

CS 5710 –Machine Learning Programming Assignment-3

GITHUB LINK: <https://github.com/vijender6/vijender>

RECORDING VIDEO LINK:

https://drive.google.com/file/d/1tbkWAF3TCJC7tKidJeF6a5Zh9GFFAcVA/view?usp=drive_link

**NAME: VIJENDER REDDY KOOTURU
700765220**

- a) Using NumPy create random vector of size 15 having only Integers in the range 1-20.
1. Reshape the array to 3 by 5
 2. Print array shape.
 3. Replace the max in each row by 0

Create a 2-dimensional array of size 4 x 3 (composed of 4-byte integer elements), also print the shape, type and data type of the array.

```
+ Code + Text Cannot save changes
import numpy as np

#Create random vector of size 15 with integers in the range 1-20
vector = np.random.randint(1, 21, size=15)

#Reshape the vector to 3x5
reshaped_array = vector.reshape(3, 5)

#Print array shape
array_shape = reshaped_array.shape

#Replace the max in each row by 0
for row in reshaped_array:
    row[np.argmax(row)] = 0

#Create a two-dimensional array of size 4x3 (composed of 4-byte integer elements)
array_4x3 = np.zeros((4, 3), dtype=np.int32)

#Print the shape, type, and data type of the 4x3 array
array_4x3_shape = array_4x3.shape
array_4x3_type = type(array_4x3)
array_4x3_dtype = array_4x3.dtype

print("Reshaped array with max in each row replaced by 0:")
print(reshaped_array)
print("4x3 array:")
print(array_4x3)
print("Shape of 4x3 array:", array_4x3_shape)
print("Type of 4x3 array:", array_4x3_type)
print("Data type of 4x3 array:", array_4x3_dtype)
```

```
+ Code + Text Cannot save changes
Reshaped array with max in each row replaced by 0:
[[ 5  2  3  8  0]
 [ 0 11  2  9  4]
 [ 7  0 14  4  6]]
4x3 array:
[[0 0 0]
 [0 0 0]
 [0 0 0]
 [0 0 0]]
Shape of 4x3 array: (4, 3)
Type of 4x3 array: <class 'numpy.ndarray'>
Data type of 4x3 array: int32
```

b. Write a program to compute the eigenvalues and right eigenvectors of a given square array given below:

```
+ Code + Text Cannot save changes Reconnect Gemini
```

```
import numpy as np

#Given square array
array = np.array([[3, -2],
                  [1, 0]])

#Computing eigenvalues and eigenvectors
eigenvalues, eigenvectors = np.linalg.eig(array)

print("Eigenvalues:")
print(eigenvalues)

print("Eigenvectors:")
print(eigenvectors)
```

```
Eigenvalues:
[2. 1.]
Eigenvectors:
[[0.89442719 0.70710678]
 [0.4472136  0.70710678]]
```

- c. Compute the sum of the diagonal element of a given array.

```
Welcome To Colab Share Gemini
```

```
File Edit View Insert Runtime Tools Help Cannot save changes
```

```
+ Code + Text Copy to Drive
```

```
#Import numpy library
import numpy as np

#Define the given array (matrix)
array = np.array([[0, 1, 2],
                  [3, 4, 5]])

#Extract the diagonal elements using numpy's diag function
diagonal_elements = np.diag(array)

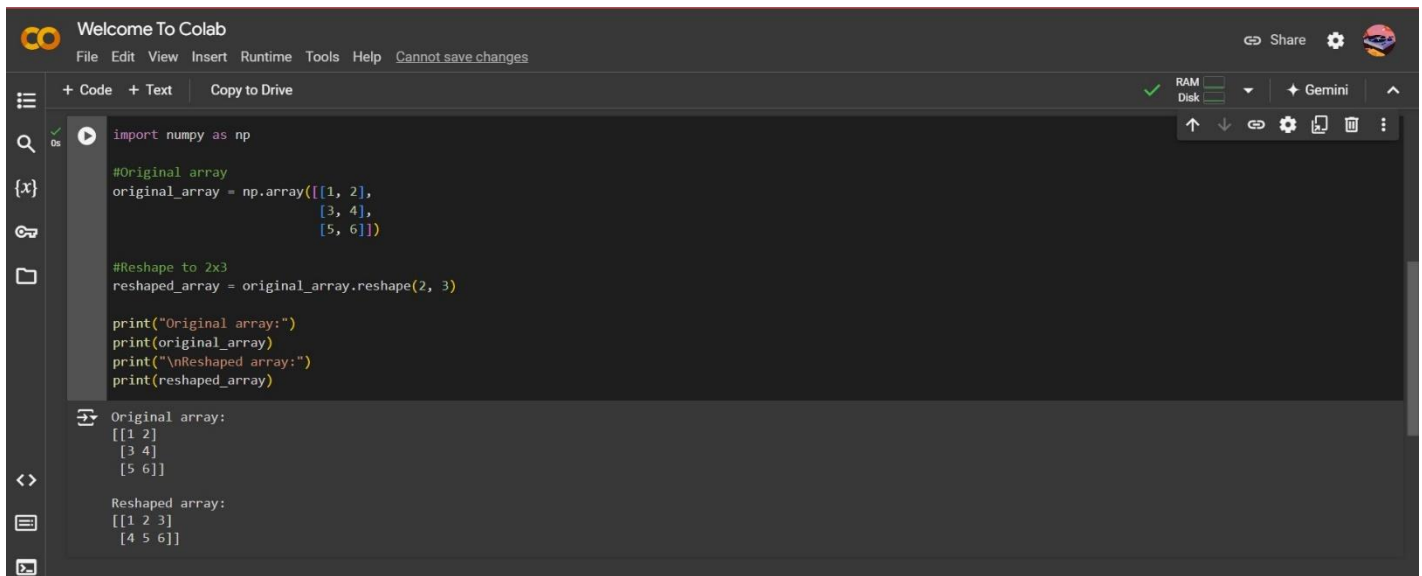
#Compute the sum of the diagonal elements
sum_of_diagonal_elements = np.sum(diagonal_elements)

#Print the sum of the diagonal elements
print("Sum of diagonal elements:")
print(sum_of_diagonal_elements)
```

```
Sum of diagonal elements:
4
```

d. Write a NumPy program to create a new shape to an array without changing its data.

Reshape 3x2. Reshape 2x3:



The screenshot shows a Google Colab notebook interface. The top bar includes the Colab logo, 'Welcome To Colab', and a menu with 'File', 'Edit', 'View', 'Insert', 'Runtime', 'Tools', 'Help', and a status message 'Cannot save changes'. Below the menu is a toolbar with '+ Code', '+ Text', and 'Copy to Drive'. On the right side of the toolbar, there are icons for RAM and Disk usage, a 'Share' button, a 'Gemini' button, and a user profile icon. The main code area contains the following Python code:

```
import numpy as np

#Original array
original_array = np.array([[1, 2],
                           [3, 4],
                           [5, 6]])

#Reshape to 2x3
reshaped_array = original_array.reshape(2, 3)

print("Original array:")
print(original_array)
print("\nReshaped array:")
print(reshaped_array)
```

The output of the code is displayed in a separate cell below the code cell:

```
Original array:
[[1 2]
 [3 4]
 [5 6]]

Reshaped array:
[[1 2 3]
 [4 5 6]]
```