Machine Learning (Assignment)

Jagadeeswar Chimata
700731649

Question 1:

In [17]:

```
# 1) Sort the list and find the min and max age
# here we declare the function to check for maximum number
def max(x):
  max = x[0]
  for i in x:
    if i > max:
      max = i
  return max

# here we declare the function to check for minimum number
def min(x):
  min = x[0]
  for j in x:
    if j < min:
      min = j
  return min

#ages array
ages = [19, 22, 19, 24, 20, 25, 26, 24, 25, 24]
ages.sort()

#final output of max and min age
print("sorted array:",ages)
print("Maximum age:", max(ages))
print("Minimum age:", min(ages))S

#Add the min age and the max age again to the list
ages.append(min(ages))
ages.append(max(ages))
ages.sort()
print(ages)

#Find the median age (one middle item or two middle items divided by two)
print("sorted array",ages)
mid = len(ages) // 2
res = (ages[mid] + ages[~mid]) / 2
```

```
# Printing result
print("Median of the ages list is : " + str(res))


#Find the average age (sum of all items divided by their number)

def Average(ages):
    return sum(ages) / len(ages)

# Driver Code
average = Average(ages)
print("ages array list : ",ages)

#average age
print("Average age =", round(average, 2))

#Find the range of the ages (max minus min)
ages = [19, 22, 19, 24, 20, 25, 26, 24, 25, 24]
temp = sorted(ages)

# sorted list

min = temp[0]
max = temp[-1]

# index -1 will give the last element

print ("min", min, ages.index(min))
print ("max", max, ages.index(max))
print ("Range : ", max - min)
```

```
sorted array: [19, 19, 20, 22, 24, 24, 24, 25, 25, 26]
Maximum age: 26
Minimum age: 19
[19, 19, 19, 20, 22, 24, 24, 24, 25, 25, 26, 26]
sorted array [19, 19, 19, 20, 22, 24, 24, 24, 25, 25, 26, 26]
Median of the ages list is : 24.0
ages array list :  [19, 19, 19, 20, 22, 24, 24, 24, 25, 25, 26, 26]
Average age = 22.75
min 19 0
max 26 6
Range :  7
```

•we declare the function to check for maximum number

•we declare the function to check for minimum number

• List items are indexed, and we can access them by referring to the index number and negative indexing means start from the end.

• To add an item to the end of the list, use the append () method

• Sort () method that will sort the list alphanumerically, ascending, by default

Question 2:

In [2]:
```
#Create an empty dictionary called dog
dog = {}

#Add name, color, breed, legs, age to the dog dictionary
dog = {'name','color','breed','legs','age'}

#Create a student dictionary and add first_name, last_name, gender, age, marital status,
#skills, country, city and address as keys for the dictionary
student = {'first_name','last_name','gender','age','marital status','skills','country','city','address'}

#Get the length of the student dictionary
student = {'first_name','last_name','gender','age','marital status','skills','country','city','address'}
print(len(student))

#Get the value of skills and check the data type, it should be a list
student = {'first_name ': 'jagadeeswar','last_name': 'chimata','gender': 'male','age': '16','marital status': 'single','skills':
print(student.get('first_name'))
print(student.get('country'))
print(student.get('skills'))
print(student.get('city'))

#Modify the skills values by adding one or two skills
student = {'first_name ': 'jagadeeswar','last_name': 'chimata','gender': 'male','age': '16','marital status': 'single','skills':
student['students'] = 'studying'
student['skills'].append('java script')
print(student)

#Get the dictionary keys as a list
student = {'first_name ': 'jagadeeswar','last_name': 'chimata','gender': 'male','age': '16','marital status': 'single','skills':
keys = student.keys()
print(keys)

#Get the dictionary values as a list
student = {'first_name ': 'jagadeeswar','last_name': 'chimata','gender': 'male','age': '16','marital status': 'single','skills':
values = student.values()
print(values)
```

```
9
None
united states
['mean stack development', 'java', 'machine learning']
kansas
{'first_name ': 'jagadeeswar', 'last_name': 'chimata', 'gender': 'male', 'age': '16', 'marital status': 'single', 'skills': ['m
ean stack development', 'java', 'machine learning', 'java script'], 'country': 'united states', 'city': 'kansas', 'address': 'o
verland park', 'students': 'studying'}
dict_keys(['first_name ', 'last_name', 'gender', 'age', 'marital status', 'skills', 'country', 'city', 'address'])
dict_values(['jagadeeswar', 'chimata', 'male', '16', 'single', ['mean stack development', 'java', 'machine learning'], 'united
states', 'kansas', 'overland park'])
```

