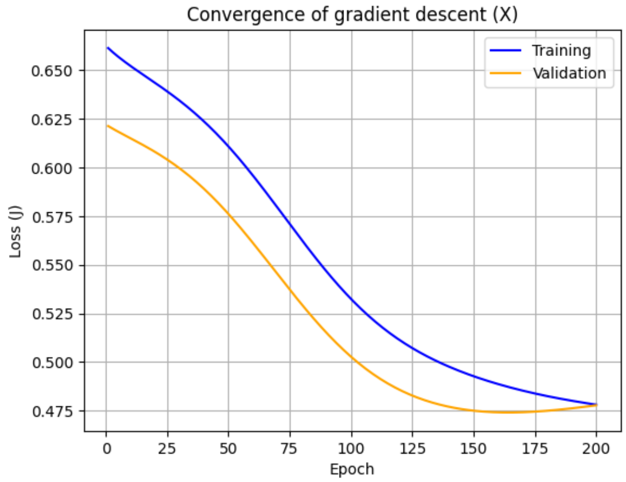
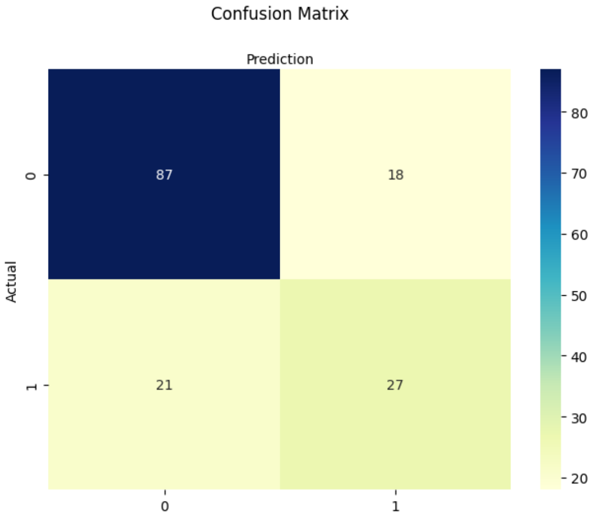
ECGR 5105 Homework 6

Owen Bailey-Waltz (801488178)
GitHub Link: https://github.com/obaileyw-uncc/ecgr5105/tree/main/hw06_neuralnetworks

