

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Тольяттинский государственный университет»

Б1.В.12

(индекс дисциплины)

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Инженерия машинного обучения для производственных процессов

(наименование дисциплины)

по направлению подготовки
09.03.04 Программная инженерия

направленность (профиль)
Программная инженерия с применением ИИ-технологий

Форма обучения: заочная

Год набора: 2024

Общая трудоемкость: 3 ЗЕ

Распределение часов дисциплины по семестрам

Семестр	8	Итого
Форма контроля	зачет	
Вид занятий		
Лекции	4	4
Лабораторные		
Практические		
Руководство: курсовые работы (проекты) / РГР		
Промежуточная аттестация	0,25	0,25
Контактная работа	4,25	4,25
Самостоятельная работа	100	100
Контроль	3,75	3,75
Итого	108	108

Рабочую программу составил(и):

Доцент института цифровых технологий, канд.пед.наук., Ерофеева Е.А.

(должность, ученое звание, степень, Фамилия И.О.)

Рецензирование рабочей программы дисциплины:



Отсутствует



Рецензент

(должность, ученое звание, степень, Фамилия И.О.)

Рабочая программа дисциплины составлена на основании ФГОС ВО и учебного плана направления подготовки

09.03.04 Программная инженерия

Срок действия рабочей программы дисциплины до «31» августа 2031 г.

УТВЕРЖДЕНО

На заседании института цифровых технологий

(протокол заседания № 1 от «05» сентября 2025 г.).

1. Цель освоения дисциплины

Цель дисциплины сформировать у обучающихся системные знания и практические навыки разработки, внедрения и эксплуатации решений машинного обучения в производственных процессах, включая анализ данных, выбор и построение моделей, их интеграцию в программные системы, оценку эффективности и обеспечение качества работы ИИ-компонентов в условиях реальных производственных сред.

2. Место дисциплины в структуре ОПОП ВО

Дисциплины и практики, на освоении которых базируется данная дисциплина: Тестирование и верификация систем с ИИ-компонентами, Системы искусственного интеллекта, Архитектура информационных систем и методы интеграции.

Дисциплины и практики, для которых освоение данной дисциплины необходимо как предшествующее: Производственная практика (преддипломная практика), Выполнение и защита выпускной квалификационной работы.

3. Планируемые результаты обучения

Формируемые и контролируемые компетенции (код и наименование)	Индикаторы достижения компетенций (код и наименование)	Планируемые результаты обучения
ПК-6 Способен разрабатывать программные продукты со встроенной аналитикой больших данных	ПК-6.1. Знает понятие больших данных, методы работы с большими данными	Знать: технологии и инструменты для работы с большими данными Уметь: выбирать подходящие технологии для хранения и обработки больших объемов данных. Владеть: навыками проектирования моделей данных для аналитических задач.
	ПК-6.2. Умеет анализировать большие данные, разрабатывать программные продукты	Знать: как интегрировать аналитические модули в основное приложение. Уметь: проектировать архитектуру ПО, которая поддерживает сбор, обработку и анализ больших данных. Владеть: навыками создания прототипов аналитических функций.
	ПК-6.3. Владеет навыками создания программных продуктов со встроенной аналитикой больших данных	Знать: особенности реализации алгоритмов для распределенной обработки данных. Уметь: реализовывать компоненты для аналитики

Формируемые и контролируемые компетенции (код и наименование)	Индикаторы достижения компетенций (код и наименование)	Планируемые результаты обучения
		больших данных на практике. Владеть: навыками работы с фреймворками для встраивания аналитики в приложения.
ПК-7 Способен разрабатывать сервисы на основе аналитики больших данных	ПК-7.1. Понимает принципы разработки сервисов, приемы аналитики больших данных	Знать: принципы разработки сервисов; методы аналитики (дескриптивная, диагностическая, предиктивная, предписывающая аналитика). Уметь: проектировать сервисно-ориентированную архитектуру. Владеть: навыками проектирования API для аналитических сервисов.
	ПК-7.2. Умеет разрабатывать сервисы на основе аналитики больших данных	Знать: как работать с потоковыми данными. Уметь: разрабатывать микросервисы, которые предоставляют аналитические функции (например, рекомендации, прогнозирование). Владеть: практическими навыками создания веб-сервисов с интеграцией аналитических библиотек.

