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

Question 1

1) MLP Model

In [9]:

```
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 [10]:

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

In [11]:

```
X_train.shape
```

Out[11]:

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

In [12]:

```
X_test.shape
```

Out[12]:

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

In [13]:

```
Y_train.shape
```

Out[13]:

```
(60000,)
```

In [14]:

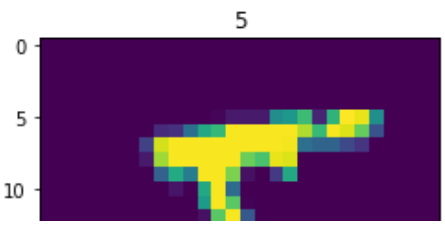
```
Y_test.shape
```

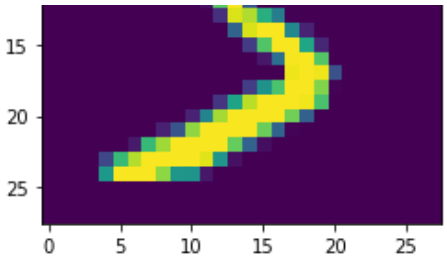
Out[14]:

```
(10000,)
```

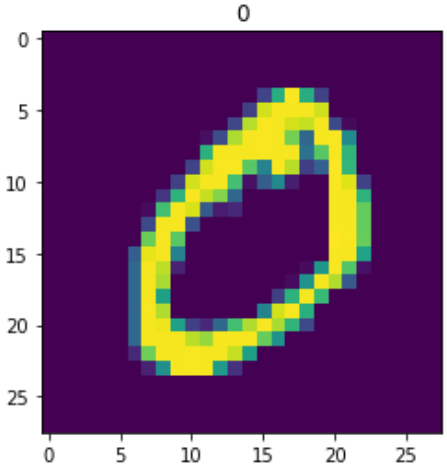
In [15]:

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

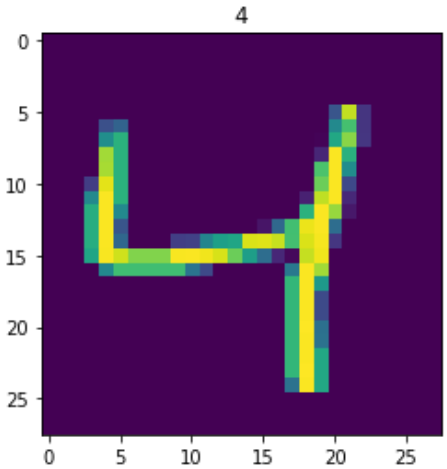




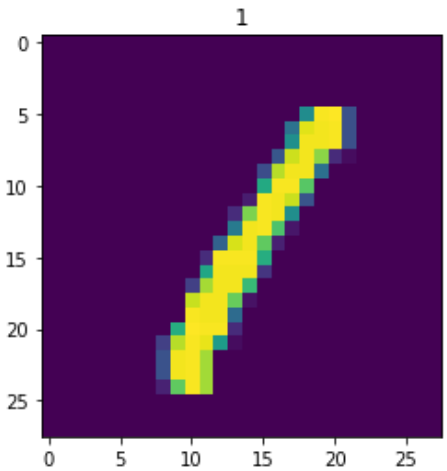
0



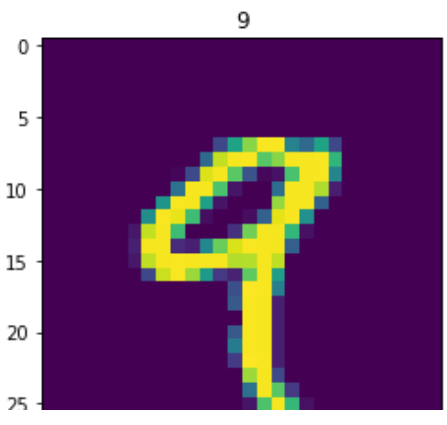
4

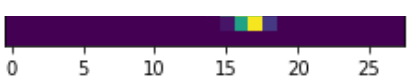


1



9





Printing the shape

In [16]:

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

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

Reshaping the data

In [17]:

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

In [18]:

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

One hot encoding

In [19]:

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

In [20]:

```
Y_train
```

Out[20]:

```
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 [21]:

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

In [22]:

```
Y_test
```

Out[22]:

```
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 [23]:

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

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

In [24]:

```
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.layers import GlobalAveragePooling1D
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 [25]:

```
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 [26]:

```
model=in_model()
```

In [27]:

```
model.summary()
```

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense (Dense)                (None, 50)                39250
_________________________________________________________________
batch_normalization (BatchNo (None, 50)                200
_________________________________________________________________
activation (Activation)      (None, 50)                0
_________________________________________________________________
dropout (Dropout)            (None, 50)                0
_________________________________________________________________
dense_1 (Dense)              (None, 50)                2550
```

```
batch_normalization_1 (Batch (None, 50)                200
_________________________________________________________________
activation_1 (Activation)    (None, 50)                0
_________________________________________________________________
dropout_1 (Dropout)          (None, 50)                0
_________________________________________________________________
dense_2 (Dense)              (None, 50)                2550
_________________________________________________________________
batch_normalization_2 (Batch (None, 50)                200
_________________________________________________________________
activation_2 (Activation)    (None, 50)                0
_________________________________________________________________
dropout_2 (Dropout)          (None, 50)                0
_________________________________________________________________
dense_3 (Dense)              (None, 50)                2550
_________________________________________________________________
batch_normalization_3 (Batch (None, 50)                200
_________________________________________________________________
activation_3 (Activation)    (None, 50)                0
_________________________________________________________________
dropout_3 (Dropout)          (None, 50)                0
_________________________________________________________________
dense_4 (Dense)              (None, 10)                510
_________________________________________________________________
activation_4 (Activation)    (None, 10)                0
=================================================================
Total params: 48,210
Trainable params: 47,810
Non-trainable params: 400
_________________________________________________________________
```

In [28]:

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

Custom callback function

This function stops the training once accuracy reaches 90%

In [29]:

```
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 [30]:

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

Training the model

In [31]:

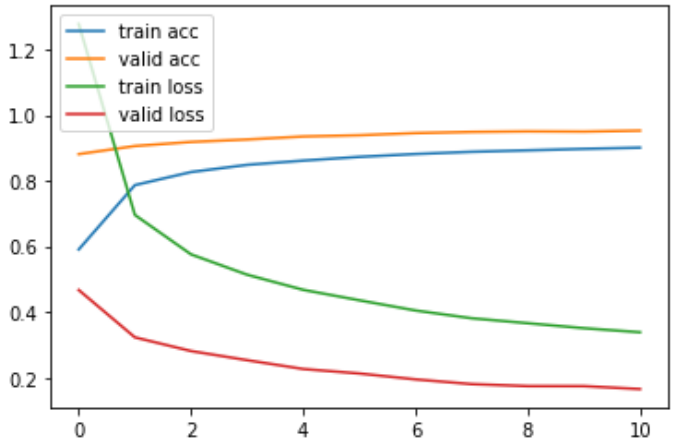
```
train1=model.fit(X_train,Y_train,validation_split=0.3,epochs=100,callbacks=[callback],ver
bose=1)
```

```
Epoch 1/100
```

```
1313/1313 [==============================] - 24s 16ms/step - loss: 1.2775 - accuracy: 0.5
902 - val_loss: 0.4666 - val_accuracy: 0.8808
Epoch 2/100
1313/1313 [==============================] - 21s 16ms/step - loss: 0.6955 - accuracy: 0.7
864 - val_loss: 0.3224 - val_accuracy: 0.9056
Epoch 3/100
1313/1313 [==============================] - 20s 16ms/step - loss: 0.5754 - accuracy: 0.8
262 - val_loss: 0.2807 - val_accuracy: 0.9179
Epoch 4/100
1313/1313 [==============================] - 20s 15ms/step - loss: 0.5133 - accuracy: 0.8
481 - val_loss: 0.2528 - val_accuracy: 0.9254
Epoch 5/100
1313/1313 [==============================] - 20s 15ms/step - loss: 0.4672 - accuracy: 0.8
611 - val_loss: 0.2259 - val_accuracy: 0.9346
Epoch 6/100
1313/1313 [==============================] - 20s 15ms/step - loss: 0.4353 - accuracy: 0.8
725 - val_loss: 0.2122 - val_accuracy: 0.9384
Epoch 7/100
1313/1313 [==============================] - 20s 15ms/step - loss: 0.4042 - accuracy: 0.8
812 - val_loss: 0.1942 - val_accuracy: 0.9450
Epoch 8/100
1313/1313 [==============================] - 20s 15ms/step - loss: 0.3806 - accuracy: 0.8
879 - val_loss: 0.1798 - val_accuracy: 0.9484
Epoch 9/100
1313/1313 [==============================] - 20s 15ms/step - loss: 0.3654 - accuracy: 0.8
923 - val_loss: 0.1738 - val_accuracy: 0.9501
Epoch 10/100
1313/1313 [==============================] - 20s 15ms/step - loss: 0.3503 - accuracy: 0.8
969 - val_loss: 0.1737 - val_accuracy: 0.9494
Epoch 11/100
1313/1313 [==============================] - 20s 15ms/step - loss: 0.3379 - accuracy: 0.9
006 - val_loss: 0.1648 - val_accuracy: 0.9524
Epoch 10: Reached baseline, terminating training
```

In [32]:

```
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 [33]:

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

```
313/313 [==============================] - 2s 5ms/step - loss: 0.1565 - accuracy: 0.9538
Test Accuracy is: 95.38000226020813%
```

In [34]:

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

```
['loss', 'accuracy']
[0.15651117265224457, 0.9538000226020813]
```

Printing Confusion Matrix

In [35]:

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

In [36]:

```
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 [37]:

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

In [38]:

```
import seaborn as sns
```

In [39]:

```
cm1
```

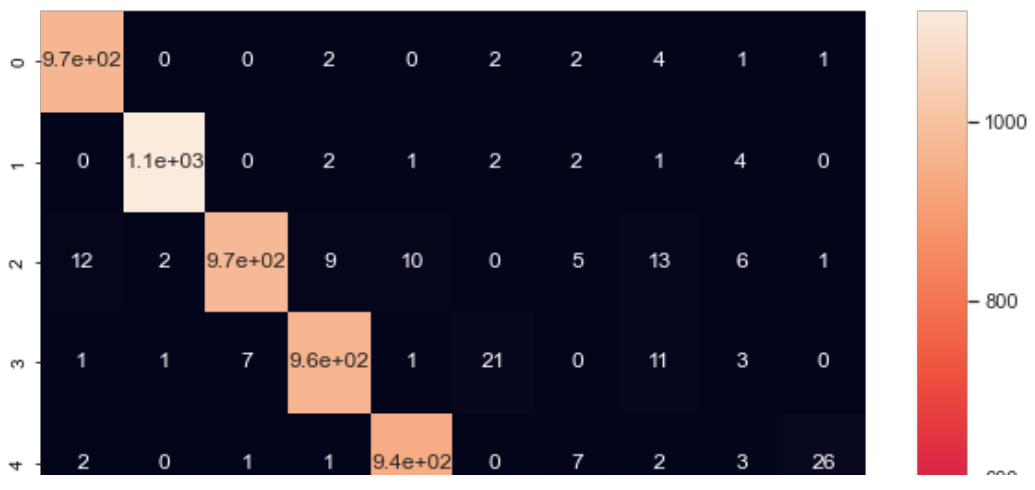
Out[39]:

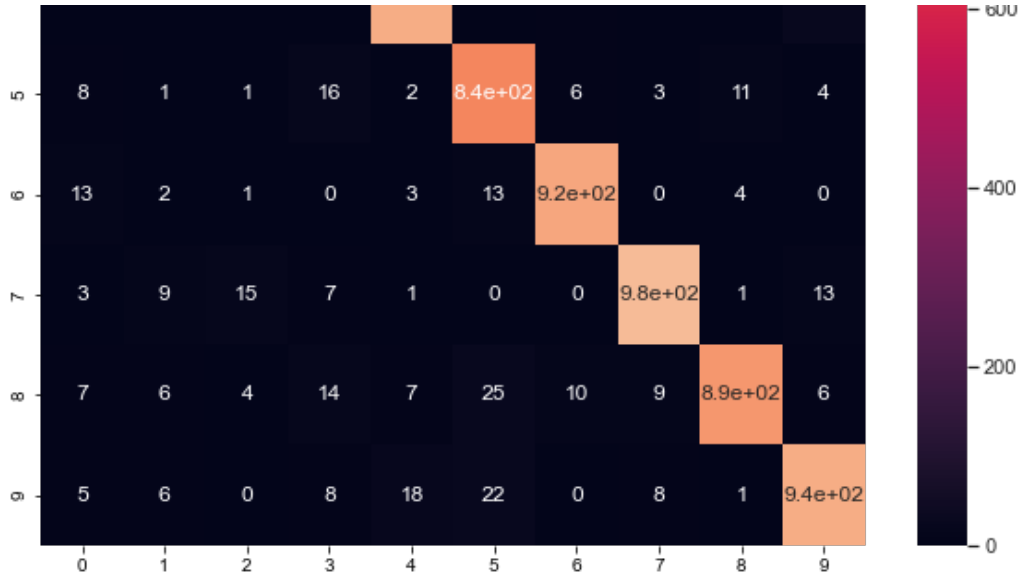
```
array([[ 968,    0,    0,    2,    0,    2,    2,    4,    1,    1],
       [   0, 1123,    0,    2,    1,    2,    2,    1,    4,    0],
       [  12,    2,  974,    9,   10,    0,    5,   13,    6,    1],
       [   1,    1,    7,  965,    1,   21,    0,   11,    3,    0],
       [   2,    0,    1,    1,  940,    0,    7,    2,    3,   26],
       [   8,    1,    1,   16,    2,  840,    6,    3,   11,    4],
       [  13,    2,    1,    0,    3,   13,  922,    0,    4,    0],
       [   3,    9,   15,    7,    1,    0,    0,  979,    1,   13],
       [   7,    6,    4,   14,    7,   25,   10,    9,  886,    6],
       [   5,    6,    0,    8,   18,   22,    0,    8,    1,  941]],
      dtype=int64)
```

In [40]:

```
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 [41]:

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

In [42]:

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

In [43]:

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

Initializing 3 models for Ensemble Learning

In [44]:

```
emodel1=KerasClassifier(build_fn=in_model,epochs=100,validation_split=0.3,callbacks=[call
back1],verbose=1)
emodel2=KerasClassifier(build_fn=in_model,epochs=100,validation_split=0.3,callbacks=[call
back1],verbose=1)
emodel3=KerasClassifier(build_fn=in_model,epochs=100,validation_split=0.3,callbacks=[call
back1],verbose=1)
emodel1._estimator_type="classifier"
emodel2._estimator_type="classifier"
emodel3._estimator_type="classifier"
```

In [45]:

```
ensemble=VotingClassifier(estimators=[('model1',emodel1),('model2',emodel2),('model3',emo
del3)],voting='soft')
```

In [46]:

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

```
Epoch 1/100
1313/1313 [==============================] - 21s 15ms/step - loss: 1.3198 - accuracy: 0.5
711 - val_loss: 0.4990 - val_accuracy: 0.8661
Epoch 2/100
1313/1313 [==============================] - 19s 14ms/step - loss: 0.7342 - accuracy: 0.7
762 - val_loss: 0.3414 - val_accuracy: 0.8962
Epoch 3/100
1313/1313 [==============================] - 20s 15ms/step - loss: 0.5948 - accuracy: 0.8
236 - val_loss: 0.2900 - val_accuracy: 0.9132
```

```
Epoch 4/100
1313/1313 [==============================] - 20s 15ms/step - loss: 0.5165 - accuracy: 0.8
467 - val_loss: 0.2486 - val_accuracy: 0.9273
Epoch 5/100
1313/1313 [==============================] - 20s 15ms/step - loss: 0.4755 - accuracy: 0.8
594 - val_loss: 0.2196 - val_accuracy: 0.9363
Epoch 6/100
1313/1313 [==============================] - 20s 15ms/step - loss: 0.4368 - accuracy: 0.8
717 - val_loss: 0.2141 - val_accuracy: 0.9373
Epoch 7/100
1313/1313 [==============================] - 20s 15ms/step - loss: 0.4117 - accuracy: 0.8
780 - val_loss: 0.1877 - val_accuracy: 0.9458
Epoch 8/100
1313/1313 [==============================] - 20s 15ms/step - loss: 0.3835 - accuracy: 0.8
863 - val_loss: 0.1771 - val_accuracy: 0.9491
Epoch 9/100
1313/1313 [==============================] - 20s 15ms/step - loss: 0.3671 - accuracy: 0.8
901 - val_loss: 0.1679 - val_accuracy: 0.9508
Epoch 10/100
1313/1313 [==============================] - 20s 15ms/step - loss: 0.3443 - accuracy: 0.8
977 - val_loss: 0.1729 - val_accuracy: 0.9501
Epoch 11/100
1313/1313 [==============================] - 20s 15ms/step - loss: 0.3344 - accuracy: 0.9
007 - val_loss: 0.1582 - val_accuracy: 0.9542
Epoch 10: Reached baseline, terminating training
Epoch 1/100
1313/1313 [==============================] - 20s 15ms/step - loss: 1.3162 - accuracy: 0.5
829 - val_loss: 0.4376 - val_accuracy: 0.8896
Epoch 2/100
1313/1313 [==============================] - 19s 15ms/step - loss: 0.7079 - accuracy: 0.7
870 - val_loss: 0.3212 - val_accuracy: 0.9095
Epoch 3/100
1313/1313 [==============================] - 19s 15ms/step - loss: 0.5721 - accuracy: 0.8
307 - val_loss: 0.2627 - val_accuracy: 0.9241
Epoch 4/100
1313/1313 [==============================] - 19s 15ms/step - loss: 0.5082 - accuracy: 0.8
509 - val_loss: 0.2509 - val_accuracy: 0.9261
Epoch 5/100
1313/1313 [==============================] - 19s 15ms/step - loss: 0.4633 - accuracy: 0.8
637 - val_loss: 0.2302 - val_accuracy: 0.9342
Epoch 6/100
1313/1313 [==============================] - 19s 15ms/step - loss: 0.4334 - accuracy: 0.8
716 - val_loss: 0.2126 - val_accuracy: 0.9377
Epoch 7/100
1313/1313 [==============================] - 19s 15ms/step - loss: 0.3933 - accuracy: 0.8
825 - val_loss: 0.1842 - val_accuracy: 0.9463
Epoch 8/100
1313/1313 [==============================] - 19s 15ms/step - loss: 0.3777 - accuracy: 0.8
888 - val_loss: 0.1782 - val_accuracy: 0.9480
Epoch 9/100
1313/1313 [==============================] - 20s 15ms/step - loss: 0.3597 - accuracy: 0.8
935 - val_loss: 0.1741 - val_accuracy: 0.9492
Epoch 10/100
1313/1313 [==============================] - 19s 15ms/step - loss: 0.3428 - accuracy: 0.8
987 - val_loss: 0.1622 - val_accuracy: 0.9536
Epoch 11/100
1313/1313 [==============================] - 19s 14ms/step - loss: 0.3371 - accuracy: 0.9
004 - val_loss: 0.1524 - val_accuracy: 0.9560
Epoch 10: Reached baseline, terminating training
Epoch 1/100
1313/1313 [==============================] - 20s 15ms/step - loss: 1.3121 - accuracy: 0.5
799 - val_loss: 0.4615 - val_accuracy: 0.8786
Epoch 2/100
1313/1313 [==============================] - 19s 15ms/step - loss: 0.7033 - accuracy: 0.7
875 - val_loss: 0.3199 - val_accuracy: 0.9064
Epoch 3/100
1313/1313 [==============================] - 19s 15ms/step - loss: 0.5721 - accuracy: 0.8
287 - val_loss: 0.2817 - val_accuracy: 0.9161
Epoch 4/100
1313/1313 [==============================] - 19s 15ms/step - loss: 0.5040 - accuracy: 0.8
501 - val_loss: 0.2536 - val_accuracy: 0.9239
Epoch 5/100
```

```
1313/1313 [==============================] - 19s 15ms/step - loss: 0.4680 - accuracy: 0.8
597 - val_loss: 0.2202 - val_accuracy: 0.9347
Epoch 6/100
1313/1313 [==============================] - 19s 15ms/step - loss: 0.4360 - accuracy: 0.8
724 - val_loss: 0.2094 - val_accuracy: 0.9387
Epoch 7/100
1313/1313 [==============================] - 20s 15ms/step - loss: 0.4042 - accuracy: 0.8
797 - val_loss: 0.1934 - val_accuracy: 0.9416
Epoch 8/100
1313/1313 [==============================] - 19s 15ms/step - loss: 0.3856 - accuracy: 0.8
852 - val_loss: 0.1823 - val_accuracy: 0.9472
Epoch 9/100
1313/1313 [==============================] - 19s 15ms/step - loss: 0.3683 - accuracy: 0.8
904 - val_loss: 0.1751 - val_accuracy: 0.9489
Epoch 10/100
1313/1313 [==============================] - 19s 15ms/step - loss: 0.3492 - accuracy: 0.8
957 - val_loss: 0.1679 - val_accuracy: 0.9513
Epoch 11/100
1313/1313 [==============================] - 19s 14ms/step - loss: 0.3315 - accuracy: 0.9
011 - val_loss: 0.1569 - val_accuracy: 0.9541
Epoch 10: Reached baseline, terminating training
```

