Improved Uniform Test Error Bounds

Eric Bax*

August 18, 1997

Abstract

We derive distribution-free uniform test error bounds that improve on VC-type bounds for validation. We show how to use knowledge of test inputs to improve the bounds. The bounds are sharp, but they require intense computation. We introduce a method to trade sharpness for speed of computation. Also, we compute the bounds for several test cases.

Key words machine learning, learning theory, generalization, Vapnik-Chervonenkis.

*Computer Science Department, California Institute of Technology 256-80, Pasadena, California, 91125 (eric@cs.caltech.edu).

1 Introduction

In learning theory, bounds on out-of-sample error measure the ability of a learning scheme to generalize, i.e., to perform on data not used for training. Bounds that are uniform over a set of classifiers have applications to model selection and validation.

Model selection is the choice, without reference to the training data, of a class of classifiers from which to select a classifier through training. VC analysis [2, 3, 7, 12, 11] shows that if the class is sufficiently restricted and there are enough training data, then the error rates on training data are uniformly good estimates of the error rates on out-of-sample data with high probability. Hence, the classifier selected because it has low training error is likely to have low out-of-sample error as well.

Validation is the use of some in-sample data to evaluate trained classifiers. First, the in-sample data is partitioned into training and validation data. Next, classifiers are selected using the training data. Then the error rates over the validation data are evaluated for the trained classifiers. Since the trained classifiers are chosen without reference to the validation data, the trained classifiers play the role of a class and the validation data plays the role of training data in a simplified version of VC analysis [1]. If there are sufficiently few trained classifiers and enough validation data, then the error rates on validation data are uniformly good estimates of the error rates on out-of-sample data with high probability. Hence, the classifier or classifiers selected because they have low validation error are likely to have low out-of-sample error as well.

In this paper, we develop improved uniform bounds. These bounds apply directly to validation schemes with few classifiers, including early stopping with central classifiers [4] and a method to validate voting committees and other stacked classifiers [5]. In these schemes, uniform error bounds are required over only a few classifiers, and error bounds for the remaining classifiers are derived by inference. The bounds that we develop are computationally expensive for large sets of classifiers, so we introduce methods to mediate the tradeoff between bound strength and computational efficiency by merging the new bounds with those provided by VC analysis. The merged bounds apply in the full VC framework.

The paper proceeds as follows. First, we introduce our machine learning framework and review VC-style bounds for validation. Next, we derive improved bounds. Then, we discuss the computation required to calculate the bounds, and we present a method to trade bound strength for computational efficiency. We conclude by calculating the improved bounds for several cases.

2 VC-Style Uniform Error Bounds

2.1 Framework

Our machine learning framework has the following structure. There is an unknown boolean-valued target function and an unknown distribution over its input space. For example, the input distribution could be typical data about credit card applicants, and the target function could be 1 if the applicant defaults within 5 years of being issued a credit card and 0 otherwise.

We have a set of classifiers $g_1, \ldots, g_M$. We have d validation examples, which were not used to train the classifiers. We want uniform bounds on the error rates of the classifiers over d' as-yet-unseen test examples. (The error rate of a classifier on a data set is defined as the rate of disagreement between the classifier and the target function over the inputs in the data set.) The validation and test inputs are drawn independently at random according to the underlying input distribution.

2.2 Single-Classifier Bound

The first step in developing VC-style bounds that are uniform over classifiers is to develop a bound for a single classifier g_m. Let ν_m be the validation error of g_m, and let ν'_m be the test error. Let $n = d + d'$, the number of inputs in the validation and test data combined. The probabilities in our error bounds are over partitions of the n inputs into d validation examples and d' test examples. Since the inputs are drawn i.i.d., each partition is equally likely.

Let w be the number of the n inputs for which classifier g_m produces the incorrect output. The probability that the validation error is $\frac{k}{d}$ is

$$\binom{n}{d}^{-1} \binom{w}{k} \binom{n-w}{d-k} \tag{1}$$

If the validation error is $\frac{k}{d}$, then the test error is $\frac{w-k}{d'}$. So

$$\Pr\{\nu'_m \geq \nu_m + \epsilon | w\} = \sum_{\{k | \frac{w-k}{d'} \geq \frac{k}{d} + \epsilon\}} \binom{n}{d}^{-1} \binom{w}{k} \binom{n-w}{d-k} \tag{2}$$

Bound by maximizing over w.

$$\Pr\{\nu'_m \geq \nu_m + \epsilon\} \leq \max_{w \in \{0,\ldots,n\}} \Pr\{\nu'_m \geq \nu_m + \epsilon | w\} \tag{3}$$

We refer to the bound as $B(\epsilon)$.

