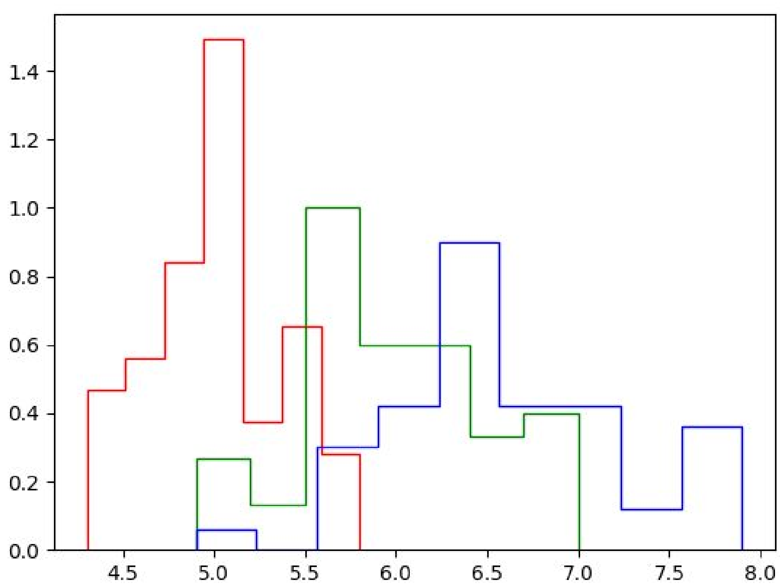
Q1:
I felt that the certain things required were Domain knowledge specific though answers are given as per the Data provided and written in description.

S.No		1	2	3	4	5	6	7	8
	NAME OF THE DATASET	IMAGE SEGMENTATION	IONOSPHERIC	IRIS	LETTER RECOGNITION	LIVER DISORDERS	SOLAR FLARE	SPECT HEART	WISCONSIN BREAST CANCER
1	Feature Vector Dimension	19	34	4	17	7	13	45	31
2	No. of Classes	7	2	3	26		3	2	2
3	Prior Prob. for Each class	1/7	g:225/351 b:126/351	1/3	1/26		1/3	1/2	1/2
4	Mean Vector Dimension	19	34	4	17	7	13	45	31
5	Covariance Matrix Dimension	19 * 19	34 * 34	4 * 4	17 * 17	7 * 7	13 * 13	45 * 45	31 * 31

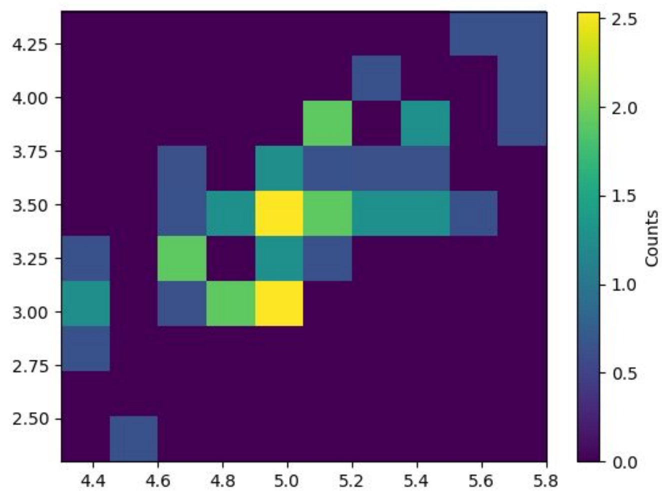
Q2:

For this problem, I, selected the Iris dataset 3 having 3 classes.
The following plots are obtained. Code is in the file hist.py (python3)

