

# Introduction to Machine Learning

## Problem Set: Decision Trees, Ensembles, Support Vector Classifier

Summer 2021

1. For this question, you will use the Tennis dataset presented in the lecture:

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

- (a) In the lecture, we showed how the first node in a decision tree classifier would be built. Continue this procedure, and build the rest of the tree, using information gain as the criterion. (Show your computations at each stage, as in the lecture handout; don't just use a `DecisionTreeClassifier` from `sklearn`.) Stop building your tree when it achieves zero classification error on the training set.
- (b) Next, you will perform weakest-link pruning on the tree you built in (a). Show your computations and the optimal subtree at each stage.

2. (*AdaBoost notebook.*) This question is about the *Demo: AdaBoost Classifier* notebook we used in class. Review that notebook, and try running it yourself, before answering the questions.

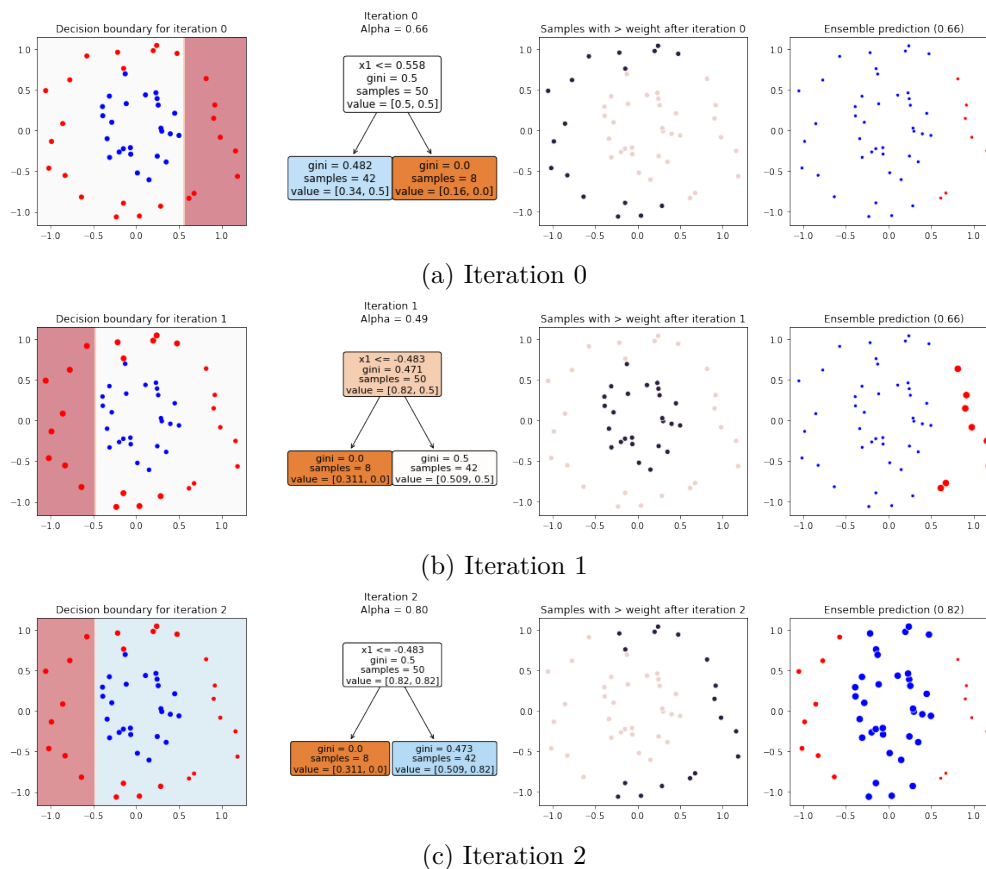


Figure 1: First three iterations of an AdaBoost classifier

- (a) Figure 1a and Figure 1b show iteration 0 and 1 of an AdaBoost classifier. In both iterations, the predicted class labels of the ensemble are the same. However, in the second iteration, the ensemble output is more *confident* about eight samples in the red class. Why is the ensemble more confident about the class of those eight samples after two iterations? Why is the ensemble not as confident about any of the other samples after two iterations?
- (b) Figure 1b and Figure 1c show iteration 1 and iteration 2 of an AdaBoost classifier. In both iterations, the decision stump splits on the same feature, at the same cutpoint. In iteration 1, the red class is predicted in both regions. In iteration 2, however, the blue class is predicted in the region on the right. Why are the predictions different, even though the boundary is the same? Explain what changed between iteration 1 and iteration 2, and why.

3. (Based on a problem by CMU/M. Kolar.) This question should be completed by hand; no programming is involved.

Suppose we are given a dataset of feature-label pairs in  $\mathbb{R}^1$ :

$$(-1, -1), (0, -1), (1, -1), (-3, +1), (-2, +1), (3, +1)$$

(The first value in each tuple is the feature  $x$ , the second value is the label  $y$ .)

