

Shea Durgin & Loic Sine

<https://github.com/sheaDurgin/NLP23/tree/main/Lab6>

My FNN model has a ~73.3% accuracy on the test set.

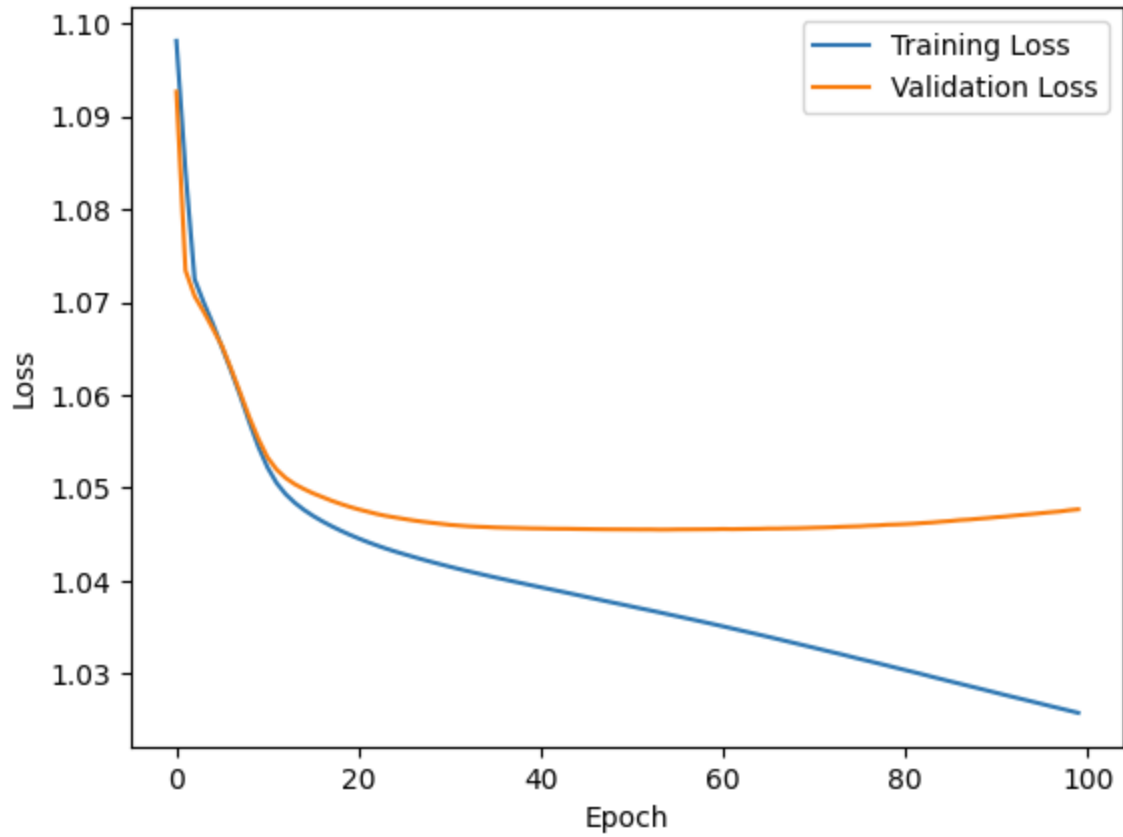
My parameters are as such. The batch size is something I tested with a lot. I ultimately decided on 512 as going above or below would give me worse results. I had 3 hidden layers hidden_dim that went 128 -> 64 -> 32. This is what I mainly tuned at the start and found 3 layers in this setup was giving me good results. My dropout is 0.5, which is in the normal range of 0.2-0.5, it was the most consistently good number. I stuck with CrossEntropyLoss as my research suggested it was the ideal for this type of classification. My optimizer is Adam with a learning rate of 0.0003 and a weight decay of 0.0001. This was the result of a lot of trial and error. Weight decay helped a bit with overfitting (although it is still doing that if you look at the graph) and learning rate is ideal for the amount of epochs I am running. My best epoch is usually in the latter half of the 100 epoch range, I am saving models based on best validation accuracy.

```
# Test the model
y_test_pred = model_module.predict(test_x_tensor)

# Calculate test accuracy
test_accuracy = model_module.calculate_accuracy(test_y_tensor.to(model_module.device), y_test_pred).item()
print(f"Test accuracy: {test_accuracy}")
print(f"best epoch: {count}")
# Test accuracy: 0.7328190803527832
# best epoch: 69
#input_dim = 100
#hidden_dim_1 = 128
#hidden_dim_2 = 64
#hidden_dim_3 = 32
#p = 0.5
#output_dim = 3
#batch size = 512
#optimizer = torch.optim.Adam(self.model.parameters(), lr=0.0003, weight_decay= 0.0001)
```

```
Test accuracy: 0.7328190803527832
best epoch: 69
```

Training and Validation Loss



Training and Validation Accuracy

