Notebook by Suryanarayan.B (CB.EN.U4CSE19056)

Question 1

1) MLP Model

In [143]:

```
import pandas as pd
import numpy as np
import random
import matplotlib.pyplot as plt
import tensorflow as tf
from tensorflow.keras import datasets, layers, models
from tensorflow.keras.utils import to_categorical
```

In [144]:

```
(X_train, Y_train), (X_test, Y_test) = datasets.mnist.load_data()
```

In [145]:

```
X_train.shape
```

Out[145]:

```
(60000, 28, 28)
```

In [146]:

```
X_test.shape
```

Out[146]:

```
(10000, 28, 28)
```

In [147]:

```
Y_train.shape
```

Out[147]:

```
(60000,)
```

In [148]:

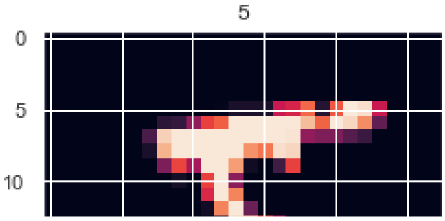
```
Y_test.shape
```

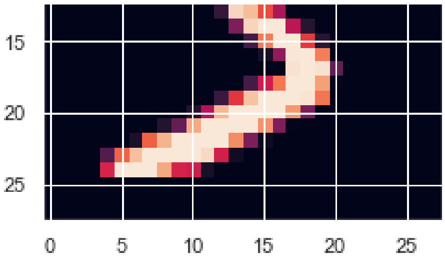
Out[148]:

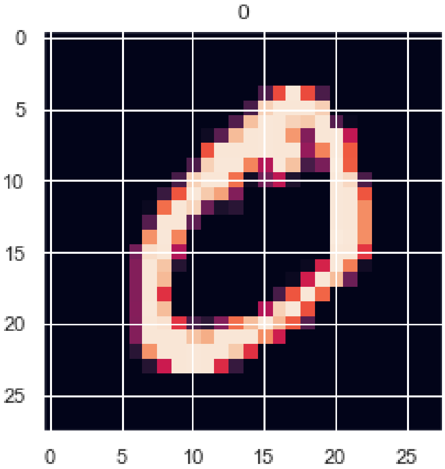
```
(10000,)
```

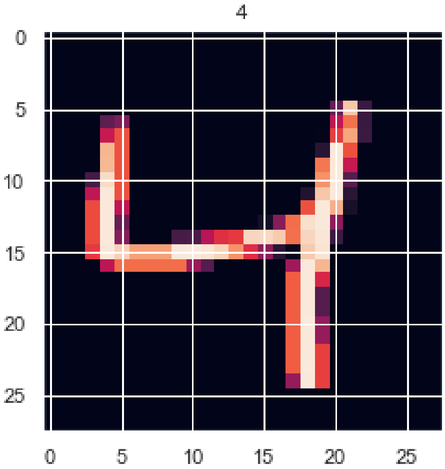
In [149]:

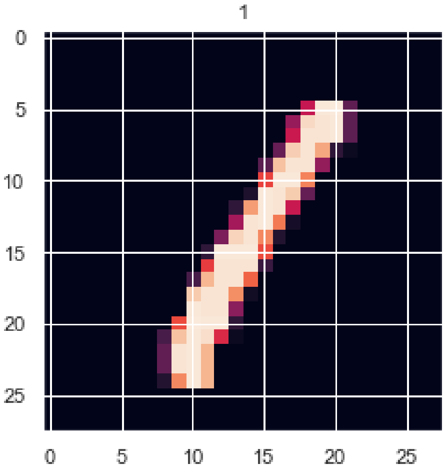
```
for i in range(5):
    plt.imshow(X_train[i])
    plt.title(Y_train[i])
    plt.show()
```

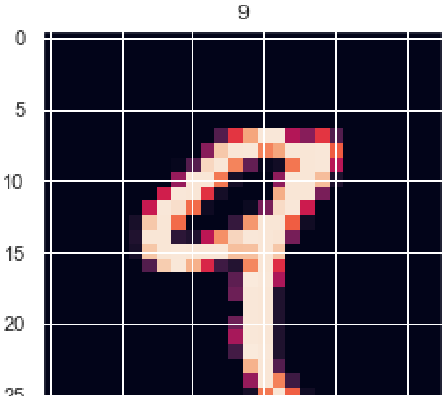



15
20
25
0
5
10
15
20
25


0
0
5
10
15
20
25
0
5
10
15
20
25


4
0
5
10
15
20
25
0
5
10
15
20
25


1
0
5
10
15
20
25
0
5
10
15
20
25


9
0
5
10
15
20
25

Printing the shape

In [150]:

```
print(X_train[0].shape)
print(X_train.shape)
```

```
(28, 28)
(60000, 28, 28)
```

Reshaping the data

In [151]:

```
X_train=X_train.reshape(X_train.shape[0],-1)
```

In [152]:

```
X_test=X_test.reshape(X_test.shape[0],-1)
```

One hot encoding

In [153]:

```
Y_train=to_categorical(Y_train)
```

In [154]:

```
Y_train
```

Out[154]:

```
array([[0., 0., 0., ..., 0., 0., 0.],
       [1., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.],
       ...,
       [0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 1., 0.]], dtype=float32)
```

In [155]:

```
Y_test=to_categorical(Y_test)
```

In [156]:

```
Y_test
```

Out[156]:

```
array([[0., 0., 0., ..., 1., 0., 0.],
       [0., 0., 1., ..., 0., 0., 0.],
       [0., 1., 0., ..., 0., 0., 0.],
       ...,
       [0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.]], dtype=float32)
```

Printing Shapes of train and test

In [157]:

```
print(X_train.shape,X_test.shape,Y_train.shape,Y_test.shape)
```

```
(60000, 784) (10000, 784) (60000, 10) (10000, 10)
```

In [158]:

```
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Flatten
from tensorflow.keras.layers import Dense
from tensorflow.keras.layers import Activation
from tensorflow.keras.models import load_model
from tensorflow.keras.layers import BatchNormalization,Dropout
from tensorflow.keras import optimizers
```

Here we use 3 hidden layers with 50 neurons and 4 batch normal function after each dense layer and 4 Dropout function after each activation layer.

Here the optimizer used is adam , kernel_initializer is he_normal and activation function is sigmoid

In [159]:

```
def in_model(initializer='he_normal'):
    model=Sequential()
    model.add(Dense(50, input_shape = (784,),kernel_initializer=initializer))
    model.add(BatchNormalization())
    model.add(Activation('sigmoid'))
    model.add(Dropout(0.2))
    model.add(Dense(50,kernel_initializer=initializer))
    model.add(BatchNormalization())
    model.add(Activation('sigmoid'))
    model.add(Dropout(0.2))
    model.add(Dense(50,kernel_initializer=initializer))
    model.add(BatchNormalization())
    model.add(Activation('sigmoid'))
    model.add(Dropout(0.2))
    model.add(Dense(50,kernel_initializer=initializer))
    model.add(BatchNormalization())
    model.add(Activation('sigmoid'))
    model.add(Dropout(0.2))
    model.add(Dense(10,kernel_initializer=initializer))
    model.add(Activation('softmax'))

    ad = optimizers.Adam(learning_rate = 0.001)
    model.compile(optimizer = ad, loss = "categorical_crossentropy", metrics = ['accurac
y'])
    return model
```

In [160]:

```
model=in_model()
```

In [161]:

```
model.summary()
```

```
Model: "sequential_14"
_________________________________________________________________
Layer (type)                 Output Shape              Param #   
=================================================================
dense_70 (Dense)             (None, 50)                39250     
_________________________________________________________________
batch_normalization_56 (Batc (None, 50)                200       
_________________________________________________________________
activation_70 (Activation)   (None, 50)                0         
_________________________________________________________________
dropout_56 (Dropout)         (None, 50)                0         
_________________________________________________________________
dense_71 (Dense)             (None, 50)                2550      
```

```
_________________________________________________________________
batch_normalization_57 (Batc (None, 50)                200       
_________________________________________________________________
activation_71 (Activation)   (None, 50)                0         
_________________________________________________________________
dropout_57 (Dropout)         (None, 50)                0         
_________________________________________________________________
dense_72 (Dense)             (None, 50)                2550      
_________________________________________________________________
batch_normalization_58 (Batc (None, 50)                200       
_________________________________________________________________
activation_72 (Activation)   (None, 50)                0         
_________________________________________________________________
dropout_58 (Dropout)         (None, 50)                0         
_________________________________________________________________
dense_73 (Dense)             (None, 50)                2550      
_________________________________________________________________
batch_normalization_59 (Batc (None, 50)                200       
_________________________________________________________________
activation_73 (Activation)   (None, 50)                0         
_________________________________________________________________
dropout_59 (Dropout)         (None, 50)                0         
_________________________________________________________________
dense_74 (Dense)             (None, 10)                510       
_________________________________________________________________
activation_74 (Activation)   (None, 10)                0         
=================================================================
Total params: 48,210
Trainable params: 47,810
Non-trainable params: 400
_________________________________________________________________
```

In [162]:

```python
from tensorflow.keras.callbacks import Callback
```

Custom callback function

This function stops the training once accuracy reaches 90%

In [163]:

```python
class TerminateOnBaseline(Callback):
    def __init__(self, monitor='accuracy', baseline=0.9):
        super(TerminateOnBaseline, self).__init__()
        self.monitor = monitor
        self.baseline = baseline

    def on_epoch_end(self, epoch, logs=None):
        logs = logs or {}
        acc = logs.get(self.monitor)
        if acc is not None:
            if acc >= self.baseline:
                print('Epoch %d: Reached baseline, terminating training' % (epoch))
                self.model.stop_training = True
```

In [164]:

```python
callback = [TerminateOnBaseline()]
```

Initializing the Filepath to save the Checkpoint and initializing checkpoint

In [166]:

```python
checkpoint_filepath = 'C:\\Users\\Surya\\Desktop\\College\\6th Semester\\Neural Networks and Deep Learning\\Lab Assignments\\Lab-4\\model1.h5'
```

In [168]:

```
checkpoint1 = tf.keras.callbacks.ModelCheckpoint(
    filepath=checkpoint_filepath,
    save_weights_only=False,
    monitor='accuracy',
    mode='auto',
    save_best_only=True)
```

Training the model

In [169]:

```
train1=model.fit(X_train,Y_train,validation_split=0.3,epochs=100,callbacks=[callback,chec
kpoint1],verbose=1)
```

```
Epoch 1/100
1313/1313 [==============================] - 22s 16ms/step - loss: 1.3036 - accuracy: 0.5
851 - val_loss: 0.4848 - val_accuracy: 0.8736
Epoch 2/100
1313/1313 [==============================] - 20s 15ms/step - loss: 0.7186 - accuracy: 0.7
843 - val_loss: 0.3383 - val_accuracy: 0.9032
Epoch 3/100
1313/1313 [==============================] - 21s 16ms/step - loss: 0.5834 - accuracy: 0.8
268 - val_loss: 0.2828 - val_accuracy: 0.9160
Epoch 4/100
1313/1313 [==============================] - 22s 16ms/step - loss: 0.5191 - accuracy: 0.8
463 - val_loss: 0.2545 - val_accuracy: 0.9253
Epoch 5/100
1313/1313 [==============================] - 20s 15ms/step - loss: 0.4736 - accuracy: 0.8
615 - val_loss: 0.2360 - val_accuracy: 0.9314
Epoch 6/100
1313/1313 [==============================] - 20s 15ms/step - loss: 0.4397 - accuracy: 0.8
717 - val_loss: 0.2103 - val_accuracy: 0.9379
Epoch 7/100
1313/1313 [==============================] - 20s 15ms/step - loss: 0.4056 - accuracy: 0.8
809 - val_loss: 0.1961 - val_accuracy: 0.9432
Epoch 8/100
1313/1313 [==============================] - 20s 15ms/step - loss: 0.3869 - accuracy: 0.8
858 - val_loss: 0.1866 - val_accuracy: 0.9474
Epoch 9/100
1313/1313 [==============================] - 20s 15ms/step - loss: 0.3693 - accuracy: 0.8
914 - val_loss: 0.1754 - val_accuracy: 0.9505
Epoch 10/100
1313/1313 [==============================] - 20s 15ms/step - loss: 0.3514 - accuracy: 0.8
967 - val_loss: 0.1668 - val_accuracy: 0.9528
Epoch 11/100
1313/1313 [==============================] - 23s 17ms/step - loss: 0.3384 - accuracy: 0.9
008 - val_loss: 0.1619 - val_accuracy: 0.9527
Epoch 10: Reached baseline, terminating training
```

Loading the weights into the model from model1.h5

In [170]:

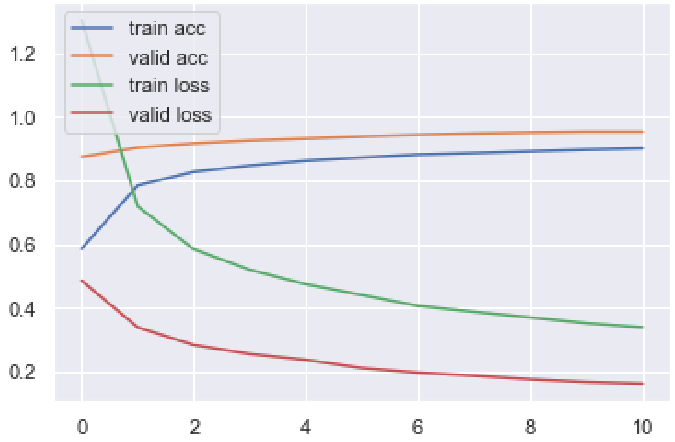
```
modelmn1=in_model()
```

In [171]:

```
modelmn1.load_weights('model1.h5')
```

In [172]:

```
plt.plot(train1.history['accuracy' ])
plt.plot(train1.history['val_accuracy'])
plt.plot(train1.history['loss'])
plt.plot(train1.history['val_loss'])
plt.legend(['train acc', 'valid acc', 'train loss', 'valid loss'], loc = 'upper left')
plt.show()
```



Printing the accuracy

In [173]:

```
model_out1=model.evaluate(X_test,Y_test)
print(f"Test Accuracy is: {model_out1[1]*100}%")
```

```
313/313 [==============================] - 2s 6ms/step - loss: 0.1506 - accuracy: 0.9546
Test Accuracy is: 95.45999765396118%
```

In [174]:

```
print(model.metrics_names)
print(model_out1)
```

```
['loss', 'accuracy']
[0.1505972146987915, 0.9545999765396118]
```

