ESTR2004 Course Project Fall Semester:1–10, 2023

Test Set Bound in Binary Classification Theory

Hok Fong WONG (1155189917) SILASWHF@LINK.CUHK.EDU.HK
The Chinese University of Hong Kong, Hong Kong SAR, China

1. Introduction

This essay serves as a tutorial for understanding the "Test Set Bound" in binary classification theory (Langford, 2005a). To evaluate a model's performance, we would need to calculate its error rate. However, the calculation is usually not viable due to either the large number of inputs or the lack of observation samples. Hence, we could only heuristically estimate a test set error rate and argue that the error rate is not too far away from the genuine error rate of the classifier. Surprisingly, we can bound the true error rate with confidence intervals without relying on the properties of the classifier by elementary probabilistic methods and techniques in inequalities. The importance of the test set bound can be seen for reporting classifiers' performance, as it has provided a better method generalizable to most of the commonly used learning models without giving bogus claims.

2. Preliminaries

This essay assumes that the readers are equipped with basic knowledge in discrete mathematics.

2.1. Classification Models

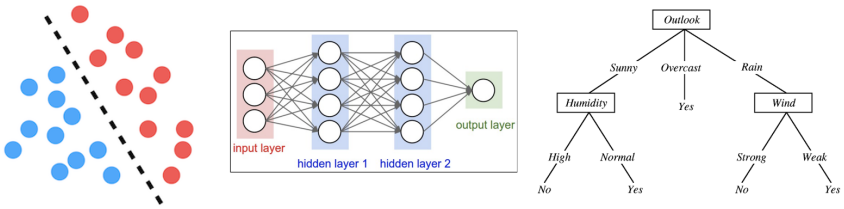

Figure 1: The above shows three common classification models, from left to right: (a) Support Vector Machine, (b) Neural Network, (c) Decision Tree.[1]

1. These are all famous machine learning models. If you are interested in learning them, you can find several online courses on the Internet.

Classifiers are algorithms driven by data sets, which can make predictions or decisions. A model is initially fit on a training data set described by a number of $(\text{observation}, \text{result})$ pairs. The observation is the input to the model, then the model produces the corresponding output for us to compare to the real-world results and thus enabling the adjustment of a model. This is called the **training phase**.

In this essay, we mainly focus on finalized binary classifiers after training. Then, results $\in \{0, 1\}$. To test the performance of a model (testing phase), we use a test data set independent of the training data set for an unbiased evaluation and **observe the number of errors made by the classifier**. Below is a formalized description of the testing phase:

Input: n test samples $(x^1, y^1), (x^2, y^2), \cdots, (x^n, y^n)$
Output: hypothesis made by the classifier $c(x^i)$, where $c \to \{0, 1\}$, $i = 1, 2, \cdots, n$

2.2. Important Definitions in Learning Theory

The goal of a learning model is to find a function capable of predicting the output based on the input. A good classifier is expected to learn a close approximation to the target concept with high probability and efficiency. Classifiers are almost error-prone, but we can be confident in different models and argue that some of the models are more accurate. To further formalize our statement, we will define several variables for consistency throughout the essay.

Variable	Defintion
X	The space of input to the classifier
$Y = \{0, 1\}$	The output of a classification
D	An unknown distribution over $X \times Y$
S	A sequence of examples drawn independently from D, called the test set
m	$= \lvert S \rvert$, which is the number of examples in the test set
c	A function mapping X to Y

Again, note that our argument does not rely on the classifier space since our error estimation only relies on a fixed true error rate of a classifier. Below is an example dedicated to showing the usage of these newly assigned variables.

Example 1 *Spam email identification. X would be the content of the email, sender or other possible information as sensory inputs and Y would be 0 if the prediction is "not spam" and 1 otherwise. The distribution D is over sensory inputs and outcomes. The sample set S might consist of $m = 100$ (email, result) pairs such as $(\textit{email}_1, \text{"spam"})$, $(\textit{email}_2, \text{"spam"})$, $\cdots$, $(\textit{email}_{100}, \text{"not spam"})$. A classifier c, is a function which predicts "spam" or "not spam" based on the email.*

Two more derived variables are crucial for the discussion of the test set bound.

Definition 1 (True Error) *The true error c_D of a classifier c is defined as the probability that the classifier errs:*

$$c_D \equiv \mathop{\mathbb{P}}_{(x,y)\sim D}(c(x) \neq y)$$

under draws from the distribution D.