In [47]:

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

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

In [48]:

```
Y_pred1.shape
```

Out[48]:

```
(10000,)
```

Printing the accuracy

In [49]:

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

In [50]:

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

```
Test Accuracy is : 96.1  %
```

Printing the confusion matrix

In [51]:

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

In [52]:

```
cm2
```

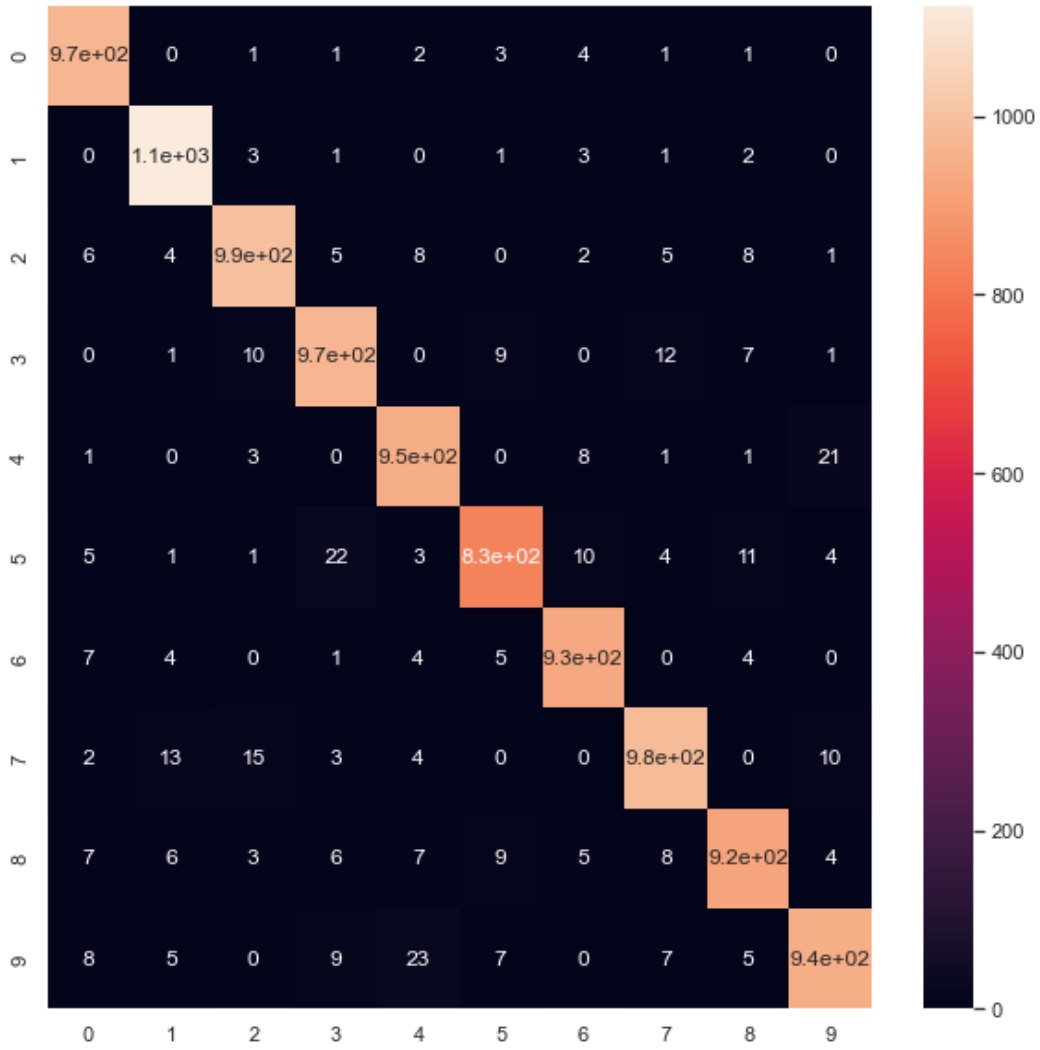
Out[52]:

```
array([[ 967,    0,    1,    1,    2,    3,    4,    1,    1,    0],
       [   0, 1124,    3,    1,    0,    1,    3,    1,    2,    0],
       [   6,    4,  993,    5,    8,    0,    2,    5,    8,    1],
       [   0,    1,   10,  970,    0,    9,    0,   12,    7,    1],
       [   1,    0,    3,    0,  947,    0,    8,    1,    1,   21],
       [   5,    1,    1,   22,    3,  831,   10,    4,   11,    4],
       [   7,    4,    0,    1,    4,    5,  933,    0,    4,    0],
```

```
       [   2,   13,   15,    3,    4,    0,    0,  981,    0,   10],
       [   7,    6,    3,    6,    7,    9,    5,    8,  919,    4],
       [   8,    5,    0,    9,   23,    7,    0,    7,    5,  945]],
      dtype=int64)
```

In [53]:

```
matrix(cm2)
```



Question 2

1) MLP Model

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

In [54]:

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

In [55]:

```
df.head()
```

Out[55]:

	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	small	low	unacc
4	vhigh	vhigh	2	2	med	high	unacc

Splitting the data

In [56]:

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

Out[56]:

	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 [57]:

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

Out[57]:

```
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 [58]:

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

In [59]:

```
X=X.values
```

In [60]:

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

In [61]:

```
Y=Y.values
```

In [62]:

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

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

Train-test-split

In [63]:

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

In [64]:

```
X_train, X_test, y_train, y_test = train_test_split(X, Y, test_size=0.33, random_state=4
2)
```

In [65]:

```
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 [66]:

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

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

Initializing the NN Model

In [67]:

```
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 = ['accurac
y'])
```

```
    return model
```

In [68]:

```
cmodel1=car_model()
```

In [69]:

```
cmodel1.summary()
```

```
Model: "sequential_4"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense_20 (Dense)             (None, 50)                1100
_________________________________________________________________
batch_normalization_16 (Batc (None, 50)                200
_________________________________________________________________
activation_20 (Activation)   (None, 50)                0
_________________________________________________________________
dropout_16 (Dropout)         (None, 50)                0
_________________________________________________________________
dense_21 (Dense)             (None, 50)                2550
_________________________________________________________________
batch_normalization_17 (Batc (None, 50)                200
_________________________________________________________________
activation_21 (Activation)   (None, 50)                0
_________________________________________________________________
dropout_17 (Dropout)         (None, 50)                0
_________________________________________________________________
dense_22 (Dense)             (None, 50)                2550
_________________________________________________________________
batch_normalization_18 (Batc (None, 50)                200
_________________________________________________________________
activation_22 (Activation)   (None, 50)                0
_________________________________________________________________
dropout_18 (Dropout)         (None, 50)                0
_________________________________________________________________
dense_23 (Dense)             (None, 50)                2550
_________________________________________________________________
batch_normalization_19 (Batc (None, 50)                200
_________________________________________________________________
activation_23 (Activation)   (None, 50)                0
_________________________________________________________________
dropout_19 (Dropout)         (None, 50)                0
_________________________________________________________________
dense_24 (Dense)             (None, 4)                 204
_________________________________________________________________
activation_24 (Activation)   (None, 4)                 0
=================================================================
Total params: 9,754
Trainable params: 9,354
Non-trainable params: 400
_________________________________________________________________
```

