**MACHINE LEARNING FROM DATA**

**Report: Lab Session 1 – MAP and Gaussian data**

**Classification criteria based on maximizing posterior probability**

**Names: Abdullah Qureshi, S M Rakib Hasan and Niina Hietamäki**

**Instructions**

Handling your work:

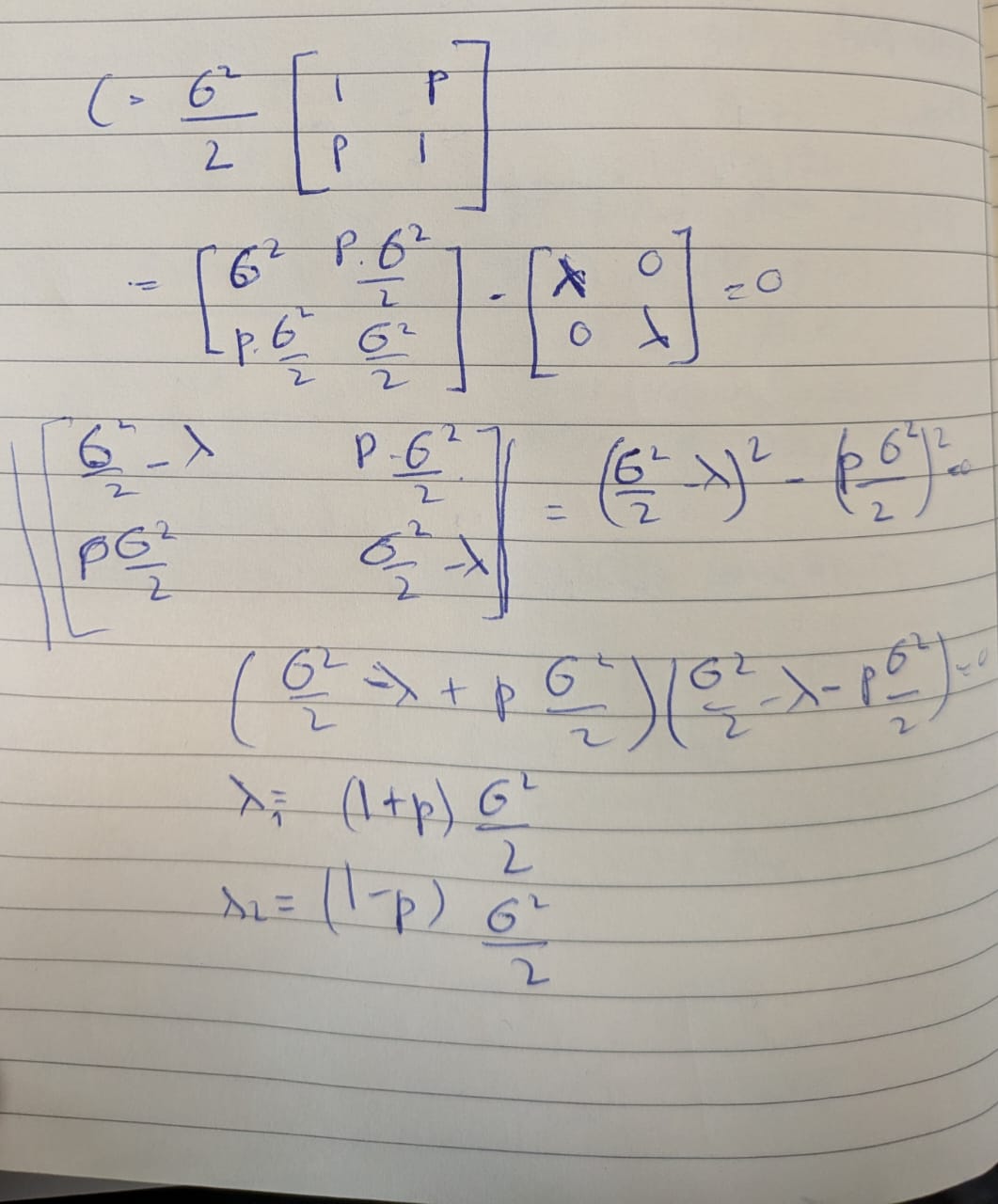
* Answer the questions in this document with the name **Mlearn\_Lab1\_report\_team\_surnames.doc**

**Questions**

Q1: Derive the expression for the eigenvalues of the matrix as a function of the parameters and . (edit equations or solve by hand and scan and insert an image with the solution)



solved



Q2. Create a table including error probabilities obtained by the linear classifier (LC) and error probabilities obtained by the quadratic classifier (QC), for each SNR value on the test set. Discuss the results.

|  | 3 dB | 0 dB | -3 dB | -10 dB |
| --- | --- | --- | --- | --- |
| LC | 0.009500 | 0.025500 | 0.111000 | 0.289000 |
| QC | 0.010000 | 0.026000 | 0.111000 | 0.292500 |

