Boosting

10-701/15-781, Recitation

April 15, 2010

Ni Lao

Today's Schedule

- Review of AdaBoost
  - Boosting as sequential optimization

- Preview of HW5 AdaBoost questions

The AdaBoost Algorithm

- Given N examples (x_i, y_i)

1. Initialize $w_i^1 = 1/N \ (i = 1, ..., N)$
2. For $t = 1, ..., T$,
   a. Learn a weak classifier $h_t(x)$ by minimizing the weighed error function J_t, where $J_t = \sum_{i=1}^N w_i^t I(h_t(x_i) \neq y_i)$;
   b. Compute the error rate for the learnt weak classifier $h_t(x)$: $\epsilon_t = \sum_{i=1}^N w_i^t \mathrm{I}(h_t(x_i) \neq y_i)$;
   c. Compute the weight for $h_t(x)$: $\alpha_t = \frac{1}{2} \ln \frac{1-\epsilon_t}{\epsilon_t}$;
   d. Update the weight for each example: $w_i^{t+1} = \frac{w_i^t \exp\{-\alpha_t y_i h_t(x_i)\}}{Z_t}$, where Z_t is the normalization factor for w_i^{t+1}: $Z_t = \sum_{i=1}^N w_i^t \exp\{-\alpha_t y_i h_t(x_i)\}$.
3. Output the final classifier: $H(x) = sign(f_T(x))$, where $f_T(x)$ is a linear combination of the weak classifiers, i.e., $f_t(x) = \sum_{m=1}^t \alpha_t h_m(x)$.

The Objective Function

- From class we know that the objective function is

$$E = \sum_{i=1}^{N} \exp\{-y_i f_T(x_i)\}$$

- But why is it so?
  - People invented AdaBoost much earlier than they discover its objective function …

Optimization in Functional Space

- Similar to gradient decent (in vector space), we can minimize an objective function in function space.

	Vector space	**Functional space**
Objective	$\min_x F(x)$ s.t. $x \in R^p$	$\min_f E(f)$ s.t. $f(x)$ is a function
Gradient	$\nabla_x F(x)$	$\nabla_{f(x)} E(f)$
Update	$x^t = x^{t-1} - \alpha^t \nabla_x F(x)$	$f^t(x) = f^{t-1}(x) - \alpha^t \nabla_{f(x)} E(f)$

Think a function f as
an infinite long vector

Optimization in Functional Space

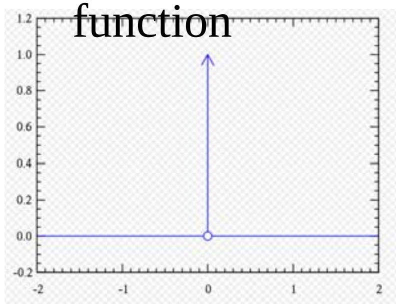
- Now we constrain that $f(x) = \sum_{m=1..T} \alpha^m h^m(x)$, where $h \in M$ comes from a family of functions (which is easy to represent)

	Vector space	**Functional space**
Objective	$\min_x F(x)$ s.t. $x \in R^p$	$\min_f E(f)$ s.t. $f(x)$ is a function
Gradient	$\nabla_x F(x)$	$h^t = \arg\max_{h \in M} \langle h, \nabla_{f(x)} E(f) \rangle$
Update	$x^t = x^{t-1} - \alpha^t \nabla_x F(x)$	$f^t(x) = f^{t-1}(x) - \alpha^t h^t(f)$

where $\langle h, g \rangle = \int h(x) g(x) dx$ is the dot product of two functions

AdaBoost

δ () is the Dirac delta function



$$\int \delta(x)dx = 1$$

$$\int \delta(x-y) f(x)dx = f(y)$$

- For AdaBoost the functional gradient is

$$\nabla_{f(x)} E(f) = -\sum_{i=1..N} \delta(x - x_i) y_i \exp(-y_i f(x_i))$$

- Its dot product with *h(x)* is

$$\left\langle h, \nabla_{f(x)} E(f) \right\rangle = -\sum_{i=1..N} y_i \exp(-y_i f(x_i)) h(x_i)$$

  – which is the objective function of a classifier with weighted samples
    - no matter what kind of base classifier we choose, we are still minimizing E()

  – In home work you will see that $w_i^t \propto exp\{-y_i f_{t-1}(x_i)\}$

  – You will prove that $\alpha_t = \frac{1}{2}\ln\frac{1-\epsilon_t}{\epsilon_t}$ is also minimizing E()

Sequential Optimization

Prove this is minimizing E(), assuming existing α and h() of are fixed

1. Initialize $w_i^1 = 1/N \ (i = 1, ..., N)$

2. For $t = 1, ..., T$,

   a. Learn a weak classifier $h_t(x)$ by minimizing the weighed error function J_t, where $J_t = \sum_{i=1}^{N} w_i^t I(h_t(x_i) \neq y_i)$;

   b. Compute the error rate for the learnt weak classifier $h_t(x)$: $\epsilon_t = \sum_{i=1}^{N} w_i^t \mathrm{I}(h_t(x_i) \neq y_i)$;

   c. Compute the weight for $h_t(x)$: $\alpha_t = \frac{1}{2} \ln \frac{1-\epsilon_t}{\epsilon_t}$;

   d. Update the weight for each example: $w_i^{t+1} = \frac{w_i^t \exp\{-\alpha_t y_i h_t(x_i)\}}{Z_t}$, where Z_t is the normalization factor for w_i^{t+1}: $Z_t = \sum_{i=1}^{N} w_i^t \exp\{-\alpha_t y_i h_t(x_i)\}$.

3. Output the final classifier: $H(x) = sign(f_T(x))$, where $f_T(x)$ is a linear combination of the weak classifiers, i.e., $f_t(x) = \sum_{m=1}^{t} \alpha_t h_m(x)$.

- Other HW questions
  - AdaBoost objective function 1

$$E = \sum_{i=1}^{N} \exp\{-y_i f_T(x_i)\}$$

  - AdaBoost objective function 2 (Extra credit)
    - the training error of AdaBoost, is upper bounded by

$$E = \sum_{i=1}^{N} \exp\{-y_i f_T(x_i)\}$$

  - Derive update equation for a different objective function

$$E = \sum_{i=1}^{N} (y_i - f_T(x_i))^2$$

  - Under stand the margin of classifiers
    - What does “margin” mean? Do logistic regression and Adaboost have margins?

Summary of AdaBoost

- Both of the following steps are minimizing the functional E()
  - Find h() by training a classifier with weighted samples
  - Setting $\alpha_t = \frac{1}{2}\ln\frac{1-\epsilon_t}{\epsilon_t}$

- The End
- Thanks