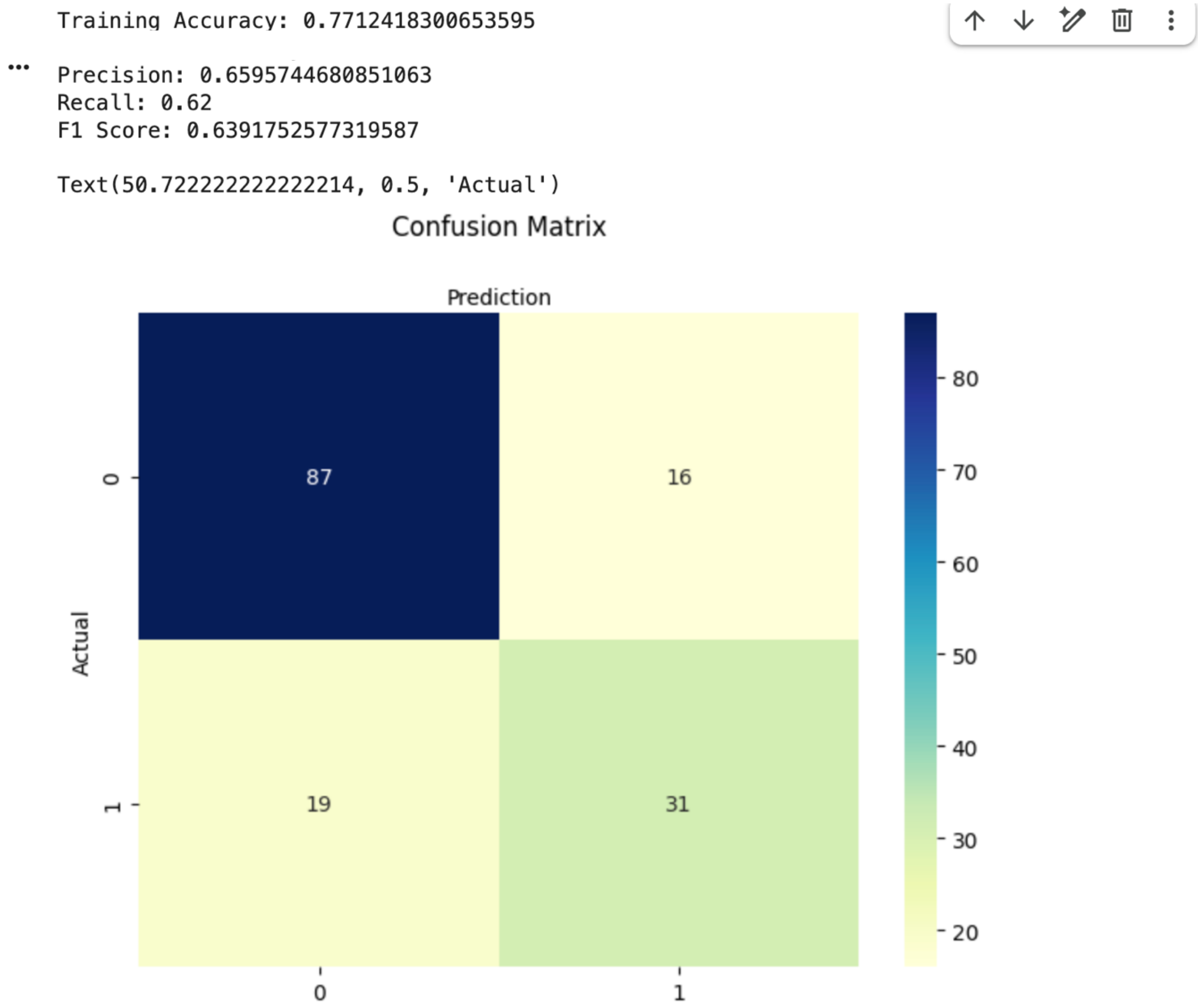
Problem 1: Diabetes dataset





Training Accuracy 0.7789
Validation Accuracy 0.7451
Precision 0.6000
Recall 0.5625
F1 Score 0.5806

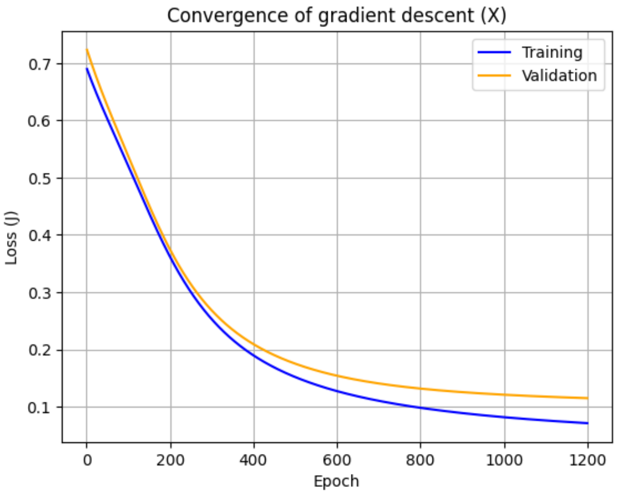
The behavior of stochastic gradient descent leads to a massive spread in the final model convergence. Large networks have a tendency to overfit, making small networks a good choice for this problem. The best performing network that was achieved had a training and validation accuracy of 77.1%, a precision of 66.0%, a sensitivity of 62.0% and an F1 score of 63.9%. The confusion matrix below is from that training before the runtime was reset and the result was lost.