•The value of skills and check the data type, it should be a list with few sets for example

• Dictionaries are written with curly brackets, and have keys and values

• Dictionaries are used to store data values in key: value pairs .A dictionary is changeable and do not allow duplicates

• len () is used to get length of the dictionary

Question 3:

In [5]:
```
#Create a tuple containing names of your sisters and your brothers (imaginary siblings are fine)
cousins1 = ('rahul', 'varun', 'joseph')
cousins2 = ('priyanka', 'satwika', 'anusha')

#Join brothers and sisters tuples and assign it to siblings
cousins1 = ('rahul', 'varun', 'joseph')
cousins2 = ('priyanka', 'satwika', 'anusha')
siblings = cousins1 + cousins2
print(siblings)

#How many siblings do you have?
len(siblings)

#Modify the siblings tuple and add the name of your father and mother and assign it to family_members
siblings = ("papireddy","swarna")+siblings
family_members = siblings
print("family members:",family_members)
```

```
('rahul', 'varun', 'joseph', 'priyanka', 'satwika', 'anusha')
family members: ('papireddy', 'swarna', 'rahul', 'varun', 'joseph', 'priyanka', 'satwika', 'anusha')
```

• A tuple is a collection which is ordered and unchangeable.

• Tuples are written with round brackets

• To join two or more tuples you can use the + operator

Question 4:

In [7]:
```
#given data set
it_companies = ["Facebook", "Google"," Microsoft", "Apple", "IBM", "Oracle", "Amazon"]
A = {19, 22, 24, 20, 25, 26}
B = {19, 22, 20, 25, 26, 24, 28, 27}
age = [22, 19, 24, 25, 26, 24, 25, 24]

#Find the length of the set it_companies
print(len(it_companies))

#Add 'Twitter' to it_companies
it_companies = ["Facebook", "Google"," Microsoft", "Apple", "IBM", "Oracle", "Amazon"]
it_companies.append('twitter')
print(it_companies)

#Insert multiple IT companies at once to the set it_companies
it_companies.extend(["cognizent","paytm"])
print(it_companies)

#Remove one of the companies from the set it_companies
it_companies.remove('paytm')
print(it_companies)

#What is the difference between remove and discard
print("remove is a method where it shows an error when value does not exist but discard method does not show an error.")

#Join A and B
a_union_b=A.union(B)
print(a_union_b)

#Find A intersection B
a_intersection_b=A.intersection(B)
print(a_intersection_b)
```

```
#Is A subset of B ?
a_subset_b=A.issubset(B)
print(a_subset_b)

#Are A and B disjoint sets

a_disjoint_b = A.isdisjoint(B)
print(a_disjoint_b)


#What is the symmetric difference between A and B
a_symmetric_difference_b = A.symmetric_difference(B)
print(a_symmetric_difference_b)

#Delete the sets completely
A.clear()
B.clear()
print (A)
print (B)

#Convert the ages to a set and compare the length of the list and the set.
age_set=set(age)
print(age_set)
print ("difference between length of age",len(age)-len(age_set))
```

```
7
['Facebook', 'Google', ' Microsoft', 'Apple', 'IBM', 'Oracle', 'Amazon', 'twitter']
['Facebook', 'Google', ' Microsoft', 'Apple', 'IBM', 'Oracle', 'Amazon', 'twitter', 'cognizent', 'paytm']
['Facebook', 'Google', ' Microsoft', 'Apple', 'IBM', 'Oracle', 'Amazon', 'twitter', 'cognizent']
remove is a method where it shows an error when value does not exist but discard method does not show an error.
{19, 20, 22, 24, 25, 26, 27, 28}
{19, 20, 22, 24, 25, 26}
True
False
{27, 28}
set()
set()
{19, 22, 24, 25, 26}
difference between length of age 3
```

• Sets are written with curly brackets, set is a collection which is unordered, unchangeable, and unindexed

• To add one item to a set use the add () method and To add items from another set into the current set, use the update () method

• To remove an item in a set, use the remove () or the discard() method. remove method raises an error when value doesn't exist whereas discard method doesn't raise an error.

