# Лабораторна робота №2

## Візуалізації даних в Python

***Мета роботи:*** набути навичок роботи з модулями Matplotlib та Seaborn, опанувати основні методи бібліотек Seaborn і Matplotlib.pyplot, навчитися проводити візуальний аналіз даних на представленому набір даних

***Література***

*Галерея прикладів різної графіки в matplotlib -*[*https://matplotlib.org/gallery.html*](https://matplotlib.org/gallery.html) *Приклади створення геометричних фігур і форм –*

*https://matplotlib.org/examples/shapes\_and\_collections/artist\_reference.html Повний список команд для pyplot - https://matplotlib.org/api/pyplot\_summary.html*

*Документація: https://seaborn.pydata.org/generated/seaborn.heatmap.html https://seaborn.pydata.org/generated/seaborn.JointGrid.html https://seaborn.pydata.org/generated/seaborn.jointplot.html*

### Зміст роботи

#### **Завдання 1.** Провести розвідувальний аналіз даних. Вхідні дані знаходяться у csv-файлі: med.csv

Datasetсформований з реальних даних, і в ньому використовуються ознаки, які можна розбити на 3 групи:

*Об’єктивні ознаки:*

* Вік (age)
* Зріст (height)
* Вага (weight)
* Стать (gender)

*Результати вимірювання:*

* Артеріальний тиск верхній і нижній (ap\_hi, ap\_lo)
* Холестерин (cholesterol)
* Глюкоза (gluc)

*Суб'єктивні ознаки (зі слів пацієнта):*

* Куріння (smoke)
* Вживання алкоголю (alco)
* Фізична активність (active)

Вік заданий в днях. Значення показників холестерину і глюкози представлені одним з трьох класів: *норма, вище норми, значно вище норми*. Значення суб’єктивних ознак - бінарні.

З бібліотек знадобляться :

import numpy as np import pandas as pd import matplotlib.ticker import matplotlib.pyplot as plt import seaborn as sns

# ігноруємо warnings import warnings

warnings.filterwarnings("ignore")

Провести налаштування зовнішнього вигляду графіків у seaborn:

sns.set\_context( "notebook", font\_scale = 1.5, rc = {

"figure.figsize" : (12, 9),

"axes.titlesize" : 18

}

)

Зчитати дані з CSV-файлу в об'єкт pandas *DataFrame*. df = pd.read\_csv('mlbootcamp5\_train.csv', sep=';',index\_col='id') Подивитися на перші 5 записів та розмір Dataset:

df.head() print(df.shape) Результат:



В рамках завдання для простоти необхідно буде працювати з вибіркою даних, що має кількісні і категоріальні ознаки. Чистити дані від викидів і помилок НЕ ПОТРІБНО, крім тих випадків, де про це явно зазначено.

Всі візуалізації рекомендуємо проводити за допомогою бібліотеки *Seaborn.*

***Проведемо невеликий EDA (розвідувальний аналіз даних)***

*Розвідувальний аналіз займається попереднім експрес-аналізом даних шляхом їх перетворення та/або представлення у зручному вигляді: графічному, табличному, за допомогою схем, діаграм тощо.*

*Термін «розвідувальний аналіз» (*exploratory data analysis *—* EDA*) був уперше введений Дж. Тьюкі, він же сформулював основні його завдання:*

* *максимальне проникнення в дані;*
* *вибір найважливіших ознак:*
* *аналіз основних структур;*
* *виявлення відхилень і аномалій;*
* *перевірка основних гіпотез щодо законів розподілу і взаємозв’язків;*
* *апробація моделей.*

*Отже, на етапі розвідувального аналізу формується уявлення про тип даних, оцінюється їхня однорідність, з’ясовується структура об’єкта моделювання, виявляються взаємозв’язки між ознаками. За допомогою дескриптивних статистик описуються й узагальнюються основні властивості об’єкта моделювання, частотний аналіз і графічна візуалізація допомагають визначитися щодо методів подальшого аналізу і моделей, які треба застосувати, а також яких результатів можна очікувати. Без розвідувального аналізу даних моделювання буде наосліп.*

Для початку завжди потрібно подивитися на значення, які приймають змінні.

