МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯРОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»(ФГБОУ ВО «ВГТУ», ВГТУ)

Факультет информационных технологий и компьютерной безопасности

Кафедра компьютерных интеллектуальных технологий проектирования

ОТЧЁТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №3

По дисциплине: «Искусственный интеллект в программно-информационных системах»

Тема: «Визуализация данных»

Выполнил работу студент группы змИИВТ-231:                            Енин М. М.

подпись, дата

Принял:                                                                                             Короленко В.В.

подпись, дата

Воронеж 2024

Содержание

1. Содержание 2
2. Введение 3
3. Описание работы 3
4. Построение графиков 5
5. Заключение 9
6. Контрольные вопросы 13
7. Литература 14

# **ВВЕДЕНИЕ**

**Jupyter Notebook** – это веб-приложение, которое позволяет создавать и делиться документами, содержащими код на различных языках программирования, математические выражения, графики и текстовые комментарии. Оно широко используется в научных исследованиях, анализе данных и образовании благодаря своей интуитивной и интерактивной природе.

Jupyter Notebook состоит из ячеек двух типов: ячейки кода и ячейки markdown. Ячейки кода содержат исходный код, который можно выполнять для получения результатов, в то время как ячейки markdown позволяют добавлять форматированный текст, изображения и математические формулы.

**Описание работы.**

Визуализация данных — это метод, который позволяет специалистам по анализу данных преобразовывать сырые данные в диаграммы и графики, которые несут ценную информацию. Диаграммы уменьшают сложность данных и делают более понятными для любого пользователя.

Есть множество инструментов для визуализации данных, таких как Tableau, Power BI, ChartBlocks и других, которые являются no-code инструментами. Они очень мощные, и у каждого своя аудитория. Однако для работы с сырыми данными, требующими обработки, а также в качестве песочницы, Python подойдет лучше всего. Несмотря на то, что этот путь сложнее и требует умения программировать, Python позволит вам провести любые манипуляции, преобразования и визуализировать ваши данные. Он идеально подходит для специалистов по анализу данных.

Python — лучший инструмент для data science и этому много причин, но самая важная — это его экосистема библиотек. Для работы с данными в Python есть много замечательных библиотек, таких как numpy, pandas, matplotlib, tensorflow.

Seaborn — это библиотека для создания статистических графиков на Python. Она основывается на matplotlib и тесно взаимодействует со структурами данных pandas. Архитектура Seaborn позволяет вам быстро изучить и понять свои данные. Seaborn захватывает целые фреймы данных или массивы, в которых содержатся все ваши данные, и выполняет все внутренние функции, нужные для семантического маппинга и статистической агрегации для преобразования данных в информативные графики. Она абстрагирует сложность, позволяя вам проектировать графики в соответствии с вашими нуждами.

Установка Seaborn Установить seaborn так же просто, как и любую другую библиотеку, для этого вам понадобится ваш любимый менеджер пакетов Python.

Во время установки seaborn библиотека установит все зависимости, включая matplotlib, pandas, numpy и scipy.

Давайте уже установим seaborn и, конечно же, также пакет notebook, чтобы получить доступ к песочнице с данными. pipenv install seaborn notebook

Помимо этого, перед началом работы давайте импортируем несколько модулей.

import seaborn as sns

import pandas as pd

import numpy as np

import matplotlib.

# **Построение графиков**

# Перед тем, как мы начнем строить графики, нам нужны данные. Прелесть seaborn в том, что он работает непосредственно с объектами dataframe из pandas, что делает ее очень удобной. Более того, библиотека поставляется с некоторыми встроенными наборами данных, которые можно использовать прямо из кода, и не загружать файлы вручную.В качестве данных будет использоваться атрибуты для наборов данных Student-mat.csv (курс математики) скаченный с сайта <https://www.kaggle.com/datasets/uciml/student-alcohol-consumption/data> - употребление алкоголя студентами.

