

Machine Learning Lab 3: Bayesian Learning and Boosting

Tianxiao Zhao
Feiyang Liu

Mar. 1, 2017

Why Adaboost ?

- Convert weak learners to strong ones
- Weak learners are usually easy to get
- Adaboost is a forward stage-wise additive algorithm using the exponential loss function, tending to reduce bias and variance

Pros & Cons of Adaboost

■ Pros

- Reduce bias and variance
- Reduce dimensions of the model and the optimization problem (By using linear combination of weak learners)
- Not easy to overfit in general

■ Cons

- Sensitive to Outliers (Because boosting gets base learner on previous errors. Outliers will have get much higher attention than non-outliers)

How Adaboost works ?

- Training

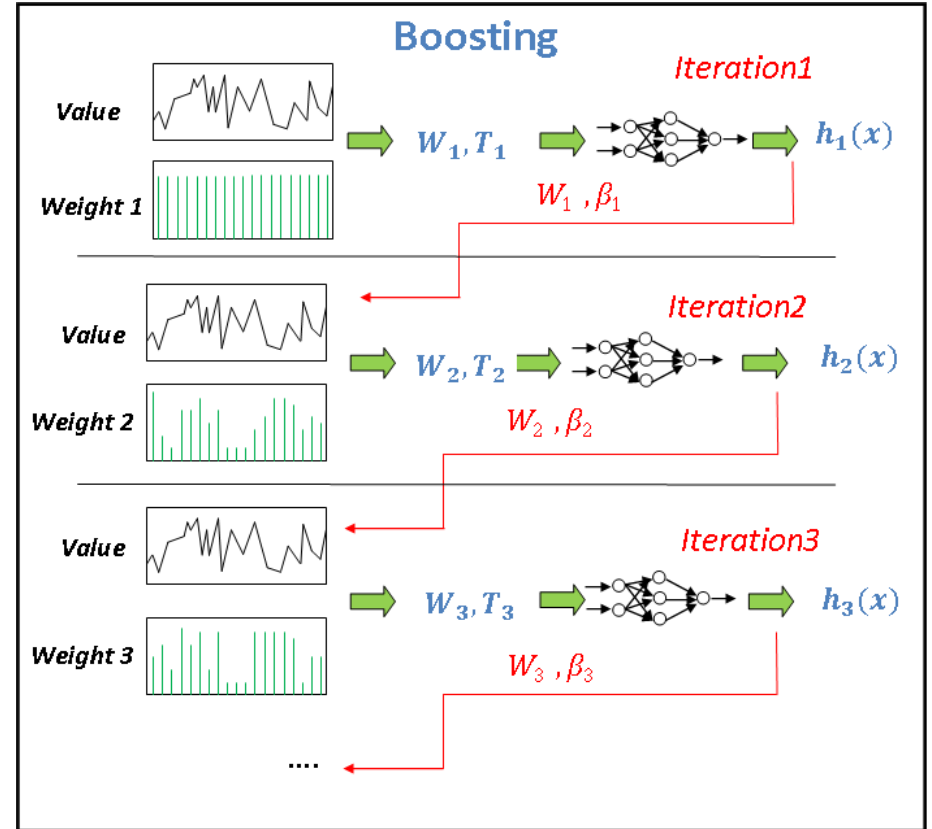
- Step 1: Assign equal weight to each data
- Step 2: Train weak learner with weights
- Step 3: Get weak learner and its error with respect to weights
- Step 4: Compute alpha based on error
- Step 5: Update weights (pay higher focus on misclassified examples)

■ Testing

- Step 1: Classify using each weak learner
- Step 2: Linear combination of outputs of all weak learners

