Образовательное частное учреждение высшего образования
«МОСКОВСКИЙ УНИВЕРСИТЕТ ИМЕНИ А.С. ГРИБОЕДОВА»
(ИМПЭ им. А.С. Грибоедова)

ИНСТИТУТ МЕЖДУНАРОДНОЙ ЭКОНОМИКИ, ЛИДЕРСТВА И МЕНЕДЖМЕНТА

УТВЕРЖДАЮ
Директор института
международной экономики,
лидерства и менеджмента
_____ А.А. Панарин
«23» декабря 2024 г.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

по учебной дисциплине:

СИСТЕМЫ АНАЛИЗА ДАННЫХ

Направление подготовки
09.03.03 Прикладная информатика
(уровень бакалавриат)

Направленность (профиль):
«Анализ данных»

Форма обучения: очная, заочная

Москва

Фонд оценочных средств для дисциплины «Системы анализа данных». Направление подготовки/специальность 09.03.03 Прикладная информатика, направленность (профиль/специализация): Анализ Данных/ О.С. Зеленская – М.: ИМПЭ им. А.С. Грибоедова.

Фонд оценочных средств является неотъемлемой частью рабочей программы дисциплины.

Разработчик: О.С. Зеленская

Заведующий кафедрой: _____/ Н. Н. Загускин

1. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ, ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ

Настоящий Фонд оценочных средств (ФОС) по дисциплине «Системы анализа данных» является неотъемлемым приложением к рабочей программе дисциплины (РПД) «Системы анализа данных». На данный ФОС распространяются все реквизиты утверждения, представленные в РПД по данной дисциплине.

2. ПЕРЕЧЕНЬ ОЦЕНОЧНЫХ СРЕДСТВ

Для определения качества освоения обучающимися учебного материала по дисциплине используются следующие оценочные средства:

№ п/п	Оценочное средство	Краткая характеристика оценочного средства	Представление оценочного средства в ФОС
1	Тестирование	Вид контроля, позволяющий оценить изученный теоретический материал.	Вопросы для проведения тестирования
2	Практические задания	Вид контроля, позволяющий оценить умение обучающегося применять осваиваемую компетенцию в практических ситуациях и при решении производственных задач	Задания к практическому занятию
3	Контрольная работа	Вид контроля, позволяющий определить результат освоения компетенций по дисциплине в рамках рассматриваемой темы, оцениваемый с помощью соответствующих индикаторов достижения компетенций	Задания контрольной работы
4	Самостоятельная работа	Вид контроля, позволяющий оценить проработку теоретического материала, изучение рекомендуемой литературы, выполнение практико-ориентированных заданий (заполнение таблиц, проведение сравнительного анализа, составление схем и др.), решение практических задач, создание презентаций, написание рефератов, подборку нормативного и иного материала и выполнение других заданий	Задания самостоятельной работы
5	Курсовая работа	Вид контроля, позволяющий выявить степень владения базовыми знаниями, умениями и навыками, необходимыми для обучения, и определить уровень владения новым материалом	Индивидуальные задания (темы) для курсовой работы
6	Зачет/Зачет с оценкой/Экзамен	Вид контроля, позволяющий выявить степень овладения знаниями, умениями и навыками, необходимыми для дальнейшего освоения образовательной программы подготовки	Вопросы для подготовки к зачету/зачету с оценкой/экзамену

3. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ ДИСЦИПЛИНЕ

3.1. Сопроводительная информация.

Разработчик	О.С. Зеленская
Кафедра	Прикладной информатики и информационных технологий
Наименованиедисциплины	Системы анализа данных
Факультет / институт	Институт международной экономики, лидерства и менеджмента
Направлениеподготовки / специальность	09.03.03 Прикладная информатика
Количество вопросоввоценочных заданиях (диапазон)	От 15 до 30
Общее время тестирования(мин)	90
Общееколичествовопросов/заданийвФОС	30
Размещенность на сайте Университета примерного перечня вопросов, заданий ФОС – для подготовки обучающихся к прохождению оценки (да / нет)	Да

3.2. Характеристика оцениваемых компетенций.

Код компетенции	Формулировка компетенции	Индикаторы достижения компетенции
ПК-5	Способен осуществлять проектирование структур данных	**ИПК-5.1 Знать:** Методы и средства проектирования компьютерного программного обеспечения **ИПК-5.2 Уметь:** Применять методы и средства проектирования компьютерного программного обеспечения, структур данных, баз данных, программных интерфейсов
ОПК-6	Способен анализировать и разрабатывать организационно-технические и экономические процессы с применением методов системного анализа и математического моделирования	**ИОПК-6.1.** Знать основы теории систем и системного анализа, дискретной математики, теории вероятностей и математической статистики, методов оптимизации и исследования операций, нечётких вычислений, математического и имитационного моделирования на базовом уровне **ИОПК-6.2.** Уметь применять методы теории систем и системного анализа, математического, статистического и имитационного моделирования для автоматизации задач принятия решений, анализа информационных потоков, расчёта экономической эффективности и надёжности информационных систем технологий на базовом уровне

4. СОДЕРЖАНИЕ ОЦЕНОЧНЫХ СРЕДСТВ ТЕКУЩЕГО КОНТРОЛЯ

4.1. ТИПОВЫЕ ТЕСТОВЫЕ ЗАДАНИЯ.

Тесты содержат набор вопросов, в полном объеме охватывающие изученный теоретический материал по указанной теме. Выполнение тестов позволяет определить результат освоения компетенций по дисциплине в рамках рассматриваемой темы, оцениваемый с помощью соответствующих индикаторов достижения компетенций. Индивидуальный тестовый сеанс для каждого обучающегося формируется по специальному алгоритму, обеспечивающему заданную тематическую структуру и пропорциональное наличие вопросов разного типа и сложности.

При формировании тестов необходимо использовать задания следующих типов:

Тип задания 1. Задания закрытого типа на установление соответствия.

Тип задания 2. Задания закрытого типа на установление последовательности.

Тип задания 3. Задания комбинированного типа, предполагающие выбор одного правильного ответа из предложенных

Тип задания 4. Задания комбинированного типа, предполагающие выбор нескольких ответов из предложенных

Тип задания 5. Задания открытого типа с развернутым ответом.

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
Тема 1.1 Системы анализа данных	**ПК- 5** Способен осуществлять проектирование структур данных	**ИПК-5.1 Знать:** Методы и средства проектирования компьютерного программного обеспечения	**Задание 1.1** Какие из приведенных утверждений корректно раскрывают сущность системы анализа данных (САД) как целостной платформы? 1. Это любая программа, в названии которой есть слова «анализ» или «data». 2. Это интегрированный комплекс программных, аппаратных и методологических средств, поддерживающий полный жизненный цикл работы с данными. 3. Это система, основной задачей которой является только генерация красивых графиков и дашбордов. 4. Это платформа, обеспечивающая сбор, хранение, очистку, анализ, визуализацию данных и управление этими процессами. Ответ: 2, 4. Пояснение: Ответ 2 является классическим, полным определением САД, подчеркивающим ее комплексность (софт, железо, методики) и охват полного цикла. Ответ 4 конкретизирует ключевые функции, которые интегрирует такая платформа, что раскрывает ее суть как единого инструментария.

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			Задание 1.2 Какие из перечисленных функций относятся к ключевым аналитическим функциям САД в рамках классификации типов анализа? 1. Описательная аналитика (Descriptive Analytics) – ответ на вопрос «Что произошло?». 2. Нормативная аналитика (Prescriptive Analytics) – рекомендация действий для достижения цели. 3. Хостинг веб-сайтов компании. 4. Администрирование корпоративной электронной почты. Ответ: 1, 2. Пояснение: Ответ 1 описывает базовую и обязательную функцию любой САД – констатацию и визуализацию произошедшего на основе исторических данных. Ответ 2 описывает одну из самых продвинутых функций САД, направленную на оптимизацию решений. **Задание 1.3** Установите правильную последовательность этапов работы с данными, которые должна поддерживать полнофункциональная система анализа данных, в рамках типового аналитического проекта. 1. Интерпретация результатов, формулирование выводов и подготовка отчета. 2. Построение, обучение и валидация прогнозных или классификационных моделей. 3. Определение целей и постановка бизнес-задачи анализа. 4. Развертывание модели или аналитического правила в производственную среду (Deployment). 5. Загрузка, очистка и преобразование данных (ETL/ELT).

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			Ответ: 35214 Пояснение: Данная последовательность отражает логику стандартных методологий (CRISP-DM, KDD). Анализ всегда начинается с понимания задачи (3). Затем в систему попадают и подготавливаются данные (5). На подготовленных данных выполняются ключевые аналитические операции, такие как моделирование (2). Результаты модели интерпретируются для принятия решений (1). После успешной проверки и интерпретации, модель внедряется для регулярного использования (4). Этот порядок демонстрирует полный цикл, поддерживаемый САД. **Задание 1.4** Расположите архитектурные слои типичной современной системы анализа данных в порядке «от сырых данных к бизнес-решению». 1. Слой аналитической обработки и моделирования (Data Science, ML Engine). 2. Слой источников данных (операционные БД, лог-файлы, внешние API). 3. Слой представления и визуализации (BI-дашборды, отчеты). 4. Слой консолидированного хранения (Data Warehouse / Data Lake). Ответ: 2413 Пояснение: Эта последовательность описывает поток данных в архитектуре. Все начинается с источников (2). Данные из них извлекаются и загружаются в специально организованное хранилище для анализа (4). Находясь в этом хранилище, данные подвергаются углубленной аналитической обработке (1). Итоговые результаты, инсайты или предсказания модели доносятся до конечного пользователя через интерфейсы визуализации (3). Нарушение этого порядка (например, визуализация до хранения) делает систему неработоспособной.
Тема 1.2 Статистические методы библиотеки Pandas.	**ОПК-6** Способен осуществлять проектирование баз данных	**ИОПК-6.1.** Знать основы теории систем и системного анализа, дискретной	**Задание 2.1** Установите правильную последовательность шагов для проведения типового анализа с группировкой данных в Pandas: от исходного DataFrame до получения агрегированной статистики по группам. 1. Применить одну или несколько агрегирующих функций (например, .mean(), .sum()) к

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
		математики, теории вероятностей и математической статистики, методов оптимизации и исследования операций, нечётких вычислений, математического и имитационного моделирования на базовом уровне	сгруппированному объекту. 2. Прочитать данные в DataFrame (например, с помощью pd.read_csv()). 3. Выбрать столбец или несколько столбцов, по которым будет проводиться группировка. 4. Вызвать метод .groupby() на DataFrame, передав выбранные столбцы. Ответ: 2341 Пояснение: Это логический поток операций при работе с группировкой. Сначала данные должны быть загружены в объект DataFrame (2). Затем аналитик определяет признак (признаки) для группировки (3). После этого создается объект DataFrameGroupBy с помощью метода .groupby() (4). И только затем к этому «ленивому» объекту применяются агрегирующие функции для получения конкретного результата (1). Нарушение порядка (например, попытка сгруппировать до загрузки данных) невозможно. **Задание 2.2** Представьте, что вам нужно проверить гипотезу о наличии выбросов в данных. Расположите следующие шаги по увеличению информативности анализа, начиная с самого общего обзора. 1. Визуализировать распределение данных с помощью boxplot (df.boxplot()) для наглядного обнаружения выбросов. 2. Вычислить базовые описательные статистики с помощью df.describe(), чтобы увидеть минимум, максимум и квартили. 3. Вычислить стандартное отклонение (df.std()) и, возможно, интерквартильный размах (IQR) вручную для количественной оценки разброса. 4. Отфильтровать потенциальные выбросы, используя условие на основе IQR (например, Q1 - 1.5*IQR, Q3 + 1.5*IQR). Ответ: 2314 Пояснение: Последовательность отражает переход от общего к частному и от числового анализа к визуальному и активным действиям. Сначала получаем общую числовую картину (describe()) (2). Затем углубляемся в конкретные меры разброса (std, расчет IQR) (3). После этого переходим к визуальной проверке (boxplot), которая подтверждает численные расчеты

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			(1). И, наконец, на основе проведенного анализа выполняем фильтрацию данных (4). Такой порядок является методологически верным. **Задание 2.3** Установите соответствие между методами агрегации/статистики в Pandas и результатом, который они возвращают при применении к числовому столбцу DataFrame df['price']. Список 1 (Метод): 1. df['price'].count() 2. df['price'].median() 3. df['price'].std() 4. df['price'].quantile(0.75) Список 2 (Результат): A. Значение, ниже которого расположены 75% наблюдений (третий квартиль, Q3). B. Количество непустых (не-NaN) значений в столбце. C. Мера разброса данных вокруг среднего значения (среднеквадратическое отклонение). D. Значение, которое делит упорядоченный набор данных пополам. Ответ: 1B, 2D, 3C, 4A. **Задание 2.4** Установите соответствие между методом Pandas, используемым при группировке данных, и его типичным применением или результатом. Список 1 (Метод/Конструкция): 1. df.groupby('category')['sales'].sum() 2. df.groupby('category').agg({'sales':'mean', 'profit':'max'}) 3. df.groupby('category').size() 4. df.groupby(['category', 'month']).mean() Список 2 (Применение/Результат): A. Получение средней выручки и максимальной прибыли для каждой категории.

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			B. Подсчет количества строк (записей) в каждой категории. C. Вычисление общей суммы продаж для каждой категории. D. Создание сводной таблицы со средними значениями по всем числовым столбцам для каждой комбинации категории и месяца. Ответ: 1C, 2A, 3B, 4D.
Тема 2.1 Построение графиков и визуализация с помощью библиотеки Matplotlib.		**ИПК-5.1 Знать:** Методы и средства проектирования компьютерного программного обеспечения	**Задание 3.1** Установите соответствие между типом графика/диаграммы и основной функцией Matplotlib pyplot, используемой для его построения. Список 1 (Тип графика): 1. Линейный график 2. Столбчатая диаграмма (вертикальная) 3. Круговая диаграмма 4. Гистограмма Список 2 (Функция plt.*): A. bar() B. hist() C. pie() D. plot() Ответ: 1D, 2A, 3C, 4B. **Задание 3.2** Установите соответствие между параметром настройки в Matplotlib и типом графика, для которого он является наиболее характерным и часто используемым. Список 1 (Параметр): 1. wedgeprops (свойства секторов, например, ширина границы) 2. width (ширина столбцов) 3. linestyle (стиль линии) 4. whis (длина "усов" относительно межквартильного размаха)

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			Список 2 (Тип графика): A. Линейный график (plot) B. График "ящик с усами" (boxplot) C. Круговая диаграмма (pie) D. Столбчатая диаграмма (bar/barh) Ответ: 1C, 2D, 3A, 4B. **Задание 3.3** Вы хотите построить сгруппированную столбчатую диаграмму для сравнения двух показателей (А и Б) по трем категориям (X, Y, Z). Какой из следующих подходов к подготовке данных и вызову функции plt.bar() является концептуально верным? 1. Однократный вызов plt.bar() с передачей всех 6 значений в один список. 2. Два последовательных вызова plt.bar() для показателя А и показателя Б со смещением параметра x (позиции столбцов) для каждого вызова. 3. Вызов plt.pie() с параметром labels, содержащим 6 значений. 4. Использование plt.boxplot() и передача двух списков по три элемента в каждый. Ответ: 2. Пояснение: Классический способ построения сгруппированных столбцов — рисовать серии столбцов рядом. Для этого каждая серия (показатель) рисуется своим вызовом bar(), с расчетом позиций (x) так, чтобы столбцы разных серий для одной категории стояли рядом. **Задание 3.4** При построении круговой диаграммы с помощью plt.pie() какой параметр отвечает за подписи к каждому сектору (например, названия категорий)? 1. titles 2. labels

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			3. legend 4. sizes Ответ: 2. Пояснение: Параметр labels функции plt.pie() принимает список строк для подписей секторов. Ответ 1: Параметра titles не существует, заголовок всей диаграммы задается plt.title(). Ответ 3: legend — это функция для создания легенды, а не параметр pie. Ответ 4: sizes — это не параметр, а фактические числовые данные для диаграммы, которые передаются первым аргументом в pie(sizes, ...).
Тема 2.2 Построение графиков и визуализация с помощью библиотеки Seaborn.		**ИОПК-6.1.** Знать основы теории систем и системного анализа, дискретной математики, теории вероятностей и математической статистики, методов оптимизации и исследования операций, нечётких вычислений, математического и имитационного моделирования	**Задание 4.1** Вы хотите построить точечную диаграмму (scatter plot) в Seaborn, где цвет точек будет зависеть от значений третьей, категориальной переменной 'type', а размер точек — от значений четвертой, числовой переменной 'importance'. Какой из следующих вызовов функции является синтаксически и концептуально правильным? Предположим, что df — это pandas DataFrame. 1. sns.scatterplot(data=df, x='var1', y='var2', color='type', size='importance') 2. sns.scatterplot(data=df, x='var1', y='var2', hue='type', scale='importance') 3. sns.scatterplot(data=df, x='var1', y='var2', hue='type', size='importance') 4. sns.scatterplot(data=df, x='var1', y='var2', group='type', volume='importance') Ответ: 3. Пояснение: В Seaborn для кодирования категорий цветом используется параметр hue, а для кодирования числовой величины размером — параметр size. **Задание 4.2** При построении гистограммы с графиком плотности (KDE) с помощью sns.histplot() какой параметр непосредственно отвечает за отображение или скрытие гладкой кривой оценки плотности поверх столбцов гистограммы? 1. bins

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
		на базовом уровне	2. kde 3. stat 4. alpha Ответ: 2. Пояснение: Параметр kde принимает булево значение (kde=True или kde=False) и управляет отображением ядерной оценки плотности (Kernel Density Estimate). **Задание 4.3** Как называется функция высшего уровня (high-level function) в Seaborn, которая создает сетку точечных диаграмм (scatter plots) для попарного изучения взаимосвязей между всеми числовыми столбцами в DataFrame, а также гистограммы для каждого столбца на диагонали? Ответ: sns.pairplot() (или seaborn.pairplot). Пояснение: Функция sns.pairplot() является одной из ключевых для EDA в Seaborn. Она автоматически строит матрицу графиков (pair grid), где вне диагонали — scatter plots, а на диагонали по умолчанию — KDE (или можно задать гистограмму). Она часто является первым шагом в многомерном анализе. **Задание 4.4** Какой единственный и обязательный параметр необходимо передать в функцию sns.heatmap() для построения тепловой карты, если мы хотим визуализировать уже рассчитанную матрицу корреляций corr_matrix? Ответ: data (или позиционный первый аргумент, т.е. сама матрица corr_matrix). Пояснение: Функция sns.heatmap() ожидает первым и основным аргументом двумерный набор данных для визуализации (например, pandas DataFrame или массив NumPy). Вызов выглядит как sns.heatmap(corr_matrix, ...) или явно sns.heatmap(data=corr_matrix, ...). Все

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			остальные параметры (annot, cmap, fmt) являются необязательными для настройки отображения.
Тема 3.1 Знакомство с библиотекой Scikit-Learn.		**ИПК-5.1 Знать:** Методы и средства проектирования компьютерного программного обеспечения	**Задание 5.1** Какие из следующих утверждений верно описывают ключевые принципы и соглашения (API) библиотеки Scikit-Learn, общие для большинства её алгоритмов? 1. Все модели (оценщики, estimators) реализуют методы .fit(X, y) для обучения и .predict(X) (или .transform(X)) для применения. 2. Параметры модели задаются в конструкторе класса, а гиперпараметры настраиваются через метод .set_params() после обучения. 3. Данные для обучения (X) должны быть представлены исключительно в виде списка списков Python (list of lists), другие форматы недопустимы. 4. Разделение данных на обучающую и тестовую выборки рекомендуется выполнять с помощью функции train_test_split из модуля sklearn.model_selection. Ответ: 1, 4. Пояснение: Ответ 1: Это центральное соглашение API Scikit-Learn. Единый интерфейс fit/predict/transform — основа библиотеки. Ответ 4: Функция train_test_split — это стандартный и рекомендуемый инструмент Scikit-Learn для простого случайного разбиения данных, который следует знать на начальном этапе. **Задание 5.2** Какие из перечисленных задач относятся к области обучения с учителем (supervised learning), для решения которых в Scikit-Learn есть соответствующие классы алгоритмов? 1. Кластеризация данных на группы без заранее известных меток. 2. Прогнозирование цены дома на основе его характеристик (площадь, район и т.д.). 3. Классификация изображений рукописных цифр на классы от 0 до 9. 4. Уменьшение размерности данных для визуализации (например, с помощью PCA).

