

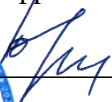
Наименование учредителя: Правительство Российской Федерации
Федеральное государственное образовательное бюджетное учреждение
высшего образования
**«Финансовый университет при Правительстве Российской Федерации»
(Финансовый университет)**

ЦЕНТР КОМПЕТЕНЦИЙ «ЦИФРОВАЯ ЭКОНОМИКА»

УТВЕРЖДАЮ

Директор Центра компетенций
«Цифровая экономика»



 Н.Ю. Сурова

«09» июля 2025 г.

ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА
«Программирование и обучение нейронных сетей
(базовый уровень)»

Формы реализации – очная без применения дистанционных образовательных технологий (в том числе, с применением средств электронного обучения)

Москва 2025 г.

Дополнительная общеобразовательная программа «Программирование и обучение нейронных сетей (базовый уровень)»

1. Пояснительная записка

Актуальность

Изучение программирования и обучения нейронных сетей актуально из-за потенциала этих сквозных технологий для прогресса в различных областях (медицина, образование, геологоразведка и пр.). Нейронные сети имитируют работу человеческого мозга, и их изучение позволяет решать сложные задачи, анализировать большие объёмы данных и создавать интеллектуальные системы.

Навык программирования и обучения нейронных сетей – составляющая цифровой компетенции.

Особенности и новизна программы на базовом уровне:

- Содержание учебного материала включает вопросы, формирующие концептуальное представление о нейронных сетях с переходом на узкоспециализированные области (компьютерное зрение, распознавание речи). Обеспечивается целостность образовательных результатов.
- Практикоориентированный подход к реализации программы предполагает практические задания для отработки умений и работу над собственным учебным проектом в контексте конкретных тем.
- Использование современных технологий и инструментов для программирования и обучения нейронных сетей (Pandas, Matplotlib, методы загрузки данных и пр.).
- Решение кейсов вводит обучающихся в процесс работы над прикладными задачами, где применимы навыки программирования и работа с данными.
- Предусмотрены темы ориентированные на профориентацию в сфере ИИ, что поможет в самоопределении и проектировании будущего профессионального пути.
- Программа учитывает возрастные особенности и интересы школьников 8-11 классов и студентов колледжей в необходимости изучения востребованных умений и навыков в интересных форматах.

Описание программы

Данная программа предназначена для получения фундаментальных знаний и практических навыков в области искусственного интеллекта, опираясь на уверенное владение языком программирования Python.

Формы реализации – очная без применения дистанционных образовательных технологий (в том числе, с применением средств электронного обучения).

Общий объем программы 146 академических часов. Длительность и количество занятий – 2 или 3 раза в неделю по 2 академических часа (1 академический час равен 45 минутам, не включая перерыв).

Программа включает 4 модуля, каждый из которых завершается промежуточной аттестацией – выполнением теста. В каждом модуле в рамках практической и самостоятельной работы предусмотрено решение практических задач. По мере освоения программы

обучающиеся выполняют разнообразные упражнения по созданию игр, приложений с использованием нейронных сетей.

Содержание модуля 1 программы включает рассмотрение типов данных, особенности загрузки данных из различных источников, основы кибербезопасности и соблюдения конфиденциальности при использовании различных фреймворков и библиотек. Основная часть программы (модули 2–4) посвящен всестороннему изучению машинного обучения, принцип работы базовых моделей (модуль 2); принцип работы перцептрона как базовой модели нейронной сети с использованием библиотек TensorFlow, PyTorch, Keras (модуль 3); применение методов машинного обучения и использование нейронных сетей для анализа изображений, практическое использование библиотек и фреймворков для решения прикладных задач (модуль 4).

В рамках указанных модулей обучающиеся также познакомятся с новыми профессиями в сфере искусственного интеллекта и рассмотрят практическое использование библиотек и фреймворков для решения прикладных задач.

Программа обеспечена учебно-методическими ресурсами для качественного освоения содержания учебного материала, что также обеспечивает результативность освоения модулей. Обучающимся обеспечен доступ к современному практикоориентированному содержанию электронных образовательных и информационных ресурсов. Предусмотрены материально-технические условия – программное обеспечение и оборудование.

Преподавательский состав включает IT-специалистов, имеющих большой педагогический стаж в области преподавания языков программирования и создания нейронных сетей.

Аннотация

Дополнительная общеобразовательная программа «Программирование и обучение нейронных сетей (базовый уровень)» включает 4 модуля, каждый из которых завершается промежуточным тестированием. При освоении образовательной программы предусмотрены теоретическая подготовка, практические занятия, самостоятельная работа, работа над собственным учебным проектом, контроль знаний. Направленность дополнительной общеобразовательной программы – техническая.

Программа базового уровня сложности подойдет школьникам 8-11 классов и студентам колледжей, которые имеют базовые знания по основам программирования и математики, умеют писать простые программы, имеют представление о линейной алгебре и теории вероятностей и хотят научиться основам машинного обучения нейросетей.

Обучающиеся научатся:

- Работать с типами и структурами данных.
- Проводить предобработку данных для анализа и моделирования.
- Проводить очистку данных и их подготовку для использования в моделях.
- Анализировать данные с помощью статистических методов и визуализации.
- Различать типы задач ML.
- Строить и оценивать простые модели.
- Работать с scikit-learn.
- Понимать структуру нейронных сетей и их работу.
- Применять базовые алгоритмы обучения нейросетей.
- Создавать и обучать модели с помощью библиотек.
- Применять ML и ИИ для реальных задач.
- Работать с нейросетями в компьютерном зрении.
- Обработать текстовые данные с помощью ИИ.
- Использовать популярные библиотеки для решения прикладных задач.

В результате обучения по программе обучающиеся смогут научиться программированию и обучению нейронных сетей для использования в реальной жизни и профессиональных задачах.

Обучение предусматривает очную форму с применением электронного обучения, без применения дистанционных образовательных технологий, и проводится очно на площадке. Рекомендуется последовательно изучать каждую тему в модулях, читать краткие конспекты и выполнять практические задания. Полезно задавать вопросы, чтобы лучше понять материал тем. При завершении каждого модуля нужно пройти небольшой тест.

После освоения программы базового уровня обучающиеся смогут работать с данными, применять библиотеки для создания простых нейросетевых моделей и ИИ для решения прикладных задач. Эти умения полезны в разных сферах (например, в медицине, финансах, геологоразведке и пр.), где требуется контроль, классификация или прогнозирование. Дополнительная общеобразовательная программа носит практикоориентированный характер и включает более 50% часов практической работы, что поможет освоить конкретные умения и навыки по программированию и обучению нейронных сетей. Предлагается разработка собственного учебного проекта.