■ Summary of the Idea

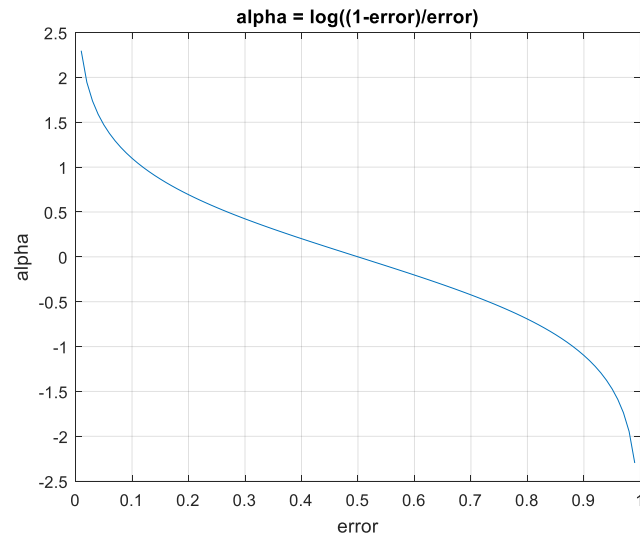
- Use weighted average
- Consider prediction of higher vote



How Adaboost works ? (Cont')

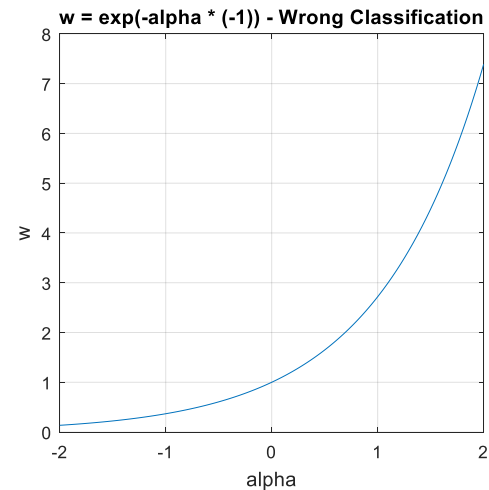
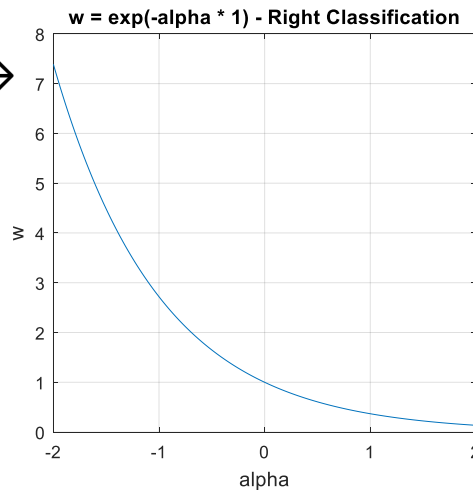
■ Alpha vs. Error

- $\text{error} > 0.5 \rightarrow \alpha < 0$
- $\text{error} \uparrow \alpha \downarrow$
- Vice versa

