CSI 5340 Assignment 1

Name: Alim Manjiyani

Student Number: 300229095

20th September 2021

This assignment deals with my explorations of fitting, generalization and regularization of regression models via simulation, according to the steps given in the assignment. I have separated the code in two files (regularized and non-regularized) for readability and easiness in output generation.

I tried to explore the aspect of regularization in every experiment possible. I have implemented all the estimators (GD, SGD, Mini-batch SGD) but decided to use the Mini-batch SGD for my experiments as it is faster, and the performance is as good as GD.

I begin my experiments with the set rules given in the assignment by fixing one factor and accessing the effect of other factors on the error values. Below are the experiments, their results and their respective conclusions.

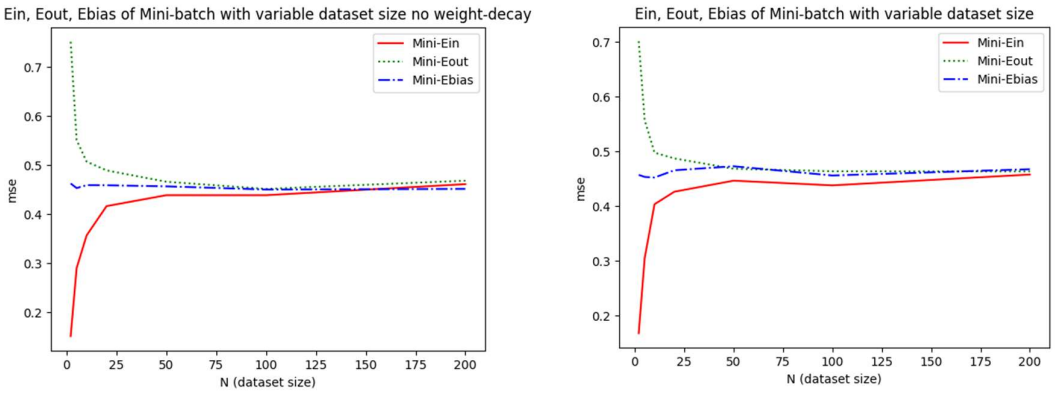
1 Variable Sample Size (N)

Firstly, we observe the effects of increasing dataset size on a fixed model. Then, we see the changes in the performance with changes in the model complexity and noise level.

We compare it side by side with a regularized estimator as well

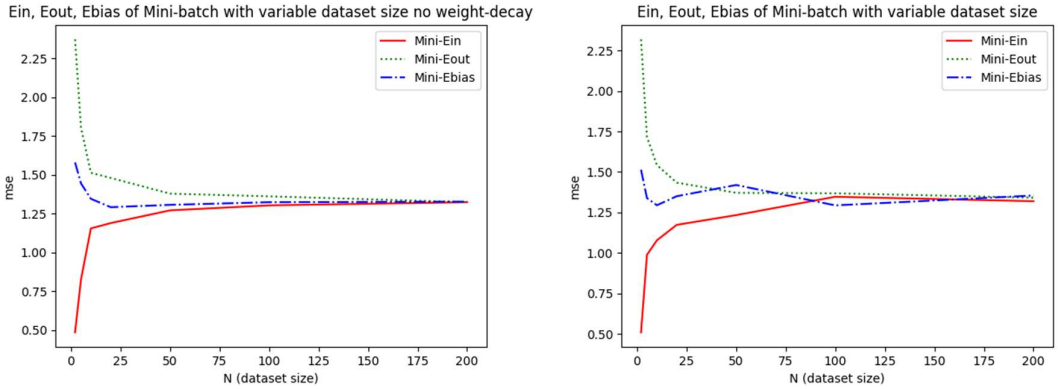
Parameter Settings:

- Data Size (N): N ∈ {2,5,10,20,50,100,200}
- Complexity (d): 3
- Standard Deviation (σ): 0.1
- Number of Trials (M): 50
- Learning Rate (lr): 0.001
- Batch Size: 40
- Epochs: 2000
- Weight decay (λ): 0.1



