

Лабораторна робота №2 Програмування і алгоритми

У цій лабораторній роботі я Розробив програмний засіб, який відображує за заданим датасетом точки на координатній площині і зберігає зображення в одному з графічних форматів.

Для виконання цієї задачі я використав дві бібліотеки:

- Matplotlib
- Plotly

Ці дві є одні з найпопулярніших бібліотек для малювання різних графіків.

До цієї лабораторної я вже мав досвід роботи саме з Matplotlib коли проходив курси з Data Science від IBM і на платформі Udemu. Тому вирішив додатково попрацювати з Plotly. Відразу скажу, що я для себе відкрив набагато кращий варіант (Plotly) ніж Matplotlib, адже Plotly більш гнучкий саме під твої запити. Я читав навіть декілька статей на цю тему і Plotly робить дивовижні речі.

Залишу посилання де я це читав :) ([Посилання](#)).

Тепер перейду до кода і поясню свої кроки:

```
with open("DS2.txt", 'r') as dataset_file:
    dataset = dataset_file.read()

dataset = dataset.replace(" ", ",")
dataset_list = [x for x in dataset.split("\n")]
dataset_list.remove("")
X = []
Y = []
for el in dataset_list:
    row = el.split(",")
    X.append(int(row[0]))
    Y.append(int(row[1]))
```

Тут я зчитав дані з файлу. Далі замінив пробіли на кому. Оскільки формат файлу .txt то рядки можна було розділити "\n". Ну і далі що залишилось це просто для кожного елемента те що йде перед комою помістити в X як 'x' координату. А те що після коми в Y, як 'y' координату.

```
def figure_plot(lib_num):
    if lib_num == 1:
        plt.figure(figsize=(960/100, 540/100))
        plt.title("Scatter Plot of Points (Matplotlib)")
        plt.xlabel("X-axis")
        plt.ylabel("Y-axis")
        plt.scatter(X, Y, c="black")
        plt.show()
    if lib_num == 2:
        fig = pg.Figure(pg.Scatter(x = X, y = Y, mode="markers",
                                   marker=dict(color=Y, size=5, colorscale="icefire", colorbar=dict(title="Colorbar"))))

        fig.update_layout(
            title='Scatter Plot of Points (Plotly)',
            xaxis_title='X-axis',
            yaxis_title='Y-axis',
            width=960,
            height=540,
            margin=dict(l=40, r=40, t=40, b=40), # Відстані від країв
        )
        fig.show()
```

Тут я створив умови, що якщо я в терміналі напишу 1 то буде виконуватися Matplotlib, а якщо 2 то Plotly:

- В першій умові я спочатку задаю розмір полотна. Далі даю заголовок і назви осей. Далі я малюю датасет за допомогою scatter (plt це імпортований модуль matplotlib.pyplot і для комфортного використання я дав plt бо так в цілому прийнято писати саме для pyplot). Scatter приймає два подібних до масива типа даних. Тобто можна подати як і **DataFrame**, **numpy array**, **tuple** і **list**. На скільки я розумію підходять всі ітеровані об'єкти. Колір точок я вказав як чорний.
- В другій умові я вже пробував погратися з кольорами і градієнтом. Звісно функціонал не обмежується тільки цим, але я намагався не виходити за межі завдання.

Спочатку я створив об'єкт Figure, що приймає scatter plot або інший графік. Scatter приймає X, Y як і минулий варіант з matplotlib. Ну і я вирішив додати градієнт в залежності від значення Y. Ось всі можливі варіанти:

One of the following named colorscales:

```
[ 'aggrnyl', 'agsunset', 'algae', 'amp', 'armyrose', 'balance',
  'blackbody', 'bluered', 'blues', 'blugrn', 'bluyl', 'brbg', 'brwnyl', 'bugn', 'bupu', 'burg', 'burgyl',
  'cividis', 'curl', 'darkmint', 'deep', 'delta', 'dense', 'earth', 'edge', 'electric', 'emrld', 'fall',
  'geyser', 'gnbu', 'gray', 'greens', 'greys', 'haline', 'hot', 'hsv', 'ice', 'icefire', 'inferno', 'jet',
  'magenta', 'magma', 'matter', 'mint', 'mrybm', 'mygbm', 'oranges', 'orrd', 'oryel', 'oxy',
  'peach', 'phase', 'picnic', 'pinkyl', 'piyg', 'plasma', 'plotly3', 'portland', 'prgn', 'pubu', 'pubugn',
  'puor', 'purd', 'purp', 'purples', 'purpor', 'rainbow', 'rdbu', 'rdgy', 'rdpu', 'rdylbu', 'rdylgn',
  'redor', 'reds', 'solar', 'spectral', 'speed', 'sunset', 'sunsetdark', 'teal', 'tealgrn', 'tealrose',