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			Ответ: 2, 3. Пояснение: Ответ 2: Это классическая задача регрессии (предсказание непрерывной величины) – подраздел обучения с учителем. Ответ 3: Это классическая задача классификации (предсказание дискретной метки) – подраздел обучения с учителем. **Задание 5.3** Как называется фундаментальный объект в Scikit-Learn, который инкапсулирует алгоритм машинного обучения и реализует стандартные методы .fit(), .predict() или .transform()? Ответ: Оценщик (Estimator). Пояснение: Термин "estimator" (оценщик) — это ключевое понятие в API Scikit-Learn. Все алгоритмы обучения (как с учителем, так и без) представлены в виде классов, которые являются оценщиками и следуют этому единому интерфейсу. Это обеспечивает согласованность и простоту использования библиотеки. **Задание 5.4** Какой основной метод используется у обученного оценщика классификации или регрессии в Scikit-Learn для того, чтобы получить предсказания целевой переменной для новых (ранее не виденных моделью) данных? Ответ: .predict() (или метод predict). Пояснение: Метод .predict() — это универсальный интерфейс для получения итоговых предсказаний модели на новых данных. Для классификации он возвращает метки классов, для регрессии — числовые значения. Это отличает его от .predict_proba() (вероятности) или .decision_function() (решающая функция).

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
Тема 3.2 Методы классификации и библиотеки Scikit-learn.		**ИОПК-6.1.** Знать основы теории систем и системного анализа, дискретной математики, теории вероятностей и математической статистики, методов оптимизации и исследования операций, нечётких вычислений, математического и имитационного моделирования на базовом уровне	**Задание 6.1** Какие из следующих утверждений верно описывают процесс и принципы обучения моделей классификации, связанные с валидацией и переобучением? 1. Тестовая выборка (test set) используется для финальной, однократной оценки обобщающей способности модели, которую не видели ни при обучении, ни при настройке гиперпараметров. 2. Валидационная выборка (validation set) часто используется в процессе настройки гиперпараметров (например, силы регуляризации C или глубины дерева max_depth). 3. Если точность модели на обучающей выборке значительно выше, чем на тестовой, это верный признак недообучения (underfitting). 4. Для надежной оценки модели при малом объеме данных лучшей альтернативой простому разделению на train/val/test является кросс-валидация (cross-validation). Ответ: 1, 2, 4. Пояснение: Ответ 1: Это строгое определение назначения тестовой выборки. Её трогать до финального этапа нельзя. Ответ 2: Это прямое назначение валидационной выборки в классическом workflow настройки гиперпараметров. Ответ 4: Кросс-валидация (например, KFold) позволяет более эффективно использовать данные для оценки и настройки модели, когда их мало. **Задание 6.2** Какие из перечисленных методов/алгоритмов в Scikit-Learn являются ансамблевыми методами (ensemble methods), основанными на комбинации нескольких базовых моделей для повышения качества и устойчивости? 1. LogisticRegression 2. RandomForestClassifier 3. DecisionTreeClassifier

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			4. GradientBoostingClassifier (из sklearn.ensemble) Ответ: 2, 4. Пояснение: Ответ 2: Случайный лес (RandomForestClassifier) — это классический ансамблевый метод, объединяющий множество решающих деревьев, построенных на бутстрап-выборках и случайных подмножествах признаков. Ответ 4: Градиентный бустинг (GradientBoostingClassifier) — это мощный ансамблевый метод, который последовательно строит деревья, каждое из которых исправляет ошибки предыдущих. **Задание 6.3** Установите правильную последовательность этапов полного цикла разработки и оценки модели классификации с настройкой гиперпараметров, начиная с исходных данных. 1. Провести финальную оценку качества лучшей модели на тестовой выборке (test set), которую модель никогда не видела. 2. Разделить исходные данные на обучающую+валидационную (train+val) и тестовую (test) части. 3. На валидационной выборке (validation set) оценить качество моделей с разными гиперпараметрами и выбрать лучшую комбинацию. 4. На обучающей выборке (train set) обучить несколько моделей классификации (например, RandomForestClassifier) с разными значениями гиперпараметров. 5. Разделить обучающую+валидационную часть на непосредственно обучающую и валидационную выборки (или использовать K-Fold CV). Ответ: 25431 Пояснение: Эта последовательность отражает строгий и корректный pipeline, предотвращающий "утечку" данных из тестового набора. Сначала мы страхуем себе чистый, нетронутый тестовый набор (2). Затем работаем с оставшимися данными: создаем

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			из них механизм для настройки (валидационную схему) (5). На обучающей части этой схемы производим обучение (4). На валидационной части — выбор лучшей модели (3). И только в самом конце, убедившись в выборе, проводим финальный беспристрастный тест (1). Нарушение этого порядка (например, настройка на тестовой выборке) делает оценку невалидной. **Задание 6.4** Расположите в логическом порядке этапы построения решающего дерева (Decision Tree) для классификации, начиная с корня. 1. Рекурсивно повторить шаги выбора признака и разделения для каждой новой подгруппы (дочерней вершины), пока не будет выполнено условие остановки (например, достигнута максимальная глубина). 2. Для текущего набора данных в узле выбрать признак и пороговое значение, которые наилучшим образом разделяют объекты по классам (например, максимизируют прирост информации information gain или уменьшают gini impurity). 3. Создать корневую вершину дерева, содержащую все объекты обучающей выборки. 4. Назначить листовой вершине метку класса, наиболее часто встречающегося среди объектов, попавших в этот лист. Ответ: 3214 Пояснение: Эта последовательность описывает жадный алгоритм построения дерева "сверху вниз". Начинаем с корня, содержащего все данные (3). В каждом узле (начиная с корня) мы ищем наилучшее правило разделения (2). Это правило применяется, создавая дочерние узлы, и процесс рекурсивно повторяется для каждого из них (1). Когда рекурсия завершается (выполнено условие остановки), в получившихся листьях определяется итоговый прогноз — модальный класс (4). Этот порядок универсален для подобных алгоритмов.
Тема 3.3 Метрики качества классификаци		**ИПК-5.1 Знать:** Методы и средства	**Задание 7.1** Установите правильную последовательность шагов для оценки бинарного классификатора с использованием ROC-кривой и метрики AUC-ROC в Scikit-Learn.

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
и в библиотеке Scikit-learn.		проектирования компьютерного программного обеспечения	1. Вычислить метрику roc_auc_score(y_true, y_pred_proba) для количественной оценки. 2. Получить предсказанные вероятности положительного класса для тестовых данных: y_pred_proba = model.predict_proba(X_test)[:, 1]. 3. Построить ROC-кривую, используя roc_curve(y_true, y_pred_proba) для получения точек и визуализировать её. 4. Обучить модель бинарной классификации (например, LogisticRegression) на тренировочных данных. 5. Проанализировать кривую: чем ближе к верхнему левому углу и чем больше AUC-ROC (ближе к 1), тем лучше модель. Ответ: 42315 Пояснение: Последовательность отражает логичный pipeline. Сначала модель должна быть обучена (4). Затем, для построения ROC-кривой, нужны не "жёсткие" предсказания (predict), а вероятности (или decision function) (2). На основе этих вероятностей и истинных метрик строятся точки кривой (3). Площадь под этой кривой (AUC) вычисляется как интегральная числовая метрика (1). Завершается процесс интерпретацией результатов (5). Критически важно, что для ROC-кривой используются именно вероятности (шаг 2). **Задание 7.2** Расположите этапы анализа качества модели при дисбалансе классов в порядке перехода от простой к более информативной и надежной оценке. 1. Посчитать Accuracy (долю правильных ответов). 2. Построить и проанализировать матрицу ошибок (confusion matrix). 3. Вычислить метрики, устойчивые к дисбалансу: Precision, Recall, F1-score. 4. Построить ROC-кривую и вычислить AUC-ROC, чтобы оценить качество модели независимо от выбранного порога классификации. Ответ: 1234 Пояснение: Эта последовательность показывает эволюцию понимания проблемы. Сначала студент может посчитать простую, но вводящую в заблуждение при дисбалансе Accuracy

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			(1). Затем, увидев распределение ошибок в матрице (2), он понимает, в каком классе проблема. Далее он использует более тонкие метрики, сфокусированные на интересующем классе (Precision/Recall) (3). И наконец, для комплексной оценки, не зависящей от произвольного порога 0.5, он применяет ROC-AUC (4). Такой порядок учит критическому мышлению при выборе метрики. **Задание 7.3** Установите соответствие между метрикой качества в Scikit-Learn и её оптимальным значением, к которому нужно стремиться (в идеальном случае). Список 1 (Метрика): 1. mean_absolute_error (MAE) 2. accuracy_score 3. r2_score (коэффициент детерминации) 4. roc_auc_score (AUC-ROC) Список 2 (Идеальное значение): A. 0 (ноль) B. 1.0 (единица) C. 1.0 (единица) D. 1.0 (единица) Ответ: 1A, 2B, 3C, 4D. **Задание 7.4** Установите соответствие между элементом матрицы ошибок (confusion matrix) для бинарной классификации и его описанием. Список 1 (Элемент матрицы / Сокращение): 1. TP (True Positive) 2. FP (False Positive) 3. FN (False Negative)

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			4. TN (True Negative) Список 2 (Описание): A. Объект отрицательного класса, который модель ошибочно предсказала как положительный (Ошибка I рода). B. Объект положительного класса, который модель верно предсказала как положительный. C. Объект положительного класса, который модель ошибочно предсказала как отрицательный (Ошибка II рода). D. Объект отрицательного класса, который модель верно предсказала как отрицательный. Ответ: 1B, 2A, 3C, 4D.
Тема 3.4 Методы кластеризации библиотеки Scikit-learn.		**ИОПК-6.1.** Знать основы теории систем и системного анализа, дискретной математики, теории вероятностей и математической статистики, методов оптимизации и исследования операций, нечётких вычислений, математического и имитационного	**Задание 8.1** Установите соответствие между методом кластеризации в Scikit-Learn и его ключевой характеристикой или ограничением. Список 1 (Метод / Алгоритм): 1. K-Means (KMeans) 2. Агломеративная кластеризация (AgglomerativeClustering) Список 2 (Ключевая характеристика): A. Плоский алгоритм, требующий указания числа кластеров заранее; чувствителен к выбросам и начальной инициализации; хорошо масштабируется. B. Иерархический алгоритм, строящий дендрограмму; позволяет выбрать число кластеров после построения; имеет высокую вычислительную сложность на больших выборках. Ответ: 1A, 2B. **Задание 8.2** Установите соответствие между метрикой качества кластеризации в Scikit-Learn и типом метрики (внутренняя/внешняя), а также принципом её работы. Список 1 (Метрика): 1. silhouette_score

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
		моделирования на базовом уровне	2. calinski_harabasz_score 3. adjusted_rand_score 4. davies_bouldin_score Список 2 (Описание): A. Внешняя метрика, сравнивающая найденные метки кластеров с истинными, с поправкой на случайное угадывание. B. Внутренняя метрика, оценивающая компактность и разделённость кластеров на основе среднего расстояния между объектами одного кластера и ближайшего другого кластера. C. Внутренняя метрика, вычисляющая отношение суммы межкластерных дисперсий к сумме внутрикластерных дисперсий. D. Внутренняя метрика, вычисляющая среднее «сходство» между каждым кластером и его наиболее похожим кластером, где сходство — это отношение внутрикластерных расстояний к расстоянию между кластерами (чем меньше значение, тем лучше). Ответ: 1B, 2C, 3A, 4D. **Задание 8.3** Почему крайне важно масштабировать (стандартизировать) числовые признаки перед применением алгоритмов кластеризации, основанных на расстояниях, таких как K-Means или агломеративная кластеризация? 1. Чтобы ускорить сходимость алгоритма. 2. Чтобы избежать ситуации, когда признак с большим размахом значений (например, зарплата) будет доминировать в расчёте расстояния и неоправданно сильно влиять на результат, по сравнению с признаком с малым размахом (например, возраст). 3. Чтобы гарантировать, что алгоритм найдёт глобальный оптимум. 4. Чтобы иметь возможность использовать категориальные признаки. Ответ: 2. Пояснение: Алгоритмы, использующие евклидово расстояние (или подобные), чувствительны к масштабу. Признак с большей дисперсией будет вносить больший вклад в

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			расстояние, искажая реальную "близость" объектов. Масштабирование (например, StandardScaler) приводит все признаки к единому масштабу, устраняя это смещение. **Задание 8.4** При построении дендрограммы для агломеративной кластеризации по оси Y обычно откладывается определённая величина. Что она чаще всего представляет? 1. Номер итерации, на которой произошло слияние двух кластеров. 2. Расстояние (или меру несходства), на котором были объединены два соответствующих кластера. 3. Среднее значение всех признаков в объединяемом кластере. 4. Индекс силуэта для данного кластера. Ответ: 2. Пояснение: Высота (по оси Y) в дендрограмме показывает расстояние между объединяемыми кластерами на момент их слияния согласно выбранной метрике расстояния (metric) и методу связи (linkage). Это позволяет визуально оценить, на каком уровне "разрезать" дендрограмму: большие скачки высоты между слияниями часто указывают на естественное число кластеров.
Тема 4.1 Построение линейных регрессионных моделей.		**ИПК-5.1 Знать:** Методы и средства проектирования компьютерного программного обеспечения	**Задание 9.1** В чем заключается ключевое различие между моделями Ridge и Lasso регрессии (обе являются регуляризованными версиями линейной регрессии)? 1. Ridge регрессия может использоваться только для бинарной классификации, а Lasso — для регрессии. 2. Lasso регрессия (L1-регуляризация) может обнулять веса неважных признаков, выполняя тем самым отбор признаков, в то время как Ridge (L2-регуляризация) лишь уменьшает их, но не до нуля. 3. Ridge регрессия всегда дает более высокое значение R^2 на тесте, чем Lasso. 4. Lasso регрессия не имеет гиперпараметра силы регуляризации, в отличие от Ridge. Ответ: 2.

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			Пояснение: Это фундаментальное практическое различие. L1-норма в Lasso приводит к разреженным решениям, что полезно для интерпретации и отбора признаков. L2-норма в Ridge лишь сжимает коэффициенты. **Задание 9.2** При построении множественной линейной регрессии в Scikit-Learn (LinearRegression) целевая функция, которую минимизирует алгоритм, — это: 1. Сумма модулей остатков (MAE). 2. Сумма квадратов остатков (RSS, Residual Sum of Squares). 3. Средняя абсолютная процентная ошибка (MAPE). 4. Коэффициент детерминации (R^2), который нужно максимизировать. Ответ: 2. Пояснение: Классический метод наименьших квадратов (OLS), реализованный в LinearRegression, минимизирует именно Residual Sum of Squares (RSS) или, что эквивалентно, Mean Squared Error (MSE). **Задание 9.3** Если все признаки в модели множественной линейной регрессии были стандартизированы (приведены к нулевому среднему и единичной дисперсии с помощью StandardScaler), то как можно интерпретировать свободный член (intercept) модели? Ответ: Это предсказанное значение целевой переменной при средних значениях всех признаков (поскольку после стандартизации среднее каждого признака равно 0). Пояснение: После стандартизации точка, где все X = 0, соответствует центру данных (средним значениям всех исходных признаков). Поэтому intercept (w0) представляет собой базовый уровень целевой переменной для "среднего" объекта. Это ключевой момент для осмысленной интерпретации модели. **Задание 9.4**

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			Как называется метрика, которая является квадратным корнем из средней квадратичной ошибки (MSE) и, в отличие от MSE, измеряется в тех же единицах, что и целевая переменная, что делает её более удобной для интерпретации? Ответ: RMSE (Root Mean Squared Error) или Среднеквадратичная ошибка (СКО). Пояснение: RMSE = sqrt(MSE). Поскольку ошибки возводились в квадрат (в MSE), извлечение корня возвращает метрику к исходной размерности (например, рубли, метры, часы). Это самая распространенная метрика для итоговой оценки регрессии "в единицах измерения".
Тема 4.2 Анализ и прогнозирование временных рядов.		**ИПК-5.1 Знать:** Методы и средства проектирования компьютерного программного обеспечения **ИОПК-6.1.** Знать основы теории систем и системного анализа, дискретной математики, теории вероятностей и математической статистики, методов оптимизации и	**Задание 10.1** При исследовательском анализе временного ряда (EDA) перед построением моделей используются автокорреляционные функции. Какие из следующих утверждений о коррелограмме (графике ACF и PACF) являются верными? 1. ACF (Autocorrelation Function) показывает чистую корреляцию между наблюдениями ряда с лагом k, без учета влияния промежуточных лагов. 2. PACF (Partial Autocorrelation Function) показывает корреляцию между наблюдениями с лагом k, исключив влияние наблюдений с лагами от 1 до k-1. 3. Медленное (экспоненциальное или синусоидальное) затухание ACF до нуля может указывать на нестационарность ряда или наличие тренда. 4. Резкий обрыв (срез) PACF после лага p является индикатором порядка q для модели MA(q). Ответ: 2, 3. Пояснение: Ответ 2: Это точное определение частной автокорреляционной функции (PACF). Ответ 3: Медленное затухание ACF — классический признак нестационарности (например, наличия единичного корня). Для стационарного ряда ACF обычно быстро стремится к нулю. **Задание 10.2**

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
		исследования операций, нечётких вычислений, математического и имитационного моделирования на базовом уровне	В рамках методологии Box-Jenkins (ARIMA) выбор порядка модели (p,d,q) является ключевым. Какие из следующих шагов/методов являются стандартными и корректными для этой цели? 1. Использование расширенного теста Дики-Фуллера (ADF test) для проверки стационарности и определения порядка интегрирования (d). 2. Визуальный анализ графиков ACF и PACF стационарного ряда для выявления потенциальных порядков p и q. 3. Автоматический подбор параметров (p,d,q) с использованием критерия информативности, такого как AIC (Akaike Information Criterion), с помощью функции auto_arima. 4. Проверка остатков обученной модели на автокорреляцию с помощью теста Льюнга-Бокса (Ljung-Box test) для оценки адекватности модели. Ответ: 1, 2, 3, 4. Пояснение: Ответ 1: Тест Дики-Фуллера — стандартный статистический тест для определения необходимости взятия разностей (определение d). Ответ 2: Анализ ACF/PACF — это классический, хотя и субъективный, метод идентификации моделей AR и MA. Ответ 3: Использование auto_arima (из библиотеки pmdarima) или аналогичных методов, минимизирующих AIC/BIC, — это современный и распространенный автоматизированный подход. Ответ 4: Проверка остатков на "белошумность" — обязательный этап диагностики модели. Если остатки автокоррелированы, модель неадекватна. Все утверждения верно описывают части стандартного рабочего процесса ARIMA. **Задание 10.3** Если p-value расширенного теста Дики-Фуллера (ADF test) для исходного временного ряда составил 0.82, а после взятия первых разностей (d=1) p-value стал равен 0.03 (при уровне значимости $\alpha=0.05$), какой порядок интегрирования (d) следует выбрать для построения ARIMA-модели?