(a)non-regularized *(b)regularized*

Figure 1: Result of variable sample size on a simple model



(a)non-regularized (σ=1) *(b)regularized (σ=1)*

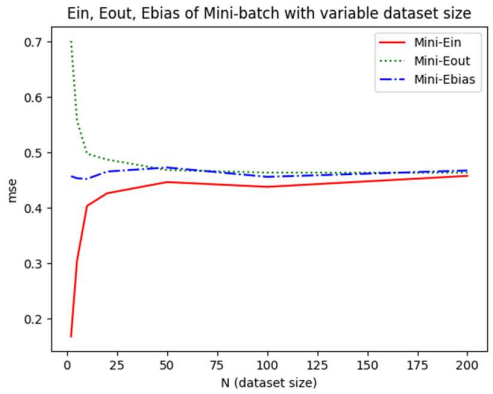
Figure 2: Results of increased noise in the data sample

Conclusion

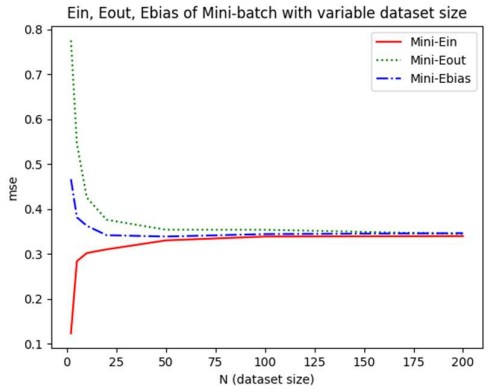
As the dataset size increases, initially, E_{out} decreases while E_{in} increases and then they ultimately reach a stable point where they converge. Whereas, E_{bias} seems to be stable from the beginning and shows that averaging the polynomial can give better results at lower sample size. Furthermore, it can be inferred that increasing the sample size can increase the performance of the model if not decrease.

From Figure 2., we can observe the effects of regularization as the E_{gen} (E_{in} - E_{out}) gap is reduced at an early stage (N=100) in regularized model than that in the non-reg model (N=150) when the noise level is increased in the sample.

Theoretically, an increase in complexity (from d=3 to d=12) can help in reducing the error significantly (as seen in Figure 3.) but a highly complex model has a great tendency to overfit the data (as seen in Figure 4.) and hence reduce the accuracy of the model.

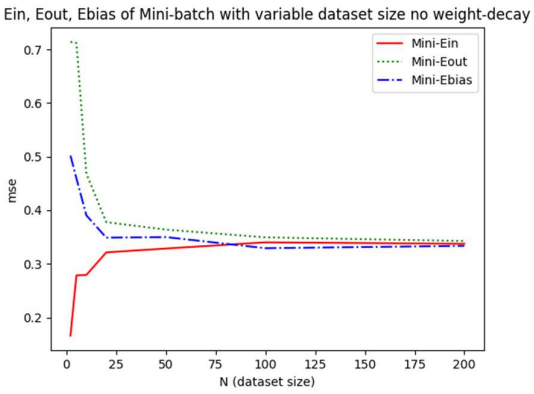


(a)regularized simple(d=3)

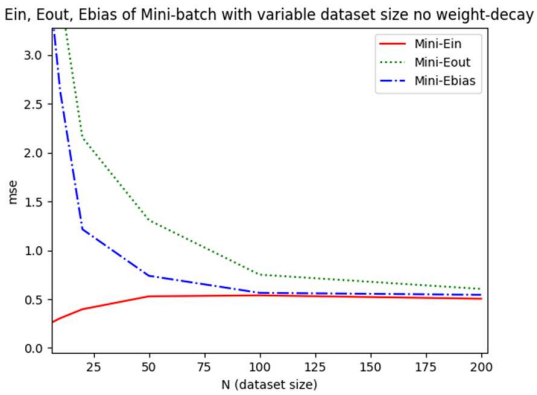


(b)regularized complex (d=12)

Figure 3: Increasing complexity can reduce error



(a)non-regularized complex (d=12)



(b)non-regularized highly complex(d=50)

