

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ПОВОЛЖСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ



УТВЕРЖДАЮ
Декан РТФ

УТВЕРЖДАЮ /А.Н. Дедов/
(Ф.И.О. декана (директора института))

13.02.2025 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Б.1.2.2 Основы программирования систем ИИ на языке Python

(код и наименование дисциплины по учебному плану)

Направление подготовки
(специальность)

09.03.02 Информационные системы и технологии

Квалификация выпускника

Бакалавр

(бакалавр/магистр/специалист)

Направленность

Интеллектуальные информационные системы и
технологии

Курс 2
Семестр 3

Распределение учебного времени

Трудоемкость по учебному плану	288 / 8	часов/зачетных единиц
Лекции	36	часов
Лабораторные работы	54	часов
Практические занятия	-	часов
Иная контактная работа	-	часов
Всего контактной работы (без учета экз.)	90	часов
Контактная работа по экзамену	6	часов
Курсовой проект (работа)	-	семестр
Самостоятельная работа обучающихся (без учета экз.)	162	часов
Самостоятельная работа по подготовке к экзамену	30	часов
Экзамен	3	семестр
Зачет	-	семестр
БРК, ДЗ	-	семестр

(год)

Программа составлена в соответствии с требованиями ФГОС ВО направления подготовки (специальности) 09.03.02 Информационные системы и технологии

Программу составили:

старший преподаватель	ИТРС	СОГЛАСОВАНО	Н.А. Конкин
(должность)	(кафедра)		(И.О. Фамилия)

РАССМОТРЕНА и ОДОБРЕНА на заседании кафедры, за которой закреплена дисциплина
Кафедра информационных технологий, радиотехники и связи

(наименование кафедры)		
13.01.2025	протокол №	2
(дата)		

Заведующий кафедрой	СОГЛАСОВАНО	Н.В. Рябова
		(И.О. Фамилия)

Рабочая программа СОГЛАСОВАНА с факультетом (институтом), выпускающей(ими)
кафедрой(ами).

СООТВЕТСТВУЕТ действующей ОП.

Заведующий кафедрой	СОГЛАСОВАНО	Н.В. Рябова
		(И.О. Фамилия)

Председатель методической комиссии факультета (института), в который входит
выпускающая кафедра

СОГЛАСОВАНО	А.Н. Дедов
	(И.О. Фамилия)

Эксперт(ы): Дудин Александр Николаевич , Исполнительный директор ООО «Трэвел Лайн
Системс»

Рабочая программа проверена и зарегистрирована в УМЦ 17.02.2025 г.

Специалист учебно-методического центра СОГЛАСОВАНО /Т.А. Смирнова/

Раздел 1. ЦЕЛЬ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Целью освоения дисциплины является достижение планируемых результатов обучения, соответствующих установленным в ОПОП индикаторам достижения компетенций:

Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Результаты обучения
1. PL-1 Способен применять язык программирования Python для решения задач в области искусственного интеллекта	PL-1.1. Разрабатывает и отлаживает прикладные решения разной сложности и для разного круга конечных пользователей с использованием языка программирования Python, тестирует, испытывает и оценивает качество таких решений	знания: Знает: принципы оптимизации кода, архитектуру CPU и GPU, а также математические основы алгоритмов в библиотеках scikit-learn, PyTorch и TensorFlow для осознанного выбора моделей. умения: Умеет: применять методы векторизации, проводить профилирование для устранения "узких мест", модифицировать готовые модели и создавать собственные компоненты для решения нестандартных задач. навыки: Владеет сквозной оптимизацией всего конвейера данных, от предобработки до инференса, а также искусством распределенных вычислений для обучения на нескольких GPU, создавая высокопроизводительные решения для промышленного внедрения.
	PL-1.2. Осуществляет выбор инструментов разработки на Python, приемлимых для создания прикладной системы обработки научных данных, машинного обучения и визуализации с заданными требованиями	знания: Знает: принципы работы и основные конструкции библиотек для ввода-вывода, а также архитектурные особенности фреймворков FastAPI, Flask и Django REST Framework для построения эффективных серверных приложений. Понимает теорию многопоточности и ограничения модуля `threading` в Python, включая проблему GIL. умения: Умеет: применять эти знания для самостоятельной разработки и поддержки серверных приложений, создавая надежные API-эндпоинты и обрабатывая конкурентные запросы. Умеет структурировать код в рамках крупных проектов, интегрировать различные компоненты и использовать многопоточность для выполнения параллельных задач, где это целесообразно. навыки: Владеет: навыком использования этих технологий в комплексе для решения большинства рутинных задач в бэкенд-разработке. Владеет полным циклом создания серверного приложения — от проектирования маршрутов и реализации бизнес-логики до его развертывания и последующей поддержки в

	PL-1.3. Разрабатывает и поддерживает системы обработки больших данных различной степени сложности	знания: Знает: основные концепции распределенных вычислений и архитектуру фреймворка PySpark, включая работу с RDD и DataFrame API. Понимает принципы оркестрации данных и структуру Directed Acyclic Graph (DAG), лежащую в основе работы Apache Airflow. умения: Умеет: применять ключевые функции PySpark для обработки и анализа больших данных, включая фильтрацию, агрегацию и соединение датасетов. Умеет самостоятельно создавать и настраивать DAG-ы в Airflow для построения отказоустойчивых ETL-процессов, определяя зависимости между задачами и планируя их выполнение. навыки: Владеет навыком комплексного построения сквозного процесса обработки данных — от написания распределенных заданий в PySpark до их организации в воспроизводимые и мониторируемые рабочие процессы с помощью Airflow. Владеет компетенцией полного цикла работы с большими данными: от сырых данных на распределенном хранилище до получения готовых результатов.
--	--	--

Раздел 2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП

Дисциплина относится к части, формируемой участниками образовательных отношений ОПОП.

