

---

# Generalization Bounds for Binary Classification with Applications to Statistical Verification of Complex Controllers

---

Vu Ha      Tariq Samad

Honeywell International

3660 Technology Dr.

Minneapolis, MN 55418

{vu.ha,tariq.samad}@honeywell.com

## Abstract

Statistical learning theory (SLT), pioneered by Vapnik & Chervonenkis (1971), concerns the problem of choosing desired functions on the basis of empirical data. A central issue in SLT is the derivation of Vapnik-Chervonenkis (VC) style generalization bounds and sample complexities. While the SLT literature contains a vast collection of such results, somewhat surprisingly it contains almost no results regarding the problem of learning a binary classifier with weighted misclassification error. In this paper we aim to fill this gap. We show how existing bounds for unweighted binary classification can be extended to weighted binary classification, especially when false negatives have zero penalty. These bounds are an order-of-magnitude sharper than Vapnik's basic result. We demonstrate the practical relevance of these results by presenting an SLT approach to verifying computational properties of complex controller algorithms.

## 1 INTRODUCTION

In this paper we are interested in deriving VC style generalization bounds and sample complexities for binary classification where the two types of misclassifications – false negatives and false positives are assigned different losses.

The problem of learning a binary classifier based on a data sample is a special case of learning a general function in the SLT framework. Let  $X$  and  $Y$  be non-empty sets that will be called the *input* and *output space* respectively, and  $F(x, y)$  be a fixed, unknown probability function<sup>1</sup> on  $Z = X \times Y$ . A *training set*

is a finite sample  $S_n = \{(x_i, y_i)\}_{i=1}^n \in Z^n$  drawn independently according to  $F$ . A *hypothesis space* is a set  $H \subset Y^X$  of functions from  $X$  to  $Y$ . A *loss function* is a real-valued function  $L : Y^2 \rightarrow \mathbb{R}$ . The *expected* or *real loss* of a hypothesis  $h \in H$  is defined as:  $L(h) = \int L(h(x), y) dF(x, y)$ . The informal goal of the learning problem is

*to find  $h$  with small expected loss, based on the training sample  $S_n$ .*

There are several well-known special cases of the general learning problem. An important special case is where the conditional distribution  $F(y|x)$  is degenerate, corresponding to a function  $f : X \rightarrow Y$ . Function  $f$  is called the *target function*. When  $Y$  is finite, we have a *classification* problem. When  $|Y| = 2$ , we have a *binary classification* problem, and the two elements of  $Y$  are denoted as  $\{-1, 1\}$ . A frequent choice of  $L$  for classification problems is the *misclassification error*:

$$L(y, y') = \mathbb{I}(y \neq y') = \begin{cases} 0 & \text{if } y = y' \\ 1 & \text{if } y \neq y'. \end{cases}$$

In binary classification, sometimes we wish to use the *weighted classification error*:

$$L(y, y') = \begin{cases} 0 & \text{if } y = y' \\ \rho & \text{if } y = -1, y' = 1 \text{ (false negative)} \\ 1 & \text{if } y = 1, y' = -1 \text{ (false positive)}. \end{cases}$$

Here  $\rho$  is a number between 0 and 1. The idea is that false positive misclassifications are more costly than false negative ones. In  $m$ -class classification, we can generalize this idea by defining a  $m \times m$  classification loss matrix.

In this paper, we will assume that the loss function has range  $[0, 1]$ . This assumption clearly holds for weighted binary classification.

sition rigorous by ignoring measurability and integrability issues. This simplification is often adapted in presenting SLT to a computer scientist audience.

---

<sup>1</sup>For the sake of simplicity, we will not make the expo-

## 2 VC ANALYSIS

How should one go about finding a hypothesis with small expected loss? This question initially appears impossibly difficult, especially since no assumption is made about the sampling distribution  $F$ . SLT reformulates the informal goal of learning as:

*to find  $h$  with a small bound on its expected loss, based on the training sample  $S_n$ .*

The crucial modification lies in the phrase “small bound”. In the classical VC analysis, the expected loss of a hypothesis  $h \in H$  is bounded using the sum of two quantities. The first quantity is the *empirical loss* of  $h$  on  $S_n$ :

$$L_n(h) = L(h, S_n) = \frac{1}{n} \sum_{i=1}^n L(h(x_i), y_i).$$

The second quantity depends on some measure of complexity of the hypothesis space  $H$  from which  $h$  is selected, and is referred to as the *VC confidence*. Formally, let  $\epsilon$  and  $\delta$  be small ( $< 1$ ) positive numbers, we will be interested in inequalities of the form

$$P \left\{ \sup_{h \in H} (L(h) - L_n(h)) < \epsilon \right\} \geq 1 - \delta,$$

or succinctly:  $\mathcal{P}_\delta : L(h) < L_n(h) + \epsilon$ . (2.1)

Such an inequality provides an upper bound ( $L_n(h) + \epsilon$ ) with confidence  $1 - \delta$  for the expected loss of *any* hypothesis  $h \in H$ . The classical approach is thus based on a *worst-case* analysis. Typically, there is some functional relationship between  $\epsilon$ ,  $\delta$ ,  $n$ , and a measure of complexity of  $H$ . There are several quantities that can serve as a complexity measure of  $H$ , the most well-known of which is the so-called *VC dimension*. We will give a precise definition of the VC dimension later, but throughout the paper we will use  $d$  to denote the VC dimension of  $H$ .

