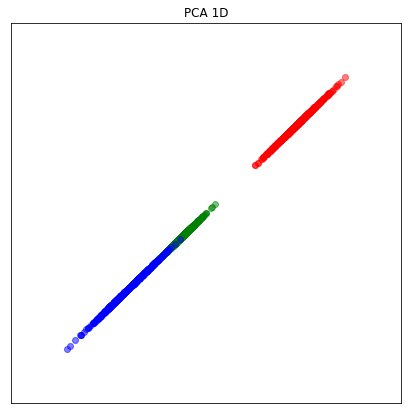
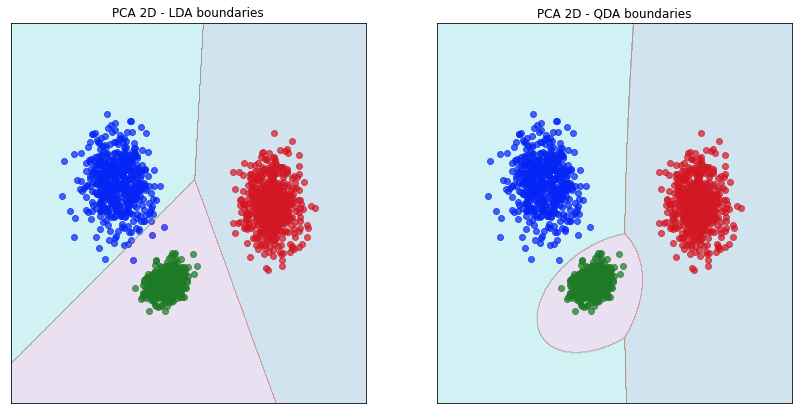
**MACHINE LEARNING FROM DATA**

**Report: Lab Session 2 – Feature selection – PCA and MDA**

**Questions**

**Q1**: Complete the table with the training and test errors for the linear (LC) and the quadratic (QC) classifiers when using three, two and one feature, and **SNR=10dB**. In this case **PCA** is used for feature selection. Discuss the results. Analyse the scatter plots in two dimensions and in one dimension.

|  | 3 features | | 2 features | | 1 feature | |
| --- | --- | --- | --- | --- | --- | --- |
| Test | Train | Test | Train | Test | Train |
| LC | 0.000667 | 0.0 | 0.002667 | 0.002000 | 0.039333 | 0.024667 |
| QC | 0.0 | 0.0 | 0.000667 | 0.0 | 0.032000 | 0.022000 |



It does not make sense to reduce to 3 features, as it is already the amount of features we have at the start.

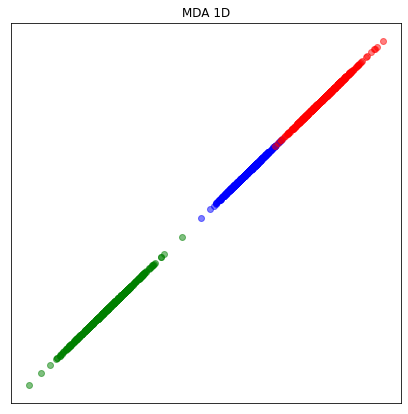
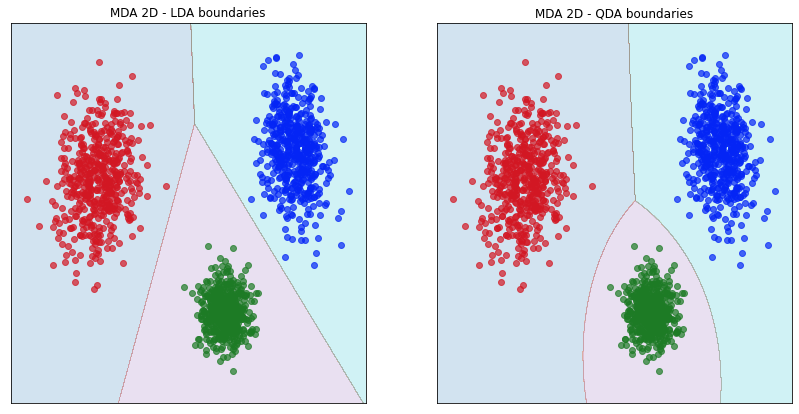
With the current noise, which is quite low, we can see that the data is quite concentrated or densely clustered.

When reducing dimensionality to 2 features with PCA we see that both classifiers, LC and QC do a very good job classifying the dataset. The QC is a bit better than the LC.