Дисциплина является обязательной

Изучаемая дисциплина является основой для продолжения формирования указанных компетенций в следующих практиках: Производственная практика. Технологическая (проектно-технологическая) практика (рассредоточенная) (PL-1), Производственная практика. Технологическая (проектно-технологическая) практика (PL-1), Производственная практика. Научно-исследовательская работа (PL-1); государственной итоговой аттестации в форме: Выполнение и защита выпускной квалификационной работы (PL-1)

Раздел 3. ОПИСАНИЕ ОБРАЗОВАТЕЛЬНЫХ ТЕХНОЛОГИЙ

Для формирования заявленных компетенций используются методологические технологии, реализующие деятельностный, личностно-ориентированный, практико-ориентированный подходы.

Основными стратегическими технологиями являются: лекционные занятия, практические и лабораторные занятия

На достижение конкретных целей обучения направлены применяемые тактические технологии: задания, информационные, классическая лекция, мини-проекты

Раздел 4. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Виды и темы занятий	Количество часов	Формируемые компетенции
№1 Инструментальные средства разработки на Python	63	PL-1
Лекция. Лекция 1 (2 часа): Экосистема Python для разработки систем ИИ • Настройка окружения разработки и управление зависимостями • Обзор библиотек для научных вычислений и обработки данных	2	
Лекция. Лекция 2 (2 часа): Научные вычисления и обработка данных • Векторизация операций и оптимизация вычислений • Работа с табличными данными и временными рядами • Методы загрузки, очистки и трансформации данных	2	
Лекция. Лекция 3 (2 часа): Визуализация и многопоточное программирование • Построение графиков и визуализация для анализа данных • Многопоточность и параллелизм в Python • Оптимизация обработки данных через параллельные	2	
Лабораторная работа. • Настройка окружения и работа с основными библиотеками МО и анализа данных • Операции с массивами и табличными данными • Exploratory Data Analysis на реальных датасетах • Создание визуализаций для ML-проектов • Реализация многопоточной обработки данных	12	
Задания для самостоятельной работы, в том числе выполнение • Изучение документации основных библиотек • Комплексный анализ выбранного датасета • Разработка эффективных pipeline для обработки данных • Benchmarking различных подходов к оптимизации	45	
№2 Машинное обучение и классические алгоритмы	69	PL-1
Лекция. Лекция 4 (4 часа): Основы машинного обучения • Типы задач и pipeline машинного обучения • Методы оценки качества моделей и кросс-валидация • Препроцессинг и feature engineering	4	
Лекция. Лекция 5 (4 часа): Классические алгоритмы машинного обучения • Линейные модели и метрические методы • Древовидные алгоритмы и ансамбли • Методы оптимизации гиперпараметров	4	
Лекция. Лекция 6 (4 часа): Практические аспекты разработки ML-систем • Работа с несбалансированными данными • Построение полных ML Pipeline • Интерпретация моделей и анализ ошибок	4	
Лабораторная работа. • Разделение данных и baseline модели • Построение полного ML Pipeline с препроцессингом	12	

Задания для самостоятельной работы, в том числе выполнение - Участие в соревновании по машинному обучению - Разработка оптимальной модели для выбранной задачи - Эксперименты с feature engineering - Создание reproducible ML pipeline	45	
№3 Глубокое обучение и нейронные сети	60	PL-1
Лекция. Лекция 7 (2 часа): Основы глубокого обучения - Архитектура нейронных сетей и процесс обучения - Фреймворки PyTorch и TensorFlow: сравнение и применение	2	
Лекция. Лекция 8 (2 часа): Архитектуры нейронных сетей - Сверточные нейронные сети - Рекуррентные сети для обработки последовательностей - Transfer learning и использование предобученных	2	
Лекция. Лекция 9 (2 часа): Техники обучения нейронных сетей - Методы регуляризации и оптимизации - Стратегии обучения и мониторинг процесса - Оптимизация производительности и deployment моделей	2	
Лабораторная работа. • Создание простых нейронных сетей и обучающих циклов - Реализация CNN для классификации изображений - Обработка текстов и временных рядов с RNN/LSTM - Работа с TensorFlow/Keras	18	
Задания для самостоятельной работы, в том числе выполнение - Решение задачи компьютерного зрения или NLP - Эксперименты с различными архитектурами - Портирование проекта между фреймворками - Оптимизация и подготовка модели к разворачиванию	36	
№4 Серверные приложения и обработка больших данных	60	PL-1
Лекция. Лекция 10 (4 часа): Разработка серверных приложений для ML - REST API для ML-моделей: FastAPI, Flask, Django REST Framework - Контейнеризация и разворачивание ML-сервисов - MLOps: версионирование, мониторинг и CI/CD	4	
Лекция. Лекция 11 (2 часа): Обработка больших данных - Основы работы с PySpark: RDD, DataFrame, Spark SQL - Распределенная обработка данных и машинное обучение с MLlib - Оптимизация Spark-приложений	4	
Лекция. Лекция 12 (2 часа): Оркестрация ML-пайплайнов - Apache Airflow: архитектура и концепция DAG - Построение ETL и ML-пайплайнов - Автоматизация процессов обучения и мониторинга	4	
Лабораторная работа. • Создание REST API для ML-модели с FastAPI - Контейнеризация и документирование сервиса - Настройка experiment tracking и версионирования - Работа с большими датасетами в PySpark - Распределенное обучение моделей с MLlib - Создание Airflow DAG для автоматизации ML-	12	

Задания для самостоятельной работы, в том числе выполнение реферата	
Задания для самостоятельной работы, в том числе выполнение учебных задач от индустриального партнера:	
• Анализ влияния температуры на вероятность дефектов продукции. • Выявление «узких мест» в технологическом процессе по данным сенсоров. • Построение цифровых метрик здоровья оборудования (equipment health indicators). • Оценка качества продукции на основе объединённого анализа нескольких источников данных. • Агрегация данных разных цехов в единый хранилище и их консолидация средствами Python. • Анализ временных рядов производственных сенсоров (температура, вибрации, давление). • Автоматическое выявление аномалий в параметрах оборудования (статистические и простые ML-подходы). • Сравнение производственных линий по стабильности параметров. • Построение системы мониторинга на основе Python + Pandas + Matplotlib. • Обработка больших CSV-файлов с использованием chunk-загрузки. • Предварительная очистка данных (вбросы, пропуски, коррекция некорректных значений). • Построение корреляционной матрицы и анализ взаимосвязей между параметрами оборудования.	36
Иная контактная работа:	0
Подготовка к экзамену	30
Проведение экзамена	6

Раздел 5. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Изучение дисциплины рекомендуется начать с ознакомления с рабочей программой, ее структурой и содержанием разделов. Учебный материал структурирован, изучение дисциплины осуществляется в тематической последовательности.

