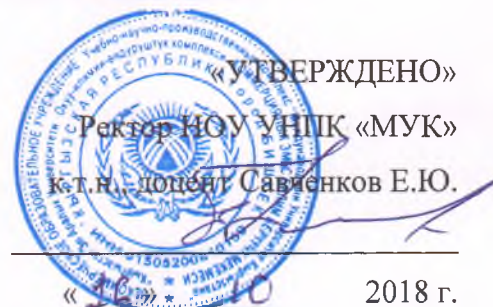


УЧЕБНО-НАУЧНО-ПРОИЗВОДСТВЕННЫЙ КОМПЛЕКС
«МЕЖДУНАРОДНЫЙ УНИВЕРСИТЕТ КЫРГЫЗСТАНА»



«УТВЕРЖДЕНО»
Ректор ИОУ УНПК «МУК»
к.т.н., доцент Савченко Е.Ю.

«16» 10 2018 г.

БАКАЛАВРИАТ

Кафедра «Компьютерные информационные системы и управление»

Учебно-методический комплекс дисциплины

Нейросетевые технологии

Направление: 710100 «Информатика и вычислительная техника»

Профиль: Компьютерные информационные системы в бизнесе

Академическая степень - бакалавр

Форма обучения (очная)

График проведения модулей 8-семестр

неделя	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
лекц. зан.	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
прак./лаб. зан.	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2

«РАССМОТРЕНО»

Протокол заседания кафедры

«КИСиУ»

№ 2 от 16.10.2018

Зав. кафедрой д.т.н., проф. Миркин Е.Л.

«СОГЛАСОВАНО»

Проректор по академ.
вопросам
проф. Мадалиев М.М.

Составитель

Директор Научной библиотеки

к.т.н., доцент
Савченко Е.Ю.

Асанова Ж.Ш.

БИШКЕК 2018

ОГЛАВЛЕНИЕ

АННОТАЦИЯ	4
УЧЕБНО-МЕТОДИЧЕСКИЙ КОМПЛЕКС ДИСЦИПЛИНЫ (МОДУЛЕЙ)	5
1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА	5
1.1. Миссия и стратегия	5
1.2. Цель и задачи дисциплины	5
1.3. Формируемые компетенции, а также перечень планируемых результатов обучения по дисциплине	5
1.4. Место дисциплины (модулей) в структуре ООП ВПО	7
2. СТРУКТУРА ДИСЦИПЛИНЫ	7
3. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ	9
4. КОНСПЕКТ ЛЕКЦИЙ	11
Лекция Введение в нейронные сети	11
Лекция Биологический нейрон и его математическая модель	12
Лекция Классификация нейронных сетей и их свойства. Представление знаний в нейронных сетях	15
Лекция Представление нейронных сетей с помощью ориентированных графов	17
Лекция Обучение нейронных сетей. Парадигмы обучения	19
Лекция Обучение, основанное на коррекции ошибок	20
Лекция GUI-интерфейс для пакета Neural Networks Toolbox программной среды Matlab 6.5. Назначение и обзор	22
Лекция Персептрон	34
Лекция Линейная разделимость и персептронная представляемость	36
Лекция Многослойная нейронная сеть прямого распространения	37
Лекция Алгоритм обучения сети обратного распространения	40
Лекция Последовательный и пакетный режимы обучения	41
Лекция Классификация и кластеризация	43
5. ИНФОРМАЦИОННЫЕ И ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ	46
6. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ТЕКУЩЕГО, РУБЕЖНОГО И	

ИТОГОВОГО КОНТРОЛЕЙ ПО ИТОГАМ ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЕЙ)	49
6.1. Перечень компетенций с указанием этапов их формирования в процессе освоения дисциплины	49
6.2. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности	51
6.3. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания	53
6.4. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности.	54
Контрольные вопросы	54
Тематика рефератов	55
Тест	56
Контрольная работа	61
Контрольная работа	62
Самостоятельная работа студентов	64
7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ	65
7.1. Список источников и литературы	65
7.2. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимый для освоения дисциплины (модулей)	65
8. ПЕРЕЧЕНЬ УЧЕБНО-МЕТОДИЧЕСКОГО ОБЕСПЕЧЕНИЯ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ	66
8.1. Планы практических (семинарских) и лабораторных занятий. Методические указания по организации и проведению	66
8.2. Методические указания для обучающихся по освоению дисциплины (модулей)	69
8.3. Методические рекомендации по подготовке отчетов по лабораторным работам	70
8.4. 8.4. Иные материалы	70
9. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ	70
10. ГЛОССАРИЙ	71
11. ПРИЛОЖЕНИЯ	72
12.	73

АННОТАЦИЯ

Дисциплина «Нейронные сети» представляет собой специализированный курс, который является одним из важнейших при подготовке специалистов в области информационных технологий. Данный курс дает полное представление специалистам об областях применения нейронных сетей, о существующих нейросетевых структурах. Приводится детальный обзор и описание важнейших методов обучения нейронных сетей различной структуры. Данный курс состоит из двух модулей, итоговый контроль в виде экзамена.

УЧЕБНО-МЕТОДИЧЕСКИЙ КОМПЛЕКС ДИСЦИПЛИНЫ (МОДУЛЕЙ)

1. Пояснительная записка

1.1. Миссия и стратегия

Миссия НОУ УНПК "МУК" – подготовка международно - признанных, свободно мыслящих специалистов, открытых для перемен и способных трансформировать знания в ценности на благо развития общества.

Видение НОУ УНПК «МУК»- создание динамичного и креативного университета с инновационными научно-образовательными программами и с современной инфраструктурой, способствующие достижению академических и профессиональных целей.

Стратегии развития - модернизация образовательной деятельности университета – совершенствование образовательного процесса в соответствии с требованиями Болонского процесса.

1.2. Цель и задачи дисциплины

Цель дисциплины: подготовить выпускника в области теоретических основ и принципов устройства систем с искусственным интеллектом, введение в проблематику неформализованных задач и методы их решения, а также приобретение ими практических навыков в построении информационных систем с искусственным интеллектом с помощью нейросетевых технологий.

Задачи дисциплины:

- формирование у студентов знаний об основах нейросетевых технологий, истории и предпосылках развития этой области научных знаний;
- ознакомление с принципами представления знаний;
- изучения принципов устройства и функционирования искусственных нейронных сетей, экспертных систем;
- ознакомление с проблемой распознавания образов, кластеризации и классификации данных, основными алгоритмами решения задач в области данной проблемы;
- получение студентами практических навыков решения задач в области искусственного интеллекта: программирование задач в Neural Network Toolbox Matlab;
- подготовить специалистов к использованию нейросетевых технологий в научно-исследовательской деятельности.

1.3. Формируемые компетенции, а также перечень планируемых результатов обучения по дисциплине

Дисциплина «Нейросетевые технологии» направлена на формирование следующих компетенций:

- общенаучными (ОК-4):
- способен понимать и применять традиционные и инновационные идеи, находить подходы к их реализации и участвовать в работе над проектами, используя базовые методы исследовательской деятельности (ОК-4);
- профессиональными (ПК-3, ПК-8):
 - разрабатывать интерфейсы «человек - электронно-вычислительная машина» (ПК-3);
- способен готовить конспекты и проводить занятия по обучению сотрудников применению программно-методических комплексов, используемых на предприятии (ПК-8).

○

В результате освоения дисциплины обучающийся должен демонстрировать следующие результаты образования:

1. Знать:

- принципы постановки задач для решения с помощью нейронных сетей;

(ОК-4)

- методы представления данных для обучения и использования нейронных сетей; (ПК-3)
- методы обучения нейронных сетей и оценки качества обучения нейронной сети; (ПК-8)
- этапы решения задач с помощью нейронных сетей. (ПК-8)

2. Уметь:

- ориентироваться в различных типах прикладных систем, основанных на использовании нейронных сетей (ОК-4)
- ориентироваться в различных методах представления, данных для обучения нейронной сети (ПК-3).

3. Владеть:

- навыком выбирать и ставить задачу для решения ее нейронной сетью (ПК-8);
- навыком выбирать модель нейронной сети для решения задачи (ПК-8).

Результаты обучения могут быть представлены в виде таблицы

Например:

Коды компетенции	Содержание компетенций	Перечень планируемых результатов обучения по дисциплине
(ОК-4)	<i>общенаучные</i>	<i>Знать:</i> различные модели нейронных сетей, их особенности; принципы постановки задач для решения с помощью нейронных сетей; <i>Уметь:</i> ориентироваться в различных типах прикладных систем, основанных на использовании нейронных сетей <i>Владеть:</i> принципами устройства и функционирования искусственных нейронных сетей, экспертных систем;
(ПК-8)	<i>инструментальные</i>	<i>Знать:</i> существующие прикладные системы, основанные на применении нейронных сетей; <i>Уметь:</i> -планировать, организовывать и проводить научные исследования в области нейросетевых технологий; <i>Владеть:</i> практическими навыками решения задач в области искусственного интеллекта: программирование задач в Neural Network Toolbox Matlab;
(ПК-3)	<i>профессиональные</i>	<i>Знать:</i> различные модели нейронных сетей, их особенности; <i>Уметь:</i> -планировать, организовывать и проводить научные исследования;

		<i>Владеть - навыками самостоятельной научно-исследовательской и научно- педагогической деятельности.</i>
--	--	---

1.4. Место дисциплины (модулей) в структуре ООП ВПО

Дисциплина (модуль) «Нейронные сети» является частью профессионального цикла (блока) дисциплин учебного плана по направлению подготовки 710100 «Информатика и вычислительная техника» подготовки бакалавров (специализации Компьютерные информационные системы).

Для освоения дисциплины (модулей) необходимы компетенции, сформированные в ходе изучения следующих дисциплин и прохождения практик: основные разделы математики, программирования.

В результате освоения дисциплины (модулей) формируются компетенции, необходимые для изучения следующих дисциплин и прохождения практик: интеллектуальные системы управления, нейросетевые технологии, написание магистерской диссертации.

2. Структура дисциплины

Общая трудоемкость дисциплины составляет 3 кредита, 90ч., в том числе аудиторная работа обучающихся с преподавателем 48ч., самостоятельная работа обучающихся 42 ч.

№ п/п	Раздел, Темы Дисциплины	Семестр	Неделя семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)				Формы текущего контроля успеваемости (по неделям семестра) Форма промежуточной аттестации (по семестрам)
				Лекции 16ч	Лаб. Зан. 32ч	СРС 28ч	СРСиП 14ч	
1	Введение в нейронные сети. Биологический нейрон и его математическая модель. Классификация нейронных сетей и их свойства.			2	2	1	0.5	

	Представление знаний в нейронных сетях.							
2	Обучение нейронных сетей. Парадигмы обучения. Обучение, основанное на коррекции ошибок.			2	2	1	0.5	
3	Проблемы практического использования искусственных нейронных сетей и их свойства			2	2	2	1	Сдача модуля
4	Многослойная нейронная сеть прямого распространения. Алгоритм обучения сети обратного распространения			2	2	2	1	
5	Линейные сети. Персептрон			2	2	2	1	Сдача модуля
6	Кластеризация и Классификация данных			1	2	2	1	
7	Сети с самоорганизацией на основе конкуренции			1	2	2	1	
8	Сеть с самоорганизацией корреляционного типа			1	2	2	1	
9	LVQ сети			1	2	2	1	
10	Рекуррентные сети			1	2	1	0.5	Сдача модуля
11	Вероятностная нейронная сеть			1	2	1	0.5	

3. Содержание дисциплины

№	Наименование раздела, темы дисциплины	Краткое содержание
1	Введение в нейронные сети. Биологический нейрон и его математическая модель. Классификация нейронных сетей и их свойства. Представление знаний в нейронных сетях.	Биологические основы функционирования нейрона. Первые модели нейронной сети. Прикладные возможности нейронных сетей. Определение искусственных нейронных сетей. Свойства биологических и искусственных нейронных сетей. Способы реализации нейросетей. Типы задач, решаемых нейронными сетями. Недостатки и ограничения нейронных сетей.
2	Обучение нейронных сетей. Парадигмы обучения. Обучение, основанное на коррекции ошибок.	Перцептрон. Сигмоидальный нейрон. Нейрон типа «адалайн». Парадигмы обучения. Обучение, основанное на коррекции ошибок.
3	Проблемы практического использования искусственных нейронных сетей и их свойства.	Предварительный подбор архитектуры сети. Подбор оптимальной архитектуры сети. Методы наращивания сети. Подбор обучающих выборок. Добавление шума в обучающие выборки. Распознавание и классификация образов. Нейронная сеть для сжатия данных. Идентификация динамических объектов.
4	Многослойная нейронная сеть прямого распространения. Алгоритм обучения сети обратного распространения.	Однослойная сеть. Многослойный перцептрон. Структура перцептронной сети. Алгоритм обратного распространения ошибки. Градиентные алгоритмы обучения сети: основные положения, алгоритм наискорейшего спуска, алгоритм переменной метрики, алгоритм Левенберга-Марквардта, алгоритм сопряженных градиентов. Подбор коэффициента обучения. Методы инициализации весов.
5	Линейные сети. Перцептрон.	Линейные сети. Основные понятия, область применения и алгоритмы обучения.
6	Кластеризация и Классификация данных.	Понятия кластеризации и классификации данных. Примеры задач.
7	Сети с самоорганизацией на основе конкуренции.	Нейронные сети встречного распространения. Сети Кохонена. Обучение слоя Кохонена. Примеры обучения сети Кохонена. Применение сети Кохонена для сжатия данных. Слой Гроссбера: обучение слоя Гроссбера, модификации.
8	Сеть с самоорганизацией корреляционного типа.	Энергетическая функция корреляционных сетей. Нейронные сети РСА: математическое введение, определение первого главного элемента, алгоритм определения множества главных компонентов.

9	LVQ сети.	LVQ нейронные. Предпосылки разработки. Область применения. Алгоритмы обучения нейронной сети.
10	Рекуррентные сети.	Сети с обратными связями. Послойность сети и матричное умножение. Расчет градиента квадратичной формы. Выбор начальной точки и длины шага. Сеть Хопфилда. Сеть Хемминга. Сеть ART –1
11	Вероятностная нейронная сеть.	Сети, использующие статистический подход. Метод «модельной закалки». Пример алгоритма минимизации функции. Машина Больцмана. Архитектура нейронной сети PNN. Пример модульной нейронной сети.

4. Конспект лекций

Лекция Введение в нейронные сети

Что такое нейронные сети.

Исследования по искусственным нейронным сетям связаны с тем, что способ обработки информации человеческим мозгом в корне отличается от методов, применяемых обычными цифровыми компьютерами. Мозг представляет собой чрезвычайно *сложный, нелинейный, параллельный* компьютер (систему обработки информации). Он обладает способностью организовывать свои структурные компоненты, называемые *нейронами* (neuron), так, чтобы они могли выполнять конкретные задачи (такие как распознавание образов, обработку сигналов органов чувств, моторные функции) во много раз быстрее, чем могут позволить самые быстродействующие современные компьютеры. Примером такой задачи обработки информации может служить обычное *зрение*. В функции зрительной системы входит создание *представления* окружающего мира в таком виде, который обеспечивает возможность *взаимодействия* с этим миром. Более точно мозг последовательно выполняет ряд задач распознавания (например, распознавание знакомого лица в незнакомом окружении). На это у него уходит около 100- 200 миллисекунд, в то время как выполнение аналогичных задач даже меньшей сложности на компьютере может занять несколько дней.

Нейронная сеть — это распределенный параллельный процессор, состоящий из элементарных единиц обработки информации, накапливающих экспериментальные знания и предоставляющих их для последующей обработки. Нейронная сеть сходна с мозгом с двух точек зрения.

- *Знания поступают в нейронную сеть из окружающей среды и используются в процессе обучения.*
- *Для накопления знаний применяются связи между нейронами, называемые синаптическими весами.*

Краткая историческая справка.

Термин «нейронные сети» сформировался к середине 50-х годов XX века. Основные результаты в этой области связаны с именами У. Маккалоха, Д. Хебба, Ф. Розенблатга, М. Минского, Дж. Хопфилда.

В 1943 г. У. Маккалох (W. McCulloch) и У. Питтс (W. Pitts) предложили модель нейрона и сформулировали основные положения теории функционирования головного мозга.

В 1949 г. Д. Хебб (D. Hebb) высказал идеи о характере соединений нейронов мозга и их взаимодействии и описал правила обучения нейронной сети.

В 1957 г. Ф. Розенблатг (F. Rosenblatt) разработал принципы организации и функционирования перцептронов, предложил вариант технической реализации первого в мире нейрокомпьютера Mark.

В 1969 г. была опубликована книга М. Минского (M. Minsky) и С. Пейперга (S. Papert) «Перцептроны», в которой доказывается принципиальная ограниченность возможностей перцептронов, что послужило причиной угасания интереса к искусственным нейронным сетям.

В начале 80-х годов происходит возобновление интереса к искусственным нейронным сетям, как следствие накопления новых знаний о деятельности мозга, а также значительного прогресса в области микроэлектроники и компьютерной техники.

В 1982-1985 гг. Дж. Хопфилд (J. Hopfield) предложил семейство оптимизирующих нейронных сетей, моделирующих ассоциативную память.

1987 г. послужил началом широкомасштабного финансирования разработок в области ИНС и нейрокомпьютеров (НК) в США, Японии и Западной Европе.

В 1989 г. разработки и исследования в области ИНС и НК ведутся практически всеми крупными электротехническими фирмами.

В 1997 г. годовой объем продаж на рынке ИИС и НК превысит 2 млрд. долларов, а ежегодный прирост составил 50%.

В 2000 г. благодаря переходу на субмикронные и нанотехнологии, а также успехам молекулярной и биомолекулярной технологии происходит переход к принципиально новым архитектурным и технологическим решениям по созданию нейрокомпьютеров.

Основные проблемы, решаемые искусственными нейронными сетями

Классификация образов. Задача состоит в указании принадлежности входного образа, представленного вектором признаков, одному или нескольким предварительно определенным классам. К известным приложениям относятся распознавание букв, распознавание речи, классификация сигнала электрокардиограммы, классификация клеток крови.

Кластеризация / категоризация. При решении задачи кластеризации, которая известна также как классификация образов без учителя, отсутствует обучающая выборка с образцами классов. Алгоритм кластеризации основан на подобии образов и размещает близкие образы в один кластер. Известны случаи применения кластеризации для извлечения знаний, сжатия данных и исследования свойств данных.

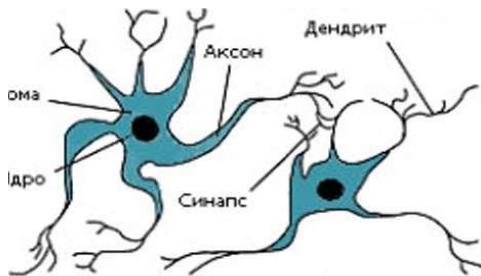
Аппроксимация функций. Предположим, что имеется обучающая выборка $((X_1, Y_1), (X_2, Y_2), \dots, (X_N, Y_N))$, которая генерируется неизвестной функцией, искаженной шумом. Задача аппроксимации состоит в нахождении оценки этой функции.

Предсказание/прогноз. Пусть заданы N дискретных отсчетов $\{y(t_1), y(t_2), \dots, y(t_k)\}$ в последовательные моменты времени t_1, t_2, \dots, t_k . Задача состоит в предсказании значения $y(t_{k+1})$ в момент t_{k+1} . Прогнозы имеют значительное влияние на принятие решений в бизнесе, науке и технике.

Оптимизация. Многочисленные проблемы в математике, статистике, технике, науке, медицине и экономике могут рассматриваться как проблемы оптимизации. Задачей оптимизации является нахождение решения, которое удовлетворяет системе ограничений и максимизирует или минимизирует целевую функцию.

Лекция Биологический нейрон и его математическая модель

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 кожи, ушей и глаз к мозгу, процессы мышления и управления действиями - все это реализовано в живом организме как передача электрических импульсов между нейронами. *Нейрон* (нервная клетка) является особой биологической клеткой, которая обрабатывает информацию (рис. 2.1.). Он состоит из *тела*, или *сомы*, и отростков нервных волокон двух типов - *дендритов*, по которым принимаются импульсы, и единственного *аксона*, по которому нейрон может передавать импульс. Тело нейрона включает *ядро*, которое содержит информацию о наследственных свойствах, и *плазму*, обладающую молекулярными средствами для производства необходимых нейрону материалов. Нейрон получает сигналы (импульсы) от аксонов других нейронов через дендриты (приемники) и передает сигналы, сгенерированные телом клетки, вдоль своего аксона (передатчика), который в конце разветвляется на волокна. На окончаниях этих волокон находятся специальные образования - *синапсы*, которые влияют на величину импульсов.



ис. 2.1. Взаимосвязь биологических нейронов

Искусственный нейрон

Каждый нейрон характеризуется своим текущим состоянием по аналогии с нервными клетками головного мозга, которые могут быть возбуждены или заторможены. Он обладает группой *синапсов* - однонаправленных входных связей, соединенных с выходами других нейронов, а также имеет *аксон* - выходную связь данного нейрона, с которой сигнал (возбуждения или торможения) поступает на синапсы следующих нейронов. Общий вид искусственного нейрона приведен на рисунке 2.2.

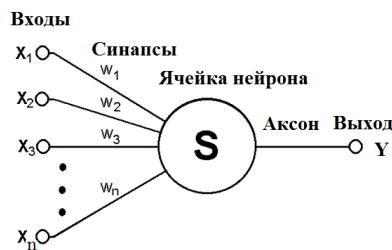


Рис.2.2. Искусственный нейрон

На рисунке 1.3 представлена модель нейрона, лежащего в основе искусственных нейронных сетей. В этой модели можно выделить три основных элемента.

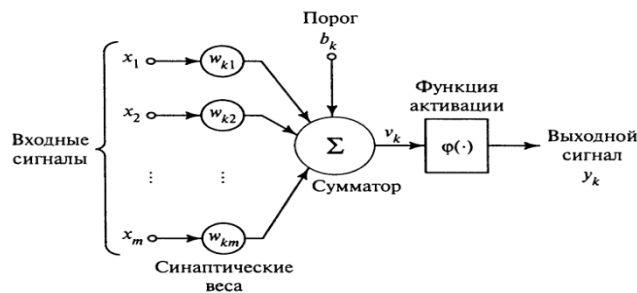


Рис. 2.3. Нелинейная модель нейрона

1. Набор синапсов или связей, каждый из которых характеризуется своим весом или силой. В частности сигнал x_j на входе синапса j , связанного с нейроном k , умножается на вес w_{kj} . Важно обратить внимание на то, в каком порядке указаны индексы синаптического веса w_{kj} . Первый индекс относится к рассматриваемому нейрону, второй – ко входному окончанию синапса, с которым связан данный вес. В отличие от синапсов головного мозга, синапсы в искусственных нейронных сетях могут принимать как положительные, так и отрицательные значения.
2. Сумматор складывает входные сигналы, взвешенные относительно соответствующих синапсов нейрона. Эту операцию можно описать как линейную комбинацию.

3. Функция активации ограничивает амплитуду выходного сигнала нейрона. Эта функция также называется функцией сжатия. Обычно нормированный диапазон амплитуд выхода нейрона лежит в интервале $[0,1]$, $[-1,1]$. В модель нейрона на рис. 3 включен пороговый элемент, который обозначен символом b_k . Эта величина отражает увеличение или уменьшение входного сигнала, подаваемого на функцию активации.

В математическом представлении функционирование нейрона k можно описать следующей парой уравнений:

$$u_k = \sum_j^m w_{kj} x_j, \quad (1)$$

$$y_k = \varphi(u_k + b_k) \quad (2)$$

Где x_1, x_2, \dots, x_m - входные сигналы; $w_{k1}, w_{k2}, \dots, w_{km}$ - синаптические веса нейрона k ; u_k - линейная комбинация входных воздействий; b_k - порог; $\varphi(\cdot)$ - функция активации; y_k - выходной сигнал нейрона. В модели, показанной на рис.3, постсинаптический потенциал вычисляется следующим образом:

$$v_k = u_k + b_k \quad (3)$$

В частности, в зависимости от того, какое значение принимает порог b_k , положительное или отрицательное, индуцированное локальное поле v_k нейрона v_k изменяется так, как показано на рисунке 1.4.

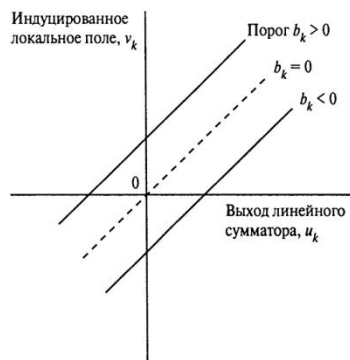


Рис. 2.4. Аффинное преобразование, вызванное наличием порога.

Обратите внимание, что в точке, где $u_k = 0$, $v_k = b_k$

Порог b_k является внешним параметром искусственного нейрона k . Тогда формулу (1) можно преобразовать к следующему виду:

$$u_k = \sum_{j=0}^m w_{kj} x_j, \quad (4)$$

$$y_k = \varphi(u_k) \quad (5)$$

В выражении 4 добавился новый синапс. Его входной сигнал равен:

$$x_0 = +1, \quad (6)$$

а его вес :

$$w_{k0} = b_k \quad (7)$$

Тогда модель нейрона будет выглядеть следующим образом рисунок 5.

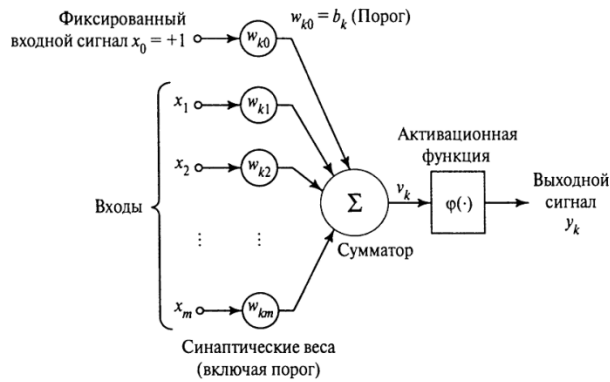


Рис.2.5 Нелинейная модель нейрона

Хотя внешне модели показанные на рис. 2.5 и рис.2.3. внешне не схожи, математически они эквивалентны.

Типы функций активации

Функции активации, представленные в формулах как $\varphi(\cdot)$, определяют выходной сигнал нейрона в зависимости от индуцировано локального поля v . В таблице приведены примеры активационных функций и их графики рис.1.6.

название	формула	область значений
пороговая	$\varphi(v) = \begin{cases} 0, & v < 0 \\ 1, & v \geq 0 \end{cases}$	[0, 1)
сигмовидная	$\varphi(v) = \frac{1}{1 + e^{-\alpha v}}$	(0, 1)

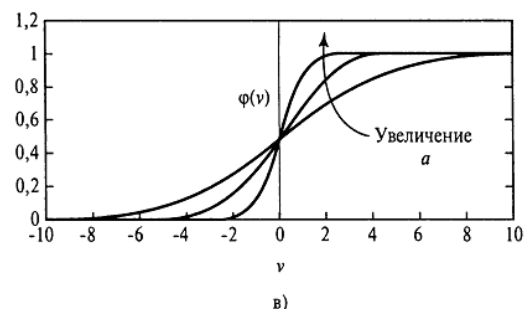
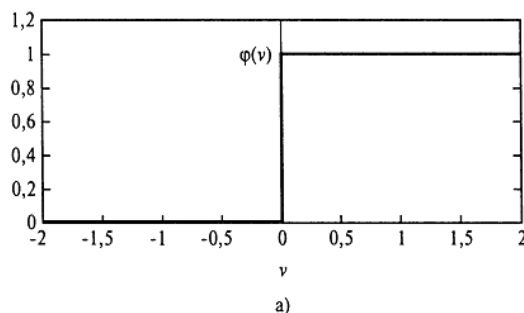


Рис.1.6. Виды активационных функций а) пороговая в) сигмовидальная

Лекция Классификация нейронных сетей и их свойства. Представление знаний в нейронных сетях

Классификация нейронных сетей и их свойства.

Нейронные сети (Neural Networks) - это модели биологических нейронных сетей мозга, в которых нейроны имитируются относительно простыми, часто однотипными, элементами (искусственными нейронами), топология соединений которых зависит от типа сети. В зависимости от функций, выполняемых нейронами в сети, можно выделить три их типа:

– *входные нейроны* – это нейроны, на которые подается входной вектор, кодирующий входное воздействие или образ внешней среды; в них обычно не осуществляется

вычислительных процедур, информация передается с входа на выход нейрона путем изменения его активации;

- *выходные нейроны* – это нейроны, выходные значения которых представляют выход сети;
- *промежуточные нейроны* – эти нейроны составляют основу искусственных нейронных сетей.

Классифицируя нейронные сети по топологии, можно выделить три основных типа таких сетей рис. 7:

1. *Полносвязные сети;*
2. *Многослойные или слоистые сети;*
3. *Слабосвязные сети.*

Полносвязные сети представляют собой Искусственные нейронные сети (ИНС), каждый нейрон которой передает свой выходной сигнал остальным нейронам, в том числе и самому себе. Все входные сигналы подаются всем нейронам. Выходными сигналами сети могут быть все или некоторые выходные сигналы нейронов после нескольких тактов функционирования сети

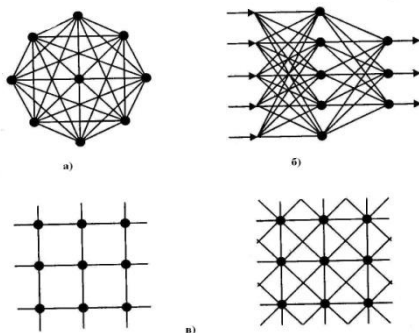


Рис 7. Архитектура нейронных сетей:

- а) -полносвязная сеть, б)-многослойная нейронная сеть с последовательными связями, в) – слабосвязные сети.

В *многослойных сетях* нейроны объединяются в слои. Слой содержит совокупность нейронов с едиными входными сигналами. Число нейронов в каждом слое может быть любым и никак заранее не связано с количеством нейронов в других слоях. Внешние входные сигналы подаются на входы нейронов первого слоя, а выходами сети являются выходные сигналы последнего слоя. Кроме входного и выходного слоев в многослойной нейронной сети есть один или несколько промежуточных скрытых слоев .

В свою очередь, среди многослойных сетей выделяют следующее типы:

- *сети без обратных связей* (классический вариант - многослойная сеть прямого распространения) рис.8;

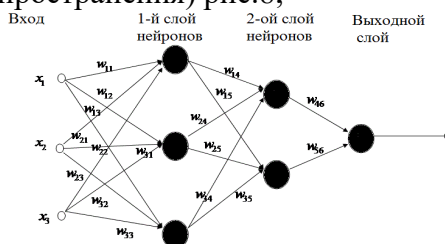


Рис. 8 Многослойная сеть прямого распространения.

- *сети с обратными связями*. Это сети, у которых информация с последующих слоев передается на предыдущие;

Представление знаний в нейронных сетях

Под знаниями понимается хранимая информация или модели, используемые человеком или машиной для интерпретации, предсказания и реакции на внешние события.

К вопросам *представления знаний* (knowledge representation) относятся следующие: какую информацию необходимо хранить и как эту информацию представить физически для ее последующего использования. Таким образом, исходя из самой природы знаний, способ их представления определяется поставленной целью. Относительно реальных приложений "интеллектуальных" систем можно утверждать, что успех решения зависит от хорошего представления знаний. Это касается и нейронных сетей, представляющих собой отдельный класс интеллектуальных систем. Форма представления входных сигналов может быть самой разной. Это приводит к тому, что разработка приемлемых нейросетевых решений становится творческим процессом.

Основной задачей нейронной сети является наилучшее обучение модели окружающего мира для решения поставленной задачи. Знания о мире включают два типа информации.

1. Известное состояние окружающего мира, представленное имеющимися в наличии достоверными фактами. Такая информация называется *априорной* (prior).
2. Наблюдения за окружающим миром (измерения), полученные с помощью сенсоров, адаптированных для конкретных условий, в которых должна функционировать данная нейронная сеть. Обычно такие измерения в значительной мере зашумлены, что потенциально может стать источником ошибок. В любом случае измерения, полученные таким способом, формируют множество информации, *примеры* из которого используются для обучения нейронной сети.

Примеры могут быть *маркированными* (labeled) и *немаркированными* (unlabeled). В маркированных примерах *входному сигналу* (input signal) соответствует *желаемый отклик* (desired response). Немаркированные примеры состоят из нескольких различных реализаций одного входного сигнала. В любом случае набор примеров, будь то маркированных или нет, представляет собой знания об интересующей предметной области, на основании которых и проводится обучение нейронной сети.

Множество пар сигналов вход-выход, каждая из которых состоит из входного сигнала и соответствующего ему желаемого выхода, называют *обучающими данными* (training data) или *обучающей выборкой* (training sample).

Лекция Представление нейронных сетей с помощью ориентированных графов

Блочные диаграммы (block diagram), представленные на рис. 1 и 2, в первой лекции обеспечивают функциональное описание различных элементов, из которых состоит модель искусственного нейрона. Внешний вид модели можно в значительной мере упростить, применив идею графов прохождения сигнала. Графы передачи сигнала с наборами правил были введены Мейсоном (Mason) в 1953 году для описания линейных сетей. *Граф передачи (или прохождения) сигнала* (signal-flow graph) — это сеть направленных *связей* (links) (или *ветвей* (branches)), соединяющих отдельные точки (*узлы*). С каждым узлом j связан *сигнал* x_j . Обычная направленная связь начинается в некотором узле j и заканчивается в другом узле k . С ней связана некоторая *передаточная функция* (transfer function), определяющая зависимость сигнала y_k узла k от сигнала x_j узла j . Прохождение сигнала по различным частям графа подчиняется трем основным правилам.

Правило 1. Направление прохождения сигнала вдоль каждой связи определяется направлением стрелки. При этом можно выделить два типа связей.

- *Синаптические связи* (synaptic link). Их поведение определяется *линейным* (linear) соотношением вход-выход. А именно, как показано на рисунке 9., сигнал узла x_j умножается на синаптический вес w_{kj} , в результате чего получается сигнал узла y_k

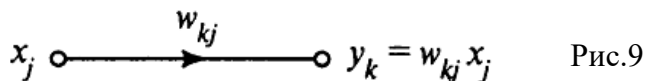


Рис.9

- *Активационные связи* (activation link). Их поведение определяется *нелинейным* (nonlinear) соотношением вход-выход. Этот вид связи показан на рис. 10, где $\varphi(\cdot)$ нелинейная функция активации.

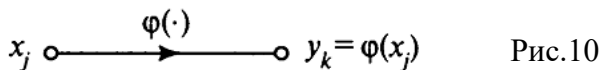
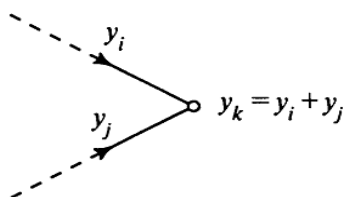


Рис.10

Правило 2. Сигнал узла равен алгебраической сумме сигналов, поступающих на его вход. На рис. 11 это правило проиллюстрировано для случая *синаптической сходимости* (synaptic convergence).

Рис.11



Правило 3. Сигнал данного узла передается по каждой исходящей связи без учета передаточных функций исходящих связей. Это правило проиллюстрировано на рис. 12 для *синаптической дивергенции* (synaptic divergence) или расходимости.

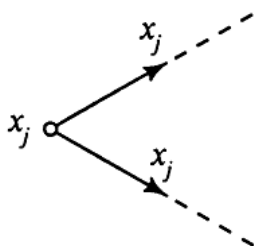


Рис.12

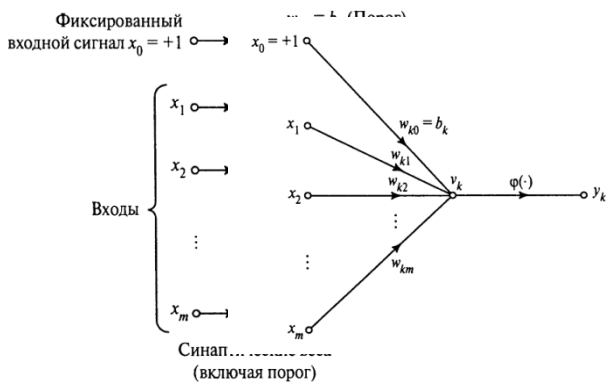


Рис. 13

графом рис.2.6.

Принимая в качестве модели нейрона граф прохождения сигнала, показанный на рис. 13, можно сформулировать еще одно определение нейронной сети.

Таким образом, рис.2 из лекции 1 можно представить с помощью ориентированного

Нейронная сеть — это направленный граф, состоящий из узлов, соединенных синаптическими и активационными связями, который характеризуется следующими четырьмя свойствами.

1. Каждый нейрон представляется множеством линейных синаптических связей, внешним порогом и, возможно, нелинейной связью активации. Порог, представляемый входной синаптической связью, считается равным +1.
2. Синаптические связи нейрона используются для взвешивания соответствующих входных сигналов.
3. Взвешенная сумма входных сигналов определяет индуцированное локальное поле каждого конкретного нейрона.
4. Активационные связи модифицируют индуцированное локальное поле нейрона, создавая выходной сигнал.

Лекция Обучение нейронных сетей. Парадигмы обучения

Самым важным свойством нейронных сетей является их способность обучаться на основе данных окружающей среды. Процесс обучения - это процесс, в котором свободные параметры нейронной сети настраиваются посредством моделирования среды, в которую сеть встроена. Тип обучения определяется способом надстройки этих параметров. Это определение процесса обучения предполагает следующую последовательность событий:

1. В нейронную сеть поступают стимулы из внешней среды.
2. В результате этого изменяются свободные параметры нейронной сети.
3. После изменения внутренней структуры нейронная сеть отвечает на возбуждения уже иным образом.

Данный список четких правил решения проблемы обучения называется алгоритмом обучения. В теории нейронных сетей выделяют две фундаментальные парадигмы обучения нейронных сетей: обучение с учителем и без учителя. И пять основных моделей обучения:

- на основе коррекции ошибок;
- с использованием памяти;
- Хеббовское обучение;
- конкурентное обучение;
- обучение Больцмана.

Для реализации перечисленных моделей обучения могут применяться различные алгоритмы, приведем некоторые из них:

алгоритмы локальной оптимизации с вычислением частных производных первого порядка:

- градиентный алгоритм;
- метод Ньютона;
- методы оптимизации с разреженными матрицами Гессе;
- метод Левенберга—Марквардта;
- поиск в случайном направлении;
- алгоритм обратного распространения ошибки.

Рассмотрим парадигму обучения нейронной сети *с учителем*.

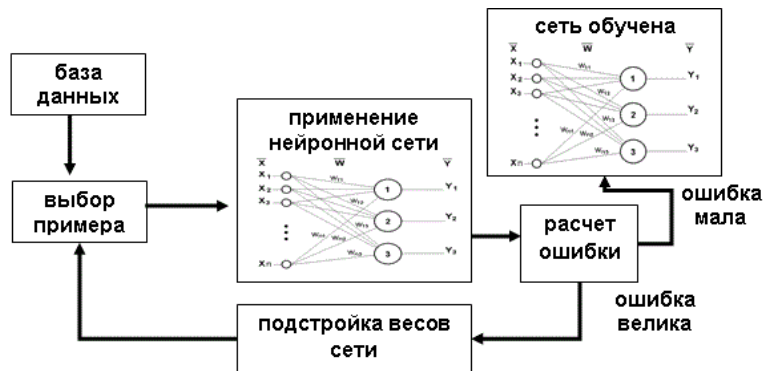


Рис.14 Процесс обучения

нейронной сети

Обобщенная схема обучения представлена на рисунке 14. Процесс обучения нейронной сети заключается в следующем. Формируется база данных примеров на основе сведений окружающей среды, из базы данных примеров выбирается один пример, который состоит из входного вектора данных и эталонного вектора. На вход нейронной сети подается входной вектор. В результате вычислений нейронной сети получаем отклик нейронной сети на данный пример, сравниваем выходное значение нейронной сети с эталоном, если ошибка мала, нейронная сеть считается обученной, иначе необходимо настраивать весовые коэффициенты сети (синапсы) согласно алгоритмов обучения. После окончания обучения учителя можно отключить и позволить нейронной сети работать со средой самостоятельно.

Рассмотрим парадигму обучения нейронной сети без учителя.

Обучение без учителя (или **обучение на основе самоорганизации**) осуществляется без вмешательства внешнего учителя или корректора, контролирующего процесс обучения. Существует лишь *независимая от задачи мера качества* (task-independent measure) представления, которому должна научиться нейронная сеть, и свободные параметры сети оптимизируются по отношению к этой мере. После обучения сети на статистические закономерности входного сигнала она способна формировать внутреннее представление кодируемых признаков входных данных и, таким образом, автоматически создавать новые классы.

Для обучения без учителя можно воспользоваться правилом конкурентного обучения. Например, можно использовать нейронную сеть, состоящую из двух слоев — входного и выходного. Входной слой получает доступные данные. Выходной слой состоит из нейронов, конкурирующих друг с другом за право отклика на признаки, содержащиеся во входных данных. В простейшем случае нейронная сеть действует по принципу "победитель получает все".

Эпоха обучения - это один просмотр всех примеров обучающей выборки с одновременной коррекцией весов сети (на этих примерах, в зависимости от правильности их решения сетью). Чтобы сеть обучилась, может потребоваться несколько (и даже несколько десятков, и даже несколько сотен,...) эпох. В течение эпохи не на каждом обучающем примере может происходить коррекция весов сети, т.к. если сеть уже обучилась правильно решать данный конкретный пример, то можно перестать требовать постоянных "улучшений" качества решения этого примера (поскольку эти улучшения уже будут несущественными) и позволить сети направить её внимание на другие примеры, которые пока ещё решаются с недостаточной точностью.

Лекция Обучение, основанное на коррекции ошибок

Для того чтобы проиллюстрировать правило обучения, рассмотрим простейший случай нейрона k единственного вычислительного узла выходного слоя нейронной сети прямого

распространения рис. а). Нейрон k работает под управлением вектора сигнала $x(n)$, производимого одним или несколькими скрытыми слоями нейронов, которые, в свою очередь, получают информацию из входного вектора (возбуждения), передаваемого начальным узлам (входному слою) нейронной сети. Под n подразумевается дискретное время, или, более конкретно, — номер шага итеративного процесса настройки синаптических весов нейрона k . Выходной сигнал нейрона k обозначается $y_k(n)$. Этот сигнал является единственным выходом нейронной сети. Он будет сравниваться с желаемым выходом, обозначенным $d_k(n)$. В результате получим сигнал ошибки $e_k(n)$. По определению

$$e_k(n) = d_k(n) - y_k(n). \quad (2.1)$$

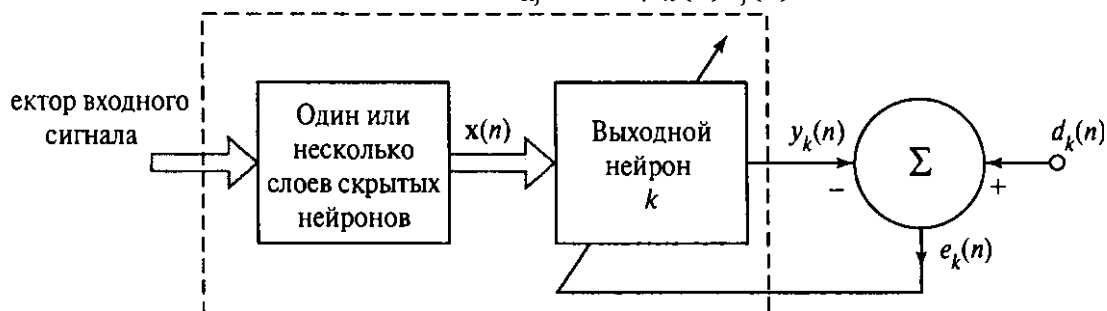
Сигнал ошибки инициализирует механизм управления (control mechanism), цель которого заключается в применении последовательности корректировок к синаптическим весам нейрона k . Эти изменения нацелены на пошаговое приближение выходного сигнала $y_k(n)$ к желаемому $d_k(n)$. Эта цель достигается за счет минимизации функции стоимости или индекса производительности $E(n)$, определяемой в терминах сигнала ошибки следующим образом:

$$E(n) = \frac{1}{2} e_k^2(n) \quad (2.2)$$

где $E(n)$ — текущее значение энергии ошибки. Пошаговая корректировка синаптических весов нейрона k продолжается до тех пор, пока система не достигнет устойчивого состояния (т.е. такого, при котором синаптические веса практически стабилизируются). В этой точке процесс обучения останавливается.

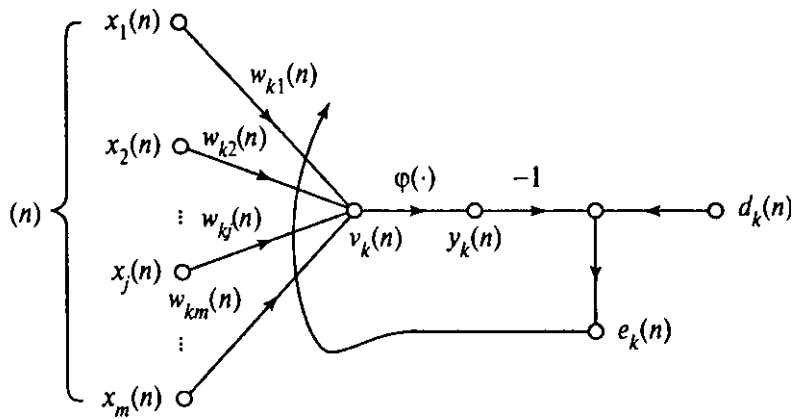
Процесс, описанный выше, называется обучением, основанном на коррекции ошибок (error-correction learning). Минимизация функции стоимости $E(n)$ выполняется по так называемому дельта-правилу, или правилу Видроу-Хоффа, названному так в честь его создателей. Обозначим $w_{kj}(n)$ текущее значение синаптического веса w_{kj} нейрона k , соответствующего элементу $x_j(n)$ вектора $x(n)$ на шаге дискретизации n . В соответствии с дельта-правилом изменение $\Delta w_{kj}(n)$, применяемое к синаптическому весу w_{kj} на этом шаге дискретизации, задается выражением

$$\Delta w_{kj}(n) = \eta e_k(n) x_j(n)$$



Иногослойная сеть прямого распространения

) Блочная диаграмма нейронной сети; показаны только нейроны выходного слоя



б) Граф передачи сигнала выходного нейрона
ис. 6.1. Обучение, основанное на коррекции ошибок

где η — некоторая положительная константа, определяющая *скорость обучения* и используемая при переходе от одного шага процесса к другому. Из формулы (2.3) видно, что эту константу естественно именовать *параметром скорости обучения*. Вербально дельта-правило можно определить следующим образом.

Корректировка, применяемая к синаптическому весу нейрона, пропорциональна произведению сигнала ошибки на входной сигнал, его вызвавший.