Цель программы: сформировать у обучающихся базового понимания технологий ИИ, освоение методов машинного обучения и анализа данных, а также получение первоначальных навыков применения ИИ для решения простых прикладных задач.

Задачи:

научить методам работы с данными, включая их предобработку и базовый анализ для подготовки к использованию в моделях ИИ.

сформировать представление об остальных типах задач машинного обучения и базовых моделей для их решения.

сформировать представление о базовых принципах работы нейронных сетей и возможности глубокого обучения.

сформировать умения и навыки применения методов и моделей ИИ для решения прикладных задач.

Формы и режим занятий

Формы занятий – теоретическая подготовка, практическая работа, самостоятельная работа.

Режим занятий – 2 или 3 раза в неделю по 2 академических часа (1 академический час равен 45 минутам, не включая перерыв). Объем часов в неделю – 6 ак. часов.

Описание планируемых результатов обучения:

Сформированные у обучающихся начальные умения и навыки создания нейронных сетей на Python.

Обучающиеся будут знать:

принцип работы перцептрона, роль функций активации;

типы данных и способами их представления;

типы задач машинного обучения (классификация, регрессия, кластеризация);

архитектуру нейронных сетей: нейроны, слои, веса и смещения;

основы обучения нейронных сетей: обратное распространение ошибки и градиентный спуск.

Обучающиеся будут уметь:

осуществлять загрузку данных из различных источников;
 использовать библиотеку scikit-learn для решения задач;
 использовать библиотеку для создания простых нейросетевых моделей;
 использование нейронных сетей для анализа изображений и основами компьютерного зрения;

использовать библиотеки и фреймворки для решения прикладных задач;
 применять базовый статистический анализ и визуализация данных;
 использовать специализированные библиотеки для работы с данными (Pandas).

Обучающиеся будут владеть:

методами очистки данных: удаление пропусков и дубликатов;
 базовыми алгоритмами: линейная регрессия, k-NN, логистическая регрессия, деревья решений, SVM;

методами машинного обучения для решения задач классификации и регрессии;
 базовыми методами обработки и анализа текстовых данных с помощью ИИ.

Личностные результаты:

осознание ценности самообразования и непрерывного развития в сфере информационных технологий;
 формирование критического мышления и способности к анализам сложных проблем;
 развитие ответственности за результаты своей работы и умение работать в команде;
 проявление инициативности и творческого подхода к решению поставленных задач;
 осознание важности соблюдения этических норм при использовании технологий машинного обучения.

Метапредметные результаты:

умение самостоятельно планировать и организовывать свою деятельность при решении учебных и практических задач;
 умение самостоятельно анализировать и обрабатывать информацию из различных источников;
 умения структурировать информацию и представлять её в понятной форме.
 умение применять математические методы и алгоритмы для решения задач машинного обучения.

Предметные результаты:

представление о нейронных сетях;
 умение разрабатывать и обучать нейронные сети с использованием современных библиотек и фреймворков;
 знания методов оптимизации и регуляризации нейронных сетей;
 умение оценивать качество работы обученных моделей и проводить их отладку;
 приобретение практических навыков применения нейронных сетей для решения задач.

2. Учебный план

Наименование модулей/тем программы	Всего, час	Виды учебных занятий				Формы контроля
		Теория	Практика	Самостоятельная работа	Контроль/аттестация, ак.час	
Модуль 1. «Основы работы с данными»	36	6	22	6	2	
Тема 1.1. Типы данных и их представление	6	1	4	1	0	
Тема 1.2. Загрузка данных из различных источников. Изучение кибербезопасности и конфиденциальность при использовании различных фреймворков и библиотек.	10	2	6	1	1	Тестирование для отслеживания текущих результатов в обучения.
Тема 1.3. Очистка данных: удаление пропусков и дубликатов	7	2	4	1	0	
Тема 1.4. Базовый анализ данных: статистика и визуализация при помощи Seaborn и Matplotlib. Предобработка данных для машинного обучения на примере технологии Блокчейн.	8	1	6	1	0	
Тема 1.5. Работа над учебным проектом	4	0	2	2	0	
Промежуточная аттестация по модулю 1.	1	0	0	0	1	Тестирование
Модуль 2. «Введение в машинное обучение»	36	8	20	6	2	
Тема 2.1. Основные типы задач машинного обучения: классификация, регрессия и	7	2	4	1	0	

Наименование модулей/тем программы	Всего, час	Виды учебных занятий				Формы контроля
		Теория	Практика	Самостоятельная работа	Контроль/аттестация, ак.час	
их применение в робототехнике.						
Тема 2.2. Принципы работы базовых моделей: линейная регрессия, метод ближайших соседей	7	1	4	1	1	Тестирование для отслеживания текущих результатов в обучения.
Тема 2.3. Алгоритмы классификации: логистическая регрессия, дерево решений, метод опорных векторов	8	1	6	1	0	
Тема 2.4. Использование библиотеки scikit-learn	9	4	4	1	0	
Тема 2.5. Работа над учебным проектом	4	0	2	2	0	
Промежуточная аттестация по модулю 2.	1	0	0	0	1	Тестирование
Модуль 3. «Основы нейронных сетей и глубокого обучения»	36	8	20	6	2	
Тема 3.1 Архитектура нейронных сетей: нейроны, слои, веса и смещения	7	2	4	1	0	
Тема 3.2. Принцип работы перцептрона как базовой модели нейронной сети. Функции активации и их роль.	8	2	4	1	1	Тестирование для отслеживания текущих результатов в обучения.
Тема 3.3. Основы обучения нейронных сетей: метод обратного	7	2	4	1	0	