Подготовка к лабораторным занятиям включает ознакомление с планом занятия, работу с учебной и учебно-методической литературой, научными изданиями и электронными образовательными ресурсами, рекомендованными рабочей программой дисциплины. Содержание самостоятельной работы определяется рабочей программой дисциплины, оценочными и методическими материалами, заданиями и указаниями преподавателя. Самостоятельная работа может осуществляться в аудиторной и внеаудиторной формах. Эффективным средством осуществления самостоятельной работы является электронная информационно-образовательная среда университета, которая обеспечивает доступ к образовательной программе, рабочей программе дисциплины (модуля), к электронным библиотечным системам, профессиональным базам данных и информационным справочным системам.

Контроль и аттестация

Периодичность проведения, формы текущего контроля успеваемости, система оценивания хода освоения дисциплины представлены в рабочей программе.

Формой промежуточной аттестации по дисциплине является экзамен.

1) Необходимое программное обеспечение и инструментальные средства

Среды разработки

В качестве среды разработки для языка программирования Python рекомендуется использовать следующие решения:

1. **Visual Studio Code** (<https://code.visualstudio.com/>) — универсальный редактор с богатыми возможностями для Python-разработки
2. **PyCharm Community Edition** (<https://www.jetbrains.com/pycharm/>) — специализированная IDE для Python
3. **Jupyter Notebook/JupyterLab** (<https://jupyter.org/>) — интерактивная среда для исследовательской работы с данными

Облачные платформы для машинного обучения

Для организации удаленного коллективного решения задач машинного обучения рекомендуется использовать следующие платформы:

1. **Google Colaboratory** (<https://colab.research.google.com/>) — облачная среда с бесплатным доступом к GPU/ TPU, интеграцией с Google Drive и поддержкой Jupyter Notebook
2. **Kaggle** (<https://www.kaggle.com/>) — платформа для соревнований по машинному обучению с большим разнообразием датасетов, примеров решений и возможностью работы в облачных ноутбуках с GPU
3. **Kaggle Learn** (<https://www.kaggle.com/learn>) — бесплатные интерактивные курсы по Python, машинному обучению и глубокому обучению

Дополнительные инструменты

- **Git/GitHub** (<https://github.com/>) — система контроля версий для управления кодом проектов
- **Anaconda Distribution** (<https://www.anaconda.com/>) — дистрибутив Python с предустановленными библиотеками для Data Science
- **Docker** (<https://www.docker.com/>) — для контейнеризации приложений

2) Содержание дисциплины включает изучение исследований и разработок, представленных в научных публикациях на конференциях уровня А*, статьях, опубликованных в журналах 1 уровня Белого списка научных журналов Минобрнауки России (УММ на основе публикаций в научных журналах 1 или 2 уровня Белого списка научных журналов Минобрнауки России, публикаций на конференциях уровня А*/А.

Ниже приведен перечень конференций, рекомендуемых для подачи работ, соответствующих уровню подготовки студентов, освоивших принципы программирования и аналитики данных на языке Python.

Машинное обучение и искусственный интеллект (Machine Learning & AI)

International Conference on Learning Representations (ICLR) Сайт: <https://iclr.cc/> Международная конференция по обучению представлением. Имеет открытый процесс рецензирования и специальные секции (*workshop tracks*), ориентированные на студентов и исследователей, представляющих прикладные решения и обучающие эксперименты с использованием библиотек Python.

International Conference on Machine Learning (ICML) Сайт: <https://icml.cc/> Одна из ведущих конференций по машинному обучению. Включает постерные и демонстрационные треки (*demo, education*), где принимаются проекты по анализу данных, визуализации результатов и разработке обучающих приложений на Python.

AAAI Conference on Artificial Intelligence Сайт: <https://aaai.org/conference/> Конференция Ассоциации по развитию искусственного интеллекта. Содержит специальную программу *Student Abstract and Poster Program*, ориентированную на студенческие и начальные исследовательские работы по искусственному интеллекту, включая разработку алгоритмов и систем на Python.

- **International Joint Conference on Artificial Intelligence (IJCAI)** Сайт: <https://www.ijcai.org/> Международная объединённая конференция по искусственному интеллекту. Имеет секцию *AI Education Track*, где рассматриваются проекты, направленные на развитие методов обучения ИИ и использование Python в образовательных целях.

Анализ данных и интеллектуальные системы (Data Science & Knowledge Discovery)

ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)
Сайт: <https://www.kdd.org/kdd2025/> Конференция по интеллектуальному анализу данных. В рамках *Applied Data Science Track* рассматриваются работы, демонстрирующие применение методов машинного обучения и анализа данных на реальных наборах данных, реализованных с использованием Python.

IEEE International Conference on Data Mining (ICDM)
Сайт: <https://waset.org/data-mining-conference-in-july-2026-in-istanbul> Конференция по интеллектуальному анализу данных, принимающая исследовательские и прикладные работы, включая студенческие проекты по обработке данных и моделированию с использованием Python.

ACM International Conference on Information and Knowledge Management (CIKM)
Сайт: <https://cikm2025.org/> Конференция, ориентированная на вопросы интеграции систем управления знаниями и анализа данных. Включает треки, посвящённые образовательным приложениям и Python-инструментам для обработки информации.

3) Предусмотрено участие ИТ-практиков в различных форматах обучения (лекции, семинары с ИТ-практиками, конкурсы, хакатоны, кейс-чемпионаты, мастер-классы, командные проектные задания и др.) и в качестве наставников.