2.3 Uniform Bound

The single-classifier bound is

$$\Pr\{\nu'_m \geq \nu_m + \epsilon\} \leq B(\epsilon) \tag{4}$$

Equivalently,

$$\Pr\{\nu'_m < \nu_m + \epsilon\} \geq 1 - B(\epsilon) \tag{5}$$

To obtain uniform bounds, consider the probability of at least one single-classifier bound failure:

$$\Pr\{\nu'_1 \geq \nu_1 + \epsilon \text{ or } \ldots \text{ or } \nu'_M \geq \nu_M + \epsilon\} \tag{6}$$

Bound the probability of the union event by the sum of event probabilities.

$$\leq \Pr\{\nu'_1 \geq \nu_1 + \epsilon\} + \ldots + \Pr\{\nu'_M \geq \nu_M + \epsilon\} \tag{7}$$

Use the single-classifier bound for each probability.

$$\leq MB(\epsilon) \tag{8}$$

Subtract $MB(\epsilon)$ from one to bound the probability of the complement of (6).

$$\Pr\{\nu'_1 < \nu_1 + \epsilon \text{ and } \ldots \text{ and } \nu'_M < \nu_M + \epsilon\} \geq 1 - MB(\epsilon) \tag{9}$$

This is the VC-style uniform bound.

3 Improved Uniform Error Bounds

In this section, we improve the uniform error bound by extending the derivation of the single-classifier bound to develop a uniform bound. Instead of estimating the probability of one or more single-classifier bound failures by summing over the individual failure probabilities, we consider the probability of the union event directly. As a result, we produce sharper bounds.

Just as the single-classifier bound is based on the worst-case number w of incorrect outputs over the n validation and test inputs, the multiple classifier bound is based on the worst-case pattern of incorrect outputs among the classifiers over the inputs. For $S \subseteq \{1, \ldots, M\}$, let w_S be the number of inputs for which the classifiers indexed by S are incorrect, and the other classifiers are correct. Define

$$\mathbf{w} = (w_\emptyset, \ldots, w_{\{1,\ldots,M\}}) \tag{10}$$

Also, let

$$H_m = \{S \subseteq \{1, \ldots, M\} | m \in S\} \tag{11}$$

Note that H_m is the set of subsets that index classifier g_m. Define

$$\mathbf{w}_m \cdot \mathbf{1} = \sum_{S \in H_m} w_S \tag{12}$$

Note that $\mathbf{w}_m \cdot \mathbf{1}$ is the number of inputs for which g_m is incorrect.

For a given partition of the n inputs into validation and test data, let c_S be the number of the inputs counted by w_S that are in the validation set. Define $\mathbf{c}$ and $\mathbf{c}_m \cdot \mathbf{1}$ corresponding to $\mathbf{w}$ and $\mathbf{w}_m \cdot \mathbf{1}$. Note that $\mathbf{c}_m \cdot \mathbf{1}$ is the number of validation examples for which g_m errs.

The probability that the validation errors assume the values $\frac{k_1}{d}, \ldots, \frac{k_M}{d}$ is

$$\Pr\{\forall m | \nu_m = \frac{k_m}{d} | \mathbf{w}\} = \binom{n}{d}^{-1} \sum_{\substack{\mathbf{c} \geq \mathbf{0} | \\ \sum_{S \subseteq \{1,\ldots,M\}} c_S = d \\ \text{and } \forall m \; \mathbf{c}_m \cdot \mathbf{1} = k_m}} \prod_{S \subseteq \{1,\ldots,M\}} \binom{w_S}{c_S} \tag{13}$$

where $\forall m$ abbreviates $\forall m \in \{1, \ldots, M\}$, and $\mathbf{c} \geq \mathbf{0}$ represents the condition that each entry of $\mathbf{c}$ is nonnegative.

