PS3 solution

1.1 [15 points]

Problem 1: VC Dimension

1. Consider a binary classification problem for data points in $\mathbb{R}^3$ with a hypothesis space consisting of axis aligned 3-D boxes such that any point in the box is labeled with a + and any point outside the box is labeled with a −. What is the VC dimension of this hypothesis space? Prove it. How many samples would be sufficient to guarantee that an optimal learning algorithm will attain an accuracy of .8 with probability at least .95? Can you generalize your argument to axis aligned boxes in $\mathbb{R}^d$?

Solution: the VC dimension of this hypothesis space is 6.

Let h be such a hypothesis, i.e. h is an axis aligned 3-D box. As we know for such a box, there are 6 faces. Moreover, let us make this box with length, weight and height are all equal to 3 units. (This is not necessary, but easy to explain.) Let us pick the center from each face of h, there are 6 of them, as my sample points. I am going to show that this 6 points can be shattered with any labeling.

For any point from my sample set, if the point has a + label, we can move the face that containing this point 1 unit away from the center of h; while if the point has a - label, we can move the face that containing this point 1 unit towards to the center of h. In either case, this point is correctly classified by the updated hypothesis. In addition, we can do this update on the hypothesis with respect to each point separately, without affecting the classification of any other point. Thus, the VC dimension of this hypothesis space is at least 6.

Then, I have to show that the VC dimension of this hypothesis space is at most 6, i.e. any sample set that includes 7 or more points can not be shattered by this hypothesis space. Let us construct a sample set that includes 7 points, one can always find the extrema values (max and min) of the 3 coordinates (if not, it means at least 3 points are contained in the same coordinate-plane-paralleled plane and clearly this sample set can not be shattered), and these

extrema values (6 of them) will generate a 3-D box with each face contains 1 extrema point, and there will be 1 point left over. It is obvious that if the left over point lies on the boundary of the 3-D box, this sample set can not be shattered. Thus the left over point has to be inside of the 3-D box. If this point has a - label and all other points have a + label, then this sample set with this labeling will not able to be shattered. Thus, The VC dimension is less than 7. In conclusion, The VC dimension of this hypothesis space is 6.

To find how many samples would be sufficient to guarantee that an optimal learning algorithm will attain an accuracy of .8 with probability at least .95, we need the following formula:

$$M \geq \frac{1}{\epsilon}(4 \ln \frac{2}{\delta} + 8 \cdot VC(H) \ln \frac{13}{\epsilon})$$

where $\epsilon = 1 - 0.8 = 0.2$, $\delta = 1 - 0.95 = 0.05$ and $VC(H) = 6$. Thus:

$$M \geq \frac{1}{0.2}(4 \ln \frac{2}{0.05} + 8 \times 6 \ln \frac{13}{0.2}) \approx 1076$$

For the general axis aligned boxes in $\mathbb{R}^d$, the VC dimension will be $2d$. Since for any dimension, there are 2 extrema (one max and one min), and a sample set that has one point on each extrema (there are $2d$ of them) will able to be shattered. In addition, any more point will not able to be shattered.

1.2 [15 points]

[You should discuss the affect of different alpha1 and alpha2 on the VC dimension]

Case 1: alpha1 ≠ alpha2:

It's easy to see that the final hypothesis space is one rectangle. As we know, the VC dimension set of rectangles in 2D is 4 (assuming the points inside the rectangle is Positive). In this question, there is one difference: the points inside the rectangle could be "+" or "-". It's easy to see that the VC dimension is 5 in this case.

Case 2: alpha1 = alpha2 (and the two rectangles are different):

In this case, there will appear this circumstance: H' = sign(0)=0. You need to assume the labels of points at this time. For example, you assume the labels of points are "+" when sign(0)=0. In this case, there will have two different rectangles. As we know, the VC dimension set of rectangles in 2D is 4 (assuming the points inside the rectangle is Positive). So the final VC is 8.

[welcome to discuss with Prof and me if you have different opinion.]

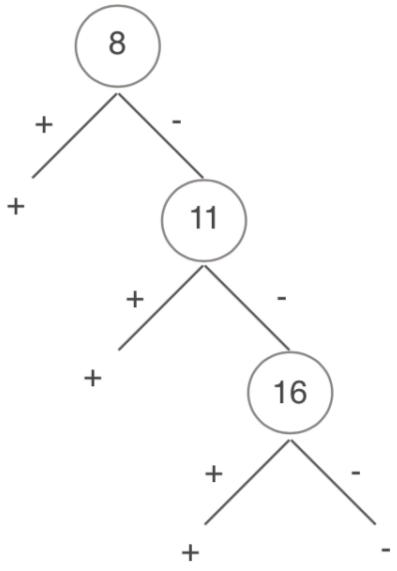
For Problem 2, each question is worth 10 points.

