Risk Bounds for Classifier Evaluation: Possibilities and Challenges

Mohak Shah MOHAK@CIM.MCGILL.CA
Center for Intelligent Machines, McGill University, Montreal, H3A 2A7 Canada

Abstract

We discuss the possibilities and issues in using learning theoretic risk bounds for classifier evaluation. We show how test set bounds already compare favorably to existing comparable measures and what possibilities can be explored for the training set bounds. When used properly, these bounds can yield more meaningful evaluaton measures.

1. Introduction

Evaluation of Machine Learning algorithms is crucial to both our ability to assess the effectiveness of the proposed approach as well as our understanding of its applicability to the domain of interest. With respect to the classification scenario, the focus of our discussion here, several approaches have been explored that can help not only in assessing the *chosen* classifier but also *selecting the best classifier* from the classifier space. The former class of problem is widely known as the problem of *error estimation or classifier evaluation* while the later as *model selection*. Although the two problems as well as various approaches in both these cases are related and have a considerable overlap, there are crucial differences between what these approaches address in each scenario. The problem of model selection is the one that helps decide what is the *best classifier in the space of classifiers* that a particular algorithm explores *given the training data*. This generally appear, in addition to a particular algorithm's learnng bias, as selecting the best set of parameters, e.g., k in the case of a k-Nearest Neighbor classifier. On the other hand, the problem of error estimation pertains to the problem of assessing the performance of the *chosen classifier* given some test data. Hence, the error estimation approaches basically not only aim at giving reliable performance estimates for a given classifier but also provides a platform to compare two competing classifiers (resulting from two different learning algorithms) on a given domain of interest (assumed distribution over a representative test set).

Most of the classifier evaluation approaches that have appeared are empirical in nature (although have theoretical foundations in statistics). For instance, parameteric approaches such as ones based on t-test essentially assumes a distribution over the classification error when comparing two estimates and aims to discover if the differences are indeed statistically significant. Other approaches such as ROC curves aim at gauging the classifier performance over a range of parameter values (and hence can essentially function as model selection criteria too) resulting in the AUC estimate for evaluation. Furthermore, there are approaches that obtains confidence intervals around the test error of a classifier and give more weights to the non-overlapping intervals. Similarly there are resampling based approaches that result in error bars around the estimates and are effective in the case of limited sample sizes.

All these approaches have so far helped and continue to guide the process of assessing classifiers' performances in the domains of interest. However, there is one crucial factor to take note of with regard to such approaches. Each of these approaches take into account only the empirical performance of the classifier on a given test set. These are independent of the nature of the learning algorithms in question or *the guarantees on their future performance*. It is indeed possible that a learning algorithm performs really well on a given test set but does not generalize well in future. The only assumption made about the test set performance is that it is representative of the future unseen examples (and hence overall distribution) of the domain. This is not an unreasonable assumption. However, the estimates sometime break. For instance, a confidence interval based estimate around the empirical error essentially yields an interval of zero size for a consistent classifier (classifier with zero empirical error). However, just as a training set can be insufficiently representative of the underlying distribution so can the test set. In dealing with the (un)representativeness of the

Appearing in *Proceedings of the 3^{rd} Workshop on Evaluation Methods for Machine Learning in conjunction with 25^{th} Intl Conf on Machine Learning*, Helsinki, Finland, 2008.

training set, approaches have evolved so that the algorithms do not underfit or overfit the data. Any algorithm that aims to be practical enough deals with the issue in one form or the other. For instance, decision trees are generally pruned (after building a complete tree) while a higher value of k secures against overfitting in the case of k-nearest neighbor. Similarly, underfitting is dealt with too. But notice the crucial difference between the manner in which we address this issue in the case of training data. The approaches do take into account the very nature of the learning algorithm.

Statistical learning theory has made attempts to characterize the performance of the classifier as well as the guarantees over its future performance. Such results have generally appeared in the form of generalization error bounds. These guarantees basically provide upper (and sometimes lower) bounds on the deviation of the true error of the classifier from its empirical error and take into account the precise quantities that a classifier learns from the data. In this position paper, we wish to examine if two or more classifiers (and hence learning algorithms) can be compared in this theoretical premise of the generalization error bounds. We discuss some of the main challanges that need to be overcome, the issues that need to be addressed and the opportunities that exist in doing so.

