Name: Sumanasekara W.K.G.G.
Index: 190610E

In [ ]:

```
import numpy as np
import tensorflow as tf
from tensorflow import keras
from keras import datasets, layers, models
```

In [ ]:

```
(train_images, train_labels), (test_images, test_labels) = datasets.mnist.load_data()
class_names = [str(i) for i in range(10)]

paddings = tf.constant(((0, 0), (2, 2), (2, 2)))
train_images = tf.pad(train_images, paddings, constant_values=0)
test_images = tf.pad(test_images, paddings, constant_values=0)

print("train_images.shape:,", train_images.shape)
print("train_labels.shape:", train_labels.shape)
print("test_images.shape:", test_images.shape)
print("test_labels.shape:", test_labels.shape)

train_images = tf.dtypes.cast(train_images, tf.float32)
test_images = tf.dtypes.cast(test_images, tf.float32)
train_images, test_images = train_images[..., np.newaxis]/255.0, test_images[..., np.r
```

```
train_images.shape:, (60000, 32, 32)
train_labels.shape: (60000,)
test_images.shape: (10000, 32, 32)
test_labels.shape: (10000,)
```

In [ ]:

```
model = models.Sequential()
model.add(layers.Convolution2D(6, (5, 5), activation='relu', input_shape=(32, 32,1)))
model.add(layers.AveragePooling2D((2, 2)))
model.add(layers.Conv2D(16, (5, 5), activation='relu'))
model.add(layers.AveragePooling2D((2, 2)))
model.add(layers.Flatten())
model.add(layers.Dense(120, activation='relu'))
model.add(layers.Dense(84, activation='relu'))
model.add(layers.Dense(10))

model.compile(optimizer=keras.optimizers.Adam(), loss=keras.losses.SparseCategoricalCr
print(model.summary())
```

```
Model: "sequential_15"
_________________________________________________________________
 Layer (type)                Output Shape              Param #   
=================================================================
 conv2d_26 (Conv2D)          (None, 28, 28, 6)         156       
                                                                 
 average_pooling2d_18 (Avera  (None, 14, 14, 6)        0         
 gePooling2D)                                                    
                                                                 
 conv2d_27 (Conv2D)          (None, 10, 10, 16)        2416      
                                                                 
 average_pooling2d_19 (Avera  (None, 5, 5, 16)         0         
 gePooling2D)                                                    
                                                                 
 flatten_7 (Flatten)         (None, 400)               0         
                                                                 
 dense_16 (Dense)            (None, 120)               48120     
                                                                 
 dense_17 (Dense)            (None, 84)                10164     
                                                                 
 dense_18 (Dense)            (None, 10)                850       
                                                                 
=================================================================
Total params: 61,706
Trainable params: 61,706
Non-trainable params: 0
_________________________________________________________________
None
```

In [ ]:

```
(train_images, train_labels), (test_images, test_labels) = datasets.cifar10.load_data(
class_names = ['airplane', 'automobile', 'bird', 'cat', 'deer', 'dog', 'frog', 'horse'

train_images, test_images = train_images / 255.0, test_images / 255.0

print("train_images.shape:,", train_images.shape)
print("train_labels.shape:", train_labels.shape)
print("test_images.shape:", test_images.shape)
print("test_labels.shape:", test_labels.shape)
```

```
train_images.shape:, (50000, 32, 32, 3)
train_labels.shape: (50000, 1)
test_images.shape: (10000, 32, 32, 3)
test_labels.shape: (10000, 1)
```

In [ ]:

```
model = models.Sequential()
model.add(layers.Conv2D(32, (5, 5), activation='relu', input_shape=(32, 32, 3)))
model.add(layers.MaxPool2D((2, 2)))
model.add(layers.Conv2D(64, (3, 3), activation='relu'))
model.add(layers.MaxPool2D((2, 2)))
model.add(layers.Conv2D(128, (3, 3), activation='relu'))
model.add(layers.Flatten())
model.add(layers.Dense(64, activation='relu'))
model.add(layers.Dense(10))

model.compile(optimizer=keras.optimizers.Adam(learning_rate=0.001), loss=keras.losses.
print(model.summary())
```

```
Model: "sequential_17"
_________________________________________________________________
 Layer (type)                Output Shape              Param #   
=================================================================
 conv2d_31 (Conv2D)          (None, 28, 28, 32)        2432      
                                                                 
 max_pooling2d_6 (MaxPooling  (None, 14, 14, 32)       0         
 2D)                                                             
                                                                 
 conv2d_32 (Conv2D)          (None, 12, 12, 64)        18496     
                                                                 
 max_pooling2d_7 (MaxPooling  (None, 6, 6, 64)         0         
 2D)                                                             
                                                                 
 conv2d_33 (Conv2D)          (None, 4, 4, 128)         73856     
                                                                 
 flatten_9 (Flatten)         (None, 2048)              0         
                                                                 
 dense_21 (Dense)            (None, 64)                131136    
                                                                 
 dense_22 (Dense)            (None, 10)                650       
                                                                 
=================================================================
Total params: 226,570
Trainable params: 226,570
Non-trainable params: 0
_________________________________________________________________
None
```