4. Структура и содержание дисциплины

Модуль (раздел)	Вид учебной работы	Наименование тем занятий (учебной работы)	Семестр	Объем, ч.	Баллы	Интерактив, ч.	Формы текущего контроля (наименование оценочного средства)
Модуль 1. Машинное обучение для производственных процессов	Лек 1	Тема 1. Производственные данные и задачи ML в промышленности	8	2		-	
	Ср	Тема 2. Предобработка и очистка производственных данных. Фильтрация, ресемплирование, заполнение пропусков, выбросы.	8	7		-	
	Ср	Тема 3. Базовые модели ML в производстве. Деревья, ансамбли, линейные модели, ввод в модели для временных рядов.	8	7		-	
	Ср	Тема 4. Модели обнаружения аномалий в производственных системах Isolation Forest, AutoEncoder, статистические методы.	8	7		-	
	Ср	ПР1. Анализ промышленного датасета (датчики, временные ряды)	8	8	8	-	Отчет по практической работе 1
	Ср	ПР2. Подготовка производственного датасета (feature engineering).	8	8	8	-	Отчет по практической работе 2
	Ср	ПР3. Обучение первой модели (качество продукции / отказ оборудования).	8	8	8	-	Отчет по практической работе 3
	Ср	ПР4. Обнаружение аномалий в технологическом процессе.	8	8	8	-	Отчет по практической работе 4
Модуль 2. Интеграция и эксплуатация ML в производственных ИТ-системах	Лек 2	Тема 5. ML-сервис: архитектура и API. REST, gRPC, структура сервиса, сервис-модель, ML-node.	8	2		-	
	Ср	Тема 6. Контейнеризация ML-моделей. Docker, структура образа, оптимизация, reproducibility.	8	7		-	

Модуль (раздел)	Вид учебной работы	Наименование тем занятий (учебной работы)	Семестр	Объем, ч.	Баллы	Интерактив, ч.	Формы текущего контроля (наименование оценочного средства)
	Ср	Тема 7. MLOps и мониторинг моделей MLflow, drift, degrade, переобучение.	8	7		-	
	Ср	Тема 8. Надёжность и безопасность ML в производственных системах логирование, fault-tolerance, защита API, ресурсная оптимизация.	8	7		-	
	Ср	ПР5. Создание простого ML-API (FastAPI / Flask / Spring). (классификация/рекомендации)	8	8	8	-	Отчет по практической работе 5
	Ср	ПР6. Создание Docker-образа ML-сервиса.	8	8	8	-	Отчет по практической работе 6
	Ср	ПР7. Мониторинг модели и drift-детекция. ПР8. ML-сервис для производственного процесса.	8	10	6 6	-	Отчет по практической работе 7 Отчет по практической работе 8
	ПА	Промежуточная аттестация	8	0,25		—	
	Контроль	зачет	8	3,75	40	-	Итоговый тест
Итого:				108	100		

Схема расчета итогового балла:

Текущий рейтинг (все занятия и промежуточные тесты) + Результат итогового теста.

5. Образовательные технологии

При изучении дисциплины используются следующие образовательные технологии:

- **технологии традиционного обучения** – лекции, практические занятия и самостоятельная работа обучающихся, направленные на освоение принципов машинного обучения, особенностей производственных данных, методов анализа и интеграции моделей;
- **технология проектного обучения**, предусматривающая разработку фрагментов производственных ML-решений: подготовка датасетов, обучение моделей прогнозирования отказов, создание сервисов для обнаружения аномалий, реализация API и контейнеризация ML-модулей. Обучающиеся защищают результаты практических работ и демонстрируют работоспособность созданных прототипов;
- **практико-ориентированное обучение**, основанное на выполнении реальных инженерных задач: подготовка данных промышленного процесса, разработка и тестирование моделей, построение ML-сервисов, интеграция моделей с веб-инфраструктурой, работа с Docker, анализ корректности интеграции и функционирования алгоритмов в условиях производственной среды.

Технологии традиционного обучения включают объяснительно-иллюстративный формат, применяемый при изучении теоретических основ машинного обучения, особенностей производственных данных, моделей и алгоритмов анализа, принципов построения ML-сервисов и интеграции ИИ-компонентов.

