ML Labb 2

Magnus Axén
Oscar de Brito Lingman

March 7, 2022

Assignment 1

In the first assignment we are asked to write functions that compute the mean and variance matrix for a multi-variate Gaussian distribution. We are given labeled data, 0,1,2,3,4, and for each of the labeled data the mean and variance is computed. We return in our mean function a $C \times d$ matrix where C is the amount of classes, and d the chunks of data sets. In the variance we return a $C \times d \times d$ tensor to compute for each class and data chunks the co-variance between the labeled data. Seen below we plot the boundary's for the Maximum likelihood estimate of mu and sigma.

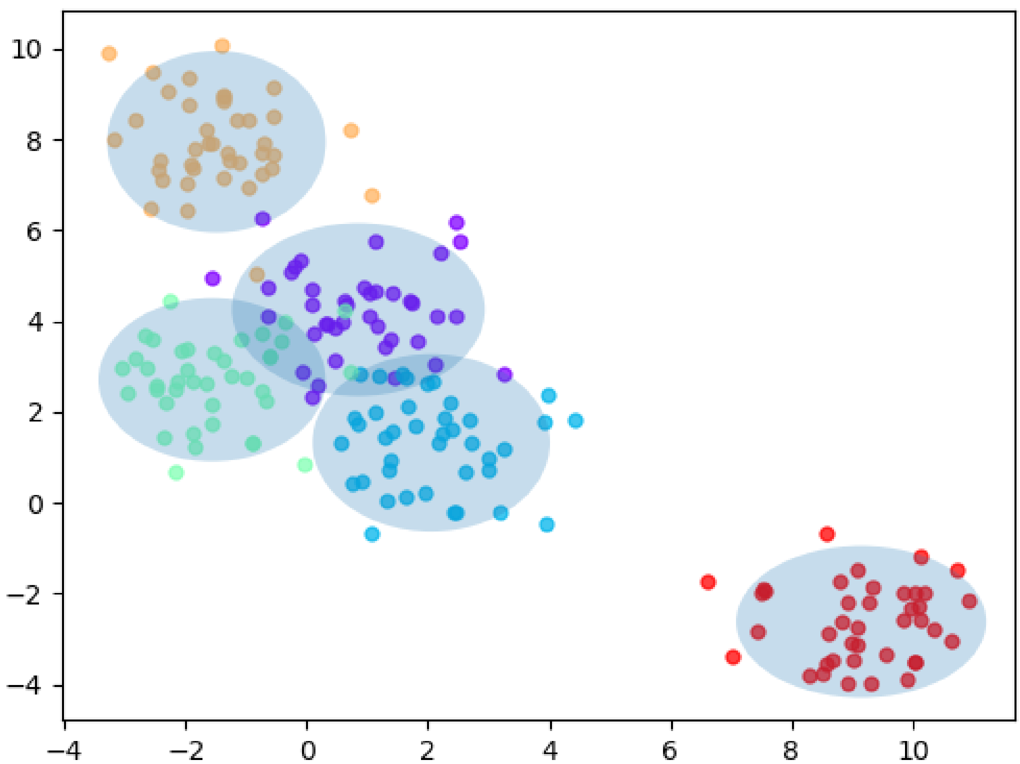

Figure 1: Gaussian Plot

Assignment 2

1)

ComputePrior is simply just implementing

$$p(k) = \frac{N_k}{N}$$

i.e compute the frequency of the class divided by total amount. That is how common the class is in percentage.

2)

In classifyBayes we implement equation 11, that is the the discriminant function. Here we use the co variance and mean to compute for each class and each point the discriminant function. Afterwards the argmax() of all the points are taken to assign the point to the class for which it is argmax. So we return a vector with length N i.e amount of data points with all the labels.

Assignment 3

Trials	Accuracy
0	84.4
10	95.6
20	93.3
30	86.7
40	88.9
50	91.1
60	86.7
70	91.1
80	86.7
90	91.1

Table 1: **Iris** dataset, bayes classifier
Final mean classification accuracy 89 with standard deviation 4.16

Trials	Accuracy
0	61
10	66.2
20	74
30	66.9
40	59.7
50	64.3
60	66.9
70	63.6
80	62.3
90	70.8

Table 2: **Vowel** dataset, bayes classifier
Final mean classification accuracy 64.7 with standard deviation 4.03

1)

For the conditional Independence assumption to be true we would have to have either small dependence or none at all. If coefficients are too highlty correlated this will indeed result in too messy conclusions. Else wise a lot of information will go lost on the model.

2)

By manipulating the data we could possibly remove the overlap s.t. the method more easily identifies a more reasonable boundary. As for changing methods in the previous labs we were taught about SVM's with slack. It seems like a more suitable choice here, a linear one at that.

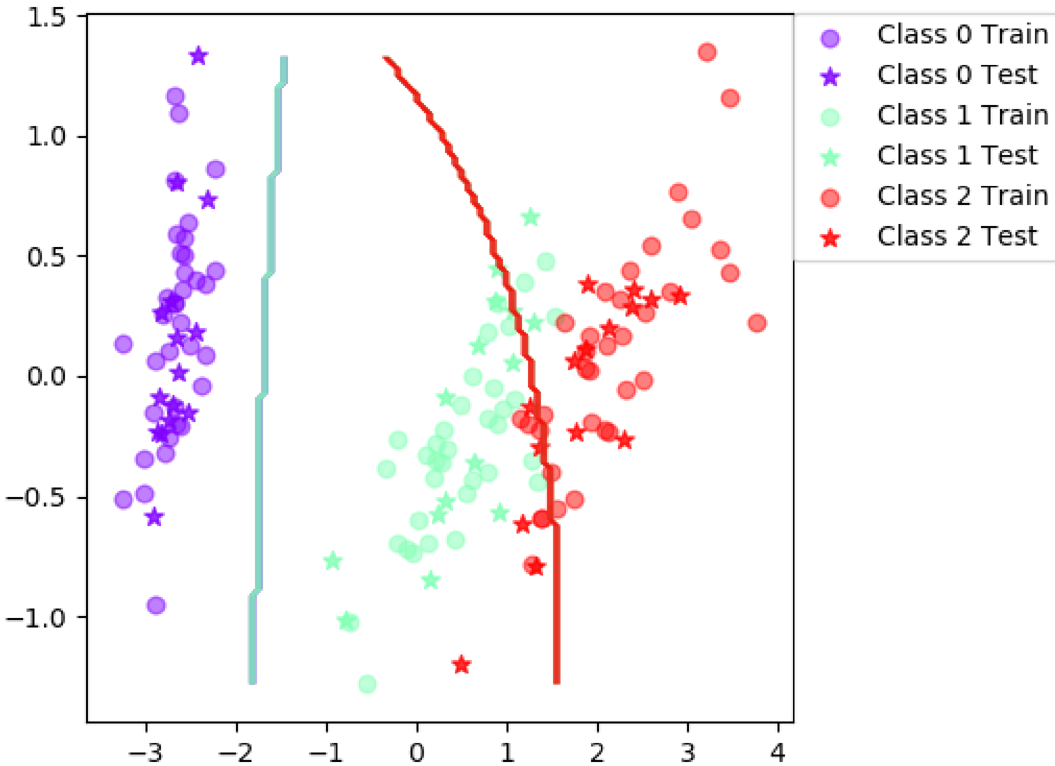

Figure 2: **Iris** dataset plot, bayes classifier

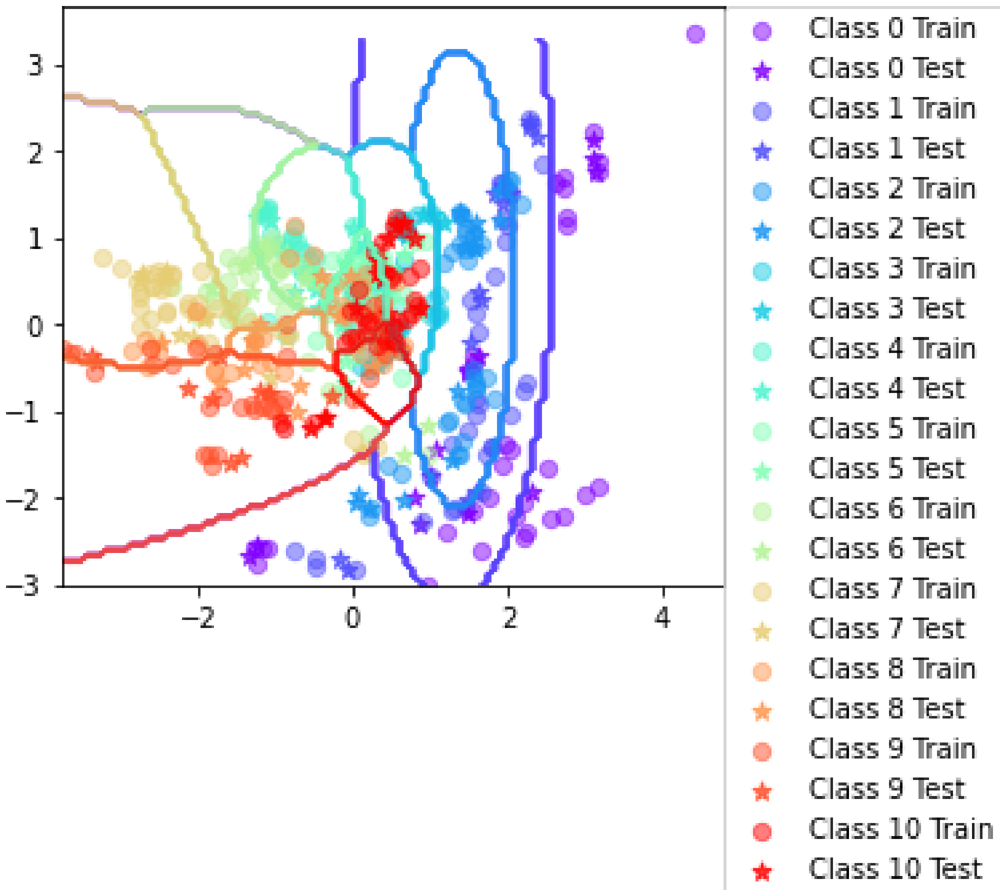

Figure 3: **Vowel** dataset plot, bayes classifier

Assignment 4

In assignment 4 the programming that was done was implementing the input W into the function sigma and mu. W is a vector of size $1 \times N$ that is simply just multiplied on all of the entries. Then we iteratively update the weights by looking at the error. The error in turn is computed by looking at the wrongly/correctly classified labeled data points. Its adjusted s.t. the error gets smaller and smaller.

