10-701/15-781, Machine Learning: Homework 5

Eric Xing, Tom Mitchell, Aarti Singh
Carnegie Mellon University
Updated on March 24, 2010

1 AdaBoost [Ni, 30 pt]

Given N examples (x_i, y_i), where y_i is the label and $y_i = +1$ or $y_i = -1$. Let $\mathrm{I}(\cdot)$ be the indicator function, which is 1 if the condition in () is true and 0 otherwise. In this exercise, we use the following version for AdaBoost algorithm:

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., $f_t(x) = \sum_{m=1}^t \alpha_m h_m(x)$.

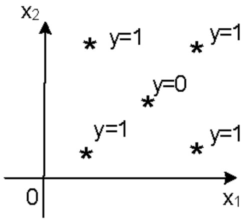
1.1 Sequential Optimization [5 pts]

In class, we learnt that AdaBoost tries to minimize the negative exponential loss: $E = \sum_{i=1}^N \exp\{-y_i f_T(x_i)\}$ sequentially. That is to say, at the t^{th} $(1 \leq t \leq T)$iteration, we want to choose appropriate weight α_t and the corresponding weak classifier $h_t(x)$ so that the overall loss E (accumulated up to t^{th} iteration) is minimized. It was proved that this strategy leads to the update rule: in the t^{th} iteration, $\alpha_t = \frac{1}{2}\ln\frac{1-\epsilon_t}{\epsilon_t}$.

Now, if we change the objective function to square loss $E = \sum_{i=1}^N (y_i - f_T(x_i))^2$ and we still want to optimize it sequentially. What is the new update rule for α_t?

1.2 [5 pts]

As shown in the figure below, we have five training samples with label 0.0 or 1.0.



Now let's assume that the base functions $h_t(x)$ are linear classifiers. Will the boosting algorithm (with square loss) always get zero training error after sufficient iterations? What is the minimum number of iterations before it can reach zero training error?

1.3 About Margin[5 pts]

Draw the objective functions of SVM, logistic regression, and Adaboost together. Assume we have a single training sample and a single feature.

hints: for AdaBoost assume that $h_t(x)$ is given, and the parameter is α_t.

1.4 [5 pts]

What does "margin" mean? Do logistic regression and Adaboost have margins?

1.5 Overfitting [5 pts]

There is an interesting applet written by Yoav Freund (`http://cseweb.ucsd.edu/~yfreund/adaboost/`). With it, you can create your own data set and train AdaBoost models.

Please design a dataset showing that AdaBoost does overfit. You can print a screen shot which includes both data points and error curves.

1.6 [5 pts]

Can you think of a strategy to prevent Boosting from overfitting?