Machine Learning from Data HW 5

Shane O'Brien

October 2017

Exercise 2.8

a

The definition of $\bar{g}$ is:

$\bar{g}(x) \approx \frac{1}{K}\Sigma_{k=1}^{K} g_k(x)$

This averaging definition is, by definition, linear.

b

Most binary classifications won't have $\bar{g}$ in the $\mathcal{H}$. The $\bar{g}$ won't be a -1 or $+1$ unless $\mathcal{H}$ labels all the points the same each time, which would be a pointless $\mathcal{H}$.

c

No, $\bar{g}$ won't be a binary function. $\bar{g}$ will most likely land somewhere between -1 and 1.

Problem 2.14

a

We want to show that $d_{vc}(\mathcal{H}) < K(d_{vc} + 1)$

Let's define a point $k^* = d_{vc} + 1$. We know $d_{vc} < k^*$

We also know that $d_{vc}(\mathcal{H})$ is at most K times our previously defined k^*, so:

$d_{vc}(\mathcal{H}) < Kk^*$

Which means...

$d_{vc}(\mathcal{H}) < K(d_{vc} + 1)$

b

From part a, we know that for some number ℓ:

$m_{\mathcal{H}}(\ell) \leq \Sigma_K m_{\mathcal{H}_k}(\ell)$

$m_{\mathcal{H}}(\ell) \leq \Sigma_K (\ell^{d_{vc}} + 1)$

We can now do some mathematical manipulation:

$m_{\mathcal{H}}(\ell) \leq \Sigma_K (\ell^{d_{vc}} + 1)$

$m_{\mathcal{H}}(\ell) \leq K\ell^{d_{vc}} + K$

$m_{\mathcal{H}}(\ell) \leq 2K\ell^{d_{vc}}$

So our results is:

$m_{\mathcal{H}}(\ell) \leq 2K\ell^{d_{vc}} < 2^{\ell}$

c

I started by getting the derivative of our problem

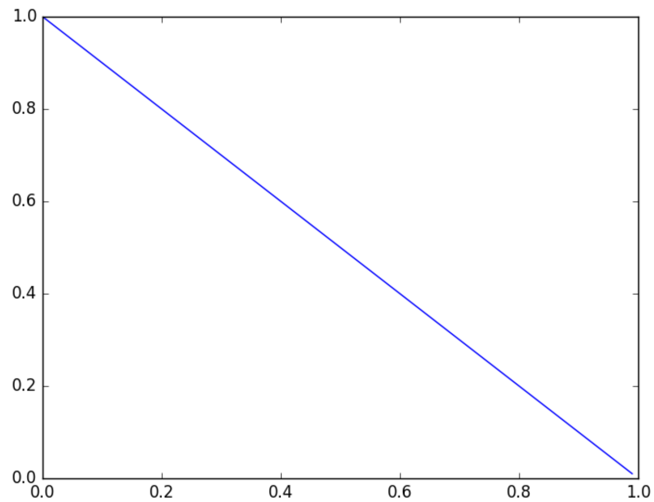
$K(d_{vc} + 1), 7(d_{vc} + K) log_2(d_{vc}K))$

and then setting it equal to zero. However, when I do this, I get a very large problem that I can't set to 0 easily. I was stuck here and didn't know how to continue.

Problem 2.15

a

A simple example of a monotonic classifier is a linear perceptron that only goes down and to the right. The image below shows this:



The upper right quadrant is +1, while the bottom left quadrant is -1.

b

$m_{\mathcal{H}}(N) = 2^N$, $d_{vc} = \infty$

If we consider the set in the example:

> A set of N points generated by first choosing one point, and then generating the next point by increasing the first component and decreasing the second component until N points are obtained

None of these points have $x_1 > x_2$, so our hypotheses set has no specification on how it can label these points. Therefore, it can always be shattered.

Problem 2.24

a

If we analytically calculate $\bar{g}$, it would be in the form $ax + b$. In this case, since our uniform random variable is between [-1,1], our $\bar{g}$ is $0x + 0$, or 0.

b

We start by generating 100 data sets. Then, our learning algorithm uses these two points to return to us a g in the form of $ax + b$. We will use our set of 100 gs to calculate bias(x) and var(x). These will be mean squared error.

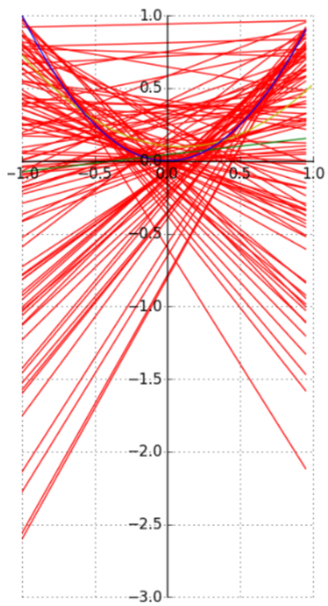
bias(x) $= \frac{1}{K} * \sum_{k=1}^{K}(g_k(x) - f(x))^2$

var(x) $= \frac{1}{K} * \sum_{k=1}^{K} (\bar{g} - f(x))^2$

Then, we will integrate bias(x) and var(x) from [-1,1] and multiply by $\frac{1}{b-a}$ to get var and bias. Then, we can add these to get $E_x[E_{out}]$

c

I ran the experiment with 100 data sets. I have achieved this result:



The red lines are our gs, the blue line is $f(x)$, and the green line is $\bar{g}$. When I run these results into our calculation, I get bias = 0.195 and var = 0.139. This totals $E_x[E_{out}]$ to 0.335

d

We know that $\bar{g}(x) = 0$ right from the start because the expected value of both a and b in $ax + b$ are 0.

bias $= E_x$[bias(x)]

$= E_x[(\bar{g}(x) - f(x))^2]$

$= E_x[(0 - x^2)^2]$

$= E_x[x^4]$

We use the expectation formula, with [a,b] as [-1,1]:

$\frac{1}{b-a}\int_a^b f(x)dx$

bias = 0.2

var is calculated in a very similar way, yielding 0.2 as well.

This gives us a calculated $E_x[E_{out}]$ of 0.4. This is given unlimited data sets.