18th of October 2021

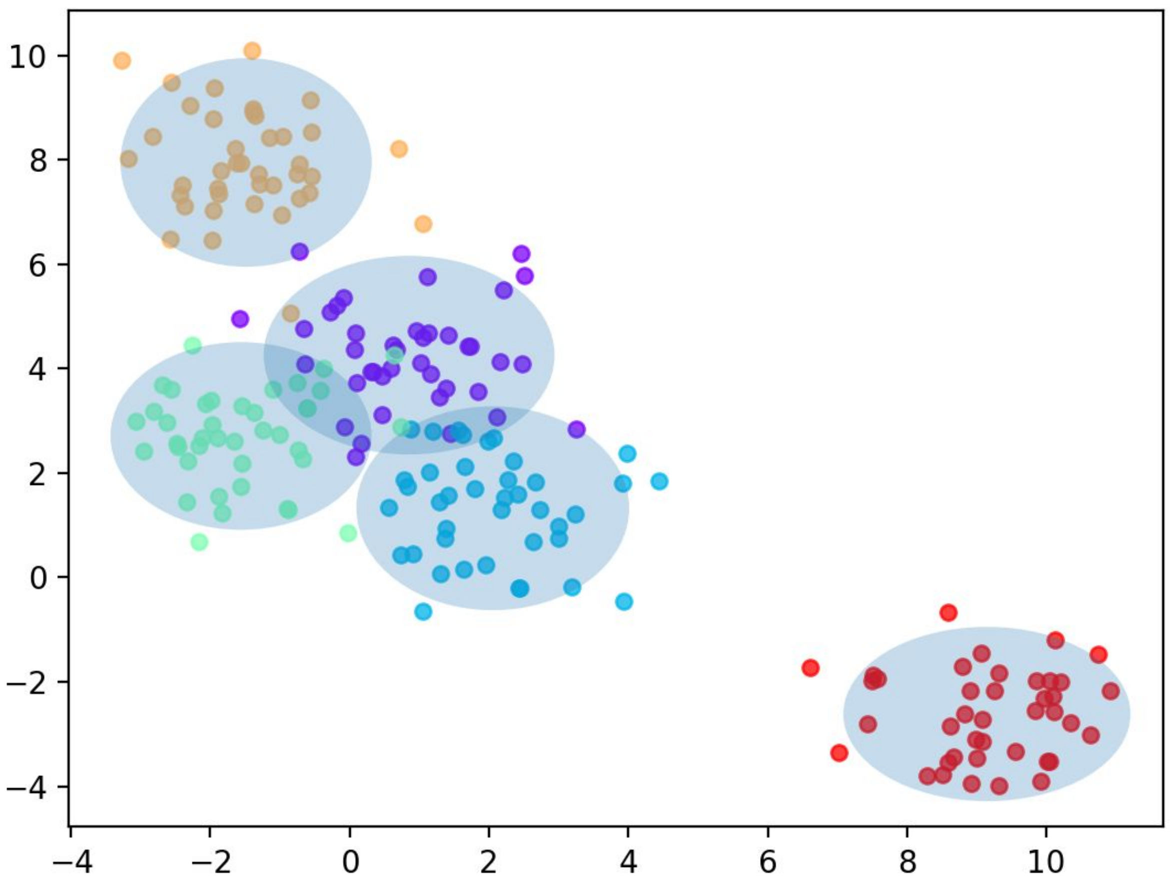
Machine Learning Lab 3

Pablo Laso, Antonia Schroff

Assignment 1

Output:

Classname: 0
Mu: 4.256886388219375
Sigma: 0.9050044083418618
Classname: 1
Mu: 1.324450006940761
Sigma: 0.9536173425007328
Classname: 2
Mu: 2.709027959660036
Sigma: 0.8082382688292771
Classname: 3
Mu: 7.947609901133649
Sigma: 0.998350167242589
Classname: 4
Mu: -2.6166770056472632
Sigma: 0.6968776368014792

