Introduction to Machine Learning (67577)

Exercise IV – Classification

Hadar Sharvit – 208287599

**Theoretical Questions**

**PAC learnability**

1. Prove that s.t : is equivalent to :

Using *Markov’s Inequality*

since then , and therefore, there exists some for which is arbitrarily close to , which is equivalent to writing , as needed.

1. Prove that is learnable with :

Similar to what we have seen in class, we will use the following learning algorithm-

Given points in , return the circle with the smallest area that includes all of the samples and none of the samples. If there are no such circles, return .

For convenience’s sake, we will denote a ring with inner radius and outer radius as , and the radius of some circle as .

Since the tightest fit circle is always contained in the true circle , the error can only come from positive samples , and If we are able to guarantee that the weight under of is at most , then the error of is at most .

Suppose the ring weighing exactly is for some circle . now some points to consider:

* The ring has weight exceeding iff .
* iff there are no points of in .
* if some point is in , then the algorithm would have made include .
* The probability of sampling a point that misses is (and for samples if would be

From those we understand that if we choose such that , then with probability over the random samples the weight of the error is at most . Now, since , let us choose , thus

**VC-dimension**

1. For finite class : :

A set is of maximal size , therefore .Now, since is all function , its size is , and since we have

1. Find of :

will show that

that is, subset of size for which we can generate both and . We will choose where is a unit vector with 1 in the row and otherwise. Given labels , can generate them all because when summing over all subsets in , we can get any permutation of . in other words, by definition of , using the singleton for some (see below) we have , and from here for any union of singletons , the returned label is any permutation of ones and zeros, such that we will see a in index if the singleton is a part of the union.

generates every label:

And every non singleton is a combination of a set of singleton labels, which generates all the remaining labels.

This can also be explained as followed: for every , to generate choose which does not contain , and for choose which contains .

This in turn shows that is shattered by H, so , but from section (3) we have , therefore .

1. We will show that , which is if is not bounded:

* : any label of size can be generated using intervals. the worst-case scenario for a label, in terms of the number of intervals we need to use, is when there are no adjacent ’s. This is because for any repetition of ’s we can simply use one interval to include them all. If so, for non-adjacent s (that are separated with ’s), we must have intervals (interval for each ). Furthermore, a label of size has at most non-adjacent ’s, because between two non-adjacent ’s there is at least one .
* : assume on a way of contradiction that there exists a group of size that can be shattered by . If so, the label with non-adjacent ’s and non-adjacent ’s can be generated. This is a contradiction because non-adjacent ’s can only be generated using intervals (as previously discussed), and we use only .

1. We will show , where .

* : assume by contradiction , so a set is shattered by . Consider the hypothesis

Since there are variables, and contains of them, there exists two literals, call them and , that represent the same (either or ).

If : then but . This is a contradiction because

If : suppose w.l.o.g that . By definition of we have , though since then . This is a contradiction.

* : consider the group of unit vectors . For some label vector we can choose the following hypothesis:

For such we have , therefore

**Agnostic-PAC**

1. has uniform convergence property with is agnostic-PAC learnable with sample complexity :

has uniform convergence property with on . Since that claim is true let us choose , therefore . Now, since is representative, we know that , where .

Thus, we have , which is the definition of Agnostic-PAC learnability, w.r.t sample complexity

1. is not agnostic PAC learnable:

The claim suggests that can depend on , even though it should not according to the definition of PAC learnability. This dependance (of on ) suggests that any hypothesis class satisfies the statement. As a counter example, consider the hypothesis class of all functions over some non-finite domain . We have seen that such is not PAC learnable. Furthermore, we have seen in class that is PAC learnable iff is agnostic PAC learnable, therefore, since the class above is not PAC learnable, it is not agnostic PAC learnable even though it satisfies the question’s claim.

**Monotonicity**

1. : assume by contradiction that but . By definition, is the minimal number of samples needed to achieve , but from our assumption we have using samples. In other words, we have used less samples to achieve better resolution, which is a contradiction.

The claim for is very similar:

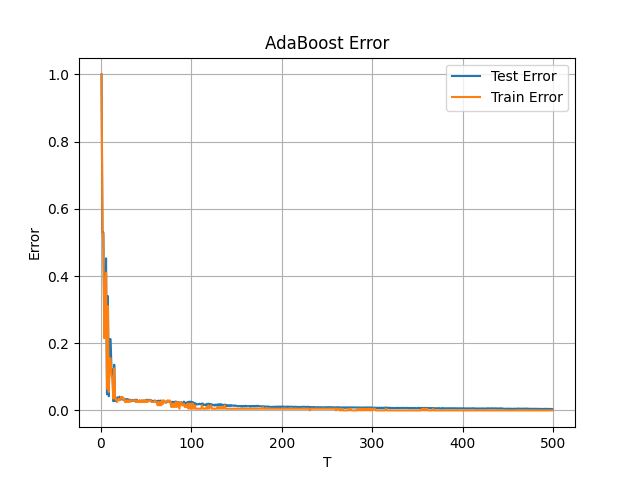
assume by contradiction that but . By definition, is the minimal number of samples needed to achieve , but from out assumption we have , using . In other words, we have used less samples to achieve better accuracy, which is a contradiction.

1. Denote for . Let us assume by contradiction that . If so, there exists some that is shattered by and not by , which in turns mean that there exists some that, given , generates every label. Since , then , therefore also shatters , in contradiction to the claim that it’s dimension is smaller than ’s . From here we conclude
2. Not for submission

**Practical Questions**

**Separate the Inseparable – AdaBoost**

1. code
2. Ada-Boost Error graph:



1. Decision boundaries of the learned classifiers:

Background pattern

Description automatically generated

1. , and the given error is

A picture containing text, fabric

Description automatically generated

1. Training set with size proportional to :

Chart, bubble chart

Description automatically generated

It is harder to classify points near the edges, therefore the classifier increments their respective weights many times (so they are represented with bigger points). Other points are easier to classify, which corresponds to smaller points with smaller weights.

1. Repeating the process with noise