### 2.1 VAPNIK’S GENERAL BOUND

When  $H$  contains a single hypothesis, 2.1 becomes the problem of *convergence in probability*, of bounding the discrepancy between the expectation and the empirical mean of some random function, to which large deviation inequalities such as Chebyshev’s, Chernoff’s, Hoeffding’s, Bernstein’s, and McDiarmid’s can be applied (see (McDiarmid, 1997) for a recent survey of these inequalities). In the general case, the problem is one of *uniform convergence in probability*. The word “uniform” is a consequence of the supremum in 2.1. The classical approach employs a set of techniques to reduce the general case to the special case. The following result is an example of such results.

**Theorem 2.1 (Vapnik’s General Bound (Vapnik, 2000, Equation 3.26)).**

$$\mathcal{P}_\delta : L(h) < L_n(h) + \sqrt{\frac{1}{n} \left( d \ln \frac{2en}{d} + \ln \frac{4}{\delta} \right)}.$$

### 2.2 BOUNDS FOR UNWEIGHTED BINARY CLASSIFICATION

VC analysis for binary classification has been the focus of much research in statistical learning theory, and more recently, in the theory of learnability (Blumer et al., 1989; Anthony & Shawe-Taylor, 1993). But to our knowledge, heretofore all of the work in this area assumes that the classification error is *unweighted*. In this section we describe the main results on generalization bounds in unweighted binary classification.

When the empirical loss  $L_n(h)$  is zero – in which case we call  $h$  *consistent* – it is possible to derive sharper bounds, as the following well-known result shows.

**Theorem 2.2 (Consistent Unweighted Binary Classifiers (Blumer et al., 1989)).**

$$L_n(h) = 0 \Rightarrow \mathcal{P}_\delta : L(h) < \frac{2}{n} \left( d \log \frac{2en}{d} + \log \frac{2}{\delta} \right).$$

The requirement that  $h$  is consistent is quite restrictive, especially for unweighted binary classification. If it is possible to achieve small, but non-zero empirical loss, the following result can be used instead.

**Theorem 2.3 (Unweighted Binary Classifiers with Small Empirical Loss (Anthony & Shawe-Taylor, 1993)).**

$$\mathcal{P}_\delta : L(h) < 2L_n(h) + \frac{4}{n} \left( d \ln \frac{2en}{d} + \ln \frac{4}{\delta} \right).$$

## 3 VC ANALYSIS FOR WEIGHTED BINARY CLASSIFICATION

Practically nothing is known about generalization bounds for weighted binary classification. Granted, Theorem 2.1 makes no assumption about the loss function  $L$ , and thus can be applied for weighted binary classification. Even then, there is one subtlety with Theorem 2.1 in the definition of the VC dimension. It is important to note that the VC dimension of a hypothesis space  $H$  *depends* on the loss function  $L$ . In (Vapnik, 2000), the concept of VC dimension is first defined for the case of *unweighted* binary classification, and then generalized to the case of arbitrary loss function  $L$ . It is not immediately clear from the existing literature how the VC dimension for unweighted binary classification is related to the VC dimension for weighted binary classification. We offer the following observation to address this issue.

**Proposition 3.1.** *In weighted binary classification, the VC dimension, as defined in (Vapnik, 2000) does not depend on the parameter  $\rho$ .*

As a consequence, we can apply Theorem 2.1 to weighted binary classification using the familiar definition of VC dimension. The problem is, the VC confidence term in Theorem 2.1 is  $O(\sqrt{\ln n/n})$ , significantly worse than  $O(\log n/n)$  in Theorems 2.2 and 2.3. The question is then: can we extend Theorems 2.2 and 2.3 to the weighted binary classification problem? This is the motivation of our work, and we shall show that the answer is affirmative when  $\rho = 0$ .

### 3.1 CONSISTENT BINARY CLASSIFIERS

We first consider consistent binary classifiers, and thus statements of the form

$$\mathcal{P}_\delta : \sup_{h \in H, L_n(h)=0} L(h) < \epsilon.$$

The approach taken here is similar to that described in Vidyasagar (2003), with extensions to include weighted binary loss.

The main difficulty in deriving inequalities of the form 2.1 is the unknown quantity  $L(h)$ , as we make no assumptions about the distribution function  $F$ . The first step in the VC analysis is to eliminate  $L(h)$ . It is referred to as *symmetrization step*.