Наименование модулей/тем программы	Всего, час	Виды учебных занятий				Формы контроля
		Теория	Практика	Самостоятельная работа	Контроль/аттестация, ак.час	
распространения ошибки и градиентный спуск						
Тема 3.4.Использование библиотек TensorFlow, PyTorch, Keras для работы с нейронными сетями и создание простых моделей	9	2	6	1	0	
Тема 3.5. Работа над учебным проектом	4	0	2	2	0	
Промежуточная аттестация по модулю 3.	1	0	0	0	1	Тестирование
Модуль 4. «Применение искусственного интеллекта для типовых задач»	36	6	22	6	2	
Тема 4.1. Применение методов машинного обучения для решения задач классификации и регрессии	8	1	6	1	0	
Тема 4.2. Использование нейронных сетей для анализа изображений: основы компьютерного зрения. Применение моделей в проектах дополненной реальности AR совместно с OpenCV.	9	1	6	1	1	Тестирование для отслеживания текущих результатов в обучения.
Тема 4.3. Базовые методы обработки и анализа текстовых данных с помощью ИИ	7	2	4	1	0	
Тема 4.4. Новые профессии в ИИ. Практическое использование библиотек и фреймворков для	7	2	4	1	0	

Наименование модулей/тем программы	Всего, час	Виды учебных занятий				Формы контроля
		Теория	Практика	Самостоятельная работа	Контроль/аттестация, ак.час	
решения прикладных задач.						
Тема 4.5. Работа над учебным проектом	4	0	2	2	0	
Промежуточная аттестация по модулю 4.	1	0	0	0	1	Тестирование
Итоговый контроль	0	0	0	0	2	Тестирование
Итого	146	28	84	2	12	
ИТиСИ	Часы ИТиСИ не входят в общую трудоемкость ДОП, но участие в ИТиСИ обязательно для всех успешно прошедших итоговый контроль/аттестацию по ДОП					

3. Календарно-тематическое планирование

Название ДОП	№ потока	Дата начала обучения по ДОП	№ модуля ДОП	Начало обучения по модулю ДОП	Календарный период (количество дней) Длительности модуля	Дата окончания обучения по ДОП
Программирование и обучение нейронных сетей (базовый уровень)	1	2025-10-01	1	2025-10-01	59	2026-05-29
			2	2025-12-01	68	
			3	2026-02-09	54	
			4	2026-04-06	54	

4. Рабочая программа

Порядковый номер модуля	Наименование темы	Лекции. Количество академических часов	Содержание лекций	Практические занятия. Количество академических часов	Содержание практических занятий	Самостоятельная работа. Количество академических часов	Содержание самостоятельной работы
1	Тема 1.1. Типы данных и их представление	1	Изучение основных типов данных (числовые, категориальные, текстовые и др.), форматов представления данных, особенностей их хранения и обработки. Понимание структуры данных и способов их представления в программных средствах.	4	Изучение дополнительного материала по теме, выполнение задания для самостоятельной работы для отработки умений и навыков по написанию простой программы. Выполнение заданий на тренажере на проверку данных, создание списков, создание кортежа, написание функции.	1	Изучение дополнительного учебного материала о множествах и замороженных множествах в Python. Выполнение задания для самостоятельной работы.
1	Тема 1.2. Загрузка данных из различных источников. Изучение кибербезопасности и конфиденциальности при использовании различных	2	Ознакомление с методами загрузки данных из файлов (CSV, Excel, JSON и др.), базами данных. Изучение и практическое применение библиотек для работы с данными	6	Выполнение заданий по написанию программы, на загрузку данных из SQLite и вывод результата. Выполнение заданий на тренажере на создание базы данных, на добавление записи в базу данных, на загрузку CSV-	1	Изучение дополнительного учебного материала о библиотеке Pandas в Python. Выполнение задания для самостоятельной работы.

	фреймворков и библиотек.		(например, Pandas).		файла с помощью Pandas и на загрузку Excel-файла, на проверку существования файла, на подсчет строк в файле, на чтение базы.		
1	Тема 1.3. Очистка данных: удаление пропусков и дубликатов	2	Изучение методов очистки данных: обнаружение и удаление пропущенных значений, дубликатов, обработка аномалий. Практические приемы подготовки данных для анализа и моделирования.	4	Выполнение задания с разбором на загрузку и первичный анализ датасета. Выполнение заданий на тренажере на заполнение пропусков медианой, на удаление строк с пропусками, на проверку и удаление дубликатов, на удаление ненужных столбцов и вывод уникальных значений.	1	Изучение дополнительного учебного материала про объединение таблиц в Pandas. Выполнение задания для самостоятельной работы. Проведение самопроверки знаний в игровой форме – выполнение тестовых заданий с игровым правилом по ограничению во времени и с получением мгновенного результата.
1	Тема 1.4. Базовый анализ данных: статистика и визуализация при помощи Seaborn и Matplotlib. Предобработка данных для машинного обучения на примере технологии Блокчейн.	1	Основы статистического анализа данных: расчет основных статистических показателей (среднее, медиана, дисперсия и др.). Введение в визуализацию данных с помощью графиков и диаграмм (библиотеки Matplotlib, Seaborn).	6	Выполнение задания с разбором на создание сводной таблицы с использованием pivot_table, на группировку и усреднение с использованием groupby и unstack. Выполнение заданий на тренажере на определение среднего значения данных, на вычисление медианного значения, на определение максимального значения	1	Изучение дополнительного учебного материала о блокчейне. Выполнение задания для самостоятельной работы.

					данных, на подсчет количества, на создание сводных таблиц.		
1	Тема 1.5. Работа над учебным проектом	0	-	2	Выполнение учебного проекта согласно варианту и плану. Подготовка отчета о выполнении учебного проекта	2	Изучение дополнительного учебного материала о типах данных (маленькие хитрости). Завершение работы над учебным проектом в соответствии с вариантом и планом. Подготовка отчета о его выполнении.
2	Тема 2.1. Основные типы задач машинного обучения: классификация, регрессия и их применение в робототехнике.	2	Изучение основных типов задач ML: классификация, регрессия, краткий обзор других задач (кластеризация, снижение размерности, аномалии). Формирование представления, для чего и как решаются разные задачи.	4	Выполнение задания с разбором на классификацию данных, на регрессию. Оработка навыков в ходе выполнения заданий на тренажере на классификацию и регрессию.	1	Изучение дополнительного учебного материала о машинном обучении. Выполнение задания для самостоятельной работы.
2	Тема 2.2. Принципы работы базовых моделей: линейная регрессия, метод	1	Изучение линейной регрессии и метода ближайших соседей (k-NN): принципы работы, примеры,	4	Выполнение задания с разбором на линейную регрессию, на использование метода ближайших соседей (k-	1	Изучение дополнительного учебного материала по сравнению базовых моделей (линейная регрессия, логистическая регрессия,