As the SNR decreases the error probabilities for both the linear classifier and quadratic classifier increase. This makes sense as the noise level increases it becomes harder to classify the two classes accurately. Both these classifiers perform similarly across all SNRs but linear classifier has smaller error probabilities in most cases

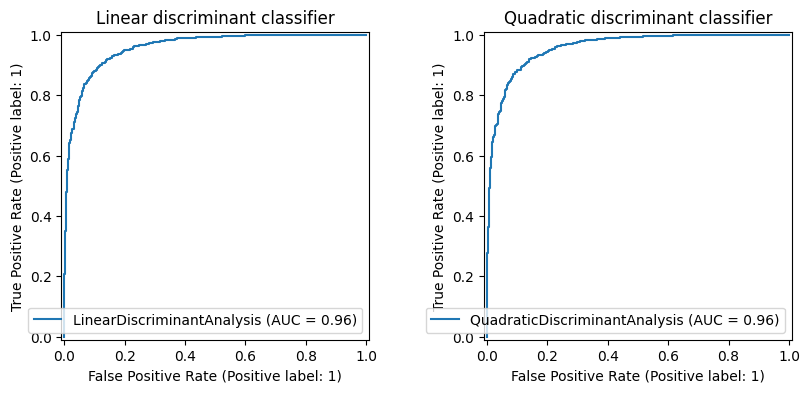
Q3. Include in the report the confusion matrices obtained for SNR=-10db and SNR=-3dB and the two classifiers on the test set. Discuss the results.

|  | -3 dB | -10 dB |
| --- | --- | --- |
| LC | [[881 119]  [103 897]] | [[708 292]  [286 714]] |
| QC | [[881 119]  [103 897]] | [[702 298]  [287 713]] |

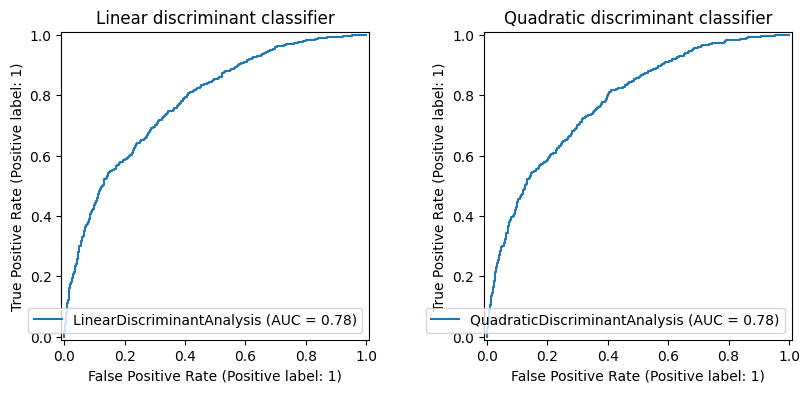
At -3 dB, both Linear and Quadratic classifiers performed well correctly classifying most samples. However, at -10 dB their performance dropped significantly with both classifiers misclassifying many more samples. This is because of increase in the noise level which makes it hard for the classifier to distinguish between 2 classes. While the Linear Classifier was slightly better at this lower SNR both classifiers showed almost similar results indicating that the added complexity of QDA did not provide a noticeable advantage in this case.

Q4. Include in the report the ROC curves obtained for SNR=-10db and SNR=-3dB and the two classifiers on the test set. Discuss the results.

**ROC Curve -3dB**



**ROC Curve -10dB**



At SNR = -3 dB the ROC curves for both Linear Discriminant Analysis and Quadratic Discriminant Analysis are closer to the top left corner indicating high accuracy and low false positives with the area under the curve near to 1. This shows effective class separation.

However, at SNR = -10 dB, the curves shift downward and to the right, reflecting increased misclassifications at higher noise levels resulting in a lower AUC. This shows poorer performance. Linear and Quadratic classifier curves are almost same which means that the Linear classifier performs almost the same as quadratic classifier under these challenging conditions.

Q5. Compute the Mahalanobis distance between the two classes on the test set for SNR= 3, 0, -3,-10 dB. Compare the results. Explain why the result differs depending on the order of the parameters.

