

ФЕДЕРАЛЬНОЕ АГЕНСТВО ПО ОБРАЗОВАНИЮ  
ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ  
МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ПРИБОРОСТРОЕНИЯ И ИНФОРМАТИКИ

П.Г.Круг

# **МОДЕЛИРОВАНИЕ ИСКУССТВЕННЫХ НЕЙРОННЫХ СЕТЕЙ**

Учебное пособие по курсу  
«Нейронные вычислительные сети»

Москва 2009

УДК 004.032.26

ББК 32.818

К-84

*Рекомендовано к изданию в качестве учебного пособия  
редакционно–издательским советом МГУПИ*

*Рецензент: проф., д-р. техн. наук Желбаков И.Н.*

**П.Г.Круг.**

**Моделирование искусственных нейронных сетей:** учебное пособие по курсу «Нейронные вычислительные сети» – М.: МГУПИ, 2009 – 96 с.

Рассматриваются основы искусственных нейронных сетей, современные программные нейросимуляторы и примеры моделирования нейронных сетей типа «Многослойный персептрон».

Содержит практический курс, базирующийся на нейросимуляторе *TRAJAN*.

Отражает десятилетний опыт обучения студентов в области нейронных сетей в Московском государственном университете приборостроения и информатики и Московском энергетическом институте (техническом университете).

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

Табл. 15 Ил. 18. Библиограф.: 14 назв.

**УДК 004.032.26**

**ББК 32.818**

© П.Г.Круг, 2009

## ОГЛАВЛЕНИЕ

<b>Введение .....</b>	<b>5</b>
<b>1. Направления применения искусственных нейронных сетей .....</b>	<b>7</b>
1.1. Типовые решаемые задачи .....	7
1.2. Обзор областей применения .....	8
1.3. Распознавание символов .....	13
1.4. Искусственный нос .....	15
1.5. Прогнозирование .....	25
<b>2. Модель искусственного нейрона .....</b>	<b>44</b>
2.1. Биологический нейрон .....	44
2.1. Искусственный нейрон .....	45
2.1. Активационная функция .....	46
<b>3. Методы обучения искусственных нейронных сетей .....</b>	<b>48</b>
3.1. Обучение «с учителем» .....	48
3.2. Обучение «без учителя» .....	49
<b>4. Многослойный персептрон .....</b>	<b>50</b>
4.1. Определения .....	50
4.2. Алгоритм «Обратного распространения» ( <i>Backpropagation</i> ) .....	50
4.3. Требования к обучающим выборкам .....	55
<b>5. Самообучающиеся карты Кохонена .....</b>	<b>57</b>
5.1. Определения .....	57
5.2. Принцип работы искусственной нейронной сети Кохонена .....	57
5.3. Сходимость алгоритма самообучения .....	61
<b>6. Программное обеспечение для моделирования нейронных сетей .</b>	<b>64</b>
3.1. Обзор программных продуктов для моделирования .....	64
3.2. Краткое описание программного продукта <i>TRAJAN</i> .....	64
3.3. Описание основных этапов моделирования в среде <i>TRAJAN</i> .....	67
<b>7. Лабораторный практикум .....</b>	<b>71</b>
Лаб. работа № 1. Создание и обучение простейшей нейронной сети.	71
Лаб. работа № 2. Определение направления двоичного сдвига .....	75
Лаб. работа № 3. Распознавание символов .....	78
Лаб. работа № 4. Искусственный нос .....	80

Лаб. работа № 5. Прогнозирование .....	83
<b>Заключение .....</b>	<b>86</b>
<b>Вопросы для самопроверки .....</b>	<b>87</b>
<b>Литература .....</b>	<b>88</b>
<b>Приложения .....</b>	<b>89</b>
П1. Программные продукты для моделирования искусственных нейронных сетей .....	93
П2. Варианты заданий к лаб. работе №2 .....	94
П3. Варианты заданий к лаб. работе №3 .....	94
П4. Варианты заданий к лаб. работе №5 .....	95

## ВВЕДЕНИЕ

Учебное пособие посвящено моделированию на персональных компьютерах искусственных нейронных сетей (ИНС) типа «Многослойный персептрон». ИНС все чаще применяются при классификации, распознавания образов и прогнозировании, поэтому подготовка специалистов, владеющих нейросетевыми технологиями, является, безусловно, актуальной задачей.

Рассмотрим историю создания и развития искусственных нейронных сетей. Впервые о них заговорили в 1940-х годах.

Считается, что теория нейронных сетей, как научное направление, была обозначена в классической работе *Мак Каллока* и *Питтса* в 1943 г., в которой утверждалось, что, в принципе, любую арифметическую или логическую функцию можно реализовать с помощью простой нейронной сети.

Среди основополагающих работ следует выделить модель *Д. Хэбба*, который в 1949 г. предложил закон обучения, явившийся стартовой точкой для алгоритмов обучения искусственных нейронных сетей, а также теоремы *М. Минского* и исследования им ряда типовых задач, в том числе, популярной задачи «Исключающего «ИЛИ».

В 1958 г. *Ф. Розенблатт* предложил нейронную сеть, названную персептроном, и построил первый нейрокомпьютер «Марк-1». Персептрон был предназначен для классификации объектов и получал на этапе обучения от «учителя» сообщение, к какому классу принадлежит предъявляемый объект. Обученный персептрон был способен классифицировать объекты, в том числе, не использовавшиеся при обучении, делая при этом мало ошибок.

Затем, после разработок 1950-х и 1960-х наступил период затишья, длившийся с 1968 по 1985 гг.

В 1985-1986 гг. теория нейронных сетей получила «технологический импульс», который был вызван возможностью моделирования нейронных сетей на появившихся в то время персональных компьютерах. Искусственные нейронные сети превратились из объекта фундаментальных исследований в инструмент решения прикладных задач в самых разных отраслях.

Настольной книгой специалиста, моделирующего и применяющего нейронные сети, стала работа *Ф. Уоссермена* «Нейрокомпьютерная техника» (издана на русском языке в 1992 году).

В ближайшие десятилетия следует ожидать дальнейший технологический рост в области проектирования нейронных сетей и нейрокомпьютеров, и связанный с этим рост интереса к ним со стороны специалистов. Развивается как фундаментальная наука, где искусственные нейронные сети являются объектом исследований, так и прикладная наука о технологиях применения нейронных сетей при решении разнообразных задач во многих отраслях. За последние годы уже открыто немало новых возможностей нейронных сетей, а работы в данной области зачастую становятся важным вкладом в промышленность, науку и технологии, имеют большое экономическое значение.

Искусственные нейронные сети во многом копируют свой естественный прототип – человеческий мозг. Однако, все попытки понять и моделировать объективные процессы обработки информации мозгом человека пока особого успеха не имели. Несмотря на то, что разработки по нейронному моделированию ведутся нейробиологами уже более 50 лет, нет ни одной области мозга, где процесс обработки информации был бы им ясен до конца. Также ни для одного нейрона в мозге пока невозможно определить код, который он использует для передачи информации в виде последовательности импульсов. Поэтому, будущие открытия в области нейрокомпьютерных технологий по всей вероятности будут связаны с новыми открытиями в области биологического нейро-моделирования. Предполагается, что открытие биологических основ обработки информации вызовет существенную активизацию работ в построении искусственного мозга и инициацию беспрецедентного по своему размаху научного и технологического проекта. Новый проект будет способен достаточно быстро дать значительный экономический эффект и, наконец-то, появится возможность широко применять «умные» системы, способные вместо людей выполнять монотонные, скучные и опасные задания [8, 9].

Справедливости ради нужно отметить, что для достижения этих целей, также важно развитие и ряда других областей информатики, микроэлектроники и искусственного интеллекта.

Данное учебное пособие содержит 6 теоретических разделов, введение, практикум, состоящий из 5 лабораторных работ, заключение, список литературы, приложения и список контрольных вопросов для самопроверки.

Выражаю благодарность аспиранту МГУПИ Яковицкому Сергею Владимировичу за помощь в подготовке учебного пособия.

## **1. НАПРАВЛЕНИЯ ПРИМЕНЕНИЯ ИСКУССТВЕННЫХ НЕЙРОННЫХ СЕТЕЙ**

### **1.1. Типовые решаемые задачи**

Потенциальными областями применения ИНС являются те области, где человеческий интеллект малоэффективен, а традиционные вычисления трудоемки или физически неадекватны (т.е., не отражают или плохо отражают реальные физические процессы и объекты).

Действительно, **актуальность применения нейронных сетей и нейрокомпьютеров многократно возрастает тогда, когда появляется необходимость решения *плохо формализованных* задач.**

Нейронные сети продемонстрировали свою способность решать сложные задачи. Они имеют уникальные потенциальные возможности, хотя не свободны от ограничений и вопросов, на которые до сих пор не существует исчерпывающих ответов. Такая ситуация настраивает на умеренный оптимизм.



**Рис.1.1. Типовые задачи, решаемые с помощью нейронных сетей и нейрокомпьютеров**

Типовые задачи, решаемые с помощью ИНС, следующие (рис.1.1):

- автоматизация процесса классификации;
- автоматизация прогнозирования;
- автоматизация процесса предсказания;
- автоматизация процесса принятия решений;
- управление;
- кодирование и декодирование информации;
- аппроксимация зависимостей.

## **1.2. Обзор областей применения**

ПЦОС широко применяются в следующих основных областях:

- промышленность;
- бизнес и финансы;
- наука и техника;
- высокие технологии;
- оборонные отрасли;
- здравоохранение.



Примеры применения ИНС и представлены в таблице 1.1. Безусловно, данный перечень не полон, однако он позволяет получить представление о характере востребованности нейросетевых технологий. В качестве иллюстрации успешного применения ИНС рассмотрим некоторые примеры.

#### Проектирование и оптимизация сетей связи

С помощью ИНС успешно решается важная задача в области телекоммуникаций – нахождение оптимального пути трафика между узлами.

Учитываются две особенности: во-первых, решение должно быть адаптивным, т. е. учитывать текущее состояние сети связи и наличие сбойных участков, а во-вторых, оптимальное решение необходимо находить в реальном времени. Кроме управления маршрутизацией потоков, ИНС используются для получения эффективных решений в области проектирования новых телекоммуникационных сетей.

#### Управление ценами и производством

Часто недооцениваются потери от неоптимального планирования производства. В связи с тем, что спрос и условия реализации продукции зависят от времени, сезона, курсов валют и многих других факторов, то и объем производства должен гибко варьироваться с целью оптимального использования ресурсов.

Таблица 1.1. Основные области применения нейрокомпьютеров

Промышленность		
Управление технологическими процессами	Идентификация химических компонент	Контроль качества артезианских вод
Оценка экологической обстановки	Прогнозирование свойств синтезируемых полимеров	Управление водными ресурсами
Оптимальное планирование	Разработка нефти и газа	Управление прессом
Идентификация вида полимеров	Управление ценами и производством	Оптимизация работы моторов
Обнаружение повреждений	Оптимизация закупок сырья	Контроль качества
Приложения аналитической химии	Анализ функционирования заводов и магазинов	Прогнозирование потребления энергии
Высокие технологии		Оборона
Проектирование и оптимизация сетей связи	Идентификация и верификация говорящего	Анализ визуальной аэрокосмической информации
Видеонаблюдение	Речевой ввод текста в ПК	Отбор целей
Распознавание печатных и рукописных символов	Автоматизированное распознавание речевых команд	Обнаружение взрывчатых веществ и наркотиков
Фальсификации в пищевой и парфюмерной отраслях	Распознавание слитной речи с (без) настройки на диктора	Анализ биометрических баз данных изображений
Обслуживание кредитных карт	Анализ и сжатие изображений	Целесообразность

		условного освобождения
Наука и техника		Здравоохранение
Поиск неисправностей в научных приборах	Спектральный анализ и интерпретация спектров	Идентификация микробов и бактерий
Диагностика печатных плат	САПР	Диагностика заболеваний
Идентификация продуктов	Моделирование систем	Интерпретация ЭКГ
Синтез новых материалов	Анализ данных в ботанике	Анализ качества лекарств
Интерпретация показаний сенсоров	Планирование химических экспериментов	Обработка и анализ медицинских тестов
Оптимизация биологических экспериментов	Отбор сенсоров для контроля химических процессов	Прогнозирование результатов лечения
Геофизические и сейсмологические исследования	Прогноз температур технологических процессов	Оптимизация атлетической подготовки
Диагностика сигнализации	Распознавание ингредиентов	Диагностика слуха
Бизнес и финансы		
Выбор сбытовой политики	Прогноз прибыли ( <i>Cash-flow</i> )	Прогнозирование продаж
Принятие административных решений	Предсказание и расшивка «узких мест»	Анализ целей маркетинговой политики
Предсказания на фондовой бирже	Прогноз эффективности кредитования	Прогнозирование индикаторов
Анализ финансового рынка	Исследование фактора спроса	Анализ страховых исков
Прогнозирование валютного курса	Прогнозирование и анализ цен	Отбор перспективных кадров
Моделирование бизнес-стратегии	Построение макро- и микроэкономических моделей	Стратегии в области юриспруденции
Предсказание наступления финансовых кризисов	Предсказание необходимых трудностей	Оценка стоимости недвижимости

Нейросетевая система (компания *Neural Innovation Ltd.*), предназначенная для планирования затрат при издании газет, обнаруживает сложные зависимости между затратами на рекламу, объемами продаж, ценой, ценами конкурентов, днем недели, сезоном и т.д. В результате использования системы осуществляется выбор оптимальной стратегии издательства с точки зрения максимизации объема продаж или прибыли.

#### Анализ потребительского рынка

Один из популярных маркетинговых механизмов – распространение купонов, дающих право покупки определенного товара со скидкой.

Так как затраты на рассылку купонов довольно велики, решающим фактором является *эффективность рассылки*, т. е. повышение доли клиентов, воспользовавшихся скидкой. Для повышения эффективности купонной системы необходимо проведение предварительной сегментации рынка, а затем адресация клиентам каждого сегмента именно тех купонов, которыми они с большей вероятностью воспользуются.

Нейросетевая система (компания *IBM Consulting*), прогнозирующая свойства потребительского рынка пищевых продуктов, решает задачу *кластеризации* с помощью сетей Кохонена. На втором этапе для потребителей каждого из кластеров подбираются подходящие коммерческие предложения, а затем строится прогноз объема продаж для каждого сегмента. Другой популярный маркетинговый механизм – распространение поощрительных товаров (когда, например, присылая пять этикеток от кофе, клиент бесплатно получает кружку). Здесь обычные методы прогнозирования отклика потребителей могут быть недостаточно точны: иногда, спрос на кружки оказывается слишком велик и многие покупатели годами ждут получения приза.

Прогнозирующая нейросетевая система (компания *GoalAssist Corp.*) использует сеть с адаптивной архитектурой нейросимулятора *NeuroShell Classifier* (компании *Ward Systems Group*).

На входы ИНС, применяемой для классификации возможных откликов потребителей, подаются различные параметры товаров и рекламной политики для разделения входов на четыре вида откликов. Те же входы вместе с ответом первой сети подаются на вход сети нейросимулятора *NeuroShell Predictor* (той же компании), предназначенной для решения задачи количественного прогнозирования.

При этом средняя ошибка предсказаний эффекта от распространения поощрительных товаров составляет всего около 4 %.

#### Исследование спроса

Для сохранения бизнеса в условиях конкуренции компании приходится поддерживать постоянный контакт с потребителями – «обратную связь».

Крупные компании проводят опросы потребителей, позволяющие выяснить, какие факторы являются для них решающими при покупке данного товара или услуги, почему в некоторых случаях предпочтение отдается конкурентам, и какие товары потребитель хотел бы увидеть в будущем. Анализ результатов такого опроса – достаточно сложная задача, так как существует большое количество коррелированных параметров.

Нейросетевая система (компания *Neural Technologies*) позволяет выявлять сложные зависимости между факторами спроса, прогнозировать поведение потребителей при изменении маркетинговой политики, находить наиболее значимые факторы и оптимальные стратегии рекламы, а также очерчивать сегмент потребителей, наиболее перспективный для данного товара. В частности, система применяется для исследований предпочтений

различных сортов пива в зависимости от возраста, дохода, семейного положения потребителя и других параметров.

#### Анализ страховых исков

Нейросетевая система *Claim Fraud Analyser* (компания *Neural Innovation Ltd.*) предназначена для оперативного выявления подозрительных страховых исков, поступающих в связи с повреждениями автомобилей. На входы системы подаются такие параметры, как возраст и опыт водителя, стоимость автомобиля, наличие подобных происшествий в прошлом и др.

В результате обработки такой информации ИНС определяет вероятность того, что данный иск не связан с мошенничеством. Система позволяет не только обнаруживать фальсификации, но и улучшать отношения с клиентами за счет более быстрого удовлетворения справедливых исков.

#### Обслуживание кредитных карт

Нейросетевая система *Falcon* (компания *HNC*), разработанная для отслеживания операций с крадеными кредитными картами и поддельными чеками, позволяет по частоте сделок и характеру покупок выделить подозрительные сделки и сигнализировать об этом в контролирующие службы.

Благодаря данной системе, отслеживающей более 260 миллионов счетов 16 крупнейших эмитентов кредитных карт, потери банков от таких операций заметно уменьшились.

Аналогичная система (компания *ITC*), используемая для обработки операций с кредитными картами *VISA*, предотвратила в 1995 г. нелегальные сделки на сумму более 100 млн долл.

#### Медицинская диагностика

Система объективной диагностики слуха у грудных детей (Российская компания Нейропроект) обрабатывает зарегистрированные «вызванные потенциалы» (отклики мозга), проявляющиеся в виде всплесков на электроэнцефалограмме, в ответ на звуковой раздражитель, синтезируемый в процессе обследования.

Обычно, для уверенной диагностики слуха ребенка опытному эксперту-аудиологу необходимо провести около 2000 тестов, что занимает около часа. Система на основе ИНС способна с той же достоверностью определить уровень слуха уже по 200 наблюдениям в течение всего нескольких минут, причем без участия квалифицированного персонала.

#### Обнаружение фальсификаций

Подсчитано, что потери бюджета США от мошенничеств в области здравоохранения и фальсификаций составляют около 730 млн долл. в год.

Тестирование системы обнаружения (стоимость – 2,5 млн долл., компания *ИТС*) показало, что ИНС позволяет обнаруживать 38% мошеннических случаев, в то время как существовавшая ранее экспертная система – только 14%.

#### Оценка недвижимости

Стоимость квартиры или дома зависит от большого числа факторов, таких как общая площадь, удаленность от центра, экологическая обстановка, престижность, тип дома, и т.д. Так как вид этих зависимостей неизвестен, то стандартные методы анализа неэффективны в задаче оценки стоимости. Как правило, эта задача решается экспертами-оценщиками, работающими в агентстве по недвижимости.

Недостатком такого подхода является субъективность оценщика, а также возможные разногласия между различными экспертами.

Система на основе ИНС (компания *Attrasoft*) способна эффективно решать широкий спектр задач объективной оценки стоимости недвижимости, в частности, с учетом 13 факторов при оценке стоимости домов в г.Бостон (США).

Группа исследователей из университета г.Портсмут (Великобритания) в системе на основе ИНС использовала данные по оценке недвижимости из обзоров риэлтерских фирм и списков аукционных цен. Результаты исследования показали, что система делает оценки стоимости близкие к оценкам лучших экспертов и специалистов данного профиля.

#### Распознавание речи

Распознавание речи – одна из наиболее популярных областей применений нейронных сетей.

Демонстрационная система для дикторo-независимого речевого управления встроенным калькулятором *Microsoft Windows* (Российская компания Нейропроект) способна распознавать 36 команд, сказанных в стандартный микрофон. Для классификации слов используется двухкаскадная иерархическая нейронная сеть, где первый каскад состоит из одного персептрона (1000 входов, 24 нейрона в скрытом слое, 6 выходов), а второй каскад – из 6 персептронов с различными параметрами слоев.

При этом первый персептрон осуществляет грубое распознавание слова, относя его к одному из шести классов. Роль второго каскада – точно

классифицировать команду внутри класса. Для построения данной нейронной сети используется библиотека *NeuroWindows*, а также специальный алгоритм иерархического обучения. В обучении сети принимали участие 19 дикторов.

### **1.3. Распознавание символов**

Распознавание букв и символов, с одной стороны – одна из наиболее разработанных и освещенных в специальной литературе проблем, а с другой – не смотря на кажущуюся простоту, чрезвычайно трудно реализуемая на практике задача.

Рассмотрим особенности применения нейронной сети (компания *AT&T Bell Laboratories*) при сортировке писем на почте в г.Буффало, США.

Задача состоит в применении нейросетевых методов при разработке системы распознавания рукописных цифр, которые отправители писем указывали на конвертах в качестве индекса. Исследовались две строчки индексов: первые – написанные быстро и, как правило, неразборчиво, и вторые – написанные более тщательно печатными буквами. Разработчики наполнили базу данных более 9000 символами, переведенных с конвертов, которые прошли через почтовую службу г.Буффало в 1988 (рис.1.2).

80322-4129 80206

40004 14310

37872 05453

5502 75216

35460 44209

1011915485726803226414186  
6359720299299722510046701  
3084111591010615406103631  
1064111030475262009979966  
8912056708557131427955460  
6018730187112993089970984  
0109707597331972015519055  
1075318255182814358090943  
1787521655460354603546055  
18255108503047520439401

Рис.1.2. Рукописные индексы, обрабатываемые почтовой службой г. Буффало (сверху) и взятые из них отдельные изолированные цифры (снизу)

Показаны некоторые почтовые индексы (вверху) и уже изолированные цифры, подготовленные для распознавания (внизу). Видно, что индексы пишутся крайне неразборчиво, так, что сотрудники почты считают, что некоторые отправители в действительности не желают, чтобы их письма доходили по назначению. В связи с этим, к сожалению, большинство подобных систем распознавания имеют точность 95%, что является едва приемлемым показателем.

В целом ряде случаев, наиболее трудная проблема при распознавании символов – не собственно распознавание, а обнаружение символов и выяс-

нение их местоположения, т.е. – верная интерпретация индекса, состоящего из известного количества позиций, изолирование и подготовка к распознаванию отдельных цифр индекса. Поэтому для простоты будем полагать далее, что процесс распознавания начинается уже после изолирования цифры. Для каждой цифры разработчики строили решетку (сетку) размерностью  $16 \times 16$  пикселей. В ходе исследований в 1988 г. было выяснено, что применение стандартного подхода, основанного на применении обученной ИНС обратного распространения к сырому массиву пикселей и чисел, приемлемого результата не дает.

