8.1.1 A mathematical decomposition (for regression)

To formally state the bias-variance tradeoff for regression problems, we consider the following setup (which is an extension of the beginning paragraph of Section 8.1).

- Draw a training dataset $S = \{x^{(i)}, y^{(i)}\}_{i=1}^{n}$ such that $y^{(i)} = h^\star(x^{(i)}) + \xi^{(i)}$ where $\xi^{(i)} \in N(0, \sigma^2)$.
- Train a model on the dataset S, denoted by $\hat{h}_S$.
- Take a test example (x, y) such that $y = h^\star(x) + \xi$ where $\xi \sim N(0, \sigma^2)$, and measure the expected test error (averaged over the random draw of the training set S and the randomness of ξ)[5][6]

$$\text{MSE}(x) = \mathbb{E}_{S,\xi}[(y - h_S(x))^2] \tag{8.2}$$

We will decompose the MSE into a bias and variance term. We start by stating a following simple mathematical tool that will be used twice below.

Claim 8.1.1: Suppose A and B are two independent real random variables and $\mathbb{E}[A] = 0$. Then, $\mathbb{E}[(A + B)^2] = \mathbb{E}[A^2] + \mathbb{E}[B^2]$.

As a corollary, because a random variable A is independent with a constant c, when $\mathbb{E}[A] = 0$, we have $\mathbb{E}[(A + c)^2] = \mathbb{E}[A^2] + c^2$.

The proof of the claim follows from expanding the square: $\mathbb{E}[(A+B)^2] = \mathbb{E}[A^2] + \mathbb{E}[B^2] + 2\mathbb{E}[AB] = \mathbb{E}[A^2] + \mathbb{E}[B^2]$. Here we used the independence to show that $\mathbb{E}[AB] = \mathbb{E}[A]\mathbb{E}[B] = 0$.

Using Claim 8.1.1 with $A = \xi$ and $B = h^\star(x) - \hat{h}_S(x)$, we have

$$\begin{aligned} \text{MSE}(x) = \mathbb{E}[(y - h_S(x))^2] &= \mathbb{E}[(\xi + (h^\star(x) - h_S(x)))^2] && (8.3) \\ &= \mathbb{E}[\xi^2] + \mathbb{E}[(h^\star(x) - h_S(x))^2] \quad \text{(by Claim 8.1.1)} \\ &= \sigma^2 + \mathbb{E}[(h^\star(x) - h_S(x))^2] && (8.4) \end{aligned}$$

Then, let's define $h_{\text{avg}}(x) = \mathbb{E}_S[h_S(x)]$ as the "average model"—the model obtained by drawing an infinite number of datasets, training on them, and averaging their predictions on x. Note that h_{avg} is a hypothetical model for analytical purposes that can not be obtained in reality (because we don't

[5] For simplicity, the test input x is considered to be fixed here, but the same conceptual message holds when we average over the choice of x's.

[6] The subscript under the expectation symbol is to emphasize the variables that are considered as random by the expectation operation.

have infinite number of datasets). It turns out that for many cases, h_{avg} is (approximately) equal to the the model obtained by training on a *single* dataset with infinite samples. Thus, we can also intuitively interpret h_{avg} this way, which is consistent with our intuitive definition of bias in the previous subsection.

We can further decompose $\text{MSE}(x)$ by letting $c = h^\star(x) - h_{\text{avg}}(x)$ (which is a constant that does not depend on the choice of S!) and $A = h_{\text{avg}}(x) - h_S(x)$ in the corollary part of Claim 8.1.1:

$$\begin{aligned}
\text{MSE}(x) &= \sigma^2 + \mathbb{E}[(h^\star(x) - h_S(x))^2] && (8.5)\\
&= \sigma^2 + (h^\star(x) - h_{\text{avg}}(x))^2 + \mathbb{E}[(h_{\text{avg}} - h_S(x))^2] && (8.6)\\
&= \underbrace{\sigma^2}_{\text{unavoidable}} + \underbrace{(h^\star(x) - h_{\text{avg}}(x))^2}_{\triangleq \text{ bias}^2} + \underbrace{\text{var}(h_S(x))}_{\triangleq \text{ variance}} && (8.7)
\end{aligned}$$

We call the second term the bias (square) and the third term the variance. As discussed before, the bias captures the part of the error that are introduced due to the lack of expressivity of the model. Recall that h_{avg} can be thought of as the best possible model learned even with infinite data. Thus, the bias is not due to the lack of data, but is rather caused by that the family of models fundamentally cannot approximate the $h^\star$. For example, in the illustrating example in Figure 8.2, because any linear model cannot approximate the true quadratic function $h^\star$, neither can h_{avg}, and thus the bias term has to be large.

The variance term captures how the random nature of the finite dataset introduces errors in the learned model. It measures the sensitivity of the learned model to the randomness in the dataset. It often decreases as the size of the dataset increases.