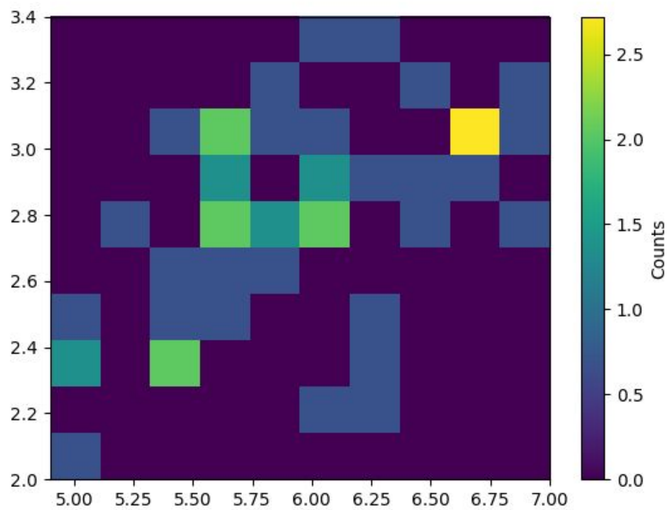
In the plot below 1D histogram of three classes, Iris-setosa(class1 in red), Iris-versicolor(class2 in green), Iris-virginica(class3 in blue) are plotted with sepal length in cm as feature.



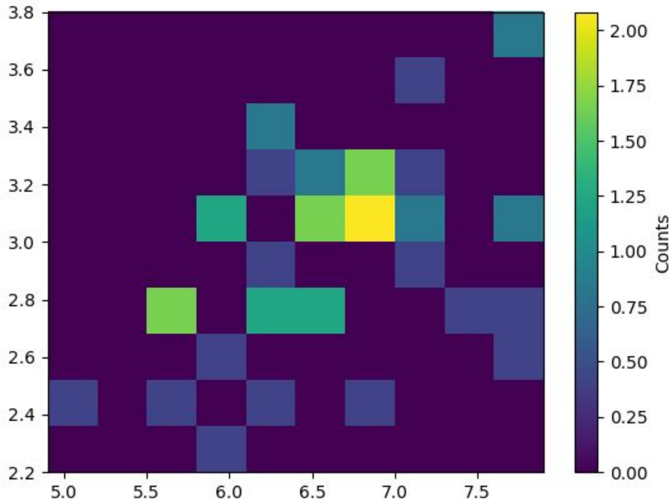
2D Histogram of class1 (Iris-setosa) with sepal length in cm and sepal width in cm as features.



2D Histogram of class2 (Iris-versicolor) with sepal length in cm and sepal width in cm as features.



2D Histogram of class3 (Iris-virginica) with sepal length in cm and sepal width in cm as features.



xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

Q3:

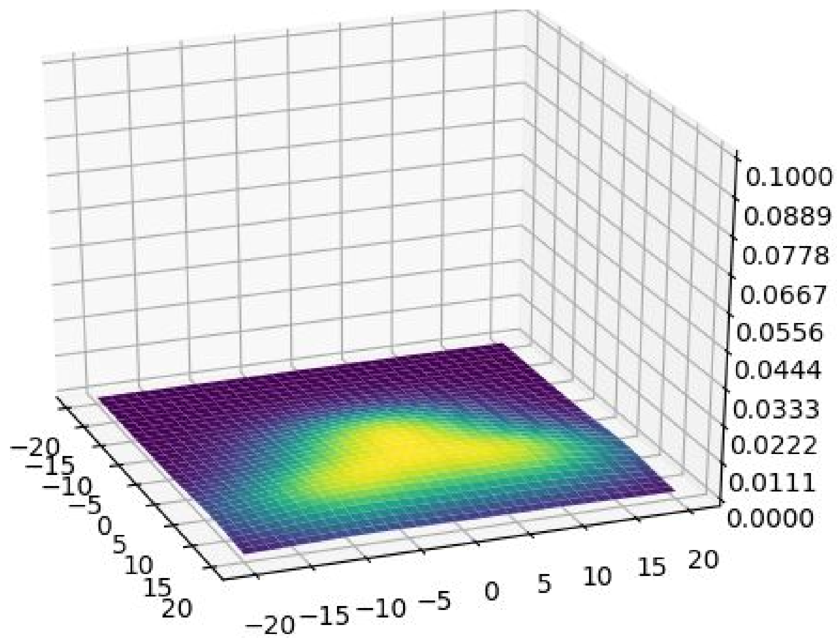
For the first part use file linearly_separable.py and helper.py. Execute the first.

