7.1

Intro

This exercise only ask us to explain the behaviour of the training set error as our number of instances increases. More specifically, it ask us why the training error of high capacity models might increase as the size of the dataset also increases, before reaching a plateau. I will give two answers to this question. Don't worry, they are both equal, in the core content. But the first will be the expected answer that is short and based on the high-level concept of overfitting and the second will be a more personal view of the same problem, given more detail of what is happening.

Solution

First answer

Complex models are prone to overfitting, which means when they are paired with little data, they will "memorize" the training dataset and give low errors on the training dataset. Nevertheless, since the model has a lot of parameters and not enough training data to learn them well, the model don't generalize beyond the training set and give high errors on the test dataset and on real data as well. When we increase the dataset size, the model starts learning instead of memorizing the dataset. The final result is that both the training and test errors start to converge to the same answer. In another words, the training/test errors increases/decreases to the best generalization error of the model.

Second answer

When we are working with regresion in supervised learning, we usually are interested in modeling the following equation $y = f(x) + \epsilon$, where f is the mapping from the input to the output and ϵ is a intrinsic noise that we can't remove from our model. The presence of the intrinsic noise tell us that our model will never be perfect using just x as input. Therefore we should accept that there will be a minimal amount of erros in our predictions, which is represented as the plateau that both the training and test erros converge in Figure 7.10.

In this scenario, our goal is to minimize the amount of errors by finding a function f^* such that $f^*(x) = f(x)$ for all $x \in X$. In practice, it is usual to work with parameterized models and the quest then becomes to find the real parameters w such that $f^*_w(x) = f(x)$. In the particular case where we have a complex model and little data to train it, it often happens that the training error is a overestimation of the true capabilities of the model. To explain this, let's use the probabilistic tools that we learned in this book.

As we saw in chapter 3, training a model usually means to compute the MLE or the MAP estimation of its parameters. Let's focus on the MLE for now. The MLE is a estimator based only on the data that the model has seen (training data). Furthermore, the MLE is the solution of an optimization problem. Thus, the MLE shows the best model to explain what the model has seen. The real danger of this approach is that the MLE does not put any tought on all the data that it has not seen. Remembering the black swan paradox of chapter 3: if the training data don't have a tail as one of its instances, the MLE will say that tail is not possible, because this is the optimal explanation of the data.

Since the MLE is the optimal explanation for the data, it has a very low training error. Unfortunatly, the test error does not follow the same pattern, since the MLE was not optimized to explain all data, but the training data. This scenario is one of the faces of overfitting and can be fought with the use of regularization.

The two most common regularization approaches to solve the shortcomings of the MLE are l_p penalty and gather more data. l_p penalty assume a prior over the weights, so now the model cannot bend itself to explain the data on the training set. Gathering more data also works because now the MLE is being optimized over a wide range of scenarios, so the danger associated with ignoring the unseen datapoints decreases. Since the optimization is across a broader range of possibilities its minimum value will be bigger (but more truthful to reality), expalining why the training error increases.

Conclusion

In this exercise we explained the behaviour of the training set error as our number of instances increases in complex models. Two answers were given: the short, more conventional one and the second based more closely on my personal view on the matter. Both say the same thing but are based on different perspectives and with the second going in more depth.