2. Learning Theoretic Bounds

Providing generalization error bounds on the classifier involves characterizing an algorithm in a given theoretical framework. Different frameworks exploit different criteria for characterizing an algorithm. The Probably Approximately Correct (PAC) framework, probably the oldest such framework and most widely used, provides approximate guarantees on the true error of a classifier with a given confidence parameter δ as can be seen in the sample bounds presented below. The generalization error bounds or risk bounds appear in two main variants: *the test set (or holdout) bounds* and *the training set bounds*. The holdout bounds are the guarantees on the true error of the classifier obtained on a given test set and can be obtained without reference to the learning algorithm in question. The training set bounds on the other hand, inspired by resampling requirement as a result of limited data availability and model selection considerations, almost always take into account (or rather exploits) the characteristics of the learning algorithm. We will explore how such theoretical frameworks can be put to use in classifier evaluation.

We explore the utility of both types of bounds in classifier evaluation. As we will see, while the test set bounds can readily be utilized for the purpose, there are many challenges with regard to the training set bounds. Before we proceed further, we list some of the notations that we use from here onward. The empirical error or risk of a classifier means the error that the classifier makes on a given training set (in the case of training set bounds) or a test set (for test set bounds) and is denoted by $R_S(f)$ for a classifier f and training/test set S of m examples. The true risk, denoted by $R(f)$ is f's true risk on future unseen examples over the distribution $\mathcal{D}$ from which the samples are drawn i.i.d.. The rest of the notation are explained in the context.

3. Test Set Bound

We start by showing a sample test set bound:

Theorem 1 [1] *For all $\delta \in (0, 1]$, $\forall f$:*

$$\Pr_{S \sim \mathcal{D}^m} \left(R(f) \leq R_S(f) + \sqrt{\frac{\ln \frac{1}{\delta}}{2m}} \right) \leq 1 - \delta$$

where $R(f)$ and $R_S(f)$ are the true and empirical risk of classifier f.

Note in the above bound that $R_S(f)$ represents the empirical risk of the classifier on a *test* or *holdout* set. The important point to note in the above bound is that the quantifier $\forall f$ appears outside the probability. That is, the bound holds true for *any* classifier f and not f in some fixed classifier space $\mathcal{H}$ uniformly. This shows that the test set bounds generally are applicable to any independent estimate of any classifier on a test or holdout set. It appears that, with regard to traditional statistics, this corresponds to providing a confidence interval around $R_S(f)$. However, the test set bound approach have some distinct advantage.

Applying confidence intervals around $R_S(f)$ implies computing a mean and variance of the estimate and then providing a confidence interval with, say, 2 standard deviations around the mean. This implicitly assumes the distribution of the observed empirical error to be Gaussian asymptotically (i.e., in the limit that the number of examples goes to infinity). The test set bound works, on the other hand, by considering the error distribution to be binomial. The effect of the approach is that the test set bound approach lead to (upper and lower) estimates that lie in the $[0, 1]$ interval while this might not be the case in the confidence interval approach. Further, the bound approach has a more predictable behavior when a consistent classifier is found. That is, a classifier with zero empirical risk. In this case, unlike confidence interval approach, the bound estimate is not zero for a finite test set. [1] discusses this in more details.

Hence, the test set bounds can be better evaluation measures as opposed to the traditional confidence interval estimates and can lead to more meaningful classifier comparisons. Let us now discuss the training set bound scenario.

