

Problem 1

Write a code for that carries out leave one out cross validation and displays the classification rate. You should use the dataset as input and the output will be the average classification rate.

Solution: For classes $\{\omega_i\}_{1 \leq i \leq n}$, we assign observation x to class ω_i if $P(i|x) \geq P(j|x)$ for all $j \neq i$. We assume dataset of each class follows a normal distribution. Let the PDF for class i be given by p_i . Then:

$$p(i|x) = \frac{p(x|i)p(i)}{p(x)} = p_i(x) \times \text{No. of obs from class } i \times \frac{1}{p(x) \times \text{Total No. of obs}}$$

As $\left(\frac{1}{p(x) \times \text{Total No. of obs}}\right)$ is constant for a data set, we can consider $g_i(x) = p_i(x) \times \text{No. of obs from class } i$ to be the discriminant function. For IRIS data we get a classification rate of 97.33% and for wine data we get a rate of 99.44%. The relevant code is given below

```

1  %Closing everything for next function call
2  close ALL;
3  clear ALL;
4
5
6  %data Prep of IRIS data
7  load("irisdata.mat")
8  data(:,5)=data(:,5)+1;
9
10
11 %Function Call
12 disp("IRIS data LOO rate:")
13 disp(LOO_CV(data));
14
15 %Closing everything for next function call
16 close ALL;
17 clear ALL;
18
19 %Importing Wine dataset
20
21 load("wine_dataset.mat")
22
23 labels=wineTargets.';
24 labelsRow=labels(:,1)+labels(:,2)*2+labels(:,3)*3;
25 data=horzcat(wineInputs.',labelsRow);
26
27
28 %Function Call
29 disp("Wine data LOO rate:")
30 disp(LOO_CV(data));
31
32 function rate=LOO_CV(data)
33
34 %clearing
35 close ALL;
36 clear ALL;
37 %getting rows columns and classes
38 [r,c] = size(data);
39 [class,~]=size(unique(data(:,c)));
40
41 %We set the arr_rate[i] to the error on classifying data[i] as Cross
42 %checking data. It is 1 for proper classification, 0 otherwise
43
44 arr_rate=zeros(r,1);

```

```

45
46 for i=1:r
47     arr_g=zeros(class,1);
48     LOO_data=data([1:i-1,i+1:r],:);%Creating Dset without i^{th} row
49     for j=1:class
50
51         lastColumn = LOO_data(:, c);
52         LOO_data_class=LOO_data(lastColumn==j,:);
53         mu=mean(LOO_data_class(:,1:c-1));
54         Sigma=cov(LOO_data_class(:,1:c-1));
55         prior=size(LOO_data_class,1);
56         arr_g(j)=mvnpdf(data(i,1:c-1),mu,Sigma)*prior;
57
58     end
59     [~,I]=max(arr_g);
60     if I==data(i,c)
61         arr_rate(i)=1;
62     else
63         arr_rate(i)=0;
64     end
65
66 end
67 rate=sum(arr_rate)/r;
68 end

```