Данные по умолчанию:

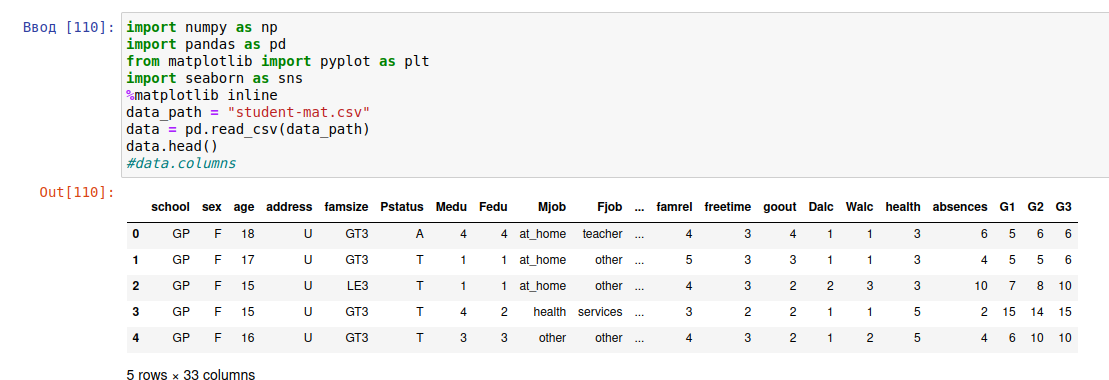


Рисунок 1- Загрузка данных, получение и первичный анализ признаков

* school - школа ученика (двоичный код: 'GP' - Габриэль Перейра или 'MS' - Мусиньо да Силвейра)
* sex — пол студента (двоичный код: «Ж» — женский или «М» — мужской)
* age - возраст ученика (числовой: от 15 до 22)
* адрес — тип домашнего адреса студента (двоичный: «U» — городской или «R» — сельский)
* famsize — размер семейства (двоичный: «LE3» — меньше или равно 3 или «GT3» — больше 3)
* Pstatus — статус совместного проживания родителей (двоичный код: «Т» — проживание вместе или «А» — раздельно)
* Medu— образование матери (числовое: 0 — нет, 1 — начальное образование (4 класс), 2 — с 5 по 9 классы, 3 — среднее образование или 4 — высшее образование)
* Fedu— образование отца (числовое: 0 — нет, 1 — начальное образование (4 класс), 2 — с 5 по 9 классы, 3 — среднее образование или 4 — высшее образование)
* Mjob - работа матери (номинал: «учитель», «связанный со здравоохранением», гражданские «службы» (например, административные или полицейские), «на дому» или «другое»)
* Fjob - работа отца (номинал: «учитель», «связанный со здравоохранением», гражданские «службы» (например, административные или полицейские), «на дому» или «другое»)
* reason- причина выбрать эту школу (номинал: близок к «дому», «репутация» школы, предпочтение «курса» или «другое»)
* guardian- опекун ученика (номинальное: «мать», «отец» или «другой»)
* traveltime — время в пути домой в школу (числовое: 1 – <15 минут, 2 – 15 – 30 минут, 3 – 30 минут до 1 часа или 4 – >1 часа)
* Studytime — еженедельное время обучения (числовое: 1 — <2 часов, 2 — 2–5 часов, 3 — 5–10 часов или 4 — >10 часов)
* failures- количество прошлых сбоев класса (числовое: n, если 1<=n<3, иначе 4)
* schoolup - дополнительная образовательная поддержка (двоичный код: да или нет)
* famsup — образовательная поддержка семьи (двоичный код: да или нет)
* paid— дополнительные платные занятия по предмету курса (математика или португальский язык) (двоичный код: да или нет)
* activities– внеклассные мероприятия (бинарный вариант: да или нет)
* nursery - посещал детский сад (двоичный вариант: да или нет)
* higher– хочет получить высшее образование (двоичный вариант: да или нет)
* Internet — доступ в Интернет дома (двоичный код: да или нет)
* romantic– с романтическими отношениями (бинарный: да или нет)
* famrel- качество семейных отношений (числовое: от 1 - очень плохо до 5 - отлично)
* freetime – свободное время после школы (числовое: от 1 – очень низкое до 5 – очень высокое)
* goout – встреча с друзьями (числовое: от 1 – очень низкий до 5 – очень высокий)
* Dalc - потребление алкоголя в течение рабочего дня (числовое: от 1 - очень низкое до 5 - очень высокое)
* Walc – потребление алкоголя в выходные дни (числовое: от 1 – очень низкое до 5 – очень высокое)
* health– текущее состояние здоровья (числовое: от 1 – очень плохое до 5 – очень хорошее)
* absences- количество пропусков занятий (число: от 0 до 93)

