KANIGALPULA GEETHA SAI THRILOKYA
Y20CS078

Lab Cycle-1

1. Down load the dataset winequality-red.csv file(each column is separated by a semicolon (;)) from the UCI Machine Learning Repository

```
import numpy as np
np.set_printoptions(suppress=True)
wines=np.genfromtxt("C:/Users\y20cs78\Desktop\winequality-red.csv",delimiter=";",skip_header=1)
```

2. Convert it to numPy array, name it as wines (leave the first row of the list) and specify the data type of array as float.

```
wines.dtype="float"
```

3. Identify the shape of the array.

```
print("shape of the array",wines.shape)
```

Output:

```
shape of the array (1599, 12)
```

4. Display the element at row 3 and column 4.

```
print("element at row 3 column 4 is ",wines[2,3])
```

Output:

```
element at row 3 column 4 is  2.3
```

5. Display the first three items from the fourth column.

```
print("first 3 items from 4th column is:",wines[0:3,3])
```

Output:

```
first 3 items from 4th column is: [1.9 2.6 2.3]
```

6. Display third column from each row.

```
print("items in 3 rd column",wines[:,2])
```

Output:

```
items in 3 rd column [0.   0.   0.04 ... 0.13 0.12 0.47]
```

7. Display fourth row.

```
print("fourth row:",wines[3])
```

Output:

```
fourth row: [11.2    0.28   0.56   1.9    0.075 17.    60.    0.998  3.16   0.58  9.8    6.   ]
```

8. Assign value 10 to 2nd row and 6th column element.

```
wines[1,5]=10
```

9. Take the 10th column from wines array and name that slice as slice_new and assign value 666 to all elements of slice_new.

```
slice_new=wines[:,9]
slice_new[:]=666
print(slice_new)
```

Output:

```
array([666., 666., 666., ..., 666., 666., 666.])
```

10. Display wines array.

```
print("wines array",wines)
```

Output:

```
wines array [[  7.4     0.7     0.    ... 666.      9.4     5.   ]
 [  7.8     0.88    0.    ... 666.      9.8     5.   ]
 [  7.8     0.76    0.04  ... 666.      9.8     5.   ]
 ...
 [  6.3     0.51    0.13  ... 666.     11.      6.   ]
 [  5.9     0.645   0.12  ... 666.     10.2     5.   ]
 [  6.      0.31    0.47  ... 666.     11.      6.   ]]
```

11. Find the data type of wines array and Change the data type to int.

```
print("datatype of wines array before changing",wines.dtype)
wines=wines.astype(np.int32)
print("datatype of wines array after changing",wines2.dtype)
```

Output:

```
datatype of wines array before changing float64
datatype of wines array after changing int32
```

12. Add 10 points to each quality score.

```
wines[:,11]=wines[:,11]+10
print("after adding 10 to quality column")
wines[:,11]
```

Output:

```
after adding 10 to quality column
array([15, 15, 15, ..., 16, 15, 16])
```

13. Find the sum of all the elements in an array

```
s=np.sum(wines)
print("sum of all elements",s)
```

Output:

```
sum of all elements 1226388
```

14. Find the sum of all the values in every column.

```
print("sum of all values in every column",np.sum(wines,axis=0))
```

Output:

```
sum of all values in every column [  12589      24       1    3350       0   25367   74301      81
4770
 1064934   15969   25002]
```

15. Find the sum of all the values in every row.

```
print("sum of all values in every row",np.sum(wines,axis=1))
```

Output:

```
sum of all values in every row [746 779 771 ... 773 777 765]
```

16. Add the quality column to itself.

```
wines[:,11]=wines[:,11]+wines[:,11]
#wines[:,11]=wines[:,11]*2
print("after adding quality column to itself")
wines[:,11]
```

Output:

```
after adding quality column to itself
array([30, 30, 30, ..., 32, 30, 32])
```

17. Multiply alcohol by quality.

```
a=wines[:,10]*wines[:,11]
print(a)
```

Output:

```
array([270, 270, 270, ..., 352, 300, 352])
```

18. Display which wines have a quality rating higher than 5.

```
wines[wines[:,11]>5]
```

Output:

```
array([[  7,   0,   0, ..., 666,   9,  30],
       [  7,   0,   0, ..., 666,   9,  30],
       [  7,   0,   0, ..., 666,   9,  30],
       ...,
       [  6,   0,   0, ..., 666,  11,  32],
       [  5,   0,   0, ..., 666,  10,  30],
       [  6,   0,   0, ..., 666,  11,  32]])
```

19. Check if any wines have a quality rating equal to 10.

```
wines[wines[:,11]==10]
```

Output:

```
array([], shape=(0, 12), dtype=int32)
```

20. Select rows in wines where the quality is over 7

```
wines[wines[:,11]>7]
```

Output:

```
array([[  7,   0,   0, ..., 666,   9,  30],
       [  7,   0,   0, ..., 666,   9,  30],
       [  7,   0,   0, ..., 666,   9,  30],
       ...,
       [  6,   0,   0, ..., 666,  11,  32],
       [  5,   0,   0, ..., 666,  10,  30],
       [  6,   0,   0, ..., 666,  11,  32]])
```

21. Display wines with alcohol greater than 10 and quality greater than 7.

```
wines[(wines[:,10]>10) & (wines[:,11]>7)]
```

Output:

```
array([[ 4,   0,   0, ..., 666,  13,  28],
       [ 4,   0,   0, ..., 666,  12,  32],
       [ 5,   0,   0, ..., 666,  13,  30],
       ...,
       [ 5,   0,   0, ..., 666,  11,  32],
       [ 6,   0,   0, ..., 666,  11,  32],
       [ 6,   0,   0, ..., 666,  11,  32]])
```

22. Change the shape of wines array.

```
wines.reshape(12,1599)
```

Output:

```
array([[ 7,   0,   0, ...,   6,   0,   0],
       [ 1,   0,  17, ...,   3,   0,  26],
       [ 61,   1,   3, ...,  42,   0,   3],
       ...,
       [ 2,   0,  23, ...,   1,   0,   6],
       [ 25,   0,   3, ...,  53,   0,   3],
       [666,   9,  30, ..., 666,  11,  32]])
```