# ML UNIT 1 PART B Material

Computer organisation (Jawaharlal Nehru Technological University, Kakinada)

**Dr K N MADHAVI LATHA**
ASSOCIATE PROFESSOR
DEPT OF CSE
SIRCRRCOE

# MACHINE LEARNING UNIT 1 PART B

**Unit 1 PART B SYLLABUS:**

**Statistical Learning**:
Introduction, Supervised and Unsupervised Learning, Training and Test Loss, Tradeoffs in Statistical Learning, Estimating Risk Statistics, Sampling distribution of an estimator, Empirical Risk Minimization.

## Introduction

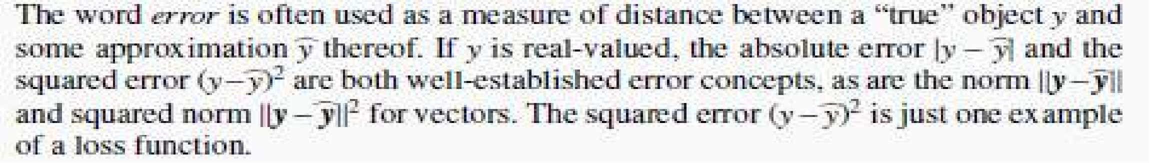
The structuring and visualizing data are important aspects of data science, the main challenge lies in the mathematical analysis of the data. When the goal is to interpret the model and quantify the uncertainty in the data, this analysis is usually referred to as statistical learning. In contrast, when the emphasis is on making predictions using large-scale statistical data, then it is common to speak about machine learning or data mining
There are two major goals for modeling data: 1) to accurately predict some future quantity of interest, given some observed data, and 2) to discover unusual or interesting patterns in the data. To achieve these goals, one must rely on knowledge from three important pillars of the mathematical sciences.

- Function approximation
- Optimization
- Probability and Statistics

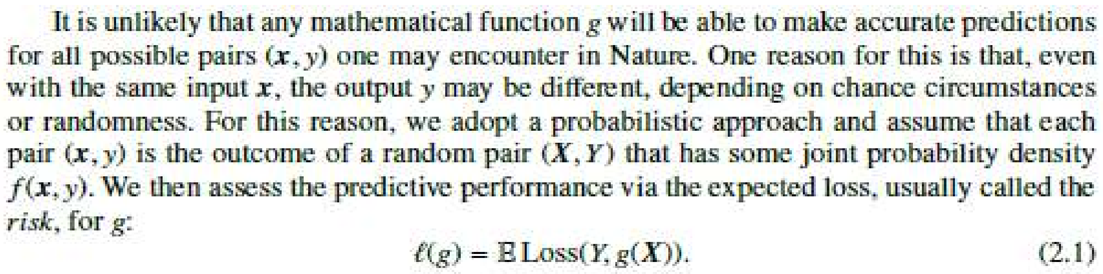
## Supervised and Unsupervised Learning

Given an input or feature vector x, one of the main goals of machine learning is to predict response an output or response variable y. For example, x could be a digitized signature and y a binary variable that indicates whether the signature is genuine or false. Another example is where x represents the weight and smoking habits of an expecting mother and y the birth weight of the baby. The data science attempt at this prediction is encoded in a mathematical prediction function g, called the prediction function function, which takes as an input x and outputs a guess g(x) for y. In a sense, g encompasses all the information about the relationship between the variables x and y, excluding the effects of chance and randomness in nature.
In regression problems, the response variable y can take any real value. In contrast, regression when y can only lie in a finite set, say y 2 f0; : : : ; c □ 1g, then predicting y is conceptually the same as classifying the input x into one of c categories, and so prediction becomes a classification classification problem.
We can measure the accuracy of a prediction by with respect to a given response y by loss function using some loss function Loss(y, y). In a regression setting the usual choice is the squarederror loss (y,y)2. In the case of classification, the zero–one (also written 0–

1) loss function Loss(y,y) = 1fy , byg is often used, which incurs a loss of 1 whenever the predicted class
by is not equal to the class y. Later on in this book, we will encounter various other useful loss functions, such as the cross-entropy and hinge loss functions.

