

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  
Осовец А.О.  
Проверил:  
Крощенко А.А.

Брест 2025

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.

### Вариант 7

Выборка Auto MPG. Содержит технические характеристики различных автомобилей и данные о расходе топлива (миль на галлон).

Задачи:

1. Загрузите данные. Обратите внимание, что пропуски в столбце horsepower могут быть обозначены знаком ?.
2. Преобразуйте столбец horsepower в числовой формат и заполните пропуски средним значением.
3. Постройте диаграмму рассеяния, чтобы изучить зависимость расхода топлива (mpg) от веса автомобиля (weight).
4. Преобразуйте категориальный признак origin (страна производства) в числовой.
5. Создайте новый признак age, рассчитав возраст автомобиля относительно года, когда были собраны данные (например, 1983 - model year).
6. Визуализируйте распределение количества цилиндров (cylinders) с помощью столбчатой диаграммы.

```
import os
import pandas as pd
import matplotlib.pyplot as plt
from datetime import datetime

current_dir = os.path.dirname(os.path.abspath(__file__))

project_root = os.path.abspath(os.path.join(current_dir, '..', '..', '..', '..'))

file_path = os.path.join(project_root, 'auto-mpg.csv')
print("Читаем файл из:", file_path)

columns = [
    'mpg', 'cylinders', 'displacement', 'horsepower', 'weight',
    'acceleration', 'model_year', 'origin', 'car_name'
]

df = pd.read_csv(
    file_path,
    sep=";",
    names=columns,
    na_values='?',
    header=None
```

```

)

numeric_cols = [
    'mpg', 'cylinders', 'displacement', 'horsepower',
    'weight', 'acceleration', 'model_year', 'origin'
]

for col in numeric_cols:
    df[col] = pd.to_numeric(df[col], errors='coerce')

df['model_year'] = 1900 + df['model_year']

df['car_name'] = df['car_name'].astype(str)

df[numeric_cols] = df[numeric_cols].fillna(df[numeric_cols].mean())

current_year = datetime.now().year
df['age'] = current_year - df['model_year']

pd.set_option('display.max_columns', None)

print("\nТипы данных:")
print(df.dtypes)

print("\nКоличество пропусков:")
print(df.isnull().sum())

print("\nОсновные статистические показатели:")
print(df.describe())

print("\nПример новых данных (model_year и age):")
print(df[['model_year', 'age']].head())

# Зависимость расхода топлива от веса
fig1, ax1 = plt.subplots(figsize=(8,6))
ax1.scatter(df['weight'], df['mpg'], alpha=0.7)
ax1.set_xlabel("Вес")
ax1.set_ylabel("MPG (миль на галлон)")
ax1.set_title("Зависимость расхода топлива от веса автомобиля")
ax1.grid(True)
plt.show()
plt.close(fig1)

# 2. Распределения мощности
fig2, ax2 = plt.subplots(figsize=(8,6))
ax2.hist(df['horsepower'], bins=30, color='skyblue', edgecolor='black')
ax2.set_xlabel("Horsepower")
ax2.set_ylabel("Количество автомобилей")

```

```

ax2.set_title("Распределение мощности автомобилей")
ax2.grid(True)
plt.show()
plt.close(fig2)

# Boxplot для сравнения MPG по количеству цилиндров
fig3, ax3 = plt.subplots(figsize=(8,6))
df.boxplot(column='mpg', by='cylinders', ax=ax3)
ax3.set_xlabel("Цилиндры")
ax3.set_ylabel("MPG")
ax3.set_title("MPG по количеству цилиндров")
plt.suptitle("")
ax3.grid(True)
plt.show()
plt.close(fig3)

# Распределения возраста автомобилей
fig4, ax4 = plt.subplots(figsize=(8,6))
ax4.hist(df['age'], bins=20, color='lightgreen', edgecolor='black')
ax4.set_xlabel("Возраст автомобиля (лет)")
ax4.set_ylabel("Количество автомобилей")
ax4.set_title("Распределение возраста автомобилей")
ax4.grid(True)
plt.show()
plt.close(fig4)

# Зависимость MPG от возраста
fig5, ax5 = plt.subplots(figsize=(8,6))
ax5.scatter(df['age'], df['mpg'], alpha=0.7, color='orange')
ax5.set_xlabel("Возраст автомобиля (лет)")
ax5.set_ylabel("MPG (миль на галлон)")
ax5.set_title("Зависимость расхода топлива от возраста автомобиля")
ax5.grid(True)
plt.show()
plt.close(fig5)