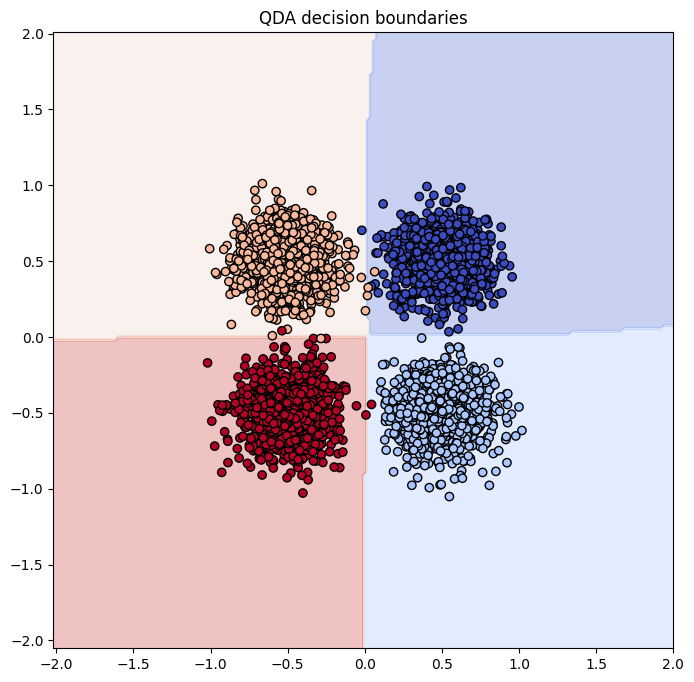
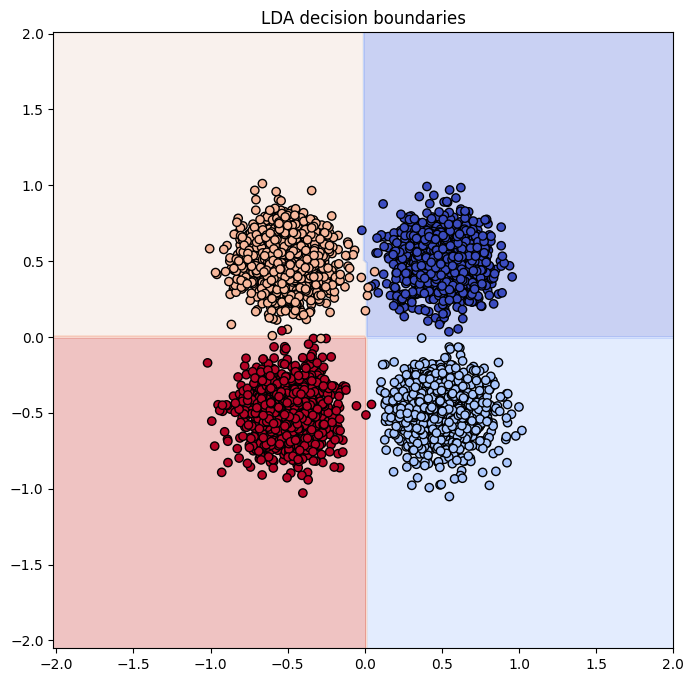
|  | 3 dB | 0 dB | -3 dB | -10 dB |
| --- | --- | --- | --- | --- |
| 00 | 26.46 | 14.93 | 8.94 | 0.289000 |
| 01 | 26.82 | 14.94 | 9.04 | 0.292500 |

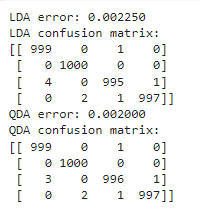
As the SNR decreases from 3 dB to -10 dB, the Mahalanobis distance between the two classes drops significantly. This means that the classes become harder to distinguish due to increasing noise. At 3 dB, the distance is high (around 26) showing clear separation, at -10 dB, the distance is very small (around 0.29), meaning the classes almost overlap.

The slight differences between the "00" and "01" distances are due to minor numerical variations when switching the order of the parameters but they don't affect the overall result.

**QPSK and covariances of all classes identical but arbitrary (case 2)**

Q6. Include the scatter plot, decision boundary, confusion matrices and error probabilities obtained using the linear classifier (LC) and the quadratic classifier (QC) for *ρ* = 0. Compare the metrics for the two classifiers and discuss the results.