Организация учебного процесса предполагает активное вовлечение обучающихся в практическую деятельность: выполнение работ по обработке производственных данных, обучению моделей, реализации API, контейнеризации решений, настройке мини-MLOps-процессов, анализу ошибок и работе с мониторингом.

На практических занятиях обучающиеся:

- демонстрируют работоспособность разработанных ML-моделей и сервисов;
- обосновывают выбранные методы предобработки данных, архитектуру сервиса и вариант интеграции;
- обсуждают возникающие технические проблемы и пути их решения;
- проводят мини-защиту выполненной работы.

Итоговая аттестация проводится в форме **экзамена**, включающего оценку теоретических знаний и практических навыков разработки и интеграции моделей машинного обучения для производственных процессов.

6. Методические указания по освоению дисциплины

6.1. Рекомендации по подготовке к практическим занятиям

Обучающимся следует:

- использовать при подготовке не только материалы лекций и учебную литературу, но и официальную документацию по библиотекам машинного обучения (scikit-learn, PyTorch/TF при необходимости), инструментам обработки данных, средствам контейнеризации и фреймворкам для реализации ML-сервисов;
- заранее ознакомиться с форматами производственных данных (временные ряды датчиков, технологические события, параметры оборудования), примерами исходного кода, структурой API и особенностями передачи данных к ML-модели;
- в начале занятия уточнить у преподавателя вопросы, вызвавшие затруднения, чтобы избежать ошибок при предобработке данных, обучении модели, построении сервиса или интеграции ML-модуля;

- стремиться доводить каждое практическое задание до полностью работоспособного состояния, уметь продемонстрировать корректность работы модели и сервиса, объяснить принятые инженерные и архитектурные решения.

Практические занятия тесно связаны с лекционным материалом: особенностями производственных данных, методами машинного обучения, принципами построения ML-сервисов, механизмами интеграции и контейнеризации, а также основами мониторинга моделей. Только после освоения этих теоретических основ возможно качественное выполнение заданий по реализации данных и сервисных модулей.

При самостоятельной работе над заданиями необходимо:

- обосновывать каждый этап разработки, включая выбор метода предобработки данных, архитектуры модели, схемы API, способа интеграции ML-модуля, стратегии обработки ошибок и структуры хранения данных;
- анализировать несколько вариантов решения (например, прямой вызов модели из веб-сервиса или вынесение её в отдельный ML-узел) и выбирать наиболее рациональный вариант с точки зрения производительности, отказоустойчивости, масштабируемости и удобства поддержки;
- предварительно составлять план реализации, включая структуру проекта, взаимодействующие компоненты, источники данных, последовательность шагов и точки интеграции;
- сопровождать решения комментариями, схемами, диаграммами потоков данных или архитектурными эскизами, если требуется пояснить структуру сервиса или логику работы модели.

Каждая практическая работа должна завершаться получением функционирующего результата — ML-модуля, модели или веб-сервиса, готового к тестированию, мониторингу или развёртыванию. Обучающийся должен самостоятельно проверить разработанный компонент: протестировать основные маршруты API, корректность взаимодействия с моделью, поведение при некорректных данных и устойчивость к ошибкам.

Развитие навыков инженерии машинного обучения достигается регулярной самостоятельной практикой, анализом примеров, экспериментированием с моделями и архитектурами, а также изучением профессиональной документации и рекомендаций по разработке производственных ML-систем.

6.2. Рекомендации по подготовке к зачету

Подготовка к зачёту направлена на систематизацию, закрепление и обобщение теоретических знаний и практических навыков, полученных в ходе изучения дисциплины. Зачёт проверяет сформированность элементов компетенций, связанных с анализом производственных данных, разработкой моделей машинного обучения, построением ML-сервисов и обеспечением их корректной работы в производственной среде.

Для успешной подготовки обучающимся рекомендуется последовательно повторить ключевые темы лекционного курса:

- особенности производственных данных (датчики, временные ряды, технологические события);
- методы предобработки данных, устранение выбросов и пропусков, формирование признаков;
- базовые алгоритмы машинного обучения и их применение к производственным задачам;
- методы обнаружения аномалий и прогнозирования отказов;
- архитектура ML-сервисов и принципы проектирования API;
- основы контейнеризации (Docker) и развёртывания моделей;
- основы мониторинга моделей, drift-детекции и контроля качества.

Рекомендуется проанализировать выполненные практические работы, уделив внимание:

- корректности подготовки данных и формированию обучающей выборки;
- качеству реализованных ML-моделей и обоснованию выбранных методов;
- способам передачи данных между сервисом и моделью, структуре API;
- работе ML-сервиса и корректности обработки запросов;
- созданию и использованию Docker-контейнеров;
- настройке окружения, установке зависимостей и использованию документации.

Также важно пересмотреть примеры интеграции ML-моделей, изученные в ходе курса:

- схемы взаимодействия веб-сервиса и ML-модуля;
- особенности предобработки производственных данных;
- распространённые ошибки интеграции и способы их устранения;
- методы повышения устойчивости сервиса и улучшения качества модели.

Для систематизации знаний рекомендуется повторить основные архитектурные решения, применяемые при построении ML-сервисов:

- проектирование сервисов с учётом особенностей производственной среды;
- разделение функциональности между компонентами (обработка данных, обучение, инференс);
- типовые точки отказа, способы их минимизации, логирование и мониторинг.

Подготовка к зачёту должна проводиться на протяжении всего семестра. Желательно активно участвовать в обсуждениях на занятиях, выполнять практические работы в полном объёме, анализировать примеры реализации ML-сервисов и регулярно обращаться к документации по используемым библиотекам. Такая работа формирует глубокое понимание материала и обеспечивает успешное прохождение зачёта.

7. Оценочные средства

7.1. Паспорт оценочных средств

Семестр	Код контролируемой компетенции (или ее части)	Наименование оценочного средства
8	ПК-7 ПК-6	Тестовые задания 1-450 Вопросы к зачету 1-80 Отчеты по практическим работам 1-8

7.2. Типовые задания или иные материалы, необходимые для текущего контроля

7.2.1. Пример практической работы

Цель работы

Изучить структуру типичных промышленных данных (датчиков и событий), научиться выполнять первичный анализ временных рядов, определять основные характеристики технического процесса и выявлять потенциальные проблемы данных: выбросы, шум, пропуски, сдвиги.

Исходные данные

Используется публичный промышленный датасет одного из следующих типов (преподаватель выбирает заранее):

- **NASA Turbofan Engine Degradation Dataset (C-MAPSS)** — временные ряды датчиков двигателя.
- **SECOM Manufacturing Sensor Dataset** — показания датчиков оборудования.
- **Kaggle: Pump Sensor Data / Bearing Failure Data** — вибрации, температура, нагрузки.

Формат данных:

CSV / TSV

Содержит: timestamp, sensor_1 ... sensor_N, operational_settings.

Задания для выполнения

Задание 1. Загрузка и первичный просмотр

1. Загрузить датасет (CSV).
2. Вывести первые 10 строк.
3. Определить:
 - количество датчиков,
 - количество записей,
 - временной охват данных.

Задание 2. Анализ временной структуры

1. Проверить наличие пропусков (NaN) и заполнить таблицу:
 - «Количество пропусков на каждый датчик»
 - «Доля пропусков в %»
2. Проверить равномерность временного шага (например, есть ли пропуски по времени).

Задание 3. Выявление выбросов

1. Для 2–3 выбранных датчиков построить:
 - график временного ряда;
 - boxplot.
2. Сделать выводы:
 - есть ли выбросы;
 - стабильна ли работа датчиков;
 - присутствует ли дрейф значений.

Задание 4. Первичный анализ корреляций

1. Построить корреляционную матрицу между датчиками.
2. Определить:
 - какие датчики наиболее связаны,
 - есть ли признаки избыточности.

Задание 5. Формулировка наблюдений

Записать краткие выводы по каждому пункту:

- особенности данных;
- обнаруженные проблемы;
- датчики со стабильной/нестабильной динамикой;
- возможные гипотезы о работе оборудования.

4. Вопросы для обсуждения

1. Почему производственные датасеты часто содержат шум и выбросы?
2. Чем отличаются «пропуски данных» от «неравномерного временного шага»?
3. Какие датчики могут быть наиболее важными для прогноза отказа?
4. Почему корреляция между датчиками помогает в feature engineering?