```

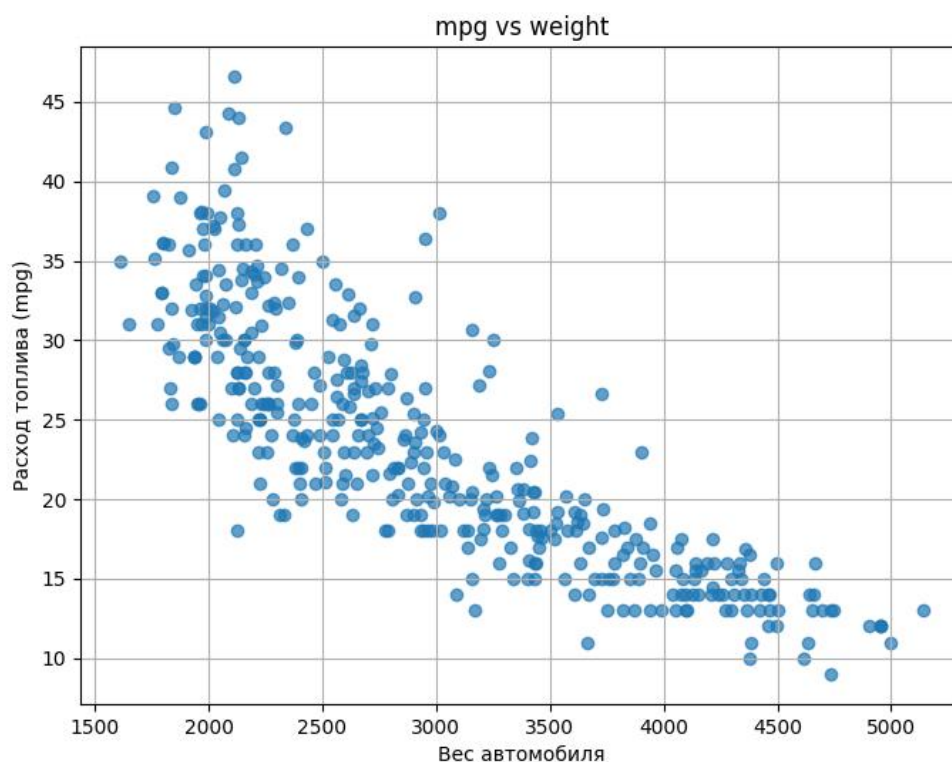
Типы данных:		Количество пропусков:	
mpg	float64	mpg	0
cylinders	int64	cylinders	0
displacement	float64	displacement	0
horsepower	float64	horsepower	0
weight	float64	weight	0
acceleration	float64	acceleration	0
model_year	int64	model_year	0
origin	int64	origin	0
car_name	object	car_name	0
dtype: object		dtype: int64	

