

MACHINE LEARNING

ASSIGNMENT1

Name:Rupa mallempati

Github link: https://github.com/rupamallempati/ML_Assignment1.git

Video link: [https://drive.google.com/file/d/17kGWrVLI RcTys3iep1_o-j05VvCEPAk6/view?usp=share link](https://drive.google.com/file/d/17kGWrVLI RcTys3iep1_o-j05VvCEPAk6/view?usp=share_link)

Question 1

The following is a list of 10 students ages:

ages = [19, 22, 19, 24, 20, 25, 26, 24, 25, 24]

- Sort the list and find the min and max age
- Add the min age and the max age again to the list
- Find the median age (one middle item or two middle items divided by two)
- Find the average age (sum of all items divided by their number)
- Find the range of the ages (max minus min)

Source code:

```
import statistics
```

```
Ages = [19, 22, 19, 24, 20, 25, 26, 24, 25, 24]
```

```
#Sort the list and find the min and max age
```

```
Ages.sort()
```

```
print("After Sorting the list: ", Ages)
```

```
#Add the min age and the max age again to the list
```

```
Min = min(Ages)
```

```
Max = max(Ages)
print("Minimum value in list : " , Min ,"\n" "Maximum value in list : " , Max)
Ages.append(Min) #Appending
Ages.append(Max)
print("After appending the list:", Ages)
#Find the median age
Med = statistics.median(Ages)
print("Median of the list:", Med)
#Find the average age
Sum = sum(Ages);
Length = len(Ages);
Average = Sum/Length;
print("Average of the list : " , Average);
#Range
Range = Max - Min;
print("Range of the list :", Range);
```

OUTPUT:

```
After Sorting the list:  [19, 19, 20, 22, 24, 24, 24, 25, 25, 26]
Minimum value in list :  19
Maximum value in list :  26
After appending the list: [19, 19, 20, 22, 24, 24, 24, 25, 25, 26, 19, 26]
Median of the list: 24.0
Average of the list :  22.75
Range of the list : 7
```

```

In [1]: import statistics
Ages = [19, 22, 19, 24, 20, 25, 26, 24, 25, 24]

#Sort the list and find the min and max age
Ages.sort()
print("After Sorting the list: ", Ages)

#Add the min age and the max age again to the list
Min = min(Ages)
Max = max(Ages)
print("Minimum value in list : " , Min , "\n" "Maximum value in list : " , Max)
Ages.append(Min) #Appending
Ages.append(Max)
print("After appending the list:", Ages)

#Find the median age
Med = statistics.median(Ages)
print("Median of the list:", Med)

#Find the average age
Sum = sum(Ages);
Length = len(Ages);
Average = Sum/Length;
print("Average of the list : ", Average);

#Range
Range = Max - Min;
print("Range of the list :", Range);

After Sorting the list: [19, 19, 20, 22, 24, 24, 24, 25, 25, 26]
Minimum value in list : 19
Maximum value in list : 26
After appending the list: [19, 19, 20, 22, 24, 24, 24, 25, 25, 26, 19, 26]
Median of the list: 24.0
Average of the list : 22.75
Range of the list : 7

```

Question 2

- Create an empty dictionary called dog
- Add name, color, breed, legs, age to the dog dictionary
- Create a student dictionary and add first_name, last_name, gender, age, marital status, skills, country, city and address as keys for the dictionary
- Get the length of the student dictionary
- Get the value of skills and check the data type, it should be a list
- Modify the skills values by adding one or two skills
- Get the dictionary keys as a list
- Get the dictionary values as a list

