



Домашнее задание: Визуализация табличных данных в Python

Домашнее задание

Цель

Сделать визуальное представление табличных данных с использованием библиотек `matplotlib`, `seaborn` и `plotly`, а также научиться выбирать тип визуализации в зависимости от задачи анализа.

Задание

Шаг 1. Подготовка данных

1. Сделать загрузку набора данных с клиническими случаями:
 - использовать данные из предыдущих домашних заданий.
2. Сделать первичный осмотр данных.
3. Приложить вывод первых строк таблицы.

Шаг 2. Связь `pandas` и визуализации

1. Сделать не менее **двух графиков**, используя встроенные возможности `pandas`:
 - распределение возраста пациентов;
 - количество клинических случаев по диагнозам **или** исходам лечения.
2. Использовать метод `.plot()`.
3. Приложить полученные графики.

Шаг 3. `Matplotlib`

1. Сделать не менее **двух визуализаций** с помощью `matplotlib`, например:
 - линейный график;
 - столбчатую диаграмму;

- гистограмму.
2. Добавить:
 - заголовок графика;
 - подписи осей.
 3. Приложить результат.

Шаг 4. Seaborn

1. Сделать не менее **двух визуализаций** с помощью `seaborn`, например:
 - `boxplot` возраста пациентов по диагнозам;
 - `countplot` количества случаев по врачам;
 - `heatmap` агрегированных данных.
2. Использовать данные из `pandas.DataFrame`.
3. Приложить результат.

Шаг 5. Plotly

1. Сделать не менее **одной интерактивной визуализации** с помощью `plotly`, например:
 - интерактивный столбчатый график;
 - интерактивную диаграмму распределения;
 - `scatter plot`.
2. Проверить, что график масштабируется и отображает подсказки при наведении.
3. Приложить результат (скриншот или ссылку, если используется Colab).

Шаг 6. Сравнение визуализаций

1. Сделать краткое сравнение (текстом):
 - какой тип визуализации лучше подходит для анализа распределений;
 - какой — для сравнения категорий;
 - в чём отличие интерактивных графиков от статических.
2. Приложить текстовое описание (5–7 предложений).

Подсказки по ключевым частям

- `pandas` : `.plot(kind=...)`

- `matplotlib` : `plt.plot()` , `plt.bar()` , `plt.hist()`
- `seaborn` : `sns.countplot()` , `sns.boxplot()` , `sns.heatmap()`
- `plotly` : `plotly.express`
- Не забывайте:
 - подписывать оси;
 - добавлять заголовки;
 - выбирать подходящий тип графика под задачу.

Что проверить перед отправкой (чек-лист)

- Сделаны визуализации с помощью `pandas` .
- Сделаны визуализации с помощью `matplotlib` .
- Сделаны визуализации с помощью `seaborn` .
- Сделана хотя бы одна интерактивная визуализация с помощью `plotly` .
- Графики читаемы и корректно подписаны.
- Приложены все результаты.
- Код выполняется без ошибок.

Советы по улучшению работы

- Делайте отдельную ячейку под каждый график.
- Подписывайте, **что именно показывает визуализация**.
- Старайтесь не перегружать графики лишними элементами.
- Подумайте, какие графики были бы полезны врачу или аналитику.

Ответ

Ссылка на решение:

<https://colab.research.google.com/drive/1qHe5eDsY1mSGO8xptF6HJGmlspE6ZS9O?usp=sharing>