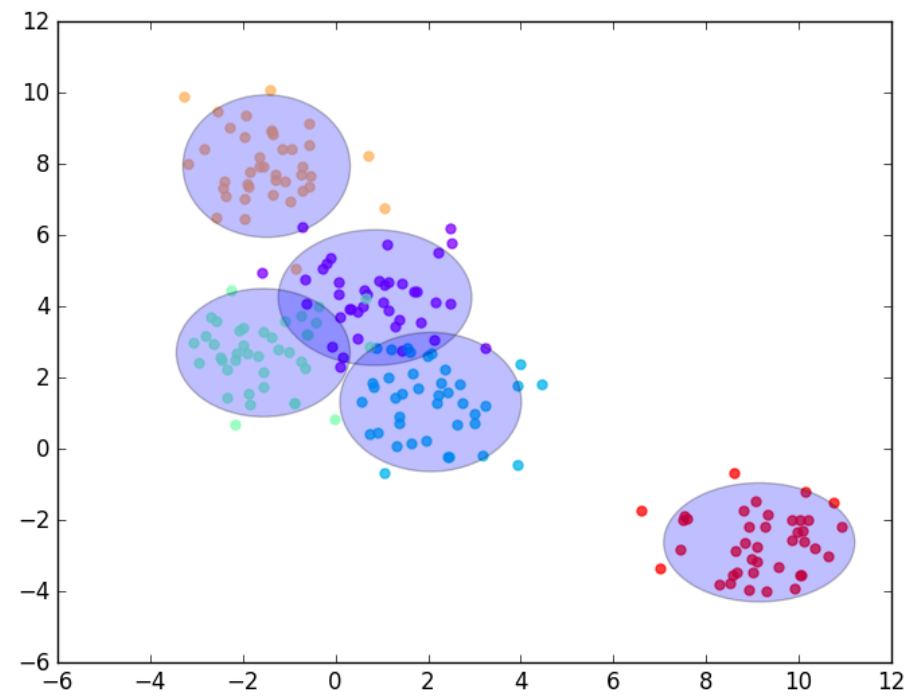


■ Weights vs. Alpha

- Right Classification & $\alpha > 0 \rightarrow$ weights \downarrow
- Wrong Classification & $\alpha > 0 \rightarrow$ weights \uparrow
- Vice versa



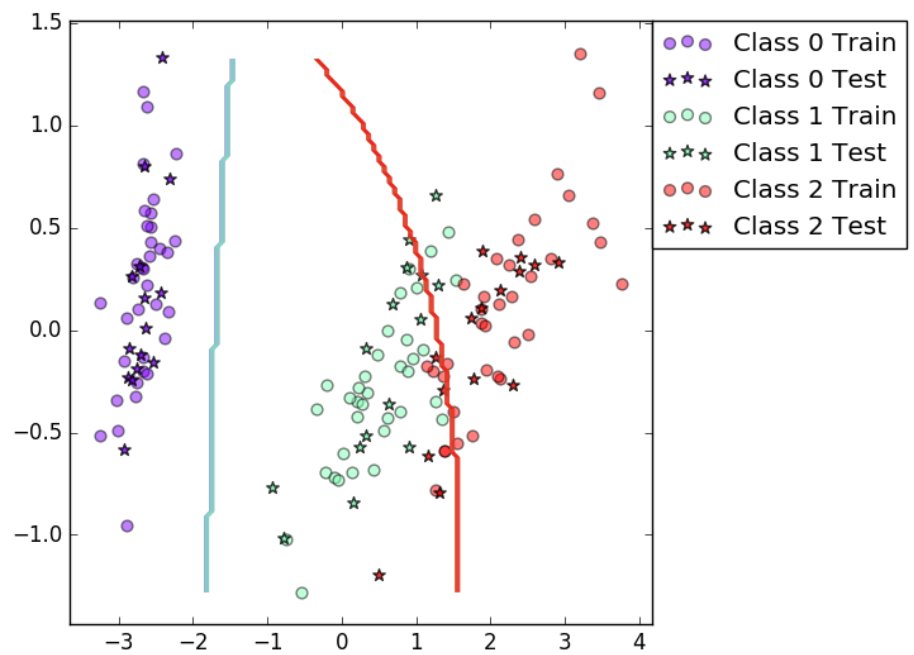
Experiment 1: Naïve Bayesian Maximum Likelihood Estimates



- Assumption: All of the feature dimensions are **uncorrelated**
- Covariance matrix is diagonal

