Adaboost Implementation

David Torpey

May 20, 2017

Contents

1 Introduction

In order to perform the Adaboost (Adaptive Boosting) algorithm, we need to define two hyper-parameters:

1. M - number of models in the Adaboost cascade.
2. C_i - a base classifier to use in the Adaboost cascade.

Usually, C_i is some simple classifier such as a decision stump (a decision tree of depth one). An optimal M can be chosen through cross-validation.

2 Algorithm

Below is pseudocode for the Adaboost algorithm.

Algorithm 1 Adaboost

Initialize observation weights $D_t(i) = \frac{1}{m}, \forall i = 1, \ldots, m$.
for t = 1 to T **do**
  (1) Train weak learner h_t using distribution D_t (sample m samples from X using D_t.
  (2) Given above weak hypothesis $h_t : X \rightarrow \{-1, +1\}$, find $\epsilon_t = P_{i \sim D_t}[h_t(x_i) \neq y_i]$.
  (3) Calculate $\alpha_t = \frac{1}{2} ln(\frac{1-\epsilon_t}{\epsilon_t})$.
  Update the weight distribution for the samples: $D_{t+1}(i) = \frac{D_t(i) exp(-\alpha_t y_i h_t(x_i))}{Z_t}$, where Z_t is a normalization factor that ensures D_{t+1} is a distribution. This can be, for example, $Z_t = \sum_{i=1}^{m}(D_t(i))$.
end for

The final classifier/hypothesis will then be:

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