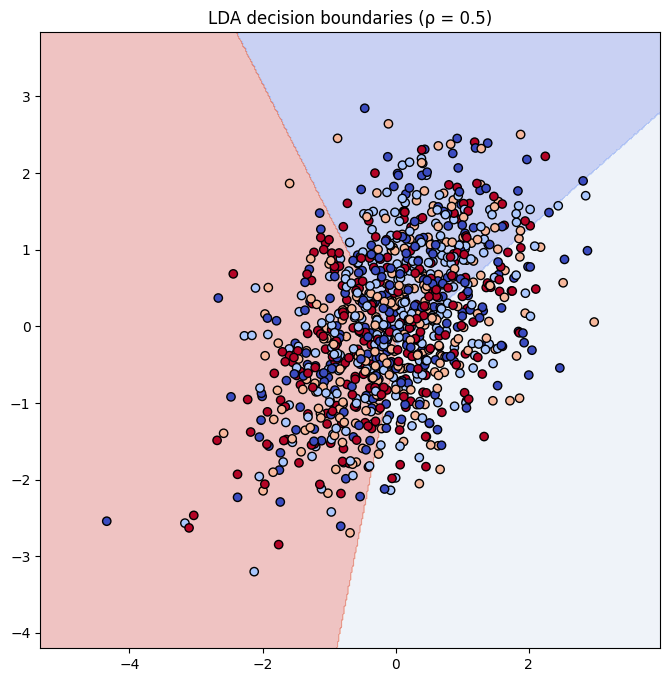
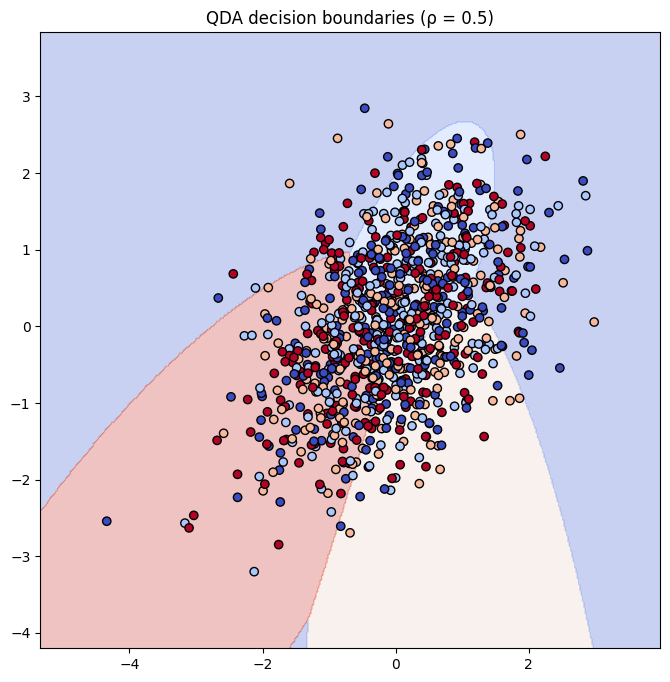


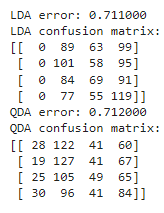
Scatter Plots and Decision Boundaries: QC decision boundaries, on the other hand, shows some curvature, especially at the upper boundary regions, because QC can create non-linear, quadratic decision boundaries, making it more flexible compared to LC.

Confusion Matrices: LC has almost perfectly classified the data points, with only a few misclassifications. Specifically, there are 4 misclassifications in class 2 and 2 misclassifications in class 3. The QC confusion matrix shows nearly identical results to LC, but there are fewer misclassifications in class 2 (only 3) and one fewer misclassification in class 3.

Error Rates: LC is 0.00225 (or 0.225%), indicating that the classifier made only a small number of errors and QC is slightly better at 0.00200 (or 0.2%), showing that QDA performed slightly more accurately in this case.

Summary: Both classifiers performed extremely well, and the difference between them is small. However, since QC can adapt better to more complex boundaries, it provides slightly better accuracy than LC in this situation where the data points are well-separated.

Q7. Repeat the previous analysis (Q6) for *ρ* = 0,5. Compare the metrics for the two classifiers and discuss the results.

Scatter Plots and Decision Boundaries: In the LC decision boundary, the boundaries are linear but don't seem to separate the classes very well due to the correlation. In contrast, QC decision boundaries show significant curvature and better adjust to the data, especially around the upper and lower regions. 

Confusion Matrices:The LC confusion matrix shows more misclassifications, especially due to the inability to handle the correlation well. In comparison, the QC confusion matrix has fewer misclassifications, as its quadratic nature helps it better distinguish between classes.

Error Rates: LC error rate is higher due to its rigid, linear decision boundaries that struggle with the correlation between features. QC performs better than LC because its non-linear boundaries adapt more effectively to the increased complexity of the data, resulting in a lower error rate.

Summary:Overall, QC performs better in this case due to its adaptability to non-linear boundaries, which is crucial when there is significant correlation between features. While LC struggles with misclassifications, QC is better suited for the task, demonstrating improved accuracy.

Q8. Compare and discuss the results obtained in Q6 and Q7

In Q6, with ρ = 0, both LC and QC did a good job, with only a few misclassifications. QC did a bit better overall because its non-linear decision boundaries made it easier to handle more complex situations compared to LC. The error rates were low for both, but QC was a bit more accurate.

Q7 As correlation increases (ρ = 0.5), the classifiers face more difficulties in separating the classes. LC performs worse because its linear decision boundaries cannot handle the complexity introduced by correlation, while QC adapts better with its non-linear decision boundaries, resulting in lower error rates.

Summary: In both cases (Q6 and Q7), QC does better than LC, especially when the data becomes more complex. Both classifiers handle the simpler situations just fine, but QC's non-linear decision boundaries help it adapt better as the correlation increases, which leads to fewer errors. LC, on the other hand, has a harder time dealing with more complex relationships in the data, which causes more misclassifications. Overall, QC comes out as the more reliable option in these scenarios.