5. Отчёт по работе (что сдаёт студент)

Отчёт должен включать:

1. Краткое описание датасета.
2. Таблицу пропусков.
3. Графики временных рядов (2–3 датчика).
4. Корреляционную матрицу.

5. Выводы:

- качество данных;
- найденные проблемы;
- интерпретация динамики датчиков.

Формат отчёта: PDF или DOCX.

6. Минимальный код-шаблон (для практики)

Только структура, без лишних деталей — чтобы студент мог запустить и дополнить.

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# 1. Загрузка данных
df = pd.read_csv("dataset.csv")

# 2. Первичный осмотр
print(df.head())
print(df.info())

# 3. Анализ пропусков
print(df.isna().sum())

# 4. Построение временного ряда по выбранному датчику
plt.figure(figsize=(12,4))
plt.plot(df["sensor_1"])
plt.title("Sensor 1 Time Series")
plt.show()

# 5. Boxplot
sns.boxplot(x=df["sensor_1"])
plt.show()

# 6. Корреляционная матрица
plt.figure(figsize=(10,8))
sns.heatmap(df.corr(), annot=False, cmap="viridis")
plt.show()
```

Требования к оформлению

Отчет должен содержать подробное описание (включая иллюстрации). Отчёт по практическому занятию выполняется на страницах формата А4 в электронном виде.

При оформлении отчёта используется сквозная нумерация страниц, считая титульный лист первой страницей. Номер страницы на титульном листе не ставится. Номера страницы ставятся по центру сверху.

При оформлении отчёта соблюдать следующие требования:

- Для заголовков: полужирный шрифт, 14 пт, центрированный.
- Для основного текста: нежирный шрифт, 14 пт, выравнивание по ширине.
- Во всех случаях тип шрифта – Times New Roman, отступ абзаца 1.25 см, полуторный междустрочный интервал.
- Поля: левое – 2 см, правое, верхнее и нижнее – 1 см.

Процедура оценивания

Оценка выполненной практической работы проводится по следующим критериям:

1. Наличие всей существенной информации по работе
2. Точность и полнота предоставляемых сведений
3. Непротиворечивость приводимой информации
4. Правильность интерпретаций и выводов, которые сделаны по результатам работы
5. Степень достижения обучающимся поставленной цели
6. Обоснованность применяемого решения
7. Грамотность (содержательная) используемых формулировок

Критерии оценки за отчеты по практическим работам:

Формы текущего контроля	Критерии и нормы оценки
Отчеты по практическим работам 1-6	8 баллов – задание выполнено в полном объеме без замечаний 6-7 баллов – задание выполнено в полном объеме, присутствуют замечания 4-5 баллов – задание выполнено в объеме 70% 1-3 балла – задание выполнено в объеме менее 50% 0 баллов – задание не выполнено.
Отчеты по практическим работам 7, 8	5-6 баллов – задание выполнено в полном объеме без замечаний 3-4 балла – задание выполнено в объеме 70%, 1-4 балл – задание выполнено в объеме менее 50%, 0 баллов – задание не выполнено.

7.2.2. Тестовые задания

1. Какие данные чаще всего встречаются в промышленности?
 - a) Изображения
 - b) Аудио
 - c) **Временные ряды датчиков**
 - d) Текстовые данные
2. Что является распространённой проблемой производственных данных?
 - a) Слишком много изображений
 - b) **Шум и выбросы**
 - c) Малая ширина таблиц
 - d) Много категорий
3. Что такое выброс?
 - a) Отсутствующий признак
 - b) Повторяющееся значение
 - c) **Значение, резко отличающееся от нормальных**
 - d) Лишний столбец
4. Какой алгоритм подходит для обнаружения аномалий?
 - a) Random Forest
 - b) Logistic Regression
 - c) KNN
 - d) **Isolation Forest**
5. Что такое feature engineering?
 - a) Переименование файлов
 - b) **Создание новых признаков**

- c) Настройка графиков
- d) Контейнеризация
- 6. Что означает предобработка данных?
 - a) Настройка API
 - b) **Очистка, нормализация, обработка пропусков**
