

**Recitation 11: Boosting, Notes**<sup>1</sup>

December 2, 2005

**1. AdaBoost Algorithm**

Given training examples  $(x_1, y_1), \dots, (x_m, y_m)$  such that  $x_i \in X, y_i \in Y = \{-1, +1\}$ .

Initialize  $D_1(i) = 1/m$ . ( $D_t(i)$  represents how much weight is given to example  $i$  on iteration  $t$ .)

For  $t = 1, \dots, T$ :

- (a) Train weak learner using distribution  $D_t$ : Outputs a weak classifier  $h_t : X \rightarrow Y$  ( $h_t$  can be an ID tree, a NN-based classifier, ...)
- (b) Compute the error  $\epsilon_t$  of the classifier  $h_t$ :  $\epsilon_t =$  sum of the weights of the data samples that  $h_t$  classifies incorrectly, or more mathematically,

$$\epsilon_t = \sum_{i: h_t(x_i) \neq y_i} D_t(i)$$

- (c) Use the error to compute  $\alpha_t \in R$ :

$$\alpha_t = \frac{1}{2} \ln \left( \frac{1 - \epsilon_t}{\epsilon_t} \right)$$

( $\alpha_t$  represents the weight on each classifier.)

- (d) Update

$$D_{t+1}(i) = \frac{D_t(i) \exp(-\alpha_t y_i h_t(x_i))}{Z_t}$$

where  $Z_t$  is a normalization factor (chosen so that  $D_{t+1}$  will be a distribution, that is, sum to 1)

Output the final classifier to be a weighted majority vote of the  $T$  base classifiers:

$$H(x) = \text{sign} \left( \sum_{t=1}^T \alpha_t h_t(x) \right)$$

Slogan: "T heads are better than 1."

**2. Important properties of Adaboost**

- Integrates disparate classifiers together (*i.e.*, combine classifiers that concentrate on different aspects of the problem or, in other words, put more weight to different data points)
- Theoretical bounds – adding a new classifier can't hurt (in terms of training error)
- Easy to program: can use *any* weak learner; Doesn't get stuck in local minima (in terms of minimizing training error)
- Sensitive to outliers, thus could overfit *in theory*, but not typical *in practice*.

---

<sup>1</sup>These notes were prepared in conjunction with Sourabh Niyogi. (Orig. date: Nov. 18, 2004; Last updated: Dec. 18, 2005)