**Эти оценки связаны с предметом курса**

* G1 – оценка первого урока (числовая: от 0 до 20)
* G2 – оценка за второй период (числовая: от 0 до 20)
* G3 — итоговая оценка (числовая: от 0 до 20, выходная цель)ель)

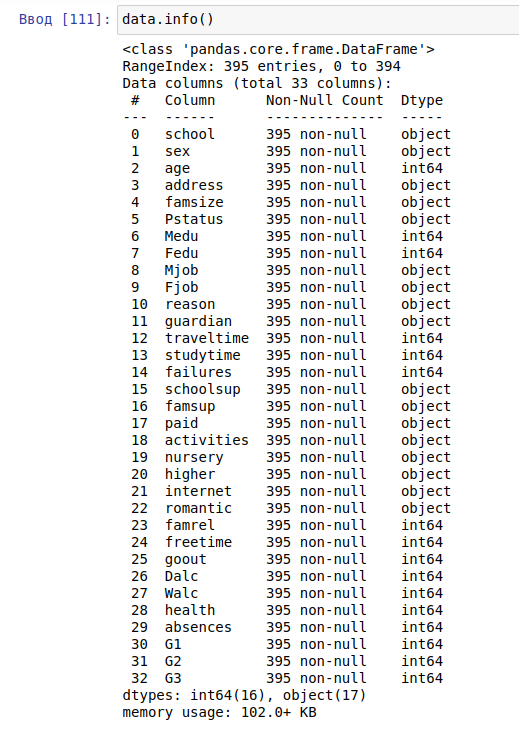


Рисунок 2 - Информация о признаках набора данных

#### **Количественные признаки**

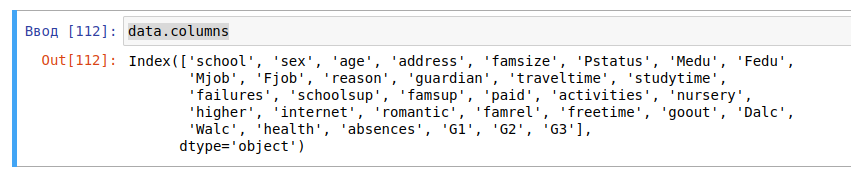


Диаграмма возраста. Показывает количество студентов в зависимости от возраста.

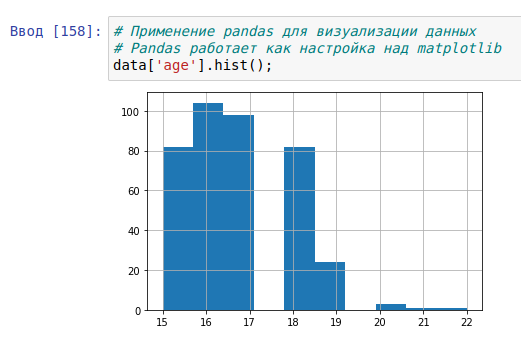


Рисунок 3 – Информация о признаках набора данных

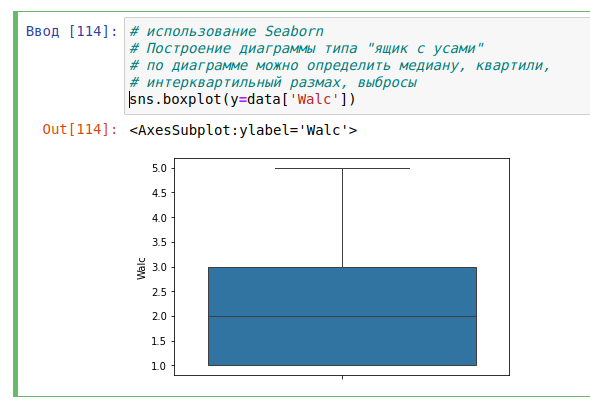


Рисунок 4 – График «ящик с усами» для отдельного признака

Из «ящика с усами» показывает потребление алкоголя по шкале от 1-5.