	ближайших соседей		преимущества и ограничения.		NN). Отработка навыков в ходе выполнения заданий на тренажере на линейную регрессию, на использование метода ближайших соседей (k-NN).		метод ближайших соседей (k-NN)). Выполнение задания для самостоятельной работы.
2	Тема 2.3. Алгоритмы классификации: логистическая регрессия, дерево решений, метод опорных векторов	1	Рассмотрение алгоритмов логистической регрессии, дерева решений и метода опорных векторов: как они работают, где применяются.	6	Выполнение задания с разбором на использование дерева решений, на использование метода опорных векторов (SVM). Отработка навыков в ходе выполнения заданий на тренажере на использование дерева решений, на использование метода опорных векторов (SVM).	1	Изучение дополнительного учебного материала про индекс Джини и Энтропии (понятие и использование). Проведение самопроверки знаний в игровой форме – выполнение тестовых заданий с игровым правилом по ограничению во времени и с получением мгновенного результата.
2	Тема 2.4. Использование библиотеки scikit-learn	4	Ознакомление с библиотекой scikit-learn: загрузка данных, создание моделей, обучение, оценка качества.	4	Выполнение задания с разбором на обучение логистической регрессии на датасете Iris и на сгенерированных данных. Отработка навыков в ходе выполнения заданий на тренажере на загрузку датасета Iris и разделение на выборки, на обучение логистической регрессии и вычисление F1-меры, на генерацию	1	Повторение учебного материала по теме (тест). Выполнение задания для самостоятельной работы. Изучение дополнительного учебного материала о проклятии размерности (знакомство с термином и понятием).

					регрессионных данных и разделение на выборки, на обучение линейной регрессии и вычисление MSE.		
2	Тема 2.5. Работа над учебным проектом	0	-	2	Выполнение учебного проекта согласно варианту и плану. Подготовка отчета о выполнении учебного проекта.	2	Изучение дополнительного учебного материала про автоматический подбор гиперпараметров. Завершение работы над учебным проектом в соответствии с вариантом и планом. Подготовка отчета о его выполнении.
3	Тема 3.1 Архитектура нейронных сетей: нейроны, слои, веса и смещения	2	Структура нейронной сети: что такое нейроны, слои, как работают веса и смещения, их роль в формировании выходных данных сети.	4	Выполнение задания с разбором на имитацию работы одного нейрона с двумя входами, на имитацию прямого прохода в однослойной нейронной сети с тремя нейронами. Выполнение заданий на тренажере на вычисление выхода одного нейрона, на прямой проход однослойной нейронной сети, на реализацию функции активации ReLU, на вычисление взвешенной суммы с использованием NumPy, на имитацию функции активации сигмоиды.	1	Изучение дополнительного учебного материала о разнице между классическим ML и нейросетями. Выполнение задания для самостоятельной работы.

3	Тема 3.2. Принцип работы перцептрона как базовой модели нейронной сети. Функции активации и их роль.	2	Описание модели перцептрона: входные нейроны, веса, суммирование, функция активации (например, сигмоида, ReLU, ступенчатая). Роль функции активации в нелинейности и способности сети решать сложные задачи.	4	Выполнение задания с разбором на расчет вероятности, моделирование операции XOR с помощью простой нейронной сети и ReLU. Выполнение заданий на тренажере на расчет вероятности, на реализацию функции ReLU и прямого прохода нейронной сети XOR, на классификацию выходов нейронной сети XOR, на классификацию выходов нейронной сети XOR с порогом.	1	Изучение дополнительного учебного материала о функции потерь и эпохи в обучении нейросетей. Выполнение задания для самостоятельной работы.
3	Тема 3.3. Основы обучения нейронных сетей: метод обратного распространения ошибки и градиентный спуск	2	Изучение алгоритма обратного распространения ошибки, принципа градиентного спуска для настройки весов. Понимание процесса обучения нейросети на примерах.	4	Выполнение задания с разбором на реализацию градиентного спуска с моментумом для функции. Выполнение заданий на тренажере на реализацию шага градиентного спуска с моментумом, на реализацию цикла градиентного спуска с моментумом, на поиск минимума функции после градиентного спуска, на реализацию градиентного спуска с моментумом и	1	Изучение дополнительного учебного материала о работе градиентного спуска. Выполнение задания для самостоятельной работы. Проведение самопроверки знаний в игровой форме – выполнение тестовых заданий с игровым правилом по ограничению во времени и с получением мгновенного результата.

					определение изменения функции.		
3	Тема 3.4. Использование библиотек TensorFlow, PyTorch, Keras для работы с нейронными сетями и создание простых моделей	2	Ознакомление с популярными библиотеками (например, PyTorch, TensorFlow). Создание и обучение простых моделей нейронных сетей с их помощью.	6	Выполнение задания с разбором на создание и анализ тензора в TensorFlow, на создание переменной и вычисление градиентов в PyTorch. Выполнение заданий на тренажере на создание константного тензора в TensorFlow, нулевого тензора в PyTorch, случайного тензора в PyTorch и на переменной TensorFlow, на проведение арифметических операций с тензорами в PyTorch, на срезы (slicing) тензоров в PyTorch, на определение и вычисление функции с обратным распространением в PyTorch, на изменение формы (reshape) тензора в PyTorch.	1	Изучение дополнительного учебного материала о библиотеке TensorFlow. Выполнение задания для самостоятельной работы.
3	Тема 3.5. Работа над учебным проектом	0	-	2	Выполнение учебного проекта согласно варианту и плану. Подготовка отчета о выполнении учебного проекта.	2	Изучение дополнительного учебного материала про интересные фишки для нейросетей (кастомные функции активации в PyTorch, визуализация весов

							в Keras, заморозка слоев и пр.). Завершение работы над учебным проектом в соответствии с вариантом и планом. Подготовка отчета о его выполнении.
4	Тема 4.1. Применение методов машинного обучения для решения задач классификации и регрессии	1	Изучение реальных примеров и методов применения классических алгоритмов ML для задач классификации и регрессии в практических сценариях.	6	Выполнение задания с разбором на обучение простой линейной модели с TensorFlow, на реализацию алгоритма К-средних (K-means) на TensorFlow, на оценку качества кластеризации с помощью суммы квадратов ошибок (SSE). Выполнение заданий на тренажере на создание и обучение простой линейной модели, на инициализацию центроидов для К-средних, на обновления центроидов в К-средних, на удаление элементов из массива, на построение графика функции с локальным минимумом (без визуализации).	1	Изучение дополнительного учебного материала про ансамблирование моделей. Выполнение задания для самостоятельной работы.
4	Тема 4.2. Использование нейронных сетей для анализа	1	Обзор компьютерного зрения и роли нейронных сетей в	6	Выполнение задания с разбором на обнаружение объектов на изображении с помощью YOLOv8.	1	Изучение дополнительного учебного материала о VR и его связь с компьютерным зрением.