В программе предусмотрено изучение кейсов (кейсы для изучения, практической и проектной работы) и реальных задач (проектов по разработке, внедрению, развитию цифровых технологий, ИТ-разработок и др.) из практического опыта Индустриальных партнеров в объеме не менее 1 кейса на 1 з.е.

Учебные задач от индустриального партнера:

- Анализ влияния температуры на вероятность дефектов продукции.
- Выявление «узких мест» в технологическом процессе по данным сенсоров.
- Построение цифровых метрик здоровья оборудования (equipment health indicators).
- Оценка качества продукции на основе объединённого анализа нескольких источников данных.
- Агрегация данных разных цехов в единый хранилище и их консолидация средствами Python.
- Анализ временных рядов производственных сенсоров (температура, вибрации, давление).
- Автоматическое выявление аномалий в параметрах оборудования (статистические и простые ML-подходы).
- Сравнение производственных линий по стабильности параметров.
- Построение системы мониторинга на основе Python + Pandas + Matplotlib.
- Обработка больших CSV-файлов с использованием chunk-загрузки.
- Предварительная очистка данных (вбросы, пропуски, коррекция некорректных значений).
- Построение корреляционной матрицы и анализ взаимосвязей между параметрами оборудования.

4) Рекомендованные источники учебно-методической литературы из:

Дополнительные ссылки на источники информации (КРМ)

1. Разрабатывает и отлаживает прикладные решения разной сложности и для разного круга конечных пользователей с использованием языка программирования Python, тестирует, испытывает и оценивает качество таких решений:

- 1) Лутц М. Изучаем Python. — СПб.: Диалектика, 2019. — 832 с. — ISBN 978-5-907144-85-0

• 2) Любанович Б. Простой Python. Современный стиль программирования. — СПб.: Питер, 2021. — 592 с. — ISBN 978-5-4461-1456-6

3) Шоу Э. Внутри CPython: гид по интерпретатору Python. — СПб.: Питер, 2023. — 352 с. — ISBN 978-5-4461-1925-7

2. Осуществляет выбор инструментов разработки на Python, приемлимых для создания прикладной системы обработки научных данных, машинного обучения и визуализации с заданными требованиями:

1) Жерон О. Прикладное машинное обучение с помощью Scikit-Learn и TensorFlow / пер. А. А. Слинкина. — М.: ДМК, 2023. — 688 с. — ISBN 978-5-93700-123-7

2) Рашка С. Машинное обучение с PyTorch и Scikit-Learn. — М.: ДМК, 2023. — 790 с. — ISBN 978-5-93700-145-9

3) ПНСТ 838-2023 Искусственный интеллект. Структура описания систем искусственного интеллекта, использующих машинное обучение

4) ГОСТ Р 71752-2024 Искусственный интеллект. Техническое задание. Требования к содержанию

3. Разрабатывает и поддерживает системы обработки больших данных различной степени сложности

1) Сопов Е. А. Технологии обработки больших данных на Python: учебник. — М.: ИНФРА-М, 2023. — 156 с. — ISBN 978-5-16-019627-5

2) Силен Д., Мейсман А., Али М. Основы Data Science и Big Data. Python и наука о данных. — СПб.: Питер, 2017. — 336 с. — ISBN 978-5-496-02545-3

3) МакГрегор С. Обработка данных на Python. Data Wrangling и Data Quality. — БХВ, 2024. — 320 с. — ISBN 978-5-9775-1846-8

Языки программирования и библиотеки (Python, C++)

Ввод-вывод, типы данных, PEP-8:

1. Palladain. Python_1_HSE_2024: курс «Язык программирования Python» для 1 курса ФКН ВШЭ 2024-2025 год [Электронный ресурс]. — Режим доступа: https://github.com/Palladain/Python_1_HSE_2024?tab=readme-ov-file
2. Основы Python. Хендбук. Яндекс Образование [Электронный ресурс]. — Режим доступа: <https://education.yandex.ru/handbook/python>
3. Задачник по Python. Тема 4. Работа со списками [Электронный ресурс]. Smartika.ru. — Режим доступа: <https://smartika.ru/python-workbook/list#2>

Ввод-вывод с файла, циклы и условия:

4. the_compiler. Python f-string cheat sheet [Электронный ресурс]. — Режим доступа: <https://fstring.help/cheat/>
5. Шультайс Эдьюкейшн. f-строки в Python: применение и производительность [Электронный ресурс]. — Режим доступа: <https://shultais.education/blog/python-f-strings>

Collections + itertools, регулярные выражения:

1. re — Regular expression operations (раздел библиотеки Python) [Электронный ресурс]. — Режим доступа: <https://docs.python.org/3/library/re.html>
2. Regexp101: build, test, and debug regex [Электронный ресурс]. — Режим доступа: <https://regex101.com/>

ООП:

3. Auriga. Алгоритмы автоматической проверки требований к ... [Электронный ресурс]. — Режим доступа: <https://habr.com/ru/companies/auriga/articles/724030/>
4. Python Descriptor Guide. Descriptor HowTo Guide. [Электронный ресурс]. — Режим доступа: <https://docs.python.org/3/howto/descriptor.html>

Parallelism & Concurrency:

5. Wunder Fund. Разбираемся с параллельными и конкурентными вычислениями в Python [Электронный ресурс]

Изучение современных технологий по материалам конференций:

1. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python / Virtanen P. et al. // *Nature Methods*, 2020. DOI:10.1038/s41592-019-0686-2

Данная статья представляет обзор экосистемы Python для научных вычислений, включая ключевые библиотеки — SciPy, NumPy и Matplotlib. Авторы рассматривают внутренние механизмы оптимизации вычислений, векторизацию операций и модульную структуру Python-инструментов для прикладных задач инженерного анализа. Материал связан с лекциями №1–3, охватывающими настройку окружения, работу с массивами данных, а также оптимизацию и визуализацию вычислений.