#### The Symmetrization Argument

Let  $S'_n$  be a *ghost sample*, i.e. a training set of size  $n$  and independent of  $S_n$ , and  $L'_n(h) = L(S'_n, h)$ . We will prove that

$$\begin{aligned} pP\left\{ \sup_{h: L_n(h)=0} L(h) > \epsilon \right\} \\ \leq P\left\{ \sup_{h: L_n(h)=0} L'_n(h) > \frac{\epsilon}{2} \right\}, \end{aligned} \quad (3.1)$$

for some  $p \in (0, 1)$ . The intuition here is that if the empirical losses on two independent training sets are close, then they should also be close to the expected loss. Let

$$\begin{aligned} Q &= \left\{ S_n : \sup_{L_n(h)=0} L(h) > \epsilon \right\}, \\ R &= \left\{ S_n, S'_n : \sup_{L_n(h)=0} L'_n(h) > \frac{\epsilon}{2} \right\}. \end{aligned}$$

Then 3.1 becomes  $pP(Q) \leq P(R)$ , and we can set  $p = P(R|Q)$ . To derive a lower bound on  $p$ , fix  $h$  and

$S_n$  such that  $L_n(h) = 0, L(h) > \epsilon$ , and observe that

$$\begin{aligned} 1 - p &\leq P\{L'_n(h) \leq \epsilon/2\} \\ &\leq P\{L'_n(h) - L(h) < -\epsilon/2\} \\ \Rightarrow p &\geq P\{L'_n(h) \geq L(h) - \epsilon/2\} =: \alpha_n(\epsilon). \end{aligned} \quad (3.2)$$

We will derive further lower bounds on  $\alpha_n(\epsilon)$  later; for now let us remember that  $\alpha_n(\epsilon)$  is the probability that the empirical loss (of size  $n$ ) is large (at least the real loss minus  $\epsilon/2$ ).

#### The Covering Argument

We now bound  $P(R)$ . This step is referred to as the *covering argument*. Let

$$\begin{aligned} H_{2n} &= \{(h(x_i))_{i=1}^{2n} | h \in H, x_i \in X, i = 1, \dots, 2n\} \\ &\subseteq \{-1, 1\}^{2n}. \end{aligned}$$

Elements of  $H_{2n}$  are denoted as  $(h_n, h'_n)$ , where  $h_n \in \{-1, 1\}^n$  are the first  $n$  coordinates, and  $h'_n \in \{-1, 1\}^n$  are the second  $n$  coordinates. We then write  $L_n(h)$  as  $L(h_n)$ , and  $L'_n(h)$  as  $L(h'_n)$ . Now  $P(R) = P(R')$  where

$$R' = \left\{ \exists (h_n, h'_n) \in H_{2n} : L(h_n) = 0, L(h'_n) > \frac{\epsilon}{2} \right\}.$$

Let  $\Sigma_n = \{\sigma : \{1, \dots, n\} \rightarrow \{0, 1\}\}$ , and thus  $|\Sigma_n| = 2^n$ . Each function  $\sigma$  can be viewed as a bijective mapping between a pair  $(S_n, S'_n) \in Z^n \times Z^n$  and another pair  $(\sigma S_n, \sigma S'_n) \in Z^n \times Z^n$  so that

$$\begin{aligned} \sigma(i) = 1 &\Rightarrow \sigma S_n(i) = S'_n(i) \wedge \sigma S'_n(i) = S_n(i), \\ \sigma(i) = 0 &\Rightarrow \sigma S_n(i) = S_n(i) \wedge \sigma S'_n(i) = S'_n(i). \end{aligned}$$

In words, each  $n$ -bit 0/1 vector  $\sigma$  encodes the interchanging of the  $n$  coordinates of  $S_n$  and  $S'_n$ . Similarly, denote by  $(\sigma h_n, \sigma h'_n)$  the interchanging of coordinates of  $(h_n, h'_n)$  according to  $\sigma$ . We write  $L(\sigma h_n, \sigma S_n)$  as  $L(\sigma h_n)$  and  $L(\sigma h'_n, \sigma S'_n)$  as  $L(\sigma h'_n)$ . Finally, let

$$\begin{aligned} \sigma R' &= \left\{ L(\sigma h_n) = 0, L(\sigma h'_n) > \frac{\epsilon}{2} \right\}, \\ \sigma R' &= \left\{ \exists (h_n, h'_n) \in H_{2n} : \sigma R'(h_n, h'_n) \right\}. \end{aligned}$$

Since  $\sigma$  merely relabels the samples in  $S_n$  and  $S'_n$ , we have

$$\begin{aligned} P(R') &= P(\sigma R') = \frac{1}{2^n} \sum_{\sigma \in \Sigma_n} P\{\sigma R'\} \\ &= \frac{1}{2^n} \int_{Z^{2n}} \sum_{\sigma \in \Sigma_n} \mathbb{I}\{\sigma R'\} dS_n dS'_n \\ &\leq \frac{1}{2^n} \int_{Z^{2n}} \sum_{\sigma \in \Sigma_n} \sum_{(h_n, h'_n) \in H_{2n}} \mathbb{I}\{\sigma R'(h_n, h'_n)\} dS_n dS'_n \\ &= \frac{1}{2^n} \int_{Z^{2n}} \sum_{(h_n, h'_n) \in H_{2n}} \sum_{\sigma \in \Sigma_n} \mathbb{I}\{\sigma R'(h_n, h'_n)\} dS_n dS'_n. \end{aligned} \quad (3.3)$$

In the above,  $\mathbb{I}$  denotes the indicator function. We next bound the integrand in 3.3. To bound the inside sum, we proceed as follows. Note that for any fixed realizations of  $(S_n, S'_n)$  and  $(h_n, h'_n)$ , if  $L(h'_n, S'_n) > \epsilon/2$ , then there are more than  $n\epsilon/2$  coordinates of  $S'_n$  that have positive loss. These coordinates can not be interchanged with  $S_n$  if we want to keep  $L(h_n, S_n) = 0$ . Consequently, there are less than  $n - n\epsilon/2$  coordinates of  $S'_n$  that can be interchanged with  $S_n$  while still keeping  $L(h_n, S_n) = 0$ . Thus, the inside sum in the integrand in 3.3 is at most  $2^{n-n\epsilon/2}$ .

