James Hahn
CS1675 Machine Learning
Dr. Adriana Kovashka

Discussion of results

With the Pima Indians dataset, origin labels were 0 and 1. After preprocessing, they were converted to -1 and 1. Additionally, 10-fold cross-validation was used to train our adaboost algorithm. The algorithm was run with 10 iterations, 20 iterations, and 50 iterations.

This specific implementation of the adaboost algorithm breaks the loop when an alpha value less then zero is computed. After training the adaboost algorithm with decision stumps as the weak classifiers, the following results were achieved:

Iterations	Accuracy
10	.7528
20	.7601
50	.7667

The accuracies are an average over five simulations.

As one can see, the number of iterations largely does not impact the boosting algorithm's accuracy. There is a slight trend, but it is negligible and is not worth the extra computational time to attempt an increase in accuracy.

Adaboost Extra

For 5 points of extra credit, another version of the adaboost algorithm, implemented in adaboost_extra.m, never terminates the loop, even if the alpha value is negative. Instead, by using a negative alpha value in the final linear combination of the weak classifiers, the algorithm is inverting that algorithm's prediction. Results were computed below for 10, 20, and 50 iterations with this new implementation.

Iterations	Accuracy
10	.7534
20	.7628
50	.7649

Clearly, the same general trend in accuracy follows with the number of iterations. The accuracies are averages across five simulations. On any given simulation, the accuracies for adaboost_extra.m are the same as the accuracies for adaboost.m. The only reason the above computed accuracies are different is because the dataset is randomized on any given simulation. There is no notable difference in the algorithms. One explanation for these similar accuracies is because adaboost.m never in fact has a negative alpha value, so the loop never terminates, and adaboost_extra.m never computes a negative alpha value, so it performs identically to adaboost.m.

Bishop Exercise 1.3

Suppose that we have three coloured boxes r (red), b (blue), and g (green). Box r contains 3 apples, 4 oranges, and 3 limes, box b contains 1 apple, 1 orange, and 0 limes, and box g contains 3 apples, 3 oranges, and 4 limes. If a box is chosen at random with probabilities p(r)=0.2, p(b)=0.2, p(g)=0.6, and a piece of fruit is removed from the box (with equal probability of selecting any of the items in the box), then what is the probability of selecting an apple? If we observe that the selected fruit is in fact an orange, what is the probability that it came from the green box?

P(apple) = P(apple | r)P(r) + P(apple | b)P(b) + P(apple | g)P(g)
= 3/10 * 2/10 + ½ * 2/10 + 3/10 * 6/10
= 6/100 + 10/100 + 18/100
= 34/100
= **34%**

P(g | orange) = P(orange | g)P(g) / [P(orange | g)P(g) + P(orange | r)P(r) + P(orange | b)P(b)]
= [3/10 * 6/10] / [3/10 * 6/10 + 4/10 * 2/10 + ½ * 2/10]
= [18/100] / [18/100 + 8/100 + 10/100]
= 18 / (18 + 8 + 10)
= 18 / 36
= ½
= **50%**