Call back function

In [70]:

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

Training the model

In [71]:

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

```
Epoch 1/100
```

```
26/26 [==============================] - 2s 30ms/step - loss: 1.2413 - accuracy: 0.4400 -
val_loss: 1.0090 - val_accuracy: 0.6667
Epoch 2/100
26/26 [==============================] - 0s 17ms/step - loss: 0.9385 - accuracy: 0.6700 -
val_loss: 0.9272 - val_accuracy: 0.6667
Epoch 3/100
26/26 [==============================] - 0s 17ms/step - loss: 0.7890 - accuracy: 0.7046 -
val_loss: 0.9071 - val_accuracy: 0.6667
Epoch 4/100
26/26 [==============================] - 0s 17ms/step - loss: 0.7585 - accuracy: 0.7194 -
val_loss: 0.8838 - val_accuracy: 0.6667
Epoch 5/100
26/26 [==============================] - 0s 16ms/step - loss: 0.7269 - accuracy: 0.7182 -
val_loss: 0.8213 - val_accuracy: 0.6667
Epoch 6/100
26/26 [==============================] - 0s 16ms/step - loss: 0.6733 - accuracy: 0.7355 -
val_loss: 0.7654 - val_accuracy: 0.6667
Epoch 7/100
26/26 [==============================] - 0s 16ms/step - loss: 0.6294 - accuracy: 0.7392 -
val_loss: 0.6921 - val_accuracy: 0.6667
Epoch 8/100
26/26 [==============================] - 0s 17ms/step - loss: 0.6127 - accuracy: 0.7590 -
val_loss: 0.6403 - val_accuracy: 0.6667
Epoch 9/100
26/26 [==============================] - 0s 15ms/step - loss: 0.5826 - accuracy: 0.7812 -
val_loss: 0.5915 - val_accuracy: 0.6983
Epoch 10/100
26/26 [==============================] - 0s 15ms/step - loss: 0.5469 - accuracy: 0.7923 -
val_loss: 0.5484 - val_accuracy: 0.7557
Epoch 11/100
26/26 [==============================] - 0s 16ms/step - loss: 0.5319 - accuracy: 0.7849 -
val_loss: 0.5216 - val_accuracy: 0.7672
Epoch 12/100
26/26 [==============================] - 0s 15ms/step - loss: 0.5302 - accuracy: 0.8010 -
val_loss: 0.5070 - val_accuracy: 0.7874
Epoch 13/100
26/26 [==============================] - 0s 16ms/step - loss: 0.5131 - accuracy: 0.7936 -
val_loss: 0.5029 - val_accuracy: 0.7874
Epoch 14/100
26/26 [==============================] - 0s 15ms/step - loss: 0.5371 - accuracy: 0.7923 -
val_loss: 0.5023 - val_accuracy: 0.7787
Epoch 15/100
26/26 [==============================] - 0s 16ms/step - loss: 0.5139 - accuracy: 0.7960 -
val_loss: 0.4926 - val_accuracy: 0.7874
Epoch 16/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4720 - accuracy: 0.8146 -
val_loss: 0.4842 - val_accuracy: 0.7960
Epoch 17/100
26/26 [==============================] - 0s 16ms/step - loss: 0.5045 - accuracy: 0.8010 -
val_loss: 0.4717 - val_accuracy: 0.8075
Epoch 18/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4839 - accuracy: 0.8096 -
val_loss: 0.4669 - val_accuracy: 0.8132
Epoch 19/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4616 - accuracy: 0.8133 -
val_loss: 0.4626 - val_accuracy: 0.8161
Epoch 20/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4574 - accuracy: 0.8257 -
val_loss: 0.4587 - val_accuracy: 0.8190
Epoch 21/100
26/26 [==============================] - 0s 16ms/step - loss: 0.5008 - accuracy: 0.8183 -
val_loss: 0.4537 - val_accuracy: 0.8190
Epoch 22/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4710 - accuracy: 0.8171 -
val_loss: 0.4543 - val_accuracy: 0.8218
Epoch 23/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4479 - accuracy: 0.8269 -
val_loss: 0.4551 - val_accuracy: 0.8075
Epoch 24/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4615 - accuracy: 0.8084 -
val_loss: 0.4558 - val_accuracy: 0.8161
Epoch 25/100
```

```
26/26 [==============================] - 0s 16ms/step - loss: 0.4546 - accuracy: 0.8158 -
val_loss: 0.4510 - val_accuracy: 0.8190
Epoch 26/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4585 - accuracy: 0.8245 -
val_loss: 0.4523 - val_accuracy: 0.8103
Epoch 27/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4589 - accuracy: 0.8146 -
val_loss: 0.4465 - val_accuracy: 0.8190
Epoch 28/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4549 - accuracy: 0.8158 -
val_loss: 0.4387 - val_accuracy: 0.8247
Epoch 29/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4314 - accuracy: 0.8307 -
val_loss: 0.4369 - val_accuracy: 0.8218
Epoch 30/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4480 - accuracy: 0.8245 -
val_loss: 0.4346 - val_accuracy: 0.8218
Epoch 31/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4280 - accuracy: 0.8220 -
val_loss: 0.4305 - val_accuracy: 0.8305
Epoch 32/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4211 - accuracy: 0.8443 -
val_loss: 0.4316 - val_accuracy: 0.8247
Epoch 33/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4211 - accuracy: 0.8344 -
val_loss: 0.4296 - val_accuracy: 0.8218
Epoch 34/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4141 - accuracy: 0.8418 -
val_loss: 0.4251 - val_accuracy: 0.8333
Epoch 35/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4374 - accuracy: 0.8220 -
val_loss: 0.4212 - val_accuracy: 0.8362
Epoch 36/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4253 - accuracy: 0.8294 -
val_loss: 0.4211 - val_accuracy: 0.8305
Epoch 37/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4282 - accuracy: 0.8356 -
val_loss: 0.4198 - val_accuracy: 0.8305
Epoch 38/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4358 - accuracy: 0.8158 -
val_loss: 0.4168 - val_accuracy: 0.8276
Epoch 39/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4168 - accuracy: 0.8269 -
val_loss: 0.4165 - val_accuracy: 0.8305
Epoch 40/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4054 - accuracy: 0.8381 -
val_loss: 0.4098 - val_accuracy: 0.8362
Epoch 41/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3880 - accuracy: 0.8480 -
val_loss: 0.4080 - val_accuracy: 0.8333
Epoch 42/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4014 - accuracy: 0.8492 -
val_loss: 0.4051 - val_accuracy: 0.8333
Epoch 43/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3991 - accuracy: 0.8356 -
val_loss: 0.4002 - val_accuracy: 0.8391
Epoch 44/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3991 - accuracy: 0.8405 -
val_loss: 0.3985 - val_accuracy: 0.8362
Epoch 45/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3790 - accuracy: 0.8554 -
val_loss: 0.3949 - val_accuracy: 0.8362
Epoch 46/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3718 - accuracy: 0.8455 -
val_loss: 0.3913 - val_accuracy: 0.8391
Epoch 47/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3920 - accuracy: 0.8430 -
val_loss: 0.3888 - val_accuracy: 0.8391
Epoch 48/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3768 - accuracy: 0.8381 -
val_loss: 0.3889 - val_accuracy: 0.8276
Epoch 49/100
```

```
26/26 [==============================] - 0s 15ms/step - loss: 0.3878 - accuracy: 0.8307 -
val_loss: 0.3929 - val_accuracy: 0.8218
Epoch 50/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3752 - accuracy: 0.8541 -
val_loss: 0.3845 - val_accuracy: 0.8362
Epoch 51/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3924 - accuracy: 0.8430 -
val_loss: 0.3773 - val_accuracy: 0.8305
Epoch 52/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3772 - accuracy: 0.8467 -
val_loss: 0.3761 - val_accuracy: 0.8333
Epoch 53/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3590 - accuracy: 0.8492 -
val_loss: 0.3747 - val_accuracy: 0.8362
Epoch 54/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4071 - accuracy: 0.8183 -
val_loss: 0.3716 - val_accuracy: 0.8391
Epoch 55/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4057 - accuracy: 0.8455 -
val_loss: 0.3661 - val_accuracy: 0.8362
Epoch 56/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3706 - accuracy: 0.8455 -
val_loss: 0.3646 - val_accuracy: 0.8391
Epoch 57/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3348 - accuracy: 0.8616 -
val_loss: 0.3650 - val_accuracy: 0.8333
Epoch 58/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3727 - accuracy: 0.8331 -
val_loss: 0.3584 - val_accuracy: 0.8420
Epoch 59/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3471 - accuracy: 0.8529 -
val_loss: 0.3581 - val_accuracy: 0.8420
Epoch 60/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3485 - accuracy: 0.8591 -
val_loss: 0.3582 - val_accuracy: 0.8333
Epoch 61/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3449 - accuracy: 0.8616 -
val_loss: 0.3531 - val_accuracy: 0.8391
Epoch 62/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3457 - accuracy: 0.8640 -
val_loss: 0.3501 - val_accuracy: 0.8391
Epoch 63/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3486 - accuracy: 0.8541 -
val_loss: 0.3518 - val_accuracy: 0.8391
Epoch 64/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3664 - accuracy: 0.8480 -
val_loss: 0.3454 - val_accuracy: 0.8391
Epoch 65/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3416 - accuracy: 0.8554 -
val_loss: 0.3445 - val_accuracy: 0.8448
Epoch 66/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3427 - accuracy: 0.8517 -
val_loss: 0.3442 - val_accuracy: 0.8448
Epoch 67/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3431 - accuracy: 0.8578 -
val_loss: 0.3437 - val_accuracy: 0.8420
Epoch 68/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3444 - accuracy: 0.8653 -
val_loss: 0.3436 - val_accuracy: 0.8362
Epoch 69/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3438 - accuracy: 0.8480 -
val_loss: 0.3399 - val_accuracy: 0.8420
Epoch 70/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3304 - accuracy: 0.8702 -
val_loss: 0.3401 - val_accuracy: 0.8420
Epoch 71/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3520 - accuracy: 0.8566 -
val_loss: 0.3412 - val_accuracy: 0.8534
Epoch 72/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3042 - accuracy: 0.8739 -
val_loss: 0.3397 - val_accuracy: 0.8506
Epoch 73/100
```

```
26/26 [==============================] - 0s 16ms/step - loss: 0.3230 - accuracy: 0.8591 -
val_loss: 0.3354 - val_accuracy: 0.8506
Epoch 74/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3403 - accuracy: 0.8541 -
val_loss: 0.3302 - val_accuracy: 0.8592
Epoch 75/100
26/26 [==============================] - 0s 19ms/step - loss: 0.3364 - accuracy: 0.8554 -
val_loss: 0.3279 - val_accuracy: 0.8592
Epoch 76/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3334 - accuracy: 0.8628 -
val_loss: 0.3281 - val_accuracy: 0.8736
Epoch 77/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3092 - accuracy: 0.8739 -
val_loss: 0.3265 - val_accuracy: 0.8563
Epoch 78/100
26/26 [==============================] - 0s 18ms/step - loss: 0.3142 - accuracy: 0.8739 -
val_loss: 0.3226 - val_accuracy: 0.8649
Epoch 79/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3263 - accuracy: 0.8677 -
val_loss: 0.3185 - val_accuracy: 0.8592
Epoch 80/100
26/26 [==============================] - 0s 18ms/step - loss: 0.3070 - accuracy: 0.8813 -
val_loss: 0.3189 - val_accuracy: 0.8678
Epoch 81/100
26/26 [==============================] - 0s 18ms/step - loss: 0.2942 - accuracy: 0.8826 -
val_loss: 0.3179 - val_accuracy: 0.8621
Epoch 82/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3175 - accuracy: 0.8665 -
val_loss: 0.3144 - val_accuracy: 0.8678
Epoch 83/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3268 - accuracy: 0.8541 -
val_loss: 0.3179 - val_accuracy: 0.8678
Epoch 84/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3039 - accuracy: 0.8801 -
val_loss: 0.3188 - val_accuracy: 0.8621
Epoch 85/100
26/26 [==============================] - 0s 15ms/step - loss: 0.2877 - accuracy: 0.8813 -
val_loss: 0.3151 - val_accuracy: 0.8678
Epoch 86/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3397 - accuracy: 0.8554 -
val_loss: 0.3126 - val_accuracy: 0.8764
Epoch 87/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3349 - accuracy: 0.8665 -
val_loss: 0.3109 - val_accuracy: 0.8707
Epoch 88/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3012 - accuracy: 0.8727 -
val_loss: 0.3102 - val_accuracy: 0.8736
Epoch 89/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3031 - accuracy: 0.8764 -
val_loss: 0.3057 - val_accuracy: 0.8707
Epoch 90/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3130 - accuracy: 0.8813 -
val_loss: 0.3060 - val_accuracy: 0.8707
Epoch 91/100
26/26 [==============================] - 0s 15ms/step - loss: 0.2854 - accuracy: 0.8875 -
val_loss: 0.3036 - val_accuracy: 0.8764
Epoch 92/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3075 - accuracy: 0.8752 -
val_loss: 0.3045 - val_accuracy: 0.8736
Epoch 93/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3116 - accuracy: 0.8554 -
val_loss: 0.3018 - val_accuracy: 0.8707
Epoch 94/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3137 - accuracy: 0.8764 -
val_loss: 0.3008 - val_accuracy: 0.8707
Epoch 95/100
26/26 [==============================] - 0s 15ms/step - loss: 0.2889 - accuracy: 0.8739 -
val_loss: 0.2996 - val_accuracy: 0.8736
Epoch 96/100
26/26 [==============================] - 0s 15ms/step - loss: 0.2840 - accuracy: 0.8888 -
val_loss: 0.2985 - val_accuracy: 0.8764
Epoch 97/100
```

```
26/26 [==============================] - 0s 15ms/step - loss: 0.3045 - accuracy: 0.8739 -
val_loss: 0.2950 - val_accuracy: 0.8736
Epoch 98/100
26/26 [==============================] - 0s 15ms/step - loss: 0.2804 - accuracy: 0.8875 -
val_loss: 0.2904 - val_accuracy: 0.8764
Epoch 99/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3101 - accuracy: 0.8727 -
val_loss: 0.2891 - val_accuracy: 0.8793
Epoch 100/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3176 - accuracy: 0.8752 -
val_loss: 0.2856 - val_accuracy: 0.8736
```