If the validation error of g_m is $\nu_m = \frac{k_m}{d}$, then the test error is $\nu'_m = \frac{\mathbf{w}_m \cdot \mathbf{1} - k_m}{d'}$. To find the probability that the uniform bound over classifiers fails, sum the probabilities of cases in which one or more single-classifier bounds fail, i.e., $\nu'_m \geq \nu_m + \epsilon$ for some m. Define the failure set F as follows.

$$F = \{(k_1, \ldots, k_M) | (\exists m | \frac{\mathbf{w}_m \cdot \mathbf{1} - k_m}{d'} \geq \frac{k_m}{d} + \epsilon) \text{ and } (\forall m | 0 \leq k_m \leq d) \text{ and } (\forall m | 0 \leq \mathbf{w}_m \cdot \mathbf{1} - k_m \leq d')\} \tag{14}$$

(The last two conditions follow from requiring that the validation and test error rates be between zero and one.) Summation of (13) over the failure set produces a uniform bound conditioned on $\mathbf{w}$.

$$\Pr\{\exists m | \nu'_m \geq \nu_m + \epsilon | \mathbf{w}\} = \sum_{(k_1,\ldots,k_M) \in F} \Pr\{\forall m | \nu_m = \frac{k_m}{d} | \mathbf{w}\} \tag{15}$$

To remove the dependence on the (unknown) value of $\mathbf{w}$, maximize over all possible values.

$$\Pr\{\exists m | \nu'_m \geq \nu_m + \epsilon\} \leq \max_{\mathbf{w} \geq \mathbf{0} | \mathbf{w} \cdot \mathbf{1} = n} \Pr\{\exists m | \nu'_m \geq \nu_m + \epsilon | \mathbf{w}\} \tag{16}$$

where $\mathbf{w} \cdot \mathbf{1}$ is the sum of entries in $\mathbf{w}$. Denote the bound by $B_M(\epsilon)$.

The uniform bound has been derived under the assumption that the validation inputs and outputs are known, since we need them to compute ν_m. We have assumed that the test inputs and outputs are unknown. Now suppose that we know the test inputs. We still cannot compute the pattern of incorrect outputs, $\mathbf{w}$, because the test outputs are unknown. However, we can compute the pattern of agreements among classifiers over the inputs, and we can use this information to constrain $\mathbf{w}$ in bound (16).

For $S \subseteq \{1, \ldots, M\}$, let a_S be the number of the n validation and test inputs for which each classifier indexed by S returns 1, and each other classifier returns 0. For each input counted by a_S, the classifiers indexed by S are either all right or all wrong. So the input is counted by either w_S or $w_{\overline{S}}$. Hence we have the following additional constraints for (16).

$$\forall S \subseteq \{1, \ldots, M\} \quad w_S + w_{\overline{S}} = a_S + a_{\overline{S}} \tag{17}$$

Denote the bound obtained with these constraints by $B(\epsilon, \mathbf{a})$.

Even if we do not know the test inputs, we can use knowledge of the validation inputs to constrain $\mathbf{w}$. Let v_S be the number of validation inputs for which the classifiers indexed by S are incorrect, and the other classifiers are correct. The inputs counted by v_S are a subset of the inputs counted by w_S, so we have the following additional constraints for (16).

$$\forall S \subseteq \{1, \ldots, M\} \quad w_S \geq v_S \tag{18}$$

4 Computation and Mixed Bounds

Computation of bound $B_M(\epsilon)$ by formula (16) is intractable for large numbers of classifiers and large data sets. The computation can be reduced by using a few tricks. For example, since the classifiers have interchangeable roles in the computation, the maximization over $\mathbf{w}$ can be constrained to the values such that the error rates over the combined validation and test data are ordered by classifiers: $\{\mathbf{w} | \mathbf{w}_1 \cdot \mathbf{1} \geq \ldots \geq \mathbf{w}_M \cdot \mathbf{1}\}$. Also, many terms in subformulas (13) and (15) can be combined through a recursive algorithm that assigns a value to an entry of $\mathbf{c}$ at each level. (For details, contact the author.) Still, the computation of the bound (16) remains intractable for large cases.

