MACHINE LEARNING CS5710

CRN 30460

PROGRAMMING ASSIGNMENT 1

GITHUB LINK - https://github.com/ravitejachakilam7/ML_5170_summer2023_Assignment1.git

a) Numpy: 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.

```
# question 1.a

randomVector = np.random.randint(1, 21, size=15)

# 1.a.1 Reshape the array to 3 by 5
reshapeArray = randomVector.reshape(3, 5)

#1.a.2 Print Array shape
print("Array Shape: ", reshapeArray.shape)

print(reshapeArray)

#1.a.3 Replace the max in each row by 0
reshapeArray[np.arange(reshapeArray.shape[0]), np.argmax(reshapeArray, axis=1)] = 0

print("replaced Max with 0 in each row\n",reshapeArray)

#Create a 2-dimensional array of size 4 x 3 (composed of 4-byte integer elements)
array = np.empty((4, 3), dtype=np.int32)
# Print array shape
print("Array shape:", array.shape)

# Print array type
print("Array type:", type(array))

# Print array data type
print("Array data type:", array.dtype)
```

b) Write a program to compute the eigenvalues and right eigenvectors of a given square array given below: [[3 -2] [1 0]]

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

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

eigenvalues, rightEigenvectors = np.linalg.eig(array)
```

```
# Print the eigenvalues
print("Eigenvalues:")
for eigenvalue in eigenvalues:
    print(eigenvalue)

# Print the right eigenvectors
print("\nRight Eigenvectors:")
for i in range(len(rightEigenvectors)):
    print("Eigenvector", i+1, ":", rightEigenvectors[:, i])
```

c) Compute the sum of the diagonal element of a given array. [[0 1 2] [3 4 5]]

```
#1.c. Compute the sum of the diagonal element of a given array

array = np.array([[0, 1, 2], [3, 4, 5]])
diagonalSum = np.trace(array)
print("Given Array\n", array)
print("Sum of the diagonal elements:", diagonalSum)
```

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]]

```
# 1.d. d. Write a NumPy program to create a new shape to an array without changing its data.
array = np.array([[1, 2], [3, 4], [5, 6]])

# Reshape the array to 3x2
reshapedArray1 = array.reshape(3, 2)

# Reshape the array to 2x3
reshapedArray2 = array.reshape(2, 3)

# Print the original array
print("Original array:\n")
print(array)

# Print the reshaped arrays
print("\nReshaped array 1 (3x2):\n")
print(reshapedArray1)

print("\nReshaped array 2 (2x3):")
print(reshapedArray2)
```

2. Write a Python programming to create a below chart of the popularity of programming Languages. 2. Sample data: Programming languages: Java, Python, PHP, JavaScript, C#, C++ Popularity: 22.2, 17.6, 8.8, 8, 7.7, 6.7

```
# Problem solution 2.1, 2.2

import matplotlib.pyplot as plt

#input data
```

```
programmingLanguages = ['Java', 'Python', 'PHP', 'JavaScript', 'C#', 'C++']
popularity = [22.2, 17.6, 8.8, 8, 7.7, 6.7]

# Creating a pie chart
plt.pie(popularity, labels=programmingLanguages, autopct='%1.1f%%')
plt.title('Popularity of the Programming Languages given below')

# Displaying the chart
plt.show()
```