Ensemble Learning

March 30, 2010
CS 489/698
University of Waterloo

Outline

- Ensemble Learning
  - Bagging
  - Boosting
- Reading:
  - Bishop Sect 14.2, 14.3
  - Russell & Norvig Sect 18.4

Supervised Learning

- So far…
  - Decision trees
  - Statistical learning
    - Bayesian Learning
    - Maximum a posteriori
    - Maximum likelihood
- Which technique should we pick?

Ensemble Learning

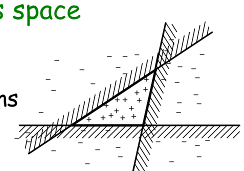
- Sometimes each learning technique yields a different hypothesis
- But no perfect hypothesis…
- Could we combine several imperfect hypotheses into a better hypothesis?

Ensemble Learning

- Analogies:
  - Elections combine voters' choices to pick a good candidate
  - Committees combine experts' opinions to make better decisions
- Intuitions:
  - Individuals often make mistakes, but the "majority" is less likely to make mistakes.
  - Individuals often have partial knowledge, but a committee can pool expertise to make better decisions.

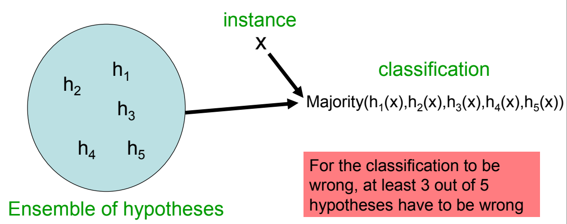
Ensemble Learning

- Definition: method to select and combine an ensemble of hypotheses into a (hopefully) better hypothesis
- Can enlarge hypothesis space
  - Perceptrons
    - linear separators
  - Ensemble of perceptrons
    - polytope



Bagging

- Majority Voting



Bagging

- Assumptions:
  - Each h_i makes error with probability p
  - The hypotheses are independent
- Majority voting of n hypotheses:
  - k hypotheses make an error: $\binom{n}{k} p^k(1-p)^{n-k}$
  - Majority makes an error: $\Sigma_{k>n/2} \binom{n}{k} p^k(1-p)^{n-k}$
  - With n=5, p=0.1 → err(majority) < 0.01

Weighted Majority

- In practice
  - Hypotheses rarely independent
  - Some hypotheses have less errors than others
- Let's take a weighted majority
- Intuition:
  - Decrease weight of correlated hypotheses
  - Increase weight of good hypotheses

Boosting

- Most popular ensemble technique
- Computes a weighted majority
- Can "boost" a "weak learner"
- Operates on a weighted training set

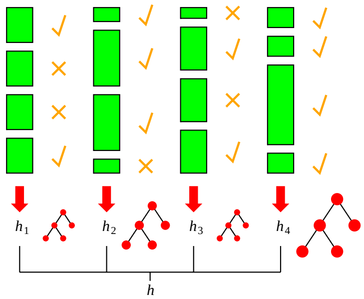
Weighted Training Set

- Learning with a weighted training set
  - Supervised learning → minimize train. error
  - Bias algorithm to learn correctly instances with high weights
- Idea: when an instance is misclassified by a hypothesis, increase its weight so that the next hypothesis is more likely to classify it correctly

Boosting Framework

- Set all instance weights w_x to 1
- Repeat
  - h_i ← learn(dataset, weights)
  - Increase w_x of misclassified instances x
- Until sufficient number of hypotheses
- Ensemble hypothesis is the weighted majority of h_i's with weights w_i proportional to the accuracy of h_i

Boosting Framework



AdaBoost (Adaptive Boosting)

w: vector of N instance weights
z: vector of M hypoth. weights

- $w_j \leftarrow 1/N \quad \forall_j$
- For m=1 to M do
  - $h_m \leftarrow$ learn(dataset,w)
  - err $\leftarrow$ 0
  - For each (x_j,y_j) in dataset do
    - If $h_m(x_j) \neq y_j$ then err $\leftarrow$ err + w_j
  - For each (x_j,y_j) in dataset do
    - If $h_m(x_j) = y_j$ then $w_j \leftarrow w_j$ err / (1-err)
  - w $\leftarrow$ normalize(w)
  - $z_m \leftarrow$ log [(1-err) / err]
- Return *weighted-majority(h,z)*

What can we boost?

- Weak learner: produces hypotheses at least as good as random classifier.
- Examples:
  - Rules of thumb
  - Decision stumps (decision trees of one node)
  - Perceptrons
  - Naïve Bayes models

Boosting Paradigm

- Advantages
  - No need to learn a perfect hypothesis
  - Can boost any weak learning algorithm
  - Boosting is very simple to program
  - Good generalization
- Paradigm shift
  - Don't try to learn a perfect hypothesis
  - Just learn simple rules of thumbs and boost them

Boosting Paradigm

- When we already have a bunch of hypotheses, boosting provides a principled approach to combine them
- Useful for
  - Sensor fusion
  - Combining experts

Boosting Applications

- Any supervised learning task
  - Spam filtering
  - Speech recognition/natural language processing
  - Data mining
  - Etc.