В 1988–1990 гг. был предложен метод локализации информативных участков, вокруг которых строились решетки  $5 \times 5$  и  $7 \times 7$  пикселей, после чего на вход ИНС обратного распространения поступал 180-мерный вектор. Конечная нейросетевая система распознавания представляет собой аппаратный модуль, реализованный на базе ПЦОС и соединенный с ПК. Обучение нейронной сети системы проводится единожды, однако, достаточно медленно с использованием 167693 представительских выборок. Ошибка системы в процессе распознавания символов – 0,14% при предъявлении обучающей пары из набора представительских выборок, использованной при обучении, и 5% при распознавании «новых» символов. Таким образом, разработчики и пользователи приняли решение о приемлемости результатов и необходимости использования системы для предварительной сортировки конвертов.

#### **1.4. Искусственный нос**

##### Принцип действия искусственного носа

Среди пяти чувств, чувство запаха наиболее загадочное.

Человеческий нос стал объектом исследования ученых и инженеров, специализирующихся в области высоких технологий и пытающихся понять, как нос функционирует. Такой повышенный интерес к обонятельной системе человека возник в связи с последними достижениями в области проектирования электронного (искусственного) носа. В классическом понимании электронный нос представляет собой мультисенсорное цифровое устройство, предназначенное для анализа содержимого воздушной среды путем классификации запахов.

Несмотря на то, что электронный нос сегодня не способен заменить человеческую обонятельную систему, сфера применения данной технологии достаточно широка.



В производственных целях возможности обонятельной системы человека широко используется во многих странах, например, для проверки различных продовольственных продуктов. Тренированный человеческий нос, детально изучив запахи продовольственных продуктов, таких, например, как зерно, сыр, вино, водка, рыба, способен в последствие определять их качество и свежесть. Аналогичным образом «нюхачи» оценивают перспективность того или иного парфюмерного запаха, обнаруживают фальсифицированные духи и дезодоранты. Запахи учитываются также и докторами при выявлении общих заболеваний: такие болезни, как пневмония или диабет, вызывающие специфическое дыхание или жидкие выделения с характерными запахами, могут быть замечены квалифицированными врачами.

Если искусственный интеллект электронного носа окажется способен классифицировать запахи подобным образом, то тогда электронный нос смог бы справиться с той же работой, причем гораздо лучшим образом. Проблема в том, что человеческая обонятельная система чрезвычайно субъективная: зачастую, разные люди по-разному реагируют на запахи. Электронный нос решает эту проблему, наверняка устанавливая «стандарт» для каждого требуемого запаха, например, строго определяя запах испорченного зерна.

В настоящее время различные прототипы электронного носа уже широко используются в промышленности. В частности, в агрокомплексе Швеции электронный нос применяется для независимого контроля качества зерна путем автоматической классификации проб зерна на кондиционное и испорченное (достоверность – 90 %).

Другие проблемно-ориентированные разновидности электронного носа позволяют контролировать испарения вредных для здоровья химических, в том числе аллергических, веществ.

Привлекательность использования электронного носа в этих целях заключается в следующем:

- обнаружение тех или иных компонент носит объективный, а не субъективный характер;
- собственно процесс обнаружения токсичных веществ путем вдыхания воздуха через нос может быть вреден;
- некоторые химические вещества и комбинации веществ, которые легко обнаруживаются электронным носом, в традиционном понимании могут не иметь запаха;

- электронный нос способен функционировать в местах, где не может функционировать человеческий, например, в условиях крайне высоких и низких температур, внутри тела человека, в масляных или бензиновых резервуарах, в сточных трубах, на космических спутниках и т.д.

Для понимания принципа действия электронного носа, уточним, каким образом возникает запах: иногда его создает какое-либо одно химическое вещество, но чаще – комбинация множества различных химических компонент. Например, запах кофе формируют сотни различных молекул. Электронный нос должен реагировать на определенную концентрацию требуемых молекул и их комбинаций.

Существуют два принципиально разных подхода к реализации электронного носа:

- с использованием методов газовой хроматографии и масс-спектрометрии;
- с использованием ИНС.

Однако применение методов хроматографии и масс-спектрометрии, в отличие от нейросетевых методов, не позволяет регистрировать запахи, а лишь присутствие тех или иных отдельных химических компонент. Поэтому, хроматографы и масс-спектрометры могут называться электронным носом лишь условно.

*Достоинства* использования нейросетевого подхода следующие:

- Применение ИНС позволяет отказаться от использования редких и дорогостоящих сенсоров. Используется, как правило, мультисенсорная комбинация («головка» или матрица), состоящая из набора (от 5 до 15) слабоселективных доступных по цене химических сенсоров.
- ИНС способны обнаруживать большее количество химических компонент, чем количество сенсоров нейросетевой системы.
- Продолжительность измерений существенно короче. Следует отметить, что временные затраты меньше как на собственно регистрацию, так и на этапы подготовки и обработки измерений.

Говоря о *недостатках* использования нейросетевого подхода, можно упомянуть о необходимости предварительной настройки электронного носа на требуемые запахи (подготовки базы данных «стандартов» запахов), выражающейся, в частности, в предварительном обучении ИНС.

Далее, под электронным носом будем подразумевать интеллектуальные системы, реализованные на основе ИНС.

#### Аппаратура искусственного носа

Электронный нос состоит из двух основных функциональных частей (рис.1.3): мультисенсорного измерительного модуля и интеллектуального цифрового блока.

Содержание кислорода в воздушной среде изменяется из-за присутствия в ней определенных химических веществ. Содержание кислорода, в свою очередь, изменяет выходное напряжение сенсора, которое измеряется как разность между текущим значением и нормальным (или стандартным) уровнем. Аналоговый сигнал затем преобразовывается АЦП в цифровой код, готовый для дальнейшей цифровой обработки. Измерительный модуль содержит аналоговый или цифровой препроцессор агрегирующий, а также усиливающий измерительный сигнал для уменьшения шума и «повышения» чувствительности сенсора.

Интеллектуальный цифровой блок (в частности ПК) регистрирует сигналы, поступившие от всех сенсоров, и формирует *исследуемый вектор*.

На *этапе предварительной настройки* электронного носа производится обучение ИНС путем установления «стандартов» для каждого требуемого запаха. Процесс обучения производится путем предъявления ИНС *представительских выборок*, состоящих из входного и выходного вектора нейронной сети. Роль входного вектора играет исследуемый вектор, сформированный из результатов измерений. Размерность выходного вектора соответствует количеству запахов, которые электронный нос должен распознать.

На *этапе тестирования* определяется насколько ИНС качественно обучилась, а электронный нос способен достоверно распознавать предъявляемые ему запахи.

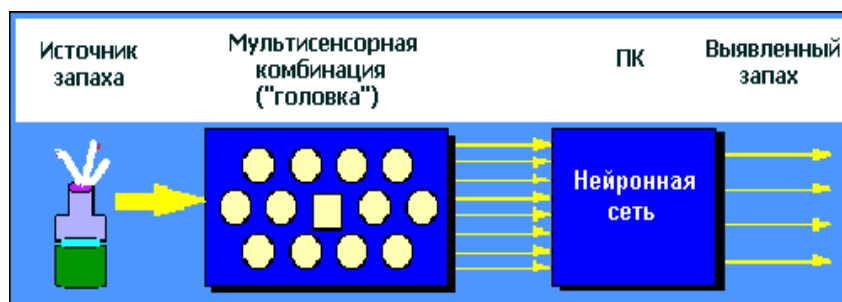


Рис.1.3. Принцип действия искусственного носа

Этапы предварительной настройки и тестирования, как правило, итерационные – если результат предварительной настройки не устраивает потребителя, то осуществляются необходимые изменения и настройка повторяется заново.

#### Пример реализации искусственного носа

Рассмотрим в качестве примера один из зарубежных прототипов электронного носа (рис. 2.3). Данная система, реализованная в одном из университетов Великобритании в учебных целях, предназначена для автоматического определения запахов ряда изделий пищевой, бытовой и офисной химии, таких как ацетон, аммиак, изопропанол, белый «штрих» и уксус.

Мультисенсорная «головка» сформирована из девяти недорогих газовых сенсоров компании Figaro Co. Ltd. (сенсор 1 – TGS 109; сенсоры 2 и 3 – TGS 822; сенсор 4 – TGS 813; сенсор 5 – TGS 821; сенсор 6 – TGS 824; сенсор 7 – TGS 825; сенсор 8 – TGS 842; сенсор 9 – TGS 880), а также датчика влажности (сенсор 10 – NH-02) и двух датчиков температуры (сенсоры 11 и 12 – 5KD-5).

Датчики влажности и температуры встроены в «головку» для контроля условий проведения измерительных экспериментов, и их показания также включены в исследуемый вектор.

Хотя каждый из газовых сенсоров изначально был задуман и создан как моноселективный (т.е. для реагирования на вполне конкретный химический компонент), каждый из них, в силу своей конструкции, реагирует на широкий ряд химических веществ.

При этом, те или иные комбинации показаний всех сенсоров «головки» являются уникальными, и, следовательно, могут указывать на присутствие самых разных химических веществ и их комбинаций.

На этапе предварительной настройки ИНС предъявляются показания сенсоров (входной слой) и указываются химические вещества, которые этим показаниям соответствуют (выходной слой).

Таким образом, на данных представительских выборках ИНС обучается обнаруживать предъявленные настройщиком химические вещества.

Интеллектуальный цифровой блок данного прототипа реализован на основе ПК (рис.1.4).

Следует отметить, что быстроедействие электронного носа ограничивается практически лишь временем отклика химических сенсоров, что соответствует единицам секунд.



Рис.1.4. Внешний вид прототипа искусственного носа

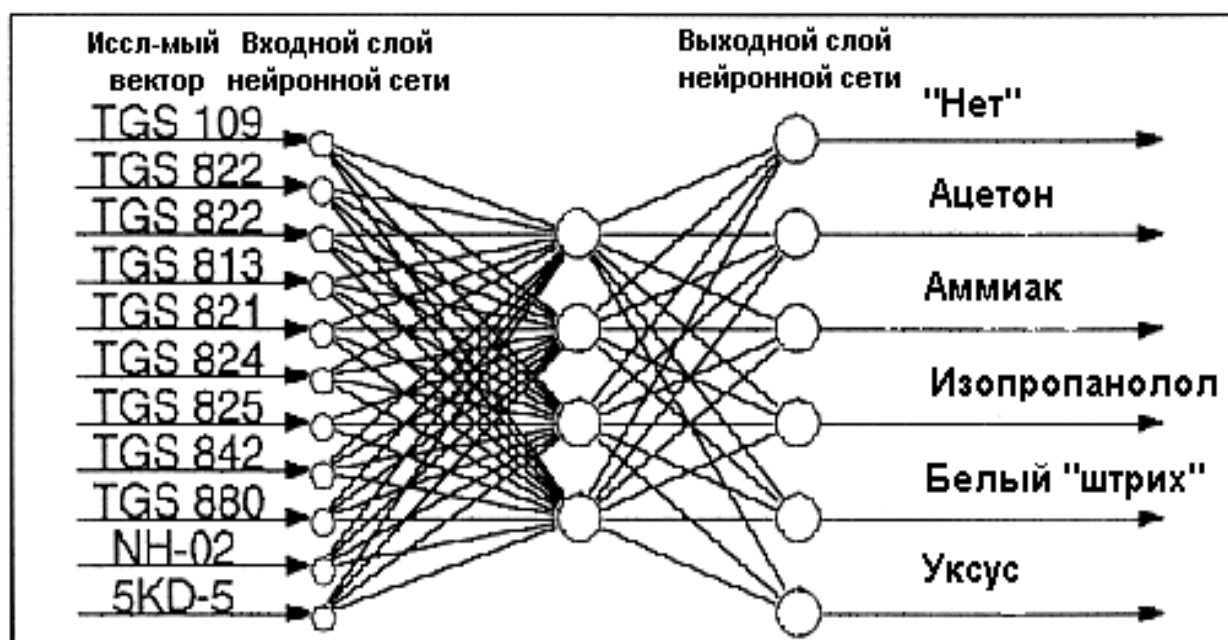


Рис. 1.5. Структура нейронной сети обратного распространения, используемой в искусственном носе

Для сравнения реализованы две разновидности ИНС – стандартный многослойный персептрон со структурой: 11 – 6 – 6, обучаемый по алгоритму обратного распространения (рис.1.5) и нечеткая ART-сеть.

Для настройки и тестирования прототипа электронного носа были использованы пробы пяти из вышеперечисленных веществ: ацетон, аммиак, изопропанол, белый «штрих» и уксус. Также добавлена категория «Нет» для обозначения ситуации, когда отсутствуют все перечисленные запахи.

Параметры, используемые для обучения и тестирования ИНС с алгоритмом обратного распространения и нечеткой ART-сети, представлены в таблице 1.2. ИНС были обучены с использованием произвольно выбранных представительских выборок. Следует отметить, что при обучении во многих случаях не ставится задача, чтобы подготавливаемая ИНС определяла уровни концентрации того или иного вещества. Наоборот, при обучении ИНС могут последовательно предъявляться одни и те же вещества, но в различной концентрации. Благодаря этому нейронная сеть становится способной одинаково успешно определять как «густые» запахи, так и еле ощутимые.

Таблица 1.2. Параметры нейронных сетей, применяемых в электронном носе

Нейронная сеть с алгоритмом обратного распространения		
1	Структура нейронной сети	11 (входной слой); 6 (внутренний слой); 11 (выходной слой)
2	Активационная функция	Сигмоидальная
3	Скорость обучения	0,10
4	Момент обучения	0,90
5	Количество итераций	1369
Нечеткая ART-сеть		
1	Чувствительность обучения	0,98
2	Чувствительность тестов	0,80
3	Количество итераций	3

Таблица 1.3. Результаты тестирования искусственного носа

Количество обучающих пар	Кол-во тестов	Предъявляемое вещество	Вероятность идентификации, %	
			Алгоритм обратного распространения	Нечеткая ART-сеть
67	28	Нет	96,4	96,4
75	22	Ацетон	100,0	100,0
64	14	Аммиак	100,0	100,0
93	28	Изопропанол	92,9	100,0
5	3	Аммиак и изопропанол	0,0	66,7
106	25	Белый «штрих»	100,0	96,0
74	27	Аммиак и белый «штрих»	100,0	92,6
66	21	Уксус	81,0	95,2
68	26	Аммиак и уксус	92,3	76,9
1	2	Изопропанол и уксус	0,0	0,0
619	196	В целом	92,9	93,4

Тестирование показало приблизительно одинаковые результаты для примененных видов ИНС. При этом достоверность обнаружения запахов (ошибка идентификации) изменялась на интервале от 89,7 до 98,2 % в зависимости от используемых проб, которые выбирались произвольным образом.

В таблице 1.3 приведены количественные результаты тестирования при обнаружении различных составов.

Показания сенсоров и результаты обнаружения запахов, осуществленных интеллектуальным цифровым блоком, иллюстрируют тот факт, что качественно обученная ИНС способна правильно классифицировать предъявляемые запахи с приемлемым уровнем достоверности (выше 90,0 %).

Показания сенсоров свидетельствуют, что по отношению к классифицируемым запахам они не являются моноселективными и реагируют с разной степенью интенсивности на все предъявляемые им запахи (рис.1.6).

#### Искусственный нос для контроля окружающей среды

В связи с объективной тенденцией распространения высоких технологий и высокотехнологических производств, существенно возрастет спрос на те или иные прототипы электронного носа.

Ожидается, что в ближайшем будущем электронный нос будет востребован для реализации крупных экологических программ, направленных на защиту окружающей воздушной и водной среды.

Гигантские объемы опасных (ядерных, химических и комбинированных) отходов накоплены за более чем 40 лет производства оружия в США. По поручению отдела энергетического оружия Минобороны США Северозападная атлантическая национальная лаборатория изучает технологии по воссозданию окружающей среды и рентабельной утилизации опасных отходов. Данная программа подразумевает, в том числе, разработку портативных, недорогих систем, таких как электронный нос, способных в реальном масштабе времени идентифицировать опасные загрязняющие вещества в воздушных и жидких средах.

Прототипы электронного носа могут быть широко использованы при:

- контроле и идентификации составов токсичных выбросов в атмосферу;
- анализе топливных смесей;
- при обнаружении выбросов масляных смесей;
- при исследовании качества, в том числе и запаха, артезианских вод;
- при контроле качества воздуха в помещениях;
- при обнаружении наркотиков и взрывчатых веществ и др.

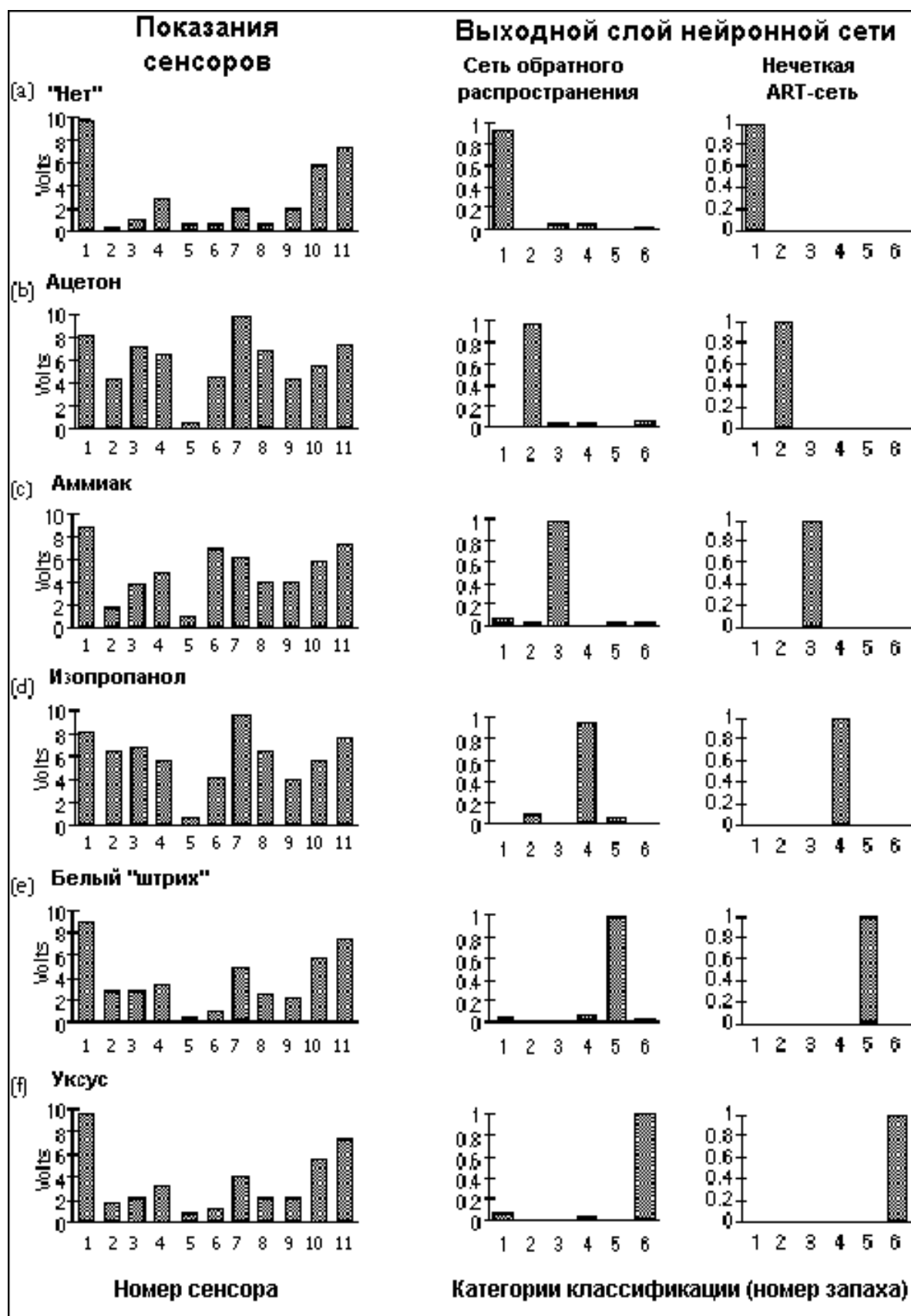


Рис.1.6. Показания сенсоров (столбец графиков слева) и результаты автоматической классификации запахов (столбцы справа)



### Искусственный нос в медицине

Применение электронного носа в качестве диагностического прибора обусловлено тем, что запах широко используется медиками при диагностике целого ряда болезней. Функциональные возможности электронного носа позволяют диагностировать те или иные отклонения организма при исследовании дыхания, запаха ран, различных физиологических и других жидкостей и т.п..

Так, запахи при дыхании могут свидетельствовать о желудочно-кишечных болезнях, диабете, болезнях печени и некоторых инфекционных заболеваниях. Инфекционные раны и ткани также выделяют отчетливые запахи, характер которых может анализироваться с применением электронного носа. В настоящее время прототип такого электронного носа используется для исследований инфекционных ран в университетской больнице южного района г. Манчестер (Великобритания).

Анализ запахов жидкостей тела, может быть использован также для диагностики заболеваний печени и других органов.

Сотрудники медучреждений при возникновении подозрений с успехом используют электронный нос для обнаружения просроченных, недоброкачественных или фальсифицированных лекарств и препаратов.

Ожидается, что в будущем электронный нос будет востребован в дистанционной хирургии. В настоящее время в этих целях широко распространена передача визуальных, звуковых и механических сигналов, в то время как значение запаха игнорируется. Потенциально, электронный нос способен идентифицировать операционные запахи и предоставлять для передачи обонятельные сигналы, создавая для дистанционной хирургии полноту среды так называемой виртуальной реальности.

### Искусственный нос в пищевой промышленности

В настоящее время наибольший рынок для электронного носа представляется в агропромышленном комплексе.

Прототипы электронного носа применяются как для оценки качества продуктов питания, так и контроля качества приготовления пищи. Предприятия агрокомплекса расширяют применение данных технологий, в частности, для контроля свежести рыбы на промыслах и оптовых складах; контроля процессов брожения; обследования контейнеров, резервуаров и элеваторов; проверки натуральности апельсинового сока; проверки прогорклости майо-

неза; ограничения лукового запаха; установления сортности (выдержанности) виски и коньяков; автоматического управления вкусом и т.д.

Важнейшее направление применения электронного носа – обнаружения на оптовых складах и в торговле фальсифицированных продуктов питания, напитков (в частности, водки) и пищевых добавок.

В ряде случаев результаты работы электронного носа могут быть использованы в качестве дополнительной аналитической информации для экспертов – специалистов в области запахов.

В других случаях, использование электронного носа при химическом анализе в агрокомплексе предпочтительнее приобретения аналитической аппаратуры, характеризующейся большей стоимостью и продолжительностью экспериментов. Это особенно заметно, когда требуется получение не количественных, а качественных результатов типа «свежее – не свежее», «натуральное – не натуральное», «качественное – не качественное», «настоящее – фальсифицированное» и т.д.

## **1.5. Прогнозирование**

Прогнозирование – важнейший элемент современных информационных технологий принятия решений в управлении.