2. Python for Data Analytics: A Systematic Literature Review of Tools, Techniques and Applications // *International Journal of Advanced Computer Science and Applications*, 2024. [ResearchGate](#)

В статье выполнен систематический обзор инструментов для анализа данных на Python. Особое внимание уделено библиотекам *pandas*, *NumPy*, *matplotlib*, *seaborn* и *scikit-learn*. Описаны типовые методы предобработки, трансформации и визуализации данных, применяемые в современных аналитических системах. Работа будет полезна при изучении лекций №2–3, связанных с обработкой табличных данных, временных рядов и реализацией EDA (Exploratory Data Analysis).

3. Machine Learning: Algorithms, Real-World Applications and Future Directions / Jordan M. et al. // *Frontiers in Artificial Intelligence*, 2021. [Ссылка на PubMed Central](#)

Авторы обобщают современные подходы к машинному обучению, рассматривая линейные модели, деревья решений, ансамблевые методы и метрические алгоритмы. Статья содержит сравнительный анализ классических моделей, примеры оптимизации гиперпараметров и рекомендации по применению различных методов к прикладным задачам. Материал соотносится с содержанием лекций №4–6, включая построение ML Pipeline, оценку качества моделей и анализ ошибок.

4. Review of Deep Learning: Concepts, CNN Architectures, Challenges and Applications / Alzubaidi L. et al. // *Journal of Big Data*, 2021. [Ссылка на SpringerOpen](#)

В статье представлен развёрнутый обзор концепций глубокого обучения, архитектур сверточных и рекуррентных нейронных сетей, а также современных подходов к обучению моделей. Рассмотрены проблемы переобучения, стратегии регуляризации и применения transfer learning. Работа тесно связана с лекциями №7–9, где изучаются основы нейронных сетей, архитектуры CNN и RNN, методы оптимизации и мониторинга обучения.

5. MLOps: A Review / Wazir S., Ahmad M., Shah S. et al. // *arXiv preprint*, 2023. [Ссылка на arXiv](#)

В данной обзорной статье рассматриваются современные методы внедрения и сопровождения систем

1. машинного обучения, включая принципы CI/CD, контейнеризацию, мониторинг и автоматизацию пайплайнов. Особое внимание уделено инструментам *FastAPI*, *Docker*, *Airflow* и *MLflow*. Статья напрямую коррелирует с лекциями №10–12, посвящёнными разработке серверных приложений, обработке больших данных и оркестрации ML-процессов.

5) Эффективным средством осуществления самостоятельной работы является электронная информационно-образовательная среда университета, которая обеспечивает доступ к образовательной программе, рабочей программе дисциплины, к электронным библиотечным системам, профессиональным базам данных и информационным справочным системам. А также рекомендованные ОТКРЫТЫЕ ОБРАЗОВАТЕЛЬНЫЕ РЕСУРСЫ (на платформе LMS Moodle (elearning.volgatech.net), интегрированного с федеральной платформой ГИС СЦОС online.edu.ru открытый образовательный портал (mooped.net) и ресурсы открытых образовательных платформ Яндекс-Практикум, Сберуниверситет.

Рекомендуемые образовательные ресурсы

Базовые курсы по Python

1. **"Поколение Python": курс для начинающих** (<https://stepik.org/course/58852/>)

- Основы синтаксиса Python
- Работа со встроенными типами данных
- Функции и модули

1. **"Поколение Python": курс для продвинутых** (<https://stepik.org/course/68343/>)

- ООП в Python
- Работа с файлами
- Обработка исключений

Курсы по машинному обучению и искусственному интеллекту

1. **"Введение в Data Science и машинное обучение"** (<https://stepik.org/course/4852/>)

- Основы анализа данных
- Классические алгоритмы ML
- Библиотеки NumPy, pandas, scikit-learn

1. **"Нейронные сети и обработка текста"** (<https://stepik.org/course/54098/>)

- Архитектуры нейронных сетей
- Natural Language Processing
- Практика на реальных задачах

1. **"Введение в глубокое обучение"** от HSE (<https://stepik.org/course/135003/>)

- Теория глубокого обучения
- PyTorch и TensorFlow
- Компьютерное зрение и NLP

Специализированные курсы

1. **"MLOps: практики и инструменты"** (<https://stepik.org/course/181476/>)

- Версионирование моделей и данных
- CI/CD для ML-проектов
- Мониторинг моделей в продакшене

Интерактивные платформы для практики

1. **LeetCode** (<https://leetcode.com/>) — алгоритмические задачи на Python
2. **HackerRank** (<https://www.hackerrank.com/domains/python>) — задачи по программированию и ML

Kaggle Competitions (<https://www.kaggle.com/competitions>) — соревнования по анализу данных

Раздел 6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ И УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

6.1. Учебно-методическое обеспечение

№№ п/п	Список используемой литературы	Количество экземпляров печатных изданий, имеющих в библиотеке, или электронный адрес издания (ресурса) в сети Интернет
УЧЕБНЫЕ, УЧЕБНО-МЕТОДИЧЕСКИЕ И НАУЧНЫЕ ИЗДАНИЯ		
1.	Федоров, Дмитрий Юрьевич. Программирование на языке высокого уровня Python [Текст : Электронный ресурс] : учебное пособие для вузов / Д. Ю. Федоров. 5-е изд. Москва: Юрайт, 2025. - 210 с ISBN 978-5-534-14638-7.	https://urait.ru/book/programirovanie-na-python-556864
2.	Трофимов, Валерий Владимирович. Информатика [Текст : Электронный ресурс] : учебник для вузов / В. В. Трофимов [и др.] ; ответственный редактор В. В. Трофимов. 4-е изд. Москва: Юрайт, 2025. - 795 с ISBN 978-5-534-17577-6.	https://urait.ru/book/informatika-568691
3.	Искусственный интеллект в интеллектуальных системах управления биотехнологическими робототехническими системами [Электронный ресурс] : Учебное пособие / Шафрай А. В. Москва: Ай Пи Ар Медиа, 2023. - 217 с. ISBN 978-5-4497-2071-9.	https://www.iprbookshop.ru/128381.html
4.	Комашинский, Владимир Ильич. Нейронные сети и их применение в системах управления и связи [Текст] / В. И. Комашинский, Д. А. Смирнов. М.: Горячая линия - Телеком, 2003. - 93 с. ISBN 5-93517-094-9. Экземпляры: всего 4.	4
5.	Барский, Аркадий Бенционович. Нейронные сети [Текст] : распознавание, управление, принятие решений / А. Б. Барский. М.: Финансы и статистика, 2004. - 175 с. ISBN 5-279-02757-X. Экземпляры: всего 9.	9
6.	Методы и модели анализа данных: OLAP и Data Mining [Текст] : учеб. пособие по специальности 071900 "Информ. системы и технологии" направления 654700 "Информ. системы" / А. А. Барсегян, М. С. Куприянов, В. В. Степаненко, И. И. Холод. СПб.: БХВ-Петербург, 2004. - 336 с. ISBN 5-94157-522-X. Экземпляры: всего 16.	16
7.	Гвоздева, Т. В. Проектирование информационных	