In [ ]:
```python
model.fit(train_images, train_labels, epochs=5)
test_loss, test_acc = model.evaluate(test_images, test_labels, verbose=2)
```

```
Epoch 1/5
1563/1563 [==============================] - 48s 31ms/step - loss: 1.5140 - accuracy:
0.4488
Epoch 2/5
1563/1563 [==============================] - 51s 33ms/step - loss: 1.1575 - accuracy:
0.5914
Epoch 3/5
1563/1563 [==============================] - 44s 28ms/step - loss: 0.9839 - accuracy:
0.6542
Epoch 4/5
1563/1563 [==============================] - 44s 28ms/step - loss: 0.8724 - accuracy:
0.6929
Epoch 5/5
1563/1563 [==============================] - 46s 30ms/step - loss: 0.7826 - accuracy:
0.7265
313/313 - 2s - loss: 0.8982 - accuracy: 0.6881 - 2s/epoch - 8ms/step
```

In [ ]:
```python
(train_images, train_labels), (test_images, test_labels) = datasets.mnist.load_data()
class_names = [str(i) for i in range(10)]

paddings = tf.constant(((0, 0), (2, 2), (2, 2)))
train_images = tf.pad(train_images, paddings, constant_values=0)
test_images = tf.pad(test_images, paddings, constant_values=0)

print("train_images.shape:,", train_images.shape)
print("train_labels.shape:", train_labels.shape)
print("test_images.shape:", test_images.shape)
print("test_labels.shape:", test_labels.shape)
```

```
train_images = tf.dtypes.cast(train_images, tf.float32)
test_images = tf.dtypes.cast(test_images, tf.float32)
train_images, test_images = train_images[..., np.newaxis]/255.0, test_images[..., np.r

model = models.Sequential()
model.add(layers.Convolution2D(32, (3, 3), activation='relu', input_shape=(32, 32,1)))
model.add(layers.MaxPool2D((2, 2)))
model.add(layers.Conv2D(64, (3, 3), activation='relu'))
model.add(layers.MaxPool2D((2, 2)))
model.add(layers.Conv2D(64, (3, 3), activation='relu'))
model.add(layers.Flatten())
model.add(layers.Dense(64, activation='relu'))
model.add(layers.Dense(10))

model.compile(optimizer=keras.optimizers.Adam(), loss=keras.losses.SparseCategoricalCr
print(model.summary())

model.fit(train_images, train_labels, epochs=2)
test_loss, test_acc = model.evaluate(test_images, test_labels, verbose=2)
model.save_weights('saved_weights/')
```

```
train_images.shape:, (60000, 32, 32)
train_labels.shape: (60000,)
test_images.shape: (10000, 32, 32)
test_labels.shape: (10000,)
Model: "sequential_18"
_________________________________________________________________
 Layer (type)                Output Shape              Param #   
=================================================================
 conv2d_34 (Conv2D)          (None, 30, 30, 32)        320       
                                                                 
 max_pooling2d_8 (MaxPooling  (None, 15, 15, 32)       0         
 2D)                                                             
                                                                 
 conv2d_35 (Conv2D)          (None, 13, 13, 64)        18496     
                                                                 
 max_pooling2d_9 (MaxPooling  (None, 6, 6, 64)         0         
 2D)                                                             
                                                                 
 conv2d_36 (Conv2D)          (None, 4, 4, 64)          36928     
                                                                 
 flatten_10 (Flatten)        (None, 1024)              0         
                                                                 
 dense_23 (Dense)            (None, 64)                65600     
                                                                 
 dense_24 (Dense)            (None, 10)                650       
                                                                 
=================================================================
Total params: 121,994
Trainable params: 121,994
Non-trainable params: 0
_________________________________________________________________
None
Epoch 1/2
1875/1875 [==============================] - 46s 24ms/step - loss: 0.1306 - accuracy:
0.9595
Epoch 2/2
1875/1875 [==============================] - 47s 25ms/step - loss: 0.0407 - accuracy:
0.9869
313/313 - 2s - loss: 0.0295 - accuracy: 0.9905 - 2s/epoch - 7ms/step
```