Printing the accuracy from model1.h5

In [176]:

```
model_outmn1=modelmn1.evaluate(X_test,Y_test)
print(f"Test Accuracy is: {model_outmn1[1]*100}%")
```

```
313/313 [==============================] - 2s 6ms/step - loss: 0.1506 - accuracy: 0.9546
Test Accuracy is: 95.45999765396118%
```

Printing Confusion Matrix

In [93]:

```
from sklearn.metrics import confusion_matrix
```

In [94]:

```
def print_conf(model):
    labels=Y_test
    y_pred=model.predict(X_test)
    diffmatrix = confusion_matrix(labels.argmax(axis=1), y_pred.argmax(axis=1))
    return diffmatrix
```

In [95]:

```
cm1=print_conf(model)
```

In [96]:

```
import seaborn as sns
```

In [97]:

```
cm1
```

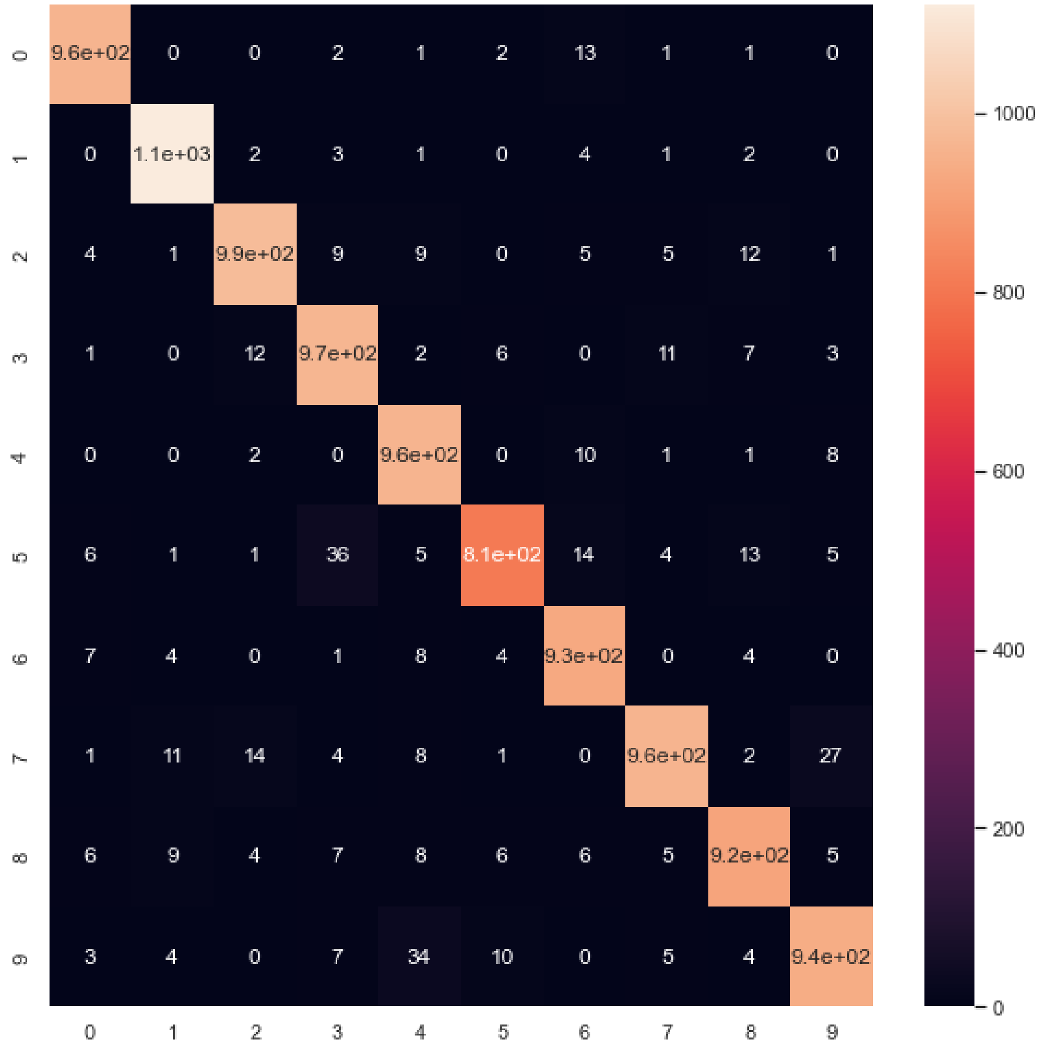
Out[97]:

```
array([[ 960,    0,    0,    2,    1,    2,   13,    1,    1,    0],
       [   0, 1122,    2,    3,    1,    0,    4,    1,    2,    0],
       [   4,    1,  986,    9,    9,    0,    5,    5,   12,    1],
       [   1,    0,   12,  968,    2,    6,    0,   11,    7,    3],
       [   0,    0,    2,    0,  960,    0,   10,    1,    1,    8],
       [   6,    1,    1,   36,    5,  807,   14,    4,   13,    5],
       [   7,    4,    0,    1,    8,    4,  930,    0,    4,    0],
       [   1,   11,   14,    4,    8,    1,    0,  960,    2,   27],
       [   6,    9,    4,    7,    8,    6,    6,    5,  918,    5],
       [   3,    4,    0,    7,   34,   10,    0,    5,    4,  942]],
      dtype=int64)
```

In [98]:

```
def matrix(cm):
    fig, ax = plt.subplots(figsize=(10,10))
    sns.set(font_scale=1)
    sns.heatmap(cm, annot=True, ax=ax)

matrix(cm1)
```



2) Ensemble Learning model using Voting Classifier

In [99]:

```
y_train=np.argmax(Y_train,axis=1)
y_test=np.argmax(Y_test,axis=1)
```

In [100]:

```
from tensorflow.keras.wrappers.scikit_learn import KerasClassifier
from sklearn.ensemble import VotingClassifier
from sklearn.metrics import accuracy_score
```

In [101]:

```
callback1 = [TerminateOnBaseline()]
```

In [102]:

```
checkpoint_filepath1 = 'C:\\Users\\Surya\\Desktop\\College\\6th Semester\\Neural Networks and Deep Learning\\Lab Assignments\\Lab-4\\emodel.h5'
```

In [103]:

```
checkpoint2 = tf.keras.callbacks.ModelCheckpoint(
    filepath=checkpoint_filepath1,
    save_weights_only=False,
    monitor='accuracy',
    mode='auto',
    save_best_only=True)
```

Initializing 3 models for Ensemble Learning

In [104]:

```
emodel1=KerasClassifier(build_fn=in_model,epochs=100,validation_split=0.3,callbacks=[callback1,checkpoint2],verbose=1)
emodel2=KerasClassifier(build_fn=in_model,epochs=100,validation_split=0.3,callbacks=[callback1,checkpoint2],verbose=1)
emodel3=KerasClassifier(build_fn=in_model,epochs=100,validation_split=0.3,callbacks=[callback1,checkpoint2],verbose=1)
emodel1._estimator_type="classifier"
emodel2._estimator_type="classifier"
emodel3._estimator_type="classifier"
```

In [105]:

```
ensemble=VotingClassifier(estimators=[('model1',emodel1),('model2',emodel2),('model3',emodel3)],voting='soft')
```

In [106]:

```
etrain=ensemble.fit(X_train,y_train)
```

```
Epoch 1/100
1313/1313 [==============================] - 26s 19ms/step - loss: 1.2967 - accuracy: 0.5
851 - val_loss: 0.4560 - val_accuracy: 0.8869
Epoch 2/100
1313/1313 [==============================] - 27s 20ms/step - loss: 0.7083 - accuracy: 0.7
874 - val_loss: 0.3385 - val_accuracy: 0.9021
Epoch 3/100
1313/1313 [==============================] - 21s 16ms/step - loss: 0.5812 - accuracy: 0.8
276 - val_loss: 0.2803 - val_accuracy: 0.9193
Epoch 4/100
1313/1313 [==============================] - 21s 16ms/step - loss: 0.5152 - accuracy: 0.8
472 - val_loss: 0.2483 - val_accuracy: 0.9267
Epoch 5/100
1313/1313 [==============================] - 21s 16ms/step - loss: 0.4670 - accuracy: 0.8
619 - val_loss: 0.2209 - val_accuracy: 0.9371
Epoch 6/100
1313/1313 [==============================] - 21s 16ms/step - loss: 0.4400 - accuracy: 0.8
700 - val_loss: 0.2066 - val_accuracy: 0.9378
Epoch 7/100
1313/1313 [==============================] - 21s 16ms/step - loss: 0.4042 - accuracy: 0.8
813 - val_loss: 0.1932 - val_accuracy: 0.9426
Epoch 8/100
1313/1313 [==============================] - 21s 16ms/step - loss: 0.3804 - accuracy: 0.8
876 - val_loss: 0.1811 - val_accuracy: 0.9461
Epoch 9/100
Epoch 9/100
1313/1313 [==============================] - 26s 20ms/step - loss: 0.3603 - accuracy: 0.8
925 - val_loss: 0.1715 - val_accuracy: 0.9502
Epoch 10/100
1313/1313 [==============================] - 20s 16ms/step - loss: 0.3556 - accuracy: 0.8
956 - val_loss: 0.1668 - val_accuracy: 0.9514
Epoch 11/100
1313/1313 [==============================] - 19s 14ms/step - loss: 0.3342 - accuracy: 0.9
007 - val_loss: 0.1556 - val_accuracy: 0.9540
Epoch 10: Reached baseline, terminating training
Epoch 1/100
1313/1313 [==============================] - 23s 17ms/step - loss: 1.3153 - accuracy: 0.5
815 - val_loss: 0.4487 - val_accuracy: 0.8823
Epoch 2/100
1313/1313 [==============================] - 21s 16ms/step - loss: 0.6934 - accuracy: 0.7
933 - val_loss: 0.3150 - val_accuracy: 0.9073
Epoch 3/100
1313/1313 [==============================] - 23s 17ms/step - loss: 0.5666 - accuracy: 0.8
309 - val_loss: 0.2696 - val_accuracy: 0.9211
Epoch 4/100
1313/1313 [==============================] - 22s 17ms/step - loss: 0.4985 - accuracy: 0.8
514 - val_loss: 0.2379 - val_accuracy: 0.9297
Epoch 5/100
1313/1313 [==============================] - 23s 17ms/step - loss: 0.4655 - accuracy: 0.8
631 - val_loss: 0.2183 - val_accuracy: 0.9362
Epoch 6/100
1313/1313 [==============================] - 19s 15ms/step - loss: 0.4344 - accuracy: 0.8
719 - val_loss: 0.2083 - val_accuracy: 0.9387
Epoch 7/100
1313/1313 [==============================] - 19s 15ms/step - loss: 0.4077 - accuracy: 0.8
786 - val_loss: 0.1923 - val_accuracy: 0.9439
Epoch 8/100
1313/1313 [==============================] - 19s 15ms/step - loss: 0.3861 - accuracy: 0.8
857 - val_loss: 0.1819 - val_accuracy: 0.9470
Epoch 9/100
1313/1313 [==============================] - 20s 15ms/step - loss: 0.3684 - accuracy: 0.8
903 - val_loss: 0.1779 - val_accuracy: 0.9467
Epoch 10/100
1313/1313 [==============================] - 20s 15ms/step - loss: 0.3538 - accuracy: 0.8
968 - val_loss: 0.1698 - val_accuracy: 0.9516
Epoch 11/100
1313/1313 [==============================] - 22s 17ms/step - loss: 0.3366 - accuracy: 0.9
010 - val_loss: 0.1633 - val_accuracy: 0.9523
Epoch 10: Reached baseline, terminating training
Epoch 1/100
1313/1313 [==============================] - 27s 20ms/step - loss: 1.3157 - accuracy: 0.5
773 - val_loss: 0.4858 - val_accuracy: 0.8784
Epoch 2/100
1313/1313 [==============================] - 22s 16ms/step - loss: 0.7183 - accuracy: 0.7
784 - val_loss: 0.3334 - val_accuracy: 0.9081
Epoch 3/100
1313/1313 [==============================] - 21s 16ms/step - loss: 0.5872 - accuracy: 0.8
258 - val_loss: 0.2833 - val_accuracy: 0.9173
Epoch 4/100
1313/1313 [==============================] - 21s 16ms/step - loss: 0.5086 - accuracy: 0.8
490 - val_loss: 0.2438 - val_accuracy: 0.9303
Epoch 5/100
1313/1313 [==============================] - 21s 16ms/step - loss: 0.4614 - accuracy: 0.8
660 - val_loss: 0.2177 - val_accuracy: 0.9382
Epoch 6/100
1313/1313 [==============================] - 21s 16ms/step - loss: 0.4188 - accuracy: 0.8
771 - val_loss: 0.2006 - val_accuracy: 0.9406
Epoch 7/100
1313/1313 [==============================] - 37s 28ms/step - loss: 0.4020 - accuracy: 0.8
805 - val_loss: 0.1879 - val_accuracy: 0.9448
Epoch 8/100
1313/1313 [==============================] - 29s 22ms/step - loss: 0.3752 - accuracy: 0.8
891 - val_loss: 0.1777 - val_accuracy: 0.9488
Epoch 9/100
1313/1313 [==============================] - 20s 15ms/step - loss: 0.3634 - accuracy: 0.8
920 - val_loss: 0.1709 - val_accuracy: 0.9508
Epoch 10/100
1313/1313 [==============================] - 19s 14ms/step - loss: 0.3389 - accuracy: 0.9
```

```
000 - val_loss: 0.1683 - val_accuracy: 0.9501
Epoch 11/100
1313/1313 [==============================] - 19s 14ms/step - loss: 0.3336 - accuracy: 0.9
029 - val_loss: 0.1608 - val_accuracy: 0.9527
Epoch 10: Reached baseline, terminating training
```

In [107]:

```
Y_pred1 = ensemble.predict(X_test)
```

```
313/313 [==============================] - 1s 3ms/step
313/313 [==============================] - 1s 2ms/step
313/313 [==============================] - 1s 3ms/step
```

Printing the accuracy

In [108]:

```
ex1=accuracy_score(y_test,Y_pred1)
```

In [109]:

```
print("Test Accuracy is :", ex1*100 ," %")
```

```
Test Accuracy is : 95.97  %
```

In [110]:

```
model13=in_model()
```

Loading the model from emodel.h5 and printing the accuracy

In [111]:

```
model13.load_weights('emodel.h5')
```

In [112]:

```
model_out12=model13.evaluate(X_test,Y_test)
print(f"Test Accuracy is: {model_out12[1]*100}%")
```

```
313/313 [==============================] - 2s 5ms/step - loss: 0.1504 - accuracy: 0.9555
Test Accuracy is: 95.55000066757202%
```

Printing the confusion matrix

In [113]:

```
cm2=confusion_matrix(y_test, Y_pred1)
```

In [114]:

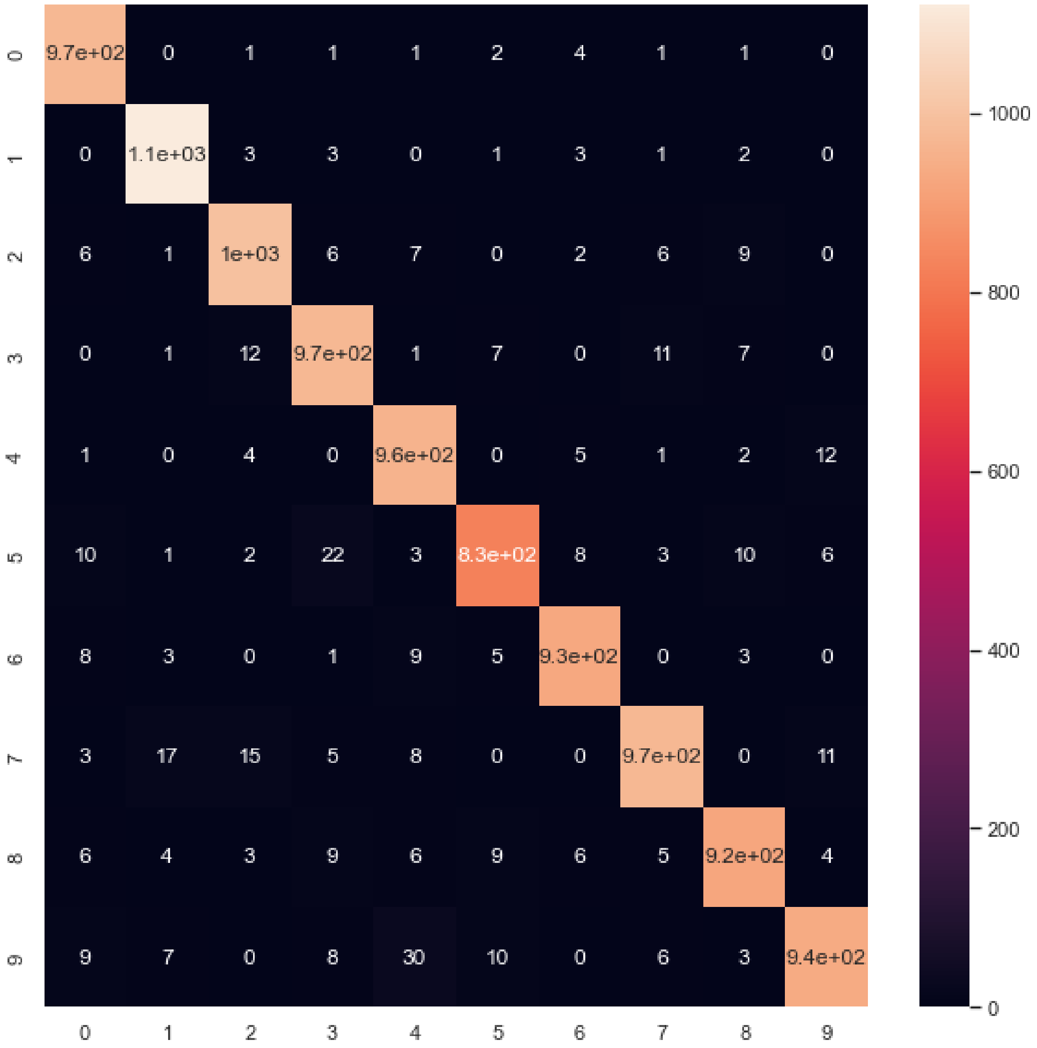
```
cm2
```

Out[114]:

```
array([[ 969,    0,    1,    1,    1,    2,    4,    1,    1,    0],
       [   0, 1122,    3,    3,    0,    1,    3,    1,    2,    0],
       [   6,    1,  995,    6,    7,    0,    2,    6,    9,    0],
       [   0,    1,   12,  971,    1,    7,    0,   11,    7,    0],
       [   1,    0,    4,    0,  957,    0,    5,    1,    2,   12],
       [  10,    1,    2,   22,    3,  827,    8,    3,   10,    6],
       [   8,    3,    0,    1,    9,    5,  929,    0,    3,    0],
       [   3,   17,   15,    5,    8,    0,    0,  969,    0,   11],
       [   6,    4,    3,    9,    6,    9,    6,    5,  922,    4],
       [   9,    7,    0,    8,   30,   10,    0,    6,    3,  936]],
      dtype=int64)
```

In [115]:

```
matrix(cm2)
```



	0	1	2	3	4	5	6	7	8	9
0	9.7e+02	0	1	1	1	2	4	1	1	0
1	0	1.1e+03	3	3	0	1	3	1	2	0
2	6	1	1e+03	6	7	0	2	6	9	0
3	0	1	12	9.7e+02	1	7	0	11	7	0
4	1	0	4	0	9.6e+02	0	5	1	2	12
5	10	1	2	22	3	8.3e+02	8	3	10	6
6	8	3	0	1	9	5	9.3e+02	0	3	0
7	3	17	15	5	8	0	0	9.7e+02	0	11
8	6	4	3	9	6	9	6	5	9.2e+02	4
9	9	7	0	8	30	10	0	6	3	9.4e+02

Question 2

1) MLP Model

Here the dataset used is https://archive.ics.uci.edu/ml/machine-learning-databases/car/car.data

In [182]:

```
df=pd.read_csv('https://archive.ics.uci.edu/ml/machine-learning-databases/car/car.data')
```

In [183]:

```
df.head()
```

Out[183]:

	vhigh	vhigh.1	2	2.1	small	low	unacc
0	vhigh	vhigh	2	2	small	med	unacc
1	vhigh	vhigh	2	2	small	high	unacc
2	vhigh	vhigh	2	2	med	low	unacc
3	vhigh	vhigh	2	2	med	med	unacc
4	vhigh	vhigh	2	2	med	high	unacc

Splitting the data

In [184]:

```
X=df.drop('unacc',axis=1)
X
```

Out[184]:

	vhigh	vhigh.1	2	2.1	small	low
0	vhigh	vhigh	2	2	small	med
1	vhigh	vhigh	2	2	small	high
2	vhigh	vhigh	2	2	med	low
3	vhigh	vhigh	2	2	med	med
4	vhigh	vhigh	2	2	med	high
...	...	...	...	...	...	...
1722	low	low	5more	more	med	med
1723	low	low	5more	more	med	high
1724	low	low	5more	more	big	low
1725	low	low	5more	more	big	med
1726	low	low	5more	more	big	high

1727 rows × 6 columns

In [185]:

```
Y=df.iloc[:,-1]
Y
```

Out[185]:

```
0       unacc
1       unacc
2       unacc
3       unacc
4       unacc
        ...  
1722     good
1723    vgood
1724    unacc
1725     good
1726    vgood
Name: unacc, Length: 1727, dtype: object
```

One-Hot Encoding

In [186]:

```
X=pd.get_dummies(X)
```

In [187]:

```
X=X.values
```

In [188]:

```
Y=pd.get_dummies(Y)
```

In [189]:

```
Y=Y.values
```

In [190]:

```
print(X.shape,Y.shape)
```

```
(1727, 21) (1727, 4)
```

Train-test-split

In [191]:

```
from sklearn.model_selection import train_test_split
```

In [192]:

```
X_train, X_test, y_train, y_test = train_test_split(X, Y, test_size=0.33, random_state=42)
```

In [193]:

```
print("Train shape",X_train.shape)
print("Test shape",X_test.shape)
print(y_train.shape)
print(y_test.shape)
```

```
Train shape (1157, 21)
Test shape (570, 21)
(1157, 4)
(570, 4)
```

In [194]:

```
X_train=np.asarray(X_train).astype(float)

y_train=np.asarray(y_train).astype(float)
```

Initializing the NN Model

In [195]:

```
def car_model(initializer='he_normal'):
    model=Sequential()
    model.add(Dense(50, input_shape = (21,),kernel_initializer=initializer))
    model.add(BatchNormalization())
    model.add(Activation('sigmoid'))
    model.add(Dropout(0.2))
    model.add(Dense(50,kernel_initializer=initializer))
    model.add(BatchNormalization())
    model.add(Activation('sigmoid'))
    model.add(Dropout(0.2))
    model.add(Dense(50,kernel_initializer=initializer))
    model.add(BatchNormalization())
    model.add(Activation('sigmoid'))
    model.add(Dropout(0.2))
    model.add(Dense(50,kernel_initializer=initializer))
    model.add(BatchNormalization())
    model.add(Activation('sigmoid'))
    model.add(Dropout(0.2))
    model.add(Dense(4,kernel_initializer=initializer))
    model.add(Activation('softmax'))

    ad = optimizers.Adam(learning_rate = 0.001)
    model.compile(optimizer = ad, loss = "categorical_crossentropy", metrics = ['accuracy'])
    return model
```

In [196]:

```
cmodel1=car_model()
```

```
cmodel1=car_model()
```

In [197]:

```
cmodel1.summary()
```

```
Model: "sequential_17"
_________________________________________________________________
Layer (type)                 Output Shape              Param #   
=================================================================
dense_85 (Dense)             (None, 50)                1100      
_________________________________________________________________
batch_normalization_68 (Batc (None, 50)                200       
_________________________________________________________________
activation_85 (Activation)   (None, 50)                0         
_________________________________________________________________
dropout_68 (Dropout)         (None, 50)                0         
_________________________________________________________________
dense_86 (Dense)             (None, 50)                2550      
_________________________________________________________________
batch_normalization_69 (Batc (None, 50)                200       
_________________________________________________________________
activation_86 (Activation)   (None, 50)                0         
_________________________________________________________________
dropout_69 (Dropout)         (None, 50)                0         
_________________________________________________________________
dense_87 (Dense)             (None, 50)                2550      
_________________________________________________________________
batch_normalization_70 (Batc (None, 50)                200       
_________________________________________________________________
activation_87 (Activation)   (None, 50)                0         
_________________________________________________________________
dropout_70 (Dropout)         (None, 50)                0         
_________________________________________________________________
dense_88 (Dense)             (None, 50)                2550      
_________________________________________________________________
batch_normalization_71 (Batc (None, 50)                200       
_________________________________________________________________
activation_88 (Activation)   (None, 50)                0         
_________________________________________________________________
dropout_71 (Dropout)         (None, 50)                0         
_________________________________________________________________
dense_89 (Dense)             (None, 4)                 204       
_________________________________________________________________
activation_89 (Activation)   (None, 4)                 0         
=================================================================
Total params: 9,754
Trainable params: 9,354
Non-trainable params: 400
_________________________________________________________________
```

Call back function

In [198]:

```
callback_c1 = [TerminateOnBaseline()]
```

In [199]:

```
checkpoint_filepath2 = 'C:\\Users\\Surya\\Desktop\\College\\6th Semester\\Neural Networks and Deep Learning\\Lab Assignments\\Lab-4\\car1.h5'
checkpoint3 = tf.keras.callbacks.ModelCheckpoint(
    filepath=checkpoint_filepath2,
    save_weights_only=False,
    monitor='accuracy',
    mode='auto',
    save_best_only=True)
```

