

**1. What impact does changing of training samples and the number of test samples have on the results. Current code uses 1000 and 100. See how much you can vary this and then show how the results vary. Do fewer make it better/worse/same? Do more make it better/worse same?**

- <https://www.loom.com/share/6e671b00adf146b8a7f737f2e59932b1>
- Lowering training will generally lower accuracy and loss
- Lowering testing will generally falsely raise accuracy and lower loss
- Lowering both will significantly change accuracy and loss lower and higher depends on the random tests selected
- Increasing training slightly decreases loss
- Increasing testing will also slightly impact loss
- Increasing both will slightly increase accuracy and loss
- More training data should help prevent overfitting (just don't add too much) while decreasing it to much will make it more likely. More testing data helps to make sure there are no bias in validation.
- Tuning models is a balancing act to much or to little of either data set can impact the model the best number to use can be felt out and the author stated this comes with experience in one of the early chapters

**2. What impact does the number of epochs have on the results? Current code uses 10,000. Try more and try less.**

- <https://www.loom.com/share/08ffe078559044109deb67ca5af43deb>
- Not enough results in lower accuracy
- The correct amount results in high accuracy
- Too many can result in overfitting and a lower accuracy

**3. Using strategic print statements, show and explain the details of what takes place in class Optimizer\_Adam**

- <https://www.loom.com/share/a6bf2e886eef4fabab209f51ec1fae63>
- Adam is adaptive momentum
- This has momentum and a cache of a set number of recent epochs
- Each neuron has separate weight and bias
- Adam changes the learning rate down slowly as the global minimum is approached
- Due to time to run I stopped video at 100 but the change in learning rate is more significant as the number of epochs goes up

4. **Using strategic print statements, show and explain the details of what takes place in class `Activation_Softmax`.**
  1. <https://www.loom.com/share/e133d6b8d24441ef8cee6f9eab3385fc>
  2. This normalized the output to get the confidence scores for what the network thinks the is for the video series this would be the likelihood of each hand drawn number and what the answer is the highest confidence score is the number output by that network
  3. As said in video forward pass has initial values, then Initial val, normalized to get the probability distribution b
  4. backwards reshapes single output, Calculate Jacobian matrix, then calculates the sample gradient and adds it to the gradients array
5. **Using strategic print statements, show and explain the details of what takes place in class `Layer_Dense`**
  - <https://www.loom.com/share/c4578e35125b45f9b37eb33f3e1ca0d3>
  - Initialize weights and biases
  - Weights are set randomly (inputs, neurons avoids having to use transposition at every forward pass)
  - Biases are all set to zero
  - Forward pass
    - Passes `np.dot(inputs, self.weights) + self.biases` to the next layer of the network
  - Backwards pass
    - L1 and L2 functions to reduce generalization error
    - L1 sum of absolute values
    - L2 sum of squared weights and bias
    - Resulting gradient is `np.dot(dvalues, self.weights.T)` stored in `dinput`
6. **EXTRA CREDIT: show the impact of various learning rates on this code. Does changing the learning rate improve the machine learning?**
  - <https://www.loom.com/share/ab374ddd911c46948dec657eb1c01679>
  - Changing the learning rate helps to find the global minimum of the function however to high or to low will cause this minimum to be missed which is why optimizers and other methods are used to adjust the model over time
  - The learning rate needed depends on the rate of change number of epochs and the other optimization methods used