Эффективность того или иного управленческого решения оценивается по событиям, возникающим уже после его принятия. Поэтому прогноз неуправляемых аспектов таких событий перед принятием решения позволяет сделать наилучший выбор, который без прогнозирования мог бы быть не таким удачным.

Прогнозирование – одна из самых востребованных, но при этом одна из самых сложных задач интеллектуального анализа данных. Проблемы прогнозирования связаны с недостаточным качеством и количеством исходных данных, изменениями среды (в которой протекает процесс) воздействием субъективных факторов. Прогноз всегда осуществляется с некоторой погрешностью, зависящей от используемой модели прогноза и полноты исходных данных. При увеличении информационных ресурсов, используемых в модели, увеличивается точность прогноза, а убытки, связанные с неопределенностью при принятии решений, уменьшаются.

Характер затрат, связанных с прогнозированием, таков, что за определенным пределом дополнительные затраты не приведут к снижению потерь. Это связано с тем, что объективно невозможно снизить погрешность прогно-

зирования ниже определенного уровня, вне зависимости от того насколько хорош примененный метод прогнозирования. Поэтому определение погрешности прогноза, наряду с самим прогнозом, позволяет значительно снизить риск при принятии решений.

Известны и широко применяются различные методы прогнозирования: алгоритмы экстраполяции экспериментальных данных в несложных инженерных расчетах и программных продуктах, а также более громоздкие статистические методы, использующие параметрические модели.

В последние десятилетия для прогнозирования широко применяются другие подходы, и в частности, ИНС.

Рассмотрим особенности применения ИНС, которые показывают их *преимущества* по сравнению с другими существующими методами при выборе модели прогноза.

- *Результативность при решении неформализованных или плохо формализованных задач.* Из общеизвестных преимуществ методов на основе нейронных сетей следует выделить одно самое привлекательное – отсутствие необходимости в строгой математической спецификации модели, что особенно ценно при прогнозировании плохо формализуемых процессов. Известно, что большинство финансовых, бизнес и других подобных задач плохо формализуется.
- *Устойчивость к частым изменениям среды.* Достоинства нейронных сетей становятся заметными, когда часто изменяются «правила игры»: среда, в которой существует прогнозируемый процесс, а также характер воздействия влияющих факторов. Поэтому, ИНС наилучшим образом подходят для решений таких задач, как прогнозирование тенденций фондового рынка, характеризующихся влиянием целого набора постоянно изменяющихся факторов.
- *Результативность при работе с большим объемом противоречивой информации.* Нейронные сети будут предпочтительнее там, где имеется много анализируемых данных, в которых скрыты закономерности. В этом случае автоматически учитываются также различные нелинейные взаимодействия между влияющими факторами. Это особенно важно, в частности, для предварительного анализа или отбора исходных данных, выявления «выпадающих фактов» или грубых ошибок при принятии решений.

- *Результативность при работе с неполной информацией.* Целесообразно использование ИНС в задачах с неполной или «зашумленной» информацией, а также в задачах, для которых характерны интуитивные решения.

#### Постановка задачи прогнозирования

Задача прогнозирования в общем случае сводится к получению оценки будущих значений упорядоченных во времени данных на основе анализа уже имеющихся, а также (при необходимости) тенденции изменения влияющих факторов.

Прогнозируемой величиной являются значения временного ряда на интервале  $[T(n+1), T(n+f)]$ , где  $T(n)$  – текущий момент времени, а  $f$  – интервал прогнозирования. Иногда возникает необходимость не в прогнозе значений временного ряда на заданном интервале, а в прогнозе вероятности того, что они будут вести себя тем или иным образом (возрастать, убывать, находиться в некоторых пределах и т.д.).

Рассмотрим типовой алгоритм прогнозирования, осуществляемого с использованием нейронных сетей (рис.1.7).

#### *Отбор значащих факторов*

На первом этапе выделяется максимальное число из значащих, влияющих на прогноз, факторов. Такие дополнительные факторы, влияющие на поведение прогнозируемой величины, называют экзогенными (внешними) или артефактами.

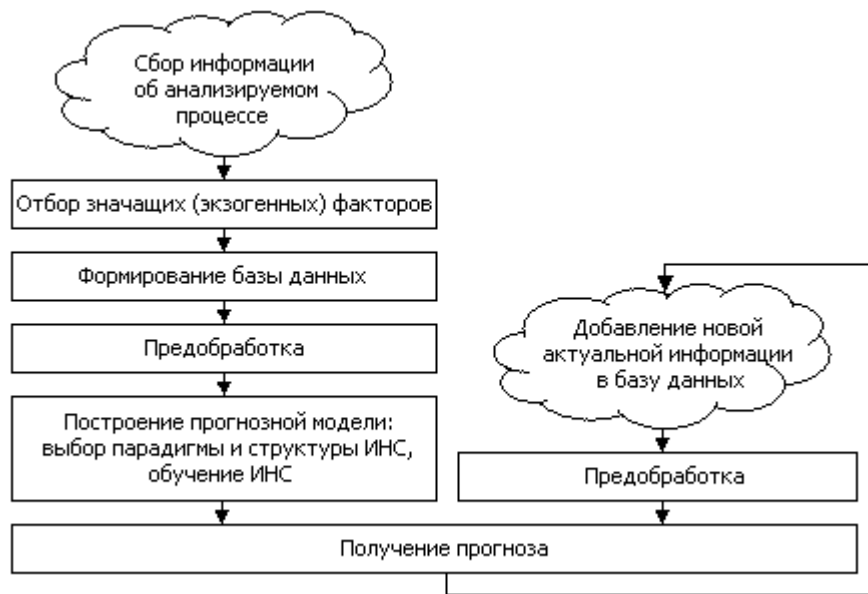
Здесь же выбирается интервал наблюдения (окно скольжения), т.е. выясняется, по какому количеству предшествующих значений временного ряда осуществляется прогноз.

#### *Предобработка данных*

На втором этапе устраняются несущественные, по мнению эксперта, и не влияющие на прогноз, данные. При необходимости, также восстанавливается пропущенная информация, устраняются аномальные выбросы, убираются высокочастотные шумы. Умело проведенная предобработка данных позволяет значительно улучшить качество прогноза.

#### *Построение модели*

На следующем этапе для данного анализируемого процесса выбирается наиболее подходящая парадигма и структура ИНС, а также алгоритм и параметры ее обучения.



**Рис.1.7. Типовой алгоритм прогнозирования, осуществляемого с использованием нейронных сетей**

*Собственно прогнозирование (получение результата)*

Эксперименты осуществляются по схеме, аналогичной той, при которой производилось обучение.

Рассмотрим так называемый метод скользящих окон (Пример 1.1). Он предполагает использование двух окон  $W_i$  и  $W_o$  с фиксированными размерами  $n$  и  $m$  соответственно. Эти окна перемещаются с некоторым шагом скользящего  $s$  по временной последовательности имеющихся данных, начиная с первого элемента.

При этом первое окно  $W_i$  длиной  $n$  формирует входной вектор нейронной сети, а второе –  $W_o$  – выходной вектор размерностью  $m$ .

Последовательность обучающих выборок (обучающих пар)  $W_i \rightarrow W_o$  формирует так называемый блок обучающих или представительских выборок.

**Пример 1.1. Простейшая модель прогнозирования продаж с помощью нейронной сети**

*Дано:*

Информация об еженедельных продажах компьютеров (таблица 1.4) за четыре месяца (число недель  $k = 16$ ).

Таблица 1.4. Данные о еженедельных продажах компьютеров

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
100	94	90	96	91	94	95	99	95	98	100	97	99	98	96	98

*Получить:*

Многошаговый и одношаговый прогноз продаж.

*Пример решения:*

В данной временной последовательности предполагается наличие скрытых нелинейных зависимостей. Поэтому для построения модели прогноза применим ИНС:

1. Синтезируем ИНС со следующими параметрами:

- число нейронов входного слоя (ширина окна)  $n = 4$ ;
- число нейронов выходного слоя  $m = 1$ ;
- сдвиг (скольжения)  $s = 1$ .

2. С помощью метода скользящих окон для ИНС формируется блок обучающих (представительских) выборок (таблица 1.5).

Очередная обучающая выборка получается в результате сдвига окон  $W_i$  и  $W_o$  вправо на один элемент ( $s = 1$ ).

ИНС обучается на данных выборках, настраивая свои коэффициенты, и формирует в качестве результата требуемую функцию прогноза  $P$ .

Собственно процесс прогнозирования осуществляется после обучения ИНС, проводимого по тому же принципу, что и процесс формирования обучающей выборки.

Таблица 1.5. Блок обучающих выборок ИНС, применяемой для целей прогнозирования продаж

№№ обучающих выборок	Входной слой				Выходной слой
	1	2	3	4	
1	100	94	90	96	91
2	94	90	96	91	94
3	90	96	91	94	95
4	96	91	94	95	99
5	91	94	95	99	95
и т.д.					

ИНС должна обучиться на временной последовательности, взятой из таблицы 1.4 с использованием блока обучающих выборок таблицы 1.5.

Затем, при подаче на вход ИНС последней из известных выборок 99, 98, 96, 98 (см. конец таблицы 1.4), прогнозируется  $(k+1)$ -й элемент последовательности, например 95.

На данном простейшем примере рассмотрим действия при *многошаговом* и *одношаговом* прогнозировании.

#### *Одношаговое прогнозирование*

Применяется для краткосрочных прогнозов на один шаг вперед. На очередном текущем шаге в качестве исходной информации используются только объективные данные (результаты прогнозов, полученных на предыдущих шагах, не используются).

Если на шаге  $(k+1)$ -м для временной последовательности, взятой из табл. 2.5, прогнозируется 95, а на самом деле продается не 95, а 96 компьютеров, то на шаге  $(k+2)$ -м в качестве входного вектора будет использована выборка 98, 96, 98, 96.

#### *Многошаговое прогнозирование*

Применяется для осуществления долгосрочного прогноза и предназначено для определения основного тренда и главных точек изменения тренда для некоторого интервала времени. При этом прогнозирующая система использует результаты прогноза (выходные данные), полученные для моментов времени  $k+1$ ,  $k+2$  и т.д. в качестве входных данных для прогнозирования на моменты времени  $k+2$ ,  $k+3$ ,  $k+4$  и т.д.

Многошаговое прогнозирование на  $(k+2)$ -м шаге продолжается при подаче на вход нейронной сети выборки 98, 96, 98, 95, в которой последний элемент является результатом прогноза на предыдущем шаге. И так далее.

#### Прогнозирование в сфере бизнеса и финансов

Подавляющее большинство задач прогнозирования на основе нейронных сетей так или иначе связаны со сферой бизнеса и финансов. Это – краткосрочные и долгосрочные прогнозы тенденций следующих финансовых рынков:

- рынков купонных и бескупонных облигаций;
- фондовых рынков (рынков акций);
- валютных рынков.

Сюда же можно отнести прогнозы:

- платежеспособного спроса;
- продаж и выручки;
- рисков кредитования;
- финансирования экономических и инновационных проектов;
- фьючерсных контрактов и ряд других.

По оценкам экспертов, бум вокруг систем искусственного интеллекта в финансовой индустрии пришелся на период 1984–1989 гг. В основном он затронул США и Великобританию, где создатели сложных систем для военных (таких как программа «Звездные войны») решили попытать счастья на Уолл-стрит.

Фондовая биржа в Нью-Йорке в 1987 г. начала использовать программный продукт прогнозирования *Stockwatch Alert Terminal (SWAT)* II и вела переговоры о его внедрении с рядом бирж Европы и региона Юго-Восточной Азии. В этот период времени на рынке появляются программные продукты моделирования и прогнозирования банкротства, анализа портфеля ценных бумаг, оптимальной торговли акциями, а также предназначенные для определения вероятности риска при выдаче кредита. Подразделение *Mellon Bank* в Питтсбурге (США) применило программную систему *Neural-Works Professional II/Plus 5* компании *Neural-Ware* (Питтсбург, США) для распределения фондов и специальной селекции акций, так как в ходе работы было обнаружено, что между влияющими факторами и прогнозируемыми параметрами существуют нелинейные связи, не поддающиеся точному учету с помощью стандартных статистических методов.

Департамент торговли и промышленности правительства Великобритании финансирует две программы, направленные на развитие нейронных вычислений в финансовой сфере. Первая – «Нейропрогнозирование», иницированная Лондонской школой бизнеса совместно с университетским колледжем Лондона (*UCL*). Вторая – «Нейронные сети для финансовых услуг» создана *TBS Bank Technology* совместно с *UCL* и Центром прогнозирования *Henley*. Среди финансовых институтов, использующих технологию нейронных сетей – *Chemical Bank*, *Citibank*, *JP Morgan* и др.

Специалисты программы «Нейропрогнозирование» разработали модель для выработки тактики распределения фондов на глобальных рынках облигаций. Модель охватывает семь географических регионов: Великобританию, Францию, Германию, Японию, США, Канаду, Австралию, каждый из которых моделируется с помощью ИНС различной структуры. Для получе-



ния краткосрочных прогнозов обучение ИНС производилось с использованием статистической информации, характеризующей ситуацию на этом рынке за каждый месяц. Далее, полученные локальные прогнозы объединяются в центре управления единым портфелем ценных бумаг. С ноября 1992 г. данная программная система использовалась Североамериканской страховой компанией (г. Бостон, США). В результате использования капитал компании увеличился с 25 до 50 млн долл., а портфель ценных бумаг повысил доходность на 25% в первый год внедрения системы.

В качестве инструмента для оптимизации параметров ИНС часто используются генетические алгоритмы. В частности, компанией *Hill Samuel Investment Management* разработана программная система для прогнозирования результатов контрактов по долгосрочным ценным бумагам повышенной надежности. При моделировании нескольких стратегий торгов в задаче прогнозирования направлений движения рынка она достигла точности 57%.

В страховой компании *TSB General Insurance* (г. Ньюпорт, США) используется сходная методика для прогноза уровня риска при страховании частных кредитов. Эта ИНС самообучается на статистических данных о состоянии безработицы в стране.

#### Прогнозирование краткосрочных и долгосрочных тенденций финансовых рынков

Задачей автоматизированной системы прогнозирования краткосрочных и долгосрочных тенденций финансовых рынков является анализ некоторого набора влияющих факторов с последующим выводом о дальнейшем краткосрочном или долгосрочном поведении прогнозируемой величины.

Возможными прогнозируемыми величинами для подобных систем являются доходность и ценовые показатели: средневзвешенная цена, цены закрытия и открытия, максимальная и минимальная цены. Причем прогнозироваться могут как показатели, определенные для целой группы инструментов или некоторого рынка в целом, так и показатели, определенные, только для одного инструмента финансового рынка. Как для совокупности инструментов, так и для каждого индивидуально может определяться доходность; ценовые показатели определяются для каждого конкретного инструмента. Целями прогноза (прогнозируемой величиной) в области финансовых рынков могут являться, например, средневзвешенная доходность бескупонных облигаций (для группы инструментов), средневзвешенная цена акции РАО «РЖД», курс американского доллара к рублю и др.

В качестве исходных данных (влияющих факторов) для такого прогноза могут использоваться различные макро- и микроэкономические показатели, информация с торговых площадок, данные, предоставляемые информационно-торговыми агентствами, экспертные оценки специалистов. Количество влияющих на прогноз факторов зависит от рассматриваемого рыночного инструмента и конкретной рыночной ситуации (временного момента), т.е. одни факторы оказывают влияние на все финансовые рынки, другие – только на определенные. Кроме того, влияние факторов на рынки может меняться с течением времени (меняются рыночные тенденции). Так как в определенные моменты времени прослеживается явная взаимосвязь между финансовыми рынками и инструментами рынков, целесообразно в качестве исходных данных для прогнозирования одного рынка или его инструментов использовать информацию о тенденциях других рынков. Например, при прогнозировании цены открытия «сегодня» для любых инструментов всех финансовых рынков, этот прогноз сильно зависит от цены закрытия «вчера» и обе эти величины могут выступать как прогнозируемые.

Прогнозирование краткосрочных и долгосрочных тенденций финансовых рынков включает следующие этапы.

1. Сбор и хранение статистических данных – возможной исходной информации для прогноза (либо в качестве исходных данных, либо в качестве прогнозируемой величины, либо, как и то и другое).
2. Определение для рассматриваемого рынка или инструмента прогнозируемой величины и набора влияющих факторов (причем не всегда могут быть использованы данные, непосредственно хранящиеся в базе данных, зачастую требуется произвести некоторые преобразования данных: например, часто в качестве таких факторов используются относительные изменения величин).
3. Выявление зависимости между прогнозируемой величиной и набором влияющих факторов в виде некоторой функции.
4. Вычисление интересующей величины в соответствии с определенной функцией, значениями влияющих факторов на прогнозируемый момент и видом прогноза (краткосрочный или долгосрочный).

Процедура выполнения краткосрочного прогноза отличается от процедуры долгосрочного на первом и четвертом этапах. В случае краткосрочного прогноза считается, что все участвующие в нем значимые влияющие факторы на прогнозируемую дату известны и хранятся в базе данных. Горизонт

краткосрочного прогноза не превышает три-четыре дня. В случае долгосрочного прогноза считается, что значимые влияющие факторы на прогнозируемую дату неизвестны и должны быть указаны ожидаемые значения и погрешности. Соответственно погрешность определения прогнозируемой величины существенно увеличивается (чем дальше горизонт прогноза, тем больше погрешность определения влияющих факторов и вероятность ошибки аналитика). Горизонт долгосрочного прогноза, как правило, превышает три-четыре дня.

### Кредитование

Характерный пример успешного применения нейронных сетей в финансовой сфере – управление кредитными рисками.

Перед выдачей кредита для оценки вероятности собственных убытков от несвоевременного возврата финансовых средств крупные банки, как правило, предпринимают сложные статистические расчеты по определению финансовой надежности заемщика.

Такие расчеты обычно базируются на оценке кредитной истории, динамики развития компании, стабильности ее основных финансовых показателей и многих других факторов. Так, *Bank of New York*, США опробовав метод нейронных вычислений и применив его для оценки 100 тыс. банковских счетов, выявил свыше 90% потенциальных неплательщиков.

### Прогнозирование тенденций фондового рынка (рынка акций)

Важная область применения ИНС в сфере финансов – прогнозирование ситуации на фондовом рынке. Стандартный подход к решению этой задачи (не использующий нейронные сети) базируется на жестко фиксированном наборе «правил игры», который со временем теряет свою актуальность из-за изменения условий торгов на фондовой бирже. Помимо того, системы, построенные на основе такого стандартного подхода, оказываются слишком медленными для ситуаций, требующих от трейдера (участника торгов) мгновенного принятия решений.

Рассмотрим, некоторые особенности фондового рынка.

Трейдер, принимающий решения о купле-продаже акций, имеет доступ к одному или нескольким электронным источникам информации (*Reuters, Dow Jones Telerate, Bloomberg, Tenfore*). Он наблюдает текущие значения и графики интересующих его индексов на мировых фондовых биржах, основные кросс-курсы валют и другие показатели валютного, фон-

дового и кредитного рынков в многооконной среде с различной степенью детализации. На принятие его решения о купле–продаже акций, естественно, влияют макроэкономические и общественно-политические события, сообщения о которых через каждые 5–10 минут появляются в текстовом окне монитора и сопровождаются комментариями экспертов, озвучивающих разнообразные слухи и прогнозы. Трейдеру также доступна дополнительная информация, такая как сообщения из Центрального банка России и от других значащих источников об основных показателях рынков.

Обязательно учитывается психология конкурирующих трейдеров, для которых важную роль играют ожидания того или иного поведения ряда влияющих процессов. Например, в 16:00 многие московские трейдеры внутренне не готовы к изменениям тенденции поведения индекса Доу-Джонса на Нью-Йоркской фондовой бирже, которая с учетом сдвига по часовым поясам открывается лишь в 17:30 по московскому времени.

Фондовый рынок характеризуется еще и другими *особенностями*:

- рыночные процессы весьма неоднородны во времени: например, состояние рынка осенью существенно отличается от его состояния летом того же года; поэтому не всегда имеет смысл формировать обучающие выборки большого объема;
- «загрязнениями» данных и их неоднородностью;
- наличием малоинформативных показателей при относительно малом объеме статистики.

В целом, задача краткосрочного прогноза котировок акций пусть и с использованием нейронных сетей представляется достаточно сложной, особенно на стремительно изменяющемся российском фондовом рынке.

Примером прогнозирования тенденций фондового рынка может служить нейросетевая система (компания *Alela Corp.*), предназначенная для прогноза изменения биржевых индексов *Dow Jones*, *S&P500* и *Merval*.

На сайте компании можно бесплатно воспользоваться прогнозом изменения данных индексов и, используя его в качестве дополнительной информации, убедиться, что доля верных прогнозов составляет не менее 80%.

Японские компании, оперирующие на рынке ценных бумаг, также широко применяют нейронные сети (компания *Mitsubishi*).

Для входа нейронной сети использовалась информация о деловой активности нескольких организаций, полученная за 33 года, включая также оборот, предыдущую стоимость акций, уровни дохода и т.д. Данная ИНС

самообучалась на реальных примерах и показала высокую точность прогнозирования, а также быстрое действие. Общая результативность прогноза по сравнению с системами, использующими стандартные статистические подходы, улучшилась на 19%.

#### Оптимальное распределение свободных средств банка между различными финансовыми рынками

Успешное прогнозирование поведения как финансовых рынков в целом, так и их отдельных инструментов позволяет банку эффективнее управлять имеющимися в его распоряжении средствами.

Задача оптимального распределения свободных средств между различными финансовыми рынками и их инструментами встает перед банком ежедневно. Любой банк имеет в своем распоряжении «портфель», куда могут входить различные ценные бумаги и валюта. Принцип формирования портфеля – получение прибыли с вложенного в финансовые инструменты капитала не ниже некоторого фиксированного уровня при минимальном для этого уровня риске.

Ежедневно могут происходить следующие взаимоисключающие процессы: поступление денежных средств для их вложения в финансовые инструменты и отток денежных средств для выполнения обязательств банка, т. е. существуют следующие причины для изменения состава портфеля:

- с течением времени отдельные финансовые инструменты начинают терять свою привлекательность и необходимо выполнить оптимальное (доходность не ниже фиксированного уровня, риск – минимальный) перераспределение средств между финансовыми инструментами внутри портфеля;
- банку необходимо выполнить некие требования, для чего реализуется некоторое количество финансовых инструментов, входящих в портфель, на определенную сумму; естественно, что финансовые инструменты должны быть выбраны таким образом, чтобы характеристики портфеля по возможности не ухудшились;
- у банка увеличился объем свободных денежных средств и необходимо произвести их оптимальное распределение между различными финансовыми инструментами.

Независимо от причины и механизма изменения состава портфеля расчет выгодности этих изменений производится на фиксированную дату, называемую горизонтом портфеля.

Исходными данными для задачи оптимального распределения свободных средств между различными финансовыми рынками и их инструментами являются либо результаты долгосрочного прогноза для всех рассматриваемых инструментов, либо вероятностный анализ поведения рассматриваемых инструментов в сходных рыночных ситуациях.