Finally we see that even when reducing to 1 feature there is still a good class separability because of the low noise. Consequently, the error is still very low. The QC slightly outperformed the LC

**Q2**: Complete the table with the training and test errors for the linear (LC) and the quadratic (QC) classifiers when using three, two and one feature, and **SNR=10dB**. In this case **MDA** is used for feature selection. Discuss the results. Analyse the scatter plots in two dimensions and in one dimension.

|  | 3 features | | 2 features | | 1 feature | |
| --- | --- | --- | --- | --- | --- | --- |
| Test | Train | Test | Train | Test | Train |
| LC | 0.000667 | 0.0 | 0.000667 | 0.0 | 0.011333 | 0.004000 |
| QC | 0.0 | 0.0 | 0.0 | 0.0 | 0.006667 | 0.003333 |



Again, it does not make sense to reduce to 3 features, as it is already the amount of features we have at the start. The errors for both classifiers are the same when using PCA and MDA, because we are not reducing the features.

With the current noise, which is quite low, we can see that the data is quite concentrated or densely clustered.

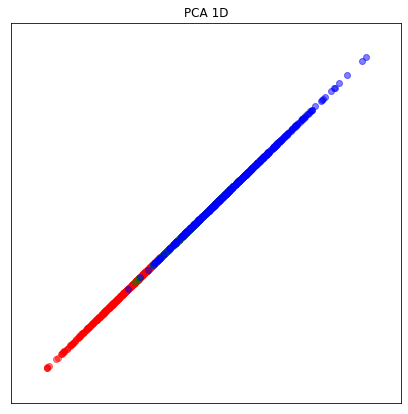
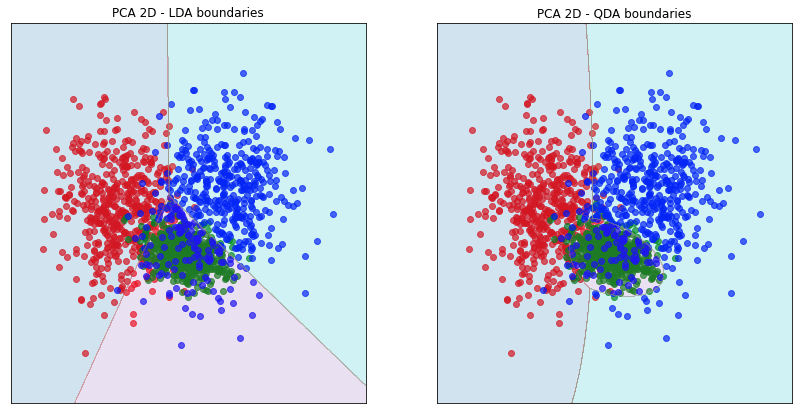
When reducing dimensionality to 2 features with MDA we see that both classifiers, LC and QC do a very good job classifying the dataset. The QC is a bit better than the LC. IN fact, QC es a perfect classifier in this case.

Finally we see that even when reducing to 1 feature there is still a good class separability because of the low noise. Consequently, the error is still very low. The QC outperformed the LC.

When comparing MDA and PCA, we see that in all cases, when using MDA, the projections to 2 and 1 dimensions result in a better separability (which implies better classifiers).

**Q3**: Use **PCA** for feature selection. Complete the table with the training and test errors for the linear (LC) and the quadratic (QC) classifiers when using three, two and one feature, and **SNR= 0 dB**. Discuss the results. Analyse the scatter plots in two dimensions and in one dimension.

|  | 3 features | | 2 features | | 1 feature | |
| --- | --- | --- | --- | --- | --- | --- |
| Test | Train | Test | Train | Test | Train |
| LC | 0.133333 | 0.115333 | 0.175333 | 0.147333 | 0.244667 | 0.250667 |
| QC | 0.062000 | 0.070667 | 0.144000 | 0.122000 | 0.239333 | 0.244667 |