	технология автоматизированного проектирования. Лабораторный практикум [Электронный ресурс] / Гвоздева Т. В., Баллод Б. А. 2-е изд., стер. Санкт-Петербург: Лань, 2020. - 156 с. ISBN 978-5-8114-5147-0.	https://e.lanbook.com/book/133477
8.	Чубукова, И. А. Data Mining [Электронный ресурс] / Чубукова И. А. 2-е изд. Москва: ИНТУИТ, 2016. - 470 с. ISBN 978-5-94774-819-2.	https://e.lanbook.com/book/100582
9.	Макшанов, А. В. Большие данные. Big Data [Электронный ресурс] : учебник для вузов / Макшанов А. В., Журавлев А. Е., Тындыкарь Л. Н.; Журавлев А. Е., Тындыкарь Л. Н. 3-е изд., стер. Санкт-Петербург: Лань, 2023. - 188 с. ISBN 978-5-507-46866-9.	https://e.lanbook.com/book/322664

6.2. Материально-техническая база и программное обеспечение

№№ п/п	Аудитории для проведения учебных занятий, самостоятельной работы и проведения государственной итоговой аттестации	Перечень основного оборудования	Программное обеспечение
1.	333б (II)	Комплект учебной мебели (1)	Microsoft Windows Enterprise, Справочная правовая система "Консультант Плюс", Microsoft Office Standard, Агент Dr.Web, Комплект ГАРАНТ-Мастер, Microsoft Access, Microsoft Visio Professional, Microsoft Project Professional, Microsoft Visual Studio Enterprise, Комплект ПО для решения основных пользовательских задач, Mathcad University Classroom Perpetual - 40, LABVIEW
2.	333г (III)	Измерительный прибор "BerCut-E" (1), Комплекс ПАИК/77100/КПВ (1), Комплект дополнит.оборудования к ПАИК/7710/КПВ(автогенератор AnCom и автоответчи (1), Компьютер P4-3.0/2*256Mb/HDD 200Gb/128 6600GT/DVD-RW/KM/FDD/MBi945P/UPS (1), Ксерокс Canon FC-860 (1), Междисциплинарная лабораторная платформа в комплекте с аппаратно-программным кон (10), Монитор 19"Samsung 940N (LKSB) TFT (1).	Microsoft Windows Enterprise, Справочная правовая система "Консультант Плюс", Microsoft Office Standard, Агент Dr.Web, Комплект ГАРАНТ-Мастер, Microsoft Access, Microsoft Visio Professional, Microsoft Project Professional, Microsoft Visual Studio Enterprise, Комплект ПО для решения основных

		Принтер HP Laser Jet 1100 (1), Систем.блок Core2 DUOE6300/1024Мб*2/320Gb/DVD-RW/клав.мышь.ковр. (1), Комплект учебной мебели (1)	пользовательских задач, Mathcad University Classroom Perpetual - 40, LABVIEW
3.	431 (III)	Доска маркерная 120x240 см. ауд.431 (2), Кабель VGA 30.5 М KRAMER (П-П) (1), Проектор мультимедийный Hitachi CP-X8150 (1), Экран настенный с электроприводом 400x300см (1), Комплект учебной мебели (1)	Microsoft Windows Enterprise, Справочная правовая система "Консультант Плюс", Microsoft Office Standard, Агент Dr.Web, Комплект ГАРАНТ-Мастер, Microsoft Access, Microsoft Visio Professional, Microsoft Project Professional, Microsoft Visual Studio Enterprise, Комплект ПО для решения основных пользовательских задач, Mathcad University Classroom Perpetual - 40, LABVIEW
4.	433 (III)	Акустический комплект с микшером CRATE PA 8FX (1), Микшерный пульт компактный малошумящий Behringer XENYX 1202FX (1), Подавитель шума ALTAIR NG-440 4-х канальный (1), Экран настенный с электроприводом 400*300 Braun (1), Комплект учебной мебели (1)	Microsoft Windows Enterprise, Справочная правовая система "Консультант Плюс", Microsoft Office Standard, Агент Dr.Web, Комплект ГАРАНТ-Мастер, Microsoft Access, Microsoft Visio Professional, Microsoft Project Professional, Microsoft Visual Studio Enterprise, Комплект ПО для решения основных пользовательских задач, Mathcad University Classroom Perpetual - 40, LABVIEW
5.	538 (III)	Системный блок RAY P360 3 ,клав,мышь оптич, коврик+ монитор 19" ViewSonic VA916 (1), Комплект учебной мебели (1)	Microsoft Windows Enterprise, Справочная правовая система "Консультант Плюс", Microsoft Office Standard, Агент Dr.Web, Комплект ГАРАНТ-Мастер, Microsoft Access, Microsoft Visio Professional, Microsoft Project Professional,

		Microsoft Visual Studio Enterprise, Комплект ПО для решения основных пользовательских задач, Mathcad University Classroom Perpetual - 40, LABVIEW
--	--	---

Раздел 7. ФОРМЫ КОНТРОЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ/ ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

Критерии оценивания индикаторов достижения компетенций направлены на:

- усвоение теоретического материала (объем знаний, глубина усвоения), предусмотренного рабочей программой;
- умение излагать материал (четкость, грамотность изложения материала, точность и полнота воспроизведения учебного материала);
- умение применять теоретические знания при решении практических заданий.