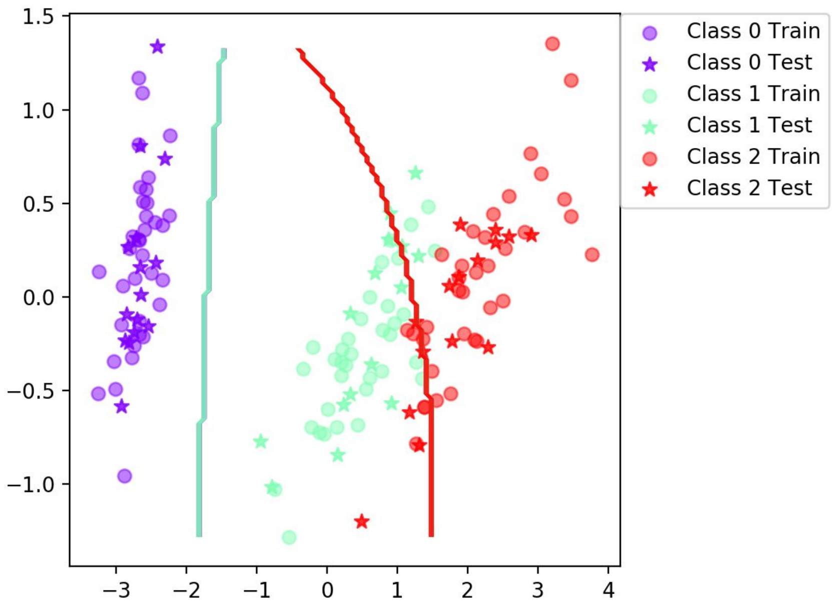


Assignment 2 & 3 - Iris Dataset

***test**Classifier(**BayesClassifier**(), dataset='iris', split=0.7)*

***plot**Boundary(**BayesClassifier**(), dataset='iris',split=0.7)*

Trial: 0 Accuracy 84.4
Trial: 10 Accuracy 97.8
Trial: 20 Accuracy 91.1
Trial: 30 Accuracy 86.7
Trial: 40 Accuracy 88.9
Trial: 50 Accuracy 91.1
Trial: 60 Accuracy 86.7
Trial: 70 Accuracy 91.1
Trial: 80 Accuracy 86.7
Trial: 90 Accuracy 91.1
Final mean classification accuracy 89.2 with standard deviation 4.19

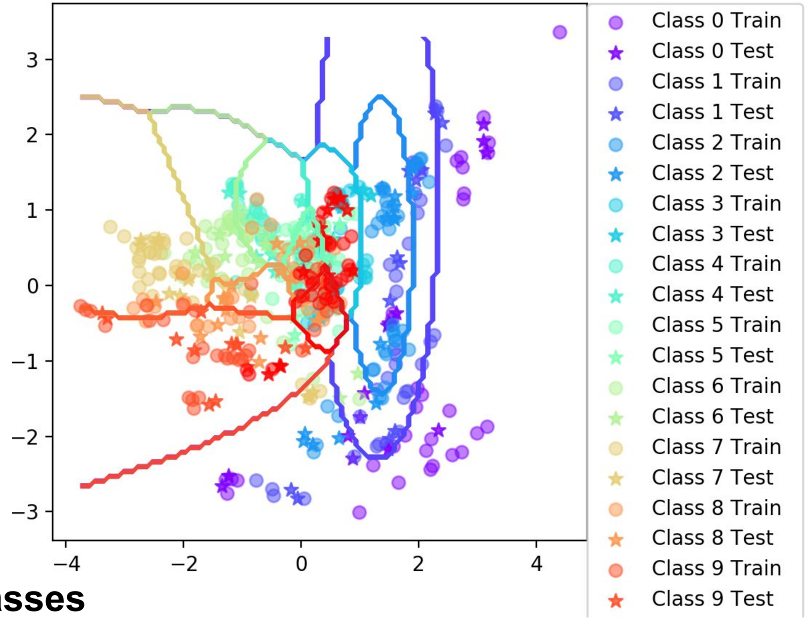


Assignment 2 & 3 - Vowel Dataset

***test**Classifier(**BayesClassifier**(), dataset='vowel', split=0.7)*

Trial: 0 Accuracy 52.6
Trial: 10 Accuracy 61.7
Trial: 20 Accuracy 68.2
Trial: 30 Accuracy 62.3
Trial: 40 Accuracy 56.5
Trial: 50 Accuracy 63
Trial: 60 Accuracy 64.3
Trial: 70 Accuracy 62.3
Trial: 80 Accuracy 60.4
Trial: 90 Accuracy 65.6
Final mean classification accuracy 61.3 with standard deviation 3.48