	изображений: основы компьютерного зрения. Применение моделей в проектах дополненной реальности AR совместно с OpenCV.		анализе изображений. Знакомство с задачами классификации, детектирования и сегментации изображений. Основы сверточных нейронных сетей (CNN)		Выполнение заданий на тренажере на создание одномерного массива с помощью NumPy и двумерного массива из списка списков, на создание массива случайных чисел с нормальным распределением, массива нулей и единиц, массива и суммирование элементов в цикле, на умножение элементов массива на константу в цикле, на поэлементное сравнение массива и подсчёт количества элементов, на создание двумерного массива и вывод элементов по строкам, на объектное детектирование на искусственном изображении с YOLOv8 и на обработку серии искусственных кадров с YOLOv8.		Выполнение задания для самостоятельной работы.
4	Тема 4.3.Базовые методы обработки и анализа текстовых данных с помощью ИИ	2	Изучение методов обработки текстовых данных: токенизация, векторизация, основы работы с текстом в ИИ. Применение моделей	4	Выполнение задания с разбором на векторизацию текста с помощью CountVectorizer. Выполнение заданий на векторизацию текста с	1	Изучение дополнительного учебного материала про векторизацию текста с помощью GloVe и fastText. Выполнение задания для самостоятельной работы. Проведение самопроверки

			для анализа и классификации текстов.		помощью CountVectorizer, на получение словаря из CountVectorizer на другом корпусе, на векторизацию текста с помощью TfidfVectorizer, на обучение Word2Vec на списке предложений, на вычисление сходства слов с помощью Word2Vec.		знаний в игровой форме – выполнение тестовых заданий с игровым правилом по ограничению во времени и с получением мгновенного результата.
4	Тема 4.4. Новые профессии в ИИ. Практическое использование библиотек и фреймворков для решения прикладных задач.	2	Ознакомление с популярными библиотеками и фреймворками (scikit-learn, TensorFlow, PyTorch и др.) для решения задач ML и ИИ. Практические примеры применения.	4	Выполнение задания с разбором на подсчёт частоты слов с использованием Counter, Выполнение заданий на подсчёт частоты слов в другом тексте, на подсчёт частоты слов с другими стоп-словами, на токенизацию текста на слова с помощью NLTK и на предложения с помощью NLTK, на токенизацию с помощью токенизатора Hugging Face Transformers.	1	Изучение дополнительного учебного материала о практическом использовании библиотек и фреймворков для решения прикладных задач в области искусственного интеллекта. Выполнение задания для самостоятельной работы.
4	Тема 4.5. Работа над учебным проектом	0	-	2	Выполнение учебного проекта согласно варианту и плану. Подготовка отчета о выполнении учебного проекта.	2	Изучение дополнительного учебного материала про ускорение обучения с mixed precision, оптимизацию гиперпараметров с Optuna, деплой моделей с ONNX,

							автоматическую маркировку данных с Snorkel, кросс-платформенные пайплайны с Kedro. Завершение работы над учебным проектом в соответствии с вариантом и планом. Подготовка отчета о его выполнении.
--	--	--	--	--	--	--	---

5. Формы аттестации и оценочные материалы, включая примеры контрольных заданий

5.1 Текущий контроль

В таблице ниже представлены по каждому из модулей:

- количество часов текущего контроля;
- формы текущего контроля с подробным описанием процедуры оценивания результатов обучения;
- диагностические инструменты с примерами (контрольные задания, материалы, промежуточные тесты и задачи);
- показатели и критерии оценивания;
- шкала оценивания.

Текущий контроль. Количество ак. часов	1	1	1	1
Текущий контроль. Формы контроля	Выполнение тестовых заданий для текущего контроля знаний и навыков по пройденному материалу.	Выполнение тестовых заданий для текущего контроля знаний и навыков по пройденному материалу.	Выполнение тестовых заданий для текущего контроля знаний и навыков по пройденному материалу.	Выполнение тестовых заданий для текущего контроля знаний и навыков по пройденному материалу.
Текущий контроль. Диагностические инструменты	Тестовые вопросы. Примеры вопросов. 1. Какой тип данных будет у переменной $x = 5.5$?	Тестовые вопросы. Примеры вопросов. 1. Какой метод Pandas используется для чтения CSV-файла?	Тестовые вопросы. Примеры вопросов. 1. Какой тип активации используется в выходном слое логистической регрессии?	Тестовые вопросы. Примеры вопросов. 1. Какой параметр градиентного спуска

a) int б) float в) str г) bool Ответ: б) float 2. Какое имя переменной соответствует PEP 8? а) MyVar б) my_var в) myVar г) lvar Ответ: б) my_var 3. Что выведет код print("""10"" * 2)? а) 20 б) ""20"" в) ""1010"" г) Ошибку Ответ: в) ""1010"" 4. Открытый вопрос. Что выведет этот код? print(len("""Python""")) Ответ: 6 5. Открытый вопрос. Что выведет этот код? print("""Hello"".upper())	а) read_csv() б) load_csv() в) open_csv() г) get_csv() Ответ: а) read_csv() 2. Как проверить наличие пропущенных значений в DataFrame df? а) df.isna() б) df.missing() в) df.has_nulls() г) df.check_null() Ответ: а) df.isna() 3. Как удалить дубликаты в DataFrame? а) df.drop_duplicates() б) df.remove_duplicates() в) df.unique() г) df.deduplicate() Ответ: а) df.drop_duplicates() 4. Открытый вопрос. Какой метод Pandas выведет основные статистики (count, mean, std, min, max) для числовых столбцов? Ответ: df.describe()	а) ReLU б) Сигмоида в) Tanh г) Softmax Ответ: б) Сигмоида 2. Какой параметр дерева решений ограничивает его глубину? а) min_samples_split б) max_depth в) criterion г) n_estimators Ответ: б) max_depth 3. Какой ядерный метод НЕ используется в SVM? а) Линейный б) Полиномиальный в) Экспоненциальный г) RBF Ответ: в) Экспоненциальный 4. Открытый вопрос. Что выведет следующий код? from sklearn.linear_model import LogisticRegression model = LogisticRegression() print(model.penalty) Ответ: 'l2'	контролирует размер шага обновления весов? а) batch_size б) learning_rate в) epochs г) momentum Ответ: б) learning_rate 2. В PyTorch для создания тензора из списка используется: а) torch.tensor() б) torch.Tensor() в) torch.from_list() г) torch.array() Ответ: а) torch.tensor() 3. Какой слой в Keras используется для свертки 2D изображений? а) Conv1D б) Conv2D в) MaxPooling2D г) Dense Ответ: б) Conv2D 4. Открытый вопрос. Что выведет код: import torch x = torch.tensor([1.0, 2.0], requires_grad=True)
---	---	--	--

