

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

**Лабораторная работа №1**

По дисциплине: «ОМО»

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

Выполнил:  
Студент 3-го курса  
Группы АС-66  
Ануфриенко М. А.  
Проверил:  
Крощенко А. А.

**Цель:** 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.

**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.

**Задания по вариантам:**

**Вариант 1**

Выборка Titanic. Содержит информацию о пассажирах лайнера, включая их возраст, пол, класс каюты и факт выживания.

**Задачи:**

1. Загрузите данные и выведите первые 5 строк, а также общую информацию о столбцах (.info()).
2. Найдите и визуализируйте количество выживших и погибших пассажиров с помощью столбчатой диаграммы.
3. Обработайте пропуски в столбце Age, заполнив их медианным значением.
4. Преобразуйте категориальные признаки Sex и Embarked в числовые с помощью One-Hot Encoding.
5. Постройте гистограмму распределения возрастов пассажиров.
6. Создайте новый признак FamilySize путем сложения значений из столбцов SibSp и Parch.

## Код программы:

```
# Импорт библиотек

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import io # для буфера info()

# 1. Загрузка данных

df = pd.read_csv(r"D:\ЛАБЫ\ОМО\Titanic-Dataset.csv") # используем сырую строку для
пути

# --- Задача 1: первые строки и общая информация ---

report1 = "Первые 5 строк датасета:\n" + df.head().to_string() + "\n\n"
report1 += "Общая информация о данных:\n"

buffer = io.StringIO() # создаем буфер
df.info(buf=buffer)

report1 += buffer.getvalue() # записываем содержимое info()

with open("report_task1.txt", "w", encoding="utf-8") as f:
    f.write(report1)
```

## Report\_task1

Первые 5 строк датасета:

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch
Ticket	Fare	Cabin	Embarked				
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1 0
A/5 21171	7.2500	NaN	S				
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Thayer)	female	38.0	1
0	PC 17599	71.2833	C85	C			
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0 0
STON/O2. 3101282	7.9250	NaN	S				
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1 0
113803	53.1000	C123	S				
4	5	0	3	Allen, Mr. William Henry	male	35.0	0 0
373450	8.0500	NaN	S				

Общая информация о данных:

```
<class 'pandas.core.frame.DataFrame'>
```

```
RangeIndex: 891 entries, 0 to 890
```

```
Data columns (total 12 columns):
```

```
#   Column      Non-Null Count  Dtype
```

```
---  -----  -
```

```
0   PassengerId  891 non-null    int64
```

```
1   Survived     891 non-null    int64
```

```
2   Pclass       891 non-null    int64
```

```
3   Name         891 non-null    object
```

```
4   Sex          891 non-null    object
```

```
5   Age          714 non-null    float64
```

```
6   SibSp        891 non-null    int64
```

```
7   Parch        891 non-null    int64
```

```
8   Ticket       891 non-null    object
```

```
9   Fare         891 non-null    float64
```

```
10  Cabin        204 non-null    object
```

```
11  Embarked     889 non-null    object
```

```
dtypes: float64(2), int64(5), object(5)
```

```
memory usage: 83.7+ KB
```

```
# --- Задача 2: количество выживших и погибших ---
```

```
survived_counts = df['Survived'].value_counts()
```

```
plt.figure(figsize=(6,4))
```

```
survived_counts.plot(kind='bar', color=['red','green'])
```

```
plt.title("Количество выживших (1) и погибших (0)")
```

```
plt.xlabel("Статус выживания")
```

```
plt.ylabel("Количество пассажиров")
```

```
plt.xticks(rotation=0)
```

```
plt.savefig("survival_counts.png") # сохраняем график
```