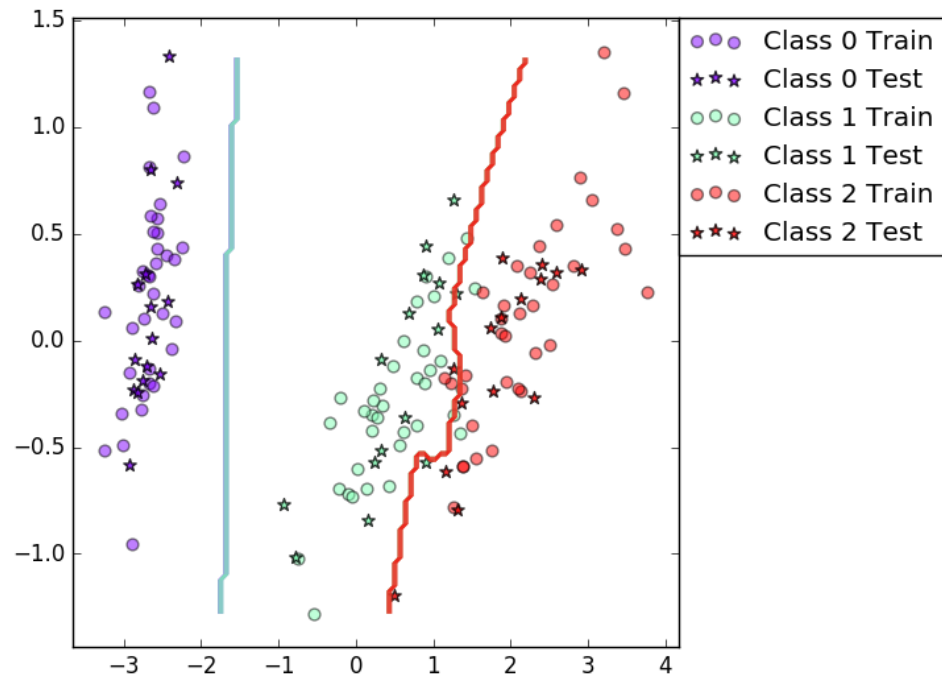
Experiment 2: Naïve Bayes Classifier on the Iris Dataset

Non-boosting



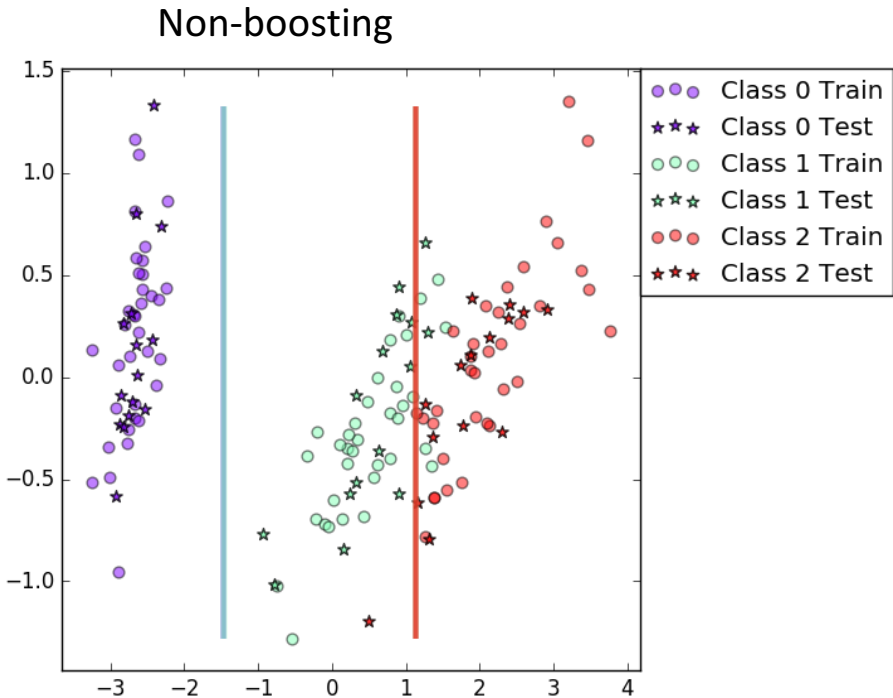
■ Final mean classification accuracy 89
with standard deviation 4.16