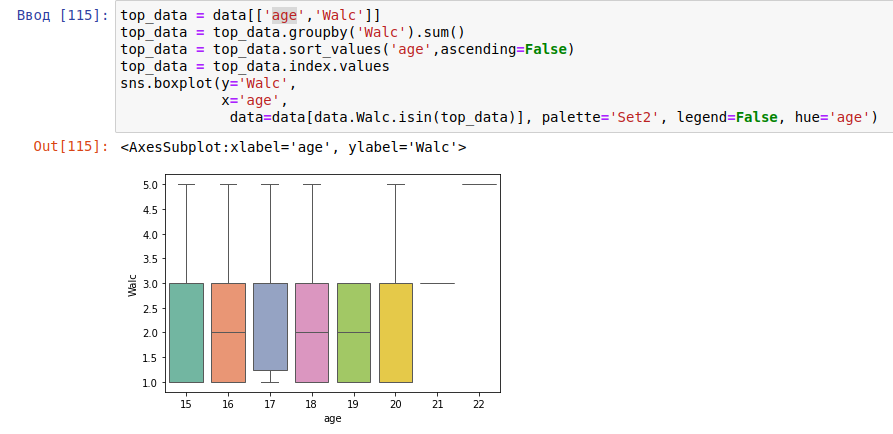


Рисунок 5 – Использование boxplot для анализа признака по возрасту

Да множественной диаграмме «ящик с усами» можно увидеть потребление алкоголя по шкале от 1-5 в зависимости от возраста.

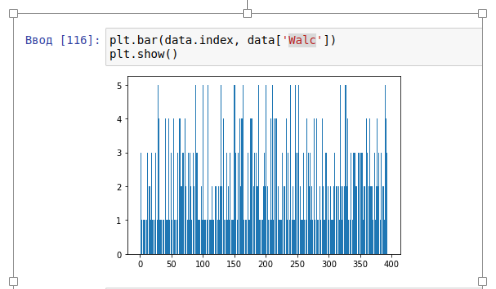


Рисунок 6 – Построение гистограммы с использованием matplotlib

**Категориальные признаки**

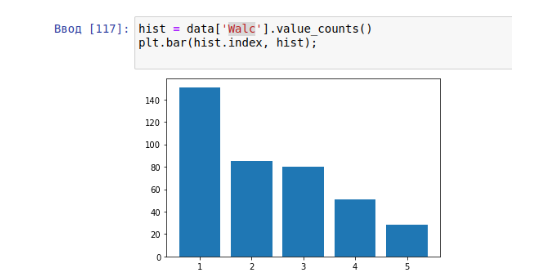


Рисунок 7 – Распределение количества употребляющих алкоголь по пятибальной шкале.

