

Кафедра

прикладной информатики и высшей математики

Б1. В.12

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ (СРЕДСТВА)

Учебная дисциплина	<i>Основы нейронных сетей</i>
По направлению подготовки	<i>09.03.03</i>
	<i>«Прикладная информатика»</i>
Профиль (программа бакалавриата)	<i>«Прикладная информатика в цифровой экономике»</i>
Форма обучения	<i>Очная</i>

Оценочные материалы (средства) дисциплины рассмотрены (актуализированы) и утверждены на заседании кафедры прикладной информатики и высшей математики

Протокол заседания № 10 от «25» мая 2026 г.

Заведующий кафедрой Стрекалова Наталья Борисовна

Оглавление

1. ПЕРЕЧЕНЬ КОМПЕТЕНЦИЙ И ЭТАПЫ ИХ ФОРМИРОВАНИЯ В ПРОЦЕССЕ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ	3
2. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ (СРЕДСТВА) ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ И ОЦЕНКИ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ, ОБЕСПЕЧИВАЮЩИХ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ В ПРОЦЕССЕ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ	4
3. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ (СРЕДСТВА) ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ДИСЦИПЛИНЕ	13
4. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНОВАНИЯ ЗНАНИЙ, УМЕНИЙ И НАВЫКОВ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ	16
5. ДИАГНОСТИЧЕСКИЕ ЗАДАНИЯ ДЛЯ ОЦЕНКИ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ, ХАРАКТЕРИЗУЮЩИХ УРОВЕНЬ СФОРМИРОВАННОСТИ КОМПЕТЕНЦИЙ	21
6. ЛИСТ СОГЛАСОВАНИЯ	22

1. ПЕРЕЧЕНЬ КОМПЕТЕНЦИЙ И ЭТАПЫ ИХ ФОРМИРОВАНИЯ В ПРОЦЕССЕ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Оценочные материалы (средства) сформированы в соответствии с ФГОС ВО - бакалавриат по направлению подготовки 09.03.03 «Прикладная информатика», утвержденный приказом Министерства образования и науки Российской Федерации от 19.09.2017 № 922 (с изменениями и дополнениями от 26.11.2020, 08.02.2021). В соответствии с матрицей компетенций основной профессиональной образовательной программы «Прикладная информатика в цифровой экономике» (уровень бакалавриата) в процессе обучения по дисциплине «Основы нейронных сетей» происходит формирование закрепленных за дисциплиной компетенций обучающихся. Оценка сформированности компетенций на каждом этапе обучения происходит через оценку планируемых результатов обучения по дисциплине (знаний, умений, навыков).

Результаты освоения образовательной программы (компетенции)	Индикаторы достижения компетенций	Планируемые результаты обучения по дисциплине	Этапы формирования компетенции (семестры и темы)	Оценочное средство
УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1.2. Анализирует поставленную задачу и предлагает варианты ее решения с применением принципов системного подхода	Знать: - возможность и нейронных сетей (как программ особого вида) для решения профессиональных задач; - способы построения нейросетей (бесформульный, самообучение, воплощение и т.п.) и способы их применения для решения практических задач; Уметь: - проектировать нейронную сеть на логическом и алгоритмическом уровне; Владеть: - навыками анализа и сложности построенной нейронной сети	Тема 1. Введение в теорию нейронных сетей Тема 2. Обучение нейронных сетей Тема 3. Система принятия решений на основе математической логики событий Лабораторная работа 1. Алгоритм 1. Персептрон «4-3-1» Лабораторная работа 2. Алгоритм 2. Повторное обучение персептрона «4-3-1» Лабораторная работа 3. Модификация модели «Вавилон». Алгоритм 3. Персептрон с архитектурой «2-3-1» Лабораторная работа 4. Тестирование нейронной сети. Алгоритм 4. Тестирование персептрона «2-3-1» Лабораторная работа 5. Алгоритм 5. Нейросетевой предсказатель «2-3-1»	- устный опрос по темам 1-3 - лабораторные работы 1-5 - тестирование - ответ на теоретический вопрос зачета
ПК-2 Способен проектировать информационные системы	ПК-2.2. Разрабатывает модель информационной системы, в том числе с опорой на современные облачные решения	Знать: - понятие нейрона, базовых моделей нейронов и нейронных сетей, виды биологических нейронных сетей и структуру мозга; Уметь: - использовать модели и	Тема 1. Введение в теорию нейронных сетей Лабораторная работа 1. Алгоритм 1. Персептрон «4-3-1» Лабораторная работа 2. Алгоритм 2. Повторное обучение персептрона «4-3-1» Лабораторная работа 3. Модификация модели «Вавилон». Алгоритм 3.	- устный опрос по теме 1 - тестирование - лабораторные работы 1-5 - ответ на теоретический вопрос зачета

		алгоритмы работы нейронных сетей для решения прикладных задач; Владеть: - навыками решения задач по построению нейронной сети.	Персептрон с архитектурой «2-3-1» Лабораторная работа 4. Тестирование нейронной сети. Алгоритм 4. Тестирование персептрона «2-3-1» Лабораторная работа 5. Алгоритм 5. Нейросетевой предсказатель «2-3-1»	
	ПК-2.4. Проводит анализ современных программно-технических средств, специализированных цифровых платформ и облачных решений, сквозных цифровых технологий и отраслевых решений на их основе с целью выбора средств разработки информационных систем	Знать: - современные направления развития нейросетевых технологий и их возможности для разных сфер деятельности; - перечень программных средств для моделирования и построения нейронных сетей, их возможности и особенности работы; Уметь: - выбирать программные средства и схемотехнические решения для реализации нейронной сети; Владеть: - навыками работы с программными средствами для построения нейронных сетей.	Тема 4. Нейросетевые самообучающиеся и адаптивные системы управления Тема 5. Нейросетевые технологии в экономике и бизнесе, технике Лабораторная работа 1. Алгоритм 1. Персептрон «4-3-1» Лабораторная работа 2. Алгоритм 2. Повторное обучение персептрона «4-3-1» Лабораторная работа 3. Модификация модели «Вавилон». Алгоритм 3. Персептрон с архитектурой «2-3-1» Лабораторная работа 4. Тестирование нейронной сети. Алгоритм 4. Тестирование персептрона «2-3-1» Лабораторная работа 5. Алгоритм 5. Нейросетевой предсказатель «2-3-1»	- устный опрос по теме 3 - тестирование - ответ на теоретический вопрос зачета - лабораторные работы 1-5

2. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ (СРЕДСТВА) ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ И ОЦЕНКИ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ, ОБЕСПЕЧИВАЮЩИХ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ В ПРОЦЕССЕ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Устные опросы

*Вопросы для проведения устного опроса
по теме «Введение в теорию нейронных сетей»*

1. Биологический и искусственный нейрон. Основные функции активации нейронов.
2. Преимущества нейронных сетей. Сопоставление традиционных ЭВМ и нейрокомпьютеров.
3. Классификации нейронных сетей, области применения и решаемые задачи. Основные направления развития нейрокомпьютинга.

4. Персептрон Розенблата.
5. Алгоритм обучения персептрона и правило Хебба.
6. Теорема о сходимости алгоритма обучения персептрона для линейно-разделимых множеств.
7. Проблема исключающего «или».
8. Многослойный персептрон. Представление булевых функций.
9. Преодоление ограничения линейной разделимости и решение проблемы исключающего «или».

*Вопросы для проведения устного опроса
по теме «Обучение нейронных сетей»*

1. Модели искусственного нейрона.
2. Функции активации. Нейрон с векторным входом.
3. Глобальные и локальные данные программы, их область видимости и время жизни.
4. Решение задач классификации на основе подсчёта вероятности принадлежности векторов к рассматриваемым классам.
5. Методы и алгоритмы обучения искусственных нейронных сетей.
6. Алгоритмы обучения, основанные на использовании метода сопряженных градиентов.
7. Самоорганизующихся слои Кохонена.
8. Архитектуры самоорганизующихся нейронных слоев Кохонена и специальные функции для их создания, инициализации, взвешивания, накопления, активации, настройки весов и смещений, адаптации и обучения.
9. Применение самоорганизующихся слоев для исследования топологической структуры данных, их объединением в кластеры (группы) и распределением по классам.
10. Самоорганизующихся карты Кохонена.

*Вопросы для проведения устного опроса
по теме «Система принятия решений на основе математической логики событий»*

1. Предмет теории принятия решений;
2. Системный подход к принятию решений
3. Функции участников в процессе выработки решений
4. Объект и предмет исследования теории принятия решений (ТПР)
5. Эволюция теории принятия решений
6. ЭВМ в принятии решений
7. Формирование информационных ресурсов и использование информационных технологий в процессе разрешения проблемных ситуаций

*Вопросы для проведения устного опроса
по теме «Нейросетевые самообучающиеся и адаптивные системы управления»*

1. На чем основана идея «бесформульных» вычислений и как она реализуется с помощью логической нейронной сети?
2. Как осуществляется ситуационное управление?
3. Как идеи ситуационного управления развиваются для построения самообучающихся систем управления?
4. Как реализуется самообучающаяся система управления с помощью логической нейронной сети?

5. В чем заключается способ учета текущих и прогнозируемых характеристик потока запросов при выборе оптимального плана их параллельного (многоканального) обслуживания?
6. Как производится настройка (адаптация) диспетчера оптимального распараллеливания по текущим и прогнозируемым характеристикам потока запросов?

*Вопросы для проведения устного опроса
по теме «Нейросетевые технологии в экономике и бизнесе, технике»*

1. Нейросетевые системы в экономике привести пример.
2. Нейросетевые системы в медицине привести пример
3. Нейросетевые системы в отраслях связи привести пример
4. Нейросетевые системы автоматизации производства привести пример
5. Нейросетевые системы в политических технологиях привести пример
6. Нейросетевые технологии в охранных системах привести пример
7. Нейросетевые системы в геологоразведке и сейсмический мониторинг.
8. Нейросетевые системы в Авионике привести пример.

Критерии оценивая устного опроса:

- оценка «отлично» выставляется студенту, если он ответил развернуто на один из заданных вопросов, активно дополнял ответы других студентов или задавал им дополнительные вопросы;
- оценка «хорошо» выставляется студенту, если он ответил развернуто на один из заданных вопросов или ответил кратко на ряд вопросов;
- оценка «удовлетворительно» выставляется студенту, если он кратко ответил на один из задаваемых вопросов или просто дополнял ответы других студентов, задавал им дополнительные вопросы;
- оценка «неудовлетворительно» выставляется студенту, если он не смог ответить правильно на заданный вопрос, либо не отвечал на вопросы и не участвовал в их обсуждении.

Тестирование:

1. Что представляет собой модель «живого» нейрона? Как строится абстрактная (математическая) модель нейрона – средства реализации функции активации?
2. Каковы механизмы направленного распространения сигналов в нейронной сети с помощью синаптических весов, позволяющие запоминать причинно-следственные связи?
3. Какую информацию обрабатывают нейроны, и почему задание этой информации на рецепторах следует интерпретировать как достоверность высказываний о принадлежности данных?
4. В каких режимах работает нейронная сеть?
5. Как производится обучение нейронной сети?
6. В чем принципиальное отличие логической нейронной сети от персептрона, реализующего «классический» подход?
7. Какая информация отображается на рецепторах, и как с помощью их возбуждения задавать нечеткие данные или данные, не совпадающие с теми, что использованы при обучении?
8. В чем преимущества работы с признаками изучаемых объектов?
9. Для чего и по какому принципу производится локализация возбуждения нейронов выходного слоя в длинных логических цепочках вывода?

