64 GENERALIZATION THEORY

We described several data-based methods for inference and learning in the previous chapters. These methods operate directly on the data to arrive at classification or inference decisions. One key challenge these methods face is that the available training data need not provide sufficient representation for the sample space. For example, the training data that may be available in the neighborhood of any feature location, $h \in \mathbb{R}^M$, will generally provide only a *sparse* representation (i.e., a few examples) of the sought-after classifier behavior within this volume of space. It is for this reason that the design of reliable inference methods in higher-dimensional spaces is more challenging than normal. In particular, algorithms that work well in lower-dimensional feature spaces need not work well in higher-dimensional spaces. This property is a reflection of the phenomenon known as *curse of dimensionality*. We examine these difficulties in this chapter and arrive at some important conditions for reliable learning from a finite amount of training data.

64.1 CURSE OF DIMENSIONALITY

To illustrate the curse of dimensionality effect, we refer to Fig. 64.1. Consider initially a one-dimensional space, with $M = 1$, and assume all N training points $\{h_n\}$ (which are now scalars) are randomly distributed within the interval $[0, 1]$. In this case, we say that we have a sample density of $d = N$ samples/dimension.

Let us now consider the two-dimensional case, with $M = 2$, and let us assume, similarly, that the N training points are randomly distributed within the square region $[0, 1] \times [0, 1]$. In this case, the resulting sample density will be $d = N^{1/2}$ samples/dimension. This can be seen as follows. Referring to the diagram in the left part of Fig. 64.2, we partition the horizontal and vertical dimensions of the square region $[0, 1] \times [0, 1]$ into $N^{1/2}$ sub-intervals in each direction. This division results in a total of N smaller squares. Since the total number of training samples is N, and since these samples are assumed to be uniformly distributed within the region $[0, 1] \times [0, 1]$, we conclude that the expected number of samples per small square is equal to one. Consequently, if we consider any horizontal (or vertical) stripe, the average number of samples in that stripe will be $N^{1/2}$, from which we infer that the sample density is $d = N^{1/2}$ samples/dimension. Likewise, for

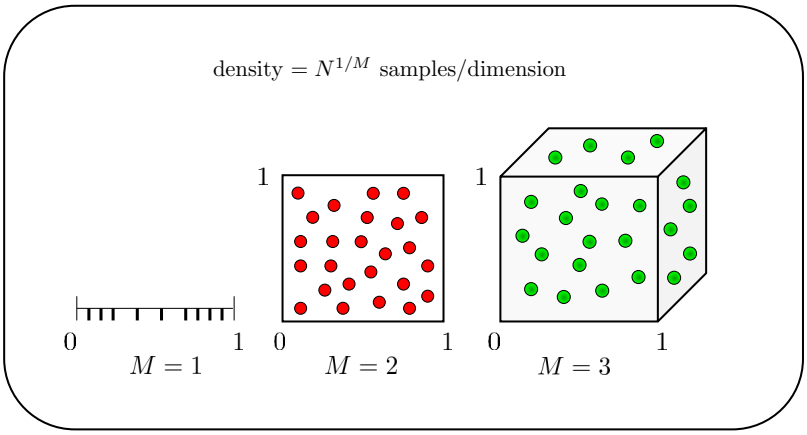

Figure 64.1 The plots illustrate how sample density varies with the dimension values $M = 1, 2, 3$. For a generic $M-$dimensional space, the density is equal to $N^{1/M}$ samples per dimension.

$M = 3$, the density will be $d = N^{1/3}$ and, more generally, for $M-$dimensional spaces, the density will be

$$d = N^{1/M} \text{ samples/dimension} \tag{64.1}$$

If we were to consider a density value of $d = 100$ samples/dimension to be reasonable for one-dimensional problems within the interval $[0, 1]$, then to attain this same density in $M-$dimensions, we will need a total number N of training samples that satisfies $N^{1/M} = 100$ or

$$N = 100^M \text{ samples} \tag{64.2}$$

For example, for $M = 20$, which is a relatively small feature dimension, we would need to collect 10^{40} samples (that is a huge number of samples). For $M = 40$, we would need 10^{80} samples (that is equal to the estimated number of atoms in the universe)! In other words, as the dimension of the feature space increases, we will be needing substantially more training data to maintain the sampling density uniform. Conversely, if we keep N fixed and increase M, then the higher-dimensional space will become more sparsely populated by the training data.

One other way to visualize this effect is to consider a small hypercube of edge length $\ell < 1$ embedded within the larger $[0, 1]^M$ hypercube in $M-$dimensional space, whose volume is equal to one. The volume of the smaller hypercube is ℓ^M, which is a fraction of the larger volume — see the right plot in Fig. 64.2. If the larger $[0, 1]^M$ hypercube has N samples distributed randomly within it, then the smaller hypercube will contain, on average, a fraction of these samples and their number will be $\ell^M N$. Observe that as M increases, this fraction of samples will decrease in number since $\ell < 1$ and the smaller hypercube will become less populated.

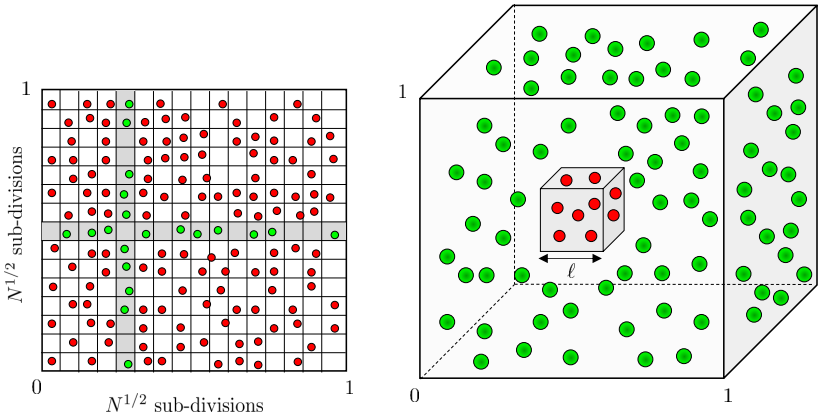

Figure 64.2 The plot on the left illustrates the density expression of $N^{1/2}$ samples per dimension for the case $M = 2$. The plot on the right illustrates that the fraction of training samples inside the smaller cube is equal to $\ell^3 N$ on average.

Example 64.1 (**Numerical example**) We illustrate the curse of dimensionality effect by means of an example. A collection of $N = 2000$ feature vectors $h_n \in \mathbb{R}^M$ are generated randomly for increasing values of M. The entries of each h_n are uniformly distributed within the range $[-0.5, 0.5]$ so that the feature vectors lie inside a hypercube of unit edge centered at the origin. For each fixed M, we determine the distance to the closest neighbor for each feature vector and average these distances over all $N = 2000$ vectors. The numerical values listed in the table below are obtained in this manner.

Table 64.1 Average minimum distance to nearest neighbor for different M, obtained by averaging over $N = 2000$ random feature vectors.

dimension, M	**average minimum distance**
1	0.00026
10	0.46
50	2.07
100	3.28
500	8.34
1000	12.13
5000	28.10
10000	40.06

The values in the table indicate that the minimum distance between uniformly distributed feature vectors increases quickly with the feature dimension, M, so that the feature vectors become more dispersed in higher dimensions. Actually, as the dimension M increases, the feature vectors tend to concentrate at the corners of the unit hypercube. To see this, assume we insert a sphere of radius $1/2$ inside this hypercube; it is centered at the center of the cube — see Fig. 64.3. Its volume is given by the expression

$$\text{volume} = \left(\frac{1}{2}\right)^M \frac{\pi^{M/2}}{\Gamma\left(\frac{M}{2}+1\right)} \tag{64.3}$$

in terms of the Gamma function, $\Gamma(x)$, defined earlier in Prob. 4.3. Since the feature vectors are uniformly distributed in space, we find that the ratio of points that lie inside the sphere relative to the points that lie inside the hypercube is equal to the above volume expression. Taking the limit as $M \to \infty$, the volume expression approaches zero (see Prob. 64.8), which confirms that most of the volume of the hypercube is at its 2^M corners and not in the center. Consequently, the feature vectors become more spread out as M increases.

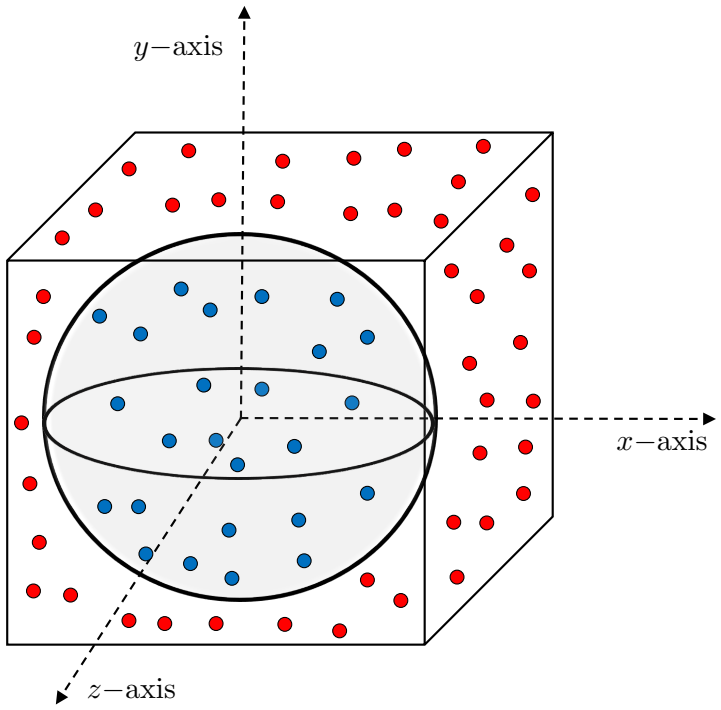

Figure 64.3 A cube centered at the origin with unit edge length, along with a sphere of radius 1/2 inserted inside the cube and touching its surfaces. Feature vectors are randomly distributed inside the cube.

Implication for classification

The curse of dimensionality is problematic when one is searching for classification mappings, $\widehat{\gamma}(h) : \mathbb{R}^M \to \mathbb{R}$, over the set of *all* possible classifiers. This is because the problem of determining a classifier is essentially one of fitting a function $\widehat{\gamma}(h)$ to the training data $\{\gamma(n), h_n\}$ and using it to classify test features, h, for example, by examining the sign of $\widehat{\gamma}(h)$ when $\gamma \in \{\pm 1\}$. As the feature dimension increases, a significantly larger amount of training data will be necessary for a better fit. The larger amount of data allows to sample the feature space more

densely so that the behavior of the training data $\{\gamma(n), h_n\}$ is informative enough to obtain a classifier that performs well over the entire feature space.

One important question then is whether it is possible to design a good classifier in high-dimensional spaces. We address this question in the next section and answer it in the affirmative under some conditions. Specifically, it will turn out that as long as the size of the training data is large enough *and* the complexity of the classification mapping that we are seeking is moderate, then we will be able to learn reasonably well. We have two main tools at our disposal to deal with the curse of dimensionality:

(a) (**Moderate classifier complexity**) One first approach is to limit the complexity of the classification model by restricting the class of classifiers, as will be done further ahead in (64.11). This is one reason why we often rely on *affine* or linear classifiers (and not arbitrary classifier structures).

(b) (**Dimensionality reduction**) A second approach is to reduce the dimension of the feature space. We already encountered two dimensionality-reduction procedures in the earlier chapters in the form of the Fisher discriminant analysis (FDA) method of Sec. 56.4 and the principal component analysis (PCA) method of Chapter 57.

In this chapter, we focus on the first approach, which relies on reducing the classifier complexity. In particular, we will examine the feasibility of the learning problem and explain how it is affected by the size of the training data, N, and by the complexity of the classifier model.

64.2 EMPIRICAL RISK MINIMIZATION

The available information for learning is limited to the training data:

$$\{\gamma(n), h_n,\ n = 0, 1, \ldots, N-1\} \tag{64.4}$$

where n is the running variable and $\gamma(n) \in \{\pm 1\}$ is the binary label associated with the $n-$th feature vector $h_n \in \mathbb{R}^M$. There will be no prior information about the underlying joint data distribution, $f_{\boldsymbol{\gamma},\boldsymbol{h}}(\gamma, h)$. As such, we will rarely be able to solve directly the problem of minimizing the *actual* risk, $R(c)$, defined as the probability of erroneous classifications:

$$c^{\bullet}(h) \triangleq \underset{c(h)}{\operatorname{argmin}} \left\{ R(c) \triangleq \mathbb{P}(c(\boldsymbol{h}) \neq \boldsymbol{\gamma}) = \mathbb{E}\, \mathbb{I}\,[c(\boldsymbol{h}) \neq \boldsymbol{\gamma}] \right\} \tag{64.5}$$

where $c(h) : \mathbb{R}^M \to \{\pm 1\}$ is a classifier mapping from h to the label space. In (64.5), the minimization is over all possible choices for $c(h)$ and the optimal classifier is denoted by the bullet superscript, $c^{\bullet}(h)$. Observe that we are writing the risk $R(c)$ in two equivalent forms: as the probability of misclassification and as the expected value of the indicator function. The second form is valid because

the indicator function is either one or zero, and it assumes the value of one when an error occurs. In the notation used in (64.5), the variable $\boldsymbol{\gamma}$ refers to the true label associated with the feature vector $\boldsymbol{h}$.

We already know that the solution to the above problem is given by the Bayes classifier (28.28):

$$c^{\bullet}(h) \;=\; \begin{cases} +1, & \text{when } \mathbb{P}(\boldsymbol{\gamma}=+1|\boldsymbol{h}=h) \geq 1/2 \\ -1, & \text{otherwise} \end{cases} \tag{64.6}$$

This solution requires knowledge of the conditional probability distribution of $\boldsymbol{\gamma}$ given $\boldsymbol{h}$, which is rarely available beforehand. For this reason, as we already saw in several examples in previous chapters, we will need to deviate from seeking the optimal Bayes solution and settle on approximating it from the training data $\{\gamma(n), h_n\}$. One first approximation to consider is to minimize the empirical error rate over the training data, i.e., to replace (64.5) by

$$c^{\blacktriangle}(h) \;\triangleq\; \underset{c(h)}{\operatorname{argmin}} \left\{ R_{\text{emp}}(c) = \frac{1}{N}\sum_{n=0}^{N-1} \mathbb{I}\left[c(h_n) \neq \gamma(n)\right] \right\} \tag{64.7}$$

where we are now counting only the misclassification errors that occur over the training data. We are denoting the solution to this problem by $c^{\blacktriangle}(h)$ with a filled triangle. We reserve the *filled* circle and triangle superscripts to minimizations over *all* classifiers without limitation.

Four optimal classifiers

Problem (64.7) continues to be challenging because it does not restrict the class of classifiers over which the minimization of $R_{\text{emp}}(c)$ is performed. We have seen in several of the learning algorithms we studied before that it is customary to limit the search space over some restricted set of classifiers, denoted by $c \in \mathcal{C}$. This classifier space $\mathcal{C}$ is sometimes called the *hypothesis* space in learning theory where it is denoted by the letter $\mathcal{H}$. We will refer to it instead as the classifier space and use the notation $\mathcal{C}$. One popular classifier class $\mathcal{C}$, which we have employed extensively before is the class of "linear" or affine classifiers of the form:

$$\left\{ c(h) \;=\; \text{sign}\left(h^{\mathsf{T}}w - \theta\right), \quad w \in \mathbb{R}^M,\ \theta \in \mathbb{R} \right\} \tag{64.8}$$

where the sign function is defined by

$$\text{sign}(x) \;\triangleq\; \begin{cases} +1, & \text{if } x \geq 0 \\ -1, & \text{if } x < 0 \end{cases} \tag{64.9}$$

This class of classifiers is parameterized by (w, θ); each choice for (w, θ) results in one particular classifier. Once optimal values $(w^{\star}, \theta^{\star})$ are selected (based on some design criterion), for any test feature vector, h, the classification decision is based on examining the sign of $h^{\mathsf{T}}w^{\star} - \theta^{\star}$. Other families of classifiers are of course possible such as nonlinear models that are based on kernel representations or neural network models (which are studied in future chapters).

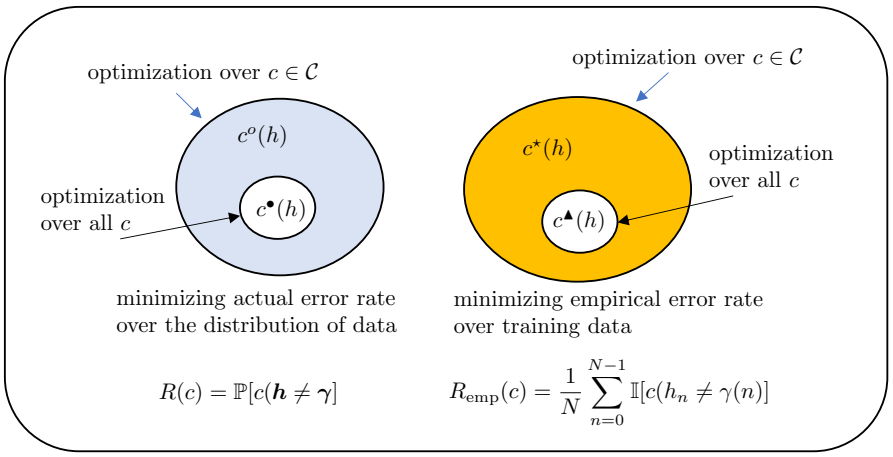

Figure 64.4 Four inter-related optimization problems. Two classifiers $\{c^\bullet(h), c^o(h)\}$ minimize the actual risk, while two other classifiers $\{c^\blacktriangle(h), c^\star(h)\}$ minimize the empirical error rate. Moreover, two classifiers $\{c^o(h), c^\star(h)\}$ restrict the search class to $c \in \mathcal{C}$, while two other classifiers $\{c^\bullet(h), c^\blacktriangle(h)\}$ do not. The smaller circles are meant to indicate that the respective optimal classifiers attain smaller risk values because they are optimizing over a larger pool of classifiers.

Whether we restrict or not the class of classifiers, and whether we minimize the actual or empirical risk, we end up with four inter-related optimization problems that we can compare against each other — see Fig. 64.4. We denote the minimizer for the actual risk (64.5) by $c^\bullet(h)$, which uses the bullet superscript notation. This is the optimal Bayes classifier (the ideal solution that we aim for but is generally unattainable). This solution results from minimizing the risk (or error rate) $R(c)$ over *all* possible classifier mappings and not only over any restricted set $c \in \mathcal{C}$, i.e.,

$$c^\bullet(h) \triangleq \underset{c(h)}{\text{argmin}}\ R(c) \tag{64.10}$$

Once we limit the minimization to some classifier set, say, $c \in \mathcal{C}$, the resulting minimizer need not agree with $c^\bullet(h)$ anymore, and we will denote it instead by $c^o(h)$ using the circle superscript notation to refer to optimality over a restricted search space:

$$c^o(h) \triangleq \underset{c(h)\in\mathcal{C}}{\text{argmin}}\ R(c) \tag{64.11}$$

The larger the space $\mathcal{C}$ is, the closer we expect the solution $c^o(h)$ to get to the optimal Bayes classifier, $c^\bullet(h)$. We say that the restriction $c(h) \in \mathcal{C}$ introduces a form of *inductive bias* by moving the solution away from $c^\bullet(h)$. Problem (64.11) continues to require knowledge of the underlying joint data distribution to evaluate the risk $R(c)$. In data-based learning methods, we move away from this requirement by relying solely on the training data. In that case, we replace $R(c)$

in (64.11) by the empirical error rate over the training data and denote the solution by $c^\star(h)$:

$$c^\star(h) \triangleq \underset{c(h)\in\mathcal{C}}{\operatorname{argmin}} \; R_{\text{emp}}(c) \tag{64.12}$$

We reserve the star superscript notation to solutions that result from using the training data. Thus, note that we use the circle (o) superscript to refer to optimality over the entire distribution of the data, and the star ($\star$) superscript to refer to optimality relative to the training data. In contrast to $c^\blacktriangle(h)$ from (64.7), the sought-after classifiers in (64.12) are limited to the set $c \in \mathcal{C}$. Problem (64.12) is referred to as the *Empirical Risk Minimization* (ERM) problem and its solution is solely dependent on the training data. This is the problem that the various learning procedures that we have been studying focus on and its performance should generally be compared against $c^o(h)$ in (64.11). Table 64.2 lists the four classifiers discussed in this section and indicates whether they minimize the actual or empirical risk and whether they restrict the class of classifiers.

Table 64.2 Four optimal classification problems and their respective classifiers.

	actual risk, $R(c)$	**empirical risk,** $R_{\text{emp}}(c)$
minimization over all c	$c^\bullet(h)$	$c^\blacktriangle(h)$
minimization over $c \in \mathcal{C}$	$c^o(h)$	$c^\star(h)$

64.3 GENERALIZATION ABILITY

Our main focus will be on designing $c^\star(h)$, namely, classifiers that minimize the empirical error rate over some classifier set $c \in \mathcal{C}$, and on examining how close their *actual* error performance, $R(c^\star)$, gets to the solution $c^o(h)$ that minimizes the actual risk. Ideally, we would like the performance of $c^\star(h)$ to approximate the performance of the optimal Bayes solution, $c^\bullet(h)$, as $N \to \infty$. However, this objective is generally impossible to meet. This is because the determination of $c^\star(h)$ is limited to the restricted set $c \in \mathcal{C}$, while the determination of $c^\bullet(h)$ is over all possible classifier mappings.

We therefore need to formulate a more realistic expectation. Since we are limiting the search space to some set $c \in \mathcal{C}$ (such as the space of affine classifiers), it is the two classifiers $\{c^\star(h), c^o(h)\}$ that matter the most in our discussions. For this reason, it is the risk value of $c^o(h)$ that we would like the empirical solution $c^\star(h)$ to approach and not that of $c^\bullet(h)$. This is an attainable objective. We will show below in (64.20) that, under some reasonable conditions, the risk value of $c^\star(h)$ can be made to approach asymptotically, as $N \to \infty$ and with high probability $1-\epsilon$, the risk value of $c^o(h)$. This is a remarkable conclusion, especially since it will hold irrespective of the joint distribution of the data $(\boldsymbol{\gamma}, \boldsymbol{h})$.

64.3.1 Vapnik-Chervonenkis Bound

To arrive at this important conclusion, we let VC denote the so-called *Vapnik-Chervonenkis* dimension of the classifier set, $\mathcal{C}$ (we will not be limiting this set to linear classifiers in the current discussion). We will define the VC dimension later in Sec. 64.4. Here, it is sufficient to know that this nonnegative number serves as a measure of the complexity of the classification set: more complex classifier models will have larger VC dimension than simpler models (but this will not necessarily translate into better learning)! For example, for the case of affine classifiers, we will find that $\mathrm{VC} = M + 1$.

The argument that leads to future conclusion (64.20) relies on a fundamental result in statistical learning theory, known as the Vapnik-Chervonenkis bound. The result is motivated in Probs. 64.24–64.25 under some simplifying conditions, and is proven more generally in Appendix 64.C. To state the result, we introduce an auxiliary parameter. Given any small $\epsilon > 0$, we introduce a positive factor $\delta > 0$ that depends on N, VC, and ϵ as follows:

$$\delta(N, \mathrm{VC}, \epsilon) \triangleq \sqrt{\frac{32}{N}\left\{\mathrm{VC}\,\ln\left(\frac{Ne}{\mathrm{VC}}\right) + \ln\left(\frac{8}{\epsilon}\right)\right\}} \tag{64.13}$$

where the letter "e" refers to the base number for natural logarithms, $e \approx 2.7183$. Observe that the value of δ is *independent* of the distribution of the data, $f_{\boldsymbol{\gamma},\boldsymbol{h}}(\gamma, h)$, and that δ is small when N is large (i.e., under sufficient training data) and VC is small (i.e., for moderately complex classification models).

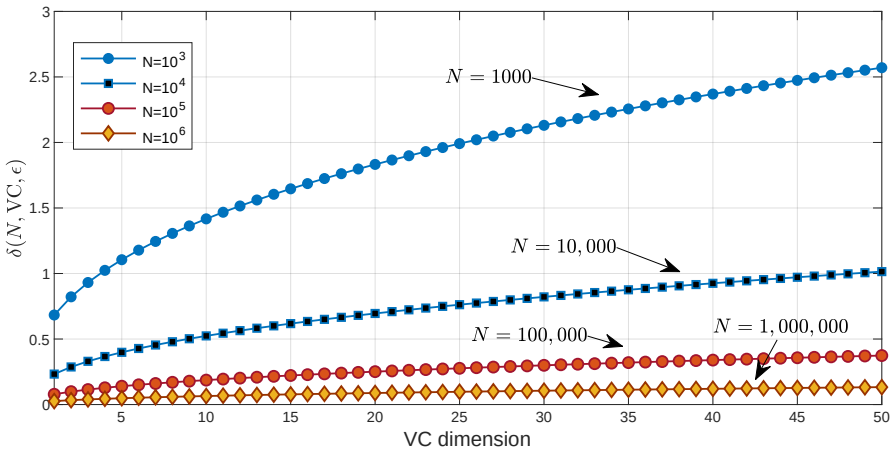

Figure 64.5 The plot illustrates the behavior of the bound $\delta(N, \mathrm{VC}, \epsilon)$ in (64.13) as a function of the VC dimension for various values of N and $\epsilon = 0.01$.

Figure 64.5 illustrates the behavior of δ as a function of the VC dimension for several values of N and $\epsilon = 0.01$. Observe from the plot that, for example for $N = 10^4$, increasing the complexity of the model (i.e., increasing its VC

dimension), enlarges the value of δ. Observe further from (64.13) that if, on the other hand, we fix the values of VC and ϵ and let the size of the training set increase, we obtain:

$$\lim_{N\to\infty} \delta(N) \;=\; 0, \quad \text{for fixed VC and } \epsilon \tag{64.14}$$

Now, using the value of δ defined by (64.13), it can be shown that, *regardless* of the distribution of the data and for *any* $c \in \mathcal{C}$, it holds with high probability of at least $(1-\epsilon)$, that:

$$|R_{\text{emp}}(c) - R(c)| \;\leq\; \delta, \quad \text{for any } c \in \mathcal{C} \tag{64.15}$$

That is, the empirical error rate of classifier c evaluated on the training data is $\delta-$close to its actual error rate over the entire data distribution with high probability $1-\epsilon$. We restate result (64.15) in another equivalent form as follows.

Theorem 64.1. (VC bound) *Consider a collection of N training data points $\{\gamma(n), h_n\}$ and let $\mathcal{C}$ denote the classifier space. Let $R_{\text{emp}}(c)$ denote the empirical risk for any classifier $c \in \mathcal{C}$ over the training data, and let $R(c)$ denote its actual risk (i.e., its probability of misclassification) over the entire data distribution:*

$$R_{\text{emp}}(c) = \frac{1}{N}\sum_{n=0}^{N-1} \mathbb{I}\left[c(h_n) \neq \gamma(n)\right], \quad R(c) = \mathbb{P}\left(c(\boldsymbol{h}) \neq \boldsymbol{\gamma}\right) \tag{64.16}$$

Introduce the parameter δ defined by (64.13) in terms of the VC dimension for $\mathcal{C}$. Then, for any small $\epsilon > 0$, it holds that

$$\mathbb{P}\left(\sup_{c\in\mathcal{C}} |R_{\text{emp}}(c) - R(c)| \leq \delta\right) \;\geq\; 1-\epsilon \tag{64.17}$$

where the supremum is over the classifier set. This useful result is known as the Vapnik-Chervonenkis bound.

Proof: See Appendix 64.C. ■

The result of the theorem provides a bound on the size of the difference between the empirical and actual risks, $R_{\text{emp}}(c)$ and $R(c)$, for any *finite* N and for any classifier, $c \in \mathcal{C}$. Loosely, it states that the difference between these risk values is relatively small (when δ is small) with high probability. The result implies roughly that for any classifier, $c \in \mathcal{C}$:

$$\mathbb{P}\left\{\begin{pmatrix}\text{error rate on}\\ \text{training data}\end{pmatrix} \approx \begin{pmatrix}\text{error rate on}\\ \text{test data}\end{pmatrix}\right\} \;\geq\; 1-\epsilon \tag{64.18}$$

where we are using the symbol $a \approx b$ to indicate that the values a and b are similar up to a small difference of magnitude δ. Obviously, since the risk values $R_{\text{emp}}(c)$ and $R(c)$ amount to misclassification error rates (and are therefore probability measures), their individual values must lie within the interval $[0,1]$. This means that the bound (64.15) is meaningful only for parameter values (N, VC, ϵ) that

result in small δ; this typically requires large sample size, N, as already illustrated by Fig. 64.5.