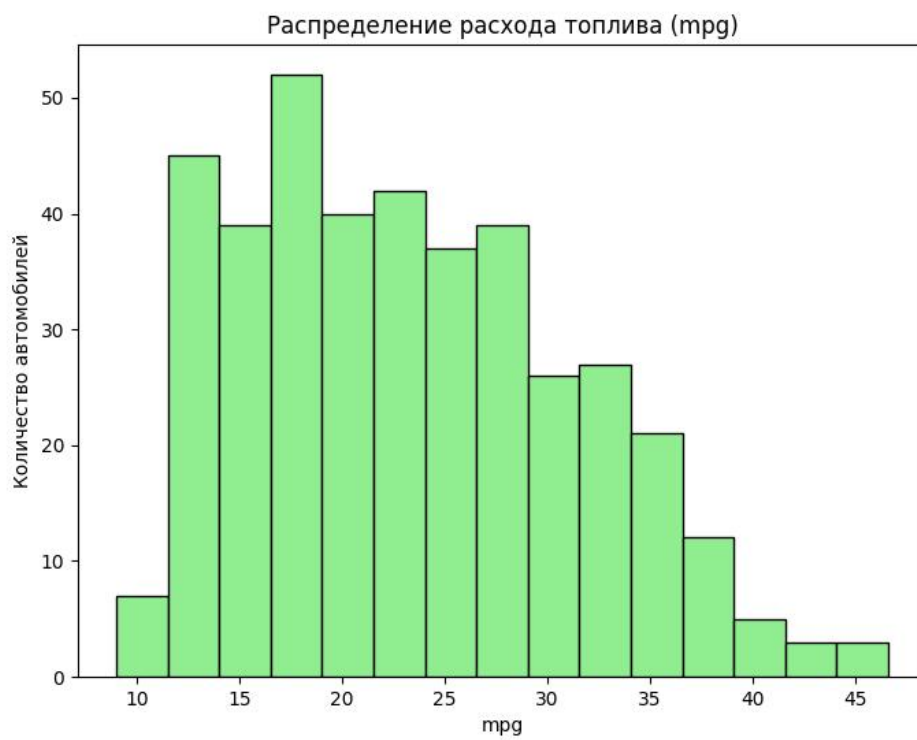
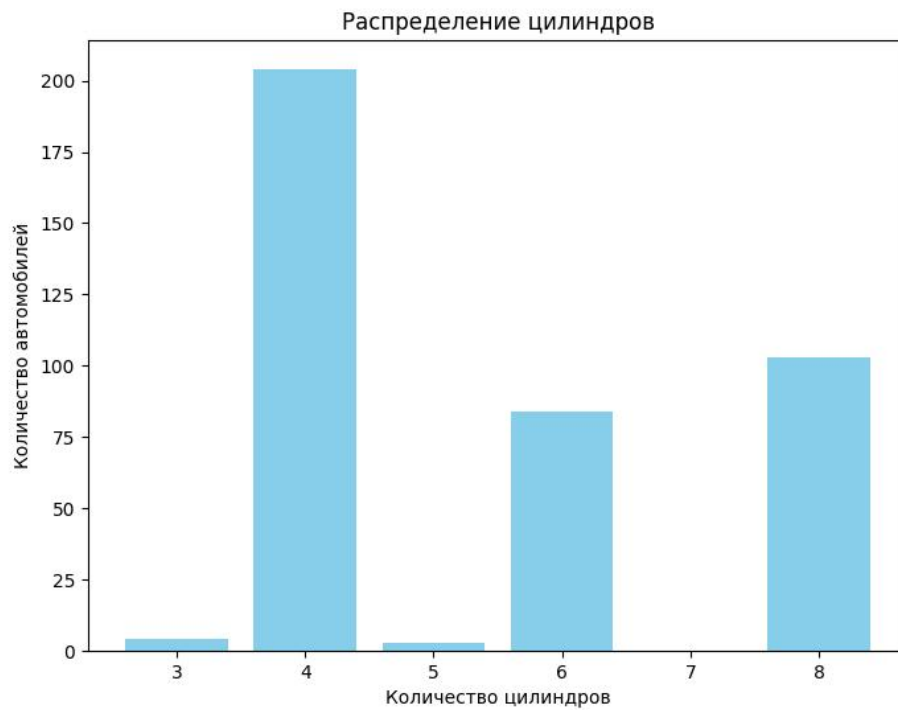
Основные статистические показатели:

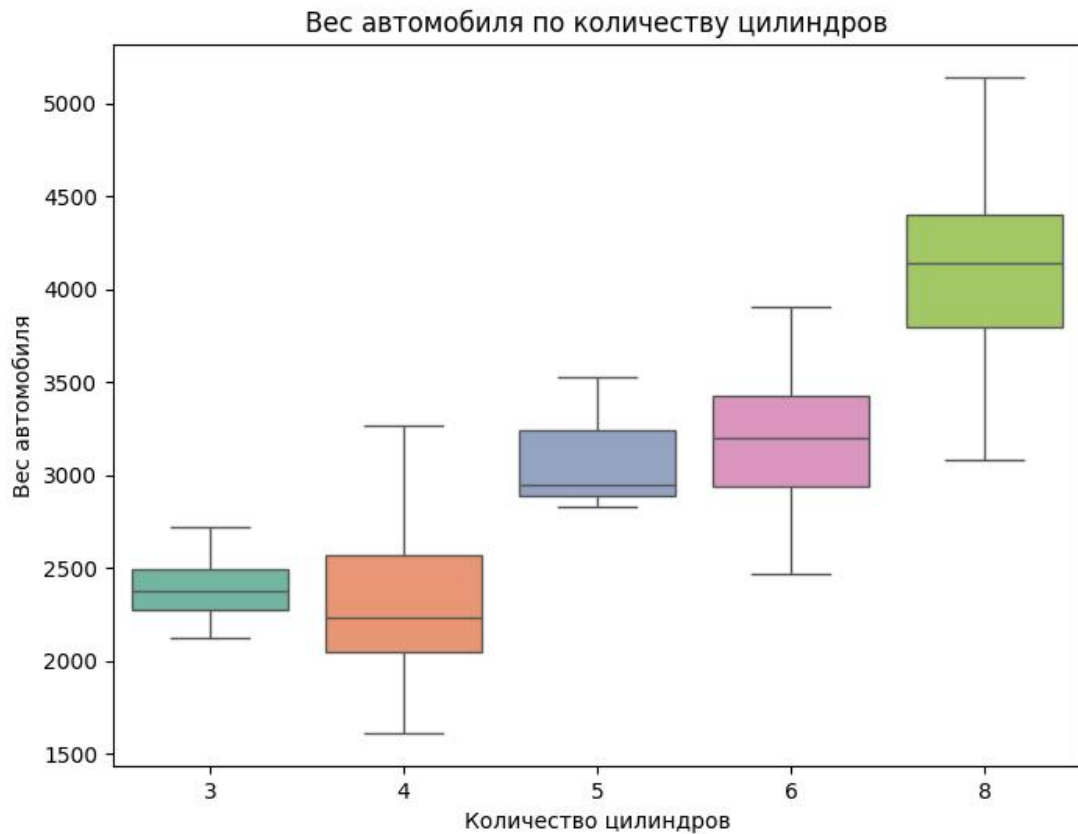
	mpg	cylinders	displacement	horsepower	weight \
count	399.000000	399.000000	399.000000	399.000000	399.000000
mean	23.514573	5.454774	193.425879	104.469388	2970.424623
std	7.806159	1.698866	104.138764	38.151168	845.777234
min	9.000000	3.000000	68.000000	46.000000	1613.000000
25%	17.500000	4.000000	104.500000	76.000000	2224.500000
50%	23.000000	4.000000	151.000000	95.000000	2807.000000
75%	29.000000	8.000000	262.000000	125.000000	3607.000000
max	46.600000	8.000000	455.000000	230.000000	5140.000000

	acceleration	model_year	origin	age
count	399.000000	399.000000	399.000000	399.000000
mean	15.568090	1976.010050	1.572864	48.989950
std	2.754222	3.692978	0.801047	3.692978
min	8.000000	1970.000000	1.000000	43.000000
25%	13.850000	1973.000000	1.000000	46.000000
50%	15.500000	1976.000000	1.000000	49.000000
75%	17.150000	1979.000000	2.000000	52.000000
max	24.800000	1982.000000	3.000000	55.000000

Графики:







**Вывод:** в результате выполнения данной лабораторной работы получили практические навыки работы с данными с использованием библиотек Pandas для манипуляции и Matplotlib для визуализации. Научились выполнять основные шаги предварительной обработки данных, такие как очистка, нормализация и работа с различными типами признаков.