Suppose we have too many classifiers (M) to compute $B_M(\epsilon)$ in reasonable time. We can still derive some benefit from the improved bounds as follows. For simplicity, let k divide M. Let $G_1 = \{g_1, \ldots, g_k\}, \ldots, G_{\frac{M}{k}} = \{g_{M-k+1}, \ldots, g_M\}$. Use (16) on each set of k classifiers.

$$\forall i \in \{1, \ldots, \frac{M}{k}\} \ \Pr\{\exists g_m \in G_i | \nu'_m \geq \nu_m + \epsilon\} \leq B_k(\epsilon) \tag{19}$$

Now consider the probability of uniform bound failure.

$$\Pr\{\exists g_m \in \{g_1, \ldots, g_M\} | \nu'_m \geq \nu_m + \epsilon\} \tag{20}$$

A single-classifier bound failure in $\{g_1, \ldots, g_M\}$ represents a failure in some subset $G_1, \ldots, G_{\frac{M}{k}}$.

$$= \Pr\{(\exists g_m \in G_1 | \nu'_m \geq \nu_m + \epsilon) \text{ or } \ldots \text{ or } (\exists g_m \in G_{\frac{M}{k}} | \nu'_m \geq \nu_m + \epsilon)\} \tag{21}$$

Bound the probability of the union of events by the sum of event probabilities.

$$\leq \Pr\{\exists g_m \in G_1 | \nu'_m \geq \nu_m + \epsilon\} + \ldots + \Pr\{\exists g_m \in G_{\frac{M}{k}} | \nu'_m \geq \nu_m + \epsilon\} \leq \frac{M}{k} B_k(\epsilon) \tag{22}$$

To find a confidence level for success of the uniform bound, subtract $\frac{M}{k} B_k(\epsilon)$ from one to bound the probability of the complement of uniform bound failure.

$$\Pr\{\forall g_m \in \{g_1, \ldots, g_M\} | \nu'_m < \nu_m + \epsilon\} \geq 1 - \frac{M}{k} B_k(\epsilon) \tag{23}$$

Bound (23) combines the methods of bounds (16) and (9). As we increase classifier subset size k, we produce stronger bounds that require more computation. For every problem,

$$k > k' \Rightarrow \frac{M}{k} B_k(\epsilon) \leq \frac{M}{k'} B_{k'}(\epsilon) \tag{24}$$

However, the inequality is not strict in all cases.

For example, when $d = d'$, and the maximizing w for a single classifier in (3) is $\frac{n}{2}$, bounding by pairs of classifiers (k=2) does not produce a stronger bound than bounding classifiers individually (k=1), i.e., $B_2(\epsilon) = 2B_1(\epsilon)$. The reason is as follows. For the bound $B_2(\epsilon)$, consider the error distribution $\mathbf{w}$ in which each classifier has $\frac{n}{2}$ errors, and the classifiers always disagree: $w_\emptyset = 0, w_{\{1\}} = \frac{n}{2}, w_{\{2\}} = \frac{n}{2}, w_{\{1,2\}} = 0$. Each single-classifier bound $\nu'_m < \nu_m + \epsilon$ fails for $B_1(\epsilon)$ of the partitions since each classifier has the worst-case number of errors, i.e. $\frac{n}{2}$. This alone does not ensure $B_2(\epsilon) = 2B_1(\epsilon)$. For this, there must be no partition of the data into validation and test sets such that both single-classifier bounds fail. Indeed, the bounds never fail together. Since the classifiers always disagree, $\nu_2 = 1 - \nu_1$ and $\nu'_2 = 1 - \nu'_1$ for each partition. Hence, $\nu'_2 - \nu_2 = -(\nu'_1 - \nu_1)$. So if $\nu'_1 \geq \nu_1 + \epsilon$ then $\nu'_2 < \nu_2 + \epsilon$.

Using knowledge of the validation data and the test inputs to constrain $\mathbf{w}$ (as outlined in the previous section) reduces the necessary computation and can only improve the bounds. If we use this information for the mixed bound (23), then the partitioning of classifiers can affect the bound. As we see from the previous example, the uniform bound over a subset of classifiers is generally weaker when the classifiers within the subset have high rates of disagreement. Thus, it makes sense to place classifiers with high rates of agreement together in subsets.