**Визуализация соотношения количественных признаков**

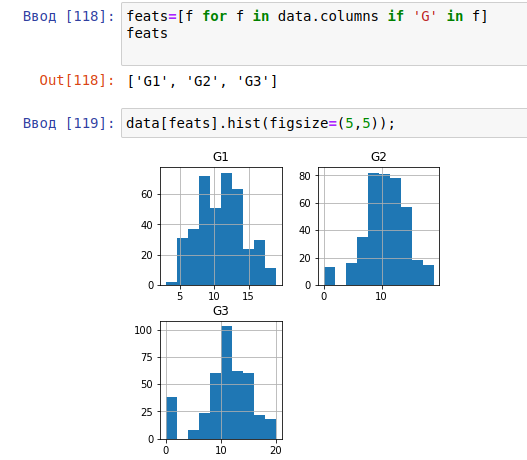


Рисунок 8 – Диаграммы для сравнения распределения числовых показателей

**Попарное распределение**

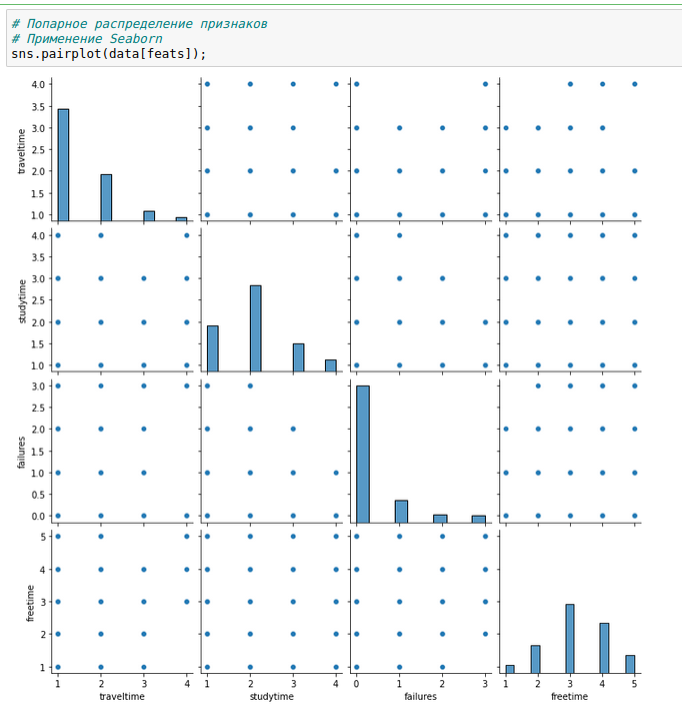
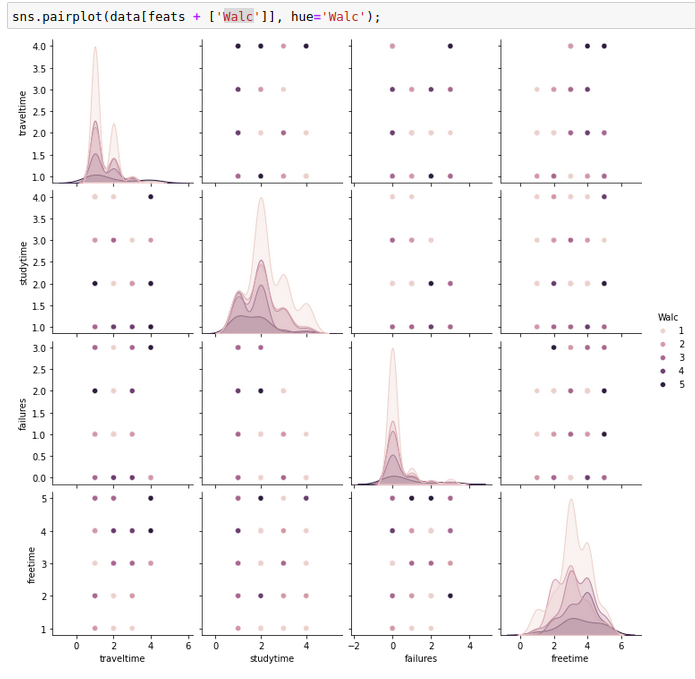


Рисунок 9 – Попарное распределение признаков

### **Корреляция признаков**

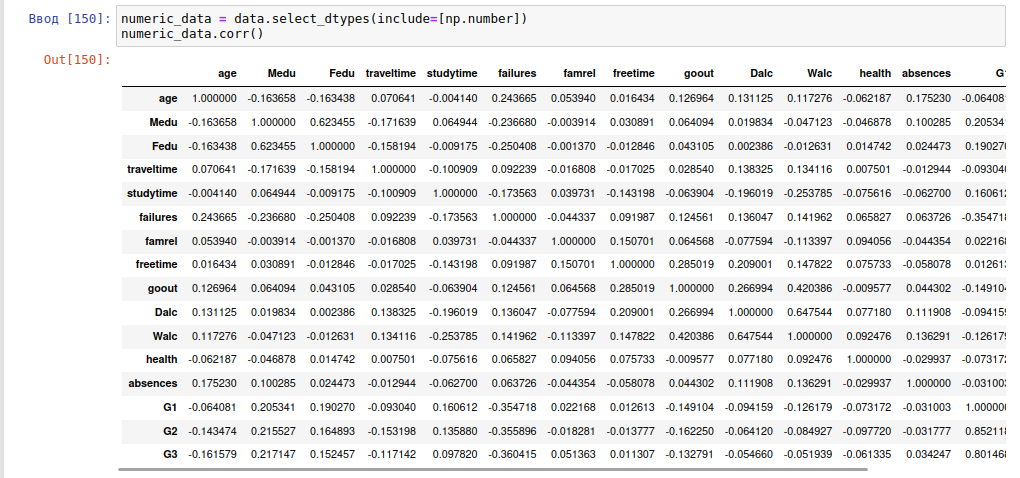


Рисунок 10 – Определение коррелирующих признаков набора данных

В реальных задачах анализ такой матрицы затруднен, необходимо или автоматизировать (скрипты), или визуализировать (графики)

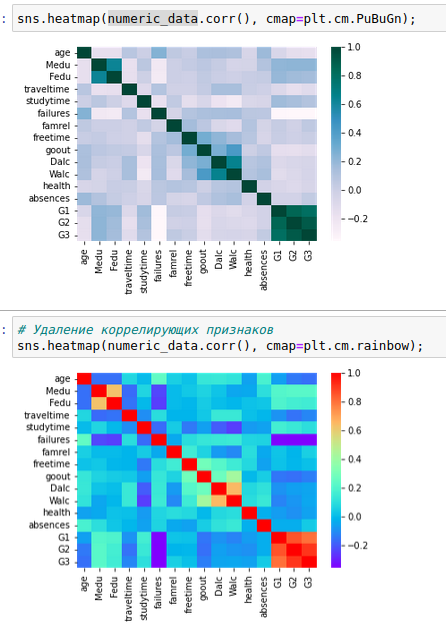


Рисунок 11 - Визуализация матрицы корреляции с использованием графика типа heatmap.