 - c) Подготовка интерфейса
 - d) Обучение модели
- 7. Тип задачи для прогнозирования отказов оборудования:
 - a) Кластеризация
 - b) Сегментация
 - c) **Классификация**
 - d) Генерация данных
- 8. Что такое временной ряд?
 - a) Случайные события
 - b) Категории
 - c) **Последовательность значений во времени**
 - d) Неподписанные данные
- 9. Алгоритм для прогноза RUL:
 - a) GAN
 - b) CNN
 - c) **Линейная регрессия / Gradient Boosting**
 - d) DBSCAN
- 10. Что выявляет пропуски?
 - a) Dockerfile
 - b) Voxplot
 - c) Графики
 - d) **isna()**
- 11. Что показывает корреляция?
 - a) Размер памяти
 - b) Бронь оборудования
 - c) **Связь между признаками**
 - d) Время работы

Критерии оценки за пройденный тест:

- 40 баллов выставляется обучающемуся, если он ответил правильно на все вопросы случайной выборки 30 тестовых заданий;
- 0-39 баллов выставляется обучающемуся в зависимости от количества верных ответов на вопросы случайной выборки 30 тестовых заданий.

7.3. Оценочные средства для промежуточной аттестации по итогам освоения дисциплины

7.3.1. Вопросы к промежуточной аттестации

Семестр 8

№	Вопросы к зачету
1.	Что относится к производственным данным?
2.	Какой тип данных чаще всего представлен временными рядами?

№	Вопросы к зачету
3.	Что является распространённой проблемой датчиков?
4.	Что такое выброс в данных?
5.	Что такое пропуски данных?
6.	Что чаще всего вызывает дрейф данных?
7.	Что такое временной шаг временного ряда?
8.	Какой метод используется для обнаружения аномалий?
9.	Какой алгоритм является ансамблевым?
10.	Какой алгоритм подходит для прогноза отказов оборудования?
11.	Какой формат используется для хранения временных рядов?
12.	Что такое feature engineering?
13.	Что является преимуществом нормализации данных?
14.	Что показывает корреляционная матрица?
15.	Какой метод помогает выявить зависимость между датчиками?
16.	Что такое ML-модель?
17.	Что является целевой переменной?
18.	Что принимает на вход /predict?
19.	Что такое API?
20.	Что такое REST-сервис?
21.	Какой формат данных используется в API?
22.	Что такое ML-сервис?
23.	Какой элемент обеспечивает обмен данными в сервисе?
24.	Что такое endpoint?
25.	Что делает контейнер Docker?
26.	Что содержит Dockerfile?
27.	Что делает команда docker build?
28.	Для чего используется docker run?
29.	Зачем использовать requirements.txt?
30.	Что обеспечивает reproducibility?
31.	Что делает MLflow?
32.	Что такое эксперимент в MLflow?
33.	Что такое параметр модели?
34.	Что такое метрика качества?
35.	Что показывает accuracy?

№	Вопросы к зачету
36.	Что показывает MAE?
37.	Что такое обучающая выборка?
38.	Что такое тестовая выборка?
39.	Что означает overfitting?
40.	Что означает underfitting?
41.	Что такое кросс-валидация?
42.	Какие данные называются производственными?
43.	Что такое SCADA?
44.	Что такое IoT?
45.	Что измеряет датчик вибрации?
46.	Что показывает температура оборудования?
47.	Зачем анализировать тренды данных?
48.	Что такое деградация модели?
49.	Что является причиной drift?
50.	Что такое data drift?
51.	Что такое concept drift?
52.	Зачем нужен мониторинг модели?
53.	Что делает alert в мониторинге?
54.	Что такое логирование?
55.	Что обычно логируется в ML-сервисе?
56.	Что такое ошибка 500 в API?
57.	Что означает ошибка 400?
58.	Что такое JSON?
59.	Что такое timestamp?
60.	Для чего нужен pandas?
61.	Для чего используется matplotlib?
62.	Чем полезны boxplot?
63.	Что показывает график временного ряда?
64.	Что такое окно сглаживания (moving average)?
65.	Что помогает выявить резкие скачки?
66.	Что такое baseline-модель?
67.	Что является задачей классификации?
68.	Что является задачей регрессии?

№	Вопросы к зачету