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			Ответ: d=1. Пояснение: Высокий p-value (0.82 > 0.05) для исходного ряда означает, что мы не можем отвергнуть нулевую гипотезу о нестационарности (H0). Ряд нестационарен. После взятия первых разностей p-value (0.03) становится меньше α (0.05), что позволяет отвергнуть H0 и принять альтернативу о стационарности преобразованного ряда. Следовательно, для получения стационарного ряда достаточно одного дифференцирования: d=1. Это практическое применение теста. **Задание 10.4** Как называется статистический тест, который применяется после построения ARIMA-модели для проверки гипотезы о том, что остатки модели являются "белым шумом" (не автокоррелированы)? Этот тест обычно проверяет автокорреляцию остатков на нескольких первых лагах одновременно. Ответ: Тест Льюнга-Бокса (Ljung-Box test). Пояснение: Это стандартный диагностический тест в анализе временных рядов. Нулевая гипотеза H0: остатки независимы (отсутствует автокорреляция до выбранного лага). Если p-value теста мал (например, < 0.05), мы отвергаем H0, что означает наличие значимой автокорреляции в остатках и, следовательно, неадекватность модели. Успешное прохождение этого теста — важный критерий качества модели.

Критерии оценивания тестового задания:

Оценка	Критерии оценивания
отлично	от 90 до 100 % правильно выполненных заданий
хорошо	от 70 до 89 % правильно выполненных заданий
удовлетворительно	от 50 до 69 % правильно выполненных заданий
неудовлетворительно	менее 50 % правильно выполненных заданий

4.2 ТИПОВЫЕ ПРАКТИЧЕСКИЕ ЗАДАНИЯ

Практические задания должны отражать умение обучающегося применять осваиваемую компетенцию в практических ситуациях и при решении производственных задач.

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Задания к практическому занятию

Раздел №1 «Общая характеристика систем анализа данных. Обработка данных с помощью библиотеки Pandas» Тема 1.1 Системы анализа данных Тема 1.2 Статистические методы библиотеки Pandas.	ПК-5 Способен осуществлять проектирование структур данных	ИПК-5.2 Уметь: Применять методы и средства проектирования компьютерного программного обеспечения, структур данных, баз данных, программных интерфейсов	**Задание №1**
			Вам предоставлен фрагмент данных о транзакциях интернет-магазина. Выполните полный цикл первичного исследования и предобработки:
			1. Загрузите данные и изучите общую информацию (размер, типы, пропуски).
			2. Удалите полностью пустые строки и дубликаты.
			3. Заполните пропуски: в числовом столбце order_total - медианой, в категориальном payment_method - значением 'Unknown'.
			4. Преобразуйте тип данных в столбце order_date в формат datetime.
			5. Сформулируйте выводы о качестве данных и проведенных преобразованиях.
			
			Ответ:
			`import pandas as pd`
			`import numpy as np`
			`from datetime import datetime`
			
			`# 1. Создание и загрузка синтетических данных`
			`np.random.seed(42)`
			`data = {`
			`'order_id': [1001, 1002, 1003, 1004, 1005, 1006, 1007],`
			`'customer_id': [201, 202, 203, 204, 202, 205, 206],`
			`'order_date': ['2023-10-01', '2023-10-02', None, '2023-10-04', '2023-10-02', '2023-10-05', '2023-10-06'],`
			`'order_total': [150.0, 89.99, 200.0, None, 89.99, 120.5, 75.0],`
			`'payment_method': ['Card', 'PayPal', 'Card', 'Card', 'PayPal', None, 'Card'],`
			`'region': ['North', 'South', 'East', 'West', 'South', 'North', 'East']`
			`}`
			`df = pd.DataFrame(data)`
			
			`print("1. ИСХОДНЫЕ ДАННЫЕ И ОБЩАЯ ИНФОРМАЦИЯ:")`
			`print(df)`

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Задания к практическому занятию
			print(f"\nРазмер данных: {df.shape}") print("\nТипы данных:") print(df.dtypes) print("\nПропуски по столбцам:") print(df.isnull().sum()) print("\nКоличество дубликатов:", df.duplicated().sum()) # 2. Удаление полностью пустых строк и дубликатов df_cleaned = df.dropna(how='all') df_cleaned = df_cleaned.drop_duplicates() print(f"\n2. После удаления дубликатов: {df_cleaned.shape[0]} строк") # 3. Заполнение пропусков # Числовой столбец - медианой if df_cleaned['order_total'].isnull().any(): median_val = df_cleaned['order_total'].median() df_cleaned['order_total'].fillna(median_val, inplace=True) print(f"Заполнены пропуски в 'order_total' медианой: {median_val:.2f}") # Категориальный столбец - значением 'Unknown' df_cleaned['payment_method'].fillna('Unknown', inplace=True) # 4. Преобразование типа данных df_cleaned['order_date'] = pd.to_datetime(df_cleaned['order_date'], errors='coerce') print("\n4. Преобразован тип 'order_date' в datetime") # 5. Выводы print("\n" + "="*60) print("5. ИТОГОВЫЕ ВЫВОДЫ:") print("="*60) print(f"1. Исходный набор содержал {df.shape[0]} строк, {df.shape[1]} столбцов.") print(f"2. Обнаружено пропусков: order_date - {df['order_date'].isnull().sum()}, order_total - {df['order_total'].isnull().sum()}, payment_method - {df['payment_method'].isnull().sum()}.") print(f"3. Удалено дубликатов: {df.shape[0] - df_cleaned.shape[0]}.") print(f"4. После предобработки датасет содержит {df_cleaned.shape[0]} уникальных записей.") print(f"5. Все пропуски заполнены, типы данных корректны.") print(f"\nИтоговый датасет:") print(df_cleaned) print(f"\nТипы данных после обработки:") print(df_cleaned.dtypes)

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Задания к практическому занятию

			Задание №2
			На основе предобработанных данных о продажах выполните статистический анализ:
			1. Вычислите общую выручку, средний чек и количество заказов по каждому региону.
			2. Найдите регион с максимальной и минимальной средней выручкой на один заказ.
			3. Проведите корреляционный анализ между customer_id и order_total.
			4. Сгруппируйте данные по payment_method и вычислите для каждой категории среднюю сумму заказа и количество транзакций. Сформулируйте выводы.
			
			Ответ:
			`import pandas as pd`
			`import numpy as np`
			
			`# Используем очищенный датафрейм df_cleaned из предыдущей работы`
			`# Создадим расширенный набор для анализа`
			`np.random.seed(42)`
			`data = {`
			`'order_id': range(1001, 1021),`
			`'customer_id': np.random.randint(200, 220, 20),`
			`'order_total': np.random.uniform(50, 500, 20).round(2),`
			`'payment_method': np.random.choice(['Card', 'PayPal', 'Apple Pay'], 20),`
			`'region': np.random.choice(['North', 'South', 'East', 'West'], 20)`
			`}`
			`df = pd.DataFrame(data)`
			
			`print("СТАТИСТИЧЕСКИЙ АНАЛИЗ ПРОДАЖ ПО РЕГИОНАМ")`
			`print("="*50)`
			`print("Исходные данные (первые 5 строк):")`
			`print(df.head())`
			
			`# 1. Анализ по регионам`

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Задания к практическому занятию

			print("\n1. СТАТИСТИКА ПО РЕГИОНАМ:")
			region_stats = df.groupby('region').agg(
			  total_revenue=('order_total', 'sum'),
			  avg_check=('order_total', 'mean'),
			  orders_count=('order_id', 'count')
			).round(2)
			print(region_stats)
			
			# 2. Регион с максимальной и минимальной средней выручкой
			max_region = region_stats['avg_check'].idxmax()
			min_region = region_stats['avg_check'].idxmin()
			print(f"\n2. Регион с максимальным средним чеком: {max_region} ({region_stats.loc[max_region, 'avg_check']:.2f})")
			print(f" Регион с минимальным средним чеком: {min_region} ({region_stats.loc[min_region, 'avg_check']:.2f})")
			
			# 3. Корреляционный анализ
			correlation = df['customer_id'].corr(df['order_total'])
			print(f"\n3. Корреляция между customer_id и order_total: {correlation:.4f}")
			print(" Интерпретация: практически отсутствует линейная связь" if abs(correlation) < 0.3 else "Есть заметная линейная связь")
			
			# 4. Группировка по способам оплаты
			print("\n4. СТАТИСТИКА ПО СПОСОБАМ ОПЛАТЫ:")
			payment_stats = df.groupby('payment_method').agg(
			  avg_order=('order_total', 'mean'),
			  transaction_count=('order_id', 'count')
			).round(2)
			print(payment_stats)
			
			# Выводы
			print("\n" + "="*50)
			print("ВЫВОДЫ:")
			print("="*50)
			print("1. Наибольшая выручка наблюдается в регионе", region_stats['total_revenue'].idxmax())
			print("2. Самый высокий средний чек в регионе", max_region)
			print("3. Корреляция между ID клиента и суммой заказа практически отсутствует (неинформативный признак).")
			print("4. Наиболее популярный способ оплаты:", payment_stats['transaction_count'].idxmax())
			print(f"5. Всего проанализировано {len(df)} транзакций на общую сумму {df['order_total'].sum():.2f}")

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Задания к практическому занятию

Раздел №3 «Методы машинного обучения библиотеки Scikit-Learn»
Тема 3.1 Знакомство с библиотекой Scikit-Learn.
Тема 3.2 Методы классификации библиотеки Scikit-learn.
Тема 3.3 Метрики качества классификации в библиотеке Scikit-learn.
Тема 3.4 Методы кластеризации библиотеки Scikit-learn.
	ОПК-6 Способен анализировать и разрабатывать организационно-технические и экономические процессы с применением методов системного анализа и математического моделирования	ИОПК-6.2. Уметь применять методы теории систем и системного анализа, математического, статистического и имитационного моделирования для автоматизации задач принятия решений, анализа информационных потоков, расчёта экономической эффективности и надёжности информационных систем технологий на базовом уровне	**Задание №3**
			Постройте модель для прогнозирования оттока клиентов (churn).
			1. Разделите данные в соотношении 80/20 на обучающую и тестовую выборки с random_state=42.
			2. Обучите две модели: логистическую регрессию и дерево решений (максимальная глубина=5).
			3. На тестовой выборке оцените обе модели по метрикам accuracy, precision, recall, F1-score.
			4. Определите, какая модель показала лучший результат по F1-score.
			
			Ответ:
			`import pandas as pd`
			`import numpy as np`
			`from sklearn.model_selection import train_test_split`
			`from sklearn.linear_model import LogisticRegression`
			`from sklearn.tree import DecisionTreeClassifier`
			`from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score, classification_report`
			
			`# Создание синтетических данных банковских клиентов`
			`np.random.seed(42)`
			`n_samples = 200`
			`data = {`
			`'age': np.random.randint(18, 70, n_samples),`
			`'balance': np.random.exponential(50000, n_samples).round(2),`
			`'credit_score': np.random.normal(650, 100, n_samples).round(),`
			`'products_count': np.random.randint(1, 5, n_samples),`
			`'churn': np.random.choice([0, 1], n_samples, p=[0.7, 0.3])`
			`}`
			`# Искусственная зависимость для реализма`
			`data['churn'] = ((data['age'] > 60) | (data['balance'] < 10000) | (data['credit_score'] < 600)).astype(int)`
			`df = pd.DataFrame(data)`
			
			`print("ДАННЫЕ ДЛЯ КЛАССИФИКАЦИИ ОТТОКА КЛИЕНТОВ")`
			`print("="*50)`
			`print(f"Размер датасета: {df.shape}")`
			`print(f"Классовый баланс:\n{df['churn'].value_counts()}")`

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Задания к практическому занятию

			print(f"Доля оттока: {df['churn'].mean():.2%}")
			
			# 1. Разделение данных
			X = df[['age', 'balance', 'credit_score', 'products_count']]
			y = df['churn']
			X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42, stratify=y)
			print(f"\n1. Разделение данных: train {X_train.shape[0]} / test {X_test.shape[0]}")
			
			# 2. Обучение моделей
			print("\n2. ОБУЧЕНИЕ МОДЕЛЕЙ:")
			# Логистическая регрессия
			lr_model = LogisticRegression(random_state=42, max_iter=1000)
			lr_model.fit(X_train, y_train)
			print(" Логистическая регрессия обучена")
			
			# Дерево решений
			dt_model = DecisionTreeClassifier(max_depth=5, random_state=42)
			dt_model.fit(X_train, y_train)
			print(" Дерево решений обучено (max_depth=5)")
			
			# 3. Предсказание и оценка
			y_pred_lr = lr_model.predict(X_test)
			y_pred_dt = dt_model.predict(X_test)
			
			print("\n3. МЕТРИКИ КАЧЕСТВА НА ТЕСТОВОЙ ВЫБОРКЕ:")
			print("-"*50)
			
			def print_metrics(y_true, y_pred, model_name):
			  acc = accuracy_score(y_true, y_pred)
			  prec = precision_score(y_true, y_pred)
			  rec = recall_score(y_true, y_pred)
			  f1 = f1_score(y_true, y_pred)
			  print(f"{model_name}:")
			  print(f" Accuracy: {acc:.3f}")
			  print(f" Precision: {prec:.3f}")
			  print(f" Recall: {rec:.3f}")
			  print(f" F1-score: {f1:.3f}")
			  return f1
			
			f1_lr = print_metrics(y_test, y_pred_lr, "Логистическая регрессия")
			print()

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Задания к практическому занятию

			f1_dt = print_metrics(y_test, y_pred_dt, "Дерево решений")
			
			# 4. Сравнение моделей
			print("\n4. СРАВНЕНИЕ МОДЕЛЕЙ:")
			print("-"*50)
			if f1_lr > f1_dt:
			  print(f"Логистическая регрессия показала лучший F1-score: {f1_lr:.3f} против {f1_dt:.3f}")
			elif f1_dt > f1_lr:
			  print(f"Дерево решений показало лучший F1-score: {f1_dt:.3f} против {f1_lr:.3f}")
			else:
			  print("Модели показали одинаковый F1-score")
			
			print("\nДетальный отчет по логистической регрессии:")
			print(classification_report(y_test, y_pred_lr, target_names=['Не уйдет', 'Уйдет']))
			
			Задание №4
			Выполните кластеризацию клиентов по признакам balance и credit_score.
			1. Масштабируйте данные с помощью StandardScaler.
			2. Примените алгоритм K-means с k=3 и агломеративную кластеризацию с параметрами: n_clusters=3, linkage='ward'.
			3. Оцените качество кластеризации с помощью силуэтного коэффициента для обоих методов.
			4. Визуализируйте результаты кластеризации на scatter plot. Сделайте вывод о лучшем методе.
			
			Ответ:
			import pandas as pd
			import numpy as np
			import matplotlib.pyplot as plt
			from sklearn.preprocessing import StandardScaler
			from sklearn.cluster import KMeans, AgglomerativeClustering
			from sklearn.metrics import silhouette_score
			
			# Создание синтетических данных клиентов

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Задания к практическому занятию

			np.random.seed(42)
			n_samples = 150
			# Создаем 3 искусственных кластера
			cluster1 = np.random.multivariate_normal([50000, 750], [[20000, 0], [0, 50]], n_samples//3)
			cluster2 = np.random.multivariate_normal([150000, 650], [[30000, 0], [0, 40]], n_samples//3)
			cluster3 = np.random.multivariate_normal([250000, 550], [[40000, 0], [0, 30]], n_samples//3)
			
			data = np.vstack([cluster1, cluster2, cluster3])
			df = pd.DataFrame(data, columns=['balance', 'credit_score'])
			print(f"Данные для кластеризации: {df.shape[0]} клиентов")
			
			# 1. Масштабирование данных
			scaler = StandardScaler()
			X_scaled = scaler.fit_transform(df)
			print("1. Данные масштабированы (StandardScaler)")
			
			# 2. Применение алгоритмов кластеризации
			# K-means
			kmeans = KMeans(n_clusters=3, random_state=42, n_init=10)
			kmeans_labels = kmeans.fit_predict(X_scaled)
			print("2. K-means кластеризация выполнена")
			
			# Агломеративная кластеризация
			agglo = AgglomerativeClustering(n_clusters=3, linkage='ward')
			agglo_labels = agglo.fit_predict(X_scaled)
			print(" Агломеративная кластеризация выполнена (linkage='ward')")
			
			# 3. Оценка качества
			kmeans_silhouette = silhouette_score(X_scaled, kmeans_labels)
			agglo_silhouette = silhouette_score(X_scaled, agglo_labels)
			
			print("\n3. ОЦЕНКА КАЧЕСТВА КЛАСТЕРИЗАЦИИ:")
			print(f" K-means силуэтный коэффициент: {kmeans_silhouette:.3f}")
			print(f" Агломеративная силуэтный коэффициент: {agglo_silhouette:.3f}")
			
			# 4. Визуализация
			fig, axes = plt.subplots(1, 3, figsize=(15, 4))
			
			# Исходные данные
			axes[0].scatter(df['balance'], df['credit_score'], alpha=0.6, edgecolor='k')
			axes[0].set_title('Исходные данные')

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Задания к практическому занятию

			axes[0].set_xlabel('Баланс')
			axes[0].set_ylabel('Кредитный рейтинг')
			
			# K-means
			scatter1 = axes[1].scatter(df['balance'], df['credit_score'], c=kmeans_labels, cmap='viridis', alpha=0.6, edgecolor='k')
			axes[1].scatter(kmeans.cluster_centers_[:, 0]*scaler.scale_[0] + scaler.mean_[0],
			kmeans.cluster_centers_[:, 1]*scaler.scale_[1] + scaler.mean_[1],
			s=200, marker='X', c='red', label='Центроиды')
			axes[1].set_title(f'K-means (Silhouette: {kmeans_silhouette:.3f})')
			axes[1].set_xlabel('Баланс')
			axes[1].legend()
			
			# Agglomerative
			scatter2 = axes[2].scatter(df['balance'], df['credit_score'], c=agglo_labels, cmap='plasma', alpha=0.6, edgecolor='k')
			axes[2].set_title(f'Agglomerative (Silhouette: {agglo_silhouette:.3f})')
			axes[2].set_xlabel('Баланс')
			
			plt.tight_layout()
			plt.show()
			
			# 5. Выводы
			print("\n" + "="*50)
			print("ВЫВОДЫ ПО КЛАСТЕРИЗАЦИИ:")
			print("="*50)
			print(f"1. K-means выделил {len(np.unique(kmeans_labels))} кластера.")
			print(f"2. Агломеративная кластеризация выделила {len(np.unique(agglo_labels))} кластера.")
			if kmeans_silhouette > agglo_silhouette:
			print(f"3. K-means показал лучшее качество (силуэтный коэффициент {kmeans_silhouette:.3f} > {agglo_silhouette:.3f})")
			elif agglo_silhouette > kmeans_silhouette:
			print(f"3. Агломеративная кластеризация показала лучшее качество (силуэтный коэффициент {agglo_silhouette:.3f} > {kmeans_silhouette:.3f})")
			else:
			print(f"3. Оба метода показали одинаковое качество")
			print("4. Визуально кластеры хорошо разделяются по балансу и кредитному рейтингу.")

Раздел №4 «Методы статистического	ПК-5 Способен осуществлять	ИПК-5.2 Уметь: Применять методы и средства проектирования	**Задание№ 5** Постройте модель линейной регрессии для прогнозирования цены дома.

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Задания к практическому занятию

анализа библиотеки StatsModels» Тема 4.1 Построение линейных регрессионных моделей. Тема 4.2 Анализ и прогнозирование временных рядов.	проектирование структур данных	компьютерного программного обеспечения, структур данных, баз данных, программных интерфейсов	1. 1) Проведите предобработку: удалите выбросы в цене (верхние 5%), создайте признак area_per_room.
			2. 2) Выполните корреляционный анализ и отберите признаки с корреляцией с целевой переменной > 0.3.
			3. 3) Постройте модель линейной регрессии, разделив данные в соотношении 80/20.
			4. 4) Оцените модель с помощью R^2 и RMSE. Проинтерпретируйте коэффициенты модели.
			
			Ответ:
			`import pandas as pd`
			`import numpy as np`
			`import matplotlib.pyplot as plt`
			`from sklearn.model_selection import train_test_split`
			`from sklearn.linear_model import LinearRegression`
			`from sklearn.metrics import r2_score, mean_squared_error`
			
			`# Создание синтетических данных о недвижимости`
			`np.random.seed(42)`
			`n_samples = 200`
			`data = {`
			`'area': np.random.uniform(50, 200, n_samples),`
			`'rooms': np.random.randint(1, 6, n_samples),`
			`'floor': np.random.randint(1, 25, n_samples),`
			`'metro_distance': np.random.exponential(2, n_samples).round(1),`
			`'price': 0 # Будем вычислять`
			`}`
			`df = pd.DataFrame(data)`
			
			`# Генерация целевой переменной с зависимостью от признаков`
			`df['price'] = (df['area'] * 1000 +`
			`df['rooms'] * 50000 -`
			`df['metro_distance'] * 10000 +`
			`np.random.normal(0, 50000, n_samples))`
			
			`print("ДАННЫЕ О НЕДВИЖИМОСТИ")`
			`print("="*50)`
			`print(f"Размер датасета: {df.shape}")`

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Задания к практическому занятию

			print("\nПервые 5 строк:")
			print(df.head())
			
			# 1. Предобработка
			# Удаление выбросов в цене (верхние 5%)
			price_threshold = df['price'].quantile(0.95)
			df_clean = df[df['price'] <= price_threshold].copy()
			print(f"\n1. ПРЕДОБРАБОТКА:")
			print(f" Удалено выбросов: {len(df) - len(df_clean)} записей (верхние 5% по цене)")
			
			# Создание нового признака
			df_clean['area_per_room'] = df_clean['area'] / df_clean['rooms']
			print(f" Создан новый признак 'area_per_room'")
			
			# 2. Корреляционный анализ
			correlation_matrix = df_clean.corr()
			price_corr = correlation_matrix['price'].sort_values(ascending=False)
			print("\n2. КОРРЕЛЯЦИЯ С ЦЕЛЕВОЙ ПЕРЕМЕННОЙ 'price':")
			for feature, corr in price_corr.items():
			  print(f" {feature}: {corr:.3f}")
			
			# Отбор признаков с корреляцией > 0.3 (исключая саму цену)
			selected_features = price_corr[price_corr > 0.3].index.tolist()
			selected_features.remove('price') if 'price' in selected_features else None
			print(f"\n Отобранные признаки (corr > 0.3): {selected_features}")
			
			# 3. Построение модели
			X = df_clean[selected_features]
			y = df_clean['price']
			X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
			
			model = LinearRegression()
			model.fit(X_train, y_train)
			print(f"\n3. МОДЕЛЬ ЛИНЕЙНОЙ РЕГРЕССИИ ПОСТРОЕНА")
			print(f" Обучающая выборка: {X_train.shape[0]} записей")
			print(f" Тестовая выборка: {X_test.shape[0]} записей")
			
			# 4. Оценка модели
			y_pred = model.predict(X_test)
			r2 = r2_score(y_test, y_pred)
			rmse = np.sqrt(mean_squared_error(y_test, y_pred))

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Задания к практическому занятию

			print("\n4. ОЦЕНКА КАЧЕСТВА МОДЕЛИ:")
			print(f" R² (коэффициент детерминации): {r2:.3f}")
			print(f" RMSE (среднеквадратичная ошибка): {rmse:.0f} у.е.")
			print(f" Средняя цена в тестовой выборке: {y_test.mean():.0f} у.е.")
			print(f" Относительная ошибка (RMSE/Средняя цена): {rmse/y_test.mean():.1%}")
			
			# Интерпретация коэффициентов
			print("\nИНТЕРПРЕТАЦИЯ КОЭФФИЦИЕНТОВ:")
			for feature, coef in zip(selected_features, model.coef_):
			print(f" {feature}: {coef:.1f} (при увеличении на 1 единицу, цена изменяется на {coef:.1f} у.е.)")
			print(f" Intercept (базовая цена): {model.intercept_:.1f} у.е.")
			
			# Визуализация
			plt.figure(figsize=(10, 4))
			
			plt.subplot(1, 2, 1)
			plt.scatter(y_test, y_pred, alpha=0.6)
			plt.plot([y_test.min(), y_test.max()], [y_test.min(), y_test.max()], 'r--', lw=2)
			plt.xlabel('Фактическая цена')
			plt.ylabel('Предсказанная цена')
			plt.title(f'Предсказание vs Факт (R² = {r2:.3f})')
			
			plt.subplot(1, 2, 2)
			residuals = y_test - y_pred
			plt.scatter(y_pred, residuals, alpha=0.6)
			plt.axhline(y=0, color='r', linestyle='--')
			plt.xlabel('Предсказанная цена')
			plt.ylabel('Остатки')
			plt.title('Анализ остатков')
			
			plt.tight_layout()
			plt.show()
			
			Задание №6
			Проанализируйте временной ряд месячных продаж.
			1. Проверьте ряд на стационарность с помощью теста Дики-Фуллера.

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Задания к практическому занятию

			2. Если ряд нестационарен, приведите его к стационарному виду с помощью сезонного дифференцирования (лаг=12).
			3. Постройте модель ARIMA с параметрами (1,1,1) и сделайте прогноз на 3 месяца вперед.
			4. Визуализируйте исходный ряд, прогноз и доверительные интервалы.
			
			Ответ:
			import pandas as pd
			import numpy as np
			import matplotlib.pyplot as plt
			from statsmodels.tsa.stattools import adfuller
			from statsmodels.tsa.arima.model import ARIMA
			from statsmodels.graphics.tsaplots import plot_acf, plot_pacf
			import warnings
			warnings.filterwarnings('ignore')
			
			# Создание синтетического временного ряда с трендом и сезонностью
			np.random.seed(42)
			n_periods = 60 # 5 лет по месяцам
			time_index = pd.date_range(start='2018-01-01', periods=n_periods, freq='M')
			
			# Компоненты ряда: тренд + сезонность + шум
			trend = np.linspace(100, 300, n_periods)
			seasonality = 50 * np.sin(2 * np.pi * np.arange(n_periods) / 12)
			noise = np.random.normal(0, 15, n_periods)
			
			sales = trend + seasonality + noise
			df = pd.DataFrame({'sales': sales}, index=time_index)
			
			print("АНАЛИЗ ВРЕМЕННОГО РЯДА ПРОДАЖ")
			print("="*50)
			print(f"Период: {df.index[0].date()} - {df.index[-1].date()}")
			print(f"Длина ряда: {len(df)} месяцев")
			print("\nПервые 12 значений:")
			print(df.head(12))
			
			# 1. Проверка на стационарность

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Задания к практическому занятию

			print("\n1. ТЕСТ ДИКИ-ФУЛЛЕРА (ПРОВЕРКА СТАЦИОНАРНОСТИ):")
			adf_result = adfuller(df['sales'])
			print(f" ADF статистика: {adf_result[0]:.3f}")
			print(f" p-value: {adf_result[1]:.3f}")
			print(f" Критические значения:")
			for key, value in adf_result[4].items():
			  print(f" {key}: {value:.3f}")
			
			if adf_result[1] > 0.05:
			  print(" Вывод: Ряд НЕ стационарен (p-value > 0.05)")
			else:
			  print(" Вывод: Ряд стационарен")
			
			# 2. Приведение к стационарному виду (сезонное дифференцирование)
			df_diff = df['sales'].diff(12).dropna() # Сезонное дифференцирование (лаг=12)
			adf_result_diff = adfuller

Критерии оценивания практических занятий:

Оценка	Критерии оценивания
отлично	Выставляется, если обучающийся умеет увязывать теорию с практикой (решает задачи, формулирует выводы, умеет пояснить полученные результаты), владеет понятийным аппаратом, полно и глубоко овладел материалом по заданной теме, обосновывает свои суждения и даёт правильные ответы на вопросы преподавателя
хорошо	Выставляется, если обучающийся умеет увязывать теорию с практикой (решает задачи и формулирует выводы, умеет пояснить полученные результаты), владеет понятийным аппаратом, полно и глубоко овладел материалом по заданной теме, но содержание ответов имеют некоторые неточности и требуют уточнения и комментария со стороны преподавателя
удовлетворительно	Выставляется, если обучающийся знает и понимает материал по заданной теме, но изложение неполное, непоследовательное, допускаются неточности в определении понятий, студент не может обосновать свои ответы на уточняющие вопросы преподавателя
неудовлетворительно	Выставляется, если обучающийся допускает ошибки в определении понятий, искажающие их смысл, беспорядочно и неуверенно излагает материал. Делает ошибки в ответах на уточняющие вопросы преподавателя

4.3. ТИПОВЫЕ ЗАДАНИЯ ДЛЯ КОНТРОЛЬНЫХ РАБОТ

Контрольные работы содержат несколько практических заданий по индивидуальным вариантам, в полном объеме охватывающих изученный материал по указанной теме. Выполнение контрольных работ позволяет определить результат освоения компетенций по дисциплине в рамках рассматриваемой темы, оцениваемый с помощью соответствующих индикаторов достижения компетенций.

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Типовой вариант контрольной работы
Тема 3.1 Знакомство с библиотекой Scikit-Learn.	**ПК- 5** Способен осуществлять проектирование структур данных **ПК-6** Способен осуществлять	**ИПК-5.2 Уметь:** Применять методы и средства проектирования компьютерного программного обеспечения, структур данных, баз данных, программных интерфейсов **ИОПК-6.1.**	**Задание 5.1** Какие из следующих утверждений верно описывают ключевые принципы и соглашения (API) библиотеки Scikit-Learn, общие для большинства её алгоритмов? 1. Все модели (оценщики, estimators) реализуют методы .fit(X, y) для обучения и .predict(X) (или .transform(X)) для применения. 2. Параметры модели задаются в конструкторе класса, а гиперпараметры настраиваются через метод .set_params() после обучения. 3. Данные для обучения (X) должны быть представлены исключительно в виде списка списков Python (list of lists), другие форматы недопустимы.

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Типовой вариант контрольной работы
	проектирование баз данных	Знать основы теории систем и системного анализа, дискретной математики, теории вероятностей и математической статистики, методов оптимизации и исследования операций, нечётких вычислений, математического и имитационного моделирования на базовом уровне **ИОПК-6.2.** Уметь применять методы теории систем и системного анализа, математического, статистического и имитационного моделирования для автоматизации задач принятия решений, анализа информационных потоков, расчёта экономической эффективности и надёжности информационных систем технологий на базовом уровне	4. Разделение данных на обучающую и тестовую выборки рекомендуется выполнять с помощью функции train_test_split из модуля sklearn.model_selection. Ответ: 1, 4. Пояснение: Ответ 1: Это центральное соглашение API Scikit-Learn. Единый интерфейс fit/predict/transform — основа библиотеки. Ответ 4: Функция train_test_split — это стандартный и рекомендуемый инструмент Scikit-Learn для простого случайного разбиения данных, который следует знать на начальном этапе. **Задание 5.2** Какие из перечисленных задач относятся к области обучения с учителем (supervised learning), для решения которых в Scikit-Learn есть соответствующие классы алгоритмов? 1. Кластеризация данных на группы без заранее известных меток. 2. Прогнозирование цены дома на основе его характеристик (площадь, район и т.д.). 3. Классификация изображений рукописных цифр на классы от 0 до 9. 4. Уменьшение размерности данных для визуализации (например, с помощью PCA). Ответ: 2, 3. Пояснение: Ответ 2: Это классическая задача регрессии (предсказание непрерывной величины) – подраздел обучения с учителем. Ответ 3: Это классическая задача классификации (предсказание дискретной метки) – подраздел обучения с учителем.

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Типовой вариант контрольной работы
			Задание 5.3 Как называется фундаментальный объект в Scikit-Learn, который инкапсулирует алгоритм машинного обучения и реализует стандартные методы .fit(), .predict() или .transform()? Ответ: Оценщик (Estimator). Пояснение: Термин "estimator" (оценщик) — это ключевое понятие в API Scikit-Learn. Все алгоритмы обучения (как с учителем, так и без) представлены в виде классов, которые являются оценщиками и следуют этому единому интерфейсу. Это обеспечивает согласованность и простоту использования библиотеки. **Задание 5.4** Какой основной метод используется у обученного оценщика классификации или регрессии в Scikit-Learn для того, чтобы получить предсказания целевой переменной для новых (ранее не виденных моделью) данных? Ответ: .predict() (или метод predict). Пояснение: Метод .predict() — это универсальный интерфейс для получения итоговых предсказаний модели на новых данных. Для классификации он возвращает метки классов, для регрессии — числовые значения. Это отличает его от .predict_proba() (вероятности) или .decision_function() (решающая функция). **Задание 6.1** Какие из следующих утверждений верно описывают процесс и принципы обучения моделей классификации, связанные с валидацией и переобучением? 1. Тестовая выборка (test set) используется для финальной, однократной оценки обобщающей способности модели, которую не видели ни при обучении, ни при настройке гиперпараметров. 2. Валидационная выборка (validation set) часто используется в процессе настройки гиперпараметров (например, силы регуляризации C или глубины дерева max_depth).

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Типовой вариант контрольной работы
			3. Если точность модели на обучающей выборке значительно выше, чем на тестовой, это верный признак недообучения (underfitting). 4. Для надежной оценки модели при малом объеме данных лучшей альтернативой простому разделению на train/val/test является кросс-валидация (cross-validation). Ответ: 1, 2, 4. Пояснение: Ответ 1: Это строгое определение назначения тестовой выборки. Её трогать до финального этапа нельзя. Ответ 2: Это прямое назначение валидационной выборки в классическом workflow настройки гиперпараметров. Ответ 4: Кросс-валидация (например, KFold) позволяет более эффективно использовать данные для оценки и настройки модели, когда их мало. **Задание 6.2** Какие из перечисленных методов/алгоритмов в Scikit-Learn являются ансамблевыми методами (ensemble methods), основанными на комбинации нескольких базовых моделей для повышения качества и устойчивости? 1. LogisticRegression 2. RandomForestClassifier 3. DecisionTreeClassifier 4. GradientBoostingClassifier (из sklearn.ensemble) Ответ: 2, 4. Пояснение: Ответ 2: Случайный лес (RandomForestClassifier) — это классический ансамблевый метод, объединяющий множество решающих деревьев, построенных на бутстрап-выборках и случайных подмножествах признаков.

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Типовой вариант контрольной работы
			Ответ 4: Градиентный бустинг (GradientBoostingClassifier) — это мощный ансамблевый метод, который последовательно строит деревья, каждое из которых исправляет ошибки предыдущих. **Задание 6.3** Установите правильную последовательность этапов полного цикла разработки и оценки модели классификации с настройкой гиперпараметров, начиная с исходных данных. 1. Провести финальную оценку качества лучшей модели на тестовой выборке (test set), которую модель никогда не видела. 2. Разделить исходные данные на обучающую+валидационную (train+val) и тестовую (test) части. 3. На валидационной выборке (validation set) оценить качество моделей с разными гиперпараметрами и выбрать лучшую комбинацию. 4. На обучающей выборке (train set) обучить несколько моделей классификации (например, RandomForestClassifier) с разными значениями гиперпараметров. 5. Разделить обучающую+валидационную часть на непосредственно обучающую и валидационную выборки (или использовать K-Fold CV). Ответ: 25431 Пояснение: Эта последовательность отражает строгий и корректный pipeline, предотвращающий "утечку" данных из тестового набора. Сначала мы страхуем себе чистый, нетронутый тестовый набор (2). Затем работаем с оставшимися данными: создаем из них механизм для настройки (валидационную схему) (5). На обучающей части этой схемы производим обучение (4). На валидационной части — выбор лучшей модели (3). И только в самом конце, убедившись в выборе, проводим финальный беспристрастный тест (1). Нарушение этого порядка (например, настройка на тестовой выборке) делает оценку невалидной.