***plot**Boundary(**BayesClassifier**(), dataset='vowel',split=0.7)*



lower accuracy than in iris, but there are more classes

Assignment 3

*When can a **feature independence assumption** be reasonable and when not?*

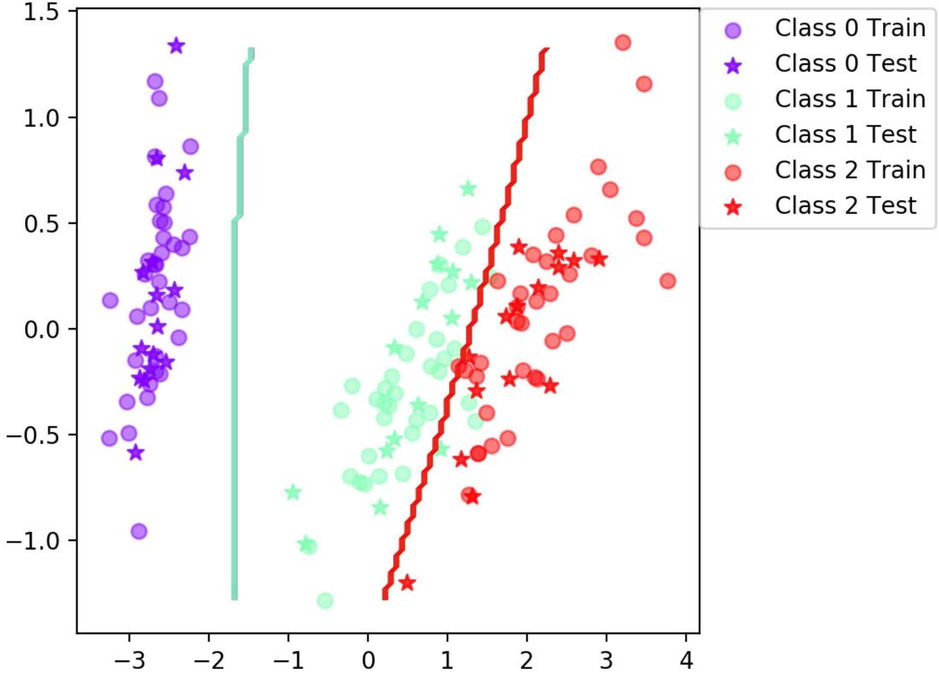
- to make use of the **naive bayes classifier**, we assume that events are independent
  - -> lower computational time
- features **actually** need to be independent!

*How does the **decision boundary** look for the Iris dataset? How could one **improve** the classification results for this scenario by **changing classifier** or, alternatively, **manipulating the data**?*

- blue decision boundary looks good, red one doesn't match.
- if we used a **SVM**, it would create a linear decision boundary that fits better.
- **KNN** could work as well, as it works well on small datasets.
- **data augmentation** to create more data points.

