Deep Learning

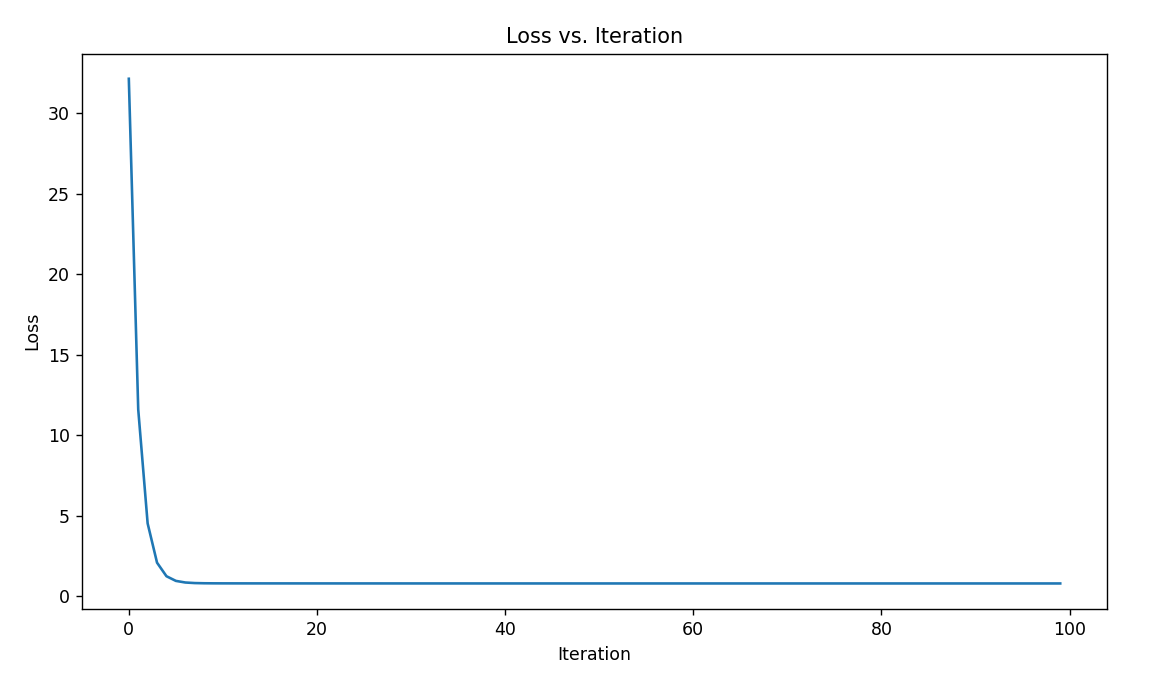
**QS1:**

**1.a**

**1.b**

We create our SGD in a way that it doesn’t use the softmax activation function on the output layer if there is only one output neoron in the least squared error example in for this case we don’t want the softmax or the sigmoid, because our network is targeted to predict the y value (a regression).

We then run our code for the MSE problem and get amazing results, even better then the closed form solution.

\*It is worth mentioning that we used early stopping if loss doesn’t decrease for 10 iteration in a row, but it just kept on decreasing. It is hard to see that in the graph because the loss starts at a very high value.

A line graph with blue dots

Description automatically generated

In values we get:

The closed form equation is: Our solution equation is:

Mean Squared Error (Closed-form): 6.1851108724937465

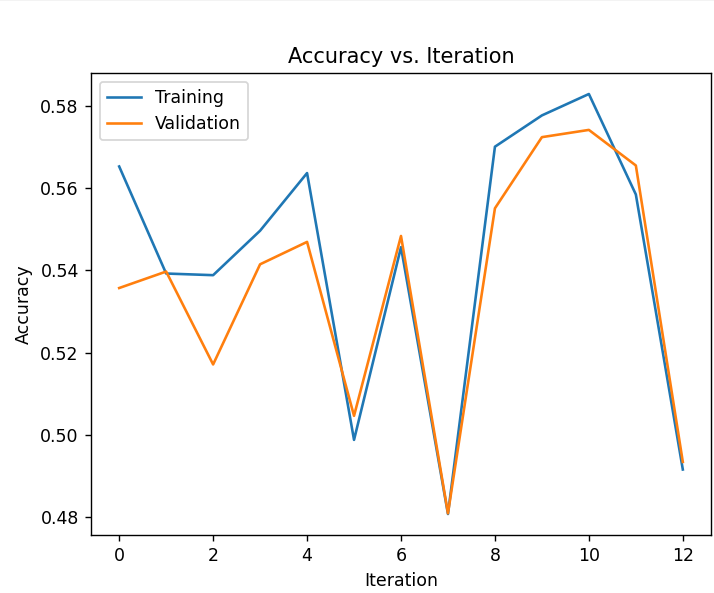
Mean Squared Error (SGD): 0.8065947869466693

1.c

We test our minimization of the softmax function first using PeaksData dataset. Running our SGD on multiple combinations of learning rate and batch size. Before moving in, we have to hold in mind that one layer with a softmax activation function cant solve this well because the simple fact that having 0 inner layers means that the NN can learn only a linear separator, as it did well in the previous question. As this task isn’t really in the power of our “mini NN”, we have no high hopes for significant differences between the parameter combinations, and they can be highly dependent on the initialization of the random weights.  
Running all the possible combinations out of the options we can see in the code(Function Qs3), we find that the optimal combination to be: Learning rate: 0.1 and Batch size: 10.

We once again use early stopping when for 10 iterations the loss doesn’t decrease on the validation set.

\*in the graph, Iteration is equal to an epoch, meaning that it is an iteration over all the data(/batches)

A graph with blue lines

Description automatically generated