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Типовой вариант контрольной работы
			Задание 6.4 Расположите в логическом порядке этапы построения решающего дерева (Decision Tree) для классификации, начиная с корня. 1. Рекурсивно повторить шаги выбора признака и разделения для каждой новой подгруппы (дочерней вершины), пока не будет выполнено условие остановки (например, достигнута максимальная глубина). 2. Для текущего набора данных в узле выбрать признак и пороговое значение, которые наилучшим образом разделяют объекты по классам (например, максимизируют прирост информации information gain или уменьшают gini impurity). 3. Создать корневую вершину дерева, содержащую все объекты обучающей выборки. 4. Назначить листовой вершине метку класса, наиболее часто встречающегося среди объектов, попавших в этот лист. Ответ: 3214 Пояснение: Эта последовательность описывает жадный алгоритм построения дерева "сверху вниз". Начинаем с корня, содержащего все данные (3). В каждом узле (начиная с корня) мы ищем наилучшее правило разделения (2). Это правило применяется, создавая дочерние узлы, и процесс рекурсивно повторяется для каждого из них (1). Когда рекурсия завершается (выполнено условие остановки), в получившихся листьях определяется итоговый прогноз — модальный класс (4). Этот порядок универсален для подобных алгоритмов.
Тема 4.2 Анализ и прогнозирование временных рядов.	**ПК- 5** Способен осуществлять проектирование структур данных	**ИПК-5.2 Уметь:** Применять методы и средства проектирования компьютерного программного обеспечения, структур данных,	**Задание 10.1** При исследовательском анализе временного ряда (EDA) перед построением моделей используются автокорреляционные функции. Какие из следующих утверждений о коррелограмме (графике ACF и PACF) являются верными? 1. ACF (Autocorrelation Function) показывает чистую корреляцию между наблюдениями ряда с лагом k, без учета влияния промежуточных лагов.

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Типовой вариант контрольной работы
	ПК-6 Способен осуществлять проектирование баз данных	баз данных, программных интерфейсов **ИОПК-6.2.** Уметь применять методы теории систем и системного анализа, математического, статистического и имитационного моделирования для автоматизации задач принятия решений, анализа информационных потоков, расчёта экономической эффективности и надёжности информационных систем технологий на базовом уровне	2. PACF (Partial Autocorrelation Function) показывает корреляцию между наблюдениями с лагом k, исключив влияние наблюдений с лагами от 1 до k-1. 3. Медленное (экспоненциальное или синусоидальное) затухание ACF до нуля может указывать на нестационарность ряда или наличие тренда. 4. Резкий обрыв (срез) PACF после лага p является индикатором порядка q для модели MA(q). Ответ: 2, 3. Пояснение: Ответ 2: Это точное определение частной автокорреляционной функции (PACF). Ответ 3: Медленное затухание ACF — классический признак нестационарности (например, наличия единичного корня). Для стационарного ряда ACF обычно быстро стремится к нулю. **Задание 10.2** В рамках методологии Box-Jenkins (ARIMA) выбор порядка модели (p,d,q) является ключевым. Какие из следующих шагов/методов являются стандартными и корректными для этой цели? 1. Использование расширенного теста Дики-Фуллера (ADF test) для проверки стационарности и определения порядка интегрирования (d). 2. Визуальный анализ графиков ACF и PACF стационарного ряда для выявления потенциальных порядков p и q. 3. Автоматический подбор параметров (p,d,q) с использованием критерия информативности, такого как AIC (Akaike Information Criterion), с помощью функции auto_arima. 4. Проверка остатков обученной модели на автокорреляцию с помощью теста Льюнга-Бокса (Ljung-Box test) для оценки адекватности модели. Ответ: 1, 2, 3, 4.

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Типовой вариант контрольной работы
			Пояснение: Ответ 1: Тест Дики-Фуллера — стандартный статистический тест для определения необходимости взятия разностей (определение d). Ответ 2: Анализ ACF/PACF — это классический, хотя и субъективный, метод идентификации моделей AR и MA. Ответ 3: Использование auto_arima (из библиотеки pmdarima) или аналогичных методов, минимизирующих AIC/BIC, — это современный и распространенный автоматизированный подход. Ответ 4: Проверка остатков на "белошумность" — обязательный этап диагностики модели. Если остатки автокоррелированы, модель неадекватна. Все утверждения верно описывают части стандартного рабочего процесса ARIMA. **Задание 10.3** Если p-value расширенного теста Дики-Фуллера (ADF test) для исходного временного ряда составил 0.82, а после взятия первых разностей (d=1) p-value стал равен 0.03 (при уровне значимости α=0.05), какой порядок интегрирования (d) следует выбрать для построения ARIMA-модели? Ответ: d=1. Пояснение: Высокий p-value (0.82 > 0.05) для исходного ряда означает, что мы не можем отвергнуть нулевую гипотезу о нестационарности (H0). Ряд нестационарен. После взятия первых разностей p-value (0.03) становится меньше α (0.05), что позволяет отвергнуть H0 и принять альтернативу о стационарности преобразованного ряда. Следовательно, для получения стационарного ряда достаточно одного дифференцирования: d=1. Это практическое применение теста. **Задание 10.4** Как называется статистический тест, который применяется после построения ARIMA-модели для проверки гипотезы о том, что остатки модели являются "белым шумом" (не автокоррелированы)? Этот тест обычно проверяет автокорреляцию

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Типовой вариант контрольной работы
			остатков на нескольких первых лагах одновременно. Ответ: Тест Льюнга-Бокса (Ljung-Box test). Пояснение: Это стандартный диагностический тест в анализе временных рядов. Нулевая гипотеза H0: остатки независимы (отсутствует автокорреляция до выбранного лага). Если p-value теста мал (например, < 0.05), мы отвергаем H0, что означает наличие значимой автокорреляции в остатках и, следовательно, неадекватность модели. Успешное прохождение этого теста — важный критерий качества модели.

Критерии оценивания контрольной работы:

Оценка	Критерии оценивания
отлично	Выставляется, если обучающийся умеет увязывать теорию с практикой (решает задачи, формулирует выводы, умеет пояснить полученные результаты), владеет понятийным аппаратом, полно и глубоко овладел материалом по заданной теме, обосновывает свои суждения и даёт правильные ответы на вопросы преподавателя
хорошо	Выставляется, если обучающийся умеет увязывать теорию с практикой (решает задачи и формулирует выводы, умеет пояснить полученные результаты), владеет понятийным аппаратом, полно и глубоко овладел материалом по заданной теме, но содержание ответов имеют некоторые неточности и требуют уточнения и комментария со стороны преподавателя.
удовлетворительно	Выставляется, если обучающийся знает и понимает материал по заданной теме, но изложение неполное, непоследовательное, допускаются неточности в определении понятий, студент не может обосновать свои ответы на уточняющие вопросы преподавателя
неудовлетворительно	Выставляется, если обучающийся допускает ошибки в определении понятий, искажающие их смысл, беспорядочно и неуверенно излагает материал. Делает ошибки в ответах на уточняющие вопросы преподавателя

4.4. ТИПОВЫЕ ЗАДАНИЯ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ

Самостоятельная работа включает в себя проработку теоретического материала, изучение рекомендуемой литературы, выполнение практико-ориентированных заданий (заполнение таблиц, проведение сравнительного анализа, составление схем и др.), решение практических задач, создание презентаций, написание рефератов, подборка нормативного и иного материала и выполнение других заданий.

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Типовое задание для самостоятельной работы
Тема 2.2 Построение графиков и визуализация с помощью библиотеки Seaborn.	**ПК- 5** Способен осуществлять проектирование структур данных **ПК-6** Способен осуществлять проектирование баз данных	**ИПК-5.1 Знать:** Методы и средства проектирования компьютерного программного обеспечения **ИПК-5.2 Уметь:** Применять методы и средства проектирования компьютерного программного обеспечения, структур данных, баз данных, программных интерфейсов **ИОПК-6.1.** Знать основы теории систем и системного анализа, дискретной математики, теории вероятностей и математической	Подготовить реферат на тему “Построение 3D-диаграммы рассеивания” Что может быть отражено в реферате: Seaborn — библиотека для создания статистических графиков на Python, построена на основе библиотеки Matplotlib и тесно интегрирована со структурами данных из Pandas. С её помощью можно создавать диаграммы рассеяния, которые отображают точки на двумерной оси, чтобы показать взаимосвязь между двумя переменными. Теоретические основы Функция sns.scatterplot(). Принимает набор данных, который нужно визуализировать, и столбцы, которые будут выступать как оси x и y. Возможность настраивать элементы диаграмм. Например, можно изменить цвет и размер каждой точки на графике. Семантическое отображение. Можно показать взаимосвязь между x и y для разных подмножеств данных с помощью параметров hue, size и style. Автоматическое добавление меток осей и условных обозначений. Функции Seaborn автоматически добавляют информативные метки осей и условные обозначения, которые объясняют семантические отображения на графике. Пример кода Создание базовой точечной диаграммы с помощью набора данных, например, ФМРТ. Можно нанести момент времени на ось x, а сигнал — на ось y, чтобы наблюдать, как сигнал меняется со временем. Настройка диаграммы. Например, можно задать оттенок к регионам — это означает, что данные по каждому из них будут раскрашены по-разному. Также можно задать пропорции точек в зависимости от уровня свободы — чем больше значение аргумента size, тем крупнее точка на диаграмме. Создание сетки графиков с помощью функции FacetGrid. Это позволяет визуализировать данные для разных категорий в разных подграфиках. Рекомендации по оформлению Указать цель визуализации — например, показать взаимосвязь между несколькими переменными и выявить скрытые паттерны в данных.

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Типовое задание для самостоятельной работы
		статистики, методов оптимизации и исследования операций, нечётких вычислений, математического и имитационного моделирования на базовом уровне **ИОПК-6.2.** Уметь применять методы теории систем и системного анализа, математического, статистического и имитационного моделирования для автоматизации задач принятия решений, анализа информационных потоков, расчёта экономической эффективности и	Привести примеры — например, диаграмму рассеяния, где на оси X расположена одна переменная, на оси Y — другая, а размер точек зависит от третьей переменной. Настроить внешний вид графиков — можно использовать встроенные функции Seaborn для изменения стиля, цветов и шрифтов. Например, для создания минималистичного графика можно использовать стиль whitegrid, который добавляет сетку на фон. **Вариант 2** Подготовить реферат на тему “Скрипичная диаграмма. ” Что может быть отражено в реферате: Violin Plot — гибридный инструмент визуализации, сочетающий элементы box plot (ящик с усами) и kernel density plot (график плотности вероятности). Название происходит от характерной формы графика, напоминающей скрипку или виолончель. sky.pro Основная ценность Violin Plot — возможность одновременно отображать: полное распределение данных (через кривые плотности вероятности); медиану и квартили (через внутренние элементы «ящика»); минимумы и максимумы (через «усы»); выбросы (через отдельные точки); сравнительную плотность вероятности на разных участках распределения (через ширину «скрипки»). Принцип построения В Seaborn для построения скрипичных диаграмм используется функция sns.violinplot(). Она позволяет: Отображать распределение количественных данных после группировки по одной или более категориальным переменным. Использовать оценку плотности ядра для отображения распределения. Настраивать внешний вид диаграммы: регулировать цветовую палитру, добавлять метки, изменять стили линий.

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Типовое задание для самостоятельной работы
		надёжности информационных систем технологий на базовом уровне	Параметры Некоторые параметры функции sns.violinplot(): data — набор данных для построения графика. x — категориальная ось. y — числовая переменная для отображения на графике. hue — параметр для создания сгруппированной скрипичной диаграммы. split — позволяет разделить скрипки, чтобы сравнить распределение подгрупп внутри категорий. inner — параметр для определения внутренних точек скрипичной диаграммы. orient — параметр для определения ориентации графика (может быть горизонтальным или вертикальным). Примеры В реферате можно рассмотреть, например: Базовый скрипичный график с использованием встроенного набора данных tips из Seaborn. Например, показать распределение счетов по дням недели. Скрипичный график с дополнительными настройками. Например, разделить скрипку для сравнения по гендеру, показать внутри скрипки диаграмму размаха. Скрипичный график с статистическими элементами. Например, показать отдельные наблюдения, ограничить диапазон скрипки диапазоном данных, масштабировать ширину скрипок.
Тема 4.1 Построение линейных регрессионных моделей.	**ПК- 5** Способен осуществлять проектирование структур данных **ПК-6** Способен осуществлять	**ИПК-5.1 Знать:** Методы и средства проектирования компьютерного программного обеспечения **ИПК-5.2 Уметь:** Применять методы и средства	Самостоятельно изучите тему “Построение регрессии с категориальными переменными.” и подготовьте краткий доклад по изученному материалу Теоретические основы Понятие категориальной регрессии. Она подходит, когда цель анализа — предсказание зависимой переменной из набора независимых (предикторных) переменных. Категориальные переменные могут быть перекодированы как индикаторные или обработаны тем же способом, что и переменные интервального уровня. Процедура подготовки данных. Категориальные предикторы не можно помещать в регрессию без специальной подготовки — параметры модели невозможно

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Типовое задание для самостоятельной работы
	проектирование баз данных	проектирования компьютерного программного обеспечения, структур данных, баз данных, программных интерфейсов **ИОПК-6.1.** Знать основы теории систем и системного анализа, дискретной математики, теории вероятностей и математической статистики, методов оптимизации и исследования операций, нечётких вычислений, математического и имитационного моделирования на базовом уровне	проинтерпретировать. Процедура подготовки — разложение каждого категориального предиктора на ряд дихотомических («фиктивных») переменных, число которых всегда получается на единицу меньше числа значений (категорий) исходной переменной. Альтернативные методы. Например, логлинейный анализ, который работает с категориальными шкалами, не требуя преобразовывать их в фиктивные переменные. Методы Логистическая регрессия. Предсказывает вероятности появления зависимой переменной при определённых независимых признаках (предикторах), выраженных в категориальных переменных. Мультиномиальная логистическая регрессия — подходит, если зависимая переменная категориальная, но имеет более двух категорий. Порядковая регрессия — используется, когда зависимые переменные относятся к порядковой шкале. Регрессия с оптимальным масштабированием — для каждой переменной назначаются значения масштаба таким образом, чтобы они были оптимальными для регрессии. Сочетания этих уровней могут объяснить широкий диапазон нелинейных взаимосвязей, для которых ни один «стандартный» метод по отдельности не подходит. Ошибки Ошибка отсеивания незначимых предикторов. При таком подходе есть риск упустить из виду присутствие в системе предикторов многомерных связей и взаимодействий. Поскольку последние нельзя разложить на ряд парных связей, проверка предикторов на мультиколлинеарность может их не выявить. Ошибка создания фиктивных переменных. Некоторые методологические аргументы против этой процедуры: создание фиктивных переменных занижает величину парной связи и уменьшает статистическую мощность выборки для выявления истинного эффекта независимой переменной в генеральной совокупности. Ошибка неуместного объединения данных. Регрессионные модели объединяют данные из разных выборок, которые не следует объединять.

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Типовое задание для самостоятельной работы
		ИОПК-6.2. Уметь применять методы теории систем и системного анализа, математического, статистического и имитационного моделирования для автоматизации задач принятия решений, анализа информационных потоков, расчёта экономической эффективности и надёжности информационных систем технологий на базовом уровне	

Критерии оценивания самостоятельной работы:

Оценка	Критерии оценивания
отлично	выставляется если работа носит научно-исследовательский характер, проанализирован и сделан сравнительный анализ нескольких литературных источников, приведены примеры
хорошо	выставляется если проанализирован и сделан сравнительный анализ нескольких литературных источников, приведены примеры
удовлетворительно	выставляется если проведен сравнительный анализ научно-методической литературы, приведены примеры
неудовлетворительно	выставляется если работа прошла проверку на антиплагиат и соответствует требованиям оформления

4.5. ИНДИВИДУАЛЬНЫЕ ЗАДАНИЯ ДЛЯ КУРСОВЫХ РАБОТ

Курсовые работы не предусмотрены

Курсовая работа позволяет выявить степень владения базовыми знаниями, умениями и навыками, необходимыми для обучения, и определить уровень владения новым материалом.

Примерные индивидуальные задания (темы) для курсовых работ:

Критерии оценивания курсовой работы:

Оценка	Критерии оценивания
отлично	Содержание курсовой работы полностью соответствует заданию, содержащемуся в методических указаниях, и плану. Представлены результаты структурированного и логически последовательного обзора литературных и иных источников по теме исследования. Структура курсовой работы логически и методически выдержана. Верно определены исходные данные для расчетов. Все аналитические расчеты выполнены верно, корректно применены методы экономического анализа, не нарушена методика анализа предмета исследования. Все выводы и предложения убедительно аргументированы. При защите курсовой работы обучающийся правильно и уверенно отвечает на вопросы преподавателя, демонстрирует глубокое знание теоретического материала, способен аргументировать собственные утверждения и выводы
хорошо	Содержание курсовой работы полностью соответствует заданию, содержащемуся в методических указаниях, и плану. Представлены результаты структурированного и логически последовательного обзора литературных и иных источников по теме исследования. Структура курсовой работы логически и методически выдержана. Верно определены исходные данные для расчетов. В расчетах допускаются незначительные (не искажающие общего итога оценки) погрешности/ошибки. Большинство выводов и предложений аргументировано, корректно применены методы экономического анализа, не нарушена методика анализа предмета исследования. Оформление курсовой работы и полученные результаты в целом отвечают требованиям, изложенным в методических указаниях. Имеются одна-две несущественные ошибки в использовании терминов, в построенных диаграммах и схемах, в оформлении таблиц. Наличествует незначительное количество грамматических и/или стилистических ошибок. При защите курсовой работы обучающийся правильно и уверенно отвечает на большинство вопросов преподавателя, демонстрирует хорошее знание

	теоретического материала, но не всегда способен аргументировать собственные утверждения и выводы. При наводящих вопросах преподавателя исправляет ошибки в ответе
удовлетворительно	Содержание курсовой работы полностью соответствует заданию, содержащемуся в методических указаниях, и плану. Результаты обзора литературных и иных источников представлены недостаточно полно, недостаточно логично и последовательно. Верно определены исходные данные для расчетов, но имеются грубые ошибки в расчетах. Аргументация выводов и предложений слабая или отсутствует. Экономические выводы носят констатирующий (описательный) характер. Имеются одно-два существенных отклонений от требований в оформлении курсовой работы. Полученные результаты в целом отвечают требованиям, изложенным в методических указаниях. Имеются одна-две существенных ошибки в использовании терминов, в построенных диаграммах и схемах. Много грамматических и/или стилистических ошибок. При защите курсовой работы обучающийся допускает грубые ошибки при ответах на вопросы преподавателя, демонстрирует слабое знание теоретического материала, в большинстве случаев не способен уверенно аргументировать собственные утверждения и выводы
неудовлетворительно	Содержание курсовой работы не соответствует заданию, содержащемуся в методических указаниях, и плану. Неверно определены исходные данные для расчетов, неверно и не корректно применены методы экономического анализа. Экономические выводы содержат неверную экономическую оценку. Имеются более двух существенных отклонений от требований в оформлении курсовой работы. Большое количество существенных ошибок по сути работы, много грамматических и стилистических ошибок и др. Полученные результаты не отвечают требованиям, изложенным в методических указаниях. При защите курсовой работы обучающийся демонстрирует слабое понимание программного материала, студент не может защитить свои решения, допускает грубые фактические ошибки при ответах на поставленные вопросы или вовсе не отвечает на них. Курсовая работа не представлена преподавателю. Обучающийся не явился на защиту курсовой работы

5. МАТЕРИАЛЫ ДЛЯ ОЦЕНКИ РЕЗУЛЬТАТОВ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

Промежуточная аттестация проводится в форме зачета/зачета с оценкой/экзамена.