10. В чем заключаются преимущества конструктивного решения проблемы формирования длинных логических цепочек в виде коры (головного мозга)?
11. Как формируется обученная нейронная сеть типа персептрона для распознавания символов с помощью эталонных «ловушек»?
12. Как формируется обученная логическая нейронная сеть с помощью системы правил вывода по всем возможным ситуациям?
13. Как задается информация и как формируются решения по нечетким данным?
14. Как строится логическая нейронная сеть по логическому описанию системы принятия решений?
15. «Схемотехнический» подход к построению нейросети «под задачу».
16. Каковы принципы построения нечеткой логики?
17. На чем основана принципиальная возможность сведения логических нейронных сетей к однослойным?
18. Что представляет собой исчерпывающее множество событий?
19. Что представляет собой факторное пространство событий и как его структуризация с помощью дерева логических возможностей помогает корректно сформировать логическое описание системы принятия решений?
20. Как формируется полное и непротиворечивое логическое описание системы принятия решений?
21. Как логическое описание преобразуется в форму, предполагающую однослойную нейронную сеть?
22. Как осуществляется переход к нечетким данным – к достоверностям высказываний о наступлении событий?
23. Как на основе «схемотехнического» подхода производится обоснование однослойных логических нейронных сетей, реализующих системы принятия решений?
24. Когда возникает проблема обучения логической нейронной сети и в чем ее идея?
25. Как (на иллюстративном уровне) производится обучение нейронной сети заданной структуры распознаванию символов с помощью эталонных «ловушек»?
26. Как описание структуры логической нейронной сети с помощью матрицы следования способствует формальному решению задачи обучения?
27. Вводятся ли при обучении новые связи между нейронами, не отображенные в исходной структуре?
28. Как на основе анализа транзитивных связей корректируется обучаемая структура нейронной сети?
29. В чем заключается (на иллюстративном уровне) идея трассировки логической нейронной сети?
30. Как на основе оценок признаков объектов временного ряда производится логическое описание системы распознавания?
31. Как строится логическая нейронная сеть для обработки нечетких данных?
32. Как строится матрица следования для однослойной логической нейронной сети?
33. Почему попытка «механического» объединения ситуаций в рамках однослойной логической нейронной сети способна привести к некорректности ее решений?
34. Как производится развитие однослойной логической нейронной сети?
35. Как выбирается функция активации нейрона, если все или некоторые веса связей меньше единицы?
36. Для чего и как вводятся обратные связи?
37. Философия ВЕАМ технологий, в чем суть данного направления робототехники?
38. Что такое стимул реакция, основные принципы формирования?
39. Какие типы нейронных сетей существуют?
40. Методы реализации нейронных цепей на основе схемотехнических решений?

41. На чем основана идея «бесформульных» вычислений и как она реализуется с помощью логической нейронной сети?
42. Как осуществляется ситуационное управление?
43. Как идеи ситуационного управления развиваются для построения самообучающихся систем управления?

Критерии оценивая тестирования:

- оценка «отлично» выставляется обучающемуся, если он правильно ответил на 85% вопросов теста;
- оценка «хорошо» выставляется обучающемуся, если он правильно ответил на 75% вопросов теста;
- оценка «удовлетворительно» выставляется обучающемуся, если он правильно ответил на 50% вопросов теста;
- оценка «неудовлетворительно» выставляется обучающемуся, если он ответил менее, чем на 50% вопросов теста.

Практические работы

Практическая работа № 1

Ознакомление с визуальной средой программирования.
персептрон. Обучение персептрона. Решение проблемы классификации векторов

1.1 Цель работы

Ознакомление с визуальной средой программирования на языке Python (на примере Spyder). Освоение методики создания персептрона и процедуры его настройки при помощи библиотеки Neurolab на языке программирования Python. Приобретение навыков решения задачи классификации векторов.

1.2 Методические указания по самостоятельной работе студентов

План работы:

1. Установка среды и библиотек (SciPy, Neurolab). Проверка работоспособности модулей numpy, matplotlib, neurolab.
2. Изучение визуальной среды по предлагаемым и встроенным примерам.
3. Повторение теоретического материала об однослойном персептроне, выполнение численных экспериментов.
4. Создание однослойного персептрона, выполнение предложенных и встроенных экспериментов.
5. Проведение контрольных экспериментов по вариантам.

Каждый пункт должен быть отражён в отчёте.

2.4 Порядок выполнения работы

1. Провести моделирование персептрона.
2. Провести настройку параметров персептрона.
3. Провести классификацию векторов (по варианту в таблице) при помощи персептрона.

Таблица 2.1 – Варианты заданий

Вариант	Координаты точек	Номера точек первого класса	Номера точек второго класса
нт			

1	(-2;1), (-1;0), (0;0), (0.5;-1), (0.5;1)	1,2	3,4,5
2	(-1;-1), (0.5;0), (0;0), (1;1)	1	2,3,4
3	(-0.5;-1), (-1;0), (-1.5;1.5), (0;0), (0.5;-1)	1,2,3	4,5
4	(1;-1), (-1;-1), (0.5;0), (-1.5;1)	1,3	2,4
5	(0;-2), (0;-1), (0.5;-0.5), (-1;-1), (-0.5;1)	1,2,3	4,5
6	(-0.5;-0.5), (-1;0), (0;0.5), (0.5;1)	1,2	3,4
7	(0;0.5), (-1;0), (0.5;-1), (-1.5;1.5), (0.5;0)	1,3,5	2,4
8	(0;0), (0.5;0.5), (-0.5;-0.5), (-1;-1)	1,2	3,4
9	(-1;-0.5), (-0.5;0), (0;0), (0.5;0.5), (1;1)	1,2,3	4,5
10	(0;-1), (-0.5;0), (0;1), (1;1)	1,2,3	4
11	(1;0.5), (0.5;1), (-0.5;0), (-0.5;1), (-1;0.5)	1,2	3,4,5
12	(0;0), (0.5;1), (-0.5;-1), (-1;-0.5)	1,2	3,4
13	(0.5;0), (0;1), (-0.5;-0.5), (-1;0), (-0.5;1)	1,2,3	4,5
14	(0;0.5), (0;-0.5), (-1;0.5), (-1;-0.5)	2,3	1,4
15	(0.5;0.5), (0.5;-1), (-0.5;0.5), (0;0), (0;1)	2,4	1,3,5

4. Внести результаты в отчёт.

5. При помощи приведённого ниже фрагмента сгенерировать классы векторов:

```
import neurolab as nl
n1 = 3          # количество векторов первого класса
n2 = 3          # количество векторов второго класса
x0 = 10
alpha = 15
y0 = 50
beta = 20
X = numpy.random.normal(x0,alpha,(n1,2))
Y = numpy.random.normal(y0,beta,(n2,2))
P = numpy.concatenate((X,Y)) # формирование входных векторов
# формирование выходных векторов
T = numpy.zeros((1,
len(P))) T[:,n1] =
numpy.ones(n1)
T =
T.transpose()
min =
numpy.min(P)
max =
numpy.max(P)
net = neurolab.net.newp([[min,max],[min,max]],1)
```

6. Сделать классификацию векторов.

7. С помощью следующего кода сделать тестовый пример для классификации:

```
x1 = 45
y1 = 50
x1 = numpy.random.normal(x1,beta,1)
y1 = numpy.random.normal(y1,beta,1)
p = numpy.concatenate(x1,y1)
```

Сделать классификацию

8. Результат внести в отчёт

1.5 Содержание отчёта

Содержание отчёта должно соответствовать плану работы.

Обязательные пункты:

- 1) Название работы
- 2) Цель работы
- 3) Краткое описание исследованной нейронной сети и её архитектуры
- 4) Краткое описание проведённых исследований в виде графиков и

числовых данных

- 5) Листинг программы моделирования нейронной сети
- 6) Полученные результаты в виде графиков и числовых данных
- 7) Расширенные выводы из лабораторной работы

Дополнительные задания:

- 1) Провести персептроном классификацию на 4 класса.
- 2) Использовать персептрон для обработки реальных данных (своя задача или подготовленные выборки, например из репозитория UCI

(<http://archive.ics.uci.edu/ml/index.html>), например
(<http://archive.ics.uci.edu/ml/datasets/Iris>)

3) Использовать персептрон для распознавания образов (база MNIST:
<http://code.google.com/p/vlabdownload/downloads/detail?name=MNIST.rar&can=2&q=#makechanges>— 60 тысяч подготовленных изображений рукописных цифр от 0 до 9).

1.6 Контрольные вопросы и задания

1. Что такое персептрон? Какова его архитектура?
2. Какие алгоритмы обучения персептрона вы знаете?
3. Охарактеризуйте правила настройки параметров персептрона.
4. Охарактеризуйте процедуру адаптации параметров персептрона.
5. Каковы достоинства и недостатки персептрона

Критерии оценивая:

- оценка «отлично» выставляется студенту, если задание выполнено в полном объеме, студент хорошо ориентируется в технологии разработки нейронных сетей, программа логично и корректно демонстрирует работу всех функций;
- оценка «хорошо» выставляется студенту, если он выполнил задание почти в полном объеме (может отсутствовать пункт), допускает небольшие «проектно-технологические» ошибки в разработке нейронных сетей;
- оценка «удовлетворительно» выставляется студенту, если он выполнил написание программы по нейронным сетям, но не составил отчет;
- оценка «неудовлетворительно» выставляется студенту, если он не выполнил написание программы по нейронным сетям, но не составил отчет.

Практическая работа № 2

Обучение персептрона

1.1 Цель работы

Ознакомление с визуальной средой программирования на языке Python (на примере Spyder). Освоение методики создания персептрона и процедуры его настройки при помощи библиотеки Neurolab на языке программирования Python. Приобретение навыков решения задачи классификации векторов.

1.2 Методические указания по самостоятельной работе студентов

План работы:

1. Установка среды и библиотек (SciPy, Neurolab). Проверка работоспособности модулей numpy, matplotlib, neurolab.
2. Изучение визуальной среды по предлагаемым и встроенным примерам.
3. Повторение теоретического материала об однослойном персептроне, выполнение численных экспериментов.
4. Создание однослойного персептрона, выполнение предложенных и встроенных экспериментов.
5. Проведение контрольных экспериментов по вариантам.

Каждый пункт должен быть отражён в отчёте.

Порядок выполнения работы

1. Провести моделирование персептрона.
2. Провести настройку параметров персептрона.
3. Провести обучение персептрона при различных параметрах, занести результаты в таблицу.
4. Сделать выводы

Содержание отчёта

Содержание отчёта должно соответствовать плану работы.

