Министерство науки и высшего образования Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования

«Ульяновский государственный технический университет»

Кафедра «Вычислительная техника»

Дисциплина: : «Системы искусственного интеллекта»

Лабораторная работа №1

«Анализ пассажиров корабля "Титаник"»

Выполнил:

студент группы ИВТАПбд-41

Князев А.Д.

Проверил:

преподаватель кафедры «ВТ»

Святов К.В.

Ульяновск, 2025

Цель работы:  
Необходимо дополнить программу, которая дана в условии лабораторной работы, чтобы она могла определять:

1. Сколько мужчин/женщин было на борту?
2. Определите распределение функции Pclass. Теперь Для мужчин и женщин отдельно. Сколько людей из второго класса было на борту?
3. Каковы медиана и стандартное отклонение Fare? Округлить значение до 2-х знаков после запятой.
4. Правда ли, что средний возраст выживших людей выше, чем у пассажиров, которые в конечном итоге умерли?
5. Это правда, что пассажиры моложе 30 лет. выжили чаще, чем те, кому больше 60 лет. Каковы доли выживших людей среди молодых и пожилых людей?
6. Правда ли, что женщины выживали чаще мужчин? Каковы доли выживших людей среди мужчин и женщин?
7. Какое имя наиболее популярно среди пассажиров мужского пола?
8. Как средний возраст мужчин / женщин зависит от Pclass?

### Реализация:

Файл titanic\_train.csv содержит информацию о пассажирах.

С помощью библиотеки pandas (Panel Data Systems) был выполнен анализ информации.

Сначала происходит загрузка данных:

|  |
| --- |
| import pandas as pd  data = pd.read\_csv('titanic\_train.csv', index\_col='PassengerId') |

Далее добавляются возрастные характеристики:

|  |
| --- |
| def age\_category(age):  if age < 30: return 1 # молодые  elif age < 55: return 2 # средний возраст  elif age >= 55: return 3 # пожилые  data['Age\_category'] = data['Age'].apply(age\_category) |

Затем начинается работа над дополнениями кода, согласно заданию:

1. Сколько мужчин/женщин было на борту?

|  |
| --- |
| gender\_counts = data['Sex'].value\_counts()  print("Количество мужчин и женщин на борту:")  print(gender\_counts)  print() |

В результате получаем вывод, представленный на рисунке 1:

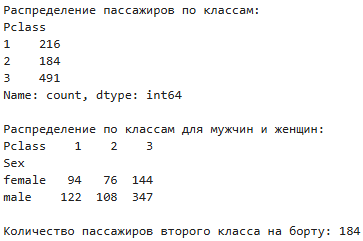


**Рисунок 1**. Количество мужчин/женщин на борту

1. Определите распределение функции PClass. Теперь для мужчин и женщин отдельно. Сколько людей из второго класса было на борту?

|  |
| --- |
| pclass\_distribution = data['Pclass'].value\_counts().sort\_index()  print("Распределение пассажиров по классам:")  print(pclass\_distribution)  print()  print("Распределение по классам для мужчин и женщин:")  gender\_pclass = data.groupby(['Sex', 'Pclass']).size().unstack()  print(gender\_pclass)  print()  # 4. Количество людей из второго класса  second\_class\_count = data[data['Pclass'] == 2].shape[0]  print(f"Количество пассажиров второго класса на борту: {second\_class\_count}") |

В результате получим вывод, представленный на рисунке 2:



**Рисунок 2**. Определение функции PClass

1. Каковы медиана и стандартное отклонение Fare? Округлите до 2-х знаков после запятой.

|  |
| --- |
| fare\_median = round(data['Fare'].median(), 2)  fare\_std = round(data['Fare'].std(), 2)  print(f"Медиана Fare: {fare\_median}")  print(f"Стандартное отклонение Fare: {fare\_std}") |

В результате будет получен вывод с рисунка 3:

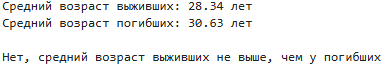


**Рисунок 3**. Медиана и стандартное отклонение

1. Правда ли, что средний возраст выживших людей выше, чем у пассажиров, которые в конечном итоге умерли?

|  |
| --- |
| survived\_age = data[data['Survived'] == 1]['Age'].mean()  not\_survived\_age = data[data['Survived'] == 0]['Age'].mean()  print(f"Средний возраст выживших: {round(survived\_age, 2)} лет")  print(f"Средний возраст погибших: {round(not\_survived\_age, 2)} лет")  if survived\_age > not\_survived\_age:  print("\nДа, средний возраст выживших действительно выше, чем у погибших")  else:  print("\nНет, средний возраст выживших не выше, чем у погибших") |

В результате увидим результат (рисунок 4):