5.1. Вопросы к зачету:

1. Что является определяющим признаком системы анализа данных в отличие от набора разрозненных инструментов?
2. Назовите две ключевые функциональные группы, которые должна обеспечивать современная система анализа данных.
3. Какова основная цель вычисления описательных статистик на этапе разведочного анализа данных (EDA)?
4. Чем принципиально отличается медиана от среднего арифметического, и в каком случае медиана является более предпочтительной мерой центра?
5. В чем основное практическое различие между коэффициентом корреляции и ковариации при оценке связи двух переменных?
6. Какова цель использования операции группировки (GROUP BY) в анализе данных? Приведите типичный пример её применения.

7. Какой параметр линейного графика в Matplotlib является ключевым для одновременного отображения нескольких линий с их идентификацией?

8. В каком случае предпочтительнее использовать столбчатую диаграмму, а в каком — круговую? Назовите по одному ключевому параметру для каждой.
9. Что отображает гистограмма и на что напрямую влияет параметр bins при её построении?
10. Какие три ключевых элемента информации о распределении данных можно быстро оценить с помощью графика «ящик с усами»?
11. Какой параметр тепловой карты (heatmap) в Seaborn является обязательным для её содержательной интерпретации при визуализации матрицы корреляций и почему?
12. Какую практическую аналитическую задачу помогает решить таблица коэффициентов корреляции предикторов с целевой переменной на этапе подготовки данных для машинного обучения?
13. Назовите параметр точечной диаграммы (scatter plot) в Seaborn, который позволяет добавить третье измерение (категориальное) к двумерному графику, и как он это делает?
14. В чем заключается главное принципиальное соглашение (API) библиотеки Scikit-learn, общее для подавляющего большинства её алгоритмов?
15. Каков ключевой критерий, на основе которого задача машинного обучения относится к обучению с учителем или без учителя?
16. Несмотря на название «регрессия», почему логистическая регрессия является алгоритмом классификации, а не регрессии?
17. Для каких целей, соответственно, используются обучающая (train), валидационная (validation) и тестовая (test) выборки?
18. Опишите простой диагностический признак, по которому можно заподозрить переобучение модели.
19. Какова основная цель добавления регуляризации (например, L1-Lasso или L2-Ridge) в функцию потерь линейной регрессии?
20. На каком принципе основан жадный алгоритм построения решающего дерева, и какую локальную задачу он решает в каждом узле?
21. За счет каких двух источников случайности достигается разнообразие деревьев в ансамбле «Случайный лес»?
22. Какое ключевое свойство MSE (средней квадратичной ошибки) отличает её от MAE (среднего модуля ошибки) и в каких случаях это может быть важно?
23. Какую содержательную информацию предоставляет коэффициент детерминации R^2, и что означает его отрицательное значение?
24. Какие две принципиально разные типовые бизнес-ситуации можно описать с помощью элементов FP (False Positive) и FN (False Negative) из матрицы ошибок?
25. В каком распространенном сценарии метрика Accuracy (доля правильных ответов) становится абсолютно бесполезной для выбора модели, и почему?
26. Какое ключевое преимущество метрики AUC-ROC (площадь под ROC-кривой) перед Accuracy или F1-score?
27. Какую уникальную возможность при выборе числа кластеров предоставляет агломеративная кластеризация по сравнению с K-Means, и с помощью какого инструмента это реализуется?
28. Каков главный содержательный недостаток алгоритма K-Means, вытекающий из того, что он минимизирует внутрикластерное расстояние до центроидов?
29. Почему для оценки качества кластеризации нельзя использовать стандартные метрики точности (accuracy), и какой тип метрик для этого применяют?

30. Какая фундаментальная проблема возникает при интерпретации коэффициентов множественной линейной регрессии, если признаки сильно коррелированы между собой?
31. Почему при оценке регрессионной модели недостаточно смотреть только на значение ошибки (например, MSE), и какая дополнительная метрика помогает это понять?
32. Какую ключевую гипотезу о временном ряде позволяет проверить визуальный анализ его автокорреляционной функции (ACF)?

33. Какую нулевую и альтернативную гипотезу проверяет расширенный тест Дики-Фуллера (ADF test) и как интерпретируется его p-value?
34. Назовите три ключевых этапа методологии Бокса-Дженкинс (ARIMA) и их основное содержание.

Критерии оценивания зачета с оценкой/экзамена:

Оценка	Критерии оценивания
зачтено	Обучающийся должен: уметь строить ответ в соответствии со структурой излагаемого вопроса; продемонстрировать прочное, достаточно полное усвоение знаний программного материала; продемонстрировать знание основных теоретических понятий; правильно формулировать определения; последовательно, грамотно и логически стройно изложить теоретический материал; продемонстрировать умения самостоятельной работы с литературой; уметь сделать достаточно обоснованные выводы по излагаемому материалу.
не зачтено	Обучающийся демонстрирует: незнание значительной части программного материала; не владение понятийным аппаратом дисциплины; существенные ошибки при изложении учебного материала; неумение строить ответ в соответствии со структурой излагаемого вопроса; неумение делать выводы по излагаемому материалу.

Критерии оценивания зачета с оценкой/экзамена:

Оценка	Критерии оценивания
отлично	Выставляется, если обучающимся правильно и полностью раскрыто содержание материала в пределах программы, чётко и правильно даны определения и раскрыто содержание понятий, точно использованы научные и технические термины, в ответе использованы ранее приобретённые теоретические знания, сделаны необходимые выводы и обобщения
хорошо	Выставляется, если обучающимся раскрыто основное содержание материала в пределах программы, даны определения и раскрыто содержание понятий, в ответе использованы ранее приобретённые теоретические знания, сделаны необходимые выводы и обобщения, но присутствуют незначительные нарушения в последовательности изложения, имеются одна-две неточности в содержании ответа
удовлетворительно	Выставляется, если обучающимся содержание учебного материала изложено фрагментарно, не всегда последовательно, не даны определения, не раскрыто содержание понятий, или они изложены с ошибками, допускаются ошибки и неточности в

	использовании научной терминологии, отсутствуют выводы и обобщения из предыдущего материала, или возможны ошибки в их изложении
неудовлетворительно	Выставляется, если обучающимся основное содержание учебного материала не раскрыто, не даются ответы на основные вопросы, допускаются грубые ошибки в определении понятий, в использовании терминологии, отсутствуют выводы и обобщения

0. МАТЕРИАЛЫ ДЛЯ ДИАГНОСТИЧЕСКОЙ РАБОТЫ

Задания для диагностической работы должны обеспечивать оценку полностью или частично сформированных компетенций. Каждое задание должно быть привязано к тому или иному индикатору сформированности компетенций.

При формировании заданий для диагностической работы необходимо использовать тестовые задания следующих типов:

Тип задания 1. Задания закрытого типа на установление соответствия.

Тип задания 2. Задания закрытого типа на установление последовательности.

Тип задания 3. Задания комбинированного типа, предполагающие выбор одного правильного ответа из предложенных

Тип задания 4. Задания комбинированного типа, предполагающие выбор нескольких ответов из предложенных

Тип задания 5. Задания открытого типа с развернутым ответом.

Все типы заданий должны быть представлены не менее одного раза.

№ п/п	Тема занятия	Код компетенции	Индикатор	Тип задания	Задание		Уровень освоения компетенции
					Вариант 1	**Вариант 2**	
1	Тема 1.1 Системы анализа данных	**ПК- 5** Способен осуществлять проектирование структур данных	**ИПК-5.1 Знать:** Методы и средства проектирования компьютерного программного обеспечения	4,4	**1.** Какие из приведенных утверждений корректно раскрывают сущность системы анализа данных (САД) как целостной платформы? 1. Это любая программа, в названии которой есть слова «анализ» или «data». 2. Это интегрированный комплекс программных, аппаратных и методологических средств, поддерживающий полный жизненный цикл работы с данными. 3. Это система, основной задачей которой является только генерация красивых графиков и дашбордов. 4. Это платформа, обеспечивающая сбор, хранение, очистку, анализ, визуализацию данных и управление этими процессами. Ответ: 2, 4. Пояснение: Ответ 2 является классическим, полным определением САД, подчеркивающим ее комплексность (софт, железо, методики) и охват полного цикла. Ответ 4 конкретизирует ключевые функции, которые интегрирует такая платформа, что раскрывает ее суть как единого инструментария.	**1.** Какие из перечисленных функций относятся к ключевым аналитическим функциям САД в рамках классификации типов анализа? 1. Описательная аналитика (Descriptive Analytics) – ответ на вопрос «Что произошло?». 2. Нормативная аналитика (Prescriptive Analytics) – рекомендация действий для достижения цели. 3. Хостинг веб-сайтов компании. 4. Администрирование корпоративной электронной почты. Ответ: 1, 2. Пояснение: Ответ 1 описывает базовую и обязательную функцию любой САД – констатацию и визуализацию произошедшего на основе исторических данных. Ответ 2 описывает одну из самых продвинутых функций САД, направленную на оптимизацию решений.	**Повышенный** **(3-5 минут)**

2	Тема 1.2 Статистические методы библиотеки Pandas.	**ОПК-6** Способен осуществлять проектирование баз данных	**ИОПК-6.1.** Знать основы теории систем и системного анализа, дискретной математики, теории вероятностей и математической статистики, методов оптимизации и исследования операций, нечётких вычислений, математического и имитационного моделирования на базовом уровне	2,2	**2.** Установите правильную последовательность шагов для проведения типового анализа с группировкой данных в Pandas: от исходного DataFrame до получения агрегированной статистики по группам. 1. Применить одну или несколько агрегирующих функций (например, .mean(), .sum()) к сгруппированному объекту. 2. Прочитать данные в DataFrame (например, с помощью pd.read_csv()). 3. Выбрать столбец или несколько столбцов, по которым будет проводиться группировка. 4. Вызвать метод .groupby() на DataFrame, передав выбранные столбцы. Ответ: 2341 Пояснение: Это логический поток операций при работе с группировкой. Сначала данные должны быть загружены в объект DataFrame (2). Затем аналитик определяет признак (признаки) для группировки (3). После этого создается объект DataFrameGroupBy с помощью метода .groupby() (4). И только затем к этому «ленивому» объекту применяются агрегирующие функции для	**2.** Представьте, что вам нужно проверить гипотезу о наличии выбросов в данных. Расположите следующие шаги по увеличению информативности анализа, начиная с самого общего обзора. 1. Визуализировать распределение данных с помощью boxplot (df.boxplot()) для наглядного обнаружения выбросов. 2. Вычислить базовые описательные статистики с помощью df.describe(), чтобы увидеть минимум, максимум и квартили. 3. Вычислить стандартное отклонение (df.std()) и, возможно, интерквартильный размах (IQR) вручную для количественной оценки разброса. 4. Отфильтровать потенциальные выбросы, используя условие на основе IQR (например, Q1 - 1.5*IQR, Q3 + 1.5*IQR). Ответ: 2314 Пояснение: Последовательность отражает переход от общего к частному и от числового анализа к визуальному и активным действиям. Сначала получаем общую числовую картину (describe()) (2). Затем углубляемся в конкретные меры разброса (std, расчет IQR) (3). После этого переходим к визуальной	**Повышенный (3-5 минут)**

					получения конкретного результата (1). Нарушение порядка (например, попытка сгруппировать до загрузки данных) невозможно.	проверке (boxplot), которая подтверждает численные расчеты (1). И, наконец, на основе проведенного анализа выполняем фильтрацию данных (4). Такой порядок явл методологически верным.	
3	Тема 2.1 Построение графиков и визуализация с помощью библиотеки Matplotlib.		**ИПК-5.1 Знать:** Методы и средства проектирования компьютерного программного обеспечения	1,1	**3.** Установите соответствие между типом графика/диаграммы и основной функцией Matplotlib pyplot, используемой для его построения. Список 1 (Тип графика): 1. Линейный график 2. Столбчатая диаграмма (вертикальная) 3. Круговая диаграмма 4. Гистограмма Список 2 (Функция plt.*): A. bar() B. hist() C. pie() D. plot() Ответ: 1D, 2A, 3C, 4B.	**3.** Установите соответствие между параметром настройки в Matplotlib и типом графика, для которого он является наиболее характерным и часто используемым. Список 1 (Параметр): 1. wedgeprops (свойства секторов, например, ширина границы) 2. width (ширина столбцов) 3. linestyle (стиль линии) 4. whis (длина "усов" относительно межквартильного размаха) Список 2 (Тип графика): A. Линейный график (plot) B. График "ящик с усами" (boxplot) C. Круговая диаграмма (pie) D. Столбчатая диаграмма (bar/barh) Ответ: 1C, 2D, 3A, 4B.	**Базовый (1-3 минуты)**
4	Тема 2.2 Построение графиков и визуализация с помощью библиотеки Seaborn.		**ИОПК-6.1.** Знать основы теории систем и системного анализа, дискретной математики,	3,3	**4.** Вы хотите построить точечную диаграмму (scatter plot) в Seaborn, где цвет точек будет зависеть от значений третьей, категориальной переменной 'type', а размер точек — от значений четвертой, числовой переменной 'importance'. Какой из следующих вызовов функции	**4.** При построении гистограммы с графиком плотности (KDE) с помощью sns.histplot() какой параметр непосредственно отвечает за отображение или скрытие гладкой кривой оценки плотности поверх столбцов гистограммы?	**Повышенный (3-5 минут)**

			теории вероятностей и математической статистики, методов оптимизации и исследования операций, нечётких вычислений, математического и имитационного моделирования на базовом уровне		является синтаксически и концептуально правильным? Предположим, что df — это pandas DataFrame. 1. sns.scatterplot(data=df, x='var1', y='var2', color='type', size='importance') 2. sns.scatterplot(data=df, x='var1', y='var2', hue='type', scale='importance') 3. sns.scatterplot(data=df, x='var1', y='var2', hue='type', size='importance') 4. sns.scatterplot(data=df, x='var1', y='var2', group='type', volume='importance') Ответ: 3. Пояснение: В Seaborn для кодирования категорий цветом используется параметр hue, а для кодирования числовой величины размером — параметр size.	1. bins 2. kde 3. stat 4. alpha Ответ: 2. Пояснение: Параметр kde принимает булево значение (kde=True или kde=False) и управляет отображением ядерной оценки плотности (Kernel Density Estimate).	
5	Тема 3.1 Знакомство с библиотекой Scikit-Learn.		**ИПК-5.1 Знать:** Методы и средства проектирования компьютерного программного обеспечения	5,5	**5.** Как называется фундаментальный объект в Scikit-Learn, который инкапсулирует алгоритм машинного обучения и реализует стандартные методы .fit(), .predict() или .transform()? Ответ: Оценщик (Estimator). Пояснение: Термин "estimator"	**5.** Какой основной метод используется у обученного оценщика классификации или регрессии в Scikit-Learn для того, чтобы получить предсказания целевой переменной для новых (ранее не виденных моделью) данных? Ответ: .predict() (или метод predict).	**Высокий (5-10 минут)**

					(оценщик) — это ключевое понятие в API Scikit-Learn. Все алгоритмы обучения (как с учителем, так и без) представлены в виде классов, которые являются оценщиками и следуют этому единому интерфейсу. Это обеспечивает согласованность и простоту использования библиотеки.	Пояснение: Метод .predict() — это универсальный интерфейс для получения итоговых предсказаний модели на новых данных. Для классификации он возвращает метки классов, для регрессии — числовые значения. Это отличает его от .predict_proba() (вероятности) или .decision_function() (решающая функция).	
6	Тема 3.2 Методы классификации библиотеки Scikit-learn.		**ИОПК-6.1.** Знать основы теории систем и системного анализа, дискретной математики, теории вероятностей и математической статистики, методов оптимизации и исследования операций, нечётких вычислений, математического и имитационного моделирования на базовом уровне		**6.** Какие из следующих утверждений верно описывают процесс и принципы обучения моделей классификации, связанные с валидацией и переобучением? 1. Тестовая выборка (test set) используется для финальной, однократной оценки обобщающей способности модели, которую не видели ни при обучении, ни при настройке гиперпараметров. 2. Валидационная выборка (validation set) часто используется в процессе настройки гиперпараметров (например, силы регуляризации C или глубины дерева max_depth). 3. Если точность модели на обучающей выборке значительно выше, чем на тестовой, это верный признак недообучения (underfitting). 4. Для надежной оценки модели при малом объеме данных лучшей альтернативой простому	**6.** Какие из перечисленных методов/алгоритмов в Scikit-Learn являются ансамблевыми методами (ensemble methods), основанными на комбинации нескольких базовых моделей для повышения качества и устойчивости? 1. LogisticRegression 2. RandomForestClassifier 3. DecisionTreeClassifier 4. GradientBoostingClassifier (из sklearn.ensemble) Ответ: 2, 4. Пояснение: Ответ 2: Случайный лес (RandomForestClassifier) — это классический ансамблевый метод, объединяющий множество решающих деревьев, построенных на бутстрап-выборках и случайных подмножествах признаков. Ответ 4: Градиентный бустинг	**Базовый (1-3 минуты)**