  'tempo', 'temps', 'thermal', 'tropic', 'turbid', 'turbo', 'twilight', 'viridis', 'ylgn', 'ylgnbu', 'ylorbr',
  'ylorrd' ].
```

Все що залишається це додати назви осей і вказати розмір полотна з відступами.

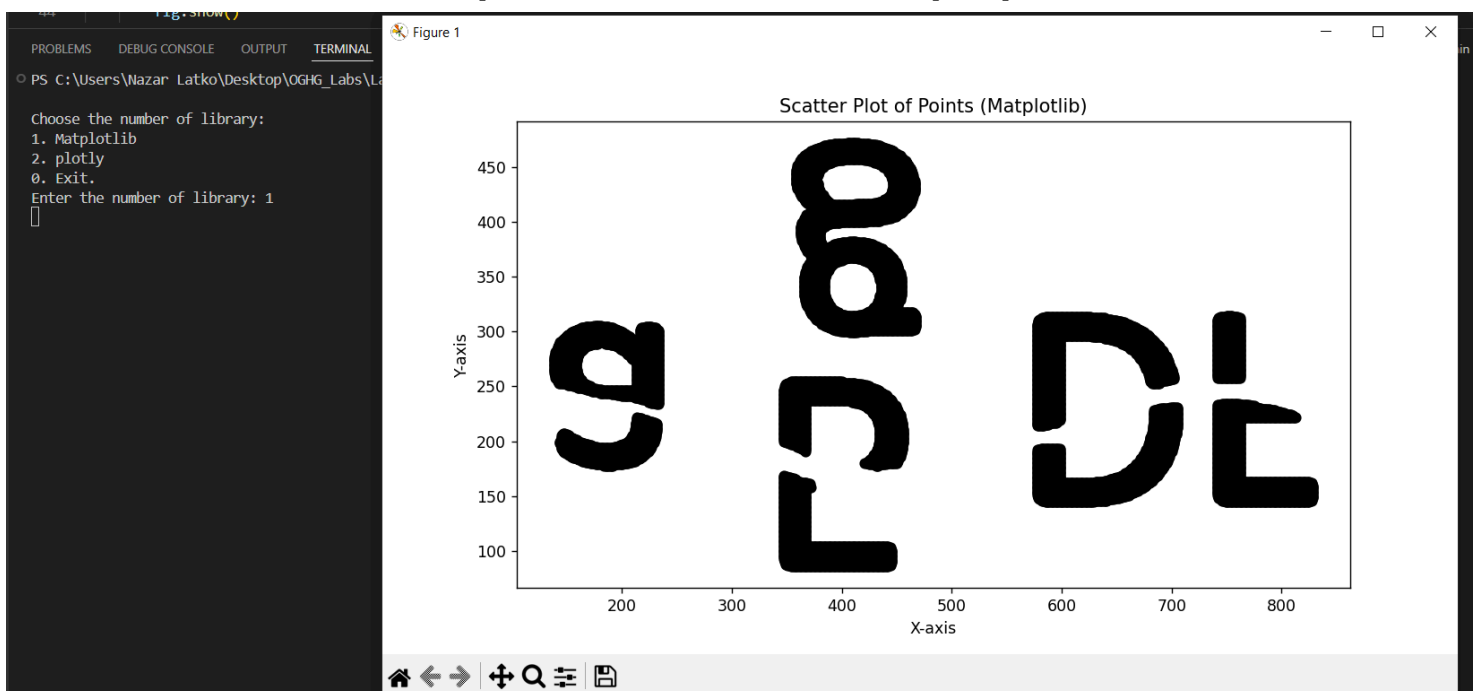
```
def menu():
    while True:
        print('\nChoose the number of library:
1. Matplotlib
2. plotly
0. Exit.')

        try:
            choice = int(input("Enter the number of library: "))
            if choice == 1:
                figure_plot(1)
            elif choice == 2:
                figure_plot(2)
            elif choice == 0:
                print('Thank you')
                break
            else:
                print("Try again!!!")
        except ValueError:
            print("Error!")

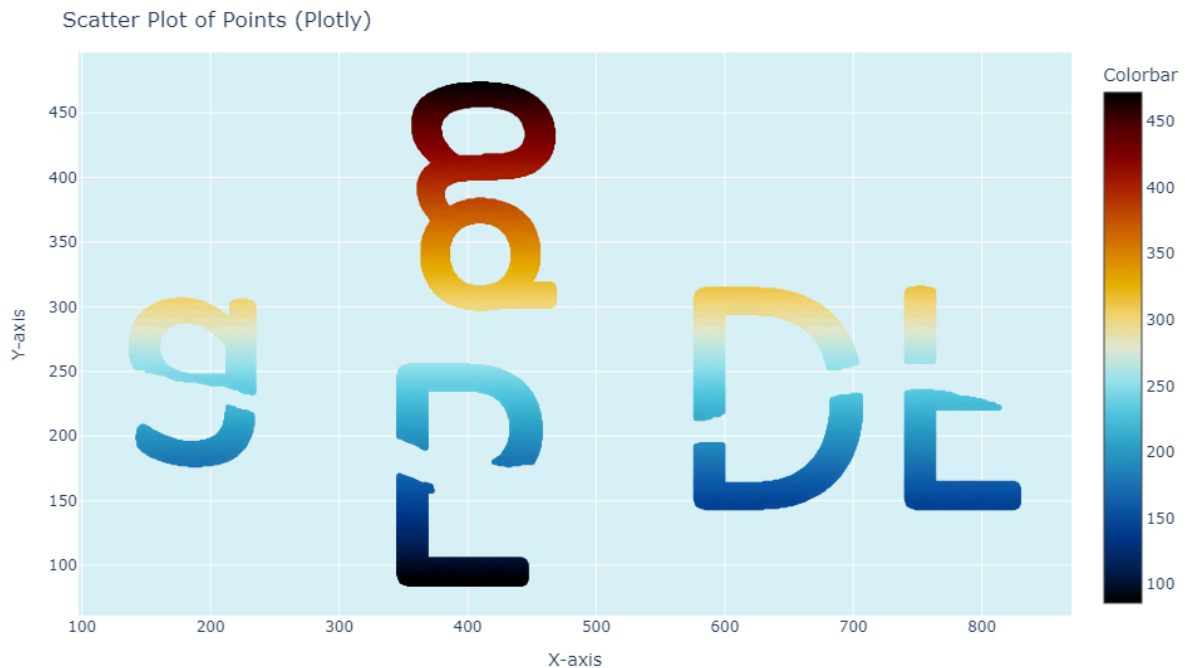
menu()
```