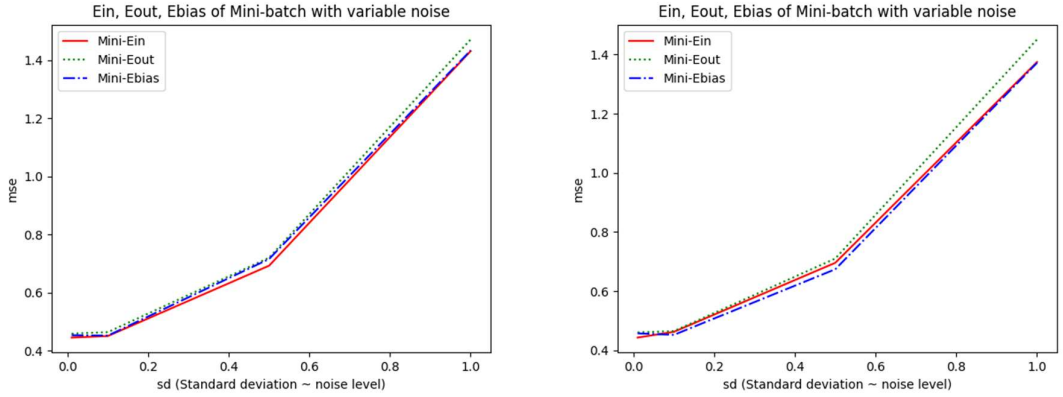
Figure 4: Highly complex model causes overfitting

2 Variable Noise/Variance (σ^2)

This experiment shows the result of varying complexity over a fixed sample size and noise level. Also, we will examine the change in error as the complexity and sample size changes

Parameter Settings:

- Data Size (N): 100
- Complexity (d): 3
- Standard Deviation (σ): $\sigma \in \{0.01,0.1,0.5,1\}$
- Number of Trials (M): 50
- Learning Rate (lr): 0.001
- Batch Size: 40
- Epochs: 2000
- Weight decay (λ): 0.1



(a)non-regularized *(b)regularized*

Figure 5: Result of changing noise levels

Conclusion

Although the two results look similar there are many things that can be inferred. Firstly, E_{in}, E_{out} and E_{bias}, all increase with increase of variance in the dataset. Secondly, the regularized model perform slightly better than the other as we can see the slope (rate of change of error) of the later is greater. Also, the increased E_{gen} gap in the regularized model can be attributed the small sample size and will be less comparatively to the non-regularized model when the dataset size is large.

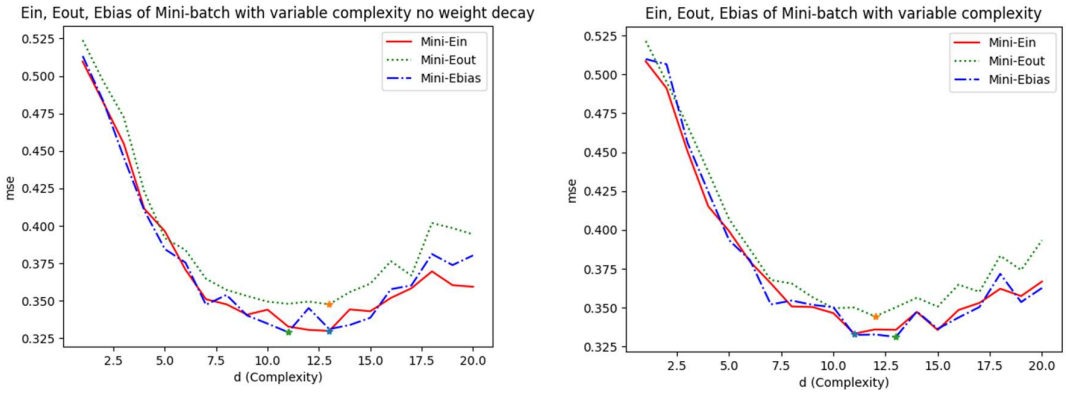
The result of increasing complexity and sample size show little to no change in error as it remains proportional to the noise level.

3 Variable complexity (*d*)

Varying the complexity and plotting the errors can tell us the best complexity of the model for the given dataset. This helps us distinguish between underfitting and overfitting of the chosen model.

Parameter Settings:

- Data Size (*N*): 100
- Complexity (*d*): d ∈ {1, 2…, 20}
- Standard Deviation (σ): 0.1
- Number of Trials (*M*): 50
- Learning Rate (*lr*): 0.001
- Batch Size: 50
- Epochs: 2000
- Weight decay (λ): 0.1



