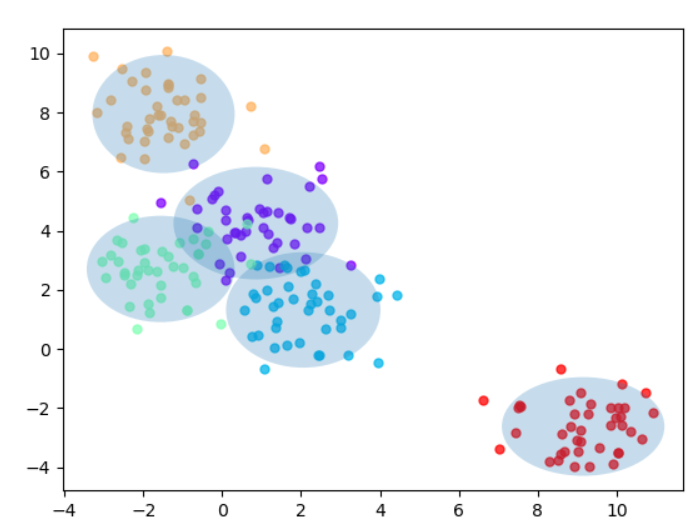
|  |  |
| --- | --- |
|  |  |
|  | Machine Learning |
|  |  |
|  |  |
|  | **Lab 3: Bayesian Learning and Boosting**  LinusGroß  DanielMensah |
|  |  |

|  |
| --- |
|  |

# Bayesian Learning

1. *Maximum-likelihood estimates of μ and ∑*



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

When we have a high number of features (/Dimension) it can be assumed to be reasonable to use the naive Bayes Classifier as an approximation. We consider the dimensions as conditionally independent. This simplifies all our calculations and reduces the model complexity.

If we know that the features are strongly correlated or dependent, this will of course lead to imprecise classification.

On the other hand, if we only have low dimensions, it could prove more useful to use another classifier that can work on fewer data and it will not be too much of a calculation hurdle.

1. *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?*

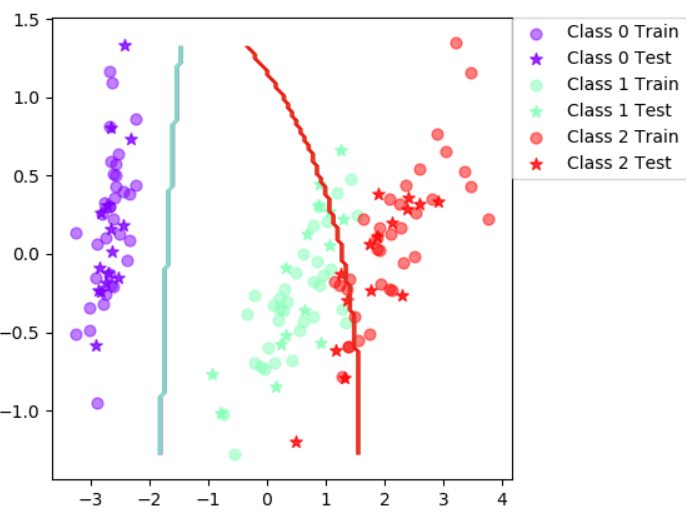


Figure 1: Naïve Bayes classifier, Accuracy 89%

The decision boundary doesn’t really fit the boundary between class 1 and class 2. This leads to the misclassification of test and even training samples. This could be due to the fact that these two classes are located close to each other and they even overlap. The decision boundary between class 0 and 1 is however very simple.

A clear decision boundary could be achieved be using e.g. a SVM (as used in the last Lab), because it is able to better deal with overlapping and more complex datasets (Slack, kernel).

# Boosting

*Compute the classification accuracy of the boosted classifier on some data sets using testClassifier from labfuns.py and compare it with those of the basic classifier on the vowels and iris data sets (see Assignment 3):*

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

The ‘iris’ dataset shows an improvement in its classification accuracy. The accuracy increases from 89% up to 94,1%. Since only 2 attributes are considered the unboosted classifier already shows a decent result

The vowel dataset shows an even bigger increase in accuracy, from 64,7% up to 80,2%. This is due to multiple reasons. First, the dataset is much bigger than the iris dataset. Furthermore, the 10 attributes are not easily separable by a single classifier and the boosted one increases the accuracy.

1. *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?*



Figure 2: Boosted Naive Bayes classifier, Accuracy 94,1%

The decision boundary separates the two classes better and this is because it takes a more complex shape than without boosting.