The data was first imported from the text file and store into lists namely class1, class2, class3 as described in the question.

Each class was then sliced into two list training dataset and testing dataset. The method used to slice is taking the assumption that data is random and consistent.

Case 1:
Same Covariance Matrix for all the classes.

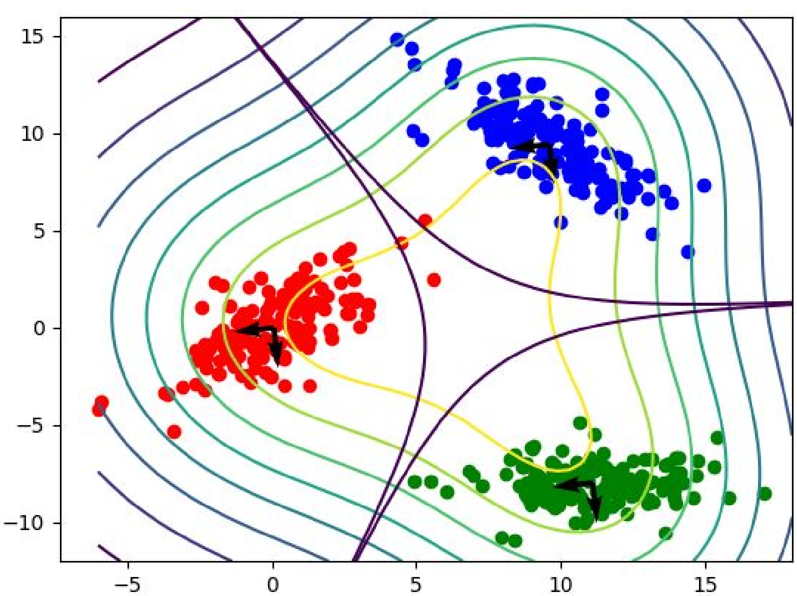
The results were confusing as the plot was overlapping and didn't have much z component. Slight elevation or curve can be seen if observed closely.



The contour, eigenvectors and the scatter plot of test data point was pretty simple and inline with the expectation.

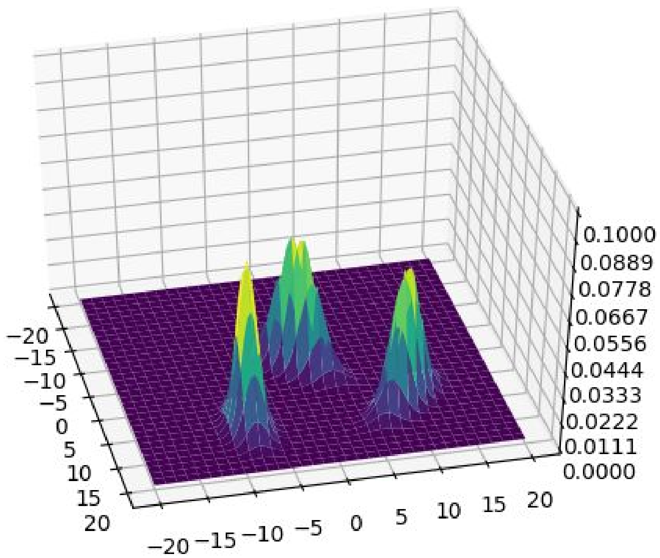
The decision boundary may be messy has I tried various ways to get the required plot, I am not aware of the actual method of making the decision boundary but intuition and calculation wise these line serve the purpose.

Here class1 test data points are in red class2 in blue and class3 in green.