Определенное таким образом дельта-правило предполагает возможность *прямого измерения* сигнала ошибки. Для обеспечения такого измерения требуется поступление желаемого отклика от некоторого внешнего источника, непосредственно доступного для нейрона k . Другими словами, нейрон k должен быть *видимым* для внешнего мира (рис. 2.1, а). На этом рисунке видно, что обучение на основе коррекции ошибки по своей природе является *локальным*. Это прямо указывает на то, что корректировка синаптических весов по дельта-правилу может быть локализована в отдельном нейроне k . Вычислив величину изменения синаптического веса $\Delta w_{kj}(n)$, можно определить его новое значение для следующего шага дискретизации:

$$\Delta w_{kj}(n+1) = w_{kj}(n) + \Delta w_{kj}(n) \quad (2.4)$$

Таким образом, $\Delta w_{kj}(n+1)$ и $w_{kj}(n)$ можно рассматривать как *старое* и *новое* значения синаптического веса $w_{kj}(n)$.

Лекция GUI-интерфейс для пакета Neural Networks Toolbox программной среды Matlab 6.5. Назначение и обзор

Цель работы: изучение основных свойств и основ работы с GUI – интерфейсом пакета Neural Networks Toolbox в программной среде MatLab 6.5.

Теоретическая часть

Окно GUI-интерфейса пакета нейронных сетей

GUI-интерфейс – это специальное инструментальное средство организации диалога с пользователем. Это, например, пакет по нейронным сетям, в состав которого входит инструментальное средство NNTool. Этот графический интерфейс позволяет, не обращаясь к командному окну системы MATLAB, выполнять создание, обучение, моделирование, а также импорт и экспорт нейронных сетей и данных, используя только инструментальные возможности GUI-интерфейса. Однако такие инструменты наиболее эффективны лишь на начальной стадии работы с пакетом, поскольку имеют определенные ограничения. В частности, интерфейс NNTool допускает работу только с простейшими однослойными и двухслойными нейронными сетями, но при этом пользователь выигрывает во времени и эффективности решения прикладных задач [3].

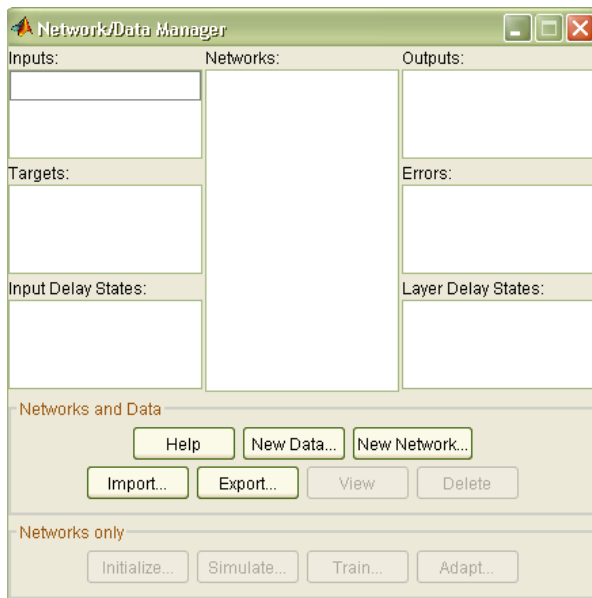


Рис. 1. Окно управления сетью/данными

Вызов GUI-интерфейса NNTool осуществляется командой `nntool` из командной строки.

После вызова появляется окно **Network/Data Manager (Управление сетью/данными)** – см. рис. 1.

Здесь имеются следующие области и кнопки:

Inputs – последовательность входов;

Targets – последовательность целей;

Input Daley States – начальные условия линии задержки входов;

Networks – список нейронных сетей;

Outputs – последовательность выходов;

Errors – последовательности ошибок сети;

Layer Delay States – начальные условия линии задержки слоя;

Help – кнопка вызова окна подсказки (рис. 2);

New Data... – кнопка вызова окна формирования данных (рис. 3);

New Network... – кнопка вызова окна создания новой нейронной сети (рис. 4);

Import... – кнопка вызова окна импорта или загрузки данных (рис. 5);

Export... – кнопка вызова окна экспорта или загрузки данных в файл (рис. 6).

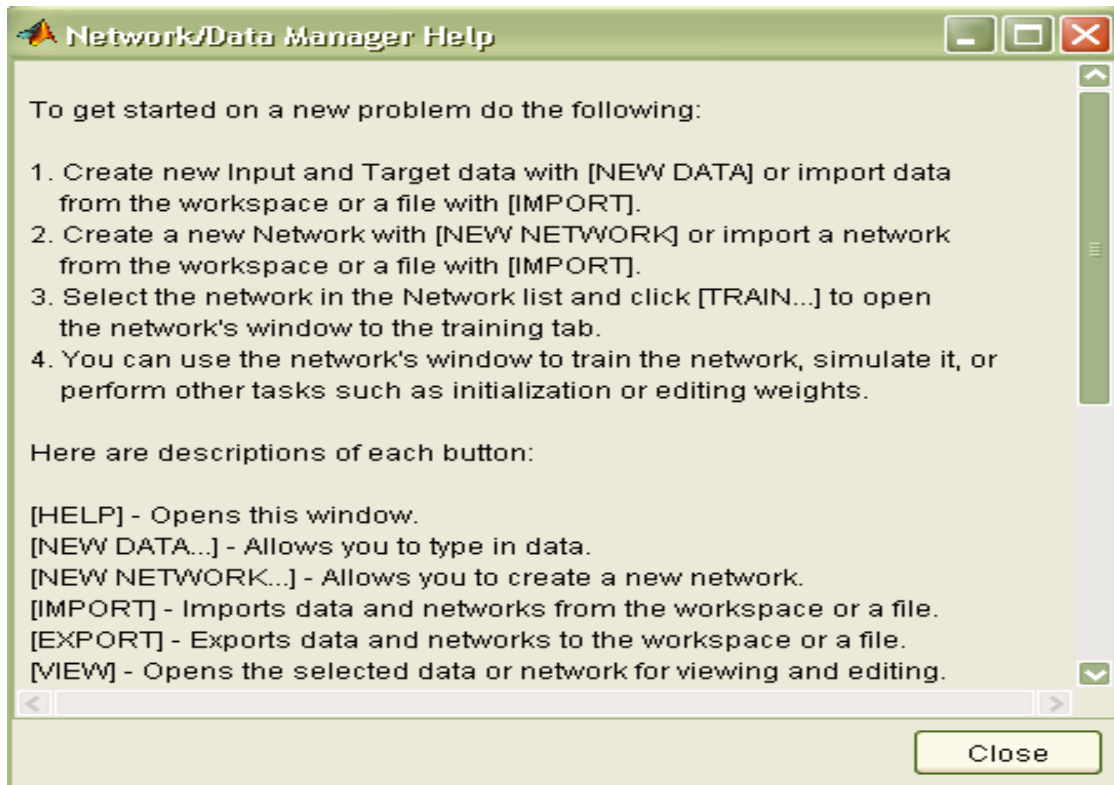


Рис. 2. Окно подсказки

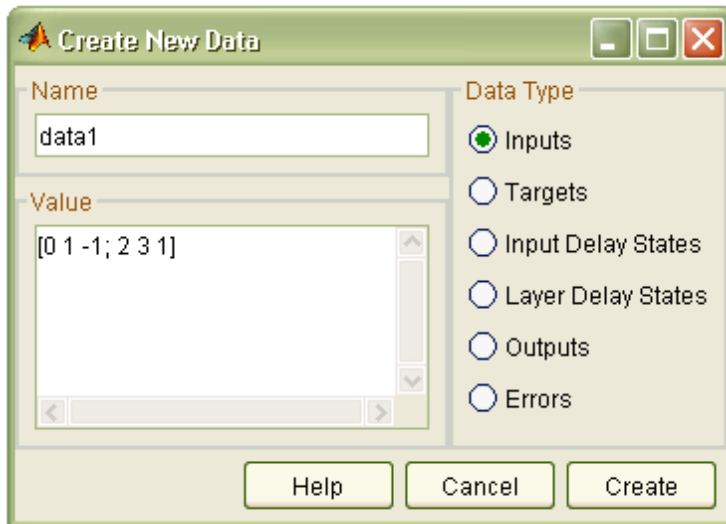


Рис. 3. Окно форматирования данных

Кнопки View, Delete становятся активными только после создания и активизации данных, относящихся к последовательностям входа, цели, выхода или ошибок сети. Кнопка View позволяет просмотреть, а кнопка Delete удалить активизированные данные.

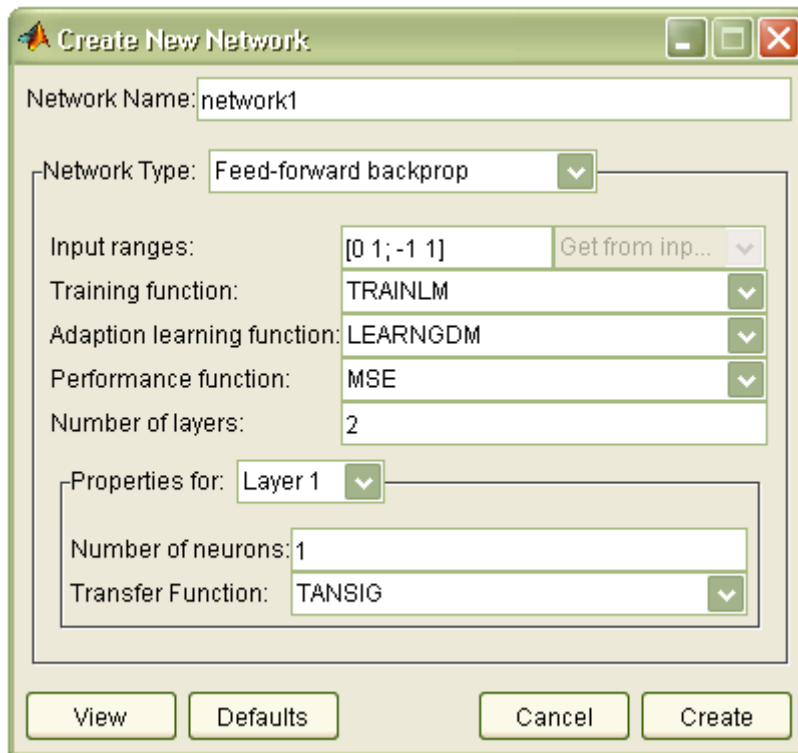


Рис. 4. Окно создания новой нейронной сети

Кнопки View, Delete, Initialize..., Simulate..., Train..., Adapt... становятся активными после создания и активации самой нейронной сети. Они позволяют просмотреть, удалить, инициализировать, промоделировать, обучить или адаптировать нейронную сеть.

Работа с инструментальными средствами GUI

Окно подсказки (**Network/Data Manager Help**) показано на рис. 2. и описывает правила работы диспетчером **Network/Data Manager** при создании нейронной сети.

При создании нейронной сети, необходимо выполнить следующие операции:

- 1) сформировать последовательность входов и целей (кнопка New Data) либо загрузить их из рабочей области системы MatLab или из файла (кнопка Import);
- 2) создать новую нейронную сеть (кнопка New Network) либо загрузить ее из рабочей области систем MatLab или из файла (кнопка Import);
- 3) выбрать тип нейронной сети и нажать кнопку Train..., чтобы открыть окно для задания параметров процедуры обучения;
- 4) открыть окно **Network** для просмотра, инициализации, моделирования, обучения и адаптации сети.

Окно формирования данных (**Create New Data**), показанное на рис. 3, содержит две области редактирования текста для записи имени вводимых данных (область Name) и ввода самих данных (область Value), а также 6 кнопок для указания типа вводимых данных:

Inputs (Входы) – последовательность значений входов;

Targets (Цели) – последовательность значений целей;

Input Delay States (Состояния ЛЗ входа) – начальные условия линии задержки на входе;

Layer Delay States (Состояния ЛЗ слоя) – начальные условия линии задержки в слое;

Outputs – последовательность значений выходов сети;

Errors – разность значений целей и выходов.

Окно создания новой нейронной сети (**Create New Network**) показано на рис. 4 и включает поля для задания параметров создаваемой сети. В зависимости от типа сети количество полей и их названия изменяются.

Приведем описания полей.

Network Name (Имя сети) – стандартное имя сети, присваиваемое GUI-интерфейсом NNTool; в процессе создания новых сетей порядковый номер будет изменяться автоматически.

Network Type (Тип сети) – список сетей, доступных для работы с интерфейсом NNTool. Для удобства этот список повторен в табл. 1. Интерфейс NNTool позволяет создавать нейронные сети только с одной или двумя слоями.

Таблица 1. Типы сетей, доступных с интерфейсом NNTool

№ пп.	Тип сети	Название сети	Число слоев
1	Competitive	Конкурирующая сеть	1
2	Cascade-forward backprop	Каскадная сеть с прямым распространением сигнала и обратным распространением ошибки	2
3	Elman backprop	Сеть Элмана с обратным распространением ошибки	2
4	Feed-forward backprop	Сеть с прямым распространением сигнала и обратным распространением ошибки	2
5	Time delay backprop	Сеть с запаздыванием и обратным распространением ошибки	2
6	Generalized regression	Обобщенная регрессионная сеть	2
7	Hopfield	Сеть Хопфилда	1
8	Linear layer (design)	Линейный слой (создание)	1
9	Linear layer (train)	Линейный слой (обучение)	1
10	LVQ	Сеть для классификации входных векторов	2
11	Perceptron	Персептрон	1
12	Probabilistic	Вероятностная сеть	2
13	Radial basis (exact fit)	Радиально базисная сеть с нулевой ошибкой	2
14	Radial basis (fewer neurons)	Радиально базисная сеть с минимальным числом нейронов	2
15	Self organizing map	Самоорганизующаяся карта Кохонена	1

Примечание:

- 1) для сетей 2, 3, 7 в данной версии NNTool не обеспечивается просмотр структурных схем;
- 2) сети 5, 9 допускают введение линий задержек на входе;
- 3) сети 3 допускают введение линий задержек в слое;
- 4) сети с двумя слоями имеют последовательную структуру, когда выход первого слоя служит входом второго слоя. Исключение составляют сети 3, которые допускают наличие обратной связи в первом слое и передачу входного сигнала на входы обоих слоев.

Input ranges (Диапазон входа) – допустимые границы входов, которые либо назначаются пользователем, либо определяются автоматически по имени входной последовательности, выбираемой из списка Get from Inp...

Training function (Функция обучения) – список обучающих функций.

Performance function (Функция качества обучения) – список функций оценки качества обучения.

Number of layers (Количество слоев) – количество слоев нейронной сети.

Properties for (Свойства) – список слоев.

Number of neurons (Количество нейронов) – количество нейронов в слое.

Transfer function (Функция активации) – функции активации слоя.

Окно для импорта и загрузки данных показано на рис. 5.

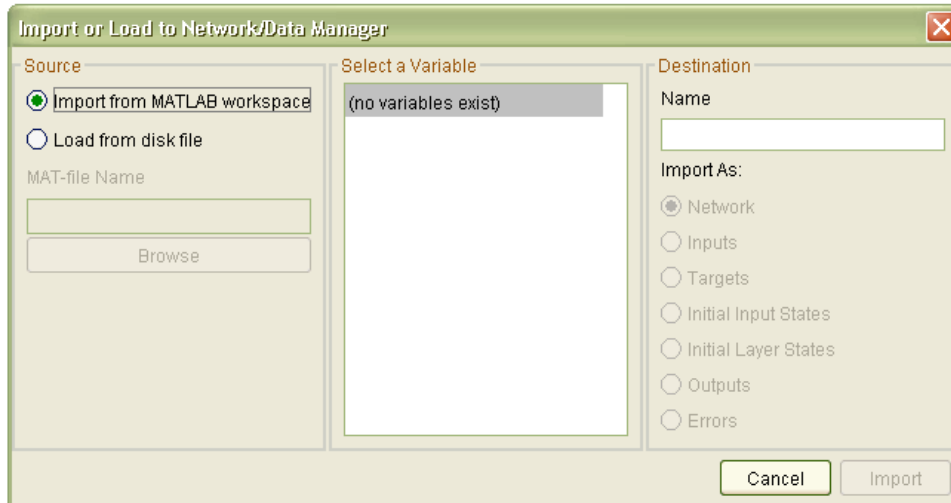


Рис. 5. Окно для импорта и загрузки данных

Source (Источники) – поле для выбора источника данных. Это либо рабочая область системы MatLab (кнопка выбора Input from Matlab Workspace), либо файл (кнопка выбора Load from disk file).

Если выбрана первая кнопка, то в поле **Select a Variable** можно увидеть все переменные рабочей области, и, выбрав одну из них, например x , можно передать ее в поле **Destination (Назначение)** как последовательность входа **Inputs (Входы)**.

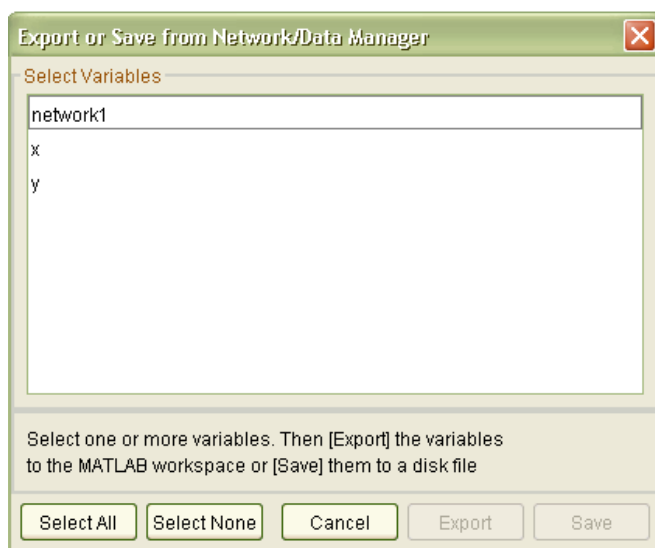


Рис. 6. Окно для экспорта или записи данных в файл

При выборе кнопки Load from disk file активизируется поле **MAT-file Name** и кнопка Browse, что позволяет начать поиск и загрузку файла из файловой системы.

Окно для экспорта или записи данных в файл (**Export or Save from Network/Data Manager**) показано на рис. 6 и позволяет передавать данные из рабочей области GUI-интерфейса NNTool в рабочую область системы MatLab или записать их в виде файла на диске.

Диалоговая панель Network показана на рис. 7.

Данная диалоговая панель открывается только в том случае, когда в окне **Network/Data Manager** выделена созданная сеть и становятся активными кнопки View, Initialize, Simulate, Train, Adapt.

Панель имеет 6 закладок:

View (Просмотреть) – структура сети;

Initialize (Инициализация) – задание начальных весов и смещений;

Simulate (Моделирование) – моделирование сети;

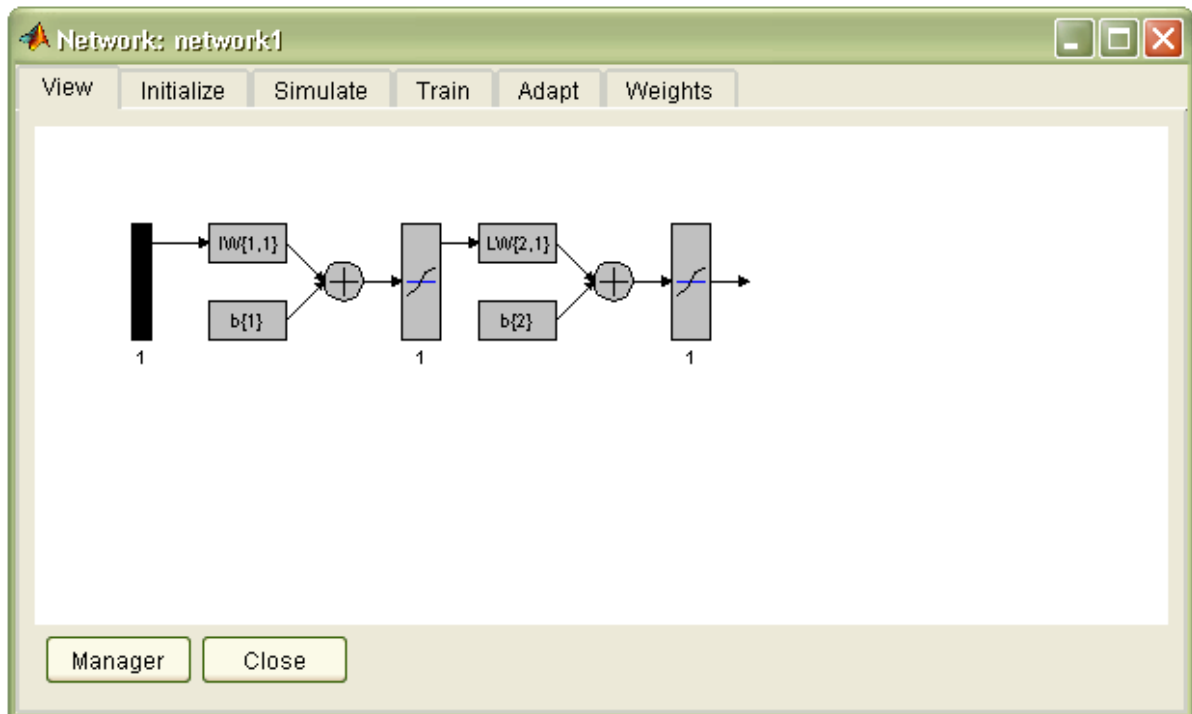


Рис. 7. Диалоговая панель Network

Train (Обучение) – обучение сети;

Adapt (Адаптация) – адаптация и настройка параметров сети;

Weights (Весы) – просмотр установленных весов и смещений.

Пример 1. Создадим, используя графический интерфейс пользователя, нейронную сеть для выполнения операции $y=x^2$ при задании векторов входа $x=[-1 -0.8 -0.5 -0.2 0 0.1 0.3 0.6 0.9 1]$ и цели $y=[1 0.64 0.25 0.4 0 0.01 0.09 0.36 0.81 1]$

Откроем с помощью функции *nntool* основное окно интерфейса, затем сформируем последовательность входов и целей в рабочей области GUI-интерфейса, используя окно **Create New Data**.

С этой целью сначала нажмем кнопку **New Data** и далее – в поле Name окна **Create New Data** – введем сначала имя переменной *x*, затем – в области значений Value – вектор значений $[-1 -0.8 -0.5 -0.2 0 0.1 0.3 0.6 0.9 1]$ и, используя радиокнопку Inputs (в правой части окна), укажем тип переменных (Inputs - Входы). Ввод завершим нажатием радиокнопки Create (Создать).

Аналогичную операцию сделаем для вектора y с указанием (с помощью радиокнопки Targets), что это – вектор целевых данных.