Обязательные пункты:

- 1) Название работы
- 2) Цель работы
- 3) Краткое описание исследованной нейронной сети и её архитектуры
- 4) Краткое описание проведённых исследований в виде графиков и числовых данных
- 5) Листинг программы моделирования нейронной сети
- 6) Полученные результаты в виде графиков и числовых данных
- 7) Расширенные выводы из лабораторной работы

Критерии оценивая:

- оценка «отлично» выставляется студенту, если задание выполнено в полном объеме, студент хорошо ориентируется в технологии разработки нейронных сетей, программа логично и корректно демонстрирует работу всех функций;
- оценка «хорошо» выставляется студенту, если он выполнил задание почти в полном объеме (может отсутствовать пункт), допускает небольшие «проектно-технологические» ошибки в разработке нейронных сетей;
- оценка «удовлетворительно» выставляется студенту, если он выполнил написание программы по нейронным сетям, но не составил отчет;
- оценка «неудовлетворительно» выставляется студенту, если он не выполнил написание программы по нейронным сетям, но не составил отчет.

Практическая работа № 3-4

Эксперименты с многослойным персептроном в Keras

1.1 Цель работы

Ознакомление с визуальной средой программирования на языке Python (на примере Keras). Освоение методики создания персептрона и процедуры его настройки при помощи библиотеки Neurolab на языке программирования Python. Приобретение навыков решения задачи классификации векторов.

1.2 Методические указания по самостоятельной работе студентов

План работы:

- 1 Описание Keras
- 2 Создание однослойного перцептрона
- 3 2.1 Подключение numpy и Keras
- 4 Загрузка обучающих и тестовых примеров из MNIST
- 5 Создание сети - однослойного перцептрона
- 6 Компиляция модели
- 7 Обучение сети
- 8 Запись и считывание из файла
- 9 Тестирование модели
- 10 Эксперименты с однослойной сетью
- 11 ОУвеличение числа прогонов
- 12 Выключение нормализации входных данных
- 13 Двуслойный перцептрон
- 14 Трехслойный перцептрон
- 15 Выводы.

Каждый пункт должен быть отражён в отчёте.

Порядок выполнения работы

1. Провести моделирование персептронов.
2. Провести настройку параметров персептронов.
3. Провести обучение персептронов при различных параметрах, занести результаты в таблицу.
4. Сделать выводы

Содержание отчёта

Содержание отчёта должно соответствовать плану работы.

Обязательные пункты:

- 1) Название работы
- 2) Цель работы
- 3) Краткое описание исследованной нейронной сети и её архитектуры
- 4) Краткое описание проведённых исследований в виде графиков и числовых данных
- 5) Листинг программы моделирования нейронной сети
- 6) Полученные результаты в виде графиков и числовых данных
- 7) Расширенные выводы из лабораторной работы

Критерии оценивая:

- оценка «отлично» выставляется студенту, если задание выполнено в полном объеме, студент хорошо ориентируется в технологии разработки нейронных сетей, программа логично и корректно демонстрирует работу всех функций;
- оценка «хорошо» выставляется студенту, если он выполнил задание почти в полном объеме (может отсутствовать пункт), допускает небольшие «проектно-технологические» ошибки в разработке нейронных сетей;
- оценка «удовлетворительно» выставляется студенту, если он выполнил написание программы по нейронным сетям, но не составил отчет;

- оценка *«неудовлетворительно»* выставляется студенту, если он не выполнил написание программы по нейронным сетям, но не составил отчет.

Практическая работа № 5

Нейронная сеть на Python для диагностики диабета

1.1 Цель работы

Ознакомление с визуальной средой программирования на языке Python (на примере Keras). Освоение методики создания персептрона и процедуры его настройки при помощи библиотеки Neurolab на языке программирования Python. Приобретение навыков решения задачи классификации векторов.

1.2 Методические указания по самостоятельной работе студентов

План работы:

1. Загрузка необходимых модулей и данных
2. Нейронная сеть: создание модели
3. Нейронная сеть: обучение и оценка результата
4. Протестировать.
5. Вывести результат
6. Сделать выводы

Содержание отчёта

Содержание отчёта должно соответствовать плану работы.

Обязательные пункты:

- 1) Название работы
- 2) Цель работы
- 3) Краткое описание исследованной нейронной сети и её архитектуры
- 4) Краткое описание проведённых исследований в виде графиков и числовых данных
- 5) Листинг программы моделирования нейронной сети
- 6) Полученные результаты в виде графиков и числовых данных
- 7) Расширенные выводы из лабораторной работы

Критерии оценивая:

- оценка *«отлично»* выставляется студенту, если задание выполнено в полном объеме, студент хорошо ориентируется в технологии разработки нейронных сетей, программа логично и корректно демонстрирует работу всех функций;
- оценка *«хорошо»* выставляется студенту, если он выполнил задание почти в полном объеме (может отсутствовать пункт), допускает небольшие «проектно-технологические» ошибки в разработке нейронных сетей;
- оценка *«удовлетворительно»* выставляется студенту, если он выполнил написание программы по нейронным сетям, но не составил отчет;
- оценка *«неудовлетворительно»* выставляется студенту, если он не выполнил написание программы по нейронным сетям, но не составил отчет.

3. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ (СРЕДСТВА) ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ДИСЦИПЛИНЕ

ОЦЕНКА ЗНАНИЙ ОБУЧАЮЩИХСЯ НА ЗАЧЕТЕ ВОПРОСЫ ДЛЯ ПОДГОТОВКИ К ЗАЧЕТУ

Компетенция УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач.

Индикатор УК-1.2. Анализирует поставленную задачу и предлагает варианты ее решения с применением принципов системного подхода

Обучающийся знает: возможности нейронных сетей (как программ особого вида) для решения профессиональных задач; способы построения нейросетей (бесформульный, самообучение, воплощение и т.п.) и способы их применения для решения практических задач;.

Оценка достижения обучающимся запланированного результата обучения осуществляется по результатам его участия в устных опросах, а также по результатам ответа на зачете:

1. Обучение искусственной нейронной сети.
2. Нейросеть на основе графа произвольной структуры.
3. «Схемотехнический» подход к построению нейросети «под задачу».
4. Как формируется обученная логическая нейронная сеть с помощью системы правил вывода по всем возможным ситуациям?
5. Как строится логическая нейронная сеть по логическому описанию системы принятия решений?
6. Каковы принципы построения нечеткой логики?
7. Проблема обучения «готовых» нейросетей.
8. Дерево логических возможностей.
9. Достоверность высказываний о событиях.
10. Как строится абстрактная (математическая) модель нейрона – средства реализации функции активации?
11. Каковы механизмы направленного распространения сигналов в нейронной сети с помощью синаптических весов, позволяющие запоминать причинно-следственные связи?
12. Какую информацию обрабатывают нейроны, и почему задание этой информации на рецепторах следует интерпретировать как достоверность высказываний о принадлежности данных?
13. В каких режимах работает нейронная сеть?
14. Как производится обучение нейронной сети?
15. В чем принципиальное отличие логической нейронной сети от персептрона, реализующего «классический» подход?
16. Какая информация отображается на рецепторах, и как с помощью их возбуждения задавать нечеткие данные или данные, не совпадающие с теми, что использованы при обучении?
17. На чем основана принципиальная возможность сведения логических нейронных сетей к однослойным?
18. Что представляет собой факторное пространство событий и как его структуризация с помощью дерева логических возможностей помогает корректно сформировать логическое описание системы принятия решений?
19. Как на основе «схемотехнического» подхода производится обоснование однослойных логических нейронных сетей, реализующих системы принятия решений?
20. Когда возникает проблема обучения логической нейронной сети и в чем ее идея?
21. Как описание структуры логической нейронной сети с помощью матрицы следования способствует формальному решению задачи обучения?
22. Вводятся ли при обучении новые связи между нейронами, не отображенные в исходной структуре?

Компетенция ПК-2 Способен проектировать информационные системы

Индикатор ПК-2.2. Разрабатывает модель информационной системы, в том числе с опорой на современные облачные решения

Обучающийся знает: понятие нейрона, базовых моделей нейронов и нейронных сетей, виды биологических нейронных сетей и структуру мозга;

Оценка достижения обучающимся запланированного результата обучения осуществляется по результатам его участия в устных опросах, а также по результатам ответа на зачете.

1. История нейронных сетей. Аналогия с моделью мозга. Формирование информации на рецепторном слое. Пространство признаков.
2. Что представляет собой модель «живого» нейрона?
3. Ключевые термины. Классификация нейронных сетей и их свойств.
4. Классификация информационных интеллектуальных систем. Нейронная сеть.

Компетенция ПК-2 Способен проектировать информационные системы

Индикатор ПК-2.4. Проводит анализ современных программно-технических средств, специализированных цифровых платформ и облачных решений, сквозных цифровых технологий и отраслевых решений на их основе с целью выбора средств разработки информационных систем

Обучающийся знает: современные направления развития нейросетевых технологий и их возможности для разных сфер деятельности; перечень программных средств для моделирования и построения нейронных сетей, их возможности и особенности работы;.

Оценка достижения обучающимся запланированного результата обучения осуществляется по результатам его участия в устном опросе, а также по результатам ответа на зачете.

1. Философия BEAM технологий, в чем суть данного направления робототехники?
2. Что такое стимул реакция, основные принципы формирования?
3. Какие типы нейронных сетей существуют?
4. Методы реализации нейронных цепей на основе схмотехнических решений?
5. Маршрутизация по смежным узлам (технология Wi-Fi) на основе логической нейронной сети с обратными связями.
6. Современные направления развития нейросетевых технологий.
7. Динамический выбор оптимальной стратегии распараллеливания в многопроцессорной вычислительной системе.

ОЦЕНКА УМЕНИЙ И НАВЫКОВ ОБУЧАЮЩИХСЯ НА ЗАЧЕТЕ

Компетенция УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач.

Индикатор УК-1.2. Анализирует поставленную задачу и предлагает варианты ее решения с применением принципов системного подхода

Обучающийся умеет: проектировать нейронную сеть на логическом и алгоритмическом уровне;

Обучающийся владеет: навыками анализа и сложности построенной нейронной сети;

Оценка достижения обучающимся запланированного результата обучения осуществляется по результатам выполнения лабораторных работ

Компетенция ПК-2 Способен проектировать информационные системы

Индикатор ПК-2.2. Разрабатывает модель информационной системы, в том числе с опорой на современные облачные решения

Обучающийся умеет: использовать модели и алгоритмы работы нейронных сетей для решения прикладных задач;

Обучающийся владеет: навыками решения задач по построению нейронной сети.

Оценка достижения обучающимся запланированного результата обучения осуществляется по результатам выполнения лабораторных работ.

Компетенция ПК-2 Способен проектировать информационные системы

Индикатор ПК-2.4. Проводит анализ современных программно-технических средств, специализированных цифровых платформ и облачных решений, сквозных цифровых технологий и отраслевых решений на их основе с целью выбора средств разработки информационных систем

Обучающийся умеет: выбирать программные средства и схмотехнические решения для реализации нейронной сети;

Обучающийся владеет: навыками работы с программными средствами для построения нейронных сетей.

Оценка достижения обучающимся запланированного результата обучения осуществляется по результатам выполнения лабораторных работ, что свидетельствует о накоплении опыта работы с программными средствами построения нейронных сетей и опыта разработки схмотехнических решений..

4. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНОВАНИЯ ЗНАНИЙ, УМЕНИЙ И НАВЫКОВ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ

КРИТЕРИИ ОЦЕНИВАНИЯ СФОРМИРОВАННОСТИ КОМПЕТЕНЦИЙ

Планируемые результаты обучения (показатели достижения заданного уровня освоения компетенций)	Критерии оценивания результатов обучения				
	1	2	3	4	5
Компетенция УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач. Индикатор УК-1.2. Анализирует поставленную задачу и предлагает варианты ее решения с применением принципов системного подхода					
Знать: - возможности нейронных сетей (как программ особого вида) для решения профессиональ ных задач; - способы построения нейросетей (бесформульн ый, самообучение, воплощение и т.п.) и способы	Не знает	Фрагментарн ые знания о возможностях нейронных сетей (как программ особого вида) для решения профессионал ьных задач; способах построения нейросетей (бесформульн ый, самообучение , воплощение	Общие, но не структуриров анные знания о возможностях нейронных сетей (как программ особого вида) для решения профессионал ьных задач; способах построения нейросетей (бесформульн ый,	Сформированны е, но содержащие отдельные пробелы знания о возможностях нейронных сетей (как программ особого вида) для решения профессиональн ых задач; способах построения нейросетей (бесформульный , самообучение,	Сформированны е систематизирова нные прочные знания о возможностях нейронных сетей (как программ особого вида) для решения профессиональн ых задач; способах построения нейросетей (бесформульный , самообучение,

Планируемые результаты обучения (показатели достижения заданного уровня освоения компетенций)	Критерии оценивания результатов обучения				
	1	2	3	4	5
их применения для решения практических задач		и т.п.) и способах их применения для решения практических задач	самообучение, воплощение и т.п.) и способах их применения для решения практических задач	воплощение и т.п.) и способах их применения для решения практических задач	воплощение и т.п.) и способах их применения для решения практических задач
Уметь: проектировать нейронную сеть на логическом и алгоритмическом уровне	Не умеет	Частично освоенные умения проектировать нейронную сеть на логическом и алгоритмическом уровне	В целом освоенные, но не подкрепляемые знаниями и самостоятельностью выполнения умения проектировать нейронную сеть на логическом и алгоритмическом уровне	В целом сформированные и самостоятельно выполняемые, но не всегда интеллектуально обоснованные умения проектировать нейронную сеть на логическом и алгоритмическом уровне	Успешно сформированные, самостоятельно и осознанно выполняемые умения проектировать нейронную сеть на логическом и алгоритмическом уровне
Владеть: навыками анализа и сложности построенной нейронной сети	Не владеет	Фрагментарно сформированные навыки анализа и сложности построенной нейронной сети	В целом сформированные, но выполняемые на элементарном уровне навыки анализа и сложности построенной нейронной сети	Сформированные, но не устойчивые в сложных ситуациях навыки анализа и сложности построенной нейронной сети	Успешно сформированные, устойчивые, технологически грамотно выполняемые навыки анализа и сложности построенной нейронной сети
Компетенция ПК-2 Способен проектировать информационные системы Индикатор ПК-2.2. Разрабатывает модель информационной системы, в том числе с опорой на современные облачные решения					
Знать: - понятие нейрона, базовых моделей нейронов и нейронных сетей, виды биологических нейронных сетей и структуру	Не знает	Фрагментарные знания о нейронах, базовых моделях нейронов и нейронных сетей	Общие, но не структурированные знания о нейронах, базовых моделях нейронов и нейронных сетей	Сформированные, но содержащие отдельные пробелы знания о нейронах, базовых моделях нейронов и нейронных сетей	Сформированные систематизированные прочные знания о нейронах, базовых моделях нейронов и нейронных сетей

Планируемые результаты обучения (показатели достижения заданного уровня освоения компетенций)	Критерии оценивания результатов обучения				
	1	2	3	4	5
мозга					
Уметь: - использовать модели и алгоритмы работы нейронных сетей для решения прикладных задач	Не умеет	Частично освоенные умения использовать модели и алгоритмы работы нейронных сетей для решения прикладных задач	В целом освоенные, но не подкрепляемые знаниями и самостоятельностью выполнения умения использовать модели и алгоритмы работы нейронных сетей для решения прикладных задач	В целом сформированные и самостоятельно выполняемые, но не всегда интеллектуально обоснованные умения использовать модели и алгоритмы работы нейронных сетей для решения прикладных задач	Успешно сформированные, самостоятельно и осознанно выполняемые умения использовать модели и алгоритмы работы нейронных сетей для решения прикладных задач
Владеть: - навыками решения задач по построению нейронной сети	Не владеет	Фрагментарно сформированные навыки решения задач по построению нейронной сети	В целом сформированные, но выполняемые на элементарном уровне навыки решения задач по построению нейронной сети	Сформированные, но не устойчивые в сложных ситуациях навыки решения задач по построению нейронной сети	Успешно сформированные, устойчивые, технологически грамотно выполняемые навыки решения задач по построению нейронной сети
Компетенция ПК-2 Способен проектировать информационные системы Индикатор ПК-2.4. Проводит анализ современных программно-технических средств, специализированных цифровых платформ и облачных решений, сквозных цифровых технологий и отраслевых решений на их основе с целью выбора средств разработки информационных систем					
Знать: - современные направления развития нейросетевых технологий и их возможности для разных сфер деятельности, перечень программных средств для моделирования и	Не знает	Фрагментарные знания современных направлений развития нейросетевых технологий и их возможностей для разных сфер деятельности, перечня программных средств для моделирования	Общие, но не структурированные знания современных направлений развития нейросетевых технологий и их возможностей для разных сфер деятельности, перечня программных средств для	Сформированные, но содержащие отдельные пробелы знания современных направлений развития нейросетевых технологий и их возможностей для разных сфер деятельности, перечня программных средств для	Сформированные, систематизированные прочные знания современных направлений развития нейросетевых технологий и их возможностей для разных сфер деятельности, перечня программных средств для

Планируемые результаты обучения (показатели достижения заданного уровня освоения компетенций)	Критерии оценивания результатов обучения				
	1	2	3	4	5
построения нейронных сетей, их возможности и особенности работы		я и построения нейронных сетей, их возможности и особенности работы	моделирования и построения нейронных сетей, их возможности и особенности работы	моделирования и построения нейронных сетей, их возможности и особенности работы	моделирования и построения нейронных сетей, их возможности и особенности работы
Уметь: - выбирать программные средства и схемотехнические решения для реализации нейронной сети	Не умеет	Частично освоенные умения выбирать программные средства и схемотехнические решения для реализации нейронной сети	В целом освоенные, но не подкрепляемые знаниями и самостоятельно выполняемые умения выбирать программные средства и схемотехнические решения для реализации нейронной сети	В целом сформированные и самостоятельно выполняемые, но не всегда интеллектуально обоснованные умения выбирать программные средства и схемотехнические решения для реализации нейронной сети	Успешно сформированные, самостоятельно и осознанно выполняемые умения выбирать программные средства и схемотехнические решения для реализации нейронной сети
Владеть: - навыками работы с программными средствами для построения нейронных сетей	Не владеет	Фрагментарно сформированные навыки работы с программными средствами для построения нейронных сетей	В целом сформированные, но выполняемые на элементарном уровне навыки работы с программными средствами для построения нейронных сетей	Сформированные, но не устойчивые в сложных ситуациях навыки работы с программными средствами для построения нейронных сетей	Успешно сформированные, устойчивые, технологически грамотно выполняемые навыки работы с программными средствами для построения нейронных сетей

ПРОЦЕДУРА ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ И КРИТЕРИИ ОЦЕНКИ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Для контроля усвоения обучающимися данной дисциплины учебным планом предусмотрен зачет или по результатам текущего контроля. Зачет в третьем семестре проводится в форме устного ответа на теоретический вопрос, сдачи отчета по практическим работам, беседы с преподавателем по материалам отчета. В ходе промежуточной аттестации

перевод результатов работы обучающихся (результатов текущего контроля, рейтинговых баллов) в систему оценки знаний осуществляется с ориентацией на критерии оценивания сформированности компетенций следующим образом:

Отметка **«зачтено»** выставляется обучающемуся, который в итоговом отчете выполнил все практические задания на отметку не ниже «удовлетворительно», ответил на вопросы по некоторым заданиям, ответил на устный вопрос преподавателя из списка вопросов к зачету; показал сформированные систематизированные прочные знания о нейронах, базовых моделях нейронов и нейронных сетей, видах биологических нейронных сетей и о структуре мозга; о классификации интеллектуальных информационных систем; об основных этапах обучения нейронной сети; о типах и принципах построения нейронных сетей.

Отметка **«не зачтено»** выставляется обучающемуся, который не выполнил в полном объеме все практические задания, соответственно не предоставил на контрольное мероприятие итоговый отчет, имеет фрагментарные знания о нейронах, базовых моделях нейронов и нейронных сетей, видах биологических нейронных сетей и о структуре мозга; о классификации интеллектуальных информационных систем; об основных этапах обучения нейронной сети; о типах и принципах построения нейронных сетей.

5. ДИАГНОСТИЧЕСКИЕ ЗАДАНИЯ ДЛЯ ОЦЕНКИ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ, ХАРАКТЕРИЗУЮЩИХ УРОВЕНЬ СФОРМИРОВАННОСТИ КОМПЕТЕНЦИЙ

Компетенция УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач.

Обучающийся знает: возможности нейронных сетей (как программ особого вида) для решения профессиональных задач; способы построения нейросетей (бесформульный, самообучение, воплощение и т.п.) и способы их применения для решения практических задач.

Обучающийся умеет: проектировать нейронную сеть на логическом и алгоритмическом уровне.

Обучающийся владеет: навыками анализа и сложности построенной нейронной сети.

ЗАДАНИЯ ЗАКРЫТОГО ТИПА

Задание 1. Какое из следующих утверждений о нейронных сетях верно?

- a) Нейронные сети моделируют структуру мозга и работают точно так же.
- b) Нейронные сети используют только один слой для обработки данных.
- c) Нейронные сети не могут обучаться на небольших объемах данных. Нейронные сети применяются только в области изображений и распознавания речи.

Ответ: a

Задание 2. Что такое «вес» нейрона в нейронной сети?

- a) Скорость обучения нейрона. Число нейронов в слое.
- b) Выходной сигнал нейрона.
- c) Коэффициент, определяющий важность входных данных для выхода нейрона.

Ответ: c

Задание 3. Как называется функция активации, используемая для преобразования выхода нейрона в диапазон от 0 до 1?

- a) ReLU
- b) Сигмоид
- c) TanH
- d) Softmax

Ответ: b

Задание 4. Какие из перечисленных задач могут быть решены с использованием нейронных сетей?

- a) Распознавание изображений
- b) Машинный перевод текста
- c) Распознавание речи
- d) Все вышеперечисленные задачи.

Ответ: d

Задание 5. Что представляет собой «обучение с учителем» в контексте нейронных сетей?

- a) Модель обучается без наличия учителя.
- b) Модель обучается с использованием только положительных примеров.
- c) Модель обучается с использованием только отрицательных примеров.
- d) Модель обучается на основе пар вход-выход (учительских примеров).

Ответ: d

Задание 6. Какое из нижеперечисленных утверждений описывает структуру нейрона в искусственных нейронных сетях?

- a) Содержит в себе только один слой для входных данных.
- b) Имеет три входа и один выход.
- c) Состоит из весов, активационной функции и входных данных.

d) Содержит веса, входы, сумматор и активационную функцию.

Ответ: d

Задание 7. Какая из перечисленных функций обычно используется для вычисления ошибки в процессе обучения нейронных сетей?