Definition 2 (Empirical Error) *Given a sample test set S, the empirical error rate $\widehat{c_S}$ is defined as the observed number of errors $\widehat{m}$ divided by the number of test samples m:*

$$\widehat{c_S} \equiv \mathop{\mathbb{P}}_{(x,y)\sim S}(c(x) \neq y) = \frac{1}{m}\sum_{i=1}^{m}|c(x_i) - y_i| = \frac{\widehat{m}}{m}.$$

Note that the two definitions describe different probability with respect to the input space.

3. Establishing the Test Set Bound

Given a classifier with an unknown error rate c_D, the only major observable quantity is the number of errors the classifier has made. Since the samples from the test set are drawn independently, the probability of the classifier making an error is exactly the true error of the classifier. Therefore, we can imitate a classifier using independent biased coin flip experiments, where heads are shown with the probability c_D as making errors.

Hereby, we question the probability of observing k or fewer errors (heads) out of m examples in the test set (coin-flip experiments). This answer should be rather simple since it matches the concept of the cumulative distribution of a binomial.

To refresh our memory about binomial random variables, suppose we have m placeholders for the ongoing coin-flipping experiment. Determining the result for all m placeholders would incur in the probability $c_D^k(1-c_D)^{m-k}$, where c_D, k is the probability of showing a head and there are k specified placeholders to be heads, respectively. However, the number of such sequences is $\binom{m}{k}$, then the probability that k heads occur in m placeholders will be

$$\mathbb{P}(X = k) = \binom{m}{k} c_D^k (1-c_D)^{m-k}.$$

Here, X is called a binomial random variable with success rate c_D.

Definition 3 (Binomial Tail Distribution)

$$Bin(m, k, c_D) \equiv \mathop{\mathbb{P}}_{X_1, X_2, \cdots, X_n \sim \textit{Bernouli}(c_D)^m}\left(\sum_{i=1}^{m} X_i \leq k\right) = \sum_{i=1}^{k}\binom{m}{i} c_D^j (1-c_D)^{m-j},$$

which is equivalent to the cumulative mass function of a binomial random variable up to k. Here, the bernouli random variable is defined to takes value 1 with probability c_D and 0 otherwise.

True error bounds are usually served in the form of "over an independent and identically distributed (i.i.d.) draw of some sample, the expected error rate of a classifier is bounded by $f(\delta, \text{error rate on sample})$ with probability $1-\delta$". Formally, we can describe the statement using the following expression (this shows the upper bound):

$$\mathbb{P}(c_D \leq f(\delta, \widehat{m})) \geq 1 - \delta. \tag{1}$$

The term $1-\delta$ represents a confidence interval showing the event defined will be true at least for this portion of time.

We ought to establish an upper bound for the true error c_D using the major observable quantity $\widehat{c_S}$. Nonetheless, we ought to identify two key properties.

1. $m\widehat{c_S} \sim Binomial(m, c_D)$.
2. For a fixed m and k, $Bin(m, k, p)$ decreases as p increases (See Figure 2).

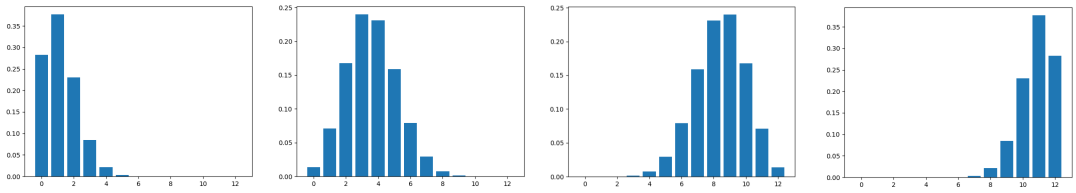

Figure 2: Above shows four binomials when $m = 12$ with probability $p = 0.1, 0.3, 0.7, 0.9$, respectively, from left to right, and the maximum tend to move rightward.

To start with, we should first appreciate how beautiful the relationship of $\widehat{c_S}$ and c_D is, since c_S represents c_D at some level. We are actually investigating several possible c_D's and by looking at a binomial constructed by these possible c_D's we realize that those which are too far away from the true error value should be rejected. Hence, let us first fiddle with c_D and $\widehat{m}$ in the probability expression in (1). We would like to turn $\widehat{m}$ as a random variable in an equivalent event space. This should be

$$\mathbb{P}(\widehat{m} \geq l) \geq 1 - \delta,$$

where l satisfies

$$\sum_{k=0}^{l} \binom{m}{k} c_D^k (1 - c_D)^{m-k} = f^{-1}(\delta).$$

Investigating the complement of $\mathbb{P}(\widehat{m} \geq l)$, i.e. $\mathbb{P}(\widehat{m} \leq l) < \delta$, we found out that this assembles with the setup of l, motivating us to let $f^{-1}(\delta) := \delta$. This led to the definition of Binomial Tail Inversion.