There is nothing we can do about the first term σ^2 as we can not predict the noise ξ by definition.

Finally, we note that the bias-variance decomposition for classification is much less clear than for regression problems. There have been several proposals, but there is as yet no agreement on what is the "right" and/or the most useful formalism.

8.2 The double descent phenomenon

Model-wise double descent. Recent works have demonstrated that the test error can present a "double descent" phenomenon in a range of machine

learning models including linear models and deep neural networks.[7] The conventional wisdom, as discussed in Section 8.1, is that as we increase the model complexity, the test error first decreases and then increases, as illustrated in Figure 8.8. However, in many cases, we empirically observe that the test error can have a second descent—it first decreases, then increases to a peak around when the model size is large enough to fit all the training data very well, and then decreases again in the so-called overparameterized regime, where the number of parameters is larger than the number of data points. See Figure 8.10 for an illustration of the typical curves of test errors against model complexity (measured by the number of parameters). To some extent, the overparameterized regime with the second descent is considered as new to the machine learning community—partly because lightly-regularized, overparameterized models are only extensively used in the deep learning era. A practical implication of the phenomenon is that one should not hold back from scaling into and experimenting with over-parametrized models because the test error may well decrease again to a level even smaller than the previous lowest point. Actually, in many cases, larger overparameterized models always lead to a better test performance (meaning there won't be a second ascent after the second descent).

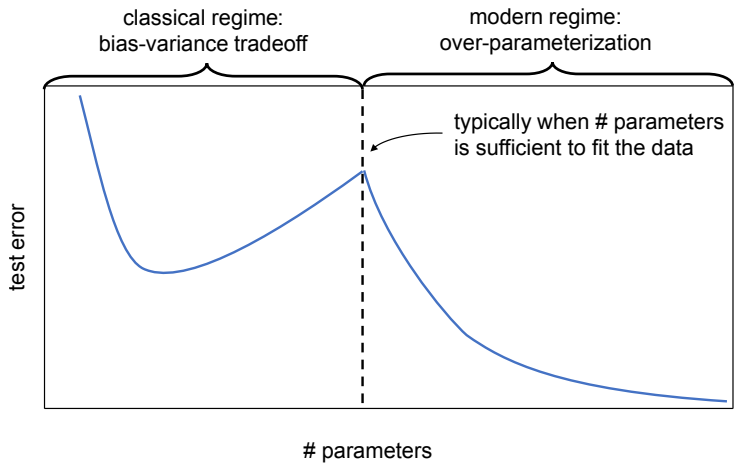

Figure 8.10: A typical model-wise double descent phenomenon. As the number of parameters increases, the test error first decreases when the number of parameters is smaller than the training data. Then in the overparameterized regime, the test error decreases again.

[7] The discovery of the phenomenon perhaps dates back to Opper [1995, 2001], and has been recently popularized by Belkin et al. [2020], Hastie et al. [2019], etc.

Sample-wise double descent. A priori, we would expect that more training examples always lead to smaller test errors—more samples give strictly more information for the algorithm to learn from. However, recent work [Nakkiran, 2019] observes that the test error is not monotonically decreasing as we increase the sample size. Instead, as shown in Figure 8.11, the test error decreases, and then increases and peaks around when the number of examples (denoted by n) is similar to the number of parameters (denoted by d), and then decreases again. We refer to this as the sample-wise double descent phenomenon. To some extent, sample-wise double descent and model-wise double descent are essentially describing similar phenomena—the test error is peaked when $n \approx d$.

Explanation and mitigation strategy. The sample-wise double descent, or, in particular, the peak of test error at $n \approx d$, suggests that the existing training algorithms evaluated in these experiments are far from optimal when $n \approx d$. We will be better off by tossing away some examples and run the algorithms with a smaller sample size to steer clear of the peak. In other words, in principle, there are other algorithms that can achieve smaller test error when $n \approx d$, but the algorithms evaluated in these experiments fail to do so. The sub-optimality of the learning procedure appears to be the culprit of the peak in both sample-wise and model-wise double descent.

Indeed, with an optimally-tuned regularization (which will be discussed more in Section 9), the test error in the $n \approx d$ regime can be dramatically improved, and the model-wise and sample-wise double descent are both mitigated. See Figure 8.11.

The intuition above only explains the peak in the model-wise and sample-wise double descent, but does not explain the second descent in the model-wise double descent—why overparameterized models are able to generalize so well. The theoretical understanding of overparameterized models is an active research area with many recent advances. A typical explanation is that the commonly-used optimizers such as gradient descent provide an implicit regularization effect (which will be discussed in more detail in Section 9.2). In other words, even in the overparameterized regime and with an unregularized loss function, the model is still implicitly regularized, and thus exhibits a better test performance than an arbitrary solution that fits the data. For example, for linear models, when $n \ll d$, the gradient descent optimizer with zero initialization finds the *minimum norm* solution that fits the data (instead of an arbitrary solution that fits the data), and the minimum norm regularizer turns out to be a sufficiently good for the overparameterized regime (but it's not a good regularizer when $n \approx d$, resulting in the peak of test

error).