To bound the outside sum in the integrand in 3.3, we need to bound the cardinality of  $H_{2n}$ . The quantity  $|H_t|$  is called the *growth function* of  $H$ . Note that  $H_t$  is the set of all  $-1/1$  vectors of length  $t$  that can be realized using the hypotheses in  $H$ , and thus  $|H_t| \leq 2^t, \forall t > 0$ . If  $|H_t| = 2^t, \forall t > 0$ , we say that  $H$  has *infinite VC dimension*. Otherwise, the greatest integer  $t$  that satisfies  $|H_t| = 2^t$  is called the *VC dimension* of  $H$ . In this paper we assume that  $H$  has finite VC dimension  $d$ . Sauer-Shelah lemma (Sauer, 1972; Shelah, 1972) provides a bound on the growth function based on the VC dimension.

**Sauer-Shelah Lemma.**  $|H_n| \leq (en/d)^d$ .

For proofs of this lemma, see, for example, (Vapnik, 1998; Vidyasagar, 2003). Thus the integrand in 3.3 is less than  $(2en/d)^d 2^{n-n\epsilon/2}$ , and hence  $P(R) \leq (2en/d)^d 2^{-n\epsilon/2}$ . Combining this with 3.2 and 3.1, we obtain the following result.

**Lemma 3.2 (Consistent Weighted Binary Classifiers).** *Any consistent hypothesis  $h$  ( $L_n(h) = 0$ ) has expected loss bounded by  $\epsilon$  with probability at least*

$$1 - \alpha_n(\epsilon)^{-1} \left( \frac{2en}{d} \right)^d 2^{-n\epsilon/2}.$$

It remains to estimate  $\alpha_n(\epsilon)$  from the left. In the case when  $\rho$  is either 0 or 1, we can use Lemma 3.6:  $\alpha_n(\epsilon) \geq 1/2$ , provided that  $n\epsilon > 2$ . This leads to the following result which extends Theorem 2.2 to include the case when false negatives have zero penalty.

**Theorem 3.3 (Consistent Weighted Binary Classifiers,  $\rho = 0, 1$ ).**

$$L_n(h) = 0 \Rightarrow \mathcal{P}_\delta : L(h) \leq \frac{2}{n} \left( d \log \frac{2en}{d} + \log \frac{2}{\delta} \right).$$

### 3.2 BINARY CLASSIFIERS WITH SMALL EMPIRICAL LOSS

When zero empirical loss is achievable, Theorem 3.3 provides a sharp (up to a logarithmic factor for  $\rho = 1$  (Ehrenfeucht & Haussler, 1989)) bound on the VC confidence (and hence the expected loss). The key is the

combinatorial argument in bounding the inner sum in 3.3. This argument is not extensible to the case where the empirical loss is non-zero. To derive bounds for the case the empirical loss is small, say  $\epsilon_n > 0$ , we need a different approach. The approach we describe in this section is based on that of Anthony & Shawe-Taylor (1993), with extensions to include weighted binary loss. We will be interested in bounding

$$P\left\{ \sup_{L_n(h) \leq \epsilon_n} L(h) > \epsilon \right\}, \text{ where } \epsilon > \epsilon_n.$$

Let  $\epsilon_n/\epsilon < \lambda \leq 1$ . We begin with a similar symmetrization argument to that in Section 3.1. Let

$$Q = \left\{ S_n : \sup_{L_n(h) \leq \epsilon_n} L(h) > \epsilon \right\},$$

$$R = \left\{ S_n, S'_n : \sup_{L_n(h) \leq \epsilon_n} L'_n(h) > \lambda\epsilon \right\},$$

and we proceed to upper bound  $P(R)$  and lower bound  $P(R|Q) =: \beta_n(\lambda, \epsilon)$ . We will derive lower bounds for  $\beta_n(\lambda, \epsilon)$  later; for now it suffices to remember that it is the probability that the empirical loss (of size  $n$ ) is at least  $\lambda\epsilon$ , given that the real loss is at least  $\epsilon$ .

To derive an upper bound on  $P(R)$ , note that if  $L'_n(h) > \lambda\epsilon > \epsilon_n \geq L_n(h)$ , then

$$\frac{L'_n(h) - L_n(h)}{\sqrt{L'_n(h) + L_n(h)}} > \frac{\lambda\epsilon - \epsilon_n}{\sqrt{\lambda\epsilon + \epsilon_n}}.$$