Выбор финансового инструмента с максимальной предполагаемой доходностью не составляет большого труда, но задача усложняется необходимостью учитывать риск предполагаемых вложений, т.е. возможность инструмента не реализовать эту доходность. Как правило, чем выше доходность, тем выше риск, и снижение риска ведет к снижению доходности. Поэтому при планировании распределения средств банка рассматриваются две задачи:

- вложение средств с минимальным риском;
- вложение средств с доходностью не ниже фиксированного уровня и минимальным для этого уровня риском.

Классическим примером снижения риска портфеля в целом является сочетание в нем инструментов с отрицательным коэффициентом корреляции.

#### *Поступление данных в систему*

В программном продукте, применяемом в Промстройбанке, реализован автоматизированный ввод в нейронную сеть новой информации из следующих источников:

- информационно-торговые данные агентства *Reuters*;
- торговые данные с площадок ММВБ и РТС;
- прочие данные с использованием ручного ввода.

#### *Выбор и подготовка данных для участия в прогнозе*

Задача данного этапа прогнозирования – выбор из более, чем 200 видов информационно-торговых данных наиболее значимых влияющих факторов для прогноза интересующей стоимостной величины некоторого финансового инструмента или группы финансовых инструментов. Первичный выбор влияющих факторов зависит от опыта и интуиции специалиста, в виду того, что автоматизация этого процесса, как правило, неэффективна.

В помощь специалисту предоставляются инструменты технического анализа в виде графиков, анализируя которые можно уловить скрытые взаимосвязи. Специалист также может использовать доступные ему матрицы корреляции и ковариации для указанной выборки влияющих факторов и

прогнозируемой величины, однако, с помощью матриц корреляции и ковариации не удастся уловить нелинейную, редко возникающую зависимость, которая, тем не менее, может оказать существенное влияние на прогнозируемую величину.

После осуществления прогноза аналитик может определить значимость участвовавших в нем влияющих факторов по изменению функции оценки и выходных сигналов системы с целью окончательной коррекции участвующих в прогнозе влияющих факторов.

Достаточно часто возникает ситуация, когда в качестве влияющего фактора или прогнозируемой величины полезно использовать информационно-торговые данные в преобразованном с помощью некоторой функции виде. Например, в качестве значимого влияющего фактора при прогнозе цены часто используется та же самая цена, но с однодневным сдвигом.

Поэтому для преобразования влияющих факторов и прогнозируемых величин были определены операции, которые применяются в любой последовательности. Кроме того, с помощью соответствующих последовательностей данных операций реализуются все наиболее популярные инструменты технического анализа.

Процесс определения величин, участвующих в прогнозе, как в качестве значимых влияющих факторов, так и в качестве прогнозируемой величины, является наиболее субъективным и трудоемким. И, естественно, нет необходимости повторять его каждый день для всех интересующих аналитика финансовых инструментов.

Существует возможность сохранения перечня выбранных влияющих факторов, участвующих в прогнозе, и выполненных с ними преобразований для некоторого финансового инструмента или группы инструментов.

#### Применение нейронных сетей для прогнозирования курсов валют

##### **Пример 1.2. Прогнозирования валютных курсов швейцарского франка к доллару и швейцарского франка к немецкой марке**

Такое моделирование с использованием нейронных сетей и технической базы *Sun SPARCstation LX* провели специалисты компании *Logica* по заказу банка *Chemical Bank*.

Выбор именно этих валют в то время объяснялся высоким уровнем подвижности первого соотношения и малым – второго (до кризиса в 1993 г.). Данные о динамике кросс-курсов этих валют собирались с 1 октября 1992 г. по 1 октября 1993 года, при этом ценовые прогнозы характеризовались пя-

тью категориями: большой рост, малый рост, без изменений, малый спад, большой спад. В итоге применяемая ИНС позволила синтезировать прогноз за вышеупомянутый период 55% данных, совпавших с реальными, по первому соотношению валют и 23% – по второму.

### Пример 1.3. Прогнозирование курса украинского карбованца к доллару

Следующий пример иллюстрирует результаты прогнозирования курса американского доллара по отношению к украинскому карбованцу (*UKB/USD*).

Исследования проводились с использованием ИНС типа «Многослойный персептрон», обучаемой по алгоритму обратного распространения.

Целью экспериментов являлось прогнозирование курса *UKB/USD*. Для чего было исследовано влияние представления исторических и прогнозируемых данных на ошибку прогнозирования. Также были рассмотрены вопросы влияния структуры ИНС на скорость обучения и ошибку прогнозирования.

При этом ставились следующие задачи поиска:

- значимых влияющих факторов;
- оптимальной формы представления статистических данных о валютном курсе;
- оптимального представления результата прогнозирования;
- оптимального размера окна «скольжения»;
- поиск оптимальной структуры ИНС.

Прогнозирование курса *UKB/USD* проводилось на основе временной последовательности ежедневных данных о курсе.

Такой подход основан на предположении экономистов, что для прогнозирования некоторых экономических показателей вполне достаточно исследования истории их изменения.

Исходными данными для экспериментов послужили ежедневные изменения курса *UKB/USD* с 15.06.93 по 26.06.95 всего 842 измерений (данные взяты из архивов банка *Porto-Franco*). Прогнозировалось среднее значение курса за день (среднее арифметическое дневных курсов покупки и продажи).

Каждый из экспериментов состоял из несколько этапов:

#### 1. Формирование обучающей выборки.

На этом этапе определялся вид представления исторических и прогнозируемых данных, осуществлялось формирование блока представительских



(обучающих) выборок. Большинство проведенных экспериментов было направлено на прогноз не фактического курса валют, а его относительного изменения:

$$\sigma_{Kt} = (K_{t+1} - K_t) / K_t. \quad (1.1)$$

*2. Обучение ИНС с использованием сформированного на первом этапе блока обучающих выборок.*

Качество обучения характеризовалось ошибкой обучения, определяемой как суммарное квадратичное отклонение значений на выходах ИНС в обучающей выборке от реальных значений, полученных на выходах ИНС. Критерий прекращения обучения – 1500 итераций или уменьшение ошибки на выходах сети на два порядка, по сравнению с первичной ошибкой. В том случае, если при описании опыта не указано, что произошло снижение ошибки на два порядка, обучение останавливается по первому критерию.

*3. Третий этап – тестирование ИНС.*

Определяется качество прогнозирования при подаче на вход 4–5% наборов из обучающей выборки. Эксперимент является успешным, если относительная достоверность не менее 80%.

*4. На четвертом этапе осуществляется пробное прогнозирование.*

На входе нейронной сети – наборы, которые не были внесены в обучающую выборку, но результат по которым (прогноз) известен.

Ограничения и недостатки, связанные с использованием ИНС для прогнозирования

1. Для эффективного прогнозирования, как правило, необходим некоторый минимум наблюдений (более пятидесяти и даже ста). Однако существует много задач, когда такое количество статистических данных недоступно. Например, при производстве сезонного товара, статистики предыдущих сезонов недостаточно для прогноза на текущий сезон из-за изменения стиля продукта, политики продаж и т.д. Даже при прогнозировании потребностей в достаточно стабильном товаре на основе информации о ежемесячных продажах невозможно накопить статистику за период от 50 до 100 месяцев. Для сезонных процессов эта проблема еще более выражена: каждый сезон фактически представляет собой одно наблюдение. Следует отметить, что удовлетворительная модель прогноза с использованием нейронной сети все же может быть построена даже в условиях нехватки данных. При этом модель будет уточняться при поступлении в нее свежих данных.

2. Другим недостатком моделей на основе ИНС являются значительные временные затраты для достижения удовлетворительного результата.

Эта проблема не столь существенна, если исследуется небольшое число временных последовательностей, однако обычно прогнозирующая система в области управления производством включает от нескольких сотен до нескольких тысяч временных последовательностей. Отметим, что завышенные ожидания эффекта от внедрения ИНС в ряд финансовых структур в США и Великобритании не оправдались. Так, один крупный инвестиционный банк на Уолл-стрит потратил более 1,0 млн долларов на разработку такой системы для оптимизации финансовых операций, однако, спустя некоторое время вынужден был вернуться к старой системе. Основной причиной неудачи стал недостаточный по сравнению с ожидаемым уровень производительности, полученный в результате внедрения системы.

3. Обучить и эксплуатировать нейронную сеть для решения многих задач, как правило, может и не специалист, но надежно интерпретировать результаты, а также численно оценивать значимость получаемых прогнозов способны специалисты, имеющие навыки в моделировании ИНС.

#### Программные продукты прогнозирования на основе нейронных сетей

Импульсом для более широкого использования ИНС в финансовых прогнозах стало появление в 1990 системы моделирования ИНС *Brain Maker* компании *California Scientific Software*. Данный программный продукт – наиболее продаваемый в своем классе – имеет следующие достоинства:

- используемая модель нейронной сети является надежной и удобной при прогнозировании в сфере бизнеса и финансов;
- для его освоения от аналитика не требуется углубленных знаний в области математики или программирования;
- эффективен при работе в случаях, когда правила, по которым изменяется прогнозируемая величина, неизвестны и трудновывяемы.

Рассмотрим также ряд особенностей и затруднений, связанных с использованием данного и других подобных программных продуктов нейросетевого моделирования:

1. На фондовом рынке лишь немногие из специалистов успешно справляются с эффективной настройкой нейросимуляторов особенно в тех случаях, когда к прогнозированию приходится привлекать малозначимые влияющие факторы и требуется правильно интерпретировать результаты

настройки ИНС. Для эффективного использования нейросимуляторов необходимо также хорошо понимать сущность моделируемого процесса.

2. При использовании нейронной сети необходимо учитывать влияние детерминированной периодической функции называемой в теории временных рядов «аддитивной сезонной компонентой» и определяемой методами спектрального анализа. Период сезонной компоненты составляет от 7 до 14 дней. Она может учитывать, например, то, что в первые два–три дня каждого месяца обычно наблюдается локальный подъем котировок акций, а в середине месяца существуют дни, когда на денежный рынок оказывают влияние обязательства по контрактам на куплю–продажу валюты по заранее оговоренной цене и т. д. На этапе прогноза сезонная компонента автоматически добавляется в одну из колонок электронной таблицы с данными и, таким образом, учитывается в нейросимуляторе при оценке прогнозируемого приращения котировок.

3. Практика работы с нейросимуляторами на финансовом рынке свидетельствует о том, что создание и тщательное ведение обширной, постоянно обновляемой и хорошо структурированной базы финансовых, макроэкономических и политических данных крайне важно, поскольку они существенно влияют на ситуацию и качество прогноза. Так как ситуация на рынке непрерывно изменяется, то и набор значащих влияющих факторов (или их порядок внутри этого набора) также изменяется во времени. В связи с этим, нейронную сеть необходимо время от времени настраивать и обучать заново.

4. Наличие подробной документации крайне важно при работе с нейросимулятором.

Документация обычно включает подробное описание методов и примеров, индексный и предметный указатели, а также обучающий курс. Некоторые компании–разработчики нейросимуляторов поддерживают «горячую линию» по телефону и Интернет, а также проводят семинары пользователей по обучению приемам эффективной работы с нейросимуляторами.

### Прогнозирование потребления электроэнергии

Система анализа данных о потреблении электроэнергии (компания ZSolutions) использует данные, полученные в результате обработки показаний счетчиков частных и корпоративных клиентов. Измерения проводятся каждые 15 минут, причем известно, что некоторые из них – неверные.

С помощью ИНС был построен алгоритм выявления неверных измерений, а также алгоритм прогнозирования потребления энергии в зимний период времени.

Использование данного прогноза позволило энергетической компании применить гибкую тарифную политику и сократить риск возникновения энергетического кризиса в регионе [99].

#### Прогнозирование свойств полимеров

Реализована технология прогнозирования свойств материалов в химических полимерных производствах с помощью ИНС (компания *Aspen Technology* и *NeuralWare Inc.*, 1997 г.). Данный подход оказался более эффективным и дешевым, чем разработка теоретической модели полимеров.

Так, с помощью нейросимулятора *NeuroShell* разработан новый сорт безопасного стекла (компания *DuPont*).

В заключение отметим, что прогнозирование цены нефти – одна из наиболее интересных задач, связанная с применением ИНС для целей прогноза. В виду того, что такой прогноз имеет для России принципиальное значение: цена существенно влияет на наполнение бюджета и уровень жизни – успехи в развитии технологий прогнозирования в финансово-экономической сфере трудно переоценить.

## 2. МОДЕЛЬ ИСКУССТВЕННОГО НЕЙРОНА

### 2.1. Биологический нейрон

Нервная система и мозг человека состоят из нейронов, соединенных между собой нервными волокнами. Нервные волокна способны передавать электрические импульсы между нейронами. Все процессы передачи раздражений от нашей кожи, ушей и глаз к мозгу, процессы мышления и управления действиями – все это реализовано в живом организме как передача электрических импульсов между нейронами.

*Биологический нейрон (Cell)* имеет ядро (*Nucleus*), а также отростки нервных волокон двух типов (рис.2.1) – *дендриты (Dendrites)*, по которым принимаются импульсы (*Carries signals in*), и единственный *аксон (Axon)*, по которому нейрон может передавать импульс (*Carries signals away*). Аксон контактирует с дендритами других нейронов через специальные образования – *синапсы (Synapses)*, которые влияют на силу передаваемого импульса.

Структура, состоящая из совокупности большого количества таких нейронов, получила название *биологической* (или *естественной*) нейронной сети.

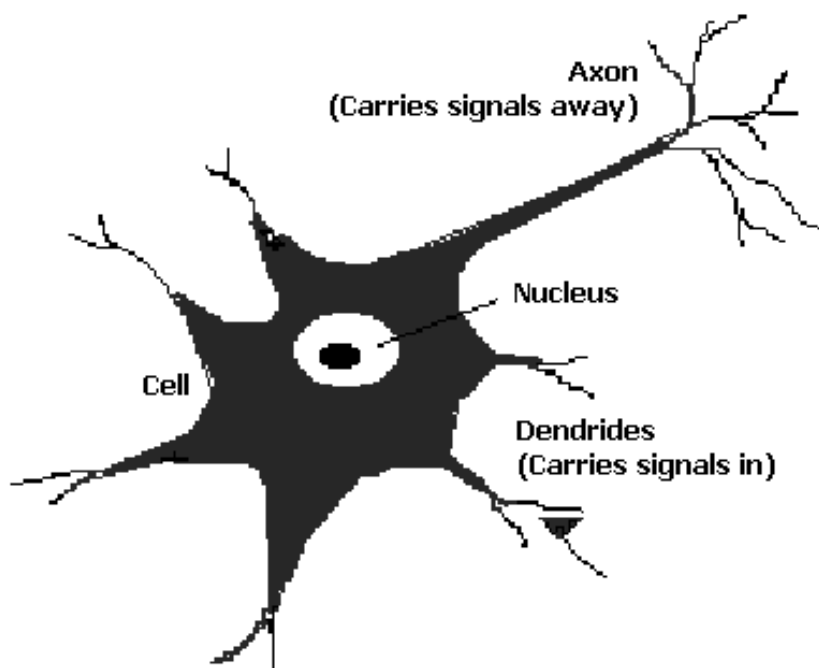


Рис. 2.1. Биологический (или естественный) нейрон

## 2.2. Искусственный нейрон

*Искусственный нейрон* (далее – нейрон) является основой любой искусственной нейронной сети.

Нейроны представляют собой относительно простые, однотипные элементы, имитирующие работу нейронов мозга. Каждый нейрон характеризуется своим текущим состоянием по аналогии с нервными клетками головного мозга, которые могут быть возбуждены и заторможены.

Искусственный нейрон, также как и его естественный прототип, имеет группу синапсов (входов), которые соединены с выходами других нейронов, а также аксон – выходную связь данного нейрона – откуда сигнал возбуждения или торможения поступает на синапсы других нейронов.

Общий вид искусственного нейрона представлен на рис 2.2.

Каждый синапс характеризуется величиной *синаптической связи* или *весом*  $w_{ij}$ , который по своему физическому смыслу эквивалентен электрической проводимости.

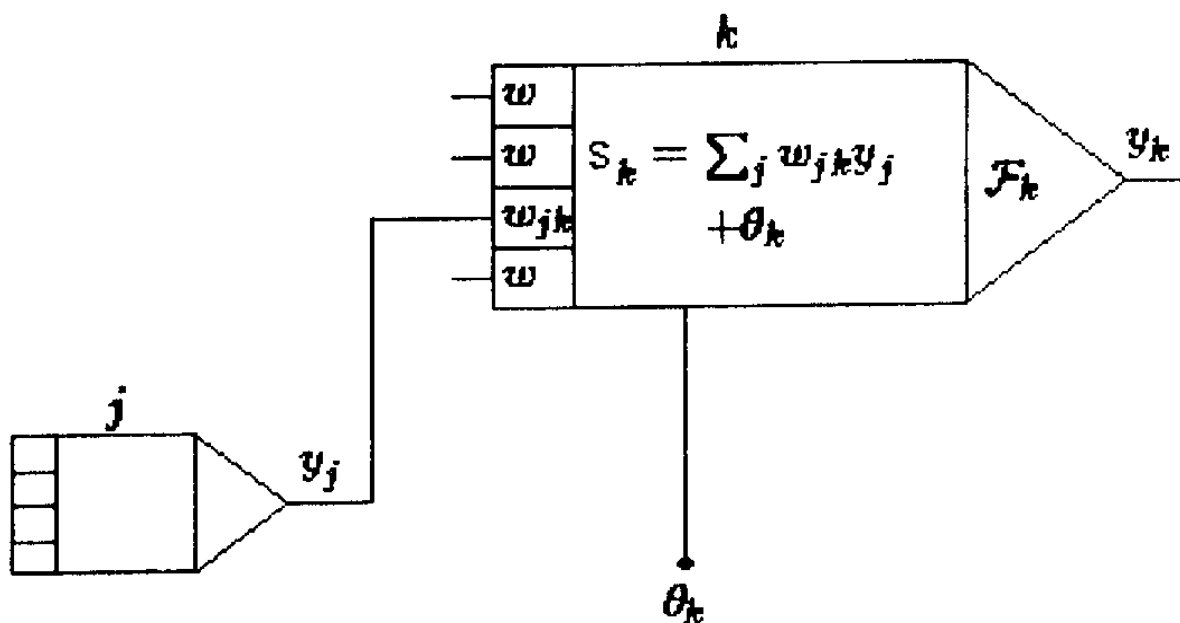


Рис.2.2. Искусственный нейрон –  
простейший элемент искусственной нейронной сети

$y_j$  – сигнал, поступающий от нейрона  $j$ ;

$s_k$  – скалярное произведение вектора входных сигналов и вектора весов;

$f_k$  – функция возбуждения;

$y_k$  – выходной сигнал нейрона

Текущее состояние нейрона определяется как взвешенная сумма его входов:

$$s = \sum_{i=1}^n x_i w_i , \quad (2.1)$$

где  $x$  – вход нейрона,  
 $w$  – соответствующий этому входу вес,  
 $n$  – число нейронов предыдущего слоя.

### 2.3. Активационная функция

Выход нейрона есть функция его состояния, т.е.

$$y = f(s) . \quad (2.2)$$

Нелинейная функция  $f(s)$  называется *активационной, сжимающей* функцией или функцией *возбуждения* нейрона.

Основные разновидности активационных функций, применяемых в ИНС, представлены на рис.2.3.

В качестве активационной функции часто используется *сигмоидальная* (*s-образная* или *логистическая*) функция, показанная на рис.2.3,с. Эта функция математически выражается по формуле (2.3):

$$f(s) = \frac{1}{1 + e^{-\alpha s}} \quad (2.3)$$

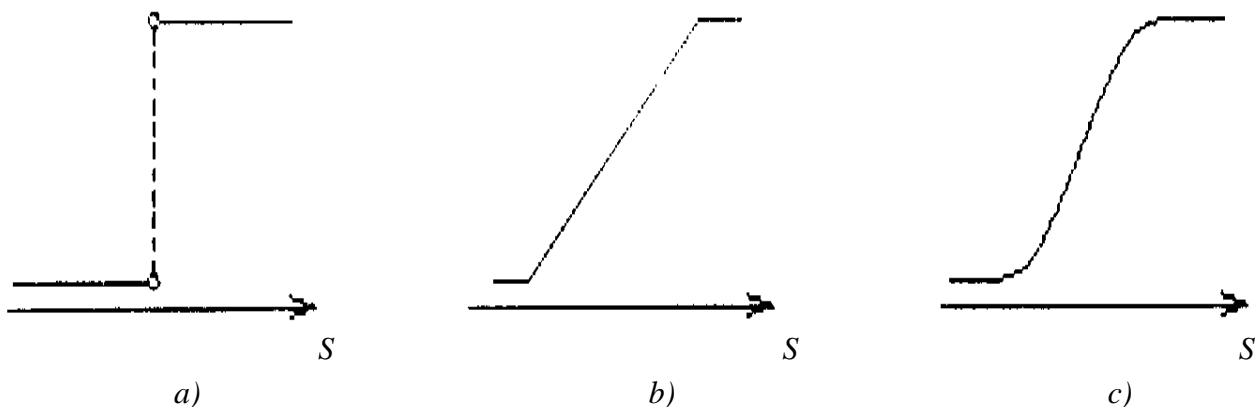


Рис.2.3. Активационная функция  
а) пороговая; б) полулинейная; в) сигмоидальная

При уменьшении  $\alpha$  сигмоидальная функция становится более пологой, в пределе при  $\alpha=0$  вырождаясь в горизонтальную линию на уровне 0,5; при увеличении  $\alpha$  сигмоидальная функция приближается по внешнему виду к функции единичного скачка с порогом  $T$  в точке  $x=0$ . Из выражения для сигмоидальной функции видно, что выходное значение нейрона лежит в диапазоне  $[0,1]$ . Одно из полезных свойств сигмоидальной функции – простое выражение для ее производной, применение которого будет рассмотрено в дальнейшем:

$$f'(x) = \alpha f(x)(1 - f(x)). \quad (2.4)$$

Следует отметить, что сигмоидальная функция дифференцируема на всей оси абсцисс, что используется в некоторых алгоритмах обучения. Кроме того, сигмоидальная функция обладает свойством усиливать малые сигналы лучше, чем большие, тем самым предотвращая насыщение от больших сигналов, так как они соответствуют областям аргументов, где сигмоидальная функция имеет пологий наклон.

Выбор структуры ИНС осуществляется в соответствии с особенностями и сложностью задачи. Для решения некоторых отдельных типов задач уже существуют оптимальные, на сегодняшний день конфигурации.

Если же задача не может быть сведена ни к одному из известных типов, разработчику приходится решать сложную проблему синтеза новой конфигурации.