Definition 4 (Binomial Tail Inversion)

$$\overline{Bin}(m, k, \delta) \equiv \max_p \{p : Bin(m, k, p) \geq \delta\}$$

equals the largest true error such that the probability of observing k or less "heads" is at least δ.

With these in hand, we can establish a formal proof for test set bound, i.e.

$$\mathbb{P}(c_D > \overline{Bin}(m, \widehat{m}, \delta)) \geq 1 - \delta.$$

Proof Consider the event

$$\mathcal{E} = \{S : c_D > \overline{Bin}(m, \widehat{m}, \delta)\},$$

we just need to show that $\mathbb{P}(\mathcal{E}) < \delta$ and thus $\mathbb{P}(\overline{\mathcal{E}}) \geq 1 - \delta$, which is the test set bound.

The derivation requires quite a bit of patience, and hope the following solves the problem in a step-by-step manner and a clear flow.

1. **Basic Concept:** By definition, $\mathcal{E}$ includes scenarios where the estimated error rate $\widehat{c_S}$ exceeds the maximum q with a tail probability less than δ.

2. **Analysis of the Event**: The event can be understood as

$$\mathcal{E} \subseteq \left\{ S : \delta > \sum_{i=0}^{\widehat{m}} \binom{m}{i} c_D^i (1 - c_D)^{m-i} \right\}.$$

3. **Bounding the Probability**: Let l be an integer satisfying

$$\sum_{i=0}^{l} \binom{m}{i} c_D^i (1 - c_D)^{m-i} < \delta \leq \sum_{i=0}^{l+1} \binom{m}{i} c_D^i (1 - c_D)^{m-i}.$$

   Then,

$$\left\{ S : \sum_{i=0}^{l} \binom{m}{i} c_D^i (1 - c_D)^{m-i} > \sum_{i=0}^{\widehat{m}} \binom{m}{i} c_D^i (1 - c_D)^{m-i} \right\} \Leftrightarrow \{S : \widehat{m} < l\}.$$

4. **Probability Identification** Since $\widehat{m} \sim$ Binomial(m, c_D), we have

$$\mathbb{P}(\mathcal{E}) \leq \mathbb{P}(\widehat{m} < l) < \sum_{i=0}^{l} \binom{m}{i} c_D^i (1 - c_D)^{m-i} < \delta.$$

■

4. Practical Test Set Bound Approximations

The Binomial Tail Inversion might not be easy to calculate when a computer is not beside. Hence, we can relax the test set bound with inequality techniques.

The Chernoff bound shows that

$$Bin(m, \widehat{m}, c_D) \leq \exp\left(-2m\left(c_D - \widehat{c_S}\right)^2\right).$$

Now we would like to establish a similar result with fundamental inequality techniques:

$$Bin(m, \widehat{m}, c_D) \leq \frac{1}{1-r}\exp\left(-2m\left(c_D - \widehat{c_S}\right)^2\right), \text{with } r = \frac{\widehat{c_S}(1-c_D)}{c_D(1-\widehat{c_S})}, 0 \leq \widehat{m} < c_D m.$$

Two inequalities are useful in this derivation and we shall leave their proof in the appendix (or as an interesting exercise, see Appendix A and B):

For any $0 \leq m \leq n$ and m is an integer,

$$\binom{n}{m} \leq \left(\frac{n}{m}\right)^m \left(\frac{n}{n-m}\right)^{n-m}; \tag{2}$$

$$\binom{n}{m-i} \leq \left(\frac{m/n}{1-m/n}\right)^i \binom{n}{m}, \text{ for any } i = 0, 1, 2, \cdots, m. \tag{3}$$

