Adversarial Attack- FGSM

CIFAR10 & MNIST

By: Abdul Momin

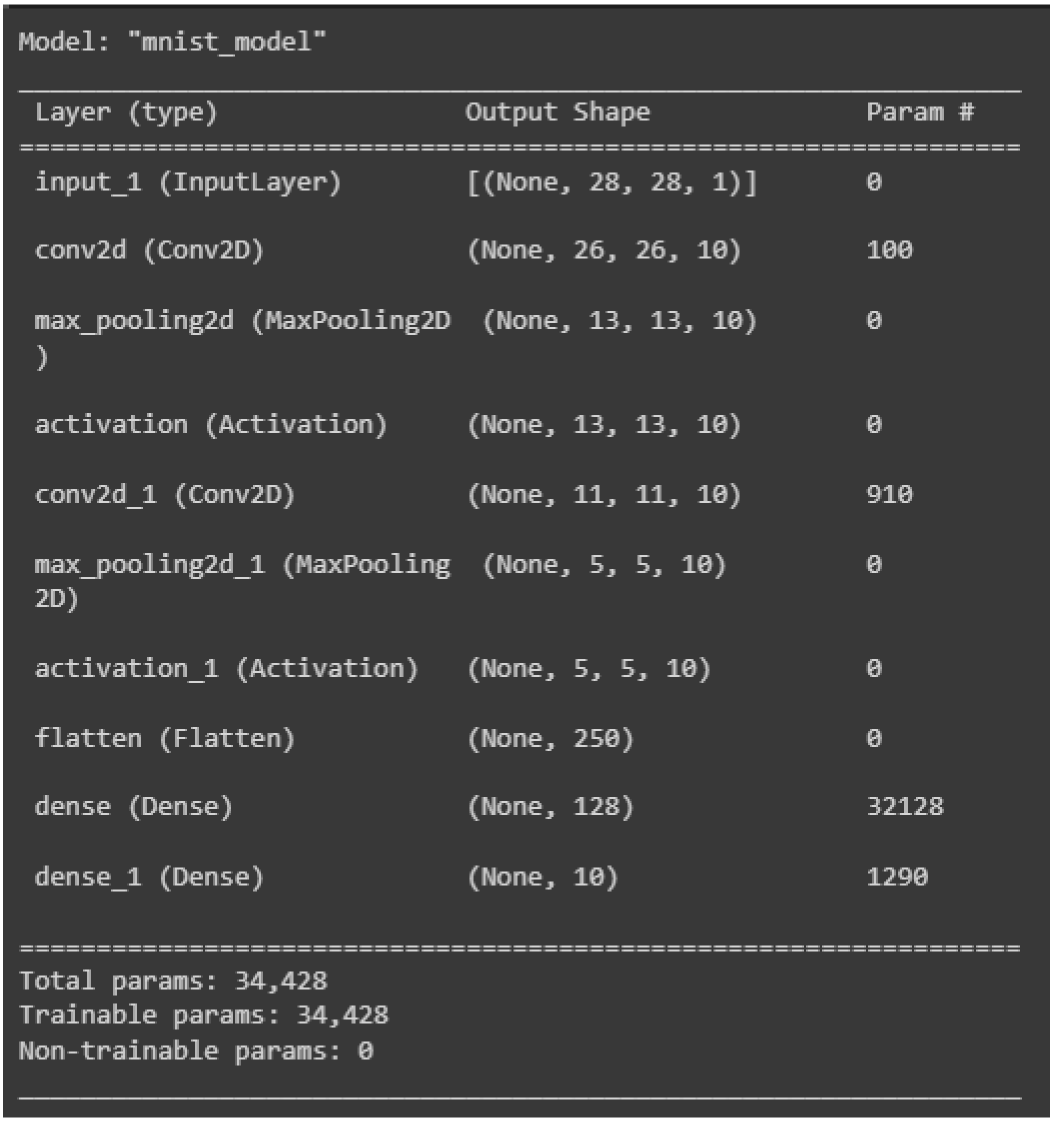
Sunday, 30 July 2023

GitHub Link

Contents

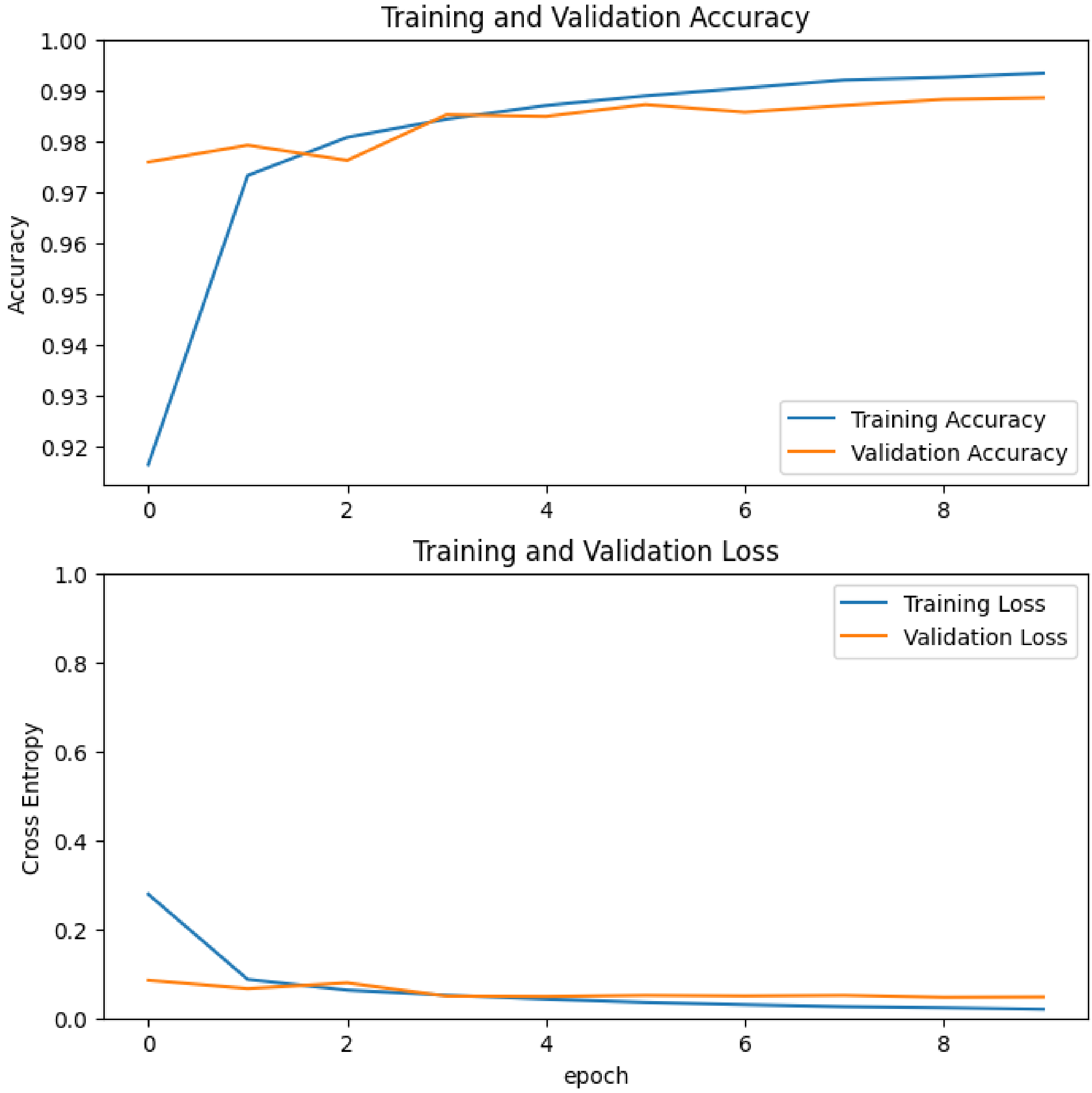
MNIST

Model

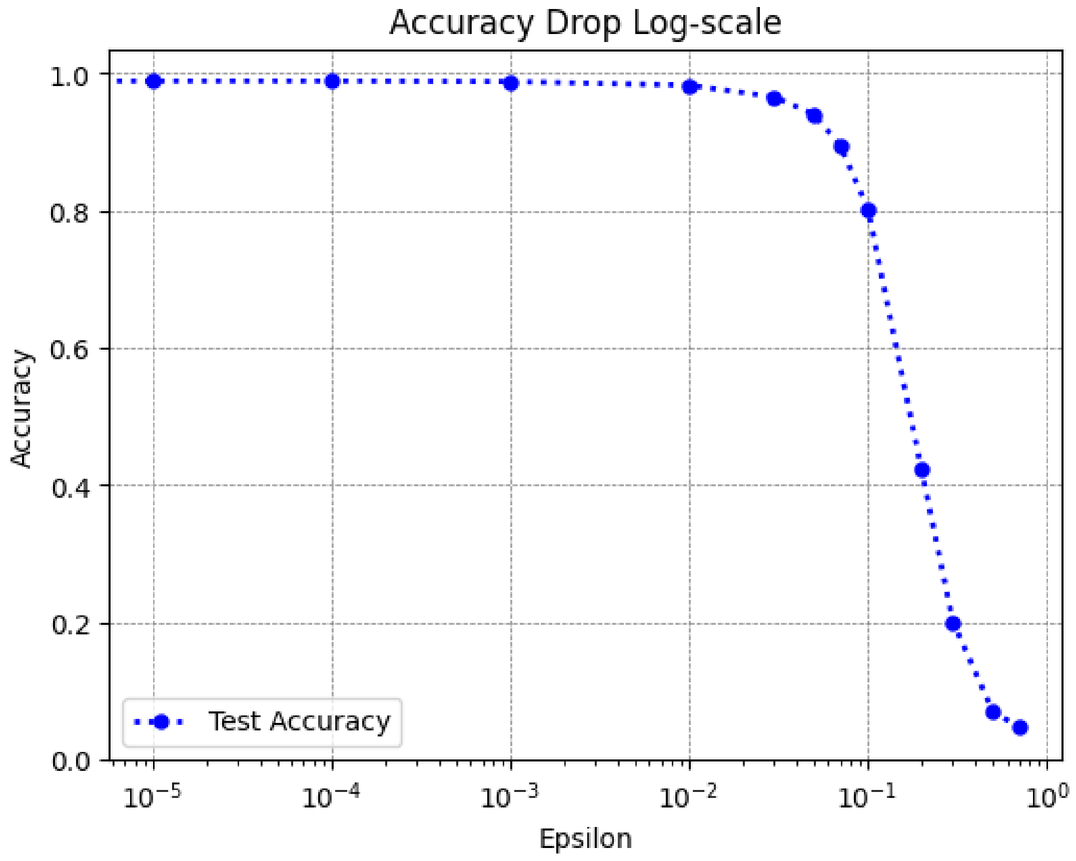


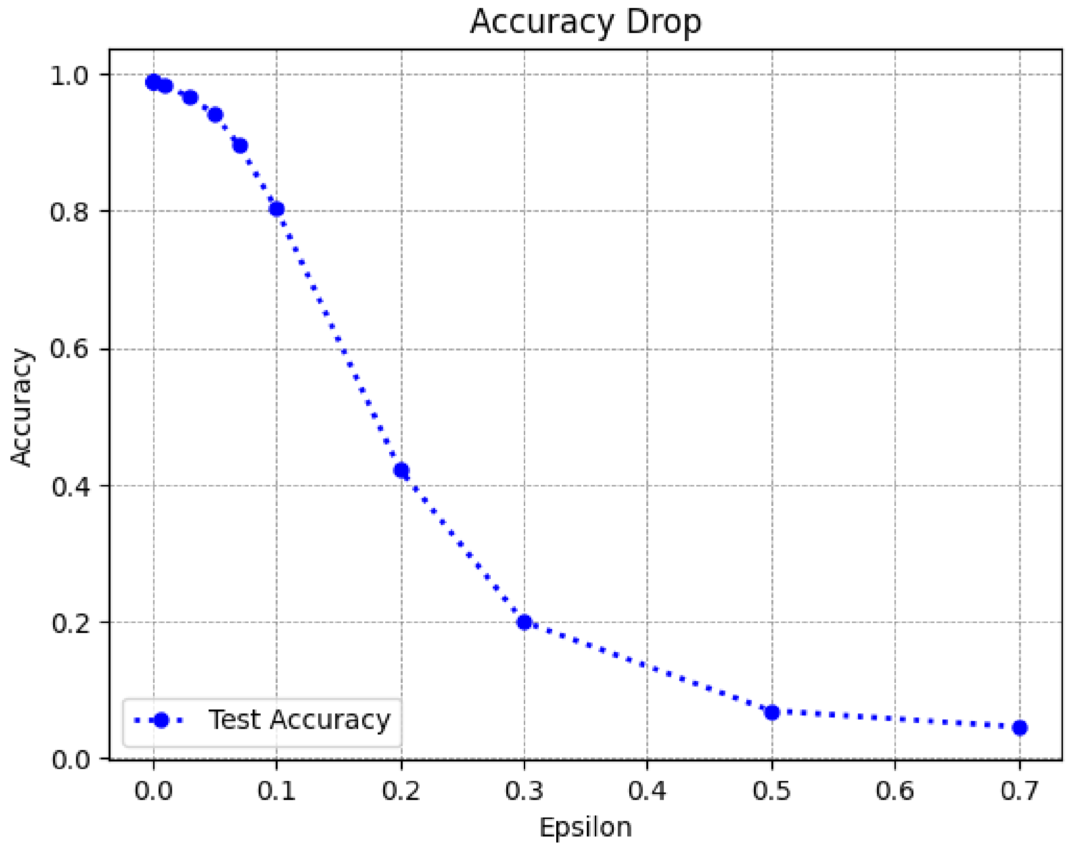
```
Model: "mnist_model"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 input_1 (InputLayer)        [(None, 28, 28, 1)]       0

 conv2d (Conv2D)             (None, 26, 26, 10)        100

 max_pooling2d (MaxPooling2D  (None, 13, 13, 10)       0
 )

 activation (Activation)     (None, 13, 13, 10)        0

 conv2d_1 (Conv2D)           (None, 11, 11, 10)        910

 max_pooling2d_1 (MaxPooling  (None, 5, 5, 10)         0
 2D)

 activation_1 (Activation)   (None, 5, 5, 10)          0

 flatten (Flatten)           (None, 250)               0

 dense (Dense)               (None, 128)               32128

 dense_1 (Dense)             (None, 10)                1290

=================================================================
Total params: 34,428
Trainable params: 34,428
Non-trainable params: 0
_________________________________________________________________
```

