Министерство образования Республики Беларусь
Учреждение образования
«Брестский Государственный технический университет»
Кафедра ИИТ

Лабораторная работа №1

По дисциплине: «Основы машинного обучения»

Тема: «Знакомство с анализом данных: предварительная обработка и визуализация»

Выполнил:
Студентка 3 курса
Группы АС-65
Сергиевич М.А.

Проверил:
Крощенко А. А.

Брест 2025

Цель работы: получить практические навыки работы с данными с использованием библиотек Pandas для манипуляции и Matplotlib для визуализации. Научиться выполнять основные шаги предварительной обработки данных, такие как очистка, нормализация и работа с различными типами признаков.

Ход работы

Общее задание:

1. Загрузить предложенный набор данных (по вариантам) в DataFrame библиотеки Pandas.
2. Провести исследовательский анализ: изучить типы данных, количество пропусков, основные статистические показатели (среднее, медиана, стандартное отклонение).
3. Обработать пропущенные значения (например, заполнить средним значением или удалить строки/столбцы).
4. Преобразовать категориальные признаки в числовые с помощью метода One-Hot Encoding.
5. Выполнить нормализацию или стандартизацию числовых признаков.
6. Построить несколько графиков для визуализации данных (гистограммы, диаграммы рассеяния) и сделать выводы о зависимостях между признаками.
7. **Написать отчет, создать пул-реквест в репозиторий с кодом решения и отчетом в формате pdf.**

Используемые инструменты: Python, Pandas, Matplotlib, NumPy, Jupyter Notebook / Google Colab / PyCharm

Вариант 6

Выборка Heart Disease. Содержит медицинские данные пациентов, такие как возраст, пол, уровень холестерина, и наличие заболевания сердца.

Задачи:

1. Загрузите данные и выведите информацию о них. Проверьте на наличие пропусков.

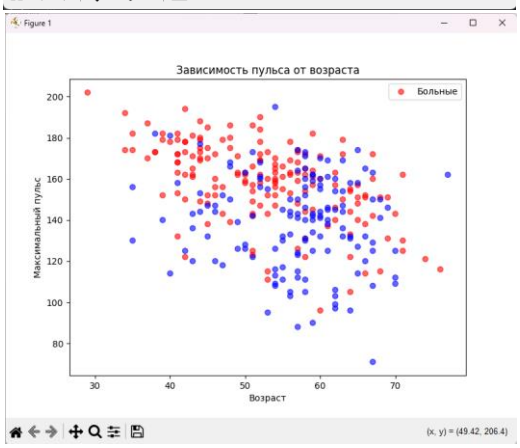
2. Постройте столбчатую диаграмму, сравнивающую количество здоровых и больных пациентов.
3. Создайте диаграмму рассеяния, показывающую зависимость максимального пульса (thalach) от возраста (age). Раскрасьте точки в зависимости от наличия болезни.
4. Преобразуйте признак sex (0 = женщина, 1 = мужчина) в более читаемый формат с категориями 'female' и 'male', а затем примените к нему One-Hot Encoding.
5. Рассчитайте средний уровень холестерина (chol) для больных и здоровых пациентов.
6. Выполните нормализацию признаков age, trestbps, chol и thalach.

Код программы:

```
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.preprocessing import MinMaxScaler

# загрузка данных
df = pd.read_csv("heart.csv")

print("1. ИНФОРМАЦИЯ О ДАННЫХ:")
df_info = df.info()
print(df_info)

print("\nПропуски в данных:")
print(df.isna().sum())

print("\nОСНОВНЫЕ СТАТИСТИКИ:")
print(df.describe())

print("\nМедианы:")
print(df.median(numeric_only=True))

print("\nСтандартные отклонения:")
print(df.std(numeric_only=True))

# проверка пропусков
if df.isna().sum().sum() == 0:
    print("\nПропусков нет, обрабатывать не нужно.")
else:
    df.fillna(df.mean(numeric_only=True), inplace=True)
    print("\nПропуски заменены средними значениями.")

# график количества пациентов
plt.figure(figsize=(6, 4))
df["target"].value_counts().plot.bar()
plt.title("Количество здоровых и больных пациентов")
plt.xlabel("Target (0-здоров, 1-болен)")
plt.ylabel("Количество")
plt.xticks(rotation=0)
plt.show()
print("Вывод: больных пациентов больше, чем здоровых.")

# график пульс vs возраст
plt.figure(figsize=(8, 6))
plt.scatter(df["age"], df["thalach"],
            c=["red" if t == 1 else "blue" for t in df["target"]],
            alpha=0.6)
plt.title("Зависимость пульса от возраста")
plt.xlabel("Возраст")
plt.ylabel("Максимальный пульс")
plt.legend(["Больные", "Здоровые"])
plt.show()
print("Вывод: у молодых пациентов чаще выше пульс.")

# кодирование пола
df["sex"] = df["sex"].map({0: "female", 1: "male"})
df = pd.concat([df, pd.get_dummies(df["sex"], prefix="sex")], axis=1)

print("\n4. Результат One-Hot Encoding:")
print(df[["sex", "sex_female", "sex_male"]].head())

# сравнение холестерина
chol_sick = df.loc[df["target"] == 1, "chol"].mean()
chol_healthy = df.loc[df["target"] == 0, "chol"].mean()
```

```
print("\n5. Средний уровень холестерина:")
print(f"Больные: {chol_sick:.2f}")
print(f"Здоровые: {chol_healthy:.2f}")

# нормализация признаков
scaler = MinMaxScaler()
cols = ["age", "trestbps", "chol", "thalach"]
df[cols] = scaler.fit_transform(df[cols])

print("\n6. Данные после нормализации:")
print(df[cols].head())
```