This is a consequence of the easily checked fact that function  $f(x) = (x - \epsilon_n)/\sqrt{x + \epsilon_n}$  is monotonically increasing in  $(\epsilon_n, \infty)$ . Thus  $P(R) \leq P(R')$ , where

$$R' = \left\{ S_n, S'_n : \exists h : \frac{L'_n(h) - L_n(h)}{\sqrt{L'_n(h) + L_n(h)}} > \eta \right\},$$

and  $\eta = \frac{\lambda\epsilon - \epsilon_n}{\sqrt{\lambda\epsilon + \epsilon_n}}$ . Now, proceed analogously to the permutational covering argument in the case of zero empirical loss (see 3.3), we have

$$P(R') \leq \int_{Z^{2n}} \sum_{\sigma \in \Sigma_n} \sum_{(h_n, h'_n) \in H_{2n}} \frac{1}{2^n} \mathbb{I} \left\{ \frac{L'(\sigma h_n) - L(\sigma h_n)}{\sqrt{L'(\sigma h_n) + L(\sigma h_n)}} > \eta \right\} dS_n dS'_n. \quad (3.4)$$

Denote  $l_i = L(h(x_i), y_i)$  and  $l'_i = L(h(x'_i), y'_i)$ , and define the independent random variables  $Y_i, i = 1 \dots n$ , such that

$$P\{Y_i = l_i - l'_i\} = P\{Y_i = l'_i - l_i\} = \frac{1}{2}.$$

The inner sum in the integrand in 3.4 becomes

$$P \left( \frac{1}{n} \sum_{i=1}^n Y_i > \eta \sqrt{\frac{1}{n} \sum_{i=1}^n (l_i + l'_i)} \right)$$

which, by Hoeffding's inequality, is bounded by

$$\exp\left(-\frac{n\eta^2 \sum_{i=1}^n (l_i + l'_i)}{2 \sum_{i=1}^n (l_i - l'_i)^2}\right) \leq \exp\left(-\frac{n\eta^2}{2}\right),$$

since  $l_i, l'_i \in [0, 1]$ . Again using the Sauer-Shelah lemma to bound the outer sum, we have

$$P(R) \leq P(R') \leq \left(\frac{2en}{d}\right)^d \exp\left(-\frac{n(\lambda\epsilon - \epsilon_n)^2}{2(\lambda\epsilon + \epsilon_n)}\right).$$

Putting everything together, we have the following general result.

**Lemma 3.4 (Weighted Binary Classifiers with Small Empirical Loss).** *Any hypothesis  $h$  with  $L_n(h) \leq \epsilon_n$  has expected loss bounded by  $\epsilon$  with probability at least*

$$1 - \beta_n(\epsilon)^{-1} \left(\frac{2en}{d}\right)^d \exp\left(-\frac{n(\lambda\epsilon - \epsilon_n)^2}{2(\lambda\epsilon + \epsilon_n)}\right),$$

provided that  $\lambda \in (\epsilon_n/\epsilon, 1]$ .

In the case when  $\rho$  is 0 or 1, and  $\lambda = 1$ , Lemma 3.6, Equation 3.7 states that  $\beta_n(1, \epsilon) > 1/4$ , provided that  $n\epsilon > 1$ , leading to the following result, which supersedes Theorem 2.3.

**Theorem 3.5 (Weighted Binary Classifiers with Small Empirical Loss,  $\rho = 0, 1$ ).**

$$L_n(h) \leq \epsilon_n \Rightarrow \mathcal{P}_\delta : L(h) \leq \epsilon,$$

if  $\epsilon$  satisfies that

$$\frac{(\epsilon - \epsilon_n)^2}{\epsilon + \epsilon_n} \geq \frac{2}{n} \left( d \ln \frac{2en}{d} + \ln \frac{4}{\delta} \right).$$

In particular, we can take

$$\epsilon = 2\epsilon_n + \frac{4}{n} \left( d \ln \frac{2en}{d} + \ln \frac{4}{\delta} \right). \quad (3.5)$$

*Remark 3.1.* Note that the bound in Theorem 3.5 (Equation 3.5) is not in the form of Inequality 2.1, due to the factor two in front of  $\epsilon_n$ . This is the reason why this result is most useful for small  $\epsilon_n$ . The VC confidence term can be rewritten using base-two logarithm as

$$\frac{4 \ln 2}{n} \left( d \log \frac{2en}{d} + \log \frac{2}{\delta} + 1 \right),$$

which is roughly 38% larger than the VC confidence term in Theorem 3.3. This is due to the use of Hoeffding's inequality as opposed to the combinatorial argument in bounding the inner sums in 3.4 and 3.3.

### 3.3 THE ROLE OF FALSE NEGATIVE PENALTY

In this section we look at the differences between the unweighted binary classification case ( $\rho = 1$ ) and the weighted one ( $0 \leq \rho < 1$ ), and provide additional insight into why the existing bounds for the case  $\rho = 1$  can be carried over to the case  $\rho = 0$  without change. For the case when  $0 < \rho < 1$ , we discuss several ways to derive bounds.

First, let us note that the covering argument for bounding  $P(R)$  in both Theorem 3.3 and 3.5 does not make any assumption regarding  $\rho$ . The only place where  $\rho$  matters is in the symmetrization argument, in deriving lower bounds for  $\alpha_n(\epsilon)$  for Theorem 3.3 and  $\beta_n(\lambda, \epsilon)$  for Theorem 3.5. When  $\rho = 0, 1$ , the lower bounds for  $\alpha_n(\epsilon)$  and  $\beta_n(1, \epsilon)$  are obtained via the following result.