Source code:

```
#Create an empty dictionary called dog
```

```
Dog = {}
```

```
#Add name, color, breed, legs, age to the dog dictionary
```

```
Dog.update({'Name' : 'Vikky','Color' : 'Black','Breed' : 'Shitzu','Legs' : '4','Age' : '4'})
```

```
#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' : 'Rupa',  
           'Last_name' : 'Mallempati',  
           'Gender' : 'Female',  
           'Age' : '23',  
           'Marital status' : 'Unmarried',  
           'Skills' : ["C","Java","Python","MySQL"],  
           'Country' : 'India',  
           'City' : 'Hyderabad', 'Address' : 'Khammam, 533105'}
```

```
#Get the length of the student dictionary
```

```
print("Length of the Student dictionary is :", len(Student))
```

```
#Get the value of skills and check the data type, it should be a list
```

```
print("Skills of the student are :", Student['Skills'])
```

```
print("Datatype of the skills is :", type(Student['Skills']))
```

```
#Modify the skills values by adding one or two skills
```

```
Student['Skills'].extend(["HTML", "Azure Sentinel"])
```

```
print("Modified skills in the list are :", Student['Skills'])
```

#Get the dictionary keys as a list

```
print("Keys in the student dictionary are :", list(Student.keys()))
```

#Get the dictionary values as a list

```
print("values in the student dictionary are :", list(Student.values()))
```

Output:

```
Length of the Student dictionary is : 9
Skills of the student are : ['C', 'Java', 'Python', 'MySQL']
Datatype of the skills is : <class 'list'>
Modified skills in the list are : ['C', 'Java', 'Python', 'MySQL', 'HTML',
'Azure Sentinel']
Keys in the student dictionary are : ['First_name', 'Last_name', 'Gender',
'Age', 'Marital status', 'Skills', 'Country', 'City', 'Address']
values in the student dictionary are : ['Rupa', 'Mallempati', 'Female', '2
3', 'Unmarried', ['C', 'Java', 'Python', 'MySQL', 'HTML', 'Azure Sentinel'
], 'India', 'Hyderabad', 'Khammam, 533105']
```

```
In [13]: #Create an empty dictionary called dog
Dog = {}
#Add name, color, breed, legs, age to the dog dictionary
Dog.update({'Name': 'Vikky', 'Color': 'Black', 'Breed': 'Shitzu', 'Legs': '4', 'Age': '4'})

#Create a student dictionary and add first_name, last_name, gender, age, marital status, skills, country, city and address as
Student = {'First_name': 'Rupa',
          'Last_name': 'Mallempati',
          'Gender': 'Female',
          'Age': '23',
          'Marital status': 'Unmarried',
          'Skills': ['C', 'Java', 'Python', 'MySQL'],
          'Country': 'India',
          'City': 'Hyderabad', 'Address': 'Khammam, 533105'}

#Get the Length of the student dictionary
print("Length of the Student dictionary is :", len(Student))

#Get the value of skills and check the data type, it should be a list
print("Skills of the student are :", Student['Skills'])
print("Datatype of the skills is :", type(Student['Skills']))

#Modify the skills values by adding one or two skills
Student['Skills'].extend(['HTML', 'Azure Sentinel'])
print("Modified skills in the list are :", Student['Skills'])

#Get the dictionary keys as a list
print("Keys in the student dictionary are :", list(Student.keys()))
#Get the dictionary values as a list
print("values in the student dictionary are :", list(Student.values()))

Length of the Student dictionary is : 9
Skills of the student are : ['C', 'Java', 'Python', 'MySQL']
Datatype of the skills is : <class 'list'>
Modified skills in the list are : ['C', 'Java', 'Python', 'MySQL', 'HTML', 'Azure Sentinel']
Keys in the student dictionary are : ['First_name', 'Last_name', 'Gender', 'Age', 'Marital status', 'Skills', 'Country', 'Ci
ty', 'Address']
values in the student dictionary are : ['Rupa', 'Mallempati', 'Female', '23', 'Unmarried', ['C', 'Java', 'Python', 'MySQL',
'HTML', 'Azure Sentinel'], 'India', 'Hyderabad', 'Khammam, 533105']
```

Question 3:

