

Machine learning Programing assignment -3

Harshitha Boyapati

700761387

Github Link: <https://github.com/Harshitha-Boyapati/ML-Assignment-3>

```
+ Code + Text

# 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

import numpy as np
r = np.random.randint(1,20,size= 15)
print("Random values are")
print(r)

# 1. Reshape the array to 3 by 5
s=r.reshape(3,5)
print("Array after reshape")
print(s)

# 2. Print array shape.
print("Array :",s)
print ("The shape of array is:",s.shape)

# 3. Replace the max in each row by 0
New_Array = np.where(s == [ [i] for i in np.amax(s, axis = 1) ], 0, s)
print(New_Array)

Random values are
[ 5  1 14  5 12 19  9  4 11  1  9 11  8 19 17]
Array after reshape
[[ 5  1 14  5 12]
 [19  9  4 11  1]
 [ 9 11  8 19 17]]
Array : [[ 5  1 14  5 12]
 [19  9  4 11  1]
 [ 9 11  8 19 17]]
The shape of array is: (3, 5)
[[ 5  1  0  5 12]
 [ 0  9  4 11  1]
 [ 9 11  8  0 17]]
```

✓
0s

```
# 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.  
import numpy as np  
  
arr = np.array([[3,3,3], [3,5,9], [9,4,7], [1,4,3]], dtype=np.int32)  
print("The shape of an array is ", arr.shape)  
print("The type of an array is ", type(arr))  
print("The data type of an array is ", arr.dtype)
```

☞ The shape of an array is (4, 3)
The type of an array is <class 'numpy.ndarray'>
The data type of an array is int32

✓
0s

```
#1(b) Write a program to compute the eigenvalues and right eigenvectors of a given square array given below:  
#[[ 3 -2]  
# [ 1 0]]  
import numpy as np  
# defining the square array  
arr = np.array([[3, -2], [1, 0]])  
  
# compute the eigenvalues and right eigenvectors  
eigenvalues, eigenvectors = np.linalg.eig(arr)  
  
# print the eigenvalues and right eigenvectors  
print("Eigenvalues for a given array:", eigenvalues)  
print("Right eigenvectors for a given array:\n", eigenvectors)
```

☞ Eigenvalues for a given array: [2. 1.]
Right eigenvectors for a given array:
[[0.89442719 0.70710678]
 [0.4472136 0.70710678]]

✓
0s

```
#1(c) Compute the sum of the diagonal element of a given array.  
#[[0 1 2]  
# [3 4 5]]  
  
import numpy as np  
  
arr = np.array([[0, 1, 2], [3, 4, 5]])  
sum = np.trace(arr)  
print("Sum of diagonal elements are+:", sum)
```

☞ Sum of diagonal elements are+: 4

✓
0s

▶ #1(d)Write a NumPy program to create a new shape to an array without changing its data.

```
#Reshape 3x2:
#[[1 2]
#[3 4]
#[5 6]]
#Reshape 2x3:
#[[1 2 3]
#[4 5 6]]
import numpy as np

# define the original array
arr = np.array([[1, 2], [3, 4], [5, 6]])

# reshape to 3x2
arr_3x2 = arr.reshape(3, 2)

# reshape to 2x3
arr_2x3 = arr.reshape(2, 3)

print("Original array:\n", arr_3x2)
print("Modified array:\n", arr_2x3)
```

📄 Original array:

```
[[1 2]
 [3 4]
 [5 6]]
```

Modified array:

```
[[1 2 3]
 [4 5 6]]
```