64.3.2 PAC Learning

The VC bound (64.17) is important because it implies, as we now explain, that learning is feasible when N is large and the VC dimension is relatively small (so that δ is small). Indeed, note that for the classifiers $\{c^\star(h), c^o(h)\}$ that we are interested in, it holds with probability at least $1-\epsilon$ that:

$$\begin{aligned} R(c^\star) &\leq R_{\text{emp}}(c^\star) \;+\; \delta \qquad \text{(by (64.15) applied to } c^\star) \\ &\leq R_{\text{emp}}(c^o) \;+\; \delta \qquad \text{(since } c^\star \text{ minimizes } R_{\text{emp}}(c)) \\ &\leq R(c^o) \;+\; 2\delta \qquad \text{(by (64.15) applied to } c^o) \end{aligned} \tag{64.19}$$

That is,

$$\boxed{\mathbb{P}\Big(R(c^\star) - R(c^o) \leq 2\delta\Big) \;\geq\; 1-\epsilon} \tag{64.20}$$

Recall that, by design, $R(c^\star) \geq R(c^o)$ since $c^o(h)$ minimizes the actual risk $R(c)$. The above result is known as the PAC bound, where the letters PAC stand for *"Probably Approximately Accurate"* learning. When δ is small (e.g., when N is large and VC is small), the result shows that a classifier $c^\star(h)$ determined from the training data is able to produce misclassification errors over the distribution of the data that are comparable to the best possible value, $R(c^o)$, i.e., $R(c^\star) \approx R(c^o)$. However, we still do not know how small $R(c^o)$ is. This value can be assessed from the empirical risk, $R_{\text{emp}}(c^\star)$. Using (64.15) and (64.19), we can verify, again with high probability $1-\epsilon$ that

$$\begin{aligned} |R_{\text{emp}}(c^\star) - R(c^o)| &= \Big|\Big(R_{\text{emp}}(c^\star) - R(c^\star)\Big) \;+\; \Big(R(c^\star) - R(c^o)\Big)\Big| \\ &\leq |R_{\text{emp}}(c^\star) - R(c^\star)| \;+\; |R(c^\star) - R(c^o)| \\ &\overset{(a)}{\leq} |R_{\text{emp}}(c^\star) - R(c^\star)| \;+\; (R(c^\star) - R(c^o)) \\ &\leq \delta + 2\delta \quad \text{(using (64.15) and (64.19))} \\ &= 3\delta \end{aligned} \tag{64.21}$$

where step (a) is because c^o minimizes $R(c)$ over $\mathcal{C}$ and hence $R(c^\star) \geq R(c^o)$. Result (64.21) provides one useful way to assess $R(c^o)$ (and $R(c^\star)$) through $R_{\text{emp}}(c^\star)$; this latter value is readily obtained from the training data.

In summary, we conclude from results (64.15), (64.19), and (64.21) that, with high probability of at least $1-\epsilon$ and for small δ, the empirical and actual risk values (or empirical and actual error rates) for the classifiers $c^\star(h)$ and $c^o(h)$ are

clustered together and satisfy the relations:

$$R(c^o) \leq R(c^\star) \leq R(c^o) \; + \; 2\delta \tag{64.22a}$$

$$|R(c^\star) - R_{\text{emp}}(c^\star)| \leq \delta \tag{64.22b}$$

$$|R(c^o) - R_{\text{emp}}(c^\star)| \leq 3\delta \tag{64.22c}$$

Figure 64.6 illustrates these relations graphically. The first relation states that the risk of the empirical classifier, $R(c^\star)$, does not exceed the optimal risk value $R(c^o)$ by more than 2δ. The second and third relations state that the empirical risk, $R_{\text{emp}}(c^\star)$, provides a good indication of the actual risks $R(c^\star)$ and $R(c^o)$.

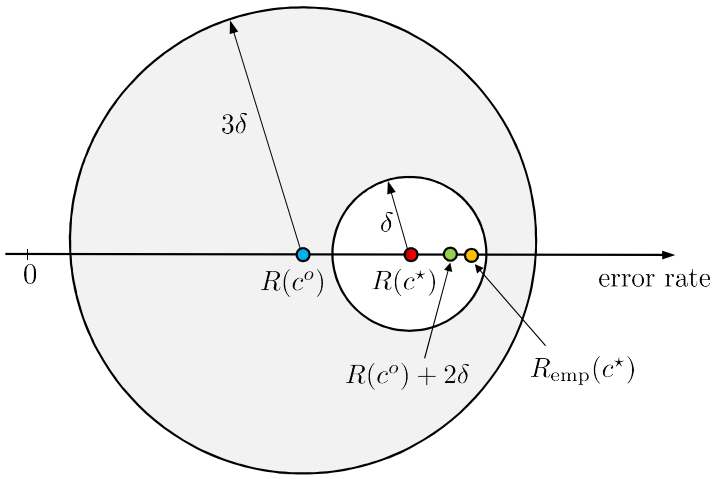

Figure 64.6 The figure illustrates relations (64.22a)–(64.22c). The first relation states that the risk of the empirical classifier, $R(c^\star)$, does not exceed the optimal risk value $R(c^o)$ by more than 2δ. The second and third relations state that the empirical risk, $R_{\text{emp}}(c^\star)$, provides a good indication of the actual risks $R(c^\star)$ and $R(c^o)$.

The main conclusion from the above analysis is the following. Assume the size of the training data, N, is large enough and the complexity of the classification model, VC, is moderate enough such that the corresponding δ parameter from (64.13) is sufficiently small. Assume further that we use the training data to determine a classifier $c^\star(h)$ that minimizes the empirical risk $R_{\text{emp}}(c)$ defined by (64.12) over the set of classifiers, $c \in \mathcal{C}$. If $R_{\text{emp}}(c^\star)$ is small, then the actual risk, $R(c^\star)$, that corresponds to this classifier (i.e., its generalization ability, which corresponds to the probability of misclassification on test data apart from the training data), will also be small. We refer to the test error or error on the test data as the *generalization error*. Moreover, the value of the empirical risk, $R_{\text{emp}}(c^\star)$, will be close to the optimal value $R(c^o)$. These results hold *irrespective* of the distribution of the data, $f_{\boldsymbol{\gamma},\boldsymbol{h}}(\gamma, h)$. In other words, learning from data is feasible under these conditions. By feasible learning we therefore mean any learning procedure that is able to satisfy the PAC property (64.20) with suffi-

ciently small δ. The size of δ can be made small by choosing the sample size, N, large enough.

64.4 VC DIMENSION

We are ready to explain the meaning of the VC parameter. This so-called *Vapnik-Chervonenkis* (VC) dimension of the class of classifiers $\mathcal{C}$, also referred to as the modeling *capacity* of $\mathcal{C}$, is a measure of the complexity of $\mathcal{C}$. We will use the set of linear classifiers to illustrate this concept and subsequently extend it more generally.

Consider a collection of K feature vectors h_n in $M-$dimensional space. In a binary classification setting, each of these feature vectors can be assigned to class $+1$ or -1. There are 2^K possibilities (also called dichotomies) for assigning the K feature vectors over the two classes. We say that a class of classifiers $\mathcal{C}$ is able to *shatter* the K feature vectors if every possible assignment among the 2^K possibilities can be separated by a classifier from the set. We illustrate this definition in Fig. 64.7. The figure considers $K = 3$ feature vectors in $\mathbb{R}^2$ (i.e., $M = 2$ in this case). There are $2^3 = 8$ possibilities for assigning these feature vectors to the classes ± 1. All eight possibilities are shown in the figure on the left. Observe that in each of the eight assignments, we can find at least one line that is able to separate the feature vectors into the classes ± 1. We therefore say that the three feature vectors in this example can be shattered by linear classifiers. In contrast, the figure on the right shows four feature vectors and one particular assignment for them that cannot be separated by linear classifiers.

Motivated by this example, we define the VC dimension for a general class of classifiers, $\mathcal{C}$, as the *largest* value of K for which *at least one* set of K feature vectors can be found that can be shattered by $\mathcal{C}$. For the class of linear classifiers over $\mathbb{R}^2$, the above example shows that $K = 3$. Therefore, VC $= 3$ when $M = 2$. It is important to observe that the definition of the VC dimension is *not* stating that the value of K should be such that *every* set of K feature vectors can be shattered. The definition is only requiring that *at least one* set of K feature vectors should exist that can be shattered.

Example 64.2 (**VC dimension for a finite number of classifiers**) Assume the set of classifiers (linear or otherwise) consists of a finite number, L, of possibilities denoted by $\{c_1, c_2, \ldots, c_L\}$. In this case, the solution of the binary classification problem amounts to selecting one classifier from this collection. Then, it is easy to verify that the VC dimension for this set of classifiers is bounded by:

$$\text{VC}\,(L \text{ classifiers}) \;\leq\; \log_2(L) \tag{64.23}$$

Observe how (64.23) illustrates that the VC dimension of a set $\mathcal{C}$ provides an indication of how complex that set is.

Proof: If the VC dimension of the set of classifiers is denoted by VC, then this means

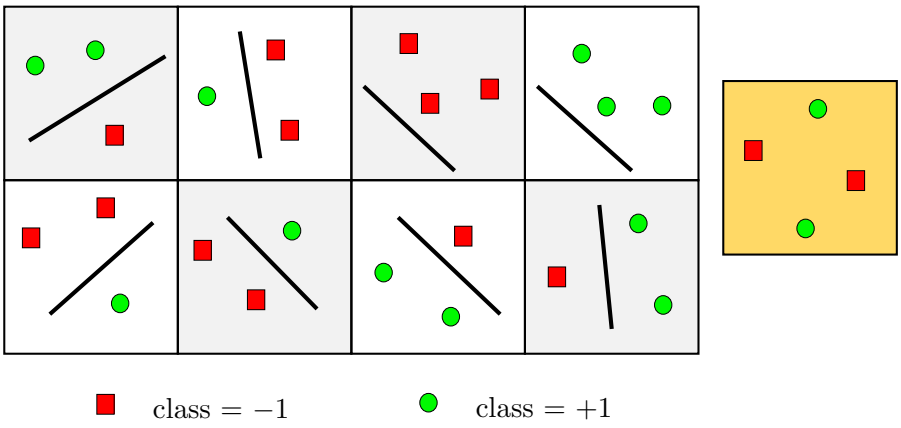

Figure 64.7 The eight squares on the left show all possible assignments of the *same* three feature vectors in $\mathbb{R}^2$. In each case, a line exists that separates the classes ± 1 from each other. We therefore say that the three feature vectors in this example can be shattered by linear classifiers. In contrast, the figure on the right shows four feature vectors in the same space $\mathbb{R}^2$ and an assignment of classes that cannot be separated by a linear classifier.

that we can find a set of VC feature vectors that can be shattered by the L classifiers. This set of VC feature vectors admits 2^{VC} possible labeling assignments. Therefore, the size L should be at least equal this value, i.e., $L \geq 2^{\text{VC}}$, from which we obtain (64.23). ■

The next statement identifies the VC dimension of the class of affine classifiers of the form $c(h) = \text{sign}(h^{\mathsf{T}}w - \theta)$, for some parameters (w, θ).

Lemma 64.1. (Affine classifiers) *The VC dimension for the class of affine classifiers over $\mathbb{R}^M$ is equal to $M+1$, i.e.,*

$$\text{VC}(\text{affine classifiers over } \mathbb{R}^M) \;=\; M+1 \tag{64.24}$$

Proof: See Appendix 64.A. ■

64.5 BIAS-VARIANCE TRADEOFF

The size of δ in (64.13) depends on the VC dimension of the classification set, $\mathcal{C}$. The particular situation illustrated in Fig. 64.5 indicates that the value of δ becomes worse (i.e., larger) for larger VC values. This behavior seems to be

counter-intuitive in that it suggests that using more complex models is not necessarily beneficial for learning and can degrade performance (since it can increase the probability of misclassification and lead to poor generalization).

There are at least two ways to explain this apparent dilemma. One explanation is more intuitive and relies on the *Occam razor principle*, which we already encountered in Sec. 63.2. As was indicated in Fig. 63.2, more complex models can succeed in weaving through the training points and separating them into their respective classes almost flawlessly. However, this "perfect" fitting that happens during the training phase ends up modeling spurious effects and causes poor performance over test data. In the same token, simplistic models need not fit the training data well and can similarly lead to poor misclassification.

64.5.1 Bias-Variance Curve

The second explanation for the dilemma is more formal and relies on an important bias-variance tradeoff that occurs in the design of optimal classifiers.

As explained earlier, we desire the optimal Bayes classifier, $c^{\bullet}(h)$, but can only work with $c^{\star}(h) \in \mathcal{C}$, which is obtained from the training data. The risk associated with $c^{\bullet}(h)$ is denoted by $R(c^{\bullet})$, and the risk associated with $c^{\star}(h)$ is denoted by $R(c^{\star})$. This latter risk is data-dependent and, therefore, it can be viewed as a realization for a random variable: each training dataset leads to one value for $R(c^{\star})$. We use the boldface notation to emphasize this random nature and write $\boldsymbol{R}(c^{\star})$. Computing the expectation of $\boldsymbol{R}(c^{\star})$ over the distribution of the data allows us to evaluate the expected risk value for $c^{\star}(h)$. It is instructive to compare the difference between the optimal risk, $R(c^{\bullet})$, and the expected risk from the training data, $\mathbb{E}\,\boldsymbol{R}(c^{\star})$. For this purpose, we note first that we can write, by adding and subtracting $R(c^{o})$:

$$\mathbb{E}\,\boldsymbol{R}(c^{\star}) - R(c^{\bullet}) = \underbrace{\Big(\mathbb{E}\,\boldsymbol{R}(c^{\star}) - R(c^{o})\Big)}_{\textbf{estimation error (variance)}} + \underbrace{\Big(R(c^{o}) - R(c^{\bullet})\Big)}_{\textbf{approximation error (bias)}} \tag{64.25}$$

This relation expresses the difference on the left as the sum of two components, referred to as the *estimation error* (also called variance) and the *approximation error* (also called bias) — see Fig. 64.8:

(a) (**bias**) The bias error is independent of the training data; it measures the discrepancy in the risk value that results from *restricting* the classifier models to the set $\mathcal{C}$ and by using c^{o} instead of $c^{\bullet}$. The richer the set $\mathcal{C}$ is, the smaller the bias is expected to be.

(b) (**variance**) On the other hand, each training data set results in a realization for the risk value, $R(c^{\star})$. These realizations are represented by the red circles in Fig. 64.8 and they are dispersed around $R(c^{o})$; the dispersion arises from the random nature of the training data. The estimation or variance error therefore measures how far the values of $\boldsymbol{R}(c^{\star})$ are spread around $R(c^{o})$.

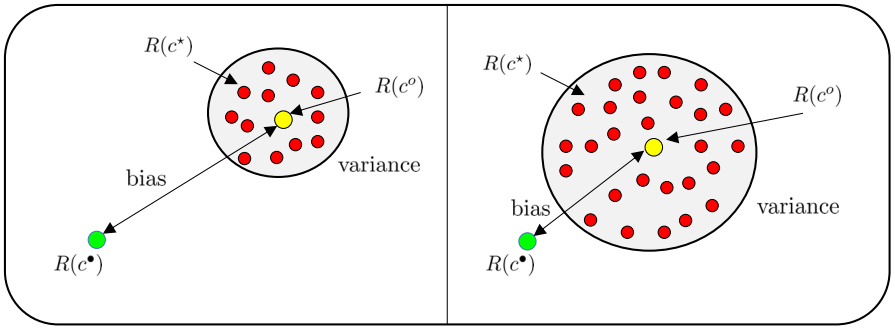

Figure 64.8 The bias quantity relates to the distance from $R(c^o)$ to the optimal Bayes risk value, $R(c^\bullet)$. The variance quantity relates to the spread of $\boldsymbol{R}(c^\star)$ around $R(c^o)$ due to randomness in the data.

The *bias* and *variance* terms behave differently as the complexity of the classification set, $\mathcal{C}$, increases. Assume, for instance, that we enlarge the class of classifiers to $\mathcal{C}' \supseteq \mathcal{C}$. Then, seeking the optimal classifier c^o over the larger set $\mathcal{C}'$ can only reduce the bias component on the right-hand side of (64.25) since

$$\min_{c \in \mathcal{C}'} R(c) \;\leq\; \min_{c \in \mathcal{C}} R(c) \tag{64.26}$$

Therefore, $R(c^o)$ will get closer to $R(c^\bullet)$ and the bias term will get smaller. On the other hand, enlarging the classifier set generally increases the variance component because the realizations $R(c^\star)$ will get dispersed farther away from $R(c^o)$, which is now smaller. Indeed, note that for a fixed N, as the complexity of class $\mathcal{C}$ increases, its VC dimension and subsequently the value of δ in (64.13) also increases. This behavior is observed in Fig. 64.5. It follows from (64.19) that the empirical solution will tend to have risk values, $R(c^\star)$, spread farther away from $R(c^o)$.

64.5.2 Overfitting and Underfitting

We conclude from the bias-variance analysis that there exists a compromise between bias and variance. A simple model set $\mathcal{C}$ may result in large bias but smaller variance. We refer to this scenario as *underfitting* since we would be fitting the data rather poorly by using simple models. In contrast, a more elaborate model set $\mathcal{C}$ may result in smaller bias but larger variance. We refer to this scenario as *overfitting* since we are likely to be overreaching by fitting the data more than is necessary. Combining these facts together we arrive at the bias-variance tradeoff curve shown in Fig. 64.9 in solid color. The curve captures the behavior of the bias and variance components as a function of the model complexity (i.e., its VC dimension). In general, good classifiers, $c^\star(h)$, would be

ones that are close to the minimum of the curve; these are classifiers for which the sum of both components on the right-hand side of (64.25) is the least possible.

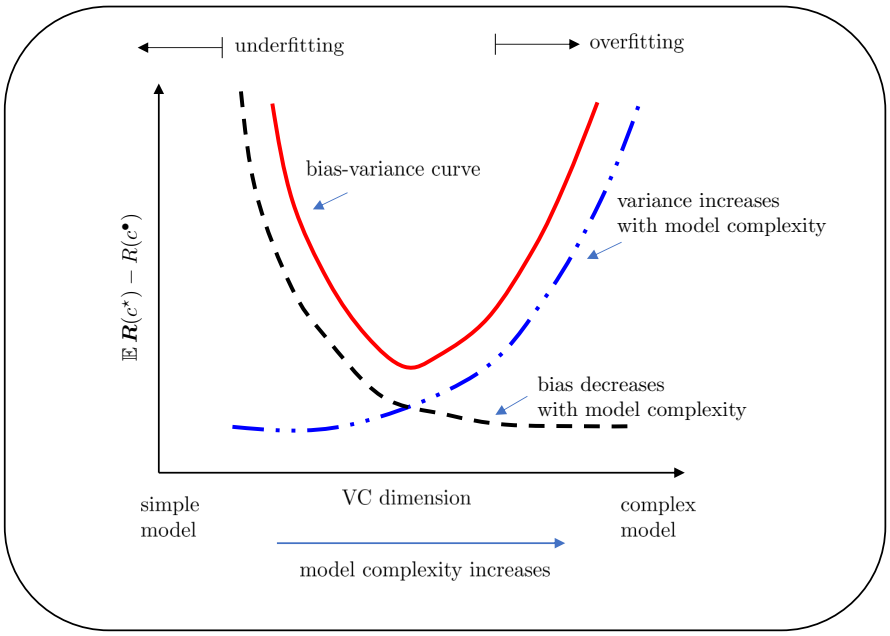

Figure 64.9 Increasing the complexity of the classifier class (i.e., increasing its VC dimension), reduces the bias but increases the variance. The behavior of the bound in (64.19) as a function of VC is illustrated by the solid curve. The figure indicates that there is generally an optimal VC value at which the bound (red curve) is minimized.

64.5.3 Requirements for Feasible Learning

Based on the discussion this far on the bias-variance tradeoff in (64.25) and on the VC bound in (64.17), we conclude that a learning algorithm is effective and able to learn well if it meets three general conditions:

(a) (**Moderate classifier complexity**) The classifier structure should be moderately complex with a reasonable VC dimension in order to limit overfitting and reduce the size of the variance component in (64.25).

(b) (**Sufficient training data**) The algorithm should be trained on a sufficient number of data points. Usually, the value of N is chosen to be some multiple of the VC dimension of the classifier set.

(c) (**Small empirical error rate**) The algorithm should result in a small empirical error rate, $R_{\text{emp}}(c^\star)$, on the training data (i.e., it should have a relatively small number of misclassifications).

When these conditions are met, learning becomes feasible *irrespective* of the probability distribution of the data. This means that the classifier $c^\star(h)$, determined from the training data, will be able to generalize and lead to small misclassification errors on test data arising from the same underlying distribution.

64.6 SURROGATE RISK FUNCTIONS

The previous discussion establishes that learning from data is feasible for a sufficient amount of training data and for moderately complex classifier models (such as affine classifiers). Specifically, if we determine a classifier $c^\star(h)$ with a small empirical error rate (misclassification error) over the training data $\{\gamma(n), h_n\}$, then it is likely that this classifier will perform equally well on test data and its performance will approach that of $c^o(h)$ (which minimizes the probability of error over the distribution of the data).

Thus, consider again the empirical risk minimization problem (64.12) and select the set $\mathcal{C}$ to be the class of affine classifiers, $c(h) = \text{sign}(h^{\mathsf{T}}w - \theta)$. For convenience, we extend the feature and weight vectors using

$$h_n \leftarrow \begin{bmatrix} 1 \\ h_n \end{bmatrix}, \quad w \leftarrow \begin{bmatrix} -\theta \\ w \end{bmatrix} \tag{64.27}$$

in which case $c(w) = \text{sign}(h^{\mathsf{T}}w)$ and the offset parameter is represented implicitly within w. The optimal $w^\star$ that determines $c^\star(w)$ is found by solving

$$w^\star \triangleq \underset{w\in\mathbb{R}^M}{\text{argmin}} \left\{ \frac{1}{N}\sum_{n=0}^{N-1} \mathbb{I}\left[h_n^{\mathsf{T}}w \neq \gamma(n)\right] \right\} \tag{64.28}$$

where we continue to denote the size of w by M. The difficulty we face now is that this problem is not only challenging to solve but is also ill-conditioned, meaning that decisions based on its solution are sensitive (and can change drastically) for minor variations in the data. To see this, we rewrite (64.28) in the equivalent form

$$w^\star \triangleq \underset{w\in\mathbb{R}^M}{\text{argmin}} \left\{ \frac{1}{N}\sum_{n=0}^{N-1} \mathbb{I}\left[\gamma(n)\widehat{\gamma}(h_n) \leq 0\right] \right\} \tag{64.29}$$

where

$$\widehat{\gamma}(h_n) \triangleq h_n^{\mathsf{T}}w \tag{64.30}$$

This alternative rewriting is based on the observation that a classification error occurs whenever the signs of $\gamma(n)$ and $h_n^{\mathsf{T}}w$ do not match each other. It is generally difficult to minimize the empirical risk in (64.29) for at least two main reasons. First, a closed-form expression for $w^\star$ is rarely possible except in some special cases. Second, and more importantly, the $0/1-$loss function,

$$Q(w;\gamma,h) \triangleq \mathbb{I}\left[\gamma\widehat{\gamma}(h) \leq 0\right], \quad \widehat{\gamma} = h^{\mathsf{T}}w \tag{64.31}$$

is nonsmooth over w and its value changes abruptly from 0 to 1. For example, if w is some classifier for which $\mathbb{I}\,[\gamma\widehat{\gamma}(h) \leq 0] = 1$ for a particular feature vector h, then a slight perturbation to this w can transition the indicator function to zero and lead to $\mathbb{I}\,[\gamma\widehat{\gamma}(h) \leq 0] = 0$. This behavior occurs because of the discontinuity of the indicator function $\mathbb{I}[y \leq 0]$ at location $y = 0$, which causes problem (64.29) to be ill-conditioned — see Fig. 64.10. The term "ill-conditioning" refers to the phenomenon in which slight variations in the input data to a problem can lead to significant variations in the outcome.

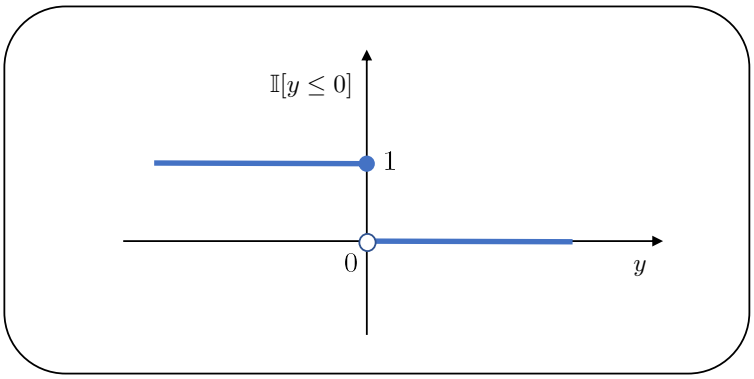

Figure 64.10 The indicator function $\mathbb{I}[y \leq 0]$ is discontinuous at $y = 0$.

To illustrate this undesirable property numerically, assume we succeed in determining a solution, $w^\star$, for (64.29). Consider further a particular training data point h in class $\gamma = -1$ and assume the value of h is such that

$$\widehat{\gamma} = h^{\mathsf{T}} w^\star \;=\; -10^{-6} \tag{64.32}$$

Since $\widehat{\gamma}$ is negative, the classifier $w^\star$ will classify this point correctly:

$$\text{sign}\,(\widehat{\gamma}) \;=\; -1 \;=\; \gamma \tag{64.33}$$

Assume next that in the process of determining $w^\star$ we end up with a slightly perturbed version of it (e.g., due to numerical errors in the optimization process or due to minor perturbations in the training data). We denote this perturbed classifier by $w^\times$. It is not difficult to envision situations in which the perturbed $w^\times$ would lead to a positive value for $\widehat{\gamma}$, say,

$$\widehat{\gamma} = h_n^{\mathsf{T}} w^\times \;=\; 10^{-6} \tag{64.34}$$

The two values $\{-10^{-6}, 10^{-6}\}$ are very close to each other and yet, the new value will cause h to be misclassified and assigned to class $+1$.

Alternate risk functions

Due to the difficulty in dealing with $0/1-$losses, it is customary to rely on surrogate loss functions that are easier to minimize and better behaved. We have

encountered several choices for alternative loss functions in the earlier chapters, such as the logistic loss, hinge loss, quadratic loss, and so forth.

For example, since $\gamma^2 = 1$, we have

$$(\gamma - \widehat{\gamma})^2 \;=\; \Big(\gamma(1 - \gamma\widehat{\gamma})\Big)^2 \;=\; (1 - \gamma\widehat{\gamma})^2 \tag{64.35}$$

so that the quadratic loss $(\gamma - \widehat{\gamma})^2$ will in effect be seeking values w that force the product $\gamma\widehat{\gamma}$ to stay close to one. We refer to the product $\gamma\widehat{\gamma}$ as the *margin* variable:

$$y \;\stackrel{\Delta}{=}\; \gamma\widehat{\gamma}(h), \qquad (\textbf{margin variable}) \tag{64.36}$$

The margin y is a function of w since $\widehat{\gamma} = h^{\mathsf{T}}w$. We can consider several surrogate loss functions defined as follows in terms of the margin variable:

$$Q(y) \;=\; (1-y)^2, \qquad \text{(quadratic loss)} \tag{64.37a}$$

$$Q(y) \;=\; \ln\left(1 + e^{-y}\right), \qquad \text{(logistic loss)} \tag{64.37b}$$

$$Q(y) \;=\; \max\{0, -y\}, \qquad \text{(Perceptron loss)} \tag{64.37c}$$

$$Q(y) \;=\; \max\{0, 1-y\}, \qquad \text{(hinge loss)} \tag{64.37d}$$

$$Q(y) \;=\; e^{-y}, \qquad \text{(exponential loss)} \tag{64.37e}$$

$$Q(y) \;=\; \mathbb{I}[y \leq 0], \qquad \text{(ideal } 0/1\text{−loss)} \tag{64.37f}$$

In each of these cases, the loss function can be interpreted as the "cost" or "price" we incur in using $\widehat{\gamma}(h)$ to predict γ. Figure 64.11 plots these various loss functions. Several observations stand out:

(a) Observe that the ideal $0/1-$loss function returns a value of zero for correct decisions and a value of one for mismatches in the signs of γ and $\widehat{\gamma}$ (i.e., whenever $y \leq 0$); this latter situation corresponds to misclassification.

(b) In comparison, the Perceptron loss (64.37c) also returns zero for correct decisions but penalizes misclassifications close to the boundary $y = 0$ less severely than misclassifications farther away from the boundary; the penalty value varies linearly in the argument y.

(c) The hinge loss (64.37d) shows similar behavior with a linear penalty component; however, this component adds some margin away from the boundary $y = 0$ and penalizes arguments y that are smaller than one (rather than smaller than zero). We already know from our study of support vector machines (SVMs) that this feature adds robustness to the operation of the learning algorithm.

