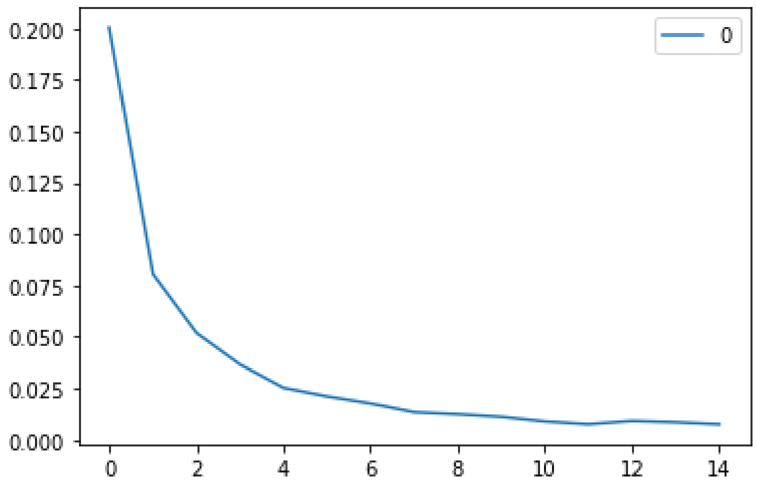
```python
import tensorflow as tf
from tensorflow import keras
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np
from sklearn.metrics import accuracy_score
```

```python
(x_train,y_train),(x_test,y_test) = keras.datasets.mnist.load_data()
x_train = x_train/255
x_test = x_test/255
```

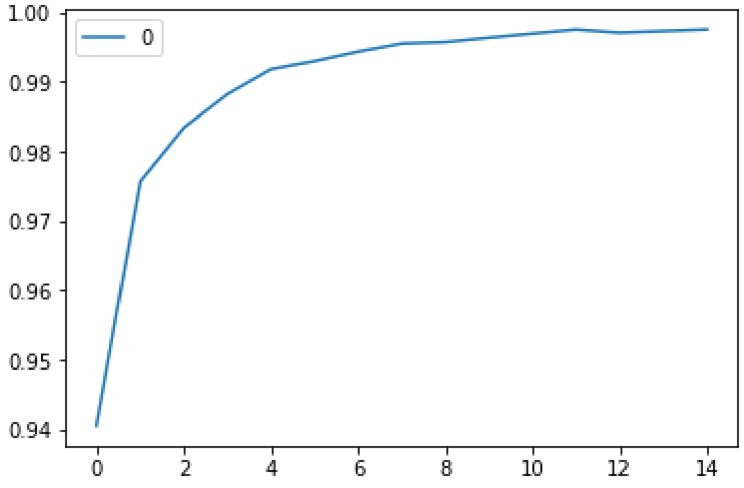
```python
model = tf.keras.Sequential()
model.add(tf.keras.layers.Flatten(input_shape = x_train[0].shape))
model.add(tf.keras.layers.Dense(532,activation = 'relu'))
model.add(tf.keras.layers.Dense(10,activation='softmax'))
model.compile(optimizer ='adam', loss = 'sparse_categorical_crossentropy',metrics = 'accur
model.fit(x_train,y_train,epochs = 15)
```

```
Epoch 1/15
1875/1875 [==============================] - 8s 4ms/step - loss: 0.2004 - accuracy: 0
Epoch 2/15
1875/1875 [==============================] - 9s 5ms/step - loss: 0.0804 - accuracy: 0
Epoch 3/15
1875/1875 [==============================] - 9s 5ms/step - loss: 0.0519 - accuracy: 0
Epoch 4/15
1875/1875 [==============================] - 8s 4ms/step - loss: 0.0366 - accuracy: 0
Epoch 5/15
1875/1875 [==============================] - 8s 4ms/step - loss: 0.0253 - accuracy: 0
Epoch 6/15
1875/1875 [==============================] - 8s 4ms/step - loss: 0.0211 - accuracy: 0
Epoch 7/15
1875/1875 [==============================] - 8s 4ms/step - loss: 0.0177 - accuracy: 0
Epoch 8/15
1875/1875 [==============================] - 8s 4ms/step - loss: 0.0135 - accuracy: 0
Epoch 9/15
1875/1875 [==============================] - 8s 4ms/step - loss: 0.0125 - accuracy: 0
Epoch 10/15
1875/1875 [==============================] - 9s 5ms/step - loss: 0.0112 - accuracy: 0
Epoch 11/15
1875/1875 [==============================] - 9s 5ms/step - loss: 0.0090 - accuracy: 0
Epoch 12/15
1875/1875 [==============================] - 8s 4ms/step - loss: 0.0076 - accuracy: 0
Epoch 13/15
1875/1875 [==============================] - 8s 4ms/step - loss: 0.0092 - accuracy: 0
Epoch 14/15
1875/1875 [==============================] - 8s 4ms/step - loss: 0.0086 - accuracy: 0
Epoch 15/15
1875/1875 [==============================] - 9s 5ms/step - loss: 0.0076 - accuracy: 0
<keras.callbacks.History at 0x7fe060b30290>
```

```python
loss = pd.DataFrame(model.history.history['loss']).plot()
```



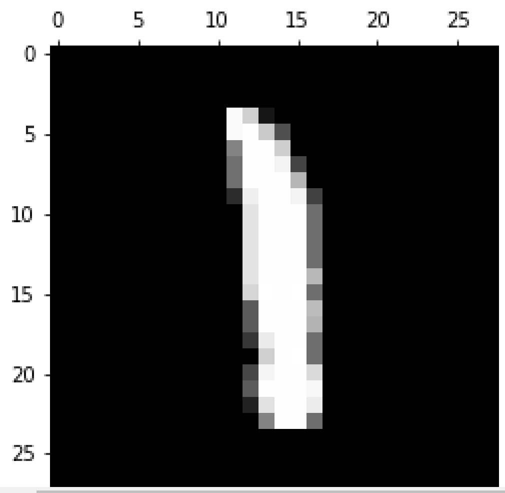
```
acc = pd.DataFrame(model.history.history['accuracy']).plot()
```



```
model.evaluate(x_test, y_test)
plt.matshow(x_test[777],cmap = 'gray')
y_predicted = model.predict(x_test)
y_predicted[777]
n = np.argmax(y_predicted[777])
print("The number is : " ,n)

model.get_weights()
model.save('Handwritten Recognition.hdf5')
```

```
313/313 [==============================] - 1s 3ms/step - loss: 0.1002 - accuracy: 0.9
The number is :  1
```



✓ 2s completed at 1:36 PM