**MACHINE LEARNING FROM DATA**

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

**Classification criteria based on maximizing posterior probability**

**Names:**

Miquel Gomez i Esteve

Ricard Abril Ferreres

**Instructions**

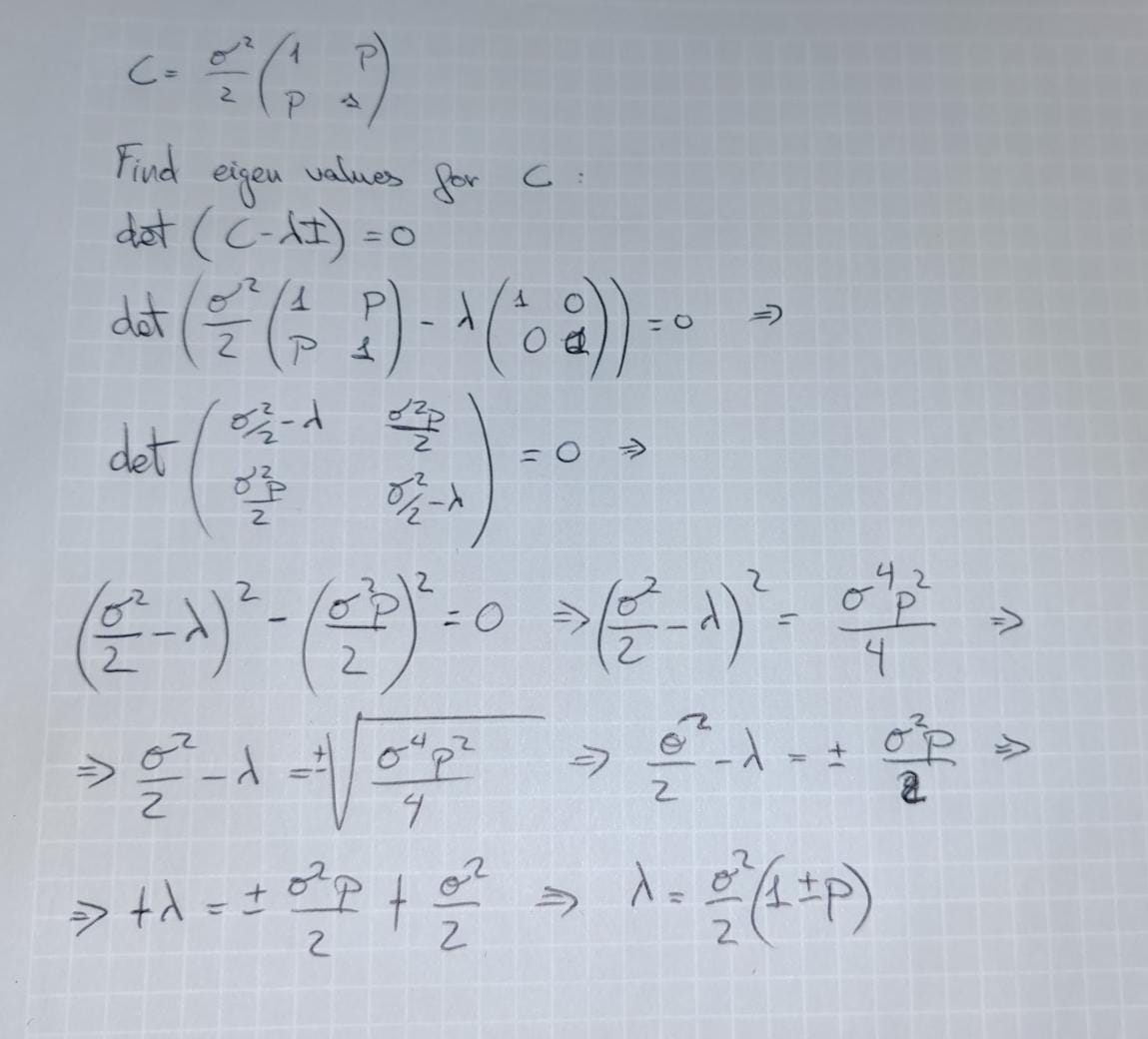
Handling your work:

* 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)





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 | 0.008500 | 0.040000 | 0.111000 | 0.300500 |
| QC | 0.008500 | 0.041000 | 0.111000 | 0.301500 |

Case 3db:

For the case with a high SNR, both classifiers perform equally, with the same error rate, this is showing that these classifiers work well when the noise is small.

Case 0db:

When the SNR is 0db the error probabilities for both classifiers are higher, performing better than the LC classifier, this is showing that both classifiers are working well with more noise.