- a) Синусоида.
- b) Тангенс.
- c) Экспоненциальная функция.
- d) Функция потерь.

Ответ: d

Задание 8. Что такое перцептрон в терминах нейронных сетей?

- a) Нейрон с нелинейной активационной функцией.
- b) Сеть, состоящая только из входного слоя.
- c) Нейрон с одним входом и одним выходом.
- d) Простейший вид искусственного нейрона с пороговой функцией активации.

Ответ: d

Задание 9. Какая функция используется для обновления весов в процессе обучения нейронных сетей?

- a) Сигмоида.
- b) Гиперболический тангенс.
- c) Градиентный спуск.
- d) Регуляризация.

Ответ: c

Задание 10. Что представляет собой эпоха в контексте обучения нейронных сетей?

- a) Слой нейронов в сети.
- b) Процесс передачи сигнала по сети.
- c) Одна итерация обучения по всем образцам в наборе данных.
- d) Период, в течение которого весь набор данных использован для обучения.

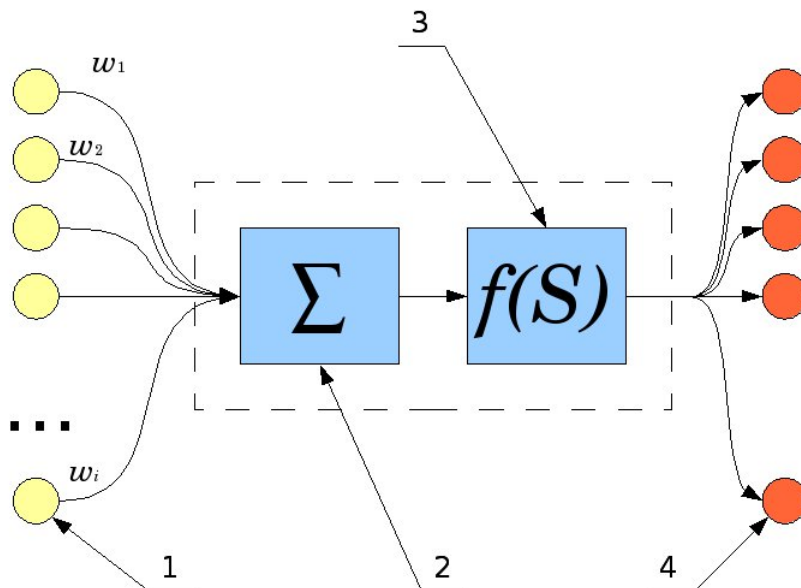
Ответ: d

ЗАДАНИЯ ОТКРЫТОГО ТИПА (15 заданий, из них 5 – с развернутым ответом)

-----Уметь-----

Задание 1. Схематично изобразить искусственный формальный нейрон (Мак-Каллока-Питтса), описать его составные части.

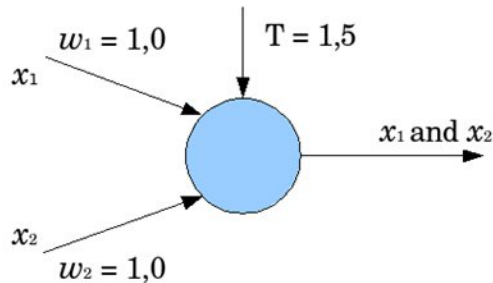
Ответ:



где 1 это входные сигналы $x_1, x_2 \dots x_i$; 2 – сумматор входных сигналов $S = \sum w_i \cdot x_i$; 3- активационная функция $Y=f(S)$; 4 – выходные сигналы нейрона y ; 5 – $w_1, w_2 \dots w_i$ весовые коэффициенты связей, множители входных сигналов.

Задание 2. Постройте схему нейрона с таблицей истинности, настроенного на моделирование логического «И»

Ответ:

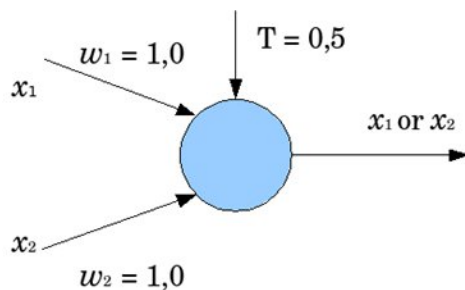


И	T	1,5			
	w_1	1,0			
	w_2	1,0			
	x_1	0	0	1	1
	x_2	0	1	0	1
	f	0	0	0	1

Задание 3. Постройте схему нейрона с настроенного на моделирование логического «ИЛИ»

таблицей истинности,

Ответ:



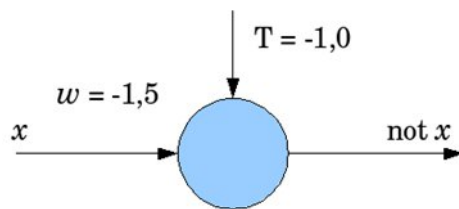
ИЛИ	T	0,5			
	w_1	1,0			
	w_2	1,0			
	x_1	0	0	1	1
	x_2	0	1	0	1
	f	0	1	1	1

Задание 4. Постройте схему нейрона с таблицей настроенного на моделирование логического

истинности, «НЕ»

Ответ:

Н Е	T	-1,0
	w	-1,5



	x	0	1
	f	1	0

Задание 5. Напишите математическую модель пороговой функции активации, постройте график, реализуйте на языке Python. Кратко охарактеризуйте ее.

Ответ: Пороговая функция активации, повторяет действие биологического нейрона:

- возможно только два состояния (активирован или нет);
- нейрон активируется при достижении порогового значения θ .

Математически данную функцию активации можно выразить формулой:

$$f(x) = \begin{cases} 1, & x \geq \theta \\ 0, & x < \theta \end{cases}$$

График функции



Реализация на *Python*:

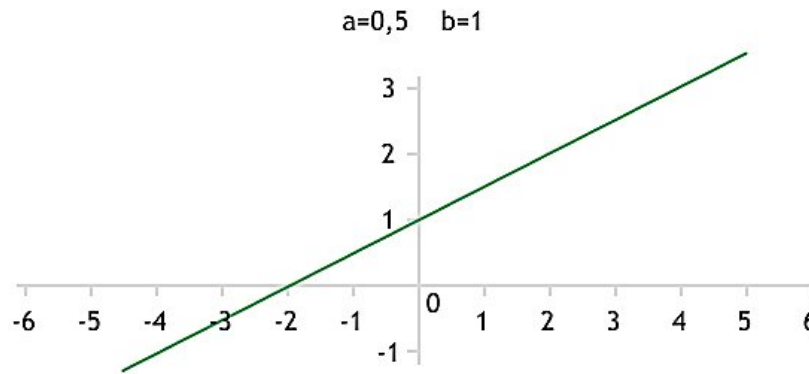
```
theta = 0
def ActStep (x):
    return 1 if x >= theta else 0
```

Задание 6. Напишите математическую модель линейной функции активации, постройте график, реализуйте на языке Python. Кратко охарактеризуйте ее.

Ответ: Применение линейной функции активации дает линейную функцию всего нейрона и нейронной сети. Данное свойство не позволяет использовать линейную функцию активации для решения нелинейных задач. В то же время, создавая нелинейность на скрытых слоях нейронной сети с помощью других функций активации, мы можем использовать линейную функцию активации в нейронах выходного слоя своей модели. Такой прием позволяет решать нелинейные задачи регрессии. Линейная функция активации задается линейной функцией:

$$f(x) = ax + b,$$

График линейной функции



Реализация на *Python*:

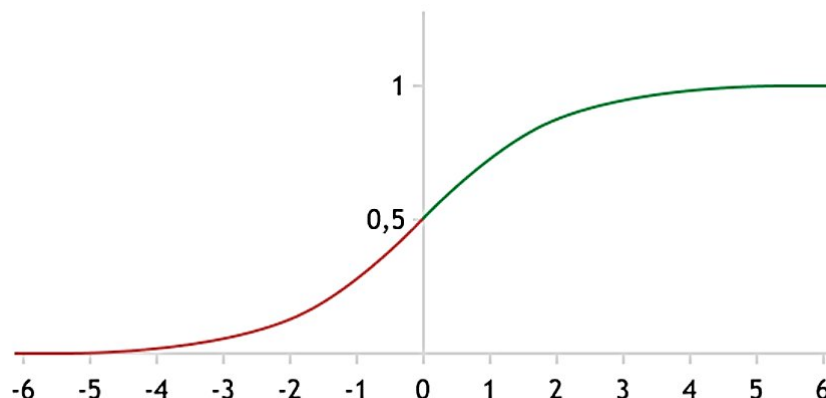
```
a = 1.0
b = 0.0
def ActLinear (x):
    return a * x + b
```

Задание 7. Напишите математическую модель логистической функции активации (Сигмоида), постройте график, реализуйте на языке Python. Кратко охарактеризуйте ее.

Ответ: Логистическая функция позволяет нормализовать выходные значения функции в диапазоне $[0, 1]$. Благодаря этому свойству использование логистической функции вводит понятие вероятности в практику нейронных сетей. Это свойство широко эксплуатируется в нейронах выходного слоя при решении задач классификации, когда количество нейронов выходного слоя равно количеству классов, а отнесение объекта к тому или иному классу определяется по максимальной вероятности (максимальному значению выходного нейрона). Математическая формула функции имеет вид:

$$f(x) = \frac{1}{1 + e^{-x}} \quad \text{или} \quad f(x) = \frac{a}{1 + e^{-x}} - b,$$

График логистической функции:



При реализации на Python перед использованием функции экспоненты необходимо импортировать библиотеку *math*:

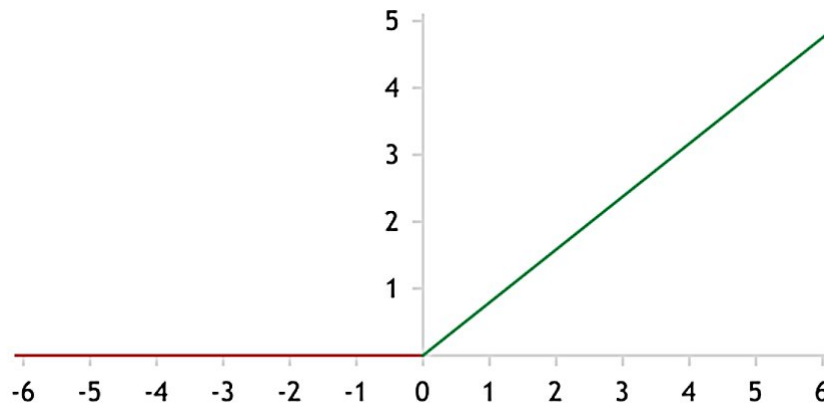
```
import math
a = 1.0
b = 0.0
def ActSigmoid (x):
    return a / (1 + math.exp(-x)) - b
```

Задание 8. Напишите математическую модель функции активации выпрямленного линейного блока (ReLU), постройте график, реализуйте на языке Python. Кратко охарактеризуйте ее.

Ответ: Еще одна широко используемая функция активации нейронов — ReLU (выпрямленный линейный блок). При входящих значениях больше нуля функция возвращает само значение, подобно линейной функции активации. При значениях меньше или равном нулю функция всегда возвращает 0. Математически данная функция выражается формулой:

$$f(x) = \begin{cases} x, & x > 0 \\ 0, & x \leq 0 \end{cases} = \max(0, x)$$

График функции выпрямленного линейного блока (ReLU):



Реализация на Python.

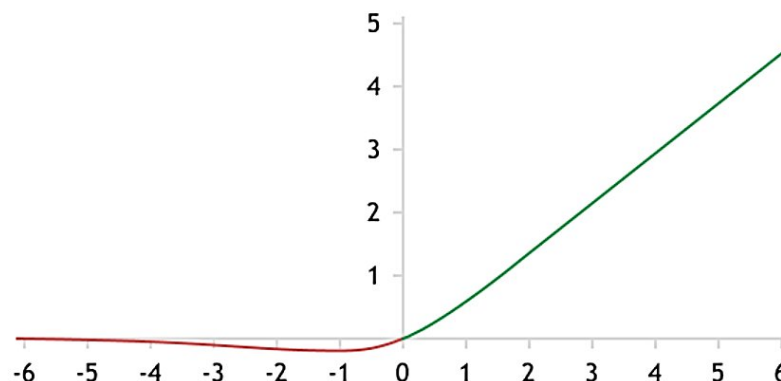
```
a = 0.01
def ActPReLU (x):
    return x if x >= 0 else a * x
```

Задание 9. Напишите математическую модель функции активации Swish, постройте график, реализуйте на языке Python. Кратко охарактеризуйте ее.

Ответ: Замена ReLU на Swish (без переобучения) позволила улучшить показатели нейронных сетей. Математическая формула функции имеет вид:

$$f(x) = x * \text{sigmoid}(\beta x) = \frac{x}{1 + e^{-\beta x}}$$

Параметр β влияет на нелинейность функции и может быть принят константой при проектировании сети или подбираться в процессе обучения. График функции



Реализация на Python.

```
import math
b=1.0
def ActSwish (x):
```

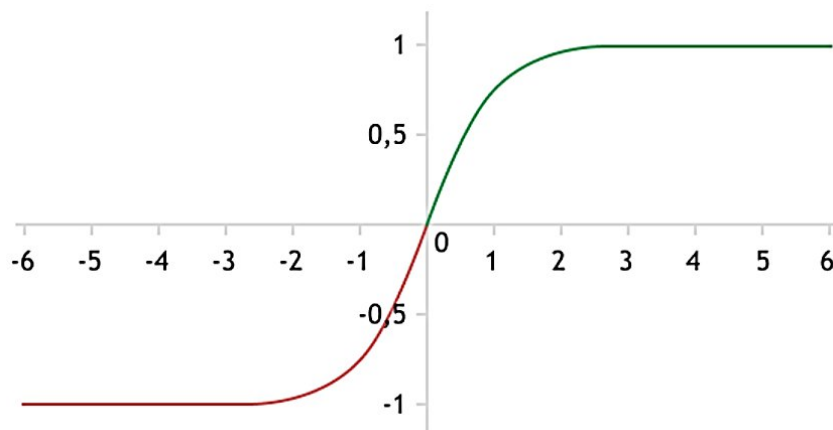
```
return x / (1 + math.exp(-b * x))
```

Задание 10. Напишите математическую модель функции активации гиперболический тангенс ($\tanh$), постройте график, реализуйте на языке Python. Кратко охарактеризуйте ее.

Ответ: Альтернативой логистической функции активации является гиперболический тангенс ($\tanh$). Он так же, как и логистическая функция, имеет S-образный график, а значения функции нормализованы. Но они принадлежат диапазону от -1 до 1 , и изменение состояния нейрона осуществляется в 2 раза быстрее. График функции также асимметричен, но в отличие от логистической функции центр асимметрии находится в центре координат.

$$f(x) = \tanh(x) = \frac{2}{1 + e^{-2x}} - 1$$

График функции гиперболический тангенс ($\tanh$):



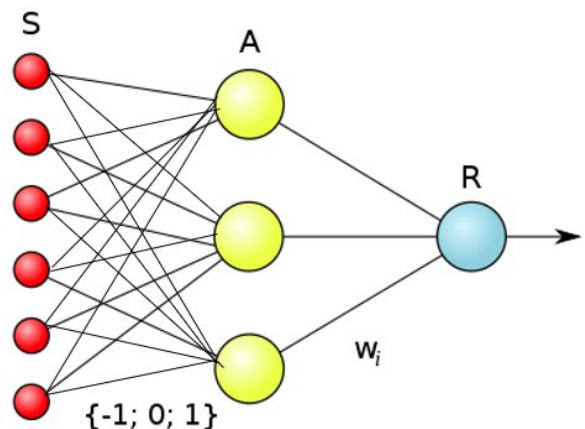
При реализации на Python необходимо импортировать библиотеку *math*:

```
import math
def ActTanh (x):
    return math.tanh(x)
```

-----Владеть-----

Задание 11. Опишите основные элементы элементарного перцептрона их типы. Нарисуйте схему элементарного перцептрона.

Ответ: Элементарный перцептрон состоит из элементов трёх типов: S-элементов, A-элементов (ассоциативные) и одного R-элемента. S-элементы — это слой сенсоров или рецепторов. Каждый рецептор может находиться в одном из двух состояний — покоя или возбуждения, и только в последнем случае он передаёт единичный сигнал в следующий слой, ассоциативным элементам.

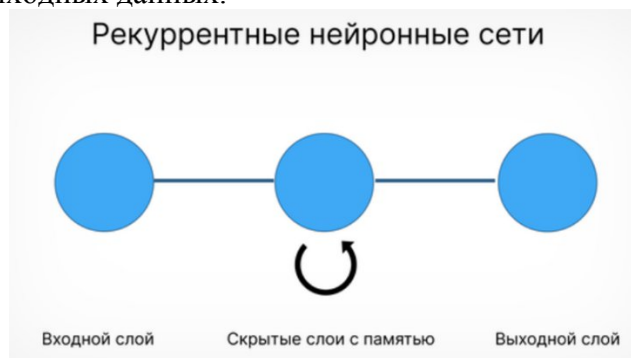


Логическая схема элементарного перцептрона:

Веса S—A связей могут иметь значения -1 , $+1$ или 0 (то есть отсутствие связи). Веса A—R связей W_i могут быть любыми

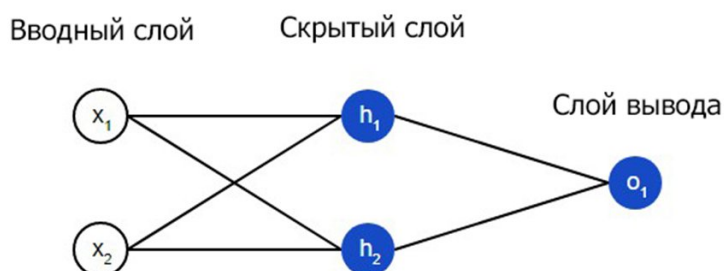
Задание 12. Опишите работу и дайте краткую характеристику рекуррентных нейронных сетей, нарисуйте упрощенную схему.

Ответ: Рекуррентные нейросети применяют для генерации текстов, обработки речи и перевода. Это сети с циклами, главная особенность которых — использование памяти. Модель будет перемещать данные вперед и возвращать их к предыдущим шагам, чтобы как можно лучше выполнить задачу. Слои между входным и выходным слоями повторяются, поскольку данные зацикливаются и сохраняются — так сеть хранит информацию о всех доступных данных. Благодаря этому модели лучше понимают контекст входных данных и улучшают прогнозы выходных данных.



Задание 13. Нарисуйте простейшую нейронную сеть прямого распространения FeedForward с двумя входами, одним скрытым слоем и одним выходом, опишите ее работу.

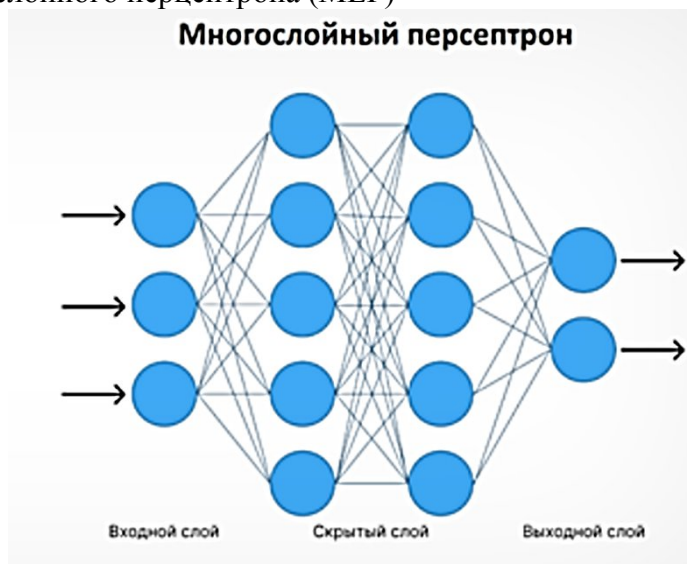
Ответ: Простая нейронная сеть прямого распространения FeedForward с двумя входами, одним скрытым слоем и одним выходом выглядит следующим образом:



Нейронная сеть, по сути, представляет собой группу связанных между собой нейронов. На вводном слое сети два входа — x_1 и x_2 . На скрытом слое два нейрона — h_1 и h_2 . На слое вывода находится один нейрон — o_1 . Входные данные для o_1 являются результатами вывода h_1 и h_2 .

Задание 14. Нарисуйте схему многослойного перцептрона (MLP), опишите его работу.

Ответ: Схема многослойного перцептрона (MLP)



Многослойный перцептрон (MLP) — это тип искусственной нейронной сети, состоящей из нескольких слоев взаимосвязанных искусственных нейронов. Это нейронная сеть с прямой

Обучающийся умеет: использовать модели и алгоритмы работы нейронных сетей для решения прикладных задач; выбирать программные средства и схемотехнические решения для реализации нейронной сети.

Обучающийся владеет: навыками решения задач по построению нейронной сети; навыками работы с программными средствами для построения нейронных сетей.

ЗАДАНИЯ ЗАКРЫТОГО ТИПА

Задание 1. Какие типы слоев чаще всего используются в сверточных нейронных сетях для анализа изображений?

- a) Слой обратного распространения.
- b) Слой декодирования.
- c) Слой агрегации.
- d) Слой свертки.

Ответ: d

Задание 2. Какая из перечисленных архитектур предназначена для передачи данных только в одном направлении, от входных к выходным нейронам?

- a) Рекуррентная нейронная сеть
- b) Сверточная нейронная сеть
- c) Прямая нейронная сеть
- d) Глубокая нейронная сеть

Ответ: c

Задание 3. Что представляет собой процесс обучения нейронной сети на наборе данных?

- a) Расчет точности сети
- b) Определение структуры сети
- c) Коррекция весов на основе ошибки
- d) Измерение времени обработки данных

Ответ: c

Задание 4. Что представляют собой веса нейронов в контексте нейронных сетей?

- a) Специальные коэффициенты для коррекции ошибок
- b) Экспоненциальные коэффициенты для активации
- c) Параметры, определяющие силу связей между нейронами
- d) Число проходов данных через сеть

Ответ: c

Задание 5. Что такое сверточные нейронные сети (CNN)?

- a) Сети, обученные с подкреплением.
- b) Сети без весов.
- c) Сети, специально предназначенные для обработки и анализа изображений.
- d) Сети с большим количеством скрытых слоев.

Ответ: c

Задание 6. Какое из нижеперечисленных утверждений о нейронных сетях является верным?

- a) Нейронные сети используются исключительно для анализа текстовой информации.
- b) Нейронные сети могут быть обучены извлекать сложные закономерности из данных.
- c) Нейронные сети не требуют обучения и могут работать непосредственно после создания.
- d) Нейронные сети применяются только в области медицины.

Ответ: b

Задание 7. Какой компонент является основным строительным блоком нейронных сетей?

- a) Алгоритмы сортировки.
- b) Графические процессоры (Gh2U).
- c) Искусственные нейроны (нейроны).
- d) Ключи шифрования.

Ответ: c

Задание 8. Что такое «обучение с учителем» в контексте нейронных сетей?

- a) Процесс самостоятельного обучения нейронов.
- b) Использование сетей для создания новых учителей.
- c) Обучение модели на основе пары входных данных и соответствующих выходных данных.
- d) Обучение нейронов на основе случайных данных.

Ответ: c

Задание 9. Какая роль у «скрытых слоев» в нейронных сетях?

- a) Скрытие внутренних параметров сети от внешнего мира.
- b) Определение целевых переменных.
- c) Извлечение сложных признаков из входных данных.
- d) Скрытие процесса обучения от пользователя.

Ответ: c

Задание 10. Что такое функция активации в нейронной сети?

- a) Вес, присваиваемый каждому связанному нейрону
- b) Специальная функция, определяющая выход нейрона
- c) Метод вычисления ошибки
- d) Число слоев в нейронной сети

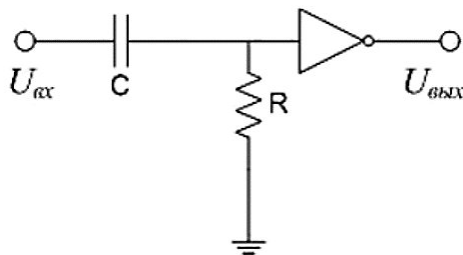
Ответ: c

ЗАДАНИЯ ОТКРЫТОГО ТИПА (15 заданий, из них 5 – с развернутым ответом)

-----Уметь-----

Задание 1. Опишите работу электронного Nv нейрона, нарисуйте схему.

Ответ: Схема электронного Nv нейрона (быстрого нейрона, аналог пороговая активационная функция)

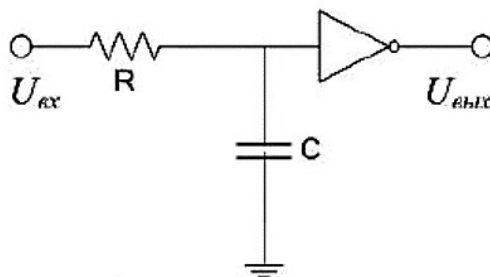


Принцип работы:

- Состояние «выключено» высокий выходной уровень.
- Возрастающий сигнал на входе (включает) заставляет нейрон мгновенно переключиться. После задержки нейрон выключается (сначала «включено», затем задержка, затем «выключено»)

Задание 2. Опишите работу электронного Nu нейрона, нарисуйте схему.

Ответ: Схема электронного Nu нейрона (медленного нейрона, аналог сигмоидальная активационная функция)



- Состояние «выключено» высокий выходной уровень.
- Высокий входной сигнал переключает нейрон – после задержки. Нейрон остается включенным, пока на вход подается высокий уровень сигнала (задержка, затем «включено», потом «включено»).

Задание 3. Проанализируйте приведенный код на Python, опишите что он реализует, изобразите схему и таблицу, закомментируйте программу.

```
import numpy as np

def activation_function(y):
    if y > 0:
        y = 1
    elif y <= 0:
        y = 0
    return y

W = np.array([-1])
b = 1

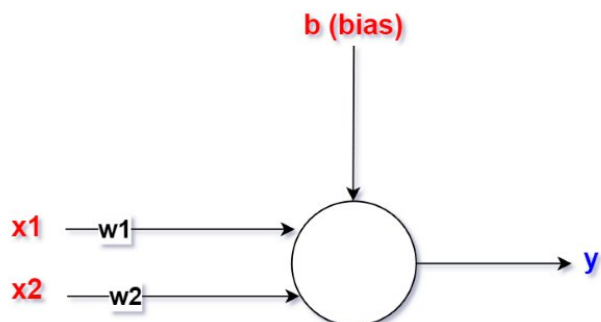
def perceptron_algorithm(x):
    y = np.dot(W, x) + b
    y = activation_function(y)
    return y

input1 = np.array([0])
input2 = np.array([1])

print('NOT Logic: \n')
print(f'x = 0 => y = {perceptron_algorithm(input1)}')
print(f'x = 1 => y = {perceptron_algorithm(input2)}')
```

Ответ. Данная программа реализует персептрон реализующим логическую функцию НЕ, который можно определить как нейронную сеть прямого распространения с одним скрытым слоем. Он принимает определенное количество входных данных (в данном случае x_1 и x_2), обрабатывает их с использованием алгоритма персептрона и затем, наконец, выдает выходной сигнал y , который может быть либо 0, либо 1.

X	Y
1	0
0	1



Комментарии кода


```

# Импорт необходимых библиотек
import numpy as np

# Определение функции активации
def activation_function(y):
    if y > 0:
        y = 1
    elif y <= 0:
        y = 0
    return y

# W = веса модели перцептрона
W = np.array([-1])
# b = смещение
b = 1

# Определение алгоритма перцептрона
def perceptron_algorithm(x):
    # y = w1x1 + b
    y = np.dot(W, x) + b
    # y = 1 if Wx+b > 0 else y = 0
    y = activation_function(y)
    return y

# Входные значения для проверки логики НЕ
input1 = np.array([0])
input2 = np.array([1])

# Печать результата
print('NOT Logic: \n')
print(f'x = 0 => y = {perceptron_algorithm(input1)}')
print(f'x = 1 => y = {perceptron_algorithm(input2)}')

```

Задание 4. Проанализируйте приведенный код на Python, опишите что он реализует, изобразите схему и таблицу, прокомментируйте программу.

```

import numpy as np

weight = np.array([1,1])
bias = -0.5
inputs = np.array([[0, 0], [0, 1], [1, 0], [1, 1]])

def OR_perceptron(x, weight, bias):

    fx = np.dot(weight, x) + bias

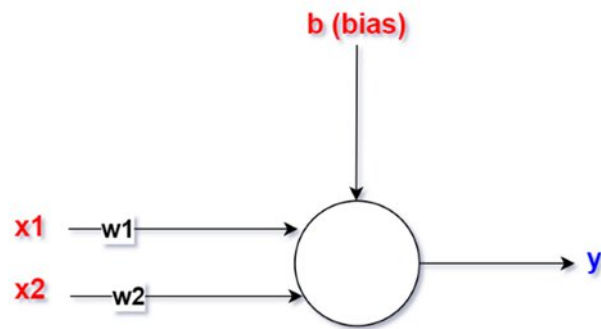
    if fx >= 0:
        return 1
    else:
        return 0

for i in inputs:
    print(OR_perceptron(i, weight, bias))

```

Ответ. Данная программа реализует персептрон реализующим логическую функцию ИЛИ, который можно определить как нейронную сеть прямого распространения со ступенчатой функцией активации.

X	Y	X Y
1	1	1
1	0	1
0	1	1
0	0	0



Комментарии кода

```
# Импорт необходимых библиотек
import numpy as np

# весовые параметры
weight = np.array([1,1])
# параметр смещения
bias = -0.5
# входные значения для проверки логики ИЛИ
inputs = np.array([[0, 0], [0, 1], [1, 0], [1, 1]])

# Определение алгоритма перцептрона
def OR_perceptron(x, weight, bias):

    fx = np.dot(weight, x) + bias

    if fx >= 0:
        return i, 1
    else:
        return i, 0
# Печать результата
for i in inputs:
    print(OR_perceptron(i, weight, bias))
```

Задание 5. Проанализируйте приведенный код на Python, опишите что он реализует, изобразите схему и таблицу, закомментируйте программу.

```

import numpy as np

def Steps(v):
    if v >= 0:
        return 1
    else:
        return 0

def perceptron(x, w, b):
    v = np.dot(w, x) + b
    y = Steps(v)
    return y

def logic_AND(x):
    w = np.array([2, 2])
    b = -1
    return perceptron(x, w, b)

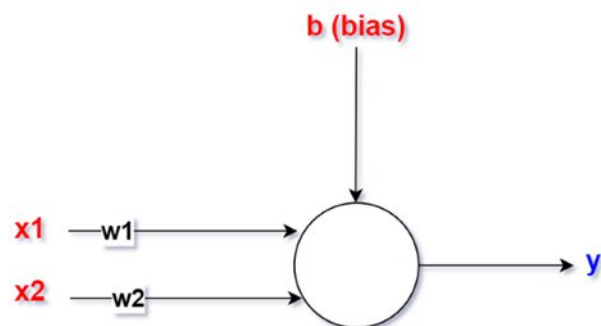
p1 = np.array([0, 1])
p2 = np.array([1, 1])
p3 = np.array([0, 0])
p4 = np.array([1, 0])

print("AND(0, 1) = {}".format( logic_AND(p1)))
print("AND(1, 1) = {}".format( logic_AND(p2)))
print("AND(0, 0) = {}".format( logic_AND(p3)))
print("AND(1, 0) = {}".format( logic_AND(p4)))

```

Ответ. Данная программа реализует персептрон реализующим логическую функцию ИЛИ, который можно определить как нейронную сеть прямого распространения.

X	Y	X & Y
0	0	0
0	1	0
1	0	0
1	1	1



Комментарий кода

```

# Импорт необходимых библиотек
import numpy as np

# Блок реализации шага
def Steps(v):
    if v >= 0:
        return 1
    else:
        return 0

# Определение алгоритма перцептрона
def perceptron(x, w, b):
    v = np.dot(w, x) + b
    y = Steps(v)
    return y

# Реализация функция логики И
def logic_AND(x):
    # весовые параметры
    w = np.array([2, 2])
    # параметр смещения
    b = -1
    return perceptron(x, w, b)

# Тестирование модели персептрона
p1 = np.array([0, 1])
p2 = np.array([1, 1])
p3 = np.array([0, 0])
p4 = np.array([1, 0])

# Печать результата
print("AND(0, 1) = {}".format( logic_AND(p1)))
print("AND(1, 1) = {}".format( logic_AND(p2)))
print("AND(0, 0) = {}".format( logic_AND(p3)))
print("AND(1, 0) = {}".format( logic_AND(p4)))

```

Задание 6. По фрагменту кода определите, что вычисляется сформулируйте свой ответ.

```

import numpy as np

actual = [12, 13, 14, 15, 15, 22, 27]
pred = [11, 13, 14, 14, 15, 16, 18]

def mse(actual, pred):
    actual, pred = np.array(actual), np.array(pred)
    return np.square(np.subtract(actual, pred)).mean()

print(mse(actual, pred))

```

Ответ. MSE (средняя квадратическая ошибка) — это оценка среднего значения квадрата ошибок, различие между предсказанием и фактическим значением. Эту метрику удобно использовать для выявления аномалий. Выражается формулой

$$\text{MSE} = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2$$

где n — количество наблюдений по которым строится модель и количество прогнозов,
 y_i — фактические значение i -го наблюдения,
 $\hat{y}_i$ значение зависимой переменной, предсказанное моделью.