(d) Ideally, under perfect operation, the value of $\widehat{\gamma}(h)$ should match γ and their product should evaluate to one. That is why the quadratic loss penalizes deviations away from one; both to the left and right. However, we know that requiring the product $\gamma\widehat{\gamma}(h)$ to be exactly one is unnecessary; it is sufficient to require the variables γ and $\widehat{\gamma}(h)$ to have the same sign (i.e., to require the margin to be sufficiently positive). For this reason, several of the other loss

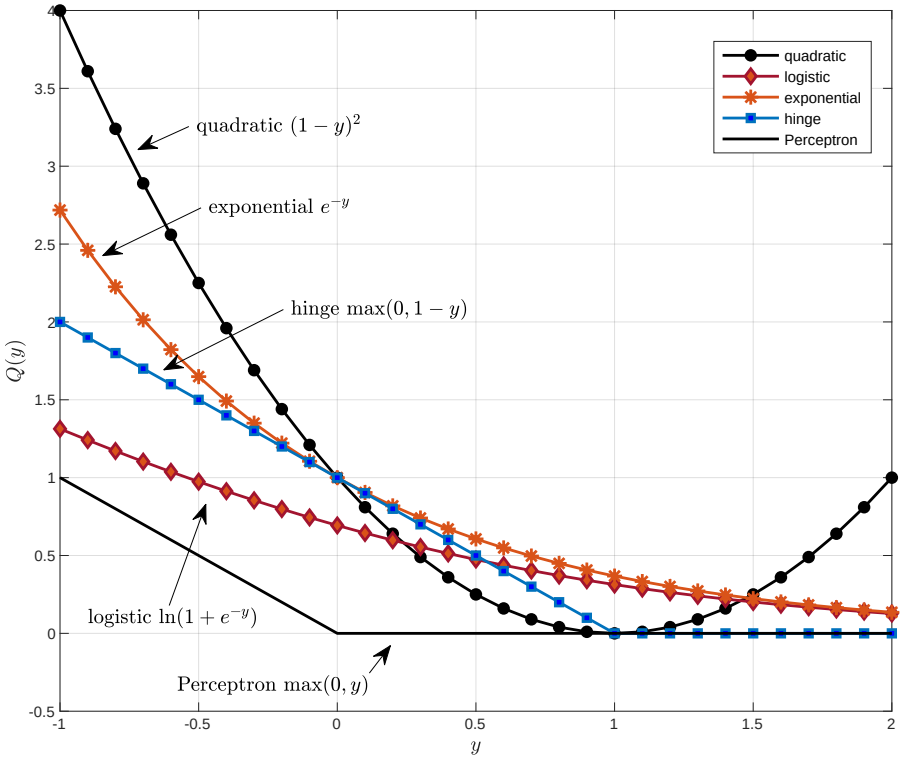

Figure 64.11 The dashed curve shows the plot of the ideal $0/1-$loss $\mathbb{I}[y \leq 0]$. The other plots show the loss functions $Q(y)$ for quadratic, exponential, logistic, hinge, and Perceptron designs — see expressions (64.37a)–(64.37f) for the definitions. It is seen from the graphs that, with the exception of the Perceptron loss, all other loss functions bound the $0/1-$loss from above. Although not seen in the figure, this fact is also true for the logistic loss if we re-scale it by $1/\ln 2$ to ensure that its value becomes one at $y = 0$.

functions assign more penalty to values of y smaller than one than to values of y larger than one.

(e) It is further seen from the figure that, with the exception of the Perceptron loss, all other loss functions bound the $0/1-$loss from above. Although not seen in the figure, this fact is also true for the logistic loss if we re-scale it by $1/\ln 2$ to ensure that its value becomes one at $y = 0$. This scaling by a constant value does not affect the solution of the corresponding optimization problem. For this reason, it is customary to list the logistic loss without the scaling by $1/\ln 2$.

(f) The five surrogate loss functions (64.37a)–(64.37e), and their corresponding empirical risk functions defined below are convex functions in w. This is a useful property because it helps ensure that optimization problems that seek to minimize the surrogate risks $P(w)$ will only have global minima.

Using the aforementioned losses, we can replace the empirical $0/1-$risk in (64.29) by any of the following expressions and continue to denote the minimizer by $w^\star$:

$$P(w) = \frac{1}{N}\sum_{n=0}^{N-1}\left(\gamma(n) - h_n^{\mathsf{T}}w\right)^2, \qquad \text{(quadratic risk)} \tag{64.38a}$$

$$P(w) = \frac{1}{N}\sum_{n=0}^{N-1}\ln\left(1+e^{-\gamma(n)h_n^{\mathsf{T}}w}\right), \qquad \text{(logistic risk)} \tag{64.38b}$$

$$P(w) = \frac{1}{N}\sum_{n=0}^{N-1}\max\left\{0,\,-\gamma(n)h_n^{\mathsf{T}}w\right\}, \qquad \text{(Perceptron risk)} \tag{64.38c}$$

$$P(w) = \frac{1}{N}\sum_{n=0}^{N-1}\max\left\{0,\,1-\gamma(n)h_n^{\mathsf{T}}w\right\}, \qquad \text{(hinge risk)} \tag{64.38d}$$

$$P(w) = \frac{1}{N}\sum_{n=0}^{N-1}e^{-\gamma(n)h_n^{\mathsf{T}}w}, \qquad \text{(exponential risk)} \tag{64.38e}$$

Example 64.3 (**Probability of misclassification**) A classifier that minimizes a surrogate empirical risk with small misclassification errors over the training data will still generalize well and deliver small misclassification errors over test data. To see this, let $w^\star$ denote the solution to one of the problems listed above, excluding the Perceptron risk. Its actual error rate is denoted by

$$R(w^\star) \;\triangleq\; \mathbb{P}(\boldsymbol{h}^{\mathsf{T}}w^\star \neq \boldsymbol{\gamma}) \;=\; \mathbb{E}\;\mathbb{I}[\boldsymbol{h}^{\mathsf{T}}w^\star \neq \boldsymbol{\gamma}] \tag{64.39}$$

whereas its empirical risk value is $P(w^\star)$ and its empirical error rate (misclassifications over the training data) is

$$R_{\text{emp}}(w^\star) = \frac{1}{N}\sum_{n=0}^{N-1}\mathbb{I}[\gamma(n)\widehat{\gamma}(n)\leq 0] \tag{64.40}$$

Now, observe from Fig. 64.11 that it is generally the case that the new loss functions bound the ideal $0/1-$loss function from above (with the exception of the Perceptron loss function, which we are excluding from this discussion), i.e., it holds that

$$\mathbb{I}[y\leq 0] \;\leq\; Q(y) \tag{64.41}$$

In this case, we get

$$R_{\text{emp}}(w^\star) \leq P(w^\star) \tag{64.42}$$

and it follows that

$$\begin{aligned} R(w^\star) &\leq R_{\text{emp}}(w^\star) \;+\; \delta \qquad \text{(by result (64.15))} \\ &\leq P(w^\star) \;+\; \delta \qquad \text{(by (64.42))} \end{aligned} \tag{64.43}$$

so that a small empirical risk value, $P(w^\star)$, translates into a small probability of misclassification, $R(w^\star)$, over the entire data distribution. A similar conclusion holds for more general classifier spaces, $\mathcal{C}$ (other than affine classifiers — see the discussion leading to (64.64) in the comments at the end of the chapter.

64.7 COMMENTARIES AND DISCUSSION

Curse of dimensionality. The designation "curse of dimensionality" is attributed to the American control theorist **Robert Bellman (1920–1964)**, who coined the term in his development of dynamic programming in Bellman (1957a); dynamic programming refers to a widely used class of mathematical optimization problems — discussed later in Chapter 44. We explained in Sec. 64.1 how the curse of dimensionality degrades the performance of learning strategies. This is because in higher dimensions, the available training data can only provide a sparse representation of the space. Moreover, as shown in Prob. 64.9, most of the training samples will concentrate close to the boundaries of the space. And it is common to encounter high-dimensional data in practice. For example, when DNA microarrays are used to measure the expression levels of a large number of genes, the dimension for this problem is on the order of $M \sim 10^4$. A useful theoretical study by Hughes (1968) illuminated how the curse of dimensionality degrades the performance of the Bayes classifier when a finite number, N, of training data is used to estimate conditional probabilities by using relative frequencies. It was shown in that work that, for a fixed N, the classification accuracy increases initially but then degrades as the dimensionality of the feature space, M, increases beyond some threshold value — see Prob. 64.30. From (64.1), we note that in order to design classifiers that perform well in higher-dimensional spaces, the number of training data, N, will need to increase exponentially fast with the dimension, M. In acknowledgment of Hughes' work, the curse of dimensionality problem is sometimes referred to as the *Hughes effect*.

Bias-variance tradeoff. The bias-variance relation (64.25) reflects an important tradeoff in the design of effective learning algorithms from training data. The relation expresses the difference between the optimal risk $R(c^\bullet)$ and the average performance $\mathbb{E}\,\boldsymbol{R}(c^\star)$ as the sum of two components. Ideally, a designer would like to keep both the bias and variance terms small. One degree of freedom that the designer has is the choice of the model set, $\mathcal{C}$. As explained in the text, a simple model set generally under-fits the data and leads to large bias but small variance. In contrast, a more complex model set generally over-fits the data and leads to small bias but large variance. A compromise needs to be struck by selecting classifier sets of moderate complexity — as illustrated in Fig. 64.9. This is one reason why it is often observed in practice that moderately complex classifiers perform better than more sophisticated classifiers. Some useful references that deal with the bias-variance tradeoff in the learning context and other related issues include the works by German, Bienenstock, and Doursat (1992), Kong and Dietterich (1995), Breiman (1994,1996a,b), Tibshirani (1996a), James and Hastie (1997), Kohavi and Wolpert (1996), Friedman (1997), Domingos (2000), James (2003), and Geurts (2005), as well as the text by Hastie, Tibshirani, and Friedman (2009).

Generalization theory. The Vapnik-Chervonenkis bound (64.17) is a reassuring statistical result; it asserts that, given a sufficient amount of training data, learning is feasible for moderately complex classifier models. This means that classifiers that perform well on the training data are able to generalize and deliver reliable classifications on test data. This result is one of the cornerstones of statistical learning theory and it resulted from the landmark work by Vapnik and Chervonenkis (1968,1971); its strength lies in the fact that the bound is *distribution-free*.

It is common to list the VC bound (64.17) in an alternative form where δ is fixed at some small constant value and the right-hand side bound is made to depend on N, δ, and the VC dimension, namely, as:

$$\mathbb{P}\left(\sup_{c\in\mathcal{C}}|R_{\text{emp}}(c)-R(c)|>\delta\right) \;\leq\; 8\,(Ne/\text{VC})^{\text{VC}}\;e^{-N\delta^2/32} \tag{64.44}$$

In comparison, the earlier form (64.17) fixes the right-hand side probability at some constant level ϵ and then specifies the attainable δ by means of relation (64.13) in

terms of ϵ, N, and the VC dimension. This earlier form motivates the *Probably Approximately Accurate* PAC designation introduced by Valiant (1984). More expansive treatments of the VC bound(s) appear in the monographs by Vapnik (1995,1998) and the textbooks by Fukunaga (1990), Kearns and Vazirani (1994), Devroye, Gyorfi, and Lugosi (1996), Vidyasagar (1997), Cherkassky and Mulier (2007), and Hastie, Tibshirani, and Friedman (2009). Accessible overviews on learning theory appear in Kulkarni, Lugosi, and Venkatesh (1998) and Vapnik (1999). An extension that applies to other *bounded* loss functions, besides the $0/1-$loss function, appears in Vapnik (1998) — see also Prob. 64.28. An interesting quote appears in Vapnik (1998) stating that *"nothing is more practical than a good theory"* repeating an earlier statement made in Lewin (1945) by the German-American social psychologist **Kurt Lewin (1890–1947)**.

We provide a derivation of the Vapnik-Chervonenkis inequality (64.44) in Appendices 64.B and 64.C; the argument is non-trivial and relies on several steps. We follow in these appendices the presentation given by Devroye, Gyorfi, and Lugosi (1996, Ch. 12). In their presentation, the coefficient appearing in the exponential factor in (64.44) is $N\delta^2/32$, while the coefficient appearing in the original bound given by Vapnik and Chervonenkis (1971) is $N\delta^2/8$ and corresponds to a tighter bound — see also the works by Blumer *et al.* (1989) and Cherkassky and Mulier (2007). This difference is not significant for the conclusions and arguments presented in our treatment; it is sufficient for our purposes to know that a bound exists and that this bound decays to zero as $N \to \infty$ at a uniform rate that is independent of the data distribution. The derivation used in Appendix 64.C relies on two famous inequalities. The first result is the *Hoeffding inequality*, which we encountered earlier in Appendix 3.B; it provides a bound on the probability of the sum of a collection of random variables deviating from their mean. This inequality is due to the Finnish statistician **Wassily Hoeffding (1914-1991)** and appeared in the work by Hoeffding (1963). Earlier related investigations appear in Chernoff (1952) and Okamoto (1958). The second inequality is known as *Sauer lemma* (also Sauer-Shelah lemma) in combinatorial analysis and is derived in Appendix 64.B. The result was derived independently by Sauer (1972) and Shelah (1972); a similar result also appeared in the work by Vapnik and Chervanenkis (1971).

Universally-consistent classifiers. The significance of the *distribution-free* property of the VC bound can be highlighted by commenting on the notion of *universal consistency*. Recalling the definitions introduced in Sec. 64.2, we let $c^{\blacktriangle}(h)$ denote the classifier that minimizes the empirical risk (64.7), while $c^{\bullet}(h)$ refers to the Bayes classifier and it minimizes the actual risk (64.10). Both solutions do not impose any restriction on the classifier set, which is indicated by the filled triangle and circle superscripts. The classifier $c^{\blacktriangle}(h)$ is determined from the training data and its structure depends on the sample size, N. This decision rule is said to be *consistent* if it satisfies the property:

$$\lim_{N\to\infty} R(c^{\blacktriangle}) \;=\; R(c^{\bullet}), \quad \text{almost surely} \tag{64.45}$$

In other words, the risk value that is attained by the empirical classifier should approach the optimal risk value for increasingly large datasets. If the consistency property holds *for all* data distributions $f_{\boldsymbol{\gamma},h}(\gamma,h)$, then the empirical decision rule, $c^{\blacktriangle}(h)$, is said to be *universally* consistent. Such decision rules would be desirable because the implication is that, regardless of the data distribution, sufficient training samples can make learning feasible. A remarkable result by Stone (1977) established that universally consistent classifiers exist. One notable example from this work is the asymptotic $k-$NN classifier when the value of k is selected to depend on N and satisfy the two conditions $k(N) \to \infty$ and $k/N \to 0$ as $N \to \infty$. However, and unfortunately, although $R(c^{\blacktriangle})$ can approach $R(c^{\bullet})$ asymptotically for any data distribution, it turns out that the convergence rate can be extremely slow; moreover, the performance for finite sample size can also be disappointing. For example, a result by Devroye (1982), strengthening an earlier conclusion by Cover (1968), shows that for any classification rule $c^{\blacktriangle}(h)$ and any $\epsilon > 0$ and finite integer N, there exists a data distribution $f_{\boldsymbol{\gamma},\boldsymbol{h}}(\gamma,h)$ with $R(c^{\bullet}) = 0$

and such that — see Prob. 64.31 and also Devroye, Gyorfi, and Lugosi (1996, p. 112):

$$R(c^{\blacktriangle}) \;\geq 0.5-\epsilon, \quad \text{for any finite } N \tag{64.46}$$

This conclusion shows that the finite-sample performance can be very bad for some distributions (in this case, the optimal Bayes risk is equal to zero and, yet, the risk by the empirical classifier is close to 1/2). It is also shown in Cover (1968) and Devroye (1982) that the convergence rate of $R(c^{\blacktriangle})$ towards $R(c^{\bullet})$ can be arbitrarily slow. Specifically, if $a(n) > 0$ denotes any monotonically decreasing sequence of positive numbers converging to zero, then for any classification rule $c^{\blacktriangle}(h)$ and any $\epsilon > 0$ and finite integer N, there exists a data distribution $f_{\boldsymbol{\gamma},\boldsymbol{h}}(\gamma, h)$ with $R(c^{\bullet}) = 0$ and such that:

$$R(c^{\blacktriangle}) \;\geq\; a(N), \quad \text{for any finite } N \tag{64.47}$$

As indicated by Devroye, Gyorfi, and Lugosi (1996, p. 114), statements (64.46)–(64.47) combined imply that *"good universally consistent classifiers do not exist."* In light of this conclusion, which also relates to the concept of "*no free lunch theorems*" discussed further ahead, we can now re-examine the VC bound (64.44). Similar to (64.20), this result implies that, for a fixed constant δ,

$$\mathbb{P}\Big(R(c^{\blacktriangle}) - R(c^{\bullet}) \geq 2\delta\Big) \;\leq\; 8\,(Ne/\mathrm{VC})^{\mathrm{VC}}\; e^{-N\delta^2/32} \tag{64.48}$$

and the bound holds for all finite N and for all data distributions. Recall that this result is obtained by restricting the search for $c^{\blacktriangle}(h)$ and $c^{\bullet}(h)$ to a set $c \in \mathcal{C}$ with a finite VC dimension (in which case $c^{\blacktriangle}(h)$ becomes $c^{\star}(h)$ and $c^{\bullet}(h)$ becomes $c^{o}(h)$). It is clear from (64.48) that $R(c^{\blacktriangle})$ can be made sufficiently close to $R(c^{\bullet})$ by selecting N large enough; moreover, with high probability, the convergence rate of $R(c^{\blacktriangle})$ towards $R(c^{\bullet})$ is $O(\ln(N)/N)$.

It is worth noting that the Vapnik-Chervonenkis bound (64.44) is a generalization of a famous result derived by the Russian mathematician **Valery Glivenko (1896–1940)** and the Italian mathematician **Francesco Cantelli (1875–1966)** in two separate publications by Glivenko (1933) and Cantelli (1933). The result is known as the Glivenko-Cantelli theorem and it describes the asymptotic behavior of the ensemble cumulative distribution function. Proofs appear in the works by Dudley (1978, 1999), Pollard (1984), Devroye, Gyorfi, and Lugosi (1996), and van der Vaart and Wellner (1996).

Glivenko-Cantelli theorem (Glivenko (1933), Cantelli (1933)). *Consider a collection of N independent and identically-distributed realizations, $\{x_n\}$, of a random variable $\boldsymbol{x}$ with a cumulative density function, $F(x)$. Introduce the ensemble average construction for $F(x)$:*

$$F_N(x) \;\triangleq\; \frac{1}{N}\sum_{n=0}^{N-1} \mathbb{I}\,[x_n \leq x] \tag{64.49}$$

where the indicator function on the right-hand side counts the number of sample values observed within the interval $(-\infty, x]$. It then holds that

$$\mathbb{P}\left(\sup_{x\in\mathbb{R}} |F_N(x) - F(x)| > \delta\right) \;\leq\; 8(N+1)e^{-N\delta^2/32} \tag{64.50}$$

No free lunch theorem. The results by Cover (1968) and Devroye (1982) revealing that the finite-sample performance of a classifier can be very bad for some distributions can also be explained from the perspective of the "no free lunch theorem," which we motivate as follows.

We have devised several learning algorithms in our treatment so far, such as logistic regression, support vector machines, kernel methods, and decision trees. We will

introduce additional learning algorithms in future chapters based on neural network architectures. But is there one "best" algorithm? It is observed in practice that some algorithms perform better on some data distributions and worse on other distributions. However, this does not mean that some algorithms are better than other algorithms. This conclusion is captured by a famous result known as the "*no free lunch theorem*" by Wolpert (1992,1996); see also Schaffer (1994) and Wolpert and Macready (1997). In broad terms, the theorem asserts that, averaged over all possible data distributions, the performance of every classification algorithm will be the *same* as other algorithms on test data! This means that no classifier can be proven to be universally better than all other classifiers and, as such, there is no "best" learning method.

Specifically, consider two learning algorithms, say, two binary classifiers $\mathbb{A}$ and $\mathbb{B}$. The first classifier could be based on logistic regression while the second classifier could be a support vector machine or a neural network. Both classifiers are trained to decide whether feature vectors belong to one class ($\gamma = +1$) or the other ($\gamma = -1$). The training data arises from some distribution $f_{\boldsymbol{\gamma},\boldsymbol{h}}(\gamma, h)$. Assume we assess the performance of the algorithms on test data generated from this same distribution and write down the classification error that each algorithm generates during this assessment phase. We may find that one of the classifiers performs better than the other, say, classifier $\mathbb{A}$ outperforms $\mathbb{B}$ in this assessment exercise (i.e., it yields a smaller classification error). Now, assume we repeat the experiment but change the data distribution this time. We train the classifiers and test their performance on a new distribution and write down the resulting classification errors for each. It may be the case for this new distribution that the same better-performing classifier $\mathbb{A}$ from the first assessment continues to outperform $\mathbb{B}$ in this second test. It may also be the case that classifier $\mathbb{B}$ outperforms $\mathbb{A}$. We could continue in this manner and compare the performance of both classifiers over all possible choices of data distributions. The "no free lunch theorem" states that, averaged over all choices of data distributions, the performance of the two classifiers will match! This means that better performance by one algorithm in some data situations will be offset by worse performance in other situations. This also means that no single learning algorithm can be expected to work best for *all* data distributions (i.e., for all types of problems). We encounter one manifestation of this property in Prob. 64.31 where we show that for any finite sample-size optimal classifier, there always exists a data distribution for which the empirical risk of the classifier is bad. The following is an alternative justification for this fact, and can be viewed as one form of a "free lunch theorem."

Consider a finite number of feature vectors, $\mathcal{H} = \{h \in \mathbb{R}^M\}$. Each feature vector has label $\gamma = +1$ or $\gamma = -1$. Let Γ be the collection of all possible mappings $\gamma(h) : \mathcal{H} \to \{+1, -1\}$. That is, every $\gamma(h) \in \Gamma$ assigns ± 1 labels to features in $\mathcal{H}$. There are a total of $2^{|\mathcal{H}|}$ possible mappings, $\gamma(h)$, in terms of the cardinality of the set $\mathcal{H}$. We will verify next that there exists some probability distribution over the feature vectors $h \in \mathcal{H}$ (which determines how they are selected or sampled from $\mathcal{H}$) and a choice of mapping $\gamma(h) \in \Gamma$ for which a trained classifier $c^\star(h)$ will perform poorly. For more details, the reader may refer to the useful discussion in Shalev-Shwartz and Ben-David (2014, Chapter 5).

Variation of no free lunch theorem (Wolpert (1992,1996)) *Consider an arbitrary learning algorithm that is trained on at most $N \leq |\mathcal{H}|/2$ data points $\{\gamma(n), h_n\}$ from $\mathcal{H}$. We denote the output generated by the algorithm by $c^\star(h) : \mathcal{H} \to \{+1, -1\}$. Then, there will exist a label mapping $\gamma(h) : \mathcal{H} \to \{+1, -1\}$ and a distribution $f_{\boldsymbol{h}}(h)$ over $\mathcal{H}$ such that*

$$\mathbb{P}\Big(c^\star(h) \neq \gamma(h)\Big) \geq 1/8 \;\; \textit{holds with probability of at least } 1/7 \tag{64.51}$$

In other words, there exists a mapping $\gamma(h)$ and a data distribution leading to bad performance.

Proof: We follow an argument similar to Shalev-Shwartz and Ben-David (2014, Sec. 5.1). We select $2N \leq |\mathcal{H}|$ independent and identically distributed (i.i.d.) feature vectors at random according to some distribution $\boldsymbol{h} \sim f_{\boldsymbol{h}}(h)$ from the set $\mathcal{H}$. We place N of these samples at random into a set $\mathcal{S}$ and use them to train a classification algorithm, say, by minimizing some empirical risk function. We keep the remaining samples for testing. The algorithm will generate some mapping $c^{\star}(h) : \mathcal{H} \rightarrow \{+1, -1\}$. For each feature vector $h \in \mathcal{H}$, the trained classifier will assign the label $c^{\star}(h)$. We have several elements of randomness involved in this setting: the distribution $\boldsymbol{h} \sim f_{\boldsymbol{h}}(h)$, the samples that end up in $\mathcal{S}$, and also the choice of the mapping $\gamma(h)$ from Γ that sets the labels of the feature vectors. We wish to examine the size of the probability of error, denoted by $P_e(\gamma) = \mathbb{P}(c^{\star}(h) \neq \gamma(h))$; this error depends on the mapping $\gamma(h)$. Obviously, the error will also depend on the distribution $f_{\boldsymbol{h}}(h)$ used to select the $2N$ feature vectors and on the randomness in defining the test set. For this reason, we will be interested in examining the average probability of error over these sources of randomness, namely, the quantity $P_{e,av}(\gamma) = \mathbb{E}_{\,\mathcal{S},h} P_e(\gamma)$. The worst value for the average error over the mappings $\gamma(h)$ is

$$\max_{\gamma(h)\in\Gamma} \Big\{ P_{e,av}(\gamma) \Big\} \overset{(a)}{\geq} \mathbb{E}_{\,\gamma}\, P_{e,av}(\gamma) \overset{(b)}{=} \mathbb{E}_{\,\mathcal{S}} \Big(\mathbb{E}_{\,\gamma,h}\, P_e(\gamma) \Big) \tag{64.52}$$

where step (a) is because the worst performance on the left is larger than the average performance on the right, and step (b) changes the order of the expectations. Now note that:

$$\begin{aligned}
&\mathbb{E}_{\,\gamma,h}\, P_e(\gamma) \\
&= \mathbb{E}_{\,\gamma,h}\, \mathbb{P}(c^{\star}(h) \neq \gamma(h)) \\
&= \mathbb{E}_{\,\gamma} \Big\{ \mathbb{P}(\boldsymbol{h} \in \mathcal{S}) \mathbb{P}\Big(c^{\star}(h) \neq \gamma(h) \,|\, h \in \mathcal{S} \Big) \;+\; \mathbb{P}(\boldsymbol{h} \notin \mathcal{S}) \mathbb{P}\Big(c^{\star}(h) \neq \gamma(h) \,|\, h \notin \mathcal{S} \Big) \Big\} \\
&\overset{(c)}{\geq} \frac{1}{2} \mathbb{E}_{\,\gamma}\, \mathbb{P}\Big(c^{\star}(h) \neq \gamma(h) \,|\, h \notin \mathcal{S} \Big)
\end{aligned} \tag{64.53}$$

where step (c) ignores the first term from the third line and uses the fact that only half of the selected features are used for training so that $\mathbb{P}(\boldsymbol{h} \notin \mathcal{S}) = 1/2$. We still need to evaluate the last expectation, which averages over the choice of the mapping $\gamma(h)$. Recall that $c^{\star}(h)$ is determined without knowledge of any of the features from outside $\mathcal{S}$. Moreover, since we are free to choose $\gamma(h)$, there are mappings that could result in $\gamma(h) = +1$ and others that could result in $\gamma(h) = -1$. Therefore, $c^{\star}(h)$ will be wrong half of the time:

$$\mathbb{P}\Big(c^{\star}(h) \neq \gamma(h) \,|\, h \notin \mathcal{S} \Big) = 1/2 \tag{64.54}$$

Substituting into (64.52) we conclude that $\max_{\gamma} P_{e,av}(\gamma) \geq 1/4$. This means that there exists a mapping $\gamma(h)$ and a distribution $f_{\boldsymbol{h}}(h)$ such that $\mathbb{E}_{\,\mathcal{S},h} P_e(\gamma) \geq 1/4$. Now, recall that $P_e(c)$ is a probability measure and it assumes values in the interval $[0, 1]$. Therefore, using the result of part (a) from Prob. 3.19 we conclude that

$$\mathbb{P}\Big(P_e(\gamma) \geq 1/8 \Big) \geq \frac{1/4 - (1 - 1/8)}{7/8} = 1/7 \tag{64.55}$$

as desired.

■

We conclude that a classifier that performs well on certain data distributions need not deliver similar performance on other distributions. This is more or less in line with intuition. An architecture that distinguishes well between images of cats and dogs need not perform well in distinguishing between poetry and prose. For this reason, when one learning algorithm is said to outperform another, this statement should be qualified to mean that one algorithm outperforms the other for the particular data distribution under consideration.

It is important to note that some criticism has been leveled at the "no free lunch theorem" and its implication for practical learning algorithms. This is because the statement of the theorem averages performance over *all* possible data distributions: these include distributions over which the classifier was not trained and, moreover, many of these distributions need not be reflective of how real-world data behave. For example, the work by Fernandez-Delgado *et al.* (2014) has shown that some learning algorithms consistently outperform other algorithms in real-data scenarios. Moreover, even if an algorithm $\mathbb{A}$ performs badly on some distributions, it may be the case that these distributions are not relevant for the problem at hand. For all practical purposes, a designer should seek learning algorithms that perform best on the problems (or distributions) of interest.

Surrogate loss functions. The Vapnik-Chervonenkis bound (64.44) is established in Appendix 64.C under the assumption that the risk values are computed relative to the ideal $0/1-$loss function. That is, the classifiers $\{c^o(h), c^\star(h)\}$ correspond to the minimizers of the actual and empirical risks defined by (64.11) and (64.12):

$$c^o(h) \triangleq \operatorname*{argmin}_{c\in\mathcal{C}} R(c), \quad c^\star(h) \triangleq \operatorname*{argmin}_{c\in\mathcal{C}} R_{\text{emp}}(c) \tag{64.56}$$

where

$$R(c) \triangleq \mathbb{E}\, \mathbb{I}[c(\boldsymbol{h}) \neq \boldsymbol{\gamma}], \quad R_{\text{emp}}(c) \triangleq \frac{1}{N}\sum_{n=0}^{N-1} \mathbb{I}[c(h_n) \neq \gamma(n)] \tag{64.57}$$

These expressions rely on the use of the ideal $0/1-$loss defined by:

$$Q(c;\boldsymbol{\gamma},\boldsymbol{h}) \triangleq \mathbb{I}[c(\boldsymbol{h}) \neq \boldsymbol{\gamma}] \tag{64.58}$$

In this way, the value of $R(c)$ is a measure of the probability of misclassification over the entire data distribution, while the value of $R_{\text{emp}}(c)$ is a measure of the fraction of erroneous classifications over the N training data points, $\{\gamma(n), h_n\}$. Given the difficulty in solving the optimal design problems (64.56), due to the nonsmooth nature of the indicator function, we motivated in Sec. 64.6 several surrogate convex losses (such as quadratic, logistic, hinge, exponential, and Perceptron functions). A natural question is to inquire about the generalization ability of classifiers designed under these alternative choices.