- Create a tuple containing names of your sisters and your brothers (imaginary siblings are fine)
- Join brothers and sisters tuples and assign it to siblings
- How many siblings do you have?
- Modify the siblings tuple and add the name of your father and mother and assign it to family_members

Source code:

```
#Create a tuple containing names of your sisters and your brothers (imaginary siblings are fine)
```

```
Sisters = ("Sindhu", "Chaithu", "Sravani", "Anusha", "Anvi")
```

```
Brothers = ("Rajesh", "Vishal", "Akash", "Sandeep")
```

```
print("Sister names: ", Sisters)
```

```
print("Brother names: ", Brothers)
```

```
#Join brothers and sisters tuples and assign it to siblings
```

```
Siblings = Sisters + Brothers
```

```
print("After join brothers and sisters in a tuple: ", Siblings)
```

```
#How many siblings do you have?
```

```
Count= len(Siblings)
```

```
print("Number of Siblings: ", Count)
```

```
#Modify the siblings tuple and add the name of your father and mother and assign it to family_members
```

```
Mother = "Triveni"
```

```
Father= "SrinivasaRao"
```

```
family_members = list(Siblings)
```

```
family_members.append(Father)
```

```
family_members.append(Mother)
```

```
family_members=tuple(family_members)
```

```
print("After appending father and mother name: ", family_members)
```

Output:

```
Sister names: ('Sindhu', 'Chaithu', 'Sravani', 'Anusha', 'Anvi')
Brother names: ('Rajesh', 'Vishal', 'Akash', 'Sandeep')
After join brothers and sisters in a tuple: ('Sindhu', 'Chaithu', 'Sravan
i', 'Anusha', 'Anvi', 'Rajesh', 'Vishal', 'Akash', 'Sandeep')
Number of Siblings: 9
After appending father and mother name: ('Sindhu', 'Chaithu', 'Sravani',
'Anusha', 'Anvi', 'Rajesh', 'Vishal', 'Akash', 'Sandeep', 'SrinivasaRao',
'Triveni')
```

```
In [14]: M #Create a tuple containing names of your sisters and your brothers (imaginary siblings are fine)
Sisters = ("Sindhu", "Chaithu", "Sravani", "Anusha", "Anvi")
Brothers = ("Rajesh", "Vishal", "Akash", "Sandeep")
print("Sister names: ", Sisters)
print("Brother names: ", Brothers)

#Join brothers and sisters tuples and assign it to siblings
Siblings = Sisters + Brothers
print("After join brothers and sisters in a tuple: ", Siblings)

#How many siblings do you have?
Count= len(Siblings)
print("Number of Siblings: ", Count)

#Modify the siblings tuple and add the name of your father and mother and assign it to family_members
Mother = "Triveni"
Father= "SrinivasaRao"
family_members = list(Siblings)
family_members.append(Father)
family_members.append(Mother)
family_members=tuple(family_members)
print("After appending father and mother name: ", family_members)

Sister names: ('Sindhu', 'Chaithu', 'Sravani', 'Anusha', 'Anvi')
Brother names: ('Rajesh', 'Vishal', 'Akash', 'Sandeep')
After join brothers and sisters in a tuple: ('Sindhu', 'Chaithu', 'Sravani', 'Anusha', 'Anvi', 'Rajesh', 'Vishal', 'Akash',
'Sandeep')
Number of Siblings: 9
After appending father and mother name: ('Sindhu', 'Chaithu', 'Sravani', 'Anusha', 'Anvi', 'Rajesh', 'Vishal', 'Akash', 'Sa
ndeep', 'SrinivasaRao', 'Triveni')
```

Question 4:

```
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
- Add 'Twitter' to it_companies
- Insert multiple IT companies at once to the set it_companies
- Remove one of the companies from the set it_companies
- What is the difference between remove and discard
- Join A and B
- Find A intersection B
- Is A subset of B
- Are A and B disjoint sets
- Join A with B and B with A
- What is the symmetric difference between A and B
- Delete the sets completely
- Convert the ages to a set and compare the length of the list and the set

Source code:

```
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("The length of set is:", len(IT_Companies))
```


#Add 'Twitter' to it_companies

```
IT_Companies.add("Twitter")
```

```
print(IT_Companies)
```

#Insert multiple IT companies at once to the set it_companies

```
Multiple_ITcompanies= ["Soceite","Samsung", "Deloitte", "Meta"]
```

```
IT_Companies.update(Multiple_ITcompanies)
```

```
print(IT_Companies)
```

#Remove one of the companies from the set it_companies

```
IT_Companies.remove("Samsung")
```

```
print(IT_Companies)
```

#What is the difference between remove and discard

#Remove: If the item to remove does not exist, remove() will raise an error

#Discard: If the item to remove does not exist, discard() will NOT raise an error

#Join A and B

```
C = A.union(B)
```

```
print("After joining A and B: ", C)
```

#Find A intersection B

```
D = A.intersection(B)
```

```
print("After intersecting A and B: ", D)
```

```
#Is A subset of B
```

```
E = A.issubset(B)
```

```
print("Is A subset of B?", E)
```

```
#Are A and B disjoint sets
```

```
F = A.isdisjoint(B)
```

```
print("Are A and B disjoint sets? ", F)
```

```
#Join A with B and B with A
```

```
G = B.union(A)
```

```
print("After joining A With B: ",C, "and joining B with A: ",G)
```

```
#What is the symmetric difference between A and B
```

```
H = A.symmetric_difference(B)
```

```
print("Symmetric difference between A and B:", H)
```

```
#Delete the sets completely
```

```
del A,B
```

```
#print(B)
```

```
#Convert the ages to a set and compare the length of the list and the set
```

```
I = set(Age)
```

```
print("Is length of list and set is same :", len(I)==len(Age))
```

Output:

```
The length of set is: 7
{'Oracle', 'Amazon', 'IBM', 'Google', 'Apple', 'Microsoft', 'Twitter', 'Facebook'}
{'Samsung', 'IBM', 'Twitter', 'Apple', 'Oracle', 'Microsoft', 'Deloitte', 'Facebook', 'Amazon', 'Tesla', 'Google', 'Meta'}
{'IBM', 'Twitter', 'Apple', 'Oracle', 'Microsoft', 'Deloitte', 'Facebook', 'Amazon', 'Tesla', 'Google', 'Meta'}
After joining A and B: {19, 20, 22, 24, 25, 26, 27, 28}
After intersecting A and B: {19, 20, 22, 24, 25, 26}
Is A subset of B? True
Are A and B disjoint sets? False
After joining A With B: {19, 20, 22, 24, 25, 26, 27, 28} and joining B with A: {19, 20, 22, 24, 25, 26, 27, 28}
Symmetric difference between A and B: {27, 28}
Is length of list and set is same : False
```

```

In [4]: 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("The length of set is:", len(IT_Companies))

#Add 'Twitter' to it_companies
IT_Companies.add("Twitter")
print(IT_Companies)

#Insert multiple IT companies at once to the set it_companies
Multiple_ITcompanies= ["Soceite", "Samsung", "Deloitte", "Meta"]
IT_Companies.update(Multiple_ITcompanies)
print(IT_Companies)

#Remove one of the companies from the set it_companies
IT_Companies.remove("Samsung")
print(IT_Companies)

#What is the difference between remove and discard
#Remove: If the item to remove does not exist, remove() will raise an error
#Discard: If the item to remove does not exist, discard() will NOT raise an error

#Join A and B
C = A.union(B)
print("After joining A and B: ", C)

#Find A intersection B
D = A.intersection(B)
print("After intersecting A and B: ", D)

#Is A subset of B
E = A.issubset(B)
print("Is A subset of B?", E)

#Are A and B disjoint sets
F = A.isdisjoint(B)
print("Are A and B disjoint sets? ", F)