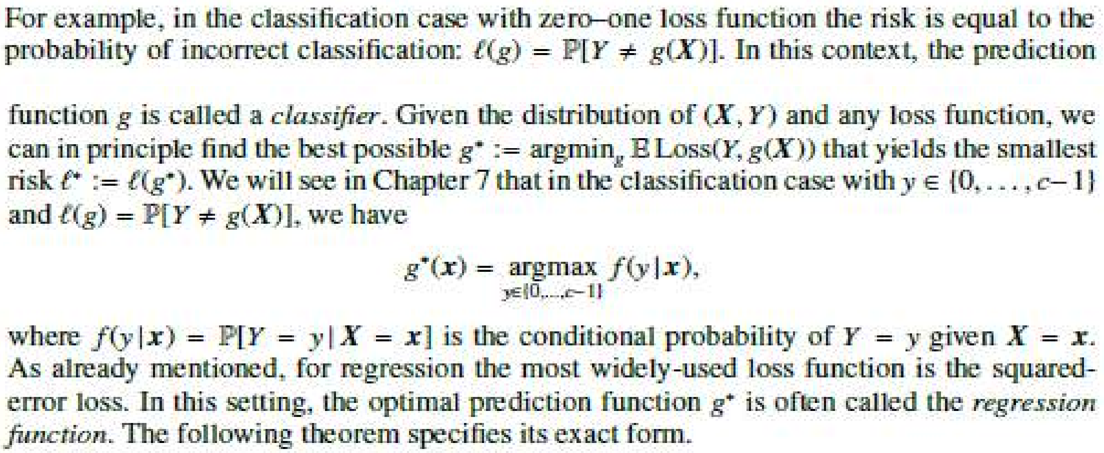


The word *error* is often used as a measure of distance between a "true" object $y$ and some approximation $\widehat{y}$ thereof. If $y$ is real-valued, the absolute error $|y - \widehat{y}|$ and the squared error $(y-\widehat{y})^2$ are both well-established error concepts, as are the norm $\|\boldsymbol{y}-\widehat{\boldsymbol{y}}\|$ and squared norm $\|\boldsymbol{y} - \widehat{\boldsymbol{y}}\|^2$ for vectors. The squared error $(y-\widehat{y})^2$ is just one example of a loss function.



It is unlikely that any mathematical function $g$ will be able to make accurate predictions for all possible pairs $(\boldsymbol{x}, y)$ one may encounter in Nature. One reason for this is that, even with the same input $\boldsymbol{x}$, the output $y$ may be different, depending on chance circumstances or randomness. For this reason, we adopt a probabilistic approach and assume that each pair $(\boldsymbol{x}, y)$ is the outcome of a random pair $(\boldsymbol{X}, Y)$ that has some joint probability density $f(\boldsymbol{x}, y)$. We then assess the predictive performance via the expected loss, usually called the *risk*, for $g$:

$$\ell(g) = \mathbb{E}\,\mathrm{Loss}(Y, g(\boldsymbol{X})). \tag{2.1}$$



For example, in the classification case with zero–one loss function the risk is equal to the probability of incorrect classification: $\ell(g) = \mathbb{P}[Y \neq g(\boldsymbol{X})]$. In this context, the prediction

function $g$ is called a *classifier*. Given the distribution of $(\boldsymbol{X}, Y)$ and any loss function, we can in principle find the best possible $g^* := \operatorname{argmin}_g \mathbb{E}\,\mathrm{Loss}(Y, g(\boldsymbol{X}))$ that yields the smallest risk $\ell^* := \ell(g^*)$. We will see in Chapter 7 that in the classification case with $y \in \{0, \dots, c-1\}$ and $\ell(g) = \mathbb{P}[Y \neq g(\boldsymbol{X})]$, we have

$$g^*(\boldsymbol{x}) = \underset{y \in \{0,\dots,c-1\}}{\operatorname{argmax}} f(y \,|\, \boldsymbol{x}),$$

where $f(y \,|\, \boldsymbol{x}) = \mathbb{P}[Y = y \,|\, \boldsymbol{X} = \boldsymbol{x}]$ is the conditional probability of $Y = y$ given $\boldsymbol{X} = \boldsymbol{x}$. As already mentioned, for regression the most widely-used loss function is the squared-error loss. In this setting, the optimal prediction function $g^*$ is often called the *regression function*. The following theorem specifies its exact form.

**Dr K N MADHAVI LATHA**
ASSOCIATE PROFESSOR
DEPT OF CSE
SIRCRRCOE

# Training and Test Loss:

The prediction accuracy of new pairs of data is measured by the *generalization risk* of the learner. For a *fixed* training set $\tau$ it is defined as

$$\ell(g^{\mathcal{G}}_{\tau}) = \mathbb{E}\,\mathrm{Loss}(Y, g^{\mathcal{G}}_{\tau}(\boldsymbol{X})), \tag{2.5}$$

where $(\boldsymbol{X}, Y)$ is distributed according to $f(\boldsymbol{x}, y)$. In the discrete case the generalization risk is therefore: $\ell(g^{\mathcal{G}}_{\tau}) = \sum_{\boldsymbol{x},y} \mathrm{Loss}(y, g^{\mathcal{G}}_{\tau}(\boldsymbol{x}))f(\boldsymbol{x}, y)$ (replace the sum with an integral for the continuous case). The situation is illustrated in Figure 2.1, where the distribution of $(\boldsymbol{X}, Y)$ is indicated by the red dots. The training set (points in the shaded regions) determines a fixed prediction function shown as a straight line. Three possible outcomes of $(\boldsymbol{X}, Y)$ are shown (black dots). The amount of loss for each point is shown as the length of the dashed lines. The generalization risk is the average loss over all possible pairs $(\boldsymbol{x}, y)$, weighted by the corresponding $f(\boldsymbol{x}, y)$.