Thus let, more generally, $Q(c;\boldsymbol{\gamma},\boldsymbol{h})$ denote an arbitrary nonnegative convex loss function. For the purposes of the discussion in this section, We denote the surrogate risk by the notation:

$$P(c) \triangleq \mathbb{E}\, Q(c;\boldsymbol{\gamma},\boldsymbol{h}) \tag{64.59}$$

and the corresponding empirical risk by

$$P_{\text{emp}}(c) = \frac{1}{N}\sum_{n=0}^{N-1} Q(c;\gamma(n),h_n) \tag{64.60}$$

The quantities $\{P(c), P_{\text{emp}}(c)\}$ play the role of $\{R(c), R_{\text{emp}}(c)\}$ when the $0/1-$loss is used. Now, however, we are using more general loss functions, $Q(c;\cdot)$. We then replace problems (64.56) by

$$c^o(h) \triangleq \operatorname*{argmin}_{c\in\mathcal{C}} P(c), \quad c^\star(h) \triangleq \operatorname*{argmin}_{c\in\mathcal{C}} P_{\text{emp}}(c) \tag{64.61}$$

where we continue to use the notation $(c^o, c^\star)$ in order to avoid an explosion of symbols. It turns out that an inequality of the Vapnik-Chervonenkis type continues to hold in this more general case if we assume that, for any $c \in \mathcal{C}$, the loss function $Q(c;\boldsymbol{\gamma},\boldsymbol{h})$ is bounded, say, its values lie within some interval $[a, b]$ for nonnegative scalars $a < b$. If

we examine the derivation of inequality (64.111) in Appendix 64.C, we will be able to recognize that, under this boundedness condition, a similar bound continues to hold for more general loss functions with the exponent $-N\delta^2/32$ now replaced by $-N\delta^2/32b^2$; see Prob. 64.28:

$$\mathbb{P}\left(\sup_{c\in\mathcal{C}}|P_{\text{emp}}(c^\star)-P(c^o)|>\delta\right) \;\leq\; Ke^{-N\delta^2/32b^2} \tag{64.62}$$

for some constant K that is independent of the data distribution. The ultimate conclusion is that the bound continues to decay to zero as $N\to\infty$ at a uniform rate that is also independent of the data distribution. Further discussion on this result can be found in Vapnik and Chervonenkis (1968,1971), Dudley, Gine, and Zinn (1991), Alon *et al.* (1997), Vapnik (1998), Cucker and Smale (2002), and Rosasco *et al.* (2004).

Next, following steps similar to the ones that led to (64.21) we can then conclude that with high probability, $|P_{\text{emp}}(c^\star)-P(c^o)|\leq 3\delta$. If the loss function further satisfies $\mathbb{I}[c(h)\neq\boldsymbol{\gamma}]\leq Q(c;\gamma,h)$ for any $c\in\mathcal{C}$, then it will hold that

$$R_{\text{emp}}(c^\star)\leq P_{\text{emp}}(c^\star) \tag{64.63}$$

Applying these inequalities to the optimal classifier $c^\star(h)$ from (64.61), we conclude that

$$R(c^\star)\overset{(64.15)}{\leq} R_{\text{emp}}(c^\star)+\delta\;\leq P_{\text{emp}}(c^\star)+\delta \tag{64.64}$$

so that a small $P_{\text{emp}}(c^\star)$ translates into a small probability of misclassification for $c^\star(h)$. In other words, learning from data for general loss functions is still feasible. The main limitation in the argument leading to this conclusion is the requirement that the loss function $Q(c;\gamma,h)$ be bounded for any $c\in\mathcal{C}$.

Rademacher complexity There is an alternative method to examine the generalization ability of learning algorithms for more general loss functions by relying on the concept of the *Rademacher complexity*. We pursue this approach in Appendix 64.D. Recall that the analysis in the body of the chapter has shown that classification structures with medium VC dimensions are able to learn well with high likelihood for *any* data distribution. In a sense, this conclusion amounts to a generalization guarantee under a *worst case* scenario since it holds irrespective of the data distribution. It is reasonable to expect that some data distributions will be more favorable than others and, therefore, it is desirable to seek generalization results that have some dependence on the data distribution. The framework that is based on the Rademacher complexity allows for this possibility and leads to tighter generalization error bounds. The approach also applies to multiclass classification problems and to other loss functions, and is not restricted to binary classification or $0/1-$losses. The analysis carried out in Appendix 64.D continues to lead to similar reassuring conclusions about the ability of learning methods to generalize for mild VC dimensions. However, the conclusions are now dependent on the data distribution and will not correspond to worst-case statements that hold for any distribution. The main results in the appendix are the one and two-sided generalization bounds (64.182a)–(64.182b) and (64.197a)–(64.197b). The derivation of these results relies on two critical tools known as the McDiarmid inequality, which we encountered earlier in (3.259a) and is due to McDiarmid (1989), and the Massart lemma (64.145) due to Massart (2000,2007). The first works to use the Rademacher complexity to study the generalization ability of learning algorithm are by Koltchinskii (2001), Koltchinskii and Panchenko (2000,2002), Bartlett, Boucheron, and Lugosi (2001), Barlett and Mendelson (2002), Mendelson (2002), Antos *et al.* (2002), and Bartlett, Bousquet, and Mendelson (2005). Overviews and further treatments appear in Boucheron, Bousquet, and Lugosi (2005), Shalev-Shwartz and Ben-David (2014), Mohri, Rostamizadeh, and Talwalkar (2018), and Wainwright (2019). The designation Rademacher complexity is motivated by the connection to the discrete *Rademacher distribution*, named after the German-American mathematician **Hans Rademacher (1892-1969)**, which refers to

random variables that assume the values ± 1 with equal probability. A sum of such variables leads to a random walk with symmetry where it is equally likely to move in one direction or the other. The Rademacher distribution is related to the standard Bernoulli distribution: the former deals with values $\{+1, -1\}$ chosen with probability $1/2$ each, while the latter deals with values $\{1, 0\}$ chosen with probabilities $\{p, 1-p\}$.

PROBLEMS

64.1 Let $t(h) = \mathbb{P}(\boldsymbol{\gamma} = +1 | \boldsymbol{h} = h)$.

(a) For any classifier c, derive the following expression for the excess-risk:

$$R(c) - R(c^{\bullet}) \;=\; \mathbb{E}_h \Big(\, |2t(\boldsymbol{h}) - 1| \; \mathbb{I}[c^{\bullet}(\boldsymbol{h}) \neq c(\boldsymbol{h})] \Big)$$

where the expectation is over the distribution of the feature data.

(b) Show that the optimal Bayes risk is given by $R(c^{\bullet}) = \mathbb{E}_h \Big\{ \min\big(t(\boldsymbol{h}),\; 1 - t(\boldsymbol{h})\big) \Big\}$.

(c) Show also that $R(c^{\bullet}) \;=\; \frac{1}{2}(1 - \mathbb{E}_h |2t(\boldsymbol{h}) - 1|)$.

64.2 Continuing with Prob. 64.1, let $\pi_{\pm 1}$ denote the prior probabilities of classes $\boldsymbol{\gamma} \in \{\pm 1\}$. That is, $\pi_{+1} = \mathbb{P}(\boldsymbol{\gamma} = +1)$ and likewise for π_{-1}. Assume the feature data, $\boldsymbol{h}$, has a continuous conditional probability distribution, $f_{\boldsymbol{h}|\boldsymbol{\gamma}}(h|\gamma)$.

(a) Verify that

$$R(c^{\bullet}) \;=\; \int_{h \in \mathcal{H}} \min\big\{ \pi_{+1} f_{\boldsymbol{h}|\boldsymbol{\gamma}}(h|\gamma = +1),\; \pi_{-1} f_{\boldsymbol{h}|\boldsymbol{\gamma}}(h|\gamma = -1) \big\} \, dh$$

where the integration is over the feature space, $h \in \mathcal{H}$.

(b) Assume $\pi_{+1} = \pi_{-1} = 1/2$. Conclude that in this case:

$$R(c^{\bullet}) \;=\; \frac{1}{2} \left\{ 1 - \frac{1}{2} \int_{h \in \mathcal{H}} \Big| f_{\boldsymbol{h}|\boldsymbol{\gamma}}(h|\gamma = +1) \;-\; f_{\boldsymbol{h}|\boldsymbol{\gamma}}(h|\gamma = -1) \Big| \, dh \right\}$$

In other words, the Bayes risk is related to the $\mathcal{L}_1-$distance between the two conditional distributions of the feature data.

64.3 Refer to expression (64.7) for the empirical risk. Assume $\{\gamma(n), h_n\}$ are independent and identically distributed realizations of $\{\boldsymbol{\gamma}, \boldsymbol{h}\}$.

(a) Argue that each term of the form $\mathbb{I}[c(\boldsymbol{h}) \neq \boldsymbol{\gamma}]$ is a binomial random variable with probability parameter $p = R(c)$.

(b) Conclude that the mean and variance of $R_{\text{emp}}(c)$ are given by p and $p(1-p)/N$, respectively,

(c) Use Chebyshev bound (3.28) to conclude that, for any scalar $\delta > 0$,

$$\mathbb{P}\Big(|R_{\text{emp}}(c) - R(c)| \;\geq \delta \Big) \;\leq\; \frac{p(1-p)}{N\delta^2}$$

64.4 Let $\{\boldsymbol{x}_n,\; n = 1, \ldots, N\}$ denote N independent random variables, with each variable satisfying $a_n \leq \boldsymbol{x}_n \leq b_n$. Let $\boldsymbol{S}_N = \sum_{n=1}^{N} \boldsymbol{x}_n$ denote the sum of these random variables. Let $\Delta = \sum_{n=1}^{N} (b_n - a_n)^2$ denote the sum of the squared lengths of the respective intervals. A famous inequality known as Hoeffding inequality was derived in Appendix 3.B; it asserts that for any $\delta > 0$:

$$\mathbb{P}\Big(|\boldsymbol{S}_N \;-\; \mathbb{E}\, \boldsymbol{S}_N| \;\geq \delta \Big) \;\leq\; 2e^{-2\delta^2/\Delta}$$

Now, refer to expression (64.7) for the empirical risk. Use Hoeffding inequality to establish that, for any particular classifier c and $\delta > 0$, it holds:

$$\mathbb{P}\Big(|R_{\text{emp}}(c) - R(c)| \geq \delta \Big) \leq 2e^{-2N\delta^2}$$

In comparison with the bound obtained in part (c) of Prob. 64.3, observe that the bound on the right-hand side of the above expression decays exponentially with the size of the training data. Let $\epsilon = 2e^{-2N\delta^2}$. Conclude that the above bound asserts that $\mathbb{P}(\,|R_{\text{emp}}(c) - R(c)| \geq \delta) \leq \epsilon$ for any small $\epsilon > 0$ and where δ and ϵ are related via:

$$\delta = \sqrt{\frac{1}{2N} \ln\left(\frac{2}{\epsilon}\right)}$$

Remark. This result shows that the true and empirical risk values get closer to each other as the number of training samples, N, increases. However, this conclusion assumes a fixed classifier, c. See the extensions studied in future Probs. 64.24 and 64.25.

64.5 We reconsider the discussion on surrogate risk functions from Sec. 64.6. Consider an arbitrary predictor function $\widehat{\gamma}(h) : \mathbb{R}^M \to \mathbb{R}$, which maps feature vectors h into real-valued predictions $\widehat{\gamma}$ for their labels. For each h, let $y = \gamma\widehat{\gamma}$ denote the corresponding margin variable with surrogate loss denoted by $Q(y) : \mathbb{R} \to \mathbb{R}$, for some loss function $Q(\cdot)$ to be selected. In the body of the chapter we listed several choices for $Q(\cdot)$ in (64.37a)–(64.37f). We associate with each $Q(\cdot)$ the *stochastic* risk function $P(\widehat{\gamma}) = \mathbb{E}\, Q(\boldsymbol{y})$, where the expectation is over the distribution of the data $\{\boldsymbol{\gamma}, \boldsymbol{h}\}$.

(a) Let $t(h) = \mathbb{P}(\boldsymbol{\gamma} = +1 | \boldsymbol{h} = 1)$. By conditioning on $\boldsymbol{h} = h$, verify that

$$\mathbb{E}\left(Q(\boldsymbol{y})|\boldsymbol{h} = h\right) = t(h)Q(\widehat{\gamma}(h)) + (1 - t(h))Q(-\widehat{\gamma}(h))$$

The right-hand side is a function of $\widehat{\gamma}$ and we denote it more compactly by $P(\widehat{\gamma}|h) = tQ(\widehat{\gamma}) + (1-t)Q(-\widehat{\gamma})$.

(b) We know that the optimal Bayes classifier assigns $\widehat{\gamma}_{\text{Bayes}}(h) = +1$ when $t(h) > 1/2$ and $\widehat{\gamma}_{\text{Bayes}}(h) = -1$ when $t(h) < 1/2$. We wish to select convex loss functions $Q(y)$ such that $P(\widehat{\gamma}|h)$ ends up having a negative minimizer $\widehat{\gamma}$ when $t < 1/2$ and a positive minimizer $\widehat{\gamma}$ when $t > 1/2$. When this happens, the sign of the minimizer $\widehat{\gamma}$ will match the optimal Bayes decision. Show that this occurs if, and only if, the convex loss $Q(y)$ is differentiable at $y = 0$ with a negative derivative value at that location (i.e., $Q'(0) < 0$). *Remark.* The reader may refer to Bartlett, Jordan, and McAuliffe (2006) for a related discussion. In the language of this reference, convex loss function $Q(\cdot)$ that satisfy these two conditions are said to be classification-calibrated.

64.6 Consider a hypercube in $M-$dimensions with edge length equal to one. Let h_o represent a particular feature vector located somewhere inside this hypercube. Assume there are a total of N feature vectors distributed uniformly inside the hypercube. We center a smaller hypercube around h_o with edge size ℓ.

(a) Assume $M = 3$. Determine the value of ℓ such that the volume of the smaller hypercube around h_o captures 10% of the N training samples.

(b) Assume now $M = 20$. Determine the value of ℓ such that the volume of the smaller hypercube around h_o captures the same fraction, 10%, of the N training samples. Compare the result with part (a).

64.7 Consider a hypercube in $M-$dimensions with edge size equal to one. Consider a smaller cube with edge size ℓ. What should the length ℓ be for the volume of the smaller cube to correspond to 1% of the volume of the larger cube? Determine ℓ for both cases of $M = 10$ and $M = 100$. What do you observe?

64.8 Refer to the volume expression (64.3).

(a) Assume $M = 2K$ is even. Show that the expression reduces to $(1/4)^K \pi^K / K!$.

(b) Show that it tends to zero as $M \to \infty$.

64.9 Assume N feature vectors are distributed uniformly inside a hyper-sphere in $M-$dimensions centered at the origin and of radius equal to one. Let $\boldsymbol{d}$ denote the distance from the origin to the closest training point; this variable is random in nature. Show that the median value of $\boldsymbol{d}$ is given by

$$\text{median}(\boldsymbol{d}) \;=\; \left(1-\frac{1}{2^{1/N}}\right)^{1/M}$$

Assume $M=20$ and $N=1000$. What is the median of $\boldsymbol{d}$? What do you conclude?

64.10 Refer to definition (64.11) for $c^o(h)$, where $R(c)=\mathbb{P}(c(\boldsymbol{h})\neq\boldsymbol{\gamma})$. Show that any solution c^o that results in $R(c^o)=0$ also leads to $R_{\text{emp}}(c^o)=0$, where the empirical risk is defined by (64.7). Conclude that if a solution c^o exists such that $R(c^o)=0$, then the Bayes classifier generates zero classification errors.

64.11 Refer to the alternate loss functions (64.37a)–(64.37e), and their corresponding risks. Show that all these functions are convex in w. Is the ideal $0/1-$loss function convex in w?

64.12 Consider feature vectors $h\in\mathbb{R}^2$. Give an example of 3 feature vectors that cannot be shattered by the class of linear classifiers. Does this fact contradict the conclusion that the VC dimension is three?

64.13 **(True or false)** The VC dimension of a class of classifiers is the value d for which any number $N>d$ of training samples cannot be shattered by this class of classifiers?

64.14 Show that the VC dimension of the class of linear classifiers $c(h)=\text{sign}(h^{\mathsf{T}}w)$ over $\mathbb{R}^M$ is equal to M.

64.15 Consider a collection of $M+2$ vectors in $\mathbb{R}^M$ denoted by $\mathcal{X}=\{x_1,x_2,\ldots,x_{M+2}\}$. *Radon theorem* states that every such set can be split into two disjoint subsets, denoted by $\mathcal{X}_1$ and $\mathcal{X}_2$, such that the convex hulls of $\mathcal{X}_1$ and $\mathcal{X}_2$ intersect with each other.

(a) Establish the validity of Radon theorem.

(b) Use Radon theorem to conclude that the VC dimension of the class of affine classifiers $c(h)=\text{sign}(h^{\mathsf{T}}w-\theta)$ over $\mathbb{R}^M$ is equal to $M+1$.

Remark. The theorem is due to Radon (1921). See Mohri, Rostamizadeh, and Talwalkar (2018) for a related discussion.

64.16 Consider the class of classifiers that consists of circles centered at the origin in $\mathbb{R}^2$, where feature vectors inside the circle belong to class -1 and feature vectors outside the circle belong to class $+1$. What is the VC dimension of this class of classifiers over $\mathbb{R}^2$?

64.17 Consider a class of classifiers defined by two scalar parameters $a\leq b$; the parameters define an interval $[a,b]$ on the real line. A scalar feature value h is declared to belong to class $+1$ if $h\in[a,b]$ (i.e., h lies inside the interval); otherwise, h is declared to belong to class -1. Show that the VC dimension of this class of classifiers is equal to 2. What is the shatter coefficient for this class of classifiers?

64.18 Consider a class of classifiers defined by four scalar parameters $a\leq b<c\leq d$; the parameters define two disjoint intervals $[a,b]$ and $[c,d]$ on the real line. A scalar feature value h is declared to belong to class $+1$ if $h\in[a,b]$ or $h\in[c,d]$ (i.e., h lies inside one of the intervals); otherwise, h is declared to belong to class -1. Show that the VC dimension of this class of classifiers is equal to 4.

64.19 Consider the class of classifiers that consists of two separate co-centric circles centered at the origin in $\mathbb{R}^2$, where feature vectors that lie in the ring between both circles belong to class -1 and feature vectors outside this area belong to class $+1$.

(a) What is the VC dimension of this class of classifiers over $\mathbb{R}^2$?

(b) If we replace the circles by co-centric spheres centered at the origin in $\mathbb{R}^3$, what would the VC dimension be?

64.20 Consider feature vectors $h\in\mathbb{R}^2$, which represent points in the plane. The classifier class consists of rectangles with vertical or horizontal edges. Points that fall inside a rectangle are declared to belong to class $+1$ and points that fall outside the rectangle are declared to belong to class -1. Show that the VC dimension for this class of classifiers is equal to 4.

64.21 Consider feature vectors $h \in \mathbb{R}^2$, which represent points in the plane. The classifier class consists of squares with vertical edges. Points that fall inside a square are declared to belong to class $+1$ and points that fall outside the square are declared to belong to class -1. Show that the VC dimension for this class of classifiers is equal to 3.

64.22 Refer to the VC bound in (64.17). How many training samples, N, do we need in order to ensure that the error between the actual and empirical risks is no larger than a prescribed value δ with probability of a least $1-\epsilon$. Compute the numerical value for N when $\delta = 5\% = \epsilon$ and VC $= 20$.

64.23 Refer again to the VC bound in (64.17). At what rate does the error between the actual and empirical risks decay as a function of the sample size, N?

64.24 The bound derived in Prob. 64.4 is applicable to a single classifier, c. We can derive a uniform bound over all classifiers as follows. Assume first that the number of classifiers in the set $\mathcal{C}$ is finite, i.e., $|\mathcal{C}| < \infty$.

(a) Argue that

$$\mathbb{P}\left(\sup_{c\in\mathcal{C}} |R_{\text{emp}}(c) - R(c)| \geq \delta\right) \leq \sum_{c\in\mathcal{C}} \mathbb{P}\Big(|R_{\text{emp}}(c) - R(c)| \geq \delta\Big)$$

(b) Conclude that, for any $\delta > 0$:

$$\mathbb{P}\left(\sup_{c\in\mathcal{C}} |R_{\text{emp}}(c) - R(c)| \geq \delta\right) \leq 2|\mathcal{C}|e^{-2N\delta^2}$$

(c) Conclude that, for any small $\epsilon > 0$:

$$\mathbb{P}\left(\sup_{c\in\mathcal{C}} |R_{\text{emp}}(c) - R(c)| \geq \delta\right) \leq \epsilon$$

where δ and ϵ are related via (compare with (64.13)):

$$\delta = \sqrt{\frac{1}{2N}\left(\ln|\mathcal{C}| + \ln\left(\frac{2}{\epsilon}\right)\right)}$$

(d) Conclude further that, for given (δ, ϵ) values, the amount of training samples that is necessary to ensure the bound from part (c) is

$$N \geq \frac{1}{2\delta^2}\ln\left(\frac{2|\mathcal{C}|}{\epsilon}\right)$$

so that more complex models require more data for training.

64.25 We continue with Prob. 64.24.

(a) When the number of classifiers in $\mathcal{C}$ is not necessarily finite, but the set has a finite VC dimension, it can be shown that the quantity $|\mathcal{C}|$ that appears on the right-hand side in the bound in part (b) should be replaced by $4(Ne/\text{VC})^{\text{VC}}$, and the scalar $2N\delta^2$ in the exponent should be replaced by $N\delta^2/32$ — see (64.111) and (64.88) in Appendix 64.C. Use this fact to conclude that for any small $\epsilon > 0$, it holds that

$$\mathbb{P}\left(\sup_{c\in\mathcal{C}} |R_{\text{emp}}(c) - R(c)| \geq \delta\right) \leq \epsilon$$

where δ and ϵ are now related via (compare with (64.13)):

$$\delta = \sqrt{\frac{8}{N}\left(\text{VC}\ln\left(\frac{Ne}{\text{VC}}\right) + \ln\left(\frac{4}{\epsilon}\right)\right)}$$

(b) An alternative bound can be obtained as follows for finite VC dimensions. It can also be shown that the quantity $|\mathcal{C}|$ that appears on the right-hand side in the bound in part (b) can be replaced by $4(N+1)^{\mathrm{VC}}$, and the scalar $2N\delta^2$ in the exponent can be replaced by $N\delta^2/32$ — see (64.111) and (64.87) in Appendix 64.C. Use this fact to conclude that for any small $\epsilon > 0$, it also holds that

$$\mathbb{P}\left(\sup_{c\in\mathcal{C}} |R_{\mathrm{emp}}(c) - R(c)| \geq \delta\right) \leq \epsilon$$

where δ and ϵ are related via:

$$\delta = \sqrt{\frac{32}{N}\left(\mathrm{VC}\,\ln(N+1) + \ln\left(\frac{8}{\epsilon}\right)\right)}$$

64.26 Refer to the result of Prob. 64.25. Show that during the training phase with N data points, it holds that

$$\mathbb{P}\Big(|R_{\mathrm{emp}}(c) - R(c)| \geq \delta\Big) \leq 4\left(\frac{2Ne}{\mathrm{VC}}\right)^{\mathrm{VC}} e^{-N\delta^2/8}$$

while during the testing phase, also using a total of N test data points, and after the classifier $c^\star(h)$ has been selected, it holds that

$$\mathbb{P}\Big(|R_{\mathrm{emp}}(c^\star) - R(c^\star)| \geq \delta\Big) \leq 2e^{-2N\delta^2}$$

Explain the difference.

64.27 Follow arguments similar to those employed in the derivation of the Vapnik-Chervonenkis inequality (64.111) in Appendix 64.C to establish the Glivenko-Cantelli inequality (64.50).

64.28 Let $Q(c;\gamma,h)$ denote an arbitrary nonnegative convex loss function that is assumed to be bounded, say, $Q(c;\gamma,h) \in (a,b)$ for some nonnegative scalars a, b and for any $c \in \mathcal{C}$ (i.e., for any choice in the classifier set under consideration). Define the corresponding surrogate risk function $P(c) = \mathbb{E}\,Q(c;\boldsymbol{\gamma},\boldsymbol{h})$. In the text, we used the indicator function $\mathbb{I}[c(\boldsymbol{h}) \neq \boldsymbol{\gamma}]$ in expression (64.5) instead of $Q(c;\boldsymbol{\gamma},\boldsymbol{h})$. Likewise, define the empirical risk over a set of N training points $\{\gamma(n), h_n\}$ as

$$P_{\mathrm{emp}}(c) \triangleq \frac{1}{N}\sum_{n=0}^{N-1} Q(c;\gamma(n),h_n)$$

Follow arguments similar to those employed in the derivation of the Vapnik-Chervonenkis inequality (64.111) in Appendix 64.C to establish that a similar bound holds for these more general loss and risk functions with the exponent $-N\delta^2/32$ replaced by $-N\delta^2/32b^2$.

64.29 Refer to Sauer inequality (64.86). Several useful bounds on the shatter coefficient are given in the text by Devroye, Gyorfi, and Lugosi (1996). In particular, verify that the following bounds hold for the shatter coefficient of a class $\mathcal{C}$ of classifiers applied to N feature vectors:

$$\begin{aligned}
&\mathcal{S}(\mathcal{C},N) \leq N^{\mathrm{VC}} + 1, && \text{for all VC}\\
&\mathcal{S}(\mathcal{C},N) \leq N^{\mathrm{VC}}, && \text{for VC} > 2\\
&\mathcal{S}(\mathcal{C},N) \leq e^{NH\left(\mathrm{VC}/N\right)}, && \text{for } N \geq 1 \text{ and VC} < N/2
\end{aligned}$$

where $H(p)$ denotes the entropy measure for a binomial random variable with parameter $p \in (0,1)$, i.e., $H(p) = -p\log_2 p - (1-p)\log_2(1-p)$.

64.30 Consider a binary classification problem with classes $\boldsymbol{\gamma} \in \{\pm 1\}$ having known prior probabilities denoted by π_{+1} and $\pi_{-1} = 1 - \pi_{+}$. Let $\boldsymbol{h}(m)$ denote the $m-$th entry of the feature vector $\boldsymbol{h} \in \mathbb{R}^M$ and assume it is a discrete random variable. Refer to the optimal Bayes classifier.

(a) Argue that the probability of correct decisions is given by

$$\mathbb{P}(\text{correct decisions}) \;=\; \sum_{m=1}^{M} \max_{\gamma=\pm 1} \Big\{ \pi_{\gamma}\, \mathbb{P}(\boldsymbol{h}(m) = h(m) \,|\, \boldsymbol{\gamma} = \gamma) \Big\}$$

(b) Assume first that $\pi_{+1} < \pi_{-1}$. The expression derived in part (a) is dependent on the data, $\{h(m)\}$. Averaging over all possible distributions for the data, show that the *average* accuracy of the Bayes classifier is given by:

$$\mathbb{P}_{av}(\text{correct decisions}) = \pi_{+1} + \pi_{-1}(M-1)\left(\frac{\pi_{+1}}{\pi_{-1}}\right)^{M} \Delta$$

where

$$\Delta = \sum_{m=0}^{M} \frac{M!}{m!(M-m)!(2M-m-1)\,[\pi_{+1}/(1-2\pi_{+1})]^m}$$

Let $M \to \infty$ and conclude that $\mathbb{P}_{av} \to (1 - \pi_{-1}\pi_{+1})$. What does this result mean?

(c) When $\pi_{+1} = \pi_{-1} = 1/2$, show that

$$\mathbb{P}_{av}(\text{correct decisions}) \;=\; \frac{3M-2}{4M-2} \;\overset{M\to\infty}{\longrightarrow}\; 0.75$$

(d) What is the value of $\mathbb{P}_{\text{av}}$ when $M = 1$? Is this expected?

64.31 In this problem, we establish result (64.46), namely, that for any finite sample-size empirical classifier, $c^{\blacktriangle}(h)$, there always exists a data distribution for which the empirical risk is bad. Here, $c^{\blacktriangle}(h)$ denotes the classifier that minimizes (64.7). Assume the data $(\boldsymbol{h}, \boldsymbol{\gamma})$ is constructed as follows. The feature variable $\boldsymbol{h}$ is a discrete scalar random variable satisfying:

$$\mathbb{P}(\boldsymbol{h} = s) = \frac{1}{K}, \quad \text{for} \;\; s = 0, ..., K-1$$

Consider a real number $b \in [0, 1)$ and introduce its binary expansion written in the form $b = 0.b_0 b_1 b_2 \cdots b_{K-1}$, where each b_j is either 0 or 1. The label $\boldsymbol{\gamma}$ corresponding to $\boldsymbol{h} = s$ is set to $\boldsymbol{\gamma} = b_s$. Observe that in this description, and without any loss in generality, we are setting the binary label to the values $\{0, 1\}$ rather than $\{-1, 1\}$ used in the body of the text.