4. Training set bounds

The idea of training set bounds is more involved. Various theoretical frameworks can be utilized to characterize the behavior of a learning algorithm and bound the true risk of the classifier. Most of these are built on the PAC framework mentioned above with a confidence measure δ. The lower the δ, the more the confidence in the estimate about the true risk (and by consequence looser the bound) and *vice versa*. Such models, generally, provide these guarantees over the future classifier performance in terms of its empirical performance and possibly some other quantities obtained from training data, and some measure on the complexity of the hypothesis class that the learning algorithm explores. Such measures have appeared in the form of VC-dimension, Rademacher Complexities and so on. There are other learning frameworks that do not include explicitly algorithm's dependence on the hypothesis class complexity in the risk bound and hence have an advantage over conventional bounds that do. This is because, the complexity measure grows with the size (and complexity) of the hypothesis class and many a times result in unrealistic bounds. See [4] for a quick review of trainng set bounds. Successful attempts in the direction of attaining practical, realizable bounds, although few, have appeared specifically for sample compression framework.

Briefly, sample compression framework relies on characterizing a classifier in terms of two complementary sources of information viz. a compression set $S_{\mathbf{i}}$, where $\mathbf{i}$ denotes the vector of indices pointing to the examples in the compression set, and a message string σ. Hence, compression set is a (preferably) small subset of the training set and the message string is the additional information that can be used to reconstruct the classifier from the compression set. Consequently, this requires the existence of such a Reconstruction function that can reconstruct the classifier solely from this information. The risk bound that we present below as an example, basically, bounds the risk of the classifier represented by $(\sigma, S_{\mathbf{i}})$, over all such reconstruction functions. The bound presented below is due to [2] who also utilized this bound to perform successful model selection in the case of Set Covering Machine algorithm [3].

Theorem 2 *For any reconstruction function $\mathcal{R}$ that maps arbitrary subsets of a training set and message strings to classifiers, for any prior distribution $P_{\mathcal{I}}$ of vectors of indices (where $\mathcal{I}$ denotes all possible 2^m realizations of $\mathbf{i}$) , for any compression set-dependent distribution of messages $P_{\mathcal{M}(S_{\mathbf{i}})}$ (where $\mathcal{M}$ denotes the set of messages that can be supplied with compression set $S_{\mathbf{i}}$), and for any $\delta \in (0, 1]$, the following holds with probability $1 - \delta$ over random draws of $S \sim \mathcal{D}^m$:*

$$\forall \mathbf{i} \in \mathcal{I}, \forall \sigma \in \mathcal{M}(S_{\mathbf{i}}) \colon R(\mathcal{R}(\sigma, S_{\mathbf{i}})) \leq 1 - \exp\left(\frac{-1}{m-d-k} \left[\ln \binom{m-d}{k} + \ln\left(\frac{1}{P_{\mathcal{I}}(\mathbf{i}) P_{\mathcal{M}(S_{\mathbf{i}})}(\sigma) \delta} \right) \right] \right) \quad (1)$$

where for any training set S, d is the sample compression set size of classifier represented by $(\sigma, S_{\mathbf{i}})$, $\mathcal{R}(\sigma, S_{\mathbf{i}})$ and k is the number of training errors that this classifier makes on the examples that are not in the compression set.

As can be seen, the above bound will be tight when the algorithm can find a classifier with small compression set d (a property known as sparsity) along with a small empirical risk (k). Also, note that the quantifier over the classifiers, $\forall \mathbf{i} \in \mathcal{I}, \forall \sigma \in \mathcal{M}$ appears *inside* the probability. This is because the above bounds applies to all the classifiers in a given classifier space uniformly, unlike the test set bound. Hence, the training set bound focuses precisely on what the learning algorithm can learn (in terms of its reconstruction) and its empirical performance on the training data. As also, discussed before, training set bounds such as the one shown above, also provide an optimization problem for learning and theoretically a classifier that minimizes the risk bound shoud be selected. However, this statement should be considered more carefully. As also discussed by [1], choosing a classifier based on the risk bound necesarily means that this gives a better worst-case bound on the true risk of the classifier. This is different from obtaining an improved estimate of true risk. Generally, measures such as empirical risk that guide the model selection have a better behavior. Some successful examples of learning from bound minimization do exist however. See for instance [2]. Further, there can also be other considerations as we discuss below.