Assignment 4 & 5 - Iris

***plot**Boundary(**Boost**Classifier(**BayesClassifier**()), dataset='iris',split=0.7)*



Assignment 5 - Questions (Iris)

*Is there any **improvement** in classification **accuracy**? Why/why not?*

- mean accuracy improved by 5 in the iris dataset

*Plot the **decision boundary** of the boosted classifier on iris and compare it with that of the basic. What differences do you notice? Is the boundary of the boosted version more complex?*

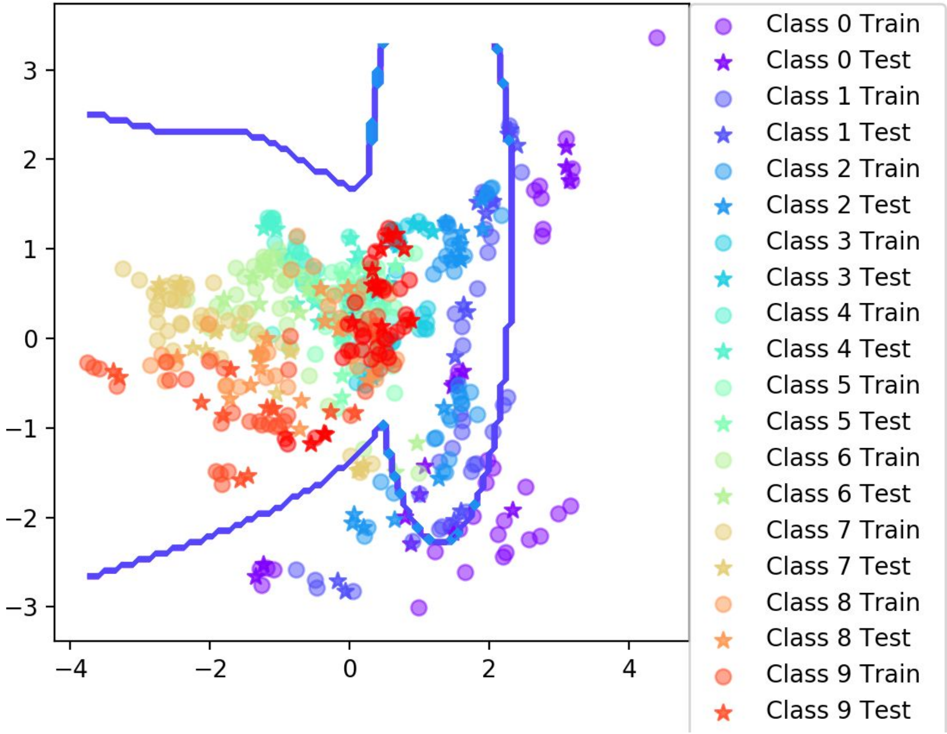
- red decision boundary a lot **more accurate** now
- **boosted** version **more complex**, look at many tiny steps in the boundary
- Booster focuses more on misclassified points, which are assigned a greater weight

*Can we make up for **not using a more advanced model** in the basic classifier (e.g. independent features) by **using boosting**?*

- YES. Boosting always makes the algorithm **more complex** and therefore **more accurate**
- risk of **overfitting**.

Assignment 4 & 5 - Vowel

***plot**Boundary(**Boost**Classifier(**Bayes**Classifier()), dataset=vowel,split=0.7)*



Assignment 5 - Questions (Vowel)

*Is there any **improvement** in classification **accuracy**? Why/why not?*

- mean accuracy improved by 12.8 (!)

*Plot the **decision boundary** of the boosted classifier on iris and compare it with that of the basic. What differences do you notice? Is the boundary of the boosted version more complex?*

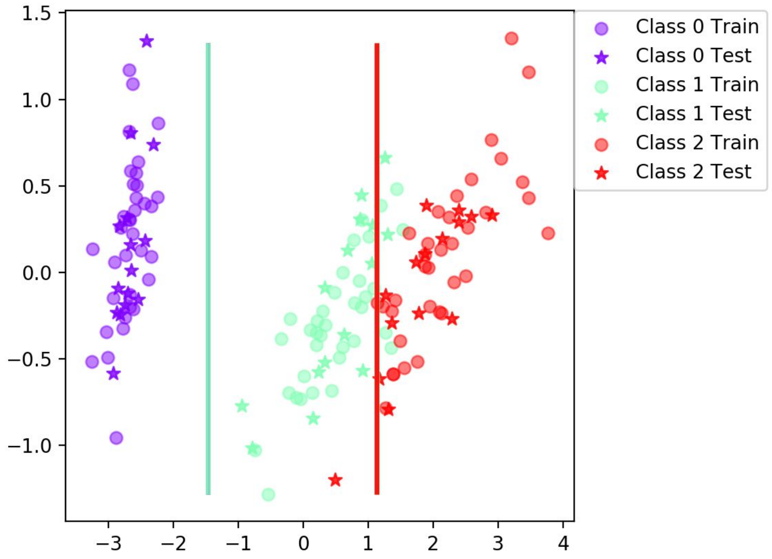
- not all decision boundaries could be computed.

