Esad İsmail Tök
21801679
GE 461- 01

GE 461—Introduction to Data Science

Project 3 – Supervised learning

Part A

The dataset given is not suitable for linear regression because it exhibits a high sum of squared loss, as shown in Figure 1. Moreover, the input and output values are not linear, as indicated by the scattered points in Figure 1. Therefore, it is necessary to utilize an Artificial Neural Network (ANN) with a single layer to reduce errors and make accurate predictions.

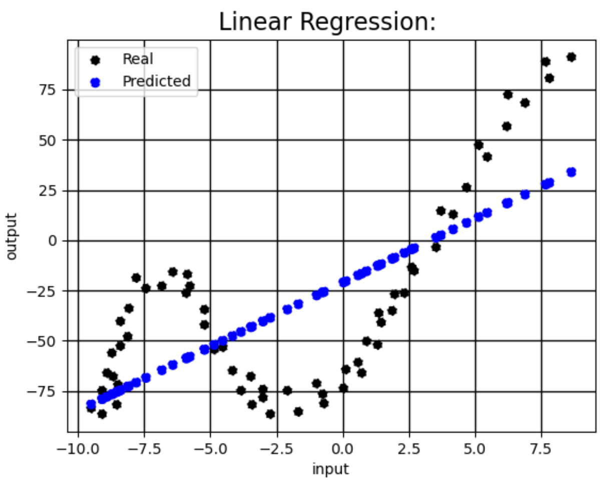


Figure-1: Plot of Linear Regression Loss

I utilized an ANN with 2 hidden units, 10000 iterations (epochs), and a learning rate of 0.001, as specified in the project description. However, this configuration resulted in an error which surpassed the error obtained from linear regression. To address this issue, I decided to modify the number of hidden units as it is suggested. As expected, when I increased the number of hidden units to 4, there was a significant decrease in the error. Based on this observation, I concluded that using at least 4 hidden units would be adequate for accurate value prediction. Furthermore, as shown in the table, increasing the number of hidden units further resulted in a reduction in the error as it is expected.

Esad İsmail Tök
21801679
GE 461- 01

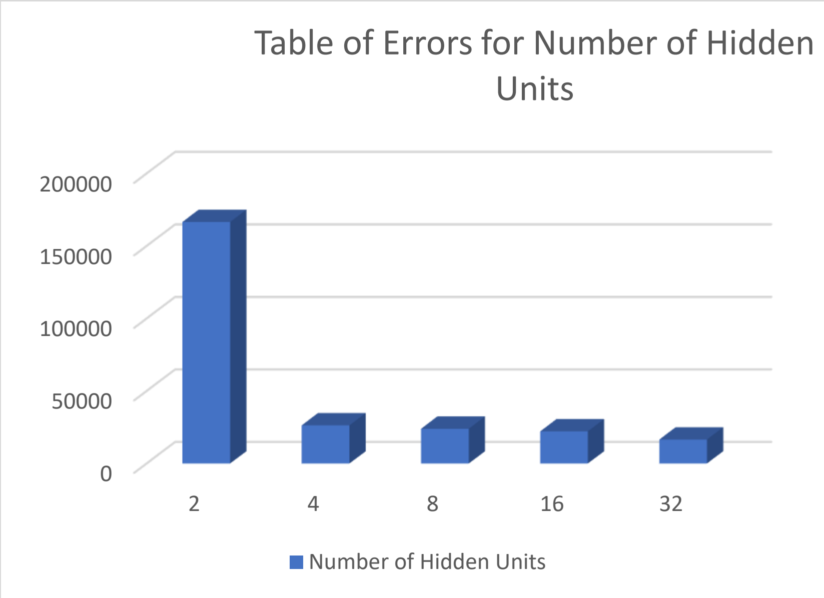


Table-1: Number of Errors for Hidden Units 2, 4, 8, 16 and 32

In order to determine the best points for prediction, I performed experiments using 32 hidden units with various learning rates. Initially, I used 32 hidden units, 10000 iterations (epochs), and a learning rate of 0.01 (1%). Then, I decreased the learning rate to 0.001 (0.01%) and further reduced it to 0.0001 (0.001%). The resulting error values are displayed in Table 2.

