Neural Network & Deep Learning

(ICP Assignment # 3)

CS 5720

Name: Satya Ishyanth Kadali

Id: 700735513

CRN: 23259

Video link:

https://drive.google.com/file/d/1qKBNQG0H_k2Rg--wAl0qojbOJGhoBHbc/view?usp=sharing

Github link:

https://github.com/Ishyanth/Deep-Learning-Assignments

Question 1:

```
1  # ICP Assignment-3
2  CS 5720 Neural Network & Deep Learning
3  700735513, Satya Ishyanth Kadali
```

In [ ]: ▶ 1 Question 1

In [2]: ▶
```
1  #creating a Employee class
2  class Employee:
3      total_salary = 0
4      emp_count = 0
5
6      def __init__(self, name, family, salary, department):
7          self.name = name
8          self.family = family
9          self.salary = salary
10         self.department = department
11         # increment total salary and employee count
12         Employee.total_salary += salary
13         Employee.emp_count += 1
14
15     # class method to calculate average salary
16     @classmethod
17     def average_salary(cls):
18         return cls.total_salary / cls.emp_count
19
20  #Creating a Fulltime Employee class with Inheritence
21  class FulltimeEmployee(Employee):
22      pass
23
24  # creating instances of the classes
25  emp1 = FulltimeEmployee("Ishyanth", "Kadali", 90000, "Research")
26  emp2 = Employee("Abhi", "Tammana", 40000, "Customer Service")
27  emp3 = Employee("Jayakar", "Dadala", 80000, "Accounting")
28
29  print("Average Salary = ", Employee.average_salary())
30  print("Number of Employees = ", Employee.emp_count)
31
```

```
Average Salary =  70000.0
Number of Employees =  3
```

1. Creating an **Employee** class with instance variables **total_salary** and **emp_count** that keeps track of the total salary and number of employees, respectively.

2. The **__init__** method initializes an instance of the class with the given parameters and increments the **total_salary** and **emp_count** class variables.

3. The **average_salary** is a class method, which calculates the average salary of all the employees using the class variables **total_salary** and **emp_count**.

4. The **FulltimeEmployee** class is defined as a subclass of the **Employee** class, which inherits all the properties and methods of the parent class. The instances of the **FulltimeEmployee** class are created and the average salary and number of employees are displayed.

Question 2:

```
1  Question 2
```

```
In [6]:    1  import numpy as np
           2
           3  # Creating a random vector of size 20 with only float values in the range 1-20
           4  r_vector = np.random.uniform(1, 20, 20)
           5
           6  # Reshape the array to 4x5
           7  new_array = r_vector.reshape(4, 5)
           8
           9  # Replace the max in each row with 0 (axis=1)
          10  np.place(new_array, new_array == np.max(new_array, axis=1, keepdims=True), 0)
          11
          12  print(new_array)
```

```
[[ 0.          7.22182483 12.85463727 11.54550355  3.78727794]
 [ 0.         10.75728244 10.07139048  7.32120781 12.60012389]
 [ 5.68009521  3.55195525  3.99239713  0.          4.95695426]
 [14.13367363  0.          9.95311926 13.16701456 16.06566808]]
```

1. The code imports the numpy library and uses the **random.uniform** function to create a vector of 20 random float values between 1 and 20.

2. The vector is then reshaped into a 4x5 array.

3. Using the **numpy place function**, the maximum value in each row of the array is replaced with 0 and output array is printed.