(a) Argue that the risk of the optimal Bayes classifier is zero.

(b) Using the training data set $\mathcal{D}_N \triangleq \{(\boldsymbol{h}_0, \boldsymbol{\gamma}(0)), ..., (\boldsymbol{h}_{N-1}, \boldsymbol{\gamma}(N-1))\}$, we estimate the label $\boldsymbol{\gamma}$ for a feature vector $\boldsymbol{h}$ by employing the empirical classifier $c^{\blacktriangle}(\cdot)$:

$$\widehat{\boldsymbol{\gamma}} = c^{\blacktriangle}(\boldsymbol{h})$$

Assume the training data set $\{(\boldsymbol{\gamma}(n), \boldsymbol{h}_n\}_{n=0}^{N-1}$ and the test data $(\boldsymbol{\gamma}, \boldsymbol{h})$ are generated by the same process described previously. Let us denote the actual risk of $c^{\blacktriangle}(\cdot)$, parameterized by b, by the notation:

$$R(c^{\blacktriangle}; b) \triangleq \mathbb{P}(c^{\blacktriangle}(\boldsymbol{h}) \neq \boldsymbol{\gamma}]$$

We next model $\boldsymbol{b}$ as a random variable that is uniformly distributed in $[0, 1)$ and has binary expansion $\boldsymbol{b} = 0.\boldsymbol{b}_0 \boldsymbol{b}_1 \boldsymbol{b}_2 \cdots \boldsymbol{b}_{K-1}$. What is the value of $\mathbb{P}(\boldsymbol{b}_j = 0)$ for any j? Prove that for any empirical classifier $c^{\blacktriangle}(h)$ we have

$$\sup_{b \in [0,1)} R(c^{\blacktriangle}; b) \geq \mathbb{E}_b \, R(c^{\blacktriangle}; \boldsymbol{b})$$

where the expectation is over the distribution of $\boldsymbol{b}$.

(c) Assume that $\boldsymbol{b}$ is independent of the test vector $\boldsymbol{h}$ and the training vectors $\{\boldsymbol{h}_n\}_{n=0}^{N-1}$. Prove that

$$\mathbb{E}_b\, R(c^{\blacktriangle};\boldsymbol{b}) \geq \frac{1}{2}\left(1-\frac{1}{K}\right)^N$$

What can we conclude about the lower bound on $\sup_{b\in[0,1)} R(c^{\blacktriangle};b)$ as $K\to\infty$? Comment on the result.

64.32 Verify that the supremum function is convex, i.e., for any two sequences $\{x_n, x'_n\}$ and $\alpha\in[0,1]$:

$$\sup_n\Big(\alpha x_n + (1-\alpha)x'_n\Big) \leq \alpha\sup_n x_n + (1-\alpha)\sup_n x'_n$$

64.33 Consider a subset $\mathcal{A}\subset\mathbb{R}^N$, with finite cardinality, and refer to its Rademacher complexity defined by (64.142). Introduce the convex hull of $\mathcal{A}$, denoted by $\mathrm{conv}(\mathcal{A})$, which is the set of all convex combinations of elements in $\mathcal{A}$. Show that the sets $\mathcal{A}$ and $\mathrm{conv}(\mathcal{A})$ have the same Rademacher complexity. *Remark.* See Bartlett and Mendelson (2002, Sec. 3) and Shalev-Shwartz and Ben-David (2014, Ch. 26).

64.34 Consider a subset $\mathcal{A}\subset\mathbb{R}^N$ and refer to its Rademacher complexity defined by (64.142). Let $\phi(x):\mathbb{R}\to\mathbb{R}$ denote a $\delta-$Lipschitz function satisfying $|\phi(x)-\phi(y)|\leq\delta|x-y|$, for all $x,y\in\mathrm{dom}(\phi)$ and some $\delta>0$. We denote the entries of each $a\in\mathcal{A}$ by $a=\mathrm{col}\{a_n\}$, for $n=1,2,\ldots,N$. We define the transformation $\phi(a)$, with vector argument a, as the vector that results from applying $\phi(\cdot)$ to each individual entry of a, i.e., $\phi(a)=\mathrm{col}\{\phi(a_n)\}$. Consider the set $\mathcal{A}_\phi=\{\phi(a),\ a\in\mathcal{A}\}$. In other words, $\mathcal{A}_\phi$ is obtained by applying the Lipschitz continuous function $\phi(\cdot)$ to the elements of $\mathcal{A}$. Show that the Rademacher complexity is modified as follows

$$\mathcal{R}_N(\mathcal{A}_\phi)\leq\delta\,\mathcal{R}_N(\mathcal{A})$$

Remark. See Ledoux and Talagrand (1991), Kakade, Sridharan, and Tewari (2008), and Shalev-Shwartz and Ben-David (2014, Ch. 26) for related discussion.

64.35 Consider a collection of N feature vectors $\{h_1,\ldots,h_N\}$ where each $h_n\in\mathbb{R}^M$. Introduce two sets $\mathcal{A},\mathcal{B}\subset\mathbb{R}^N$ consisting of $N-$dimensional vectors each defined as follows:

$$\mathcal{A}=\Big\{a=\mathrm{col}\{a_n\}\in\mathbb{R}^N\,|\,a_n=h_n^{\mathsf{T}}w,\ \ \|w\|_2\leq 1\Big\}$$
$$\mathcal{B}=\Big\{b=\mathrm{col}\{b_n\}\in\mathbb{R}^N\,|\,b_n=h_n^{\mathsf{T}}w,\ \ \|w\|_1\leq 1\Big\}$$

where the only difference is the bound on the parameter w: in the first case, we bound its Euclidean norm and in the second case we bound its ℓ_1-norm. Show that the Rademacher complexities of these two sets satisfy

$$\mathcal{R}_N(\mathcal{A})\leq\frac{1}{\sqrt{N}}\times\Big\{\max_{1\leq n\leq N}\|h_n\|_2\Big\}$$
$$\mathcal{R}_N(\mathcal{B})\leq\sqrt{\frac{2\ln(2M)}{N}}\times\Big\{\max_{1\leq n\leq N}\|h_n\|_\infty\Big\}$$

Remark. See Shalev-Shwartz and Ben-David (2014, Ch. 26) and Mohri, Rostamizadeh, and Talwalkar (2018, Ch. 10) for a related discussion.

64.36 Derive the two-sided generalization bounds (64.197a)–(64.197b) by extending the argument used to derive their one-sided counterparts in Appendix 64.D.

64.37 Refer to the binary classification context described in Example 64.9. Verify that the empirical risk admits the representation

$$R_{\mathrm{emp}}(c)=\frac{1}{2}\left\{1-\frac{1}{N}\sum_{n=1}^{N}\gamma(n)c(h_n)\right\}$$

Conclude that one can alternatively select an optimal classifier by solving

$$c^o \;=\; \operatorname*{argsup}_{c\in\mathcal{C}} \left\{ \frac{1}{N}\sum_{n=1}^{N} \gamma(n)c(h_n) \right\}$$

How does this formulation relate to the Rademacher complexity of the class of binary classifiers $\mathcal{C}$?

64.38 Refer to definitions (64.162) and (64.163) for the Rademacher complexity and its empirical version. The Gaussian complexity of a set of functions $Q \in \mathcal{Q}$ is defined similarly with each variable $\boldsymbol{\sigma}_n$ now selected from the standard Gaussian distribution:

$$\widehat{\mathcal{G}}_N(\mathcal{Q}) = \mathbb{E}_{\boldsymbol{\sigma}} \left\{ \sup_{Q\in\mathcal{Q}} \left(\frac{1}{N}\sum_{n=1}^{N} \boldsymbol{\sigma}_n Q(y_n) \right) \right\}, \;\; \boldsymbol{\sigma}_n \sim \mathcal{N}_\sigma(0,1)$$

$$\mathcal{G}_N(\mathcal{Q}) = \mathbb{E}_{\boldsymbol{y}} \left\{ \widehat{\mathcal{G}}_N(\mathcal{Q}) \right\}$$

Show that the Rademacher and Gaussian complexities are related as follows:

$$\alpha \mathcal{R}_N(\mathcal{Q}) \le \mathcal{G}_N(\mathcal{Q}) \le \beta \ln N \, \mathcal{R}_N(\mathcal{Q})$$

for some nonnegative constants α and β. *Remark.* See Tomczak-Jaegermann (1989) for a related discussion.

64.39 Consider a collection of vectors $a \in \mathcal{A} \subset \mathbb{R}^N$, with individual entries $a = \mathrm{col}\{a_n\}$. Consider also Rademacher variables $\{\boldsymbol{\sigma}_n\}$, which take values $\{\pm 1\}$ with equal probability. Establish the Khintchine-Kahane inequality:

$$\frac{1}{2}\mathbb{E}_{\boldsymbol{\sigma}} \Big\| \sum_{n=1}^{N} \boldsymbol{\sigma}_n a_n \Big\|^2 \le \left\{ \mathbb{E}_{\boldsymbol{\sigma}} \Big\| \sum_{n=1}^{N} \boldsymbol{\sigma}_n a_n \| \right\}^2 \;\le\; \mathbb{E}_{\boldsymbol{\sigma}} \Big\| \sum_{n=1}^{N} \boldsymbol{\sigma}_n a_n \Big\|^2$$

Remark. The inequality is originally due to Khintchine (1923) and was extended by Kahane (1964). Proofs and discussion appear in Latala and Oleszkiewicz (1994), Wolff (2003), and Mohri, Rostamizadeh, and Talwalkar (2018).

64.40 Consider a collection of N feature vectors $\{h_1, h_2, \ldots, h_N\}$ from the set $\{h \in \mathbb{R}^M \,|\, K(h,h) \le r^2\}$, where $K(h_a, h_b)$ denotes the kernel function. Let $\phi(h)$ represent the mapping that is implicitly defined by the choice of kernel: it maps vectors from the original feature space $h \in \mathbb{R}^M$ to a transformed space $h^\phi \in \mathbb{R}^{M_\phi}$. Introduce the set:

$$\mathcal{A} = \Big\{ a = \mathrm{col}\{a_n\} \in \mathbb{R}^N \,|\, a_n = (h_n^\phi)^{\mathsf{T}} w^\phi, \;\; \|w^\phi\| \le 1 \Big\}$$

Extend the derivation from Prob. 64.35 to show that the Rademacher complexity of this set satisfies

$$\mathcal{R}_N(\mathcal{A}) \le r/\sqrt{N}$$

Remark. See Mohri, Rostamizadeh, and Talwalkar (2018, Ch. 5) for a related discussion.

64.A VC DIMENSION FOR LINEAR CLASSIFIERS

We establish in this appendix the result of Lemma 64.1 following an argument similar to Abu-Mostafa, Magdon-Ismail, and Lin (2012), Shalev-Shwartz and Ben-David (2014, Ch. 9), and Mohri, Rostamizadeh, and Talwalkar (2018, Ch. 3). Thus, consider the class of affine classifiers defined by $c(h) = \mathrm{sign}(h^{\mathsf{T}} w - \theta)$, with parameters $w \in \mathbb{R}^M$

and $\theta \in \mathbb{R}$. If we assume the feature vectors are extended with a top unit entry, and the weight vector w is extended with $-\theta$ as leading entry, namely,

$$w \leftarrow \begin{bmatrix} -\theta \\ w \end{bmatrix}, \quad h \leftarrow \begin{bmatrix} 1 \\ h \end{bmatrix} \tag{64.65}$$

then it is sufficient to focus on linear classifiers of the form $c(h) = \text{sign}(h^{\mathsf{T}} w)$.

Assuming this extension, let us first establish that VC $\geq M + 1$. We can do so by constructing a collection of $M + 1$ features vectors that can be shattered by linear classifiers. Indeed, consider the following $M + 1$ feature vectors collected as rows into the matrix H below:

$$H \triangleq \begin{bmatrix} h_1^{\mathsf{T}} \\ h_2^{\mathsf{T}} \\ \vdots \\ h_{M+1}^{\mathsf{T}} \end{bmatrix} = \begin{bmatrix} 1 & 0_M^{\mathsf{T}} \\ 1 & e_1^{\mathsf{T}} \\ \vdots & \vdots \\ 1 & e_M^{\mathsf{T}} \end{bmatrix} \in \mathbb{R}^{(M+1)\times(M+1)} \tag{64.66}$$

Each feature vector starts with the unit entry, with the remaining corresponding to the zero vector for h_1 and to the basis vectors $\{e_m\}$ in $\mathbb{R}^M$ for the remaining features. It is easy to verify that the square matrix H is invertible. Now, let $\gamma_{\text{vec}} \in \mathbb{R}^{M+1}$ denote *any* label vector of size $M+1$: the individual entries of γ_{vec} can be $+1$ or -1 at will, so that all labeling possibilities for the $M + 1$ feature vectors in H are covered. Now, for any choice of γ_{vec}, there exists a classifier w that maps H to γ_{vec} and it can be chosen as $w = H^{-1}\gamma_{\text{vec}}$. Therefore, the above set of $M + 1$ feature vectors can be shattered and we conclude that

$$\text{VC(linear classifiers)} \geq M + 1 \tag{64.67}$$

Let us verify next that VC $\leq M+1$ so that equality must hold. To prove this second statement, it is sufficient to exhibit an example with $M + 2$ feature vectors and the corresponding labels for which no linear classifier exists. Thus, consider a collection of $M + 2$ nonzero feature vectors in $\mathbb{R}^{M+1}$. These vectors are clearly linearly dependent, which means there exists some feature vector among them, denoted by h_n, such that h_n is a linear combination of the remaining feature vectors. Specifically, we write

$$h_n = \sum_{m \neq n}^{M+1} \alpha(m) h_m \tag{64.68}$$

for some coefficients $\{\alpha(m)\}$; some of which are nonzero. We now assign the following labels to the $M + 2$ feature vectors $\{h_1, h_2, \ldots, h_{M+2}\}$:

$$\gamma(m) = \begin{cases} \text{sign}(\alpha(m)), & \text{for all } m \neq n \\ -1, & \text{for } m = n \end{cases} \tag{64.69}$$

That is, we label h_n as -1 and label all other feature vectors by the sign of the corresponding coefficient $\alpha(m)$; if $\alpha(m) = 0$, it does not matter whether we label the corresponding feature vector with $+1$ or -1. Now, note the following. For *any* classifier w that is able to classify the $M + 1$ features $\{h_m,\, m \neq n\}$ so that

$$\text{sign}(h_m^{\mathsf{T}} w) = \gamma(m) = \text{sign}(\alpha(m)) \tag{64.70}$$

this classifier will not be able to classify h_n correctly because

$$\text{sign}(h_n^{\mathsf{T}} w) = \text{sign}\Big(\sum_{m \neq n} \alpha(m) h_m^{\mathsf{T}} w\Big) \;>\; 0 \tag{64.71}$$

The positive sign contradicts the fact that the label for h_n is negative. Therefore, we

have a collection of $M+2$ feature vectors that cannot be separated by the linear classifier and we conclude that

$$\text{VC(linear classifiers)} \leq M+1 \tag{64.72}$$

Combining (64.67) and (64.72) we arrive at the desired conclusion.

64.B SAUER LEMMA

In this appendix, we establish a useful lemma that deals with a fundamental combinatorial bound and use it to establish the Vapnik-Chervonenkis inequality (64.44) in Appendix 64.C. The arguments in these two appendices are adapted from the derivation given by Devroye, Gyorfi, and Lugosi (1996) adjusted to our notation and conventions. Thus, let $\{h_n \in \mathbb{R}^M\}$ denote N feature vectors and let $\mathcal{C}$ denote a set of classifier models; this set may have a finite or infinite number of elements. Each $c \in \mathcal{C}$ maps a feature vector h_n into one of two binary classes, i.e., $c(h_n) : \mathbb{R}^M \to \{\pm 1\}$.

Shatter coefficient or growth function

There are 2^N possibilities for assigning the N feature vectors to the two classes ± 1. For each choice of a classifier $c \in \mathcal{C}$, we obtain one possible labeling vector (also called a *dichotomy*), denoted by ℓ_c, for the given feature vectors:

$$\ell_c \triangleq \text{col}\Big\{c(h_0),\, c(h_1),\, c(h_2), \ldots, c(h_{N-1})\Big\} \in \{\pm 1\}^N \tag{64.73}$$

This is a vector of size $N \times 1$ with entries ± 1.

Example 64.4 (**Illustrating dichotomies**) Figure 64.12 illustrates the construction. In this example, we assume the feature data are scalars, $h_n \in \mathbb{R}$, and that each classifier c in the set $\mathcal{C}$ is defined by some threshold parameter $t \in \mathbb{R}$. Based on the value of t, the classifier c assigns a feature vector to class $+1$ or -1 according to the following decision rule:

$$\begin{cases} \text{if } h_n \geq t, \text{ then } c(h_n) = +1 \\ \text{if } h_n < t, \text{ then } c(h_n) = -1 \end{cases} \tag{64.74}$$

The first row on the left-hand side of the figure shows three feature values, denoted by $\{h_0, h_1, h_2\}$; they occur at coordinate locations $\{0, 2, 3\}$ on the real axis. The subsequent rows in the figure indicate the classes that these feature entries are assigned to depending on where the threshold value t (denoted by the red circle) is located. In particular,

$$\text{if } t \leq 0, \text{ then } \{h_0, h_1, h_2\} \in \{+1, +1, +1\} \tag{64.75}$$

$$\text{if } 0 < t \leq 2, \text{ then } \{h_0, h_1, h_2\} \in \{-1, +1, +1\} \tag{64.76}$$

$$\text{if } 2 < t \leq 3, \text{ then } \{h_0, h_1, h_2\} \in \{-1, -1, +1\} \tag{64.77}$$

$$\text{if } t > 3, \text{ then } \{h_0, h_1, h_2\} \in \{-1, -1, -1\} \tag{64.78}$$

Therefore, in this example, the classifier set $\mathcal{C}$ is only able to generate four possible labeling vectors, $\{\ell_c\}$, which we collect into the rows of an assignment matrix $\mathcal{A}_{\mathcal{C}}$:

$$\mathcal{A}_{\mathcal{C}} = \left[\begin{array}{ccc} h_0 & h_1 & h_2 \\ \hline +1 & +1 & +1 \\ -1 & +1 & +1 \\ -1 & -1 & +1 \\ -1 & -1 & -1 \end{array}\right] \tag{64.79}$$

There are clearly assignments that are not possible to generate by this set of threshold classifiers, such as the assignment $\{+1, -1, +1\}$. Observe that even though the classifier set $\mathcal{C}$ may have an infinite number of models, the number of dichotomies in $\mathcal{A}_{\mathcal{C}}$ (i.e., the number of its rows) is always finite and bounded by 2^N.

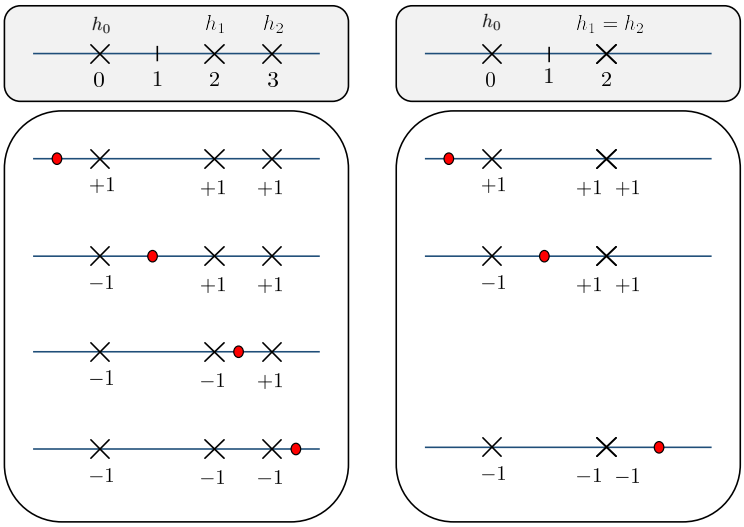

Figure 64.12 The rows on the left show three feature values on the real line and the four possible class assignments for them. The red circle represents the location of the threshold value in each case. The rows on the right show the same construction for the case in which two feature values coincide, $h_1 = h_2$. In this second case, only three assignments are possible.

As was already explained in Sec. 64.4, we say that the set of classifiers $\mathcal{C}$ is able to *shatter* the N feature vectors if every possible assignment among the 2^N possibilities can be generated by $\mathcal{C}$. We illustrated this definition in Fig. 64.7. We observe from the above example, with three scalar feature values, that it is not always possible to generate all 2^N valid assignments (or labeling vectors, ℓ_c) by the classifiers in $\mathcal{C}$. We let $\mathcal{A}_{\mathcal{C}}$ denote the collection of all assignments that can be generated by $\mathcal{C}$:

$$\mathcal{A}_{\mathcal{C}}\,(h_0, h_1, \ldots, h_{N-1}) \;\triangleq\; \{\ell_c,\; c \in \mathcal{C}\} \tag{64.80}$$

so that each choice $c \in \mathcal{C}$ generates one assignment vector, ℓ_c, and the aggregation of all these row vectors is the matrix $\mathcal{A}_{\mathcal{C}}$. Observe that the assignment set $\mathcal{A}_{\mathcal{C}}$ is a function of *both* the classifier space, $\mathcal{C}$, and the feature vectors, $\{h_n\}$. A different collection of feature vectors $\{h_n\}$ would generally lead to a different assignment set $\mathcal{A}_{\mathcal{C}}$. For instance, as shown in the second column of the same Fig. 64.12, if two of the feature values happen to occur at the same location, say, $h_0 = 0$ while $h_1 = h_2 = 2$, then, in this case, the threshold classifier set can only generate three possible labeling vectors, namely,

$$\mathcal{A}_{\mathcal{C}} = \left[\begin{array}{ccc} h_o & h_1 & h_2 \\ \hline +1 & +1 & +1 \\ -1 & +1 & +1 \\ -1 & -1 & -1 \end{array}\right] \tag{64.81}$$

To remove ambiguity due to the choice of the feature data, we introduce the *shatter coefficient* of the classifier set, $\mathcal{C}$, and denote it by $\mathcal{S}(\mathcal{C}, N)$. This coefficient, which is also called the *growth function*, is an integer value that corresponds to the largest number of assignments that can be generated by $\mathcal{C}$ over all possible choices for the feature vectors $\{h_n\}$, i.e.,

$$\mathcal{S}(\mathcal{C}, N) \triangleq \max_{\{h_n\}} |\mathcal{A}_{\mathcal{C}}(h_o, h_1, \ldots, h_{N-1})| \tag{64.82}$$

where the notation $|\mathcal{A}_{\mathcal{C}}|$ denotes the cardinality of the set $\mathcal{A}_{\mathcal{C}}$; in this case, it is the number of rows in the assignment matrix. Thus, the shatter coefficient $\mathcal{S}(\mathcal{C}, N)$ corresponds to the largest possible cardinality for $\mathcal{A}_{\mathcal{C}}$. For the example of Fig. 64.12, it is clear that $\mathcal{S}(\mathcal{C}, 3) = 4$. For this same example, if we instead had a total of N features (rather than only 3), then it is easy to see that the shatter coefficient will be $\mathcal{S}(\mathcal{C}, N) = N + 1$. Obviously, for any set of classifiers and feature vectors, it holds that

$$\mathcal{S}(\mathcal{C}, N) \leq 2^N \tag{64.83}$$

since 2^N is the maximum number of possible assignments for binary classification scenarios. Observe that this bound grows exponentially with the size of the training data. One fundamental result, derived further ahead under the designation of *Sauer lemma*, is that for classifier sets with *finite* VC dimension, their shatter coefficients are bounded by polynomial (rather than exponential) functions of N — see (64.87) and (64.88).

VC dimension

We defined in Sec. 64.4 the VC dimension of a class of classifiers $\mathcal{C}$ as the largest integer K for which at least one set of K feature vectors can shattered by $\mathcal{C}$. In other words, the VC dimension of $\mathcal{C}$ is the largest K for which $\mathcal{S}(\mathcal{C}, K) = 2^K$ or, equivalently,

$$\mathcal{S}(\mathcal{C}, \text{VC}) = 2^{\text{VC}} \tag{64.84}$$

It turns out that when $\text{VC} < \infty$, the growth function (or shatter coefficient) of $\mathcal{C}$ grows polynomially in N. This property is established in the following statement, where we employ the following definition for the combinatorial function:

$$\binom{N}{n} \triangleq \begin{cases} \dfrac{N!}{n!(N-n)!}, & 0 \leq n \leq N \\ 0, & \text{otherwise} \end{cases} \tag{64.85}$$

Sauer lemma (Sauer (1972), Shelah (1972)). *The shatter coefficient (or growth function) of a set of classifiers $\mathcal{C}$ applied to N feature vectors is bounded by the following value in terms of the VC dimension:*

$$\mathcal{S}(\mathcal{C}, N) \leq \sum_{n=0}^{\text{VC}} \binom{N}{n} \tag{64.86}$$

Two other useful bounds that follow from (64.86) when $1 \leq \text{VC} \leq N$ are:

$$\mathcal{S}(\mathcal{C}, N) \leq (1 + N)^{\text{VC}} \tag{64.87}$$

$$\mathcal{S}(\mathcal{C}, N) \leq \left(\frac{Ne}{\text{VC}}\right)^{\text{VC}} \tag{64.88}$$

where the letter "e" refers to the basis of the natural logarithm ($e \approx 2.7183$).

Proof: The argument is lengthy and involves several steps. We employ a traditional inductive argument. Let us first verify that the lemma holds for a couple of useful boundary conditions.

(Boundary conditions). For $N = 0$ and any VC, we have

$$\sum_{n=0}^{\text{VC}} \binom{0}{n} \;=\; 1, \;\; \text{and} \;\; \mathcal{S}(\mathcal{C}, 0) \leq 1 \tag{64.89}$$

where the second equality is because there are no feature data to label (therefore, we can bound the number of label possibilities by one). Likewise, for $\text{VC} = 0$ and any $N \geq 1$, we have

$$\sum_{n=0}^{0} \binom{N}{n} \;=\; 1, \;\; \text{and} \;\; \mathcal{S}(\mathcal{C}, N) = 1 \tag{64.90}$$

where the second equality is because the VC dimension is zero and, therefore, the set of classifiers can only assign the same label to all feature vectors. Similarly, for $N = 1$ and any $\text{VC} \geq 1$, we have

$$\sum_{n=0}^{\text{VC}} \binom{1}{n} \;=\; \binom{1}{0} + \binom{1}{1} + \ldots + \binom{1}{\text{VC}} \;=\; 2 \tag{64.91}$$

while $\mathcal{S}(\mathcal{C}, 1) \leq 2$. This latter inequality is because, at best, the set of classifiers may be able to assign the single feature vector into either class. We therefore assume $\text{VC} \geq 1$.

(Induction argument). We now assume that (64.86) holds up to $N - 1$ and show that it also holds for N. To do so, and in order to simplify the notation, we introduce the shorthand symbol $\mathcal{H}_N$ to refer to the collection of N feature vectors, say,

$$\mathcal{H}_N \;\triangleq\; \{h_0, \, h_1, \, \ldots, h_{N-1}\} \tag{64.92}$$

Let $\mathcal{S}(\mathcal{C}, N)$ denote the shatter coefficient for the set $\mathcal{C}$ over N feature vectors. We already know that this value is the maximal number of different ways by which the N vectors can be labeled. Let $\mathcal{C}_s \subset \mathcal{C}$ denote the smallest subset of the classifier set that attains this shatter value. That is, the number of classifiers in $\mathcal{C}_s$ is equal to the number of distinct labeling/dichotomies that can be generated on $\mathcal{H}_N$. Likewise, we write $\mathcal{H}_{N-1}$ to refer to the collection formed by excluding the last feature vector:

$$\mathcal{H}_N \;\triangleq\; \mathcal{H}_{N-1} \;\cup\; \{h_{N-1}\} \tag{64.93}$$

We also let $\mathcal{S}(\mathcal{C}, N-1)$ denote the shatter coefficient for the same set $\mathcal{C}$ over $N-1$ feature vectors. This value is the maximal number of different ways by which $N-1$ vectors can be labeled. We further let $\mathcal{C}_1 \subset \mathcal{C}_s$ denote the smallest subset of the classifier set that attains this shatter value. Again, the number of classifiers in $\mathcal{C}_1$ is equal to the number of distinct labeling/dichotomies that can be generated on $\mathcal{H}_{N-1}$. Moreover, since $\mathcal{C}_1 \subset \mathcal{C}_s$, it holds that

$$\text{VC}(\mathcal{C}_1) \leq \text{VC}(\mathcal{C}_s) \leq \text{VC}(\mathcal{C}) \tag{64.94}$$

This is because any set of feature vectors that can be shattered by $\mathcal{C}_1$ can also be shattered by $\mathcal{C}_s$. We subsequently decompose the set $\mathcal{C}_s$ into

$$\mathcal{C}_s = \mathcal{C}_1 \;\cup\; (\mathcal{C} \backslash \mathcal{C}_1) \;\;\triangleq\;\; \mathcal{C}_1 \;\cup\; \mathcal{C}_1^c \tag{64.95}$$

It is clear that each classifier in the complementary set $\mathcal{C}_1^c$ generates a labeling for the feature vectors in $\mathcal{H}_{N-1}$ that is already generated by some classifier in $\mathcal{C}_1$; otherwise,

this classifier from $\mathcal{C}_1^c$ would need to be in $\mathcal{C}_1$. This also means that for every classifier in $\mathcal{C}_1^c$ there exists a classifier in $\mathcal{C}_1$ such that both classifiers agree on their labeling of $\mathcal{H}_{N-1}$ but disagree on their labeling of h_{N-1}; they need to disagree on h_{N-1} otherwise they will be identical classifiers.