	Ответ: ""HELLO"" 6. Открытый вопрос. Что выведет этот код? <pre>def add(a, b=2): return a + b print(add(3))</pre> Ответ: 5	5. Открытый вопрос. Какой метод Scikit-learn разбивает данные на обучающую и тестовую выборки? Ответ: train_test_split() 6. Открытый вопрос. Какой метод Pandas заполняет пропущенные значения указанным числом (например, 0)? Ответ: df.fillna(0)	5. Открытый вопрос. Что вернет следующий код? <pre>import numpy as np arr = np.array([1, 2, 3]) print(arr.mean())</pre> Ответ: 2.0 6. Открытый вопрос. Что выведет следующий код? <pre>from sklearn.tree import DecisionTreeClassifier model = DecisionTreeClassifier(max_depth=1) print(model.criterion)</pre> Ответ: 'gini'	<pre>y = x.sum() y.backward() print(x.grad)</pre> Ответ: tensor([1., 1.]) 5. Открытый вопрос. Что выведет код: <pre>import tensorflow as tf x = tf.constant([[1, 2], [3, 4]]) print(tf.reduce_mean(x))</pre> Ответ: tf.Tensor(2, shape=(), dtype=int32) 6. Открытый вопрос. Что выведет код: <pre>from sklearn.cluster import KMeans import numpy as np X = np.array([[1, 2], [1, 4], [1, 0]]) kmeans = KMeans(n_clusters=2, random_state=0, n_init='auto').fit(X) print(kmeans.labels_)</pre> Ответ: [0 0 1]
Текущий контроль. Показатели и критерии оценивания	Критерии оценивания: оценивается правильность ответов на тестовые задания.	Критерии оценивания: оценивается правильность ответов на тестовые задания.	Критерии оценивания: оценивается правильность ответов на тестовые задания.	Критерии оценивания: оценивается правильность ответов на тестовые задания.

	Показатель оценивания: соответствие ответа тестируемого правильному ответу. За каждый правильный ответ начисляется 1 балл. Максимальное количество баллов — 15. Для успешного прохождения теста необходимо набрать не менее 60% правильных ответов (то есть 9 баллов из 15). Шкала оценивания: Зачтено — 9– 15 баллов Не зачтено — 0 – 8 баллов	Показатель оценивания: соответствие ответа тестируемого правильному ответу. За каждый правильный ответ начисляется 1 балл. Максимальное количество баллов — 15. Для успешного прохождения теста необходимо набрать не менее 60% правильных ответов (то есть 9 баллов из 15). Шкала оценивания: Зачтено — 9– 15 баллов Не зачтено — 0 – 8 баллов	Показатель оценивания: соответствие ответа тестируемого правильному ответу. За каждый правильный ответ начисляется 1 балл. Максимальное количество баллов — 15. Для успешного прохождения теста необходимо набрать не менее 60% правильных ответов (то есть 9 баллов из 15). Шкала оценивания: Зачтено — 9– 15 баллов Не зачтено — 0 – 8 баллов	Показатель оценивания: соответствие ответа тестируемого правильному ответу. За каждый правильный ответ начисляется 1 балл. Максимальное количество баллов — 15. Для успешного прохождения теста необходимо набрать не менее 60% правильных ответов (то есть 9 баллов из 15). Шкала оценивания: Зачтено — 9– 15 баллов Не зачтено — 0 – 8 баллов
--	--	--	--	--

5.2 Промежуточная аттестация

В таблице ниже представлены по каждому из модулей:

- количество часов промежуточной аттестации;
- формы контроля (промежуточной аттестации) с подробным описанием процедуры оценивания результатов обучения;
- диагностические инструменты с примерами (контрольные задания, материалы, промежуточные тесты и задачи);
- показатели и критерии оценивания;
- шкала оценивания.

Аттестация по итогам модуля. Количество ак. часов	1	1	1	1
--	---	---	---	---

Аттестация по итогам модуля. Формы контроля	Промежуточное тестирование	Промежуточное тестирование	Промежуточное тестирование	Промежуточное тестирование
Аттестация по итогам модуля. Диагностические инструменты	Набор тестовых заданий для промежуточного тестирования Пример тестовых вопросов: 1) Какой метод используется для загрузки данных из CSV-файла в Pandas? а) read_csv() б) read_excel() в) read_json() г) read_html() Ответ: а) read_csv() 2) Какой тип данных будет у переменной x = [1, 2, 3]? а) int б) list в) tuple г) dict Ответ: б) list 3) Как удалить дубликаты строк в DataFrame? а) drop_duplicates() б) remove_duplicates() в) delete_duplicates() г) clear_duplicates()	Набор тестовых заданий для промежуточного тестирования Пример тестовых вопросов: 1. Какой модуль в sklearn используется для разделения данных на train и test? а) sklearn.split б) sklearn.model_selection в) sklearn.preprocessing г) sklearn.datasets Ответ: б) sklearn.model_selection 2. Какой метод K-NN возвращает вероятности классов? а) predict() б) predict_proba() в) score() г) decision_function() Ответ: б) predict_proba() 3. Какой параметр логистической регрессии отвечает за тип регуляризации? а) C б) penalty в) solver г) max_iter Ответ: б) penalty	Набор тестовых заданий для промежуточного тестирования Пример тестовых вопросов: 1. Как называется элементарная единица нейронной сети? а) Слой б) Нейрон в) Тензор г) Градиент Ответ: б) Нейрон 2. Что передается между слоями нейронной сети? а) Веса б) Смещения в) Активации г) Градиенты Ответ: в) Активации 3. Какой параметр нейрона обучается в процессе тренировки?	Набор тестовых заданий для промежуточного тестирования Пример тестовых вопросов: 1. Какой тип задачи решается, когда у нас есть размеченные данные? а) С учителем б) Без учителя в) С подкреплением г) Генеративная Ответ: а) С учителем 2. Какой метод TensorFlow используется для создания последовательной модели? а) tf.keras.Sequential() б) tf.keras.Model() в) tf.keras.layers() г) tf.keras.network() Ответ: а) tf.keras.Sequential() 3. Какой алгоритм относится к обучению без учителя? а) Линейная регрессия б) К-средних в) Нейронная сеть г) Дерево решений Ответ: б) К-средних 4. Открытый вопрос. Что выведет этот код?