#Join A with B and B with A
G = B.union(A)
print("After joining A With B: ", C, "and joining B with A: ", G)

#What is the symmetric difference between A and B
H = A.symmetric_difference(B)
print("Symmetric difference between A and B:", H)

#Delete the sets completely
del A, B
#print(B)

#Convert the ages to a set and compare the Length of the List and the set
I = set(Age)
print("Is length of list and set is same :", len(I)==len(Age))

```

The length of set is: 7

{'Oracle', 'Amazon', 'IBM', 'Google', 'Apple', 'Microsoft', 'Twitter', 'Facebook'}

{'Samsung', 'IBM', 'Twitter', 'Apple', 'Oracle', 'Microsoft', 'Deloitte', 'Facebook', 'Amazon', 'Tesla', 'Google', 'Meta'}

{'IBM', 'Twitter', 'Apple', 'Oracle', 'Microsoft', 'Deloitte', 'Facebook', 'Amazon', 'Tesla', 'Google', 'Meta'}

After joining A and B: {19, 20, 22, 24, 25, 26, 27, 28}

After intersecting A and B: {19, 20, 22, 24, 25, 26}

Is A subset of B? True

Are A and B disjoint sets? False

After joining A With B: {19, 20, 22, 24, 25, 26, 27, 28} and joining B with A: {19, 20, 22, 24, 25, 26, 27, 28}

Symmetric difference between A and B: {27, 28}

Is length of list and set is same : False

Question 5:

The radius of a circle is 30 meters.

- Calculate the area of a circle and assign the value to a variable name of `_area_of_circle_`
- Calculate the circumference of a circle and assign the value to a variable name of `_circum_of_circle_`
- Take radius as user input and calculate the area.

Source code:

```
radius = 30
```

```
pi = 3.14
```

```
#Calculate the area of a circle
```

```
_area_of_circle_ = pi*radius*radius
```

```
print("Area of the circle is: ", _area_of_circle_)
```

```
#Calculate the circumference of a circle
```

```
_circum_of_circle_ = 2*pi*radius
```

```
print("Circumference of the circle is: ", _circum_of_circle_)
```

```
#Take radius as user input and calculate the area
```

```
radius = float(input ("Enter the radius of the circle : "))
```

```
Area= pi*radius*radius
```

```
print ("The area of the circle is", Area)
```

Output:

```
Area of the circle is:  2826.0
```

```
Circumference of the circle is: 188.4
Enter the radius of the circle : 5
The area of the circle is 78.5
```

```
In [5]: radius = 30
        pi = 3.14

        #Calculate the area of a circle
        _area_of_circle = pi*radius*radius
        print("Area of the circle is: ", _area_of_circle_)

        #Calculate the circumference of a circle
        _circum_of_circle = 2*pi*radius
        print("Circumference of the circle is: ", _circum_of_circle_)

        #Take radius as user input and calculate the area
        radius = float(input ("Enter the radius of the circle : "))
        Area= pi*radius*radius
        print ("The area of the circle is", Area)

        Area of the circle is: 2826.0
        Circumference of the circle is: 188.4
        Enter the radius of the circle : 5
        The area of the circle is 78.5
```

Question 6:

“I am a teacher and I love to inspire and teach people” • How many unique words have been used in the sentence? Use the split methods and set to get the unique words.

Source code:

```
#Printing unique_words
```

```
String = "I am a teacher and I love to inspire and teach people"
```

```
Unique_words=set(String.split(" "))
```

```
print("Unique words in a set:",Unique_words)
```

Output:

```
Unique words in a set: {'to', 'teacher', 'am', 'teach', 'I', 'people', 'and', 'inspire', 'a', 'love'}
```

```
In [6]: #Printing unique_words
String = "I am a teacher and I love to inspire and teach people"
Unique_words=set(String.split(" "))
print("Unique words in a set:",Unique_words)