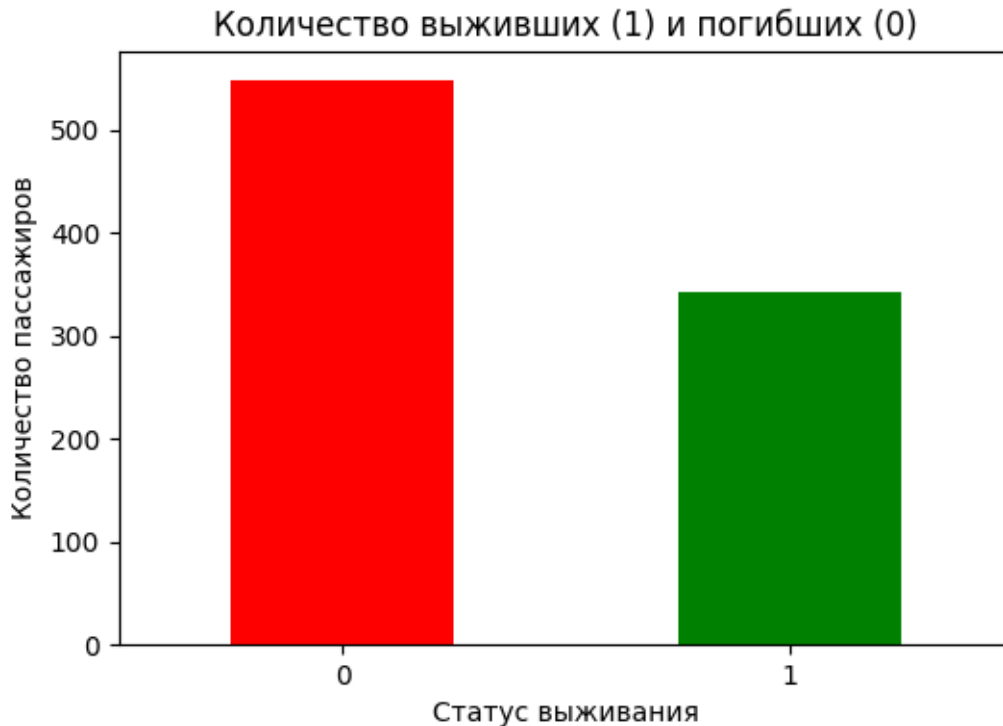
```
plt.close()
```

```
report2 = f"Количество выживших и погибших:\n{survived_counts.to_string()}\nГрафик  
сохранён в 'survival_counts.png'."
```

```
with open("report_task2.txt", "w", encoding="utf-8") as f:
```

```
f.write(report2)
```

## Report\_task2



```
# --- Задача 3: обработка пропусков в Age ---
```

```
median_age = df['Age'].median()
```

```
df['Age'].fillna(median_age, inplace=True)
```

```
report3 = f"Медианное значение возраста: {median_age}\nПропуски в столбце 'Age'\nзаполнены медианой."
```

```
with open("report_task3.txt", "w", encoding="utf-8") as f:
```

```
    f.write(report3)
```

## Report\_task3

Медианное значение возраста: 28.0

Пропуски в столбце 'Age' заполнены медианой.

```
# --- Задача 4: One-Hot Encoding ---
```

```
df = pd.get_dummies(df, columns=['Sex','Embarked'], drop_first=True)
```

```
report4 = "Преобразованы категориальные признаки 'Sex' и 'Embarked' в числовые с\nпомощью One-Hot Encoding.\n"
```

```
report4 += "Первые строки после преобразования:\n" + df.head().to_string()
with open("report_task4.txt", "w", encoding="utf-8") as f:
    f.write(report4)
```

## Report\_task4

Преобразованы категориальные признаки 'Sex' и 'Embarked' в числовые с помощью One-Hot Encoding.

Первые строки после преобразования:

PassengerId	Survived	Pclass	Name	Age	SibSp	Parch
Ticket	Fare	Cabin	Sex_male	Embarked_Q	Embarked_S	
0	1	0	3		Braund, Mr. Owen Harris	22.0 1 0 A/5
21171	7.2500	NaN	True	False	True	
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Thayer)	38.0	1 0
PC 17599	71.2833	C85	False	False	False	
2	3	1	3	Heikkinen, Miss. Laina	26.0	0 0 STON/O2.
3101282	7.9250	NaN	False	False	True	
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	35.0	1 0
113803	53.1000	C123	False	False	True	
4	5	0	3	Allen, Mr. William Henry	35.0	0 0 373450
8.0500	NaN	True	False	True		