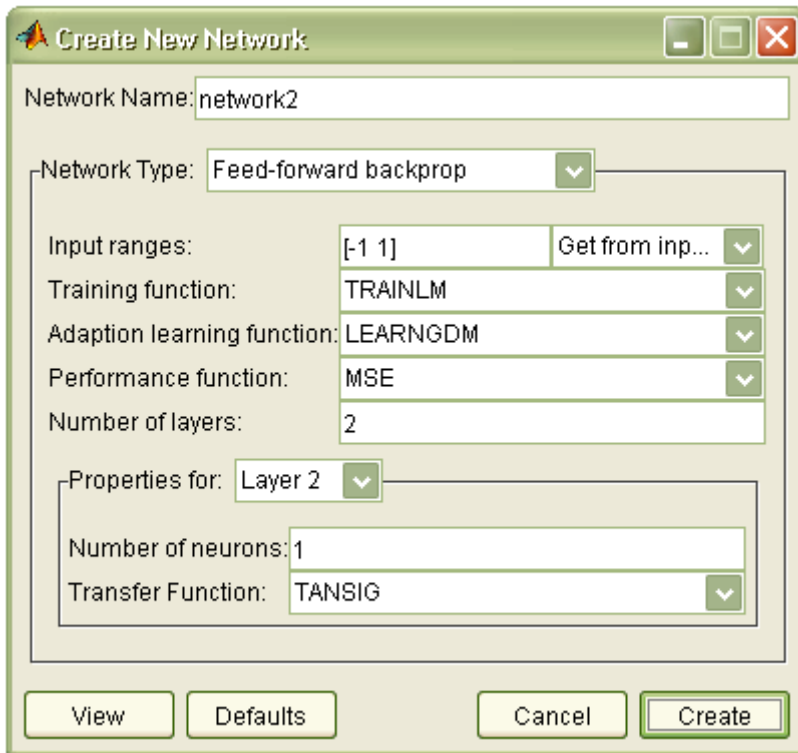


Рис. 8. Окно создания нейронной сети

Создадим новую нейронную сеть. Для этого в окне **Network/Data Manager** нажмем кнопку **New Network**. В открывшемся окне **Create New Network** выберем нейронную сеть типа **Feed-forward backprop** с прямой передачей сигнала и обратным распространением ошибки (см. выше). При создании сети сохраним ей имя, даваемое по умолчанию (**network1**), диапазон входов определим (в окне **Create New Network**) с помощью опции **Get from input**, а количество нейронов (**Number of neurons**) первого слоя (**Layer 1**) установим равным двум. Остальные установки при создании сети оставим по умолчанию (рис. 8). Создание сети завершим нажатием кнопки **Create**.

После этого в окне **Network/Data Manager**, в области **Networks** появится имя новой созданной сети – **network1**. Выберем это имя с помощью мышки, что ведет к активации всех кнопок указанного окна.

Выполним инициализацию сети, для чего с помощью кнопки **Initialize...** выберем вкладку **Initialize**. Это приведет к открытию диалоговой панели, показанной на рис. 9.

Для ввода установленных диапазонов и инициализации весов воспользуемся кнопками **Set Ranges** (Установить диапазоны) **Initialize Weights** (Инициализировать веса). Если требуется вернуться к прежним диапазонам, то следует выбрать кнопки **Revert Ranges** (Вернуть диапазоны) и **Revert Weights** (Вернуть веса), но в условиях примера это не нужно.

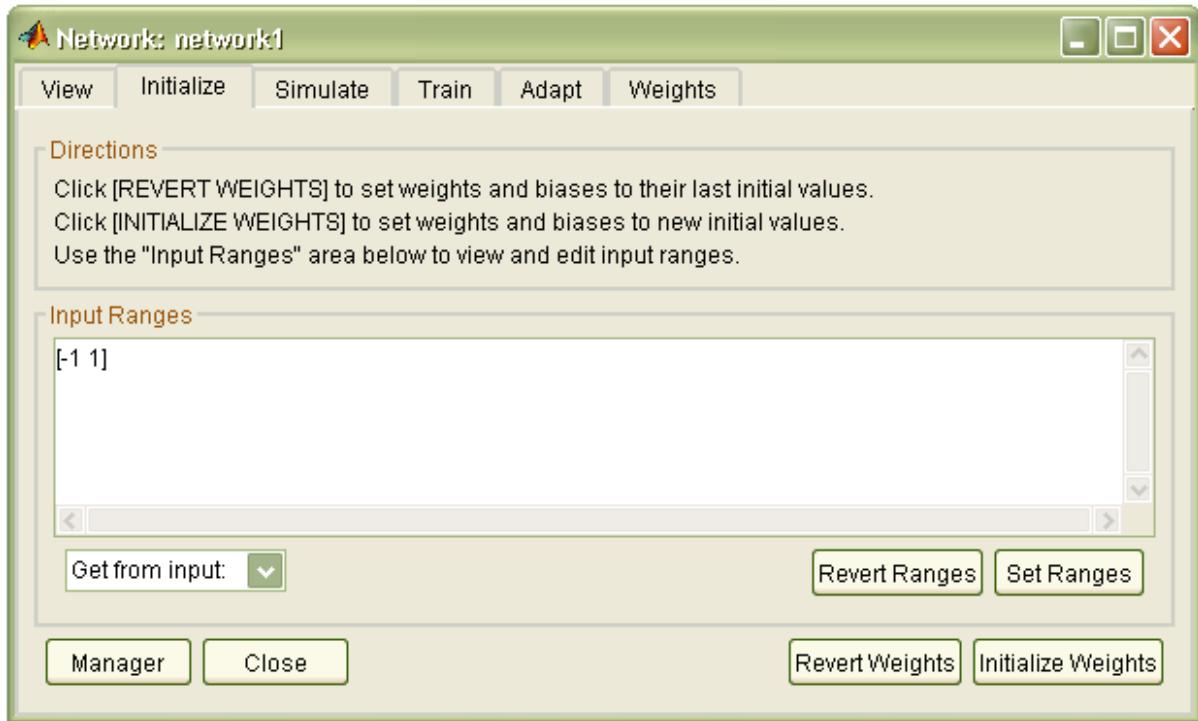


Рис. 9. Окно инициализации параметров сети

Обучение нейронной сети в GUI

Затем выполняется обучение сети, для чего выбирается закладка `Train` и открывается диалоговая панель, показанная на рис. 10.

Панель имеет три закладки:

- 1) `Training Info` (Информация об обучающих последовательностях) – рис. 10;
- 2) `Training Parameters` (Параметры обучения) – рис. 11;
- 3) `Optional Info` (Дополнительная информация) – рис. 12;

Последняя закладка применяется, когда в процессе обучения используются контрольная и тестовая последовательности.

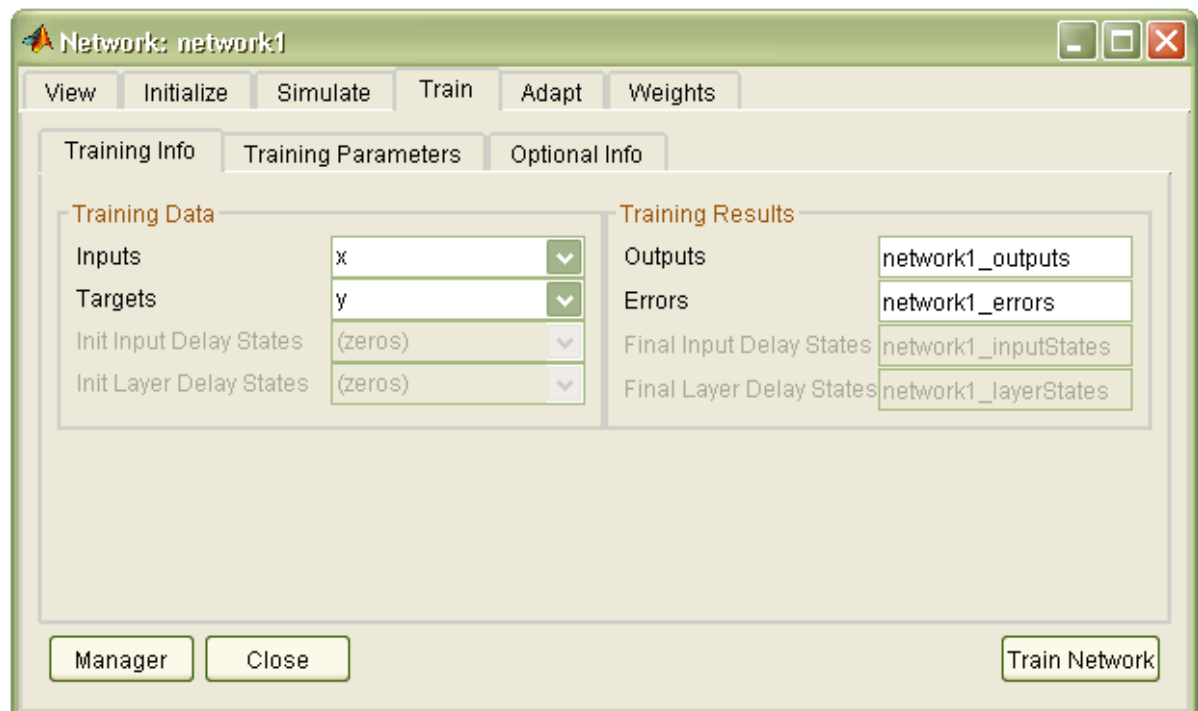


Рис. 10. Окно информации об обучающих последовательностях



Рис. 11. Окно с информацией о параметрах обучения

Применяя эти закладки, можно установить имена последовательностей входа и цели (на вкладке Training Info – в левой ее части необходимо указать x и y), а также значения параметров процедуры обучения (на вкладке Training Parameters; в условиях примера сохраняем значения по умолчанию).

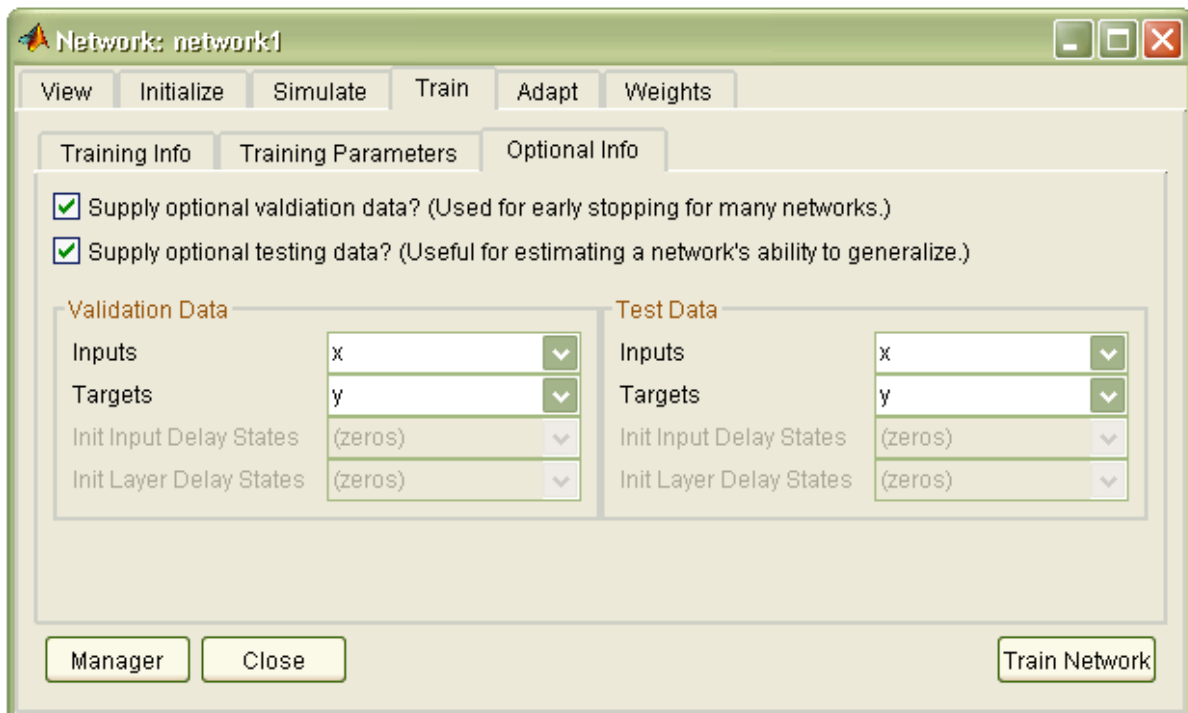


Рис. 12. Окно дополнительной информации.

Теперь нажатие кнопки Train Network вызывает обучение сети. Качество обучения сети на выбранной обучающей последовательности поясняется рис. 15. Видно, что к концу процесса обучения ошибка становится очень малой (вид данного рисунка при повторе вычислений может отличаться от приведенного).

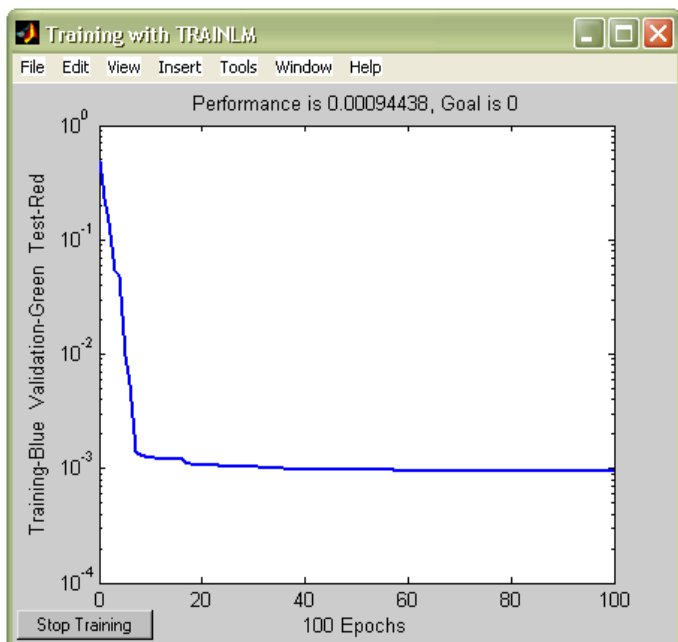


Рис. 13. Изменение ошибки сети в процессе обучения.

Результаты обучения можно просмотреть в окне **Network/Data Manager**, выбрав кнопку Manager. Появится окно (рис. 14), в котором, активизируя имена последовательностей выходов `network1_outputs` или ошибок `network1_errors`, можно просмотреть результаты, используя кнопку View.

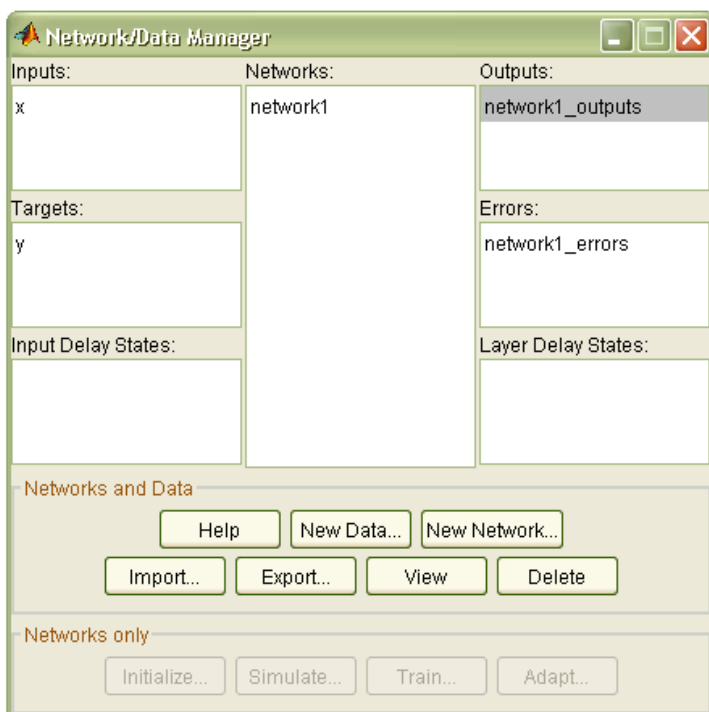


Рис. 14. Окно Network/Data Manager

Следует отметить, что в данном случае точность аппроксимации заданной функции получилась не очень высокой – максимальная абсолютная погрешность составляет 0.055, относительная – 5.5%, в чем можно убедиться, просмотрев значения ошибок `network1_errors` или выходов `network1_outputs` сети (см. рис. 15).

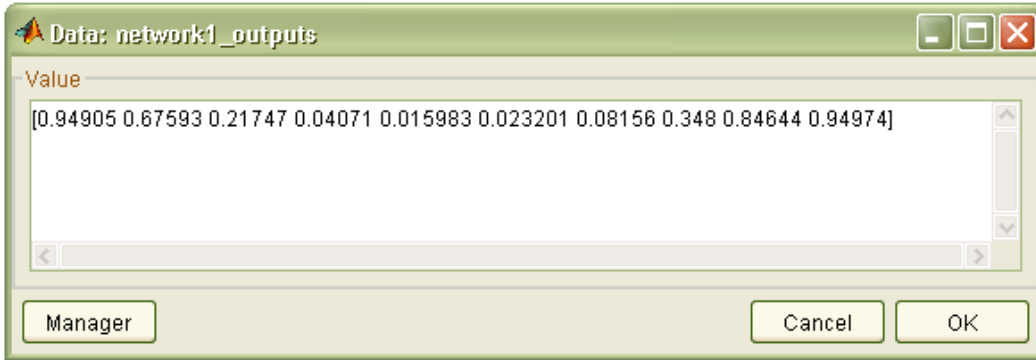


Рис. 15. Значения выходов сети

Для просмотра структурной схемы сети необходимо, выбрав имя сети (network1), воспользоваться кнопкой View (см. рис. 16).

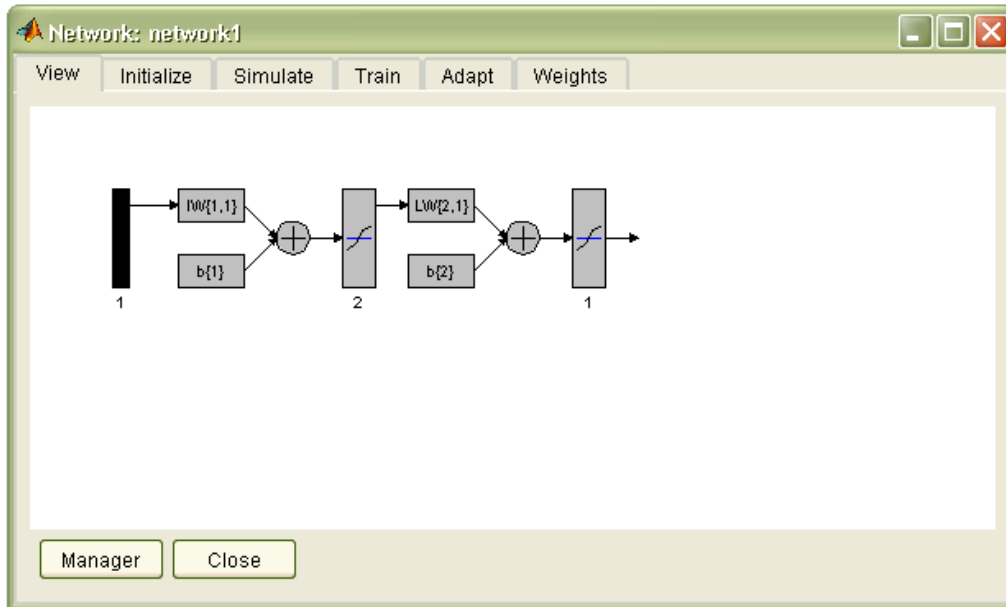


Рис. 16. Структурная схема созданной сети

При необходимости можно экспортировать созданную нейронную сеть в рабочую область системы MatLab (нажав кнопку Export и далее, в открывшемся окне Export or Save from Network/Data Manager, - кнопки Select All (Выбрать все) и Export) и получить информацию о весах и смещениях непосредственно в рабочем окне системы, выполнив команду:

```
>> network1.IW{1,1}, network1.b{1};
```

```
ans=
    1.9568
    1.9603
ans=
   -3.0678
    3.2779
```

и команду:

```
>> network1.IW{2,1}, network1.b{2};
```

```
ans=
    []
ans=
   23.4859
```

Теперь можно построить модель НС в среде Simulink и отобразить ее схему (рис. 17):

```
>> gensim(network1)
```

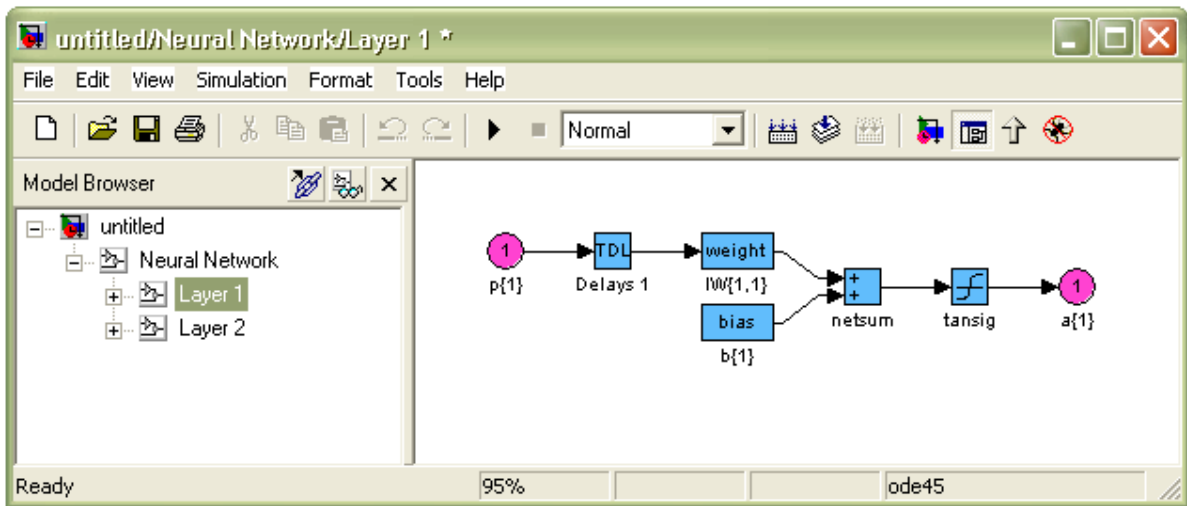


Рис. 17. Вид структурной схемы, созданной НС в среде Simulink

Эта схема является в полной мере функциональной схемой и может быть применена для моделирования нейронной сети.

Лекция Персептрон

Одной из первых искусственных сетей, способных к восприятию и формированию реакции на воспринятый стимул, явился PERCEPTRON Розенблатта. Персептроном, как правило, называют однослойную нейронную сеть, при этом каждый персептронный нейрон в качестве функции активации использует функцию единичного скачка (пороговую функцию).

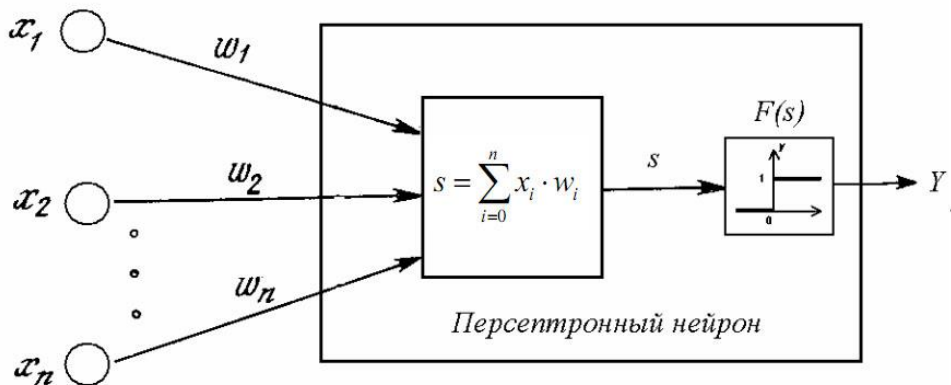


рис.1 Персептрон

Подробная схема такого персептрона изображена на рис.1. Персептрон имеет дополнительный вход x_0 , который всегда

$$y = f(s) = \begin{cases} 0, & s < 0; \\ 1, & s \geq 0 \end{cases}$$

равен 1. В таком случае, пороговое смещение и $\Theta = 0$ и

Обучение персептрона состоит в подстройке весовых коэффициентов w_i , где $i = \underline{0}, \underline{n}$. Обученный персептрон способен разделять требуемое множество образов на два класса. (К первому классу относятся входные образы, для которых на выходе персептрона получено нулевое значение, ко второму классу - образы, для которых получено единичное значение).

