CS789: Machine Learning and Neural Network

Ensemble methods

Jakramate Bootkrajang

Department of Computer Science
Chiang Mai University

Introduction

- We've seen the working of a single classifier.
- We will now explore the possibility of combining outputs of those classifiers to make (more accurate) prediction.
- The method is call **ensemble learning**

What makes good ensemble?

1. A member of the ensemble is accurate.
   - An accurate classifier is one that has error rate of better than random guessing
   - $\epsilon < 0.5$
2. The ensemble is composed of diverse classifiers.
   - Two classifiers are diverse if they make differrent errors on new data points.

More on diversity

- To see why diversity is important, imagine there are three classifier in the ensemble h_1, h_2, h_3
- If the three classifiers predict the same thing (not diverse)
  - then when h_1 makes a mistake the others will too.
- But if the classifiers are uncorrelated (diverse)
  - when h_1 makes a mistake, h_2, h_3 might not and by majority voting the final prediction is still correct.

Reasons why ensemble often be more accurate [1/3]

- It solves statistical problem related to learning from limited number of training data.
- Ensemble reduces the risk of choosing wrong hypothesis.

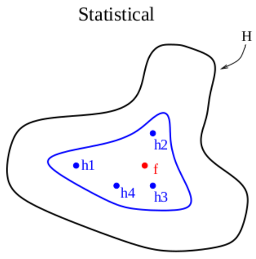


Figure: Credit: Thomas G. Dietterich, Ensemble Methods in Machine Learning

Reasons why ensemble often be more accurate [2/3]

- Even in the abundance of data, the problem might have several local optima.
- Ensemble reduces the risk of stucking in local optima.

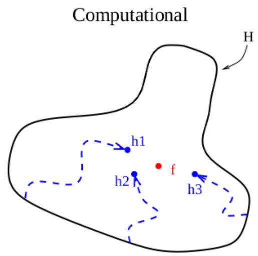


Figure: Credit: Thomas G. Dietterich, Ensemble Methods in Machine Learning

Reasons why ensemble often be more accurate [3/3]

- Ensemble alleviates the wrong choice of choosing hypothesis space.
  - That is data is not linearly-separable but linear hypothesis class is chosen.
- An ensemble of linear classifiers can have non-linear decision boundary.

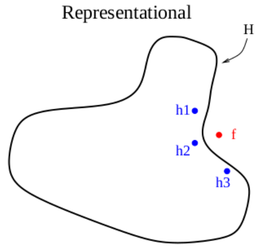


Figure: Credit: Thomas G. Dietterich, Ensemble Methods in Machine Learning

How to construct an ensemble?

- Ensemble of G members in general is given by:

$$f(x) = \sum_{i=1}^{G} w_i h_i(x)$$

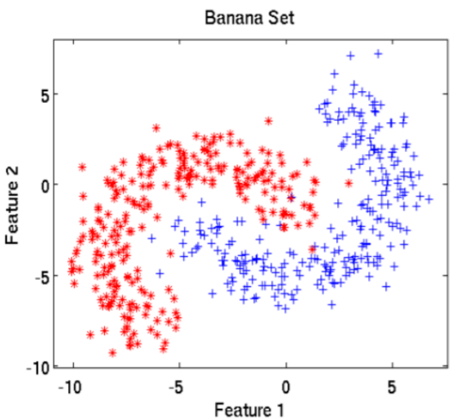
- Methods for constructing an ensemble differ in
  - How to determine w_i, the contribution of $h_i(x)$
  - How to get diverse set of $h_i(x)$.
    - Introduce some randomness to the problem or learner

Boostrap Aggregating: Bagging

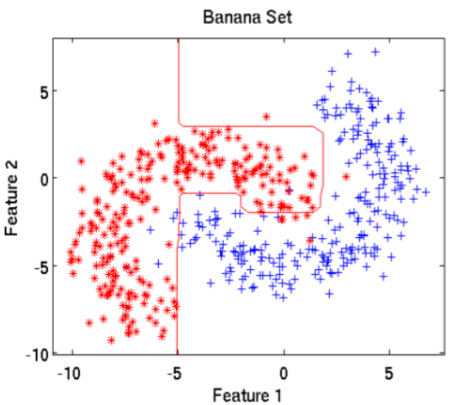
- Manipulating the input data.
- How to get diverse $h_i(x)$?
  - Sample m examples from the training set randomly, with replacement.
  - Train a classifier on the *bootstrap replicate*.
  - For each boostrap, a classifier only see part of the whole data.
- What are the weights w_i's ?
  - Classifiers are combined using identical weights.