Boosting

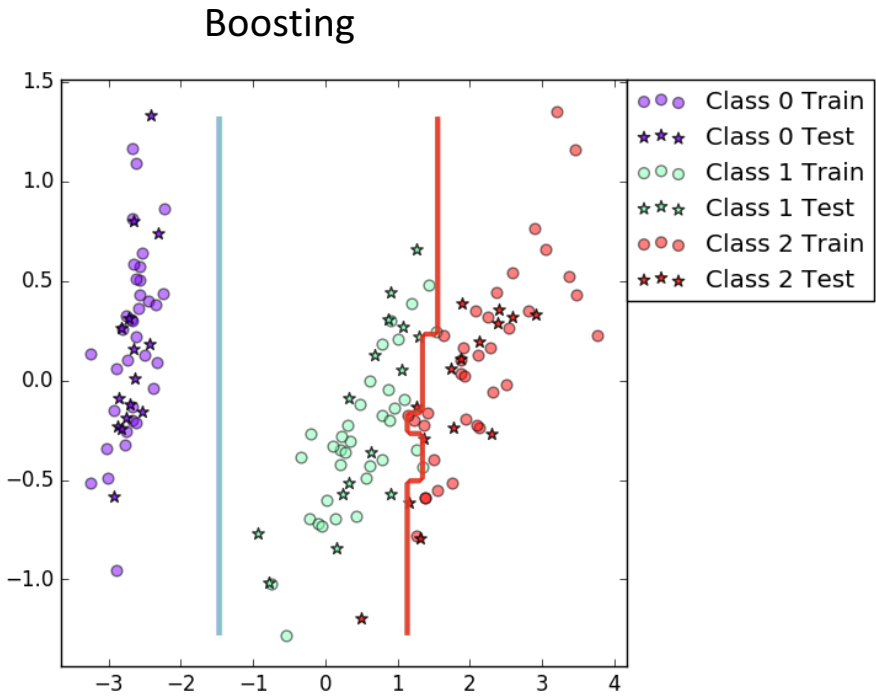


■ Final mean classification accuracy 94.1 ↑
with standard deviation 6.72 ↑

Experiment 3: Decision Tree Classifier on the Iris Dataset



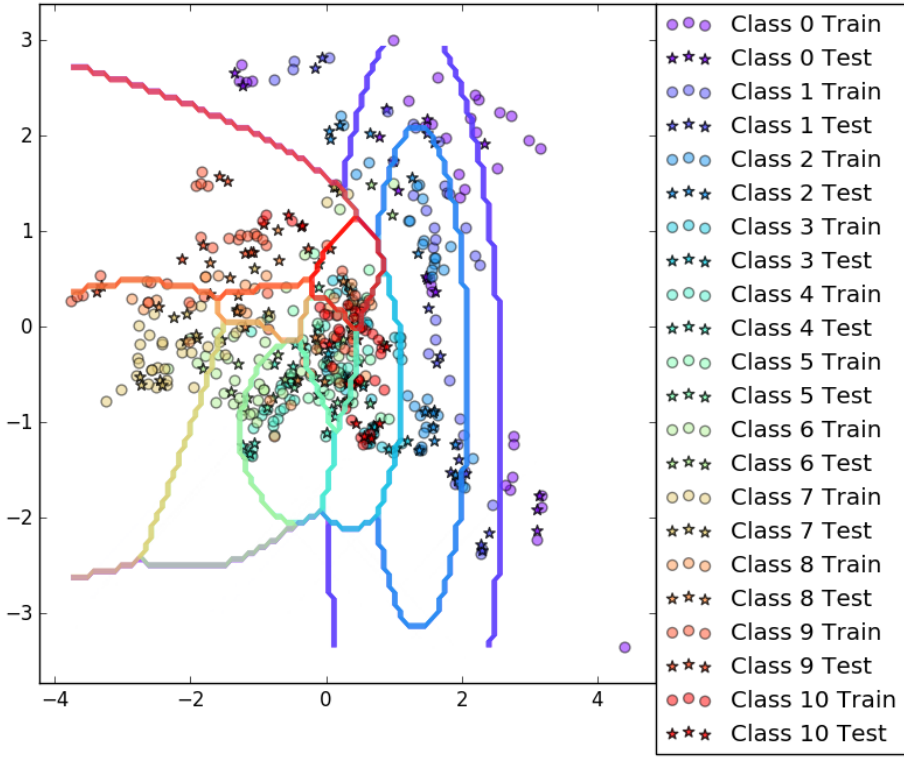
■ Final mean classification accuracy 92.4
with standard deviation 3.71