In [73]:

```
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 [75]:

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

```
18/18 [==============================] - 0s 8ms/step - loss: 0.2211 - accuracy: 0.9018
Test Accuracy is: 90.1754379272461%
```

Print Confusion Matrix

In [77]:

```
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 [79]:

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

In [80]:

```
car_cm1
```

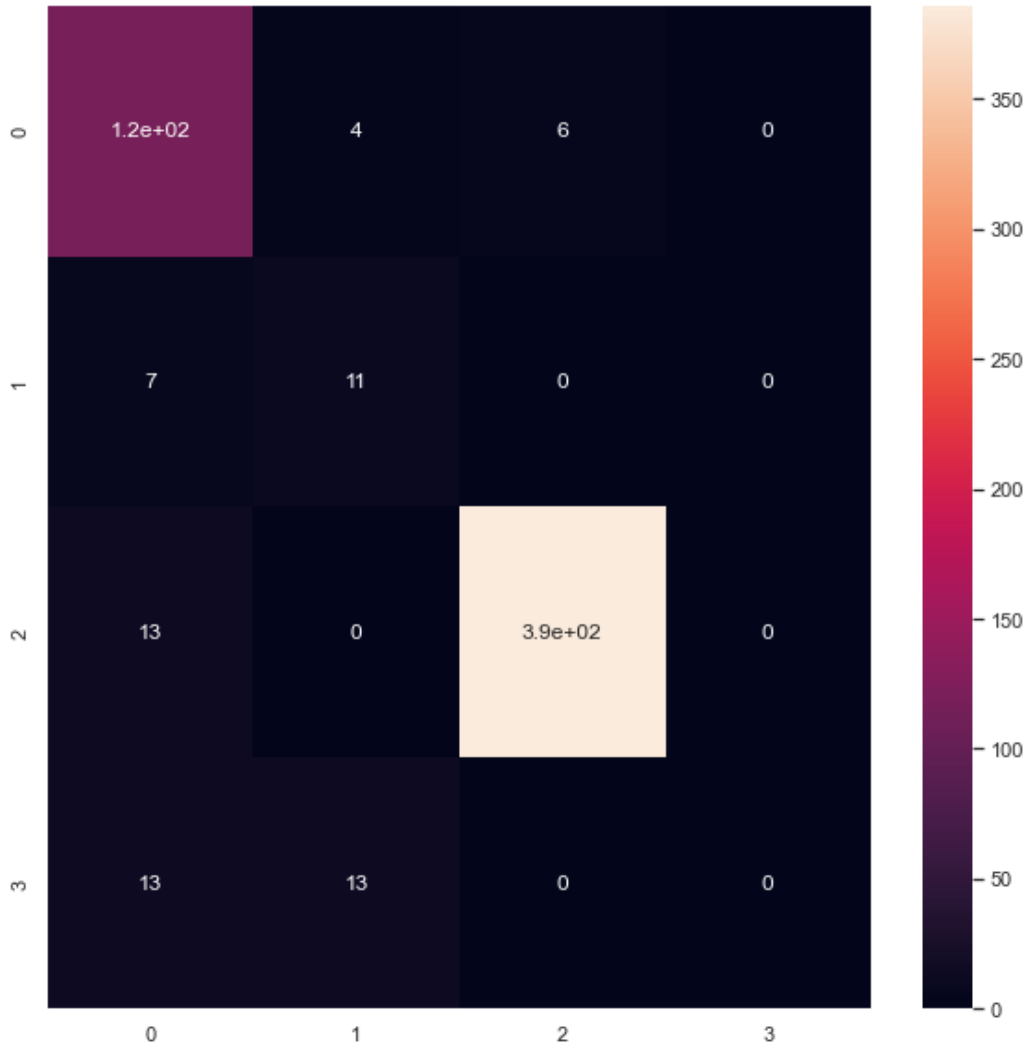
Out[80]:

```
array([[117,   4,   6,   0],
       [  7,  11,   0,   0],
```

```
[ 13,   0, 386,   0],
[ 13,  13,   0,   0]], dtype=int64)
```

In [81]:

```
matrix(car_cm1)
```



2) Ensemble Learning model using Voting Classifier

In [82]:

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

In [83]:

```
y_train
```

Out[83]:

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

In [84]:

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

Out[84]:

```
array([2, 2, 2, 0, 2, 0, 2, 2, 2, 2, 3, 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 [97]:

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

Intializing 3 models for Ensemble learning

In [98]:

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

In [99]:

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

In [100]:

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

```
Epoch 1/100
26/26 [==============================] - 2s 33ms/step - loss: 1.3295 - accuracy: 0.2991 -
val_loss: 1.1008 - val_accuracy: 0.6667
Epoch 2/100
26/26 [==============================] - 1s 20ms/step - loss: 0.9455 - accuracy: 0.6836 -
val_loss: 0.9391 - val_accuracy: 0.6667
Epoch 3/100
26/26 [==============================] - 1s 20ms/step - loss: 0.8491 - accuracy: 0.7206 -
val_loss: 0.9032 - val_accuracy: 0.6667
Epoch 4/100
26/26 [==============================] - 1s 20ms/step - loss: 0.8025 - accuracy: 0.7108 -
val_loss: 0.8804 - val_accuracy: 0.6667
Epoch 5/100
26/26 [==============================] - 0s 19ms/step - loss: 0.7674 - accuracy: 0.7206 -
val_loss: 0.8489 - val_accuracy: 0.6667
Epoch 6/100
26/26 [==============================] - 0s 19ms/step - loss: 0.7064 - accuracy: 0.7342 -
val_loss: 0.7941 - val_accuracy: 0.6667
Epoch 7/100
26/26 [==============================] - 0s 17ms/step - loss: 0.6671 - accuracy: 0.7194 -
val_loss: 0.7430 - val_accuracy: 0.7730
Epoch 8/100
26/26 [==============================] - 0s 16ms/step - loss: 0.6258 - accuracy: 0.7503 -
```

```
val_loss: 0.6913 - val_accuracy: 0.7989
Epoch 9/100
26/26 [==============================] - 0s 18ms/step - loss: 0.6204 - accuracy: 0.7429 -
val_loss: 0.6459 - val_accuracy: 0.7874
Epoch 10/100
26/26 [==============================] - 0s 16ms/step - loss: 0.5832 - accuracy: 0.7849 -
val_loss: 0.6224 - val_accuracy: 0.7787
Epoch 11/100
26/26 [==============================] - 0s 16ms/step - loss: 0.5503 - accuracy: 0.7837 -
val_loss: 0.5954 - val_accuracy: 0.7759
Epoch 12/100
26/26 [==============================] - 0s 16ms/step - loss: 0.5425 - accuracy: 0.7960 -
val_loss: 0.5822 - val_accuracy: 0.7759
Epoch 13/100
26/26 [==============================] - 0s 16ms/step - loss: 0.5291 - accuracy: 0.7948 -
val_loss: 0.5627 - val_accuracy: 0.7845
Epoch 14/100
26/26 [==============================] - 0s 16ms/step - loss: 0.5270 - accuracy: 0.7886 -
val_loss: 0.5420 - val_accuracy: 0.7989
Epoch 15/100
26/26 [==============================] - 0s 16ms/step - loss: 0.5226 - accuracy: 0.8072 -
val_loss: 0.5322 - val_accuracy: 0.7989
Epoch 16/100
26/26 [==============================] - 0s 16ms/step - loss: 0.5153 - accuracy: 0.7960 -
val_loss: 0.5201 - val_accuracy: 0.7960
Epoch 17/100
26/26 [==============================] - 0s 16ms/step - loss: 0.5029 - accuracy: 0.8035 -
val_loss: 0.5050 - val_accuracy: 0.8017
Epoch 18/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4791 - accuracy: 0.8121 -
val_loss: 0.4981 - val_accuracy: 0.8161
Epoch 19/100
26/26 [==============================] - 0s 17ms/step - loss: 0.4700 - accuracy: 0.8220 -
val_loss: 0.4822 - val_accuracy: 0.8247
Epoch 20/100
26/26 [==============================] - 0s 18ms/step - loss: 0.4892 - accuracy: 0.8035 -
val_loss: 0.4763 - val_accuracy: 0.8333
Epoch 21/100
26/26 [==============================] - 0s 17ms/step - loss: 0.4599 - accuracy: 0.8059 -
val_loss: 0.4708 - val_accuracy: 0.8247
Epoch 22/100
26/26 [==============================] - 0s 17ms/step - loss: 0.4626 - accuracy: 0.8171 -
val_loss: 0.4675 - val_accuracy: 0.8218
Epoch 23/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4595 - accuracy: 0.8183 -
val_loss: 0.4620 - val_accuracy: 0.8276
Epoch 24/100
26/26 [==============================] - 1s 27ms/step - loss: 0.4834 - accuracy: 0.8133 -
val_loss: 0.4537 - val_accuracy: 0.8276
Epoch 25/100
26/26 [==============================] - 1s 20ms/step - loss: 0.4340 - accuracy: 0.8282 -
val_loss: 0.4493 - val_accuracy: 0.8305
Epoch 26/100
26/26 [==============================] - 0s 17ms/step - loss: 0.4516 - accuracy: 0.8257 -
val_loss: 0.4512 - val_accuracy: 0.8333
Epoch 27/100
26/26 [==============================] - 0s 17ms/step - loss: 0.4406 - accuracy: 0.8158 -
val_loss: 0.4479 - val_accuracy: 0.8305
Epoch 28/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4406 - accuracy: 0.8282 -
val_loss: 0.4441 - val_accuracy: 0.8362
Epoch 29/100
26/26 [==============================] - 1s 21ms/step - loss: 0.4393 - accuracy: 0.8393 -
val_loss: 0.4381 - val_accuracy: 0.8305
Epoch 30/100
26/26 [==============================] - 0s 17ms/step - loss: 0.4274 - accuracy: 0.8282 -
val_loss: 0.4358 - val_accuracy: 0.8247
Epoch 31/100
26/26 [==============================] - 0s 17ms/step - loss: 0.4168 - accuracy: 0.8368 -
val_loss: 0.4345 - val_accuracy: 0.8333
Epoch 32/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4187 - accuracy: 0.8294 -
```

```
val_loss: 0.4323 - val_accuracy: 0.8333
Epoch 33/100
26/26 [==============================] - 1s 20ms/step - loss: 0.4320 - accuracy: 0.8245 -
val_loss: 0.4259 - val_accuracy: 0.8362
Epoch 34/100
26/26 [==============================] - 0s 17ms/step - loss: 0.4249 - accuracy: 0.8294 -
val_loss: 0.4217 - val_accuracy: 0.8333
Epoch 35/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4219 - accuracy: 0.8331 -
val_loss: 0.4198 - val_accuracy: 0.8362
Epoch 36/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4210 - accuracy: 0.8368 -
val_loss: 0.4159 - val_accuracy: 0.8333
Epoch 37/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4097 - accuracy: 0.8245 -
val_loss: 0.4153 - val_accuracy: 0.8333
Epoch 38/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4189 - accuracy: 0.8257 -
val_loss: 0.4145 - val_accuracy: 0.8305
Epoch 39/100
26/26 [==============================] - 0s 17ms/step - loss: 0.4115 - accuracy: 0.8307 -
val_loss: 0.4122 - val_accuracy: 0.8333
Epoch 40/100
26/26 [==============================] - 0s 17ms/step - loss: 0.4031 - accuracy: 0.8405 -
val_loss: 0.4057 - val_accuracy: 0.8276
Epoch 41/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3937 - accuracy: 0.8418 -
val_loss: 0.4017 - val_accuracy: 0.8218
Epoch 42/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3725 - accuracy: 0.8455 -
val_loss: 0.3942 - val_accuracy: 0.8305
Epoch 43/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3931 - accuracy: 0.8430 -
val_loss: 0.3909 - val_accuracy: 0.8333
Epoch 44/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4153 - accuracy: 0.8319 -
val_loss: 0.3885 - val_accuracy: 0.8420
Epoch 45/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3990 - accuracy: 0.8430 -
val_loss: 0.3865 - val_accuracy: 0.8391
Epoch 46/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3775 - accuracy: 0.8541 -
val_loss: 0.3825 - val_accuracy: 0.8420
Epoch 47/100
26/26 [==============================] - 0s 17ms/step - loss: 0.4002 - accuracy: 0.8245 -
val_loss: 0.3804 - val_accuracy: 0.8391
Epoch 48/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3930 - accuracy: 0.8257 -
val_loss: 0.3778 - val_accuracy: 0.8420
Epoch 49/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3885 - accuracy: 0.8393 -
val_loss: 0.3753 - val_accuracy: 0.8362
Epoch 50/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3599 - accuracy: 0.8504 -
val_loss: 0.3752 - val_accuracy: 0.8391
Epoch 51/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3434 - accuracy: 0.8566 -
val_loss: 0.3717 - val_accuracy: 0.8362
Epoch 52/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3689 - accuracy: 0.8467 -
val_loss: 0.3704 - val_accuracy: 0.8362
Epoch 53/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3773 - accuracy: 0.8393 -
val_loss: 0.3646 - val_accuracy: 0.8391
Epoch 54/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3616 - accuracy: 0.8529 -
val_loss: 0.3608 - val_accuracy: 0.8362
Epoch 55/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3406 - accuracy: 0.8541 -
val_loss: 0.3593 - val_accuracy: 0.8391
Epoch 56/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3659 - accuracy: 0.8443 -
```