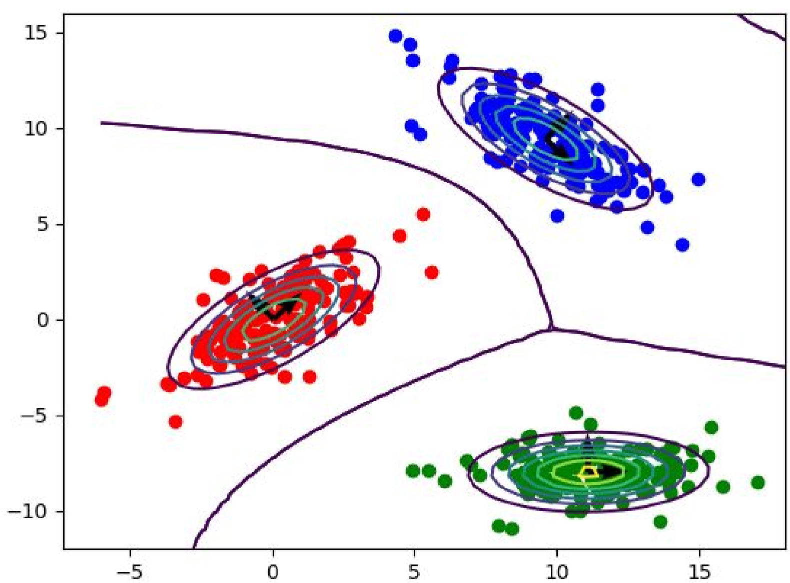


Case 2
Different Covariance matrix.

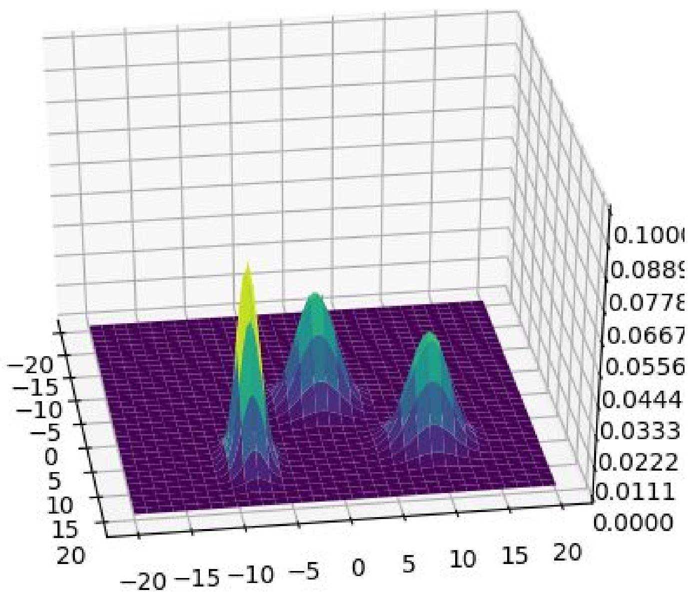
Results were as expected, I even verified by calculating some of the points. The plots are separable and test data shows no error.



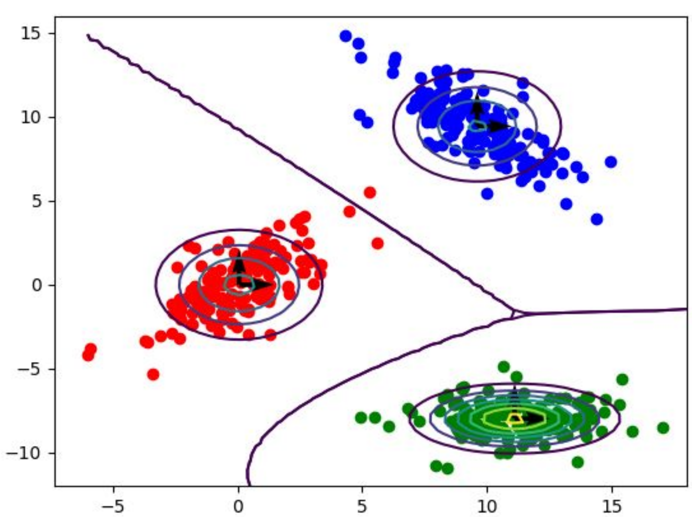
Here class1 test data points are in red class2 in blue and class3 in green.