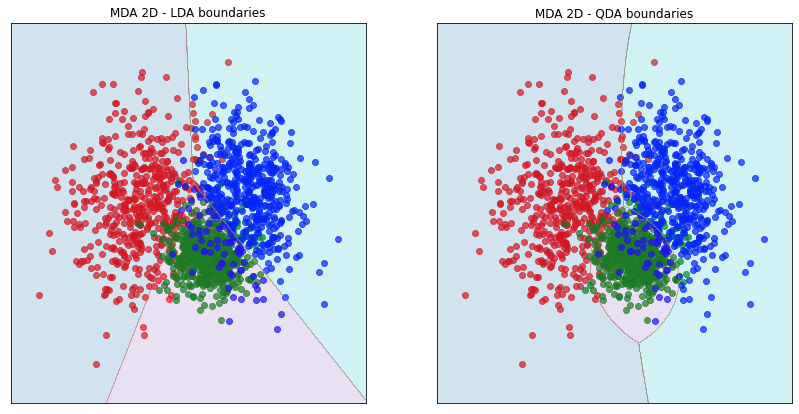
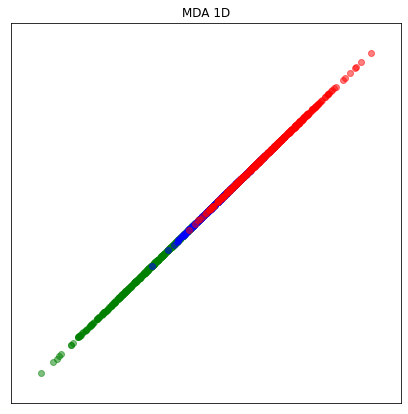
With this amount of noise, we know that the classes will be highly mixed. With 3 features the LC is not very good as it has a 13% error, the QC is a bit better with a 6% error.

When projecting to 2 features or dimensions the high noise causes a very low separability between classes. In this case, the two classifiers are not good with an error of 17% and 14%, with the QC slightly outperforming the LC.

Finally when projecting to 1 feature we can see a big overlap between classes, resulting in both classifiers performing really bad.

**Q4**: Use **MDA** for feature selection. Complete the table with the training and test errors for the linear (LC) and the quadratic (QC) classifiers when using three, two and one feature, and **SNR= 0 dB**. Discuss the results. Analyse the scatter plots in two dimensions and in one dimension.

|  | 3 features | | 2 features | | 1 feature | |
| --- | --- | --- | --- | --- | --- | --- |
| Test | Train | Test | Train | Test | Train |
| LC | 0.133333 | 0.115333 | 0.110667 | 0.115333 | 0.220667 | 0.211333 |
| QC | 0.062000 | 0.070667 | 0.125333 | 0.133333 | 0.215333 | 0.208000 |

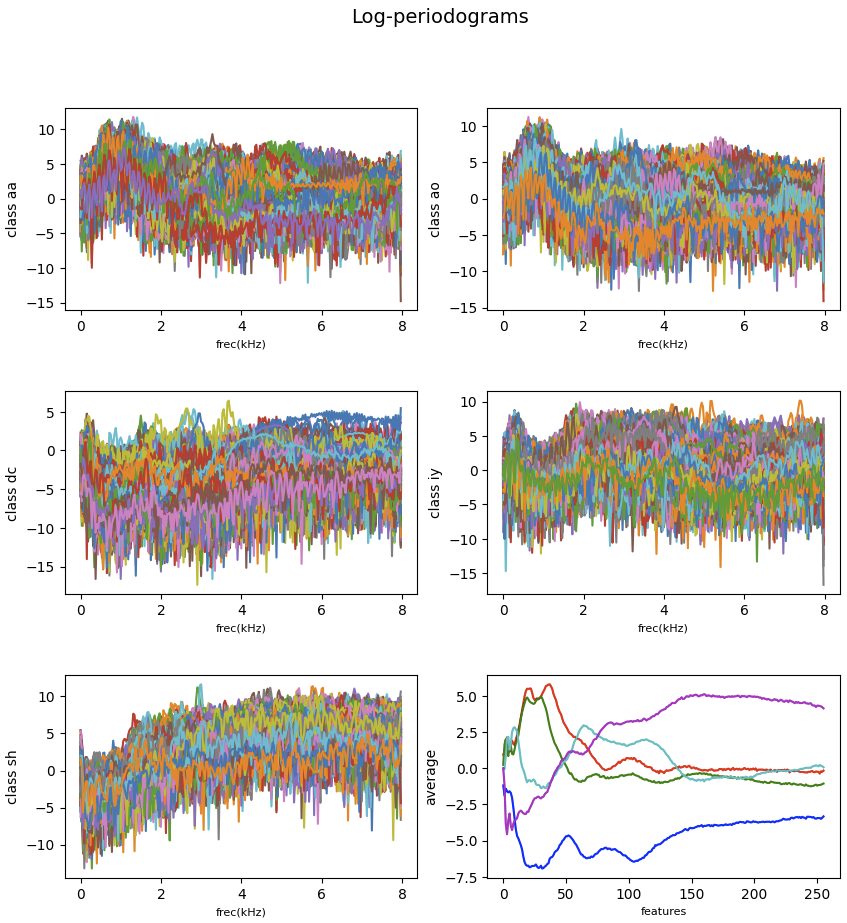
With this amount of noise, we know that the classes will be highly mixed. With 3 features the LC is not very good as it has a 13% error, the QC is a bit better with a 6% error.

