Jared Yu

Assignment 1

1. **Overfitting of polynomial matching:** We have shown that the predictor defined in Equation (2.3) leads to overfitting. While this predictor seems to be very unnatural, the goal of this exercise is to show that it can be described as a thresholded polynomial. That is, show that given a training set , there exists a polynomial such that if and only if , where is as defined in Equation (2.3). It follows that learning the class of all thresholded polynomials using the ERM rule may lead to overfitting.

Ans: (used reference [1], [7])

The equation 2.3 for will first be restated as follows,

A slight change has been made to change and to and respectively. However, the meaning of the function is the same in that it evaluates to 1 if the input example matches an example from the training set and 0 otherwise.

The question is asking us to prove an if and only if statement. Therefore, it must be shown that if , then . Also, if , then . Starting with if , then , the function evaluates to 1 in any situation where an input example exactly matches an example from the training set . Therefore, if an input example exactly matches an example from the training set , then it must be shown that . Let

In , anytime an input is given, then the function calculates the product of for all examples in the training set . If the training set contains examples where , then the product will automatically evaluate to 1, based on the power . In situations where , then it will calculate the norm of the difference between the input example and the training example. If the input example matches one of those positive examples (i.e., where ) from the training set, then that value will zero out. This leads to the entire product, , evaluating to 0, or . However, if the input does not exactly match any of the training examples, then the norm will be greater than 0 for all training examples where . Therefore, the resulting product will be calculated with no 0’s in the equation leading to a positive non-zero value which is multiplied by as seen in the front of the product. The result is that when does not match an example from the training set, then .

The other direction also holds true, that is, if , then . However, it will be first noted that the maximum value for is 0, but the rule still holds. If an input example matches an example from the training set, then as mentioned above. For those same input examples, because they match a training example, then . This is since as described in Equation 2.3 evaluates to 1 anytime the input matches a training example and 0 otherwise. If the value of , then that indicates that the input example does not match an example from the training set and so . Therefore, it has been shown that if and only if .

1. Let be a class of binary classifiers over a domain . Let be an unknown distribution over , and let be the target hypothesis in . Fix some . Show that the expected value of over the choice of equals , namely,

Ans:

Let the loss function for the sample be defined as follows,

where and is an indicator function that evaluates to 1 when the equation inside of the brackets, is true and evaluates to 0 otherwise.

Let the loss function for the population be defined as follows,

where is the size of the population of observations within domain , and is the same as described previously.

In equation 2.3, the expectation is applied the formula for described in equation 2.1. For simplicity, also let .

In equation 2.4, the expectation is moved into the summation.

In equation 2.5, the expectation of the indicator is expanded to show that it results in the probability of , or. Looking closer at the formula, we can see the following,

The reason is that the probability of the classifier being incorrect for some random observation is the same as the sum of the incorrect classifications over the population of observations. The subscript is used for clarity in the next step.

In equation (2.7), the result from equation (2.6) is plugged back into equation (2.5). The summation turns into , since equation (2.6) does not require the subscript . The in the numerator and denominator cancel out, leaving just the formula for . Therefore, it can be said that .

1. **Axis aligned rectangles:** An axis aligned rectangle classifier in the plane is a classifier that assigns the value 1 to a point if and only if it is inside a certain rectangle. Formally, given real numbers , , define the classifier by

The class of all axis aligned rectangles in the plane is defined as

Note that this is an infinite size hypothesis class. Throughout this exercise we rely on the realizability assumption.

* 1. Let be the algorithm that returns the smallest rectangle enclosing all positive examples in the training set. Show that is an ERM.

Ans: (used reference [2])

Let be the training set. The error of the classifier over the training sample is:

where . The Empirical Risk Minimization (ERM) is the learning paradigm that comes up with a predictor that minimizes . In , all the classifiers are rectangles. Within , there are infinite rectangle-shaped classifiers that can fit the sample data in . If is an algorithm that returns the smallest rectangle enclosing all positive examples in the training set, then the classifier produced by algorithm would have a corresponding , since it would enclose all positive examples of (by the realizability assumption). This is the minimum possible value for the training error. Therefore, minimizes and so it is an ERM.

* 1. Show that if receives a training set of size then, with probability of at least it returns a hypothesis with error of at most . *Hint:* Fix some distribution over , let be the rectangle that generates the labels, and let be the corresponding hypothesis. Let be a number such that the probability mass (with respect to ) of the rectangle is exactly . Similarly, let be numbers such that the probability masses of the rectangles , , are all exactly . Let be the rectangle returned by . See illustration in Figure 2.2.

Ans: (used reference [3], [4], [5], [6], [8])

(Hint 1): Let be the rectangle that generates the labels. Therefore, all points inside the rectangle generated by are labeled and all points outside are labeled . Let be the rectangle returned by algorithm . The rectangle of is such that it is a rectangle from such that the is the minimum from amongst the rectangles, where is the hypothesis associated with the rectangle that leads to a minimum error. Every rectangle has boundaries that are within the area of the rectangle , as defined in the question itself. So, for every possible , . Therefore, .Then produced by must be a subset of .

(Hint 2): Let the event . In other words, it is the event when for a sample , each rectangle is non-empty, or that the sample contains at least one positive example from them.

