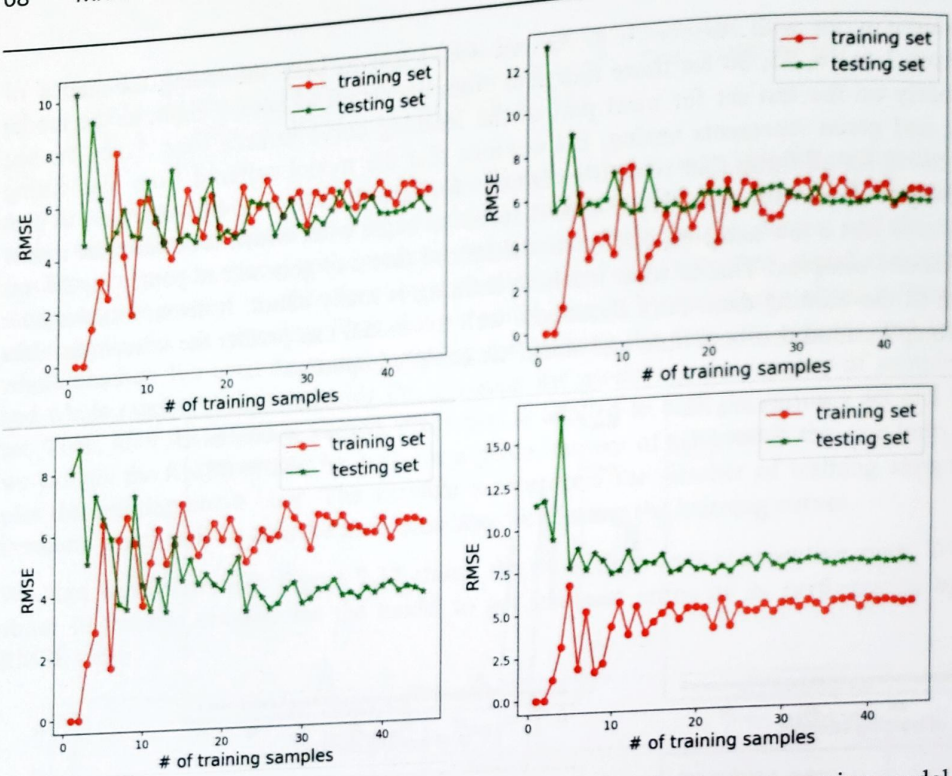

Figure 2.19 Learning curves from four consecutive runs for the linear regression model with `degree=1`.

2.8 BIAS-VARIANCE TRADE-OFFS

To some extent, machine learning is about dealing with *errors* – the errors between predicted outcome and actually observed. Therefore, it is always important to understand errors, including where errors come from, and how we can minimize errors. In this regard, there is a well-known theoretical research result that quantitatively describes the components of machine learning errors, which is known as *bias-variance trade-off*. In this section, we describe both the theoretical formulation and some numerical simulation that reinforce each other to help us understand bias-variance trade-off more precisely.

2.8.1 Theoretical formulation

Before starting discussing on how machine learning errors can be quantified theoretically, let's find a common ground for qualifying several basic elements of machine learning. If you consider all machine learning examples we have experimented with, you might realize that the basic elements of machine learning can be summarized using a pie chart as shown in Figure 2.20. This chart shows all three indispensable ingredients for a machine learning problem: model, feature set (or attributes), and sample set (or volume of data). We can call this an *M-F-S* combination, with each letter representing a corresponding element. The feature set (*F*) element helps us understand how a distribution is sampled, dense or sparse, measured by the number of features sampled from the distribution or how many times

the distribution is sampled independently in one round of sampling. The sample set (S) element describes how many samples are obtained or available. If we describe these two elements together in a matrix, F would represent the columns, while S would represent rows.

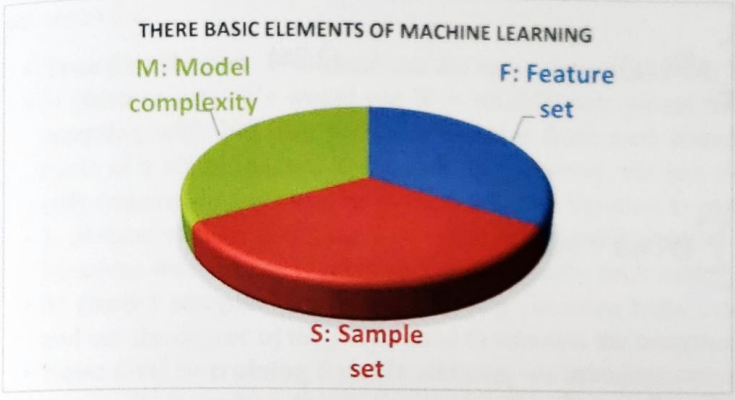

Figure 2.20 Three basic elements of machine learning.