# **Заключение**

# Вывод по работе с библиотекой Seaborn: Библиотека Seaborn является мощным инструментом для визуализации данных и предоставляет множество функций для создания привлекательных и информативных графиков. В процессе работы с Seaborn были изучены основные типы графиков, такие как линейные графики, гистограмм ящичковые диаграммы, swarm-графики и другие. Одной из ключевых особенностей Seaborn является возможность легко настраивать стиль графиков, включая цвета, шрифты и другие аспекты оформления. Это позволяет создавать визуализации, которые легко читаются и привлекают внимание. Также была изучена работа с матрицами данных и созданием сеточных графиков с помощью функции `pairplot()`. Это позволяет визуализировать взаимосвязи между различными переменными в данных и выявлять закономерности. В процессе работы с Seaborn была изучена работа с категориальными данными и созданием графиков, которые иллюстрируют распределение значений в зависимости от категорий. Это может быть полезно при анализе данных, содержащих качественные переменные. В целом, работа с библиотекой Seaborn показала, что она является удобным и эффективным инструментом для визуализации данных. Ее функции и возможности позволяют создавать привлекательные и информативные графики, которые помогают в анализе и интерпретации данных.

# **Контрольные вопросы**

Термины:

1. Для организации рабочего места специалиста Data Science используются различные инструментальные средства, такие как: - Компьютер с достаточной вычислительной мощностью и операционной системой, поддерживающей необходимыеы и библиотеки. - Операционная система, чаще всего это Linux, macOS или Windows. - Среда разработки (IDE), такие как Jupyter Notebook, PyCharm, Visual Studio Code и другие. - Библиотеки и фреймворки для работы с данными и машинного обучения, например, Python, R, Julia и другие. - Утилиты для работы с данными, такие как SQL-клиенты, NoSQL-базы данных и другие. - Средства для визуализации данных, такие как Matplotlib, Seaborn, ggplot и другие. - Средства для контроля версий, такие как Git, SVN и другие.

2. В области машинного обучения в Python используются следующие библиотеки: - Scikit-learn: библиотека для машинного обучения, предоставляет широкий набор алгоритмов для классификации, регрессии, кластеризации и других задач. - TensorFlow: фреймворк для глубинного обучения, разработанный компанией Google. - Keras: высокоуров библиотека для глубинного обучения, построенная на основе TensorFlow, Theano или CNTK. - PyTorch: фреймворк для глубинного обучения, разработанный компанией Facebook. - Pandas: библиотека для работы с данными, предоставляет удобные структуры данных и инструменты для анализа и манипуляции данными. - NumPy: библиотека для работы с массивами и математическими операциями над ними. - Matplotlib: библиотека для визуализации данных, предоставляет широкий набор инструментов для построения графиков и диаграмм.

3. Библиотеки Python получили широкое распространение в реализации систем машинного обу следующим причинам: - Python имеет простой и интуитивно понятный синтаксис, что упрощает написание и чтение кода. - Python обладает большим количеством библиотек и фреймворков для различных задач машинного обучения и анализа данных. - Python имеет мощную сообщество разработчиков, что позволяет быстро решать возникающие проблемы и получать поддержку. - Python поддерживает интеграцию с другими языками программирования, что позволяет использовать его в составе сложных систем.

4. В рамках данной лабораторной работы были изучены следующие функции Python для визуализации данных: - Функции библиотеки Matplotlib для построения различных типов графиков и диаграмм. - Функции библиотеки Seaborn для построения статистических графиков и визуализации данных в более привлекательном и информативном виде. - Функции библиотеки Pandas для визуализации данных, хранящихся в DataFrame.