Given an arbitrary prediction function $g$, it is typically not possible to compute its risk $\ell(g)$ in (2.1). However, using the training sample $\mathcal{T}$, we can approximate $\ell(g)$ via the empirical (sample average) risk

$$\ell_{\mathcal{T}}(g) = \frac{1}{n}\sum_{i=1}^{n} \mathrm{Loss}(Y_i, g(\boldsymbol{X}_i)), \tag{2.3}$$

which we call the *training loss*. The training loss is thus an unbiased estimator of the risk (the expected loss) for a prediction function $g$, based on the training data.

To approximate the optimal prediction function $g^*$ (the minimizer of the risk $\ell(g)$) we first select a suitable collection of approximating functions $\mathcal{G}$ and then take our *learner* to be the function in $\mathcal{G}$ that minimizes the training loss; that is,

$$g^{\mathcal{G}}_{\mathcal{T}} = \underset{g\in\mathcal{G}}{\operatorname{argmin}}\ \ell_{\mathcal{T}}(g). \tag{2.4}$$

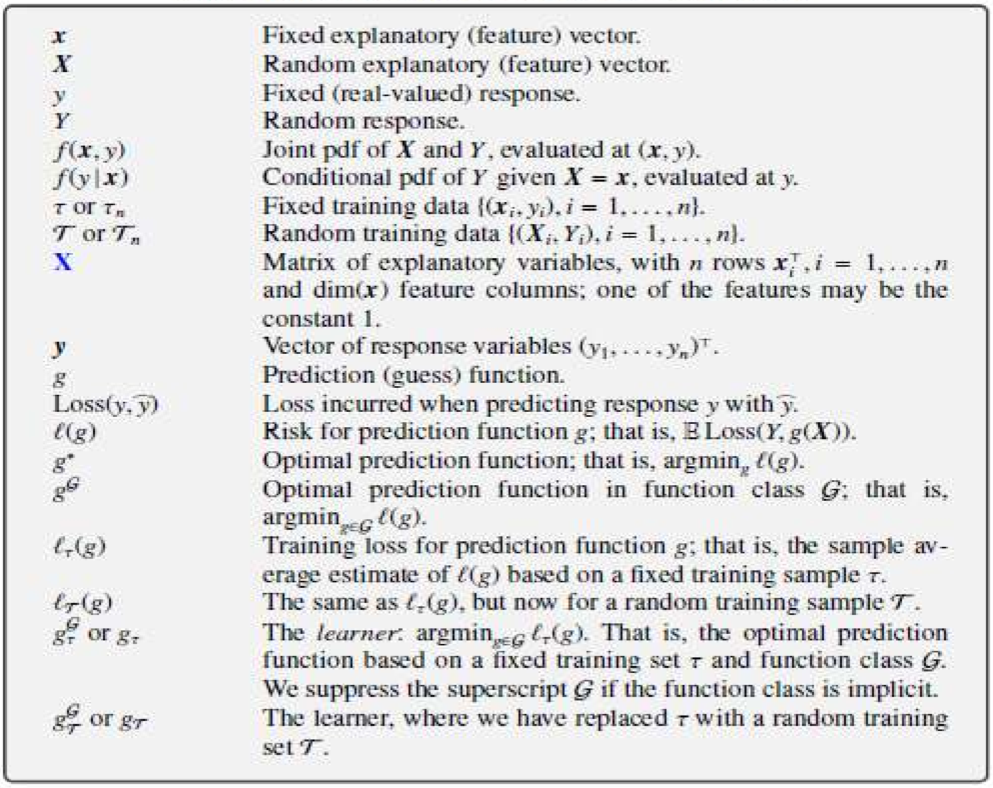
For example, the simplest and most useful $\mathcal{G}$ is the set of *linear* functions of $\boldsymbol{x}$; that is, the set of all functions $g : \boldsymbol{x} \mapsto \boldsymbol{\beta}^{\top}\boldsymbol{x}$ for some real-valued vector $\boldsymbol{\beta}$.

For any outcome $\tau$ of the training data, we can estimate the generalization risk without bias by taking the sample average

$$\ell_{\mathcal{T}'}(g^{\mathcal{G}}_{\tau}) := \frac{1}{n'}\sum_{i=1}^{n'} \mathrm{Loss}(Y'_i, g^{\mathcal{G}}_{\tau}(\boldsymbol{X}'_i)), \tag{2.7}$$

where $\{(\boldsymbol{X}'_1, Y'_1), \ldots, (\boldsymbol{X}'_{n'}, Y'_{n'})\} =: \mathcal{T}'$ is a so-called *test sample*. The test sample is completely separate from $\mathcal{T}$, but is drawn in the same way as $\mathcal{T}$; that is, via independent draws from $f(\boldsymbol{x}, y)$, for some sample size $n'$. We call the estimator (2.7) the *test loss*. For a random training set $\mathcal{T}$ we can define $\ell_{\mathcal{T}'}(g^{\mathcal{G}}_{\mathcal{T}})$ similarly. It is then crucial to assume that $\mathcal{T}$ is independent of $\mathcal{T}'$. Table 2.1 summarizes the main definitions and notation for supervised learning.