$$f_{bagging}(x) = \sum_{i=1}^{G} h_i(x)$$

Bagging: Example (1/3)



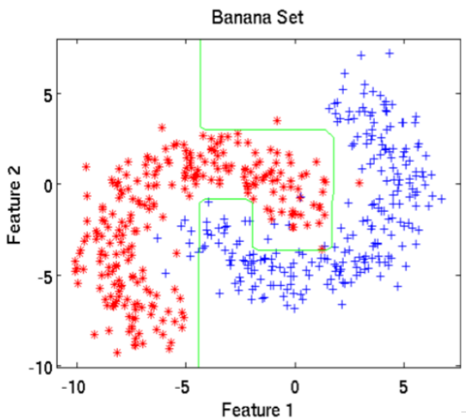
Training data



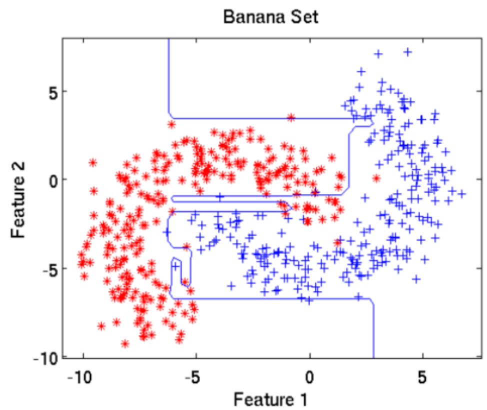
Decision boundary produced by one tree

Amit Srinet, Dave Snyder

Bagging: Example (2/3)



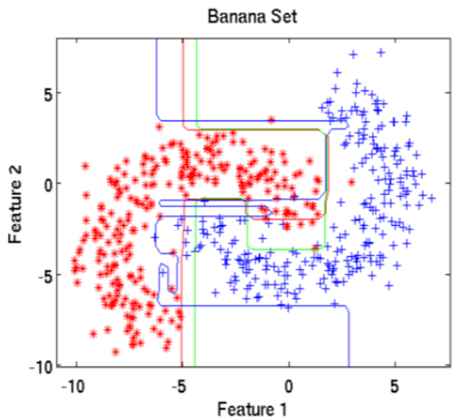
Decision boundary produced by a second tree



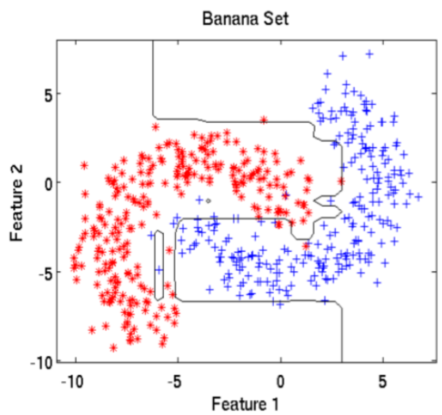
Decision boundary produced by a third tree

Amit Srinet, Dave Snyder

Bagging: Example (3/3)



Three trees and final boundary overlaid



Final result from bagging all trees.

Amit Srinet, Dave Snyder

Adaptive Boosting: AdaBoost

- Instead of sampling, uses trainining set re-weighting.
- Place more weight on 'difficult' examples.
- Classifiers are combined using

$$f_{ada}(x) = \sum_{i=1}^{G} \alpha_i h_i(x)$$

- α_i is set according to h_i's accuracy on the weighted training set.
- $h_i(x)$ is called a *weak learner*.

AdaBoost algorithm

Data: $S = \{x_i, y_i\}_{i=1}^N$, $x_i \in X$ and $y_i \in \{-1, 1\}$
initialization: uniform weight for initial data $D_1(i) = \frac{1}{N}$;
for $t = 1 \dots T$ **do**
  Learn a classifier $h_t : X \to \{-1, 1\}$ that minimises training error,
  $\epsilon_j = \sum_{i=1}^N D_t(i)[y_i \neq h_j(x_i)]$;
  if $\epsilon_t > 0.5$ **then**
    STOP;
  else
    Set $\alpha_t = \frac{1}{2} \ln \frac{1-\epsilon_t}{\epsilon_t}$;
    Reweighting by $D_{t+1}(i) = \frac{D_t(i) \exp(-\alpha_t y_i h_t(x_i))}{Z_t}$;
    (Z_t is a normaliser making $\sum_{i=1}^N D_{t+1}(i) = 1$);
  end
end
Result: $f_{ada}(x) = \sum_{t=1}^T \alpha_t h_t(x)$

AdaBoost: reweighting

- Place more weight on 'difficult' examples.