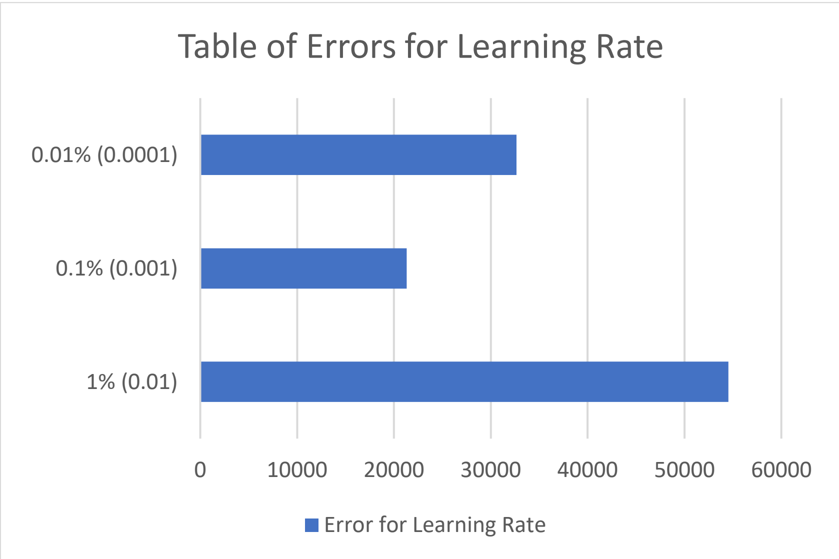
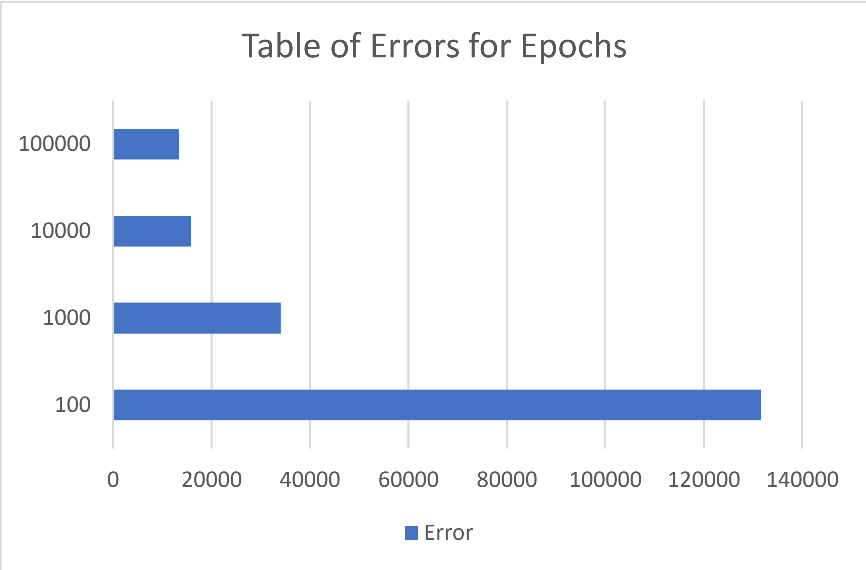


Table 2: Number of Errors for Learning Rates 0.01, 0.001 and 0.0001

The findings presented in Table 2 indicate that a learning rate of 0.1% is the most optimal choice for this project. This specific rate leads to the minimum error, as higher or lower rates result in increased errors. Regarding the weights, I rearranged the values based on the error of the predicted output. I made sure to adjust the weights of the layers at least once by sequentially utilizing every index corresponding to the input.

Appropriate number of epochs for testing to determine, I conducted experiments with different values: 100, 1000, 10000, and 100000 epochs.

Esad İsmail Tök
21801679
GE 461- 01



From the data presented in Table 3, it is evident that the most favorable outcome is obtained with 100000 epochs. Consequently, for this particular project, I have opted to utilize 100000 epochs to improve the precision of value predictions.