**Lemma 3.6.** *Let  $B(n, p)$  denotes the binomial random variable with parameters  $n$  and  $p$ . Then*

$$np > 2 \Rightarrow P\{B(n, p) > np/2\} > 1/2, \quad (3.6)$$

$$np > 1 \Rightarrow P\{B(n, p) > np\} > 1/4. \quad (3.7)$$

The crucial point is that in both cases when  $\rho = 0$  and  $\rho = 1$ , the random variable  $nL_n(h)$  is a binomial of the form  $B(n, L(h))$ ! This is no longer the case when  $\rho$  takes values in  $(0, 1)$ . For this case, we need to use other bounding methods. For example, we can bound  $\alpha_n(\epsilon)$  as

$$\begin{aligned} 1 - \alpha_n(\epsilon) &= P\{L_n(h) < L(h) - \epsilon/2\} \\ &\leq \exp(-n\epsilon^2/2) \quad (\text{Hoeffding's}) \\ \text{or} \\ &\leq \exp\left(-\frac{3n\epsilon^2}{4\epsilon + 3}\right) \quad (\text{Bernstein's}), \quad (3.8) \end{aligned}$$

with the latter bound being typically better (for  $\epsilon < 3/4$ ). When  $1/2 < \rho < 1$ , we can derive a sharper bound for  $p$  in 3.2 (instead of  $\alpha_n(\epsilon)$ ) as follows. Let  $L^*$  denote the *unweighted* loss function. We have that  $l = L^*(h) \geq L(h) > \epsilon$ ,  $L_n(h) \geq \rho L_n^*(h)$ , and thus

$$\begin{aligned} 1 - p &= \Pr\{L_n(h) \leq \epsilon/2\} \\ &\leq \Pr\left\{L_n^*(h) \leq \frac{\epsilon}{2\rho}\right\} \\ &\leq \Pr\left\{L_n^*(h) \leq \frac{l}{2\rho}\right\} \quad (\text{Multiplicative Chernoff's}) \\ &\leq \exp\left(-\left(1 - \frac{1}{2\rho}\right)^2 nl/2\right) \\ &\leq \exp\left(-\left(1 - \frac{1}{2\rho}\right)^2 n\epsilon/2\right), \quad (3.9) \end{aligned}$$

which is better than Bernstein’s bound (Equation 3.8) for  $\epsilon < 3(2\rho - 1)^2/(8\rho^2 + 16\rho - 4)$ .

As for bounding  $\beta_n(\lambda, \epsilon)$  when  $\rho \in (0, 1)$ , we can also use Hoeffding’s and Bernstein’s inequality, but only for the case  $\lambda < 1$ .

$$1 - \beta_n(\lambda, \epsilon) \leq P\{L_n(h) < L(h) - (1 - \lambda)\epsilon\} \\ \leq \exp(-n(1 - \lambda)^2\epsilon^2/2) \quad (\text{Hoeffding's}) \\ \text{or} \\ \leq \exp\left(-\frac{3n(1 - \lambda)^2\epsilon^2}{4(1 - \lambda)\epsilon + 3}\right) \quad (\text{Bernstein's}).$$

When  $\lambda = 1$ , none of the large deviation-type inequalities is applicable, and it seems like we can derive bounds only for the binomial tail ( $\rho = 0, 1$ ) and not for the general case ( $0 < \rho < 1$ ). It is worth noting that while Inequalities 3.6 and 3.7 are fairly well-known facts, we have not been able to find satisfactory proofs for them in the SLT literature. We have come up with proofs for these facts, based on case analysis. For Inequality 3.6, we can use 3.9 (with  $\rho = 1$ ) for the case  $np > 8 \ln 2 \approx 5.5$ , and a case analysis based on  $\lfloor np \rfloor$  for  $2 < np < 8 \ln 2$ . For Inequality 3.7, the proof centers on the fact that we only need to consider the case when  $np$  is an integer. Both of these two proofs are extremely long and tedious, and we omit the details here.<sup>2</sup>

## 4 APPLICATIONS

In this section we demonstrate the practical relevance of the case  $\rho = 0$ , and demonstrate the improvement of the bounds in Theorem 3.3 and 3.5 over Vapnik’s general bound in 2.1.

Our work in deriving VC bounds for weighted binary classification grew out of our interest in developing a statistical verification approach for complex controllers. The computational requirements for a complex control law, in a given implementation, can vary considerably over time. The variation arises because the control law calculation depends on a number of factors, such as the sensed (or estimated) state of the system under control, environmental disturbances, and the operational mode of the system. Since the control execution must complete in time for a command to be issued to the actuator at the next sample instant, algorithm implementations with unpredictable computing requirements can obviously not be accommodated in hard real-time systems. This is the reason that PID<sup>3</sup> controllers, with their deterministic execu-

<sup>2</sup>Ralf Herbrich, in a recent communication, has shown us a proof of 3.6, due to Mingrui Wu, that is shorter and more elegant.

<sup>3</sup>Proportional, Integral, Derivative.

tion time, are still the preferred choice in many applications despite their lesser performance.

In order to bring practical acceptance to high-performance, complex controllers, we need to provide guarantees about their computational properties. Our SLT-based verification approach is to determine the *safe operational envelope* within which the high-performance algorithm can reliably (with some statistical guarantees) terminate in the time allocated. Outside of this region, an alternative, lower performance and computationally simpler, controller could be used. The safe region is determined based on sampled data.