Training


Training and Validation Accuracy
1.00
0.99
0.98
0.97
0.96
0.95
0.94
0.93
0.92
Accuracy
0
2
4
6
8
Training Accuracy
Validation Accuracy
Training and Validation Loss
1.0
0.8
0.6
0.4
0.2
0.0
Cross Entropy
Training Loss
Validation Loss
0
2
4
6
8
epoch

Accuracy Drop

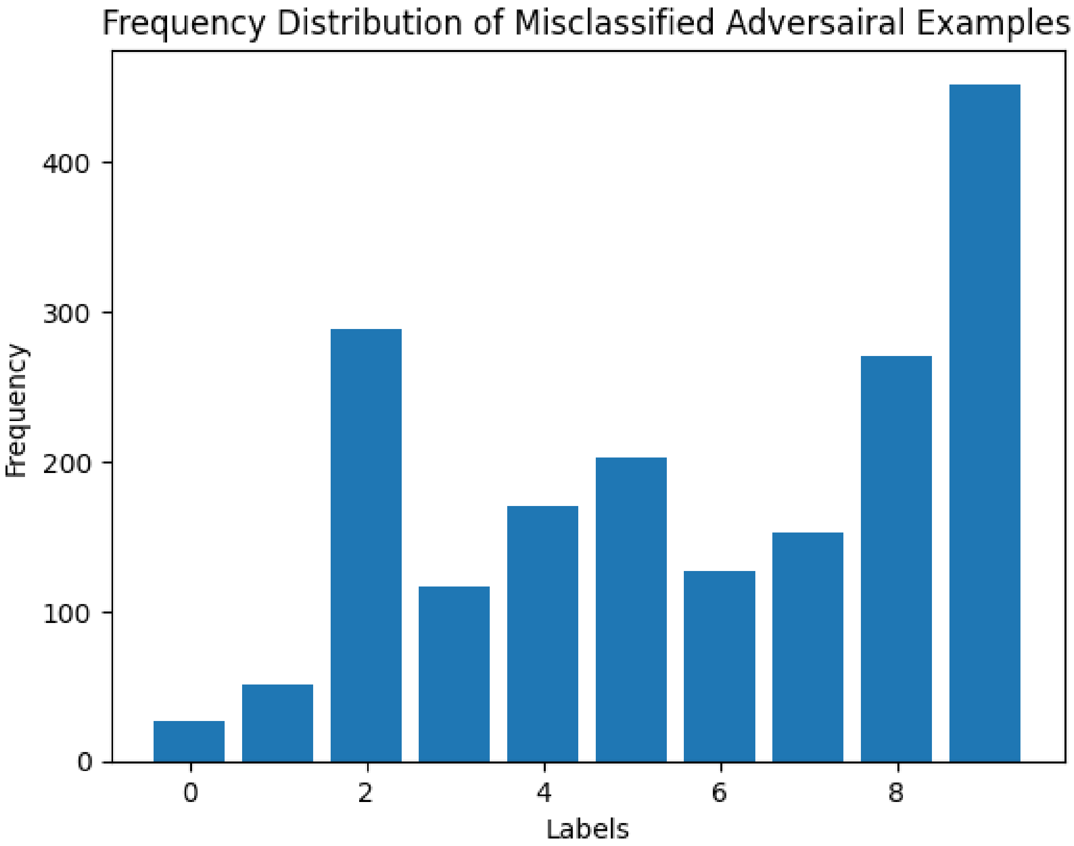




Further Analysis – Epsilon 10^{-1}

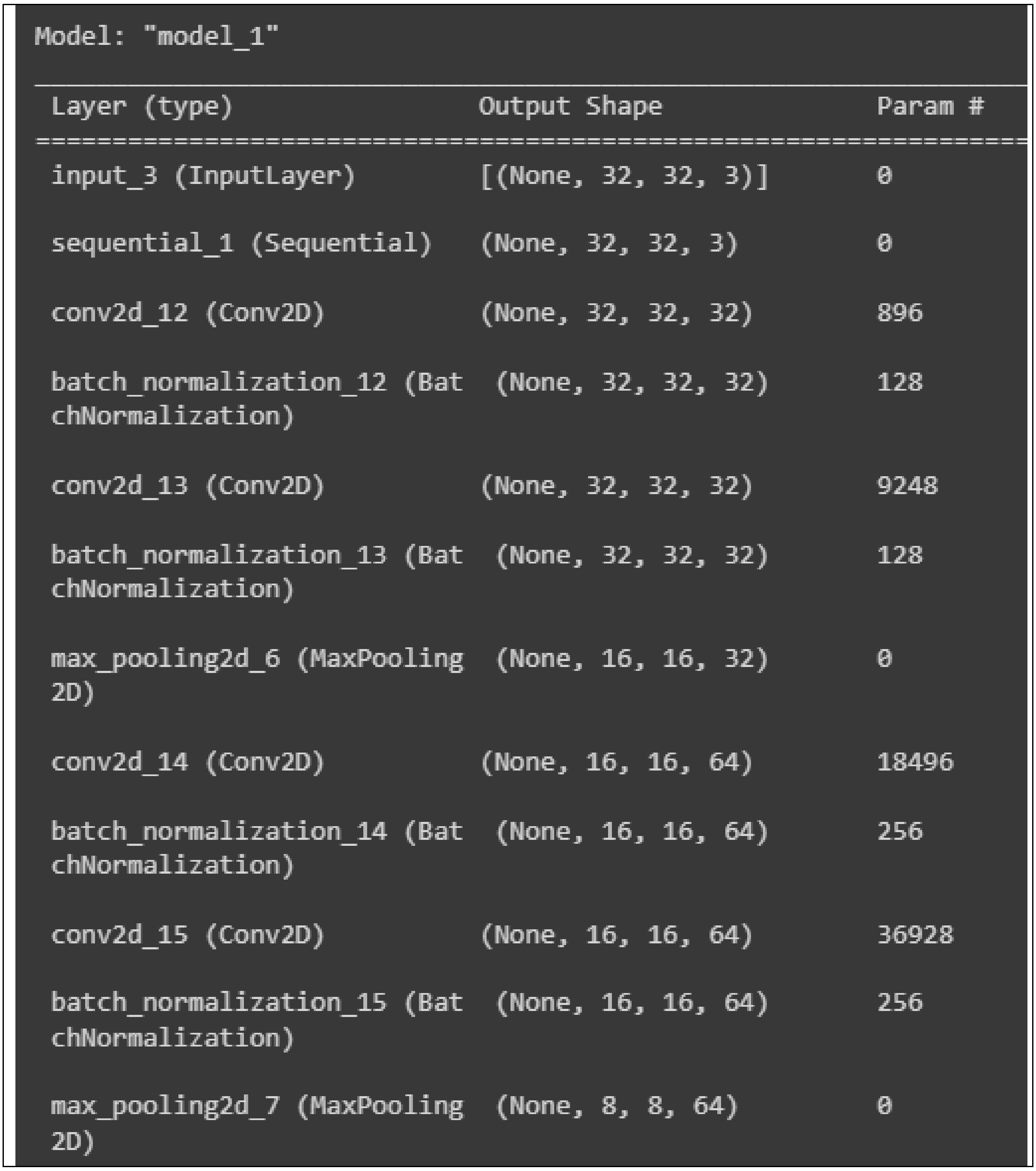
Correctly classified before and after		8029
Misclassified before and after	**Prediction unchanged**	109
	Prediction changed	5
Correctly classified, now misclassified		1857
Misclassified, now correctly classified		0

Distribution of Correctly classified, now misclassified.

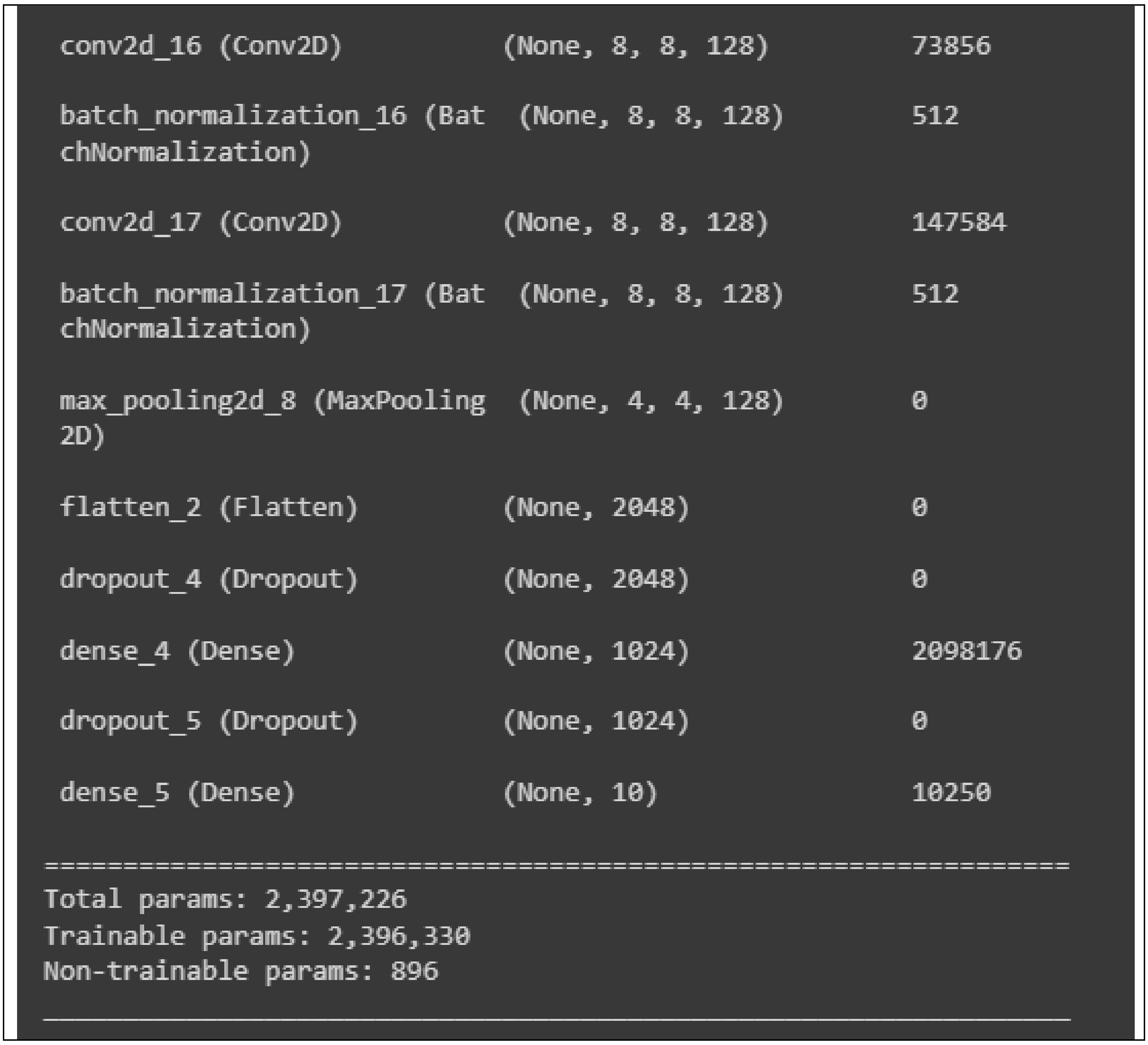


CIFAR10

Model



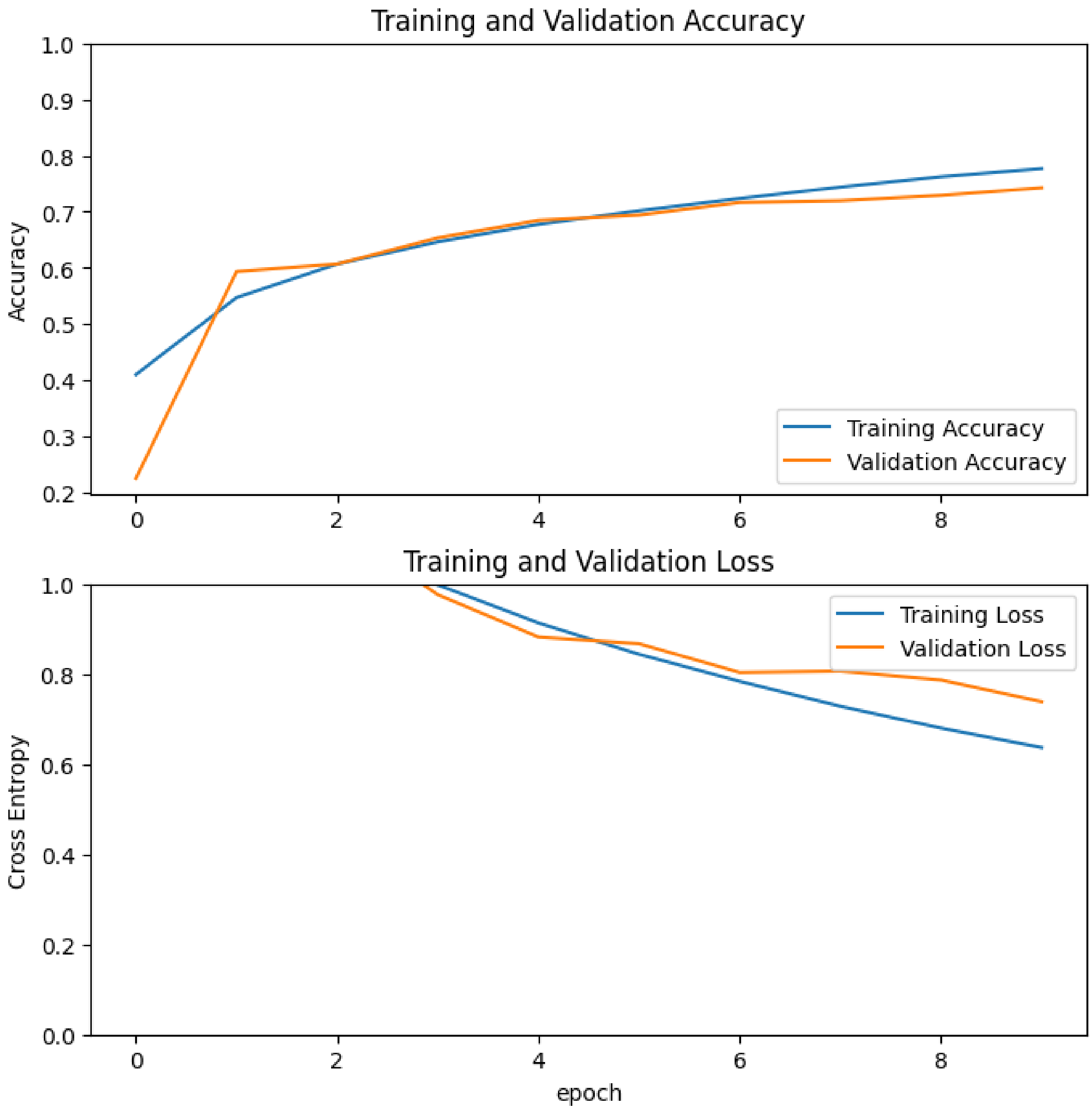
```
Model: "model_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 input_3 (InputLayer)        [(None, 32, 32, 3)]       0

 sequential_1 (Sequential)   (None, 32, 32, 3)         0

 conv2d_12 (Conv2D)          (None, 32, 32, 32)        896

 batch_normalization_12 (Bat  (None, 32, 32, 32)       128
 chNormalization)

 conv2d_13 (Conv2D)          (None, 32, 32, 32)        9248

 batch_normalization_13 (Bat  (None, 32, 32, 32)       128
 chNormalization)

 max_pooling2d_6 (MaxPooling  (None, 16, 16, 32)       0
 2D)

 conv2d_14 (Conv2D)          (None, 16, 16, 64)        18496

 batch_normalization_14 (Bat  (None, 16, 16, 64)       256
 chNormalization)

 conv2d_15 (Conv2D)          (None, 16, 16, 64)        36928

 batch_normalization_15 (Bat  (None, 16, 16, 64)       256
 chNormalization)

 max_pooling2d_7 (MaxPooling  (None, 8, 8, 64)         0
 2D)
```