Обучение персептрона - это обучение с учителем, то есть должен существовать набор векторов (X^k, y_k) , $k = 1, P$, называемый *обучающей выборкой*. Здесь $X^k = (x_1^k, x_2^k, \dots, x_n^k)$, $k = \underline{1}, \underline{P}$ - примеры входных образов, для которых заранее известна их принадлежность к одному из двух данных классов.

Будем называть персептрон обученным на данной обучающей выборке, если при подаче на вход каждого вектора X^k на выходе всякий раз получается соответствующее значение $y_k \in \{0, 1\}$. Предложенный Ф.Розенблаттом метод обучения состоит в итерационной подстройке весовых коэффициентов w_i последовательно уменьшающей выходные ошибки.

Алгоритм включает несколько шагов.

Шаг 0. Проинициализировать весовые коэффициенты w_i , $i = \underline{0}, \underline{n}$ небольшими случайными значениями (например, из диапазона $[-0.3, 0.3]$).

Шаг 1. Подать на вход персептрона один из обучающих векторов X^k и вычислить его выход y .

Шаг 2. Если выход правильный ($y=y_k$), перейти на шаг 4. Иначе вычислить ошибку - разницу между верным и полученным значениями выхода: $\delta = y_k - y$.

Шаг 3. Весовые коэффициенты модифицируются по следующей формуле:

$w_i^{t+1} = w_i^t + v \cdot \delta \cdot x_i^k$. Здесь t и $t+1$ - номера соответственно текущей и следующей итераций; v - коэффициент скорости обучения, ($0 < v \leq 1$); x_i^k - i -тая компонента входного вектора X^k .

Шаг 4. Шаги 1-3 повторяются для всех обучающих векторов. Один цикл последовательного предъявления всей выборки называется эпохой. Обучение завершается по истечении нескольких эпох, когда сеть перестанет ошибаться.

Замечание 1. Коэффициент скорости обучения v является параметром данного алгоритма. Как правило, его выбирают из диапазона $[0.5, 0.7]$. В некоторых случаях (при большом объеме обучающей выборки) целесообразно постепенно уменьшать значение v , начиная, например, с 1.

Замечание 2. Используемая на шаге 3 формула модифицирует только весовые коэффициенты, отвечающие ненулевым значениям входов x_i^k , поскольку только

они влияли на величину $s = \sum_{i=0}^n w_i$, а, следовательно, и на значение y .

Очевидно, что если $y_k > y$ (получен неправильный нулевой выход вместо правильного единичного), то, поскольку $\delta > 0$, весовые коэффициенты (а вместе с ними и величина s) будут увеличены и тем самым уменьшат ошибку. В противном случае весовые коэффициенты будут уменьшены, и s тоже уменьшится, приближая тем самым y к значению y_k .

Обобщим теперь этот алгоритм обучения на случай однослойной сети, включающей n персептронных нейронов (рис. 4). Такая сеть (при достаточно большом числе нейронов) может осуществлять разделение образов на произвольное требуемое число классов.

Пусть имеется *обучающая выборка*, состоящая из пар векторов

(X^k, Y^k) , $k = \underline{1}, \underline{P}$. Назовем нейронную сеть обученной на данной обучающей выборке, если при подаче на входы сети каждого вектора X^k на выходах всякий раз получается соответствующий вектор Y^k . Обучение заключается в итерационной подстройке матрицы весов W (w_{ij} - вес синаптической связи между i -м входом и j -м нейроном), последовательно уменьшающей ошибку в выходных векторах. Алгоритм включает следующие шаги.

Шаг 0. Проинициализировать элементы весовой матрицы W небольшими случайными значениями.

Шаг 1. Подать на входы один из входных векторов X^k и вычислить ее выход Y .

Шаг 2. Если выход правильный ($Y = Y^k$), перейти на шаг 4. Иначе вычислить вектор ошибки - разницу между идеальным и полученным значениями выхода:

$$\delta = Y^k - Y.$$

Шаг 3. Матрица весов модифицируется по следующей формуле:
 $w_{ij}^{t+1} = w_{ij}^t + v \cdot \delta \cdot x_i$. Здесь t и $t+1$ – номера соответственно текущей и следующей итераций;
 v – коэффициент скорости обучения, ($0 < v \leq 1$); x_i – i -тая компонента входного вектора X^k ; j – номер нейрона в слое.

Шаг 4. Шаги 1-3 повторяются для всех обучающих векторов. Обучение завершается, когда сеть перестанет ошибаться.

Представленный метод обучения носит название " δ - правило". Доказанная Розенблатом теорема о сходимости обучения по δ - правилу говорит о том, что персептрон способен обучиться любому обучающему набору, который он способен представить. Ниже мы более подробно обсудим возможности персептрона по представлению информации.

Лекция Линейная разделимость и персептронная представляемость

Каждый нейрон персептрона является формальным пороговым элементом, принимающим единичные значения в случае, если суммарный взвешенный вход больше некоторого порогового значения:

$$Y_j = \begin{cases} 0, & \sum_{i=1}^n x_i \cdot w_i < \Theta; \\ 1, & \sum_{i=1}^n x_i \cdot w_i \geq \Theta \end{cases}$$

Таким образом, при заданных значениях весов и порогов, нейрон имеет определенное выходное значение для каждого возможного вектора входов. Множество входных векторов, при которых нейрон активен ($Y_j=1$), отделено от множества векторов, на которых нейрон пассивен ($Y_j=0$), гиперплоскостью, уравнение

которой $\sum_{i=1}^n x_i \cdot w_i = \Theta$ Следовательно, нейрон способен отделить только такие

два множества векторов входов, для которых имеется гиперплоскость, отсекающая одно множество от другого. Такие множества называют линейно разделимыми. роиллюстрируем это понятие на примере. Рассмотрим однослойный персептрон, состоящий из одного нейрона с двумя входами. Входной вектор содержит только две булевы компоненты x_1 и x_2 , определяющие плоскость.

На данной плоскости возможные значения входных векторов отвечают вершинам единичного квадрата. В каждой вершине зададим требуемое значение выхода нейрона: 1 (на рис. 8 - белая точка) или 0 (черная точка). Требуется определить, существует ли такой набор весов и порогов нейрона, при котором нейрон сможет получить эти значения выходов? На рис. 8 представлена одна из ситуаций, когда этого сделать нельзя вследствие линейной неразделимости множеств белых и черных точек.

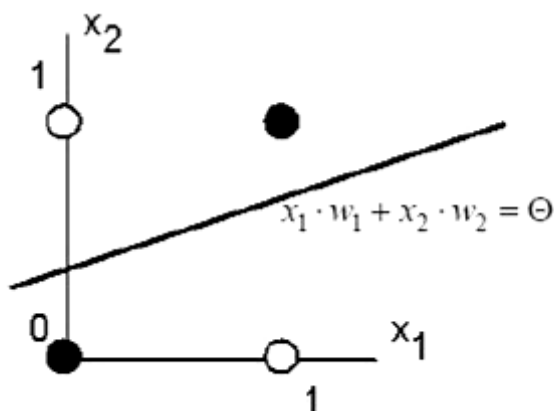


Рис. 1. Белые точки не могут быть отделены одной прямой от черных

Требуемые выходы нейрона для этого рисунка определяются таблицей, в которой легко узнать задание логической функции "исключающее или" (XOR).

1	2	3

Невозможность реализации однослойным перцептроном этой функции получила название *проблемы исключающего ИЛИ*. Видно, что однослойный перцептрон крайне ограничен в своих возможностях точно представить наперед заданную логическую функцию.

Хотя данный пример нагляден, он не является серьезным ограничением для нейросетей. Функция XOR легко реализуется простейшей двухслойной сетью,

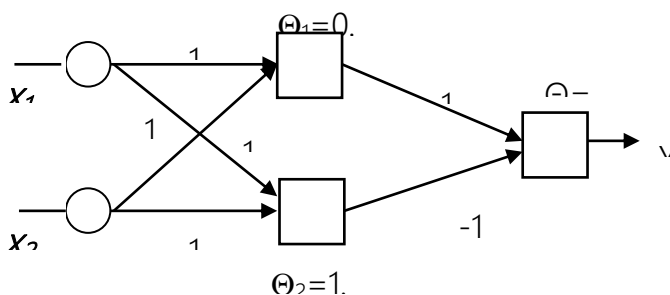


Рис. 2. Двухслойная сеть, реализующая функцию XOR

причем многими способами. Один из примеров такой сети показан на рис. 2.

Весовые коэффициенты w_{11} , w_{12} , w_{21} , w_{22} первого слоя все равны единице, весовые коэффициенты второго слоя v_1, v_2 соответственно равны 1 и -1, пороговые значения каждого нейрона указаны на рисунке.

Таблица истинности для такой сети имеет вид:

x_1	x_2	s_1	s_2	v_1	v_2	S	Y
0	0	0	0	0	0	0	0
1	0	1	1	1	0	1	1
0	1	1	1	1	0	1	1
1	1	2	2	1	1	0	0

Здесь $s_1 = x_1 * w_{11} + x_2 * w_{21}$ - значение, поступающее на вход первого нейрона первого слоя, $s_2 = x_1 * w_{12} + x_2 * w_{22}$ - вход второго нейрона первого слоя; y_1 , y_2 - выходы соответствующих нейронов первого слоя; S - входное значение нейрона второго слоя; Y - выход сети.

Лекция Многослойная нейронная сеть прямого распространения

Структура многослойной нейронной сети прямого распространения (МНСПР) изображена на рис. 1. Она состоит из простых вычислительных элементов (нейронов), которые связаны между собой посредством синаптических связей. Ее особенностью является послонная организация нейронов и односторонняя передача сигнала от предыдущего слоя к последующему. Слои нейронов со второго по предпоследний называются скрытыми слоями, а последний слой называется выходным.

Рассмотрим идею одного из самых распространенных алгоритмов обучения алгоритма обратного распространения ошибки (back propagation). Предложенный в середине 1980-х годов *алгоритм обратного распространения ошибки* стал одним из ведущих факторов, породивших современный интерес к нейронным сетям, т.к. являлся эффективным способом обучения [нейронной сети](#) достаточно произвольной топологии.

Данный алгоритм применяется для обучения такого класса нейронных сетей, как многослойные сети прямого распространения. Обычно сеть состоит из множества сенсорных элементов (входных узлов и узлов источника), которые образуют входной слой, одного или нескольких скрытых слоев вычислительных нейронов и одного выходного слоя нейронов рис.1.2.

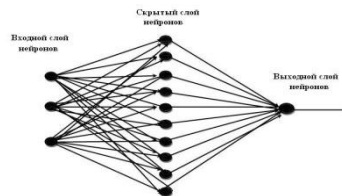


Рис.1.2. Многослойная НС прямого распространения.

Входной сигнал распространяется по сети в прямом направлении, от слоя к слою. Многослойные НС успешно применяются для решения разнообразных сложных задач. При этом обучение с учителем выполняется с помощью алгоритма обратного распространения ошибки. Этот алгоритм основывается на коррекции ошибки. Обучение методом обратного распространения ошибки предполагает два прохода по слоям: прямого и обратного рис.1.3. При прямом проходе входной вектор подается на сенсорные узлы сети, после чего распространяется по сети от слоя к слою. В результате генерируется набор выходных сигналов, который является фактической реакцией сети на данный входной образ.

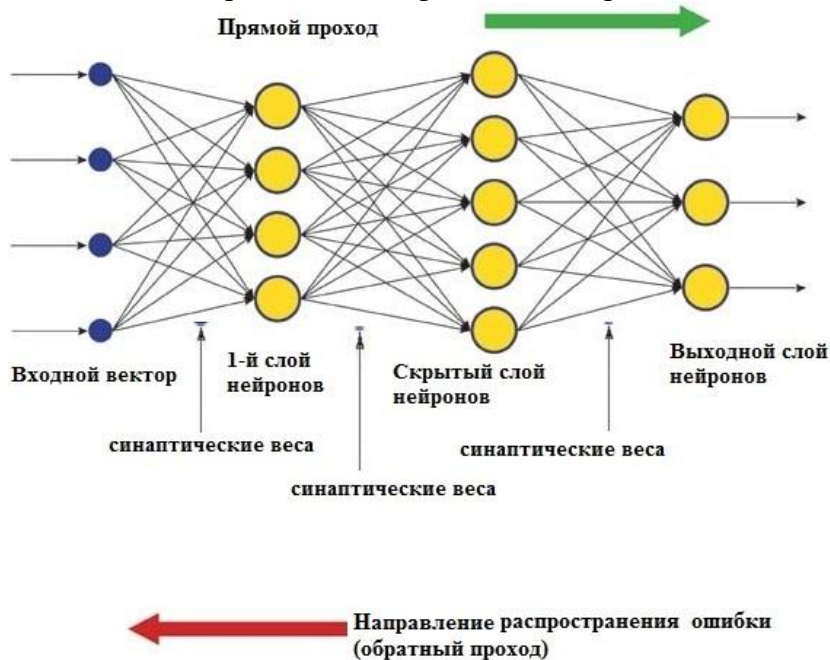


Рис.1.3. Два прохода обучения в алгоритме обратного распространения.

Во время прямого прохода все синаптические веса сети фиксированы. Во время обратного прохода все синаптические веса сети настраиваются в соответствии с правилом коррекции ошибок (рис.1.4), тем самым формируется сигнал ошибки.

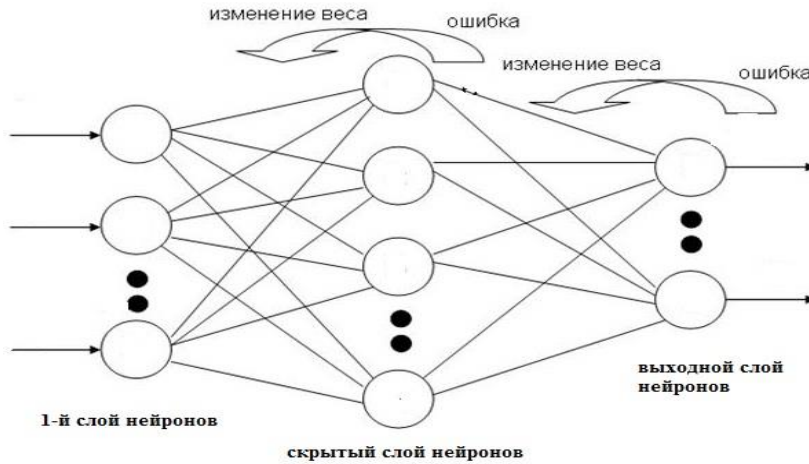


Рис.1.4. Обратный проход в алгоритме обратного распространения ошибки.

Этот сигнал впоследствии распространяется по сети в направлении, обратном направлению синаптических связей. Синаптические веса настраиваются с целью максимального приближения выходного сигнала к желаемому. Классический метод обратного распространения относится к алгоритмам с линейной сходимостью.

Рассмотрим алгоритм обратного распространения ошибки на рис.1.5 изображен нейрон j , на который поступает поток сигналов.

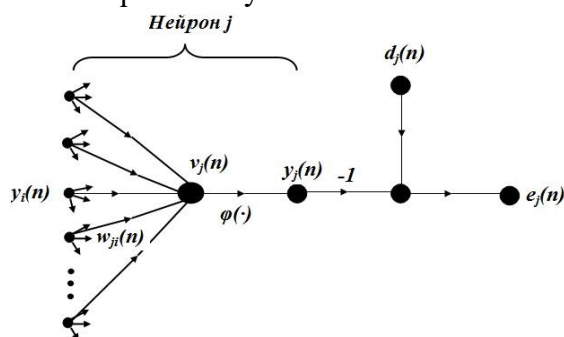


Рис.2.7. Граф передачи сигнала в пределах некоторого нейрона j .

Коррекция $\Delta w_{ji}(n)$, применяемая к синаптическому весу, соединяющему нейроны i и j , определяется следующим дельта-правилом:

$$\Delta w_{ji}(n) = \eta \delta_j(n) y_i(n) \quad , \quad (2.6)$$

где $\Delta w_{ji}(n)$ - коррекция веса, η - параметр скорости обучения, $\delta_j(n)$ - локальный градиент, $y_i(n)$ - входной сигнал нейрона j .

Значение локального градиента $\delta_j(n)$ зависит от положения нейрона в сети:

– если нейрон j - выходной, то градиент $\delta_j(n)$ равен произведению производной $f'_j(v_j(n))$ на сигнал ошибки $e_j(n)$ для нейрона j .

– если нейрон j - скрытый, то градиент $\delta_j(n)$ равен произведению производной $f'_j(v_j(n))$ на взвешенную сумму градиентов, вычисленных для нейронов следующего скрытого или выходного, которые непосредственно связаны с данным нейроном j .

Вычисление локального градиента δ для каждого нейрона многослойного персептрона требует знания производной функции активации $f(\cdot)$, связанной с этим нейроном. Примером непрерывно дифференцируемой нелинейной функции активации, которая часто

используется в многослойных перцептронах, является сигмоидальная логистическая функция:

$$f_j(v_j(n)) = \frac{1}{1 + \exp(-\alpha v_j(n))}, \quad (2.7)$$

где $v_j(n)$ - индуцированное локальное поле нейрона j . Дифференцируя (2.7) по $v_j(n)$ получим:

$$f'_j(v_j(n)) = \frac{\alpha \exp(-\alpha v_j(n))}{[1 + \exp(-\alpha v_j(n))]^2}, \quad (2.8)$$

Так как $y_j(n) = f_j(v_j(n))$, то выражение (2.8) можно преобразовать к следующему виду:

$$f'_j(v_j(n)) = \alpha y_j(n)[1 - y_j(n)], \quad (2.9)$$

для нейрона j , расположенного в выходном слое, $y_j(n) = o_j(n)$. Отсюда локальный градиент нейрона j можно выразить следующим образом:

$$\delta_j(n) = e_j(n) f'_j(v_j(n)) = \alpha [d_j(n) - o_j(n)] o_j(n) [1 - o_j(n)] \quad (2.10)$$

где $o_j(n)$ - функциональный сигнал на выходе нейрона j , $d_j(n)$ - его желаемый сигнал. На рис. 2.8 показана передача сигнала, детально отражающий связь выходного нейрона k со скрытым нейроном j .

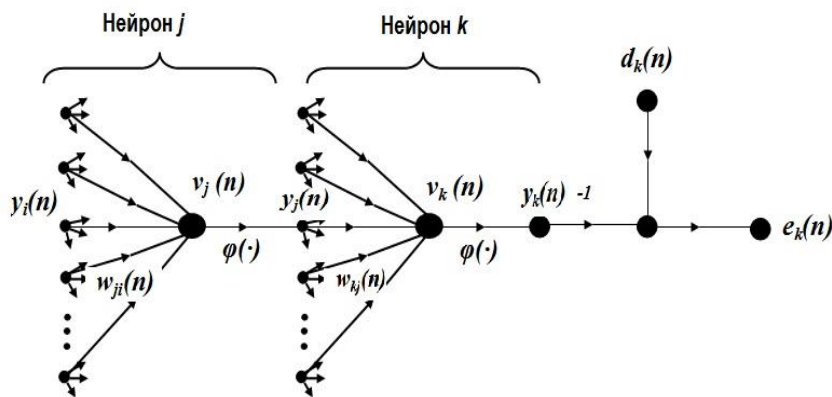


Рис.2.8. Граф передачи сигнала от одного нейрона к другому нейрону.

Для скрытого нейрона j локальный градиент можно выразить:

$$\delta_j(n) = f'_j(v_j(n)) \sum_{k=1}^M \delta_k(n) w_{kj}(n) = \alpha y_j(n) [1 - y_j(n)] \sum_{k=1}^M \delta_k(n) w_{kj}(n), \quad (2.11)$$

где M - количество нейронов в выходном слое.

Лекция Алгоритм обучения сети обратного распространения

Рассмотрим теперь полный алгоритм обучения нейросети.

Шаг 1. Инициализация сети.

Весовым коэффициентам присваиваются малые случайные значения, например из диапазона $(-0.3, 0.3)$; задаются ε - параметр точности обучения, α - параметр скорости обучения (как правило, ≈ 0.1 и может еще уменьшаться в процессе обучения), N_{\max} - максимально допустимое число итераций.

Шаг 2. Вычисление текущего выходного сигнала.

На вход сети подается один из образов обучающей выборки и определяются значения выходов всех нейронов нейросети.

Шаг 3. Настройка синаптических весов.

Рассчитать изменение весов для скрытого слоя и выходного слоя по формулам

$$\delta_j(n) = f'_j(v_j(n)) \sum_{k=1}^M \delta_k(n) w_{kj}(n) = \alpha y_j(n) [1 - y_j(n)] \sum_{k=1}^M \delta_k(n) w_{kj}(n)$$

$$\delta_j(n) = e_j(n) f'_j(v_j(n)) = \alpha [d_j(n) - o_j(n)] o_j(n) [1 - o_j(n)]$$

$$\Delta w_{ji}(n) = \eta \delta_j(n) y_i(n)$$

$$w_{ji}(n+1) = w_{ji}(n) + \eta \delta_j(n) y_i(n)$$

Шаг 4. Шаги 2-3 повторяются для всех обучающих векторов. Обучение завершается по достижении для каждого из обучающих образов значения функции ошибки, не превосходящего ε или после максимально допустимого числа итераций.

Замечание 1. На шаге 2 векторы из обучающей последовательности лучше предъявлять на вход в случайном порядке.

Замечание 2. Во многих случаях желательно наделять каждый нейрон обучаемым смещением. Это позволяет сдвигать начало отсчета логистической функции, давая эффект, аналогичный подстройке порога персептронного нейрона, и приводит к ускорению процесса обучения. Эта возможность может быть легко введена в обучающий алгоритм с помощью добавляемого к каждому нейрону веса, присоединенного к +1. Этот вес обучается так же, как и все остальные веса, за исключением того, что подаваемый на него сигнал всегда равен +1, а не выходу нейрона предыдущего слоя.

Замечание 3. Количество входов и выходов сети, как правило, диктуется условиями задачи, а размер скрытого слоя находят экспериментально. Обычно число нейронов в нем составляет 30-50% от числа входов. Слишком большое количество нейронов скрытого слоя приводит к тому, что сеть теряет способность к обобщению (она просто досконально запоминает элементы обучающей выборки и не реагирует на схожие образцы, что неприемлемо для задач распознавания). Если число нейронов в скрытом слое слишком мало, сеть оказывается просто не в состоянии обучиться.

Замечание 4. Выходы каждого нейрона сети лежат в диапазоне (0,1) -области значений логистической функции - это надо учитывать при формировании обучающей выборки. Если необходимо получить от сети бинарный выход, то, как правило, вместо 0 используют 0.1, а вместо 1 - 0.9, так как границы интервала недостижимы.

Модификации алгоритма обратного распространения связаны с использованием различных функций ошибки, других активационных функций, различных процедур определения направления и величины шага.

Обратное распространение было использовано в широкой сфере прикладных исследований. В частности фирма NEC в Японии использовала обратное распространение для визуального распознавания букв (в том числе рукописных), причем точность превысила 99%. Достигнут впечатляющий успех с Net-Talk системой, которая превращает печатный английский текст в высококачественную речь. Магнитофонная запись процесса обучения сильно напоминает звуки ребенка на разных этапах обучения речи. Но несмотря на многочисленные успешные применения обратного распространения, оно не является панацеей. Больше всего неприятностей приносит неопределенно долгий процесс обучения. В сложных задачах для обучения сети могут потребоваться часы или даже дни, она может и вообще не обучиться. Неудачи в обучении часто возникают по причине попадания сети в локальный минимум, что, к сожалению, является характерной особенностью методов градиентного спуска. Исправить ситуацию в таком случае иногда помогают небольшие случайные изменения весовых значений сети.