When projecting to 2 features or dimensions the high noise causes a very low separability between classes. In this case, the two classifiers are not good with an error of 17% and 14%, with the QC slightly outperforming the LC.

Finally when projecting to 1 feature we can see a big overlap between classes, resulting in both classifiers performing really bad.

When comparing MDA and PCA, we see that in all cases, when using MDA, the projections to 2 and 1 dimensions result in a better separability (which implies better classifiers).

**Q5**. Include the plots of the phoneme spectra.



**Q6**. Include the error probabilities for the training and test sets obtained with the linear classifier (LC) and the quadratic classifier (QC), using all the features. Discuss the results.

**LDA (LC) test error**: 0.079084

**QDA (QC) test error**: 0.112343

We can see that in this case the linear classifier behaves better than the quadratic. With this configuration we see that classes are not very separable, as both classifiers have errors of 8% and 11%.

**Q7**. Include the confusion matrices for the test set obtained with the linear classifier (LC) and the quadratic classifier (QC), using all the features. Discuss the results.

**LDA (LC) test confusion matrix**:

[[157 51 0 0 0]

[ 47 260 0 0 0]

[ 0 0 221 4 2]

[ 0 0 1 348 0]

[ 1 0 0 1 260]]

**QDA (QC) test confusion matrix**:

[[142 65 0 0 1]

[ 67 237 1 0 2]

[ 0 0 222 4 1]

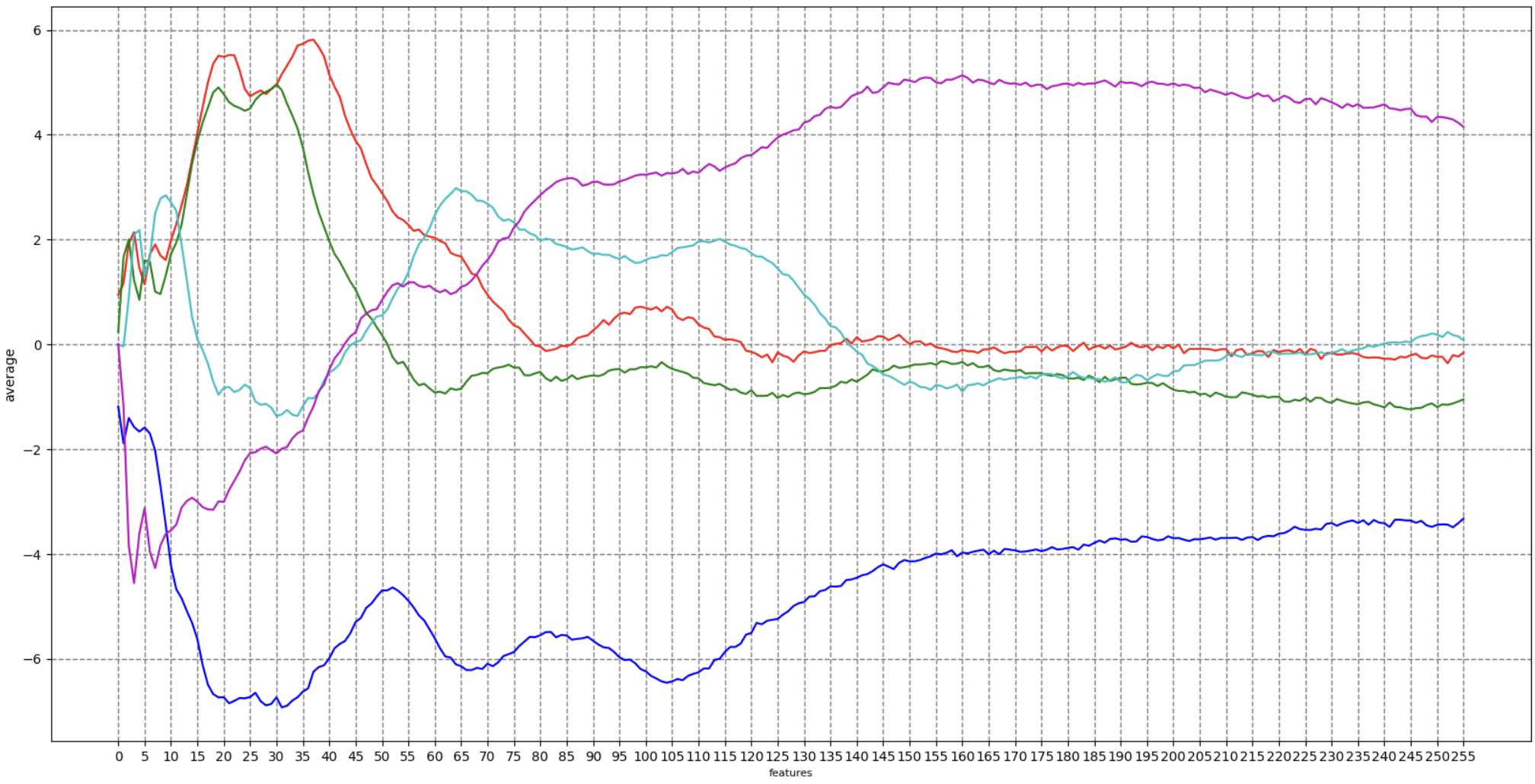
[ 0 0 9 338 2]

