

Jericho McLeod
CSI-873
Homework 8

Problem 7.1: Consider training a two-input perceptron. Give an upper bound on the number of training examples sufficient to assure with 90% confidence that the learned perceptron will have true error of at most 5%. Does this bound seem realistic?

The VC dimension of a perceptron is $r + 1$. Two mechanisms for calculating upper bounds are:

$$m \geq \frac{1}{2\epsilon^2}(\ln(\frac{1}{\delta}) + \ln |H|)$$

and

$$m \geq \frac{1}{\epsilon}(4 \log_2(\frac{2}{\delta}) + 8VC(H) \log_2(13/\epsilon))$$

Since the hypothesis space is not described, I will use the later since so that I can substitute $(r + 1)$ for $VC(H)$.

In this case, δ is the probability of failure, and ϵ is the error.

$$m \geq \frac{1}{.05}(4 \log_2(\frac{2}{0.10}) + 8(3) \log_2(13/.05))$$

$$m \geq 20(4 \log_2(20) + 8(3) \log_2(260))$$

```
In [3]: import math
m_less = 20 * ( 4 * math.log(20,2) + 24 * math.log(260,2))
print(m_less)
```

4196.490797844647

The final result, 4,197 training examples, seems excessive to train a 2 input perceptron. However, what we are really showing is that for a finite hypothesis space where the target concept is present, the perceptron will always be trained to our parameters after this many training examples. We are making no statements as to whether a particular concept can be learned in fewer training examples, though in reality many may be learnable in with fewer examples.

Problem 7.2: Consider the class C of concepts of the form $(a \leq x \leq b) \wedge (c \leq y \leq d)$, where a, b, c , and d are integers in the range $(0,99)$. Note that each concept in this class corresponds to a rectangle with integer-valued boundaries on a portion of the x, y plane. Hint: Given a region in the plane bounded by points $(0, 0)$ and $(n - 1, n - 1)$, the number of distinct rectangles with integer valued boundaries within this region is $(\frac{n(n+1)}{2})^2$.

Part A: Give an upper bound on the number of randomly drawn training examples sufficient to assure that for any target concept c in C , any consistent learner using $H = C$ will, with probability 95%, output a hypothesis with error at most 0.15.

Given

$$m \geq \frac{1}{2\epsilon^2} (\ln(\frac{1}{\delta}) + \ln |H|)$$

and the hint, $(\frac{n(n+1)}{2})^2$,

$$H = C = (\frac{100(100+1)}{2})^2$$

Thus;

```
In [13]: (1/(2*0.15**2)) * (math.log(1/0.05) + math.log((100*101/2)**2))
```

```
Out[13]: 445.55598484650665
```

Any consistent learning will learn the target concept with probability .95 and error at most 0.15 within 446 training examples.

Part B: Now suppose the rectangle boundaries a , b , c , and d take on *real* values instead of integer values. Update your answer to the first part of this question.

The VC Dimension for a Hypothesis in rectangular form can shatter at most 4 points, but not 5, logically. With $VC(4)$, $m \geq \frac{1}{\epsilon} (4 \log_2(\frac{2}{\delta}) + 8(4) \log_2(13/\epsilon))$ can be used to find the upper bound.

```
In [14]: (1/0.15) * (4 * math.log(2/0.05,2) + 8*4*math.log(13/0.15,2))
```

```
Out[14]: 1515.2312158225536
```

We need at least 1,516 training examples to reach the same point if the rectangles use real numbers instead of integers.

Problem 7.5: Consider the instances X corresponding to all points in the x, y plane. Give the VC dimension of the following hypothesis spaces:

As a note, I solved these with a whiteboard analytically. I would be happy to demonstrate these, but the collection of drawings does not lend itself to a jupyter notebook.

Part A: H_r = the set of all rectangles in the x, y plane: 4. We have already determined this for 7.2 part B.

Part B: H_c = the set of all circles in the x, y plane: 3. This is similar to what we solved in class with ellipses, but circles are more constrained and have a lower VC. Once 4 points are shown in an arrangement that resembles the XOR layout for perceptrons, circles are unable to shatter the space.

Part C: H_t = the set of all triangles in the x, y plane: I believe this is 7. I was able to shatter up to 7, but with 8 points, if you alternate classes of points on a circle and try to classify one class while excluding the other it cannot be done. Conversely, no classification or inverse of the class for 7 points fails.