In [ ]:

```
model_lw = models.Sequential()
model_lw.add(layers.Convolution2D(32, (3, 3), activation='relu', input_shape=(32, 32,1
model_lw.add(layers.MaxPool2D((2, 2)))
model_lw.add(layers.Conv2D(64, (3, 3), activation='relu'))
model_lw.add(layers.MaxPool2D((2, 2)))
model_lw.add(layers.Conv2D(64, (3, 3), activation='relu'))
model_lw.add(layers.Flatten())
model_lw.add(layers.Dense(64, activation='relu'))
model_lw.add(layers.Dense(10))

model_lw.compile(optimizer=keras.optimizers.Adam(), loss=keras.losses.SparseCategorica
print(model_lw.summary())

model_lw.load_weights('saved_weights/')

model_lw.fit(train_images, train_labels, epochs=2)
test_loss, test_acc = model_lw.evaluate(test_images, test_labels, verbose=2)

model_lw.save('saved_model/')
```

```
Model: "sequential_21"
_________________________________________________________________
 Layer (type)                Output Shape              Param #   
=================================================================
 conv2d_43 (Conv2D)          (None, 30, 30, 32)        320       
                                                                 
 max_pooling2d_14 (MaxPoolin  (None, 15, 15, 32)       0         
 g2D)                                                            
                                                                 
 conv2d_44 (Conv2D)          (None, 13, 13, 64)        18496     
                                                                 
 max_pooling2d_15 (MaxPoolin  (None, 6, 6, 64)         0         
 g2D)                                                            
                                                                 
 conv2d_45 (Conv2D)          (None, 4, 4, 64)          36928     
                                                                 
 flatten_13 (Flatten)        (None, 1024)              0         
                                                                 
 dense_29 (Dense)            (None, 64)                65600     
                                                                 
 dense_30 (Dense)            (None, 10)                650       
                                                                 
=================================================================
Total params: 121,994
Trainable params: 121,994
Non-trainable params: 0
_________________________________________________________________
None
Epoch 1/2
1875/1875 [==============================] - 46s 24ms/step - loss: 0.0303 - accuracy: 
0.9904
Epoch 2/2
1875/1875 [==============================] - 44s 23ms/step - loss: 0.0228 - accuracy: 
0.9933
313/313 - 3s - loss: 0.0286 - accuracy: 0.9917 - 3s/epoch - 8ms/step
INFO:tensorflow:Assets written to: saved_model/assets
```

In [ ]:

```
model_ld = models.load_model('saved_model/')
print(model_ld.summary())
```

```
Model: "sequential_21"
_________________________________________________________________
 Layer (type)                Output Shape              Param #   
=================================================================
 conv2d_43 (Conv2D)          (None, 30, 30, 32)        320       
                                                                 
 max_pooling2d_14 (MaxPoolin  (None, 15, 15, 32)       0         
 g2D)                                                            
                                                                 
 conv2d_44 (Conv2D)          (None, 13, 13, 64)        18496     
                                                                 
 max_pooling2d_15 (MaxPoolin  (None, 6, 6, 64)         0         
 g2D)                                                            
                                                                 
 conv2d_45 (Conv2D)          (None, 4, 4, 64)          36928     
                                                                 
 flatten_13 (Flatten)        (None, 1024)              0         
                                                                 
 dense_29 (Dense)            (None, 64)                65600     
                                                                 
 dense_30 (Dense)            (None, 10)                650       
                                                                 
=================================================================
Total params: 121,994
Trainable params: 121,994
Non-trainable params: 0
_________________________________________________________________
None
```