Assignment 5

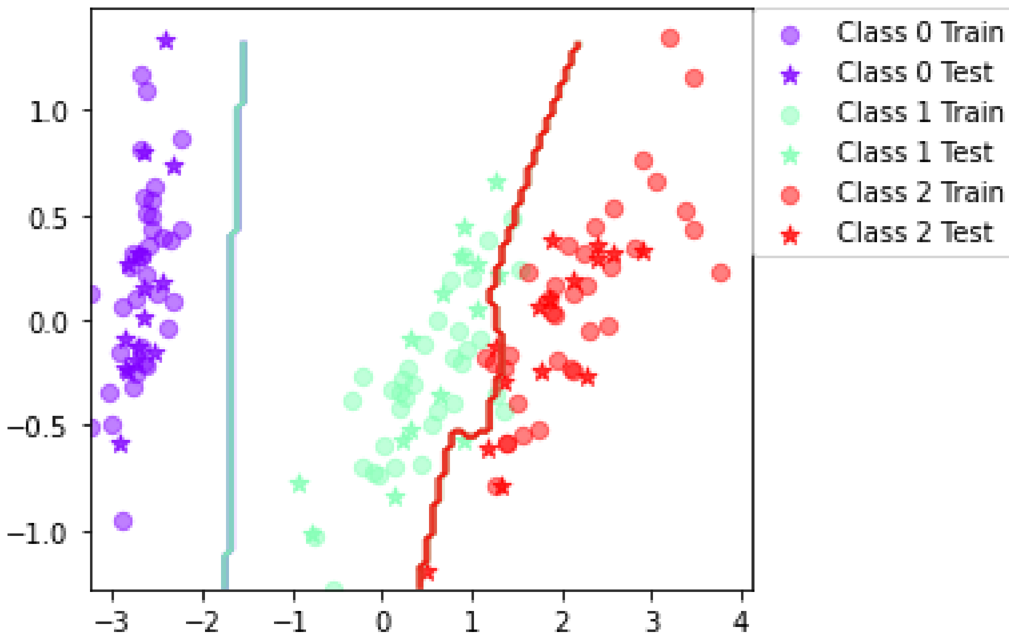

Figure 4: **Iris** dataset plot, boosted bayes classifier

Trials	Accuracy
0	95.6
10	100
20	93.3
30	91.1
40	97.8
50	93.3
60	93.3
70	97.8
80	95.6
90	93.3

Table 3: **Iris** dataset, boosted bayes classifier
Final mean classification accuracy 94.7 with standard deviation 2.82

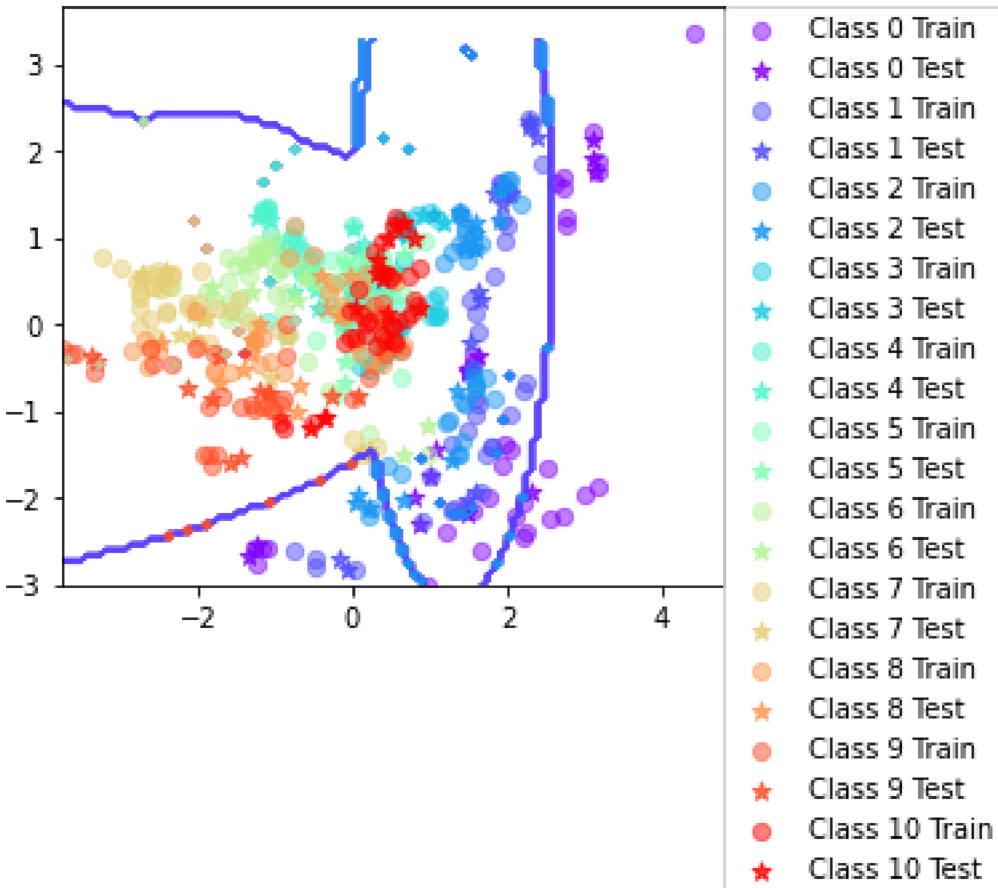

Figure 5: **Vowel** dataset plot, boosted bayes classifier

Trials	Accuracy
0	76.6
10	86.4
20	83.1
30	80.5
40	72.7
50	76
60	81.8
70	82.5
80	79.9
90	83.1

Table 4: **Vowel** dataset, boosted bayes classifier
Final mean classification accuracy 80.2 with standard deviation 3.52

(1)

	Iris	Vowel
Previous	89	64.7
Boosted	94.7	80.2
Difference	5.7	15.5

Table 5: Bayes classifier comparison between boosted and non-boosted

Yes there is. By boosting we are identify and adjusting the model by looking at the wrongly classified samples.