Шкала оценивания представлена ниже.

Уровень сформированности элементов компетенции	Критерии оценивания	Шкала оценивания
Пороговый уровень	Обучающийся имеет знания основного материала, проявляет умение логично его излагать, но может допускать неточности в изложении материала, недостаточно правильные формулировки, испытывает затруднения в выполнении практических заданий.	удовлетворительно
Продвинутый уровень	Обучающийся твердо знает программный материал, излагает его грамотно и по существу, не допускает существенных неточностей в ответе на вопрос, правильно применяет теоретические положения при решении практических вопросов и задач, владеет необходимыми навыками и приемами их выполнения	хорошо
Высокий уровень	Обучающийся глубоко и прочно усвоил программный материал, грамотно и логически стройно его излагает, дает исчерпывающие ответы на поставленные вопросы. В ответе тесно увязывается теория с практикой, при этом обучающийся не затрудняется с ответом при видоизменении задания, свободно справляется с задачами, вопросами и другими видами применения знаний, показывает знакомство с монографической литературой, периодическими изданиями, правильно обосновывает принятые решения, свободно владеет разносторонними навыками, приемами выполнения практических работ	отлично

7.1. Текущий контроль успеваемости

Текущий контроль успеваемости обеспечивает оценивание хода освоения дисциплины (модуля) и производится с применением технологии рейтингового контроля в соответствии с технологической картой дисциплины. Порядок составления технологической карты и алгоритм проведения процедуры оценивания видов деятельности обучающихся, направленных на освоение знаний, умений, навыков и/ или опыта деятельности, по

накопительной системе в баллах устанавливается положением о системе РИТМ в ФГБОУ ВО «ПГТУ»

7.2. Промежуточная аттестация обучающихся

Промежуточная аттестация обучающихся направлена на оценивание результатов обучения по дисциплине (модулю) и проводится с использованием фондов оценочных средств.

Примеры типовых контрольных заданий из базы фонда оценочных средств по образовательной программе.

Компетенция PL1 Способен применять язык программирования Python для решения задач в области ИИ

Установление соответствия между объектами двух множеств:

1. Установите соответствие между менеджерами пакетов Python и их основными файлами конфигурации. pip conda poetry pipenv

a) requirements.txt b) environment.yml c) pyproject.toml d) Pipfile e) setup.py

1. 2. Установите соответствие между библиотеками Python и их основным назначением в экосистеме разработки систем ИИ. NumPy Pandas Scikit-learn Matplotlib

a) Многомерные массивы и векторизованные вычисления b) Работа с табличными данными и временными рядами c) Реализация алгоритмов машинного обучения d) Построение графиков и визуализация данных e) Глубокое обучение и нейронные сети

1. 3. Установите соответствие между операциями NumPy и их эффектом на производительность вычислений. Векторизация через broadcasting Использование циклов Python для операций с массивами Применение универсальных функций (ufunc) Индексирование с помощью fancy indexing

a) Ускорение вычислений за счёт операций без копирования данных b) Существенное замедление из-за интерпретируемой природы языка c) Максимальная эффективность поэлементных операций на уровне C d) Увеличение потребления памяти из-за создания копии данных e) Автоматическая параллелизация на несколько ядер процессора

1. 4. Установите соответствие между методами Pandas и их назначением при очистке данных. dropna() fillna() drop_duplicates() replace()

a) Удаление строк или столбцов с пропущенными значениями b) Заполнение пропущенных значений статистиками или константами c) Удаление дублирующихся записей в датасете d) Замена конкретных значений на другие e) Нормализация числовых признаков к диапазону [0, 1]

1. 5. Установите соответствие между модулями Python для параллелизма и их основными характеристиками. threading multiprocessing asyncio concurrent.futures

a) Потоки с ограничением GIL, подходящие для I/O-операций b) Отдельные процессы для обхода GIL и использования нескольких ядер c) Асинхронное программирование на корутинах для I/O-операций d) Высокоуровневый интерфейс с пулами потоков и процессов e) Автоматическая компиляция Python-кода в машинный код

1. 6. Установите соответствие между типами визуализаций в Matplotlib/Seaborn и задачами анализа данных, для которых они наиболее подходят. Гистограмма (histogram) Диаграмма рассеяния (scatter plot) Тепловая карта (heatmap) Ящик с усами (boxplot)

1. a) Исследование распределения одной числовой переменной b) Выявление зависимости между двумя числовыми признаками c) Визуализация матрицы корреляций между признаками d) Сравнение распределений и обнаружение выбросов e) Отображение изменения временного ряда с сезонностью

1. 7. Укажите метрики качества, которые подходят для оценки моделей бинарной классификации при несбалансированных классах.

a) F1-score b) ROC-AUC (Area Under ROC Curve) c) Accuracy (точность) d) Mean Squared Error (MSE)

Выбор нескольких правильных ответов из предложенных (множественный выбор):

1. 8. Определите методы кросс-валидации, которые сохраняют временную зависимость данных в задачах прогнозирования временных рядов.

a) TimeSeriesSplit b) Walk-forward validation c) Stratified K-Fold d) K-Fold с перемешиванием (shuffle=True)

1. 9. Выберите техники feature engineering, которые создают новые признаки на основе взаимодействия существующих переменных.

a) Polynomial features (полиномиальные признаки) b) Interaction features (признаки взаимодействия) c) One-hot encoding d) StandardScaler (стандартизация)

1. 10. Укажите алгоритмы машинного обучения, которые относятся к ансамблевым методам на основе бэггинга.

a) Random Forest b) Extra Trees (Extremely Randomized Trees) c) Gradient Boosting d) AdaBoost

1. 11. Определите методы оптимизации гиперпараметров, которые используют адаптивные стратегии поиска на основе предыдущих результатов.

a) Bayesian Optimization b) Optuna (TPE - Tree-structured Parzen Estimator) c) Grid Search d) Random Search

1. 12. Выберите техники препроцессинга, которые необходимо применять отдельно к обучающей и тестовой выборкам для предотвращения утечки данных.

a) StandardScaler (стандартизация) b) MinMaxScaler (нормализация) c) Удаление дубликатов строк d) Заполнение пропусков константным значением

1. 13. Укажите методы работы с несбалансированными данными, которые изменяют распределение классов в обучающей выборке.

a) SMOTE (Synthetic Minority Over-sampling Technique) b) Random undersampling c) Class weights (веса классов) d) Stratified sampling при разделении данных

1. 14. Определите компоненты, которые должны входить в полный ML Pipeline для задачи классификации.