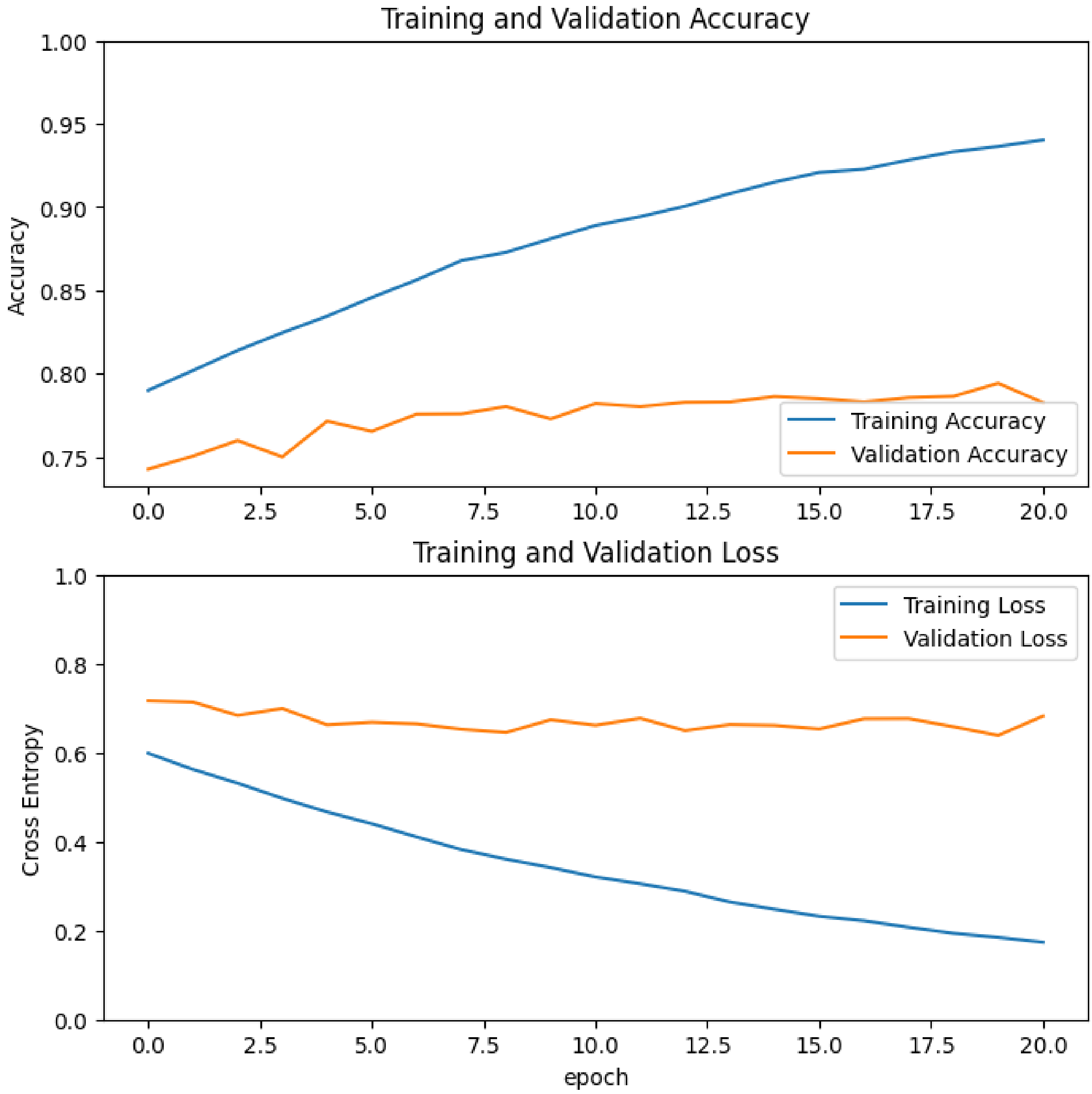
```
 conv2d_16 (Conv2D)          (None, 8, 8, 128)         73856

 batch_normalization_16 (Bat  (None, 8, 8, 128)        512
 chNormalization)

 conv2d_17 (Conv2D)          (None, 8, 8, 128)         147584

 batch_normalization_17 (Bat  (None, 8, 8, 128)        512
 chNormalization)

 max_pooling2d_8 (MaxPooling  (None, 4, 4, 128)        0
 2D)

 flatten_2 (Flatten)         (None, 2048)              0

 dropout_4 (Dropout)         (None, 2048)              0

 dense_4 (Dense)             (None, 1024)              2098176

 dropout_5 (Dropout)         (None, 1024)              0

 dense_5 (Dense)             (None, 10)                10250

=================================================================
Total params: 2,397,226
Trainable params: 2,396,330
Non-trainable params: 896
_________________________________________________________________
```