(2)

We can tell that instead of having a curved line we have now imposed a combination of linear lines which could be considered more complex. it follows the pattern clearly loads better than the earlier model.

(3)

Yes the boosting algorithm allows for a more complex model which can better fit the data. As seen in the plots.

Assignment 6

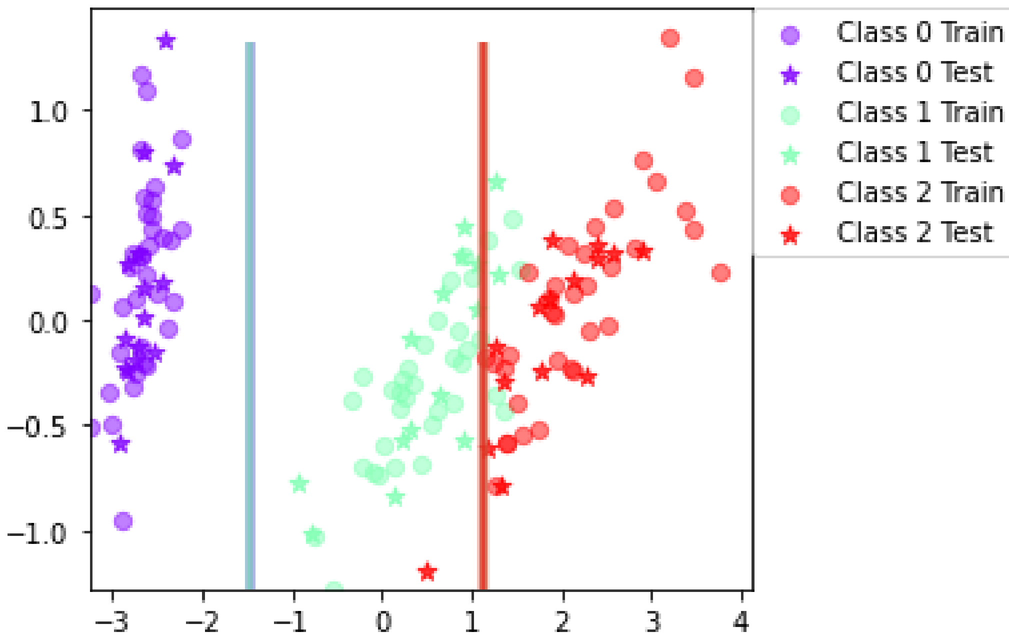

Figure 6: **Iris** dataset plot, decision tree classifier

Trials	Accuracy
0	95.6
10	100
20	91.1
30	91.1
40	93.3
50	91.1
60	88.9
70	88.9
80	93.3
90	88.9

Table 6: **Iris** dataset, decision tree classifier
Final mean classification accuracy 92.4 with standard deviation 3.71