The verification problem now becomes the problem of learning a binary classifier from simulated data, where the two types of misclassifications obviously have vastly different consequences. A false negative merely implies a conservative use of a low performance controller, while a false positive would have drastic consequences such as loss of the vehicle. For this reason, we would like to obtain a safe region that has some statistical guarantee that it is indeed safe. In other words, we would like to find a classifier that has provably small probability of false positive.

One may at first be tempted to think that we can achieve this objective by sticking to the unweighted model: as long as we can ensure small probability of misclassification, we can also ensure small probability of false positive. This argument is however problematic. If the safe region has irregular shape and is difficult to approximate, it may not be possible to obtain small error bound, due to the so-called *bias-variance tradeoff*: If we make the empirical loss small by working with a complex  $H$ , we get penalized by a *large variance* – the VC confidence term. Reversely, if we keep a bound on the VC confidence term, there may not be an  $h \in H$  that fits the training data well enough, i.e., we are stuck with a *large bias*.

The weighted classification model with  $\rho = 0$  seems to be the perfect candidate to break this deadlock. We can control the error bound by controlling the VC confidence term and the empirical loss. The former is kept small by working with  $H$  with small VC dimension, while the latter can always be made zero using the trivial hypothesis that classifies everything as negative. While this hypothesis is not very useful, it is only a starting point. We can next try to expand this hypothesis in such a way that it admits increasingly more points as positive while still maintaining zero or a small number of false positives (depending on the use of Theorem 3.3 or Theorem 3.5) in the sampled data. Table 1 summarizes this process. Note that this procedure is fairly general; it does not tell you how to choose a hypothesis space ( $H$ ), or how to find a hy-

Table 1: SLT-based verification.

1. Define the inputs of the classification problem. This includes defining the input attributes, as well as the probability distribution ( $F$ ) that accurately describes how the inputs occur.
2. Determine the parameters  $\epsilon$ ,  $\delta$ , and the space of classifiers ( $H$ ). Compute  $d$ , the VC dimension of  $H$ .
3. Determine the lower bound for  $n$ , the sample complexity based on Theorem 3.3.
4. Simulate the execution of the controller algorithms for the computed number of samples.
5. Compute a hypothesis  $h \in H$  that has zero false positives, and as few false negatives on the generated samples as possible.

pothesis with no false positives and few false negatives. These are the decisions that the control engineer has to make on a case by case basis. Also, other variants are possible. For example, if the number of samples is limited, we may have to determine  $\epsilon$ ,  $\delta$ , and  $H$  based on this limitation.

We now describe how we applied the above procedure to statistically verifying the computational property of a high-performance controller for an OAV (Figure 1, left). The OAV has a ducted fan propulsion unit, with control provided by movable vanes in the propwash. The fact that the vanes are situated in the propulsion airflow results in significant nonlinear interactions between the propulsion and the control surfaces. The trim calculation for the OAV is an iterative algorithm whose computational time depends on several factors (Elgersma & Morton, 2000). We are interested in conditions under which this calculation can be reliably used. The more accurately the range of these conditions can be assessed, the greater the operational envelope for the vehicle. Figure 1 (right) depicts the dependence of complexity of equilibrium angle of attack computation on two factors: the flight path angle and the net lift force.

For our experiment, we identify four factors that affect the computational time of the iterative algorithm, and choose 4-dimensional hyper-rectangles as our hypothesis space. The VC dimension of this hypothesis space is 8.<sup>4</sup> Thus for  $\delta = .05$ ,  $\epsilon = .05$ , we need 34,000 samples according to Theorem 2.1. But according to Theorem 3.3, we need only 3,800 samples. With 34,000 samples, if we set  $\epsilon = .05$ , then  $\delta$  can be as small as  $10^{-221}$ . Alternatively, if we fix  $\delta = .05$ , with 34,000,  $\epsilon$  can be as small as 0.0071. The improvement in both generalization bound ( $\epsilon$ ) and samples complexity ( $n$ ) is about ten-fold.

The search for the best hyper-rectangle in this experi-

<sup>4</sup>It is known that the VC dimension of hyper-rectangles in  $\mathbb{R}^m$  is  $2m$ .

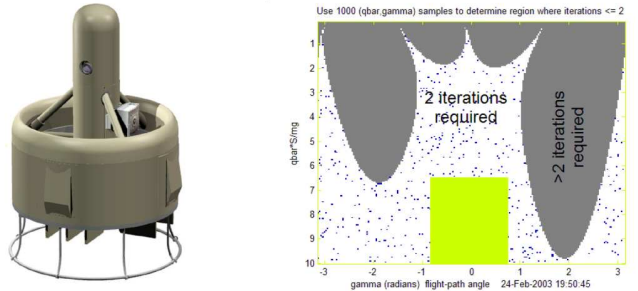


Figure 1: The Organic Air Vehicle (left), and dependence of complexity of equilibrium angle of attack computation on two factors: the flight path angle and the net lift force (right). The computational time here is equated with the number of iterations, and we are to determine under what circumstances the required number of iterations is at most 2. The small rectangle schematically depicts the current conservative safe region, that may be expanded using our SLT-based verification approach.