*Can we make up for **not using a more advanced model** in the basic classifier (e.g. independent features) by **using boosting**?*

- YES. Boosting always makes the algorithm **more complex** and therefore **more accurate**.
- risk of **overfitting**.

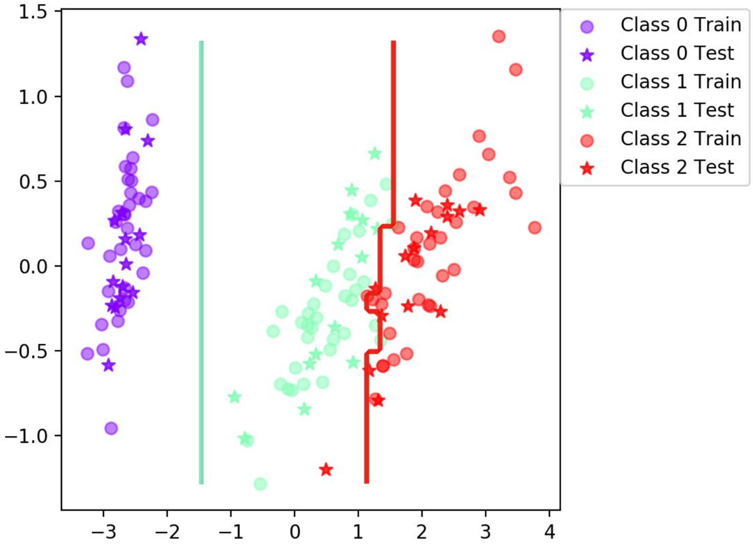
Assignment 6 - Iris

***Decision Tree** without booster*



underfitting → lower accuracy

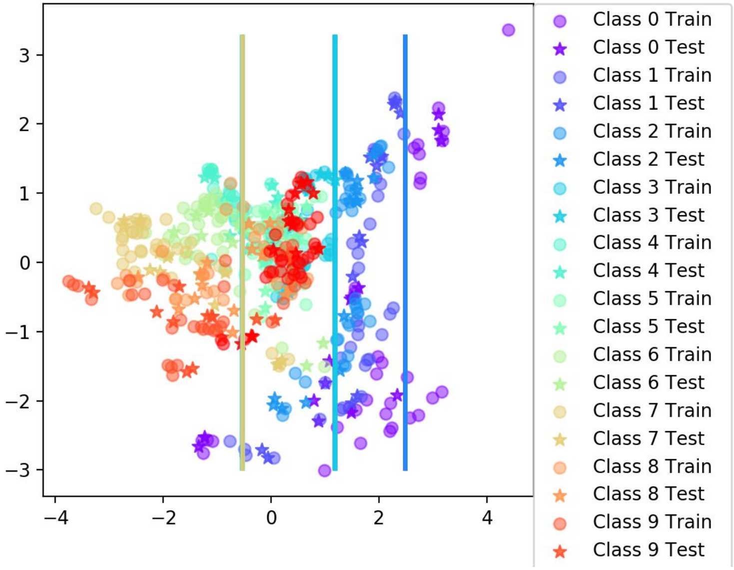
Decision Tree** with **booster



higher complexity → **higher accuracy**

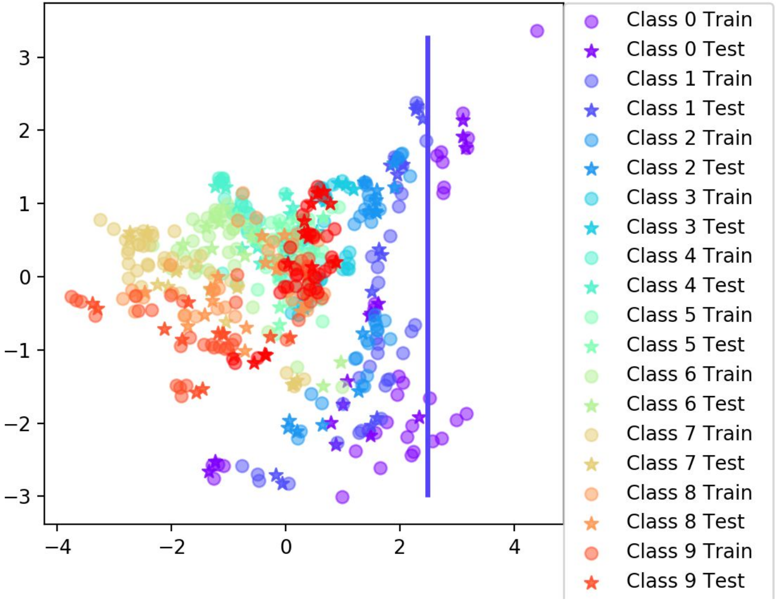
Assignment 6 - Vowel

***Decision Tree** without booster*



underfitting → lower accuracy

Decision Tree** with **booster



higher complexity → **higher accuracy**

Assignment 6 - Questions

*Is there any **improvement** in classification **accuracy**? Why/why not?*

- Iris: mean accuracy improved by 2.2
- Vowel: mean accuracy improved by 22.8

*Plot the **decision boundary** of the boosted classifier on iris and compare it with that of the basic. What differences do you notice? Is the boundary of the boosted version more complex?*

- Iris: quite similar, red class more complex
- Vowel: not all decision boundaries could be computed.

*Can we make up for **not using a more advanced model** in the basic classifier (e.g. independent features) **by using boosting**?*

- YES. Boosting always makes the algorithm **more complex** and therefore **more accurate**
- risk of **overfitting**.

Assignment 7

*If you had to **pick a classifier**, naive **Bayes or a decision tree or the boosted versions of these**, which one would you pick? Motivate from the following criteria:*

- **Outliers**:

Boosted versions might be too complex. It is risky to use them because outliers will be misclassified, and assigned a greater weight, so the algorithm will focus more on them (we risk overfitting). We could either remove outliers before training, or perhaps use less iterations, so that the boundary does not become too complex.