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

Yes, since the accuracy increases. For the vowel dataset, boosting increases the accuracy dramatically. The iris-dataset is however not really affected by boosting, using a more advanced basic classifier would be better. Using a SVM would probably lead to better results since the task is to give a decision boundary with a complex shape.

# Decision Tree Classifier

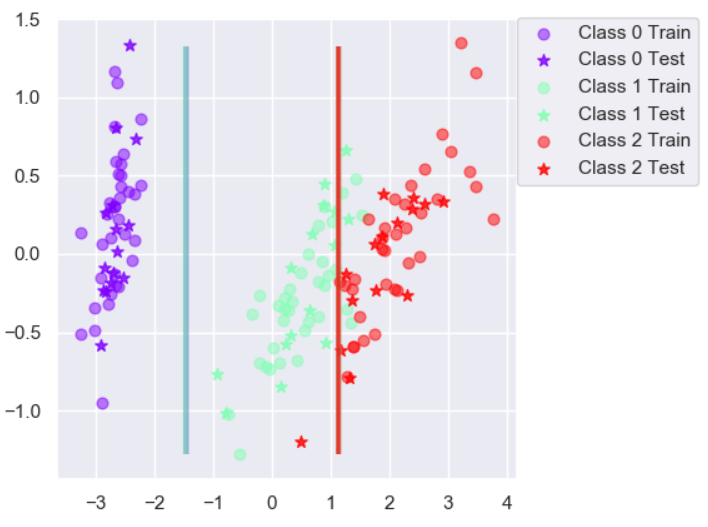


Figure 3: Decision Tree classifier, Accuracy 92,4%

The decision tree classifier leads to a linear separation between the classes. For the iris-dataset, only one attribute is used for the decision.

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

For the iris-dataset, boosting increases the accuracy from 92,4% to 94,6%. Since the unboosted classifier already delivers good results, boosting doesn’t increase the accuracy much.

The accuracy of the vowel dataset is increased by boosting from 64,1% to 86,8%. This is again due to the high number of attributes that have to be taken into account.

1. *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?*

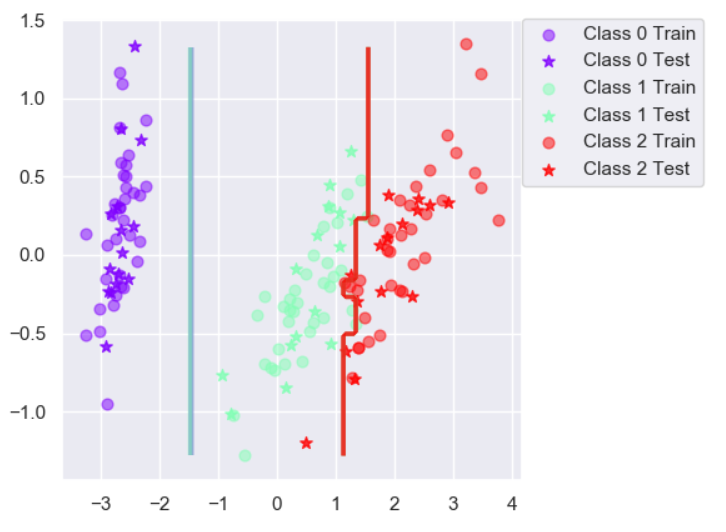


Figure 4: Boosted Decision Tree classifier, Accuracy 94,6%

In comparison to the unboosted classifier, boosting leads to a more complex decision boundary by taking the second attribute into account. Since the linear decision boundary already leads to good results, the improvement in accuracy is not very high.

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

For the vowel dataset, boosting increases the accuracy dramatically. The iris-dataset is however not really affected by boosting, using a more advanced basic classifier would be better.

*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*

Naïve bayes is a good choice because if there are few outliners, they don’t have a big impact in calculating the ML-parameters, if many training datapoints are available.

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

Decision trees can be used, e.g. by pruning the irrelevant inputs can be filtered.

* *Predictive power*

Since the boosting always increases the accuracy, this can be used to increase the prediction power

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

If we have a high dimension of the data, a decision tree is no good choice because it gets uncontrollably large. A naïve bayes classifier however can handle this good because it ignores dependencies between the dimensions.

A high number of instances can lead to large calculation effort if boosting is used, cause several classifiers have to be trained on the large dataset.