Case -3db:

For this case we can see that both classifiers have the same value, so the noise is affecting both classifiers equally.

Case -10db:

With this noise level the error rates are significant for both classifiers, being worse for QC, so we can see that the Quadratic classifier is slightly more sensitive to noise.

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 | [[897 103]  [119 881]] | [[698 302]  [299 701]] |
| QC | [[895 105]  [117 883]] | [[693 307]  [296 704]] |

Case -3db:

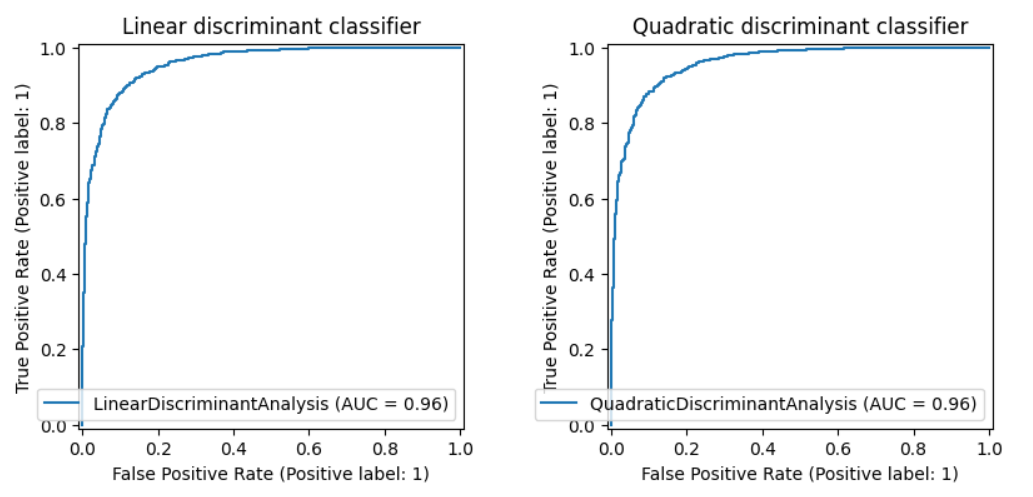
For this level of noise both classifiers perform similar (true positive rate and true negative rate are similar). For the number of false positives and false negatives LC have a slightly better performance.

Case -10db:

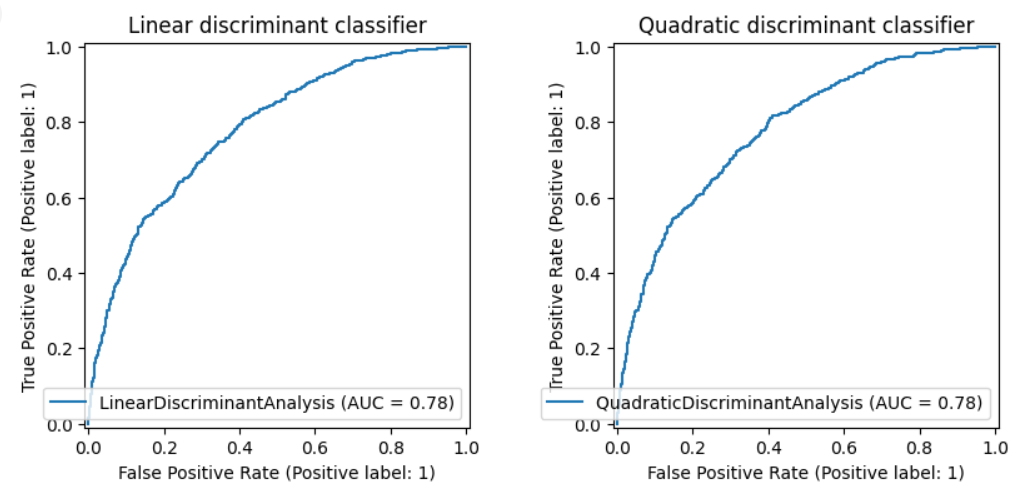
Both classifiers perform worst for this noise level, the performance for the true positive and false negative. LC have better performance compared to QC

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.

For SNR = -3



For SNR = -10



At these ROC curves, we can see that both classifiers perform almost equally at both noise levels (both curves are almost equal).

We can appreciate that at higher noises the classifiers have more difficulty to distinguish signal from noise.

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.