					разделению на train/val/test является кросс-валидация (cross-validation). Ответ: 1, 2, 4. Пояснение: Ответ 1: Это строгое определение назначения тестовой выборки. Её трогать до финального этапа нельзя. Ответ 2: Это прямое назначение валидационной выборки в классическом workflow настройки гиперпараметров. Ответ 4: Кросс-валидация (например, KFold) позволяет более эффективно использовать данные для оценки и настройки модели, когда их мало.	(GradientBoostingClassifier) — это мощный ансамблевый метод, который последовательно строит деревья, каждое из которых исправляет ошибки предыдущих.	
7	Тема 3.3 Метрики качества классификации в библиотеке Scikit-learn.		**ИПК-5.1 Знать:** Методы и средства проектирования компьютерного программного обеспечения	2,2	**7.** Установите правильную последовательность шагов для оценки бинарного классификатора с использованием ROC-кривой и метрики AUC-ROC в Scikit-Learn. 1. Вычислить метрику roc_auc_score(y_true, y_pred_proba) для количественной оценки. 2. Получить предсказанные вероятности положительного класса для тестовых данных: y_pred_proba = model.predict_proba(X_test)[:, 1]. 3. Построить ROC-кривую, используя roc_curve(y_true, y_pred_proba) для получения точек и визуализировать её. 4. Обучить модель бинарной	**7.** Расположите этапы анализа качества модели при дисбалансе классов в порядке перехода от простой к более информативной и надежной оценке. 1. Посчитать Accuracy (долю правильных ответов). 2. Построить и проанализировать матрицу ошибок (confusion matrix). 3. Вычислить метрики, устойчивые к дисбалансу: Precision, Recall, F1-score. 4. Построить ROC-кривую и вычислить AUC-ROC, чтобы оценить качество модели независимо от выбранного порога классификации. Ответ: 1234	**Базовый (1-3 минуты)**

					классификации (например, LogisticRegression) на тренировочных данных. 5. Проанализировать кривую: чем ближе к верхнему левому углу и чем больше AUC-ROC (ближе к 1), тем лучше модель. Ответ: 42315 Пояснение: Последовательность отражает логичный pipeline. Сначала модель должна быть обучена (4). Затем, для построения ROC-кривой, нужны не "жёсткие" предсказания (predict), а вероятности (или decision function) (2). На основе этих вероятностей и истинных метрик строятся точки кривой (3). Площадь под этой кривой (AUC) вычисляется как интегральная числовая метрика (1). Завершается процесс интерпретацией результатов (5). Критически важно, что для ROC-кривой используются именно вероятности (шаг 2).	Пояснение: Эта последовательность показывает эволюцию понимания проблемы. Сначала студент может посчитать простую, но вводящую в заблуждение при дисбалансе Accuracy (1). Затем, увидев распределение ошибок в матрице (2), он понимает, в каком классе проблема. Далее он использует более тонкие метрики, сфокусированные на интересующем классе (Precision/Recall) (3). И наконец, для комплексной оценки, не зависящей от произвольного порога 0.5, он применяет ROC-AUC (4). Такой порядок учит критическому мышлению при выборе метрики.	
8	Тема 3.4 Методы кластеризации библиотеки Scikit-learn.		**ИОПК-6.1.** Знать основы теории систем и системного анализа, дискретной	1,1	**8.** Установите соответствие между методом кластеризации в Scikit-Learn и его ключевой характеристикой или ограничением. Список 1 (Метод / Алгоритм):	**8.** Установите соответствие между метрикой качества кластеризации в Scikit-Learn и типом метрики (внутренняя/внешняя), а также принципом её работы.	**Базовый (1-3 минуты)**

			математики, теории вероятностей и математической статистики, методов оптимизации и исследования операций, нечётких вычислений, математического и имитационного моделирования на базовом уровне		1. K-Means (KMeans) 2. Агломеративная кластеризация (AgglomerativeClustering) Список 2 (Ключевая характеристика): A. Плоский алгоритм, требующий указания числа кластеров заранее; чувствителен к выбросам и начальной инициализации; хорошо масштабируется. B. Иерархический алгоритм, строящий дендрограмму; позволяет выбрать число кластеров после построения; имеет высокую вычислительную сложность на больших выборках. Ответ: 1A, 2B.	Список 1 (Метрика): 1. silhouette_score 2. calinski_harabasz_score 3. adjusted_rand_score 4. davies_bouldin_score Список 2 (Описание): A. Внешняя метрика, сравнивающая найденные метки кластеров с истинными, с поправкой на случайное угадывание. B. Внутренняя метрика, оценивающая компактность и разделённость кластеров на основе среднего расстояния между объектами одного кластера и ближайшего другого кластера. C. Внутренняя метрика, вычисляющая отношение суммы межкластерных дисперсий к сумме внутрикластерных дисперсий. D. Внутренняя метрика, вычисляющая среднее «сходство» между каждым кластером и его наиболее похожим кластером, где сходство — это отношение внутрикластерных расстояний к расстоянию между кластерами (чем меньше значение, тем лучше). Ответ: 1B, 2C, 3A, 4D.	
9	Тема 4.1 Построение линейных регрессионных моделей.		**ИПК-5.1 Знать:** Методы и средства		**9.** В чем заключается ключевое различие между моделями Ridge и Lasso регрессии (обе являются регуляризованными версиями	**9.** Если все признаки в модели множественной линейной регрессии были стандартизированы (приведены к нулевому среднему и единичной	**Базовый (1-3 минуты)**

			проектирования компьютерного программного обеспечения		линейной регрессии)? 1. Ridge регрессия может использоваться только для бинарной классификации, а Lasso — для регрессии. 2. Lasso регрессия (L1-регуляризация) может обнулять веса неважных признаков, выполняя тем самым отбор признаков, в то время как Ridge (L2-регуляризация) лишь уменьшает их, но не до нуля. 3. Ridge регрессия всегда дает более высокое значение R^2 на тесте, чем Lasso. 4. Lasso регрессия не имеет гиперпараметра силы регуляризации, в отличие от Ridge. Ответ: 2. Пояснение: Это фундаментальное практическое различие. L1-норма в Lasso приводит к разреженным решениям, что полезно для интерпретации и отбора признаков. L2-норма в Ridge лишь сжимает коэффициенты. **10.** При построении множественной линейной регрессии в Scikit-Learn (LinearRegression) целевая функция, которую минимизирует алгоритм, — это: 1. Сумма модулей остатков (MAE).	дисперсии с помощью StandardScaler), то как можно интерпретировать свободный член (intercept) модели? Ответ: Это предсказанное значение целевой переменной при средних значениях всех признаков (поскольку после стандартизации среднее каждого признака равно 0). Пояснение: После стандартизации точка, где все X = 0, соответствует центру данных (средним значениям всех исходных признаков). Поэтому intercept (w0) представляет собой базовый уровень целевой переменной для "среднего" объекта. Это ключевой момент для осмысленной интерпретации модели. **10.** Как называется метрика, которая является квадратным корнем из средней квадратичной ошибки (MSE) и, в отличие от MSE, измеряется в тех же единицах, что и целевая переменная, что делает её более удобной для интерпретации?	

					2. Сумма квадратов остатков (RSS, Residual Sum of Squares). 3. Средняя абсолютная процентная ошибка (MAPE). 4. Коэффициент детерминации (R^2), который нужно максимизировать. Ответ: 2. Пояснение: Классический метод наименьших квадратов (OLS), реализованный в LinearRegression, минимизирует именно Residual Sum of Squares (RSS) или, что эквивалентно, Mean Squared Error (MSE).	Ответ: RMSE (Root Mean Squared Error) или Среднеквадратичная ошибка (СКО). Пояснение: RMSE = sqrt(MSE). Поскольку ошибки возводились в квадрат (в MSE), извлечение корня возвращает метрику к исходной размерности (например, рубли, метры, часы). Это самая распространенная метрика для итоговой оценки регрессии "в единицах измерения".	
10	Тема 4.2 Анализ и прогнозирование временных рядов.		**ИОПК-6.1.** Знать основы теории систем и системного анализа, дискретной математики, теории вероятностей и математической статистики, методов оптимизации и исследования операций, нечётких вычислений, математическо		**11.** При исследовательском анализе временного ряда (EDA) перед построением моделей используются автокорреляционные функции. Какие из следующих утверждений о коррелограмме (графике ACF и PACF) являются верными? 1. ACF (Autocorrelation Function) показывает чистую корреляцию между наблюдениями ряда с лагом k, без учета влияния промежуточных лагов. 2. PACF (Partial Autocorrelation Function) показывает корреляцию между наблюдениями с лагом k, исключив влияние наблюдений с лагами от 1 до k-1. 3. Медленное (экспоненциальное или синусоидальное) затухание ACF	**11.** Если p-value расширенного теста Дики-Фуллера (ADF test) для исходного временного ряда составил 0.82, а после взятия первых разностей (d=1) p-value стал равен 0.03 (при уровне значимости α=0.05), какой порядок интегрирования (d) следует выбрать для построения ARIMA-модели? Ответ: d=1. Пояснение: Высокий p-value (0.82 > 0.05) для исходного ряда означает, что мы не можем отвергнуть нулевую гипотезу о нестационарности (H0). Ряд нестационарен. После взятия первых разностей p-value (0.03) становится меньше α (0.05), что позволяет отвергнуть H0 и принять	**Высокий (5-10 минут)**

			го и имитационного моделирования на базовом уровне		до нуля может указывать на нестационарность ряда или наличие тренда. 4. Резкий обрыв (срез) PACF после лага p является индикатором порядка q для модели MA(q). Ответ: 2, 3. Пояснение: Ответ 2: Это точное определение частной автокорреляционной функции (PACF). Ответ 3: Медленное затухание ACF — классический признак нестационарности (например, наличия единичного корня). Для стационарного ряда ACF обычно быстро стремится к нулю. **12.** В рамках методологии Box-Jenkins (ARIMA) выбор порядка модели (p,d,q) является ключевым. Какие из следующих шагов/методов являются стандартными и корректными для этой цели? 1. Использование расширенного теста Дики-Фуллера (ADF test) для проверки стационарности и определения порядка интегрирования (d). 2. Визуальный анализ графиков ACF и PACF стационарного ряда для выявления потенциальных порядков p и q.	альтернативу о стационарности преобразованного ряда. Следовательно, для получения стационарного ряда достаточно одного дифференцирования: d=1. Это практическое применение теста. **12.**Как называется статистический тест, который применяется после построения ARIMA-модели для проверки гипотезы о том, что остатки модели являются "белым шумом" (не автокоррелированы)? Этот тест обычно проверяет автокорреляцию остатков на нескольких первых лагах одновременно. Ответ: Тест Льюнга-Бокса (Ljung-Box test). Пояснение: Это стандартный диагностический тест в анализе временных рядов. Нулевая гипотеза	

					3. Автоматический подбор параметров (p,d,q) с использованием критерия информативности, такого как AIC (Akaike Information Criterion), с помощью функции auto_arima. 4. Проверка остатков обученной модели на автокорреляцию с помощью теста Льюнга-Бокса (Ljung-Box test) для оценки адекватности модели. Ответ: 1, 2, 3, 4. Пояснение: Ответ 1: Тест Дики-Фуллера — стандартный статистический тест для определения необходимости взятия разностей (определение d). Ответ 2: Анализ ACF/PACF — это классический, хотя и субъективный, метод идентификации моделей AR и MA. Ответ 3: Использование auto_arima (из библиотеки pmdarima) или аналогичных методов, минимизирующих AIC/BIC, — это современный и распространенный автоматизированный подход. Ответ 4: Проверка остатков на "белошумность" — обязательный этап диагностики модели. Если остатки автокоррелированы, модель неадекватна. Все утверждения верно описывают части стандартного рабочего процесса ARIMA.	H0: остатки независимы (отсутствует автокорреляция до выбранного лага). Если p-value теста мал (например, < 0.05), мы отвергаем H0, что означает наличие значимой автокорреляции в остатках и, следовательно, неадекватность модели. Успешное прохождение этого теста — важный критерий качества модели.	

11	Раздел 1 «Общая характеристика систем анализа данных. Обработка данных с помощью библиотеки Pandas» Тема 1.1 Системы анализа данных Тема 1.2 Статистические методы библиотеки Pandas.	ПК-5 Способен осуществлять проектирование структур данных	ИПК-5.2 Уметь: Применять методы и средства проектирования компьютерного программного обеспечения, структур данных, баз данных, программных интерфейсов	5	13. Вам предоставлен фрагмент данных о транзакциях интернет-магазина. Выполните полный цикл первичного исследования и предобработки:
					1. Загрузите данные и изучите общую информацию (размер, типы, пропуски).
					2. Удалите полностью пустые строки и дубликаты.
					3. Заполните пропуски: в числовом столбце order_total - медианой, в категориальном payment_method - значением 'Unknown'.
					4. Преобразуйте тип данных в столбце order_date в формат datetime.
					5. Сформулируйте выводы о качестве данных и проведенных преобразованиях.
					
					Ответ:
					import pandas as pd
					import numpy as np
					from datetime import datetime
					
					# 1. Создание и загрузка синтетических данных
					np.random.seed(42)
					data = {
					'order_id': [1001, 1002, 1003, 1004, 1005, 1006, 1007],
						13. На основе предобработанных данных о продажах выполните статистический анализ:
						1. Вычислите общую выручку, средний чек и количество заказов по каждому региону.
						2. Найдите регион с максимальной и минимальной средней выручкой на один заказ.
						3. Проведите корреляционный анализ между customer_id и order_total.
						4. Сгруппируйте данные по payment_method и вычислите для каждой категории среднюю сумму заказа и количество транзакций. Сформулируйте выводы.
						
						Ответ:
						import pandas as pd
						import numpy as np
						
						# Используем очищенный датафрейм df_cleaned из предыдущей работы
						# Создадим расширенный набор для анализа
						np.random.seed(42)
						data = {
						'order_id': range(1001, 1021),
						'customer_id': np.random.randint(200, 220, 20),
						'order_total': np.random.uniform(50, 500, 20).round(2),
						'payment_method': np.random.choice(['Card', 'PayPal', 'Apple Pay'], 20),
							Высокий (5-10 минут)

					'customer_id': [201, 202, 203, 204, 202, 205, 206],
					'order_date': ['2023-10-01', '2023-10-02', None, '2023-10-04', '2023-10-02', '2023-10-05', '2023-10-06'],
					'order_total': [150.0, 89.99, 200.0, None, 89.99, 120.5, 75.0],
					'payment_method': ['Card', 'PayPal', 'Card', 'Card', 'PayPal', None, 'Card'],
					'region': ['North', 'South', 'East', 'West', 'South', 'North', 'East']
					}
					df = pd.DataFrame(data)
					
					print("1. ИСХОДНЫЕ ДАННЫЕ И ОБЩАЯ ИНФОРМАЦИЯ:")
					print(df)
					print(f"\nРазмер данных: {df.shape}")
					print("\nТипы данных:")
					print(df.dtypes)
					print("\nПропуски по столбцам:")
					print(df.isnull().sum())
					print("\nКоличество дубликатов:", df.duplicated().sum())
					
					# 2. Удаление полностью пустых строк и дубликатов
					df_cleaned = df.dropna(how='all')
					df_cleaned = df_cleaned.drop_duplicates()
					print(f"\n2. После удаления дубликатов: {df_cleaned.shape[0]} строк")
					
					# 3. Заполнение пропусков
					# Числовой столбец - медианой
					if df_cleaned['order_total'].isnull().any():
					median_val = df_cleaned['order_total'].median()
					df_cleaned['order_total'].fillna(median_val, inplace=True)
					print(f"Заполнены пропуски в 'order_total' медианой: {median_val:.2f}")
					
					# Категориальный столбец - значением 'Unknown'
					df_cleaned['payment_method'].fillna('Unknown', inplace=True)
						'region': np.random.choice(['North', 'South', 'East', 'West'], 20)
						}
						df = pd.DataFrame(data)
						
						print("СТАТИСТИЧЕСКИЙ АНАЛИЗ ПРОДАЖ ПО РЕГИОНАМ")
						print("="*50)
						print("Исходные данные (первые 5 строк):")
						print(df.head())
						
						# 1. Анализ по регионам
						print("\n1. СТАТИСТИКА ПО РЕГИОНАМ:")
						region_stats = df.groupby('region').agg(
						total_revenue=('order_total', 'sum'),
						avg_check=('order_total', 'mean'),
						orders_count=('order_id', 'count')
						).round(2)
						print(region_stats)
						
						# 2. Регион с максимальной и минимальной средней выручкой
						max_region = region_stats['avg_check'].idxmax()
						min_region = region_stats['avg_check'].idxmin()
						print(f"\n2. Регион с максимальным средним чеком: {max_region} ({region_stats.loc[max_region, 'avg_check']:.2f})")
						print(f" Регион с минимальным средним чеком: {min_region} ({region_stats.loc[min_region, 'avg_check']:.2f})")
						
						# 3. Корреляционный анализ
						correlation = df['customer_id'].corr(df['order_total'])
						print(f"\n3. Корреляция между customer_id и order_total: {correlation:.4f}")
						print(" Интерпретация: практически отсутствует линейная связь" if abs(correlation) < 0.3 else " Есть заметная линейная связь")
						
						# 4. Группировка по способам оплаты
						print("\n4. СТАТИСТИКА ПО СПОСОБАМ ОПЛАТЫ:")
						payment_stats = df.groupby('payment_method').agg(	

					# 4. Преобразование типа данных
					df_cleaned['order_date'] =
					pd.to_datetime(df_cleaned['order_date'],
					errors='coerce')
					print("\n4. Преобразован тип 'order_date' в
					datetime")
					
					# 5. Выводы
					print("\n" + "="*60)
					print("5. ИТОГОВЫЕ ВЫВОДЫ:")
					print("="*60)
					print(f"1. Исходный набор содержал {df.shape[0]}
					строк, {df.shape[1]} столбцов.")
					print(f"2. Обнаружено пропусков: order_date -
					{df['order_date'].isnull().sum()}, order_total -
					{df['order_total'].isnull().sum()}, payment_method -
					{df['payment_method'].isnull().sum()}.")
					print(f"3. Удалено дубликатов: {df.shape[0] -
					df_cleaned.shape[0]}.")
					print(f"4. После предобработки датасет содержит
					{df_cleaned.shape[0]} уникальных записей.")
					print(f"5. Все пропуски заполнены, типы данных
					корректны.")
					print(f"\nИтоговый датасет:")
					print(df_cleaned)
					print(f"\nТипы данных после обработки:")
					print(df_cleaned.dtypes)
						avg_order=('order_total', 'mean'),
						transaction_count=('order_id', 'count')
						).round(2)
						print(payment_stats)
						
						# Выводы
						print("\n" + "="*50)
						print("ВЫВОДЫ:")
						print("="*50)
						print("1. Наибольшая выручка наблюдается в
						регионе", region_stats['total_revenue'].idxmax())
						print("2. Самый высокий средний чек в регионе",
						max_region)
						print("3. Корреляция между ID клиента и суммой
						заказа практически отсутствует (неинформативный
						признак).")
						print("4. Наиболее популярный способ оплаты:",
						payment_stats['transaction_count'].idxmax())
						print(f"5. Всего проанализировано {len(df)}
						транзакций на общую сумму
						{df['order_total'].sum():.2f}")	

12	Раздел 3 «Методы машинного обучения библиотеки Scikit-Learn» Тема 3.1 Знакомство с библиоте	ОПК-6 Способен анализировать и разрабатывать организационно-технические и экономические процессы с применением методов	ИОПК-6.2. Уметь применять методы теории систем и системного анализа, математического, статистического и имитационного моделирования		14. Постройте модель для прогнозирования оттока клиентов (churn). 1. Разделите данные в соотношении 80/20 на обучающую и тестовую выборки с random_state=42. 2. Обучите две модели: логистическую регрессию и дерево решений (максимальная глубина=5).	14. Выполните кластеризацию клиентов по признакам balance и credit_score. 1. Масштабируйте данные с помощью StandardScaler. 2. Примените алгоритм K-means с k=3 и агломеративную кластеризацию с параметрами: n_clusters=3, linkage='ward'.	**Высокий (5-10 минут)**

	кой Scikit-Learn. Тема 3.2 Методы классификации библиотеки Scikit-learn. Тема 3.3 Метрики качества классификации в библиотеке Scikit-learn. Тема 3.4 Методы кластеризации библиотеки Scikit-learn.	системного анализа и математического моделирования	для автоматизации задач принятия решений, анализа информационных потоков, расчёта экономической эффективности и надёжности информационных систем технологий на базовом уровне		3. На тестовой выборке оцените обе модели по метрикам accuracy, precision, recall, F1-score.
					4. Определите, какая модель показала лучший результат по F1-score.
					