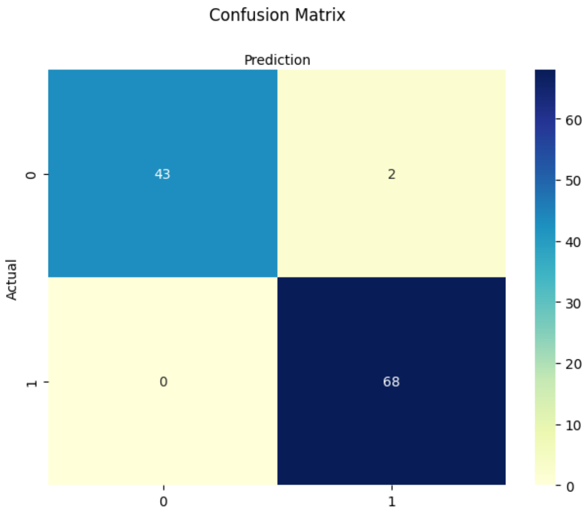


```
Training Accuracy: 0.7712418300653595

Precision: 0.6595744680851063
Recall: 0.62
F1 Score: 0.6391752577319587

Text(50.722222222222214, 0.5, 'Actual')
```

The networks that came to be from stochastic gradient descent using the LogSoftmax output layer largely underperformed the equivalent model counterparts that used vanilla gradient descent with logistic regression. Since the logistic regression example and the SVR classifier example both used APIs from SciKit Learn, we cannot probe the validation loss from these networks. However, the F1 score, which aggregates precision and recall, is 68.2% in the equivalent network trained with logistic regression compared to 58.0% or 63.9% in the neural network examples above. The neural networks achieved higher training accuracy of about 77% in each case compared to 76.22% (see Homework 3), indicating the neural networks are overfitting compared to logistic regression. The diabetes dataset was not used in support vector analysis.

Problem 2: Cancer dataset



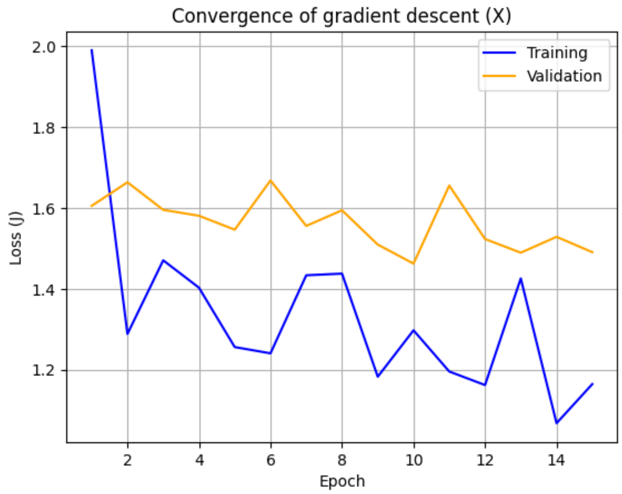


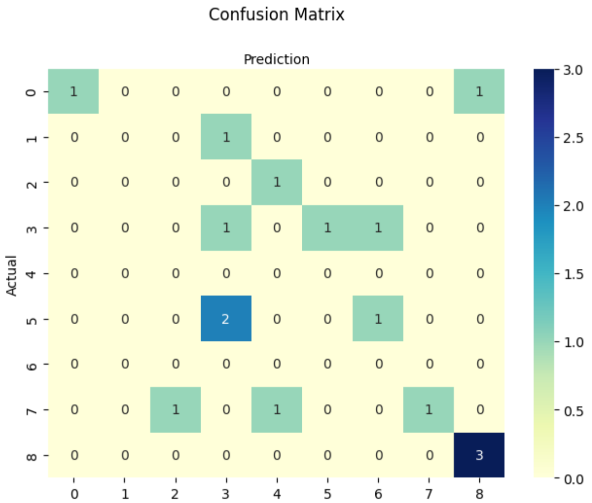
Training Accuracy 0.9824
Validation Accuracy 0.9823
Precision 0.9714
Recall 1.0000
F1 Score 0.9855

For the breast cancer dataset, a neural network-based solution outperforms both logistic regression with vanilla gradient descent and support vector analysis. Once again, the linear classifier and SVR were trained with SciKit Learn and loss is not accessible from outside the API. However, comparing F1 scores to evaluate model quality reveals that the neural network's F1 of 98.55% is significantly greater than the logistic regression F1 of 97.01% and the SVR F1 of 96.30% (see Homeworks 3 and 4). Model sensitivity has also been increased, and all false negatives have been eliminated from the validation set used in the validation of this model. This neural network therefore not only does a better job of generalizing overall, it also would perform better in the hypothetical case that it would actually be used in the early detection of breast cancer since high sensitivity and false positives are more desirable than high precision and false negatives as the latter could lead to more severe cases down the road.