To be more precise, let $\mathbf{a}|S$ be the pattern of agreements among the classifiers indexed by S, i.e., the projection of the pattern of agreements onto these classifiers. For example, $[\mathbf{a}|S]_{\{1,3\}}$ is the number of examples for which the classifiers indexed by the first and third elements of S return 1 and the other classifiers indexed by S return 0. Also, let $S_1, \ldots, S_P$ be a partition of $\{1, \ldots, M\}$. Then, by a derivation similar to that used for (23),

$$\Pr\{\forall g_m \in \{g_1, \ldots, g_M\} | \nu'_m < \nu_m + \epsilon\} \geq 1 - [B(\epsilon, \mathbf{a}|S_1) + \ldots + B(\epsilon, \mathbf{a}|S_P)] \quad (25)$$

where $B(\epsilon, \mathbf{a})$ is as defined in the previous section. The bound holds for every partition $S_1, \ldots, S_P$. Since $B(\epsilon, \mathbf{a}|S)$ depends on subset S, the confidence of the bound depends on the partition used to compute it.

5 Tests

This section presents the results of tests using the improved uniform bounds outlined in the development of (25). The tests were performed on credit card data. Each example corresponds to a credit card user. There are six inputs that correspond to user traits. The traits are unknown because the data provider has chosen to keep them secret. There is a single output that indicates whether or not the credit card user defaulted. The data were obtained from a machine-learning database site at the University of California at Irvine. The discrete-valued traits were removed, leaving the six continous-valued traits. Of the 690 examples in the original database, 24 examples had at least one trait missing. These examples were removed, leaving 666 examples. The data were cleaned by Joseph Sill. For further information, see [9].

The classifiers are artificial neural networks with six input units, six hidden units, and one output unit. The hidden and output units have tanh activation functions. The initial weights were selected independently and uniformly at random from $[-0.1, 0.1]$. The networks were trained by gradient descent on mean squared error over training examples, using sequential mode weight updates with random order of example presentation in each epoch. In all tests, the classifiers are trained for 1000 epochs.

5.1 Classifiers From a Training Sequence

The first set of tests focuses on obtaining uniform bounds over classifiers drawn from a training sequence. These tests apply to validation of the classifier chosen by early stopping using the method of central classifiers [4]. In this scheme, a snapshot of the classifier is recorded after each epoch of training. These snapshots are sampled at intervals, forming a set of central classifiers. Uniform error bounds are computed for the central classifiers. Then, the validation error is computed for all snapshots, and the snapshot with minimum validation error is delivered as the result of training. The test error of this classifier is bounded by reference to a central classifier using the fact that the difference in error rates between the chosen snapshot and the central classifier can be no greater than the rate of disagreement between the two classifiers. The confidence level for the bound by reference is the confidence level for the uniform bound over central classifiers. By using relatively few central classifiers, a higher confidence level is achieved than with uniform bounds over all snapshots. Our goal in these tests is to obtain uniform bounds over the central classifiers with high confidence.

In each of the 10 tests, the 666 examples were randomly partitioned into 444 training examples, $d = 111$ validation examples, and $d' = 111$ test examples. Over 1000 epochs of training, a central classifier was recorded after each 100 epochs, making 10 central classifiers.

Uniform bounds over the 10 classifiers were computed by the following methods:

test	10×1 classifier	5×2 classifiers	3,3,2,2 classifiers
1	92.25	95.11	95.53
2	92.25	95.49	96.11
3	92.25	95.59	96.12
4	92.25	95.21	95.61
5	92.25	95.48	95.94
6	92.25	95.28	95.74
7	92.25	95.36	95.81
8	92.25	95.58	96.10
9	92.25	94.88	95.48
10	92.25	95.76	96.17
avg	92.25	95.37	95.86

Table 1: Levels of confidence for uniform bounds over 10 classifiers from a training sequence. Results are shown for 10 tests and for three bound methods – VC-style, mixed bounds with partitions into classifier pairs, and mixed bounds with partitions into sets of two and three classifiers.