В цій частині я створив, менюшку для того щоб користувач міг вибрати бібліотеку за допомогою якої буде виведений датасет. Ну і звісно додав обробку помилок для комфортного користування.

Приклади виконання програми:



Після закриття віконечка з графіком програма пропонує вибрати знову ті ж опції або вийти з програми.

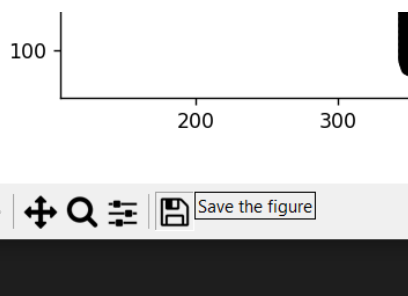


Це виконання другої умови за допомогою Plotly.
Тут я ще додав колір фону.

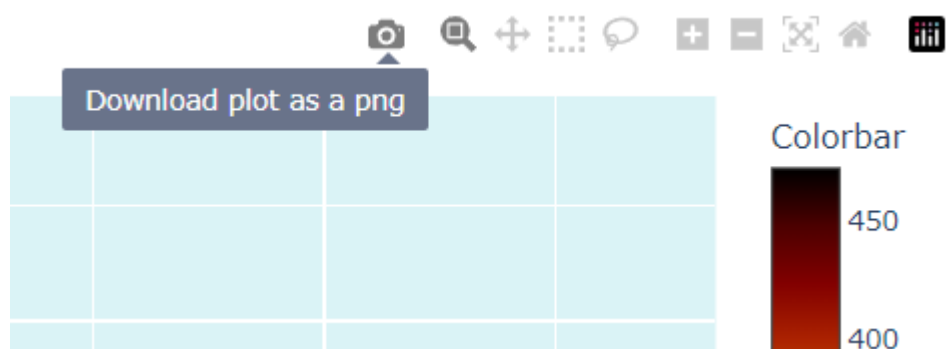
```
fig.update_layout(  
    title='Scatter Plot of Points (Plotly)',  
    xaxis_title='X-axis',  
    yaxis_title='Y-axis',  
    width=960,  
    height=540,  
    plot_bgcolor='rgba(134, 215, 227, 0.3)',  
    margin=dict(l=40, r=40, t=40, b=40), # Відстані від країв  
)  
fig.show()
```

Для кожного варіанту є можливість зберегти фото з графіком.

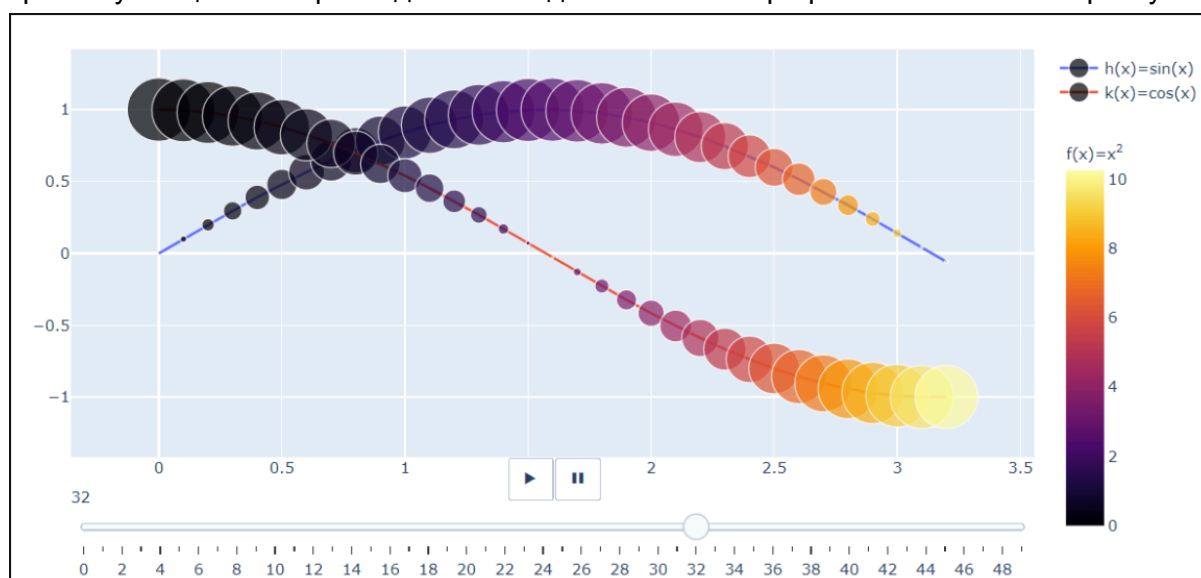
1) Matplotlib



2) Plotly



В цілому за допомогою Plotly можна багато всього зробити і він підходить більше для важливих даних. Наприклад коли робиш аналіз даних і треба подати висновки якісь або презентуєш щось. Наприклад можна подати ось такий графік який я взяв з інтернету:



Тут є ще слайдер який змінює вигляд даних певним чином.

Matplotlib в свою ж чергу підходить для швидких і простих візуалізацій, тобто щоб просто мати уявлення про дані. Причиною цьому є простий синтаксис і зручний API (Чого не можна сказати про Plotly).

Висновок: розробив програмний засіб, який відображує за заданим датасетом точки на координатній площині і зберігає зображення в одному з графічних форматів. Ознайомився з новим для себе способом візуалізації даних за допомогою Plotly. В подальшому планую використовувати саме його щоб вдосконалювати свої навички.

[Посилання на GitHub.](#)