Теоретически число слоев и число нейронов в каждом слое ИНС может быть произвольным, однако фактически оно ограничено ресурсами компьютера или специализированной микросхемы, на которых обычно реализуется ИНС.



### 3. МЕТОДЫ ОБУЧЕНИЯ ИСКУССТВЕННЫХ НЕЙРОННЫХ СЕТЕЙ

Чтобы ИНС «приобрела» интеллект она должна пройти процесс обучения. Функционирование ИНС, т.е. действия, которые она способна выполнять на этапе эксплуатации, зависит от величин коэффициентов синоптических связей. Поэтому, задавшись структурой ИНС, отвечающей определенной задаче, разработчик должен найти оптимальные значения для всех весовых коэффициентов  $w$ .

Этот этап называется обучением ИНС, и от того, насколько качественно он будет выполнен, зависит способность ИНС решать поставленные перед ней проблемы во время эксплуатации. Важнейшими параметрами обучения являются: качество подбора весовых коэффициентов и время, которое необходимо затратить на обучение. Как правило, два этих параметра связаны между собой обратной зависимостью и их приходится выбирать на основе компромисса.

В настоящее время все алгоритмы обучения ИНС можно разделить на два больших класса: «с учителем» и «без учителя».

#### 3.1. Метод обучения ИНС «без учителя»

ИНС предъявляются значения как входных, так и выходных параметров, и она по некоторому внутреннему алгоритму подстраивает веса своих синаптических связей.

Обучение с учителем предполагает, что для каждого входного вектора существует целевой вектор, представляющий собой требуемый выход. Вместе они называются *представительской* или *обучающей выборкой*. Обычно нейронная сеть обучается на некотором числе таких выборок. Предъявляется выходной вектор, вычисляется выход ИНС и сравнивается с соответствующим целевым вектором, разность (ошибка) с помощью обратной связи подается в ИНС, и веса изменяются в соответствии с алгоритмом, стремящимся минимизировать ошибку. Векторы обучающего множества предъявляются последовательно, вычисляются ошибки и веса подстраиваются для каждого вектора до тех пор, пока ошибка по всему обучающему массиву не достигнет приемлемо низкого уровня.

### 3.2. Метод обучения ИНС «с учителем»

При обучении «с учителем» ИНС предъявляются только входные сигналы, а выходы сети формируются самостоятельно с учетом только входных и производных от них сигналов.

Несмотря на многочисленные прикладные достижения, обучение с учителем критиковалось за свою биологическую неправдоподобность. Трудно вообразить обучающий механизм в естественном человеческом интеллекте, который бы сравнивал желаемые и действительные значения выходов, выполняя коррекцию с помощью обратной связи. Если допустить подобный механизм в человеческом мозге, то откуда тогда возникают желаемые выходы? Обучение без учителя является более правдоподобной моделью обучения в биологической системе. Развита Кохоненом и многими другими, она не нуждается в целевом векторе для выходов и, следовательно, не требует сравнения с предопределенными идеальными ответами.

Обучающее множество состоит лишь из входных векторов. Обучающий алгоритм подстраивает веса нейронной сети так, чтобы получались согласованные выходные векторы, т. е. чтобы предъявление достаточно близких входных векторов давало одинаковые выходы. Процесс обучения, следовательно, выделяет статистические свойства обучающего множества и группирует сходные векторы в классы. Предъявление на вход вектора из данного класса даст определенный выходной вектор, но до обучения невозможно предсказать, какой выход будет производиться данным классом входных векторов.

Следовательно, выходы подобной ИНС должны трансформироваться в некоторую понятную форму, обусловленную процессом обучения. Это не является серьезной проблемой. Обычно не сложно идентифицировать связь между входом и выходом, установленную сетью.

## 4. МНОГОСЛОЙНЫЙ ПЕРСЕПТРОН

### 4.1. Определения

Многослойный персептрон – это *полносвязанная полиморфная* ИНС, состоящая из двух замечательных слоев: входного и выходного, и как минимум одного внутреннего (скрытого) слоя (рис.4.1).

В *полносвязанной* ИНС каждый нейрон сети связан синаптической связью со всеми нейронами предыдущего слоя и со всеми нейронами последующего.

*Полиморфность* ИНС означает, что все нейроны сети одинаковые и имеют одинаковые активационные функции.

### 4.2. Алгоритм «Обратного распространения» (Backpropagation)

Обучение многослойного персептрона по алгоритму обратного распространения ошибки или, если коротко, алгоритм «Обратного распространения», заключается в распространении сигналов ошибки от выходов ИНС к ее входам, в направлении, обратном прямому распространению сигналов в обычном режиме работы.

Согласно методу наименьших квадратов, минимизируемой целевой функцией ошибки ИНС является величина:

$$E(w) = \frac{1}{2} \sum_{j,p} (y_{j,p}^{(N)} - d_{j,p})^2 \quad (4.1)$$

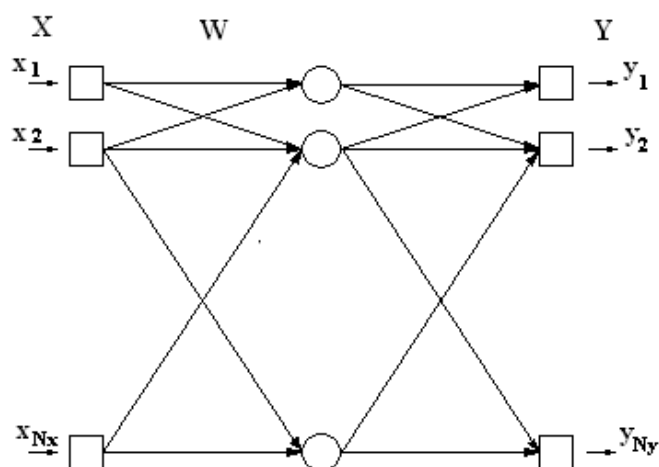


Рис. 4.1. Многослойный персептрон

где  $y_{j,p}^{(N)}$  – реальное выходное состояние нейрона  $j$  выходного слоя  $N$  нейронной сети при подаче на ее входы  $p$ -го образа;  $d_{jp}$  – идеальное (желаемое) выходное состояние этого нейрона.

Суммирование ведется по всем нейронам выходного слоя и по всем обрабатываемым сетью образам. Минимизация ведется методом градиентного спуска, что означает подстройку весовых коэффициентов следующим образом:

$$\Delta w_{ij}^{(n)} = -\eta \cdot \frac{\partial \mathcal{E}}{\partial w_{ij}} \quad (4.2)$$

Здесь  $w_{ij}$  – весовой коэффициент синаптической связи, соединяющей  $i$ -ый нейрон слоя  $n-1$  с  $j$ -ым нейроном слоя  $n$ ,  $\eta$  – коэффициент скорости обучения,  $0 < \eta < 1$ .

В то же время:

$$\frac{\partial \mathcal{E}}{\partial w_{ij}} = \frac{\partial \mathcal{E}}{\partial y_j} \cdot \frac{dy_j}{ds_j} \cdot \frac{\partial s_j}{\partial w_{ij}} \quad (4.3)$$

Здесь под  $y_j$ , как и раньше, подразумевается выход нейрона  $j$ , а под  $s_j$  – взвешенная сумма его входных сигналов, то есть аргумент активационной функции. Так как множитель  $dy_j/ds_j$  является производной этой функции по ее аргументу, из этого следует, что производная активационной функция должна быть определена на всей оси абсцисс. В связи с этим функция единичного скачка и прочие активационные функции с неоднородностями не подходят для рассматриваемых ИНС. В них применяются такие гладкие функции, как гиперболический тангенс или классический сигмоид с экспонентой. В случае гиперболического тангенса

$$\frac{dy}{ds} = 1 - s^2 \quad (4.4)$$

Третий множитель  $\partial s_j / \partial w_{ij}$ , очевидно, равен выходу нейрона предыдущего слоя  $y_i^{(n-1)}$ .

Что касается первого множителя в (4.3), он легко раскладывается следующим образом:

$$\frac{\partial \mathcal{E}}{\partial y_j} = \sum_k \frac{\partial \mathcal{E}}{\partial y_k} \cdot \frac{dy_k}{ds_k} \cdot \frac{\partial s_k}{\partial y_j} = \sum_k \frac{\partial \mathcal{E}}{\partial y_k} \cdot \frac{dy_k}{ds_k} \cdot w_{jk}^{(n+1)} \quad (4.5)$$

Здесь суммирование по  $k$  выполняется среди нейронов слоя  $n+1$ .

Введя новую переменную

$$\delta_j^{(n)} = \frac{\partial \mathcal{E}}{\partial y_j} \cdot \frac{dy_j}{ds_j} \quad (4.6)$$

мы получим рекурсивную формулу для расчетов величин  $\delta_j^{(n)}$  слоя n из величин  $\delta_k^{(n+1)}$  более старшего слоя n+1.

$$\delta_j^{(n)} = \left[ \sum_k \delta_k^{(n+1)} \cdot w_{jk}^{(n+1)} \right] \cdot \frac{dy_j}{ds_j} \quad (4.7)$$

Для выходного же слоя

$$\delta_l^{(N)} = (y_l^{(N)} - d_l) \cdot \frac{dy_l}{ds_l} \quad (4.8)$$

Теперь мы можем записать (4.2) в раскрытом виде:

$$\Delta w_{ij}^{(n)} = -\eta \cdot \delta_j^{(n)} \cdot y_i^{(n-1)} \quad (4.9)$$

Иногда для придания процессу коррекции весов некоторой инерционности, сглаживающей резкие скачки при перемещении по поверхности целевой функции, (4.9) дополняется значением изменения веса на предыдущей итерации

$$\Delta w_{ij}^{(n)}(t) = -\eta \cdot (\mu \cdot \Delta w_{ij}^{(n)}(t-1) + (1-\mu) \cdot \delta_j^{(n)} \cdot y_i^{(n-1)}) \quad (4.10)$$

где  $\mu$  – коэффициент инерционности, t – номер текущей итерации.

Таким образом, полный алгоритм обучения ИНС с помощью процедуры обратного распространения строится так:

1. Подать на входы сети один из возможных образов и в режиме обычного функционирования ИНС, когда сигналы распространяются от входов к выходам, рассчитать значения последних. Напомним, что

$$s_j^{(n)} = \sum_{i=0}^M y_i^{(n-1)} \cdot w_{ij}^{(n)} \quad (4.11)$$

где M – число нейронов в слое n-1 с учетом нейрона с постоянным выходным состоянием +1, задающего смещение;  $y_i^{(n-1)} = x_{ij}^{(n)}$  – i-ый вход нейрона j слоя n.

$$y_j^{(n)} = f(s_j^{(n)}), \text{ где } f() \text{ – сигмоид} \quad (4.12)$$

$$y_q^{(0)} = I_q, \quad (4.13)$$

где  $I_q$  – q-ая компонента вектора входного образа.

2. Рассчитать  $\delta^{(N)}$  для выходного слоя по формуле (4.8). Рассчитать по формуле (4.9) или (4.10) изменения весов  $\Delta w^{(N)}$  слоя N.

3. Рассчитать по формулам (4.7) и (4.9) (или (4.7) и (4.10)) соответственно  $\delta^{(n)}$  и  $\Delta w^{(n)}$  для всех остальных слоев, n=N-1,...,1.

4. Скорректировать все веса в ИНС

$$w_{ij}^{(n)}(t) = w_{ij}^{(n)}(t-1) + \Delta w_{ij}^{(n)}(t) \quad (4.14)$$

5. Если ошибка сети существенна, перейти на шаг 1. В противном случае – завершение обучения.

Сети на шаге 1 попеременно в случайном порядке предъявляются все тренировочные образы, чтобы сеть, образно говоря, не забывала одни по мере запоминания других. Алгоритм представлен на рис.4.2.

Из выражения (4.9) следует, что когда выходное значение  $y_i^{(n-1)}$  стремится к нулю, эффективность обучения заметно снижается.

При двоичных входных векторах в среднем половина весовых коэффициентов не будет корректироваться, поэтому область возможных значений выходов нейронов  $[0,1]$  желательно сдвинуть в пределы  $[-0.5,+0.5]$ , что достигается простыми модификациями логистических функций. Например, сигмоид с экспонентой преобразуется к виду

$$f(x) = -0.5 + \frac{1}{1 + e^{-\alpha x}} \quad (4.15)$$

Теперь коснемся вопроса емкости ИНС, то есть числа образов, предъявляемых на ее входы, которые она способна научиться распознавать.

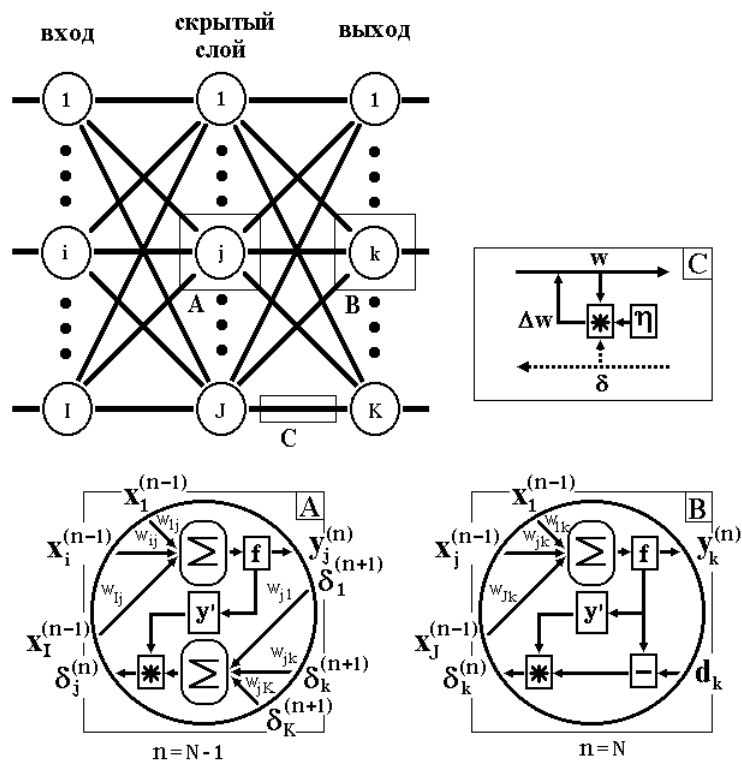


Рис. 4.2. Диаграмма сигналов в ИНС при обучении по алгоритму «Обратного распространения»

Для сетей с числом слоев больше двух, он остается открытым. Для ИНС с двумя слоями, то есть выходным и одним скрытым слоем, детерминистская емкость сети  $C_d$  оценивается так:

$$N_w/N_y < C_d < N_w/N_y \cdot \log(N_w/N_y) \quad (4.16)$$

где  $N_w$  – число подстраиваемых весов,  $N_y$  – число нейронов в выходном слое.

Следует отметить, что данное выражение получено с учетом некоторых ограничений. Во-первых, число входов  $N_x$  и нейронов в скрытом слое  $N_h$  должно удовлетворять неравенству  $N_x + N_h > N_y$ . Во-вторых,  $N_w/N_y > 1000$ . Однако, вышеприведенная оценка выполнялась для сетей с активационными функциями нейронов в виде порога, а емкость сетей с гладкими активационными функциями, например – (4.15), обычно больше. Кроме того, фигурирующее в названии емкости прилагательное «детерминистский» означает, что полученная оценка емкости подходит абсолютно для всех возможных входных образов, которые могут быть представлены  $N_x$  входами. В действительности, распределение входных образов, как правило, обладает некоторой регулярностью, что позволяет ИНС проводить обобщение и, таким образом, увеличивать реальную емкость. Так как распределение образов, в общем случае, заранее не известно. Мы можем говорить о такой емкости только предположительно, но обычно она раза в два превышает емкость детерминистскую.

В продолжение разговора о емкости ИНС логично затронуть вопрос о требуемой мощности выходного слоя сети, выполняющего окончательную классификацию образов. Дело в том, что для разделения множества входных образов, например, по двум классам достаточно всего одного выхода. При этом каждый логический уровень – «1» и «0» – будет обозначать отдельный класс. На двух выходах можно закодировать уже 4 класса и так далее. Однако результаты работы сети, организованной таким образом, можно сказать – «под завязку» – не очень надежны. Для повышения достоверности классификации, желательно, ввести избыточность путем выделения каждому классу одного нейрона в выходном слое или, что еще лучше, нескольких. Каждый из которых обучается определять принадлежность образа к классу со своей степенью достоверности, например: высокой, средней и низкой. Такие ИНС позволяют проводить классификацию входных образов, объединенных в нечеткие (размытые или пересекающиеся) множества. Это свойство приближает подобные ИНС к условиям реальной жизни.

Рассматриваемая ИНС имеет несколько «узких мест». Во-первых, в процессе обучения может возникнуть ситуация, когда большие положительные или отрицательные значения весовых коэффициентов сместят рабочую точку на сигмоидах многих нейронов в область насыщения. Малые величины производной от логистической функции приведут в соответствие с (4.7) и (4.8) к остановке обучения, что парализует ИНС. Во-вторых, применение метода градиентного спуска не гарантирует, что будет найден глобальный, а не локальный минимум целевой функции. Эта проблема связана еще с одной, а именно – с выбором скорости обучения. Доказательство сходимости обучения в процессе обратного распространения основано на производных, то есть приращения весов и, следовательно, скорость обучения должны быть бесконечно малыми, однако, в этом случае обучение будет происходить неприемлемо медленно. С другой стороны, слишком большие коррекции весов могут привести к постоянной неустойчивости процесса обучения. Поэтому в качестве  $\eta$  обычно выбирается число меньше 1, но не очень маленькое, например, 0.1, и оно, вообще говоря, может постепенно уменьшаться в процессе обучения. Кроме того, для исключения случайных попаданий в локальные минимумы иногда, после того как значения весовых коэффициентов застабилизируются,  $\eta$  кратковременно сильно увеличивают, чтобы начать градиентный спуск из новой точки. Если повторение этой процедуры несколько раз приведет алгоритм в одно и то же состояние ИНС, можно более или менее уверенно сказать, что найден глобальный максимум, а не локальный.

### 5.3. Требования к обучающим выборкам

При формировании представительских или обучающих пар (выборок) необходимо руководствоваться рядом общепринятых принципов, соблюдение которых позволяет повышать качество обучения и, как следствие, обеспечивать достоверность и надежность работы ИНС на этапе эксплуатации системы.

Выделим основные такие принципы.

1. Обучающие выборки должны обладать свойством *репрезентативности*. При обучении выборки должны «представлять» все многообразие возможных входных воздействий, которые могут возникнуть на этапе эксплуатации системы, и задача проектировщика – «научить» ИНС адекватно на всех на них реагировать.



2. При сборе информации об анализируемом процессе необходимо обеспечивать свойство *полноты* множества обучающих выборок с тем, чтобы собрать максимальное количество возможных выборок и тем самым «сообщить» ИНС во время обучения множество различные нюансов анализируемых образов или процессов. Это существенно повысит точность срабатывания ИНС на этапе эксплуатации.
3. Если множество входов ИНС типа «Многослойный персептрон» состоит из двух или более подмножеств различной природы, то необходимо *масштабировать* данные таким образом, чтобы поступающие числовые значения были соразмерны, т.е. одного порядка. Масштабирование облегчает для ИНС задачу первоначально-равного учета входных данных различной природы.
4. Обучающие выборки должны быть свободны от фоновых случайного шума и неверных (ошибочных) данных. Для этого выполняется *фильтрация* данных и ручное устранение неверных «выбросов», тем или иным способом прокраившихся в базу данных.
5. Также, обучающие выборки должны обладать свойством *непротиворечивости*. Одинаковым или почти одинаковым входным векторам ИНС во время обучения не должны соответствовать разные комбинации на выходе. В таких случаях, внутренняя логика нарушается и ИНС в принципе не может быть обучена.

Необходимо помнить, что от этапа обучения зависит эффективность работы системы в целом, и способность ИНС работать таким образом, как мы хотим, чтобы она работала. Поэтому, соблюдение вышеперечисленных принципов при проектировании ИНС является задачей первостепенной важности.

## 5. САМООБУЧАЮЩИЕСЯ КАРТЫ КОХОНЕНА

### 5.1. Определения

ИНС Кохонена или *самоорганизующиеся карты Кохонена* (Kohonen's Self-Organizing Maps) предназначены для решения задач автоматической классификации, когда обучающая последовательность образов отсутствует. Соответственно отсутствует и фиксация ошибки, на минимизации которой основаны алгоритмы обучения, например, алгоритм обратного распространения ошибки (Backpropagation).

Сеть Кохонена – это двухслойная ИНС, содержащая *входной слой* (слой входных нейронов) и *слой Кохонена* (слой активных нейронов). Слой Кохонена может быть: одномерным, двумерным или трехмерным. В первом случае активные нейроны расположены в цепочку. Во втором случае они образуют двухмерную сетку (обычно в форме квадрата или прямоугольника), а в третьем случае они образуют трехмерную конструкцию.

В силу отсутствия обучающей последовательности образов, для каждого из которых известна от учителя принадлежность к тому или иному классу, определение весов нейронов слоя Кохонена основано на использовании алгоритмов классической классификации (кластеризации или самообучения).

### 5.2. Принцип работы ИНС Кохонена

На рис.5.1 приведен пример топологической карты сети Кохонена, содержащей входной слой и слой Кохонена. Нейроны входного слоя служат для ввода значений признаков распознаваемых образов. Активные нейроны слоя Кохонена предназначены для формирования областей (кластеров) различных классов образов. На этом рисунке показаны связи всех входных нейронов лишь с одним нейроном слоя Кохонена. Каждый нейрон слоя Кохонена также соединен с соседними нейронами.

Поясним основной принцип работы ИНС Кохонена.

Введем следующие обозначения (рис.5.1):

$$W_j = (w_{j1}, w_{j2}, \dots, w_{jn})^T, \quad j = \overline{1, m} \quad (5.1)$$

– вектор весовых коэффициентов  $j$ -го нейрона слоя Кохонена,

$$X = (x_1, x_2, \dots, x_n)^T \quad (5.2)$$

– входной вектор или вектор значений признаков некоторого образца.