[ 0 0 0 0 262]]

As we have seen in the previous questions, the LC performs a bit better than the QC. This difference is found in the classes 1 and 4, where clearly the LC classifies better.

Suppose that due to computational issues we can only use two features per observation. Considering the plots in Q1, manually select two features that seem to be the most discriminative. Use the variable V\_coor

**Q8**. Which features would you choose? Show the error probabilities for the training and test sets obtained with the linear and the quadratic classifier. Compare with the previous case (using all features) and discuss the results.



By intuition we would choose the spaces where the means in the plot above are the most separated between classes. For example we can choose the features **20** and **100**.

For this pair of features the results are the following:

**LDA train error**: 0.312738

**LDA test error**: 0.309682

**QDA train error:** 0.308302

**QDA test error**: 0.305987

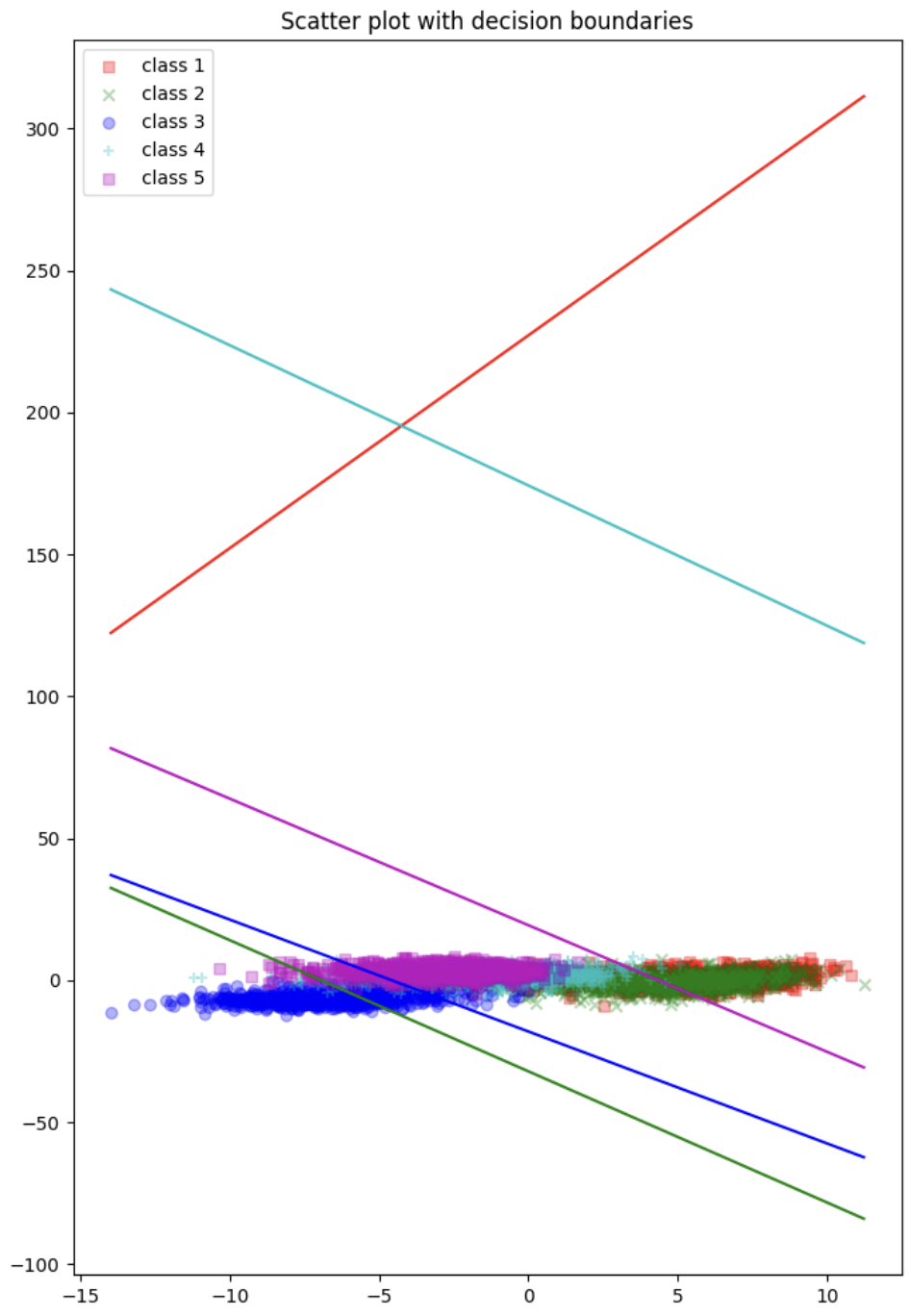
We can see that when choosing two features the classifiers work much worse than when using the 256, that is explained by the loss of information by simply deleting 254 features. In consequence the error probabilities rise from around 10% to more than 30%. However I would have expected a much bigger error based on the amount of information lost when choosing only 2 features.