- VC-style bounds (9), with confidence $1 - 10B(\epsilon)$.
- Mixed bound with test input constraints (25), using pairwise partitioning of the classifiers by sequence to obtain confidence

$$1 - [B(\epsilon, \mathbf{a}|\{1,2\}) + \ldots + B(\epsilon, \mathbf{a}|\{9,10\})] \tag{26}$$

- Mixed bound with test input constraints (25), partitioning in sequence into two sets of three classifiers followed by two sets of two classifiers, to obtain confidence

$$1 - [B(\epsilon, \mathbf{a}|\{1,2,3\}) + B(\epsilon, \mathbf{a}|\{4,5,6\}) + B(\epsilon, \mathbf{a}|\{7,8\}) + B(\epsilon, \mathbf{a}|\{9,10\})] \tag{27}$$

The value of ϵ was chosen to be the minimum value (to the nearest thousandth) that gives at least 90% confidence for the VC-style bound. This value is 0.163, and it gives confidence 92.25%, i.e., $B(\epsilon) = 0.775\%$. Since a uniform bound over more than one classifier can have no more confidence than the bound for a single classifier, $B(\epsilon, \mathbf{a}|S) \leq B(\epsilon)$ for all $\mathbf{a}|S$. Hence, the best possible confidence level for the first mixed bound is $1 - 5B(\epsilon) = 96.125\%$. For the second bound, it is $1 - 4B(\epsilon) = 96.9\%$.

The test results are shown in Table 1. Note that the confidence levels increase as we move through the three bound methods from the single-classifier-based bound to those based on larger subsets of classifiers. In other words, more computation buys better bounds.

test	10 × 1 classifier	5 × 2 classifiers	3,3,2,2 classifiers
1	92.25	94.59	95.10
2	92.25	94.67	94.91
3	92.25	94.65	94.90
4	92.25	94.53	95.10
5	92.25	94.31	94.88
6	92.25	94.68	95.10
7	92.25	94.10	94.52
8	92.25	94.49	94.60
9	92.25	94.63	95.10
10	92.25	94.49	94.98
avg	92.25	94.51	94.92

Table 2: Levels of confidence for uniform bounds over 10 separately trained classifiers. Results are shown for 10 tests and for three bound methods – VC-style, mixed bounds with partitions into classifier pairs, and mixed bounds with partitions into sets of two and three classifiers.

5.2 Separately Trained Classifiers

The second set of tests focuses on obtaining uniform bounds over classifiers that are trained separately. These uniform bounds can be used to select a classifier or to bound the test error of a voting committee or some other stacked classifier[8, 10, 13] by reference [5].

In each of 10 tests, the 666 examples were randomly partitioned into 444 training examples, $d = 111$ validation examples, and $d' = 111$ test examples. In each test, 10 classifiers were trained using early stopping. For each classifier, the training data were partitioned into 400 examples for actual training and 44 examples for early stopping. A snapshot was recorded after each epoch. The snapshot with minimum error on the 44 examples was returned as the trained classifier.

Uniform bounds over the 10 trained classifiers were computed using the same three methods used in the first set of tests. Once again, ϵ was set to 0.163. The test results are shown in Table 2. As in the first set of tests, the confidence levels increase as we move to more computationally intensive bound methods.

6 Discussion

We have derived distribution-free uniform test error bounds that improve on VC-style bounds for validation, and we have shown how to use knowledge of test inputs to strengthen the bounds and reduce computation. We have developed mixed bounds that trade sharpness for reduction in computation. Through tests on credit card data, we have shown that these bounds are effective for applications in the real world. This work presents several opportunities for further inquiry, including extension to the full VC framework, analysis with the goal of reducing computation, and extension to nonuniform bounds.

This paper used the validation framework, in which validation error is used to uniformly bound test error over a finite set of classifiers. In the full VC framework, training error is used to uniformly bound test error over a possibly infinite class of classifiers. To use the results of this paper in the VC framework, identify validation data here with training data in the VC framework. Also, identify the set of classifiers here with a subset of classifiers in the class such that there is some representative classifier for each dichotomy that the class can produce over the training and test examples. To derive distribution-free bounds, use the bound for the worst-case arrangement of training and test inputs (as in the definition of the growth function [12].) For details, see [12, 6]. For any but the smallest problems, direct use of the improved bounds is computationally infeasible. However, the mixed bounds prove useful. In [6], the representative classifiers are partitioned into small sets of classifiers with few disagreements. The mixed bounds over these partitions improve VC bounds.

The computation required for the improved bounds restricts their utility. There are several analytic and algorithmic approaches that could yield computational reductions. The bound (16) is defined in terms of the worst-case distribution $\mathbf{w}$. Through analysis, it may be possible to identify the worst-case error distribution for given values of ϵ and numbers of validation examples, test examples, and classifiers. For example, for large enough ϵ and validation and test sets of equal size, it is known that the worst-case $\mathbf{w}$ for a single classifier is $w_{\emptyset} = \frac{n}{2}$ and $w_{\{1\}} = \frac{n}{2}$ [12].