1. a) Препроцессинг данных (нормализация, кодирование категорий) b) Feature engineering (создание новых признаков) c) Обучение модели и валидация d) Визуализация распределений признаков e) Написание документации проекта

Установление правильной последовательности в предложенной совокупности объектов:

1. 15. Расположите этапы прямого прохода (forward pass) в нейронной сети в правильной последовательности.

a) Подача входных данных на вход сети b) Последовательная обработка данных скрытыми слоями с применением функций активации c) Получение выходных предсказаний модели d) Вычисление функции потерь на основе предсказаний и истинных меток

1. 16. Установите последовательность шагов алгоритма обратного распространения ошибки (backpropagation).

a) Вычисление градиента функции потерь по выходу сети b) Обратное распространение градиентов через слои с применением правила цепочки c) Вычисление градиентов функции потерь по весам каждого слоя d) Обновление весов с использованием оптимизатора

1. 17. Расположите компоненты сверточной нейронной сети (CNN) в типичном порядке их следования в архитектуре для классификации изображений.
a) Сверточные слои (Convolutional layers) b) Слои пулинга (Pooling layers) c) Flatten (преобразование в одномерный вектор) d) Полносвязные слои (Fully Connected layers)
1. 18. Установите последовательность этапов процесса transfer learning при адаптации предобученной модели к новой задаче.
a) Загрузка предобученной модели (например, ResNet, VGG) b) Замена последних слоев классификации на новые c) Добавление слоев для новой задачи с нужным количеством выходов d) Заморозка весов предобученных слоев и обучение новых слоев
1. 19. Расположите методы регуляризации в порядке возрастания их влияния на архитектуру нейронной сети.
a) L2-регуляризация (Weight Decay) b) Batch Normalization c) Dropout d) Early Stopping
1. 20 Установите последовательность этапов подготовки модели глубокого обучения к production deployment.
a) Обучение и валидация модели до достижения требуемого качества b) Оптимизация модели (квантизация, pruning) c) Экспорт модели в формат для deployment (ONNX, TorchScript) d) Создание inference API и контейнеризация (Docker)

Перечень вопросов для проведения промежуточной аттестации

Модуль 1. Инструментальные средства разработки на Python

2. Экосистема Python для разработки систем искусственного интеллекта: основные библиотеки и инструменты. Управление зависимостями и виртуальные окружения.
3. Векторизация операций в NumPy. Broadcasting и его применение для эффективных вычислений. Линейная алгебра с NumPy.
4. Обработка и анализ табличных данных с помощью pandas. DataFrame и Series: основные операции, фильтрация, группировка, агрегация.
5. Визуализация данных для машинного обучения. Библиотеки Matplotlib, Seaborn, Plotly: назначение и применение.
6. Многопоточность и параллелизм в Python. GIL и его влияние на производительность. Модуль threading и multiprocessing: сравнение и области применения.

Модуль 2. Машинное обучение и классические алгоритмы

7. Типы задач машинного обучения: классификация, регрессия, кластеризация. Pipeline машинного обучения. Разделение данных на обучающую, валидационную и тестовую выборки.
8. Метрики качества моделей машинного обучения. Переобучение и недообучение. Методы кросс-валидации.
9. Препроцессинг данных в машинном обучении: масштабирование признаков, кодирование категориальных переменных, обработка пропущенных значений.
10. Классические алгоритмы машинного обучения: линейная и логистическая регрессия, деревья решений, случайные леса. Их особенности и области применения.
11. Градиентный бустинг (XGBoost, LightGBM, CatBoost). Принципы работы и преимущества. Настройка гиперпараметров с использованием GridSearchCV и RandomizedSearchCV.

Модуль 3. Глубокое обучение и нейронные сети

12. Основы глубокого обучения. Архитектура нейронных сетей: слои, функции активации, функции

потерь. Процесс обучения: forward pass и backward propagation.

13. Фреймворки для глубокого обучения: PyTorch и TensorFlow. Сравнение подходов. Тензоры и автоматическое дифференцирование (autograd).
14. Сверточные нейронные сети (CNN): архитектура, операции свертки и pooling. Классические архитектуры: LeNet, VGG, ResNet. Применение для задач компьютерного зрения.
15. Рекуррентные нейронные сети (RNN), LSTM и GRU. Обработка последовательностей и временных рядов. Механизм внимания (Attention).
16. Техники обучения нейронных сетей: регуляризация (Dropout, Batch Normalization), оптимизаторы (SGD, Adam, RMSprop), learning rate scheduling. Transfer Learning и fine-tuning.

Модуль 4. Серверные приложения и обработка больших данных

17. Разработка REST API для ML-моделей. Веб-фреймворки Python: FastAPI, Flask, Django REST Framework. Валидация данных и документирование API.
18. Основы MLOps: версионирование кода и данных (Git, DVC), experiment tracking (MLflow, Weights & Biases), мониторинг моделей в продакшене.
19. Apache Spark и PySpark: архитектура и компоненты. RDD, DataFrame и Dataset API. Распределенная обработка данных.
20. Машинное обучение на больших данных с использованием Spark MLlib. Особенности распределенного обучения моделей.
21. Apache Airflow для оркестрации ML-пайплайнов. Концепция DAG (Directed Acyclic Graph). Операторы и задачи. Автоматизация процессов обучения и развертывания моделей.