Unique words in a set: {'to', 'teacher', 'am', 'teach', 'I', 'people', 'and', 'inspire', 'a', 'love'}
```

Question 7:

Use a tab escape sequence to get the following lines.

Name	Age	Country	City
Asabeneh	250	Finland	Helsinki

Source code:

```
#Sequence priting
```

```
print("Name\tAge\tCountry\tCity\nAsabeneh\t250\tFinland\tHelsinki")
```

Output:

Name	Age	Country	City
Asabeneh	250	Finland	Helsinki

```
In [7]: #Sequence priting
print("Name\tAge\tCountry\tCity\nAsabeneh\t250\tFinland\tHelsinki")

Name      Age      Country   City
Asabeneh  250      Finland   Helsinki
```

Question 8:

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.”

Source code:

```
radius = 10  
  
area = 3.14 * radius ** 2  
  
print("The area of a circle with radius {} is {} sq.meters".format(radius, int(area)))
```

Output:

The area of a circle with radius 10 is 314 sq.meters

```
In [8]: ▶ radius = 10  
        area = 3.14 * radius ** 2  
        print("The area of a circle with radius {} is {} sq.meters".format(radius, int(area)))
```

The area of a circle with radius 10 is 314 sq.meters

Question 9:

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)

Ex: L1: [150, 155, 145, 148]

Output: [68.03, 70.3, 65.77, 67.13]

Source code:

```
import math  
  
Num = int(input("Number of students:"))  
  
Lbs=[]  
  
Wts=[]  
  
for i in range(Num):  
    Lbs.append(int(input()))  
  
for b in Lbs:  
    a=(math.floor((b/2.2046) * 100 ) )/ 100;
```



```
Wts.append(a)
```

```
print(Wts)
```

Output:

```
Number of students:1
10
[4.53]
```

```
1 import math
  Num = int(input("Number of students:"))
  Lbs=[]
  Wts=[]
  for i in range(Num):
      Lbs.append(int(input()))
  for b in Lbs:
      a=(math.floor((b/2.2046) * 100 ) )/ 100;
      Wts.append(a)
  print(Wts)
```

```
Number of students:1
10
[4.53]
```

Question 10:

The diagram below shows a dataset with 2 classes and 8 data points, each with only one feature value, labeled f . Note that there are two data points with the same feature value of 6. These are shown as two x 's one above the other. Provide stepwise mathematical solution, do not write code for it. 1. Divide this data equally into two parts. Use first part as training and second part as testing. Using KNN classifier, for $K=3$, what would be the predicted outputs for the test samples? Show how you arrived at your answer. 2. Compute the confusion matrix for this and calculate accuracy, sensitivity and specificity values.

Given data elements are taken in the tabular form as below,

Feature	Label
---------	-------

1	0
---	---

2	O
3	X
6	X
6	X
7	O
10	O
11	O

Here, the first four rows of data are considered to be the Training dataset and the next four rows are selected as the Testing dataset.

Now, according to the KNN Classifier we shall now consider K=3 and then the distance between the testing and training data is demonstrated below.

In the below table the columns are the training dataset and rows are the testing dataset.

	1(O)	2(O)	3(X)	6(X)
6	5	4	3	0
7	6	5	4	1
10	9	8	7	4
11	10	9	8	5

The highlighted rows are the distance values.

Let us now assume 'O' as negative and 'X' as positive values. Now the prediction on testing data is as below:

	True label		Predicted label	O/P
6	X	X	Tp	
7	O	X	Fp	
10	O	X	Fp	
11	O	X	Fp	

Confusion matrix for the above prediction is:

TN FP

FN TP

The final confusion matrix is:

0 3 0 1

Accuracy of the classifier = $(TP + TN) / (P + N) = 1 / 4 = 0.25$

Sensitivity of the classifier = $TP / P = 1/1 = 1$

Specificity of the classifier = $TN / N = 0/3 = 0$