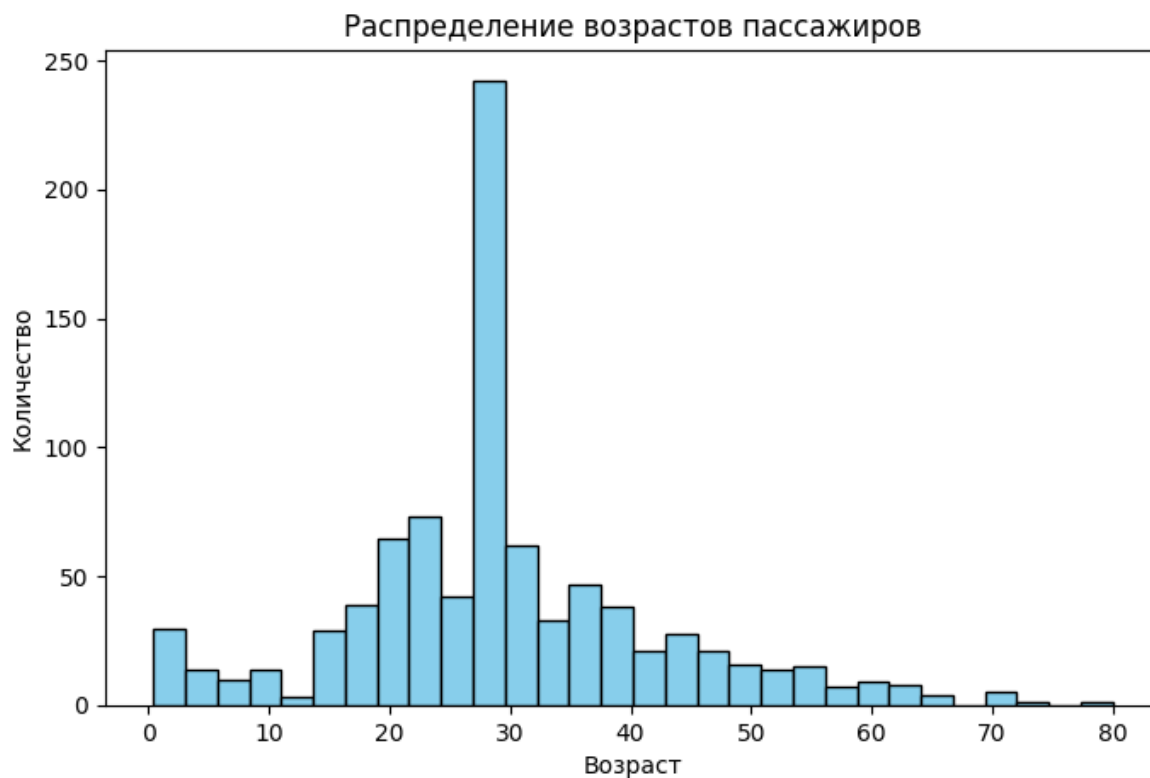
# --- Задача 5: гистограмма возрастов ---

```
plt.figure(figsize=(8,5))
plt.hist(df['Age'], bins=30, color='skyblue', edgecolor='black')
plt.title("Распределение возрастов пассажиров")
plt.xlabel("Возраст")
plt.ylabel("Количество")
plt.savefig("age_distribution.png") # сохраняем график
plt.close()
```

report5 = "Построена гистограмма распределения возрастов пассажиров.\nГрафик сохранён в 'age\_distribution.png'."

```
with open("report_task5.txt", "w", encoding="utf-8") as f:
    f.write(report5)
```

## Report\_task5



# --- Задача 6: новый признак FamilySize ---

```
df['FamilySize'] = df['SibSp'] + df['Parch']
```

```
report6 = "Создан новый признак 'FamilySize' = SibSp + Parch.\n"
```

```
report6 += "Первые строки с новым признаком:\n" +  
df[['SibSp', 'Parch', 'FamilySize']].head().to_string()
```

```
with open("report_task6.txt", "w", encoding="utf-8") as f:
```

```
    f.write(report6)
```

## Report\_task6

Создан новый признак 'FamilySize' = SibSp + Parch.

Первые строки с новым признаком:

	SibSp	Parch	FamilySize
0	1	0	1
1	1	0	1
2	0	0	0
3	1	0	1
4	0	0	0