Another property for the set $\mathcal{C}_1^c$ is the following. Assume two classifiers, say c_1 and c_2, exist in the set $\mathcal{C}_s$ that classify the $N-1$ feature vectors in $\mathcal{H}_{N-1}$ in the same manner. If this happens, then only one of these classifiers, say, c_1, must belong to the set $\mathcal{C}_1$ because otherwise $\mathcal{C}_1$ would not be the smallest classifier set that attains the shatter value for $\mathcal{H}_{N-1}$. The other classifier, say, c_2, will be in $\mathcal{C}_1^c$. Moreover, and importantly, this second classifier will label h_{N-1} differently from the classifier c_1 added to $\mathcal{C}_1$ (otherwise, both classifiers c_1 and c_2 would be identical).

The above properties are illustrated in the assignment matrix shown below for the threshold-based classifier of Fig. 64.12 for the case $N = 4$:

$$\mathcal{A}_{\mathcal{C}} = \left[\begin{array}{cccc|c} h_0 & h_1 & h_2 & h_3 & \\ \hline +1 & +1 & +1 & +1 & \\ -1 & +1 & +1 & +1 & \mathcal{C}_1 \\ -1 & -1 & +1 & +1 & \\ -1 & -1 & -1 & +1 & \\ \hline -1 & -1 & -1 & -1 & \mathcal{C}_1^c \end{array}\right] \tag{64.96}$$

In this case, the shatter coefficient is $\mathcal{S}(\mathcal{C}, 4) = 5$, so that there are at most five dichotomies that can be generated by $\mathcal{C}$. These dichotomies are listed as the rows of $\mathcal{A}_{\mathcal{C}}$ shown above. These rows represent the smallest classifier set, denoted by $\mathcal{C}_s$. Observe that the first four rows correspond to the classifiers in the set $\mathcal{C}_1$: they attain the maximal shatter value of $\mathcal{S}(\mathcal{C}, 3) = 4$ on the first three feature vectors. Observe further that the last row in $\mathcal{A}_{\mathcal{C}}$ represents the classifier set $\mathcal{C}_1^c$; it consists of a single classifier that generates the same labels on the features $\{h_0, h_1, h_2\}$ as the fourth classifier, but nevertheless leads to a different label for h_3.

We now verify that more generally, and in view of the above observations regarding the sets $\{\mathcal{C}_1, \mathcal{C}_2^c, \mathcal{C}_s\}$, it should hold that:

$$\mathrm{VC}(\mathcal{C}_1^c) \leq \mathrm{VC}(\mathcal{C}_s) - 1 \leq \mathrm{VC}(\mathcal{C}) - 1 \tag{64.97}$$

Indeed, assume $\mathcal{C}_1^c$ shatters completely some set of feature vectors $\mathcal{H}' \subset \mathcal{H}_{N-1}$. Then, it necessarily holds that $\mathcal{C}_s$ should shatter the expanded collection $\mathcal{H}' \cup \{h_{N-1}\}$. It is obvious that $\mathcal{C}_s$ shatters $\mathcal{H}'$ since $\mathcal{C}_1^c \subset \mathcal{C}_s$. With regards to h_{N-1}, we simply observe that $\mathcal{C}_s = \mathcal{C}_1 \cup \mathcal{C}_1^c$ and each of these sets contains a classifier that labels h_{N-1} differently than the other (e.g., the classifiers c_1 and c_2 mentioned before).

Now note that, by the induction assumption,

$$\mathcal{S}(\mathcal{C}_1, N-1) \leq \sum_{n=0}^{\mathrm{VC}} \binom{N-1}{n} \tag{64.98}$$

$$\mathcal{S}(\mathcal{C}_1^c, N-1) \leq \sum_{n=0}^{\mathrm{VC}-1} \binom{N-1}{n} \tag{64.99}$$

Moreover, it holds that

$$\begin{aligned}\mathcal{S}(\mathcal{C},N) &\leq \mathcal{S}(\mathcal{C}_1,N-1) \;+\; \mathcal{S}(\mathcal{C}_1^c,N-1)\\ &\leq \sum_{n=0}^{\mathrm{VC}}\binom{N-1}{n} \;+\; \sum_{n=0}^{\mathrm{VC}-1}\binom{N-1}{n}\\ &= \sum_{n=0}^{\mathrm{VC}}\binom{N-1}{n} \;+\; \sum_{n=0}^{\mathrm{VC}}\binom{N-1}{n-1}\\ &= \sum_{n=0}^{\mathrm{VC}}\binom{N}{n} \end{aligned} \tag{64.100}$$

where in the last equality we used the property

$$\binom{N}{n} \;=\; \binom{N-1}{n} \;+\; \binom{N-1}{n-1} \tag{64.101}$$

The bound (64.100) establishes result (64.86).

Now, assume $1 \leq \mathrm{VC} \leq N$. With regards to the bound (64.88), we note that since $\mathrm{VC}/N \leq 1$:

$$\begin{aligned}\left(\frac{\mathrm{VC}}{N}\right)^{\mathrm{VC}}\left\{\sum_{n=0}^{\mathrm{VC}}\binom{N}{n}\right\} &\leq \sum_{n=0}^{\mathrm{VC}}\binom{N}{n}\left(\frac{\mathrm{VC}}{N}\right)^n\\ &\overset{(a)}{\leq} \sum_{n=0}^{N}\binom{N}{n}\left(\frac{\mathrm{VC}}{N}\right)^n\\ &= \sum_{n=0}^{N}\binom{N}{n}\left(\frac{\mathrm{VC}}{N}\right)^n 1^{N-n}\\ &\overset{(b)}{=} \left(1+\frac{\mathrm{VC}}{N}\right)^N\\ &\overset{(c)}{\leq} e^{\mathrm{VC}}\end{aligned} \tag{64.102}$$

where in step (a) we replaced the upper limit on the sum by N, in step (b) we used the binomial theorem, namely,

$$(x+y)^m \;=\; \sum_{\ell=0}^{m}\binom{m}{\ell}x^\ell y^{m-\ell} \tag{64.103}$$

and in step (c) we used the fact that, for any $x \geq 0$:

$$e^x \;\leq\; \left(1+\frac{x}{N}\right)^N \tag{64.104}$$

Using (64.102) in (64.86) gives (64.88).

With regards to bound (64.87), we first note that, for any integer $n \geq 0$, it holds that

$$\binom{N}{n} \;\triangleq\; \frac{N!}{n!(N-n)!} \;\leq\; \frac{N^n}{n!} \tag{64.105}$$

Consequently, since $\mathrm{VC} \geq 1$ and $0 \leq n \leq \mathrm{VC}$,

$$\binom{N}{n} \;\leq\; \frac{N^n}{n!}\frac{\mathrm{VC}!}{(\mathrm{VC}-n)!} \;=\; \binom{\mathrm{VC}}{n}N^n \tag{64.106}$$

Using this result in (64.100) gives

$$\begin{aligned}\mathcal{S}(\mathcal{C},N) \le \sum_{n=0}^{\mathrm{VC}}\binom{N}{n} &\le \sum_{n=0}^{\mathrm{VC}}\binom{\mathrm{VC}}{n}N^n \times 1^{(\mathrm{VC}-n)} \\ &\overset{(a)}{=} (1+N)^{\mathrm{VC}}\end{aligned} \tag{64.107}$$

where in step (a) we applied the binomial theorem (64.103) again.

■

64.C VAPNIK-CHERVONENKIS BOUND

In this appendix, we establish the validity of the Vapnik-Chervonenkis bound (64.44) for binary classification problems with classes $\boldsymbol{\gamma} \in \{\pm 1\}$. The argument is adapted from the derivation given by Devroye, Gyorfi, and Lugosi (1996) adjusted to our notation and conventions. Let $\{\gamma(n), h_n \in \mathbb{R}^M\}$ denote N independent realizations arising from a joint (unknown) distribution $f_{\boldsymbol{\gamma},\boldsymbol{h}}(\gamma, h)$. Let $c^\star(h)$ denote a solution to the empirical risk minimization problem over some set $c \in \mathcal{C}$:

$$c^\star(h) \triangleq \operatorname*{argmin}_{c\in\mathcal{C}} \left\{ R_{\mathrm{emp}}(c) \triangleq \frac{1}{N}\sum_{n=0}^{N-1}\mathbb{I}\Big[c(h_n)\neq\gamma(n)\Big] \right\} \tag{64.108}$$

Likewise, let $c^o(h)$ denote the optimal solution that minimizes the probability of misclassification over the same set:

$$c^o(h) \triangleq \operatorname*{argmin}_{c\in\mathcal{C}} \left\{ R(c) \triangleq \mathbb{P}\left[c(\boldsymbol{h})\neq\boldsymbol{\gamma}\right] \right\} \tag{64.109}$$

Classifier set with finite cardinality

Assume initially that the set $\mathcal{C}$ has finite cardinality (i.e., a finite number of elements), denoted by $|\mathcal{C}|$. Using straightforward arguments, and Hoeffding inequality (3.233), we are able to establish in Probs. 64.4 and 64.24 the following useful bound:

$$\mathbb{P}\left(\sup_{c\in\mathcal{C}}|R_{\mathrm{emp}}(c)-R(c)| \ge \delta\right) \le 2|\mathcal{C}|e^{-2N\delta^2}, \quad \text{(when } |\mathcal{C}| \text{ is finite)} \tag{64.110}$$

The difficulty arises when the set $\mathcal{C}$ has uncountably infinite elements. In that case, the term on the right-hand side of (64.110) is not useful because it degenerates to an unbounded value. It turns out though that what matters is not the cardinality of $\mathcal{C}$, but rather the largest number of dichotomies that the set $\mathcal{C}$ can generate on the training data. This number is equal to the shatter coefficient of $\mathcal{C}$, which we introduced in the previous appendix and denoted it by $\mathcal{S}(\mathcal{C},N)$. We showed in (64.86), and also (64.87)–(64.88), that the shatter coefficient is bounded polynomially in N even when $|\mathcal{C}|$ is infinite.

Derivation of VC bound

We now establish the following fundamental result; the proof of which is non-trivial and relies again on several steps. We follow largely the presentation given by Devroye, Gyorfi, and Lugosi (1996, Ch. 12). As indicated in the concluding remarks, the coefficient appearing in the exponential factor in (64.111) below ends up being $N\delta^2/32$, while the coefficient appearing in the original bound given by Vapnik and Chervonenkis (1971) is $N\delta^2/8$ and corresponds to a tighter bound. This difference is not significant

since it is sufficient for our purposes to know that a bound exists and that this bound decays to zero as $N \to \infty$ at a uniform rate that is independent of the data distribution.

Vapnik-Chervonenkis inequality (Vapnik and Chervonenkis (1971)). *For any given small constant $\delta > 0$ and $N\delta^2 \geq 2$, it holds that*

$$\mathbb{P}\left(\sup_{c\in\mathcal{C}} |R_{\text{emp}}(c) - R(c)| > \delta\right) \;\leq\; 8\left(\frac{Ne}{\text{VC}}\right)^{\text{VC}} e^{-N\delta^2/32} \tag{64.111}$$

independent of the data distribution, $f_{\boldsymbol{\gamma},\boldsymbol{h}}(\gamma, h)$, and in terms of the VC dimension of the classifier set $\mathcal{C}$.

Proof: The argument is demanding and involves several steps. We remark that the condition $N\delta^2 \geq 2$ in the statement of the inequality is not a restriction. This is because for $N\delta^2 < 2$, the bound in (64.111) becomes trivial because it will be larger than $7.5(Ne/\text{VC})$, which in turn is generally larger than one (especially since we often have $\text{VC} \leq N$).

(Symmetrization step — adding fictitious samples). The first step in the argument involves replacing the difference $R_{\text{emp}}(c) - R(c)$, which involves the unknown $R(c)$, by one that involves only empirical risks — see (64.113) below. By doing so, we will be able to bound the probability expression that appears in (64.111) by a term that depends symmetrically and solely on empirical data.

To achieve this task, we start by introducing a collection of N *fictitious* data samples, denoted by $\{\gamma'(n), h'_n\}$, and which are assumed to arise from the *same* data distribution as the original samples, $\{\gamma(n), h_n\}$. These fictitious samples are added merely for the sake of argument and will not affect the final result. For any classifier $c \in \mathcal{C}$, we denote its empirical risk on the fictitious data by using the prime notation:

$$R'_{\text{emp}}(c) \;\triangleq\; \frac{1}{N}\sum_{n=0}^{N-1} \mathbb{I}\left[c(h'_n) \neq \gamma'(n)\right] \tag{64.112}$$

We now verify that

$$\mathbb{P}\left(\sup_{c\in\mathcal{C}} |R_{\text{emp}}(c) - R(c)| > \delta\right) \;\leq\; 2\,\mathbb{P}\Big(\sup_{c\in\mathcal{C}} |R_{\text{emp}}(c) - R'_{\text{emp}}(c)| > \delta/2\Big) \tag{64.113}$$

where, as desired, the term on the right-hand side involves only empirical risks in a symmetrical manner. Once established, this result relates the distance between two empirical risks to the desired distance from the empirical risk to the optimal risk — see Fig. 64.13.

Let $\bar{c} \in \mathcal{C}$ be an element in the classifier set that satisfies the bound

$$|R_{\text{emp}}(\bar{c}) - R(\bar{c})| > \delta \tag{64.114}$$

If such an element does not exist, we simply let $\bar{c}$ be an arbitrary element from $\mathcal{C}$. This classifier therefore satisfies:

$$\mathbb{P}\Big(|R_{\text{emp}}(\bar{c}) - R(\bar{c})| > \delta\Big) \geq \mathbb{P}\left(\sup_{c\in\mathcal{C}} |R_{\text{emp}}(c) - R(c)| > \delta\right) \tag{64.115}$$

This is because if a classifier $\bar{c} \in \mathcal{C}$ exists satisfying (64.114), then both probabilities in the above relation are equal to one and the inequality holds. If, on the other hand, such a $\bar{c}$ does not exist, then the probabilities will be zero and the inequality again holds.

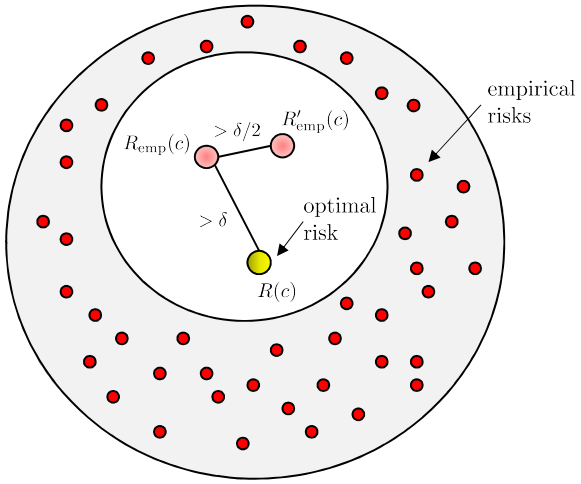

Figure 64.13 The red circles represent empirical risk values for different realizations of the data. Expression (64.113) relates the probability of the distance between $R_{\text{emp}}(c)$ and $R(c)$ being larger than δ to the probability of the distance between two empirical risks being larger than $\delta/2$. The subsequent analysis will bound this latter difference, which is a useful step since $R(c)$ is unknown.

To arrive at (64.113), we first note the following sequence of inequalities:

$$\begin{aligned}
&\mathbb{P}\Big(\sup_{c\in\mathcal{C}} |R_{\text{emp}}(c) - R'_{\text{emp}}(c)| > \delta/2\Big) \\
&\geq \mathbb{P}\left(|R_{\text{emp}}(\bar{c}) - R'_{\text{emp}}(\bar{c})| > \delta/2\right) \\
&= \mathbb{P}\left(|R_{\text{emp}}(\bar{c}) - R(\bar{c}) + R(\bar{c}) - R'_{\text{emp}}(\bar{c})| > \delta/2\right) \\
&\overset{(a)}{\geq} \mathbb{P}\left(|R_{\text{emp}}(\bar{c}) - R(\bar{c})| > \delta \ \text{ and } \ |R'_{\text{emp}}(\bar{c}) - R(\bar{c})| < \delta/2\right)
\end{aligned} \tag{64.116}$$

where step (a) follows from the property that for any two real numbers:

$$|a-b| \ \geq \ \Big|\,|a| - |b|\,\Big| \tag{64.117}$$

Indeed, assume that the following two conditions hold:

$$\left(|R_{\text{emp}}(\bar{c}) - R(\bar{c})| > \delta\right) \ \text{ and } \ \left(\big|R'_{\text{emp}}(\bar{c}) - R(\bar{c})\big| < \delta/2\right) \tag{64.118}$$

Then, property (64.117) implies that

$$\begin{aligned}
\big|R_{\text{emp}}(c) - R'_{\text{emp}}(c)\big| &= \big|\,(R_{\text{emp}}(\bar{c}) - R(\bar{c})) - \big(R'_{\text{emp}}(\bar{c}) - R(\bar{c})\big)\,\big| \\
&\geq \Big|\,\underbrace{|R_{\text{emp}}(\bar{c}) - R(\bar{c})|}_{>\delta} \ - \ \underbrace{\big|R'_{\text{emp}}(\bar{c}) - R(\bar{c})\big|}_{<\delta/2}\,\Big| \\
&> \delta/2
\end{aligned} \tag{64.119}$$

Consequently, conditions (64.118) combined imply result (64.119), which justifies step

(a). Continuing we have from (64.116) that

$$\begin{aligned}
&\mathbb{P}\Big(\sup_{c\in\mathcal{C}}|R_{\rm emp}(c)-R'_{\rm emp}(c)|>\delta/2\Big) \qquad (64.120)\\
&=\mathbb{E}\Big\{\mathbb{I}[|R_{\rm emp}(\bar{c})-R(\bar{c})|>\delta]\,\mathbb{I}\left[|R'_{\rm emp}(\bar{c})-R(\bar{c})|<\delta/2\right]\Big\}\\
&\overset{(b)}{=}\mathbb{E}\Big(\mathbb{E}\Big\{\mathbb{I}[|R_{\rm emp}(\bar{c})-R(\bar{c})|>\delta]\,\mathbb{I}\left[|R'_{\rm emp}(\bar{c})-R(\bar{c})|<\delta/2\right]\,|\,\{\gamma(n),h_n\}\Big\}\Big)\\
&=\mathbb{E}\Big\{\mathbb{I}[|R_{\rm emp}(\bar{c})-R(\bar{c})|>\delta]\Big\}\,\mathbb{E}\Big\{\mathbb{I}\left[|R'_{\rm emp}(\bar{c})-R(\bar{c})|<\delta/2\right]\,|\,\{\gamma(n),h_n\}\Big\}\\
&=\mathbb{P}\Big(|R_{\rm emp}(\bar{c})-R(\bar{c})|>\delta\Big)\,\mathbb{P}\Big(|R'_{\rm emp}(\bar{c})-R(\bar{c})|<\delta/2\,|\,\{\gamma(n),h_n\}\Big)
\end{aligned}$$

where step (b) introduces conditioning on the original training data $\{\gamma(n),h_n\}$. We now examine the rightmost probability term. For this purpose, we introduce the zero-mean, independent, and identically-distributed random variables

$$\boldsymbol{z}(n)\;\triangleq\;\mathbb{I}\left[\bar{c}(\boldsymbol{h}'_n)\neq\boldsymbol{\gamma}'(n)\right]\;-\;\mathbb{E}\Big(\mathbb{I}\left[\bar{c}(\boldsymbol{h}'_n)\neq\boldsymbol{\gamma}'(n)\right]\,|\,\{\gamma(n),h_n\}\Big) \qquad (64.121)$$

where we will be using the boldface notation for the variables $\{\boldsymbol{\gamma}(n),\boldsymbol{\gamma}'(n),\boldsymbol{h}_n,\boldsymbol{h}'_n\}$ whenever it is necessary to emphasize their stochastic nature; we will use the normal font notation to refer to their realizations. Using the fact that, by definition, $R(\bar{c})=\mathbb{E}\,\mathbb{I}\,[\bar{c}(\boldsymbol{h}')\neq\boldsymbol{\gamma}']$, it is straightforward to verify that the variance of each $\boldsymbol{z}(n)$ is given by

$$\sigma_z^2\;\triangleq\;\mathbb{E}\,(\boldsymbol{z}(n))^2\;=\;R(\bar{c})-R^2(\bar{c}) \qquad (64.122)$$

But since the risk value, $R(\bar{c})$, is a probability measure, it assumes values in the range $R(\bar{c})\in[0,1]$. It can then be verified that the quadratic expression in $R(\bar{c})$ on the right-hand side of (64.122) satisfies:

$$0\leq R(\bar{c})-R^2(\bar{c})\leq 1/4 \qquad (64.123)$$

so that $\sigma_z^2\leq 1/4$. It follows from the definition of the empirical and actual risks that:

$$\begin{aligned}
\mathbb{P}\Big(|R'_{\rm emp}(\bar{c})-R(\bar{c})|<\delta/2\,|\,\{\gamma(n),h_n\}\Big)&=\mathbb{P}\left(\left|\frac{1}{N}\sum_{n=0}^{N-1}\boldsymbol{z}(n)\right|<\delta/2\,|\,\{\gamma(n),h_n\}\right)\\
&=\mathbb{P}\left(\left|\sum_{n=0}^{N-1}\boldsymbol{z}(n)\right|<N\delta/2\,|\,\{\gamma(n),h_n\}\right)\\
&\overset{(a)}{\geq}1-\frac{4}{N^2\delta^2}N\sigma_z^2\\
&=1-\frac{4}{N\delta^2}\sigma_z^2\\
&\overset{(b)}{\geq}1-\frac{4}{N\delta^2}\frac{1}{4}\\
&\geq 1/2 \qquad (64.124)
\end{aligned}$$

where step (a) uses Chebyshev inequality (3.28), step (b) uses $\sigma_z^2\leq 1/4$, and the last

step uses the condition $N\delta^2 \geq 2$. Combining results (64.120) and (64.124) we arrive at

$$\begin{aligned}\mathbb{P}\Big(\sup_{c\in\mathcal{C}}|R_{\text{emp}}(c)-R'_{\text{emp}}(c)|>\delta/2\Big) &\geq \frac{1}{2}\,\mathbb{P}\Big(|R_{\text{emp}}(\bar{c})-R(\bar{c})|>\delta\Big)\\ &\overset{(64.115)}{\geq} \frac{1}{2}\,\mathbb{P}\Big(\sup_{c\in\mathcal{C}}|R_{\text{emp}}(c)-R(c)|>\delta\Big)\end{aligned} \tag{64.125}$$

which leads to the desired result (64.113).

(Symmetrization step — randomizing the signs). Now we work on bounding the right-hand side of (64.113) since it only involves empirical risks. Expressing these empirical risks directly in terms of the corresponding data, we can write (where we are again emphasizing the random nature of the training and fictitious data):

$$\begin{aligned}&\mathbb{P}\left(\sup_{c\in\mathcal{C}}|R_{\text{emp}}(c)-R'_{\text{emp}}(c)|>\delta/2\right)\\ &=\mathbb{P}\left(\sup_{c\in\mathcal{C}}\frac{1}{N}\left|\sum_{n=0}^{N-1}\mathbb{I}\left[c(\boldsymbol{h}_n)\neq\boldsymbol{\gamma}(n)\right]-\mathbb{I}\left[c(\boldsymbol{h}'_n)\neq\boldsymbol{\gamma}'(n)\right]\right|>\delta/2\right)\\ &=\mathbb{P}\left(\sup_{c\in\mathcal{C}}\frac{1}{N}\left|\sum_{n=0}^{N-1}\boldsymbol{y}(n)\right|>\delta/2\right)\end{aligned} \tag{64.126}$$

where we introduced the independent random variables:

$$\boldsymbol{y}(n)\;\triangleq\;\mathbb{I}\left[c(\boldsymbol{h}_n)\neq\boldsymbol{\gamma}(n)\right]-\mathbb{I}\left[c(\boldsymbol{h}'_n)\neq\boldsymbol{\gamma}'(n)\right] \tag{64.127}$$

Since the random variables $\mathbb{I}\left[c(\boldsymbol{h}_n)\neq\boldsymbol{\gamma}(n)\right]$ and $\mathbb{I}\left[c(\boldsymbol{h}'_n)\neq\boldsymbol{\gamma}'(n)\right]$ have identical probability distributions, we conclude that $\boldsymbol{y}(n)$ has zero mean and, more importantly, the distribution of $\boldsymbol{y}(n)$ is symmetric (meaning that both $\boldsymbol{y}(n)$ and $-\boldsymbol{y}(n)$ have the same distribution). This property implies that if we randomly switch the signs of the $\boldsymbol{y}(n)$ terms appearing inside the sum in (64.126), then the sum variable will continue to have the *same* distribution and, therefore, the value of the probability measure (64.126) will not change. This useful observation can be exploited as follows. We introduce N random sign variables, $\{\boldsymbol{s}(n)\}$, independently of $\{\boldsymbol{\gamma}(n),\boldsymbol{\gamma}'(n),\boldsymbol{h}_n,\boldsymbol{h}'_n\}$, such that:

$$\mathbb{P}(\boldsymbol{s}(n)=+1)\;=\;\mathbb{P}(\boldsymbol{s}(n)=-1)\;=\;1/2,\quad n=0,1,\ldots,N-1 \tag{64.128}$$

Then, in view of the symmetry of the distribution of the $\boldsymbol{y}(n)$ random variables, we have

$$\mathbb{P}\left(\sum_{c\in\mathcal{C}}\frac{1}{N}\left|\sum_{n=0}^{N-1}\boldsymbol{y}(n)\right|>\delta/2\right)\;=\;\mathbb{P}\left(\sup_{c\in\mathcal{C}}\frac{1}{N}\left|\sum_{n=0}^{N-1}\boldsymbol{s}(n)\boldsymbol{y}(n)\right|>\delta/2\right) \tag{64.129}$$

Now note from the definition of $\boldsymbol{y}(n)$ in (64.127) that the event

$$\frac{1}{N}\left|\sum_{n=0}^{N-1}\boldsymbol{s}(n)\boldsymbol{y}(n)\right|>\delta/2 \tag{64.130}$$

implies that either one of the following two events is true:

$$\frac{1}{N}\left|\sum_{n=0}^{N-1}\boldsymbol{s}(n)\mathbb{I}\left[c(\boldsymbol{h}_n)\neq\boldsymbol{\gamma}(n)\right]\right|>\frac{\delta}{4}\quad\underline{\text{or}}\quad\frac{1}{N}\left|\sum_{n=0}^{N-1}\boldsymbol{s}(n)\mathbb{I}\left[c(\boldsymbol{h}'_n)\neq\boldsymbol{\gamma}'(n)\right]\right|>\frac{\delta}{4} \tag{64.131}$$

This is because if both events are false and the two terms in the above expression are less than or equal to $\delta/4$ then, from the triangle inequality of norms, we would get:

$$\frac{1}{N}\left|\sum_{n=0}^{N-1}\boldsymbol{s}(n)\boldsymbol{y}(n)\right| \leq \frac{\delta}{2} \tag{64.132}$$

which contradicts (64.130). Therefore, for event (64.130) to hold, it must be the case that event (64.131) also holds. It follows that

$$\begin{aligned}
&\mathbb{P}\left(\sup_{c\in\mathcal{C}}\frac{1}{N}\left|\sum_{n=0}^{N-1}\boldsymbol{s}(n)\boldsymbol{y}(n)\right| > \delta/2\right)\\
&\leq \mathbb{P}\left(\sup_{c\in\mathcal{C}}\frac{1}{N}\left|\sum_{n=0}^{N-1}\boldsymbol{s}(n)\mathbb{I}\left[c(\boldsymbol{h}_n)\neq\boldsymbol{\gamma}(n)\right]\right| > \delta/4 \quad \underline{\text{or}}\right.\\
&\qquad\left.\sup_{c\in\mathcal{C}}\frac{1}{N}\left|\sum_{n=0}^{N-1}\boldsymbol{s}(n)\mathbb{I}\left[c(\boldsymbol{h}'_n)\neq\boldsymbol{\gamma}'(n)\right]\right| > \delta/4\right)\\
&\leq 2\,\mathbb{P}\left(\sup_{c\in\mathcal{C}}\frac{1}{N}\left|\sum_{n=0}^{N-1}\boldsymbol{s}(n)\mathbb{I}\left[c(\boldsymbol{h}_n)\neq\boldsymbol{\gamma}(n)\right]\right| > \delta/4\right)
\end{aligned} \tag{64.133}$$

where in the last inequality we used the union bound for probabilities to eliminate the fictitious data and arrive at a bound that depends only on the original training data. Indeed, combining with (64.126) and (64.113), we conclude that:

$$\mathbb{P}\left(\sup_{c\in\mathcal{C}}|R_{\text{emp}}(c)-R(c)|>\delta\right) \leq 4\,\mathbb{P}\left(\sup_{c\in\mathcal{C}}\frac{1}{N}\left|\sum_{n=0}^{N-1}\boldsymbol{s}(n)\mathbb{I}\left[c(\boldsymbol{h}_n)\neq\boldsymbol{\gamma}(n)\right]\right| > \delta/4\right) \tag{64.134}$$

Note further that the term on the right-hand side involves the sum of a collection of independent random variables. This property will facilitate the last step given further ahead and which will rely on the Hoeffding inequality. In order to prepare for that step, we need to explain how to move the sup operation on the right-hand side outside of the probability expression — see (64.136).