Case SNR = 3

26.886829216197846

26.72582894831203

Case SNR = 0

14.934879489896801

14.948231667445825

Case SNR = -3

8.94272634963422

9.044634863298443

Case SNR = -10

4.11967610623791

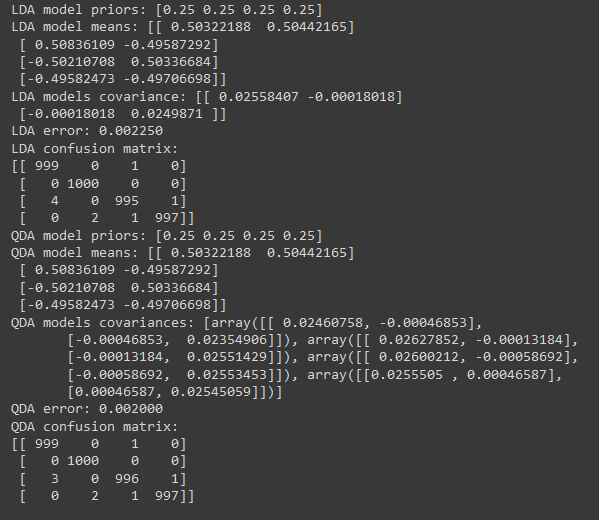
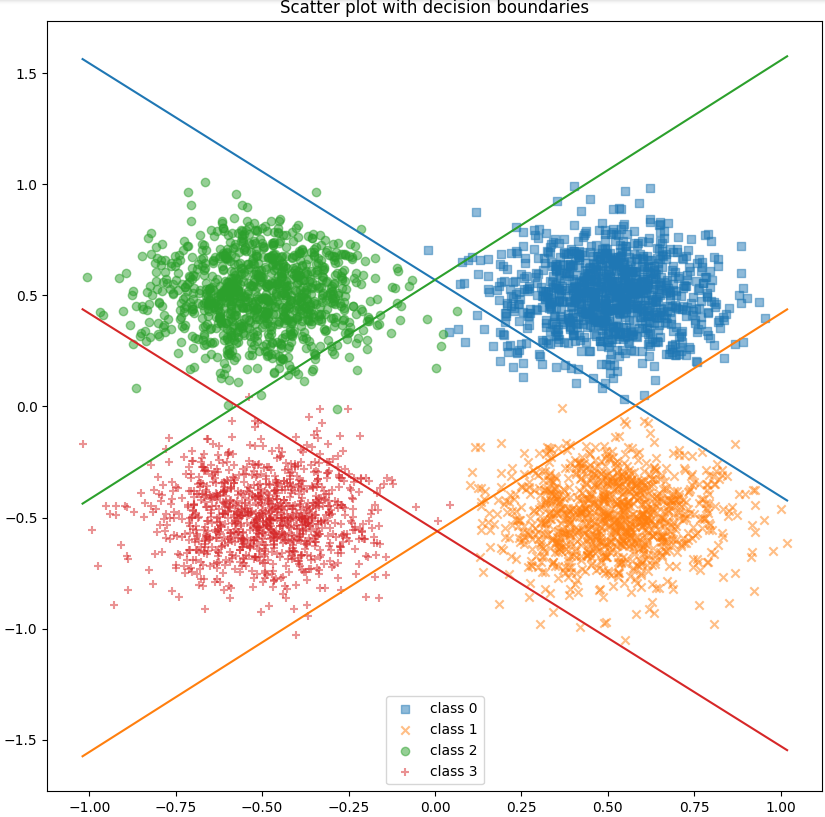
4.294965958405668