Training the model

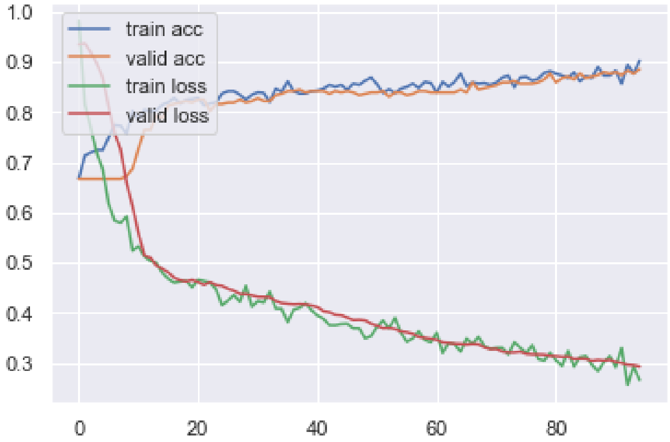
In [200]:

```
ctrain1=cmodel1.fit(X_train,y_train,validation_split=0.3,epochs=100,callbacks=[callback_
c1,checkpoint3],verbose=1)
```

```
Epoch 1/100
26/26 [==============================] - 2s 28ms/step - loss: 0.9817 - accuracy: 0.6675 -
val_loss: 0.9348 - val_accuracy: 0.6667
Epoch 2/100
26/26 [==============================] - 0s 19ms/step - loss: 0.8155 - accuracy: 0.7132 -
val_loss: 0.9367 - val_accuracy: 0.6667
Epoch 3/100
26/26 [==============================] - 0s 18ms/step - loss: 0.7681 - accuracy: 0.7194 -
val_loss: 0.9187 - val_accuracy: 0.6667
Epoch 4/100
26/26 [==============================] - 0s 18ms/step - loss: 0.7185 - accuracy: 0.7244 -
val_loss: 0.8969 - val_accuracy: 0.6667
Epoch 5/100
26/26 [==============================] - 0s 18ms/step - loss: 0.6863 - accuracy: 0.7244 -
val_loss: 0.8687 - val_accuracy: 0.6667
Epoch 6/100
26/26 [==============================] - 0s 17ms/step - loss: 0.6175 - accuracy: 0.7491 -
val_loss: 0.8110 - val_accuracy: 0.6667
Epoch 7/100
26/26 [==============================] - 0s 17ms/step - loss: 0.5834 - accuracy: 0.7738 -
val_loss: 0.7579 - val_accuracy: 0.6667
Epoch 8/100
26/26 [==============================] - 0s 18ms/step - loss: 0.5791 - accuracy: 0.7726 -
val_loss: 0.7250 - val_accuracy: 0.6667
Epoch 9/100
26/26 [==============================] - 0s 16ms/step - loss: 0.5919 - accuracy: 0.7565 -
val_loss: 0.6591 - val_accuracy: 0.6724
Epoch 10/100
26/26 [==============================] - 0s 18ms/step - loss: 0.5236 - accuracy: 0.8022 -
val_loss: 0.6124 - val_accuracy: 0.6868
Epoch 11/100
26/26 [==============================] - 0s 17ms/step - loss: 0.5321 - accuracy: 0.7849 -
val_loss: 0.5585 - val_accuracy: 0.7270
Epoch 12/100
26/26 [==============================] - 0s 16ms/step - loss: 0.5124 - accuracy: 0.8047 -
val_loss: 0.5143 - val_accuracy: 0.7644
Epoch 13/100
26/26 [==============================] - 0s 19ms/step - loss: 0.5033 - accuracy: 0.8010 -
val_loss: 0.5098 - val_accuracy: 0.7644
Epoch 14/100
26/26 [==============================] - 1s 22ms/step - loss: 0.4993 - accuracy: 0.8035 -
val_loss: 0.4939 - val_accuracy: 0.7931
Epoch 15/100
26/26 [==============================] - 0s 19ms/step - loss: 0.4794 - accuracy: 0.8146 -
val_loss: 0.4878 - val_accuracy: 0.7874
Epoch 16/100
26/26 [==============================] - 1s 20ms/step - loss: 0.4676 - accuracy: 0.8195 -
val_loss: 0.4803 - val_accuracy: 0.8132
Epoch 17/100
26/26 [==============================] - 1s 20ms/step - loss: 0.4598 - accuracy: 0.8282 -
val_loss: 0.4698 - val_accuracy: 0.8161
Epoch 18/100
26/26 [==============================] - 1s 23ms/step - loss: 0.4623 - accuracy: 0.8171 -
val_loss: 0.4647 - val_accuracy: 0.8190
Epoch 19/100
26/26 [==============================] - 0s 17ms/step - loss: 0.4635 - accuracy: 0.8245 -
val_loss: 0.4623 - val_accuracy: 0.8218
Epoch 20/100
26/26 [==============================] - 1s 24ms/step - loss: 0.4505 - accuracy: 0.8257 -
val_loss: 0.4655 - val_accuracy: 0.8190
Epoch 21/100
26/26 [==============================] - 1s 25ms/step - loss: 0.4659 - accuracy: 0.8282 -
val_loss: 0.4610 - val_accuracy: 0.8247
Epoch 22/100
26/26 [==============================] - 0s 17ms/step - loss: 0.4631 - accuracy: 0.8133 -
val_loss: 0.4552 - val_accuracy: 0.8247
Epoch 23/100
Epoch 23/100
26/26 [==============================] - 0s 18ms/step - loss: 0.4604 - accuracy: 0.8171 -
val_loss: 0.4613 - val_accuracy: 0.8017
Epoch 24/100
26/26 [==============================] - 0s 18ms/step - loss: 0.4463 - accuracy: 0.8195 -
val_loss: 0.4551 - val_accuracy: 0.8161
Epoch 25/100
26/26 [==============================] - 1s 20ms/step - loss: 0.4145 - accuracy: 0.8368 -
val_loss: 0.4535 - val_accuracy: 0.8161
Epoch 26/100
26/26 [==============================] - 0s 19ms/step - loss: 0.4243 - accuracy: 0.8418 -
val_loss: 0.4476 - val_accuracy: 0.8190
Epoch 27/100
26/26 [==============================] - 0s 18ms/step - loss: 0.4354 - accuracy: 0.8418 -
val_loss: 0.4440 - val_accuracy: 0.8190
Epoch 28/100
26/26 [==============================] - 0s 18ms/step - loss: 0.4215 - accuracy: 0.8344 -
val_loss: 0.4375 - val_accuracy: 0.8247
Epoch 29/100
26/26 [==============================] - 1s 21ms/step - loss: 0.4536 - accuracy: 0.8245 -
val_loss: 0.4371 - val_accuracy: 0.8190
Epoch 30/100
26/26 [==============================] - 0s 17ms/step - loss: 0.4117 - accuracy: 0.8344 -
val_loss: 0.4348 - val_accuracy: 0.8218
Epoch 31/100
26/26 [==============================] - 0s 17ms/step - loss: 0.4234 - accuracy: 0.8393 -
val_loss: 0.4316 - val_accuracy: 0.8276
Epoch 32/100
26/26 [==============================] - 0s 18ms/step - loss: 0.4204 - accuracy: 0.8381 -
val_loss: 0.4313 - val_accuracy: 0.8218
Epoch 33/100
26/26 [==============================] - 0s 17ms/step - loss: 0.4422 - accuracy: 0.8195 -
val_loss: 0.4310 - val_accuracy: 0.8218
Epoch 34/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4083 - accuracy: 0.8467 -
val_loss: 0.4235 - val_accuracy: 0.8333
Epoch 35/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4078 - accuracy: 0.8405 -
val_loss: 0.4192 - val_accuracy: 0.8362
Epoch 36/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3811 - accuracy: 0.8616 -
val_loss: 0.4178 - val_accuracy: 0.8420
Epoch 37/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4060 - accuracy: 0.8418 -
val_loss: 0.4167 - val_accuracy: 0.8420
Epoch 38/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4098 - accuracy: 0.8368 -
val_loss: 0.4173 - val_accuracy: 0.8448
Epoch 39/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4188 - accuracy: 0.8368 -
val_loss: 0.4165 - val_accuracy: 0.8391
Epoch 40/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4052 - accuracy: 0.8418 -
val_loss: 0.4143 - val_accuracy: 0.8420
Epoch 41/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3947 - accuracy: 0.8430 -
val_loss: 0.4112 - val_accuracy: 0.8420
Epoch 42/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3879 - accuracy: 0.8467 -
val_loss: 0.4031 - val_accuracy: 0.8420
Epoch 43/100
26/26 [==============================] - 0s 18ms/step - loss: 0.3754 - accuracy: 0.8541 -
val_loss: 0.4011 - val_accuracy: 0.8362
Epoch 44/100
26/26 [==============================] - 1s 25ms/step - loss: 0.3753 - accuracy: 0.8504 -
val_loss: 0.3961 - val_accuracy: 0.8420
Epoch 45/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3776 - accuracy: 0.8541 -
val_loss: 0.3952 - val_accuracy: 0.8391
Epoch 46/100
26/26 [==============================] - 0s 18ms/step - loss: 0.3781 - accuracy: 0.8443 -
val_loss: 0.3905 - val_accuracy: 0.8420
Epoch 47/100
Epoch 47/100
26/26 [==============================] - 1s 20ms/step - loss: 0.3690 - accuracy: 0.8566 -
val_loss: 0.3855 - val_accuracy: 0.8391
Epoch 48/100
26/26 [==============================] - 1s 22ms/step - loss: 0.3689 - accuracy: 0.8541 -
val_loss: 0.3860 - val_accuracy: 0.8333
Epoch 49/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3490 - accuracy: 0.8628 -
val_loss: 0.3847 - val_accuracy: 0.8362
Epoch 50/100
26/26 [==============================] - 1s 22ms/step - loss: 0.3537 - accuracy: 0.8690 -
val_loss: 0.3782 - val_accuracy: 0.8391
Epoch 51/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3720 - accuracy: 0.8554 -
val_loss: 0.3743 - val_accuracy: 0.8391
Epoch 52/100
26/26 [==============================] - 1s 23ms/step - loss: 0.3846 - accuracy: 0.8381 -
val_loss: 0.3692 - val_accuracy: 0.8420
Epoch 53/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3690 - accuracy: 0.8455 -
val_loss: 0.3699 - val_accuracy: 0.8305
Epoch 54/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3899 - accuracy: 0.8381 -
val_loss: 0.3675 - val_accuracy: 0.8391
Epoch 55/100
26/26 [==============================] - 0s 18ms/step - loss: 0.3599 - accuracy: 0.8467 -
val_loss: 0.3628 - val_accuracy: 0.8391
Epoch 56/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3421 - accuracy: 0.8504 -
val_loss: 0.3612 - val_accuracy: 0.8333
Epoch 57/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3490 - accuracy: 0.8566 -
val_loss: 0.3580 - val_accuracy: 0.8362
Epoch 58/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3632 - accuracy: 0.8480 -
val_loss: 0.3519 - val_accuracy: 0.8420
Epoch 59/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3454 - accuracy: 0.8603 -
val_loss: 0.3499 - val_accuracy: 0.8420
Epoch 60/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3412 - accuracy: 0.8578 -
val_loss: 0.3482 - val_accuracy: 0.8391
Epoch 61/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3601 - accuracy: 0.8393 -
val_loss: 0.3457 - val_accuracy: 0.8391
Epoch 62/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3195 - accuracy: 0.8603 -
val_loss: 0.3409 - val_accuracy: 0.8391
Epoch 63/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3403 - accuracy: 0.8504 -
val_loss: 0.3397 - val_accuracy: 0.8391
Epoch 64/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3374 - accuracy: 0.8603 -
val_loss: 0.3382 - val_accuracy: 0.8391
Epoch 65/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3223 - accuracy: 0.8764 -
val_loss: 0.3377 - val_accuracy: 0.8448
Epoch 66/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3489 - accuracy: 0.8517 -
val_loss: 0.3365 - val_accuracy: 0.8391
Epoch 67/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3377 - accuracy: 0.8591 -
val_loss: 0.3378 - val_accuracy: 0.8592
Epoch 68/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3521 - accuracy: 0.8566 -
val_loss: 0.3370 - val_accuracy: 0.8448
Epoch 69/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3334 - accuracy: 0.8578 -
val_loss: 0.3313 - val_accuracy: 0.8477
Epoch 70/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3282 - accuracy: 0.8541 -
val_loss: 0.3276 - val_accuracy: 0.8506
Epoch 71/100
Epoch 71/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3302 - accuracy: 0.8591 -
val_loss: 0.3278 - val_accuracy: 0.8534
Epoch 72/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3303 - accuracy: 0.8690 -
val_loss: 0.3225 - val_accuracy: 0.8592
Epoch 73/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3164 - accuracy: 0.8727 -
val_loss: 0.3209 - val_accuracy: 0.8592
Epoch 74/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3413 - accuracy: 0.8492 -
val_loss: 0.3209 - val_accuracy: 0.8621
Epoch 75/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3306 - accuracy: 0.8690 -
val_loss: 0.3225 - val_accuracy: 0.8563
Epoch 76/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3192 - accuracy: 0.8702 -
val_loss: 0.3180 - val_accuracy: 0.8563
Epoch 77/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3357 - accuracy: 0.8628 -
val_loss: 0.3170 - val_accuracy: 0.8563
Epoch 78/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3078 - accuracy: 0.8665 -
val_loss: 0.3168 - val_accuracy: 0.8592
Epoch 79/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3043 - accuracy: 0.8776 -
val_loss: 0.3163 - val_accuracy: 0.8649
Epoch 80/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3201 - accuracy: 0.8813 -
val_loss: 0.3137 - val_accuracy: 0.8764
Epoch 81/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3051 - accuracy: 0.8764 -
val_loss: 0.3134 - val_accuracy: 0.8592
Epoch 82/100
26/26 [==============================] - 0s 16ms/step - loss: 0.2935 - accuracy: 0.8727 -
val_loss: 0.3120 - val_accuracy: 0.8678
Epoch 83/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3243 - accuracy: 0.8677 -
val_loss: 0.3125 - val_accuracy: 0.8707
Epoch 84/100
26/26 [==============================] - 0s 17ms/step - loss: 0.2926 - accuracy: 0.8789 -
val_loss: 0.3078 - val_accuracy: 0.8649
Epoch 85/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3114 - accuracy: 0.8616 -
val_loss: 0.3085 - val_accuracy: 0.8764
Epoch 86/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3138 - accuracy: 0.8714 -
val_loss: 0.3081 - val_accuracy: 0.8678
Epoch 87/100
26/26 [==============================] - 0s 16ms/step - loss: 0.2965 - accuracy: 0.8677 -
val_loss: 0.3040 - val_accuracy: 0.8707
Epoch 88/100
26/26 [==============================] - 0s 15ms/step - loss: 0.2836 - accuracy: 0.8900 -
val_loss: 0.3057 - val_accuracy: 0.8736
Epoch 89/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3011 - accuracy: 0.8727 -
val_loss: 0.3044 - val_accuracy: 0.8793
Epoch 90/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3136 - accuracy: 0.8727 -
val_loss: 0.3059 - val_accuracy: 0.8764
Epoch 91/100
26/26 [==============================] - 0s 16ms/step - loss: 0.2917 - accuracy: 0.8826 -
val_loss: 0.3017 - val_accuracy: 0.8793
Epoch 92/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3298 - accuracy: 0.8566 -
val_loss: 0.2998 - val_accuracy: 0.8736
Epoch 93/100
26/26 [==============================] - 0s 15ms/step - loss: 0.2567 - accuracy: 0.8937 -
val_loss: 0.2962 - val_accuracy: 0.8822
Epoch 94/100
26/26 [==============================] - 0s 16ms/step - loss: 0.2928 - accuracy: 0.8764 -
val_loss: 0.2953 - val_accuracy: 0.8764
Epoch 95/100
Epoch 95/100
26/26 [==============================] - 0s 16ms/step - loss: 0.2659 - accuracy: 0.9023 -
val_loss: 0.2928 - val_accuracy: 0.8851
Epoch 94: Reached baseline, terminating training
```

In [201]:

```
plt.plot(ctrain1.history['accuracy' ])
plt.plot(ctrain1.history['val_accuracy'])
plt.plot(ctrain1.history['loss'])
plt.plot(ctrain1.history['val_loss'])
plt.legend(['train acc', 'valid acc', 'train loss', 'valid loss'], loc = 'upper left')
plt.show()
```



Test Accuracy

In [202]:

```
cmodel_out1=cmodel1.evaluate(X_test,y_test)
print(f"Test Accuracy is: {cmodel_out1[1]*100}%")
```

```
18/18 [==============================] - 0s 8ms/step - loss: 0.2250 - accuracy: 0.9053
Test Accuracy is: 90.52631855010986%
```

Loading the weights into the model from car1.h5

In [204]:

```
model1car=car_model()
```

In [205]:

```
model1car.load_weights('car1.h5')
```

In [206]:

```
car1model_out1=model1car.evaluate(X_test,y_test)
print(f"Test Accuracy is: {car1model_out1[1]*100}%")
```

```
18/18 [==============================] - 0s 9ms/step - loss: 0.2250 - accuracy: 0.9053
Test Accuracy is: 90.52631855010986%
```

Print Confusion Matrix

In [207]:

```
def cprint_conf(model):
    labels=y_test
    y_pred=model.predict(X_test)
    diffmatrix = confusion_matrix(labels.argmax(axis=1), y_pred.argmax(axis=1))
    return diffmatrix
```

In [208]:

```
car_cm1=cprint_conf(cmodel1)
```

In [209]:

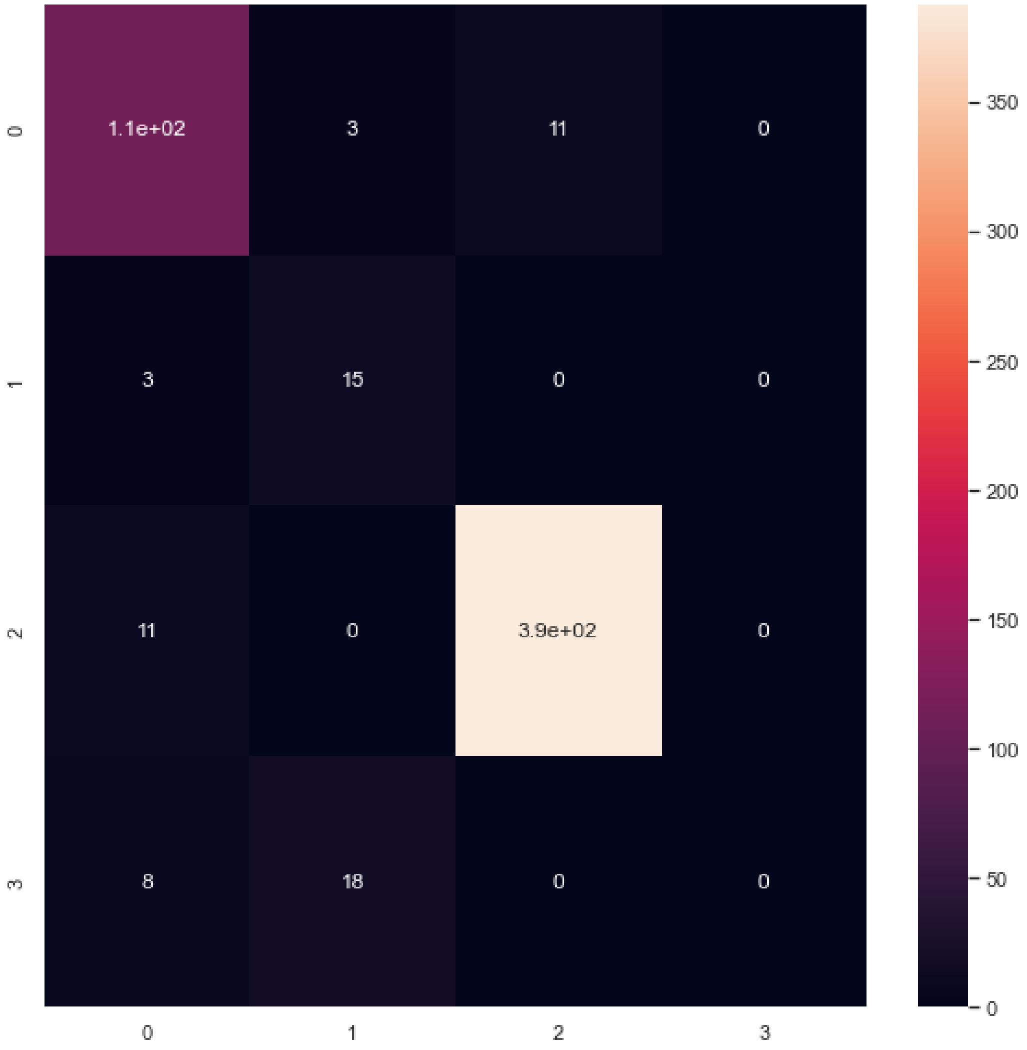
```
car_cm1
```

Out[209]:

```
array([[113,   3,  11,   0],
       [  3,  15,   0,   0],
       [ 11,   0, 388,   0],
       [  8,  18,   0,   0]], dtype=int64)
```

In [210]:

```
matrix(car_cm1)
```



2) Ensemble Learning model using Voting Classifier

In [211]:

```
y_train=np.argmax(y_train,axis=1)
```

In [212]:

```
y_train
```

Out[212]:

```
array([2, 2, 2, ..., 2, 2, 0], dtype=int64)
```

In [213]:

```
y_test=np.argmax(y_test,axis=1)
y_test
```

Out[213]:

```
array([2, 2, 2, 0, 2, 0, 2, 2, 2, 2, 3, 2, 2, 2, 2, 2, 2, 2, 2, 0, 2, 2,
       0, 2, 2, 0, 2, 2, 2, 2, 0, 2, 2, 1, 2, 2, 1, 2, 2, 3, 0, 0, 2, 2,
       0, 3, 2, 2, 2, 2, 2, 2, 0, 0, 2, 2, 2, 2, 2, 0, 2, 2, 2, 2, 2, 3,
       2, 2, 1, 2, 3, 0, 2, 2, 2, 0, 2, 2, 2, 2, 3, 2, 2, 0, 2, 0, 2, 2,
       0, 0, 2, 1, 2, 2, 2, 2, 2, 2, 2, 0, 2, 2, 2, 2, 2, 1, 0, 2, 2, 0,
       2, 0, 2, 2, 0, 2, 2, 2, 2, 2, 2, 0, 2, 0, 0, 2, 0, 2, 2, 2, 2, 2,
       2, 2, 2, 2, 2, 2, 2, 2, 2, 0, 2, 0, 2, 0, 2, 2, 0, 2, 0, 3, 2, 2,
       2, 2, 2, 2, 2, 2, 0, 0, 2, 3, 2, 3, 0, 0, 2, 3, 0, 2, 3, 2, 0, 0,
       3, 2, 0, 2, 0, 2, 2, 0, 1, 2, 2, 2, 2, 2, 2, 0, 0, 2, 2, 2, 0, 2,
       2, 2, 2, 2, 2, 3, 2, 0, 2, 0, 2, 1, 2, 2, 0, 2, 2, 2, 2, 2, 2, 0,
       3, 2, 2, 0, 1, 0, 0, 0, 2, 2, 0, 2, 2, 2, 0, 2, 0, 2, 2, 2, 2, 2,
       2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 0, 0, 2, 2, 2, 2, 2, 0, 2, 1, 2,
       2, 2, 0, 0, 2, 2, 2, 2, 0, 2, 0, 0, 2, 1, 2, 2, 2, 1, 2, 0, 1, 0,
       2, 1, 3, 2, 1, 2, 2, 2, 0, 3, 2, 2, 2, 2, 0, 0, 2, 2, 2, 2, 2, 2,
       0, 2, 3, 2, 2, 2, 2, 0, 0, 0, 2, 2, 0, 2, 2, 2, 2, 2, 1, 0, 2, 2,
       2, 2, 2, 2, 2, 2, 2, 0, 2, 0, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 0,
       0, 2, 0, 2, 2, 2, 3, 2, 2, 2, 0, 0, 0, 2, 3, 0, 2, 0, 2, 3, 2, 0,
       2, 2, 2, 0, 2, 0, 0, 2, 0, 2, 2, 2, 0, 2, 2, 2, 2, 0, 0, 2, 2, 2,
       2, 0, 0, 2, 2, 2, 2, 2, 2, 0, 2, 2, 2, 2, 2, 2, 0, 0, 0, 0, 3, 0,
       2, 2, 2, 0, 0, 0, 2, 2, 2, 2, 0, 2, 1, 2, 2, 2, 0, 0, 2, 2, 2, 2,
       2, 0, 2, 2, 2, 2, 2, 2, 2, 0, 2, 3, 2, 2, 2, 2, 2, 0, 3, 2, 2, 2,
       2, 2, 2, 2, 2, 2, 0, 2, 2, 2, 2, 2, 2, 0, 2, 2, 2, 2, 2, 2, 0, 2,
       2, 0, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 0, 2, 0, 2,
       0, 2, 2, 2, 2, 2, 0, 2, 2, 2, 2, 2, 1, 3, 2, 0, 0, 2, 2, 2, 2, 0,
       0, 2, 0, 2, 2, 2, 0, 2, 3, 2, 2, 2, 2, 2, 2, 3, 2, 2, 2, 2, 0, 2,
       2, 0, 0, 2, 2, 2, 2, 2, 2, 2, 2, 1, 2, 2, 2, 2, 2, 2, 2, 2],
      dtype=int64)
```

In [214]:

```
ecallback1 = [TerminateOnBaseline()]
```

In [215]:

```
checkpoint_filepath3 = 'C:\\Users\\Surya\\Desktop\\College\\6th Semester\\Neural Network
s and Deep Learning\\Lab Assignments\\Lab-4\\ecar.h5'
checkpoint4 = tf.keras.callbacks.ModelCheckpoint(
    filepath=checkpoint_filepath3,
    save_weights_only=False,
    monitor='accuracy',
    mode='auto',
    save_best_only=True)
```

Intializing 3 models for Ensemble learning

In [216]:

```
emodel7=KerasClassifier(build_fn=car_model,epochs=100,validation_split=0.3,callbacks=[eca
llback1,checkpoint4],verbose=1)
emodel8=KerasClassifier(build_fn=car_model,epochs=100,validation_split=0.3,callbacks=[eca
llback1,checkpoint4],verbose=1)
emodel9=KerasClassifier(build_fn=car_model,epochs=100,validation_split=0.3,callbacks=[eca
llback1,checkpoint4],verbose=1)
emodel7._estimator_type="classifier"
emodel8._estimator_type="classifier"
emodel9._estimator_type="classifier"
```

In [217]:

```
ensemble3=VotingClassifier(estimators=[('model1',emodel7),('model2',emodel8),('model3',e
model9)],voting='soft')
```