• Union() return a set that contains all items from both sets, duplicates are excluded, intersection() returns a set that contains the items that exist in both set A, and set B, is subset() return True if all items in set x are present in set y, is disjoint() return True if no items in set x is present in set
symmetric_difference() return a set that contains all items from both sets, except items that are present in both sets

•del keyword will delete the set completely.

Question 5:

```
In [8]: #Calculate the area of a circle and assign the value to a variable name of _area_of_circle_
        PI = 3.14
        radius = float(input('Enter the radius of a circle: '))

        circumference = 2 * PI * radius
        area = PI * radius * radius

        print(" Circumference Of a Circle = %.2f"%circumference)
        print(" Area Of a Circle = %.2f" %area)

        Enter the radius of a circle: 2
         Circumference Of a Circle = 12.56
         Area Of a Circle = 12.56
```

• input () allows user to pass the dynamic input

• ** is a power operator

• And by using the formula of area and circumference of the circle we can calculate those by using mathematical operators

Question 6:

```
In [9]: #How many unique words have been used in the sentence? Use the split methods and set to get the unique words.

        line = "I am a teacher and I love to inspire and teach people"
        unique_letter = set(line.split())
        print("length of unique words  ",len(unique_letter))

        length of unique words   10
```

• Split a string into a list where each word is a list item, user can specify the separator, default separator is any whitespace.

Question 7:

In [13]:
```
#Use a tab escape sequence to get the following lines.
print("name\tage\tcountry  \tcity\nJaggu\t16\tunited_states\tWarrensburg")
```

```
name    age     country         city
Jaggu   16      united_states   Warrensburg
```

- To insert characters that are illegal in a string, use an escape character.
- An escape character is a backslash \ followed by the character you want to insert.
- \t is used for a tab space, \n for new line

Question 8:

In [14]:
```
#Use the string formatting method to display the following:
#radius = 10
#area = 3.14 * radius ** 2
#"The area of a circle with radius 10 is 314 meters square."

radius = 10
area = 3.14 * radius ** 2
print("area of circle has radius %s and its square meters is %s " %(radius,area))
```

```
area of circle has radius 10 and its square meters is 314.0
```

- The % operator in python for strings is used for something called string substitution.

Question 9:

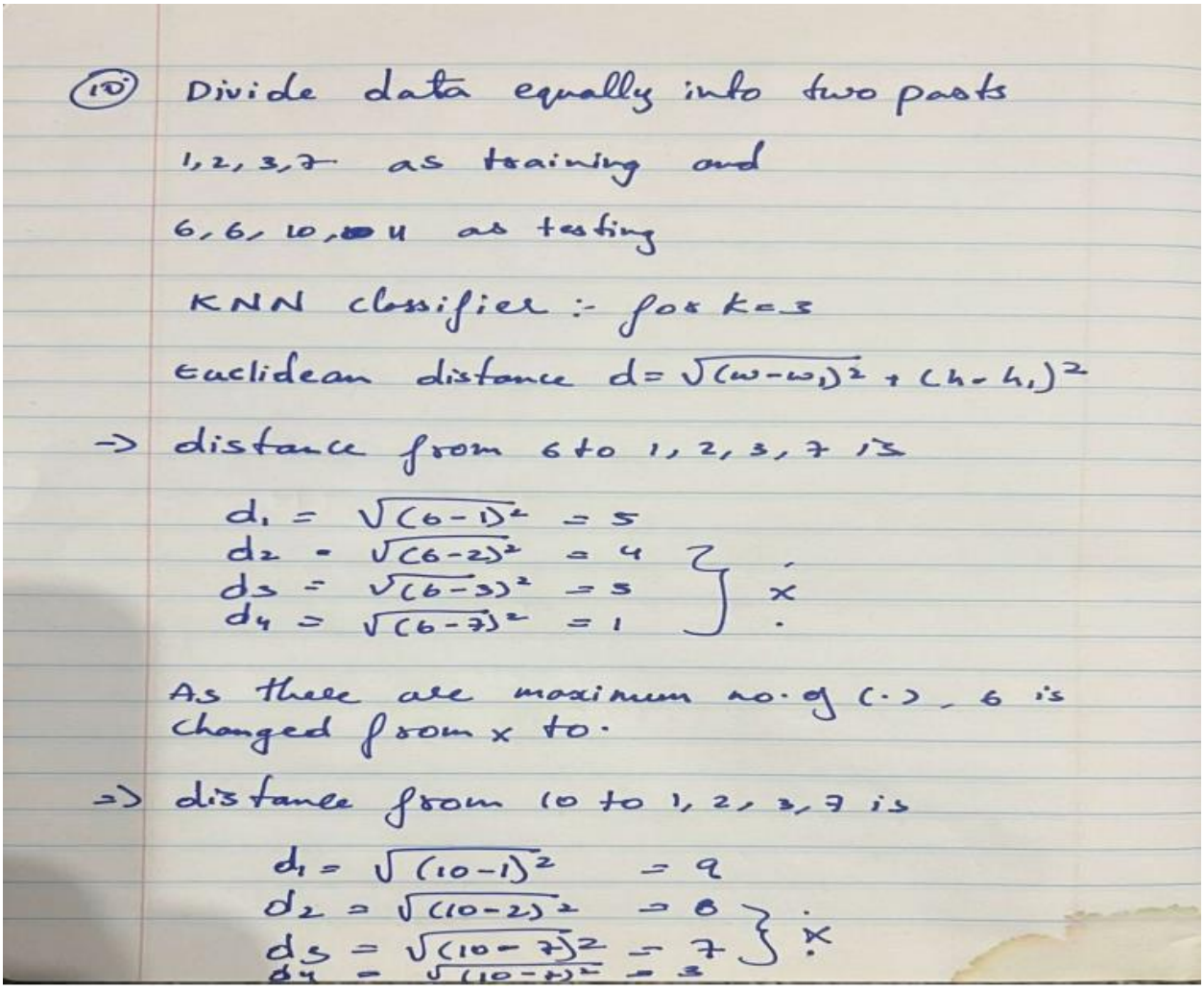
In [15]:
```
#Write a program, which reads weights (lbs.) of N students into a list and convert these weights to
#kilograms in a separate list using Loop. N: No of students (Read input from user)
students = int(input("enter no of students"))
#creating empty list
weight_lbs = []
weight_kgs = []
for stu in range(students):
    # get the input of data and adding up to the list
    weight_lbs.append(int(input("enter weight"+str(stu)+"")))
    #converting lbs to kgs
    weight_kgs.append(round(weight_lbs[stu]*0.45392,2))
    print("convertion of weight in lbs ",weight_lbs)
    print("cionvertion of weight in kgs ",weight_kgs)
```