Table 2.1: Summary of definitions for supervised learning.



| | |
|---|---|
| $\boldsymbol{x}$ | Fixed explanatory (feature) vector. |
| $\boldsymbol{X}$ | Random explanatory (feature) vector. |
| $y$ | Fixed (real-valued) response. |
| $Y$ | Random response. |
| $f(\boldsymbol{x}, y)$ | Joint pdf of $\boldsymbol{X}$ and $Y$, evaluated at $(\boldsymbol{x}, y)$. |
| $f(y \mid \boldsymbol{x})$ | Conditional pdf of $Y$ given $\boldsymbol{X} = \boldsymbol{x}$, evaluated at $y$. |
| $\tau$ or $\tau_n$ | Fixed training data $\{(\boldsymbol{x}_i, y_i), i = 1, \ldots, n\}$. |
| $\mathcal{T}$ or $\mathcal{T}_n$ | Random training data $\{(\boldsymbol{X}_i, Y_i), i = 1, \ldots, n\}$. |
| $\mathbf{X}$ | Matrix of explanatory variables, with $n$ rows $\boldsymbol{x}_i^\top, i = 1, \ldots, n$ and $\dim(\boldsymbol{x})$ feature columns; one of the features may be the constant 1. |
| $\boldsymbol{y}$ | Vector of response variables $(y_1, \ldots, y_n)^\top$. |
| $g$ | Prediction (guess) function. |
| $\text{Loss}(y, \widehat{y})$ | Loss incurred when predicting response $y$ with $\widehat{y}$. |
| $\ell(g)$ | Risk for prediction function $g$; that is, $\mathbb{E}\,\text{Loss}(Y, g(\boldsymbol{X}))$. |
| $g^*$ | Optimal prediction function; that is, $\text{argmin}_g\, \ell(g)$. |
| $g^{\mathcal{G}}$ | Optimal prediction function in function class $\mathcal{G}$; that is, $\text{argmin}_{g \in \mathcal{G}}\, \ell(g)$. |
| $\ell_\tau(g)$ | Training loss for prediction function $g$; that is, the sample average estimate of $\ell(g)$ based on a fixed training sample $\tau$. |
| $\ell_{\mathcal{T}}(g)$ | The same as $\ell_\tau(g)$, but now for a random training sample $\mathcal{T}$. |
| $g_\tau^{\mathcal{G}}$ or $g_\tau$ | The *learner*: $\text{argmin}_{g \in \mathcal{G}}\, \ell_\tau(g)$. That is, the optimal prediction function based on a fixed training set $\tau$ and function class $\mathcal{G}$. We suppress the superscript $\mathcal{G}$ if the function class is implicit. |
| $g_{\mathcal{T}}^{\mathcal{G}}$ or $g_{\mathcal{T}}$ | The learner, where we have replaced $\tau$ with a random training set $\mathcal{T}$. |