Finally, after we have data samples, we need to use a machine learning model to solve the machine learning problem. This represents the third element, the model (M), which is quantified by its complexity. We already learnt that simpler models tend to under-fit while more complex models tend to over-fit, both of which are major sources of errors for machine learning problems.

The above three basic elements for machine learning play their roles in generating errors governed by the following formula:

$$Total\ errors = (bias)^2 + variance + (irreducible\ error) \qquad (2.25)$$

Here, the third term of the irreducible error comes from the noisiness of the data itself, which cannot be reduced once samples have been gathered. The way to improve irreducible error is to take some measures at the source of data collection by any means, for example, improving the resolution of physical devices such as image sensors *actively*, or removing outliers in collected data *passively*. As machine learning scientists or engineers, what we can do is to minimize the errors from the first and second terms in Eq. (2.25), which are known as the *bias squared* and the *variance*, respectively. Next, we discuss how to define these two terms mathematically.

For convenience, let us use letters to represent the sources of errors as follows:

1. **N (feature set)**: Let us assume that we have N features for a theoretical distribution, such as a sinusoidal wave distribution. We can describe this as a row of samples expressed as (x_1, x_2, ..., x_N).
2. **L (sample set):** Let us assume that we have L sample sets, each of which is a row of (x_1, x_2, ..., x_N). This way, we can describe our data as a 2D matrix of $L \times N$.
3. **M (model complexity):** Let us assume that our model complexity is m, and we only go up to the complexity of M as the upper bound. Examples of model complexity may include the polynomial degree with polynomial fitting, regularization parameters such as α and ρ as we discussed with the Ridge, LASSO and Elastic Net regularization techniques previously, and so on.

Now we are ready to formulate the bias-variance trade-off using the above three parameters by following a 3-step procedure as described in Bishop, 2006, *Pattern Recognition and Machine Learning*, p. 151:

- Compute the *average prediction* as follows:

$$\bar{y}(x_n) = \frac{1}{L}\sum_{l=1}^{L} y^{(l)}(x_n) \tag{2.26}$$

- Compute the $(bias)^2$ as follows:

$$(bias)^2 = \frac{1}{N}\sum_{n=1}^{N}\{\bar{y}(x_n) - h(x_n)\}^2 \tag{2.27}$$

- Compute the *variance* as follows:

$$variance = \frac{1}{N}\sum_{n=1}^{N}\frac{1}{L}\sum_{l=1}^{L}\{y^{(l)}(x_n) - \bar{y}(x_n)\}^2 \tag{2.28}$$

Now, what's the difference between the bias-squared and variance errors? The answers are with the above equations. For a given or fixed model, the *bias* measures the predicted errors *relative to* the true function or regression function $h(x)$, which is unknown for practical machine learning problems (otherwise we would not need to *learn* using machine learning), whereas the *variance* measures the variations of all prediction runs using different datasets relative to the *mean* of all runs. In other words, the *bias* measures against the true unknown function, whereas the *variance* measures against the *mean* of multiple prediction runs without regarding to what the true distribution had been. As an example, let's say we are fitting a sinusoidal wave distribution using linear regression polynomial models. Then, for degree = 1, we would be fitting a sinusoidal wave with a line, which could be flat or with a non-zero slope. In this case, the bias would be large, as we are fitting a wiggled sinusoidal wave distribution with a "line," for which we know the difference is large. However, we may not care what true distribution we are dealing with and we even do not want to know what the true distribution actually is. Instead, all we care about is every time when we come up with a line from the trained model, we want to know how far its slope is relative to the mean slope averaged from all runs with the fixed degree of one. This is what the variance means. On the other hand, with increasing model complexity, the predicted fitting curve would get closer to the true sinusoidal wave distribution, so the bias would decrease, but variance may increase as the prediction fitted curve may become more oscillatory. This is actually a balance between under-fitting and over-fitting, and the minimal total error would be reached when under-fitting and over-fitting balance out with each other.

Next, we use an example to illustrate the essence of the bias-variance trade-off triage.

2.8.2 Illustration of the bias-variance trade-off triage with experiments

Bishop et al. (2006) once studied the bias-variance trade-off by fitting a sinusoidal wave distribution with the following test configurations: (1) $N = 25$ data points (or samples) per dataset to approximate a