Training – 60 Epochs

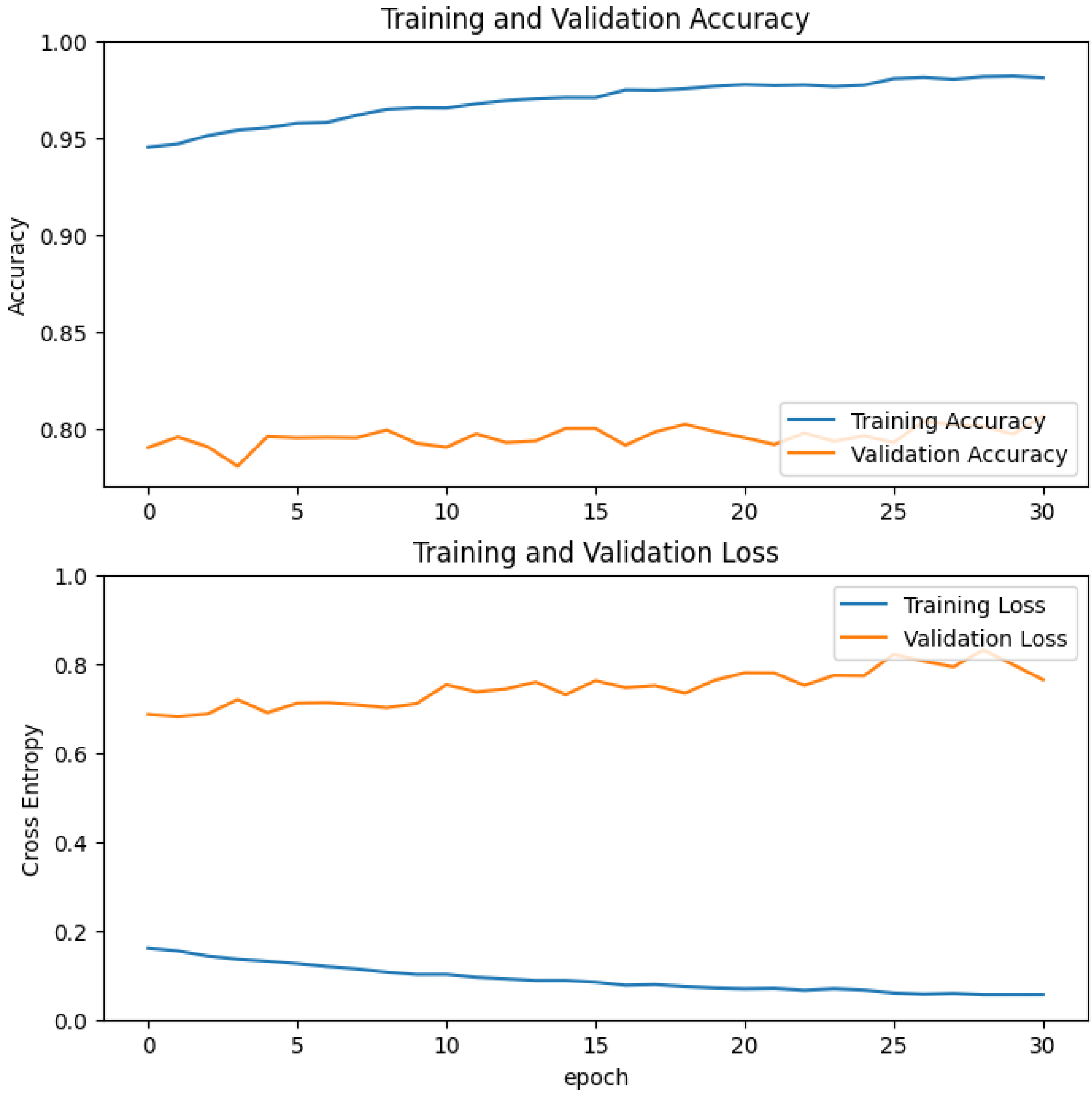
Epoch: 0 - 9



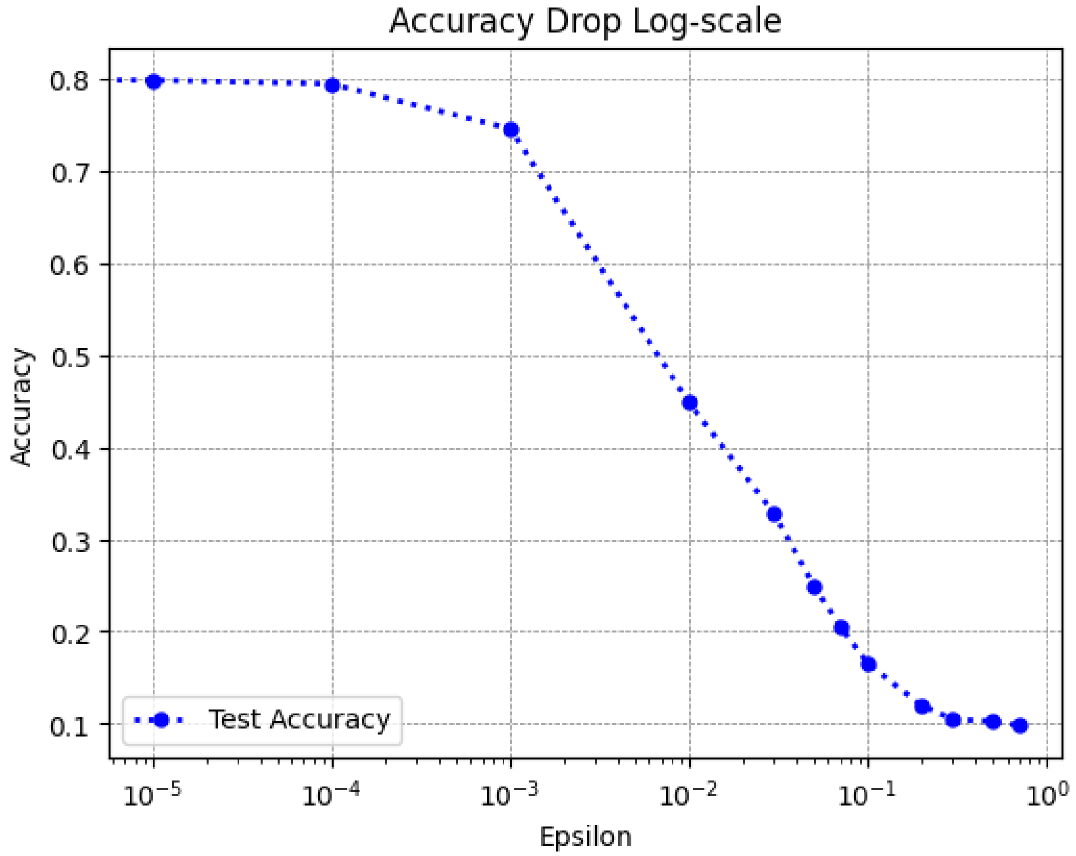
Epoch: 10 - 29

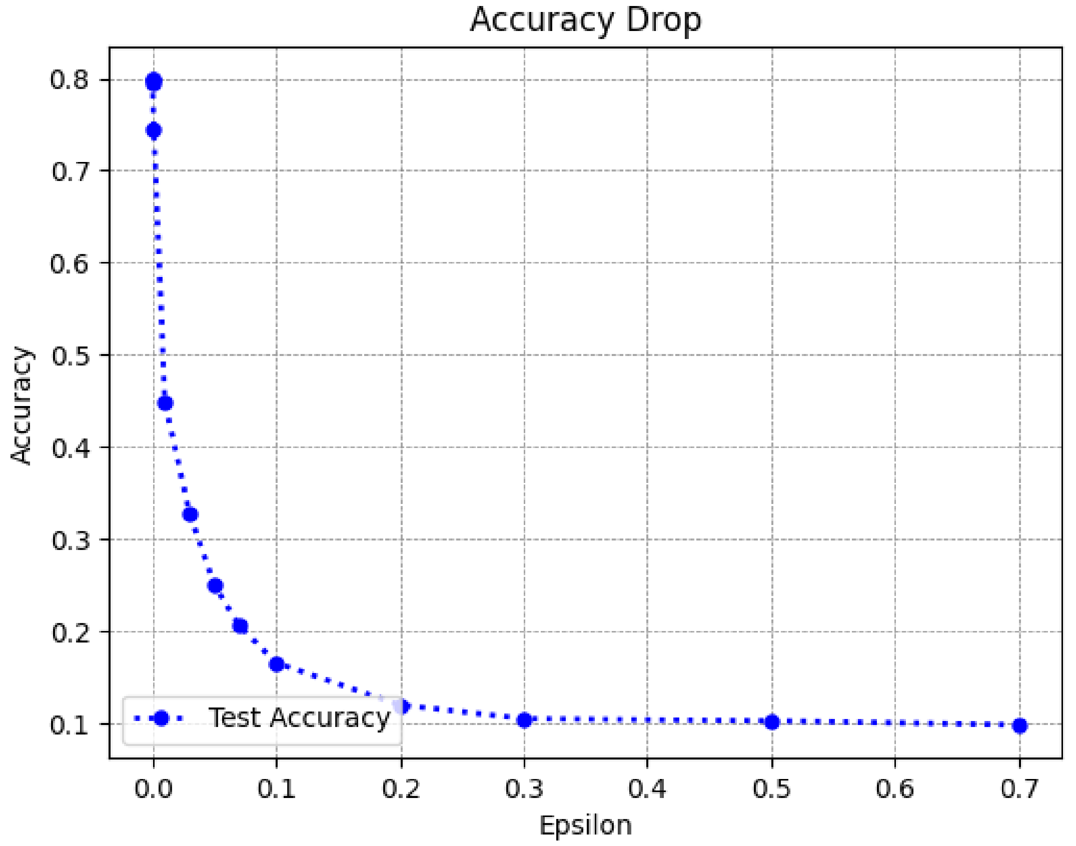

Training and Validation Accuracy
Accuracy
1.00
0.95
0.90
0.85
0.80
0.75
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
Training Accuracy
Validation Accuracy
Training and Validation Loss
Cross Entropy
1.0
0.8
0.6
0.4
0.2
0.0
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
epoch
Training Loss
Validation Loss