	Ответ: a) drop_duplicates() 4) Открытый вопрос. Что выведет этот код? print("Hello, " + "World!") Ответ: Hello, World! 5) Открытый вопрос. Что выведет этот код? numbers = [1, 2, 3, 4, 5] print(numbers[1:4]) Ответ: [2, 3, 4] 6) Открытый вопрос. Что выведет этот код? def greet(name): return f"Hello, {name}!" print(greet("Alice")) Ответ: Hello, Alice!	4. Открытый вопрос. Что выведет код? from sklearn.svm import SVC model = SVC(kernel='linear') model.fit([[0], [1]], [0, 1]) print(model.predict([[0.5]])) Ответ: 0 5. Открытый вопрос. Что выведет код? from sklearn.preprocessing import MinMaxScaler scaler = MinMaxScaler() print(scaler.fit_transform([[10], [20]])) Ответ: [[0.][1.]] 6. Открытый вопрос. Что выведет код? from sklearn.tree import DecisionTreeRegressor model = DecisionTreeRegressor(max_dept h=1) model.fit([[1], [2]], [10, 20]) print(model.predict([[1.5]])) Ответ: 15.0	a) Функция активации б) Веса в) Количество входов г) Порог срабатывания Ответ: б) Веса 4. Открытый вопрос. Что выведет этот код? import numpy as np weights = np.array([0.5, -0.3]) inputs = np.array([2.0, 1.0]) print(np.dot(weights, inputs)) Ответ: 0.7 5. Открытый вопрос. Что выведет этот код? import tensorflow as tf tensor = tf.constant([[1, 2], [3, 4]]) print(tensor.shape[1]) Ответ: 2 6. Открытый вопрос. Что выведет этот код? import torch x = torch.tensor([1.0, 2.0], requires_grad=True) y = x.mean() print(y.item())	import tensorflow as tf model = tf.keras.Sequential() model.add(tf.keras.layers.Dense(10, input_shape=(5,))) print(model.count_params()) Ответ: 60 5. Открытый вопрос. Что выведет этот код? from sklearn.feature_extraction.text import CountVectorizer corpus = ['hello world', 'world of python'] vectorizer = CountVectorizer() X = vectorizer.fit_transform(corpus) print(len(vectorizer.get_feature_names_o ut())) Ответ: 3 6. Открытый вопрос. Что выведет этот код? import nltk from nltk.stem import PorterStemmer stemmer = PorterStemmer() print(stemmer.stem('running')) Ответ: run
--	--	---	---	--

			Ответ: 1.5	
Аттестация по итогам модуля. Описание процедуры оценивания Показатели и критерии оценивания	Критерий оценивания: правильность ответа на тестовое задание. Показатель оценивания: количество правильных ответов на тестовые задания. В случае правильного ответа на тестовое задание начисляется 1 балл. В случае неправильного ответа баллы не начисляются. Полученные баллы суммируются, определяется результат выполнения промежуточного тестирования на основе соотнесения с установленным пороговым значением.	Критерий оценивания: правильность ответа на тестовое задание. Показатель оценивания: количество правильных ответов на тестовые задания. В случае правильного ответа на тестовое задание начисляется 1 балл. В случае неправильного ответа баллы не начисляются. Полученные баллы суммируются, определяется результат выполнения промежуточного тестирования на основе соотнесения с установленным пороговым значением.	Критерий оценивания: правильность ответа на тестовое задание. Показатель оценивания: количество правильных ответов на тестовые задания. В случае правильного ответа на тестовое задание начисляется 1 балл. В случае неправильного ответа баллы не начисляются. Полученные баллы суммируются, определяется результат выполнения промежуточного тестирования на основе соотнесения с установленным пороговым значением.	Критерий оценивания: правильность ответа на тестовое задание. Показатель оценивания: количество правильных ответов на тестовые задания. В случае правильного ответа на тестовое задание начисляется 1 балл. В случае неправильного ответа баллы не начисляются. Полученные баллы суммируются, определяется результат выполнения промежуточного тестирования на основе соотнесения с установленным пороговым значением.
Аттестация по итогам модуля.	0	0	0	0

Шкала оценивания, нижнее значение				
Аттестация по итогам модуля. Шкала оценивания, верхнее значение	15	15	15	15
Аттестация по итогам модуля. Шкала оценивания, минимальный проходной балл для успешной сдачи	9	9	9	9

5.3 Итоговый контроль

Итоговый контроль/Итоговая аттестация. Формы контроля	Проведение тестирования
Итоговый контроль/Итоговая аттестация. Диагностические инструменты	Набор тестовых заданий Примеры тестов: 1. Какой тип данных будет у переменной <code>x = "123"</code>? а) int б) float в) str г) bool Ответ: в) str 2. Какой метод Pandas используется для загрузки данных из CSV? а) <code>pd.read_csv()</code> б) <code>pd.load_csv()</code> в) <code>pd.open_csv()</code> г) <code>pd.from_csv()</code> Ответ: а) <code>pd.read_csv()</code> 3. Какая библиотека используется для визуализации в Python? а) numpy б) matplotlib в) pandas г) scikit-learn Ответ: б) matplotlib 4. Какой алгоритм машинного обучения используется для классификации? а) Линейная регрессия б) k-NN в) K-Means г) PCA Ответ: б) k-NN 5. Открытый вопрос. Что выведет этот код? <pre>print([1, 2, 3] + [4, 5])</pre> Ответ: [1, 2, 3, 4, 5] 6. Какой метод используется для удаления пропусков в Pandas?

	a) dropna() б) fillna() в) remove_na() г) delete_na() Ответ: а) dropna() 7. Какой тип графика используется для отображения распределения данных? а) lineplot б) histogram в) piechart г) scatterplot Ответ: б) histogram 8. Какой метод scikit-learn используется для разделения данных на обучающую и тестовую выборки? а) train_test_split() б) split_data() в) divide_dataset() г) partition() Ответ: а) train_test_split() 9. Как называется слой в нейронной сети, который преобразует входные данные? а) Выходной слой б) Скрытый слой в) Полносвязный слой г) Свёрточный слой Ответ: б) Скрытый слой 10. Открытый вопрос. Что выведет этот код? <pre>import numpy as np print(np.array([1, 2, 3]) * 2)</pre> Ответ: [2, 4, 6] 11. Какой метод OpenCV используется для детекции объектов с помощью YOLOv8? а) cv2.YOLO.detect() б) cv2.dnn.readNet() в) cv2.YOLOv8.predict() г) cv2.detectObjects()
--	--

Ответ: б) cv2.dnn.readNet()

12. Какой алгоритм обработки текста преобразует слова в их базовую форму?

