ENGINEERING DIVISION | NYU ABU DHABI

ENGR-UH 3332
Applied Machine Learning

AdaBoost for Classification

Due Date: Refer to NYU Class

Introduction

AdaBoost, short for Adaptive Boosting, is a machine learning meta-algorithm formulated by Yoav Freund and Robert Schapire, who won the 2003 Gödel Prize for their work. It can be used in conjunction with many other types of learning algorithms to improve performance. The output of the other learning algorithms ('weak learners') is combined into a weighted sum that represents the final output of the boosted classifier. AdaBoost is adaptive in the sense that subsequent weak learners are tweaked in favor of those instances misclassified by previous classifiers. An AdaBoost classifier is a meta-estimator that begins by fitting a classifier on the original dataset and then fits additional copies of the classifier on the same dataset but where the weights of incorrectly classified instances are adjusted such that subsequent classifiers focus more on difficult cases.

Dataset

Spambase: is a binary classification task and the objective is to classify email messages as being spam or not. To this end the dataset uses for seven text-based features to represent each email message. There are about 4600 instances.

Breast Cancer Dataset:
Features are computed from a digitized image of a fine needle aspirate (FNA) of a breast mass. They describe characteristics of the cell nuclei present in the image. The dataset is publicly available and was created by Dr. William H. Wolberg, physician at the University Of Wisconsin Hospital at Madison, Wisconsin, USA. The breast cancer dataset is a classic and very easy binary classification dataset.
Classes: 2
Samples per class: 212(M),357(B)
Samples total: 569 Dimensionality: 30

Process

AdaBoost (adaptive boosting) uses boosting approach to combine multiple weak classifiers into one strong classifier.

Error

$$\epsilon_t = \frac{missclassifications}{samples} = \frac{missclassifications}{N} \text{ (in the first iteration)}$$

$$\epsilon_t = \sum_{\text{miss}} \text{weights}$$

if error > 0.5, just flip the decision and the error = 1 - error

Weights

$$\omega_o = \frac{1}{N} \quad \text{for each sample}$$

$$\omega = \frac{\omega - e^{(-\alpha.y.f(X))}}{\text{sum}(\omega)}$$

where f(x) = prediction of t

Performance

$$\alpha = 0.5.\log\left(\frac{1-\epsilon_t}{\epsilon_t}\right)$$

Prediction

$$y = \text{sign}\left(\sum_{t}^{T}(\alpha_t.f(X))\right)$$

Algorithm

Initialize weights for each sample = $\frac{1}{N}$

for t in T:

- Train week classifier (search for best feature and threshold)
- Calculate error $\epsilon_t = \sum_{\text{miss}} \text{weights}$
    - flip error and decision if error > 0.5
- Calculate $\alpha = 0.5.\log\left(\frac{1-\epsilon_t}{\epsilon_t}\right)$
- Update weights: $\omega = \frac{\omega.e^{(-\alpha.f(X))}}{Z}$

Create an AdaBoost ensemble classifier based on the algorithm above to make predictions for the binary classification problem for two data sets mentioned above

Deliverables

A zip file containing the following:

1. a working project (source code, make files if needed, etc)
2. a report for the detailed description of the project
    - a. Instructions on how to run your project
    - b. Answers to the programming questions.

Before submitting your project, please make sure to test your program on the given dataset.

Notes

You may discuss the general concepts in this project with other students, but you must implement the program on your own. ***No sharing of code or report is allowed.*** *Violation of this policy can result in a grade penalty.*

Late submission is acceptable with the following penalty policy:
10 points deduction for every day after the deadline