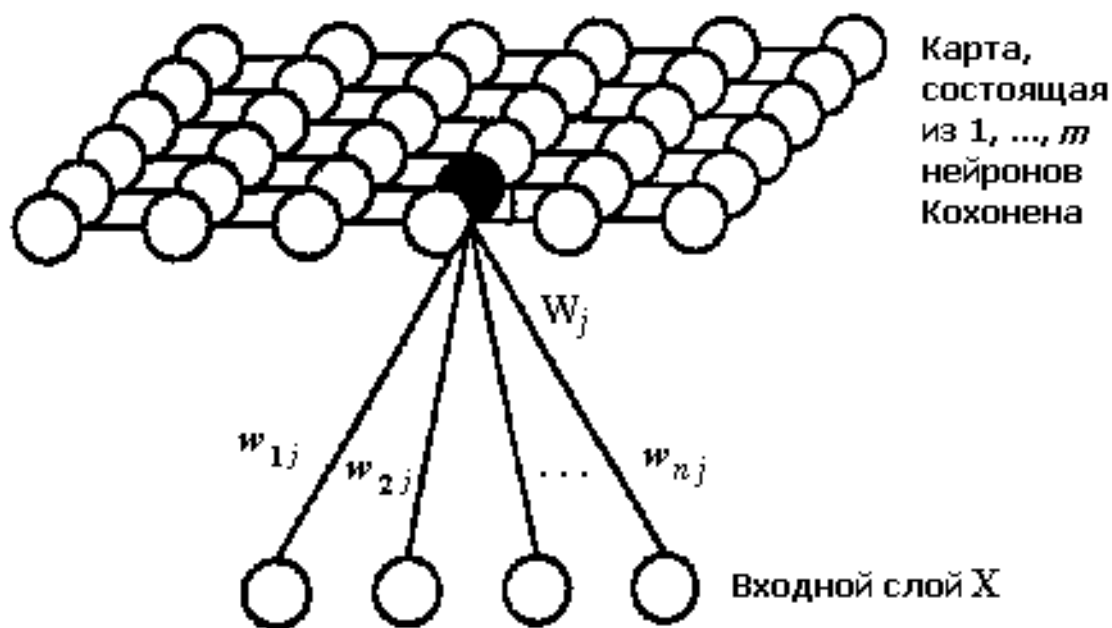


Рис.5.1. Топологическая карта сети Кохонена

На стадии обучения (точнее самообучения) сети входной вектор  $X^c$  попарно сравнивается со всеми векторами  $W_j$  всех нейронов слоя Кохонена. Вводится некоторая функция близости (например, в виде эвклидова расстояния). Активный нейрон с номером слоя Кохонена, для которого значение функции близости  $d(X, W_c)$  между входным вектором  $X$ , характеризующим некоторый образ, и вектором  $W_c$  максимально, объявляется «победителем». При этом образ, характеризующийся вектором  $X$ , относится к классу, который представляется «нейроном-победителем». В результате осуществляется преобразование  $n$ -мерного входного пространства  $R^n$  на  $m$ -мерную сетку (слой Кохонена).

Следует подчеркнуть, что это отображение реализуется в результате рекуррентной (итеративной) процедуры самообучения (*Unsupervised Learning*). Отличительная особенность этого отображения – формирование кластеров (*Cluster*) или классов. По завершении процесса самообучения на стадии реального использования ИНС Кохонена неизвестные входные образы относятся к одному из выявленных кластеров (классов).

Возникает естественный вопрос: как возникает указанное отображение топологической карты? Для ответа на него рассмотрим алгоритм самообуче-

ния сети Кохонена, полагая, что ее входной слой содержит  $n$  входных нейронов, а ее слой Кохонена –  $m$  активных нейронов.

Для определения расстояния между входным вектором  $X$  (5.2) и весовым вектором  $W_j$  (5.1)  $j$ -го нейрона слоя Кохонена можно использовать различные функции близости (обычно эвклидово расстояние).

При этом «выигрывает» тот нейрон  $c$  с весовым вектором  $W_c$ , который наиболее близок к входному вектору  $X$ :

$$\|X - W_c\| = \min_j \|X - W_j\| \quad (5.3)$$

или при использовании функции *index*, определяющей номер минимального расстояния:

$$c = \text{index} \min_j \|X - W_j\|. \quad (5.4)$$

При использовании скалярного произведения

$$X^T W_j = \sum_{i=1}^n x_i w_{ij} = \text{net}_j = z_j \quad (5.5)$$

«выигрывает» нейрон с максимальным значением этого произведения.

На стадии самообучения ИНС Кохонена осуществляется коррекция весового вектора не только «нейрона-победителя», но и весовых векторов остальных активных нейронов слоя Кохонена, однако в существенно меньшей степени – в зависимости от удаления от «нейрона-победителя». При этом форма и величина окрестности вокруг «нейрона-победителя», весовые коэффициенты нейронов которой также корректируются, в процессе обучения изменяются. Сначала начинают с очень большой области – она, в частности, может включать все нейроны слоя Кохонена.

Изменение весовых векторов осуществляется по правилу:

$$w_j(t+1) = w_j(t) + \eta(t) d_{cj}(t) [x(t) - w_j(t)], \quad j=\overline{1,m} \quad (5.6)$$

где  $w_j(t)$  – значение весового вектора на  $t$ -м шаге самообучения сети,  $d_{cj}(t)$  – функция близости между нейронами слоя Кохонена (*neighborhood Kernel*) и  $\eta(t)$  – изменяемый во времени коэффициент коррекции.

В качестве  $\eta(t)$  обычно выбирается монотонно уменьшающаяся функция:

$0 < \eta(t) < 1$ , т. е. алгоритм самообучения начинается сравнительно большими шагами адаптации и заканчивается относительно небольшими изменениями.

Обратим внимание, что в соответствии с (5.6) изменение того или иного весового вектора  $W_j$  пропорционально расстоянию между входным вектором  $X$  и этим весовым вектором  $W_j$ .

В качестве примера рассмотрим сеть Кохонена с одномерным слоем Кохонена. На рис.5.2 отображено движение весовых векторов нейронов слоя Кохонена. К входному вектору  $X$  ближе всех расположен весовой вектор  $W_5$  для нейрона  $c=5$ . Этот весовой вектор изменяется наиболее сильно: он в большей степени приближается к входному вектору  $X$ . На втором месте по степени близости находятся весовые векторы  $W_4$  и  $W_6$ . Изменение их – второе по силе (степени). Весовые векторы  $W_3$  и  $W_7$  также изменяются, однако в существенно меньшей степени.

Нейроны 1, 2, 8 и 9 расположены вне окрестности вокруг «нейрона-победителя»  $c=5$ , поэтому их весовые векторы оставляются без изменения после показа сети образца, характеризующегося вектором  $X$ .

Рассмотрим наиболее часто применяемые функции близости (двумерного слоя Кохонена).

Расстояние между нейронами  $i$  и  $j$  в двумерном пространстве:

$$z = \sqrt{(k_{i1} - k_{j1})^2 + (k_{i2} - k_{j2})^2}, \quad (5.7)$$

где  $k_{i1}$  и  $k_{i2}$  – координаты по оси  $x$  и оси  $y$  нейрона  $i$ ;

$k_{j1}$  и  $k_{j2}$  – аналогично для нейрона  $j$ .

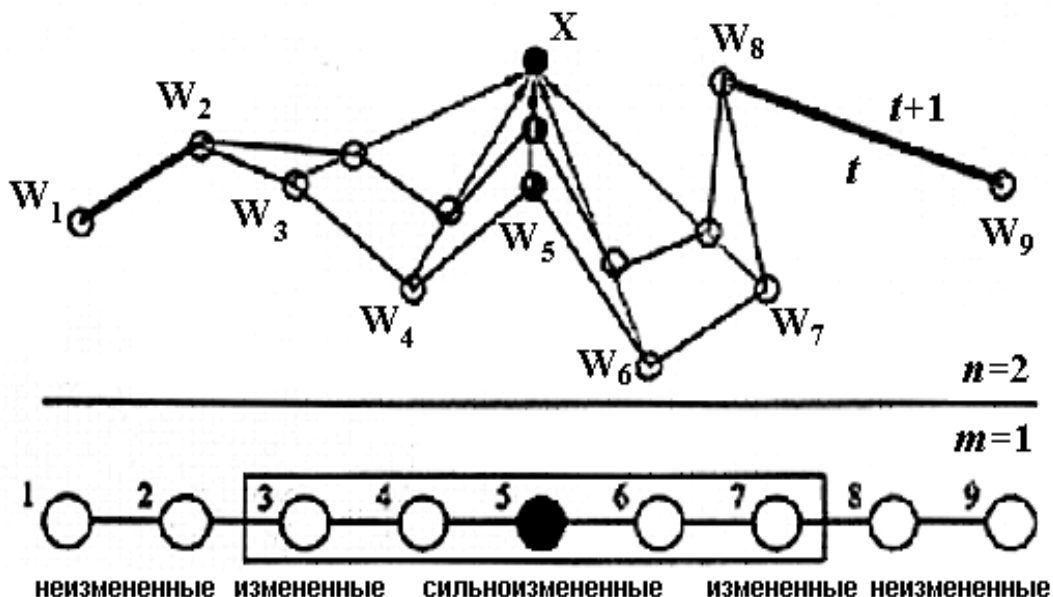


Рис.5.2. Сеть Кохонена с одномерным слоем Кохонена

При этом можно использовать следующие функции близости:

$$d_{\text{Gauss}}(z) = e^{-z^2}; \quad (5.8)$$

$$d_{\text{mexican-hat}}(z) = (1-z^2) e^{-z^2}; \quad (5.9)$$

$$d_{\cos}(z) = \begin{cases} \cos(z\pi/2), & \text{для } z < 1 \\ 0. & \end{cases} \quad (5.10)$$

Как отмечено выше, изменение весовых векторов  $W_j$  осуществляется в направлении входного вектора  $X$  многократно. В процессе самообучения варьируется как коэффициент коррекции  $\eta$ , так и радиус  $d$ , задающий окрестность вокруг «нейрона-победителя».

### 5.3. Сходимость алгоритма самообучения

При рассмотрении проблемы сходимости ограничимся одномерным случаем, когда имеется лишь один вход. Пусть  $[a, b]$  – область значений для входа (замкнутый интервал). Покажем, что алгоритм самообучения переводит вес  $x$  в середину интервала (рис.5.3).

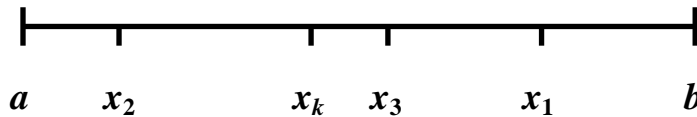


Рис.5.3. Область значений для входа

Пусть  $x_1$  – начальное значение веса одного активного нейрона слоя Кохонена. Это значение  $x_1$  выбирается случайно: интервал  $[a, b]$  разбивается на 2 подинтервала  $[a, x_1]$  и  $[x_1, b]$ . Каждое изменение  $x$  определяется его расстоянием до  $a$  и до  $b$ :

$$dx/dt = \eta (b-x)/2 + \eta(a-x)/2 = \eta((a+b)/2-x) . \quad (5.11)$$

Изменение веса  $x$  в точке  $x_1$

$$\Delta x_1 = \eta ((a+b)/2 - x_1) . \quad (5.12)$$

Обозначим  $y_i = x_i - (a+b)/2$ , тогда соотношение (5.12) можно представить так:

$$\Delta x_1 = -\eta y_1. \quad (5.13)$$

Определим математическое ожидание для значения веса  $x_2$  на следующем шаге алгоритма самообучения:

$$x_2 = x_1 + \Delta x_1 = (a+b)/2 + y_1 - \eta y_1 = (a+b)/2 + y_1(1-\eta). \quad (5.14)$$

Аналогично можно определить и  $x_3$ :

$$x_3 = (a+b)/2 + y_1(1-\eta)^2 \quad (5.15)$$

или в общем случае:

$$x_k = (a+b)/2 + y_1(1-\eta)^{k-1}. \quad (5.16)$$

При  $\eta \in [0, 1]$  значение  $x_k$  сходится к  $(a+b)/2$ .

Расширим рассмотренный одномерный случай и предположим, что одномерный слой Кохонена (линейка) содержит не один нейрон (как ранее), а  $m$  активных нейронов с весами  $x_1, x_2, \dots, x_m$ . Предположим, что эти веса упорядочены

$$0 < x_1 < x_2 < \dots < x_m < b$$

и равномерно распределены на интервале  $[a, b]$ . В этом случае в процессе самообучения весовые коэффициенты сходятся к значениям (рис.5.4):

$$x_i = a + (2i-1)(b-a)/2m, \quad i = 1, 2, \dots, m. \quad (5.17)$$

Обратим внимание, что точки (5.17) для весов  $x_i, i = 1, 2, \dots, m$  определяют наиболее устойчивые позиции, ибо

$$dx_i/dt = 0. \quad (5.18)$$

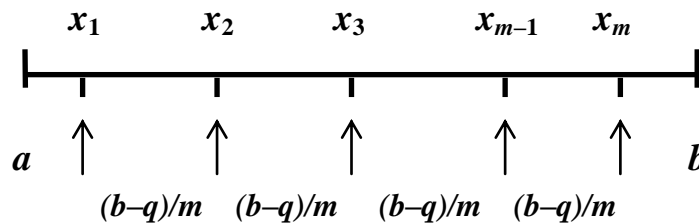


Рис.5.4. Распределение весовых коэффициентов

В двумерном случае слой Кохонена содержит  $m \times m$  активных нейронов, а областью определения для входов является декартово произведение  $[a, b] \times [c, d]$ , т. е. входной слой содержит 2 нейрона. В этом случае весовой вектор каждого нейрона слоя Кохонена имеет две составляющие – по числу входов. Каждый нейрон слоя Кохонена также характеризуется двумя координатами – по оси абсцисс и по оси ординат.

Подобно одномерному случаю можно показать, что координаты весовых векторов нейронов слоя Кохонена на оси абсцисс в процессе самообучения равномерно распределяются в интервале  $[a, b]$ :

$$a < w_1^1 < w_1^2 < \dots < w_1^m < b. \quad (2.39)$$

Аналогично для координат этих векторов по оси ординат:

$$c < w_2^1 < w_2^2 < \dots < w_2^m < d. \quad (2.40)$$

В результате самообучения сети Кохонена весовые векторы нейронов слоя Кохонена равномерно распределяются во входном пространстве.



## 6. ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ДЛЯ МОДЕЛИРОВАНИЯ НЕЙРОННЫХ СЕТЕЙ

### 3.1. Обзор программного обеспечения для моделирования

Различают универсальные и прикладные программные продукты для моделирования нейронных сетей (*Neural Network Simulators*).

*Универсальные* или *объектно-инвариантные среды* позволяют синтезировать оптимальные нейронные сети, применяемые для решения широкого класса задач, с предложением различных парадигм и алгоритмов обучения.

*Прикладные среды моделирования* ориентированы для синтеза ИНС, применяемых в той или иной отрасли, прикладной области или специфичной задаче.

Среди важнейших свойств нейросетевых симуляторов (нейросимуляторов) – способность синтезировать код программы результирующей нейронной сети на алгоритмическом языке высокого уровня (чаще всего – Си и Паскаль). Такой код впоследствии легко интегрировать в пользовательскую программу.

В таблице П.1 Приложений представлены наиболее распространенные *универсальные* программные среды для моделирования ИНС.

### 3.2. Краткое описание программного продукта *TRAJAN*

Программный продукт *TRAJAN* компании *TRAJAN Software Co.* (Великобритания) является симулятором полного цикла и предназначен для моделирования в среде *Microsoft Windows* различных видов ИНС и алгоритмов обучения.

Симулятор включает широкие возможности для графического и статистического контроля в процессе моделирования параметров и характеристик эффективности синтезируемых ИНС.

Рассмотрим основные функциональные и потребительские характеристики программного продукта *TRAJAN*.

#### Автоматизация процесса синтеза ИНС

Программное обеспечение *TRAJAN* позволяет автоматизировать следующие процессы:

- формирования представительских выборок и переменных;
- выбора вида ИНС и ее структуры;
- обучения ИНС;

- сохранения оптимальных параметров ИНС;
- выдачи графической и статистической информации о характеристиках синтезируемой ИНС.

Эти возможности позволяют работать в программной среде пользователям незнакомым с нюансами теории нейронных сетей, в то время как специалисты в данной области могут существенно сокращать рутинные этапы синтеза оптимальных ИНС и временные затраты.

При проектировании ИНС, направленной на решение особенно сложных задач, моделирование в среде программного продукта *TRAJAN*, как правило, занимает несколько дней.

#### Формирование представительской выборки

Важнейший этап моделирования – формирование репрезентативной представительской выборки нейронной сети для решения конкретной задачи. Сложность этой проблемы обусловлена возможной взаимной зависимостью (корреляцией) между входами ИНС. Программный продукт *TRAJAN* реализует ряд алгоритмов решения этой проблемы, включая форвардный и пошаговый выбор, генетические алгоритмы и ряд других.

#### Многослойные персептроны

*TRAJAN* поддерживает многослойные персептроны, обучающиеся по алгоритму «обратного распространения». Также могут быть использованы алгоритм быстрого распространения и *Delta-Bar-Delta*-алгоритм.

#### Карты Кохонена

*TRAJAN* позволяет моделировать ИНС, реализованные в виде самоорганизующихся карт Кохонена. При этом процесс моделирования сопровождается визуализацией графического окна наилучших частот и окна с топологией карты, что позволяет в реальном времени моделирования локализовать и пометить кластеры.

#### Гибридные нейронные сети

*TRAJAN* позволяет моделировать также гибридные ИНС, построенные с использованием комбинации двух или нескольких парадигм.

#### Бейесовские сети

*TRAJAN* поддерживает современные достижения в области теории Байесовских вероятностных и регрессионных ИНС. Данные парадигмы подразумевают почти мгновенное обучение ИНС, что неопределимо при модельном экспериментировании.

### Линейные модели

Большинство специалистов сравнивают результаты нелинейных моделей с результатами, получаемыми при использовании линейной модели нейронов. Линейные модели зачастую способны эффективно достигать цели за меньшее время при использовании того же программного обеспечения.

### Интерфейс пользователя

Удобный пользовательский интерфейс программного продукта позволяет обеспечивать простой доступ к большим объемам информации.

Представительские выборки и структура ИНС сохраняются в файлах с единым именем и различными расширениями, что обеспечивает легкость группировки исходных данных для моделирования.

Обновляющиеся в реальном времени графики и гистограммы позволяют наблюдать за обучением и исполнением ИНС, оперативно реагировать на ход моделирования. При решении задач классификации или аппроксимации автоматически вычисляются разнообразные статистические параметры и характеристики.

Специализированные топологические карты и кластерные диаграммы применяются при изучении и анализе результатов моделирования.

Фактически вся символьная и числовая информация доступна в *электронных таблицах (Datasheets)*, т.е. может быть мгновенно импортирована и экспортирована через буфер обмена *Microsoft Windows*. Графическая информация также может быть экспортирована, например, для составления отчета о модельных экспериментах.

### Ограничения

*TRAJAN* поддерживает ИНС «глубиной» 128 слоев, хотя, в подавляющем большинстве случаев, требуемое (наилучшее) количество слоев существенно меньше.

В *TRAJAN* первый слой является всегда слоем входа. Он используется только для ввода величин в ИНС, так как нейроны входного слоя не подразумевают никакой обработки. Последний слой является выходным, и результаты выполнения нейронов этого слоя являются и выходом ИНС в целом.

### 3.3. Описание основных этапов работы в среде TRAJAN

#### Создание ИНС

Новая ИНС создается в TRAJAN с помощью окна *Network Creating* (*Создание сети*), которое доступно из меню *File/New/Network* или по нажатию соответствующей кнопки на панели инструментов.

Для создания новой ИНС после появления на экране окна *Network Creating* следует:

- выбрать тип сети;
- определить число слоев и их размерности.

#### Выбор типа ИНС

TRAJAN предлагает несколько видов ИНС для моделирования. ИНС типа *Многоуровневый персептрон* выбрана в данном окне по умолчанию.

#### Определение числа слоев и их размерности

При задании количества слоев ИНС следует учитывать, что TRAJAN может поддерживать ИНС вплоть до 128 слоев по 128 нейронов в каждом, при этом первый из них всегда является входным и используется только для получения сетью исходных данных, а последний слой является выходным, и выходы его нейронов являются выходами всей ИНС в целом.

Задать количество нейронов в каждом слое позволяет матрица, представленная в окне *Network Creating*. Она выглядит как небольшая электронная таблица. Количество нейронов в каждом слое ИНС определяется с помощью первой ячейки этой матрицы, при этом любые слои с нулевым количеством нейронов будут проигнорированы.

После того, как задано количество нейронов в каждом слое, TRAJAN самостоятельно определит количество слоев в сети путем выбора из матрицы всех слоев, у которых количество нейронов отлично от нуля.

*Примечание. Можно заметить, что матрица содержит строку для задания ширины каждого слоя. Данная функция редко используется в TRAJAN и необходима только для нейронных сетей, использующих карты Кохонена.*

#### Подготовка нейронной сети к обучению

Одной из ключевых характеристик ИНС является их способность обучаться решать поставленные перед ними задачи с использованием представительских выборок. То есть вместо того, чтобы выставлять нейронам их веса вручную (что в принципе конечно возможно), создается некоторый набор представительских выборок, определяющих решение поставленной задачи,

для обучения предварительно подготовленной структуры ИНС. При этом используется один из алгоритмов обучения ИНС, наиболее популярным из которых является алгоритм *Обратного распространения*.

Обучающий алгоритм во время своей работы подбирает веса нейронов в ИНС, исходя из информации, содержащейся в представительской выборке.

В *TRAJAN* набор представительских выборок (обучающих пар) создается с помощью окна *Pattern Set Creating*, которое доступно из пункта меню *File/New/Pattern*.

После выбора этой команды меню на экране появляется диалоговое окно, в котором необходимо задать количество входов и выходов формируемых обучающих пар.

*Примечание. Для целей обучения количество входов и выходов обучающих пар должно совпадать с количеством нейронов, содержащихся во входных и выходных слоях ИНС.*

После этого *TRAJAN* запустит редактор вводимых обучающих пар, в котором отобразит матрицу обучающих пар. Каждая обучающая пара представлена в матрице своей строкой, которая содержит значение входных и соответствующих им выходных сигналов ИНС.

После создания ИНС и набора обучающих пар для ее обучения исходные данные моделирования сохраняются в файле. Для сохранения нейронной сети необходимо воспользоваться пунктом меню *File/Save/Network*. После выбора команды *TRAJAN* запрашивает имя файла, в котором будет сохранена моделируемая ИНС.

Для сохранения набора обучающих пар необходимо воспользоваться пунктом меню *File/Save/Pattern*. После выбора данной команды *TRAJAN* запрашивает имя файла, в котором будут сохранены введенные представительские выборки (обучающие пары).

#### Редактирование сети и представительских выборок

Средства редактирования (меню: *Edit/Network*, окно: *Network Editor*) позволяют модифицировать следующие параметры определенной ранее ИНС:

- функцию ошибки ИНС;
- изменить активационную функцию нейронов;
- изменить значения весов связей;
- удалить и добавить слои;
- удалить и добавить нейроны в текущем слое.

Электронная таблица (матрица) представительских выборок позволяет изменять любые имеющиеся обучающие пары, а также их удалять и добавлять новые.

### Обучение нейронной сети

В настоящее время программным продуктом *TRAJAN* поддерживаются наиболее распространенные алгоритмы обучения ИНС.

Рассмотрим процедуру обучения в *TRAJAN* по алгоритму *обратного распространения*.