69.	Что такое обученная модель?
70.	Что такое предсказание модели?
71.	Что делает функция predict()?
72.	Что такое сериализация данных?
73.	Что такое модельный pipeline?
74.	Зачем раздельно хранить модель и данные?
75.	Что такое архитектура ML-node?
76.	Что такое latency?
77.	Что такое throughput?
78.	Для чего используется Docker Compose?
79.	Что делает healthcheck для сервиса?
80.	Что является целью интеграции ML в производственные процессы?

7.3.2. Критерии и нормы оценки

Семестр	Форма проведения промежуточной аттестации	Критерии и нормы оценки	
8	Зачет	«зачтено»	55-100 баллов
		«не зачтено»	Менее 55 баллов

8. Учебно-методическое и информационное обеспечение дисциплины

8.1. Обязательная литература

№ п/п	Авторы, составители	Заглавие (заголовок)	Тип (учебник, учебное пособие, учебно-методическое пособие, практикум, др.)	Год издания	Количество в научной библиотеке / Наименование ЭБС
1.	Аншина М. Л.	Аншина М. Л. Архитектура приложений и данных [Электронный ресурс] : учебное пособие / М. Л. Аншина. — М. : МИРЭА — Российский технологический университет, 2024.	учебное пособие	2025	ЭБС «Лань»
2.	Турнецкая, Е. Л.	Тестирование и контроль качества программного обеспечения : учебное пособие / Е. Л. Турнецкая, А. В. Аграновский, А. А. Сенцов. — Санкт-Петербург : ГУАП, 2023. — 118 с. — ISBN 978-5-8088-1891-0. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/461498 (дата обращения: 28.11.2025)	учебное пособие	2023	ЭБС «Лань»
3.	Ю. А. Антохина, М. Л. Кричевский, Ю. А. Мартынова, А. А. Оводенко	Искусственный интеллект. Инноватика : учебное пособие / Ю. А. Антохина, М. Л. Кричевский, Ю. А. Мартынова, А. А. Оводенко. — Санкт-Петербург : ГУАП, 2023. — 320 с. — ISBN 978-5-8088-1830-9. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/341003 (дата обращения: 28.11.2025).	учебное пособие	2023	ЭБС «Лань»
4.	Баланов, А. Н.	Машинное обучение и искусственный интеллект : учебное пособие для вузов / А. Н. Баланов. — 2-е изд., стер. — Санкт-Петербург : Лань, 2025. — 172 с. — ISBN 978-5-507-52891-2. — Текст : электронный // Лань : электронно-библиотечная	Учебное пособие	2025	ЭБС «Лань»

№ п/п	Авторы, составители	Заглавие (заголовок)	Тип (учебник, учебное пособие, учебно- методическое пособие, практикум, др.)	Год издания	Количество в научной библиотеке / Наименование ЭБС
		система. — URL: https://e.lanbook.com/book/462248 (дата обращения: 28.11.2025).			
5.	Маран, М. М.	Маран, М. М. Программная инженерия : Учебное пособие для вузов / М. М. Маран. — 3-е изд., стер. — Санкт-Петербург : Лань, 2022. — 196 с. — ISBN 978-5-8114-9323-4. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/189470 (дата обращения: 28.11.2025).	Учебное пособие	2022	ЭБС «Лань»

8.2. Дополнительная литература

№ п/п	Авторы, составители	Заглавие (заголовок)	Тип (учебник, учебное пособие, учебно- методическое пособие, практикум, др.)	Год издания	Количество в научной библиотеке / Наименование ЭБС
1.	Бубнов, А. А.	Бубнов, А. А. Тестирование программного обеспечения : учебное пособие / А. А. Бубнов, С. А. Бубнов, В. В. Тишкина. — Рязань : РГРТУ, 2024. — 164 с. — ISBN 978-5-7722-0421-4. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/494540 (дата обращения: 28.11.2025).	учебное пособие	2024	ЭБС «IPRbooks»
2.	Турнецкая, Е. Л.	Турнецкая, Е. Л. Программная инженерия. Тестирование и контроль	учебное пособие	2025	ЭБС «IPRbooks»

№ п/п	Авторы, составители	Заглавие (заголовок)	Тип (учебник, учебное пособие, учебно- методическое пособие, практикум, др.)	Год издания	Количество в научной библиотеке / Наименование ЭБС
