

assignmentTask2

January 11, 2023

```
[1]: import pandas as pd
      from sklearn.model_selection import train_test_split
      from sklearn.preprocessing import StandardScaler
      from tensorflow.keras import Sequential
      from tensorflow.keras.layers import Dense
      from tensorflow.keras.optimizers import Adam
```

```
[2]: data = pd.read_csv("BankNote_Authentication.csv")
```

```
[3]: data.head()
```

```
[3]:   variance  skewness  curtosis  entropy  class
0    3.62160    8.6661   -2.8073 -0.44699     0
1    4.54590    8.1674   -2.4586 -1.46210     0
2    3.86600   -2.6383    1.9242  0.10645     0
3    3.45660    9.5228   -4.0112 -3.59440     0
4    0.32924   -4.4552    4.5718 -0.98880     0
```

```
[4]: X = data.drop("class", axis=1)
      y = data["class"]
```

```
[5]: X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2)
```

```
[6]: scaler = StandardScaler()
      X_train = scaler.fit_transform(X_train)
      X_test = scaler.transform(X_test)
```

```
[7]: model = Sequential()
      model.add(Dense(4, input_dim=4, activation="relu"))
      model.add(Dense(4, activation="relu"))
      model.add(Dense(1, activation="sigmoid"))
```

```
[8]: model.compile(loss="binary_crossentropy", optimizer=Adam(),
      ↪metrics=["accuracy"])
```

```
[9]: model.fit(X_train, y_train, epochs=50, batch_size=32, verbose=2)
```

Epoch 1/50

35/35 - 1s - loss: 0.7154 - accuracy: 0.4248 - 900ms/epoch - 26ms/step

Epoch 2/50
35/35 - 0s - loss: 0.6699 - accuracy: 0.5670 - 62ms/epoch - 2ms/step
Epoch 3/50
35/35 - 0s - loss: 0.6324 - accuracy: 0.7010 - 68ms/epoch - 2ms/step
Epoch 4/50
35/35 - 0s - loss: 0.5940 - accuracy: 0.8696 - 59ms/epoch - 2ms/step
Epoch 5/50
35/35 - 0s - loss: 0.5529 - accuracy: 0.9325 - 80ms/epoch - 2ms/step
Epoch 6/50
35/35 - 0s - loss: 0.5091 - accuracy: 0.9508 - 61ms/epoch - 2ms/step
Epoch 7/50
35/35 - 0s - loss: 0.4627 - accuracy: 0.9690 - 56ms/epoch - 2ms/step
Epoch 8/50
35/35 - 0s - loss: 0.4153 - accuracy: 0.9690 - 45ms/epoch - 1ms/step
Epoch 9/50
35/35 - 0s - loss: 0.3687 - accuracy: 0.9736 - 39ms/epoch - 1ms/step
Epoch 10/50
35/35 - 0s - loss: 0.3253 - accuracy: 0.9754 - 38ms/epoch - 1ms/step
Epoch 11/50
35/35 - 0s - loss: 0.2859 - accuracy: 0.9754 - 37ms/epoch - 1ms/step
Epoch 12/50
35/35 - 0s - loss: 0.2514 - accuracy: 0.9790 - 73ms/epoch - 2ms/step
Epoch 13/50
35/35 - 0s - loss: 0.2217 - accuracy: 0.9790 - 68ms/epoch - 2ms/step
Epoch 14/50
35/35 - 0s - loss: 0.1965 - accuracy: 0.9809 - 71ms/epoch - 2ms/step
Epoch 15/50
35/35 - 0s - loss: 0.1750 - accuracy: 0.9809 - 81ms/epoch - 2ms/step
Epoch 16/50
35/35 - 0s - loss: 0.1554 - accuracy: 0.9809 - 79ms/epoch - 2ms/step
Epoch 17/50
35/35 - 0s - loss: 0.1369 - accuracy: 0.9809 - 59ms/epoch - 2ms/step
Epoch 18/50
35/35 - 0s - loss: 0.1204 - accuracy: 0.9809 - 70ms/epoch - 2ms/step
Epoch 19/50
35/35 - 0s - loss: 0.1060 - accuracy: 0.9836 - 78ms/epoch - 2ms/step
Epoch 20/50
35/35 - 0s - loss: 0.0929 - accuracy: 0.9836 - 76ms/epoch - 2ms/step
Epoch 21/50
35/35 - 0s - loss: 0.0828 - accuracy: 0.9836 - 69ms/epoch - 2ms/step
Epoch 22/50
35/35 - 0s - loss: 0.0755 - accuracy: 0.9836 - 67ms/epoch - 2ms/step
Epoch 23/50
35/35 - 0s - loss: 0.0695 - accuracy: 0.9836 - 72ms/epoch - 2ms/step
Epoch 24/50
35/35 - 0s - loss: 0.0647 - accuracy: 0.9836 - 70ms/epoch - 2ms/step
Epoch 25/50
35/35 - 0s - loss: 0.0605 - accuracy: 0.9836 - 64ms/epoch - 2ms/step

