

Лабораторна робота 1 - Знайомство з векторними та матричними операціями на прикладі бібліотеки NumPy

Завдання 1

У машинному навчанні часто використовується логістична функція:

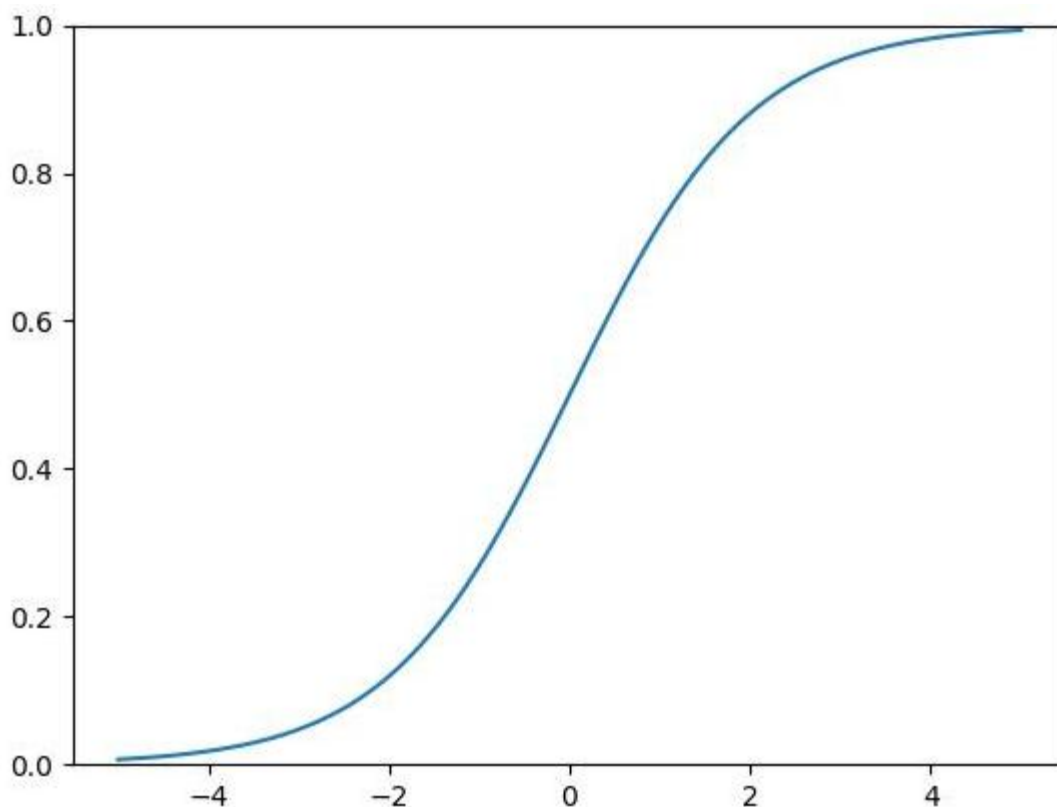
$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

Візуалізувати цю функцію на проміжку $[-5, 5]$

Код програми:

```
1 import math
2 from matplotlib import cm
3 from mpl_toolkits.mplot3d import Axes3D
4 import numpy as np
5 import matplotlib.pyplot as plt
6 from sklearn.preprocessing import PolynomialFeatures
7 def sigmoid(x):
8     return 1/(1+np.exp(-x))
9 x = np.linspace(-5, 5, 100)
10 y = sigmoid(x)
11 plt.plot(x, y)
12 plt.ylim(0, 1)
13 plt.savefig("Task 1 sigmoid.jpg")
14 plt.close()
```

Графік функції:



Завдання 2

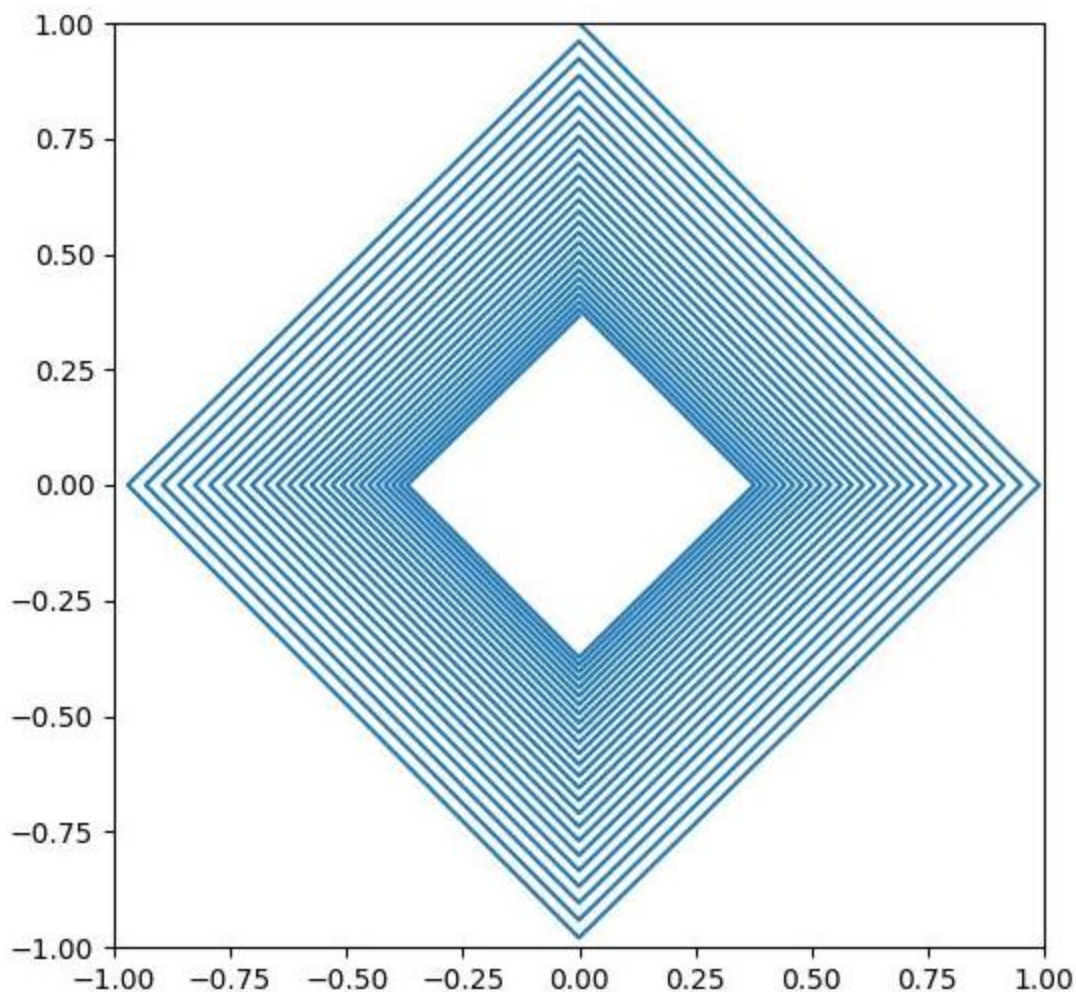
1. Створіть вектор $v = \overline{(0, 1)}$;
2. Створіть матрицю

$M = \begin{bmatrix} \cos \alpha & -\sin \alpha \\ \sin \alpha & \cos \alpha \end{bmatrix}$, де $\alpha \in R$; 3. Обчисліть $v_i, i \in \{1..100\}$, де $v_1 = v, v_{n+1} = v_n \times M \cdot 0.99$;

Код програми:

```
15 ##### TASK 2 #####
16 v=np.array([0,1])# ([i for i in range(100)])
17 alpha=np.pi/2
18 M=np.array([[np.cos(alpha),-np.sin(alpha)],[np.sin(alpha),np.cos(alpha)]])
19 vs=[v]
20 for i in range(100):
21     v=np.dot(v,M)*0.99
22     vs.append(v)
23 vs=np.array(vs)
24 plt.figure(figsize=(6,6))
25 plt.xlim(-1,1)
26 plt.ylim(-1,1)
27 plt.plot(*vs.T)
28 plt.savefig("Task 2.jpg")
29 plt.close()
```

Отримана фігура:



Завдання 3

Одним з алгоритмів навчання з учителем є так звана поліноміальна регресія. Для цього методу, потрібно маючи матрицю

$$X = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{bmatrix}$$

отримати матрицю ступенів:

$$X_p = \begin{bmatrix} x_1^0 & x_1^1 & \dots & x_1^p \\ x_2^0 & x_2^1 & \dots & x_2^p \\ \vdots & \vdots & & \vdots \\ x_m^0 & x_m^1 & \dots & x_m^p \end{bmatrix}$$

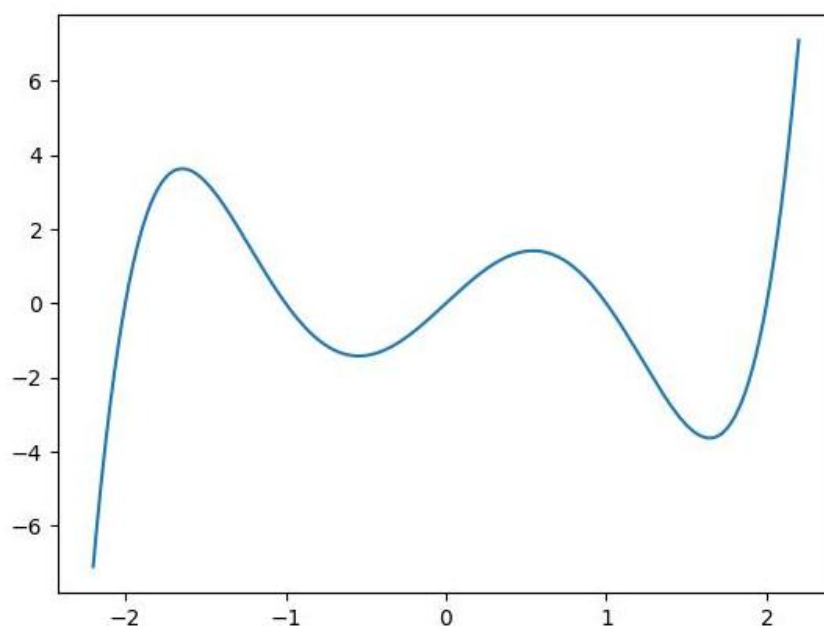
Маючи матрицю X , задану нижче, обчисліть матрицю X_p , для $p = 5$. Розв'язати задачу як з використанням циклу за степенями, так і без нього.

Корисні функції: [stack](#), [concatenate](#).

Код програми:

```
30 ##### TASK 3 #####
31 x = np.linspace(-2.2, 2.2, 1000)[: , np.newaxis]
32 print(f'Shape of x: {x.shape}')
33 X_p = np.zeros((1000,6))
34 for i in range(1000):
35     for j in range(6):
36         X_p[i][j]=pow(x[i],j)
37 X_p = np.concatenate((pow(x,0),pow(x,1),pow(x,2),pow(x,3),pow(x,4),pow(x,5)),1))
38 for i in range(len(x)):
39     for j in range(6):
40         X_p[i][j]=pow(x[i],j)
41 print(f"Shape of X_p {X_p.shape}")
42 assert X_p.shape == (1000, 6)
43 plt.plot(
44     x[... , 0],
45     X_p @ np.array([[0, 4, 0, -5, 0, 1]]).T
46 )
47 plt.savefig("Task 3.jpg")
48 plt.close()
```

Графік функції:



Завдання 4 (*)

Намалюйте графік функції Растрігіна:

$$f(x, y) = 20 + (x^2 - 10 \cos(2\pi x)) + (y^2 - 10 \cos(2\pi y)), \quad x \in [-5, 5], \quad y \in [-5, 5]$$

Використовувати цикли **заборонено**.