Epoch: 31 - 59



Accuracy Drop

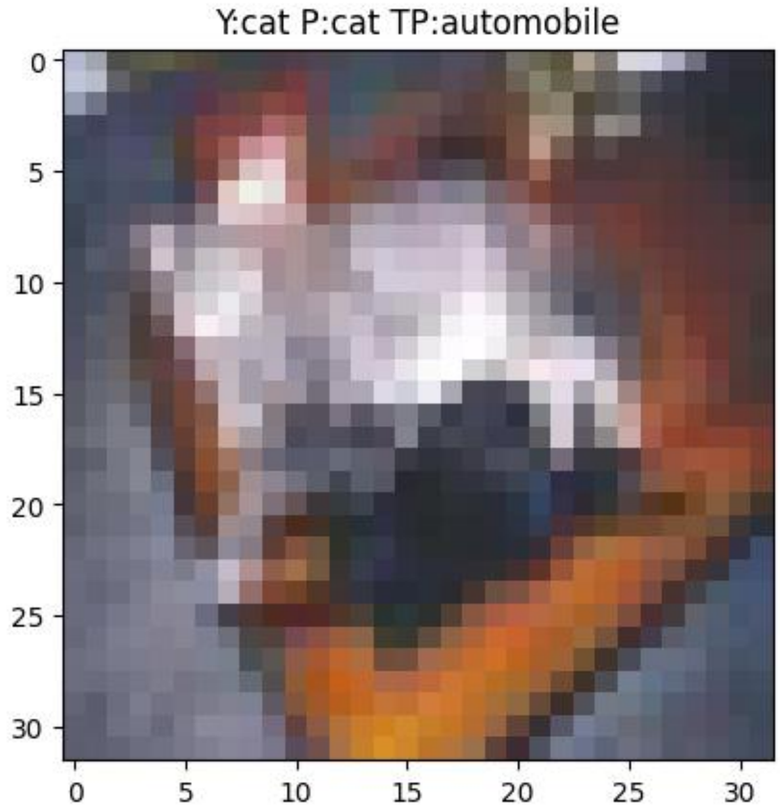

Accuracy Drop Log-scale
Accuracy
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
Test Accuracy
10^{-5}
10^{-4}
10^{-3}
10^{-2}
10^{-1}
10^{0}
Epsilon


Accuracy Drop
Accuracy
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
Test Accuracy
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
Epsilon

Further Analysis – Epsilon = 10^{-3}

Correctly classified before and after		7456
Misclassified before and after	**Prediction unchanged**	1970
