

# NEURAL NETWORK DEEP LEARNING

## ICP 5

700765227

MANDHA SAIKUMAR REDDY

### GitHub:

Repository URL for the source code:


<https://github.com/saikumarreddyMandha/Assignment-5>

### Video Link:

<https://drive.google.com/file/d/1uOCgsNDBP-aF0-WUygu8cGIJtTSFnzlx/view?usp=sharing>

### Output:

```
[ ] sgd = SGD(learning_rate=0.01, momentum=0.9, decay=1e-6)
model.compile(loss='categorical_crossentropy', optimizer=sgd, metrics=['accuracy'])
print(model.summary())
```

 /usr/local/lib/python3.10/dist-packages/keras/src/optimizers/base\_optimizer.py:33: UserWarning: Argument `decay` is no longer warnings.warn(  
Model: "sequential\_1"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 32, 32, 32)	896
dropout (Dropout)	(None, 32, 32, 32)	0
conv2d_1 (Conv2D)	(None, 32, 32, 32)	9,248
max_pooling2d (MaxPooling2D)	(None, 16, 16, 32)	0
flatten (Flatten)	(None, 8192)	0
dense (Dense)	(None, 512)	4,194,816
dropout_1 (Dropout)	(None, 512)	0
dense_1 (Dense)	(None, 10)	5,130

Total params: 4,210,090 (16.06 MB)  
Trainable params: 4,210,090 (16.06 MB)  
Non-trainable params: 0 (0.00 B)  
None

```
[ ] epochs = 5
    batch_size = 32
    model.fit(X_train, y_train, validation_data=(X_test, y_test), epochs=epochs, batch_size=batch_size)
```

```
Epoch 1/5
1563/1563 ————— 16s 7ms/step - accuracy: 0.2960 - loss: 1.9382 - val_accuracy: 0.4212 - val_loss: 1.6107
Epoch 2/5
1563/1563 ————— 6s 4ms/step - accuracy: 0.4703 - loss: 1.4611 - val_accuracy: 0.5444 - val_loss: 1.2595
Epoch 3/5
1563/1563 ————— 10s 4ms/step - accuracy: 0.5485 - loss: 1.2613 - val_accuracy: 0.5737 - val_loss: 1.1901
Epoch 4/5
1563/1563 ————— 6s 4ms/step - accuracy: 0.6093 - loss: 1.0997 - val_accuracy: 0.5973 - val_loss: 1.1197
Epoch 5/5
1563/1563 ————— 7s 4ms/step - accuracy: 0.6545 - loss: 0.9746 - val_accuracy: 0.6289 - val_loss: 1.0521
<keras.src.callbacks.history.History at 0x7d94bfb10a00>
```

```
] scores = model.evaluate(X_test, y_test, verbose=0)
print("Accuracy: %.2f%%" % (scores[1]*100))
```

```
Accuracy: 62.89%
```

```
Model: "sequential_3"
```

Layer (type)	Output Shape	Param #
conv2d_18 (Conv2D)	(None, 32, 32, 32)	896
dropout_18 (Dropout)	(None, 32, 32, 32)	0
conv2d_19 (Conv2D)	(None, 32, 32, 32)	9,248
max_pooling2d_9 (MaxPooling2D)	(None, 16, 16, 32)	0
conv2d_20 (Conv2D)	(None, 16, 16, 64)	18,496
dropout_19 (Dropout)	(None, 16, 16, 64)	0
conv2d_21 (Conv2D)	(None, 16, 16, 64)	36,928
max_pooling2d_10 (MaxPooling2D)	(None, 8, 8, 64)	0
conv2d_22 (Conv2D)	(None, 8, 8, 128)	73,856
dropout_20 (Dropout)	(None, 8, 8, 128)	0
conv2d_23 (Conv2D)	(None, 8, 8, 128)	147,584
max_pooling2d_11 (MaxPooling2D)	(None, 4, 4, 128)	0
flatten_3 (Flatten)	(None, 2048)	0
dropout_21 (Dropout)	(None, 2048)	0
dense_9 (Dense)	(None, 1024)	2,098,176
dropout_22 (Dropout)	(None, 1024)	0

[8]



flatten_3 (Flatten)	(None, 2048)	0
dropout_21 (Dropout)	(None, 2048)	0
dense_9 (Dense)	(None, 1024)	2,098,176
dropout_22 (Dropout)	(None, 1024)	0
dense_10 (Dense)	(None, 512)	524,800
dropout_23 (Dropout)	(None, 512)	0
dense_11 (Dense)	(None, 10)	5,130

Total params: 2,915,114 (11.12 MB)  
Trainable params: 2,915,114 (11.12 MB)  
Non-trainable params: 0 (0.00 B)

None

Epoch 1/5  
1563/1563 ————— 23s 10ms/step - accuracy: 0.2243 - loss: 2.0757 - val\_accuracy: 0.3266 - val\_loss: 1.9049  
Epoch 2/5  
1563/1563 ————— 10s 7ms/step - accuracy: 0.4546 - loss: 1.4943 - val\_accuracy: 0.5180 - val\_loss: 1.3072  
Epoch 3/5  
1563/1563 ————— 21s 7ms/step - accuracy: 0.5512 - loss: 1.2456 - val\_accuracy: 0.6043 - val\_loss: 1.1006  
Epoch 4/5  
1563/1563 ————— 10s 6ms/step - accuracy: 0.6202 - loss: 1.0719 - val\_accuracy: 0.6628 - val\_loss: 0.9562  
Epoch 5/5  
1563/1563 ————— 10s 7ms/step - accuracy: 0.6639 - loss: 0.9532 - val\_accuracy: 0.6895 - val\_loss: 0.8870  
Accuracy: 68.95%

```
# Print the predicted and actual labels for the first 4 images
print("Predicted labels:", predicted_labels)
print("Actual labels:  ", actual_labels)
```



1/1 ————— 0s 324ms/step  
Predicted labels: [3 8 8 0]  
Actual labels: [3 8 8 0]