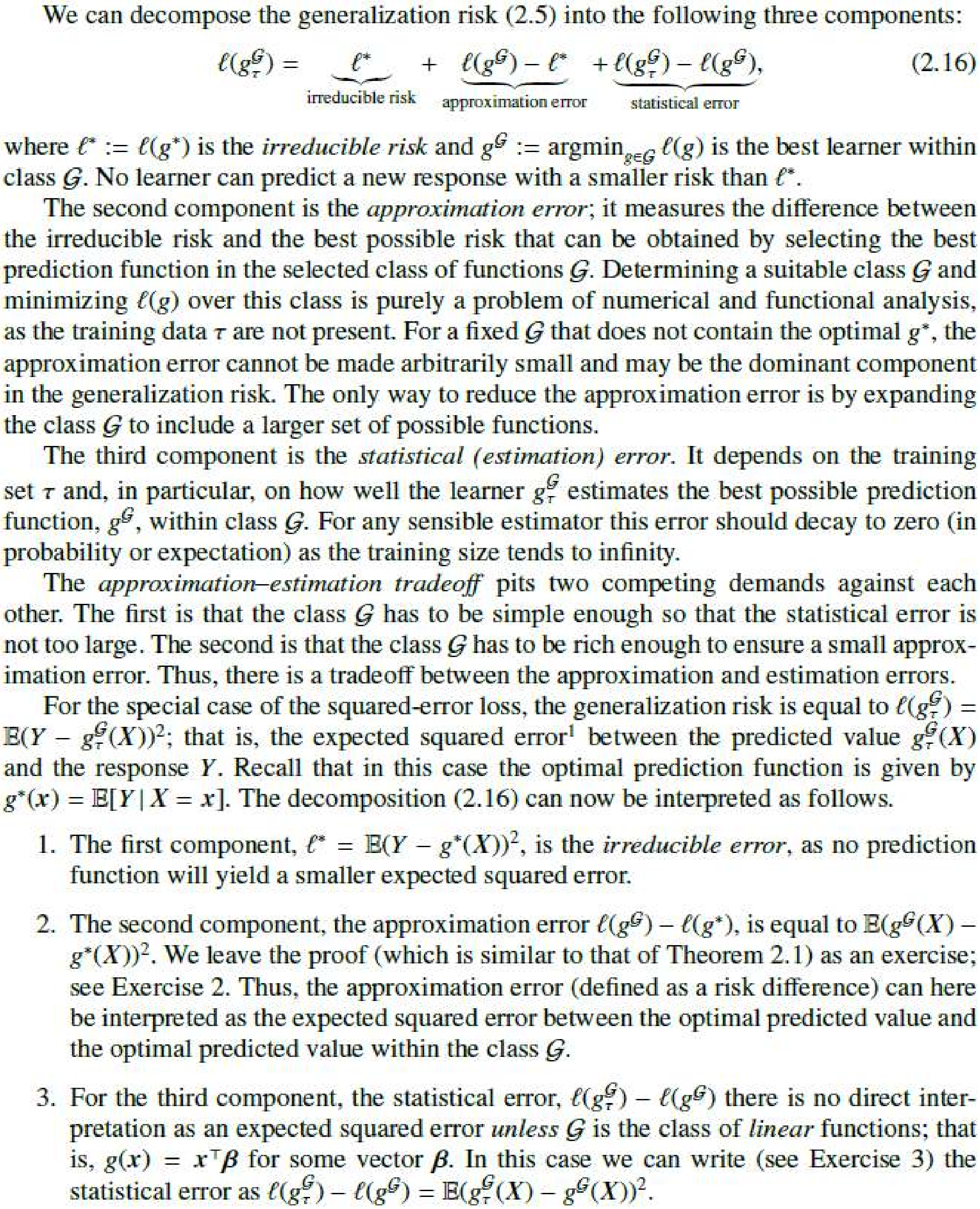
## Tradeoffs in Statistical Learning

The art of machine learning in the supervised case is to make the generalization risk or expected generalization risk as small as possible, while using as few computational resources as possible. In pursuing this goal, a suitable class G of prediction functions has to be chosen. This choice is driven by various factors, such as .

- the complexity of the class (e.g., is it rich enough to adequately approximate, or even contain, the optimal prediction function g_?),
- the ease of training the learner via the optimization program
- how accurately the training loss estimates the risk within class G,
- the feature types (categorical, continuous, etc.)

As a result, the choice of a suitable function class G usually involves a tradeoff between conflicting factors. For example, a learner from a simple class G can be trained very quickly, but may not approximate g_ very well, whereas a learner from a rich class G that contains g_ may require a lot of computing resources to train. To better understand the relation between model complexity, computational simplicity, and estimation accuracy, it is useful to decompose the generalization risk into several parts, so that the tradeoffs between these parts can be studied. We will consider two such decompositions:

**the approximation–estimation tradeoff** and **the bias–variance tradeoff**



We can decompose the generalization risk (2.5) into the following three components:

$$\ell(g_\tau^{\mathcal{G}}) = \underbrace{\ell^*}_{\text{irreducible risk}} + \underbrace{\ell(g^{\mathcal{G}}) - \ell^*}_{\text{approximation error}} + \underbrace{\ell(g_\tau^{\mathcal{G}}) - \ell(g^{\mathcal{G}})}_{\text{statistical error}}, \tag{2.16}$$

where $\ell^* := \ell(g^*)$ is the *irreducible risk* and $g^{\mathcal{G}} := \operatorname{argmin}_{g \in \mathcal{G}} \ell(g)$ is the best learner within class $\mathcal{G}$. No learner can predict a new response with a smaller risk than $\ell^*$.

The second component is the *approximation error*; it measures the difference between the irreducible risk and the best possible risk that can be obtained by selecting the best prediction function in the selected class of functions $\mathcal{G}$. Determining a suitable class $\mathcal{G}$ and minimizing $\ell(g)$ over this class is purely a problem of numerical and functional analysis, as the training data $\tau$ are not present. For a fixed $\mathcal{G}$ that does not contain the optimal $g^*$, the approximation error cannot be made arbitrarily small and may be the dominant component in the generalization risk. The only way to reduce the approximation error is by expanding the class $\mathcal{G}$ to include a larger set of possible functions.

The third component is the *statistical (estimation) error*. It depends on the training set $\tau$ and, in particular, on how well the learner $g_\tau^{\mathcal{G}}$ estimates the best possible prediction function, $g^{\mathcal{G}}$, within class $\mathcal{G}$. For any sensible estimator this error should decay to zero (in probability or expectation) as the training size tends to infinity.

The *approximation–estimation tradeoff* pits two competing demands against each other. The first is that the class $\mathcal{G}$ has to be simple enough so that the statistical error is not too large. The second is that the class $\mathcal{G}$ has to be rich enough to ensure a small approximation error. Thus, there is a tradeoff between the approximation and estimation errors.

For the special case of the squared-error loss, the generalization risk is equal to $\ell(g_\tau^{\mathcal{G}}) = \mathbb{E}(Y - g_\tau^{\mathcal{G}}(X))^2$; that is, the expected squared error[1] between the predicted value $g_\tau^{\mathcal{G}}(X)$ and the response $Y$. Recall that in this case the optimal prediction function is given by $g^*(x) = \mathbb{E}[Y \mid X = x]$. The decomposition (2.16) can now be interpreted as follows.

1. The first component, $\ell^* = \mathbb{E}(Y - g^*(X))^2$, is the *irreducible error*, as no prediction function will yield a smaller expected squared error.

2. The second component, the approximation error $\ell(g^{\mathcal{G}}) - \ell(g^*)$, is equal to $\mathbb{E}(g^{\mathcal{G}}(X) - g^*(X))^2$. We leave the proof (which is similar to that of Theorem 2.1) as an exercise; see Exercise 2. Thus, the approximation error (defined as a risk difference) can here be interpreted as the expected squared error between the optimal predicted value and the optimal predicted value within the class $\mathcal{G}$.

3. For the third component, the statistical error, $\ell(g_\tau^{\mathcal{G}}) - \ell(g^{\mathcal{G}})$ there is no direct interpretation as an expected squared error *unless* $\mathcal{G}$ is the class of *linear* functions; that is, $g(x) = x^\top \boldsymbol{\beta}$ for some vector $\boldsymbol{\beta}$. In this case we can write (see Exercise 3) the statistical error as $\ell(g_\tau^{\mathcal{G}}) - \ell(g^{\mathcal{G}}) = \mathbb{E}(g_\tau^{\mathcal{G}}(X) - g^{\mathcal{G}}(X))^2$.

Thus, when using a squared-error loss, the generalization risk for a linear class $\mathcal{G}$ can be decomposed as:

$$\ell(g_\tau^{\mathcal{G}}) = \mathbb{E}(g_\tau^{\mathcal{G}}(X) - Y)^2 = \ell^* + \underbrace{\mathbb{E}(g^{\mathcal{G}}(X) - g^*(X))^2}_{\text{approximation error}} + \underbrace{\mathbb{E}(g_\tau^{\mathcal{G}}(X) - g^{\mathcal{G}}(X))^2}_{\text{statistical error}}. \tag{2.17}$$

Note that in this decomposition the statistical error is the only term that depends on the training set.

Using again a squared-error loss, a second decomposition (for general $\mathcal{G}$) starts from

$$\ell(g_\tau^{\mathcal{G}}) = \ell^* + \ell(g_\tau^{\mathcal{G}}) - \ell(g^*),$$

where the statistical error and approximation error are combined. Using similar reasoning as in the proof of Theorem 2.1, we have

$$\ell(g_\tau^{\mathcal{G}}) = \mathbb{E}(g_\tau^{\mathcal{G}}(X) - Y)^2 = \ell^* + \mathbb{E}\left(g_\tau^{\mathcal{G}}(X) - g^*(X)\right)^2 = \ell^* + \mathbb{E}D^2(X, \tau),$$

where $D(x, \tau) := g_\tau^{\mathcal{G}}(x) - g^*(x)$. Now consider the random variable $D(x, \mathcal{T})$ for a random training set $\mathcal{T}$. The expectation of its square is:

$$\begin{aligned} \mathbb{E}\left(g_{\mathcal{T}}^{\mathcal{G}}(x) - g^*(x)\right)^2 &= \mathbb{E}D^2(x, \mathcal{T}) = (\mathbb{E}D(x, \mathcal{T}))^2 + \mathbb{V}\mathrm{ar}\, D(x, \mathcal{T}) \\ &= \underbrace{(\mathbb{E}g_{\mathcal{T}}^{\mathcal{G}}(x) - g^*(x))^2}_{\text{pointwise squared bias}} + \underbrace{\mathbb{V}\mathrm{ar}\, g_{\mathcal{T}}^{\mathcal{G}}(x)}_{\text{pointwise variance}}. \end{aligned} \tag{2.21}$$

If we view the learner $g_{\mathcal{T}}^{\mathcal{G}}(x)$ as a function of a random training set, then the *pointwise squared bias* term is a measure for how close $g_{\mathcal{T}}^{\mathcal{G}}(x)$ is on average to the true $g^*(x)$, whereas the *pointwise variance* term measures the deviation of $g_{\mathcal{T}}^{\mathcal{G}}(x)$ from its expected value $\mathbb{E}g_{\mathcal{T}}^{\mathcal{G}}(x)$. The squared bias can be reduced by making the class of functions $\mathcal{G}$ more complex. However, decreasing the bias by increasing the complexity often leads to an increase in the variance term. We are thus seeking learners that provide an optimal balance between the bias and variance, as expressed via a minimal generalization risk. This is called the *bias–variance tradeoff*.