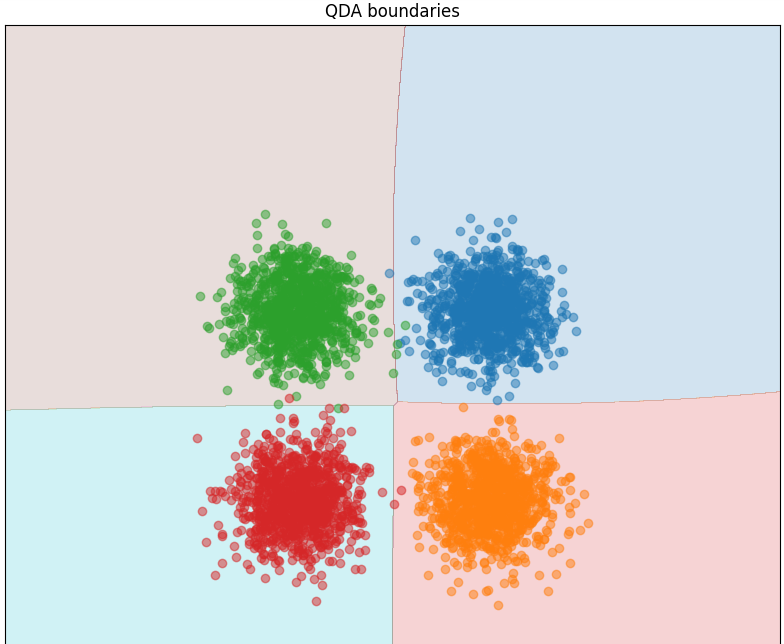
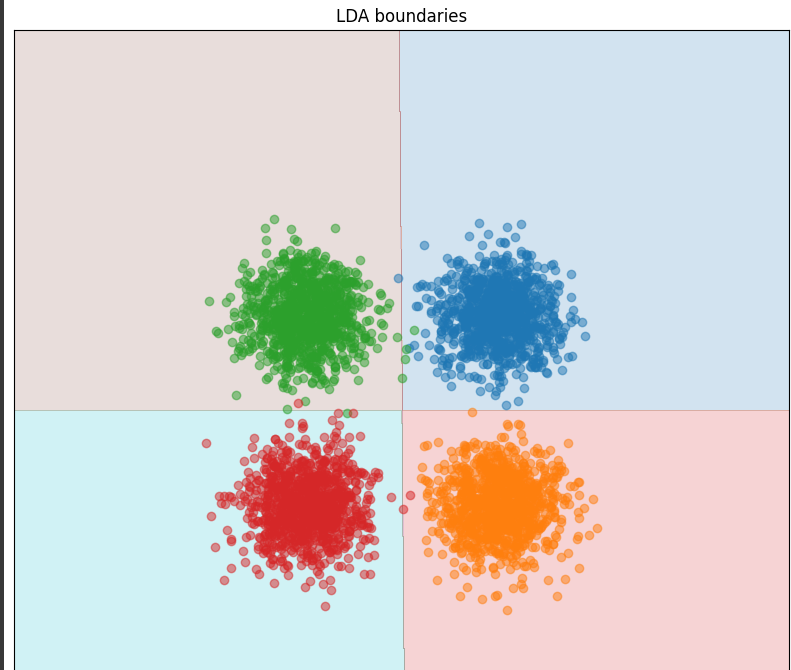
Here we can see that for lower SNR (more noise), the Mahalanobis distance is lower, reflecting the increase of difficulty to separate the classes.

The Mahalanobis distance is symmetric in its definition, meaning that switching the order of the two classes in the computation should not drastically change the result. However, slight numerical differences may arise due to rounding errors, precision in covariance matrix estimation, or small differences in the data. This is why you see slightly different values for the two distances in each case, but the differences are generally small.

**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.

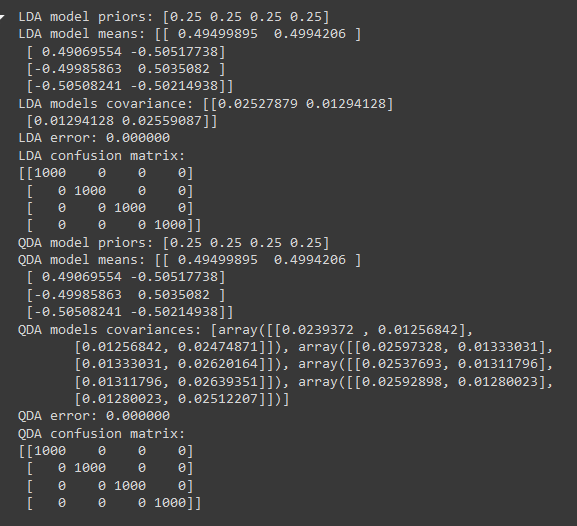
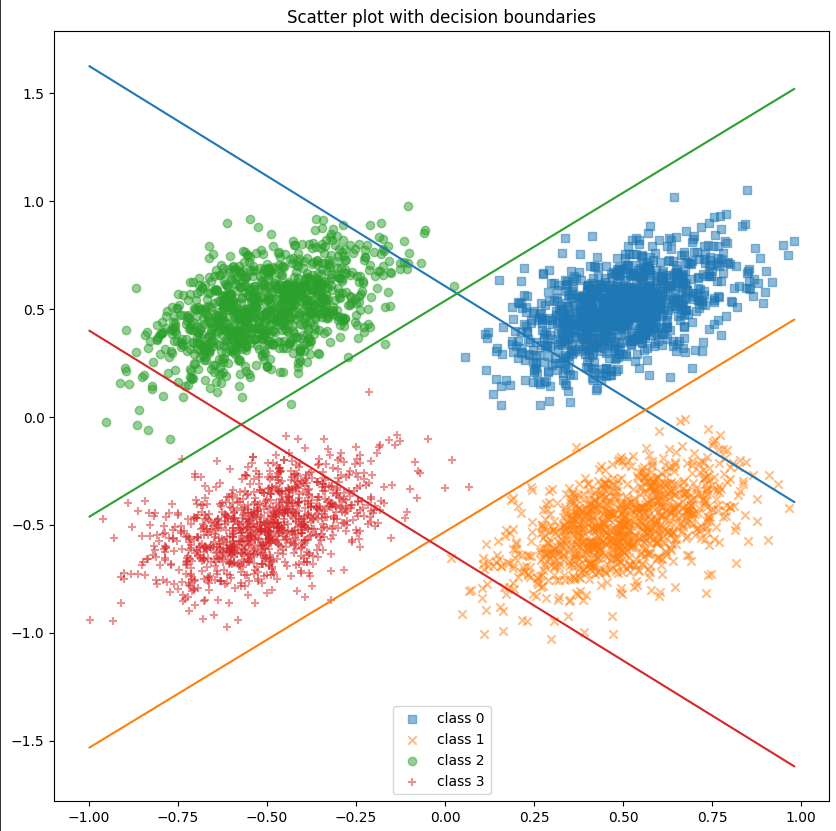