Problem 2: Medical Diagnostics

1. Suppose that the hypothesis space consists of all decision trees with exactly three attribute splits (repetition along the same path is allowed) for this data set.
   (a) Run the adaBoost algorithm for five rounds to train a classifier for this data set. Draw the 5 selected trees in the order that they occur and report the ϵ and α, generated by adaBoost, for each.

Solution:

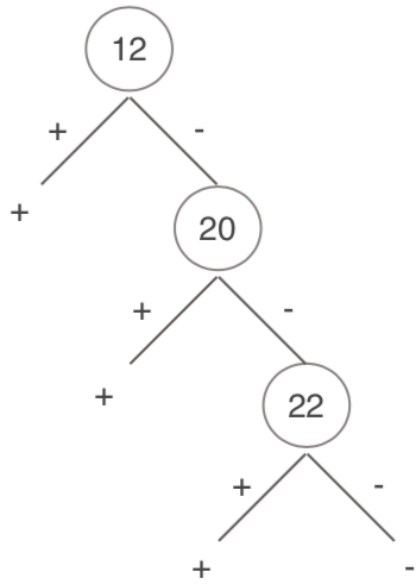
round 1:



$\epsilon_1 = 0.1875$

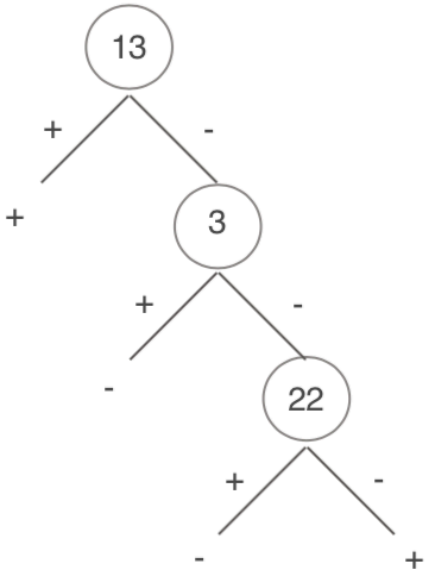
$\alpha_1 = 0.7331685343967135$

round 2:



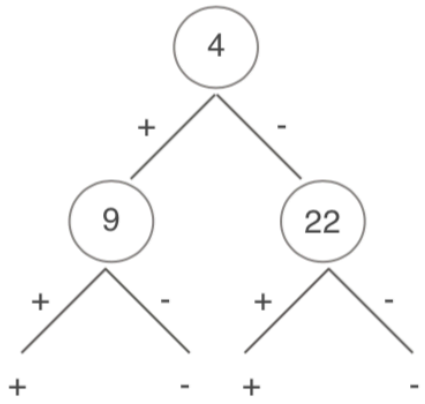
$\epsilon_2 = 0.2692307692307692$
$\alpha_2 = 0.4992644150555637$

round 3:



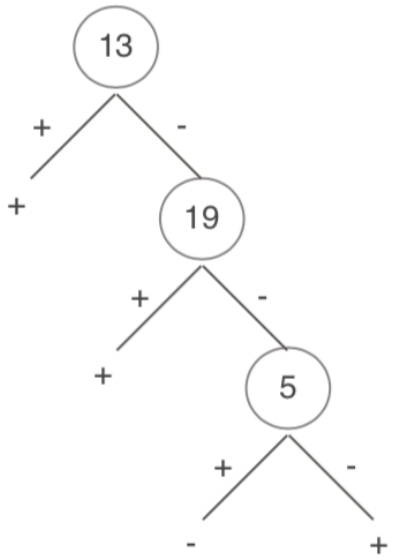
$\epsilon_3 = 0.3403508771929824$
$\alpha_3 = 0.3308654921632833$

round 4:



$\epsilon_4 = 0.3371960486322189$
$\alpha_4 = 0.3379073696386288$

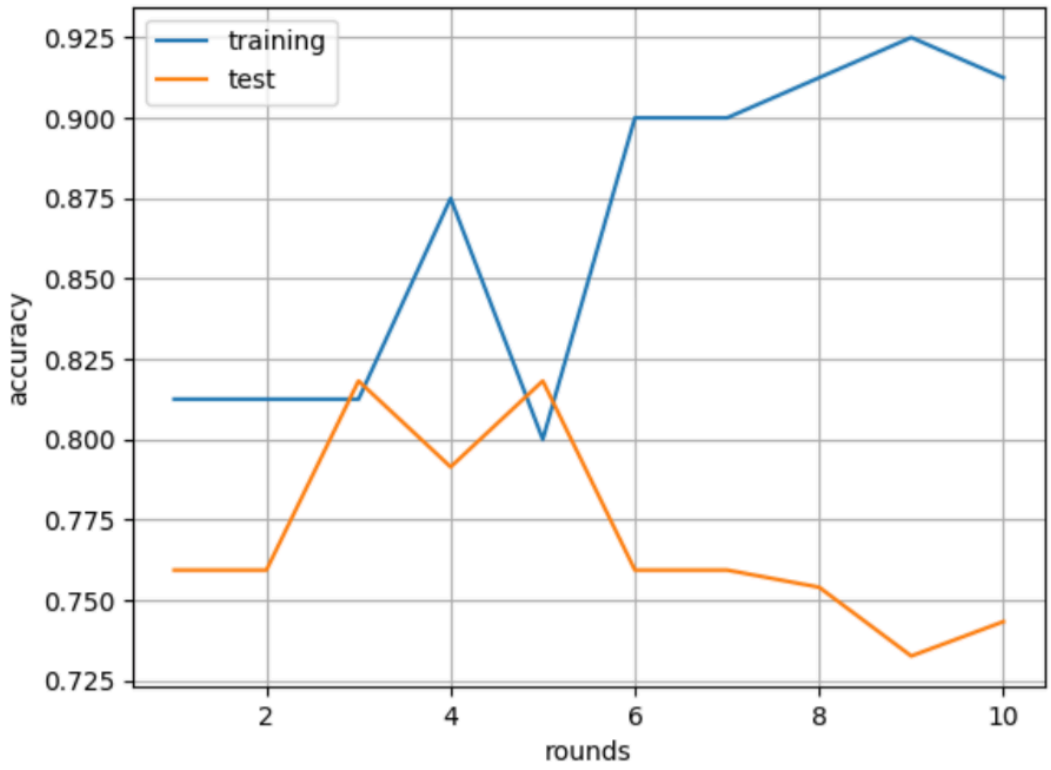
round 5:



$\epsilon_5 = 0.36971122376230825$
$\alpha_5 = 0.2667279323771729$

(b) Run the adaBoost algorithm for 10 rounds of boosting. Plot the accuracy on the training and test sets versus iteration number.

Solution:



2.2

a:

Solution: the optimal value of α is:

α = [0.28454133984432933, -0.077861227403110342, -2.1024310304417342, 0.66127884687126859, -0.11169538803919137, -0.06821395570844857, 0.45358982813308069, 1.6851097385652365, -0.82467336073930186, -0.13062102917136895, 0.16922076173905315, -0.28247673167366472, 0.84987022894178288, 0.45433129369885644, -0.77072565643270607, 0.65501444331293057, 0.42786931586013915, -1.0225327067327634, 0.1790268934074708, 0.15333755098019086, 0.17363181672150022, 0.41911520416690456]

The corresponding value of the exponential loss on the training set is: 52.15919847543013

b:

Solution: the accuracy of the resulting classifier on the test data is: 0.5721925133689839

C:

Solution:
The accuracy on training: 0.8125
The accuracy on test set: 0.6684491978609626
α = [0.48470027859405174, 0.23296515192904987, 0.30841915161660366, 0.26905996040337654, 0.12707489951194711, 0.19391360519349748, 0.166465819734366, 0.20484093019666055, 0.1123378085421135, 0.16915310437537498, 0.12721184797605212, 0.15386377069176324, 0.098070542088111307, 0.15796706354327852, 0.13458519734012858, 0.13035666473731969, 0.10144574639261549, 0.12621641283923671, 0.10744964087976935, 0.11081674484994215]

The α learned by adaBoost are the weights for the best classifier that was found in each round of boosting, these classifiers do not have the same order with the hypothesis space (if we order the hypothesis space by the order of attributes in the data set) and it allows repetition, and the number of α's is the number of the rounds we ran the boosting; while the α! learned by coordinate descent are the weights for each classifier in the hypothesis space with the same order, and the number of α's is the number of hypothesis we have in the hypothesis space.

One can tell the α learned by adaBoost contains some negative values, which means the final hypothesis allows dis-trust for some of the classifiers; while the !α leaned by coordinate descent are all positive values.

d:

Solution: the accuracy on test set is 0.5882352941176471 (This result is different each time.) The accuracy on the test set using bagging is lower than adaBoost but slightly higher than coordinate descent.

e:

Solution: In conclusion, adaBoost should be preferred for this data set. In terms of accuracy on the test set, adaBoost has the highest accuracy so it should be preferred.

Although coordinate descent can be made to perform similar as adaBoost, but the standard coordinate descent does not count the weights on an individual data point. Thus adaBoost should perform better than the standard coordinate descent.

Bagging is mainly for lowering the variance for some very expressive hypothesis spaces. However, our hypothesis space contains only height 1 decision trees, which have a very high bias and low variance. Thus, use bagging in this hypothesis space is not a good ides.