Note that the *expected* generalization risk (2.6) can be written as $\ell^* + \mathbb{E}D^2(X, \mathcal{T})$, where $X$ and $\mathcal{T}$ are independent. It therefore decomposes as

$$\mathbb{E}\,\ell(g_{\mathcal{T}}^{\mathcal{G}}) = \ell^* + \underbrace{\mathbb{E}\,(\mathbb{E}[g_{\mathcal{T}}^{\mathcal{G}}(X) \mid X] - g^*(X))^2}_{\text{expected squared bias}} + \underbrace{\mathbb{E}[\mathbb{V}\mathrm{ar}[g_{\mathcal{T}}^{\mathcal{G}}(X) \mid X]]}_{\text{expected variance}}. \tag{2.22}$$

**Dr K N MADHAVI LATHA**
ASSOCIATE PROFESSOR
DEPT OF CSE
SIRCRRCOE

# Estimating Risk Statistics

The most straightforward way to quantify the generalization risk (2.5) is to estimate it via the test loss (2.7). However, the generalization risk depends inherently on the training set, and so different training sets may yield significantly different estimates. Moreover, when there is a limited amount of data available, reserving a substantial proportion of the data for testing rather than training may be uneconomical. In this section we consider different methods for estimating risk measures which aim to circumvent these difficulties.

## 2.5.1 In-Sample Risk

We mentioned that, due to the phenomenon of overfitting, the training loss of the learner, $\ell_\tau(g_\tau)$ (for simplicity, here we omit $\mathcal{G}$ from $g_\tau^{\mathcal{G}}$), is not a good estimate of the generalization risk $\ell(g_\tau)$ of the learner. One reason for this is that we use the same data for both training the model and assessing its risk. How should we then estimate the generalization risk or expected generalization risk?

To simplify the analysis, suppose that we wish to estimate the average accuracy of the predictions of the learner $g_\tau$ at the $n$ feature vectors $x_1, \ldots, x_n$ (these are part of the training set $\tau$). In other words, we wish to estimate the *in-sample risk* of the learner $g_\tau$:

$$\ell_{\text{in}}(g_\tau) = \frac{1}{n} \sum_{i=1}^{n} \mathbb{E}\,\text{Loss}(Y_i', g_\tau(x_i)), \tag{2.23}$$

where each response $Y_i'$ is drawn from $f(y \mid x_i)$, independently. Even in this simplified setting, the training loss of the learner will be a poor estimate of the in-sample risk. Instead, the

## 2.5.2 Cross-Validation

In general, for complex function classes $\mathcal{G}$, it is very difficult to derive simple formulas of the approximation and statistical errors, let alone for the generalization risk or expected generalization risk. As we saw, when there is an abundance of data, the easiest way to assess the generalization risk for a given training set $\tau$ is to obtain a test set $\tau'$ and evaluate the test loss (2.7). When a sufficiently large test set is not available but computational resources are cheap, one can instead gain direct knowledge of the expected generalization risk via a computationally intensive method called *cross-validation*.

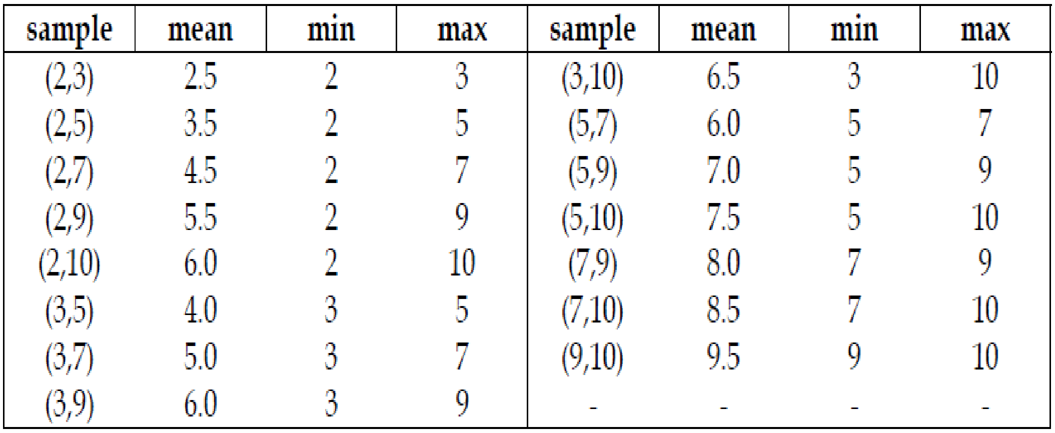
## Sampling distribution of an estimator

The "sampling distribution" of a statistic (estimator) is a probability distribution that describes the probabilities with which the possible values for a specific statistic (estimator) occur. The exact form of the sampling distribution for a given estimator will depend on the underlying population distribution from which the data were drawn. In general, knowing an estimator's sampling distribution is both useful and often necessary for constructing confidence intervals and hypothesis tests based on that estimator in order to draw inference about its corresponding population parameter. In constructing those entities, a statistician will be often interested in determining the probability that the distance between an estimator and the true parameter it seeks to estimate is smaller than a certain amount.