Proof Let $r = \dfrac{\widehat{c_S}(1-c_D)}{c_D(1-\widehat{c_S})}$, consider

$$\begin{aligned}
Bin(m, \widehat{m}, c_D) &= \sum_{i=0}^{\widehat{m}} \binom{m}{\widehat{m}-i} {c_D}^{m-i}(1-c_D)^{n-m+i} \text{ (Reversing the summation order)} \\
&\leq \binom{m}{\widehat{m}} {c_D}^{\widehat{m}}(1-c_D)^{m-\widehat{m}} \cdot \sum_{i=0}^{\widehat{m}} \left(\frac{\widehat{m}/m \cdot (1-c_D)}{(1-\widehat{m}/m)\cdot c_D}\right)^i \text{ (Using inequality (3))} \\
&= \binom{m}{\widehat{m}} {c_D}^{\widehat{m}}(1-c_D)^{m-\widehat{m}} \cdot \frac{1-r^{\widehat{m}+1}}{1-r} \text{ (Noting that } r < 1) \\
&\leq \left(\frac{m}{m-\widehat{m}}\right)^{m-\widehat{m}} \cdot \left(\frac{m}{\widehat{m}}\right)^{\widehat{m}} \cdot {c_D}^{\widehat{m}} \cdot (1-c_D)^{m-\widehat{m}} \cdot \frac{1}{1-r} \text{ (Using inequality (2))} \\
&= (1-\widehat{c_S})^{-(m-\widehat{m})} \cdot {\widehat{c_S}}^{-\widehat{m}} \cdot {c_D}^{\widehat{m}} \cdot (1-c_D)^{m-\widehat{m}} \cdot \frac{1}{1-r} \\
&= \frac{1}{1-r} \cdot \frac{c_D^{\widehat{m}}(1-c_D)^{m-\widehat{m}}}{{\widehat{c_S}}^{\widehat{m}} \cdot (1-\widehat{c_S})^{m-\widehat{m}}} \\
&= \frac{1}{1-r}\exp\left\{-m\left[\widehat{c_S}\ln\left(\frac{\widehat{c_S}}{c_D}\right) + (1-\widehat{c_S})\ln\left(\frac{1-\widehat{c_S}}{1-c_D}\right)\right]\right\}.
\end{aligned}$$

Observe that now $\widehat{c_S}$ and c_D are separable in the logarithm function, then we can establish a function $f(x) = \widehat{c_S}\ln x + (1-\widehat{c_S})\ln(1-x)$ and so $Bin(m, \widehat{m}, c_D) = \frac{1}{1-r}\exp\left(f(\widehat{c_S}) - f(c_D)\right)$.

Consider an approximation by applying the first derivative. We can obtain the approximated result by integrating the derivative back. Note that $x(1-x) \geq 1/4$ for any $x \in [0,1]$.

$$f\left(\widehat{c_S}\right) - f(c_D) = \int_{c_D}^{\widehat{c_S}} f'(x)\mathrm{d}x = \int_{\widehat{c_S}}^{c_D} \frac{x - \widehat{c_S}}{x(1-x)}\mathrm{d}x \geq 4\int_{\widehat{c_S}}^{c_D} (x - \widehat{c_S})\mathrm{d}x = 2(\widehat{c_S} - c_D)^2.$$

How to relate $\overline{Bin}(m, \widehat{m}, \delta)$ to the upper bound above? We note that

$$\begin{aligned}
\overline{Bin}(m, \widehat{m}, \delta) &= \sup\{c_D : Bin(m, \widehat{m}, c_D) \geq \delta\} \\
&\leq \sup\{c_D : \frac{1}{1-r}\exp\left(f(\widehat{c_S}) - f(c_D)\right) \geq \delta\} \\
&= \{c_D : \frac{1}{1-r}\exp\left(f(\widehat{c_S}) - f(c_D)\right) = \delta\}
\end{aligned}$$

Assuming that $r < 1$, then we can solve the equality:

$$\begin{aligned}
\frac{1}{1-r}\exp\left(-2m\left(c_D - \widehat{c_S}\right)^2\right) \leq m\exp\left(-2m\left(c_D - \widehat{c_S}\right)^2\right) &= \delta \\
\exp\left(-2m\left(c_D - \widehat{c_S}\right)^2\right) &= \frac{\delta}{m} \\
c_D &= \widehat{c_S} + \sqrt{\frac{\ln\left(m/\delta\right)}{2m}}
\end{aligned}$$

Then,

$$\mathbb{P}\left(c_D \leq \widehat{c_S} + \sqrt{\frac{\ln\left(m/\delta\right)}{2m}}\right) \geq \mathbb{P}(c_D \leq \overline{Bin}(m, \widehat{m}, \delta)) \geq 1 - \delta.$$

■

5. Test Set Bound in Applications

Many people follow the standard statistical prescription by calculating the empirical mean $\widehat{\mu} = \widehat{c_S}$ and empirical variance $\widehat{\sigma}^2 = 1/(m-1)\sum_{i=1}^{m}(|c(x_i) - y_i| - \widehat{\mu})^2$ and bound the probability by $\widehat{\mu} + 2\widehat{\sigma}$ claiming to have a 95% confidence on the bound. There are a few number of flaws:

1. Assumptions on the asymptotic growth of a Gaussian Distribution is not always accurate;
2. It returns a bound that can be smaller than 0 or larger than 1.