In [ ]:

```
base_inputs = model_ld.layers[0].input
base_outputs = model_ld.layers[-2].output
output = layers.Dense(10)(base_outputs)

new_model = keras.Model(inputs=base_inputs, outputs=output)
new_model.compile(optimizer=keras.optimizers.Adam(), loss=tf.losses.SparseCategorical(
print(new_model.summary())

new_model.fit(train_images, train_labels, epochs=3, verbose=2)
test_loss, test_acc = model_lw.evaluate(test_images, test_labels, verbose=2)
```

```
Model: "model_4"
_________________________________________________________________
 Layer (type)                Output Shape              Param #   
=================================================================
 conv2d_43_input (InputLayer  [(None, 32, 32, 1)]      0         
 )                                                               
                                                                 
 conv2d_43 (Conv2D)          (None, 30, 30, 32)        320       
                                                                 
 max_pooling2d_14 (MaxPoolin  (None, 15, 15, 32)       0         
 g2D)                                                            
                                                                 
 conv2d_44 (Conv2D)          (None, 13, 13, 64)        18496     
                                                                 
 max_pooling2d_15 (MaxPoolin  (None, 6, 6, 64)         0         
 g2D)                                                            
                                                                 
 conv2d_45 (Conv2D)          (None, 4, 4, 64)          36928     
                                                                 
 flatten_13 (Flatten)        (None, 1024)              0         
                                                                 
 dense_29 (Dense)            (None, 64)                65600     
                                                                 
 dense_39 (Dense)            (None, 10)                650       
                                                                 
=================================================================
Total params: 121,994
Trainable params: 121,994
Non-trainable params: 0
_________________________________________________________________
None
Epoch 1/3
1875/1875 - 43s - loss: 0.0997 - accuracy: 0.9747 - 43s/epoch - 23ms/step
Epoch 2/3
1875/1875 - 44s - loss: 0.0140 - accuracy: 0.9955 - 44s/epoch - 24ms/step
Epoch 3/3
1875/1875 - 42s - loss: 0.0102 - accuracy: 0.9967 - 42s/epoch - 22ms/step
313/313 - 2s - loss: 0.0286 - accuracy: 0.9917 - 2s/epoch - 7ms/step
```

In [ ]:

```
model_for_tl = models.load_model('saved_model/')
model_for_tl.trainable = False
for layer in model_for_tl.layers:
    assert layer.trainable == False

base_inputs = model_for_tl.layers[0].input
base_outputs = model_for_tl.layers[-2].output
output = layers.Dense(10)(base_outputs)

new_model = keras.Model(inputs=base_inputs, outputs=output)
new_model.compile(optimizer=keras.optimizers.Adam(), loss=tf.losses.SparseCategoricalC
print(new_model.summary())

new_model.fit(train_images, train_labels, epochs=3, verbose=2)
test_loss, test_acc = model_lw.evaluate(test_images, test_labels, verbose=2)
```

```
Model: "model_5"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d_43_input (InputLayer  [(None, 32, 32, 1)]      0
 )

 conv2d_43 (Conv2D)          (None, 30, 30, 32)        320

 max_pooling2d_14 (MaxPoolin  (None, 15, 15, 32)       0
 g2D)

 conv2d_44 (Conv2D)          (None, 13, 13, 64)        18496

 max_pooling2d_15 (MaxPoolin  (None, 6, 6, 64)         0
 g2D)

 conv2d_45 (Conv2D)          (None, 4, 4, 64)          36928

 flatten_13 (Flatten)        (None, 1024)              0

 dense_29 (Dense)            (None, 64)                65600

 dense_40 (Dense)            (None, 10)                650

=================================================================
Total params: 121,994
Trainable params: 650
Non-trainable params: 121,344
_________________________________________________________________
None
Epoch 1/3
1875/1875 - 12s - loss: 0.1938 - accuracy: 0.9566 - 12s/epoch - 6ms/step
Epoch 2/3
1875/1875 - 12s - loss: 0.0146 - accuracy: 0.9961 - 12s/epoch - 6ms/step
Epoch 3/3
1875/1875 - 12s - loss: 0.0106 - accuracy: 0.9969 - 12s/epoch - 6ms/step
313/313 - 2s - loss: 0.0286 - accuracy: 0.9917 - 2s/epoch - 7ms/step
```

In [ ]:

```
model_rs = models.Sequential()
model_rs.add(keras.applications.ResNet50V2(include_top=True, weights='imagenet', input
                                            input_shape=None, pooling=None, classes=10
                                            classifier_activation='softmax'))

model_rs.trainable = False
for layer in model_rs.layers:
    assert layer.trainable == False
```