■ Final mean classification accuracy 94.6 ↑
with standard deviation 3.65 ↓

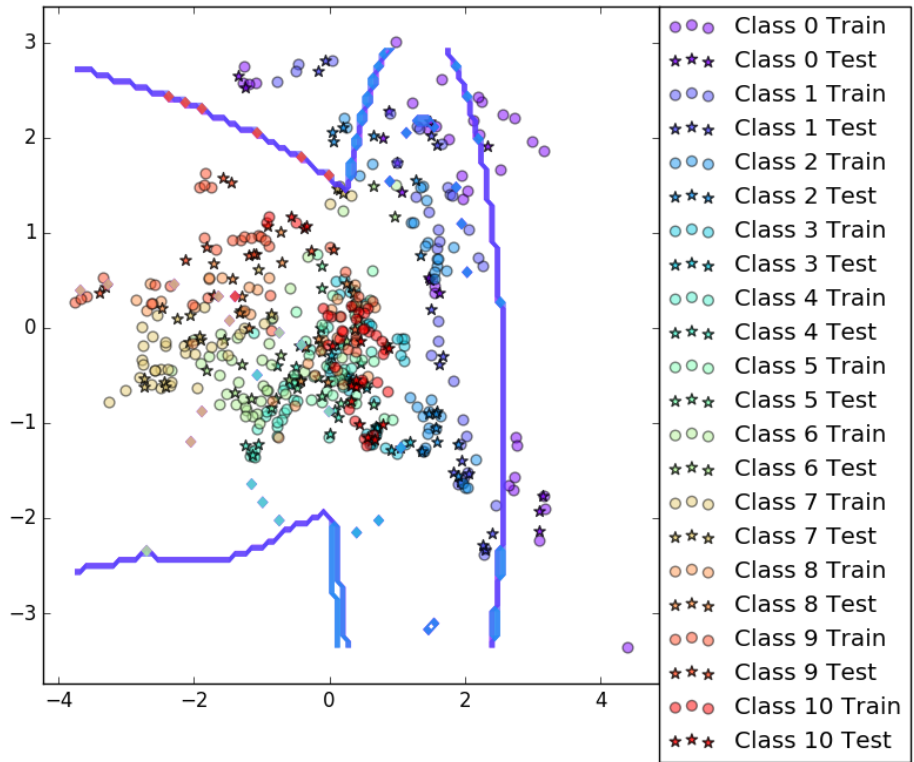
Experiment 4: Naïve Bayes Classifier on the Vowel Dataset

Non-boosting



■ Final mean classification accuracy **64.7**
with standard deviation **4.03**

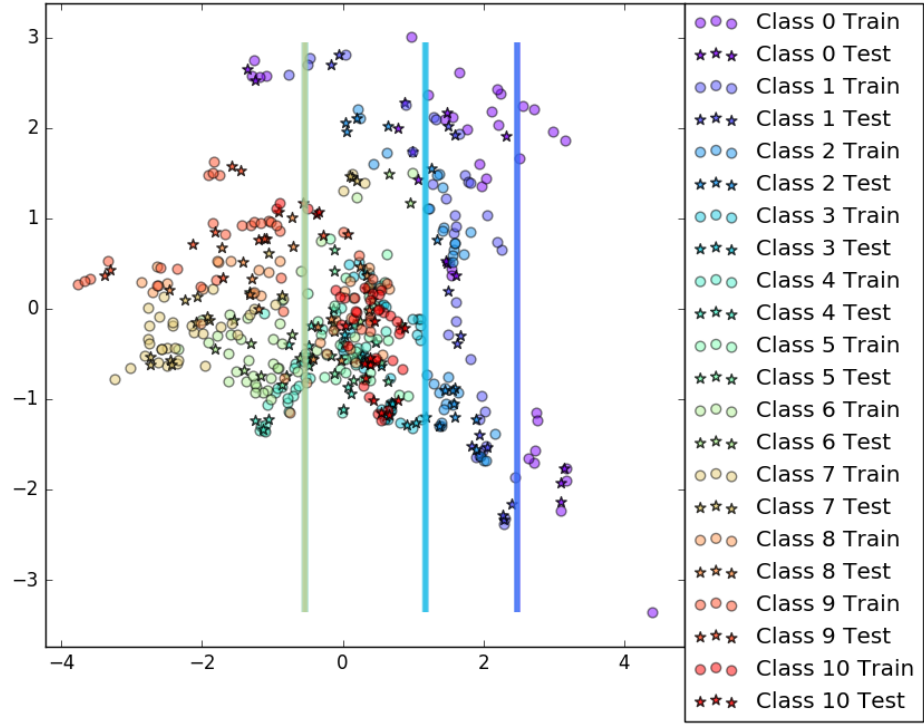
Boosting



■ Final mean classification accuracy **80.2** ↑
with standard deviation **3.52** ↓

