Phase 2

Intentional Overfit

Alexander Garcia

December 2024

Contents

1 Introduction

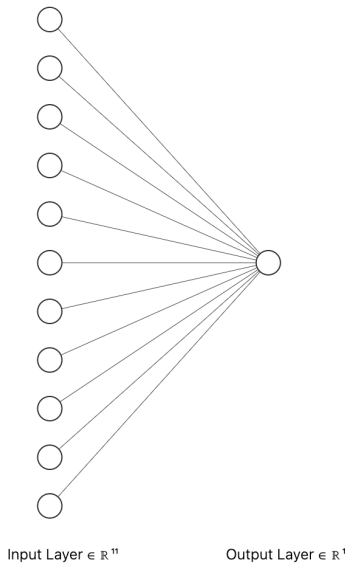
The goal of this phase is to explore which architecture will result in overfitting. Overfitting occurs when the model fits too closely to its training data and learns those patterns. The target accuracy for training data to accomplish overfitting was close to or at 100%. This will aid when later choosing an optimal network architecture as the optimal network will have to be smaller than the overfit models. For the purpose of this phase the data was **not** split into training and validation sets.

2 Logistic Regression Model

The baseline model is a Logistic Regression model. This model has an input layer equal to the amount of inputs (11 in this case) and one output layer with a single neuron that uses the sigmoid activation function. This model ran for 256 epochs The sigmoid formula is as follows:

$$\mathrm{S(x)} = \frac{1}{1 + e^{-x}}$$

Figure 1: Logistic Neural network with 11 inputs and 1 output



3 Unsuccessful Overfit Models

Three attempts using networks of sizes 2-1, 8-1, and 8-4-1, each running at 256 epochs failed to result in overfitting. The 2-1 network only achieved 85% while the other two networks both reached 90%. Each node in the hidden layers used the ReLU activation function while the output layer still used sigmoid. ReLU stands for rectified linear unit which is simply:

$$f(x) = \max(0, x)$$

4 Successful Overfit Models

Three successful attempts resulted in overfitting with two models continuing to use two hidden layers and one output layer. The first model to overfit was a 64-4-1 that ran for 256 epochs. It did not achieve 100% but ended at 98%. Second was a 64-32-1 model that ran for only 100 epochs but still achieved similar results to the 64-4-1 model. The third model was overloaded with 500 neurons in the single hidden layer (500-1) and ran for 256 epochs. Again, the hidden layers used ReLU while the output layer used sigmoid.

5 Conclusion

In summary, I ran 7 total models while mixing in combinations of layers, number of neurons, and epochs while attempting to overfit my data. The most interesting overfit to me was a single layer with 500 neurons. Given this result, I know my architecture in subsequent phases will have to be smaller than 64 neurons in a layer. However, using 3 layers with an appropriate amount of neurons seems like a good candidate.