(Union bound step). Given N feature vectors, $\{h_n\}$, there exist at most $\mathcal{S}(\mathcal{C},N)$ distinct dichotomies (or labeling) that can be generated by the set of classifiers, and where $\mathcal{S}(\mathcal{C},N)$ denotes the corresponding shatter coefficient. Let $\mathcal{C}_s$ denote the smallest subset of $\mathcal{C}$ that is able to generate all these dichotomies. Then, obviously,

$$|\mathcal{C}_s| \leq \mathcal{S}(\mathcal{C},N) \tag{64.135}$$

Using the probability union bound, we now write:

$$\begin{aligned}
&\mathbb{P}\left(\sup_{c\in\mathcal{C}}\frac{1}{N}\left|\sum_{n=0}^{N-1}\boldsymbol{s}(n)\mathbb{I}\left[c(\boldsymbol{h}_n)\neq\boldsymbol{\gamma}(n)\right]\right|>\delta/4\right)\\
&=\mathbb{P}\left(\sup_{c\in\mathcal{C}_s}\frac{1}{N}\left|\sum_{n=0}^{N-1}\boldsymbol{s}(n)\mathbb{I}\left[c(\boldsymbol{h}_n)\neq\boldsymbol{\gamma}(n)\right]\right|>\delta/4\right)\\
&=\mathbb{P}\left(\bigcup_{c\in\mathcal{C}_s}\left\{\frac{1}{N}\left|\sum_{n=0}^{N-1}\boldsymbol{s}(n)\mathbb{I}\left[c(\boldsymbol{h}_n)\neq\boldsymbol{\gamma}(n)\right]\right|>\delta/4\right\}\right)\\
&\le\sum_{c\in\mathcal{C}_s}\mathbb{P}\left(\frac{1}{N}\left|\sum_{n=0}^{N-1}\boldsymbol{s}(n)\mathbb{I}\left[c(\boldsymbol{h}_n)\neq\boldsymbol{\gamma}(n)\right]\right|>\delta/4\right)\\
&\le|\mathcal{C}_s|\sup_{c\in\mathcal{C}_s}\left\{\mathbb{P}\left(\frac{1}{N}\left|\sum_{n=0}^{N-1}\boldsymbol{s}(n)\mathbb{I}\left[c(\boldsymbol{h}_n)\neq\boldsymbol{\gamma}(n)\right]\right|>\delta/4\right)\right\}\\
&\le\mathcal{S}(\mathcal{C},N)\sup_{c\in\mathcal{C}_s}\left\{\mathbb{P}\left(\frac{1}{N}\left|\sum_{n=0}^{N-1}\boldsymbol{s}(n)\mathbb{I}\left[c(\boldsymbol{h}_n)\neq\boldsymbol{\gamma}(n)\right]\right|>\delta/4\right)\right\}\\
&=\mathcal{S}(\mathcal{C},N)\sup_{c\in\mathcal{C}}\left\{\mathbb{P}\left(\frac{1}{N}\left|\sum_{n=0}^{N-1}\boldsymbol{s}(n)\mathbb{I}\left[c(\boldsymbol{h}_n)\neq\boldsymbol{\gamma}(n)\right]\right|>\delta/4\right)\right\}
\end{aligned}\tag{64.136}$$

Observe that, as claimed earlier, the sup operation is now outside the probability calculation. Observe also that the bound involves the class size, $\mathcal{S}(\mathcal{C},N)$, as well as the independent random variables $\boldsymbol{s}(n)\mathbb{I}\left[c(\boldsymbol{h}_n)\neq\boldsymbol{\gamma}(n)\right]$.

(Hoeffding inequality). The final step is to exploit this independence along with Hoeffding inequality to bound the right-hand side of (64.136). Thus, let

$$\boldsymbol{b}(n)\;\triangleq\;\boldsymbol{s}(n)\mathbb{I}\left[c(\boldsymbol{h}_n)\neq\boldsymbol{\gamma}(n)\right]\tag{64.137}$$

Each of these random variables has zero mean and its value is $+1, 0$, or -1. In particular, the value of each $\boldsymbol{b}(n)$ is bounded between -1 and 1. It then follows from Hoeffding inequality (3.231b) by using $\Delta = 4N$, that

$$\begin{aligned}
\mathbb{P}\left(\frac{1}{N}\left|\sum_{n=0}^{N-1}\boldsymbol{b}(n)\right|>\delta/4\right)&=\mathbb{P}\left(\left|\sum_{n=0}^{N-1}\boldsymbol{b}(n)\right|>N\delta/4\right)\\
&\le 2e^{-\frac{2(N\delta/4)^2}{4N}}\\
&=2e^{-N\delta^2/32}
\end{aligned}\tag{64.138}$$

The bound on the right-hand side is independent of the classifier set, $\mathcal{C}$, so that

$$\sup_{c\in\mathcal{C}}\left\{\mathbb{P}\left(\frac{1}{N}\left|\sum_{n=0}^{N-1}\boldsymbol{s}(n)\mathbb{I}\left[c(\boldsymbol{h}_n)\neq\boldsymbol{\gamma}(n)\right]\right|>\delta/4\right)\right\}\;\le\;2e^{-N\delta^2/32}\tag{64.139}$$

Substituting into (64.136) and (64.134) we obtain

$$\mathbb{P}\left(\sup_{c\in\mathcal{C}}\frac{1}{N}\left|\sum_{n=0}^{N-1}\boldsymbol{s}(n)\mathbb{I}\left[c(\boldsymbol{h}_n)\neq\boldsymbol{\gamma}(n)\right]\right|>\delta/4\right)\;\le\;2\mathcal{S}(\mathcal{C},N)e^{-N\delta^2/32}\tag{64.140}$$

as well as

$$\mathbb{P}\left(\sup_{c\in\mathcal{C}}|R_{\text{emp}}(c)-R(c)|>\delta\right)\;\le\;8\mathcal{S}(\mathcal{C},N)e^{-N\delta^2/32}\tag{64.141}$$

We finally arrive at the desired result (64.111) by using the bound (64.88) for the shatter coefficient, $\mathcal{S}(\mathcal{C}, N)$.

■

64.D RADEMACHER COMPLEXITY

There is an alternative method to examine the generalization ability of learning algorithms by relying on the concept of the *Rademacher complexity*. Useful overviews appear in Boucheron, Bousquet, and Lugosi (2005), Shalev-Shwartz and Ben-David (2014), Mohri, Rostamizadeh, and Talwalkar (2018), and Wainwright (2019).

Recall that the analysis in the body of the chapter, and the derivations in the last appendix, focused on binary classification problems where $\boldsymbol{\gamma} \in \{\pm 1\}$ and on the $0/1-$loss function. The analysis showed that classification structures with medium VC dimensions are able to learn well with high likelihood for *any* data distribution. In a sense, this conclusion amounts to a generalization guarantee under a worst case scenario since it holds irrespective of the data distribution. It is reasonable to expect that some data distributions are more favorable than others and, therefore, it would be desirable to seek generalization results that have some dependence on the data distribution. The framework that is based on the Rademacher complexity will allow for this possibility and will lead to tighter error bounds. The approach will also apply to multiclass classification problems and to other loss functions, and is not restricted to binary classification or $0/1-$losses. The analysis will continue to lead to similar reassuring conclusions about the ability of learning methods to generalize for mild VC dimensions. However, the conclusions will now be dependent on the data distribution and will not correspond to worst-case statements that hold for any distribution.

Before formally introducing the concept, we remark that we have already encountered some elements of Rademacher complexity in the last appendix, for example, when we introduced the sign variables $\{\boldsymbol{s}(n)\}$ in (64.128) and incorporated them into the probability expression (64.129).

Definition over a set

Consider initially a subset $\mathcal{A} \subset \mathbb{R}^N$, with cardinality $|\mathcal{A}|$. Select an arbitrary vector $a \in \mathcal{A}$, which is $N-$dimensional, and denote its individual scalar entries by $a = \mathrm{col}\{a_n\}$, for $n = 1, 2, \ldots, N$. The Rademacher complexity of the set of vectors $\mathcal{A}$ is a scalar denoted by $\mathcal{R}_N(\mathcal{A})$ and defined as the following expectation:

$$\mathcal{R}_N(\mathcal{A}) \;=\; \mathbb{E}\,\boldsymbol{\sigma}\left\{\sup_{a\in\mathcal{A}}\left(\frac{1}{N}\sum_{n=1}^{N}\boldsymbol{\sigma}_n a_n\right)\right\} \tag{64.142}$$

where the $\{\boldsymbol{\sigma}_n\}$ are called the *Rademacher variables*: they are random variables chosen independently of each other with

$$\mathbb{P}(\boldsymbol{\sigma}_n = +1) = \mathbb{P}(\boldsymbol{\sigma}_n = -1) = 1/2 \tag{64.143}$$

The expectation in (64.142) is relative to the randomness in the Rademacher variables. In the definition, the entries of each $a \in \mathcal{A}$ are first modulated by (or correlated with) the binary variables $\{\boldsymbol{\sigma}_n\}$ before computing the sample average. The expected largest value for this sample average is taken as the Rademacher complexity of the set. Observe that $\mathcal{R}_N(\mathcal{A})$ depends on N.

One famous result concerning $\mathcal{R}_N(\mathcal{A})$ is the Massart lemma. Let Δ denote the largest Euclidean norm within $\mathcal{A}$:

$$\Delta \;\triangleq\; \sup_{a\in\mathcal{A}} \|a\| \tag{64.144}$$

Massart lemma (Massart (2000)). *The Rademacher complexity of a set of vectors $\mathcal{A}$ is bounded by*

$$\mathcal{R}_N(\mathcal{A}) \leq \frac{\Delta}{N} \times \sqrt{2 \ln |\mathcal{A}|} \tag{64.145}$$

Proof: We follow steps similar to Shalev-Shwartz and Ben-David (2014, Ch. 26) and Mohri, Rostamizadeh, and Talwalkar (2018, Ch. 3). The argument uses the Hoeffding lemma, which we encountered earlier in (3.233). Thus, for any positive scalar t, we consider the following sequence of calculations:

$$\begin{aligned}
e^{t\mathcal{R}_N(\mathcal{A})} &\overset{(64.142)}{=} \exp\left\{ t \times \mathbb{E}\,\boldsymbol{\sigma} \left[\sup_{a \in \mathcal{A}} \left(\frac{1}{N} \sum_{n=1}^{N} \boldsymbol{\sigma}_n a_n \right) \right] \right\} \\
&\overset{(a)}{\leq} \mathbb{E}\,\boldsymbol{\sigma} \exp\left\{ t \times \sup_{a \in \mathcal{A}} \left(\frac{1}{N} \sum_{n=1}^{N} \boldsymbol{\sigma}_n a_n \right) \right\} \\
&\overset{(b)}{=} \mathbb{E}\,\boldsymbol{\sigma} \sup_{a \in \mathcal{A}} \left[\exp\left\{ t \times \left(\frac{1}{N} \sum_{n=1}^{N} \boldsymbol{\sigma}_n a_n \right) \right\} \right] \\
&\overset{(c)}{\leq} \sum_{a \in \mathcal{A}} \mathbb{E}\,\boldsymbol{\sigma} \exp\left\{ t \times \left(\frac{1}{N} \sum_{n=1}^{N} \boldsymbol{\sigma}_n a_n \right) \right\} \\
&= \sum_{a \in \mathcal{A}} \mathbb{E}\,\boldsymbol{\sigma} \left\{ \prod_{n=1}^{N} \exp\left(t \boldsymbol{\sigma}_n a_n / N \right) \right\} \\
&\overset{(d)}{=} \sum_{a \in \mathcal{A}} \prod_{n=1}^{N} \mathbb{E}\,\boldsymbol{\sigma} \left\{ \exp(t \boldsymbol{\sigma}_n a_n / N) \right\}
\end{aligned} \tag{64.146}$$

where step (a) uses Jensen inequality (8.77) and the fact that the function e^x is convex, step (b) switches the order of the sup and exponentiation operations since $t > 0$, step (c) bounds the sup by the sum of the entries, and step (d) uses the fact that the Rademacher variables $\{\boldsymbol{\sigma}_n\}$ are independent of each other. We are now ready to apply the Hoeffding bound (3.233). Let

$$\boldsymbol{y}_n \triangleq \boldsymbol{\sigma}_n a_n \tag{64.147}$$

and note that $\mathbb{E}\,\boldsymbol{y}_n = 0$ since $\mathbb{E}\,\boldsymbol{\sigma}_n = 0$. Moreover, the value of the variable $\boldsymbol{y}(n)$ is either $-a_n$ or a_n depending on the polarity of $\boldsymbol{\sigma}_n$. It follows from (3.233) that

$$\mathbb{E}\, e^{t\boldsymbol{y}_n/N} \leq e^{t^2(2a_n)^2/8N^2} = e^{t^2 a_n^2/2N^2} \tag{64.148}$$

Substituting into (64.146) gives

$$\begin{aligned} e^{t\mathcal{R}_N(\mathcal{A})} &\le \sum_{a\in\mathcal{A}}\prod_{n=1}^{N} e^{t^2a_n^2/2N^2} \\ &= \sum_{a\in\mathcal{A}} \exp\left\{\frac{t^2}{2N^2}\sum_{n=1}^{N} a_n^2\right\} \\ &\le |\mathcal{A}| \times \exp\left\{\frac{t^2\Delta^2}{2N^2}\right\} \\ &= \exp\left\{\ln|\mathcal{A}| + \frac{t^2\Delta^2}{2N^2}\right\} \end{aligned} \tag{64.149}$$

or, equivalently,

$$\mathcal{R}_N(\mathcal{A}) \le \frac{\ln|\mathcal{A}|}{t} + \frac{t\Delta^2}{2N^2} \tag{64.150}$$

We are free to select the parameter t. We therefore minimize the upper bound over t to get

$$t = \frac{N}{\Delta}\times\sqrt{2\ln|\mathcal{A}|} \tag{64.151}$$

so that, upon substitution into the right-hand side of (64.150), we conclude that

$$\mathcal{R}_N(\mathcal{A}) \le \frac{\Delta}{N}\times\sqrt{2\ln|\mathcal{A}|} \tag{64.152}$$

■

Example 64.5 (**Finite set of classifiers**) Consider a *finite* set of binary classifiers

$$\mathcal{C} = \Big\{c(h) : \mathbb{R}^M \to \{\pm 1\}\Big\} \tag{64.153}$$

and a collection of N feature vectors $\{h_n \in \mathbb{R}^M\}$, for $n = 1, 2, \ldots, N$. Each classifier $c \in \mathcal{C}$ provides one possible labeling for the feature vectors, which we denote by

$$a \;\triangleq\; \mathrm{col}\Big\{c(h_1),\, c(h_2),\, \ldots,\, c(h_N)\Big\} \in \{\pm 1\}^N \tag{64.154}$$

This is a vector of size $N \times 1$ with entries ± 1. By constructing the label vectors $\{a\}$ for each of the classifiers $c \in \mathcal{C}$, we end up with a finite collection of vectors:

$$\mathcal{A} = \Big\{a \,|\, a = \mathrm{col}\{c(h_n)\},\; c \in \mathcal{C}\Big\} \tag{64.155}$$

In this example, the cardinality of $\mathcal{A}$ is equal to the cardinality of $\mathcal{C}$:

$$|\mathcal{A}| \;=\; |\mathcal{C}| \tag{64.156}$$

Moreover, the bound Δ is easily seen to be $\Delta = \sqrt{N}$. Using the Massart bound (64.145) we conclude that the Rademacher complexity that is associated with the set of classifiers $\mathcal{C}$ is bounded by

$$\mathcal{R}_N(A) \le \sqrt{\frac{2\ln|\mathcal{C}|}{N}} \tag{64.157}$$

Example 64.6 (**Some intuition on the Rademacher complexity**) We use the previous example to gain some intuition into the definition of $\mathcal{R}_N(A)$, which we rewrite in terms of the binary classifiers:

$$\mathcal{R}_N(A) = \mathbb{E}\,\boldsymbol{\sigma}\left\{\sup_{c\in\mathcal{C}}\left(\frac{1}{N}\sum_{n=1}^{N}\boldsymbol{\sigma}_n c(h_n)\right)\right\} \tag{64.158}$$

Observe that the summation on the right-hand side is computing the correlation between the *random* vector of Rademacher parameters $\{\boldsymbol{\sigma}_1,\ldots,\boldsymbol{\sigma}_N\}$ and the label vector $\{c(h_1),\ldots,c(h_N)\}$ that results from applying the classifier $c(h)$. A high correlation value means that this label vector is able to match relatively well the particular choice of Rademacher labels $\{\boldsymbol{\sigma}_n\}$. The Rademacher complexity is therefore assessing the largest possible correlation that the class of classifiers $\mathcal{C}$ is able to attain on average. The larger this value is, the more likely the class $\mathcal{C}$ will be able to fit randomly chosen label vectors — Prob. 64.37 provides additional motivation. It follows from this explanation that the Rademacher complexity provides an assessment of the representation power (or richness or expressiveness) of a class of classifiers, $\mathcal{C}$. In this sense, it plays a role similar to the VC dimension. However, unlike the VC concept, the Rademacher complexity is not limited to binary classification problems.

Example 64.7 (**Rademacher complexity and VC dimension**) The Massart lemma helps link the two important concepts of Rademacher complexity and VC dimension. To see this, we continue with Example 64.5 but consider now the situation in which the set of classifiers $\mathcal{C}$ has *infinitely* many elements. We already know how many different labeling vectors this set can generate for the $N-$feature vectors $\{h_n\}$. This number is given by the shatter coefficient $\mathcal{S}(\mathcal{C},N)$, which, in view of Sauer lemma, we showed in (64.88) to be bounded by

$$\mathcal{S}(\mathcal{C},N) \le (Ne/\mathrm{VC})^{\mathrm{VC}} \tag{64.159}$$

Therefore, if we again generate the set of vectors $\mathcal{A}$ that corresponds to this class of classifiers $\mathcal{C}$, its cardinality will be bounded by this same value:

$$|\mathcal{A}| \;\le\; (Ne/\mathrm{VC})^{\mathrm{VC}} \tag{64.160}$$

Using the Massart bound (64.145) we conclude that the Rademacher complexity that is associated with the class of classifiers $\mathcal{C}$ is now bounded by

$$\mathcal{R}_N(\mathcal{C}) \le \sqrt{\frac{2}{N}\mathrm{VC}\;\ln\!\left(\frac{Ne}{\mathrm{VC}}\right)} \tag{64.161}$$

Definition over functions

We can extend the definition of the Rademacher complexity to sets of scalar real-valued functions $Q\in\mathcal{Q}$, where each $Q(y):\mathbb{R}\to\mathbb{R}$. We use the letter Q because it will often correspond to the loss function in the context of learning algorithms. We also use the letter y because it will correspond to the margin variable $y=\gamma\widehat{\gamma}$. For now, we treat Q and y generically and later specialize them to the learning context.

We consider a collection of N scalar variables $\{y_n\}$ and define the *empirical* Rademacher complexity of the set $\mathcal{Q}$ as follows using the hat notation:

$$\widehat{\mathcal{R}}_N(\mathcal{Q}) \;=\; \mathbb{E}\,\boldsymbol{\sigma}\left\{\sup_{Q\in\mathcal{Q}}\left(\frac{1}{N}\sum_{n=1}^{N}\boldsymbol{\sigma}_n Q(y_n)\right)\right\} \tag{64.162}$$

where the $\{\boldsymbol{\sigma}_n\}$ continue to be the *Rademacher variables*, which assume the values $\{\pm1\}$ uniformly and independently of each other. In definition (64.162), the function $Q(\cdot)$ is evaluated at each y_n and modulated by the binary variable $\{\boldsymbol{\sigma}_n\}$ before computing

the sample average. The expected largest value for this sample average, over the set of functions, is taken as the *empirical* Rademacher complexity for the set $\mathcal{Q}$. The reason for the designation "empirical" is because the variables $\{y_n\}$ will usually correspond to independent observations of some random variable $\boldsymbol{y} \sim f_{\boldsymbol{y}}(y)$. If we then compute the expectation relative to the distribution of $\boldsymbol{y}$ we obtain the Rademacher complexity without the hat notation:

$$\mathcal{R}_N(\mathcal{Q}) \;=\; \mathbb{E}_{\,\boldsymbol{y}} \left\{ \widehat{\boldsymbol{\mathcal{R}}}_N(\mathcal{Q}) \right\} \tag{64.163}$$

where we are now treating the empirical complexity as a random variable due to its dependence on the random observations $\{\boldsymbol{y}_n\}$. Note that by computing the expectation relative to the distribution of $\boldsymbol{y}$, the Rademacher complexity becomes dependent on this distribution. This line of reasoning is unlike the analysis carried out in the previous appendix where, for example, bounds on the shatter coefficient (or growth function) were derived independently of any distribution.

Example 64.8 (**Useful property**) Consider a class of functions $c \in \mathcal{C}$, where each c can be expressed in the form $c(y) = aQ(y) + b$ for some constants $a, b \in \mathbb{R}$ and function $Q(y)$ from another set $\mathcal{Q}$. We can relate the Rademacher complexities of both sets $\{\mathcal{C}, \mathcal{Q}\}$ as follows:

$$\mathcal{R}_N(\mathcal{C}) = |a|\, \mathcal{R}_N(\mathcal{Q}) \tag{64.164a}$$

$$\widehat{\mathcal{R}}_N(\mathcal{C}) = |a|\, \widehat{\mathcal{R}}_N(\mathcal{Q}) \tag{64.164b}$$

Proof: It is sufficient to establish the result for the empirical Rademacher complexity. Thus, note from the definition that

$$\begin{aligned}
\widehat{\mathcal{R}}_N(\mathcal{C}) &= \mathbb{E}_{\boldsymbol{\sigma}} \left\{ \sup_{Q \in \mathcal{Q}} \left(\frac{1}{N} \sum_{n=1}^{N} \boldsymbol{\sigma}_n C(y_n) \right) \right\} \\
&= \mathbb{E}_{\boldsymbol{\sigma}} \left\{ \sup_{Q \in \mathcal{Q}} \left(\frac{1}{N} \sum_{n=1}^{N} \boldsymbol{\sigma}_n aQ(y_n) + \frac{1}{N} \sum_{n=1}^{N} \boldsymbol{\sigma}_n b \right) \right\} \\
&= \mathbb{E}_{\boldsymbol{\sigma}} \left\{ \sup_{Q \in \mathcal{Q}} \left(\frac{1}{N} \sum_{n=1}^{N} \boldsymbol{\sigma}_n aQ(y_n) \right) \right\} + \underbrace{\mathbb{E}_{\boldsymbol{\sigma}} \left\{ \frac{1}{N} \sum_{n=1}^{N} \boldsymbol{\sigma}_n b \right\}}_{\to 0} \\
&= |a|\, \mathbb{E}_{\boldsymbol{\sigma}} \left\{ \sup_{Q \in \mathcal{Q}} \left(\frac{1}{N} \sum_{n=1}^{N} \boldsymbol{\sigma}_n Q(y_n) \right) \right\} \\
&= |a|\, \widehat{\mathcal{R}}_N(\mathcal{Q})
\end{aligned} \tag{64.165}$$

where $|a|$ is used since the polarities of the $\{\boldsymbol{\sigma}_n\}$ can be switched between $+1$ to -1. Any value for the sample average hat is achieved using a can also be achieved using $-a$ with the polarities of $\boldsymbol{\sigma}_n$ switched. Thus, for all practical purposes, we can work with $|a|$.

■

In preparation for the main result of this appendix showing how the Rademacher complexity leads to generalization bounds, we introduce some intermediate concepts and results.

Empirical and stochastic risks. With each loss function $Q \in \mathcal{Q}$ we associate two risk

functions:

$$\mathbb{E}_{\boldsymbol{y}}\, Q(\boldsymbol{y}), \qquad (\textbf{stochastic risk}) \tag{64.166a}$$

$$\frac{1}{N}\sum_{n=1}^{N} Q(y_n), \qquad (\textbf{empirical risk}) \tag{64.166b}$$

where the first expression is the average loss value over the distribution of the data $\boldsymbol{y}$, while the second expression is a sample average value obtained from a collection of N realizations $\{y_1, y_2, \ldots, y_N\}$. Although under ergodicity, these two quantities are expected to approach each other as $N \to \infty$, they are nevertheless generally different for finite N. The difference between the two risks also varies with the choice of Q. We denote the worst case difference by the notation:

$$\phi(y_1, \ldots, y_N) \;\triangleq\; \sup_{Q\in\mathcal{Q}} \left\{ \mathbb{E}_{\boldsymbol{y}}\, Q(\boldsymbol{y}) - \frac{1}{N}\sum_{n=1}^{N} Q(y_n) \right\} \tag{64.167}$$

(**worst excess risk function**)

The function $\phi(\cdot)$ is dependent on the N variables $\{y_n\}$.

Bounded variations. We will assume that the loss functions $Q(y)$ assume values in some *bounded* interval, namely, $Q(y) : \mathbb{R} \to [a, b]$ with $a < b$. We denote the width of this interval by

$$d \;\triangleq\; b - a \tag{64.168}$$

Alternatively, we can set $d = \sup_y |Q(y)|$. It then follows that the excess risk function $\phi(\cdot)$ will have *bounded variations* (i.e., if any of its entries changes, the function will change by a bounded amount). Specifically, if y_m changes to y'_m for any entry of index m, it will hold that

$$\Big|\phi(y_{n\neq m},\, y_m) \;-\; \phi(y_{n\neq m},\, y'_m)\Big| \;\leq\; d/N \tag{64.169}$$

for all $\{y_n,\ n \neq m\}$.

Proof of (64.169): To simplify the notation, we let

$$\mathcal{Y} = \{y_1, \ldots, y_{m-1}, y_m, y_{m+1}, \ldots, y_N\} \tag{64.170}$$

$$\mathcal{Y}_m = \{y_1, \ldots, y_{m-1}, y'_m, y_{m+1}, \ldots, y_N\} \tag{64.171}$$

denote the collection of observations with y_m replaced by y'_m, while all other entries remain unchanged. Let $Q^\star(\cdot)$ be the function that attains the supremum in (64.167) with the observations $\{y_1, \ldots, y_N\}$ so that

$$\phi(\mathcal{Y}) = \mathbb{E}_{\boldsymbol{y}}\, Q^\star(\boldsymbol{y}) - \frac{1}{N}\sum_{n\in\mathcal{Y}} Q^\star(y_n) \tag{64.172}$$

Then, the desired result follows from the following sequence of inequalities:

$$
\begin{aligned}
&|\phi(\mathcal{Y}) - \phi(\mathcal{Y}')| \\
&= \left| \mathbb{E}_{\boldsymbol{y}} Q^\star(\boldsymbol{y}) - \frac{1}{N}\sum_{n\in\mathcal{Y}} Q^\star(y_n) \;-\; \sup_{Q\in\mathcal{Q}} \left\{ \mathbb{E}_{\boldsymbol{y}} Q(\boldsymbol{y}) - \frac{1}{N}\sum_{n\in\mathcal{Y}_m} Q(y_n) \right\} \right| \\
&\overset{(a)}{\leq} \left| \mathbb{E}_{\boldsymbol{y}} Q^\star(\boldsymbol{y}) - \frac{1}{N}\sum_{n\in\mathcal{Y}} Q^\star(y_n) \;-\; \left(\mathbb{E}_{\boldsymbol{y}} Q^\star(\boldsymbol{y}) - \frac{1}{N}\sum_{n\in\mathcal{Y}_m} Q^\star(y_n) \right) \right| \\
&\overset{(b)}{=} \frac{1}{N}\left| Q^\star(y_m') - Q^\star(y_m) \right| \\
&\overset{(64.169)}{\leq} d/N
\end{aligned}
$$

where step (a) is because we employed the suboptimal $Q^\star(\cdot)$ in the rightmost supremum operation, and step (b) is because the sets $\mathcal{Y}$ and $\mathcal{Y}_m$ differ by a single entry.

