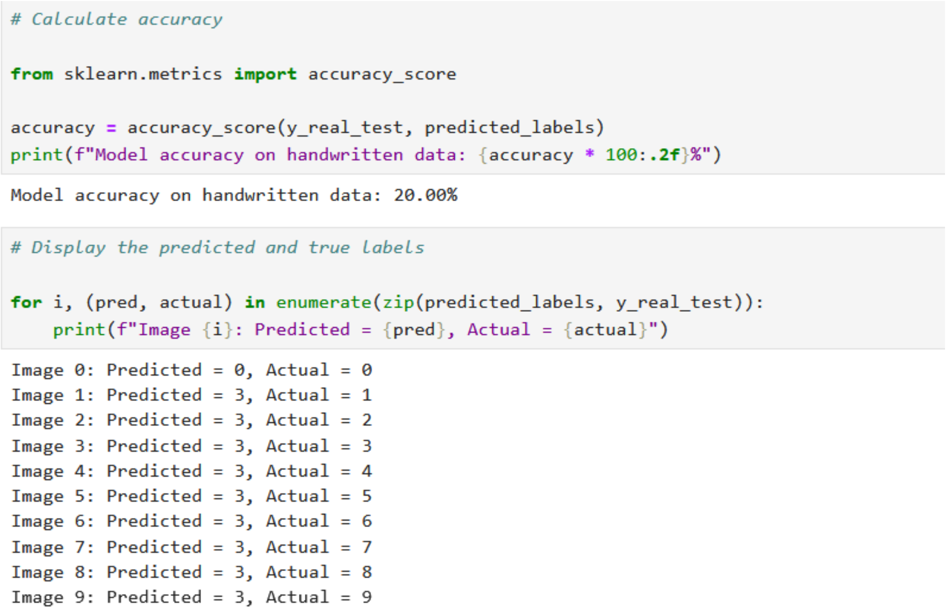
Part 1 – Handwriting Recognition



```
# Calculate accuracy

from sklearn.metrics import accuracy_score

accuracy = accuracy_score(y_real_test, predicted_labels)
print(f"Model accuracy on handwritten data: {accuracy * 100:.2f}%")
```

```
Model accuracy on handwritten data: 20.00%
```

```
# Display the predicted and true labels

for i, (pred, actual) in enumerate(zip(predicted_labels, y_real_test)):
    print(f"Image {i}: Predicted = {pred}, Actual = {actual}")
```

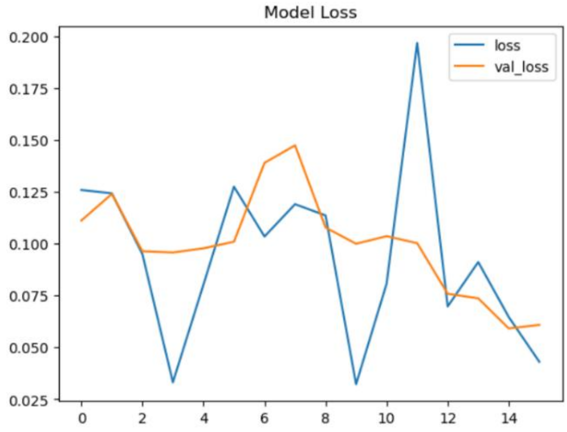
```
Image 0: Predicted = 0, Actual = 0
Image 1: Predicted = 3, Actual = 1
Image 2: Predicted = 3, Actual = 2
Image 3: Predicted = 3, Actual = 3
Image 4: Predicted = 3, Actual = 4
Image 5: Predicted = 3, Actual = 5
Image 6: Predicted = 3, Actual = 6
Image 7: Predicted = 3, Actual = 7
Image 8: Predicted = 3, Actual = 8
Image 9: Predicted = 3, Actual = 9
```

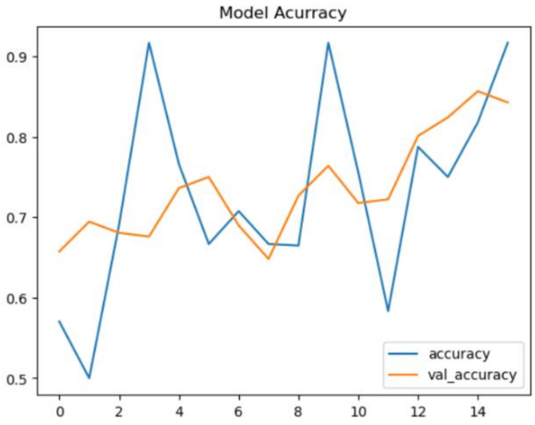
- The model correctly identified two numbers; 0 and 3. Consequently, the accuracy rate of the model was only 20%

Part 2 – Radar Recognition

Using 16 epochs, the results are the following:

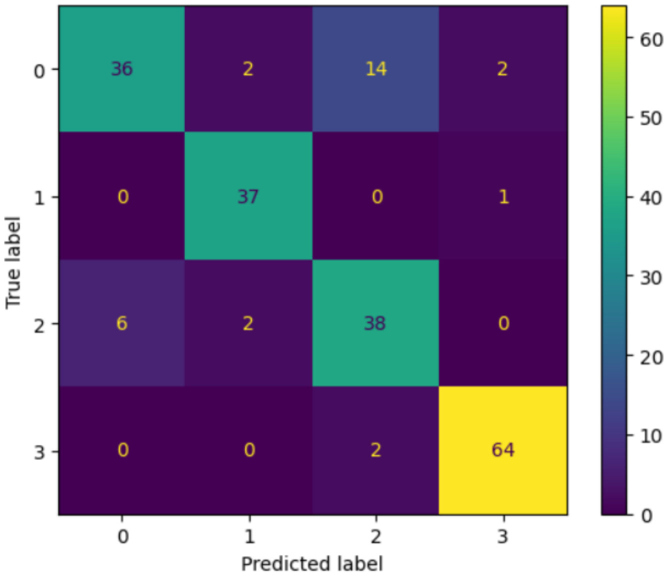
Accuracy	0.91
Val_ accuracy	0.84
Loss	0.04
Val_loss	0.06





- High validation accuracy suggests that the model is performing well on data outside of the training set. The gap between training and validation accuracy suggests the model may be slightly overfitting to the training data, as it performs better on data it has seen but struggles a bit on unseen data. The loss on the training dataset is relatively low, indicating that the model has minimized the error on the training set.

- The loss on the validation dataset is slightly higher than the training loss, which may indicate a potential overfitting issue. However, the difference may not be large enough to support this conclusion.



In the confusion matrix, we can observe that cloudy and rain images have relatively more misclassifications compared to the other two classes. Shine and sunrise images are classified with high accuracy; in fact, there are very few misclassifications for these classes. The confusion matrix highlights that the model struggles the most with distinguishing between "cloudy" and "shine", as well as between "cloudy" and "rain".

Make a proposal for the use of GANs in weather prediction

1. Using sky satellite images of a certain area, GANs could track storms or precipitation patterns.

2. GANs could be trained on historical weather data and then used to simulate future climate scenarios, helping predict how extreme weather events might become more frequent or severe in certain regions.

3. GANs could be trained to process satellite or radar imagery and automatically segment and label different meteorological phenomena (like storms and hurricanes).