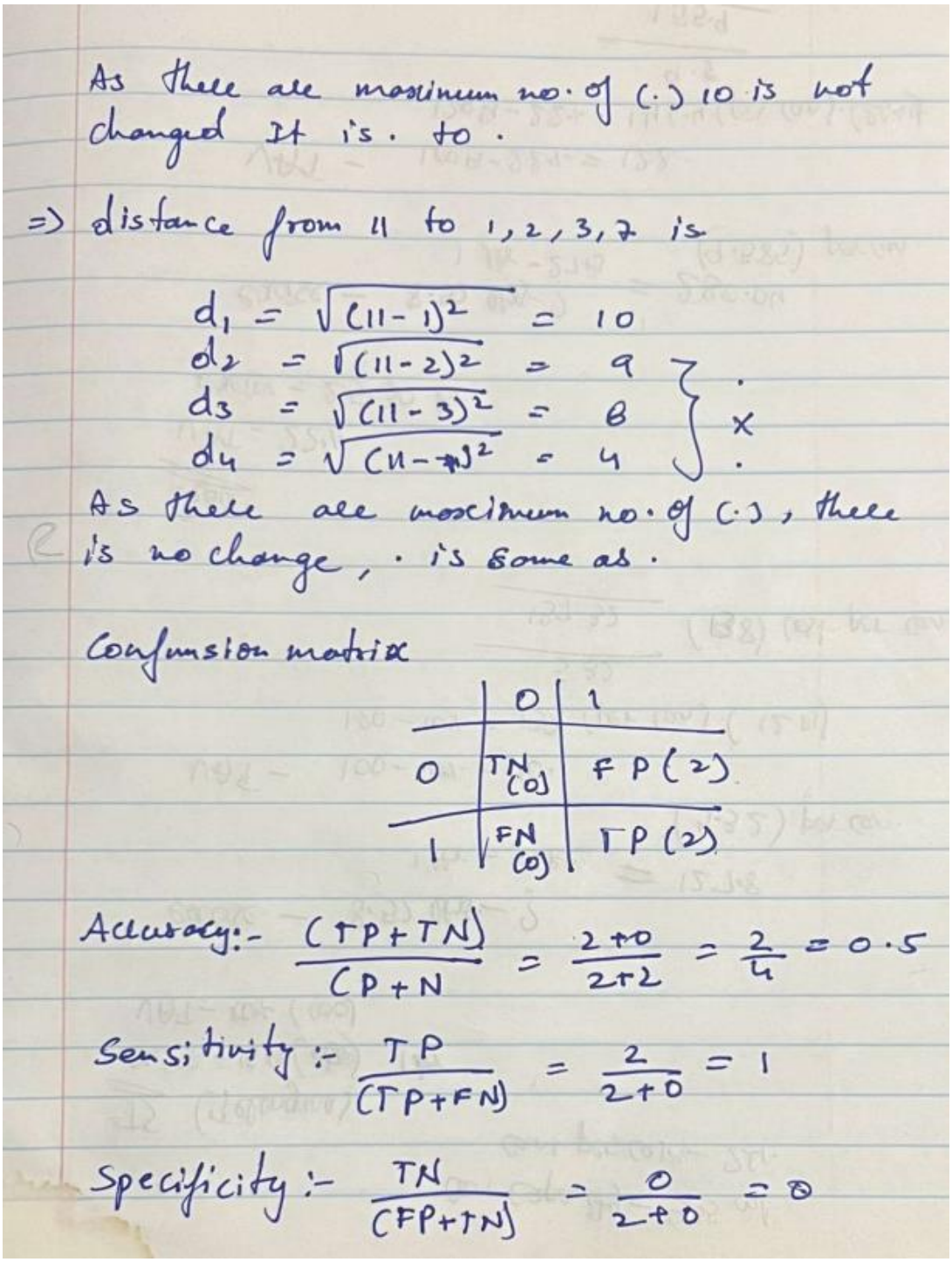
```
enter no of students6
enter weight020
convertion of weight in lbs  [20]
cionvertion of weight in kgs  [9.08]
enter weight1136
convertion of weight in lbs  [20, 136]
cionvertion of weight in kgs  [9.08, 61.73]
enter weight2369
convertion of weight in lbs  [20, 136, 369]
cionvertion of weight in kgs  [9.08, 61.73, 167.5]
enter weight33
convertion of weight in lbs  [20, 136, 369, 3]
cionvertion of weight in kgs  [9.08, 61.73, 167.5, 1.36]
enter weight422
convertion of weight in lbs  [20, 136, 369, 3, 22]
cionvertion of weight in kgs  [9.08, 61.73, 167.5, 1.36, 9.99]
enter weight566
convertion of weight in lbs  [20, 136, 369, 3, 22, 66]
cionvertion of weight in kgs  [9.08, 61.73, 167.5, 1.36, 9.99, 29.96]
```

• Using the input () function allows user input

• Using the for loop we will iterate the N no.of time and collect the students weights in lbs. and store it in the list after that performing the weight conversion calculations from lbs. to kgs the student weights in kgs is stored in another list.

Question 10:





Git repo link:

https://github.com/JagadeeswarChimata/ML-Assignment-1.git