Алгоритм обратного распространения работает, подготавливая ИНС, используя доступные ему данные, которые хранятся в наборе представительских выборок и которые были подготовлены для обучения сети. На каждой итерации (в терминах ИНС – эпохе), весь составленный набор обучающих пар предоставляется сети. Выходы, получаемые сетью, сравниваются с желаемыми результатами. Ошибка ИНС вычисляется как разность между желаемыми и фактическими результатами и используется для регулирования весов нейронов в сети.

Перед началом обучения ИНС выполняются следующие действия:

- Используя меню *Statistics/Training Graph*, открывается окно *Training error Graph*;
- Используя меню *Train/Backprop*, открывается окно *Back Propagation*;
- Окна располагаются так, чтобы они не перекрывали друг друга;
- Запускается алгоритм обучения нажатием кнопки *Train* в окне *Back Propagation*. При этом зависимость среднеквадратической ошибки от числа итераций будет показана на графике, расположенном в окне;
- Увеличивается максимальное число итераций и алгоритм запускается (с помощью кнопки *Train*) до тех пор, пока среднеквадратическая ошибка не примет приемлемого малого значения.

Вначале моделирования при использовании небольшого числа итераций среднеквадратическая ошибка уменьшается, но незначительно. Данный факт обусловлен тем, что задача «исключающего «ИЛИ» для ИНС, как не парадоксально, гораздо сложнее в решении, чем многие более сложные задачи.

Окно *Training Error Graph* отображает общую ошибку обучения ИНС, однако, иногда бывает полезно пронаблюдать за работой ИНС при использовании отдельно взятой обучающей пары, с помощью окна *Pattern Error*.

### Запуск ИНС

После обучения ИНС, она готова к запуску. Запустить ИНС на выполнение можно несколькими способами.

#### *Запуск с текущим набором представительских выборок*

Нейронная сеть может быть запущена с предъявлением либо полного набора представительских выборок, использованных ранее при обучении ИНС, либо с предъявлением одиночных выборок. При этом необходимо воспользоваться пунктом меню *Run/Single Pattern*, чтобы получить информацию о работе ИНС при предъявлении одной отдельно взятой представительской выборки или целого набора представительских выборок.

#### *Запуск индивидуальной представительской выборки, не входящей в набор обучающих пар*

Часто бывает необходимо проверить работу ИНС на представительской выборке, которая не входила в набор обучающих пар, использованных ранее при обучении, при решении следующих задач:

- Прогнозирование появления новых данных с заранее неизвестными нейронной сети выходами. Если выходы заранее известны, то можно оценить качество работы подготовленной ИНС. В противном случае, результаты, полученные при запуске ИНС, могут быть использованы в качестве прогноза. Данный тип задач для ИНС будет рассмотрен в лабораторной работе № 5.
- Распознавание образов (подробнее данная задача будет рассмотрена в лабораторной работе № 3). В этом случае, оценивается чувствительность ИНС к небольшому изменению параметров исследуемого вектора, с помощью которого проводилось обучение.

## 6. ЛАБОРАТОРНЫЙ ПРАКТИКУМ

Цель – предоставить обучаемому возможность самостоятельно решить несколько несложных, но весьма интересных задач, используя программный продукт для моделирования нейронных сетей *TRAJAN* компании *Trajan Software Ltd.* (Великобритания) или любую другую аналогичную программу на выбор (см. таблицу П.1 Приложения).

Лабораторный практикум отражает десятилетний опыт подготовки российских и иностранных студентов в области применения искусственных нейронных сетей на кафедре Информационно-измерительной техники Московского энергетического института (технического университета) и на кафедре ИТ-7 «Автоматизированные системы обработки информации и информационные технологии» Московского государственного университета приборостроения и информатики (МГУПИ).

Лабораторный практикум построен на одной из простых, и в то же время наиболее часто применяемой, модели многослойного персептрона и алгоритма обратного распространения. Позволяет закрепить теоретический материал и получить некоторые практические навыки в синтезе и обучении ИНС.

### 6.1. Лабораторная работа № 1.

#### Создание и обучение простейшей нейронной сети

Цель – освоение основных приемов работы с демонстрационной версией программного продукта *TRAJAN* в ходе создания и обучения простейшей нейронной сети [1].

##### Задание

1. Повторить соответствующий теоретический материал (глава 2: п. 2.1 – 2.3 и глава 3).
2. Создать и обучить ИНС, которая будет способна решать логическую задачу исключающего «ИЛИ». Таблица истинности для весьма полезной логической функции приведена в табл. 6.1.
3. Проверить работоспособность нейронной сети.
4. Ответить на вопросы для самопроверки № 1 – 4.

##### Создание нейронной сети

Новая ИНС создается в *TRAJAN* с помощью окна *Network Creating (Создание сети)*, которое доступно из меню *File/New/Network* или по нажатию соответствующей кнопки на панели инструментов.



Таблица 6.1. Таблица истинности для логической функции исключающего «ИЛИ»

Вход 1	Вход 2	Выход (Истина?)
0	0	1
0	1	0
1	0	0
1	1	1

После того, как на экране появится окно *Network Creating* для создания новой нейронной сети, следует произвести следующие действия.

#### Выбор типа нейронной сети

Демонстрационная версия *TRAJAN* предлагает два типа нейронных сетей. Для решения задач, представленных в данном лабораторном практикуме, рекомендуется использовать ИНС типа *многослойный персептрон*, которая выбрана в данном окне по умолчанию.

#### Определение количества слоев в нейронной сети и их размерностей

При задании количества слоев вашей ИНС, следует учитывать следующие особенности пакета *TRAJAN*:

1. Программный продукт поддерживает максимальный размер ИНС: 128 слоев по 128 нейронов в каждом, при этом первый слой всегда является входным и используется только для получения сетью исходных данных, а последний – выходным, и выходы его нейронов являются выходами всей сети в целом.
2. Для решения поставленной задачи рекомендуется использовать простейшую структуру ИНС, состоящую из трех слоев: входной слой с двумя нейронами, скрытый слой с двумя нейронами и выходной слой с одним нейроном (2-2-1).

Для задания количество нейронов в каждом слое используется матрица, представленная в окне *Network Creating*. Она выглядит как небольшая электронная таблица.

Необходимо определить количество нейронов в каждом слое сети с помощью первой ячейки этой матрицы, при этом любые слои с нулевым количеством нейронов будут проигнорированы.

После задания количества нейронов в каждом слое ИНС, *TRAJAN* самостоятельно определит количество слоев путем выбора из матрицы тех слоев, у которых количество нейронов отлично от нуля.

*Примечание. Можно заметить, что матрица содержит строку для задания «ширины» каждого слоя. Строка редко используется в TRAJAN для карт Кохонена, хотя с помощью нее можно задавать и ширину слоев для нейронных сетей некоторых других типов.*

### Обучение нейронной сети

Алгоритм обратного распространения обучает ИНС, используя доступные ему данные, которые хранятся в наборе представительских выборок для обучения. На каждой итерации (в терминах программного продукта TRAJAN – «эпохе»), ИНС предоставляется весь подготовленный набор обучающих пар. Выходы, получаемые ИНС, сравниваются с желаемыми результатами. При этом ошибка ИНС вычисляется как разность между желаемыми и фактическими результатами и используется для регулирования весов нейронов в сети.

Для обучения ИНС необходимо:

- открыть окно *Training Error Graph*, используя позицию меню *Statistics/Training Graph*;
- открыть окно *Back Propagation*, используя позицию меню *Training/Backprop*;
- расположить на экране окна так, чтобы они были оба видны и не перекрывали друг друга;
- запустить алгоритм обучения путем нажатия на кнопку *Train (Обучение)* в окне *Back Propagation*. При этом зависимость среднеквадратической ошибки обучения нейронной сети от числа используемых итераций будет вычерчиваться на графике в окне *Training Error Graph*;
- увеличить число итераций в окне *Back Propagation* и обучить ИНС вновь, нажав кнопку *Train*.

Вначале моделирования при использовании небольшого числа итераций, среднеквадратическая ошибка уменьшается, но незначительно. Это обусловлено тем, что задача «исключающего «ИЛИ» для ИНС, как не парадоксально, гораздо сложнее в решении, чем многие более сложные задачи.

Окно *Training Error Graph* отображает общую ошибку обучения ИНС, однако иногда бывает полезно пронаблюдать за работой сети при использовании отдельно взятой обучающей пары. Данный режим реализуется в TRAJAN с помощью окна *Pattern Error*.

### Запуск сети

После обучения ИНС готова к запуску, причем запустить ее на выполнение можно несколькими способами.

### Запуск, используя текущий набор представительских выборок

ИНС может быть запущена с предъявлением полного набора представительских выборок, использованных ранее при ее обучении, или выполняемыми наборами по одиночке. При этом необходимо воспользоваться пунктом меню *Run/Single Pattern*, чтобы получить информацию о работе ИНС при предъявлении одной отдельно взятой представительской выборки или целого набора представительских выборок.

### Запуск индивидуальной представительской выборки, не входящей в набор обучающих пар

При решении целого ряда задач необходимо проверять работу ИНС на представительской выборке, которая не входила в набор обучающих пар, использованных ранее при обучении. Например:

1. Прогнозирование появления новых данных с заранее неизвестными нейронной сети выходами. Если выходы заранее известны, то можно оценить качество работы подготовленной нейронной сети. В противном случае, результаты, полученные при запуске, могут быть использованы в качестве прогноза. Данный тип задач для ИНС будет рассмотрен в лабораторной работе № 5.
2. Распознавание образов (задача будет рассмотрена в работе № 3). В этом случае оценивается чувствительность ИНС к небольшому изменению параметров исследуемого вектора, с помощью которого проводилось обучение.

### Замечание

ИНС, подготовленная в данной лабораторной работе, была обучена с использованием всех возможных для нее обучающих пар, поэтому она может быть запущена на выполнение с использованием каждой из четырех представительских выборок. Следовательно, можно будет оценить работу ИНС на каждой из них.

При запуске, также возможно одновременное использование всего набора представительских выборок для оценивания общих параметров работы ИНС.

## 6.2. Лабораторная работа № 2.

### Определение направления двоичного сдвига

*Цель* – построение, обучение и тестирование ИНС, предназначенной для определения направления сдвига двоичного кода.

#### Задание

1. Повторить соответствующий теоретический материал.
2. Создать и обучить ИНС для определения направление циклического сдвига четырехпозиционного двоичного кода.
3. Проверить работоспособность ИНС.
4. Ответить на вопросы для самопроверки № 5–8.

#### О применении нейронных сетей для решения задачи классификации

Типовая задача ИНС – классификация того или иного исследуемого вектора (объекта). Получив в процессе обучения исходные данные об объекте, ИНС определяет, к какому из множества классов принадлежат исследуемые векторы. Проблема «исключающего ИЛИ», рассмотренная в предыдущей лабораторной работе, является примером решения именно такой задачи. Если исследуемый вектор может принадлежать только к одному из двух классов, то задача называется двухклассной. Задача, поставленная в данной работе, также сводится к двухклассной.

Простейший путь решения задачи двухклассной классификации при помощи ИНС – формирование у ИНС единственного выхода, который получает значение «1» для одного класса и «0» – для другого. Значения, лежащие внутри данного диапазона, характеризуют степень принадлежности объекта к тому или иному классу. Действительно, на том или ином выходе многоуровневого персептрона практически невозможно получить значения равные точно «0» или «1», хотя к этим значениям иногда можно подойти довольно близко.

Таким образом, для решения двухклассных задач с использованием одного выхода необходимо задаваться *уровнем доверия*, например: если значения выхода выше 0,95 – считать, что объект (исследуемый вектор) принадлежит к одному классу, а если ниже 0,05 – к другому.

#### Решение задачи классификации в TRAJAN

ИНС с предъявлением единственной представительской выборки запускается в окне *Run Single Pattern (Запустить единственный образец)* или

в окне *Run One-off Pattern* (Запустить одиночный образец не входящий в представительскую выборку).

TRAJAN сравнивает выходную величину ИНС с пределами доверия и определяет:

- если выход выше установленного верхнего порога, исследуемый вектор (объект) классифицируется положительно;
- если выход ниже установленного нижнего порога, то сообщается о негативной классификации;
- если значение выхода находится между порогами, то сообщается о том, что исследуемый вектор (объект) классифицировать не удалось.

Общая статистика результатов классификации осуществляется при нажатии кнопки *Run* в окне *Statistic/Classification*, которое открывается из меню *Statistic/Classification*. Статистика в этом окне отображается в виде матрицы, содержащей один столбец для каждого класса. Каждый столбец содержит две секции: «Общая статистика» и «Статистика процесса классификации», разделенные широкой горизонтальной чертой.

Секция «Общая статистика» содержит следующую информацию:

- *Total (Всего)* – количество образцов данного класса в наборе.
- *Correct (Правильные)* – количество образцов данного класса правильно классифицированных ИНС.
- *Wrong (Неправильные)* – количество образцов неправильно классифицированных ИНС (как принадлежащих к другому классу).
- *Unknown (Неизвестные)* – количество образцов данного класса, которые ИНС не смогла классифицировать.

Секция «Статистика процесса классификации» показывает, сколько исследуемых векторов (представительских выборок) было отнесено к каждому классу. При этом неклассифицированные векторы в данной секции не отображаются.

### Нейронная сеть для определения направления двоичного сдвига

Для решения поставленной задачи, следует построить и обучить ИНС, которая должна будет определять направление двоичного сдвига.

Операция двоичного сдвига является типичной для многих языков программирования. Сущность ее заключается в том, что число представляется в двоичном коде, а затем с полученной последовательностью производится операция циклического сдвига вправо или влево. Если производится сдвиг

влево, у числа самая первая (левая) цифра переставляется в конец, а если сдвиг производится вправо, то последняя (правая) цифра переставляется в начало.

Для построения ИНС представим в четырехпозиционном двоичном коде числа от «0» до «15». Далее, следует определить количество входов и выходов ИНС, необходимой для решения поставленной задачи. Очевидно, что для определения направления сдвига на входы ИНС необходимо представить исходную четырехпозиционную двоичную последовательность и четырехпозиционную двоичную последовательность, которая получилась в результате сдвига.

Выходной слой ИНС может состоять из одного нейрона. Его значение будет равно «0», если сдвиг произведен влево, и «1» – если сдвиг произведен вправо. Таким образом, для решения данной двухклассной задачи необходимо построить ИНС с *восемью* входами и *одним* выходом.

#### *Обучение нейронной сети*

Определив количество входов в сети, приступим к созданию набора обучающих пар для обучения сети. Для этого выберем шесть четырехпозиционных двоичных кодов и выполним с ними операции сдвига вправо и влево (табл. 6.2).

Таблица 6.2. Результаты сдвига влево и вправо четырехпозиционного кода

№№	Исходный двоичный код	Код после сдвига влево	Код после сдвига вправо
1	0001	0010	1000
2	0010	0100	0001
3	0011	0110	1001
4	0100	1000	0010
5	0101	1010	1010
6	0110	1100	0011
7	0111	1110	1011
8	1000	0001	0100
9	1001	0011	1100

Следует заметить, что в качестве представительских выборок нельзя выбирать числа с двоичным представлением 0000, 1111, 1010, 0101, поскольку в независимости от направления сдвига (влево или вправо) для этих чисел будет получен один и тот же результат. И, следовательно, обучающие выборки станут взаимно-противоречивые.

После подготовки набора представительских выборок (обучающих пар) и обучения, следует протестировать получившуюся ИНС (проверить качество ее обучения). Тестирование проводится на оставшихся трех четырехпозиционных двоичных кодах табл. 6.2, которые не вошли в набор представительских выборок, использованный при обучении.

### **6.3. Лабораторная работа № 3.**

#### **Распознавание символов**

*Цель* – разработать и исследовать ИНС обратного распространения, предназначенную для распознавания образов.

##### Задание

1. Повторить соответствующий теоретический материал (п. 1.3).
2. Построить и обучить ИНС, которая могла бы решать задачу распознавания символов.
3. Произвести тестирование ИНС сети при добавлении шума.

##### Описание работы

На качество решения поставленной задачи в сильной степени влияют ограничения, которые накладываются производителями на демонстрационные версии своих программных продуктов. Так, в демо-версии программного продукта *TRAJAN* количество нейронов в слое не может превышать 9, поэтому при распознавании символов будем оперировать матрицей  $3 \times 3$ .

##### *Определение структуры нейронной сети*

Представим в виде матрицы  $3 \times 3$  четыре латинские буквы «X», «Y», «L», «I» и обучим ИНС распознавать их матричное представление (см. табл. 6.3).

В соответствии с табл. 6.3 входной сигнал для ИНС может быть представлен в виде развернутого раstra – вектора длиной 9. Например, для буквы X это:

**101010101**

Таблица 6.3. Матричное представление для букв «X», «Y», «L» и «I»

<i>X</i>			<i>Y</i>			<i>I</i>			<i>L</i>		
1	0	1	1	0	1	0	1	0	1	0	0
0	1	0	0	1	0	0	1	0	1	0	0
1	0	1	0	1	0	0	1	0	1	1	1

Теперь определимся с выходами ИНС. Очевидно, что для распознавания образов ИНС должна иметь возможность формировать столько выходных сигналов, сколько образов она должна уметь распознавать.

В нашем случае таких образов четыре, поэтому возможны два варианта представления выходных данных ИНС:

- выходной слой с двумя нейронами (выходами), т.е. каждому символу ставится в соответствие двухпозиционный двоичный код;
- выходной слой с четырьмя нейронами (выходами), т.е. каждому символу свой выход.

Предлагается выбрать любой вариант.

#### *Обучение нейронной сети*

Набор обучающих пар, используемых для обучения ИНС, составляется с учетом того, какой вариант формирования выходного слоя выбран в предыдущем разделе. Если выбран вариант с двумя выходами – каждой букве ставится в соответствие двухпозиционный двоичный код, то выходной слой выглядит следующим образом:

**X – 00                      Y – 01                      I – 10                      L – 11**

Если выбран вариант с четырьмя выходами, то выходной слой такой:

**X – 0001                      Y – 0010                      I – 0100                      L – 1000**

После того, как набор представительских выборок (обучающих пар) создан, необходимо обучить ИНС и проверить, насколько корректно она решает поставленную задачу.

#### *Проверка работы нейронной сети*

После качественного обучения ИНС, следует внести в исходные данные некоторый шум (хотя это сделать непросто, так как в матрицу 3×3 очень трудно добавлять шум).



Например, вместо закодированного раstra буквы «I»

**010010010**

подайте на вход ИНС код:

**010110010**

и посмотрите: удаётся ли ИНС распознать данный символ, несмотря на «ухудшение» (зашумление) кода.

#### **6.4. Лабораторная работа № 4**

##### **Искусственный нос**

*Цель* – разработать и исследовать ИНС обратного распространения для искусственного носа, предназначенного для химического анализа воздушной и жидкой среды.

##### Задание

1. Повторить соответствующий теоретический материал (п.1.4).
2. Исследовать и проанализировать имеющиеся экспериментальные данные (табл.6.4). Определить количество вводов и выводов «многослойного персептрона» (полносвязанной ИНС обратного распространения) с учетом предложенного набора датчиков и количества обнаруживаемых веществ.
3. Создать и обучить ИНС, предназначенную для обнаружения примесей с помощью анализа показаний химических датчиков.
4. Обучить ИНС с помощью представительских выборок (обучающих пар), приведенных в табл.6.5.
5. Определить наилучшую (оптимальную) структуру ИНС с точки зрения минимизации среднеквадратической ошибки обучения.
6. Для п.5 построить графические зависимости среднеквадратической ошибки обучения от количества нейронов, используемых в скрытых слоях.
7. Ответить на контрольные вопросы, приведенные в конце описания данной лабораторной работы.

##### Определение количества вводов и выводов нейронной сети

Сформируйте представительские выборки (обучающие пары) для проведения обучения.

Для этого используйте следующие рекомендации.

1. Количество входов ИНС должно соответствовать количеству использованных химических датчиков (рис. 6.1).

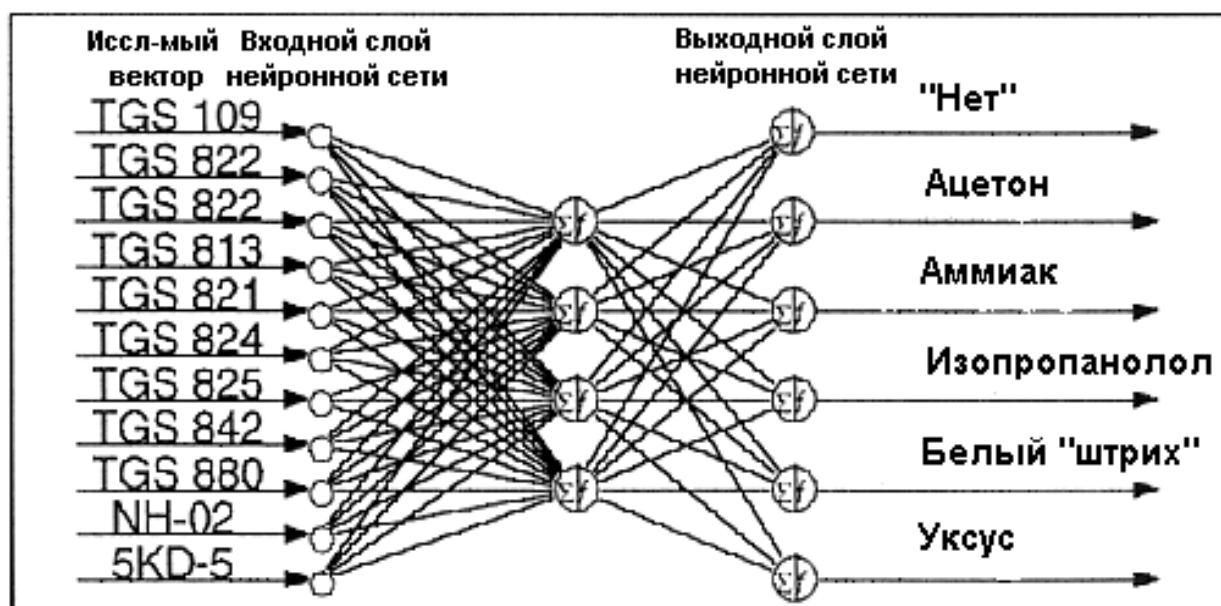


Рис.6.1. Топология нейронной сети прототипа искусственного носа

Таблица 6.4. Первичная обучающая пара

Исследуемый вектор, составленный по показаниям сенсоров и формирующий вход нейронной сети	Выходы нейронной сети
{1; 0,05; 0,1; 0,3; 0,07; 0,08; 0,2; 0,05; 0,2; 0,6; 0,8}	{0, 0, 0, 0, 0, 1}

Таблица 6.5. Набор вторичных обучающих пар

Исследуемые векторы, составленные по показаниям сенсоров и формирующие вход нейронной сети	Выходной вектор нейронной сети
{1; 0,05; 0,1; 0,3; 0,07; 0,08; 0,2; 0,05; 0,2; 0,6; 0,8}	«Нет»
{0,8; 0,4; 0,7; 0,6; 0,1; 0,5; 1,0; 0,75; 0,5; 0,7; 0,8}	Ацетон
{0,9; 0,2; 0,4; 0,5; 0,1; 0,7; 0,6; 0,5; 0,5; 0,7; 0,8}	Аммиак
{0,85; 0,7; 0,8; 0,65; 0,1; 0,4; 1,0; 0,7; 0,4; 0,6; 0,7}	Изопропанол
{0,9; 0,3; 0,3; 0,4; 0,04; 0,1; 0,5; 0,3; 0,2; 0,7; 0,8}	Белый «штрих»
{0,95; 0,18; 0,21; 0,3; 0,05; 0,1; 0,3; 0,2; 0,2; 0,5; 0,7}	Уксус

2. Существует два основных метода кодировки выхода ИНС. Первый – использование бинарного вектора: для каждой примеси только один единственный выход принимает значение «1». В этом случае количество выходов равно количеству примесей, определяемых системой. В другом случае все номинальные добавки пронумерованы и их числа перенесены в бинарную систему. В этом лабораторном исследовании используйте первый метод.
3. Исходные данные:
  - начальные экспериментальные данные, в виде показаний химических сенсоров, представлены в табл.6.4;
  - рекомендуемое число входов ИНС - 11;
  - рекомендуемое число выходов - 6;
  - начальное число нейронов скрытого (внутреннего) слоя - 4.
4. Вторичные экспериментальные данные, в виде показаний химических сенсоров, представлены в табл.6.5. Набор представительских выборок включает шесть обучающих пар, которые, естественно, подготовлены в формате первой обучающей пары (табл.6.4).