- Plot the data in  $\mathbb{R}^1$ . Is the data linearly separable in  $\mathbb{R}^1$ ?
- Using the basis function  $\phi(x) = (x, x^2)$ , the points in  $\mathbb{R}^1$  can be transformed to points in  $\mathbb{R}^2$ . Apply  $\phi(x)$  to the data and plot the transformed data in  $\mathbb{R}^2$ . Is the data linearly separable in  $\mathbb{R}^2$ ? For the rest of this problem, use the transformed data.
- Using geometric intuition (i.e. looking at the plot), identify two points  $x_+$  and  $x_-$  (from the positive class and negative class, respectively) that should be support vectors of a maximal margin classifier for this data in  $\mathbb{R}^2$ . Circle them, and explain why you selected these points.
- Construct the maximal margin classifier using the following geometric procedure: Draw a line segment  $l$  connecting the two support vectors. Mark the midpoint of this line; the separating hyperplane should pass through this point (why?). Then, draw the separating hyperplane as a solid line perpendicular to  $l$ . Finally, draw two dashed lines parallel to the separating hyperplane and passing through the support vectors, showing the margin.
- What is an equation of the line (in the form  $w_0 + w_1x_1 + w_2x_2 = 0$ ) that defines the separating hyperplane you drew?
- If we add another training sample at  $x = 5, y = +1$ , would the hyperplane change? Would the margin change? Explain.
- To find  $\alpha_i, i = 1, 2$  for the maximal margin classifier, we will solve the dual problem

$$\begin{aligned} \underset{\alpha}{\text{maximize}} \quad & \sum_i \alpha_i - \frac{1}{2} \sum_i \sum_j \alpha_i \alpha_j y_i y_j \phi(x_i)^T \phi(x_j) \\ \text{s.t.} \quad & \sum_i \alpha_i y_i = 0 \\ & \alpha_i \geq 0 \end{aligned}$$

(note: this maximal margin classifier problem is slightly different from the support vector classifier problem! There is no slack variable for the maximal margin classifier.)

We know that  $\alpha_i = 0$  for any point that is not a support vector. So, write out the expression

$$\sum_i \alpha_i - \frac{1}{2} \sum_i \sum_j \alpha_i \alpha_j y_i y_j \phi(x_i)^T \phi(x_j)$$

in terms of  $\alpha_1, \alpha_2$ , the support vectors  $x_+$  and  $x_-$ , and their labels  $y_+$  and  $y_-$ . (Note that  $x_+$  and  $x_-$  are vectors in the transformed data. The others are scalars.) Then substitute the values of  $x_+$  and  $x_-$ ,  $y_+$  and  $y_-$ , and simplify, so that the only unknowns are  $\alpha_1, \alpha_2$ .

- (h) Note that for our example, since there is one support vector in the positive class and one support vector in the negative class, the first constraint of the dual problem described above implies that  $\alpha_1 = \alpha_2$ . In your expression for the previous part, substitute  $\alpha_1 = \alpha$  and  $\alpha_2 = \alpha$ , and simplify.
- (i) Take the derivative of the expression from part (h) with respect to  $\alpha$ , and set it equal to zero. Solve to find the value of  $\alpha_1 = \alpha_2 = \alpha$ . Also find the values of the coefficients  $w_j = \sum_{i \in S} \alpha_i y_i x_{ij}, j = 1, 2$ .
- (j) To find  $w_0$ , use the fact that for points on the margin,  $y_i = w_0 + \sum_{j=1}^p w_j x_{ij}$ . Plug in values from either support vector to solve for  $w_0$ .
- (k) Show that for the values of  $w_0, w_1$ , and  $w_2$  you computed above, the separating hyperplane  $w_0 + w_1 x_1 + w_2 x_2 = 0$  is the same as the hyperplane you found using the geometric approach in part (e).
4. (*AdaBoost*. By Eric Xing, Ziv Bar-Joseph at CMU.) We are interested in building an ensemble of decision stumps (trees with only one split) with the AdaBoost algorithm. The labeled training data and the first decision stump are illustrated above.

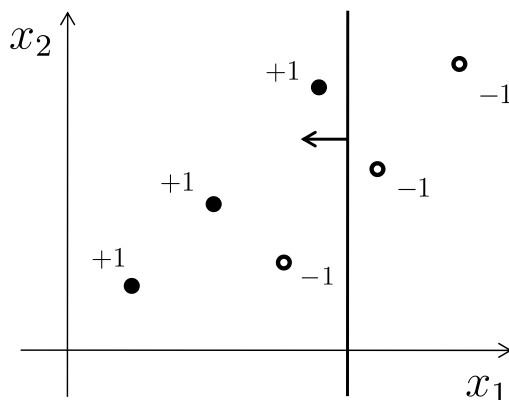


Figure 2: Labeled training data and decision boundary of first decision stump for AdaBoost. The arrow points toward the positive decision region, i.e. every point on the side of the line with the arrow is classified as positive.

For the following questions, you should be able to answer without any computations. Explain each answer.

- (a) Which training point(s) would have their weight increase for the next boosting iteration?
- (b) Draw the decision boundary of a stump that could be learned in the next boosting iteration.
- (c) Which of these two stumps will have a higher coefficient in the ensemble? In other words, will  $\alpha_2 > \alpha_1$ , or vice versa?