5. Bounds for classifier evaluation?

With the progress on the learning theory front in providing tighter risk bounds for classifiers, there lies a potential in utilizing these bounds for classifier evaluation too. An interesting direction that demands attention (or will do in near future) appears to be the

ability to utilize the risk bounds for classifier evaluation. Performing successful model selection with bounds appears an encouraging advancement. The test set bounds appear to be a more direct method for such classifier comparisons and can result in more meaningful confidence estimates around the observed empirical behavior of the classifier. The training set bounds' utilization for the purpose, however, warrants a deeper understanding as well as addressing various issues before successful application. A standardized optimal framework can result in specification of learning algorithms and enable inter-algorithm comparison on a common platform. Although, it remains to be seen how can this be done meaningfully. Many a issues remain to be addressed. For instance, if algorithms A (e.g. SCM) is characterized in certain framework, then is this framework optimal too for characterizing algorithm B (e.g., SVM) with which we wish A to be compared? This is, when at all, such characterization is possible.

A concrete example can be seen in the case of sample compression framework described above. A necessity is to have a reconstruction function that can reconstruct the classifier from compression sets and messge strings. Many a algorithms confirm to such reconstruction function existence while there are algorithms for which a direct reconstruction scheme does not exist. For instance, algorithms such as the Set Covering Machine [3] (SCM) has been designed with sparsity considerations and can be successfully characterized in this framework. Similarly, algorithms such as the Support Vector Machines can also be represented in this framework. So can the algorithms such as Decision Trees [5]. However, Algorithms such as SVM, although characterizable in sample compression framework, are not designed originally with sparsity as the learning bias. Hence, such a comparison will always yield biased estimates. On the other hand, sample compression algorithms work on hypothesis class that is defined after having the training set at hand (since each classifier is defined in terms of a subset of the training set), a notion widely known as data-dependent settings. A complexity measure such as VC-dimension, defined without reference to the data and applicable in the case of SVMs, cannot characterize the complexity of hypothesis class that sample compression algorithms explore.

5.1. Advantages and Limitations

One of the main advantage of these (risk bound) approaches is that this brings us to more evolved and meaningful performance evaluation measures for classifiers. That is, in addition to conventional quantities such as error-rates, what other qualities of algorithms can be considered in its evaluation. When considering the training set bounds, this is quite important since different algorithms exploit different learning biases and a better estimation about its generalization performance would naturally depend on how well do the algorithms exploit the concerned bias. Moreover, this opens up doors to a wide variety of research directions. For instance, we can compare algorithms that optimize similar biases on such criterion. Hence, for a given framework, one can have an idea of what kind of algorithms should be preferred. For instance, sample compression framework has algorithms that optimize a trade-off between the compression set size and the empirical risk. Hence, the classifier with increasing complexity (in terms of compression set) are preferred only when they lead to significantly better empirical performance. The question now, while comparing two learning algorithms, is how well each algorithm can perform this trade-off and what would be the repercussions of these selections in future. Other considerations also come in play here such as the resulting nature of the optimization problem when using such frameworks. Also, it is currently an active research question about obtaining tight enough training set bounds. Examples such as shown above are few. It would be interesting to see advances on this front in the near future and their impact on the classifier evaluation field. The test set bounds on the other hand provide a readily favorable alternative to the confidence interval based approaches in terms of more meaningful characterization of classifier's empirical performance.

References

[1] John Langford. Tutorial on practical prediction theory for classification. *Journal of Machine Learning Research*, 3:273–306, 2005.

[2] Francois Laviolette, Mario Marchand, and Mohak Shah. Margin-sparsity trade-off for the set covering machine. In *Proceedings of the 16th European Conference on Machine Learning, ECML 2005*, volume 3720 of *Lecture Notes in Artificial Intelligence*, pages 206–217. Springer, 2005.

[3] Mario Marchand and John Shawe-Taylor. The Set Covering Machine. *Journal of Machine Learning Reasearch*, 3:723–746, 2002.

[4] Mohak Shah. *Sample Compression, Margins and Generalization: Extensions to the Set Covering Machine*. PhD thesis, SITE, University of Ottawa, Ottawa, Canada, May 2006.

[5] Mohak Shah. Sample compression bounds for decision trees. In *ICML '07: Proceedings of the 24th international conference on Machine learning*, pages 799–806, New York, NY, USA, 2007. ACM Press.