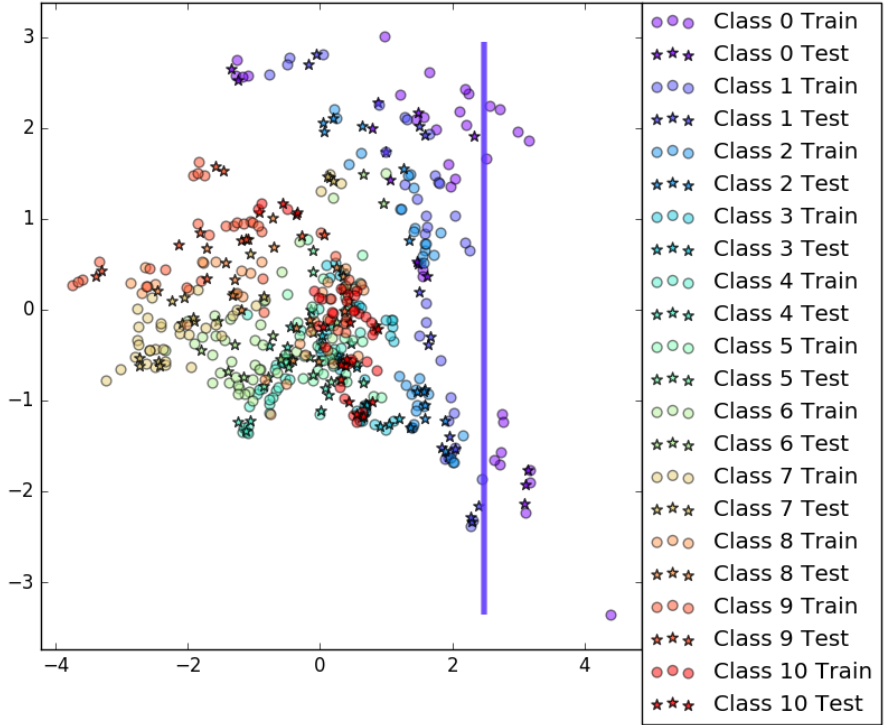
Experiment 5: Decision Tree Classifier on the Vowel Dataset

Non-boosting



■ Final mean classification accuracy **64.1**
with standard deviation **4**

Boosting



■ Final mean classification accuracy **86.9** ↑
with standard deviation **3.06** ↓

Experiment 6: Naïve Bayes Classifier on the Olivetti Dataset

- Final mean classification accuracy 87.7 with standard deviation 3.03

Experiment 7: Boosting Naïve Bayes Classifier on the Olivetti Dataset (Failed)

- Final mean classification accuracy 2.5 with standard deviation 0

Experiment 8: Decision Tree Classifier on the Olivetti Dataset (Failed)

- Final mean classification accuracy 48.5 with standard deviation 6.27

Experiment 9: Boosting Decision Tree Classifier on the Olivetti Dataset

- Final mean classification accuracy 71 ↓ with standard deviation 6.07 ↑

Experiment 10: Classify faces Using Boosting Decision Tree on Olivetti

Test image



Matched class training image 1



Matched class training image 2



Matched class training image 3



Matched class training image 4



Matched class training image 5



Matched class training image 6



Matched class training image 7



Experiment 10: Classify faces Using Boosting Decision Tree on Olivetti (cont')

Test image



Matched class training image 1



Matched class training image 2



Matched class training image 3



Matched class training image 4



Matched class training image 5



Matched class training image 6



Matched class training image 7