ment is rather simple. For each sampled input, we determine if the iterative algorithm converges. It turns out that in 32,114 instances, the algorithm converges. Next, we randomly choose a 4-dim hyper-rectangle in the input ranges as a hypothesis. If the hyper-rectangle contains a sampled point for which the algorithm does not converge (an unsafe point), then we eliminate that hyper-rectangle (since the guarantees are based on Theorem 3.3 and require zero false positives). Otherwise, we count the number of safe points that lie outside the hyper-rectangle (false negatives), and choose the hyper-rectangle that has the fewest number of false negatives. After looking at 10,000 random hyper-rectangles, we are able to come up with one that contains 18,616 safe points. This hypothesis thus has  $32,114 - 18,616 = 13,498$  false negatives. Note that the number of false negatives is still quite large. This is because we use hyper-rectangles which constitute a simple hypothesis space that does not approximate the decision surface very well. The advantage to this is that the VC dimension is low, and thus only a small number of samples are required to ensure the statistical guarantee of low probability of false positive. The exact guarantee reads:

*The found hyper-rectangle has probability of false positive bounded by 0.0071, and we have at least 95% confidence in this statement.*

## 5 CONCLUDING REMARKS

The statistical learning theory literature contains a vast collection of techniques to derive and results on

generalization bounds. We make no attempt here to give a comprehensive overview, but refer the reader to a few selected references (Devroye et al., 1996; Anthony & Bartlett, 1999; Vapnik, 2000; Vidyasagar, 2003). But to our knowledge no work has explicitly derived generalization bounds for the problem of learning binary classifiers with weighted error penalties. We aim to fill this gap with this paper. We have shown that for the case when  $\rho$ , the penalty for false negative, is zero, it is possible to derive bounds that are much sharper than the existing Vapnik’s general bound (Theorem 2.1). These bounds are applicable for consistent hypotheses – those with zero empirical loss – or for hypotheses with small empirical losses. While our proof method is very standard and based on proofs for the unweighted case, we succeeded at pinpointing the parts where a weighted loss function has effects. We also derived generalization bounds for the case when  $\rho \in (0, 1)$ . Finally, we made a case for the practical relevance of the assumption  $\rho = 0$  by presenting an SLT-based methodology to verify computational properties of complex controllers, and presenting an example in verifying a real-world controller. We expect that our generalization bounds for weighted binary classification has applications in other area, such as clinical testing as well.

The bounds derived in this paper are without exception based on the concept of VC dimension. VC dimension was originally introduced as a combinatorial parameter to estimate the so-called *annealed entropy* in the covering argument of a VC analysis (Vapnik & Chervonenkis, 1971). There are other parameters that can be and have been used for this purpose, notably the P-dimension (also pseudo-dimension), and the fat-shattering dimension (Anthony & Bartlett, 1999). These parameters are in general harder to estimate, and as such are less attractive for our purpose in statistical verification. Recent work on data-based measures of complexity such as Rademacher and Gaussian complexities (Koltchinskii, 2001; Bartlett & Mendelson, 2002) may provide additional tools for our statistical verification approach.

## References

Anthony, M. and P. Bartlett. *Neural Network Learning: Theoretical Foundations*. Cambridge University Press, 1999.

Anthony, Martin and John Shawe-Taylor. A result of Vapnik with applications. *Discrete Applied Mathematics*, 47(2):207–217, 1993.

Bartlett, P. L. and S. Mendelson. Rademacher and Gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 3:463–482, 2002.

Blumer, Anselm, A. Ehrenfeucht, David Haussler, and Manfred K. Warmuth. Learnability and the Vapnik-Chervonenkis dimension. *J. ACM*, 36(4):929–965, 1989.

Devroye, L., L. Gyrfi, and G. Lugosi. *A Probabilistic Theory of Pattern Recognition*. Springer, New York, 1996.

Ehrenfeucht, Andrzej and David Haussler. A general lower bound on the number of examples needed for learning. *Inf. Comput.*, 82(3):247–261, 1989.

Elgersma, M.R. and B.G. Morton. Nonlinear six-degree-of-freedom aircraft trim. *Journal of Guidance, Control, and Dynamics*, 23(2):305–311, 2000.

Koltchinskii, V. Rademacher penalties and structural risk minimization. *IEEE Transactions on Information Theory*, 47(5):1902–1914, July 2001.

McDiarmid, C. Concentration. In M. Habib, C. McDiarmid, J. Ramirez-Alfonsin, and B. Reed, editors, *Probabilistic method for algorithmic discrete mathematics*, pages 195–248. Springer-Verlag, New York, 1997.

Sauer, N. On the density of families of sets. *Journal of Combinatorial Theory*, 13:145–147, 1972.

Shelah, S. A combinatorial problem; stability and order for models and theories in infinitary languages. *Pacific Journal of Mathematics*, 41:247–261, 1972.

Vapnik, V. *Statistical Learning Theory*. Wiley, 1998.

Vapnik, V. *The Nature of Statistical Learning Theory*. Springer, second edition, 2000.

Vapnik, V. and A. Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability and Its Applications*, 16:264–280, 1971.

Vidyasagar, M. *Learning and Generalization, with Applications to Neural Networks*. Springer, second edition, 2003.