**QPSK and different covariance matrices (case 3)**

Q9. Include the error probabilities obtained using the linear classifier (LC) and the quadratic classifier (QC) for SNR = +5 dB and +10 dB. Compare the metrics for the two classifiers and discuss the results.

|  | 5 dB | 10 dB |
| --- | --- | --- |
| LC | 0.722000 | 0.734000 |
| QC | 0.716000 | 0.725000 |

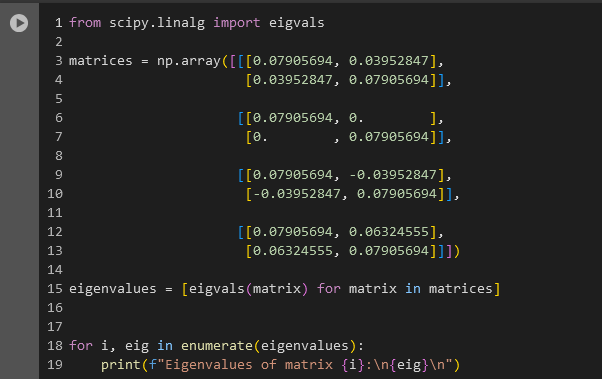
At 5 dB, the linear classifier (LC) has an error of 0.722, and the quadratic classifier (QC) shows a slightly better performance with an error of 0.716. Both classifiers perform similarly under higher noise conditions, with QC showing only a marginal improvement due to its ability to handle more complex decision boundaries. QC performs slightly better than LC at 5dB, as it can handle noise more effectively with its non-linear decision boundaries. Surprisingly, QC's error rate increases at 10 dB, while LC remains almost the same. This could be due to QC's more complex decision boundaries.

At 10 dB, the LC error increases slightly to 0.734, while QC still performs better with a reduced error of 0.725. Even though noise is lower at 10 dB, the performance gap between the two classifiers is still minor, but QC continues to have the advantage.

Q10. Complete the table with the theoretical eigenvalues using the formula obtained when answering Q1, and the eigenvalues computed using the **sample** data covariance matrices. Add the code to compute the eigenvalues of each covariance matrix; use scipy.linalg.eigvals (for just one SNR value).

Answer:

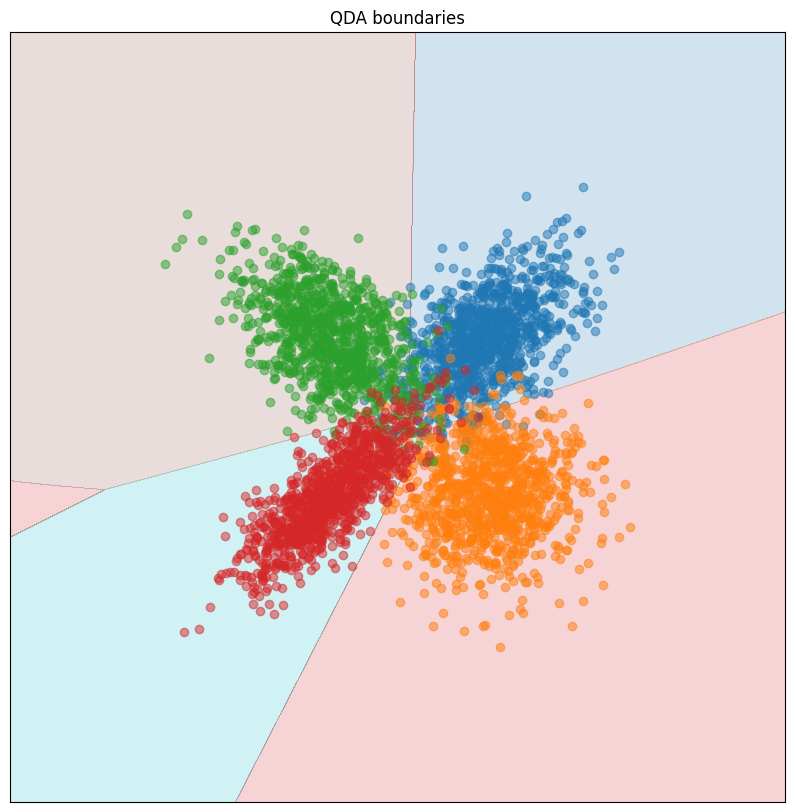
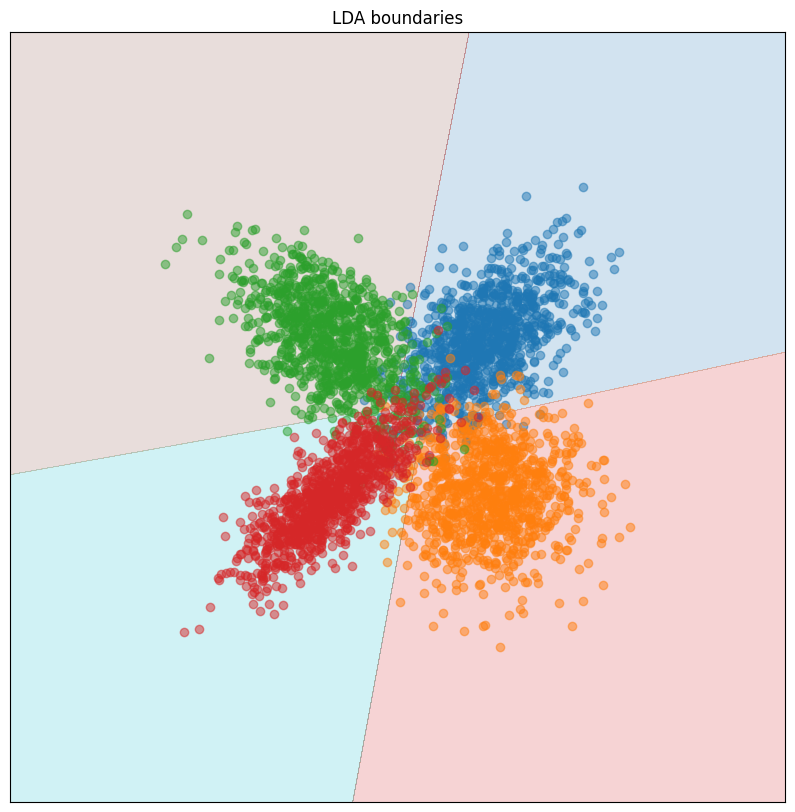
Code to calculate eigenvalues-