- а) Токенизация
- б) Стемминг
- в) Лемматизация
- г) Векторизация

Ответ: в) Лемматизация

13. Какой метод TensorFlow используется для создания полносвязного слоя?

- а) tf.keras.layers.Dense()
- б) tf.keras.layers.Conv2D()
- в) tf.keras.layers.LSTM()
- г) tf.keras.layers.Flatten()

Ответ: а) tf.keras.layers.Dense()

14. Какой метод Pandas используется для группировки данных?

- а) groupby()
- б) aggregate()
- в) summarize()
- г) cluster()

Ответ: а) groupby()

15. Открытый вопрос. Что выведет этот код?

```
def func(x):  
    return x ** 2  
print(func(3))
```

Ответ: 9

16. Какой метод используется для нормализации данных в scikit-learn?

- а) StandardScaler()
- б) Normalizer()
- в) MinMaxScaler()
- г) RobustScaler()

Ответ: в) MinMaxScaler()

17. Какой алгоритм машинного обучения использует деревья решений?

	a) SVM б) Random Forest в) k-Means г) Logistic Regression Ответ: б) Random Forest 18. Какой метод OpenCV используется для обнаружения лиц? a) cv2.detectFaces() б) cv2.CascadeClassifier() в) cv2.findContours() г) cv2.HaarDetector() Ответ: б) cv2.CascadeClassifier() 19. Какой метод NLTK используется для токенизации текста? a) nltk.tokenize() б) nltk.split() в) nltk.word_tokenize() г) nltk.Text() Ответ: в) nltk.word_tokenize() 20. Открытый вопрос. Что выведет этот код? <pre>import pandas as pd df = pd.DataFrame({"A": [1, 2], "B": [3, 4]}) print(df.shape)</pre> Ответ: (2, 2)
Итоговый контроль/Итоговая аттестация. Показатели и критерии оценивания	Критерий оценивания: правильность ответа на тестовое задание. Показатель оценивания: количество правильных ответов на тестовые задания. Алгоритм расчета итогового балла по ДОП: за каждый правильный ответ на задание теста начисляется 1 балл. В случае неправильного ответа баллы не начисляются. Полученные баллы суммируются, осуществляется автоматическое определение результата выполнения итогового теста на основании порогового балла.
Итоговый контроль/Итоговая аттестация. Шкала оценивания, нижнее значение	0

Итоговый контроль/Итоговая аттестация. Шкала оценивания, верхнее значение	20
Итоговый контроль/Итоговая аттестация. Шкала оценивания, минимальный проходной балл	12

6. Организационно-педагогические условия

6.1 Методическое обеспечение: методы, формы и технологии, применяемые при реализации программы

В образовательной программе предусмотрено использование методов модульного и проблемного обучения, лекций с мультимедийным ресурсом, а также реализация технологий традиционного и интерактивного обучения.

6.2 Материально-техническое обеспечение

Наименование требуемого оборудования и программного обеспечения:

Очная форма с применением электронного обучения, без применения дистанционных образовательных технологий:

Рабочее место обучающегося: оборудование и мебель, необходимые для обучения, включая расходные материалы.

Рабочее место педагога: оборудование (включая демонстрационное), мебель и иное необходимое оснащение.

Сетевое оборудование и подключение к сети Интернет.

Программное обеспечение.

Учебные пособия.

Средства индивидуальной защиты и охраны труда.

Зонирование площадки.

Подробная информация об оснащении представлена в технологической карте оснащённости площадки, являющейся приложением к программе.

7. Учебно-методические материалы

Ниже представлен перечень методических разработок и материалов для обучения:

Методические разработки:

Планы практических занятий.

Алгоритм выполнения самостоятельной работы.

Методические указания по выполнению практических заданий.

Методические материалы для оценивания результатов обучения.

Материалы для обучения:

Пакет материалов модуля содержит:

Опорные конспекты теоретической подготовки.

Презентации по темам.

Практические задания.

Задания для самоконтроля (самопроверки).

Задания для самостоятельной работы.

Задания системы контроля: текущего контроля, промежуточной аттестации, итоговой аттестации.

8. Перечень источников информационного сопровождения (учебная литература и др.)

Учебная литература:

1. Гэддис Т. Начинаем программировать на Python. – 5-е изд.: Пер.с англ. – СПб.: БХВ-Петербург, 2022. – 880 с.
2. Лейн Д. Машинное обучение для детей. Практическое введение в искусственный интеллект / Д. Лейн; пер. с англ.– М.: Лаборатория знаний, 2023. – 288 с.– (Школа юного программиста).
3. Лутц М., Изучаем Python. Том 1. 5-е изд. Учебник. Издательство: Диалектика, 2020. – 832 с.
4. Мэтиз Эрик Изучаем Python: программирование игр, визуализация данных, веб-приложения. 3-е изд. – СПб.: Питер, 2024. – 512 с.
5. Основы искусственного интеллекта : учебное пособие / Е.В. Боровская, Н.А. Давыдова. – 5-е изд. – М.: Лаборатория знаний, 2022. – 127 с.
6. Постолит А.В. Основы искусственного интеллекта в примерах на Python. Самоучитель. – 2-е изд., перераб. и доп. – СПб.: БХВ-Петербург, 2024. – 446 с.
7. Салахова А.А., Самылкина Н.Н. Искусственный интеллект. 10–11 классы. Учебное пособие. ФГОС: учебное пособие. – М.: Просвещение, 2023. – 144 с.
8. Флах Петер. Машинное обучение. Наука и искусство построения алгоритмов, которые извлекают знания из данных / пер. с англ. А.А. Слинкина. – М.: ДМК Пресс, 2015. – 402 с.
9. Чжен Э., Казари А. Машинное обучение. Конструирование признаков. Издательство: Бомбора, 2019. – 240 с.