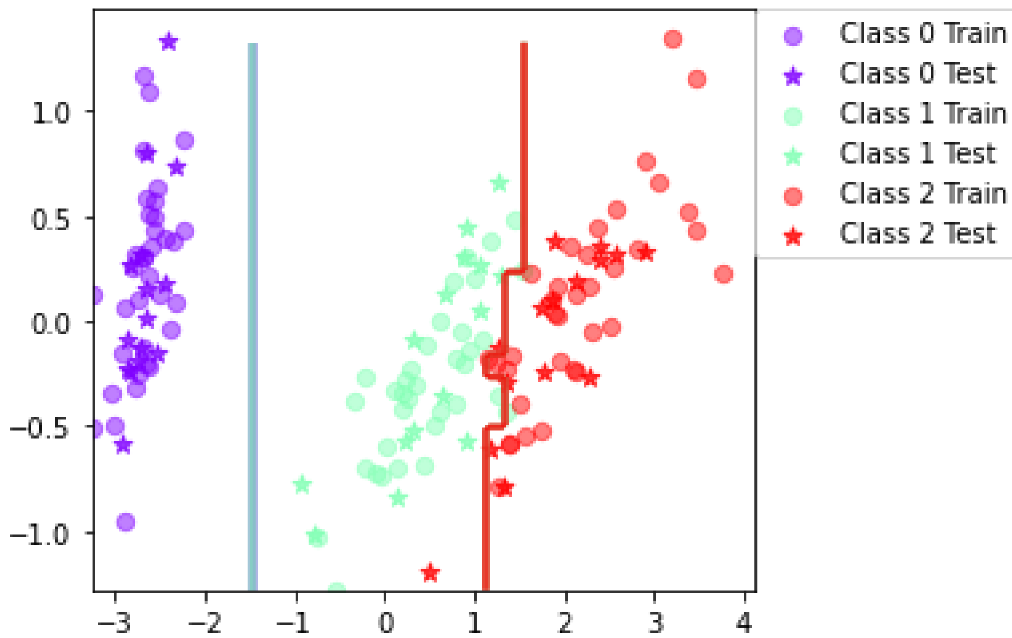

Figure 7: **Iris** dataset plot, boosted decision tree classifier

Trials	Accuracy
0	95.6
10	100
20	95.6
30	93.3
40	93.3
50	95.6
60	88.9
70	93.3
80	93.3
90	93.3

Table 7: **Iris** dataset, boosted decision tree classifier
Final mean classification accuracy 94.6 with standard deviation 3.65

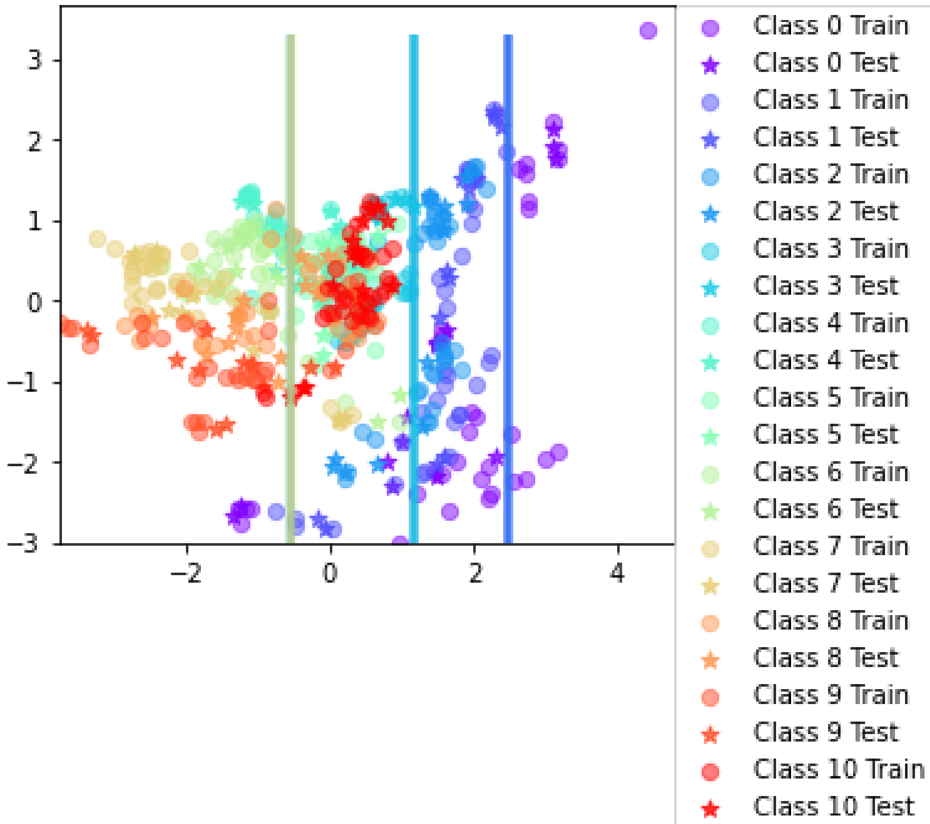

Figure 8: **Vowel** dataset plot, decision tree classifier

Trials	Accuracy
0	63.6
10	68.8
20	63.6
30	66.9
40	59.7
50	63
60	59.7
70	68.8
80	59.7
90	68.2