| SNR=5 | Class 1 | Class 2 | Class 3 | Class 4 |
| --- | --- | --- | --- | --- |
| Theoretical eigenvalues | 3  1 | 3  1 | 3  1 | 3  1 |
| Eigenvalues from sample covariance matrices | 0.118585 0.039528 | 0.079057 0.079057 | 0.118585 0.039528 | 0.142302 0.015811 |

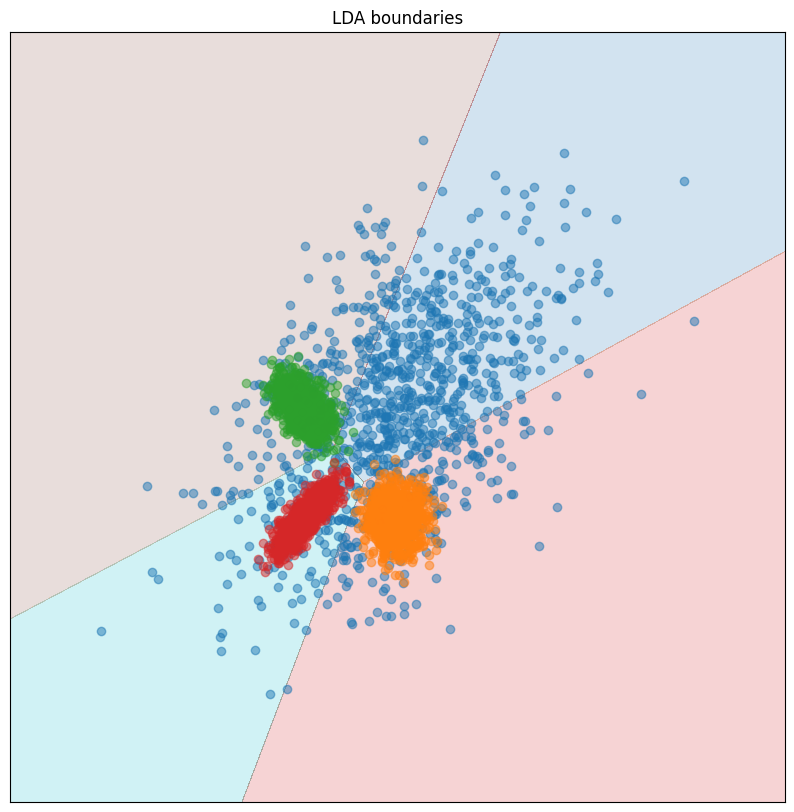
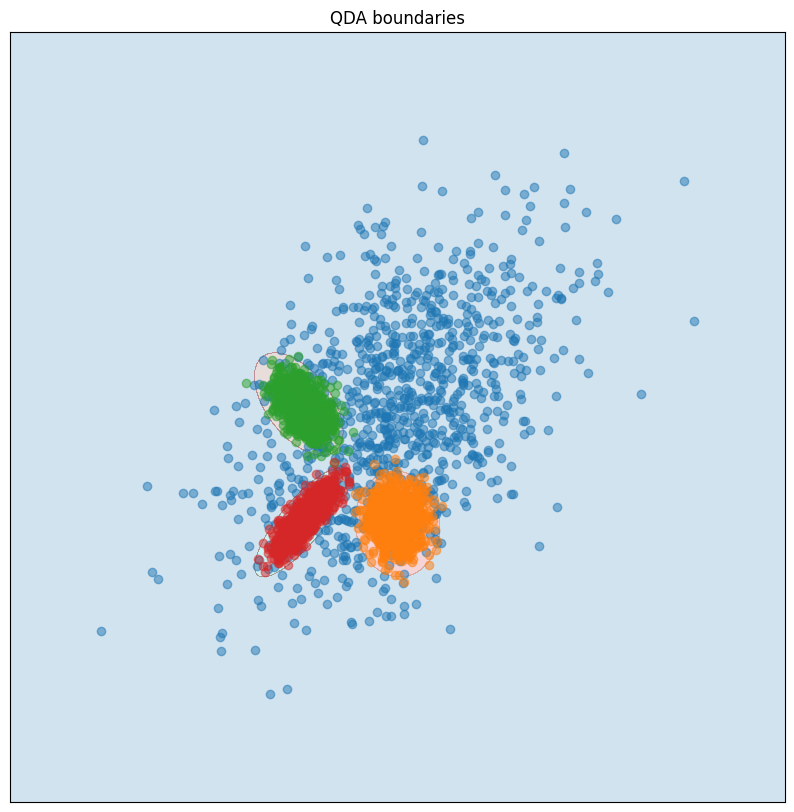
Q11. Include scatter plots for the linear and quadratic classifiers using SNR= +5 dB and SNR= +10 dB. Relate the shape of the clusters with the eigenvalues of the covariance matrices.