■

One useful consequence of the bounded variation property (64.169) is that we can bound how close the risk difference $\phi(\mathcal{Y})$ gets to its mean value. For this purpose, we appeal to the McDiarmid inequality (3.259a) and note that for any given $\delta > 0$:

$$
\mathbb{P}\Big(\phi(\mathcal{Y}) - \mathbb{E}_{\boldsymbol{y}} \phi(\boldsymbol{\mathcal{Y}}) \geq \delta \Big) \leq e^{-2\delta^2 / \sum_{n=1}^N d^2/N^2} \;=\; e^{-2N\delta^2/d^2} \tag{64.173}
$$

Thus, assume that we wish to determine the value of δ such that $\phi(\mathcal{Y})$ is $\delta-$close to its mean $\mathbb{E}_{\boldsymbol{y}} \phi(\boldsymbol{\mathcal{Y}})$ with probability $1-\epsilon$. Then, setting

$$
e^{-2N\delta^2/d^2} \leq \epsilon \tag{64.174}
$$

we can solve for δ:

$$
\delta \geq d\sqrt{\frac{1}{2N}\ln(1/\epsilon)} \tag{64.175}
$$

Substituting into (64.173) we conclude that with high probability of at least $1-\epsilon$:

$$
\boxed{\phi(\mathcal{Y}) \leq \mathbb{E}_{\boldsymbol{y}} \phi(\boldsymbol{\mathcal{Y}}) + d\sqrt{\frac{1}{2N}\ln\Big(\frac{1}{\epsilon}\Big)}} \tag{64.176}
$$

Bounding the average risk. It turns out that the mean quantity $\mathbb{E}_{\boldsymbol{y}} \phi(\boldsymbol{\mathcal{Y}})$ in the above expression can be bounded by the Rademacher complexity of the set $\mathcal{Q}$ as follows:

$$
\boxed{\mathbb{E}_{\boldsymbol{y}} \phi(\boldsymbol{\mathcal{Y}}) \;\leq\; 2\,\mathcal{R}_N(\mathcal{Q})} \tag{64.177}
$$

Proof: We follow steps similar to the proof of Theorem 8 in Bartlett and Mendelson (2002); see also Shalev-Shwartz and Ben-David (2014, Ch. 26) and Mohri, Rostamizadeh, and Talwalkar (2018, Ch. 3). We introduce a fictitious collection of samples $\{y_1', y_2', \ldots, y_N'\}$ and denote it by $\mathcal{Y}'$. This set consists of realizations of a random variable $\boldsymbol{y}'$ with the the same distribution as $\boldsymbol{y}$, except that the realizations $\{y_n'\}$ are chosen independently of the original realizations $\{y_n\}$. Then, it is clear that

$$
\mathbb{E}_{\boldsymbol{y}'} \left\{ \frac{1}{N}\sum_{n=1}^N Q(\boldsymbol{y}_n') \right\} \;=\; \mathbb{E}_{\boldsymbol{y}} Q(\boldsymbol{y}) \tag{64.178}
$$

so that

$$\begin{aligned}\mathbb{E}_{\boldsymbol{y}}\,\phi(\mathcal{Y}) &\overset{(64.167)}{=} \mathbb{E}_{\boldsymbol{y}}\left(\sup_{Q\in\mathcal{Q}}\left\{\mathbb{E}_{\boldsymbol{y}}\,Q(\boldsymbol{y}) - \frac{1}{N}\sum_{n=1}^{N}Q(\boldsymbol{y}_n)\right\}\right)\\ &\overset{(64.178)}{=} \mathbb{E}_{\boldsymbol{y}}\left(\sup_{Q\in\mathcal{Q}}\left\{\mathbb{E}_{\boldsymbol{y}'}\left[\frac{1}{N}\sum_{n=1}^{N}Q(\boldsymbol{y}'_n)\right] - \frac{1}{N}\sum_{n=1}^{N}Q(\boldsymbol{y}_n)\right\}\right)\\ &\overset{(a)}{\leq} \mathbb{E}_{\boldsymbol{y},\boldsymbol{y}'}\left\{\sup_{Q\in\mathcal{Q}}\frac{1}{N}\sum_{n=1}^{N}\Big(Q(\boldsymbol{y}'_n) - Q(\boldsymbol{y}_n)\Big)\right\} \qquad (64.179)\end{aligned}$$

where step (a) uses Jensen inequality (8.77) and the fact that the sup function is convex (see Prob. 64.32). Now note that since $\{\boldsymbol{y}_n, \boldsymbol{y}'_n\}$ are equally distributed and independent of each other, the value of the last expectation will not change if we switch the roles of $\boldsymbol{y}_n$ and $\boldsymbol{y}'_n$ for any n. In other words, it will holds that

$$\begin{aligned}&\mathbb{E}_{\boldsymbol{y},\boldsymbol{y}'}\left\{\sup_{Q\in\mathcal{Q}}\frac{1}{N}\sum_{n=1}^{N}\Big(Q(\boldsymbol{y}'_n) - Q(\boldsymbol{y}_n)\Big)\right\}\\ &= \mathbb{E}_{\boldsymbol{\sigma},\boldsymbol{y},\boldsymbol{y}'}\left\{\sup_{Q\in\mathcal{Q}}\frac{1}{N}\sum_{n=1}^{N}\boldsymbol{\sigma}_n\Big(Q(\boldsymbol{y}'_n) - Q(\boldsymbol{y}_n)\Big)\right\} \qquad (64.180)\end{aligned}$$

where we have incorporated the Rademacher parameters $\{\boldsymbol{\sigma}_n\}$ on the right-hand side; recall that they have zero mean and are chosen uniformly from $\{\pm 1\}$. We can therefore write

$$\begin{aligned}\mathbb{E}_{\boldsymbol{y}}\,\phi(\mathcal{Y}) &\leq \mathbb{E}_{\boldsymbol{\sigma},\boldsymbol{y},\boldsymbol{y}'}\left\{\sup_{Q\in\mathcal{Q}}\frac{1}{N}\sum_{n=1}^{N}\boldsymbol{\sigma}_n\Big(Q(\boldsymbol{y}'_n) - Q(\boldsymbol{y}_n)\Big)\right\}\\ &\leq \underbrace{\mathbb{E}_{\boldsymbol{\sigma},\boldsymbol{y}'}\left\{\sup_{Q\in\mathcal{Q}}\frac{1}{N}\sum_{n=1}^{N}\boldsymbol{\sigma}_n Q(\boldsymbol{y}'_n)\right\}}_{=\mathcal{R}_N(\mathcal{Q})} + \underbrace{\mathbb{E}_{\boldsymbol{\sigma},\boldsymbol{y}}\left\{\sup_{Q\in\mathcal{Q}}\frac{1}{N}\sum_{n=1}^{N}\boldsymbol{\sigma}_n Q(\boldsymbol{y}_n)\right\}}_{=\mathcal{R}_N(\mathcal{Q})}\\ &= 2\,\mathcal{R}_N(\mathcal{Q}) \qquad (64.181)\end{aligned}$$

■

Main generalization theorem

We are now ready to establish the main result, which relates the Rademacher measure of complexity to the generalization ability of learning algorithms. The bounds below, which are due to Koltchinskii and Panchenko (2000,2002) and Bartlett and Mendelson (2002), show that, with high probability, the stochastic risk of a learning algorithm will be close to its empirical risk by an amount that depends on the Rademacher complexity. One main difference between the two bounds shown in the statement is that the second result (64.182b) is data-dependent; it is stated in terms of the empirical complexity $\widehat{\mathcal{R}}_N(\mathcal{Q})$, which in principle can be estimated from the data observations $\{y_1, \ldots, y_N\}$. This is in contrast to the first bound, which employs the actual complexity $\mathcal{R}_N(\mathcal{Q})$; its computation requires averaging over the data distribution, $\boldsymbol{y} \sim f_{\boldsymbol{y}}(y)$.

One-sided generalization bounds (Koltchinskii and Panchenko (2000,2002), Bartlett and Mendelson (2002)). *Consider a set $\mathcal{Q} \in \mathcal{Q}$ of loss functions with each $Q(y) : \mathbb{R} \to [a, b]$. Let $d = b - a$. Then, for every $Q \in \mathcal{Q}$ and with high probability of at least $1 - \epsilon$,*

either of the following bounds holds in terms of the empirical or regular Rademacher complexity:

$$\mathbb{E}_{\boldsymbol{y}} Q(\boldsymbol{y}) \leq \frac{1}{N}\sum_{n=1}^{N} Q(y_n) + 2\,\mathcal{R}_N(\mathcal{Q}) + d\sqrt{\frac{1}{2N}\ln(1/\epsilon)} \tag{64.182a}$$

$$\mathbb{E}_{\boldsymbol{y}} Q(\boldsymbol{y}) \leq \frac{1}{N}\sum_{n=1}^{N} Q(y_n) + 2\,\widehat{\mathcal{R}}_N(\mathcal{Q}) + 3d\sqrt{\frac{1}{2N}\ln(2/\epsilon)} \tag{64.182b}$$

Proof: We put together several of the results derived so far to note that

$$\begin{aligned}
\mathbb{E}_{\boldsymbol{y}} Q(\boldsymbol{y}) &= \mathbb{E}_{\boldsymbol{y}} Q(\boldsymbol{y}) + \frac{1}{N}\sum_{n=1}^{N} Q(y_n) - \frac{1}{N}\sum_{n=1}^{N} Q(y_n) \\
&\overset{(64.167)}{\leq} \frac{1}{N}\sum_{n=1}^{N} Q(y_n) \;+\; \phi(y_1,\ldots,y_N) \\
&\overset{(64.176)}{\leq} \frac{1}{N}\sum_{n=1}^{N} Q(y_n) + \mathbb{E}_{\boldsymbol{y}}\,\phi(\boldsymbol{y}) + d\sqrt{\frac{1}{2N}\ln(1/\epsilon)} \\
&\overset{(64.177)}{\leq} \frac{1}{N}\sum_{n=1}^{N} Q(y_n) + 2\,\mathcal{R}_N(\mathcal{Q}) + d\sqrt{\frac{1}{2N}\ln(1/\epsilon)}
\end{aligned} \tag{64.183}$$

which establishes (64.182a). To establish the second inequality, we first note that it is straightforward to verify that the empirical Rademacher complexity $\widehat{\mathcal{R}}_N(\mathcal{Q})$ satisfies the bounded variations property with the same bound d/N as $\phi(\cdot)$. It then follows from the second McDiarmid inequality (3.259b) that, for any $\delta > 0$:

$$\mathbb{P}\Big(\Big|\,\widehat{\mathcal{R}}_N(\mathcal{Q}) - \mathcal{R}_N(\mathcal{Q})\,\Big| \geq \delta\Big) \leq 2e^{-2N\delta^2/d^2} \tag{64.184}$$

We can determine the value of δ that ensures $\widehat{\mathcal{R}}_N(\mathcal{Q})$ is $\delta-$close to its mean $\mathcal{R}_N(\mathcal{Q})$ with probability $1-\epsilon$. Then, setting

$$2e^{-2N\delta^2/d^2} \leq \epsilon \tag{64.185}$$

we can solve for δ:

$$\delta \geq d\sqrt{\frac{1}{2N}\ln(2/\epsilon)} \tag{64.186}$$

Substituting into (64.184) we find that with high probability of at least $1-\epsilon$:

$$\mathcal{R}_N(\mathcal{Q}) \leq \widehat{\mathcal{R}}_N(\mathcal{Q}) + d\sqrt{\frac{1}{2N}\ln(2/\epsilon)} \tag{64.187}$$

Using this bound in (64.183) leads to (64.182b).

■

Example 64.9 (**Application to the $0/1-$loss and VC dimension**) Consider N feature vectors $\{h_1,\ldots,h_N\}$ with binary labels $\gamma(n) \in \{\pm 1\}$. Assume we choose $Q(y)$ as the $0/1-$loss defined by

$$Q(y) = \mathbb{I}[y \leq 0] \;=\; \begin{cases} 1, & y \leq 0 \\ 0, & y > 0 \end{cases} \tag{64.188}$$

where, in this example, y is the margin variable defined as $y = \gamma c(h)$. Note that we can relate the classifier and the loss function more explicitly as follows:

$$Q(y) = \frac{1}{2}\Big(1 - \gamma\, c(h)\Big) \tag{64.189}$$

In this way, we have one loss function $Q(y)$ associated with each binary classifier $c(h)$. It then follows from property (64.164a) that

$$\mathcal{R}_N(\mathcal{Q}) = \frac{1}{2}\mathcal{R}_N(\mathcal{C}) \tag{64.190}$$

In particular, using (64.161), we find that, with high likelihood, the error probability (which is equal to $\mathbb{E}\, Q(\boldsymbol{y})$) for any classifier $c \in \mathcal{C}$ designed under the $0/1-$loss is bounded by

$$\sup_{c\in\mathcal{C}}\left(\mathbb{P}_e - \frac{1}{N}\sum_{n=1}^{N}\mathbb{I}[\gamma(n)\widehat{\gamma}(n)]\right) \leq \sqrt{\frac{2}{N}\mathrm{VC}\,\ln\Big(\frac{Ne}{\mathrm{VC}}\Big)} + \sqrt{\frac{1}{2N}\ln\Big(\frac{1}{\epsilon}\Big)} \tag{64.191}$$

We can group the last two terms by using the easily verifiable algebraic inequality $\sqrt{x}+\sqrt{y} \leq 2\sqrt{x+y}$ for any $x, y \geq 0$. Therefore, we find that

$$\sup_{c\in\mathcal{C}}\left(\mathbb{P}_e - \frac{1}{N}\sum_{n=1}^{N}\mathbb{I}[\gamma(n)\widehat{\gamma}(n)]\right) \leq \sqrt{\frac{8}{N}\left\{\mathrm{VC}\,\ln\Big(\frac{Ne}{\mathrm{VC}}\Big) + \frac{1}{4}\ln\Big(\frac{1}{\epsilon}\Big)\right\}} \tag{64.192}$$

A similar argument using the two-sided inequality (64.197a) would lead to

$$\sup_{c\in\mathcal{C}}\left|\mathbb{P}_e - \frac{1}{N}\sum_{n=1}^{N}\mathbb{I}[\gamma(n)\widehat{\gamma}(n)]\right| \leq \sqrt{\frac{8}{N}\left\{\mathrm{VC}\,\ln\Big(\frac{Ne}{\mathrm{VC}}\Big) + \frac{1}{4}\ln\Big(\frac{2}{\epsilon}\Big)\right\}} \tag{64.193}$$

where the right-most term resembles the form we encountered earlier in (64.13) but provides a tighter bound. We further remark that the quantities appearing on the left-hand side play a role similar to the risks defined in the body of the chapter:

$$R(c) = \mathbb{P}_e, \quad R_{\mathrm{emp}}(c) = \frac{1}{N}\sum_{n=1}^{N}\mathbb{I}[\gamma(n)\widehat{\gamma}(n)] \tag{64.194}$$

■

Example 64.10 (**Application to the hinge loss**) Consider next the hinge loss function

$$Q(w;\gamma,h) = \max\{0, 1-\gamma\widehat{\gamma}\}, \;\; \widehat{\gamma} = h^{\mathsf{T}}w \tag{64.195}$$

and the class of prediction functions used to generate $\widehat{\gamma}$ with vectors chosen from the set $\mathcal{W} = \{w \,|\, \|w\|_2 \leq 1\}$. Assume the feature data lies within $\|h\|_2 \leq R$. With $\gamma \in \{\pm 1\}$ fixed, the loss function $Q(\gamma, \widehat{\gamma})$ is seen to be $1-$Lipschitz with respect to the argument $\widehat{\gamma}$. Thus, using (64.182a) and the result of Probs. 64.34 and 64.35 we conclude that

$$\mathbb{E}_{\,\boldsymbol{\gamma}\boldsymbol{h}}Q(w;\boldsymbol{\gamma},\boldsymbol{h}) \leq \frac{1}{N}\sum_{n=1}^{N}\max\Big\{0, 1-\gamma(n)\widehat{\gamma}(n)\Big\} + \frac{2\delta R}{\sqrt{N}} + d\,\sqrt{\frac{1}{2N}\ln\Big(\frac{1}{\epsilon}\Big)} \tag{64.196}$$

Additional examples are given in Bartlett and Mendelson (2002, Sec. 4), Boucheron, Bousquet, and Lugosi (2005, Sec. 4), and Shalev-Shwartz and Ben-David (2014, Ch. 26).

Similar arguments can be repeated to establish two-sided versions of the generalization bounds listed before. We leave the details to Prob. 64.36.

Two-sided generalization bounds (Koltchinskii and Panchenko (2000,2002), Bartlett and Mendelson (2002)). *Consider a set $\mathcal{Q} \in \mathfrak{Q}$ of loss functions with each $Q(y) : \mathbb{R} \to [a, b]$. Let $d = b - a$. Then, with probability of at least $1 - \epsilon$, either of the following bounds holds in terms of the empirical or regular Rademacher complexity:*

$$\sup_{q \in \mathcal{Q}} \left| \mathbb{E}_{\boldsymbol{y}} Q(\boldsymbol{y}) - \frac{1}{N} \sum_{n=1}^{N} Q(y_n) \right| \leq 2\, \mathcal{R}_N(\mathfrak{Q}) + d \sqrt{\frac{1}{2N} \ln(2/\epsilon)} \tag{64.197a}$$

$$\sup_{q \in \mathcal{Q}} \left| \mathbb{E}_{\boldsymbol{y}} Q(\boldsymbol{y}) - \frac{1}{N} \sum_{n=1}^{N} Q(y_n) \right| \leq 2\, \widehat{\mathcal{R}}_N(\mathfrak{Q}) + 3d \sqrt{\frac{1}{2N} \ln(4/\epsilon)} \tag{64.197b}$$

REFERENCES

Abu-Mostafa, Y. S., M. Magdon-Ismail, and H.-T. Lin (2012), *Learning from Data*, AMLBook.com.

Alon, N., S. Ben-David, N. Cesa-Bianchi, and D. Haussler (1997), "Scale-sensitive dimensions, uniform convergence, and learnablity," *Journal of the ACM*, vol. 44, no. 4, pp. 615–631.

Antos, A., B. Kégl, T. Linder, and G. Lugosi (2002), "Data-dependent margin-based generalization bounds for classification," *J. Machine Learning Research*, vol. 3, pp. 73–98.

Bartlett, P. L., S. Boucheron, and G. Lugosi (2001), "Model selection and error estimation," *Machine Learning*, vol. 48, pp. 85–113.

Bartlett, P., O. Bousquet, and S. Mendelson (2005), "Local Rademacher complexities," *Annals of Statistics*, vol. 33, no. 4, pp. 1497–1537.

Bartlett, P. L., M. I. Jordan, and J. D. McAuliffe (2006), "Convexity, classification, and risk functions," *J. American Statistical Association*, vol. 101, no. 473, pp. 138–156.

Bartlett, P. L. and S. Mendelson (2002), "Rademacher and Gaussian complexities: Risk bounds and structural results," *J. Machine Learning Research*, vol. 3, pp. 463–482.

Bellman, R. E. (1957a), *Dynamic Programming*, Princeton University Press. Also published in 2003 by Dover Publications.

Blumer, A., A. Ehrenfeucht, D. Haussler, and M. K. Warmuth (1989), "Learnability and the Vapnik-Chervonenkis dimension," *Journal of the ACM*, vol. 36, no. 4, pp. 929–965.

Boucheron, S., O. Bousquet, and G. Lugosi (2005), "Theory of classification: A survey of recent advances," *ESAIM: Probability and Statistics*, vol. 9, pp. 323–375.

Breiman, L. (1994), "Heuristics of instability in model selection," *Annals of Statistics*, vol. 24, no. 6, pp. 2350–2383.

Breiman, L. (1996a), "Stacked regressions," *Machine Learning*, vol. 24, no. 1, pp. 41—64.

Breiman, L. (1996b), "Bagging predictors," *Machine Learning*, vol. 24, no. 2, pp. 123–140.

Cantelli, F. P. (1933), "Sulla determinazione empirica delle leggi di probabilita," *Giorn. Ist. Ital. Attuari*, vol. 4, pp. 221–424.

Cherkassky, V. and F. M. Mulier (2007), *Learning from Data: Concepts, Theory, and Methods*, 2nd edition, Wiley, NY

Chernoff, H. (1952), "A measure of asymptotic efficiency of tests of a hypothesis based on the sum of observations, " *Annals of Mathematical Statistics*, vol. 23, pp. 493–507.
Cover, T. M. (1968), "Estimation by the nearest neighbor rule," *IEEE Trans. Information Theory*, vol. 14, pp. 21–27.
Cucker, F. and S. Smale (2002), "On the mathematical foundation of learning," *Bull. Amer. Math. Soc.*, vol. 39, pp. 1–49.
Devroye, L. (1982), "Necessary and sufficient conditions for the almost everywhere convergence of nearest neighbor regression function estimates," *Z. Wahrscheinlichkeitstheorie verw. Gebiete*, vol. 61, pp. 467–481.
Devroye, L., L. Gyorfi, and G. Lugosi (1996), *A Probabilistic Theory of Pattern Recognition*, Springer, NY.
Domingos, P. (2000), "A unified bias-variance decomposition," *Proc. Intern. Conf. Machine Learning* (ICML), pp. 231–238, Stanford, CA.
Dudley, R. M. (1978), "Central limit theorems for empirical measures," *Annals of Probability*, vol. 6, pp. 899–929.
Dudley, R. M. (1999), *Uniform Central Limit Theorems*, Cambridge University Press.
Dudley, R. M., E. Gine, and J. Zinn (1991), "Uniform and universal Glivenko-Cantelli classes," *Journal of Theoretical Probability*, vol. 4, no. 3, pp. 485–510.
Friedman, J. H. (1997), "On bias, variance, 0/1 loss, and the curse-of-dimensionality," *Data Mining and Knowledge Discovery*, vol. 1, pp. 55–77.
Fukunaga, K. (1990), *Introduction to Statistical Pattern Recognition*, 2nd edition, Academic Press, NY.
German, S., E. Bienenstock and R. Doursat (1992), "Neural networks and the bias variance dilemma," *Neural Computation*, vol. 4, pp. 1–58.
Geurts, P. (2005), "Bias vs variance decomposition for regression and classification," in *Data Mining and Knowledge Discovery Handbook*, O. Maimon and L. Rokach, *Eds.*, pp. 749-763, Springer, NY.
Glivenko, V. (1933), "Sulla determinazione empirica della legge di probabilita," *Giorn. Ist. Ital. Attuari*, vol. 4, pp. 92–99.
Hastie, T., R. Tibshirani, and J. Friedman (2009), *The Elements of Statistical Learning*, 2nd edition, Springer, NY.
Hoeffding, W. (1963), "Probability inequalities for sums of bounded random variables," *Journal Amer. Stat. Assoc.*, vol. 58, pp. 13–30.
Hughes, G. F. (1968), "On the mean accuracy of statistical pattern recognizers," *IEEE Trans. Information Theory*, vol. 14, no. 1, pp. 55–63.
James, G. (2003), "Variance and bias for general loss functions," *Machine Learning*, vol. 51, pp. 115–135.
James, G. and T. Hastie (1997), "Generalizations of the bias/variance decomposition for prediction error," *Technical Report*, Department of Statistics, Stanford University, Stanford, CA.
Kahane, J.-P. (1964), "Sur les sommes vectorielles $\sum \pm u_n$, *Comptes Rendus Hebdomadaires des S'eances de l'Académie des Sciences*, Paris, vol. 259, pp. 2577–2580.
Kakade, S., K. Sridharan, and A. Tewari (2008), "On the complexity of linear prediction: Risk bounds, margin bounds, and regularization," *Proc. Neural Information Process. Systems* (NIPS), pp. 1–11, Vancouver, Canada.
Kearns, M. and U. Vazirani (1994), *An Introduction to Computational Learning Theory*, MIT Press, Cambridge, MA.
Khintchine, A. (1923), "Über dyadische brüche," *Mathematische Zeitschrift*, vol. 18, no. 1, pp. 109–116.
Kohavi, R. and D. H. Wolpert (1996), "Bias plus variance decomposition for zero-one loss functions," *Proc. Intern. Conf. Machine Learning* (ICML), pp. 275–283, Tahoe City, CA.
Koltchinskii, V. (2001), "Rademacher penalties and structural risk minimization," *IEEE Trans. Inform. Theory*, vol. 47, pp. 1902–1914.
Koltchinskii, V. and D. Panchenko (2000), "Rademacher processes and bounding the

risk of function learning," in *High Dimensional Probability II*, pp. 443–457, Springer, NY.

Koltchinskii, V. and D. Panchenko (2002), "Empirical margin distributions and bounding the generalization error of combined classifiers," *Annals of Statistics*, vol. 30, no. 1, pp. 1–50.

Kong, E. B. and T. G. Dietterich (1995), "Error-correcting output coding corrects bias and variance," *Proc. Intern. Conf. Machine Learning* (ICML), pp. 313-321, Tahoe City, CA.

Kulkarni, S., G. Lugosi, and S. Venkatesh (1998),"Learning pattern classification – A survey," *IEEE Trans. Information Theory*, vol. 44, no. 6, pp. 2178–2206.

Latala, R. and K. Oleszkiewicz (1994), "On the best constant in the khintchinekahane inequality," *Studia Math*, vol. 109, no. 1, pp. 101–104.

Lewin, K. (1945), "The research center for group dynamics at MIT," *Sociometry*, vol. 8, pp. 126–135. See also page 169 of Lewin, K. (1952), *Field Theory in Social Science: Selected Theoretical Papers by Kurt Lewin*, Tavistock, London.

Massart, P. (2000), "Some applications of concentration inequalities to statistics," *Annales de la Faculté des Sciences de Toulouse*, vol. 9, no. 2, pp. 245–303.

Massart, P. (2007), *Concentration Inequalities and Model Selection,* vol. 1896, Lecture Notes in Mathematics.

McDiarmid, C. (1989), "On the method of bounded differences," pp. 148–188, in *Surveys in Combinatorics*, J. Siemons, *Ed.*, Cambridge University Press.

Mendelson, S. (2002), "Improving the sample complexity using global data," *IEEE Trans. Inform. Theory*, vol. 48, pp. 1977–1991.

Mohri, M., A. Rostamizadeh, and A. Talwalkar (2018), *Foundations of Machine Learning*, 2nd edition, MIT Press.

Okamoto, M. (1958), "Some inequalities relating to the partial sum of binomial probabilities," *Annals Inst. Stat. Math.*, vol. 10, pp. 29–35.

Pollard, D. (1984), *Convergence of Stochastic Processes*, Springer-Verlag, NY.

Radon, J. (1921), "Mengen konvexer Körper, die einen gemeinsamen Punkt enthalten," *Mathematische Annalen*, vol. 83, no. 1–2, pp. 113–115.

Rosasco, L., E. De Vito, A. Caponnetto, M. Piana, and A. Verri (2004), "Are loss functions all the same?" *Neural Comput.*, vol. 16, no. 5, pp. 1063–1076.

Sauer, N. (1972), "On the density of families of sets," *Journal of Combinatorial Theory Series A*, vol. 13, pp. 145–147.

Schaffer, C. (1994), "A conservation law for generalization performance," in *Proc. Intern. Conf. Machine Learning* (ICML), pp. 259–265, New Brunswick, NJ.

Shalev-Shwartz, S. and S. Ben-David (2014), *Understanding Machine Learning: From Theory to Algorithms*, Cambridge University Press.

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

Stone, C. (1977), "Consistent nonparametric regression," *Annals of Statistics*, vol. 5, pp. 595–645.

Tibshirani, R. (1996a), "Bias, variance, and prediction error for classification rules," *Technical Report*, Department of Preventive Medicine and Biostatistics and Department of Statistics, University of Toronto, Toronto, Canada.

Tomczak-Jaegermann, N. (1989), *Banach-Mazur Distance and Finite-Dimensional Operator Ideals*, Pitman Monographs and Surveys in Pure and Applied Mathematics, no. 38, Pitman, London.

Valiant, L. (1984), "A theory of the learnable," *Communications of the ACM*, vol. 27, pp. 1134–1142.

van der Vaart, A. W. and J. A. Wellner (1996), *Glivenko-Cantelli Theorems*, Springer, NY.

Vapnik, V. N. (1995), *The Nature of Statistical Learning Theory*, Springer, NY.

Vapnik, V. N. (1998), *Statistical Learning Theory*, Wiley, NY.

Vapnik, V. N. (1999), "An overview of statistical learning theory," *IEEE Trans. Neural Netw.*, vol. 10, pp. 988–999.

Vapnik, V. N. and A. Y. Chervonenkis (1968), "On the uniform convergence of relative frequencies of events to their probabilities," *Doklady Akademii Nauk. SSSR*, in Russian, vol. 181, no. 4, pp. 781–783.

Vapnik, V. N. and A. Y. Chervonenkis (1971), "On the uniform convergence of relative frequencies of events to their probabilities," *Theory of Probability and its Applications*, vol. 16, no. 2, pp. 264–280.

Vidyasagar, M. (1997), *A Theory of Learning and Generalization*, Springer, NY.

Wainwright, M. J. (2019), *High-Dimensional Statistics: A Non-Asymptotic Viewpoint*, Cambridge University Press.

Wolff, T. H. (2003), *Lectures on Harmonic Analysis*, vol. 29, American Mathematical Society.

Wolpert, D. H. (1992), "On the connection between in-sample testing and generalization error," *Complex Systems*, vol. 6, pp. 47–94.

Wolpert, D. H. (1996), "The lack of a priori distinctions between learning algorithms," *Neural Computation*, vol. 8, no. 7, pp. 1341–1390.

Wolpert, D. H. and W. G. Macready (1997), "No free lunch theorems for optimization," *IEEE Trans. Evolutionary Computation*, vol. 1, no. 1, pp. 67–82.