Case 3:
Different Diagonal Covariance Matrix



Here class1 test data points are in red class2 in blue and class3 in green.



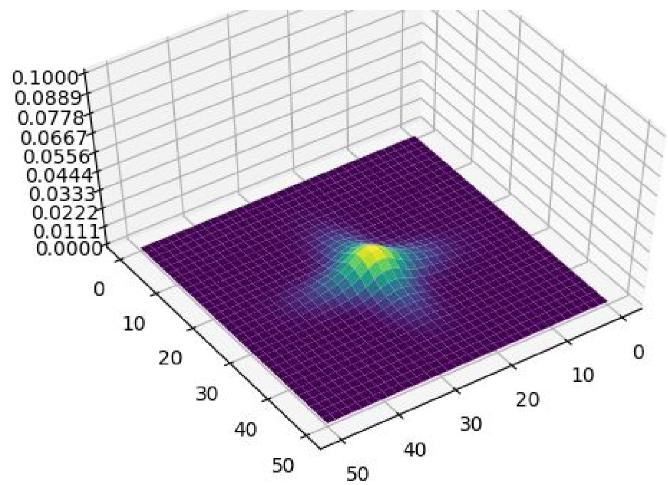
For the second part please run non-linear.py file

Case :

Different covariance matrix

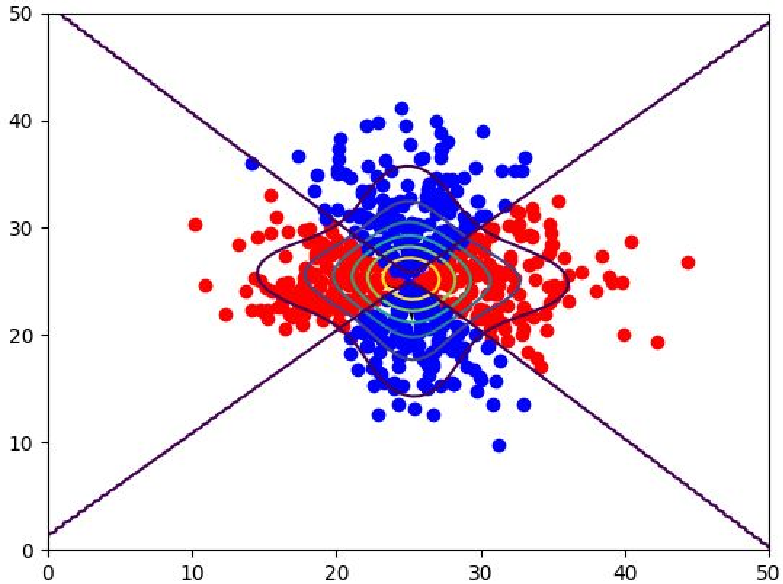
As above the data was extracted and stored in class1 and class2 respectively and divided into training and testing dataset.

The plot was overlapping alot like two elongated perpendicular bells cutting each other at center.
Still the decision boundary got good output.



Here class1 test data points are in red and class 2 data points in blue

The black line are decision boundary, though eigenvectors are there they are not visible to scatter points but can be check in code and plots i have added below without scatter points.