Графики:




```
1. ИНФОРМАЦИЯ О ДАННЫХ:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 303 entries, 0 to 302
Data columns (total 14 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   age       303 non-null    int64
 1   sex       303 non-null    int64
 2   cp        303 non-null    int64
 3   trestbps  303 non-null    int64
 4   chol      303 non-null    int64
 5   fbs       303 non-null    int64
 6   restecg   303 non-null    int64
 7   thalach   303 non-null    int64
 8   exang     303 non-null    int64
 9   oldpeak   303 non-null    float64
 10  slope     303 non-null    int64
 11  ca        303 non-null    int64
 12  thal      303 non-null    int64
 13  target    303 non-null    int64
dtypes: float64(1), int64(13)
memory usage: 33.3 KB
```

```
Пропуски в данных:
age         0
sex         0
cp          0
trestbps    0
chol        0
fbs         0
restecg     0
thalach     0
exang       0
oldpeak     0
slope       0
ca          0
thal        0
target      0
dtype: int64
```

```
ОСНОВНЫЕ СТАТИСТИКИ:
              age         sex          cp  ...          ca        thal      target
count  303.000000  303.000000  303.000000  ...  303.000000  303.000000  303.000000
mean    54.366337    0.683168    0.966997  ...    0.729373    2.313531    0.544554
std      9.082101    0.466011    1.032052  ...    1.022606    0.612277    0.498835
min     29.000000    0.000000    0.000000  ...    0.000000    0.000000    0.000000
25%     47.500000    0.000000    0.000000  ...    0.000000    2.000000    0.000000
50%     55.000000    1.000000    1.000000  ...    0.000000    2.000000    1.000000
75%     61.000000    1.000000    2.000000  ...    1.000000    3.000000    1.000000
max     77.000000    1.000000    3.000000  ...    4.000000    3.000000    1.000000

[8 rows x 14 columns]
```

```
[8 rows x 14 columns]

Медианы:
age          55.0
sex           1.0
cp            1.0
trestbps    130.0
chol        240.0
fbs           0.0
restecg       1.0
thalach     153.0
exang         0.0
oldpeak       0.8
slope         1.0
ca            0.0
thal          2.0
target        1.0
dtype: float64
```

```
Стандартные отклонения:
age          9.082101
sex          0.466011
cp           1.032052
trestbps    17.538143
chol        51.830751
fbs          0.356198
restecg      0.525860
thalach     22.905161
exang        0.469794
oldpeak      1.161075
slope        0.616226
ca           1.022606
thal         0.612277
target       0.498835
dtype: float64
```

```
Пропусков нет, обработка не требуется.
Вывод: больных пациентов больше, чем здоровых.
Вывод: у более молодых пациентов чаще встречается более высокий пульс.

4. Результат One-Hot Encoding:
      sex  sex_female  sex_male
0    male       False      True
1    male       False      True
2  female        True     False
3    male       False      True
4  female        True     False

5. Средний уровень холестерина:
Больные: 242.23
Здоровые: 251.09

6. Данные после нормализации:
        age  trestbps      chol   thalach
0  0.708333  0.481132  0.244292  0.603053
1  0.166667  0.339623  0.283105  0.885496
2  0.250000  0.339623  0.178082  0.770992
3  0.562500  0.245283  0.251142  0.816794
4  0.583333  0.245283  0.520548  0.702290
```

Вывод: В ходе данной лабораторной работы я получила практические навыки работы с данными с использованием библиотек Pandas для манипуляции и Matplotlib для визуализации.
Научилась выполнять основные шаги предварительной обработки данных, такие как очистка, нормализация и работа с различными типами признаков.