Задание 7. Что такое MNIST и для чего это надо, какой командой можно вызвать в Python.

Ответ. Название расшифровывается как Modified National Institute of Standards and Technology — «Модифицированная [база данных] Национального института стандартов и технологий США». Набор данных MNIST активно применяют в машинном обучении, в частности в создании и обучении нейронных сетей. Это такие цифровые модели, которые по структуре повторяют соединения нейронов в человеческом мозгу. Их можно обучать, но для этого нужны большие массивы данных — так программа «запомнит», как выглядит то, с чем она имеет дело. Например, MNIST есть среди данных библиотеки Keras. Чтобы добавить датасет к программе, нужно просто указать в начале кода строчку:

```
from keras.datasets import mnist # импорт набора данных MNIST
```

Задание 8. По коду определите, что получится на выходе, для чего это необходимо.

```
from faker import Faker

class FakerDemo:
    def __init__(self):
        # генерируем данные из России.
        self.fake = Faker("ru_RU")

    def person(self, count):
        msg = ""
        for i in range(count):
            num = i + 1
            name = self.fake.name()
            phone = self.fake.phone_number()
            email = self.fake.email()
            msg += f"{num},{name},{phone},{email}\n"
        return msg

if __name__ == '__main__':
    # Создание экземпляра объекта FakeDemo
    f = FakerDemo()
    data = f.person(5)

    with open("data.csv", "w") as d:
        d.write(data)
```

Ответ. В данном коде используется фреймворк Faker — это пакет Python, разработанный для упрощения генерации синтетических данных. Устанавливается командой

```
pip install Faker
```

Данный пакет прост и интуитивно понятен в использовании для обучения нейронных сетей. В данной программе генерируются синтетические данные 5 человек из России с порядковым номером, именем отчеством фамилией, номером телефона и емейлом с сохранением в csv файле с именем data.

1,Рыбаков Любим Бориславович,8 (074) 945-4234,ffedoseev@example.com
2,Кириллова Юлия Константиновна,+7 (067) 058-3680,qevseeva@example.org
3,София Антоновна Горбачева,8 144 765 4415,strelkovaelizaveta@example.net
4,Белозерова Дарья Максимовна,+7 (861) 182-8219,svjatoslav2017@example.com
5,Шестаков Ферапонт Игнатович,8 591 454 4896,erast48@example.net

Задание 9. По коду определите, что получится на выходе, для чего это необходимо.

```
from mimesis import Address
from mimesis import Person
from mimesis.locales import Locale

ru = Address(locale=Locale.RU)
person = Person(locale=Locale.RU)

print (ru.region(), ru.federal_subject(), ru.address())
print (person.full_name())
```

Ответ. В данном коде используется фреймворк Mimesis — это пакет Python, разработанный для упрощения генерации синтетических данных. Устанавливается командой

pip install mimesis

Это полноценный генератор случайной информации, построенный на базе Faker. Он может генерировать гораздо больше случайных атрибутов, чем Faker. Он также поддерживает информацию о конкретной стране для 32 локализаций (языков). Ниже мы генерируем тысячу строк поддельных данных на русском языке. В данной программе генерируется синтетические данные человека из России с регионом, субъектом РФ, адресом и именем. Формат вывода

Дагестан Астраханская область ул. Научная 90
Парфений Дмитриев

Задание 10. По приведенному коду определите какая библиотека Python используется в данном коде, что это за библиотека и что реализовано в коде. Прокомментируйте код.

```
import cv2

face_cascade = cv2.CascadeClassifier('haarcascade_frontalface_default.xml')

image = cv2.imread('matrix.jpg')

gray_image = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)

faces = face_cascade.detectMultiScale(gray_image, scaleFactor=1.1, minNeighbors=5)

for (x, y, w, h) in faces:
    cv2.rectangle(image, (x, y), (x + w, y + h), (0, 0, 255), 2)

cv2.imshow('found_faces', image)
cv2.waitKey(0)
```

Ответ. В приведенном фрагменте кода использована библиотека OpenCV (Библиотека компьютерного зрения с открытым исходным кодом) — это библиотека для обработки изображений и задач компьютерного зрения. Используя ее в Python, можно создавать эффективные приложения для таких задач, как обработка изображений, обнаружение объектов и распознавание лиц. В данном коде используется обученная каскадная модель Нагг для распознавания лиц.

```
import cv2

# создаём переменную с файлом модели
face_cascade = cv2.CascadeClassifier('haarcascade_frontalface_default.xml')

# читаем изображение
image = cv2.imread('matrix.jpg')

# обесцвечиваем изображение
gray_image = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)

# находим лица на обесцвеченном изображении
faces = face_cascade.detectMultiScale(gray_image, scaleFactor=1.1, minNeighbors=5)

# на цветном изображении рисуем квадраты там, где нашли лица на обесцвеченном
for (x, y, w, h) in faces:
    cv2.rectangle(image, (x, y), (x + w, y + h), (0, 0, 255), 2)

# открываем изображение в отдельном окне
cv2.imshow('found_faces', image)
cv2.waitKey(0)
```

-----Владеть-----

Задание 11. По приведенному коду определите какая библиотека Python используется в данном коде, что это за библиотека и что реализовано в коде.

```
import tensorflow as tf

# define the model
model = tf.keras.Sequential([
    tf.keras.layers.Dense(10, input_shape=(8,), activation='relu'),
    tf.keras.layers.Dense(1, activation='sigmoid')
])

# compile the model
model.compile(optimizer='adam', loss='binary_crossentropy', metrics=['accuracy'])
```

Ответ. В приведенном фрагменте кода использована библиотека TensorFlow — это библиотека с открытым исходным кодом, разработанная Google для создания и развёртывания моделей машинного обучения. Это одна из самых популярных библиотек для искусственного интеллекта и машинного обучения. TensorFlow подходит для построения нейронных сетей и моделей глубокого обучения, а также обладает широким спектром инструментов для построения и обучения моделей. В приведенном примере построена простая нейронная сеть с входными десятью нейронами имеющих по восемь входных параметров и активационной функцией RELU и одним выходным нейроном с сигмоидальной активационной функцией.

Задание 12. По приведенному коду определите какая библиотека Python используется в данном коде, что это за библиотека и что реализовано в коде.

```
from sklearn.linear_model import LinearRegression

# create the model
model = LinearRegression()

# fit the model to the data
model.fit(X_train, y_train)

# make predictions
y_pred = model.predict(X_test)
```

Ответ. В приведенном фрагменте кода использована библиотека Scikit-learn — это широко используемая библиотека для машинного обучения на Python. Она построена поверх NumPy и SciPy и предлагает широкий спектр инструментов для создания и оценки моделей

машинного обучения. Scikit-learn отлично подходит для построения традиционных моделей машинного обучения. В приведенном примере построена простая модель линейной регрессии.

Задание 13. По приведенному коду определите какая библиотека Python используется в данном коде, что это за библиотека и что реализовано в коде.

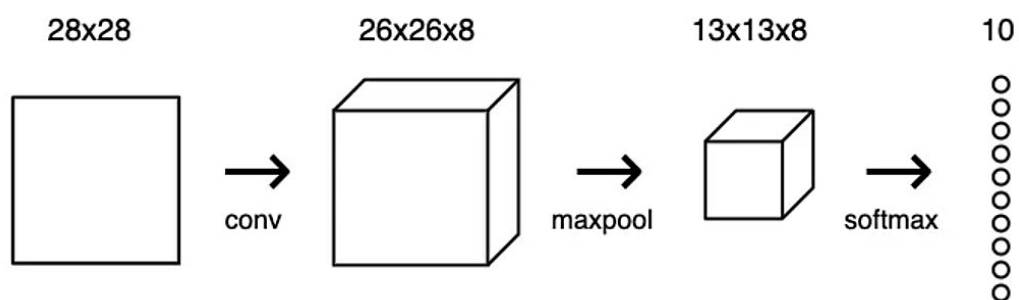
```
from keras.models import Sequential
from keras.layers import Conv2D, MaxPooling2D, Flatten, Dense

num_filters = 8
filter_size = 3
pool_size = 2

model = Sequential([
    Conv2D(num_filters, filter_size, input_shape=(28, 28, 1)),
    MaxPooling2D(pool_size=pool_size),
    Flatten(),
    Dense(10, activation='softmax'),
])
```

Ответ. В приведенном фрагменте кода использована библиотека Keras — это высокоуровневая библиотека нейронных сетей для Python. Она создана поверх TensorFlow и предназначена для того, чтобы максимально упростить построение и обучение нейронных сетей. Keras отлично подходит для построения моделей глубокого обучения и обладает широким спектром инструментов для построения и обучения моделей. В приведенном примере построена простая сверточная нейронная сеть:

- num_filters, filter_size и pool_size — это необъяснимые переменные, задающие гиперпараметры для нашей CNN.
- Первый слой в любой Sequential модели должен задавать форму входа input_shape, поэтому мы делаем это на Conv2D. После задания формы входа Keras автоматически определяет формы входов для последующих слоев.
- Выходной слой Softmax имеет 10 узлов, по одному на каждый класс.



Задание 14. По приведенному коду определите какая библиотека Python используется в данном коде, что это за библиотека и что реализовано в коде.

```
import pandas as pd

# load the data
data = pd.read_csv('data.csv')

# explore the data
print(data.head())
print(data.describe())
```

Ответ. В приведенном фрагменте кода использована библиотека Pandas — это библиотека для обработки и анализа данных на Python. Она широко используется для работы со

структурированными данными и отлично подходит для очистки, преобразования и анализа данных. Pandas имеет широкий спектр инструментов для работы с данными. В данном примере библиотека Pandas использована для загрузки данных из csv файла с именем data и распечатки набора данных/

Задание 15. По приведенному коду определите какая библиотека Python используется в данном коде, что это за библиотека и что реализовано в коде.

```
import nltk

# download the necessary resources
nltk.download('punkt')

# tokenize a sentence
sentence = "NLTK упрощает обработку текста."
tokens = nltk.word_tokenize(sentence)
print(tokens)
```

Ответ. В приведенном фрагменте кода использована библиотека NLTK (Natural Language Toolkit) — это библиотека для обработки естественного языка в Python. Она широко используется для работы с текстовыми данными и отлично подходит для таких задач, как классификация текста, анализ отношений и языковой перевод. NLTK обладает широким спектром инструментов для работы с текстовыми данными, включая токенизацию, стемминг и лемматизацию. Этот код загрузит необходимые данные punkt, которые используются для токенизации текста. Результат кода: ['NLTK', 'упрощает', 'обработку', 'текста', '.']

6. ЛИСТ СОГЛАСОВАНИЯ

Составил:

Н.А. Иванова, ст.преп.

A handwritten signature in blue ink, appearing to read 'Иванова', is written above a horizontal line.

(подпись)

Заведующий кафедрой

Н.Б.Стрекалова, д.п.н., доцент

A handwritten signature in blue ink, appearing to read 'Стрекалова', is written above a horizontal line.

(подпись)

Заведующий выпускающей кафедрой

Н.Б.Стрекалова, д.п.н., доцент

A handwritten signature in blue ink, appearing to read 'Стрекалова', is written above a horizontal line.

(подпись)