Лабораторна робота №3

Візуальний аналіз 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
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 візуальний аналіз для набору даних mlbootcamp_train_Soroka.csv

– 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.

В рамках завдання для простоти необхідно буде працювати з вибіркою даних, що має кількісні і категоріальні ознаки.

– Для початку завжди потрібно подивитися на значення, які приймають змінні. 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 * '-')
```

Висновки:

– Скільки кількісних ознак (без id)?
– Скільки категоріальних ознак?

1. Кореляційна матриця

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

2. Проведемо візуалізацію даних значень, які приймають категоріальні змінні

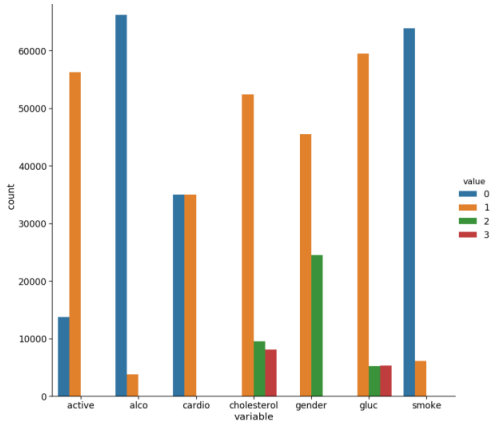
Переведемо дані в «Long Format» - представлення та проведемо візуалізацію даних за допомогою *factorplot* значень, які приймають категоріальні змінні.

```
df_uniques = pd.melt(frame=df, value_vars=['gender','choles
terol','gluc', 'smoke', 'alco','active', 'cardio'])

df_uniques = pd.DataFrame(df_uniques.groupby(['variable','v
alue'])['value'].count())\
 .sort_index(level=[0, 1]) \
 .rename(columns={'value': 'count'}) \
 .reset_index()

sns.catplot(x='variable', y='count', hue='value', data=df_u
niques, kind='bar', aspect=3)
```

Результат:



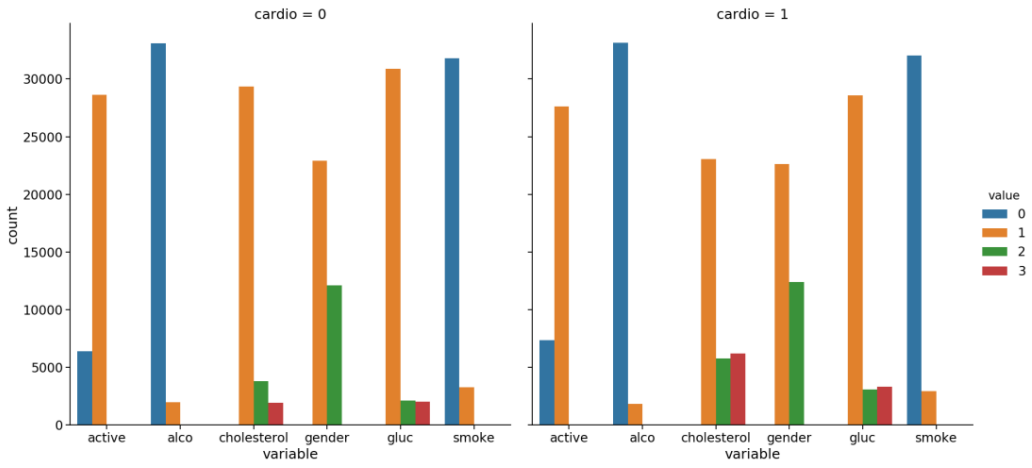
На графіку можна побачити, що класи цільової змінної *cardio* збалансовані, відмінно!

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','choles
    terol','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_uniques, kind='bar')
```

Результат:



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

3. Розподіл росту людини за гендерною ознакою

В процесі дослідження унікальних значень стать кодується значеннями 1 або 2, визначити хто є хто, для цього потрібно використовувати середні значення зросту (або ваги) при різних значеннях ознаки *gender*. Зробіть це і 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
='gender', scale='count');
```

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 ваги, 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'])
```

4. 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.

Побудуємо кореляційну матрицю, використовуючи коефіцієнт Спірмена.

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

або

df.corr(method='spearman')
```

Які ознаки найбільше корелюють одна з одною за Спірменом?

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

5. 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);
```

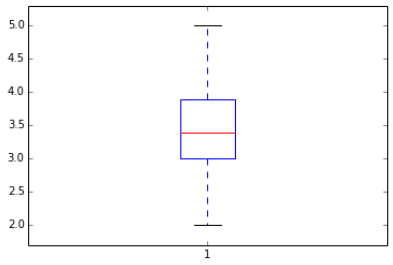
В якому віці кількість пацієнтів з ССЗ вперше стає більше, ніж здорових?

Завдання 3. Провести розвідувальний і візуальний аналіз за варіантами:

1	*https://www.kaggle.com/datasets/equilibriumm/sleep-efficiency*
2	*https://www.kaggle.com/datasets/fredericksalazar/population-world-since-1960-to-2021*
3	*https://www.kaggle.com/datasets/thedevastator/supermarket-ordering-invoicing-and-sales-analysi*
4	*https://www.kaggle.com/datasets/swatikhedekar/price-prediction-of-diamond*
5	*https://www.kaggle.com/datasets/kamilpytlak/personal-key-indicators-of-heart-disease*
6	*https://www.kaggle.com/datasets/parulpandey/palmer-archipelago-antarctica-penguin-data*
7	*https://www.kaggle.com/datasets/themrityunjaypathak/imdb-top-100-movies*
8	*https://www.kaggle.com/datasets/sidtwr/videogames-sales-dataset*
9	*https://www.kaggle.com/datasets/thedevastator/comprehensive-overview-of-52478-goodreads-best-b*
10	*https://www.kaggle.com/datasets/erenakbulut/common-bottlenose-dolphin-behavior*
11	*https://www.kaggle.com/datasets/thedevastator/netflix-top-rated-movies-and-tv-shows-2020-2022*
12	*https://www.kaggle.com/datasets/salimwid/global-billionaire-wealth-and-sources-2002-2023*
13	*https://www.kaggle.com/datasets/ahmettyilmazz/fuel-consumption*
14	*https://www.kaggle.com/datasets/themrityunjaypathak/covid-cases-and-deaths-worldwide*
15	*https://www.kaggle.com/datasets/alifarahmandfar/continuous-groundwater-level-measurements-2023*

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:



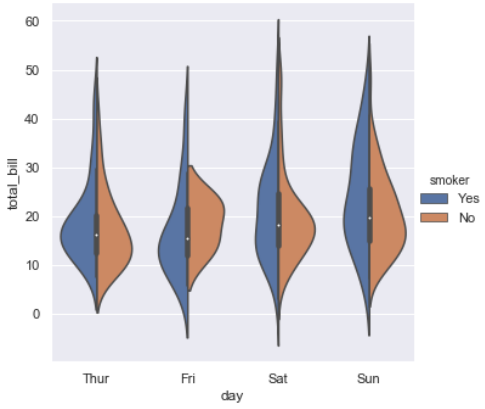
4. Якщо використовувати метод *DataFrame.plot()* з параметром *kind = 'bar'*, який вид діаграми можна отримати?

5. Тип графіків *Pie Chart* (Пиріжковий графік) відмінно підходить для відображення часток, які належать частини даних. Опішить метод побудови.

6. Опишіть призначення графіка *Heat Map* (Теплова карта).

7. Для чого призначені функції *plt.show()* і *plt.draw()?*

8. Як можна отримати наступний графік:



9. Який тип графіка можна отримати:

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