In most of the pairs tested we cannot see a very big difference between the LDA and the QDA classifiers, so in this case we cannot assume that one is better than the other.

It is important to note that this combination (features: 20 and 100) is not the best pair of features to use in order to classify. We have created an iterative solution so that we get the combination with the minimum error for both classifiers and the optimal solution has been the pair 3, 23 and the errors for both classifiers are 21%.

When checking the means for these two features in the plot above we see that at least for the feature 3, the means are not clearly separated. With that in mind, we think that this criterion (choose based on separability of means) might not be the best one.

**Q9**. Include the scatter plot and decision boundaries obtained. Discuss the results.



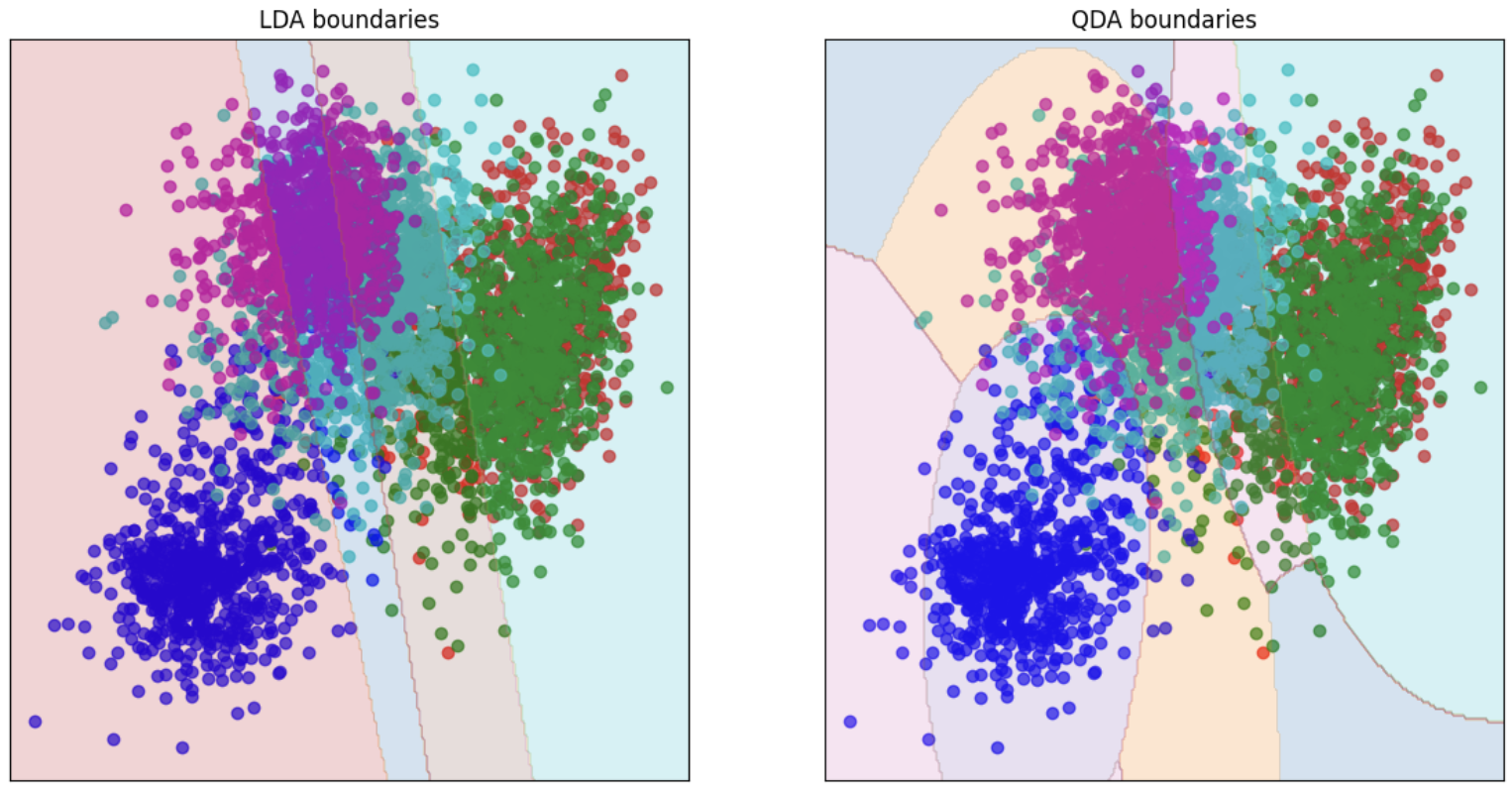
We are working with the pair of features 20 and 100.