Лекция Последовательный и пакетный режимы обучения

В практических приложениях алгоритма обратного распространения в процессе обучения многослойного персептрона ему многократно предъявляется предопределенное множество обучающих примеров. Как уже отмечалось, один полный цикл предъявления полного набора примеров обучения называют *эпохой*. Процесс обучения проводится от эпохи к эпохе, пока синаптические веса и уровни порога не стабилизируются, а среднеквадратическая ошибка на всем обучающем множестве не сойдется к некоторому минимальному значению. Целесообразно *случайным образом изменять порядок представления примеров обучения* для разных эпох. Такой принцип предъявления образов делает поиск в пространстве весов стохастическим, предотвращая потенциальную возможность появления замкнутых циклов в процессе эволюции синаптических весов.

Для данного обучающего множества алгоритм обратного распространения можно реализовать двумя способами.

1. Последовательный режим

Последовательный режим обучения по методу обратного распространения также иногда называют *стохастическим* или *интерактивным*. В этом режиме корректировка весов проводится после подачи каждого примера. Для примера рассмотрим эпоху, состоящую из N обучающих примеров, упорядоченных следующим образом: $(x(1), d(1)), \dots, (x(iV), d(iV))$. Сети предъявляется первый пример $(x(1), d(1))$ этой эпохи, после чего выполняются описанные выше прямые и обратные вычисления. В результате проводится корректировка синаптических весов и уровней порогов в сети. После этого сети предъявляется вторая пара $(x(2), d(2))$ в эпохе, повторяются прямой и обратный проходы, приводящие к следующей коррекции синаптических весов и уровня порога. Этот процесс повторяется, пока сеть не завершит обработку последнего примера (пары) данной эпохи — $(x(iV), d(iV))$.

2. Пакетный режим

В *пакетном режиме* (batch mode) обучения по методу обратного распространения корректировка весов проводится после подачи в сеть примеров обучения (эпохи). Для конкретной эпохи функция стоимости определяется как среднеквадратическая ошибка (4.2) и (4.3), представленная в составной форме:

$$\mathcal{E}_{av}(n) = \frac{1}{2N} \sum_{n=1}^N \sum_{i \in C} e_j^2(n),$$

где сигнал ошибки $e_j(n)$ соответствует нейрону j для примера обучения n и определяется формулой (4.1). Ошибка $e_j(n)$ равна разности между $d_j(n)$ и $y_j(n)$ для j -го элемента вектора желаемых откликов $d_j(n)$ и соответствующего выходного нейрона сети. В выражении (4.42) внутреннее суммирование по j выполняется по всем нейронам выходного слоя сети, в то время как внешнее суммирование по n выполняется по всем образам данной эпохи. При заданном параметре скорости обучения η корректировка, применяемая к синаптическому весу w_{ji} , связывающему нейроны i и j , определяется следующим дельта-правилом:

$$\Delta w_{ji}(n) = -\eta \frac{\partial \mathcal{E}_{av}}{\partial w_{ji}} = -\frac{\eta}{N} \sum_{n=1}^N e_j(n) \frac{\partial e_j(n)}{\partial w_{ji}}.$$

(4.43)

Для вычисления частной производной $\frac{de_j(n)}{dw_j(n)}$ нужно проделать тот же путь, что и ранее.

Согласно (4.43), в пакетном режиме корректировка веса Δw_{ji} выполняется только после прохождения по сети всего множества примеров.

В заключение можно сказать, что несмотря на многие недостатки последовательного режима алгоритма обратного распространения, он остается очень популярным (особенно при решении задач распознавания образов) по двум практическим причинам.

- Этот алгоритм прост в реализации.
- Обеспечивает эффективное решение сложных и больших задач.

Лекция Классификация и кластеризация

Задача классификации

Классификация является наиболее простой и одновременно наиболее часто решаемой задачей Data Mining. Ввиду распространенности задач классификации необходимо четкое понимания сути этого понятия.

Приведем несколько определений.

Классификация - системное распределение изучаемых предметов, явлений, процессов по родам, видам, типам, по каким-либо существенным признакам для удобства их исследования; группировка исходных понятий и расположение их в определенном порядке, отражающем степень этого сходства.

Классификация - упорядоченное по некоторому принципу множество объектов, которые имеют сходные классификационные признаки (одно или несколько свойств), выбранных для определения сходства или различия между этими объектами.

Под классификацией будем понимать отнесение объектов (наблюдений, событий) к одному из заранее известных классов.

Классификация - это закономерность, позволяющая делать вывод относительно определения характеристик конкретной группы. Таким образом, для проведения классификации должны присутствовать признаки, характеризующие группу, к которой принадлежит то или иное событие или объект (обычно при этом на основании анализа уже классифицированных событий формулируются некие правила).

Классификация относится к стратегии обучения с учителем, которое также именуют контролируемым или управляемым обучением.

Задачей классификации часто называют предсказание категориальной зависимой переменной (т.е. зависимой переменной, являющейся категорией) на основе выборки непрерывных и/или категориальных переменных.

Например, можно предсказать, кто из клиентов фирмы является потенциальным покупателем определенного товара, а кто - нет, кто воспользуется услугой фирмы, а кто - нет, и т.д. Этот тип задач относится к задачам бинарной классификации, в них зависимая переменная может принимать только два значения (например, да или нет, 0 или 1).

Другой вариант классификации возникает, если зависимая переменная может принимать значения из некоторого множества предопределенных классов. Например, когда необходимо предсказать, какую марку автомобиля захочет купить клиент. В этих случаях рассматривается множество классов для зависимой переменной.

Классификация может быть одномерной (по одному признаку) и многомерной (по двум и более признакам).

Многомерная классификация была разработана биологами при решении проблем дискриминации для классифицирования организмов. Одной из первых работ, посвященных этому направлению, считают работу Р. Фишера (1930 г.), в которой организмы разделялись на подвиды в зависимости от результатов измерений их физических параметров. Биология

была и остается наиболее востребованной и удобной средой для разработки многомерных методов классификации.

Рассмотрим задачу классификации на простом примере. Допустим, имеется база данных о клиентах туристического агентства с информацией о возрасте и доходе за месяц. Есть рекламный материал двух видов: более дорогой и комфортный отдых и более дешевый, молодежный отдых. Соответственно, определены два класса клиентов: класс 1 и класс 2. База данных приведена в таблице 5.1.

Таблица 5.1. База данных клиентов туристического агентства

Ид клиента	Возраст	Доход	Класс
	8	5	
	2	00	
	0	0	
	2	20	
	4	5	
	5	2	
	2	0	
	9	5	
	2	5	
0	0	0	

Задача. Определить, к какому классу принадлежит новый клиент и какой из двух видов рекламных материалов ему стоит отсылать.

Для наглядности представим нашу базу данных в двумерном измерении (возраст и доход), в виде множества объектов, принадлежащих классам 1 (оранжевая метка) и 2 (серая метка). На рис. 5.1 приведены объекты из двух классов.

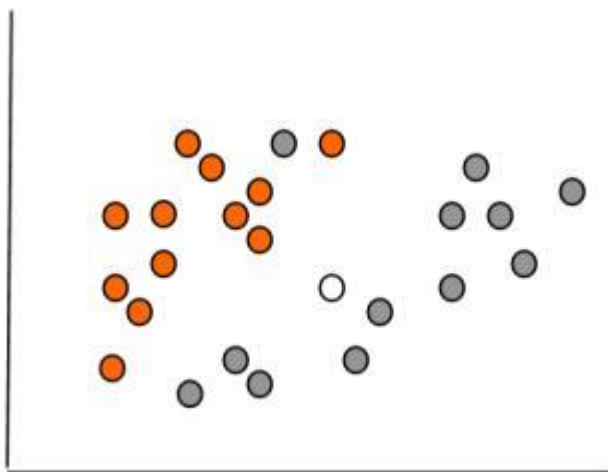


Рис. 5.1. Множество объектов базы данных в двумерном измерении

Решение нашей задачи будет состоять в том, чтобы определить, к какому классу относится новый клиент, на рисунке обозначенный белой меткой.

Процесс классификации

Цель процесса классификации состоит в том, чтобы построить модель, которая использует прогнозирующие атрибуты в качестве входных параметров и получает значение зависимого атрибута. Процесс классификации заключается в разбиении множества объектов на классы по определенному критерию.

Классификатором называется некая сущность, определяющая, какому из predetermined классов принадлежит объект по вектору признаков.

Для проведения классификации с помощью математических методов необходимо иметь формальное описание объекта, которым можно оперировать, используя математический аппарат классификации. Таким описанием в нашем случае выступает база данных. Каждый объект (запись базы данных) несет информацию о некотором свойстве объекта.

Набор исходных данных (или выборку данных) разбивают на два множества: обучающее и тестовое.

Обучающее множество (training set) - множество, которое включает данные, используемые для обучения (конструирования) модели.

Такое множество содержит входные и выходные (целевые) значения примеров. Выходные значения предназначены для обучения модели.

Тестовое (test set) множество также содержит входные и выходные значения примеров. Здесь выходные значения используются для проверки работоспособности модели.

Процесс классификации состоит из двух этапов: конструирования модели и ее использования.

1. Конструирование модели: описание множества predetermined классов.

- Каждый пример набора данных относится к одному predetermined классу.
- На этом этапе используется обучающее множество, на нем происходит конструирование модели.
- Полученная модель представлена классификационными правилами, деревом решений или математической формулой.

2. Использование модели: классификация новых или неизвестных значений.

- Оценка правильности (точности) модели.
 - Известные значения из тестового примера сравниваются с результатами использования полученной модели.
 - Уровень точности - процент правильно классифицированных примеров в тестовом множестве.
 - Тестовое множество, т.е. множество, на котором тестируется построенная модель, не должно зависеть от обучающего множества.
- Если точность модели допустима, возможно использование модели для классификации новых примеров, класс которых неизвестен.

5. Информационные и образовательные технологии

Информационные и образовательные технологии

<i>№ п/п</i>	<i>Наименование раздела</i>	<i>Виды учебной работы</i>	<i>Формируемые компетенции (указывается код компетенции)</i>	<i>Информационные и образовательные технологии</i>
<i>1</i>	<i>2</i>	<i>3</i>	<i>4</i>	<i>5</i>
1	Введение в нейронные сети. Биологический нейрон и его математическая модель. Классификация нейронных сетей и их свойства. Представление знаний в нейронных сетях.	Лекция Лабораторная работа. Самостоятельная работа	(ПК-3) (ПК-8) (ПК-3) (ОК-4)	Лекция-визуализация с применением проектора Лабораторная работа в Neural Network Toolbox Подготовка к занятию с использованием электронного курса лекций
2	Обучение нейронных сетей. Парадигмы обучения. Обучение, основанное на коррекции ошибок.	Лекция Лабораторная работа Самостоятельная работа	(ПК-3) (ПК-8) (ПК-3) (ОК-4)	Лекция-визуализация с применением проектора Лабораторная работа в Neural Network Toolbox Подготовка к занятию с использованием электронного курса лекций
3	Проблемы практического использования искусственных нейронных сетей и их свойства	Лекция Лабораторная работа Самостоятельная работа	(ПК-3) (ПК-8) (ПК-3) (ОК-4)	Лекция-визуализация с применением проектора Лабораторная работа в Neural Network Toolbox Matlab

				Подготовка к занятию с использованием электронного курса лекций
4	Многослойная нейронная сеть прямого распространения. Алгоритм обучения сети обратного распространения	Лекция Лабораторная работа Самостоятельная работа	(ПК-3) (ПК-8) (ПК-3) (ОК-4)	Лекция-визуализация с применением проектора Лабораторная работа в Neural Network Toolbox Matlab Подготовка к занятию с использованием электронного курса лекций
5	Линейные сети. Персептрон	Лекция Лабораторная работа Самостоятельная работа	(ПК-3) (ПК-8) (ПК-3) (ОК-4)	Лекция-визуализация с применением проектора Лабораторная работа в Neural Network Toolbox Matlab Подготовка к занятию с использованием электронного курса лекций
6	Кластеризация и Классификация данных	Лекция Лабораторная работа Самостоятельная работа	(ПК-3) (ПК-8) (ПК-3) (ОК-4)	Лекция-визуализация с применением проектора Лабораторная работа в Neural Network Toolbox Matlab Подготовка к занятию с использованием электронного курса лекций

7	Сети с самоорганизацией на основе конкуренции	<p><i>Лекция</i></p> <p><i>Лабораторная работа</i></p> <p><i>Самостоятельная работа</i></p>	<p>(ПК-3)</p> <p>(ПК-8) (ПК-3)</p> <p>(ОК-4)</p>	<p>Лекция-визуализация с применением проектора</p> <p>Лабораторная работа в Neural Network Toolbox Matlab</p> <p>Подготовка к занятию с использованием электронного курса лекций</p>
8	Сеть с самоорганизацией корреляционного типа	<p><i>Лекция</i></p> <p><i>Лабораторная работа</i></p> <p><i>Самостоятельная работа</i></p>	<p>(ПК-3)</p> <p>(ПК-8) (ПК-3)</p> <p>(ОК-4)</p>	<p>Лекция-визуализация с применением проектора</p> <p>Лабораторная работа в Neural Network Toolbox Matlab</p> <p>Подготовка к занятию с использованием электронного курса лекций</p>
9	LVQ сети	<p><i>Лекция</i></p> <p><i>Лабораторная работа</i></p> <p><i>Самостоятельная работа</i></p>	<p>(ПК-3)</p> <p>(ПК-8) (ПК-3)</p> <p>(ОК-4)</p>	<p>Лекция-визуализация с применением проектора</p> <p>Лабораторная работа в Neural Network Toolbox Matlab</p> <p>Подготовка к занятию с использованием электронного курса лекций</p>
10	Рекуррентные сети	<p><i>Лекция</i></p> <p><i>Лабораторная работа</i></p>	<p>(ПК-3)</p> <p>(ПК-8) (ПК-3)</p>	<p>Лекция-визуализация с применением проектора</p> <p>Лабораторная работа в Neural</p>

		<i>Самостоятельная работа</i>	(ОК-4)	<i>Network Toolbox Matlab Подготовка к занятию с использованием электронного курса лекций</i>
11	Вероятностная нейронная сеть	<i>Лекция</i>	(ПК-3)	<i>Лекция-визуализация с применением проектора</i>
		<i>Лабораторная работа</i>	(ПК-8) (ПК-3)	<i>Лабораторная работа в Neural Network Toolbox</i>
		<i>Самостоятельная работа</i>	(ОК-4)	<i>Подготовка к занятию с использованием электронного курса лекций</i>

6. Фонд оценочных средств для текущего, рубежного и итогового контролей по итогам освоению дисциплины (модулей)

6.1. Перечень компетенций с указанием этапов их формирования в процессе освоения дисциплины

Перечень компетенций с указанием этапов их формирования в процессе освоения дисциплины представляется в виде таблицы:

<i>№ п/п</i>	<i>Контролируемые разделы дисциплины (модулей)</i>	<i>Код контролируемой компетенции (компетенций)</i>	<i>Наименование оценочного средства</i>
1	Введение в нейронные сети. Биологический нейрон и его математическая модель. Классификация нейронных сетей и их свойства. Представление знаний в нейронных сетях.	(ПК-8) (ПК-3) (ОК-4)	Задача (практическое задание) Коллоквиум (Вопросы по темам/разделам дисциплины)
2	Обучение нейронных сетей. Парадигмы обучения.	(ПК-8) (ПК-3) (ОК-4)	Задача (практическое задание)

	Обучение, основанное на коррекции ошибок.		Коллоквиум (Вопросы по темам/разделам дисциплины)
3	Проблемы практического использования искусственных нейронных сетей и их свойства	(ПК-8) (ПК-3) (ОК-4)	Задача (практическое задание) Коллоквиум (Вопросы по темам/разделам дисциплины) ТЕСТ
4	Многослойная нейронная сеть прямого распространения. Алгоритм обучения сети обратного распространения	(ПК-8) (ПК-3) (ОК-4)	Задача (практическое задание) Коллоквиум (Вопросы по темам/разделам дисциплины)
5	Линейные сети. Персептрон	(ПК-8) (ПК-3) (ОК-4)	Контрольная работа
6	Кластеризация и Классификация данных	(ПК-8) (ПК-3) (ОК-4)	Задача (практическое задание) Коллоквиум (Вопросы по темам/разделам дисциплины) Темы рефератов
7	Сети с самоорганизацией на основе конкуренции	(ПК-8) (ПК-3) (ОК-4)	Задача (практическое задание) Коллоквиум (Вопросы по темам/разделам дисциплины)
8	Сеть с самоорганизацией корреляционного типа	(ПК-8) (ПК-3) (ОК-4)	Задача (практическое задание) Коллоквиум (Вопросы по темам/разделам дисциплины)
9	LVQ сети	(ПК-8) (ПК-3) (ОК-4)	Задача (практическое задание) Коллоквиум (Вопросы по темам/разделам дисциплины)

10	Рекуррентные сети	(ПК-8) (ПК-3) (ОК-4)	Контрольная работа
11	Вероятностная нейронная сеть	(ПК-8) (ПК-3) (ОК-4)	Задача (практическое задание) Коллоквиум (Вопросы по темам/разделам дисциплины)

6.2. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности

Методические материалы составляют систему текущего, рубежного и итогового (экзамена) контролей освоения дисциплины (модулей), закрепляют виды и формы текущего, рубежного и итогового контролей знаний, сроки проведения, а также его сроки и формы проведения (устный экзамен, письменный экзамен и т.п.). В системе контроля указывается процедура оценивания результатов обучения, при использовании балльно-рейтинговой системы приводится таблица с баллами и требованиями к пороговым значениям достижений по видам деятельности обучающихся; показывается механизм получения оценки (из чего складывается оценка по дисциплине (модулю)).

Текущий контроль осуществляется в виде опроса, участие в дискуссии на семинаре, выполнение самостоятельной работы и других видов работ, указанных в УМК, а также посещаемости студентов занятий - оценивается до 80 баллов.

Рубежный контроль (сдача модулей) проводится преподавателем и представляет собой письменный контроль, либо компьютерное тестирование знаний по теоретическому и практическому материалу. Контрольные вопросы рубежного контроля включают полный объем материала части дисциплины (модулей), позволяющий оценить знания, обучающихся по изученному материалу и соответствовать УМК дисциплины, которое оценивается до 20 баллов.

Итоговый контроль (экзамен) знаний принимается по экзаменационным билетам, включающий теоретические вопросы и практическое задание, и оценивается до 20 баллов.

Форма контроля	Срок отчетности	Макс. количество баллов	
		За одну работу	Всего
<i>Текущий контроль:</i>			
- Прием лабораторных работ	1,2,3,4,5,6,7,8 недели	8 баллов	До 40 баллов
- опрос	1, ,2,3,4,5,6,7,8 недели	6 баллов	До 30 баллов
- посещаемость	1, ,2,3,4,5,6,7,8 неделя	2 балла	10 баллов

<i>Рубежный контроль: (сдача модуля)</i>	<i>8 неделя</i>	<i>100%×0,2=20 баллов</i>
<i>Итого за I модуль</i>		<i>До 100 баллов</i>

<i>Форма контроля</i>	<i>Срок отчетности</i>	<i>Макс. количество баллов</i>	
		<i>За одну работу</i>	<i>Всего</i>
<i>Текущий контроль: - Прием лабораторных работ</i>	<i>9,10,11,12,13,14,15 недели</i>	<i>10 баллов</i>	<i>До 40 баллов</i>
<i>-опрос</i>	<i>9,10,11,12,13,14,15 недели</i>	<i>6 баллов</i>	<i>До 30 баллов</i>
<i>- посещаемость</i>	<i>9,10,11,12,13,14,15 недели</i>	<i>2 балла</i>	<i>10 баллов</i>
<i>Рубежный контроль: (сдача модуля)</i>	<i>15 неделя</i>	<i>100%×0,2=20 баллов</i>	
<i>Итого за II модуль</i>		<i>До 100 баллов</i>	

<i>Форма контроля</i>	<i>Срок отчетности</i>	<i>Макс. количество баллов</i>	
		<i>За одну работу</i>	<i>Всего</i>
<i>Текущий контроль: - Прием лабораторных работ</i>	<i>1,2,3,4,5,6,7,8 недели</i>	<i>8 баллов</i>	<i>До 40 баллов</i>
<i>-опрос</i>	<i>1, ,2,3,4,5,6,7,8 недели</i>	<i>6 баллов</i>	<i>До 30 баллов</i>
<i>- посещаемость</i>	<i>1, ,2,3,4,5,6,7,8 неделя</i>	<i>2 балла</i>	<i>10 баллов</i>
<i>Рубежный контроль: (сдача модуля)</i>	<i>8 неделя</i>	<i>100%×0,2=20 баллов</i>	
<i>Итого за I модуль</i>		<i>До 100 баллов</i>	

<i>Форма контроля</i>	<i>Срок отчетности</i>	<i>Макс. количество баллов</i>	
		<i>За одну работу</i>	<i>Всего</i>
<i>Текущий контроль:</i>			

- Прием лабораторных работ	9,10,11,12,13,14,15 недели	10 баллов	До 40 баллов
- опрос	9,10,11,12,13,14,15 недели	6 баллов	До 30 баллов
- посещаемость	9,10,11,12,13,14,15 недели	2 балла	10 баллов
<i>Рубежный контроль:</i> (сдача модуля)	15 неделя	100%×0,2=20 баллов	
<i>Итого за II модуль</i>			До 100 баллов

Экзаменатор выставляет по результатам балльной системы в семестре экзаменационную оценку без сдачи экзамена, набравшим суммарное количество баллов, достаточное для выставления оценки от 55 и выше баллов – автоматически (при согласии обучающегося).

Полученный совокупный результат (максимум 100 баллов) конвертируется в традиционную шкалу:

Рейтинговая оценка (баллов)	Оценка экзамена
От 0 - до 54	неудовлетворительно
от 55 - до 69 включительно	удовлетворительно
от 70 – до 84 включительно	хорошо
от 85 – до 100	отлично

6.3.Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

Текущий контроль (0 - 80 баллов)

При оценивании посещаемости, опроса и приема лабораторных работ из расчета на одну неделю учитываются:

- посещаемость (2 балла одно занятие (10 баллов за модуль)
- степень раскрытия содержания материала (2.8 балла одно занятие (14 баллов за модуль);
- изложение материала (грамотность речи, точность использования терминологии и символики, логическая последовательность изложения материала (2.8 балла одно занятие (14 баллов за модуль);
- знание теории изученных вопросов (2.8 балла одно занятие (14 баллов за модуль);
- сформированность и устойчивость используемых при ответе умений и навыков (2.8 балла одно занятие (14 баллов за модуль);
- точность решения задачи (2.8 балла одно занятие (14 баллов за модуль).

Рубежный контроль (0 – 20 баллов)

При оценивании контрольной работы учитывается:

- полнота выполненной работы (задание выполнено не полностью и/или допущены две и более ошибки или три и более неточности) – 8 баллов;
- обоснованность содержания и выводов работы (задание выполнено полностью, но обоснование содержания и выводов недостаточны, но рассуждения верны) – 14 баллов;
- работа выполнена полностью, в рассуждениях и обосновании нет пробелов или ошибок, возможна одна неточность - 17 баллов.
- работа выполнена полностью, в рассуждениях и обосновании нет пробелов или ошибок - 20 баллов.

