

CS 440: Introduction to AI

Homework 3

Due: Tuesday November 17th

Your answers must be concise and clear. Explain sufficiently that we can easily determine what you understand. We will give more points for a brief interesting discussion with no answer than for a bluffing answer.

Solutions will be posted no sooner than two days after the due date. Homework will be accepted until that point with a penalty of 10% per day that it is late. No assignments will be accepted after the solutions have been posted. Late homework will only be accepted in class, during office hours, or electronically by email to the TA.

You are expected to do each homework on your own. You may discuss concepts with your classmates, but there must be no interactions about solutions. You may consult the web but the work handed in must be done on your own.

The penalty for cheating on any assignment is straightforward. On the first occurrence, you will receive a zero for the assignment and your course grade will be reduced by one full letter grade. A second occurrence will result in course failure.

1) Determine the VC dimension of each of the hypothesis spaces below.

Ex: A line in a 2-dimensional plane (a 2d perceptron).

VC dimension: 3

a) Lines in a 2-dimensional plane, with the restriction that the line should cross the origin (a 2d perceptron with $w_0 = 0$).

3

b) Spaces inside or outside an arbitrary interval on the real line, with the points inside as + and the points outside as -

2

c) Spaces inside or outside an arbitrary interval on the real line, with the ability to choose whether the points inside the interval are classified as + or -

3

d) Spaces outside or between two parallel planes in 3d space, with the ability to choose whether the space between the line is classified as + or -

7

e) (Extra credit) Spaces inside or outside a (possibly irregular) tetragon in the plane, with the ability to choose whether the points inside are classified as + or -

9

f) (Extra credit) Spaces inside or outside an ellipsoid in the plane. You can choose whether the points inside are classified as + or -.

5

g) (Extra credit) A cosine with arbitrary frequency f , where if x is classified as + if $\cos(x\pi f) = 0$ and x is classified as - otherwise.

Infinity

2) On lecture 19, slides 9 and 10 we discussed using a cubic polynomial kernel to try to distinguish hand-written numbers in a 32×32 grayscale matrix. We concluded that using a cubic polynomial would mean that instead of considering only single pixels, we would now consider all triples of pixels as features.

We could use a polynomial kernel with degree 20 instead of a cubic polynomial kernel, and that would mean considering the correlation among all sets of 20 pixels as features.

Suppose now that we have two options: a polynomial kernel with degree 5, K_5 and a polynomial kernel with degree 80, K_{80} .

What would be one possible advantage of using K_{80} instead of K_5 . What would be the main problem of using K_{80} ? Explain your answer.

The advantage of using K_{80} is that it is more likely that we will be able to separate the points with a hyperplane in the high dimension space. The disadvantage is that since K_{80} is more expressive, we would need much more examples in order not to overfit.

3) Suppose we have an n dimensional classification problem and we are going to use an n dimensional perceptron to define a decision surface. What is the minimum number of examples needed in the training phase before evidence begins to accumulate that the perceptron will perform better than random on test examples.

A n -dimension classification problem requires n -dimensional perceptron. As we know, the VC dimension of the n -dimensional perceptron is $n+1$. Therefore, we would need to see $n+1$ examples before we can start accumulating confidence on our classification.

4) Suppose I have the following learning algorithm:

1) Start with a MLP (multi-layer perceptron) with 3 neurons in the first layer and one neuron in the second layer. Throughout the problem, consider that

all first layer outputs are input to the single second layer neuron. The training set is composed of 5000 correctly labeled examples.

2) Train the MLP by repeatedly cycling through the training set in the order it is given. Stop training when the accuracy changes by less than 0.5% over a complete pass through the training set.

3) If the resulting accuracy of the MLP over the training set is less than 100%

3.1) Add one neuron to the first layer in the MLP

3.2) Go to step 2.

4) Return the resulting MLP.

Answer and explain:

o) Does this learning algorithm always halt?

Yes since an MLP with arbitrary number of neurons is guaranteed to classify anything. The only exception is if we have two points in the same position with contrary labels (you don't need to add this remark in order to get full credit and this case is ignored for all other answers).

i) How well will we expect a resulting classifier to perform on new testing examples?

The algorithm has unbounded VC dimension and therefore we have absolutely no confidence that it will perform well on new examples.

ii) What is the VC dimension of the hypothesis space entertained by this algorithm?

It is unbounded. We can provide as many examples as we want, with as complex a labeling as we want, and the algorithm still will be able to classify the data.

iii) Suppose we stop when the training accuracy reaches 90%. How would your answer to part (i) change?

At any moment in the algorithm we will have an MLP structure. This MLP has a defined number of neurons, and therefore defined and bounded VC dimension. If this VC dimension exceeds the number of examples we would be guaranteed to achieve 100% training set accuracy. Therefore, if we stop with 90% the VC dimension of the final MLP is guaranteed to be less than the number of examples and consequently we have some confidence that the training is meaningful. (Maybe little, but still some).

iv) Suppose we train 10 MLPs as in part (iii) but with each trained on a different random permutation of the original training data. Will the resulting MLPs all have the same number of units? What can we say about their weight vectors?

They will not necessary have the same number of units or the same weight vectors.

v) We use bagging to combine the 10 MLPs of part (iv). What can we expect about its performance compared to the constituent MLPs?

We expect the ensemble to perform better than each constituent MLP.