```
val_loss: 0.3576 - val_accuracy: 0.8391
Epoch 57/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3543 - accuracy: 0.8467 -
val_loss: 0.3540 - val_accuracy: 0.8362
Epoch 58/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3703 - accuracy: 0.8504 -
val_loss: 0.3502 - val_accuracy: 0.8362
Epoch 59/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3530 - accuracy: 0.8554 -
val_loss: 0.3509 - val_accuracy: 0.8362
Epoch 60/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3448 - accuracy: 0.8566 -
val_loss: 0.3470 - val_accuracy: 0.8420
Epoch 61/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3853 - accuracy: 0.8381 -
val_loss: 0.3458 - val_accuracy: 0.8420
Epoch 62/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3391 - accuracy: 0.8603 -
val_loss: 0.3456 - val_accuracy: 0.8420
Epoch 63/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3235 - accuracy: 0.8677 -
val_loss: 0.3452 - val_accuracy: 0.8391
Epoch 64/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3666 - accuracy: 0.8480 -
val_loss: 0.3420 - val_accuracy: 0.8420
Epoch 65/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3488 - accuracy: 0.8702 -
val_loss: 0.3390 - val_accuracy: 0.8420
Epoch 66/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3472 - accuracy: 0.8517 -
val_loss: 0.3335 - val_accuracy: 0.8477
Epoch 67/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3159 - accuracy: 0.8653 -
val_loss: 0.3291 - val_accuracy: 0.8563
Epoch 68/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3492 - accuracy: 0.8554 -
val_loss: 0.3269 - val_accuracy: 0.8678
Epoch 69/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3538 - accuracy: 0.8418 -
val_loss: 0.3246 - val_accuracy: 0.8678
Epoch 70/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3219 - accuracy: 0.8677 -
val_loss: 0.3243 - val_accuracy: 0.8707
Epoch 71/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3262 - accuracy: 0.8640 -
val_loss: 0.3251 - val_accuracy: 0.8736
Epoch 72/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3253 - accuracy: 0.8677 -
val_loss: 0.3238 - val_accuracy: 0.8592
Epoch 73/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3167 - accuracy: 0.8714 -
val_loss: 0.3224 - val_accuracy: 0.8707
Epoch 74/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3343 - accuracy: 0.8640 -
val_loss: 0.3213 - val_accuracy: 0.8736
Epoch 75/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3263 - accuracy: 0.8665 -
val_loss: 0.3191 - val_accuracy: 0.8707
Epoch 76/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3164 - accuracy: 0.8690 -
val_loss: 0.3160 - val_accuracy: 0.8707
Epoch 77/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3504 - accuracy: 0.8504 -
val_loss: 0.3151 - val_accuracy: 0.8793
Epoch 78/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3308 - accuracy: 0.8665 -
val_loss: 0.3114 - val_accuracy: 0.8736
Epoch 79/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3215 - accuracy: 0.8690 -
val_loss: 0.3088 - val_accuracy: 0.8678
Epoch 80/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3130 - accuracy: 0.8603 -
```

```
val_loss: 0.3096 - val_accuracy: 0.8592
Epoch 81/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3086 - accuracy: 0.8727 -
val_loss: 0.3095 - val_accuracy: 0.8649
Epoch 82/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3126 - accuracy: 0.8739 -
val_loss: 0.3090 - val_accuracy: 0.8764
Epoch 83/100
26/26 [==============================] - 0s 16ms/step - loss: 0.2995 - accuracy: 0.8714 -
val_loss: 0.3061 - val_accuracy: 0.8707
Epoch 84/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3114 - accuracy: 0.8603 -
val_loss: 0.3078 - val_accuracy: 0.8793
Epoch 85/100
26/26 [==============================] - 0s 15ms/step - loss: 0.2978 - accuracy: 0.8776 -
val_loss: 0.3044 - val_accuracy: 0.8822
Epoch 86/100
26/26 [==============================] - 0s 16ms/step - loss: 0.2889 - accuracy: 0.8850 -
val_loss: 0.3018 - val_accuracy: 0.8851
Epoch 87/100
26/26 [==============================] - 0s 16ms/step - loss: 0.2886 - accuracy: 0.8912 -
val_loss: 0.3030 - val_accuracy: 0.8793
Epoch 88/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3223 - accuracy: 0.8677 -
val_loss: 0.2984 - val_accuracy: 0.8822
Epoch 89/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3050 - accuracy: 0.8789 -
val_loss: 0.2971 - val_accuracy: 0.8851
Epoch 90/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3129 - accuracy: 0.8714 -
val_loss: 0.2955 - val_accuracy: 0.8822
Epoch 91/100
26/26 [==============================] - 0s 16ms/step - loss: 0.2882 - accuracy: 0.8875 -
val_loss: 0.2948 - val_accuracy: 0.8764
Epoch 92/100
26/26 [==============================] - 0s 16ms/step - loss: 0.2749 - accuracy: 0.8925 -
val_loss: 0.2934 - val_accuracy: 0.8764
Epoch 93/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3064 - accuracy: 0.8690 -
val_loss: 0.2917 - val_accuracy: 0.8764
Epoch 94/100
26/26 [==============================] - 0s 15ms/step - loss: 0.2860 - accuracy: 0.8937 -
val_loss: 0.2927 - val_accuracy: 0.8764
Epoch 95/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3478 - accuracy: 0.8529 -
val_loss: 0.2888 - val_accuracy: 0.8793
Epoch 96/100
26/26 [==============================] - 0s 18ms/step - loss: 0.2945 - accuracy: 0.8727 -
val_loss: 0.2873 - val_accuracy: 0.8764
Epoch 97/100
26/26 [==============================] - 0s 16ms/step - loss: 0.2868 - accuracy: 0.8838 -
val_loss: 0.2860 - val_accuracy: 0.8793
Epoch 98/100
26/26 [==============================] - 0s 15ms/step - loss: 0.2935 - accuracy: 0.8752 -
val_loss: 0.2850 - val_accuracy: 0.8822
Epoch 99/100
26/26 [==============================] - 0s 15ms/step - loss: 0.2728 - accuracy: 0.8925 -
val_loss: 0.2817 - val_accuracy: 0.8822
Epoch 100/100
26/26 [==============================] - 0s 15ms/step - loss: 0.2835 - accuracy: 0.8888 -
val_loss: 0.2826 - val_accuracy: 0.8822
Epoch 1/100
26/26 [==============================] - 2s 24ms/step - loss: 1.3234 - accuracy: 0.3548 -
val_loss: 1.0610 - val_accuracy: 0.6667
Epoch 2/100
26/26 [==============================] - 0s 16ms/step - loss: 0.9660 - accuracy: 0.6613 -
val_loss: 0.9327 - val_accuracy: 0.6667
Epoch 3/100
26/26 [==============================] - 0s 15ms/step - loss: 0.8135 - accuracy: 0.7244 -
val_loss: 0.8914 - val_accuracy: 0.6667
Epoch 4/100
26/26 [==============================] - 0s 15ms/step - loss: 0.7490 - accuracy: 0.7182 -
```

```
val_loss: 0.8450 - val_accuracy: 0.6667
Epoch 5/100
26/26 [==============================] - 0s 15ms/step - loss: 0.7103 - accuracy: 0.7404 -
val_loss: 0.7812 - val_accuracy: 0.6667
Epoch 6/100
26/26 [==============================] - 0s 15ms/step - loss: 0.6502 - accuracy: 0.7565 -
val_loss: 0.7049 - val_accuracy: 0.7098
Epoch 7/100
26/26 [==============================] - 0s 15ms/step - loss: 0.6171 - accuracy: 0.7651 -
val_loss: 0.6363 - val_accuracy: 0.7701
Epoch 8/100
26/26 [==============================] - 0s 16ms/step - loss: 0.6039 - accuracy: 0.7689 -
val_loss: 0.5816 - val_accuracy: 0.8161
Epoch 9/100
26/26 [==============================] - 0s 15ms/step - loss: 0.5827 - accuracy: 0.7664 -
val_loss: 0.5450 - val_accuracy: 0.8218
Epoch 10/100
26/26 [==============================] - ETA: 0s - loss: 0.5580 - accuracy: 0.76 - 0s 15m
s/step - loss: 0.5569 - accuracy: 0.7701 - val_loss: 0.5199 - val_accuracy: 0.8190
Epoch 11/100
26/26 [==============================] - 0s 16ms/step - loss: 0.5386 - accuracy: 0.7936 -
val_loss: 0.4997 - val_accuracy: 0.8391
Epoch 12/100
26/26 [==============================] - 0s 16ms/step - loss: 0.5183 - accuracy: 0.8047 -
val_loss: 0.4820 - val_accuracy: 0.8247
Epoch 13/100
26/26 [==============================] - 0s 15ms/step - loss: 0.5087 - accuracy: 0.8096 -
val_loss: 0.4740 - val_accuracy: 0.8333
Epoch 14/100
26/26 [==============================] - 0s 16ms/step - loss: 0.5046 - accuracy: 0.7923 -
val_loss: 0.4710 - val_accuracy: 0.8276
Epoch 15/100
26/26 [==============================] - 0s 15ms/step - loss: 0.5191 - accuracy: 0.7824 -
val_loss: 0.4644 - val_accuracy: 0.8305
Epoch 16/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4851 - accuracy: 0.8195 -
val_loss: 0.4573 - val_accuracy: 0.8333
Epoch 17/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4892 - accuracy: 0.8035 -
val_loss: 0.4570 - val_accuracy: 0.8362
Epoch 18/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4563 - accuracy: 0.8195 -
val_loss: 0.4559 - val_accuracy: 0.8333
Epoch 19/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4624 - accuracy: 0.8269 -
val_loss: 0.4550 - val_accuracy: 0.8391
Epoch 20/100
26/26 [==============================] - 0s 17ms/step - loss: 0.4659 - accuracy: 0.8183 -
val_loss: 0.4561 - val_accuracy: 0.8333
Epoch 21/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4619 - accuracy: 0.8096 -
val_loss: 0.4492 - val_accuracy: 0.8333
Epoch 22/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4876 - accuracy: 0.8035 -
val_loss: 0.4490 - val_accuracy: 0.8333
Epoch 23/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4553 - accuracy: 0.8171 -
val_loss: 0.4526 - val_accuracy: 0.8305
Epoch 24/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4552 - accuracy: 0.8059 -
val_loss: 0.4466 - val_accuracy: 0.8333
Epoch 25/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4506 - accuracy: 0.8232 -
val_loss: 0.4450 - val_accuracy: 0.8333
Epoch 26/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4262 - accuracy: 0.8294 -
val_loss: 0.4465 - val_accuracy: 0.8305
Epoch 27/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4534 - accuracy: 0.8232 -
val_loss: 0.4412 - val_accuracy: 0.8276
Epoch 28/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4475 - accuracy: 0.8195 -
```

