Python libraries

PYTHON PROJECT

Fathima r

DG Vaishnav College

NumPy

One of the libraries in python

NumPy introduction:

NumPy is a Python library that is widely used in scientific computing and data analysis. It is short for "Numerical Python" and provides support for large, multi-dimensional arrays and matrices, along with a range of mathematical functions to operate on these arrays. NumPy is designed to provide efficient and fast computation on arrays and matrices. It provides a vectorized operation, which allows for operations to be applied to entire arrays, rather than individual elements.

NumPy is often considered as a free and open-source alternative to MATLAB, as it provides many similar functionalities for scientific computing and numerical analysis.

NumPy is used in:

Creating arrays like 1-dimensional, 2-dimensional, n-dimensional arrays

One-dimensional Arrays:

- One dimensional arrays are used to store homogeneous data
- One-dimensional arrays can be used to store and analyze data over time.

- Example: We can maintain the sales of product for a period of time and we can analyze the maximum sales, minimum sales etc.
- Another example is that we can use in machine learning models like image classification models we can first load images by cv2 library and convert into grayscale to make it two dimensional and then further by using flatten() method we can convert it and store in one dimensional array. And then it can be used for image classification

numpy1a.py

```
import numpy as np
a=np.array([1,2,3]) #creating array
print(a)
print(type(a))
print(a[0])#index can be used in one dimensional array
print(a[1:2]) #slicing can be also used in numpy one d array
b=np.array([4,5,6])
print(b)
c=np.concatenate([a,b]) #concatenation of arrays also can be done (means two arrays can be concatenated to single array)
print(c)
# ------------------------------------------------------------------------------------------------------------------------------------------------------
d,e=np.split(c, indices_or_sections=[2])
#in the above line split function i have used ,I want to split into two different arrays
#indices or sections i set is as 2 means the split occurs in 2nd index
#split can be used in vector representation using Bow for finding the unique words
print(d,e)
# ------------------------------------------------------------------------------------------------------------------------------------------------------
#for finding the unique words set function can be used
set1=np.array(["ok","fathi","iniya","first"])
set2=np.array(["fathi","good"])
set4=np.concatenate([set1,set2])
set3=set(set4)
print(set3)
# ------------------------------------------------------------------------------------------------------------------------------------------------------
a1 = np.array(['hello', 'world'])
b2= np.array(['world', 'tour'] )
c2= np.char.add(a1, b2)
print(c2) #['helloworld' 'worldtour'] this is the output
# ------------------------------------------------------------------------------------------------------------------------------------------------------
#some of the operations in 1-d array
print(a+b) #addition of arrays
print(a-b) #subraction of arrays
print(a*b) #multiplication of arrays
#dot product in array it is used with weight vector in sentiment analysis
e=np.dot(a,b)
print(e)
#zeros function it is used set values as zero,it can be used as default value for creating array before filling it with data
```

```
#zeros function it is used set values as zero,it can be used as default value for creating array before filling it with data
f=np.zeros(5,dtype=int) #datatype can also be set
print(f)
g=np.random.randint(20, 30, 10) #this line specifies that it randomly prints 10 values between 20 and 30
print(g)
#arange is used in print the range of values and we can declare when to start,stop and step is also allowed
h=np.arange(1,6,2)
print(h)
#square root
i=np.sqrt(a)
print(i)
#Square
k=np.square(a)
print(k)
#maximum
m=np.maximum(a,b)
print(m) #What happens here is that there are two arrays it compares each element of array with each other and prints the maximum among that t
#max prints the maximum value in an array
o=np.max(a)
print(o)
# ------------------------------------------------------------------------------------------------------------------------------------------
#are the values changeable in array?
a[0]=3
print(a)   #yes it can be modified
# ------------------------------------------------------------------------------------------------------------------------------------------
#can we delete values in an array
e2=np.array([78,9,0,7])
print("before delete:{}".format(e2))
f2=np.delete(e2,3)
print("after delete:{}".format(f2))
#del keyword can also be used but it is not advisable
# ------------------------------------------------------------------------------------------------------------------------------------------
# ------------------------------------------------------------------------------------------------------------------------------------------
```

Output:

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
[-3 -3 -3]
[ 4 10 18]
32
[0 0 0 0 0]
[29 25 21 21 20 28 23 23 21 22]
[1 3 5]
[1.         1.41421356 1.73205081]
[1 4 9]
[4 5 6]
3
[3 2 3]
before delete:[78  9  0  7]
after delete:[78  9  0]
PS C:\Users\mjavi\Desktop\fathi\python project>
```

Two Dimensional Arrays:

- Two-dimensional arrays are commonly used to represent data that has two-dimensional structure, such as images, tables, and matrices.
- For example image can be represented as a two-dimensional array of pixel values, where each element in the array represents the color of a pixel at a particular location in the image.
- Two-dimensional arrays can also be used to represent relationships between data points
  EXAMPLE:

We can stores the students name and students scores in 2-d array by using index we can access each student name and their score