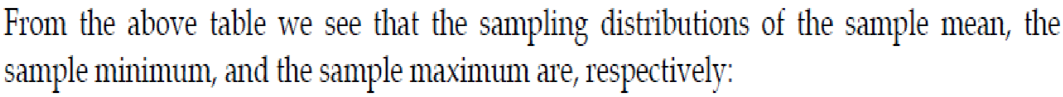
Example of sampling distributions, suppose that we have a population consisting of the values {2,3,5,7,9,10}. Suppose further that we want to take a sample of size *n*=2, "without replacement," from this population; sampling "without replacement" means that each unit in the population can appear only once in a sample.

$$\text{There are } \binom{6}{2} = \frac{6!}{2!\,(6-2)!} = 15$$

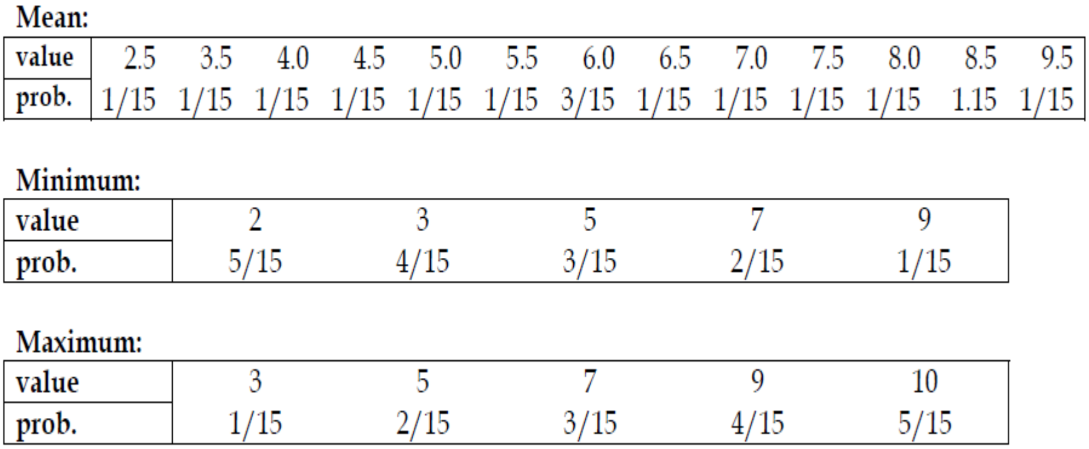
different samples of size *n* = 2 that can be taken without replacement. If we calculate the sample mean, minimum, and maximum for each of the 15 possible samples, we get the following results:



| sample | mean | min | max | sample | mean | min | max |
|---|---|---|---|---|---|---|---|
| (2,3) | 2.5 | 2 | 3 | (3,10) | 6.5 | 3 | 10 |
| (2,5) | 3.5 | 2 | 5 | (5,7) | 6.0 | 5 | 7 |
| (2,7) | 4.5 | 2 | 7 | (5,9) | 7.0 | 5 | 9 |
| (2,9) | 5.5 | 2 | 9 | (5,10) | 7.5 | 5 | 10 |
| (2,10) | 6.0 | 2 | 10 | (7,9) | 8.0 | 7 | 9 |
| (3,5) | 4.0 | 3 | 5 | (7,10) | 8.5 | 7 | 10 |
| (3,7) | 5.0 | 3 | 7 | (9,10) | 9.5 | 9 | 10 |
| (3,9) | 6.0 | 3 | 9 | - | - | - | - |



From the above table we see that the sampling distributions of the sample mean, the sample minimum, and the sample maximum are, respectively:



**Mean:**

| value | 2.5 | 3.5 | 4.0 | 4.5 | 5.0 | 5.5 | 6.0 | 6.5 | 7.0 | 7.5 | 8.0 | 8.5 | 9.5 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| prob. | 1/15 | 1/15 | 1/15 | 1/15 | 1/15 | 1/15 | 3/15 | 1/15 | 1/15 | 1/15 | 1/15 | 1.15 | 1/15 |

**Minimum:**

| value | 2 | 3 | 5 | 7 | 9 |
|---|---|---|---|---|---|
| prob. | 5/15 | 4/15 | 3/15 | 2/15 | 1/15 |

**Maximum:**

| value | 3 | 5 | 7 | 9 | 10 |
|---|---|---|---|---|---|
| prob. | 1/15 | 2/15 | 3/15 | 4/15 | 5/15 |

## Empirical Risk Minimization

Empirical risk minimization (ERM) is a principle in statistical learning theory which defines a family of learning algorithms and is used to give theoretical bounds on their performance. The core idea is that we cannot know exactly how well an algorithm will work in practice (the true "risk") because we don't know the true distribution of data that the algorithm will work on, but we can instead measure its performance on a known set of training data (the "empirical" risk).