

Numpy 3.3

Jorge Parra Hidalgo ITIT 13104

1. ¿Qué es Numpy?

NumPy es una biblioteca de Python que proporciona soporte para arreglos multidimensionales y matrices, junto con una amplia colección de funciones matemáticas de alto nivel para operar en estos arreglos.

2. ¿En qué tipo de aplicaciones podemos utilizar Numpy? Mencione y explique brevemente 3 tipos

Puedes utilizar NumPy en una variedad de aplicaciones, incluyendo:

Procesamiento de datos: NumPy es ampliamente utilizado en el ámbito de la ciencia de datos y el análisis numérico para manipular y procesar grandes conjuntos de datos de manera eficiente.

Cálculo científico: Es una herramienta esencial en campos como la física, la ingeniería y las matemáticas aplicadas para realizar cálculos complejos y resolver ecuaciones diferenciales.

Machine Learning: Muchas bibliotecas de machine learning en Python, como TensorFlow y Scikit-learn, se basan en NumPy para la manipulación de datos y cálculos numéricos.

3. Ejemplo de como trabajar con Arrays

Ejemplo de cómo trabajar con arrays en NumPy:

```
import numpy as np

# Crear un array de una dimensión
arr1d = np.array([1, 2, 3, 4, 5])

# Crear un array de dos dimensiones
arr2d = np.array([[1, 2, 3], [4, 5, 6], [7, 8, 9]])

# Acceder a elementos del array
print(arr1d[0]) # Imprimir el primer elemento del array de una dimensión
print(arr2d[1, 2]) # Imprimir el elemento en la fila 1 y columna 2 del array de dos dimensiones
```

4. Ejemplo de como hacer operaciones con matrices

```
import numpy as np

# Crear dos matrices
A = np.array([[1, 2], [3, 4]])
B = np.array([[5, 6], [7, 8]])

# Suma de matrices
C = A + B

# Producto de matrices
D = np.dot(A, B)

print("Suma de matrices:")
print(C)
print("Producto de matrices:")
print(D)
```

5. ¿Qué es Matplotlib?

Matplotlib, por otro lado, es una biblioteca de visualización de datos en 2D que produce gráficos de calidad de publicación en una variedad de formatos y entornos interactivos en Python.

6. 2 ejemplos del uso de Matplotlib

```
import matplotlib.pyplot as plt

x = np.linspace(0, 10, 100) # Crear un array de 100 puntos entre 0 y 10
y = np.sin(x) # Calcular el seno de cada punto

plt.plot(x, y) # Graficar x versus y
plt.xlabel('x') # Etiqueta del eje x
plt.ylabel('sin(x)') # Etiqueta del eje y
plt.title('Gráfico de Seno') # Título del gráfico
plt.grid(True) # Mostrar cuadrícula
plt.show() # Mostrar el gráfico
```