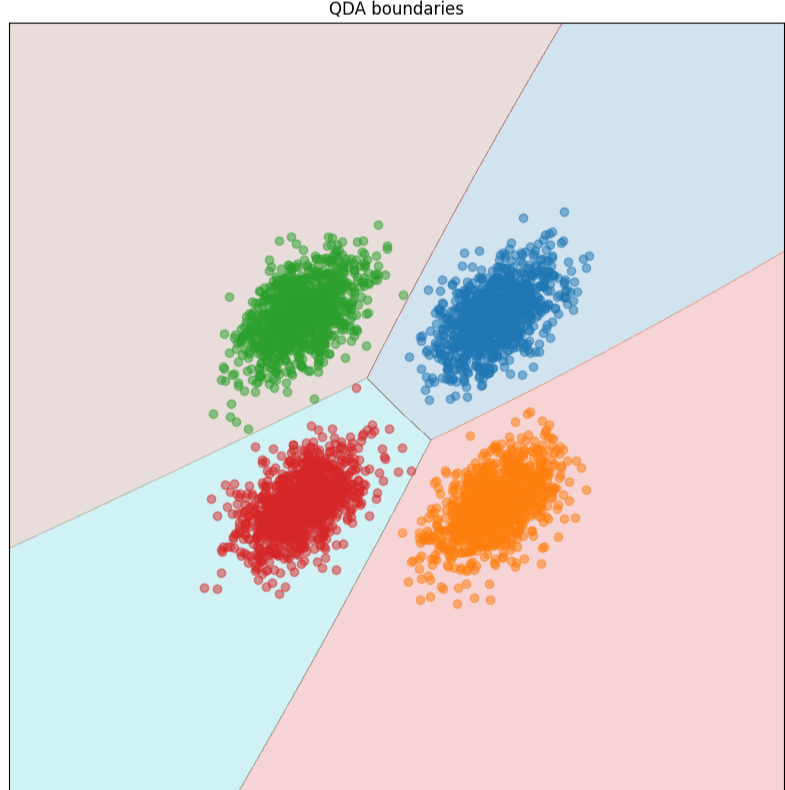
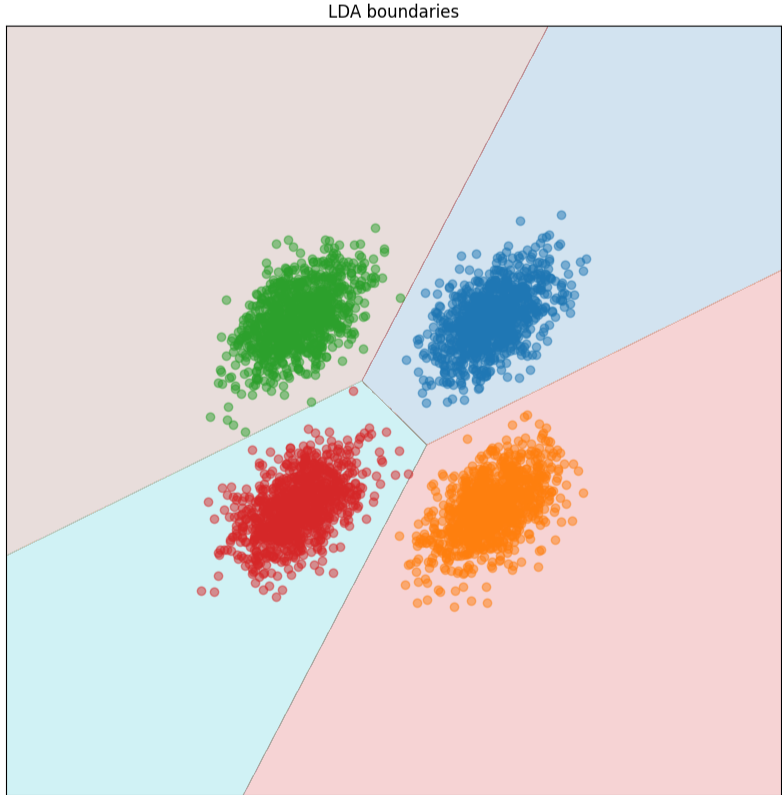