It would also be useful to identify the worst-case distributions for the bound (17) with restrictions imposed by knowledge of the test inputs. It may be relatively easy to show that the worst-case $\mathbf{w}$ has

$$w_S = \frac{1}{2}(a_S + a_{\overline{S}}) \text{ and } w_{\overline{S}} = \frac{1}{2}(a_S + a_{\overline{S}}) \tag{28}$$

for some conditions on ϵ and equal numbers of validation and test examples.

Even if closed-form solutions for the worst-case $\mathbf{w}$ cannot be found, it may be possible to prove some properties of the bound given $\mathbf{w}$ (15) that allow more efficient identification of the worst-case $\mathbf{w}$. For example, if it can be shown that all local maxima of the bound given $\mathbf{w}$ occur for worst-case $\mathbf{w}$'s, then the worst-case $\mathbf{w}$ can be identified through gradient descent instead of the present exhaustive search.

Finally, it would be useful to extend the improved uniform bound to interesting non-uniform bounds. For example, if we will use only a classifier with minimum validation error, then we are not really interested in the bounds on the other classifiers. In this case, we wish to compute

$$\Pr\{\forall g_m \text{ with minimum } \nu_m | \nu'_m \geq \nu_m + \epsilon\} \tag{29}$$

The derivation should be similar to the derivation of the uniform bound in this paper. The confidence will be at least as great as the confidence for the uniform bound. It would be interesting to observe the difference in confidence levels. For a given partition, the classifiers with minimum validation error are more likely than the average classifier to have an unusually small number of their error examples in the validation set. In this case, these classifiers have an unusually large number of error examples in the test set. Hence, these classifiers are more likely to be in violation of their bounds than a classifier chosen at random.

7 Acknowledgements

I thank Dr. Joel Franklin for his teaching, advice, and encouragement. I thank Zehra Cataltepe, Sam Roweis, and Joe Sill for their many helpful conversations and pointers to literature in the development of this work. Also, thanks to Dr. Yaser Abu-Mostafa for his informative and inspiring teaching.

References

[1] Abu-Mostafa, Y.S. (1996). What you need to know about the VC inequality. Class notes from CS156, California Institute of Technology.

[2] Abu-Mostafa, Y.S. (1989). The Vapnik-Chervonenkis dimension: information versus complexity in learning. *Neural Computation*, 1 (3), 312-317.

[3] Baum, E. B., and Haussler, D. (1989). What size net gives valid generalization? *Neural Computation*, 1 (1), 151-160.

[4] Bax, E., Cataltepe, Z., and Sill, J. (1997). Alternative error bounds for the classifier chosen by early stopping. CalTech-CS-TR-97-08.

[5] Bax, E. (1997). Validation of voting committees. CalTech-CS-TR-97-13.

[6] Bax, E. (1997). Similar classifiers and VC error bounds. CalTech-CS-TR-97-14.

[7] Blumer, A., Ehrenfeucht, A., and Haussler, D. (1989). Learnability and the Vapnik-Chervonenkis dimension. *Journal of the Association for Computing Machinery*, 36 (4), 929-965.

[8] Breiman, L. (1992). Stacked regressions. Tech. Rep. No. 367, Statistics Dept., Univ. of California at Berkeley.

[9] Sill, J. and Abu-Mostafa, Y. (1997). Monotonicity hints. to appear in *Advances in Neural Information Processing Systems, 9.*.

[10] Sridhar, D. V., Seagrave, R. C., and Bartlett, E. B. (1996). Process modeling using stacked neural networks. *AIChE Journal*, 42(9) 2529-2539.

[11] Vapnik, V. N. (1982). *Estimation of Dependences Based on Empirical Data* p.31. Springer-Verlag New York, Inc.

[12] Vapnik, V. N. and Chervonenkis, A. (1971). On the uniform convergence of relative frequencies of events to their probabilities. *Theory Prob. Appl.*. 16: 264-280.

[13] Wolpert, D. H. (1992). Stacked generalization. *Neural Networks*. 5: 241-259.