If there are positive examples in all for , then the , where is the hypothesis chosen by algorithm implementing . The reason is that if all rectangles have at least one example in them, then no rectangle can have an error of 0, since none of the rectangles can cover all the space of the other rectangles. In such a situation, this implies that if the sample error is nonzero for all hypotheses, it is not an event associated with , the set of misleading events. To be specific, let and . Furthermore, , or in other words, the event of the error being greater than (a.k.a. a bad hypothesis) is a subset of . Where is the set of misleading events and is the set of bad hypotheses.

Since event implies that the sample is not within , then the current sample (based on being part of event ) does not contain any bad hypotheses. Therefore, , when event occurs.

(Hint 3): Let event . It is when in a sample , there are no examples from rectangle . The probability mass of each of the for is . Then, the probability that does not contain an example from is . So, probability of event . This can be upper bounded using the following inequality: . Therefore, we can upper bound the probability of with . To restate the result, .

(Hint 4): Then, using the union bound, an upper bound can again be created with the following: .

(Conclusion): In the event of , there is no longer the event . Therefore, there is no longer the requirement that sample is not an element of the set of misleading samples. So, when event occurs, it contains the possibility for a misleading sample from within . Given that the sample is an element of the set , we also have the possibility of a hypothesis from . From this, we also have the following:

From this it follows that .

Let , where is the size and the right side of the inequality is as mentioned in the question itself. Then,

So, it has been shown that . Therefore, it also follows that .

* 1. Repeat the previous question for the class of axis aligned rectangles in .

Ans: (used reference [4])

Previously, the question showed that:

with

Changing the problem to , it becomes:

with

Also, change the training set to be of , where corresponds with the number of dimensions. The rectangle would change to . The rectangles , for , would have a similar design as in question 3.2, except that they would include the higher dimensions. The probability mass for each of the would be exactly .

(Hint 1): Let be the rectangle that generates the labels. Therefore, all points inside the rectangle generated by are labeled and all points outside are labeled . Let be the rectangle returned by algorithm . The rectangle chosen by is such that it is a rectangle from such that the is the minimum from amongst the rectangles, where is the hypothesis associated with the rectangle that leads to a minimum error. Every rectangle has boundaries that are within the area of the rectangle , as defined in the question itself. So, for every possible , . Therefore, .Then produced by must be a subset of .

(Hint 2): Let the event . In other words, it is the event when for a sample , each rectangle is non-empty, or that the sample contains at least one positive example from them.

If there are positive examples in all for , then the , where is the hypothesis chosen by algorithm implementing . The reason is that if all rectangles have at least one example in them, then no rectangle can have an error of 0, since none of the rectangles can cover all the space of the other rectangles. In such a situation, this implies that if the sample error is nonzero for all hypotheses, it is not an event associated with , the set of misleading events. To be specific, let and . Furthermore, , or in other words, the event of the error being greater than (a.k.a. a bad hypothesis) is a subset of . Where is the set of misleading events and is the set of bad hypotheses.

Since event implies that the sample is not within , then the current sample (based on being part of event ) does not contain any bad hypotheses. Therefore, when event occurs.

(Hint 3): Let event . It is when in a sample , there are no examples from rectangle . The probability mass of each of the for is . Then, the probability that does not contain an example from is . So, probability of event . This can be upper bounded using the following inequality: . Therefore, we can upper bound the probability of with . To restate the result, .

(Hint 4): Then, using the union bound, an upper bound can again be created with the following: .

(Conclusion): In the event of , there is no longer the event . Therefore, there is no longer the requirement that sample is not an element of the set of misleading samples. So, when event occurs, it contains the possibility for a misleading sample from within . Given that the sample is an element of the set , we also have the possibility of a hypothesis from . From this, we also have the following:

From this it follows that .

Let , where is the size and the right side of the inequality is as mentioned in the question. Then,

So, it has been shown that . Therefore, it also follows that .

* 1. Show that the runtime of applying the algorithm mentioned earlier is polynomial in , , and in .

Ans: (used reference [4])

The way that algorithm works is that it needs to scan each of the dimensions and analyze all the points in these dimensions to find the correct or values to help determine the such that it has a probability mass of exactly . Doing so implies that the runtime is , where is the number of examples in the training set and is the number of dimensions. Furthermore, the size of the training set has been upper bounded by . Therefore, taking the product of and leads to a polynomial in , , and .

**References**

[1] <http://www.math.ubc.ca/~elyse/220/2016/7Nonconditional.pdf>

Used to double-check how to prove an if-and-only-if statement.

[2] <http://www.people.vcu.edu/~rhammack/Math501/3ways.pdf>

Used to double-check how to prove an if-then statement.

[3] <https://www.sciencedirect.com/topics/computer-science/probability-mass>

Used for understanding the context of the term “probability mass” in problem 3.

[4] <https://www.cs.huji.ac.il/~shais/UnderstandingMachineLearning/MLbookSol.pdf>

Found from Piazza and used as a guidance for problem 3.

[5] <http://karlstratos.com/notes/pac.pdf>

Used as a guidance for problem 3.

[6] <https://piazza.com/class/kc0jkwru805u1?cid=38>

Question on Piazza about probability mass.

[7] <https://piazza.com/class/kc0jkwru805u1?cid=37>

Question on Piazza about problem 1.

[8] <https://piazza.com/class/kc0jkwru805u1?cid=17>

Comment on Piazza about resources for the textbook.