LDA performs well when the data is linearly separable, in this case QDA performs better (quadratic decision boundaries)

The scatter plot confirms thi, overall QDA performs better when the data doesn't have any linear relationship between classes, that can be seen here (almost perfect classification and low error)

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





For this case both classifiers perform well. For LDA were sufficient to separate the classes, and the flexibility of QDA's quadratic boundaries did not offer any significant advantage.

This suggests that while QDA generally has more expressive power, the linear separability of the data under these conditions allows LDA to perform just as well in this particular case.

Q8. Compare and discuss the results obtained in Q6and Q8

In both cases, the classifiers performed extremely well. The introduction of correlation (ρ=0.5\rho = 0.5ρ=0.5) in Q7 made the classification problem slightly more complex, but both LDA and QDA adapted and performed perfectly. **QDA**'s quadratic boundaries may offer more flexibility in more complex scenarios, but in these particular cases, **LDA**'s linear boundaries were just as effective.

The main takeaway is that while **QDA** can handle more complex data structures due to its ability to model quadratic boundaries, **LDA** can still perform well when the data is sufficiently separable, even under feature correlation.

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

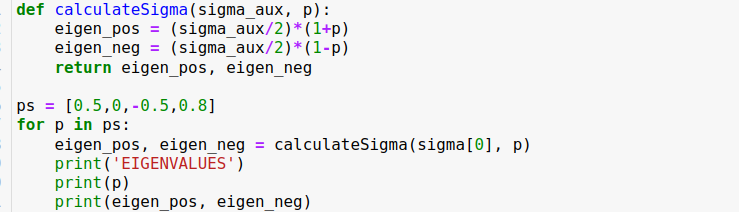
Q9. Include the error probabilities obtained using the linear classifier (LC) and the quadratic classifier (QC) for SNR = +5 dB and +10 dB. Compare the metrics for the two classifiers and discuss the results.

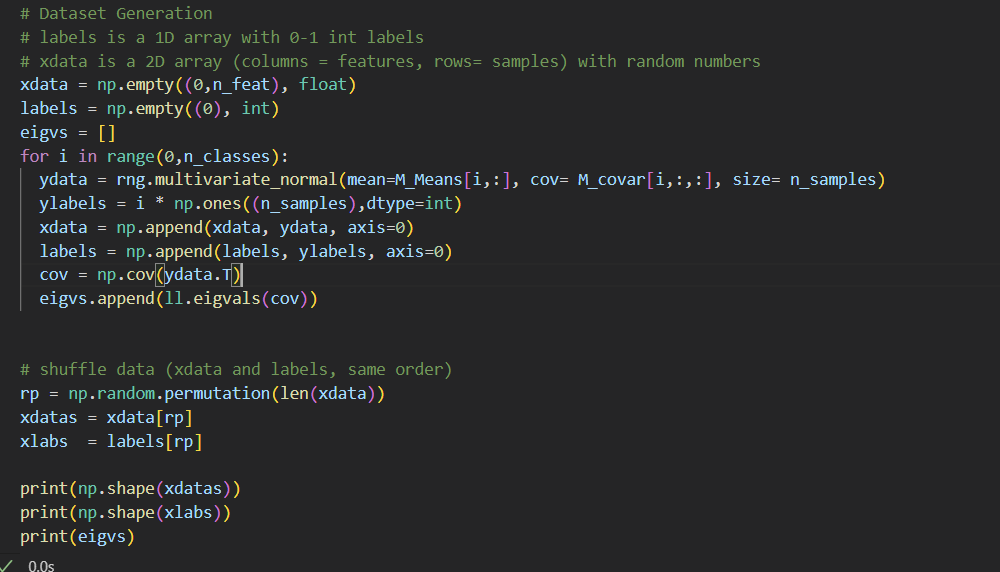
|  | 5 dB | 10 dB |
| --- | --- | --- |
| LC | 0.062750 | 0.001500 |
| QC | 0.059000 | 0.000500 |

We can see a major difference between the quadratic and the linear, the covariance matrices are all different so the hyper-quadratic decision boundaries are a better fit for our model.