При оценивании теста учитывается:

- полнота выполненной работы (задание выполнено не полностью и/или допущены две и более ошибки или три и более неточности) – до 20 баллов;

Итоговый контроль (экзаменационная сессия) - ИК = Бср × 0,8 + Бэкз × 0,2

При проведении итогового контроля обучающийся должен ответить на 3 вопроса (два вопроса теоретического характера и один вопрос практического характера).

При оценивании ответа на вопрос теоретического характера учитывается:

- теоретическое содержание не освоено, знание материала носит фрагментарный характер, наличие грубых ошибок в ответе (2 балла);
- теоретическое содержание освоено частично, допущено не более двух-трех недочетов (5 баллов);
- теоретическое содержание освоено почти полностью, допущено не более одного-двух недочетов, но обучающийся смог бы их исправить самостоятельно (8 баллов);
- теоретическое содержание освоено полностью, ответ построен по собственному плану (10 баллов).

При оценивании ответа на вопрос практического характера учитывается:

- ответ содержит менее 20% правильного решения (3 балла);
- ответ содержит 21-89 % правильного решения (7 баллов);
- ответ содержит 90% и более правильного решения (10 баллов).

6.4. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности.

Раздел УМК включает образцы оценочных средств, примерные перечни вопросов и заданий в соответствии со структурой дисциплины и системой контроля.

Контрольные вопросы

1. Что такое нейронные сети (НС)? Что дает моделирование НС? Проблемы, возникающие при моделировании. Свойства биологических и искусственных НС. Способы реализации нейросетей.
2. Место НС среди других методов решения задач. Типы задач, решаемых нейронными сетями. Недостатки и ограничения НС.
3. Биологический нейрон. Структура, функции.
4. Формальный нейрон. Виды функций активации. Ограниченность модели формального нейрона.

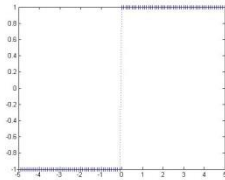
5. Многослойный перцептрон. Структура, алгоритм работы. Этапы решения задачи с помощью НС.
6. Формализация условий задачи для НС. Примеры. Подготовка входных и выходных данных. Выбор количества слоев.
7. Обучение однослойного перцептрона. Выбор шагов по W, Θ .
8. Проблема "исключающего ИЛИ" и ее решение.
9. Перцептронная представляемость.
10. Метод обратного распространения ошибки.
11. Паралич сети. Выбор шага по параметрам. Локальные минимумы. Временная неустойчивость.
12. Примеры применения перцептронов.
13. Динамическое добавление нейронов. Способность НС к обобщению.
14. Обучение без учителя. Сеть с линейным поощрением.
15. Задача классификации. Сеть Кохонена.
16. Обучение слоя Кохонена. Метод выпуклой комбинации. Примеры обучения.
17. Режимы работы сети Кохонена. Применение для сжатия данных.
18. Сеть встречного распространения. Схема, обучение, свойства.
19. Генетические алгоритмы для обучения НС. Положительные качества и недостатки.
20. Послойность сети и матричное умножение. Расчет градиента квадратичной формы с помощью НС. Выбор начальной точки и длины шага.
21. Сети с обратными связями. Сеть Хопфилда. Вычислительная энергия и ее минимизация.
22. Этапы решения задачи сетью Хопфилда. Устойчивость, сходимость к эталонам.
23. Соотношение стабильности пластичности при запоминании. Сеть АРТ 1. Структура, описание элементов сети.
24. Работа сети АРТ 1. Запоминание и классификация векторов сетью.
25. Метод имитации отжига.

Тематика рефератов

1. Формальный нейрон. Виды функций активации. Ограниченность модели формального нейрона.
2. Многослойный перцептрон. Структура, алгоритм работы. Этапы решения задачи с помощью НС.
3. Формализация условий задачи для НС. Примеры. Подготовка входных и выходных данных. Выбор количества слоев.
4. Обучение однослойного перцептрона. Выбор шагов по W, Θ .
5. Проблема "исключающего ИЛИ" и ее решение.
6. Перцептронная представляемость.
7. Метод обратного распространения ошибки.
8. Паралич сети. Выбор шага по параметрам. Локальные минимумы. Временная неустойчивость.
9. Примеры применения перцептронов.
10. Динамическое добавление нейронов. Способность НС к обобщению.
11. Обучение без учителя. Сеть с линейным поощрением.
12. Задача классификации. Сеть Кохонена.
13. Обучение слоя Кохонена. Метод выпуклой комбинации. Примеры обучения.
14. Режимы работы сети Кохонена. Применение для сжатия данных.
15. Сеть встречного распространения. Схема, обучение, свойства.

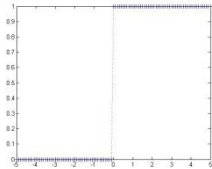
Тест

1. График какой функции изображен на рисунке



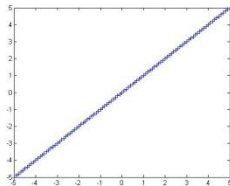
- a. Пороговая
b. Знаковая(!)

2. График какой функции изображен на рисунке (пороговая)



- a. Пороговая(!)
b. Знаковая

3. Какой формулой описывается линейная функция, график которой представлен на рисунке.



- a. $\varphi(v) = v(!)$
b. $\varphi(v) = \begin{cases} 1, & v > 0 \\ -1, & v \leq 0 \end{cases}$

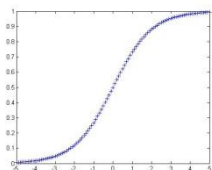
4. Как называется активационная функция формула которой $\varphi(v) = \frac{1}{1+e^{-v}}$

- a. Сигмоидальная (!)
b. гиперболический тангенс

5. Как называется активационная функция формула которой $\varphi(v) = \frac{e^v - e^{-v}}{e^v + e^{-v}}$

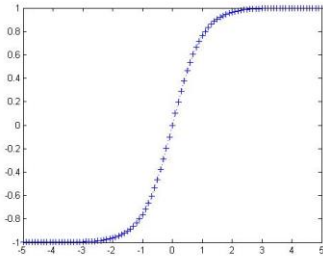
- a. сигмоидальная
b. гиперболический тангенс (!)

6. Какой функцией в Matlab описывается функция, график которой представлен на рисунке.



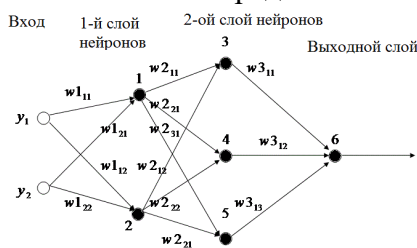
- a. $\text{logsig}(v) (!)$
b. $\text{tansig}(v)$

7. Какой функцией в Matlab описывается функция, график которой представлен на рисунке.



- a. $\text{logsig}(v)$
 - b. $\text{tansig}(v)(!)$
8. Аксон – это выходной или входной отросток нейрона?
 - a. Выходной(!)
 - b. Входной
 9. Какие нейроны называются входными нейронами:
 - a. нейроны, на которые подается входной вектор, кодирующий входное воздействие или образ внешней среды. (!)
 - b. нейроны, выходные значения которых представляют выход сети.
 - c. нейроны, находящиеся в скрытых слоях нейронной сети.
 10. Искусственные нейронные сети, каждый нейрон которой передает свой выходной сигнал остальным нейронам, в том числе и самому себе, называются:
 - a. Полносвязными сетями
 - b. Слабосвязными сетями (!)
 11. В *многослойных сетях* нейроны объединяются в слои. Как называются слои находящиеся между входным и выходным слоем нейронов.
 - a. Скрытый слой (!)
 - b. Невидимый слой
 - c. Входной слой
 - d. Выходной слой

12. Как называется сеть представленная на рисунке



- a. Многослойная сеть прямого распространения(!)
 - b. Многослойная сеть с обратными связями
13. Априорной называется информация:
 - a. Наблюдения за окружающим миром (измерения), полученные с помощью сенсоров, адаптированных для конкретных условий, в которых должна функционировать данная нейронная сеть.
 - b. Известное состояние окружающего мира, представленное имеющимися в наличии достоверными фактами(!)

14. Примеры для обучения нейронной сети могут быть *маркированными* и *немаркированными*. Как называется пример когда *входному сигналу* соответствует *желаемый отклик*

a. *маркированными(!)*

b. *немаркированными*

15. Нейрон получает входной сигнал x_m , где $m = 4$, блочная диаграмма данной



модели представлена на рисунке

$x_1 = 0, x_2 = 2, x_3 = 3, x_4 = -3$.

Соответствующие весовые коэффициенты нейрона равны

$w_{11} = 1, w_{12} = -1, w_{13} = 0.3,$

$w_{14} = 1$

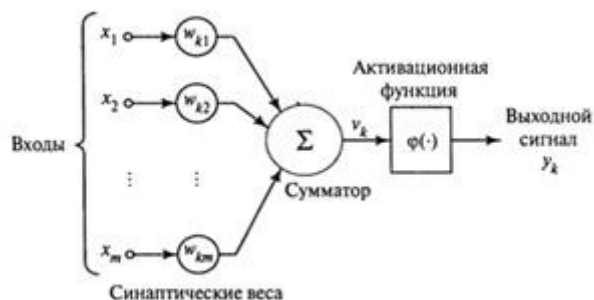
Вычислить выходное значение нейрона, модель которого описывается пороговой функцией активации. Предполагается, что порог отсутствует.

a. $0 (!)$

b. 1

c. 0.5

16. Нейрон получает входной сигнал x_m , где $m = 3$, блочная диаграмма данной



модели представлена на рисунке

$x_1 = 1, x_2 = 0.4, x_3 = 3,$

Соответствующие весовые коэффициенты нейрона равны

$w_{11} = 3, w_{12} = -1, w_{13} = 0.3,$

Вычислить выходное значение нейрона, модель которого описывается линейной функцией активации. Предполагается, что порог отсутствует.

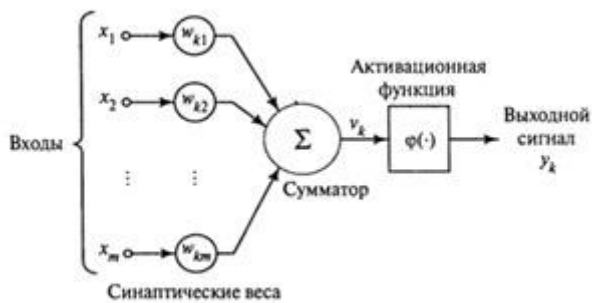
a. $3.5 (!)$

b. 3.7

c. 2.4

d.

17. Нейрон получает входной сигнал x_m , где $m = 2$, блочная диаграмма данной



модели представлена на рисунке

$$x_1 = 1, x_2 = 2$$

Соответствующие весовые коэффициенты нейрона равны

$$w_{11} = 0, w_{12} = -1$$

Вычислить выходное значение нейрона, модель которого описывается знаковой функцией активации. Предполагается, что порог отсутствует.

- 1 (!)
- 1
- 0

18. Какие функции выполняет входной слой нейронной сети?

- Транслирует сигнал на выходной слой нейронной сети
- Удаляет "шум" из сигнала
- Передает входной вектор сигналов на первый слой (!)
- Вычисляет производную для алгоритма обратного распространения ошибки

19. Сколько скрытых слоев в архитектуре 2-2-4-1

- 1 (!)
- 4
- 2

20. Сколько нейронов в скрытом слое в архитектуре 2-3-5-1

- 5(!)
- 3
- 1

21. Сколько скрытых слоев в архитектуре 2-2-4-3-1

- 1
- 4
- 2(!)

22. Сколько нейронов в первом слое в архитектуре 2-3-5-1

- 5
- 3(!)
- 2

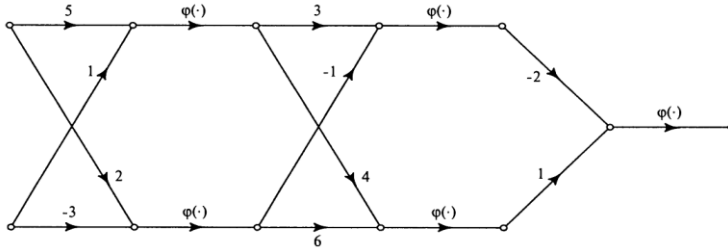
23. Сколько нейронов в первом слое в архитектуре 4-7-5-1

- 5
- 7(!)
- 4

24. Сколько элементов во входном слое в архитектуре 4-7-5-1

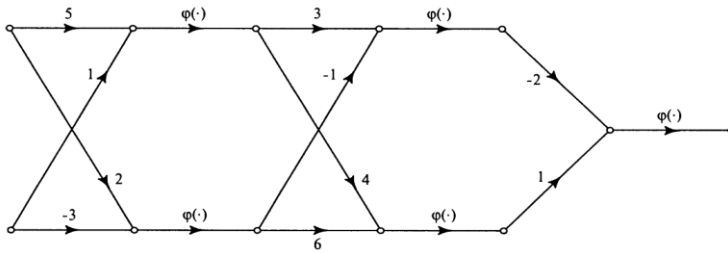
- a. 5
- b. 4(!)
- c. 7

25. Чему равны весовые коэффициенты второго слоя



- a. Весовые коэффициенты второго слоя равны 3,-1,4,6.(!)
- b. Весовые коэффициенты второго слоя равны -2, 1.
- c. Весовые коэффициенты второго слоя равны 5,1,2,-3.

26. Чему равны весовые коэффициенты первого слоя



- a. Весовые коэффициенты первого слоя равны 3,-1,4,6.
- b. Весовые коэффициенты первого слоя равны -2, 1.
- c. Весовые коэффициенты первого слоя равны 5,1,2,-3. (!)

27. Укажите чему равен диапазон значений сигмоидальной функции активации

- a. $(-\infty, \infty)$
- b. $(-1, 1)$
- c. $(0, 1)$!

28. Укажите чему равен диапазон значений полулинейной функции активации

- a. $(0, \infty)$!
- b. $(-1, 1)$
- c. $(0, 1)$

29. Укажите чему равен диапазон значений знаковой функции активации

- a. $(0, \infty)$
- b. $(-1, 1)$!
- c. $(0, 1)$

30. Укажите чему равен диапазон значений полулинейной с насыщением функции активации

- a. $(0, \infty)$
- b. $(-1, 1)$
- c. $(0, 1)$!

Контрольная работа

Нарисовать граф прохождения сигнала по сети прямого распространения вида, заданного вариантом (см. таблицу №1). Описать отображение вход-выход.

Таблица №1

№	Вид сети	Функции активации	Входное значение	Весовые коэф-ты 1 слоя	Весовые коэф-ты 2 слоя	Весовые коэф-ты 3 слоя
1	2-3-3-1	Пороговая Сигмоидальная	[1,1]	[1,2,1,3,-5,2]	[6,3,-2,1,-1, -0.5,-1,3,7,]	[2,1,3]
2	3-3-2-1	Сигмоидальная Радиальная базисная	[1,2,3]	[6,3,-2,1,-1, -0.5,-1,3,7,]	[1,2,1,3,-5,2]	[1,1]
3	2-3-2-1	Линейная Сигмоидальная	[4,6]	[2,1,3,-0.3,- 4,3]	[1,2,1,3,-5,2]	[-2,1]
4	2-2-3-1	Радиальная базисная Сигмоидальная	[-2,1]	[-4,2]	[1,2,1,3,-5,2]	[2,1,3]
5	3-2-4-1	Линейная с насыщением Сигмоидальная	[0,-1,5]	[2,1,3,-0.3,- 4,3]	[6,3,-2,1,-1, -0.5,-1,3]	[2,1,3,1]
6	2-3-4-1	Полулинейная Сигмоидальная	[-4,2]	[1,2,1,3,-5,2]	[6,3,-2,1,-1, -0.5, -1,3,7,1,2,4]	[2,1,3,1]
7	3-4-2-1	Гиперболический тангенс Сигмоидальная	[3,-3,1]	[6,3,-2,1,-1, -0.5, -1,3,7,1,2,4]	[6,3,-2,1,-1, -0.5,-1,3,7,]	[-2,1]
8	2-2-3-1	Полулинейная с насыщением Сигмоидальная	[1,-9]	[-4,2]	[1,2,1,3,-5,2]	[2,1,3]
9	3-3-4-1	Знаковая Сигмоидальная	[-3,-2,4]	[6,3,-2,1,-1, -0.5,-1,3,7,]	[6,3,-2,1,-1, -0.5, -1,3,7,1,2,4]	[2,1,3,1]
10	2-4-3-1	Полулинейная с насыщением Сигмоидальная	[5,1]	[6,3,-2,1,-1, -0.5,-1,3]	[6,3,-2,1,-1, -0.5, -1,3,7,1,2,4]	[2,1,3]
11	2-3-3-1	Пороговая Сигмоидальная	[1,1]	[2,1,3,-0.3,- 4,3]	[6,3,-2,1,-1, -0.5,-1,3,7,]	[2,1,3]
12	3-3-2-1	Сигмоидальная Радиальная базисная	[1,2,3]	[6,3,-2,1,-1, -0.5,-1,3,7,]	[2,1,3,-0.3,- 4,3]	[-2,1]
13	2-3-2-1	Линейная Сигмоидальная	[4,6]	[2,1,3,-0.3,- 4,3]	[2,1,3,-0.3,- 4,3]	[-2,1]
14	2-2-3-1	Радиальная базисная Сигмоидальная	[-2,1]	[2,1,3,1]	[2,1,3,-0.3,- 4,3]	[2,1,3]
15	3-2-4-1	Линейная с насыщением Сигмоидальная	[0,-1,5]	[2,1,3,-0.3,- 4,3]	[6,3,-2,1,-1, -0.5,-1,3]	[2,1,3,1]
16	2-3-4-1	Полулинейная Сигмоидальная	[-4,2]	[1,2,1,3,-5,2]	[6,3,-2,1,-1, -0.5,	[2,1,3,1]

					-1,3,7,1,2,4]	
17	3-4-2-1	Гиперболический тангенс Сигмоидальная	[3,-3,1]	[6,3,-2,1,-1,-0.5,-1,3,7,1,2,4]	[6,3,-2,1,-1,-0.5,-1,3,7,]	[-2,1]
18	2-2-3-1	Полулинейная с насыщением Сигмоидальная	[1,-9]	[-4,2]	[1,2,1,3,-5,2]	[2,1,3]
19	3-3-4-1	Знаковая Сигмоидальная	[-3,-2,4]	[6,3,-2,1,-1,-0.5,-1,3,7,]	[6,3,-2,1,-1,-0.5,-1,3,7,1,2,4]	[2,1,3,1]
20	2-4-3-1	Полулинейная с насыщением Сигмоидальная	[5,1]	[6,3,-2,1,-1,-0.5,-1,3]	[6,3,-2,1,-1,-0.5,-1,3,7,1,2,4]	[2,1,3]
21	2-3-3-1	Пороговая Сигмоидальная	[1,1]	[2,1,3,-0.3,-4,3]	[6,3,-2,1,-1,-0.5,-1,3,7,]	[2,1,3]
22	3-3-2-1	Сигмоидальная Радиальная базисная	[1,2,3]	[6,3,-2,1,-1,-0.5,-1,3,7,]	[2,1,3,-0.3,-4,3]	[-2,1]
23	2-3-2-1	Линейная Сигмоидальная	[4,6]	[2,1,3,-0.3,-4,3]	[2,1,3,-0.3,-4,3]	[-2,1]
24	2-2-3-1	Радиальная базисная Сигмоидальная	[-2,1]	[2,1,3,1]	[2,1,3,-0.3,-4,3]	[2,1,3]
25	3-2-4-1	Линейная с насыщением Сигмоидальная	[0,-1,5]	[2,1,3,-0.3,-4,3]	[6,3,-2,1,-1,-0.5,-1,3]	[2,1,3,1]
26	2-3-4-1	Полулинейная Сигмоидальная	[-4,2]	[1,2,1,3,-5,2]	[6,3,-2,1,-1,-0.5,-1,3,7,1,2,4]	[2,1,3,1]
27	3-4-2-1	Гиперболический тангенс Сигмоидальная	[3,-3,1]	[6,3,-2,1,-1,-0.5,-1,3,7,1,2,4]	[6,3,-2,1,-1,-0.5,-1,3,7,]	[-2,1]
28	2-3-4-1	Полулинейная Сигмоидальная	[-4,2]	[1,2,1,3,-5,2]	[6,3,-2,1,-1,-0.5,-1,3,7,1,2,4]	[2,1,3,1]
29	3-4-2-1	Гиперболический тангенс Сигмоидальная	[3,-3,1]	[6,3,-2,1,-1,-0.5,-1,3,7,1,2,4]	[6,3,-2,1,-1,-0.5,-1,3,7,]	[-2,1]
30	2-2-3-1	Полулинейная с насыщением Сигмоидальная	[1,-9]	[-4,2]	[1,2,1,3,-5,2]	[2,1,3]

Контрольная работа

Цель работы: изучение нейронных сетей для задач аппроксимации функций двух переменных

Общие сведения

Для построения графика от трех переменных использовать функцию plot3

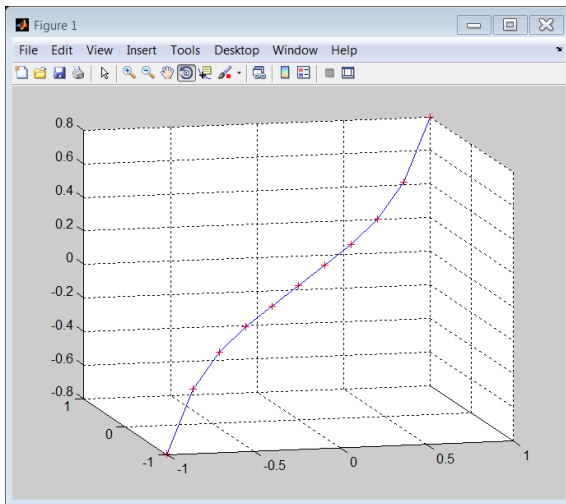


рис. 1

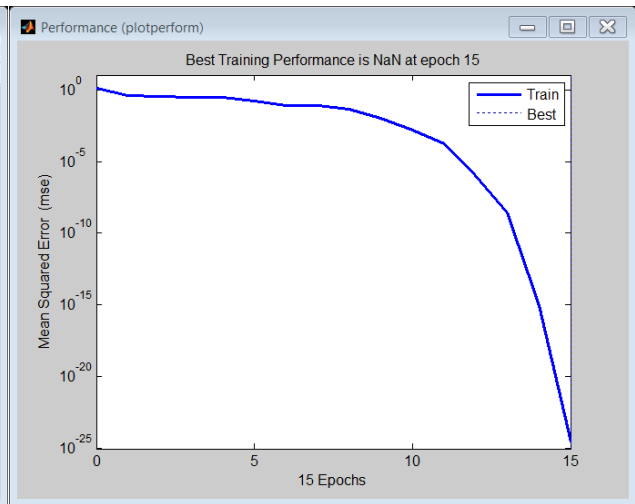


рис.2

- Для заданного преподавателем варианта (таблица) аппроксимировать функцию.
 - Проанализировать какое количество нейронов решает поставленную задачу и за какое количество эпох
 - при обучении нейронной сети использовать алгоритмы обучения `trainlm` или `traingd`
- Разработать алгоритм создания и моделирования нейронной сети.
- Реализовать разработанный алгоритм в системе MATLAB.
- Вывести параметры созданной и обученной нейронной сети (веса и смещение)
- Проверить правильность работы сети для последовательности входных векторов.
- Построить график, аналогичный представленному на рис.1 и рис.2, для своих исходных данных.
- Нарисовать модель нейронной сети (с указанием весовых коэффициентов, функций активаций)
- Составить отчет.