5. Для управления наборами данных в Python наиболее подходящей библиотекой является `pandas`. Библиотека `pandas` предоставляет мощные инструменты для работы с данными, включая структурированные и разнородные данные. Она позволяет легко импортировать, экспортировать, фильтровать, группировать, объединять и преобразовывать данные. Основными типами данных, которые предоставляет `pandas`, являются `Series` (одномерный массив) и `DataFrame` (двумерный массив с возможностью иметь разнородные типы данных в столбцах). Библиотеки `numpy`, `sklearn`, `opencv` и `matplotlib` также важны в контексте работы с данными, но они имеют более специализированное назначение: - `numpy` - это библиотека для работы с многомерными массивами и матрицами, предоставляет широкий набор функций для математических операций. - `sklearn` (scikit-learn) - это библиотека для машинного обучения, которая предоставляет реализации множества алгоритмов обучения, включая классификацию, регрессию, кластеризацию и т.д. - `opencv` - это библиотека для работы с компьютерным зрением и обработкой изображений. - `matplotlib` - это библиотека для визуализации данных, позволяет создавать различные типы графиков и диаграмм. В целом, `pandas` является основным инструментом для управления и манипуляции данными, а другие библиотеки используются для более специализированных задач, связанных с анализом и визуализацией данных.

6. Нежелательной стратегией при обработке пропусков в данных является "б) удаление строк, содержащих пропуски в данных". Это связано с тем, что удаление строк может привести к потере значительной части данных, особенно если пропуски распространены.

7. Применение OneHotEncoder к данному целевому признаку является целесообразным. Однако, важно учитывать, что использование OneHotEncoder может привести к увеличению размерности пространства признаков, что может быть нежелательным в некоторых случаях. В таких ситуациях можно рассмотреть альтернативные методы кодирования категориальных переменных, такие как LabelEncoder или TargetEncoder.

8. Принцип разбиения набора данных на обучающую и тестовую выборку заключается в разделении имеющихся данных на две части: одна часть используется для обучения модели (обучающая выборка), а другая - для оценки ее производительности (тестовая выборка). Это необходимо для того, чтобы оценить, насколько хорошо модель может обобщать свои знания на новые, невидимые ей данные. Оптимальное соотношение "тестовая:обучающая" зависит от конкретной задачи и объема имеющихся данных. Однако, часто используется соотношение 20:80 или 25:75, то есть 20-25% данных отводится для тестовой выборки, а 80-75% - для обучающей. Это позволяет иметь достаточное количество данных для обучения модели и в то же время иметь представительную выборку для оценки ее качества. Важно отметить, что при разбиении данных необходимо учитывать баланс классов, если задача является классификацией, и другие особенности данных, чтобы обеспечить представительность обеих выборок.

9. Для загрузки данных из csv-файла в Python лучше использовать функцию `read\_csv()` из библиотеки pandas. Поэтому правильный ответ: а) `dataset = read\_csv("data.csv")`

**Список литературы:**

1. Уэс, Маккинли. Python и анализ данных Электронный ресурс / Маккинли Уэс ; пер. А. А. Слинкин. - Python и анализ данных,2022-04-19. - Саратов : Профобразование, 2017. - 482 с. - Книга находится в премиум-версии ЭБС IPR BOOKS. - ISBN 978-5-4488-0046-7, экземпляров неограниченно.

2. Сузи, Р.А. Язык программирования Python Электронный ресурс : учебное пособие / Р.А. Сузи. - Язык программирования Python,2020-07-28. - Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. - 350 c. - Книга находится в базовой версии ЭБС IPRbooks. - ISBN 5-9556-0058-2, экземпляров неограниченно.

3. Стенли, Липпман. Язык программирования С++ Электронный ресурс : Полное руководство / Липпман Стенли, Лажойе Жози ; пер. А. Слинкин. - Язык программирования С++,2022-04-19. - Саратов : Профобразование, 2017. - 1104 с. - Книга находится в премиум-версии ЭБС IPR BOOKS. - ISBN 978-5-4488-0136-5, экземпляров неограниченно

4. https://github.com/enikolaev/MMO – Репозиторий с примерами кода из лабораторных работ.

5. https://archive.ics.uci.edu/ml/index.html – Репозиторий наборов данных для машинного обучения (Центр машинного обучения и интеллектуальных систем).

6. https://www.kaggle.com – Портал и система проведения соревнований по проблемам анализа данных.

7. https://www.mockaroo.com – Сайт для генерации наборов данных.

8. <https://habr.com/ru/articles/349204> – Статья на хабре: «Моделирование динамических систем: введение в GNU Octave»