Objective:

A supervised Sequential neural network model was created to compare against the other classification models. Keras open software was leveraged to create, train, and compile the model for the comparison. To prepare the data for the model, the categorical output was one-hot encoded, while the inputs were scaled using the Scikit-Learn MinMax Scaler. For the model, a Sequential neural network was developed using Dense 2D layers and a Softmax activation layer. Optimization was done by maximizing the accuracy of the model using the test data by changing the number of hidden layers, nodes in each layer, the activation function of each layer, the number of epochs, and the optimizer input for the compiler function. It should also be noted for the compiler function, the categorical cross-entropy loss function input was used.

Like with the other models, the same split data was used to train and test the model. The highest accuracy achieved with the trained model against the test data was 0.824. This accuracy was acquired with several different sets of parameters. Accuracy and loss over each epoch were also graphed to help determine if a model with a certain set of parameters was under or overfitted. Overall, with the given data, several sets of parameters generated very similar neural network models. The final model saved and used for the comparison used only 1 hidden layer with the Relu activation function, 50 nodes in said layer, along with 50 epoch cycles and the optimizer function Adam.

Model Combinations tested:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | Layer order / Layer Type / Number of Hidden Layers | Epoch | Optimizer | Test Data Accuracy | Comments |
| 1 | 1 – Relu - 90 | 10 | adam | 0.81 |  |
| 2 | 1 – Relu - 90 | 100 | adam | 0.797 |  |
| 3 | 1 – Relu - 90 | 1000 | adam | 0. 824 |  |
| 4 | 1 – Relu - 90 | 50 | adam | 0.824 |  |
| 5 | 1-Sigmoid - 90 | 50 | adam | 0.824 |  |
| 6 | 1-Sigmoid- 90 | 80 | adam | 0.824 |  |
| 7 | 1-Relu - 90 | 80 | adam | 0.811 |  |
| 8 | 1-Relu -90  2-Relu - 90 | 50 | adam | 0.811 |  |
| 9 | 1-Relu - 90  2-Relu - 90 | 35 | adam | 0.824 |  |
| 10 | 1 – Relu - 10 | 50 | adam | 0.784 |  |
| 11 | 1 – Relu - 10 | 100 | adam | 0.818 |  |
| 12 | 1 – Relu - 20 | 50 | adam | 0.824 |  |
| 13 | 1 – Relu - 50 | 50 | adam | 0.824 |  |
| 14 | 1 – Relu - 100 | 35 | adam | 0.824 |  |
| 15 | 1 – Relu – 100  2 – Relu – 50  3 – Relu - 25 | 35 | adam | 0.824 |  |
| 16 | 1 – Relu – 100  2 – Relu – 100  3 – Relu - 100 | 35 | adam | 0.797 |  |
| 17 | 1 – Relu – 100  2 – Relu – 20  3 – Relu - 2 | 35 | adam | 0.797 |  |
| 19 | 1 – Relu – 100  2 – Relu – 75  3 – Relu - 50 | 35 | adam | 0.811 |  |
| 20 | 1 – Relu – 100  2 – Relu – 75  3 – Relu – 50  4 – Relu - 25 | 35 | adam | 0.797 |  |
| 21 | 1 – Relu – 100  2 – Relu – 50  3 – Relu – 25  4 – Relu - 2 | 35 | adam | 0.824 |  |
| 22 | 1 – Relu – 50 | 50 | rmsprop | 0.818 |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |