Ensemble methods

COMP 135 Intro to Machine Learning
Liping Liu

slides modified from Roni Khardon’s with permission

Weak and Strong Learning

• Suppose we have a learning algorithm that often gives reasonable but not necessarily great performance (e.g., accuracy $\geq 0.6$).

• Can we somehow use this algorithm to do better? How?

A motivating example

• Majority vote: Suppose we have 5 completely independent classifiers,
• if accuracy is 70% for each, then majority vote accuracy is
\[(0.7)^5 + 5 \times (0.7)^4 (0.3) + 10 \times (0.7)^3 (0.3)^2 = 83.7\%\]
• With 101 such classifiers - 99.9% majority vote accuracy

Forcing Classifier Diversity

• Can we force the hypotheses produced by different runs to be different (even when base classifiers is not sensitive)?
  - Yes, several methods

Some General and Specialized Alg

• Bagging:
  - use bootstrap sample
  - Bagging of Decision Trees

• Random Forests
  - Bagging
  - Random subset of features at each node

• AdaBoost
  - Iteratively working on hard samples

Forcing Classifier Diversity

• Method 1: Bootstrapping: create different training sets by randomly selection with replacement
  - BAgging
• Create ensembles by “Bootstrap Aggregating”: generate \( L \) training sets by bootstrapping

Algorithm:

For \( t = 1 \) to \( T \):
  Draw \( n \) items from \( X \) with replacement.
  Train a base learner \( f_t(x) \)

The final classifier is:

\[
f(x) = \frac{1}{T} \sum_{t=1}^{T} f_t(x)
\]

Images are taken from Adele Cutler’s slides

---

A classification example

• A classification problem
• Decision boundary of a single tree

• Average of predictions from 25 trees

• Decision boundary of bagging 25 trees

Why it works
• Decrease error by decreasing the variance in the results of unstable classifiers
  - Unstable algorithm: when small change in the training set causes a large difference in the base learners.
  - Can be applied to arbitrary base classifiers

Stability of Base Classifiers
• Which of these classifiers are stable/sensitive?
  - kNN
  - Decision Trees
  - Linear classifiers (SVM)
  - Naive Bayes

Forcing Classifier Diversity
• Method 2: Random trees
  - Random Forest algorithm
Random forest

- Train random trees
  - Fix the structure and depth
  - Randomly choose features for splits

Algorithm:

For $t = 1$ to $T$ (of trees):
- Create an independent bootstrap sample from the training set.
- Train a random tree,
  - for each node within a maximum depth:
    - Randomly select $m$ features from $d$ features
    - Find the best split on the selected $m$ variables, create a tree node
- Average the trees to get predictions for new data.

Forcing Classifier Diversity

- Method 3: reweight training instances
  - AdaBoost

Algorithm of AdaBoost

For $t = 1$ to $T$
- Train a weak classifier $h_t(x)$ using current weights $w_i(t)$ for all $i$, by minimizing the weighted classification error.
  - $\epsilon_t = \sum w_i(t) \times [y_i \neq h_t(x_i)]$
- Compute contribution for this classifier
  - $\beta_t = 0.5 \log \frac{1 - \epsilon_t}{\epsilon_t}$
- Update weights on training points,
  - $w_{i+1}(t) \propto w_i(t) \exp(-\beta_t y_i h_t(x_i))$
  - and normalize them such that $\sum w_{i+1}(t) = 1$
- Output the final classifier
  - $h(x) = \sum \beta_t \cdot h_t(x) > 0$

Weak learners

- How weak can a weak learner be?
  - As long as the base learner has error rate less than 0.5.

Example of AdaBoost

- A classification problem
  - 10 classes, not linearly separable
  - Uniform weight at initialization

Example of AdaBoost

- First iteration
  - First classifier: split on a single feature
  - 3 misclassified => error=0.3, beta=0.42

Images are from Mihaela van der Schaar’s
Example of AdaBoost

- Second iteration
  - 3 misclassified, error=0.21, beta=0.65

Example of AdaBoost

- Second iteration
  - 3 misclassified, error=0.14, beta=0.92

Example of AdaBoost

- Final classifier
  - All data points are correctly classified

AdaBoost

- Use the same training set over and over and thus need not to be large.
- Classifiers must be simple so they do not overfit.
- Can combine an arbitrary number of base learners. (parameters)

Gradient Tree Boosting

- Calculate gradient of scores for each data point
- Fit gradient values by trees
- Add up trees to minimize training loss

Gradient Tree Boosting

Check section 10.10 of ESML book for details
Other ensemble methods

- Stacking
  - Use classifier outputs as input of a super classifier
- Cascading
  - Postpone unsure instances to next classifier

Ensemble Methods

- Main idea: voting among diverse set of hypotheses can help reduce errors
- Different schemes to take advantage of and/or force diversity
- Bagging, Random Forests, Ada-Boost
- Many variants exist
- Other ways of combining classifiers are also possible