Assignment Report

Mitchell Chatterjee
Carleton University (101141206)

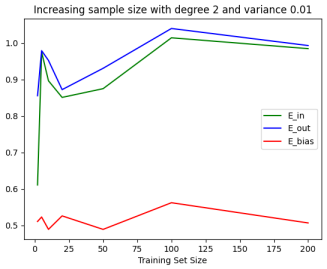
The intent of this assignment was to fit a polynomial model to a randomly generated dataset and compare the performance when modifying various hyperparameters. These hyperparameters included: model complexity (degree), variance in the dataset and the size of the training sample.

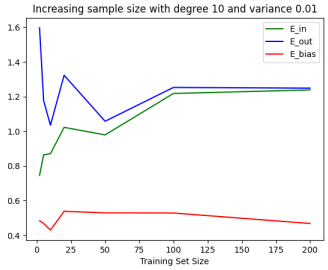
The Pytorch package was used in order to complete this assignment. Aiding in generating random data, performing tensor operations and automatic differentiation when implementing mini-batch Stochastic Gradient Descent.

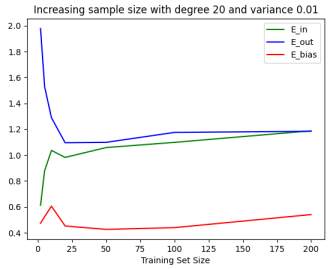
Mini-batch Stochastic Gradient Descent was used in order to train the model. This was implemented manually in the code. The random mini-batch was taken when the sample size exceeded 50 in order to improve performance while training. Note that in plotting the results regular Gradient Descent was used in order to bring greater emphasis to the effect of changing the sample size.

Part E

The results from the first set of experiments (E) are shown below. Three sets of plots were chosen In order to best illustrate the effect that modifying the three hyperparameters would have on the performance of the model.





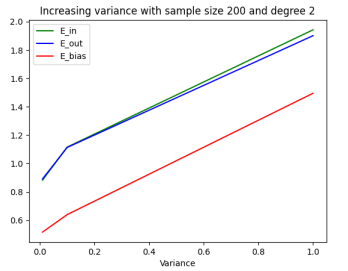


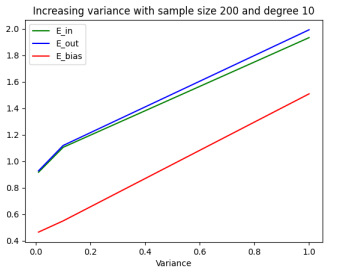
The first set of plots demonstrates the effect of increasing the sample size with a constant variance of 0.01 across each plot, and polynomials of degree 2, 10 and 20. The reasoning behind choosing these hyperparameters to demonstrate the effect of increasing the sample size is as follows. In order to reduce the effects of noise when fitting the model, a variance of 0.01 was selected. With respect to selecting the model complexity. In reference to the bias-variance trade-off: the higher the degree of the polynomial the less bias it shows to a particular model shape. Increasing the degree of the model demonstrates how a more complex

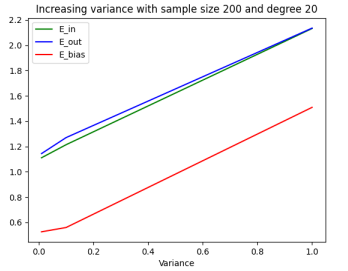
model can fit the data more closely. As a result increasing the sample size reduces the effect of overfitting as seen in slide 27 of lecture 1.

The model of degree 2 has a much higher variability, while the other two models of degree 10 and 20 fit the data much more closely. However, the model of degree 20 begins to overfit the data around the training size of 100. In this sense the capacity of the model is too high for the learning task and it is fitting to the noise in the training data, reducing its ability to generalize to unseen data.

The second set of plots demonstrates the result of increasing the degree with constant variance of 0.01 and sample sizes of 2, 20 and 200. Again a variance of 0.01 was chosen in order to decrease the noise in fitting the model. Allowing us to focus on the effects of increasing the degree of the polynomial. As the sample size increases the generalization gap between training and test data becomes smaller. This is again in reference to slide 27 of lecture 1. Where the sample size affects the generalization gap for both simple and complex models. This is a result of the bias-variance trade-off where the greater the complexity of the model the more closely we can attune to the data itself.