Epoch 26/50
35/35 - 0s - loss: 0.0570 - accuracy: 0.9836 - 89ms/epoch - 3ms/step
Epoch 27/50
35/35 - 0s - loss: 0.0538 - accuracy: 0.9836 - 73ms/epoch - 2ms/step
Epoch 28/50
35/35 - 0s - loss: 0.0509 - accuracy: 0.9845 - 68ms/epoch - 2ms/step
Epoch 29/50
35/35 - 0s - loss: 0.0482 - accuracy: 0.9863 - 81ms/epoch - 2ms/step
Epoch 30/50
35/35 - 0s - loss: 0.0459 - accuracy: 0.9845 - 75ms/epoch - 2ms/step
Epoch 31/50
35/35 - 0s - loss: 0.0433 - accuracy: 0.9863 - 69ms/epoch - 2ms/step
Epoch 32/50
35/35 - 0s - loss: 0.0412 - accuracy: 0.9863 - 70ms/epoch - 2ms/step
Epoch 33/50
35/35 - 0s - loss: 0.0393 - accuracy: 0.9863 - 81ms/epoch - 2ms/step
Epoch 34/50
35/35 - 0s - loss: 0.0377 - accuracy: 0.9872 - 79ms/epoch - 2ms/step
Epoch 35/50
35/35 - 0s - loss: 0.0362 - accuracy: 0.9872 - 78ms/epoch - 2ms/step
Epoch 36/50
35/35 - 0s - loss: 0.0345 - accuracy: 0.9872 - 64ms/epoch - 2ms/step
Epoch 37/50
35/35 - 0s - loss: 0.0331 - accuracy: 0.9872 - 57ms/epoch - 2ms/step
Epoch 38/50
35/35 - 0s - loss: 0.0319 - accuracy: 0.9872 - 80ms/epoch - 2ms/step
Epoch 39/50
35/35 - 0s - loss: 0.0308 - accuracy: 0.9872 - 76ms/epoch - 2ms/step
Epoch 40/50
35/35 - 0s - loss: 0.0298 - accuracy: 0.9881 - 68ms/epoch - 2ms/step
Epoch 41/50
35/35 - 0s - loss: 0.0287 - accuracy: 0.9909 - 87ms/epoch - 2ms/step
Epoch 42/50
35/35 - 0s - loss: 0.0279 - accuracy: 0.9936 - 81ms/epoch - 2ms/step
Epoch 43/50
35/35 - 0s - loss: 0.0273 - accuracy: 0.9900 - 59ms/epoch - 2ms/step
Epoch 44/50
35/35 - 0s - loss: 0.0261 - accuracy: 0.9936 - 64ms/epoch - 2ms/step
Epoch 45/50
35/35 - 0s - loss: 0.0253 - accuracy: 0.9936 - 80ms/epoch - 2ms/step
Epoch 46/50
35/35 - 0s - loss: 0.0244 - accuracy: 0.9936 - 75ms/epoch - 2ms/step
Epoch 47/50
35/35 - 0s - loss: 0.0236 - accuracy: 0.9936 - 81ms/epoch - 2ms/step
Epoch 48/50
35/35 - 0s - loss: 0.0229 - accuracy: 0.9936 - 79ms/epoch - 2ms/step
Epoch 49/50
35/35 - 0s - loss: 0.0223 - accuracy: 0.9936 - 65ms/epoch - 2ms/step

Epoch 50/50

35/35 - 0s - loss: 0.0213 - accuracy: 0.9973 - 82ms/epoch - 2ms/step

[9]: <keras.callbacks.History at 0x15a4aed2b50>

```
[10]: score = model.evaluate(X_test, y_test, verbose=0)
print("Test loss: %.2f%%" % (score[0] * 100))
print("Test accuracy: %.2f%%" % (score[1] * 100))
```

Test loss: 3.56%

Test accuracy: 98.91%