Машинное обучение и интеллектуальный анализ данных

Семинар 3

Г.А. Ососков*, О.И. Стрельцова*, Д.И. Пряхина*, Д.В. Подгайный*, А.В. Стадник*, Ю.А. Бутенко*

Государственный университет «Дубна»

***Лаборатория информационных технологий, ОИЯИ**

Дубна, Россия

Исследовательский анализ данных

Задача: исследовать данные из файла rus_leader_heights.csv, где представлен список некоторых правителей России (от Древней Руси до настоящего времени) и их рост (в см).

	name	height(sm)
0	Aleksandr I	185.0
1	Aleksandr II	185.0
2	Aleksandr III	190.0
3	Aleksandr Nevskiy	165.0
4	Aleksei Tishaishiy	180.0

1. Подключение библиотек

```
import pandas as pd
import numpy as np
from scipy import stats
```

2. Загрузка исходных данных

```
data = pd.read_csv("data/rus_leader_heights.csv")
print(data.head())
```

3. Извлечение метрических данных

```
height = np.array(data["height(sm)"])
```

Исследовательский анализ данных

Задача: исследовать данные из файла rus_leader_heights.csv, где представлен список некоторых правителей России (от Древней Руси до настоящего времени) и их рост (в см).

4. Изучение метрических данных

```
# Проверяем, есть ли пустые значения
np.isnan(height)
np.count_nonzero(np.isnan(height))
# Удаляем пустые значения (nan)
height = height[~np.isnan(height)]
```

5. Вычисление статистических показателей:

- среднее значение;
- медиана;
- дисперсия;
- стандартное отклонение;
- мода;
- минимальное и максимальное значения.

```
# Меры центральной тенденции (мода, медиана, среднее)
print("Mean of heights =", height.mean())
print("Mediane of heights =", np.median(height))
print("Variance of heights =", height.var())
print("Standard Deviation of height =", height.std())
print("Mode of height:", stats.mode(height))
print("Minimum height =", height.min())
print("Maximum height =", height.max())
```

Исследовательский анализ данных

Задача: исследовать данные из файла rus_leader_heights.csv, где представлен список некоторых правителей России (от Древней Руси до настоящего времени) и их рост (в см).

6. Визуализация распределения данных

```
# Подключаем библиотеки для визуализации
import matplotlib.pyplot as plt
import seaborn as sns
```

Гистограмма

```
plt.hist(height)
plt.title("Height Distribution of Leaders of Russia")
plt.xlabel("Height(sm)")
plt.ylabel("Number")
plt.show()
```

Диаграмма плотности

```
sns.histplot(height, kde=True, bins=10)
plt.ylabel("Number")
plt.show()
```

Вычисление скошенности и эксцесса

```
from scipy.stats import kurtosis, skew
print('Skewness of normal distribution (should be 0): {}'.format( skew(height) ))
print('Excess kurtosis of normal distribution (should be 0): {}'.format( kurtosis(height) ))
```

Исследовательский анализ данных

Задача: исследовать данные из файла rus_leader_heights.csv, где представлен список некоторых правителей России (от Древней Руси до настоящего времени) и их рост (в см).

6. Визуализация распределения данных

```
# Подключаем библиотеки для визуализации
import matplotlib.pyplot as plt
import seaborn as sns
```

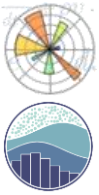

Диаграмма размаха (ящик с «усами»)

```
plt.boxplot(height)
plt.ylabel("Height(sm)")
plt.show()
```

Вычисление квартилей

```
print("25th percentile =", np.percentile(height, 25))
print("50th percentile =", np.percentile(height, 50))
print("75th percentile =", np.percentile(height, 75))
```

7. Удаление выбросов

```
filtered_data = data[ (data['height(sm)'] >= 120) & (data['height(sm)'] <= 210) ]
print(filtered_data)
```

pandas – библиотека Python для обработки и анализа данных*

Формат данных CSV (comma-separated values)

Теоретический материал к заданию: **построить гистограмму роста правителей.**

Написать собственную функцию, реализующую вычисление гистограммы (аналог функции **np.histogram**, с которой сравнить полученные результаты)

Воспользуйтесь:

- **np.linspace(a,b,N)** – генерация **N** чисел, равномерно распределенных в интервале от **a** до **b**
- **np.zeros_like(X)** – заполнения массива **X** нулями
- **np.searchsorted(arr, num)** – поиск индекса в отсортированном массиве **arr**, на место которых необходимо вставить элемент **num**, чтобы порядок следования элементов был сохранен
- **np.add.at(arr,k,t)** – вставляет элемент **k** в массив **arr** на место **k**

*Python Data Analysis Library: https://pandas.pydata.org/

Математические заметки

Новое задание в LMS: Подготовить Jupyter Notebook "**Математические заметки**", который будет содержать теоретическую справку по некоторым математическим темам:

- основные элементарные функции (см. "Математика 1.pdf") - **04.03.2025**
- минимизация функции ошибки (см. "Математика 2.pdf") - **04.03.2025**
- типы распределения вероятностей (см. "Математика 3.pdf") - **11.03.2025**

Файл должен быть оформлен по аналогии с отчетами по заданиям основных проектов.

Работа будет дополняться в течение семестра. **Сроки сдачи заданий указаны у каждой темы.**

Файл загружать в формате pdf!