In comparison, the test set bound is, not surprisingly, always giving a bound between $[0, 1]$ due to the definition of Binomial Tail Inversion. Also, the approach is not overly optimistic - our bound is proven to be true with certain confidence. Below is an example of how we can use the test set bound.

Example 2 *Suppose a classifier made* $\widehat{m} = 38$ *errors out of* $m = 100$ *examples. Let the confidence interval of correctly bounding the true error be* 95%*. Then* $\delta = 0.05$*, and thus by our derivation, the upper bound is* $c_D \leq 0.3800 + 0.1949 = 0.5749$ *with* 95% *confidence. By Chernoff bound, the upper bound is* $c_D \leq 0.3800 + 0.1224 = 0.5024$ *with* 95% *confidence. Applying binary search (can be done in* $\mathcal{O}(\widehat{m} \log eps^{-1})$*), we can calculate the exact binomial tail bound* $c_D \leq 0.4565$ *with* 95% *confidence (See Appendix C).*

6. Conclusion

This essay covers the test set bound in a way that uses fundamental knowledge of probability analysis and basic inequality properties. Test set bounds provide a better way to report error rates and confidence intervals on future error rates than some current methods. Current computers are capable of calculating the bound efficiently, and thus this test set bound method should be adopted in machine learning theory and applications.

Appendix A. Proof of Inequality 2

For any $0 \leq m \leq n$ and m is an integer,

$$\binom{n}{m} \leq \left(\frac{n}{m}\right)^m \left(\frac{n}{n-m}\right)^{n-m}.$$

Proof

By binomial theorem,

$$(m + (n-m))^n = \sum_{k=0}^{n} \binom{n}{k} m^k (n-m)^k \geq \binom{n}{m} m^m (n-m)^{n-m}.$$

Dividing n^n on both sides yields:

$$1 \geq \left(\frac{m}{n}\right) \left(1 - \frac{m}{n}\right)^{n-m} \binom{n}{m}.$$

Dividing $(m/n)^m \cdot ((n-m)/n)^{n-m}$ on both sides yields

$$\binom{n}{m} \leq \left(\frac{n}{m}\right)^m \left(\frac{n}{n-m}\right)^{n-m}.$$

■

Appendix B. Proof of Inequality 3

Let $0 \leq m \leq n$ and m is an integer, then for any $i = 0, 1, 2, \cdots, m$,

$$\binom{n}{m-i} \leq \left(\frac{m/n}{1-m/n}\right)^i \binom{n}{m}$$

Proof

Consider that $\binom{n}{m-i} = \frac{n!}{(n-(m-i))!(m-i)!}$, $\binom{n}{m} = \frac{n!}{(n-m)!m!}$, then

$$\begin{aligned}
\binom{n}{m-i} &= \frac{n!}{(n-m)! \cdot m!} \frac{m! \cdot (n-m)!}{(n-m+i)!(m-i)!} \\
&= \binom{n}{m} \prod_{k=0}^{i-1} (m-k) \cdot \prod_{k=1}^{i} \frac{1}{n-m+k} \\
&\leq \binom{n}{m} \frac{m^i}{(n-m)^i} = \binom{n}{m} \left(\frac{m/n}{1-m/n}\right)^i.
\end{aligned}$$

■

Appendix C. Program

This is a program for calculating the Binomial Tail Inversion Bound.

```
fac = [1]
s = 1
for i in range(1, 150):
    s = s * i
    fac.append(s)


def tail(n, m, p):
    s = 0.0
    for i in range(m):
        s += fac[n] / (fac[i] * fac[n - i]) * (p**i) * ((1 - p) ** (
            n - i))
    return s


def binom(n, m, delta):
    l = 0.0
    r = 1.0
    while abs(r - l) > 1e-15:
        mid = (l + r) / 2
        if tail(n, m, mid) >= delta:
            l = mid
        else:
            r = mid
    return l


print(binom(1000, 260, 0.05))
```