Переведемо дані в «Long Format» - представлення та проведемо візуалізацію даних за допомогою *factorplot* значень, які приймають категоріальні змінні.

df\_uniques = pd.melt(frame=df, value\_vars=['gender','cholesterol'

,'gluc', 'smoke', 'alco','active', 'cardio'])

df\_uniques = pd.DataFrame(df\_uniques.groupby(['variable','value']

)['value'].count())\

.sort\_index(level=[0, 1]) \

.rename(columns={'value': 'count'}) \

.reset\_index()

sns.catplot(x='variable', y='count', hue='value', data=df\_uniques , kind='bar', aspect=3)

 Результат:



На графіку можна побачити, що класи цільової змінної *cardio* збалансовані, відмінно!

Можна також розбити елементи вибірки за значеннями цільової змінної: іноді на таких графіках можна відразу побачити найбільш значиму ознаку.

df\_uniques = pd.melt(frame=df, value\_vars=['gender','cholesterol'

,'gluc','smoke', 'alco',

'active'],id\_vars=['cardio']) df\_uniques = pd.DataFrame(df\_uniques.groupby(['variable', 'value'

,

'cardio'])['value'].count()) \

.sort\_index(level=[0, 1]) \

.rename(columns={'value': 'count'}) \

.reset\_index()

sns.catplot(x='variable', y='count', hue='value', col='cardio', data=df\_un iques, kind='bar')

Результат:



По отриманим графікам можна побачити, що в залежності від цільової змінної (*cardio*) сильно змінюється розподіл холестерину і глюкози.

Статистику за унікальними значеннями ознак можна отримати за допомогою наступного коду:

for c in df.columns: n = df[c].nunique() print(c) if n <= 3:

print(n, sorted(df[c].value\_counts().to\_dict().items())) else:

print(n) print(10 \* '-')

Результат:



Разом:

* П’ять кількісних ознак (без id)
* Сім категоріальних

#### **Завдання 2.** Провести візуальний аналіз

##### Кореляційна матриця

Для того щоб краще зрозуміти ознаки в Dataset, можна порахувати матрицю коефіцієнтів кореляції між ознаками.

***Побудуйте кореляційну матрицю (****heatmap)****. Матриця формується засобами Pandas, зі стандартним значенням параметрів.***

df.corr() або

sns.heatmap(df.corr(), cmap="crest")

***Визначте які дві ознаки найбільше корелюють (за Пірсоном) з ознакою*** *height****?***

##### 2. Розподіл росту людини за гендерною ознакою

В процесі дослідження унікальних значень стать кодується значеннями 1 або 2, визначити хто є хто потрібно було у лабораторній роботі №2, для цього потрібно було використовувати середні значення зросту (або ваги) при різних значеннях ознаки *gender*. Тепер представте те ж саме, але графічно.

***Проведіть візуалізацію даних: зріст і стать*** *(violinplot)****. Використовуйте параметри:***

* *hue* ***- для розбивки за статтю;***
* *scale* ***- для оцінки кількості кожної статі.***

sns.violinplot(data=longformat, x='variable', y='value', hue='gen der', scale='count'); plt.show()

Для коректного відтворення, перетворіть *DataFrame* в «Long Format»представлення за допомогою функції *melt* в *pandas*.

longformat = pd.melt(frame=df, value\_vars='height', id\_vars='gender')

longformat.head()

***Побудуйте на одному графіку два окремих kdeplot росту і вагі*,** окремо для чоловіків і жінок. На ньому різниця буде більш наочною, але не можна буде оцінити кількість чоловіків / жінок.

sns.kdeplot(df[df['gender'] == 1]['height'])

sns.kdeplot(df[df['gender'] == 2]['height'])

plt.show(

##### 3. Рангова кореляція

У більшості випадків достатньо скористатися коефіцієнтом кореляції Пірсона для виявлення закономірностей у даних, але рангова кореляція Спірмена допоможе виявити пари, в яких менший ранг з варіаційного ряду однієї ознаки завжди передує більшому іншого (або навпаки, в разі негативної кореляції).

Рангова кореляція — метод кореляційного аналізу, який використовується для сукупностей невеликого обсягу і для кількісних ознак, якщо їхня сукупність не має нормального розподілу.

Для змінних, що належать до порядкової шкали, або для змінних, що не підкоряються нормальному розподілу, а також для змінних, що належать до інтервальної шкали, замість коефіцієнта Пірсона розраховується рангова кореляція за Спірменом.

Ще одним варіантом рангових коефіцієнтів кореляції є коефіцієнти Кендалла. У цьому методі одна змінна представляється у вигляді монотонної послідовності в порядку зростання величин; іншій змінній привласнюються відповідні рангові місця. Кількість інверсій (порушень монотонності в порівнянні з першим рядом) використовується у формулі для кореляційних коефіцієнтів. Застосування коефіцієнта Кендалла є кращим, якщо у вихідних даних зустрічаються викиди.

**Рангова кореляція Спірмена** (кореляція рангів) — найпростіший спосіб визначення міри зв'язку між факторами.

sns.heatmap(df.corr(method='spearman'))

Назва методу свідчить про те, що зв'язок визначають між рангами, тобто рядами одержаних кількісних значень, ранжованих у порядку зниження або зростання. Треба мати на увазі, що, по-перше, рангову кореляцію не рекомендовано проводити, якщо зв'язок пар менший чотирьох і більший двадцяти; по-друге, рангова кореляція дає змогу визначати зв'язок і в іншому випадку, якщо значення мають напівкількісний характер, тобто не мають числового виразу, відображають чіткий порядок прямування цих величин; по-третє, рангову кореляцію доцільно застосовувати в тих випадках, коли достатньо одержати приблизні дані.

**Побудуйте кореляційну матрицю, використовуючи коефіцієнт Спірмена.**

sns.heatmap(df.corr(method='spearman')); plt.show()

df.corr(method='spearman')

**3.1 Які ознаки найбільше корелюють одна з одною за Спірменом?**

**3.2 Чому значення рангової кореляції у цих ознаках таке велике (відносно)?**

##### 4. Вік

Порахуємо, скільки повних років було респондентам на момент їх занесення в базу.

df['age\_years'] = (df['age'] // 365.25).astype(int)

***Побудуйте Countplot, де на осі абсцис буде відзначений вік, на осі ординат - кількість. Кожне значення віку повинне мати два стовпці, що відповідають кількості осіб кожного класу cardio (здоровий / хворий) даного віку.***

sns.countplot(x='age\_years', hue='cardio', data=df);

* ***В якому віці кількість пацієнтів з ССЗ вперше стає більше, ніж здорових?***

#### **Завдання 3.** Провести розвідувальний і візуальний аналіз для набору даних: Predict students' dropout and academic success з файлу kaggle.csv

(https://www.kaggle.com/datasets/lucamorn/higher-education-predictors-of-student-retention)

## Контрольні запитання

1. Для чого використовується «Long Format»- представлення?
2. За допомогою якого графіка можна дізнатися середнє значення (мат. очікування) і розкид значень (дисперсію) для різних категорій даних.
3. Опишіть основні елементи наступної діаграми і яким чином можна її отримати:



1. Якщо використовувати метод *DataFrame*.*plot()* з параметром *kind = 'bar'*, який вид діаграми можна отримати?
2. Тип графіків *Pie Chart* (Пиріжковий графік) відмінно підходить для відображення часток, які належать частини даних. Опішить метод побудови.
3. Опишіть призначення графіка *Heat Map* (Теплова карта).
4. Для чого призначені функції *plt.show()* і *plt.draw()?*
5. Як можна отримати наступний графік:



1. Який тип графіка можна отримати:

sns.pairplot(data=iris, hue="species")