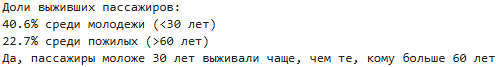


**Рисунок 4.** Средний возраст выживших

|  |
| --- |
| young\_survived = data[(data['Age'] < 30) & (data['Survived'] == 1)].shape[0]  young\_total = data[data['Age'] < 30].shape[0]  young\_percent = round(young\_survived / young\_total \* 100, 1) if young\_total > 0 else 0  old\_survived = data[(data['Age'] > 60) & (data['Survived'] == 1)].shape[0]  old\_total = data[data['Age'] > 60].shape[0]  old\_percent = round(old\_survived / old\_total \* 100, 1) if old\_total > 0 else 0  print(f"\nДоли выживших пассажиров:")  print(f"{young\_percent}% среди молодежи (<30 лет)")  print(f"{old\_percent}% среди пожилых (>60 лет)")  if young\_percent > old\_percent:  print("Да, пассажиры моложе 30 лет выживали чаще, чем те, кому больше 60 лет")  else:  print("Нет, пассажиры моложе 30 лет выживали не чаще, чем те, кому больше 60 лет") |

1. Это правда, что пассажиры моложе 30 лет. выжили чаще, чем те, кому больше 60 лет. Каковы доли выживших людей среди молодых и пожилых людей?

Результатом будет информация, представленная на рисунке 5:

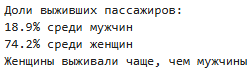


**Рисунок 5.** Возраст выживших

1. Правда ли, что женщины выживали чаще мужчин? Каковы доли выживших людей среди мужчин и женщин?

|  |
| --- |
| male\_survived = data[(data['Sex'] == 'male') & (data['Survived'] == 1)].shape[0]  male\_total = data[data['Sex'] == 'male'].shape[0]  male\_percent = round(male\_survived / male\_total \* 100, 1) if male\_total > 0 else 0  female\_survived = data[(data['Sex'] == 'female') & (data['Survived'] == 1)].shape[0]  female\_total = data[data['Sex'] == 'female'].shape[0]  female\_percent = round(female\_survived / female\_total \* 100, 1) if female\_total > 0 else 0  print(f"\nДоли выживших пассажиров:")  print(f"{male\_percent}% среди мужчин")  print(f"{female\_percent}% среди женщин")  if female\_percent > male\_percent:  print("Женщины выживали чаще, чем мужчины")  else:  print("Мужчины выживали чаще, чем женщины") |

Результат этого участка кода представлено на рисунке 7:



**Рисунок 6.** Доли выживших мужчины/женщины

1. Какое имя наиболее популярно среди пассажиров мужского пола?

|  |
| --- |
| def extract\_first\_name(full\_name):  parts = full\_name.split(', ')  if len(parts) > 1:  name\_part = parts[1]  name\_part = name\_part.split('. ')[-1]  first\_name = name\_part.split()[0]  return first\_name.strip('"()')  return None  male\_names = data[data['Sex'] == 'male']['Name'].apply(extract\_first\_name).dropna()  if not male\_names.empty:  name\_counts = male\_names.value\_counts()  top\_name = name\_counts.index[0]  top\_count = name\_counts.iloc[0]  print(f"\nСамое популярное имя среди мужчин: '{top\_name}'")  print(f"Оно встречается {top\_count} раз(а)") |

Результат проверки на самое популярное мужское имя представлено на рисунке 7:

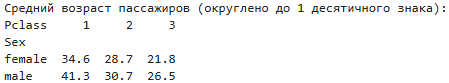


**Рисунок 7.** Самое популярное мужское имя

1. Как средний возраст мужчин/женщин зависит от PClass?

|  |
| --- |
| age\_stats = data.groupby(['Sex', 'Pclass'])['Age'].mean().unstack()  print("\nСредний возраст пассажиров (округлено до 1 десятичного знака):")  print(round(age\_stats, 1)) |

Результаты представлены на рисунке 8:

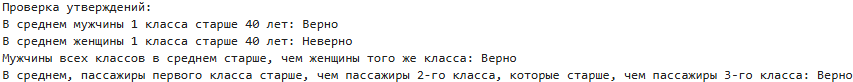


**Рисунок 8.** Округлённый средний возраст

1. Следует проверить несколько утверждений, основанных на результатах среднего возраста пассажиров.

|  |
| --- |
| male\_class1 = age\_stats.loc['male', 1]  female\_class1 = age\_stats.loc['female', 1]  statements = {  "В среднем мужчины 1 класса старше 40 лет": male\_class1 > 40,  "В среднем женщины 1 класса старше 40 лет": female\_class1 > 40,  "Мужчины всех классов в среднем старше, чем женщины того же класса": all(age\_stats.loc['male'] > age\_stats.loc['female']),  "В среднем, пассажиры первого класса старше, чем пассажиры 2-го класса, которые старше, чем пассажиры 3-го класса":  all(age\_stats[1] > age\_stats[2]) and all(age\_stats[2] > age\_stats[3])  }  print("\nПроверка утверждений:")  for statement, is\_true in statements.items():  print(f"{statement}: {'Верно' if is\_true else 'Неверно'}") |

Результатом будет следующее (рисунок 9):



**Рисунок 9.** Проверка утверждений

### Вывод

В ходе данной лабораторной работы был проведён анализ базы данных, представляющей собой базу данных пассажиров корабля “Титаник” при помощи библиотеки pandas языка python.