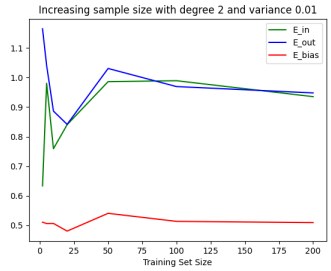


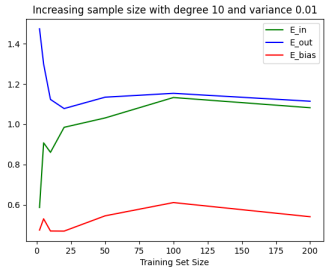


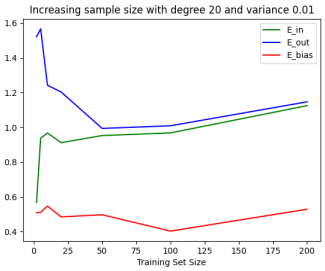
The third and final set of plots demonstrates the result of increasing the variance with a sample size of 200 and degrees of 2, 10 and 20. In order to fully demonstrate the effects of adding noise to the data a sample size of 200 was used. This gave the model the greatest chance to fit to the underlying function producing the data and to reduce the bias towards the noise of any particular point. In each plot we see a rapid rise in the $\overline{E_{in}}$, $\overline{E_{out}}$ and E_{bias}. However, it is particularly apparent in the model of degree 20. This is a result of the model overfitting to the noise in the data as the capacity of the model exceeds the learning task itself.

Part F

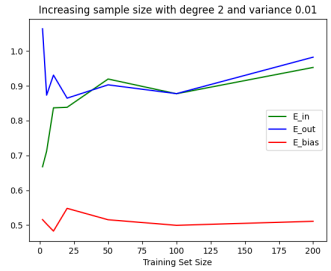
Part F resulted in repeating the experiments of Part E while implementing weight decay regularization. We repeated the first set of plots with different values of the hyperparameter lambda in order to demonstrate the effect of a larger or smaller weight decay penalty.

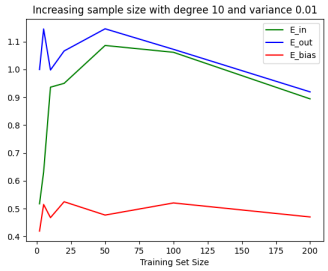


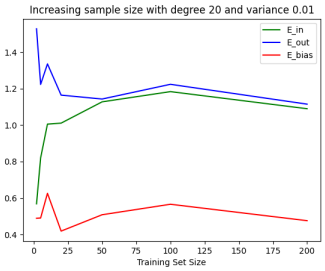




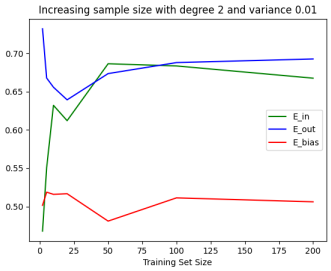
Lambda = 0.01

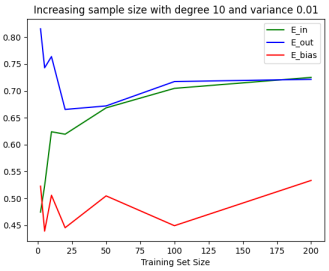


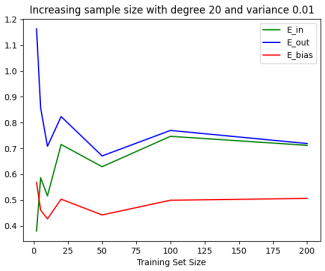




Lambda = 0.1

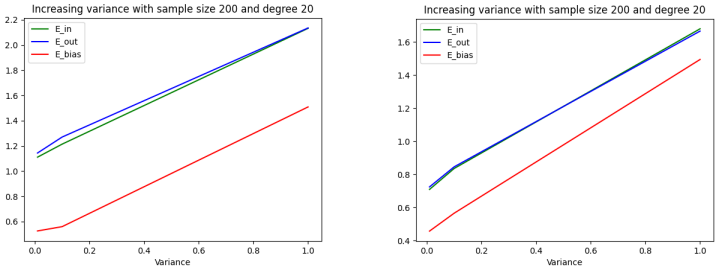






Lambda = 1

As we can see in the results. A larger weight decay penalty gave better results on the unseen data as it allowed the model to generalize more successfully. A larger value of lambda shows a preference for smaller weights in the model. This means that we are less likely to overfit to the noise in the data.



Lambda = 0 (Left), Lambda = 1 (Right)

Finally we compare the result of increasing the value of lambda and its effect on a model's ability to withstand noise in the data. In order to do this we compare the model of greatest complexity which is most likely to overfit to noise in the data. With a large value of lambda we see that the effect of increasing the variance is reduced significantly. This is again a result of the weight decay regularization preferring smaller values for the weights and ensuring the model does not overfit to the increased noise in the data.