(a)non-regularized *(b)regularized*

Figure 6: Results of variable model complexity

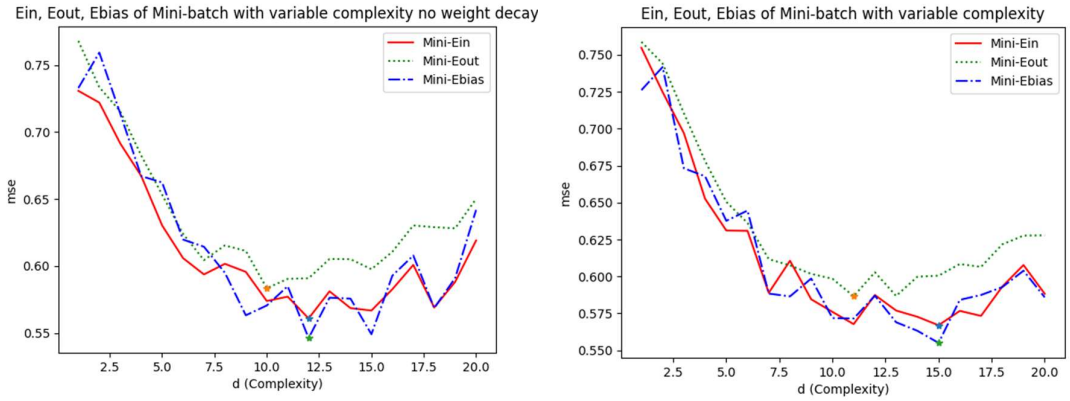
Conclusion

Initially as complexity increases, the error decreases; and then there appears a point where the error start increasing with increasing complexity, which divides the graph into two differential model categories: underfit and overfit.

From Figure 6. We can infer that, given the same data set a regularized model can bring efficient results at a little lower complexity (*d*=12) hence saving computational costs when compared to a non-regularized model (*d*=13).

As the noise level increases, the regularized model does a good job in minimizing the error and handling the noise when compared to the non-reg model (Figure 7.)

Finally, We can observe the effects of generalization in Figure 8., as it clearly shows how a regularized model can give a better picture of what E_{out} can be given its Ein (Since E_{gen} is low, $E_{in} \sim E_{out}$), whereas it is unclear when using a non-regularized model.



(a)non-regularized (d=0.5) *(b)regularized (d=0.5)*

Figure 7: Effects of increased noise level

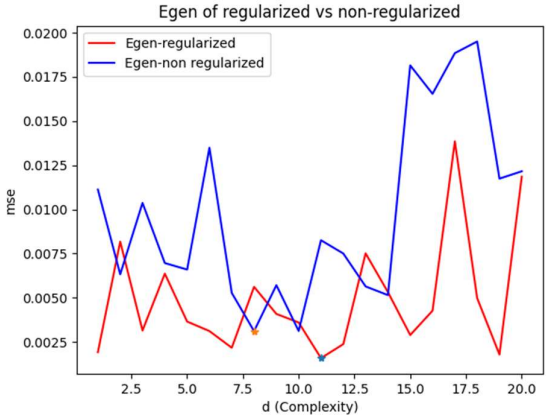


Figure 8: Generation Gap(E_{gen}) comparison

4 Regression Estimators (GD, SGD, Mini-batch SGD)

This experiment was just out of curiosity to see how the Stochastic gradient descent will perform on low sample size (as it is only recommended for huge datasets) and it also justifies the use of Mini-batch SGD for all the experiments above.

Parameter Settings:

- Data Size (N): N ∈ {2,5,10,20,50,100,200}
- Complexity (d): 10
- Standard Deviation (σ): 0.1
- Number of Trials (M): 50
- Learning Rate (lr): 0.001
- Batch Size: 50
- Epochs: 2000
- Weight decay (λ): 0.1

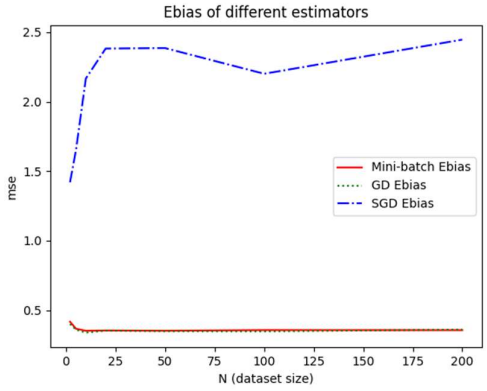


Figure 9: E_{bias} comparison of different estimators with increasing sample size

Conclusion

The fact that SGD performs poorly on low sample size as it considers only one data-value for weight calibration per epoch can be realised in Figure 9. Moreover, the low value of SGD-E_{bias} during the initial value of N can be attributed to the low sample size.

Although Mini-batch SGD and GD have similar results, Mini-batch SGD gains an advantage over GD when it comes to the processing speed for large sample sizes.