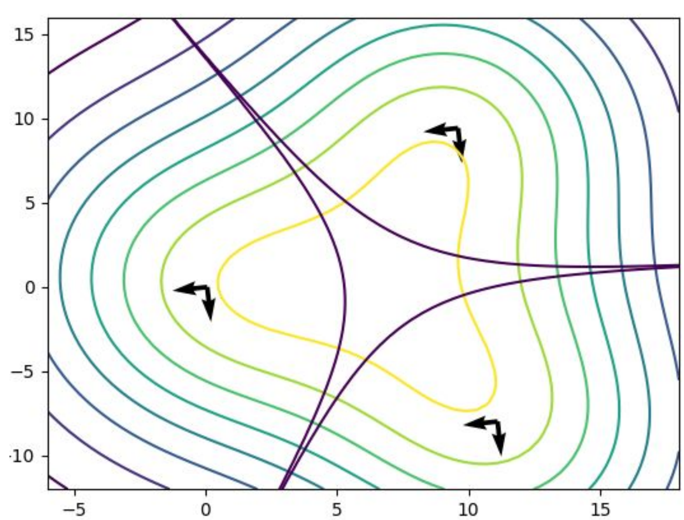


Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

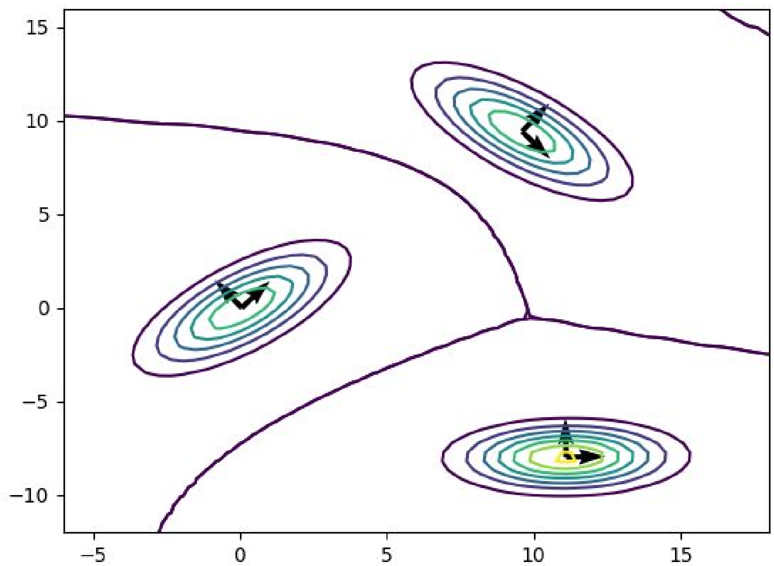
Extra Plots:

Contour plots without the test data points so that eigenvectors are clearly visible.

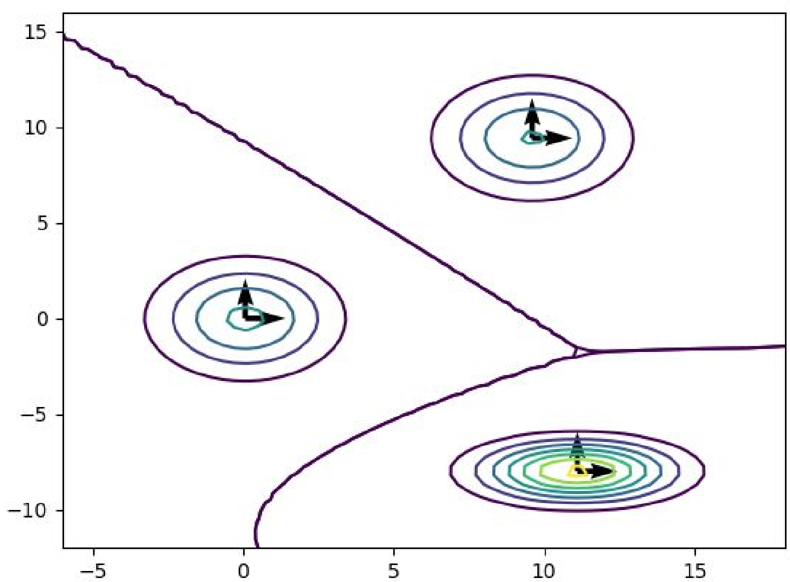
Q3 part 1 case 1:



Q3 part 1 case 2:



Q3 part 1 case 3:



Q3 part 2 case:

Red for class 1 and black for class2 eigenvectors.