#### Определение топологии нейронной сети

Практические исследования показывают, что одного внутреннего (скрытого) слоя достаточно.

#### Обучение нейронной сети

Следует проводить при помощи алгоритма обратного распространения. Используйте *TRAJAN* или любой доступный программный продукт для моделирования ИНС (см. табл. П.1 приложений), который поддерживает данный алгоритм обучения и способен моделировать данную быстродействующую ИНС искусственного носа, применяемого для контроля атмосферы в воздушной среде промышленных, офисных и домашних помещениях и в других задачах.

#### Вопросы для проверки

1. В чем заключается принцип обучения ИНС типа «многослойный персептрон по алгоритму обратного распространения»?
2. В чем основное различие методов обучения ИНС «с учителем» и «без учителя»?
3. Назовите распространенные направления применения искусственного носа.

## 6.5. Лабораторная работа № 5.

### Прогнозирование

*Цель* – разработать и исследовать ИНС обратного распространения, предназначенную для прогнозирования временных серий, а также для анализа качества генератора случайных чисел.

#### Задание

1. Повторить соответствующий теоретический материал (гл. 1: п. 1.5; гл. 2: п. 2.1 – 2.3 и гл. 3).
2. Создать и обучить ИНС, предназначенную для анализа временных серий заданной размерности и отражающую структуру данных серий.
3. Осуществить прогноз значений будущих элементов временных серий.

#### Исследуемые временные серии

Объектом исследования являются временные серии, полученные с помощью генератора случайных чисел, формирующего равномерно распределенные числовые значения.

За основу рекомендуется взять сто элементов (чисел) временной серии, значения которых лежат в диапазоне от «0» до «99». Данную временную серию следует получить с помощью генерации случайных чисел любого программного продукта для математического моделирования (*MathCAD*, *Matlab* или др.), среды программирования на языке высокого уровня (*Delphi*, *C Builder* или др.) или же взять из табл.6.5.

Выясним, есть ли какая-либо закономерность в появлении элементов данных временных серий и тем самым определим качество генератора случайных чисел, который по идее должен обладать свойством некоррелированности значений числовых последовательностей.

#### Определение начальной структуры нейронной сети

Чтобы синтезировать оптимальную структуру, необходимо подготовить и обучить несколько ИНС и проверить качество выполнения требуемых операций. Ниже приведем основные этапы синтеза такой структуры.

Аналогично предыдущим случаям, воспользуйтесь ИНС типа «много-слойный персептрон».

1. Количество входов ИНС должно соответствовать ширине «окна скольжения» (п.1.5) – количеству элементов временных серий (в нашем случае –  $N$ ).

**Таблица 6.5. Числа в диапазоне от «0» до «100», полученные с использованием генератора случайных чисел Турбо-паскаль**

Номера чисел	Числа, полученные с использованием встроенной функции Random(100) Турбо-паскаль									
<b>1 – 10</b>	0	3	86	20	27	67	31	16	37	42
<b>11 – 20</b>	8	47	7	84	5	29	91	36	77	32
<b>21 – 30</b>	69	84	71	30	16	32	46	24	82	27
<b>31 – 40</b>	48	14	87	28	77	97	49	88	82	2
<b>41 – 50</b>	14	14	50	2	59	0	77	65	77	70
<b>51 – 60</b>	55	20	68	59	95	64	99	24	67	29
<b>61 – 70</b>	8	77	49	88	50	57	95	68	33	0
<b>71 – 80</b>	70	98	77	74	19	14	91	78	58	86
<b>81 – 90</b>	68	28	9	62	28	87	16	27	54	96
<b>91 – 100</b>	17	15	26	17	57	49	28	15	60	73

2. Число нейронов во внутренних (скрытых) слоях и число таких слов зависит от сложности задачи анализа или прогнозирования временных серий.
3. Выходной слой ИНС следует составить из одного нейрона, значение которого будет соответствовать прогнозируемому элементу временных серий.

#### Обучение нейронной сети

Для обучения ИНС подготовьте блок обучающих выборок следующим образом.

Выберите кадр (временное окно) из  $N+1$  числа элементов, идущих от конца к началу временной серии, где первые  $N$  элементов – формируют вектор входного слоя, а последний  $(N+1)$ -й – элемент выходного слоя ИНС. Следующую обучающую выборку получаем, «передвигаясь» временным окном на один элемент вправо и т.д.

Таким образом, анализируя временную серию, состоящую, например, из 100 элементов, можно подготовить  $100-N-1$  обучающих пар.

### Оптимизация структуры нейронной сети

Чтобы оценить качество прогнозирования, получите в соответствии с разделом «Исследуемые временные серии» еще 3 элемента, следующих за 100 элементами временной серии, которые уже использовались при обучении. Далее, запустите ИНС, предъявив ей 3 новых вектора, и сравните результаты работы ИНС с числами, синтезированными генератором.

Разницу между значениями, синтезированными генератором, и результатами работы нейронной сети используйте как качественный показатель ИНС. Уменьшите эту разницу, модифицируя структуру ИНС.

### Замечания

Подход, используемый в данной лабораторной работе для проверки качества генератора случайных чисел, наверное, не самый популярный и эффективный. Между тем, разрабатываемая в лабораторной работе ИНС является аналогичной ИНС, применяемым в интеллектуальных системах прогнозирования различного назначения, например, в системах прогноза знаков изменения биржевых индексов, системах прогноза цен, системах расчета оптимального использования ресурсов ит.д. (п.1.5).

## ЗАКЛЮЧЕНИЕ

В ближайшие десятилетия следует ожидать дальнейший технологический рост в области проектирования нейронных сетей и нейрокомпьютеров, и связанный с этим рост, рост интереса к ним со стороны специалистов. Развивается как фундаментальная наука, где искусственные нейронные сети являются объектом исследований, так и прикладная наука о технологиях применения нейронных сетей при решении разнообразных задач во многих прикладных отраслях. За последние годы уже открыто немало новых возможностей нейронных сетей, а работы в области нейротехнологий зачастую становятся важным вкладом в промышленность и науку, имеют большое экономическое значение.

В учебном пособии весьма подробно рассматриваются теоретические вопросы: основные направления применения ИНС, основные понятия ИНС, принципы построения и методы обучения ИНС «Многослойный персептрон» и ИНС Кохонена. Теоретический материал соответствует современному состоянию теории ИНС, а содержание – государственному образовательному стандарту высшего профессионального образования по дисциплине «Нейронные вычислительные сети».

Учебное пособие включает лабораторный практикум, который обеспечивает методическую помощь студентам при моделировании ИНС типа «Многослойный персептрон» в среде нейросимулятора *TRAJAN* или ей подобной. Практикум включает 5 лабораторных работ, задания которых дифференцированы и предусматривают индивидуальные варианты.

Рекомендуется студентам и аспирантам, обучающимся по направлению «Информатика и вычислительная техника».

## ВОПРОСЫ ДЛЯ САМОПРОВЕРКИ

1. В каких случаях применение искусственных нейронных сетей актуально?
2. Назовите типовые задачи и основные области применения искусственных нейронных сетей.
3. Какие задачи решают аксоны, дендриды и синапсы?
4. Определите искусственный нейрон и его основные свойства.
5. Какую задачу решает активационная функция в искусственном нейроне?
6. Назовите виды активационных функций.
7. Дайте определение многослойного персептрона.
8. К какому типу алгоритмов обучения относится алгоритм обратного распространения, и в чем отличительные черты этого алгоритма.
9. Перечислите требования, предъявляемые к обучающим выборкам и их подготовке.
10. В чем заключается задача классификации?
11. Приведите примеры образов и процессов, которые способны классифицировать искусственные нейронные сети.
12. Сформулируйте этапы решения задачи прогнозирования с использованием искусственных нейронных сетей.
13. Что означает термин «глубина прогнозирования»?
14. В чем различие методов прогнозирования с учетом «сезонной» составляющей и без учета.
15. От каких факторов зависит достоверность и точность прогноза с использованием искусственных нейронных сетей?
16. Для решения каких задач применяются самоорганизующиеся карты Кохонена?
17. Каков смысл параметров, передаваемых функции обучения Кохонена?
18. Что представляет собой слой Кохонена?
19. Что означают термин «эвклидово расстояние»?
20. Какие задачи можно решить с помощью нейросимуляторов?



## ЛИТЕРАТУРА

1. *Галушкин А.И.* Некоторые исторические аспекты развития элементной базы вычислительных систем с массовым параллелизмом (80- и 90-ые гг) // *Нейрокомпьютер*, № 1, 2000. с. 68-82.
2. *Горбань А., Россиев Д.* Нейронные сети на персональном компьютере. Новосибирск: Наука, 1996. 276 с.
3. *Кирсанов Э.Ю.* Цифровые нейрокомпьютеры: Архитектура и схемотехника / Казань: Казанский Госуниверситет, 1995. 131 с.
4. *Круг П.Г.* Нейронные сети и нейрокомпьютеры: учебное пособие. М.: Изд-во МЭИ, 2002. 176 с.
5. *Круг П.Г.* Сигнальные процессоры и нейрокомпьютеры. М.: Изд-во МЭИ, 2002. 256 с.
6. *Минский М., Пейперт С.* Перцептроны. М.: Мир, 1971.
7. Нейросетевые и нейрокомпьютерные технологии: Метод. указания / Алексеев А.В., Круг П.Г., Петров О.М. М.: МГАПИ, 1999. 20 с.
8. *Уоссермен Ф.* Нейрокомпьютерная техника. М.: Мир, 1992.
9. *Хехт-Нильсен Р.* Нейрокомпьютинг: история, состояние, перспективы // *Открытые системы*. № 4. 1998.
10. Нейрокомпьютеры: архитектура и схемотехника / Шахнов В.А., Власов А.И., Кузнецов А.С., Поляков Ю.А. М.: Изд-во Машин-ние, 2000. 64 с.
11. *Alekseev A., Krug P., Shahidur R.* The Neural Networks. Teaching Edition. Moscow. Publishing House of MPEI, 2000. 64 p.
12. <http://www.informatik.uni-stuttgart.de/ipvr/bv/projekte/snns>
13. <http://www.trajan-software.co.uk>
14. <http://www.basegroup.ru/tasks/forecast.htm>

## ПРИЛОЖЕНИЯ

### Таблица П.1. Программные продукты моделирования нейронных сетей

[illegible]

Продолжение табл. П.1

1	2	3	4	5	6	7
Delta	Artificial Intelligence Group, Франция (Департамент компьютерных наук)	HP-UX Sun OS 4.1	Многоуровневый перцептрон, обратное распространение, сети Джордана, карты Кохонена	GUI для X-Windows	Бес- платно	Демо-версия
<a href="http://www-inf.enst.fr/~milc/dnns/dnns.us.html">http://www-inf.enst.fr/~milc/dnns/dnns.us.html</a>						
X-Sim	ИС, Испания (Мадрид)	Unix Linux	Многослойный перцептрон, обратное распространение, карты Кохонена	Режим ко- мандной строки для ОС типа Unix, GUI для X-Windows	Бес- платно	Демо-версия
<a href="http://www.iic.uam.es/xsim/Welcome.html">http://www.iic.uam.es/xsim/Welcome.html</a>						
Brain Wave	Университет Куинсланд, США	Любая	Многослойный перцептрон, обратное распространение, сети Хеббiana, карты Кохонена	Internet- броузер с поддержкой Java	Бес- платно	Реализован в виде Java-апплета - может работать с любой опера- ционной системы
<a href="http://www2.psy.uq.edu.au/~brainwav/">http://www2.psy.uq.edu.au/~brainwav/</a>						
VieNet 2	Австрийский институт ис- следования проблем ис- кусственного интеллекта	DOS Win Unix Linux	Многослойный перцептрон, обратное распространение, сети Джордана, сети Эльмана, карты Кохонена, Ассоциативная память	Формирует- ся пользова- телем	Бес- платно	Распространяется в форме исходных кодов, что позволяет активно использовать его для написания собственных программ
<a href="http://www.ai.univie.ac.at/oefai/nn/tool.html">www.ai.univie.ac.at/oefai/nn/tool.html</a>						
NeuroWindows	НейроПроект, Россия	Win	Обратное распространение, ассоциативная память, карты Кохонена	Формирует- ся пользова- телем	450\$	Библиотека динамиче- ской компоновки Visual Basic, C++ и Delphi
<a href="http://www.neuroproject.ru/">http://www.neuroproject.ru/</a>						

Продолжение табл. П.1

1	2	3	4	5	6	7
Aspirin/ MIGRAINES	Mitre Corp.	Unix	Обратное распространение	GUI для X-Windows	Бес- платно	Сохраняет веса и век- тор узлов нейронной сети на диске в до- ступном формате
Atree	Билл Армстронг, Университет г. Альберта, США	Dos, Unix	Адаптивные логические деревья	Режим командной строки для Unix-OC, окна для DOS	Бес- платно	Демо-версия
<a href="mailto:alnl@cs.ualberta.ca">alnl@cs.ualberta.ca</a>						
Cnaps	Adaptive Solutions Inc.	SunOS	Обратное распространение, карты Кохонена (одномерные и двумерные), LVQ2 и частотно-чувствительное конкурентное обучение	GUI для X-Windows	68,75 \$	Производительность при обучении по алго- ритму обратного рас- пространения 1 бил- лион CUPS
ICSIM	Международный инсти- тут компьютерных наук, Беркли, Калифорния, США	Unix	Предопределенные сети	Shell, GUI для X-Windows	Бес- платно	Демо-версия
Neural Shell	Лаборатория SPANN, Департамент инженерной энергетики, Университет Огайо, США	Unix	Сети Хопфильда, сети Хемминга, обратное распространение, карты Кохонена, адаптивное медленное обратное распространение, частотно-чувствительное конкурентное обучение	Режим командной строки, GUI для X- Windows и SUNTOOLS	Бес- платно	Демо-версия
<a href="ftp://ftp.quanta.eng.ohio-state.edu/">ftp://ftp.quanta.eng.ohio-state.edu/</a>						
Neuron	Университет Дьюка, США	Unix	Трехмерная реконструированная пирамидальная ячейка, диффузия	GUI для X- Windows	Бес- платно	Демо-версия

Окончание табл. П.1

1	2	3	4	5	6	7
Sankom	Дортмундский университет, Германия	Unix	Карты Кохонена	Режим командной строки Shell	Бесплатно	Демо-версия
SOMPAK	SOM, Лаборатория компьютерных и информационных наук, Хельсинкский университет технологий	Unix, DOS	Самоорганизующиеся карты	Формируется пользователем	Бесплатно	Демо-версия
Xerion	Университет Торонто, Департамент компьютерных наук, США	Unix	Обратное распространение, рекуррентное обратное распространение, машина Больцмана, теория среднего поля, манипуляция свободной энергией, жесткое и мягкое конкурентное обучение, карты Кохонена	GUI для X-Windows	Бесплатно	Демо-версия
<a href="ftp://ftp.cs.toronto.edu/pub/xerion">ftp://ftp.cs.toronto.edu/pub/xerion</a>						
NETS	COSMIC, Университет Джорджии, США	DOS, UNIX	Обратное распространение	Режим командной строки	-----	Демо-версия
<a href="mailto:service@cossack.cosmic.uga.edu">service@cossack.cosmic.uga.edu</a>						

Таблица П2. Варианты заданий к лабораторной работе №2

Номер варианта	Строки, используемые для обучения			Строки для проверки обучения	
	1-ая	2-ая	3-ая	1-ая	2-ая
1	1	2	3	4	6
2	2	3	4	6	7
3	3	4	6	7	8
4	4	6	7	8	9
5	6	7	8	1	9
6	7	8	9	1	2
7	1	8	9	2	3
8	1	2	9	3	4
9	1	2	4	6	7
10	1	2	6	7	8
11	1	2	7	8	9
12	1	2	8	3	4
13	1	3	4	6	7
14	1	3	6	7	8
15	1	3	7	8	9
16	1	3	8	2	9
17	1	3	9	2	4
18	1	4	6	2	5
19	1	4	7	2	6
20	1	4	8	2	7
21	1	4	9	2	8
22	1	6	7	3	4
23	1	6	8	3	7
24	1	6	9	3	8
25	1	7	8	4	6
26	1	7	9	4	8
27	2	3	6	4	7
28	2	3	7	4	8
29	2	3	8	4	9
30	2	3	9	6	7
31	2	4	6	7	8
32	2	4	7	3	6
33	2	4	8	3	9
34	2	4	9	1	3
35	2	6	7	1	5
36	2	6	8	1	6
37	2	6	9	1	7
38	2	7	8	1	9
39	2	7	9	1	8
40	2	8	9	3	5
41	3	4	6	2	7
42	3	4	7	2	8
43	3	4	8	2	9
44	3	4	9	2	6
45	3	6	7	4	8
46	3	6	8	4	9
47	3	6	9	1	4
48	3	7	8	4	6
49	3	7	9	6	8
50	3	8	9	6	7
51	4	6	7	8	9
52	4	6	8	7	9
53	4	6	9	2	3
54	4	7	8	1	3
55	4	7	9	1	5
56	4	8	9	2	6
57	6	7	8	4	9
58	6	7	9	4	8

Таблица ПЗ. Варианты заданий к лабораторной работе №3

Номер варианта	Символы для распознавания			
	1	2	3	4
1	X	I	O	П
2	X	I	O	С
3	X	I	O	Y
4	X	I	O	U
5	X	I	O	L
6	X	I	O	J
7	X	I	С	Y
8	X	I	С	U
9	X	I	С	L
10	X	I	С	J
11	X	I	С	П
12	X	I	П	Y
13	X	I	П	U
14	X	I	П	L
15	X	I	П	J
16	X	I	Y	U
17	X	I	Y	L
18	X	I	Y	J
19	X	I	U	L
20	X	I	U	J
21	X	I	L	J
22	X	O	С	Y
23	X	O	С	U
24	X	O	С	L
25	X	O	С	J
26	X	O	С	П
27	X	O	П	Y
28	X	O	П	U
29	X	O	П	L
30	X	O	П	J
31	X	O	Y	U
32	X	O	Y	L
33	X	O	Y	J
34	X	O	U	L
35	X	O	U	J
36	X	O	L	J
37	X	С	П	Y
38	X	С	П	U
39	X	С	П	L
40	X	С	П	J
41	X	С	Y	U
42	X	С	Y	L
43	X	С	Y	J
44	X	С	U	L
45	X	С	U	J
46	X	С	L	J
47	X	П	Y	U
48	X	П	Y	L
49	X	П	Y	J
50	X	П	U	L
51	X	П	U	J
52	X	П	L	J
53	X	Y	U	L
54	X	Y	U	J
55	X	Y	L	J
56	X	U	L	J
57	Y	U	L	С
58	Y	U	J	O

Таблица П4. Варианты заданий к лабораторной работе №5

Номер варианта	Ширина скольжения»	Число обучающих выборок	Число проверочных выборок
1	7	17	2
2	8	16	3
3	9	15	4
4	10	14	5
5	11	13	2
6	12	12	3
7	13	11	4
8	14	10	5
9	15	9	2
10	16	8	3
11	17	7	4
12	7	18	5
13	8	17	2
14	9	16	3
15	10	15	4
16	11	14	5
17	12	13	2
18	13	12	3
19	14	11	4
20	15	10	5
21	16	9	2
22	17	8	3
23	7	16	4
24	8	15	5
25	9	14	2
26	10	13	3
27	11	12	4
28	12	11	5
29	13	10	2
30	14	9	3
31	15	8	4
32	16	7	5
33	17	6	2
34	7	19	3
35	8	18	4
36	9	17	5
37	10	16	2
38	11	15	3
39	12	14	4
40	13	13	5
41	14	12	2
42	15	11	3
43	16	10	4
44	17	9	5
45	7	15	2
46	8	14	3
47	9	13	4
48	10	12	5
49	11	11	2
50	12	10	3
51	13	9	4
52	14	8	5
53	15	7	2
54	16	6	3
55	17	5	4
56	11	16	5
57	12	15	2
58	13	14	3



**P.G.Krug.**

**The Modeling of the Artificial Neural Networks. Moscow. MGUPI, 2009 – 96p.**

The basics of the Artificial Neural Networks, modern Neural Networks Simulators and the examples of Modeling of Multilayer Perceptrons are considered.

The book contains the practical course based on TRAJAN Software Simulator.

The book reflects the 10-years experience of teaching students in Artificial Neural Networks in the Moscow State University of Instrument Engineering and Informatics and in the Moscow Power Engineering Institute (Technical University).

For students and post-graduate dealing with Computer Science.

Учебное издание

Круг Петр Германович

**МОДЕЛИРОВАНИЕ ИСКУССТВЕННЫХ  
НЕЙРОННЫХ СЕТЕЙ**

Учебное пособие по курсу «Нейронные вычислительные сети»

ЛР № 020418 от 08 октября 1997 г.

Подписано к печати \_\_\_\_\_.2009 г. Формат 60х84. 1/16. Объем 6,0 п.л. Тираж 100 экз.

Заказ №\_\_\_\_\_.

***Московский государственный университет  
приборостроения и информатики  
107996, Москва, ул. Стромывнка, 20***