For SNR= +5 dB,



* **Class 1 & 3** (eigenvalues: 0.118585, 0.039528): The clusters are **elongated** in one direction, with more spread along the larger eigenvalue.
* **Class 2** (eigenvalues: 0.079057, 0.079057): The cluster is **circular** because the spread is equal in all directions (equal eigenvalues).
* **Class 4** (eigenvalues: 0.142302, 0.015811): The cluster is **strongly elongated**, with the large eigenvalue showing much more spread along one axis compared to the other.

For SNR +10 dB,



**Class 1**: Has a large spread along one axis (eigenvalue 1.125) and a moderate spread along the other (0.375). This creates an elongated cluster.

**Class 2**: Identical eigenvalues (0.025, 0.025) produce a circular cluster with low variance.

**Class 3**: Moderate elongation, slightly more spread in one direction.

**Class 4**: Strong elongation with one axis having much larger variance (0.045 vs. 0.005).

Q12. Include error probabilities, scatter plots and decision boundaries. Compare the performance of the classifier and justify the results. Include in your answer the new value of sigma[0].

Answer:

Error Probabilities-

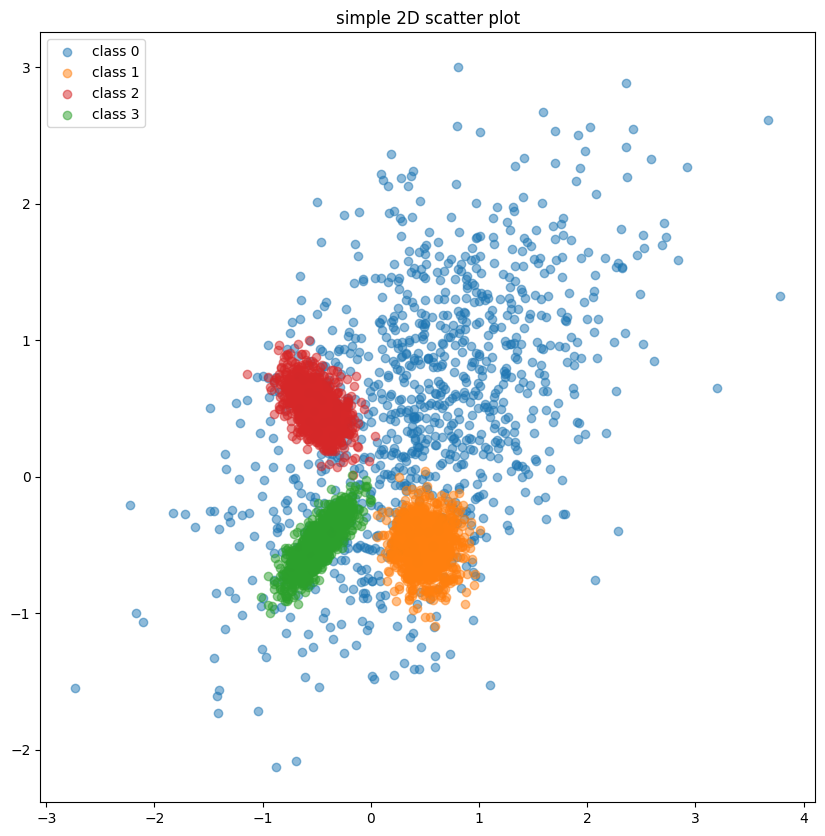
LC error (SNR = +5 dB): 0.715000

QC error (SNR = +5 dB): 0.707000

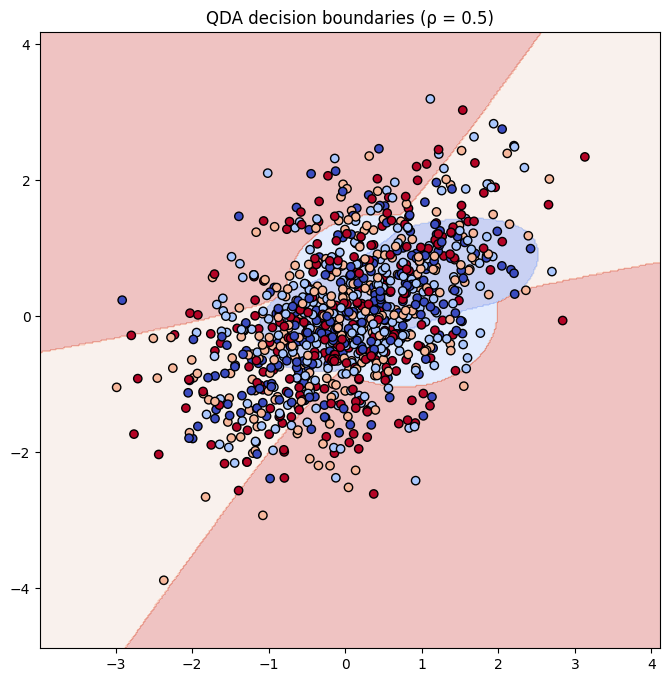
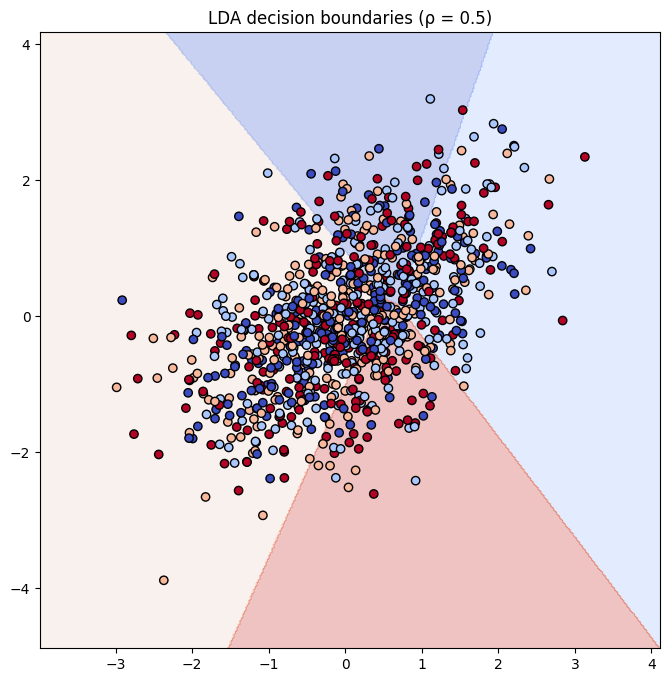
LC error (SNR = +10 dB): 0.719000

QC error (SNR = +10 dB): 0.706000

Scatter plots-