The plot shows that the feature chosen are somehow related, which is bad in order to create a good classifier later on.

In the scatter plot it is clear that the separability of the classes is almost non-existent. Particularly, for classes 1 (red) and 4 (cyan) the decision boundaries appear to be way off target, resulting in a lot of misclassifications of these two classes.

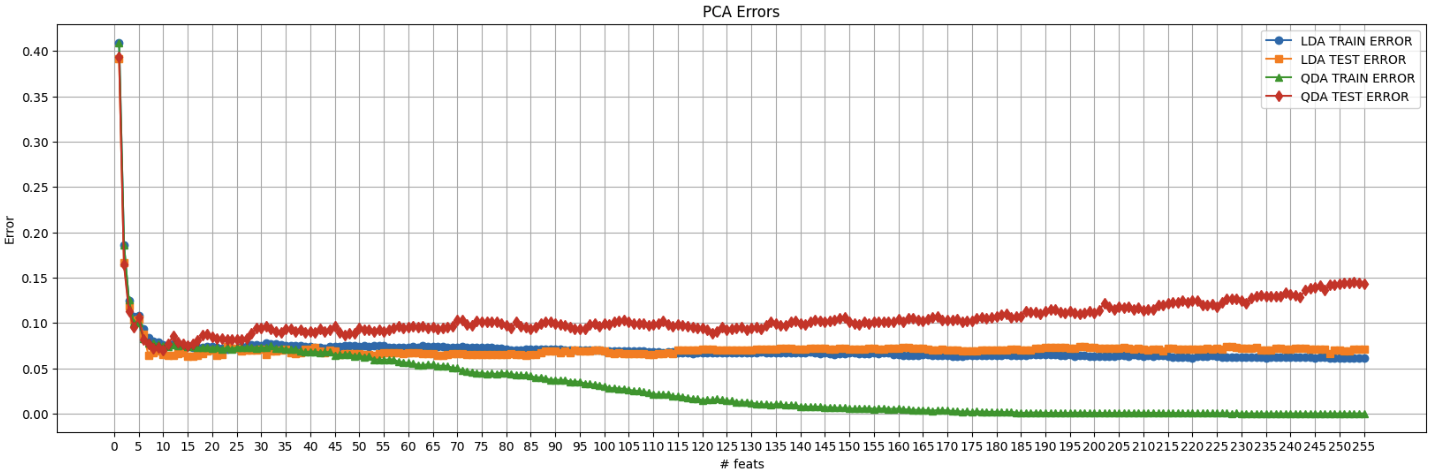
The classes which might be a bit better classified in this case would be 2(green), 3 (blue) and 5 (rose), because the decision boundaries appear to function and separate a bit better.