Problem 3: Fully connected neural network for CIFAR-10

(a) One hidden layer, 512 features



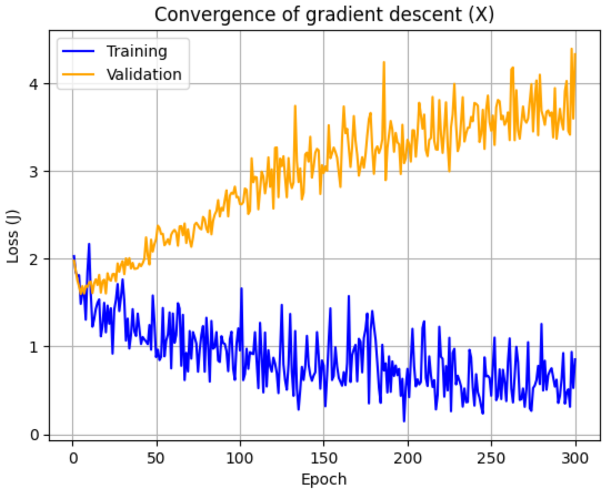


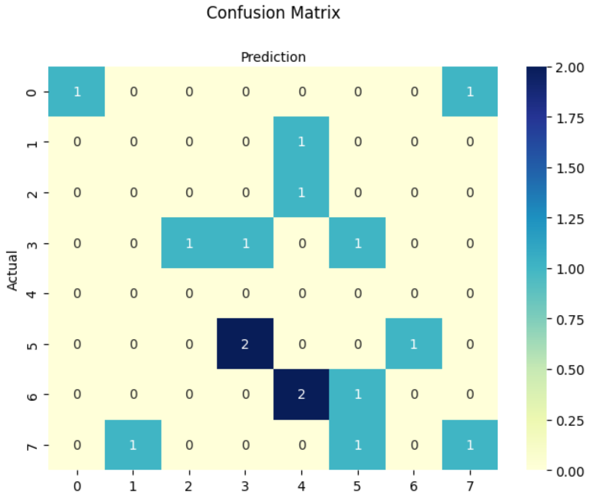
Training Epochs 15
Validation Accuracy 0.4779
Precision 0.3333
Recall 0.2407
F1 Score 0.2566

This shallow classifier began to saturate its accuracy between 15 and 20 iterations. Beyond this iteration count, the classifier begins to overfit as validation loss begins to depart from training loss.

The performance of this classifier is very poor, with only a 47.79% accuracy achieved at the end of training and even worse performance in the areas of precision and recall. Training the model with stochastic gradient descent also involves a large amount of noise on the way to convergence, likely as a result of the effect of the DataLoader’s batching functionality.

(b) Two more hidden layers




Confusion Matrix

	0	1	2	3	4	5	6	7
0	1	0	0	0	0	0	0	1
1	0	0	0	0	1	0	0	0
2	0	0	0	0	1	0	0	0
3	0	0	1	1	0	1	0	0
4	0	0	0	0	0	0	0	0
5	0	0	0	2	0	0	1	0
6	0	0	0	0	2	1	0	0
7	0	1	0	0	0	1	0	1

Training Epochs 300
Validation Accuracy 0.3798
Precision 0.2292
Recall 0.1458
F1 Score 0.1750

A deeper classifier trained for 300 epochs initially reaches approximately the same level of validation loss as the shallow classifier (approx. 1.75), but beyond this level the model begins to saturate and overfit profoundly.

This classifier with its high level of overfitting due to the 300 training epochs performs even worse than the shallow classifier, achieving a 37.98% accuracy by the end of training. Once again, PyTorch DataLoader's batching functionality and the stochastic nature of the loss function is visible as harmonic noise on the convergence trends. The model reached a maximum accuracy of 44.39% at Epoch 32 and a maximum F1 of 35.85% at Epoch 19, neither of which reflect a particularly high model fidelity.