Machine Learning Homework 1 (Python Exercise)

Mar. 17, 2021

** Please note that all homework should be your own work. You should also not copy answers from other person's, books or internet resources.*
** I didn't proofread the questions. If you find any typos/errors, let me know.*

1. Write a Python program to count the number of strings where the string length is 3 or more and the first and last character are different from a given list of strings.
Input List : ['cabc', 'xyza', 'abbc', '13221', 'xyzk']
Output : 3

2. Write a Python program to get a list, sorted in decreasing order by the second element in each inner list from a list, using a 'lambda' function.
Input List : [[2, 6], [1, 2], [3, 4], [5, 3], [4, 1]]
Output : [[2, 6], [3, 4], [5, 3], [1, 2], [4, 1]]

* Select **ONE arff file** from e-class. Change it to csv file. The csv file must contain numbers and/or strings only, each of which is separated by commas. In dong so, you have to modify arff file by removing **header part** (% and @ part) of the data.

3. Write Python code for the following tasks
0) Convert your 'arff' file to 'csv' file, refer to the code in the site below.
https://github.com/haloboy777/arfftocsv/blob/master/arffToCsv.py

1) read csvfile into a two dimension list (called **"a_list"**) using 'csv' module.

e.g.: csvfile=

1	0	2	3	A
0	1	1	2	A
0	1	0	1	B
0	0	2	3	C

a_list=[[1,0,2,3,'A'], [0,1,1,2,'A'], [0,1,0,1,'B'], [0,0,2,3,'C']]

2) show the number of columns(attributes), number of rows(records) and number of classes (label), respectively.

e.g.: csvfile=

1	0	2	3	A
0	1	1	2	A
0	1	0	1	B
0	0	2	3	C

output : number of columns 5, number of rows 4, number of class 3

3) write a Python program that shows the first 2 rows per class from the "a_list".

e.g.: csvfile=

1.1	1.2	A
0.7	0.5	A
2.1	0.5	A
5.9	0.7	B
6.1	0.8	B
5.9	0.8	B

Output : "class A : [[1.1, 1.2], [0.7, 0.5]], class B : [[5.9, 0.7], [6.1, 0.8]]

4) write a Python program which randomly shuffles 'a_list' data using 'random' module.

4. Using the "a_list" in question 3. write Python code for the following tasks
1) given two column(attribute) index numbers, write a function that return a two-dimensional list that have the values of the columns.

2) show the reversed elements of each list of q. 1) result. (We don't actually change the values a_list)
ex) q.1) output = [[0,1,2,0], [3,4,5,6]] q.2) output = [[0,2,1,0], [6,5,4,3]]

5. Using the "a_list", write Python code for the following tasks
1) define a function "divide_train_test(in_list, prop)" function where
input: 1) in_list: a 2D list, 2) prop: proportion of training data
output: train_data (first "prop" percent of in_list), test_data (the rest of in_list)
2) run divide_train_test(a_list, prop) TWO times using prop=0.7, 0.9, respectively, and show the result.
e.g.: divide_train_test([[1,2,3], [5,1,8], [8,5,2], [0,3,6], [1,7,3]], 0.8) returns
[[[1,2,3], [5,1,8], [8,5,2]], [[0,3,6], [1,7,3]]]
train_data test_data

6. Write Python code for the tasks.
1) define a function "min_max_avg_med" which takes a list of numbers and returns [minimum, maximum, average, median] of the list. (don't use any modules such as 'import statistics')
e.g.: def min_max_avg_med(in_list):

2) Using 'random' module, randomly generate 9 integer numbers (the numbers are more than -10 and less than 10) and, calculate the minimum, maximum, average and median values of generated numbers using above "min_max_avg_med" function.
e.g.: min_max_avg_med([-9,2,1,3,7,2,3,-7,-4]) returns [-9, 7, -0.222, 2]

3) define a function "equ_interval" which divides a value range into n equal intervals.
input: 1) list [min, max] of range, 2) number of intervals
output: list of (equal distance) intervals
e.g.: equ_interval([-4, 8], 3) returns [[-4,0], [0,4], [4,8]]
4) run equ_interval 2 times by using different values of list and number of intervals.

7. Write Python code for the following tasks.
1) define a function "no_of_class_values" which takes a two-dimensional list and class value (label), and returns other elements corresponding to class value.
e.g.: no_of_class_values([[1, 1, 1, 2, 'A'], [2, 4, 3, 'B'], [2, 3, 3, 3,'A']], 'A')
=> return [[1, 1, 1, 2], [2, 3, 3, 3]]
no_of_class_values([[1, 1, 1, 2, 'A'], [2, 4, 3, 'B']], 'B') => return [2, 4, 3]
no_of_class_values([[1, 1, 1, 2, 'A'], [2, 4, 3, 'B']], 'C') => return None

2) define a function "no_of_dis_val" which takes a list and returns the number of "distinct" values in the list.
e.g.: a_list=[9,9,8,7,7,8]
no_of_dis_val(a_list) returns 3 ==> 3 unique values
This means a_list contains 3 distinct values

3) for every attribute in "a_list" (except the class attribute), calculate the number of distinct values for each class, using q 1) and q 2).

e.g.: csv_file=

c1	c2	c3	class
1.1	1.5	0.7	A
0.9	1.1	0.2	A
1.5	0.2	0.1	B

output= A:5 B:3

This means class A has 5 distinct values, and class B has 3 distinct values.

4) plot a graphic table(e.g.: bar graph) by your favorite color using matplotlib as follows: X axis: class names, Y axix: the number of distinct values.

Hand In

1) In your report

...

6. .Write Python code for the tasks...

1) define a function "min_max" ...

<PROGRAM CODE> <== This is title

put your program code segment for Q 6 1) here

<RESULT> <== This is title

put the screen dump of your program run for Q 6 1) here

2) randomly generate ...

<PROGRAM CODE> <== This is title

put your program code segment for Q 6 2) here

<RESULT> <== This is title

put the screen dump of your program run here

...

2) upload the following files **separately** at e-class.

i) report file, ii) python program code

Due: 3/31(wed) 11:59PM