Table 8: **Vowel** dataset, decision tree classifier
Final mean classification accuracy 64.1 with standard deviation 4

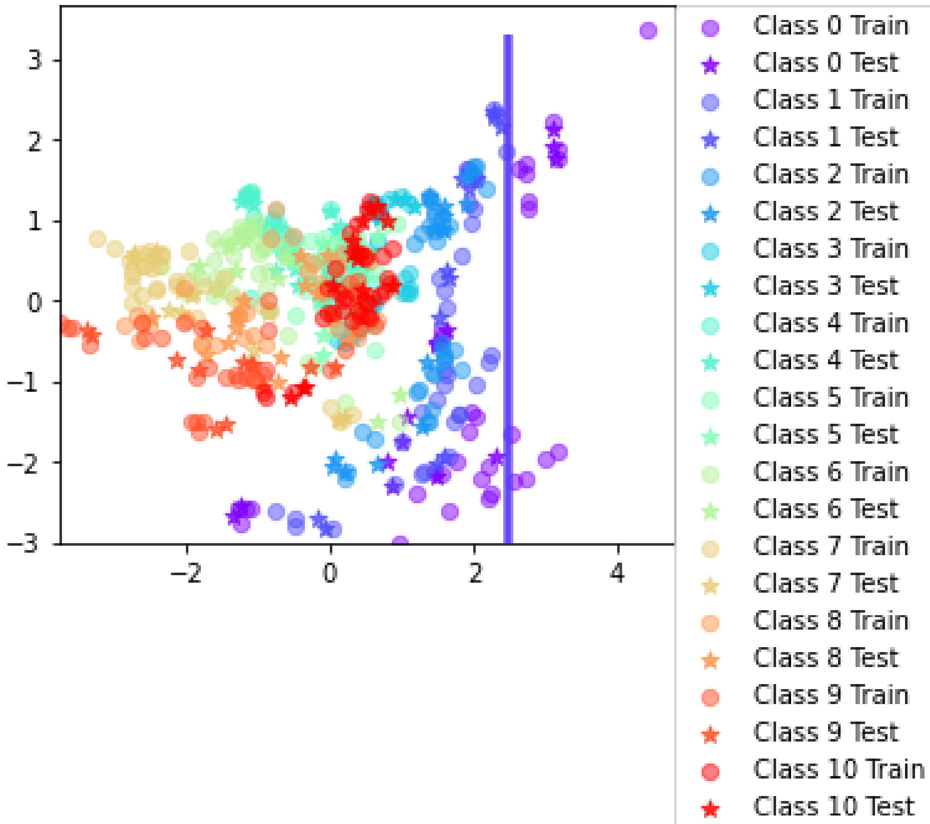

Figure 9: **Vowel** dataset plot, boosted decision tree classifier

Trials	Accuracy
0	85.1
10	85.7
20	88.3
30	91.6
40	83.8
50	79.2
60	88.3
70	86.4
80	88.3
90	89

Table 9: **Vowel** dataset, boosted decision tree classifier
Final mean classification accuracy 86.4 with standard deviation 3.11

(1)

	Iris	Vowel
Previous	92.4	64.1
Boosted	94.6	86.4
Difference	2.2	22.3

Table 10: Decision tree classifier comparison between boosted and non-boosted

Yes there is. By boosting we are identify and adjusting the model by looking at the wrongly classified samples.

Since there are more wrongly classified samples in the **Vowel** dataset, the increase in accuracy is substantially larger for this dataset.

(2)

(3)

Assignment 7

- Naive bayes, without boosting. If we were to include boosting it would overcompensate and try to adjust for said outlier, when in fact we don't want it to.
- Probably decision tree, as it computes the information gained and deems it not relevant. In pruning this would probably be removed.
- The absolute best at prediction is a naive bayes with boosting. Given that the dataset once again is not to highly correlated.
- When we are experiencing mixed data types a decision tree is better. Seeing as we have a hard time with non-continuous in the bayes classifier.
- Decision trees. As they can quite quickly categorize the data and make the problem smaller. Plus weighting in time complexity the decision tree model will be vastly faster.