From this plot we can take the same conclusion as before. With only 2 features, we lose too much information, and the classification with these classifiers is very hard, resulting in a high error rate.



Regarding the boundaries, we can clearly see that this solution misclassifies plenty of elements, and is clearly underfitting the original set of data, which is caused by the loss of information when choosing only 2 features.

**Q10**. Using **PCA**, show the error curves for the linear and the quadratic classifier on training and test set.



**Q11**. Discuss which dimension is the most adequate for the linear classifier and which is the best one for the quadratic classifier. Remember that it is important not to overfit on the training data (the test error should not be much larger than the training error).

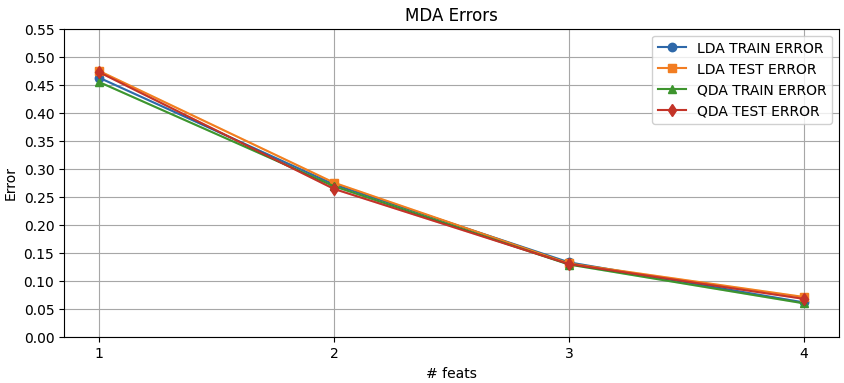
For the **quadratic classifier** we can see that when using a bigger number of features it start to overfit the data (or memorise the training set) and ends up not being able to generalise correctly as the error for the test set is much greater than the error for the training data. In comparison, for the **linear classifier** we see that after the same range of 5 to 20, it does not improve a lot, bit it still performs quite good.

With that in mind, we see that the ideal amount of features used to train the classifiers should be a number from 5 to 20, as in this range, the error for training and for test are very similar, and both classifiers seem to generalise well and not overfit or memorise the training data.

The conclusion we take from this plot is that huge amounts of features is not usually the answer to train and get good classifiers.

**Q12**. Using **MDA**, which is the maximum number of features *dmax*? Show the error curves for the linear and the quadratic classifier on the training and on the test set.

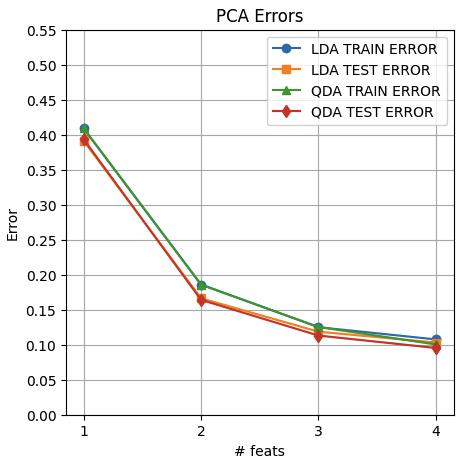
The amount of features to reduce when using MDA can be at most **min(n\_features, n\_classes - 1)**. With that in mind, the value for dmax is n\_classes-1 is 5-1 = 4.



**Q13**. Compare results and discuss the use of PCA and MDA for the Phoneme dataset using between 1 and *dmax* features.

We can see that in the **MDA** classifiers, the more features it has, the better the classifiers can work. In the case for **PCA**, the same applied but after 10 features, it just stops improving and the classifiers do not improve, in fact, the **QDA** gets worse with more features because it starts ovefittin the training data.

There isn’t a clear difference between the two types of classifiers, as they behave mostly the same in most cases. Except when doing reduction with **PCA** and more than 20 features, then while the **Linear classifier** still performs quite well, the **Quadratic Classifiers** starts performing much worse, as it memorises the training data.



When comparing directly on the range from 1 to 4 features we see that at first when reducing with PCA the classifiers work better for 1 or two features.

At 3 features the classifiers behave very similarly and have very similar error rates.

Finally at 4 features, the classifiers work much better when reducing with MDA, as the error rate is lower.