Bayes or DT really depend on the dataset. However, it is possible that NB is more sensitive to outliers, and the distribution might be greatly deformed to adapt to those outliers, by for example, increasing variance too much and losing accuracy. DT work by splitting the feature space, and are more robust to outliers.

Assignment 7

If you had to ***pick a classifier****, naive* ***Bayes or a decision tree or the boosted versions of these****, which one would you pick? Motivate from the following criteria:*

- **Irrelevant inputs**: part of the feature space is irrelevant

  NB would try to adapt to that data, probably increasing the variance and shifting along the feature space by changing the mean, thus compromising accuracy (higher bias). DT might perform better when using Information Gain to decide if that data really contributes to reduce Entropy. Pruning might also make it work better.

- **Predictive power**:

  Again, it is difficult to know beforehand, but boosting versions will be able to deal with more complex datasets and most likely give better results. Trying all and choosing the one rendering the highest test accuracy might be a nice first approach.

Assignment 7

*If you had to **pick a classifier**, naive **Bayes or a decision tree or the boosted versions of these**, which one would you pick? Motivate from the following criteria:*

• **Mixed types of data**: binary, categorical or continuous features, etc.

Binary and categorical data might be better fit by a DT, although NB may also work just as fine.

Continuous data, especially if normally distributed, will be more readily fit by a Bayes classifier.

• **Scalability**: the dimension of the data, D, is large or the number of instances, N, is large, or both.

NB might be computationally more expensive as data grows (larger matrices), while DT computational costs would largely depend on the data distribution throughout the feature space and its ability to fit those distributions.

The end :)