		качества программного обеспечения : учебное пособие для вузов / Е. Л. Турнецкая, А. В. Аграновский. — Санкт- Петербург : Лань, 2025. — 172 с. — ISBN 978-5-507-51677-3. — Текст : электронный // Лань : электронно- библиотечная система. — URL: https://e.lanbook.com/book/455672 (дата обращения: 28.11.2025).			
3.	Романов, Е. Л.	Романов, Е. Л. Программная инженерия : учебное пособие / Е. Л. Романов. — Новосибирск : НГТУ, 2017. — 395 с. — ISBN 978-5-7782-3455-0. — Текст : электронный // Лань : электронно- библиотечная система. — URL: https://e.lanbook.com/book/118221 (дата обращения: 28.11.2025).	учебное пособие	2017	ЭБС «IPRbooks»

8.3. Перечень профессиональных баз данных и информационных справочных систем

№ пп	Наименование	Ссылка
1	Springer Nature (Полнотекстовая коллекция журналов)	https://www.springernature.com/gp/products
2	Springer eBooks (Полнотекстовая коллекция электронных книг издательства Springer Nature)	https://link.springer.com/
3	«Кодекс»	https://kodeks.ru/
4	ELIBRARY.RU (электронная библиотека научных публикаций)	http://elibrary.ru
5	"Гарант"	https://www.garant.ru/
6	"КонсультантПлюс"	https://www.consultant.ru/
7	Техэксперт	https://cntd.ru/

8.4. Перечень программного обеспечения

№ п/п	Наименование ПО	Реквизиты договора (дата, номер, срок действия)
1	Python — язык программирования для анализа данных и ML	PSF License (Python Software Foundation License (свободное и бесплатное ПО))
2	Scikit-learn — классические алгоритмы ML	BSD 3-Clause License (свободное и бесплатное ПО)
3	Pandas — обработка табличных данных и временных рядов	BSD 3-Clause License (свободное и бесплатное ПО)
4	NumPy — численные вычисления	BSD License (свободное и бесплатное ПО)
5	Matplotlib / Seaborn — визуализация данных	Matplotlib — PSF/MIT; Seaborn — BSD
6	PyTorch — нейронные сети	BSD 3-Clause License (свободное и бесплатное ПО)
7	TensorFlow/Keras — ML и нейронные сети	Apache License 2.0 (свободное и бесплатное ПО)
8	MLflow — управление экспериментами, MLOps	Apache License 2.0 (свободное и бесплатное ПО)
9	FastAPI — создание REST-сервисов	MIT License (свободное и бесплатное ПО)
10	Flask — минималистичный веб-фреймворк	BSD License (свободное и бесплатное ПО)
11	Docker Desktop (Community Edition) — контейнеризация	Бесплатная лицензия для образовательных учреждений (Docker Personal / CE)
12	Docker Engine — серверная часть контейнеризации	Apache License 2.0 (свободное и бесплатное ПО)
13	Git — система контроля версий	GPL v2
14	GitHub — хостинг репозитория (free tier)	Бесплатная лицензия GitHub Free
15	VS Code — редактор кода	MIT License (свободное и бесплатное ПО)

8.5. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

№ п/п	Наименование оборудованных учебных кабинетов, лабораторий, мастерских и др. объектов для проведения практических и лабораторных занятий, помещений для самостоятельной работы обучающихся (номер аудитории)	Перечень основного оборудования
1	Компьютерный класс. Учебная аудитория для проведения занятий лекционного типа. Учебная аудитория для проведения занятий семинарского типа. Учебная аудитория для проведения лабораторных работ. Учебная аудитория для курсового проектирования (выполнения курсовых работ). Учебная аудитория для проведения групповых и индивидуальных консультаций. Учебная аудитория для проведения занятий текущего контроля и промежуточной аттестации. (ИТП-211)	Компьютер (монитор DEXP DF27N1, системный блок Intel Core i7-12700, 2100МГц, 16 Гб RAM, UHD Graphics 770), столы ученические, стол преподавательский, стулья, акустическая панель.
2	Помещение для самостоятельной работы обучающихся (УЛК-105).	Стол, стулья, стеллажи (в т.ч. выставочные) с книгами, компьютеры, мобильные рабочие места.
3	Помещение для самостоятельной работы обучающихся (УЛК-406).	Стол компьютерный, стулья, микрокомпьютеры raspberry pi 32 bit.