In [218]:

```
etrain4=ensemble3.fit(X_train,y_train)
```

```
Epoch 1/100
26/26 [==============================] - 2s 34ms/step - loss: 1.3751 - accuracy: 0.3004 -
val_loss: 1.1392 - val_accuracy: 0.6667
Epoch 2/100
26/26 [==============================] - 1s 20ms/step - loss: 0.9543 - accuracy: 0.6625 -
val_loss: 0.9548 - val_accuracy: 0.6667
Epoch 3/100
26/26 [==============================] - 0s 19ms/step - loss: 0.8112 - accuracy: 0.7070 -
val_loss: 0.9001 - val_accuracy: 0.6667
Epoch 4/100
26/26 [==============================] - 1s 20ms/step - loss: 0.7469 - accuracy: 0.7231 -
val_loss: 0.8534 - val_accuracy: 0.6667
Epoch 5/100
26/26 [==============================] - 0s 18ms/step - loss: 0.7020 - accuracy: 0.7268 -
val_loss: 0.7971 - val_accuracy: 0.6667
Epoch 6/100
26/26 [==============================] - 0s 18ms/step - loss: 0.6506 - accuracy: 0.7590 -
val_loss: 0.7352 - val_accuracy: 0.6667
Epoch 7/100
26/26 [==============================] - 1s 20ms/step - loss: 0.6291 - accuracy: 0.7454 -
val_loss: 0.6668 - val_accuracy: 0.6695
Epoch 8/100
26/26 [==============================] - 0s 18ms/step - loss: 0.5981 - accuracy: 0.7614 -
val_loss: 0.6168 - val_accuracy: 0.7443
Epoch 9/100
26/26 [==============================] - 0s 18ms/step - loss: 0.5867 - accuracy: 0.7726 -
val_loss: 0.5842 - val_accuracy: 0.7931
Epoch 10/100
26/26 [==============================] - 0s 17ms/step - loss: 0.5636 - accuracy: 0.7824 -
val_loss: 0.5581 - val_accuracy: 0.8161
Epoch 11/100
26/26 [==============================] - 0s 18ms/step - loss: 0.5607 - accuracy: 0.7763 -
val_loss: 0.5432 - val_accuracy: 0.8218
Epoch 12/100
26/26 [==============================] - 0s 17ms/step - loss: 0.5430 - accuracy: 0.7936 -
val_loss: 0.5334 - val_accuracy: 0.8276
Epoch 13/100
26/26 [==============================] - 0s 17ms/step - loss: 0.5272 - accuracy: 0.8084 -
val_loss: 0.5217 - val_accuracy: 0.8276
Epoch 14/100
26/26 [==============================] - 0s 16ms/step - loss: 0.5247 - accuracy: 0.7960 -
val_loss: 0.5129 - val_accuracy: 0.8305
Epoch 15/100
26/26 [==============================] - 0s 16ms/step - loss: 0.5103 - accuracy: 0.8022 -
val_loss: 0.5053 - val_accuracy: 0.8362
Epoch 16/100
26/26 [==============================] - 0s 17ms/step - loss: 0.5015 - accuracy: 0.8096 -
val_loss: 0.5003 - val_accuracy: 0.8276
Epoch 17/100
26/26 [==============================] - 0s 18ms/step - loss: 0.4772 - accuracy: 0.8158 -
val_loss: 0.4935 - val_accuracy: 0.8218
Epoch 18/100
26/26 [==============================] - 0s 17ms/step - loss: 0.5067 - accuracy: 0.8022 -
val_loss: 0.4874 - val_accuracy: 0.8276
Epoch 19/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4868 - accuracy: 0.8133 -
val_loss: 0.4877 - val_accuracy: 0.8247
Epoch 20/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4600 - accuracy: 0.8245 -
val_loss: 0.4841 - val_accuracy: 0.8276
Epoch 21/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4798 - accuracy: 0.8084 -
val_loss: 0.4808 - val_accuracy: 0.8218
Epoch 22/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4530 - accuracy: 0.8269 -
val_loss: 0.4778 - val_accuracy: 0.8218
Epoch 23/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4563 - accuracy: 0.8195 -
val_loss: 0.4712 - val_accuracy: 0.8247
Epoch 24/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4324 - accuracy: 0.8368 -
val_loss: 0.4694 - val_accuracy: 0.8276
```

```
Epoch 25/100
26/26 [==============================] - 0s 17ms/step - loss: 0.4471 - accuracy: 0.8344 -
val_loss: 0.4634 - val_accuracy: 0.8305
Epoch 26/100
26/26 [==============================] - 0s 17ms/step - loss: 0.4745 - accuracy: 0.8232 -
val_loss: 0.4600 - val_accuracy: 0.8305
Epoch 27/100
26/26 [==============================] - 0s 18ms/step - loss: 0.4652 - accuracy: 0.8257 -
val_loss: 0.4544 - val_accuracy: 0.8333
Epoch 28/100
26/26 [==============================] - 1s 20ms/step - loss: 0.4522 - accuracy: 0.8319 -
val_loss: 0.4525 - val_accuracy: 0.8247
Epoch 29/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4570 - accuracy: 0.8133 -
val_loss: 0.4491 - val_accuracy: 0.8333
Epoch 30/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4333 - accuracy: 0.8368 -
val_loss: 0.4446 - val_accuracy: 0.8276
Epoch 31/100
26/26 [==============================] - 0s 17ms/step - loss: 0.4336 - accuracy: 0.8307 -
val_loss: 0.4449 - val_accuracy: 0.8305
Epoch 32/100
26/26 [==============================] - 0s 17ms/step - loss: 0.4051 - accuracy: 0.8566 -
val_loss: 0.4438 - val_accuracy: 0.8362
Epoch 33/100
26/26 [==============================] - 0s 17ms/step - loss: 0.4197 - accuracy: 0.8492 -
val_loss: 0.4386 - val_accuracy: 0.8391
Epoch 34/100
26/26 [==============================] - 1s 22ms/step - loss: 0.4384 - accuracy: 0.8208 -
val_loss: 0.4394 - val_accuracy: 0.8276
Epoch 35/100
26/26 [==============================] - 0s 17ms/step - loss: 0.4269 - accuracy: 0.8319 -
val_loss: 0.4318 - val_accuracy: 0.8305
Epoch 36/100
26/26 [==============================] - 0s 17ms/step - loss: 0.4341 - accuracy: 0.8269 -
val_loss: 0.4283 - val_accuracy: 0.8305
Epoch 37/100
26/26 [==============================] - 0s 17ms/step - loss: 0.4346 - accuracy: 0.8269 -
val_loss: 0.4254 - val_accuracy: 0.8247
Epoch 38/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3883 - accuracy: 0.8504 -
val_loss: 0.4254 - val_accuracy: 0.8305
Epoch 39/100
26/26 [==============================] - 1s 20ms/step - loss: 0.4083 - accuracy: 0.8368 -
val_loss: 0.4194 - val_accuracy: 0.8362
Epoch 40/100
26/26 [==============================] - 0s 17ms/step - loss: 0.4358 - accuracy: 0.8183 -
val_loss: 0.4201 - val_accuracy: 0.8305
Epoch 41/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4047 - accuracy: 0.8319 -
val_loss: 0.4209 - val_accuracy: 0.8362
Epoch 42/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3909 - accuracy: 0.8443 -
val_loss: 0.4202 - val_accuracy: 0.8391
Epoch 43/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3876 - accuracy: 0.8504 -
val_loss: 0.4140 - val_accuracy: 0.8362
Epoch 44/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4310 - accuracy: 0.8171 -
val_loss: 0.4113 - val_accuracy: 0.8362
Epoch 45/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3910 - accuracy: 0.8541 -
val_loss: 0.4052 - val_accuracy: 0.8333
Epoch 46/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3684 - accuracy: 0.8578 -
val_loss: 0.4011 - val_accuracy: 0.8333
Epoch 47/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3886 - accuracy: 0.8517 -
val_loss: 0.4007 - val_accuracy: 0.8391
Epoch 48/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3668 - accuracy: 0.8578 -
val_loss: 0.3940 - val_accuracy: 0.8448
```

```
Epoch 49/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3653 - accuracy: 0.8616 -
val_loss: 0.3914 - val_accuracy: 0.8448
Epoch 50/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3776 - accuracy: 0.8591 -
val_loss: 0.3899 - val_accuracy: 0.8448
Epoch 51/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3615 - accuracy: 0.8443 -
val_loss: 0.3886 - val_accuracy: 0.8391
Epoch 52/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3762 - accuracy: 0.8430 -
val_loss: 0.3854 - val_accuracy: 0.8420
Epoch 53/100
26/26 [==============================] - 0s 18ms/step - loss: 0.3740 - accuracy: 0.8541 -
val_loss: 0.3778 - val_accuracy: 0.8391
Epoch 54/100
26/26 [==============================] - 0s 18ms/step - loss: 0.3941 - accuracy: 0.8405 -
val_loss: 0.3733 - val_accuracy: 0.8448
Epoch 55/100
26/26 [==============================] - 1s 19ms/step - loss: 0.3634 - accuracy: 0.8529 -
val_loss: 0.3699 - val_accuracy: 0.8420
Epoch 56/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3603 - accuracy: 0.8591 -
val_loss: 0.3697 - val_accuracy: 0.8420
Epoch 57/100
26/26 [==============================] - 0s 18ms/step - loss: 0.3459 - accuracy: 0.8566 -
val_loss: 0.3658 - val_accuracy: 0.8362
Epoch 58/100
26/26 [==============================] - 1s 20ms/step - loss: 0.3532 - accuracy: 0.8591 -
val_loss: 0.3645 - val_accuracy: 0.8362
Epoch 59/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3751 - accuracy: 0.8418 -
val_loss: 0.3654 - val_accuracy: 0.8391
Epoch 60/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3556 - accuracy: 0.8381 -
val_loss: 0.3602 - val_accuracy: 0.8333
Epoch 61/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3594 - accuracy: 0.8430 -
val_loss: 0.3565 - val_accuracy: 0.8333
Epoch 62/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3491 - accuracy: 0.8455 -
val_loss: 0.3554 - val_accuracy: 0.8391
Epoch 63/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3591 - accuracy: 0.8616 -
val_loss: 0.3541 - val_accuracy: 0.8362
Epoch 64/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3651 - accuracy: 0.8504 -
val_loss: 0.3528 - val_accuracy: 0.8391
Epoch 65/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3535 - accuracy: 0.8578 -
val_loss: 0.3509 - val_accuracy: 0.8420
Epoch 66/100
26/26 [==============================] - 0s 19ms/step - loss: 0.3650 - accuracy: 0.8504 -
val_loss: 0.3489 - val_accuracy: 0.8391
Epoch 67/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3704 - accuracy: 0.8517 -
val_loss: 0.3482 - val_accuracy: 0.8391
Epoch 68/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3362 - accuracy: 0.8529 -
val_loss: 0.3451 - val_accuracy: 0.8448
Epoch 69/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3343 - accuracy: 0.8591 -
val_loss: 0.3421 - val_accuracy: 0.8448
Epoch 70/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3589 - accuracy: 0.8517 -
val_loss: 0.3360 - val_accuracy: 0.8362
Epoch 71/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3377 - accuracy: 0.8566 -
val_loss: 0.3348 - val_accuracy: 0.8362
Epoch 72/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3503 - accuracy: 0.8541 -
val_loss: 0.3349 - val_accuracy: 0.8420
```

```
Epoch 73/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3285 - accuracy: 0.8690 -
val_loss: 0.3295 - val_accuracy: 0.8448
Epoch 74/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3280 - accuracy: 0.8677 -
val_loss: 0.3278 - val_accuracy: 0.8448
Epoch 75/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3203 - accuracy: 0.8702 -
val_loss: 0.3261 - val_accuracy: 0.8477
Epoch 76/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3231 - accuracy: 0.8690 -
val_loss: 0.3240 - val_accuracy: 0.8448
Epoch 77/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3415 - accuracy: 0.8603 -
val_loss: 0.3234 - val_accuracy: 0.8506
Epoch 78/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3154 - accuracy: 0.8665 -
val_loss: 0.3195 - val_accuracy: 0.8477
Epoch 79/100
26/26 [==============================] - 1s 21ms/step - loss: 0.3158 - accuracy: 0.8677 -
val_loss: 0.3199 - val_accuracy: 0.8592
Epoch 80/100
26/26 [==============================] - 0s 18ms/step - loss: 0.3230 - accuracy: 0.8665 -
val_loss: 0.3155 - val_accuracy: 0.8534
Epoch 81/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3234 - accuracy: 0.8727 -
val_loss: 0.3156 - val_accuracy: 0.8649
Epoch 82/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3200 - accuracy: 0.8616 -
val_loss: 0.3128 - val_accuracy: 0.8563
Epoch 83/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3126 - accuracy: 0.8727 -
val_loss: 0.3114 - val_accuracy: 0.8678
Epoch 84/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3344 - accuracy: 0.8628 -
val_loss: 0.3096 - val_accuracy: 0.8649
Epoch 85/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3251 - accuracy: 0.8603 -
val_loss: 0.3087 - val_accuracy: 0.8621
Epoch 86/100
26/26 [==============================] - 1s 20ms/step - loss: 0.3155 - accuracy: 0.8640 -
val_loss: 0.3102 - val_accuracy: 0.8649
Epoch 87/100
26/26 [==============================] - 1s 20ms/step - loss: 0.3110 - accuracy: 0.8776 -
val_loss: 0.3100 - val_accuracy: 0.8736
Epoch 88/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3067 - accuracy: 0.8714 -
val_loss: 0.3123 - val_accuracy: 0.8649
Epoch 89/100
26/26 [==============================] - 0s 18ms/step - loss: 0.3004 - accuracy: 0.8727 -
val_loss: 0.3152 - val_accuracy: 0.8506
Epoch 90/100
26/26 [==============================] - 0s 16ms/step - loss: 0.2951 - accuracy: 0.8752 -
val_loss: 0.3101 - val_accuracy: 0.8563
Epoch 91/100
26/26 [==============================] - 0s 16ms/step - loss: 0.2940 - accuracy: 0.8801 -
val_loss: 0.3006 - val_accuracy: 0.8592
Epoch 92/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3144 - accuracy: 0.8789 -
val_loss: 0.3005 - val_accuracy: 0.8678
Epoch 93/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3071 - accuracy: 0.8665 -
val_loss: 0.2972 - val_accuracy: 0.8649
Epoch 94/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3187 - accuracy: 0.8727 -
val_loss: 0.2957 - val_accuracy: 0.8707
Epoch 95/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3258 - accuracy: 0.8677 -
val_loss: 0.2964 - val_accuracy: 0.8764
Epoch 96/100
26/26 [==============================] - 0s 16ms/step - loss: 0.2952 - accuracy: 0.8813 -
val_loss: 0.2941 - val_accuracy: 0.8793
```

```
Epoch 97/100
26/26 [==============================] - 0s 16ms/step - loss: 0.2976 - accuracy: 0.8789 -
val_loss: 0.2935 - val_accuracy: 0.8822
Epoch 98/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3023 - accuracy: 0.8702 -
val_loss: 0.2941 - val_accuracy: 0.8822
Epoch 99/100
26/26 [==============================] - 0s 15ms/step - loss: 0.2869 - accuracy: 0.8739 -
val_loss: 0.2907 - val_accuracy: 0.8764
Epoch 100/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3181 - accuracy: 0.8764 -
val_loss: 0.2892 - val_accuracy: 0.8793
Epoch 1/100
26/26 [==============================] - 2s 24ms/step - loss: 1.3877 - accuracy: 0.3226 -
val_loss: 1.0172 - val_accuracy: 0.6667
Epoch 2/100
26/26 [==============================] - 0s 16ms/step - loss: 0.9661 - accuracy: 0.6354 -
val_loss: 0.9589 - val_accuracy: 0.6667
Epoch 3/100
26/26 [==============================] - 0s 16ms/step - loss: 0.8362 - accuracy: 0.7070 -
val_loss: 0.9649 - val_accuracy: 0.6667
Epoch 4/100
26/26 [==============================] - 0s 16ms/step - loss: 0.7565 - accuracy: 0.7145 -
val_loss: 0.9800 - val_accuracy: 0.6667
Epoch 5/100
26/26 [==============================] - 0s 16ms/step - loss: 0.6844 - accuracy: 0.7478 -
val_loss: 0.9812 - val_accuracy: 0.6667
Epoch 6/100
26/26 [==============================] - 0s 16ms/step - loss: 0.6492 - accuracy: 0.7417 -
val_loss: 0.9543 - val_accuracy: 0.6667
Epoch 7/100
26/26 [==============================] - 0s 15ms/step - loss: 0.6124 - accuracy: 0.7515 -
val_loss: 0.9023 - val_accuracy: 0.6667
Epoch 8/100
26/26 [==============================] - 0s 16ms/step - loss: 0.5807 - accuracy: 0.7651 -
val_loss: 0.8413 - val_accuracy: 0.6667
Epoch 9/100
26/26 [==============================] - 0s 15ms/step - loss: 0.5541 - accuracy: 0.7911 -
val_loss: 0.7642 - val_accuracy: 0.6667
Epoch 10/100
26/26 [==============================] - 0s 16ms/step - loss: 0.5579 - accuracy: 0.7787 -
val_loss: 0.6963 - val_accuracy: 0.6695
Epoch 11/100
26/26 [==============================] - 0s 16ms/step - loss: 0.5426 - accuracy: 0.7985 -
val_loss: 0.6370 - val_accuracy: 0.6810
Epoch 12/100
26/26 [==============================] - 1s 23ms/step - loss: 0.5259 - accuracy: 0.7998 -
val_loss: 0.6009 - val_accuracy: 0.7011
Epoch 13/100
26/26 [==============================] - 0s 17ms/step - loss: 0.5175 - accuracy: 0.8035 -
val_loss: 0.5495 - val_accuracy: 0.7529
Epoch 14/100
26/26 [==============================] - 0s 17ms/step - loss: 0.5021 - accuracy: 0.8022 -
val_loss: 0.5335 - val_accuracy: 0.7701
Epoch 15/100
26/26 [==============================] - 0s 17ms/step - loss: 0.5183 - accuracy: 0.7960 -
val_loss: 0.5170 - val_accuracy: 0.7931
Epoch 16/100
26/26 [==============================] - 0s 17ms/step - loss: 0.4943 - accuracy: 0.7899 -
val_loss: 0.5067 - val_accuracy: 0.8017
Epoch 17/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4945 - accuracy: 0.8047 -
val_loss: 0.4983 - val_accuracy: 0.8103
Epoch 18/100
26/26 [==============================] - 0s 18ms/step - loss: 0.4914 - accuracy: 0.7985 -
val_loss: 0.4991 - val_accuracy: 0.7960
Epoch 19/100
26/26 [==============================] - 0s 18ms/step - loss: 0.4646 - accuracy: 0.8195 -
val_loss: 0.4995 - val_accuracy: 0.7989
Epoch 20/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4481 - accuracy: 0.8368 -
val_loss: 0.4882 - val_accuracy: 0.8161
```

```
Epoch 21/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4658 - accuracy: 0.8146 -
val_loss: 0.4856 - val_accuracy: 0.8132
Epoch 22/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4274 - accuracy: 0.8443 -
val_loss: 0.4752 - val_accuracy: 0.8247
Epoch 23/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4425 - accuracy: 0.8269 -
val_loss: 0.4700 - val_accuracy: 0.8218
Epoch 24/100
26/26 [==============================] - 0s 17ms/step - loss: 0.4359 - accuracy: 0.8356 -
val_loss: 0.4659 - val_accuracy: 0.8276
Epoch 25/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4353 - accuracy: 0.8257 -
val_loss: 0.4601 - val_accuracy: 0.8305
Epoch 26/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4672 - accuracy: 0.8158 -
val_loss: 0.4611 - val_accuracy: 0.8305
Epoch 27/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4509 - accuracy: 0.8232 -
val_loss: 0.4560 - val_accuracy: 0.8305
Epoch 28/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4396 - accuracy: 0.8331 -
val_loss: 0.4572 - val_accuracy: 0.8305
Epoch 29/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4301 - accuracy: 0.8294 -
val_loss: 0.4521 - val_accuracy: 0.8276
Epoch 30/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4214 - accuracy: 0.8405 -
val_loss: 0.4514 - val_accuracy: 0.8276
Epoch 31/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4160 - accuracy: 0.8430 -
val_loss: 0.4475 - val_accuracy: 0.8247
Epoch 32/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4203 - accuracy: 0.8480 -
val_loss: 0.4480 - val_accuracy: 0.8276
Epoch 33/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4196 - accuracy: 0.8480 -
val_loss: 0.4458 - val_accuracy: 0.8276
Epoch 34/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4606 - accuracy: 0.8158 -
val_loss: 0.4461 - val_accuracy: 0.8247
Epoch 35/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3980 - accuracy: 0.8529 -
val_loss: 0.4417 - val_accuracy: 0.8247
Epoch 36/100
26/26 [==============================] - 0s 18ms/step - loss: 0.4313 - accuracy: 0.8282 -
val_loss: 0.4420 - val_accuracy: 0.8247
Epoch 37/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4101 - accuracy: 0.8529 -
val_loss: 0.4413 - val_accuracy: 0.8247
Epoch 38/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3927 - accuracy: 0.8504 -
val_loss: 0.4370 - val_accuracy: 0.8247
Epoch 39/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4078 - accuracy: 0.8393 -
val_loss: 0.4332 - val_accuracy: 0.8305
Epoch 40/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4234 - accuracy: 0.8220 -
val_loss: 0.4283 - val_accuracy: 0.8305
Epoch 41/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3988 - accuracy: 0.8517 -
val_loss: 0.4249 - val_accuracy: 0.8333
Epoch 42/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4071 - accuracy: 0.8418 -
val_loss: 0.4214 - val_accuracy: 0.8362
Epoch 43/100
26/26 [==============================] - 0s 17ms/step - loss: 0.4146 - accuracy: 0.8344 -
val_loss: 0.4184 - val_accuracy: 0.8305
Epoch 44/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3972 - accuracy: 0.8430 -
val_loss: 0.4162 - val_accuracy: 0.8362
```

```
Epoch 45/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3892 - accuracy: 0.8430 -
val_loss: 0.4137 - val_accuracy: 0.8333
Epoch 46/100
26/26 [==============================] - 0s 17ms/step - loss: 0.4072 - accuracy: 0.8356 -
val_loss: 0.4076 - val_accuracy: 0.8362
Epoch 47/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3781 - accuracy: 0.8529 -
val_loss: 0.4026 - val_accuracy: 0.8391
Epoch 48/100
26/26 [==============================] - 0s 18ms/step - loss: 0.3791 - accuracy: 0.8443 -
val_loss: 0.4013 - val_accuracy: 0.8448
Epoch 49/100
26/26 [==============================] - 0s 18ms/step - loss: 0.3875 - accuracy: 0.8566 -