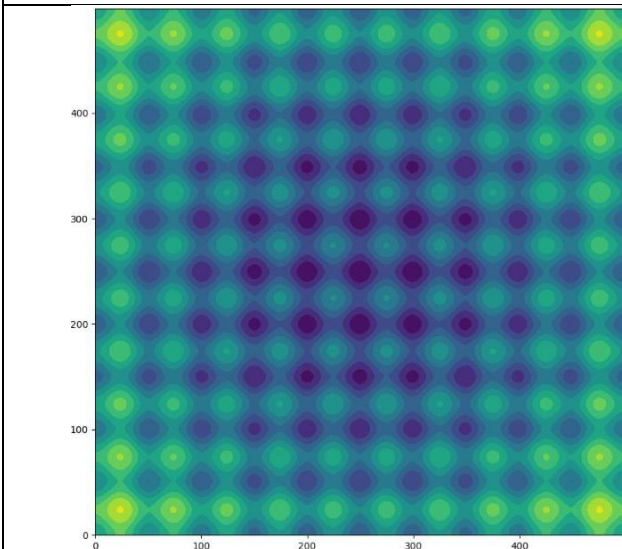
Корисні функції: [linspace](#), [meshgrid](#), [reshape](#).

(Для красивої візуалізації візьміть по 500 кроків для x і y).

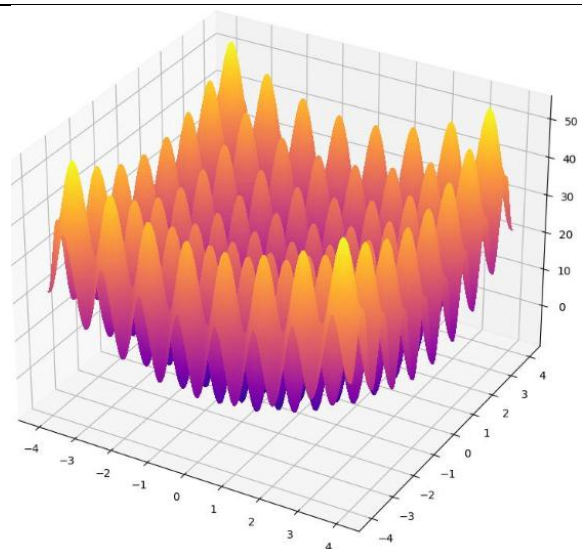
Код програми:

```
49 ##### TASK 4 #####
50 def rastrigin2d(x, y):
51     # return [20+(x**2-10*np.cos(2*math.pi*x))+(y**2-10*np.cos(2*math.pi*y)) for i in x]
52     return 20+(x**2-10*np.cos(2*np.pi*x))+(y**2-10*np.cos(2*np.pi*y))
53 x = np.linspace(-5, 5, 500)
54 y = np.linspace(-5, 5, 500)
55 x,y=np.meshgrid(x,y)
56 plt.figure(figsize=(10, 10))
57 plt.contourf(rastrigin2d(x,y), levels=10)
58 plt.savefig("Task 4 Rastrigin 2d.jpg")
59 plt.close()
60 def rastrigin3d(*X, **kwargs):
61     A = kwargs.get('A', 10)
62     return A + sum([(x**2 - A * np.cos(2 * math.pi * x)) for x in X])
63 X = np.linspace(-4, 4, 500)
64 Y = np.linspace(-4, 4, 500)
65 X, Y = np.meshgrid(X, Y)
66 Z = rastrigin3d(X, Y)
67 fig = plt.figure(figsize=(10, 10))
68 ax = fig.add_subplot(projection = '3d')
69 ax.plot_surface(X, Y, Z, rstride=1, cstride=1, cmap=cm.plasma, linewidth=0, antialiased=False)
70 plt.savefig('Task 4 Rastrigin 3d.jpg')
```

Зображення функції Растрігіна у 2D



Зображення функції Растрігіна у 3D



Висновок: При виконанні даної лабораторної роботи ми ознайомились з основами мови Python, виконали обчислення деяких математичних функцій, навчились зображувати їх графічно за допомогою команд бібліотеки matplotlib. Написання коду здійснювалось з використанням програми Notepad++, а запуск – за допомогою невеликого скрипта у bat-файлі в середовищі Windows.