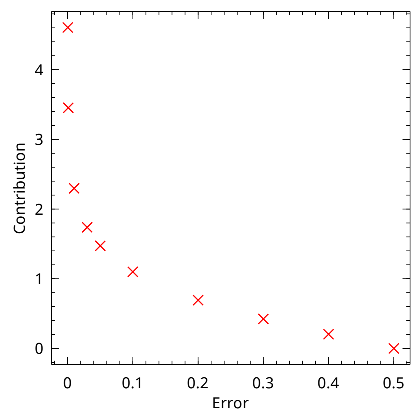
$$D_{t+1}(i) = \begin{cases} \frac{D_t(i)\exp(-\alpha_t)}{Z_t} & \text{if } y = h_t(x_i) \\ \frac{D_t(i)\exp(\alpha_t)}{Z_t} & \text{if } y \neq h_t(x_i) \end{cases} \tag{1}$$

- α_t is set according to h_t's accuracy $(1 - \epsilon_t)$ on the weighted training set.

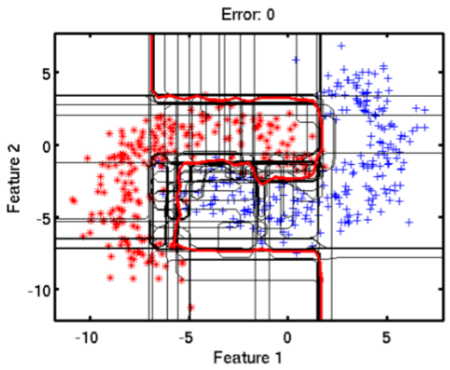
$$\alpha_t = \frac{1}{2} \ln \frac{1 - \epsilon_t}{\epsilon_t} \tag{2}$$

  - $\epsilon = 0.5$, $\alpha = 0$
  - $\epsilon = 0.4$, $\alpha = 0.20$
  - $\epsilon = 0.3$, $\alpha = 0.42$
  - $\epsilon = 0.1$, $\alpha = 1.09$

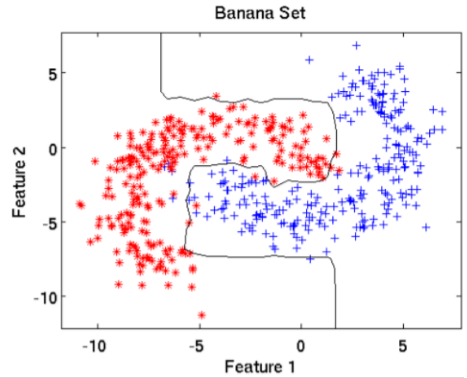
Effect of ϵ on α (contribution)



Boosting: Example (1/2)



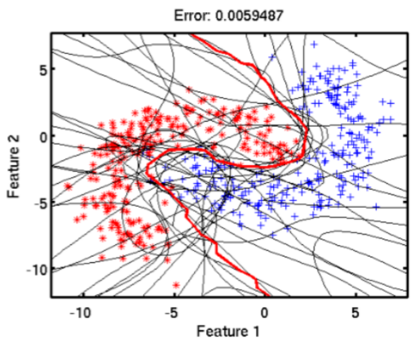
AdaBoost using 20 decision trees with default settings



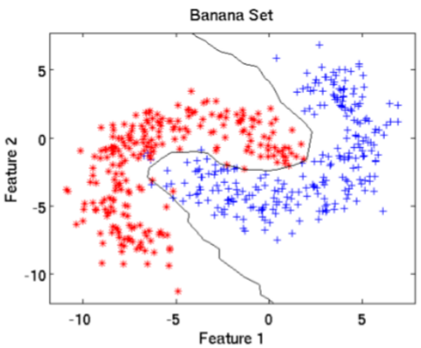
Final output of AdaBoost with 20 decision trees

Amit Srinet, Dave Snyder

Boosting: Example (2/2)



AdaBoost using 20 neural nets [bpxnc] default settings



Final output of AdaBoost with 20 neural nets

Amit Srinet, Dave Snyder

Summary

- Ensembles are method for obtaining highly accurate classifiers by combining less accurate ones.
- This is another approach to solve non-linear problem.
- Well known algorithms include, bagging and boosting.

References

- Ensemble Methods in Machine Learning by Tom Dietterich. `web.engr.oregonstate.edu/~tgd/publications/mcs-ensembles.pdf`
- Freund; Schapire (1999). "A Short Introduction to Boosting" (PDF): introduction to AdaBoost