					Ответ:
					import pandas as pd
					import numpy as np
					from sklearn.model_selection import train_test_split
					from sklearn.linear_model import LogisticRegression
					from sklearn.tree import DecisionTreeClassifier
					from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score, classification_report
					
					# Создание синтетических данных банковских клиентов
					np.random.seed(42)
					n_samples = 200
					data = {
					'age': np.random.randint(18, 70, n_samples),
					'balance': np.random.exponential(50000, n_samples).round(2),
					'credit_score': np.random.normal(650, 100, n_samples).round(),
					'products_count': np.random.randint(1, 5, n_samples),
					'churn': np.random.choice([0, 1], n_samples, p=[0.7, 0.3])
					}
					# Искусственная зависимость для реализма
					data['churn'] = ((data['age'] > 60) | (data['balance'] < 10000) | (data['credit_score'] < 600)).astype(int)
					df = pd.DataFrame(data)
					
					print("ДАННЫЕ ДЛЯ КЛАССИФИКАЦИИ ОТТОКА КЛИЕНТОВ")
						3. Оцените качество кластеризации с помощью силуэтного коэффициента для обоих методов.
						4. Визуализируйте результаты кластеризации на scatter plot. Сделайте вывод о лучшем методе.
						
						Ответ:
						import pandas as pd
						import numpy as np
						import matplotlib.pyplot as plt
						from sklearn.preprocessing import StandardScaler
						from sklearn.cluster import KMeans, AgglomerativeClustering
						from sklearn.metrics import silhouette_score
						
						# Создание синтетических данных клиентов
						np.random.seed(42)
						n_samples = 150
						# Создаем 3 искусственных кластера
						cluster1 = np.random.multivariate_normal([50000, 750], [[20000, 0], [0, 50]], n_samples//3)
						cluster2 = np.random.multivariate_normal([150000, 650], [[30000, 0], [0, 40]], n_samples//3)
						cluster3 = np.random.multivariate_normal([250000, 550], [[40000, 0], [0, 30]], n_samples//3)
						
						data = np.vstack([cluster1, cluster2, cluster3])
						df = pd.DataFrame(data, columns=['balance', 'credit_score'])
						print(f"Данные для кластеризации: {df.shape[0]} клиентов")
						
						# 1. Масштабирование данных
						scaler = StandardScaler()
						X_scaled = scaler.fit_transform(df)
						print("1. Данные масштабированы (StandardScaler)")	

					print("="*50)
					print(f"Размер датасета: {df.shape}")
					print(f"Классовый
					баланс:\n{df['churn'].value_counts()}")
					print(f"Доля оттока: {df['churn'].mean():.2%}")
					
					# 1. Разделение данных
					X = df[['age', 'balance', 'credit_score',
					'products_count']]
					y = df['churn']
					X_train, X_test, y_train, y_test = train_test_split(X,
					y, test_size=0.2, random_state=42, stratify=y)
					print(f"\n1. Разделение данных: train
					{X_train.shape[0]} / test {X_test.shape[0]}")
					
					# 2. Обучение моделей
					print("\n2. ОБУЧЕНИЕ МОДЕЛЕЙ:")
					# Логистическая регрессия
					lr_model = LogisticRegression(random_state=42,
					max_iter=1000)
					lr_model.fit(X_train, y_train)
					print(" Логистическая регрессия обучена")
					
					# Дерево решений
					dt_model = DecisionTreeClassifier(max_depth=5,
					random_state=42)
					dt_model.fit(X_train, y_train)
					print(" Дерево решений обучено (max_depth=5)")
					
					# 3. Предсказание и оценка
					y_pred_lr = lr_model.predict(X_test)
					y_pred_dt = dt_model.predict(X_test)
					
					print("\n3. МЕТРИКИ КАЧЕСТВА НА
					ТЕСТОВОЙ ВЫБОРКЕ:")
					print("-"*50)
					
					def print_metrics(y_true, y_pred, model_name):
					 acc = accuracy_score(y_true, y_pred)
					 prec = precision_score(y_true, y_pred)
					 rec = recall_score(y_true, y_pred)
					 f1 = f1_score(y_true, y_pred)
					 print(f"{model_name}:")
						# 2. Применение алгоритмов кластеризации
						# K-means
						kmeans = KMeans(n_clusters=3, random_state=42,
						n_init=10)
						kmeans_labels = kmeans.fit_predict(X_scaled)
						print("2. K-means кластеризация выполнена")
						
						# Агломеративная кластеризация
						agglo = AgglomerativeClustering(n_clusters=3,
						linkage='ward')
						agglo_labels = agglo.fit_predict(X_scaled)
						print(" Агломеративная кластеризация выполнена
						(linkage='ward')")
						
						# 3. Оценка качества
						kmeans_silhouette = silhouette_score(X_scaled,
						kmeans_labels)
						agglo_silhouette = silhouette_score(X_scaled,
						agglo_labels)
						
						print("\n3. ОЦЕНКА КАЧЕСТВА
						КЛАСТЕРИЗАЦИИ:")
						print(f" K-means силуэтный коэффициент:
						{kmeans_silhouette:.3f}")
						print(f" Агломеративная силуэтный коэффициент:
						{agglo_silhouette:.3f}")
						
						# 4. Визуализация
						fig, axes = plt.subplots(1, 3, figsize=(15, 4))
						
						# Исходные данные
						axes[0].scatter(df['balance'], df['credit_score'],
						alpha=0.6, edgecolor='k')
						axes[0].set_title('Исходные данные')
						axes[0].set_xlabel('Баланс')
						axes[0].set_ylabel('Кредитный рейтинг')
						
						# K-means
						scatter1 = axes[1].scatter(df['balance'],
						df['credit_score'], c=kmeans_labels, cmap='viridis',
						alpha=0.6, edgecolor='k')
						axes[1].scatter(kmeans.cluster_centers_[:,
						0]*scaler.scale_[0] + scaler.mean_[0],	

					print(f" Accuracy: {acc:.3f}")
					print(f" Precision: {prec:.3f}")
					print(f" Recall: {rec:.3f}")
					print(f" F1-score: {f1:.3f}")
					return f1
					
					f1_lr = print_metrics(y_test, y_pred_lr, "Логистическая регрессия")
					print()
					f1_dt = print_metrics(y_test, y_pred_dt, "Дерево решений")
					
					# 4. Сравнение моделей
					print("\n4. СРАВНЕНИЕ МОДЕЛЕЙ:")
					print("-"*50)
					if f1_lr > f1_dt:
					print(f"Логистическая регрессия показала лучший F1-score: {f1_lr:.3f} против {f1_dt:.3f}")
					elif f1_dt > f1_lr:
					print(f"Дерево решений показало лучший F1-score: {f1_dt:.3f} против {f1_lr:.3f}")
					else:
					print("Модели показали одинаковый F1-score")
					
					print("\nДетальный отчет по логистической регрессии:")
					print(classification_report(y_test, y_pred_lr, target_names=['Не уйдет', 'Уйдет']))
						kmeans.cluster_centers_[:, 1]*scaler.scale_[1] + scaler.mean_[1],
						s=200, marker='X', c='red', label='Центроиды')
						axes[1].set_title(f'K-means (Silhouette: {kmeans_silhouette:.3f})')
						axes[1].set_xlabel('Баланс')
						axes[1].legend()
						
						# Agglomerative
						scatter2 = axes[2].scatter(df['balance'], df['credit_score'], c=agglo_labels, cmap='plasma', alpha=0.6, edgecolor='k')
						axes[2].set_title(f'Agglomerative (Silhouette: {agglo_silhouette:.3f})')
						axes[2].set_xlabel('Баланс')
						
						plt.tight_layout()
						plt.show()
						
						# 5. Выводы
						print("\n" + "="*50)
						print("ВЫВОДЫ ПО КЛАСТЕРИЗАЦИИ:")
						print("="*50)
						print(f"1. K-means выделил {len(np.unique(kmeans_labels))} кластера.")
						print(f"2. Агломеративная кластеризация выделила {len(np.unique(agglo_labels))} кластера.")
						if kmeans_silhouette > agglo_silhouette:
						print(f"3. K-means показал лучшее качество (силуэтный коэффициент {kmeans_silhouette:.3f} > {agglo_silhouette:.3f})")
						elif agglo_silhouette > kmeans_silhouette:
						print(f"3. Агломеративная кластеризация показала лучшее качество (силуэтный коэффициент {agglo_silhouette:.3f} > {kmeans_silhouette:.3f})")
						else:
						print(f"3. Оба метода показали одинаковое качество")
						print("4. Визуально кластеры хорошо разделяются по балансу и кредитному рейтингу.")	

13	Раздел 4 «Методы статистического анализа библиотеки StatsModels» Тема 4.1 Построение линейных регрессионных моделей. Тема 4.2 Анализ и прогнозирование временных рядов.	ПК-5 Способен осуществлять проектирование структур данных	ИПК-5.2 Уметь: Применять методы и средства проектирования компьютерного программного обеспечения, структур данных, баз данных, программных интерфейсов		15. Постройте модель линейной регрессии для прогнозирования цены дома.
					1. 1) Проведите предобработку: удалите выбросы в цене (верхние 5%), создайте признак area_per_room.
					2. 2) Выполните корреляционный анализ и отберите признаки с корреляцией с целевой переменной > 0.3.
					3. 3) Постройте модель линейной регрессии, разделив данные в соотношении 80/20.
					4. 4) Оцените модель с помощью R^2 и RMSE. Проинтерпретируйте коэффициенты модели.
					
					Ответ:
					import pandas as pd
					import numpy as np
					import matplotlib.pyplot as plt
					from sklearn.model_selection import train_test_split
					from sklearn.linear_model import LinearRegression
					from sklearn.metrics import r2_score, mean_squared_error
					
					# Создание синтетических данных о недвижимости
					np.random.seed(42)
					n_samples = 200
						15. Проанализируйте временной ряд месячных продаж.
						1. Проверьте ряд на стационарность с помощью теста Дики-Фуллера.
						2. Если ряд нестационарен, приведите его к стационарному виду с помощью сезонного дифференцирования (лаг=12).
						3. Постройте модель ARIMA с параметрами (1,1,1) и сделайте прогноз на 3 месяца вперед.
						4. Визуализируйте исходный ряд, прогноз и доверительные интервалы.
						
						Ответ:
						import pandas as pd
						import numpy as np
						import matplotlib.pyplot as plt
						from statsmodels.tsa.stattools import adfuller
						from statsmodels.tsa.arima.model import ARIMA
						from statsmodels.graphics.tsaplots import plot_acf, plot_pacf
						import warnings
						warnings.filterwarnings('ignore')
						
						# Создание синтетического временного ряда с трендом и сезонностью
						np.random.seed(42)
						n_periods = 60 # 5 лет по месяцам
						time_index = pd.date_range(start='2018-01-01', periods=n_periods, freq='M')
							Высокий (5-10 минут)

					data = {
					'area': np.random.uniform(50, 200, n_samples),
					'rooms': np.random.randint(1, 6, n_samples),
					'floor': np.random.randint(1, 25, n_samples),
					'metro_distance': np.random.exponential(2, n_samples).round(1),
					'price': 0 # Будем вычислять
					}
					df = pd.DataFrame(data)
					
					# Генерация целевой переменной с зависимостью от признаков
					df['price'] = (df['area'] * 1000 +
					df['rooms'] * 50000 -
					df['metro_distance'] * 10000 +
					np.random.normal(0, 50000, n_samples))
					
					print("ДАННЫЕ О НЕДВИЖИМОСТИ")
					print("="*50)
					print(f"Размер датасета: {df.shape}")
					print("\nПервые 5 строк:")
					print(df.head())
					
					# 1. Предобработка
					# Удаление выбросов в цене (верхние 5%)
					price_threshold = df['price'].quantile(0.95)
					df_clean = df[df['price'] <= price_threshold].copy()
					print(f"\n1. ПРЕДОБРАБОТКА:")
					print(f" Удалено выбросов: {len(df) - len(df_clean)} записей (верхние 5% по цене)")
					
					# Создание нового признака
					df_clean['area_per_room'] = df_clean['area'] / df_clean['rooms']
					print(f" Создан новый признак 'area_per_room'")
					
					# 2. Корреляционный анализ
					correlation_matrix = df_clean.corr()
					price_corr = correlation_matrix['price'].sort_values(ascending=False)
					print("\n2. КОРРЕЛЯЦИЯ С ЦЕЛЕВОЙ ПЕРЕМЕННОЙ 'price':")
						# Компоненты ряда: тренд + сезонность + шум
						trend = np.linspace(100, 300, n_periods)
						seasonality = 50 * np.sin(2 * np.pi * np.arange(n_periods) / 12)
						noise = np.random.normal(0, 15, n_periods)
						
						sales = trend + seasonality + noise
						df = pd.DataFrame({'sales': sales}, index=time_index)
						
						print("АНАЛИЗ ВРЕМЕННОГО РЯДА ПРОДАЖ")
						print("="*50)
						print(f"Период: {df.index[0].date()} - {df.index[-1].date()}")
						print(f"Длина ряда: {len(df)} месяцев")
						print("\nПервые 12 значений:")
						print(df.head(12))
						
						# 1. Проверка на стационарность
						print("\n1. ТЕСТ ДИКИ-ФУЛЛЕРА (ПРОВЕРКА СТАЦИОНАРНОСТИ):")
						adf_result = adfuller(df['sales'])
						print(f" ADF статистика: {adf_result[0]:.3f}")
						print(f" p-value: {adf_result[1]:.3f}")
						print(f" Критические значения:")
						for key, value in adf_result[4].items():
						print(f" {key}: {value:.3f}")
						
						if adf_result[1] > 0.05:
						print(" Вывод: Ряд НЕ стационарен (p-value > 0.05)")
						else:
						print(" Вывод: Ряд стационарен")
						
						# 2. Приведение к стационарному виду (сезонное дифференцирование)
						df_diff = df['sales'].diff(12).dropna() # Сезонное дифференцирование (лаг=12)
						adf_result_diff = adfuller	

					for feature, corr in price_corr.items():
					  print(f" {feature}: {corr:.3f}")
					
					# Отбор признаков с корреляцией > 0.3 (исключая саму цену)
					selected_features = price_corr[price_corr > 0.3].index.tolist()
					selected_features.remove('price') if 'price' in selected_features else None
					print(f"\n Отобранные признаки (corr > 0.3): {selected_features}")
					
					# 3. Построение модели
					X = df_clean[selected_features]
					y = df_clean['price']
					X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
					
					model = LinearRegression()
					model.fit(X_train, y_train)
					print(f"\n3. МОДЕЛЬ ЛИНЕЙНОЙ РЕГРЕССИИ ПОСТРОЕНА")
					print(f" Обучающая выборка: {X_train.shape[0]} записей")
					print(f" Тестовая выборка: {X_test.shape[0]} записей")
					
					# 4. Оценка модели
					y_pred = model.predict(X_test)
					r2 = r2_score(y_test, y_pred)
					rmse = np.sqrt(mean_squared_error(y_test, y_pred))
					
					print("\n4. ОЦЕНКА КАЧЕСТВА МОДЕЛИ:")
					print(f" R² (коэффициент детерминации): {r2:.3f}")
					print(f" RMSE (среднеквадратичная ошибка): {rmse:.0f} у.е.")
					print(f" Средняя цена в тестовой выборке: {y_test.mean():.0f} у.е.")
					print(f" Относительная ошибка (RMSE/Средняя цена): {rmse/y_test.mean():.1%}")
					
					# Интерпретация коэффициентов		

					print("\nИНТЕРПРЕТАЦИЯ КОЭФФИЦИЕНТОВ:")
					for feature, coef in zip(selected_features, model.coef_):
					  print(f" {feature}: {coef:.1f} (при увеличении на 1 единицу, цена изменяется на {coef:.1f} у.е.)")
					print(f" Intercept (базовая цена): {model.intercept_:.1f} у.е.")
					
					# Визуализация
					plt.figure(figsize=(10, 4))
					
					plt.subplot(1, 2, 1)
					plt.scatter(y_test, y_pred, alpha=0.6)
					plt.plot([y_test.min(), y_test.max()], [y_test.min(), y_test.max()], 'r--', lw=2)
					plt.xlabel('Фактическая цена')
					plt.ylabel('Предсказанная цена')
					plt.title(f'Предсказание vs Факт (R^2 = {r2:.3f})')
					
					plt.subplot(1, 2, 2)
					residuals = y_test - y_pred
					plt.scatter(y_pred, residuals, alpha=0.6)
					plt.axhline(y=0, color='r', linestyle='--')
					plt.xlabel('Предсказанная цена')
					plt.ylabel('Остатки')
					plt.title('Анализ остатков')
					
					plt.tight_layout()
					plt.show()		

Критерии оценивания диагностической работы:

Оценка	Критерии оценивания
отлично	от 90 до 100 % правильно выполненных заданий
хорошо	от 70 до 89 % правильно выполненных заданий
удовлетворительно	от 50 до 69 % правильно выполненных заданий
неудовлетворительно	менее 50 % правильно выполненных заданий