Table-3: Number of Errors for Epochs 100, 1000, 10000 and 100000

Part B

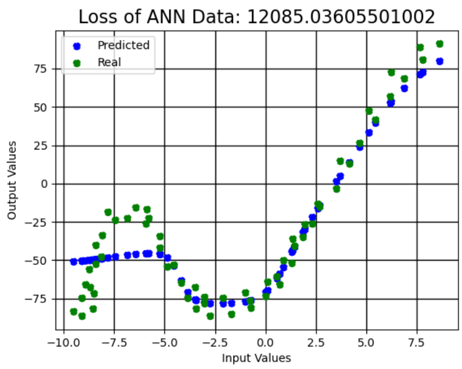


Figure-2: Plot of Train Data

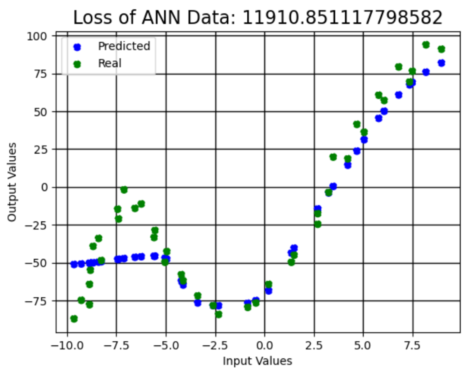


Figure-3: Plot of Test Data

ANN used (specify the number of hidden units): 32

Learning rate: 0.001

Range of initial weights: 0,1

Number of epochs: 100000

When to stop: 100000

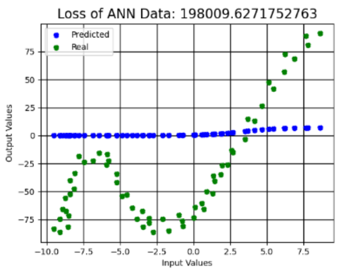
Is normalization used: No

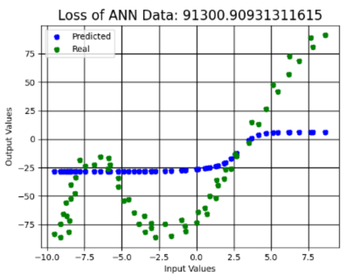
Training loss (averaged over training instances): 12085

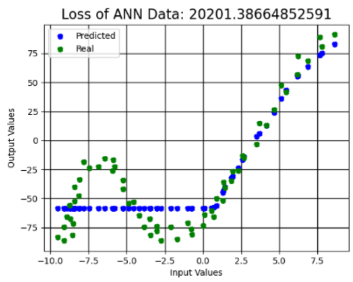
Test loss (averaged over test instances): 11910

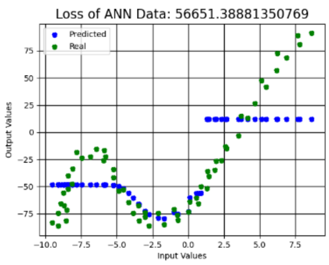
Esad İsmail Tök
21801679
GE 461- 01

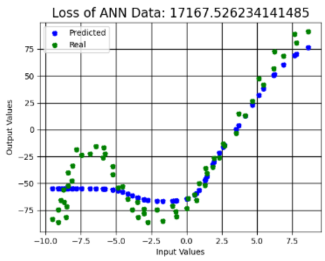
Part C











The aforementioned figures display the performance of the Artificial Neural Network (ANN) with different numbers of hidden units, namely 2, 4, 8, 16, and 32. The graphs clearly demonstrate that as the number of hidden units increases, the error decreases.

Hidden Unit	Training Loss	Test Loss	Train Loss - Avg	Test Loss - Avg	Standard Deviation (Train)	Standard Deviation (Test)
2	198009	123534	3300	3013	3093	2065
4	91300	67999	1521	1658	1319	1003
8	20201	18897	336	460	378	375
16	56651	51755	944	1262	986	1298
32	17167	16785	286	409	278	391

Table-4: Error Statistics of ANN with different hidden unit count

Nevertheless, it is crucial to acknowledge that altering the configurations of each Artificial Neural Network (ANN) may result in larger errors despite using a higher number of hidden units. This observation indicates that the performance of a neural network is not solely determined by its complexity, and other factors come into play.