val_loss: 0.3998 - val_accuracy: 0.8391
Epoch 50/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4004 - accuracy: 0.8368 -
val_loss: 0.3963 - val_accuracy: 0.8391
Epoch 51/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3819 - accuracy: 0.8430 -
val_loss: 0.3940 - val_accuracy: 0.8420
Epoch 52/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3696 - accuracy: 0.8467 -
val_loss: 0.3922 - val_accuracy: 0.8420
Epoch 53/100
26/26 [==============================] - 0s 18ms/step - loss: 0.3616 - accuracy: 0.8578 -
val_loss: 0.3890 - val_accuracy: 0.8391
Epoch 54/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3581 - accuracy: 0.8628 -
val_loss: 0.3837 - val_accuracy: 0.8448
Epoch 55/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3727 - accuracy: 0.8566 -
val_loss: 0.3794 - val_accuracy: 0.8420
Epoch 56/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3652 - accuracy: 0.8455 -
val_loss: 0.3754 - val_accuracy: 0.8391
Epoch 57/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3547 - accuracy: 0.8554 -
val_loss: 0.3719 - val_accuracy: 0.8420
Epoch 58/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3464 - accuracy: 0.8492 -
val_loss: 0.3681 - val_accuracy: 0.8305
Epoch 59/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3541 - accuracy: 0.8616 -
val_loss: 0.3647 - val_accuracy: 0.8362
Epoch 60/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3660 - accuracy: 0.8455 -
val_loss: 0.3634 - val_accuracy: 0.8333
Epoch 61/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3449 - accuracy: 0.8566 -
val_loss: 0.3579 - val_accuracy: 0.8333
Epoch 62/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3340 - accuracy: 0.8640 -
val_loss: 0.3540 - val_accuracy: 0.8448
Epoch 63/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3422 - accuracy: 0.8529 -
val_loss: 0.3510 - val_accuracy: 0.8448
Epoch 64/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3449 - accuracy: 0.8640 -
val_loss: 0.3501 - val_accuracy: 0.8448
Epoch 65/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3408 - accuracy: 0.8578 -
val_loss: 0.3474 - val_accuracy: 0.8420
Epoch 66/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3317 - accuracy: 0.8603 -
val_loss: 0.3438 - val_accuracy: 0.8420
Epoch 67/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3389 - accuracy: 0.8653 -
val_loss: 0.3428 - val_accuracy: 0.8448
Epoch 68/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3367 - accuracy: 0.8566 -
val_loss: 0.3379 - val_accuracy: 0.8420
```

```
Epoch 69/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3219 - accuracy: 0.8702 -
val_loss: 0.3361 - val_accuracy: 0.8534
Epoch 70/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3443 - accuracy: 0.8640 -
val_loss: 0.3315 - val_accuracy: 0.8448
Epoch 71/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3193 - accuracy: 0.8677 -
val_loss: 0.3259 - val_accuracy: 0.8534
Epoch 72/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3231 - accuracy: 0.8739 -
val_loss: 0.3263 - val_accuracy: 0.8506
Epoch 73/100
26/26 [==============================] - 0s 18ms/step - loss: 0.3258 - accuracy: 0.8677 -
val_loss: 0.3260 - val_accuracy: 0.8563
Epoch 74/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3426 - accuracy: 0.8492 -
val_loss: 0.3250 - val_accuracy: 0.8563
Epoch 75/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3194 - accuracy: 0.8727 -
val_loss: 0.3254 - val_accuracy: 0.8736
Epoch 76/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3319 - accuracy: 0.8541 -
val_loss: 0.3205 - val_accuracy: 0.8563
Epoch 77/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3153 - accuracy: 0.8764 -
val_loss: 0.3196 - val_accuracy: 0.8649
Epoch 78/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3214 - accuracy: 0.8752 -
val_loss: 0.3171 - val_accuracy: 0.8649
Epoch 79/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3197 - accuracy: 0.8628 -
val_loss: 0.3174 - val_accuracy: 0.8649
Epoch 80/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3081 - accuracy: 0.8603 -
val_loss: 0.3167 - val_accuracy: 0.8678
Epoch 81/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3068 - accuracy: 0.8727 -
val_loss: 0.3106 - val_accuracy: 0.8621
Epoch 82/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3025 - accuracy: 0.8826 -
val_loss: 0.3070 - val_accuracy: 0.8678
Epoch 83/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3371 - accuracy: 0.8714 -
val_loss: 0.3065 - val_accuracy: 0.8678
Epoch 84/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3430 - accuracy: 0.8616 -
val_loss: 0.3049 - val_accuracy: 0.8649
Epoch 85/100
26/26 [==============================] - 0s 15ms/step - loss: 0.2979 - accuracy: 0.8665 -
val_loss: 0.3050 - val_accuracy: 0.8707
Epoch 86/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3017 - accuracy: 0.8690 -
val_loss: 0.3037 - val_accuracy: 0.8736
Epoch 87/100
26/26 [==============================] - 0s 17ms/step - loss: 0.2961 - accuracy: 0.8789 -
val_loss: 0.3029 - val_accuracy: 0.8678
Epoch 88/100
26/26 [==============================] - 0s 16ms/step - loss: 0.2831 - accuracy: 0.8850 -
val_loss: 0.2984 - val_accuracy: 0.8678
Epoch 89/100
26/26 [==============================] - 0s 16ms/step - loss: 0.2765 - accuracy: 0.8838 -
val_loss: 0.2989 - val_accuracy: 0.8678
Epoch 90/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3034 - accuracy: 0.8813 -
val_loss: 0.2984 - val_accuracy: 0.8736
Epoch 91/100
26/26 [==============================] - 0s 16ms/step - loss: 0.2896 - accuracy: 0.8850 -
val_loss: 0.2938 - val_accuracy: 0.8736
Epoch 92/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3045 - accuracy: 0.8714 -
val_loss: 0.2901 - val_accuracy: 0.8736
```

```
Epoch 93/100
26/26 [==============================] - 0s 15ms/step - loss: 0.2792 - accuracy: 0.8900 -
val_loss: 0.2883 - val_accuracy: 0.8851
Epoch 94/100
26/26 [==============================] - 0s 16ms/step - loss: 0.2874 - accuracy: 0.8813 -
val_loss: 0.2865 - val_accuracy: 0.8822
Epoch 95/100
26/26 [==============================] - 0s 16ms/step - loss: 0.2921 - accuracy: 0.8789 -
val_loss: 0.2843 - val_accuracy: 0.8851
Epoch 96/100
26/26 [==============================] - 0s 18ms/step - loss: 0.2825 - accuracy: 0.8826 -
val_loss: 0.2838 - val_accuracy: 0.8822
Epoch 97/100
26/26 [==============================] - 0s 18ms/step - loss: 0.2998 - accuracy: 0.8776 -
val_loss: 0.2788 - val_accuracy: 0.8822
Epoch 98/100
26/26 [==============================] - 0s 17ms/step - loss: 0.2821 - accuracy: 0.8863 -
val_loss: 0.2798 - val_accuracy: 0.8851
Epoch 99/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3207 - accuracy: 0.8665 -
val_loss: 0.2837 - val_accuracy: 0.8851
Epoch 100/100
26/26 [==============================] - 0s 15ms/step - loss: 0.2969 - accuracy: 0.8813 -
val_loss: 0.2805 - val_accuracy: 0.8879
Epoch 1/100
26/26 [==============================] - 2s 23ms/step - loss: 2.0723 - accuracy: 0.0766 -
val_loss: 1.7031 - val_accuracy: 0.0460
Epoch 2/100
26/26 [==============================] - 0s 16ms/step - loss: 1.3412 - accuracy: 0.3634 -
val_loss: 1.1705 - val_accuracy: 0.6667
Epoch 3/100
26/26 [==============================] - 0s 17ms/step - loss: 0.9800 - accuracy: 0.6687 -
val_loss: 0.9448 - val_accuracy: 0.6667
Epoch 4/100
26/26 [==============================] - 0s 17ms/step - loss: 0.8008 - accuracy: 0.7256 -
val_loss: 0.8511 - val_accuracy: 0.6667
Epoch 5/100
26/26 [==============================] - 0s 16ms/step - loss: 0.7316 - accuracy: 0.7367 -
val_loss: 0.7717 - val_accuracy: 0.6667
Epoch 6/100
26/26 [==============================] - 0s 16ms/step - loss: 0.6837 - accuracy: 0.7417 -
val_loss: 0.7012 - val_accuracy: 0.6667
Epoch 7/100
26/26 [==============================] - 0s 17ms/step - loss: 0.6480 - accuracy: 0.7515 -
val_loss: 0.6528 - val_accuracy: 0.6695
Epoch 8/100
26/26 [==============================] - 0s 17ms/step - loss: 0.5917 - accuracy: 0.7738 -
val_loss: 0.5992 - val_accuracy: 0.7184
Epoch 9/100
26/26 [==============================] - 0s 17ms/step - loss: 0.5851 - accuracy: 0.7750 -
val_loss: 0.5624 - val_accuracy: 0.7586
Epoch 10/100
26/26 [==============================] - 0s 18ms/step - loss: 0.5555 - accuracy: 0.7689 -
val_loss: 0.5336 - val_accuracy: 0.7759
Epoch 11/100
26/26 [==============================] - 0s 16ms/step - loss: 0.5444 - accuracy: 0.7874 -
val_loss: 0.5120 - val_accuracy: 0.8132
Epoch 12/100
26/26 [==============================] - 0s 16ms/step - loss: 0.5493 - accuracy: 0.7713 -
val_loss: 0.4991 - val_accuracy: 0.8161
Epoch 13/100
26/26 [==============================] - 0s 16ms/step - loss: 0.5277 - accuracy: 0.7936 -
val_loss: 0.4924 - val_accuracy: 0.8103
Epoch 14/100
26/26 [==============================] - 0s 16ms/step - loss: 0.5186 - accuracy: 0.8035 -
val_loss: 0.4873 - val_accuracy: 0.8161
Epoch 15/100
26/26 [==============================] - 0s 16ms/step - loss: 0.5053 - accuracy: 0.7960 -
val_loss: 0.4803 - val_accuracy: 0.8190
Epoch 16/100
26/26 [==============================] - 0s 16ms/step - loss: 0.5056 - accuracy: 0.8109 -
val_loss: 0.4732 - val_accuracy: 0.8276
```

```
Epoch 17/100
26/26 [==============================] - 0s 17ms/step - loss: 0.5022 - accuracy: 0.8022 -
val_loss: 0.4679 - val_accuracy: 0.8190
Epoch 18/100
26/26 [==============================] - 0s 17ms/step - loss: 0.4517 - accuracy: 0.8331 -
val_loss: 0.4650 - val_accuracy: 0.8247
Epoch 19/100
26/26 [==============================] - 0s 17ms/step - loss: 0.4660 - accuracy: 0.8245 -
val_loss: 0.4634 - val_accuracy: 0.8218
Epoch 20/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4862 - accuracy: 0.8084 -
val_loss: 0.4649 - val_accuracy: 0.8247
Epoch 21/100
26/26 [==============================] - 0s 17ms/step - loss: 0.4889 - accuracy: 0.8010 -
val_loss: 0.4571 - val_accuracy: 0.8247
Epoch 22/100
26/26 [==============================] - 0s 19ms/step - loss: 0.4569 - accuracy: 0.8208 -
val_loss: 0.4545 - val_accuracy: 0.8276
Epoch 23/100
26/26 [==============================] - 0s 18ms/step - loss: 0.4682 - accuracy: 0.8121 -
val_loss: 0.4540 - val_accuracy: 0.8247
Epoch 24/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4394 - accuracy: 0.8245 -
val_loss: 0.4515 - val_accuracy: 0.8305
Epoch 25/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4417 - accuracy: 0.8109 -
val_loss: 0.4519 - val_accuracy: 0.8218
Epoch 26/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4716 - accuracy: 0.8208 -
val_loss: 0.4511 - val_accuracy: 0.8190
Epoch 27/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4431 - accuracy: 0.8331 -
val_loss: 0.4474 - val_accuracy: 0.8247
Epoch 28/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4403 - accuracy: 0.8418 -
val_loss: 0.4431 - val_accuracy: 0.8276
Epoch 29/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4508 - accuracy: 0.8245 -
val_loss: 0.4425 - val_accuracy: 0.8218
Epoch 30/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4237 - accuracy: 0.8393 -
val_loss: 0.4441 - val_accuracy: 0.8247
Epoch 31/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4524 - accuracy: 0.8294 -
val_loss: 0.4423 - val_accuracy: 0.8218
Epoch 32/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4218 - accuracy: 0.8356 -
val_loss: 0.4355 - val_accuracy: 0.8218
Epoch 33/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4416 - accuracy: 0.8257 -
val_loss: 0.4356 - val_accuracy: 0.8190
Epoch 34/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4251 - accuracy: 0.8294 -
val_loss: 0.4373 - val_accuracy: 0.8161
Epoch 35/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4218 - accuracy: 0.8455 -
val_loss: 0.4375 - val_accuracy: 0.8132
Epoch 36/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4340 - accuracy: 0.8319 -
val_loss: 0.4342 - val_accuracy: 0.8218
Epoch 37/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4020 - accuracy: 0.8331 -
val_loss: 0.4252 - val_accuracy: 0.8218
Epoch 38/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4213 - accuracy: 0.8257 -
val_loss: 0.4226 - val_accuracy: 0.8276
Epoch 39/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4219 - accuracy: 0.8356 -
val_loss: 0.4141 - val_accuracy: 0.8276
Epoch 40/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4267 - accuracy: 0.8232 -
val_loss: 0.4125 - val_accuracy: 0.8362
```

```
Epoch 41/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3820 - accuracy: 0.8504 -
val_loss: 0.4119 - val_accuracy: 0.8391
Epoch 42/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3853 - accuracy: 0.8566 -
val_loss: 0.4101 - val_accuracy: 0.8362
Epoch 43/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3861 - accuracy: 0.8344 -
val_loss: 0.4057 - val_accuracy: 0.8362
Epoch 44/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3807 - accuracy: 0.8541 -
val_loss: 0.3999 - val_accuracy: 0.8362
Epoch 45/100
26/26 [==============================] - 0s 18ms/step - loss: 0.3951 - accuracy: 0.8517 -
val_loss: 0.3949 - val_accuracy: 0.8391
Epoch 46/100
26/26 [==============================] - 0s 17ms/step - loss: 0.4095 - accuracy: 0.8319 -
val_loss: 0.3900 - val_accuracy: 0.8420
Epoch 47/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3856 - accuracy: 0.8455 -
val_loss: 0.3893 - val_accuracy: 0.8391
Epoch 48/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3742 - accuracy: 0.8529 -
val_loss: 0.3857 - val_accuracy: 0.8333
Epoch 49/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3780 - accuracy: 0.8517 -
val_loss: 0.3824 - val_accuracy: 0.8391
Epoch 50/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3703 - accuracy: 0.8578 -
val_loss: 0.3813 - val_accuracy: 0.8362
Epoch 51/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3737 - accuracy: 0.8393 -
val_loss: 0.3790 - val_accuracy: 0.8362
Epoch 52/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3696 - accuracy: 0.8492 -
val_loss: 0.3786 - val_accuracy: 0.8362
Epoch 53/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3550 - accuracy: 0.8554 -
val_loss: 0.3784 - val_accuracy: 0.8333
Epoch 54/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3529 - accuracy: 0.8578 -
val_loss: 0.3756 - val_accuracy: 0.8362
Epoch 55/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3686 - accuracy: 0.8492 -
val_loss: 0.3707 - val_accuracy: 0.8391
Epoch 56/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3717 - accuracy: 0.8480 -
val_loss: 0.3700 - val_accuracy: 0.8305
Epoch 57/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3605 - accuracy: 0.8455 -
val_loss: 0.3634 - val_accuracy: 0.8420
Epoch 58/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3414 - accuracy: 0.8640 -
val_loss: 0.3622 - val_accuracy: 0.8362
Epoch 59/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3486 - accuracy: 0.8529 -
val_loss: 0.3624 - val_accuracy: 0.8276
Epoch 60/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3513 - accuracy: 0.8529 -
val_loss: 0.3528 - val_accuracy: 0.8420
Epoch 61/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3432 - accuracy: 0.8628 -
val_loss: 0.3493 - val_accuracy: 0.8420
Epoch 62/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3310 - accuracy: 0.8690 -
val_loss: 0.3496 - val_accuracy: 0.8391
Epoch 63/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3374 - accuracy: 0.8628 -
val_loss: 0.3481 - val_accuracy: 0.8391
Epoch 64/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3702 - accuracy: 0.8504 -
val_loss: 0.3477 - val_accuracy: 0.8506
```

```
Epoch 65/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3448 - accuracy: 0.8591 -
val_loss: 0.3452 - val_accuracy: 0.8448
Epoch 66/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3503 - accuracy: 0.8665 -
val_loss: 0.3386 - val_accuracy: 0.8477
Epoch 67/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3393 - accuracy: 0.8566 -
val_loss: 0.3382 - val_accuracy: 0.8563
Epoch 68/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3556 - accuracy: 0.8492 -
val_loss: 0.3342 - val_accuracy: 0.8448
Epoch 69/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3414 - accuracy: 0.8677 -
val_loss: 0.3301 - val_accuracy: 0.8621
Epoch 70/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3236 - accuracy: 0.8702 -
val_loss: 0.3331 - val_accuracy: 0.8563
Epoch 71/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3227 - accuracy: 0.8653 -
val_loss: 0.3306 - val_accuracy: 0.8592
Epoch 72/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3449 - accuracy: 0.8554 -
val_loss: 0.3279 - val_accuracy: 0.8592
Epoch 73/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3285 - accuracy: 0.8578 -
val_loss: 0.3264 - val_accuracy: 0.8621
Epoch 74/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3350 - accuracy: 0.8677 -
val_loss: 0.3233 - val_accuracy: 0.8621
Epoch 75/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3552 - accuracy: 0.8603 -
val_loss: 0.3210 - val_accuracy: 0.8621
Epoch 76/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3224 - accuracy: 0.8628 -
val_loss: 0.3191 - val_accuracy: 0.8621
Epoch 77/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3416 - accuracy: 0.8578 -
val_loss: 0.3156 - val_accuracy: 0.8621
Epoch 78/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3255 - accuracy: 0.8603 -
val_loss: 0.3130 - val_accuracy: 0.8592
Epoch 79/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3256 - accuracy: 0.8653 -
val_loss: 0.3144 - val_accuracy: 0.8592
Epoch 80/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3121 - accuracy: 0.8702 -
val_loss: 0.3099 - val_accuracy: 0.8592
Epoch 81/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3134 - accuracy: 0.8690 -
val_loss: 0.3099 - val_accuracy: 0.8736
Epoch 82/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3328 - accuracy: 0.8653 -
val_loss: 0.3120 - val_accuracy: 0.8764
Epoch 83/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3070 - accuracy: 0.8801 -
val_loss: 0.3137 - val_accuracy: 0.8678
Epoch 84/100
26/26 [==============================] - 0s 16ms/step - loss: 0.2939 - accuracy: 0.8838 -
val_loss: 0.3141 - val_accuracy: 0.8678
Epoch 85/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3016 - accuracy: 0.8801 -
val_loss: 0.3090 - val_accuracy: 0.8764
Epoch 86/100
26/26 [==============================] - 0s 15ms/step - loss: 0.2831 - accuracy: 0.8900 -
val_loss: 0.3057 - val_accuracy: 0.8649
Epoch 87/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3022 - accuracy: 0.8764 -
val_loss: 0.3007 - val_accuracy: 0.8736
Epoch 88/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3042 - accuracy: 0.8764 -
val_loss: 0.2996 - val_accuracy: 0.8764
```

```
Epoch 89/100
26/26 [==============================] - 0s 16ms/step - loss: 0.2923 - accuracy: 0.8776 -
val_loss: 0.2979 - val_accuracy: 0.8764
Epoch 90/100
26/26 [==============================] - 0s 16ms/step - loss: 0.2971 - accuracy: 0.8801 -
val_loss: 0.2978 - val_accuracy: 0.8793
Epoch 91/100
26/26 [==============================] - 0s 15ms/step - loss: 0.2931 - accuracy: 0.8838 -
val_loss: 0.2960 - val_accuracy: 0.8793
Epoch 92/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3065 - accuracy: 0.8776 -
val_loss: 0.2949 - val_accuracy: 0.8822
Epoch 93/100
26/26 [==============================] - 0s 15ms/step - loss: 0.2998 - accuracy: 0.8764 -
val_loss: 0.2913 - val_accuracy: 0.8822
Epoch 94/100
26/26 [==============================] - 0s 15ms/step - loss: 0.2870 - accuracy: 0.8838 -
val_loss: 0.2950 - val_accuracy: 0.8793
Epoch 95/100
26/26 [==============================] - 0s 16ms/step - loss: 0.2748 - accuracy: 0.8900 -
val_loss: 0.2892 - val_accuracy: 0.8822
Epoch 96/100
26/26 [==============================] - 0s 15ms/step - loss: 0.2790 - accuracy: 0.8838 -
val_loss: 0.2875 - val_accuracy: 0.8793
Epoch 97/100
26/26 [==============================] - 0s 16ms/step - loss: 0.2868 - accuracy: 0.8702 -
val_loss: 0.2855 - val_accuracy: 0.8851
Epoch 98/100
26/26 [==============================] - 0s 16ms/step - loss: 0.2763 - accuracy: 0.8888 -
val_loss: 0.2840 - val_accuracy: 0.8793
Epoch 99/100
26/26 [==============================] - 0s 15ms/step - loss: 0.2930 - accuracy: 0.8789 -
val_loss: 0.2829 - val_accuracy: 0.8793
Epoch 100/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3015 - accuracy: 0.8813 -
val_loss: 0.2796 - val_accuracy: 0.8822
```