	Prediction changed	37
Correctly classified, now misclassified		536
Misclassified, now correctly classified		1

Misclassified, now correctly classified

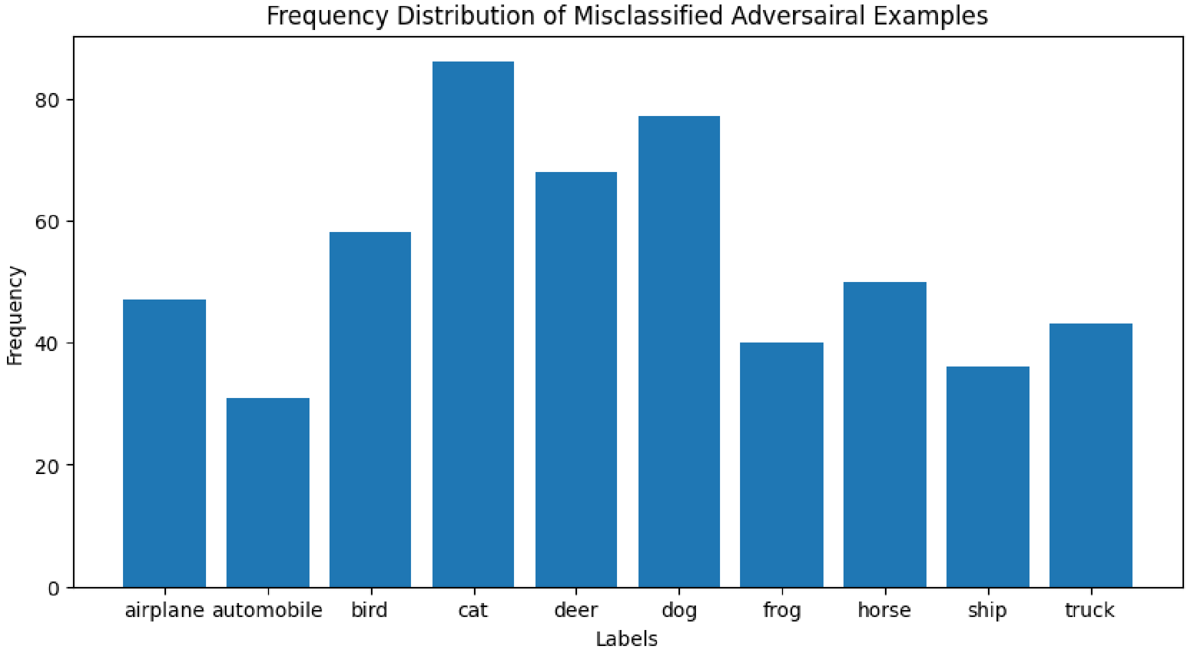


Y – True Label

TP – Prediction on clean image

P – Prediction on adversarial image

Distribution of Correctly classified, now misclassified


Frequency Distribution of Misclassified Adversairal Examples
Frequency
airplane
automobile
bird
cat
deer
dog
frog
horse
ship
truck
Labels