Decision Bounadry-



New sigma value= [1.5 0.05 0.05 0.05]

#### For ρ=0:

* LDA:
  + Error: 0.13275 (13.3%)
  + Confusion Matrix:
    - Class 1 misclassifies samples across the other classes.
    - Classes 2, 3, and 4 show almost perfect classification.
* QDA:
  + Error: 0.05375 (5.4%)
  + Confusion Matrix:
    - Class 1 has moderate misclassifications, but overall QDA does significantly better than LDA in this scenario.

#### For ρ=0.5:

* LDA:
  + Error: 0.737 (73.7%)
  + Confusion Matrix:
    - High misclassification across all classes, with very poor performance.
* QDA:
  + Error: 0.711 (71.1%)
  + Confusion Matrix:
    - Slightly better than LDA, but still poor performance across all classes with significant misclassification.

For ρ=0, where features are uncorrelated, both LDA and QDA perform well, but QDA slightly outperforms LDA since it models each class with its own covariance matrix. LDA assumes shared covariance, which works reasonably well when features are independent. However, when ρ = 0.5, indicating positive correlation between features, both LDA and QDA struggle. LDA performs poorly because it assumes linear separability and shared covariance across all classes, which is violated by the correlation. QDA, though more flexible, also performs worse because the increased correlation distorts class boundaries, making it difficult for both classifiers to separate the classes effectively.