Gautam Ramasubramanian
Rutgers DIMACS Research

Active Learning Face #1 - Efficient Search Through Hypothesis Space

Problem

In Supervised Learning Problems, you have a set of training examples - data points with input features ($\mathcal{X}$) and the correct labels ($\mathcal{Y}$). This set is passed through a learning algorithm, which spits out a hypothesis function that can predict the inputs from the outputs ($h(\mathcal{X}) = \mathcal{Y}$). If we assume that the training examples are iid from a probability distribution P of $\mathcal{X} \times \mathcal{Y}$, then it is a well known result that the hypothesis function for a reasonable learning algorithm will converge to optimality if the number of training examples goes to ∞. This is for the case when all the training data is labeled with the "correct answers". Suppose you had a set of unlabeled data, and querying for labels will have a certain cost for each query. Furthermore, suppose the unlabeled data comes to you in a stream, and you have to decide then and there whether to query for its label or not. Then you have to learn from the data you did query, so the querying should be done is such a way that the resulting hypothesis is reasonably close to the real hypothesis if all the unlabeled data were queried - supervised learning. How do you do that? You means the computer.

Algorithms

1. *Generic Active Learner*

   One Naive Approach would be to

   (a) Query the first couple of points to get a plausible hypothesis function.

   (b) After that, query only points that are close to the hypothesis function and update the hypothesis so that it can be more "percise".

   The main disadvantage this algorithm and any other version of this algorithm has is sampling bias. If you query the points that are near to the current hypothesis function, you are query a sample that doesn't represent training set probabilistically. Therefore, you miss large chunks of data far away from your hypothesis that may be relevant.

2. *CAL Learner (Mellow Learner for Separable Data)* CAL algorithm is like a generalized binary search through the hypothesis space It does the following

   (a) Maintains a version of the hypothesis space at every time interval.

   (b) If a data point is such as all hypotheses in the space agree on its label, do not query and assume the label that the hypotheses in the hypothesis class tell you.

   (c) In the case that the hypotheses disagree on the label of the point, query for the actual point label and remove the hypothesis that gave you the wrong label from the class. You are essentially updating the hypothesis space by keeping those that are able to give the correct answers to all points seen thus far.

   This is a good algorithm for data that is separable by the functions in your initial hypothesis space. You don't have the problem of sampling bias because you are not excluding any hypothesis unless you are certain it is wrong. The points queried don't have to be near to an initial hypothesis, so it is more representative of the sample.

3. *DHM Learner - CAL for non-separable data, data with noise* The CAL algorithm assumes that all unqueried labels that are inferred by all the hypotheses in H are right. This is true for separable learning problem, but not true for the non-separable case. The following algorithm is a modification of CAL to work for nonseparable data. Actually its the same algorithm, same sequence of steps, but the reasoning for why it works even in the nonseparable case is also given below.

   (a) Maintains a version of the hypothesis space at every time interval. (Just Like CAL)

   (b) If a data point is such as all hypotheses in the space agree on its label, do not query and assume the label that the hypotheses in the hypothesis class tell you. (Just like CAL). But that label *might be wrong*

(c) In the case that the hypotheses disagree on the label of the point, query for the actual point label and remove the hypothesis that gave you the wrong label from the class. You are essentially updating the hypothesis space by keeping those that are able to give the "correct answers" to all points seen thus far.

(d) Doing the same procedure as CAL will result in a hypothesis space with hypotheses with nonzero error. however, it has been proven that the hypothesis space resulting from this algorithm will have hypotheses whose errors are bounded. The more iterations of the algorithm, the smaller the bound from the minimum error. Therefore, we can get arbitrarily close to a hypothesis with minimum error in the nonseparatable case by just following the same algorithm.

Important Ideas

1. VC dimension - *See Summary on VC Dimension*
2. Label Complexity - The number of queries a learning algorithm has to make in order to come up with its optimal hypothesis. How does that compare to full supervised learning where every label is queried?