Вариант	Функция	Вариант	Функция
1	$Z = x.^3 + 0.1y$	17	$Z = x.^3 + 0.2y$
2	$Z = 0.3 * x.^3 + 0.4y$	18	$Z = 0.3 * x.^3 + y.^2 - 0.2 * x$
3	$Z = 0.1 * x.^3 + 0.2y$	19	$Z = \sin(2 * \pi * y)$
4	$Z = 0.3 * x.^3 + 0.3 * y.^2$	20	$Z = \cos(2 * \pi * x) + y.^3 + 0.2 * x$
5	$Z = 0.4 * x.^2 + 0.3y$	21	$Z = \cos(2 * \pi * x) + y.^3 + 0.2$
6	$Z = 0.4 * x.^2 + 0.4 * y.^3$	22	$Z = 0.4 * x.^2 + \cos(3 * \pi * y)$
7	$Z = 0.1 * x.^2 + 0.5y$	23	$Z = 0.5 * \pi * x.^2 + 0.1y$
8	$Z = 0.4 * x.^5 + 0.4 * y$	24	$Z = 0.4 * x.^5 - \sin(2 * \pi * y)$
9	$Z = 0.1 * x.^3 + \sin(2 * \pi * y)$	25	$Z = 0.1 * x.^3 - \sin(2 * \pi * y)$
10	$Z = 0.1 * x.^3 - \cos(0.2 * \pi * y)$	26	$Z = 0.1 * x.^3 + \sin(2 * \pi * y)$
11	$Z = 0.2 * x.^3 - 0.1y$	27	$Z = \cos(2 * \pi * x) + 0.2 * y.^3 - 0.1$
12	$Z = 0.5 * x.^3 - 0.2 * y.^2$	28	$Z = \cos(2 * \pi * x) + 0.2 * y.^2$
13	$Z = 0.4 * x.^2 - 0.6y$	29	$Z = \cos(0.4 * \pi * x) + 0.4 * y.^2 - 0.6$
14	$Z = 0.3 * x.^2 - 0.4 * y.^3$	30	$Z = 0.3 * x.^2 - 0.4 * y.^3$

15	$Z=0.2*x.^2+0.4y$	31	$Z= x.^3+0.1y$
16	$Z=0.1*x.^4+0.3*y$	32	$Z= 0.3*x.^3+0.4y$
35	$Z= 0.1*x.^2+0.1y$	33	$Z=0.2*x.^3+0.2y$
36	$Z=0.7*x.^2+0.2*y.^3$	34	$Z=0.4*x.^3+0.7*y.^2$

Самостоятельная работа студентов

Создать таблицу основных функций NNT Matlab для работы с нейронными сетями

	<i>СЕТИ С ПРЯМОЙ ПЕРЕДАЧЕЙ СИГНАЛА</i>
New networks	1.1.1.1 Формирование архитектуры сети
Using networks	Работа с нейронной сетью
	Моделирование сети
	Инициализация сети
	Адаптация сети
	Обучение сети
Weight functions	Функции взвешивания
Net input functions	Функции накопления
Transfer functions	Функции активации
Initialization functions	Функции инициализации
	Инициализация слоев
	Инициализация весов и смещений
	Инициализация нулевых весов и смещений
Performance functions	Функции оценки качества сети
Learning functions	Функции настройки параметров
Adapt functions	Функции адаптации
Training functions	Функции обучения
	Алгоритмы обучения
	Графические функции

7. Учебно-методическое и информационное обеспечение дисциплины

7.1.Список источников и литературы

Основные учебники

1. Оссовский С. Нейронные сети для обработки информации. – М.: Финансы и статистика, 2002.
2. Медведев В.С., Потемкин В.Г. Нейронные сети. Matlab 6. М.: Диалог МИФИ, 2002.
3. Роберт Каллан. Основные концепции нейронных сетей. М.: Издательский дом «Вильямс», 2001.
4. Суворцев И.С., Клюкин В.И., Пивоварова Р.П. Нейронные сети. - Воронеж: ВГУ,1994.
5. Уоссермен Ф. Нейрокомпьютерная техника: теория и практика. М.:Мир,1992.

Дополнительная литература

1. Горбань А.Н. и др. Нейроинформатика.- Электронная публикация.
2. Мкртчян С.О. Нейроны и нейронные сети. (Введение в теорию формальных нейронов)— М.:Энергия,1971.
3. А.Н. Горбань, Д.А. Россиев. Нейронные сети на персональном компьютере. - Новосибирск, Наука, 1996.
4. Гилл Ф., Мюррей У., Райт М. Практическая оптимизация.М.:Мир,1985.
5. Лоскутов А.Ю., Михайлов А.С. Введение в синергетику. М.: Наука. Гл. ред. физ. мат. лит., 1990.— 272 с.
6. Куссуль Э.М. Ассоциативные нейроподобные структуры. - Киев, Наукова думка, 1990.
7. Нейрокомпьютеры и интеллектуальные роботы. Под ред. Н.М.Амосова. – Киев, Наукова думка, 1991.

7.2.Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимый для освоения дисциплины (модулей)



1. <https://www.intuit.ru/search>
2. <https://www.twirpx.com/> Библиотека все для студента
3. <https://uk.sagepub.com/en-gb/asi/home>
4. <https://uk.sagepub.com/en-gb/asi/sage-premier>
5. <https://www.nejm.org/>
6. <https://uk.sagepub.com/en-gb/asi/imeche>
7. <http://global.oup.com/?cc=kg>
8. <https://www.cambridge.org>
9. <https://www.intellectbooks.co.uk/journals/index/>
10. <http://iopscience.iop.org/journalList>
11. <https://royalsociety.org/journals/>
12. <https://www.elibrary.imf.org/?redirect=true>
13. <https://www.elgaronline.com/page/70/journals>
14. <http://www.dukejournals.org/>
15. <http://www.iprbookshop.ru/>
16. <http://kyrlibnet.kg/ru/>

8. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся

8.1. Планы практических (семинарских) и лабораторных занятий. Методические указания по организации и проведению

- Лабораторная работа №1

Тема «Классификация типов стекол по их физико-химическим свойствам в судебно-следственной практике (UCI machine-learning database collection (Murphy & Aha, 1995))».

Форма проведения: Выполнение лабораторной работы с помощью программного средства NNT Matlab; Имеются данные по 214 образцам (крупницам и осколкам) стекла со следующими параметрами:

- 1 - RI - показатель преломления
- 2 - Na - весовая доля оксида(оксидов) натрия %
- 3 - Mg - весовая доля оксида(оксидов) марганца %
- 4 - Al - весовая доля оксида(оксидов) алюминия %
- 5 - Si - весовая доля оксида(оксидов) кремния %
- 6 - K - весовая доля оксида(оксидов) калия %
- 7 - Ca - весовая доля оксида(оксидов) кальция %
- 8 - Ba - весовая доля оксида(оксидов) бария %
- 9 - Fe - весовая доля оксида(оксидов) железа %

Выходная переменная - природа образца (один из 6 типов стекла)

- 1 (WinF) оконное стекло (float)
- 2 (WinNF) оконное стекло (non-float)
- 3 (Veh) автомобильное стекло
- 5 (Con) стеклянные сосуды
- 6 (Tabl) настольное стекло
- 7 (Head) стекло фар автомобиля

Цель: Требуется решить задачу классификации типов стеклянных осколков (типичную в судебно-следственной практике). Для обучения использовать первые 180 примеров, остальные - для тестирования.

Список источников и литературы:

<https://archive.ics.uci.edu/ml/index.php>

Материально-техническое обеспечение занятия.

Компьютерный класс, проектор, программное обеспечение NNT Matlab.

- Лабораторная работа №2

Тема «Определение злокачественности опухоли. Рак молочной железы Висконсин (оригинал)».

Форма проведения: Выполнение лабораторной работы с помощью программного средства NNT Matlab;

Построить диагностическую систему для различения злокачественных опухолей от незлокачественных образований по описанию клеток пробы (Wisconsin Breast Cancer Database).

Входные параметры (принимают значения от 1 до 10) - 9 величин (имеющих медицинскую специфику)

- 1 - толщина слоя
- 2 - степень однородности размеров клеток
- 3 - степень однородности формы клеток
- 4 - степень маргинальности спайки (слипание клеток)

5 - размер эпителиальных одиночных клеток

6 - количество свободных ядер

7 - хроматин

8 - нормальные ядра

9 - митозы

Выходной параметр - класс 0 - доброкачественная, 1 - злокачественная опухоль. Исходные данные (матрицы P, T) записаны в файле. Имеется 699 примеров, которые разбиваются на две группы - первые N примеров используются для обучения многослойного персептрона, остальные для тестирования.

Цель: Требуется построить зависимость ошибки прогноза (тестирования) от N. При каких N можно говорить о "надежном распознавании"?

Список источников и литературы:

[https://archive.ics.uci.edu/ml/datasets/breast+cancer+wisconsin+\(original\)](https://archive.ics.uci.edu/ml/datasets/breast+cancer+wisconsin+(original))

Материально-техническое обеспечение занятия.

Компьютерный класс, проектор, программное обеспечение NNT Matlab.

- Лабораторная работа №3

Тема «Прогноз стоимости недвижимости в районах Бостона (Boston database, DELVE benchmark collection)».

Форма проведения: Выполнение лабораторной работы с помощью программного средства NNT Matlab; В данной задаче требуется построить прогностическую систему для оценки стоимости жилья в Бостоне (США), основываясь на следующих 13 входных параметрах:

1 CRIM - уровень преступности на душу населения

2 ZN - доля жилых массивов на площади 25,000 кв. футов.

3 NDUS - доля предприятий, не связанных с розничной торговлей

4 CHAS - близость к реке Charles River (1 если район граничит с рекой, 0 - в противном случае)

5 NOX - концентрация окисей азота (в долях 1/10 миллионов)

6 RM - среднее число комнат в жилище

7 AGE - доля частных владений, построенных до 1940 года

8 DIS - взвешенное расстояние до 5 центров сосредоточения работы в Бостона

9 RAD - индекс доступности радиальных автострад

10 TAX - полный налог на недвижимость в расчете на \$10,000

11 PTRATIO - доля школьных учителей в районе

12 B - коэффициент $1000(Bk - 0.63)^2$, где Bk - доля чернокожего населения

13 LSTAT - процент населения ниже черты бедности

Выходной параметр

14 - MEDV средняя цена частного жилья в \$1000's

База данных содержит 506 записей, из которых первые 400 используются для обучения, остальные - для тестирования.

Цель: В данной задаче требуется построить прогностическую систему для оценки стоимости жилья в Бостоне (США). Требуется построить нейросетевую классификационную систему (на выходном слое используется линейная переходная функция) с наименьшей ошибкой тестирования.

Список источников и литературы:

<https://www.cs.toronto.edu/~delve/data/boston/bostonDetail.html>

<http://lib.stat.cmu.edu/datasets/boston>

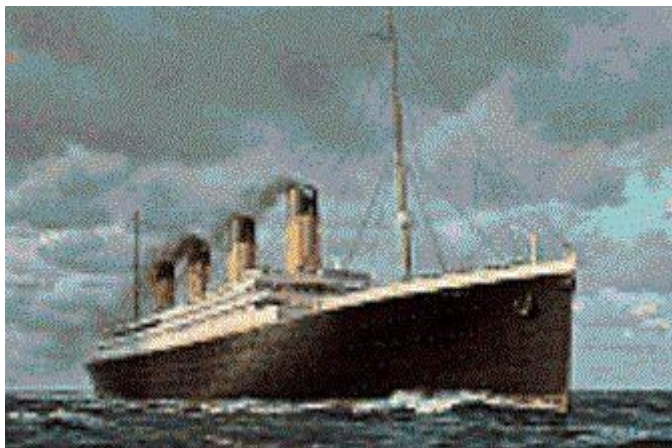
Материально-техническое обеспечение занятия.

Компьютерный класс, проектор, программное обеспечение NNT Matlab.

- Лабораторная работа №4

Тема «Оценка влияния социальных факторов на выживание пассажиров потерпевшего аварию судна "Титаник" (Titanic database, DELVE benchmark collection)».

Форма проведения: Выполнение лабораторной работы с помощью программного средства NNT Matlab;



В этой задаче рассматривается печальная статистика трагедии 14 апреля 1912 года с океанским лайнером "Титаник". Среди 2201 пассажира судна 711 удалось спастись. Требуется исследовать статистическое влияние на выживаемость людей следующих факторов:

1 CLASS - класс каюты 0-1й, 1 - 2й, 2 - 3й, 3 - член команды

2 AGE - возраст 0 - ребенок, 1 - взрослый

3 SEX - пол 0 - женщина, 1 - мужчина

Выходной параметр

4 SURVIVED - 0 - погиб, 1 - выжил.

Цель: Требуется исследовать статистическое влияние на выживаемость людей **Список источников и литературы:**

<https://www.cs.toronto.edu/~delve/data/datasets.html>

<https://www.cs.toronto.edu/~delve/data/titanic/desc.html>

Материально-техническое обеспечение занятия.

Компьютерный класс, проектор, программное обеспечение NNT Matlab.

- Лабораторная работа №5

Тема «Классификация данных, полученных со спутника Landstat (Satimage database, ELENA Project, STATLOG ESPRIT project)».

Форма проведения: Выполнение лабораторной работы с помощью программного средства NNT Matlab;

Исходные данные получены из наблюдений различных участков земной поверхности в различных спектральных диапазонах с использованием радаров (36 входных параметров - 4 спектральных диапазона для матрицы из 9 пикселей) спутником Landstat NASA. База данных содержит 6435 записей. Выход нейросети принимает одно из следующих значений:

1 - глинистая (красная) почва

2 - посевы технических культур (хлопок)

3 - серая почва

4 - увлажненная серая почва

5 - покос (стерня)

6 - смесь всех перечисленных покрытий (в базе данных отсутствует)

7 - очень влажная серая почва

Цель: Требуется построить нейросетевую классификационную систему (на выходном слое используется линейная переходная функция) с наименьшей ошибкой тестирования. Для обучения использовать первые 5000 записей, остальные для тестирования. Каковы преимущества и недостатки кодирования 7 классов одним выходным нейроном?

Список источников и литературы:

[https://archive.ics.uci.edu/ml/datasets/Statlog+\(Landsat+Satellite\)](https://archive.ics.uci.edu/ml/datasets/Statlog+(Landsat+Satellite))

Материально-техническое обеспечение занятия.

Компьютерный класс, проектор, программное обеспечение NNT Matlab.

8.2. Методические указания для обучающихся по освоению дисциплины (модулей)

Методические указания предназначены для рационального распределения времени студента по видам самостоятельной работы и разделам дисциплины. Они составляются на основе сведений о трудоемкости дисциплины, ее содержании и видах работы по ее изучению, а также учебно-методического и информационного обеспечения. В раздел включаются: рекомендации по изучению дисциплины (модулей) или отдельных тематических разделов, вопросы и задания для самостоятельной работы, материалы, необходимые, для подготовки к занятиям (разделы книг, статьи и т.д.). Раздел может быть представлен в табличной форме.

Например:

<i>Вид работы</i>	<i>Содержание (перечень вопросов)</i>	<i>Трудоемкость самостоятель ной работы (в часах)</i>	<i>Рекомендации</i>
<i>Раздел № __ (наименование раздела)</i>			
<i>Подготовка к лекции №</i>	<i>Перечень вопросов лекции</i>	<i>N</i>	<i>Список литературы по теме лекции с указанием страниц (разделов), а также других материалов, необходимых для подготовки (конспекты лекций, интернет- ресурсы, программное обеспечение и др.), вопросы и задания для самоконтроля,</i>
<i>Написание реферата</i>	<i>Тематика рефератов по разделу</i>	<i>N</i>	<i>Список литературы для реферирования, интернет-ресурсов. Рекомендации по написанию (объём, структура текста, требования к оформлению).</i>
<i>И т.п.</i>			
<i>Итого</i>		<i>N</i>	
<i>Раздел (Тема) № __ (наименование раздела, темы)</i>			
<i>Итого по дисциплине</i>		<i>N</i>	

8.3. Методические рекомендации по подготовке отчетов по лабораторным работам

Требования при оформлении лабораторных работ:

1. Требования

- Первая страница Титульный лист
- Условия задачи, цели, этапы выполнения
- Программный код
- Графики
- Результаты
- Выводы

Правила оформления лабораторных работ:

- текст печатается на странице формата А4;
- шрифт – Times New Roman;
- размеры полей: левое – 3 см, верхнее – 2 см, правое – 2 см и нижнее – 2 см;
- выравнивание по ширине.
- размер шрифта основного текста – 12;
- интервал межстрочный (полуторный) – 1,5;
- название работы печатается полужирным, размер шрифта – 14;
- заголовки печатаются жирным шрифтом 14-ым размером, перед ними следует оставить пустую строку, выравниваются по центру;
- подзаголовки печатаются жирным шрифтом 12-ым размером выравниваются по центру;
- нумерация страниц – внизу по центру.
- Нумерация рисунков, графиков и т.п. Например: (рис.1 Название рисунка) рисунки нумеруются снизу и по центру, таблица (Таблица 1. Название таблицы) таблицы нумеруются сверху выравнивание к правому краю.
 - Библиографические ссылки при цитировании приводятся в конце статьи и нумеруются согласно порядку цитирования в тексте. Указываются автор (сначала фамилия, потом инициалы), название, место и год издания, страница. Порядковые номера ссылок должны быть написаны внутри квадратных скобок (например: [1], [2]). Источники приводятся с указанием в алфавитном порядке фамилий и инициалов всех авторов, сначала отечественных, затем иностранных, полного названия статьи, названия источника, где напечатана статья, том, номер, страницы (от и до) или полное название книги, место и год издания. Фамилии иностранных авторов, название и выходные данные их работ даются в оригинальной транскрипции. Каждый источник приводится с новой строки.

8.4.8.4. Иные материалы

9. Материально-техническое обеспечение дисциплины

Минимальные требования к материально-техническому обеспечению дисциплины:

- Компьютерный класс
- проектор, экран
- колонки
- программное обеспечение NNT Matlab.

10. Глоссарий

База знаний - семантическая модель, описывающая предметную область и позволяющая отвечать на такие вопросы из этой предметной области, ответы на которые в явном виде не присутствуют в базе. База знаний является основным компонентом интеллектуальных и экспертных систем.

Декларативные знания - знания, которые записаны в памяти интеллектуальной системы так, что они непосредственно доступны для использования после обращения к соответствующему полю памяти. Обычно декларативные знания используются для представления информации о свойствах и фактах предметной области. По форме представления декларативные знания противопоставляются процедурным знаниям.

Знания о предметной области - совокупность сведений о предметной области, хранящихся в базе знаний интеллектуальной системы. Знания о предметной области подразделяются на:

- факты, относящиеся к предметной области;
- закономерности, характерные для предметной области;
- гипотезы о возможных связях между явлениями, процессами и фактами;
- процедуры для решения типовых задач в данной предметной области.

Знания - в информатике - вид информации, отражающей опыт специалиста (эксперта) в определенной предметной области, его понимание множества текущих ситуаций и способы перехода от одного описания объекта к другому. По Д.А.Поспелову для знаний характерны внутренняя интерпретируемость, структурированность, связанность и взаимная активность.

Интеллектуальная система - система или устройство с программным обеспечением, имеющие возможность с помощью встроенного процессора настраивать свои параметры в зависимости от состояния внешней среды.

Извлечение знаний - процесс взаимодействия инженера по знаниям с источником знаний (экспертом), в результате которого становятся явными процесс рассуждений специалистов при принятии решения и структура их представлений о предметной области.

Информация - по законодательству РФ - сведения о лицах, предметах, фактах, событиях, явлениях и процессах независимо от формы их представления.

Информация уменьшает степень неопределенности, неполноту знаний о лицах, предметах, событиях и т.д.

Концептуализация - определение понятий, отношений и механизмов управления, необходимых для описания процессов решения задач в избранной предметной области.

Научное знание - система знаний о законах природы, общества, мышления. Научное знание составляет основу научной картины мира и отражает законы его развития.

Научное знание:

- является результатом постижения действительности и когнитивной основой человеческой деятельности;
- социально обусловлено;
- обладает различной степенью достоверности.

Нейронные модели памяти и обучения - модели нейронных сетей, направленные на изучение их способности к формированию следов памяти и извлечению записанной информации.

Нейрокибернетика - научное направление, изучающее основные закономерности организации и функционирования нейронов и нейронных образований. Основным методом нейрокибернетики является математическое моделирование, при этом данные физиологического эксперимента используются в качестве исходного материала для создания моделей.

Нейробионика - научное направление, изучающее возможность использования принципов строения и функционирования мозга с целью создания более совершенных технических устройств и технологических процессов.

Нейронная сеть - вычислительная или логическая схема, построенная из однородных процессорных элементов, являющихся упрощенными функциональными моделями нейронов.

Формальный нейрон - в нейронных сетях - процессорный элемент, преобразователь данных, получающий входные данные и преобразующий их в соответствии с заданной функцией и параметрами.

Сумматор - в нейронных сетях - блок, суммирующий сигналы, поступающие от нейронов через синапсы. В общем случае сумматор может преобразовывать сигналы и передавать их нейронам или сумматорам тоже через синапсы.

Синапс - в нейронных сетях - связь между формальными нейронами.

Выходной сигнал от нейрона поступает в синапс, который передает его другому нейрону. Сложные синапсы могут иметь память.

Представление знаний - структурирование знаний с целью формализации процессов решения задач в определенной проблемной области.

Приобретение знаний - способ автоматизированного построения базы знаний посредством диалога эксперта и специальной программы, изначально ориентированной на конкретный способ представления знаний.

Процедурные знания - знания, хранящиеся в памяти интеллектуальной системы в виде описаний процедур, с помощью которых их можно получить. Обычно процедурные знания используются для представления информации о способах решения задач в проблемной области, а также различные инструкции, методики и т.п. По форме представления процедурные знания противопоставляются декларативным знаниям.

Сведения - часть знаний, критерий истинности которых не одинаков у различных участников познавательного процесса.

Факт - знание в форме утверждения, достоверность которого строго установлена.

Чанк - в искусственном интеллекте - фрагмент знаний, хранимый и используемый как единое целое.

Эвристика - эмпирическое правило, упрощающее или ограничивающее поиск решений в (сложной) предметной области.

Эвристические знания - знания, накапливаемые интеллектуальной системой в процессе ее функционирования, а также знания, заложенные в ней априорно, но не имеющие статуса абсолютной истинности в данной проблемной области. Обычно эвристические знания связаны с отражением в базе знаний неформального опыта решения задач.

Эксперт - специалист, который за годы обучения и практической деятельности научился эффективно решать задачи, относящиеся к конкретной предметной области.

Экспертные знания - знания, которыми располагает специалист в некоторой предметной области

11. Приложения

12.

Приложение 3
 СОГЛАСОВАНО
 Протокол заседания кафедры
 № _____ от _____

УТВЕРЖДЕНО
 Ректор УНПК «МУК»

 (название)

 (подпись, ф.и.о.)

ЛИСТ ИЗМЕНЕНИЙ

в учебно-методический комплекс (модуля) дисциплины

 (название дисциплины)

по направлению подготовки (специальности) _____

на 20__/20__ учебный год

1. В _____ вносятся следующие изменения:
 (элемент УМК)

1.1.;

1.2.;

...

1.9.

2. В _____ вносятся следующие изменения:
 (элемент УМК)

2.1.;

2.2.;

...

2.9.

3. В _____ вносятся следующие изменения:
 (элемент УМК)

3.1.;

3.2.;

...

3.9.

Составитель
 дата

подпись

расшифровка подписи