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)

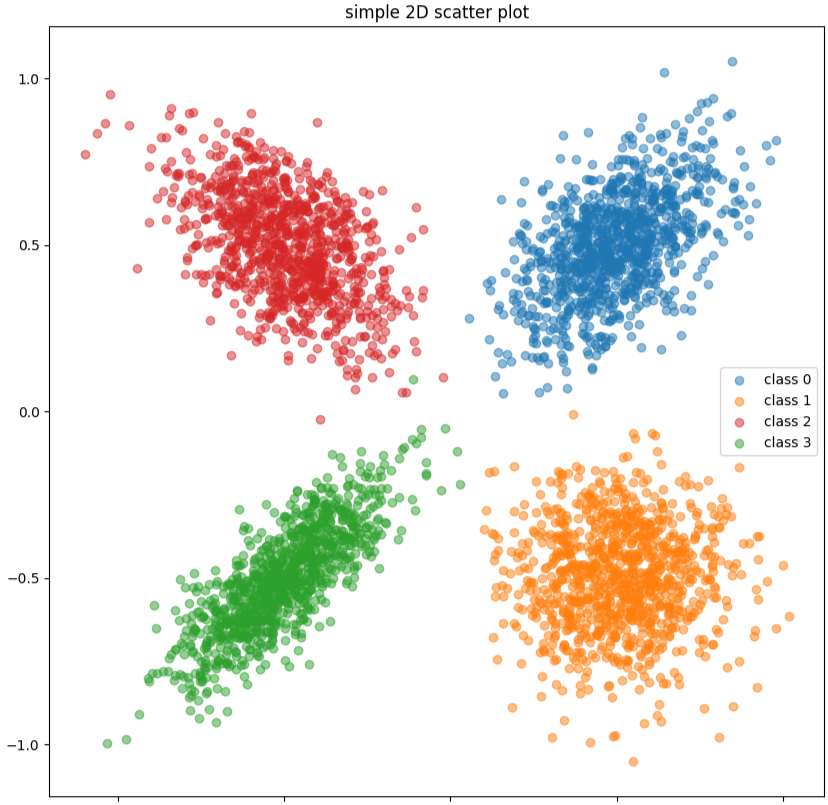
| SNR=?? | Class 1 (0.5) | Class 2 (0) | Class 3 (0.5) | Class 4 (0.8) |
| --- | --- | --- | --- | --- |
| Theoretical eigenvalues | 0.11858541225631422 0.03952847075210474 | 0.07905694150420949 0.07905694150420949 | 0.03952847075210474 0.11858541225631422 | 0.14230249470757708 0.015811388300841892 |
| Eigenvalues from sample covariance matrices | 0.12316786, 0.04012092 | 0.08756795, 0.0760404 | 0.03804094, 0.11019306 | 0.12880184, 0.01609249 |