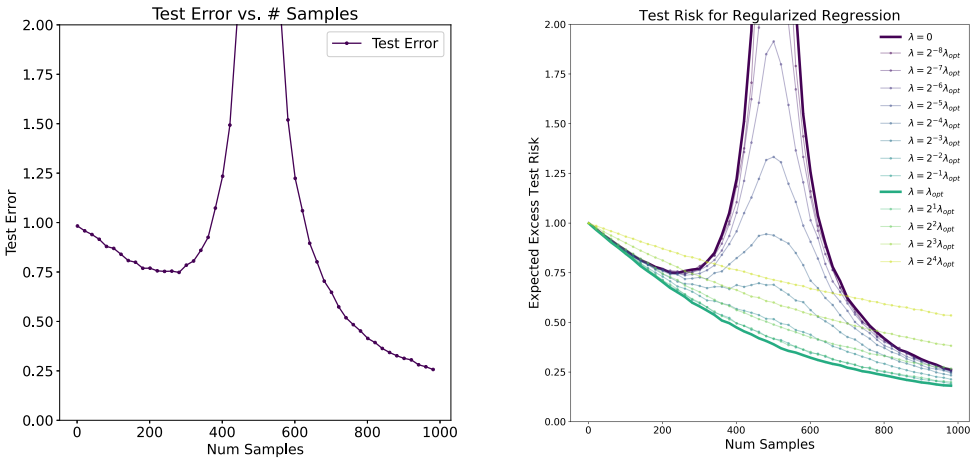

Figure 8.11: **Left:** The sample-wise double descent phenomenon for linear models. **Right:** The sample-wise double descent with different regularization strength for linear models. Using the optimal regularization parameter λ (optimally tuned for each n, shown in green solid curve) mitigates double descent. **Setup:** The data distribution of (x, y) is $x \sim \mathcal{N}(0, I_d)$ and $y \sim x^\top \beta + \mathcal{N}(0, \sigma^2)$ where $d = 500, \sigma = 0.5$ and $\|\beta\|_2 = 1$.[8]

Finally, we also remark that the double descent phenomenon has been mostly observed when the model complexity is measured by the number of parameters. It is unclear if and when the number of parameters is the best complexity measure of a model. For example, in many situations, the norm of the models is used as a complexity measure. As shown in Figure 8.12 right, for a particular linear case, if we plot the test error against the norm of the learnt model, the double descent phenomenon no longer occurs. This is partly because the norm of the learned model is also peaked around $n \approx d$ (See Figure 8.12 (middle) or Belkin et al. [2019], Mei and Montanari [2022], and discussions in Section 10.8 of James et al. [2021]). For deep neural networks, the correct complexity measure is even more elusive. The study of double descent phenomenon is an active research topic.

[8] The figure is reproduced from Figure 1 of Nakkiran et al. [2020]. Similar phenomenon are also observed in Hastie et al. [2022], Mei and Montanari [2022]

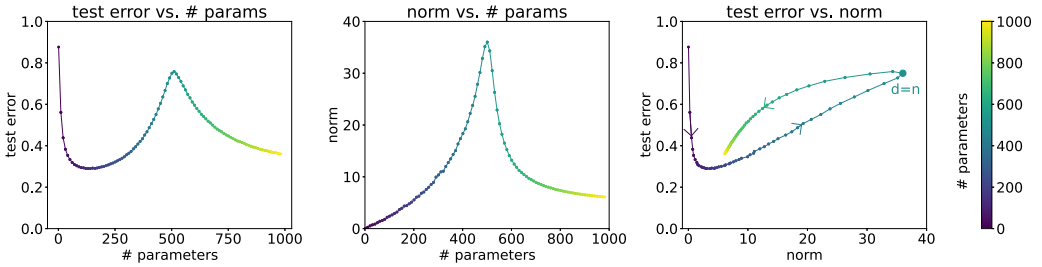

Figure 8.12: **Left:** The double descent phenomenon, where the number of parameters is used as the model complexity. **Middle:** The norm of the learned model is peaked around $n \approx d$. **Right:** The test error against the norm of the learnt model. The color bar indicate the number of parameters and the arrows indicates the direction of increasing model size. Their relationship are closer to the convention wisdom than to a double descent. **Setup:** We consider a linear regression with a fixed dataset of size $n = 500$. The input x is a random ReLU feature on Fashion-MNIST, and output $y \in \mathbb{R}^{10}$ is the one-hot label. This is the same setting as in Section 5.2 of Nakkiran et al. [2020].