In [219]:

```
y_pred = ensemble3.predict(X_test)
```

```
18/18 [==============================] - 0s 2ms/step
18/18 [==============================] - 0s 3ms/step
18/18 [==============================] - 0s 3ms/step
```

Test Accuracy

In [220]:

```
x=accuracy_score(y_test,y_pred)
```

In [221]:

```
print("Test Accuracy is :", x*100 ," %")
```

```
Test Accuracy is : 91.05263157894737  %
```

Loading the weights into the model from ecar.h5

In [223]:

```
emodel1car=car_model()
emodel1car.load_weights('ecar.h5')
```

Confusion Matrix

In [228]:

```
ecm=confusion_matrix(y_test, y_pred)
```

In [229]:

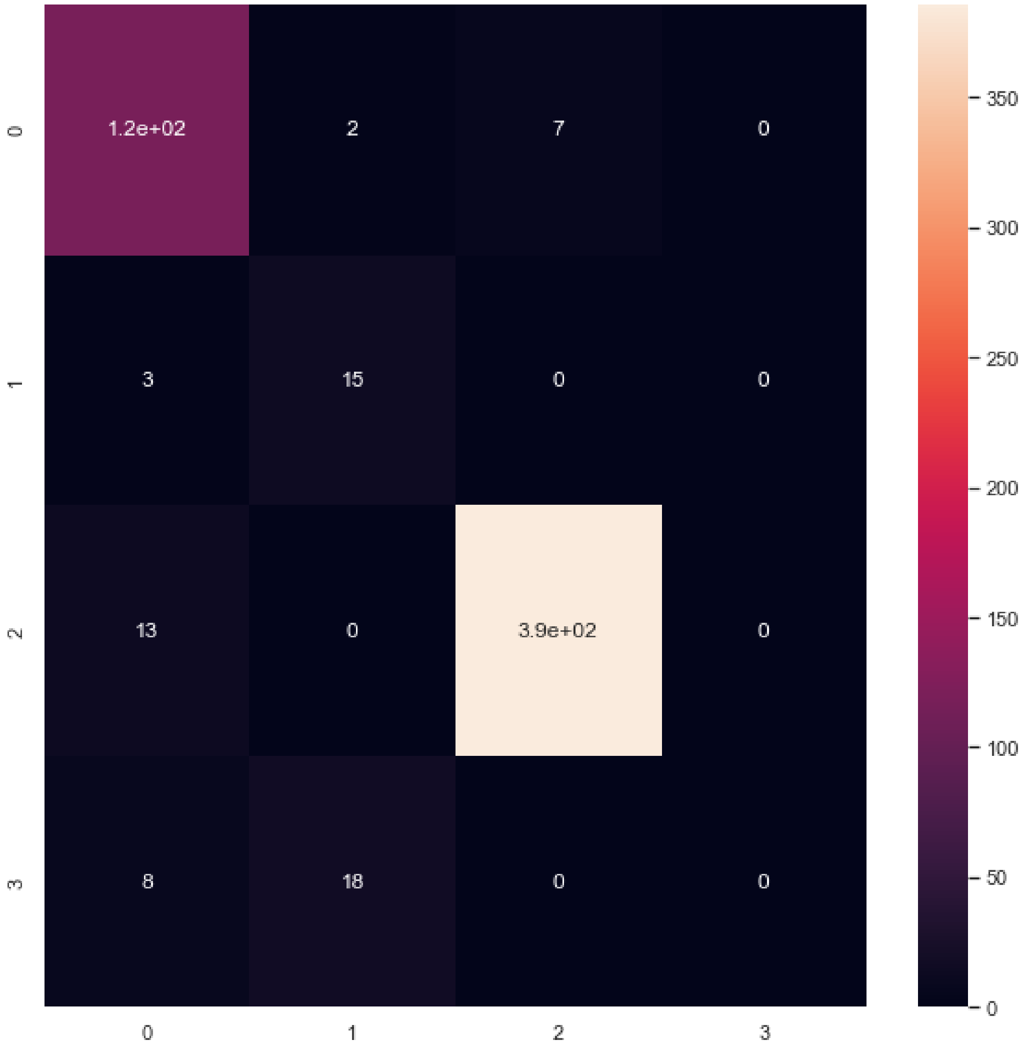
```
ecm
```

Out[229]:

```
array([[118,   2,   7,   0],
       [  3,  15,   0,   0],
       [ 13,   0, 386,   0],
       [  8,  18,   0,   0]], dtype=int64)
```

In [230]:

```
matrix(ecm)
```



Inference:

From the above experiments we infer that ensemble model gives slight or same accuracy compared to baseline MLP model