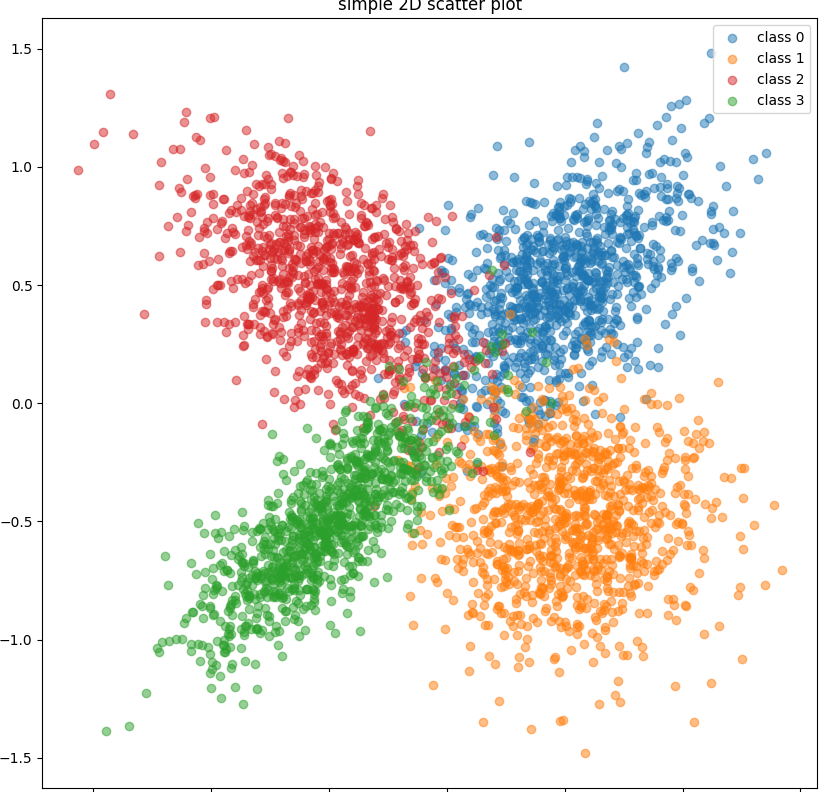


Q11. Include scatter plots for the linear and quadratic classifiers using SNR= +5 dB and SNR= +10 dB. Relate the shape of the clusters with the eigenvalues of the covariance matrices.

SNR = 10



SNR = 5



**Eigenvalues** (λ\lambdaλ): These values indicate the amount of variance that can be explained by each principal component (direction in which the data varies).

The orange class or class 1 has a p=0 (the one with the spherical shape) where the eigenvalues are the same.

For class 0 or the blue class (p=0.5) or class 2 (red, p=-0.5) the eigenvalues are the same but transposed, and the same happens in the graph. That’s why they look mirrored.

Finally, for class 3 or the green class, the difference between the eigenvalues is the greatest, which is why the ellipse looks elongated

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].

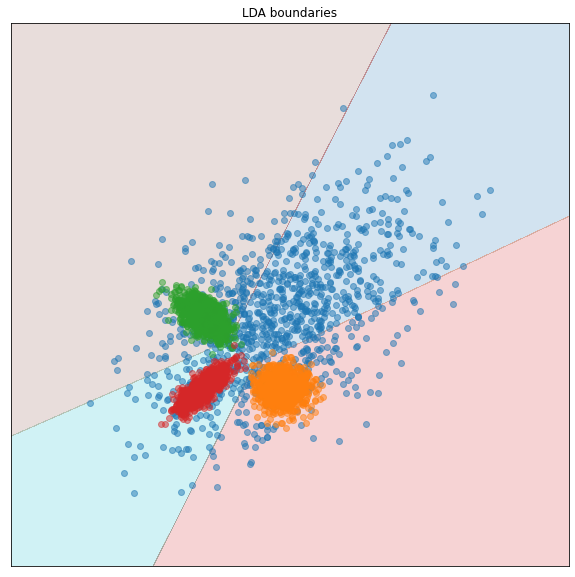
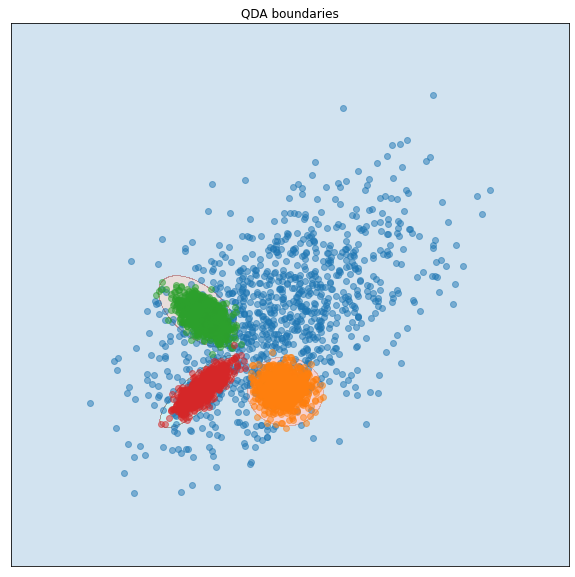
For SNR = 10 dB, multiply the covariance matrix of class 1 by a large number (for example sigma(0)\*=30). Compute the classification error, scatter plots and boundaries for the linear and the quadratic classifiers. Observe that in this case the quadratic discriminant outperforms the linear one.

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].

**SIGMA:**

[1.5 0.05 0.05 0.05]

|  | LDA | QDA |
| --- | --- | --- |
| 10db | 0.127000 | 0.048750 |



As we can see the QDA’s decision boundaries are hyper-quadratic instead of only hyper-planes, that is why in the QDA we have hyper-spheres for class 1,2 and 3 and for LDA we have hyper-planes and the QDA outperforms LDA. This phenomenon is produced by having a 30 times the original variance (sigma[0] = sigma[0] \* 30)