

Лабораторная работа № 3

Тема: «Визуализация данных на языке Python с помощью библиотеки seaborn»

Цель работы: изучить основы работы с библиотекой seaborn и отдельные виды графиков применительно к реальным данным.

Теоретическая справка

Диаграммы Seaborn создают два разных объекта типов:

- FacetGrid - можно создавать подграфики
- AxesSubplot - только одна диаграмма

Чтобы построить столбчатую диаграмму в Seaborn нужно вызвать функцию `barplot()`. Параметры данной функции таковы:

- `data=dataframe`,
- `x` – ось абсцисс
- `y` – ось ординат
- `hue` - тон

Если поменять местами координаты `x` и `y` диаграмма превратится в линейчатую (столбы будут вытянуты вдоль горизонтальной оси)

Чтобы построить столбчатую диаграмму в Seaborn нужно вызвать функцию `scatterplot()`. Параметры данной функции таковы:

- `data=dataframe`,
- `x` – ось абсцисс
- `y` – ось ординат
- `hue` - тон
- `size` – размер точки

Чтобы построить тепловую карту в Seaborn нужно вызвать функцию `heatmap()`. Параметры данной функции таковы:

- `vmin`, `vmax` — устанавливают диапазон значений, которые служат основой для цветовой карты (`colormap`).
- `cmap` — определяет конкретную `colormap`, которую мы хотим использовать (ознакомьтесь с полным диапазоном цветовых палитр здесь).
- `center` — принимает число с плавающей точкой, для центрирования цветовой карты; если `cmap` не указан, используется `colormap` по умолчанию; если установлено значение `True` — все цвета заменяются на синий.
- `annot` — при значении `True` числовые значения отображаются внутри ячеек.
- `cbar` — если установлено значение `False`, цветовая шкала не выводится.

«Визуальная аналитика», Киреев В.С.

Для создания ящичковой диаграммы нужно вызвать функцию `boxplot()`. Параметры данной функции таковы:

- `data=dataframe`,
- `x` – ось абсцисс
- `y` – ось ординат
- `order` - используется для отображения блочных диаграмм в определенном порядке.

Для создания скрипичной диаграммы нужно вызвать функцию `violinplot()`.

Для создания категориальной диаграммы рассеивания можно использовать функцию `swarmplot()`

При помощи объекта `FacetGrid`, библиотека `Seaborn` позволяет обрабатывать данные и строить на их основе групповые взаимосвязанные графики.

- `col` – группировка по колонкам
- `hue` задает каждому региону собственный тон.
- `col_wrap` ширина области `Figure` ограничивается `n` диаграммами. По достижении этого предела следующие графики будут построены на новой строке.

метод `map` используется для подготовки сетки перед выводом на экран. Тип диаграммы передается в первом аргументе, а в качестве осей служат переменные из следующих позиционных параметров. Например:

1. `g = sns.FacetGrid(df, col="Region", hue="z", col_wrap=5)`
2. `g.map(sns.scatterplot, "x", "y")`

Для изменения палитры графиков можно использовать конструкцию вида:

1. `colors = ['#78C850', '#F08030', '#6890F0', '#F8D030', '#F85888', '#705898', '#98D8D8']`
2. `sns.boxplot(x= df["x"], y= df["y"], palette=colors)`

Несмотря на то, что `Seaborn` автоматически подписывает оси `X` и `Y`, можно изменить подписи с помощью функций `set_title()` и `set_label()` после создания объекта `axes`. Надо просто передать название, которое хотим дать графику, функции `set_title()`. Для того чтобы подписать оси, используется функция `set()` с аргументами `xlabel` и `ylabel` или функции-обертки `set_xlabel()/set_ylabel()`

Для изменения стиля графиков используется конструкция вида: `sns.set_style("whitegrid")`.

Для поворота меток можно использовать подход в сочетании с `pyplot` –

1. `sns.catplot(x="x", y="y", data=df, kind="box")`
2. `plt.xticks(rotation=90)`
3. `plt.show()`

Самостоятельное задание

1. Импортировать данные lab3.csv в соответствии с кодом:

```
import pandas as pd
df=pd.read_csv('/content/lab3.csv')
```

2. Построить линейный график зависимости суммы всех заказов в день, от даты (год-месяц-день). Здесь и далее пользоваться библиотекой seaborn для построения графиков. Библиотеку matplotlib использовать по необходимости, для задания размера графика.
 - 2.1. Повернуть подписи по оси x (даты) под углом 45 градусов.
 - 2.2. Добавить название диаграммы
3. Построить ящичковую диаграмму по распределению суммы заказа в зависимости от года заказа
 - 3.1. Добавить параметр тон (hue) по полу клиента
4. Построить гистограмму распределения суммы заказа в зависимости от пола клиента
 - 4.1. Изменить цвет гистограммы на ярко-красный цвет, добавить темно-серую заливку области построения графика
5. Построить столбчатую диаграмму зависимости максимальной суммы заказа от даты (месяц, без года)
 - 5.1. Изменить ориентацию диаграммы на горизонтальную столбчатую
 - 5.2. Добавить группировку столбцов по полу
6. Построить тепловую карту по таблице – средняя сумма заказа по дате- (год-месяц) и полу клиента
7. Построить категориальную диаграмму рассеяния зависимости суммы заказа от пола
 - 7.1. Изменить стиль графика
8. Построить фасетчатую диаграмму, с гистограммами распределения суммы заказа, в зависимости от года заказа (столбцы) и пола клиента (строки).
 - 8.1. Заменить гистограммы на скрипичные графики
9. Построить совместную диаграмму зависимости суммы заказа мужчин от женщин, в день (по краям графика должны идти гистограммы распределения сумм заказов)(предварительно нужно создать таблицу, где индекс – день, а два столбца соответствуют суммам заказов мужчин отдельно и женщин отдельно, на каждый день)
10. Построить диаграмму двумерной плотности распределения сумм заказов мужчин и женщин в день
 - 10.1. Построить свою палитру цвета и применить ее к графику