```
val_loss: 0.4399 - val_accuracy: 0.8305
Epoch 29/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4188 - accuracy: 0.8356 -
val_loss: 0.4376 - val_accuracy: 0.8276
Epoch 30/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4503 - accuracy: 0.8232 -
val_loss: 0.4354 - val_accuracy: 0.8362
Epoch 31/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4291 - accuracy: 0.8257 -
val_loss: 0.4324 - val_accuracy: 0.8333
Epoch 32/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4244 - accuracy: 0.8405 -
val_loss: 0.4315 - val_accuracy: 0.8362
Epoch 33/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4223 - accuracy: 0.8344 -
val_loss: 0.4305 - val_accuracy: 0.8333
Epoch 34/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4130 - accuracy: 0.8418 -
val_loss: 0.4288 - val_accuracy: 0.8362
Epoch 35/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4008 - accuracy: 0.8554 -
val_loss: 0.4264 - val_accuracy: 0.8362
Epoch 36/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4243 - accuracy: 0.8282 -
val_loss: 0.4220 - val_accuracy: 0.8362
Epoch 37/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4071 - accuracy: 0.8455 -
val_loss: 0.4213 - val_accuracy: 0.8362
Epoch 38/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3950 - accuracy: 0.8381 -
val_loss: 0.4177 - val_accuracy: 0.8391
Epoch 39/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4118 - accuracy: 0.8331 -
val_loss: 0.4123 - val_accuracy: 0.8333
Epoch 40/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3839 - accuracy: 0.8529 -
val_loss: 0.4101 - val_accuracy: 0.8362
Epoch 41/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3797 - accuracy: 0.8492 -
val_loss: 0.4094 - val_accuracy: 0.8333
Epoch 42/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3897 - accuracy: 0.8480 -
val_loss: 0.4074 - val_accuracy: 0.8247
Epoch 43/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3950 - accuracy: 0.8455 -
val_loss: 0.4071 - val_accuracy: 0.8305
Epoch 44/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3864 - accuracy: 0.8541 -
val_loss: 0.4020 - val_accuracy: 0.8333
Epoch 45/100
26/26 [==============================] - 0s 17ms/step - loss: 0.4040 - accuracy: 0.8381 -
val_loss: 0.4027 - val_accuracy: 0.8362
Epoch 46/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3964 - accuracy: 0.8393 -
val_loss: 0.3943 - val_accuracy: 0.8362
Epoch 47/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3988 - accuracy: 0.8393 -
val_loss: 0.3936 - val_accuracy: 0.8362
Epoch 48/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4040 - accuracy: 0.8220 -
val_loss: 0.3944 - val_accuracy: 0.8333
Epoch 49/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3939 - accuracy: 0.8393 -
val_loss: 0.3914 - val_accuracy: 0.8420
Epoch 50/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3811 - accuracy: 0.8443 -
val_loss: 0.3935 - val_accuracy: 0.8391
Epoch 51/100
26/26 [==============================] - 0s 17ms/step - loss: 0.4241 - accuracy: 0.8381 -
val_loss: 0.3873 - val_accuracy: 0.8420
Epoch 52/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4002 - accuracy: 0.8480 -
```

```
val_loss: 0.3873 - val_accuracy: 0.8362
Epoch 53/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3616 - accuracy: 0.8467 -
val_loss: 0.3865 - val_accuracy: 0.8362
Epoch 54/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3535 - accuracy: 0.8677 -
val_loss: 0.3838 - val_accuracy: 0.8333
Epoch 55/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3763 - accuracy: 0.8405 -
val_loss: 0.3788 - val_accuracy: 0.8391
Epoch 56/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3677 - accuracy: 0.8653 -
val_loss: 0.3735 - val_accuracy: 0.8362
Epoch 57/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3716 - accuracy: 0.8492 -
val_loss: 0.3695 - val_accuracy: 0.8362
Epoch 58/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3592 - accuracy: 0.8566 -
val_loss: 0.3667 - val_accuracy: 0.8391
Epoch 59/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3547 - accuracy: 0.8517 -
val_loss: 0.3659 - val_accuracy: 0.8391
Epoch 60/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3427 - accuracy: 0.8566 -
val_loss: 0.3601 - val_accuracy: 0.8448
Epoch 61/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3710 - accuracy: 0.8443 -
val_loss: 0.3565 - val_accuracy: 0.8391
Epoch 62/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3621 - accuracy: 0.8517 -
val_loss: 0.3558 - val_accuracy: 0.8477
Epoch 63/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3533 - accuracy: 0.8591 -
val_loss: 0.3552 - val_accuracy: 0.8391
Epoch 64/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3724 - accuracy: 0.8405 -
val_loss: 0.3523 - val_accuracy: 0.8420
Epoch 65/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3485 - accuracy: 0.8578 -
val_loss: 0.3483 - val_accuracy: 0.8362
Epoch 66/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3700 - accuracy: 0.8418 -
val_loss: 0.3490 - val_accuracy: 0.8420
Epoch 67/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3642 - accuracy: 0.8443 -
val_loss: 0.3458 - val_accuracy: 0.8391
Epoch 68/100
26/26 [==============================] - 0s 14ms/step - loss: 0.3436 - accuracy: 0.8541 -
val_loss: 0.3437 - val_accuracy: 0.8362
Epoch 69/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3412 - accuracy: 0.8628 -
val_loss: 0.3446 - val_accuracy: 0.8420
Epoch 70/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3324 - accuracy: 0.8714 -
val_loss: 0.3429 - val_accuracy: 0.8420
Epoch 71/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3605 - accuracy: 0.8492 -
val_loss: 0.3355 - val_accuracy: 0.8477
Epoch 72/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3253 - accuracy: 0.8752 -
val_loss: 0.3344 - val_accuracy: 0.8391
Epoch 73/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3492 - accuracy: 0.8603 -
val_loss: 0.3322 - val_accuracy: 0.8391
Epoch 74/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3229 - accuracy: 0.8616 -
val_loss: 0.3327 - val_accuracy: 0.8391
Epoch 75/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3431 - accuracy: 0.8529 -
val_loss: 0.3335 - val_accuracy: 0.8362
Epoch 76/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3204 - accuracy: 0.8665 -
```

```
val_loss: 0.3306 - val_accuracy: 0.8362
Epoch 77/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3258 - accuracy: 0.8628 -
val_loss: 0.3256 - val_accuracy: 0.8420
Epoch 78/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3238 - accuracy: 0.8702 -
val_loss: 0.3266 - val_accuracy: 0.8391
Epoch 79/100
26/26 [==============================] - 0s 15ms/step - loss: 0.2955 - accuracy: 0.8776 -
val_loss: 0.3260 - val_accuracy: 0.8391
Epoch 80/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3192 - accuracy: 0.8665 -
val_loss: 0.3260 - val_accuracy: 0.8362
Epoch 81/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3193 - accuracy: 0.8690 -
val_loss: 0.3213 - val_accuracy: 0.8448
Epoch 82/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3232 - accuracy: 0.8727 -
val_loss: 0.3180 - val_accuracy: 0.8563
Epoch 83/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3305 - accuracy: 0.8616 -
val_loss: 0.3144 - val_accuracy: 0.8592
Epoch 84/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3330 - accuracy: 0.8690 -
val_loss: 0.3162 - val_accuracy: 0.8420
Epoch 85/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3100 - accuracy: 0.8702 -
val_loss: 0.3176 - val_accuracy: 0.8506
Epoch 86/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3154 - accuracy: 0.8702 -
val_loss: 0.3164 - val_accuracy: 0.8534
Epoch 87/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3367 - accuracy: 0.8616 -
val_loss: 0.3179 - val_accuracy: 0.8563
Epoch 88/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3126 - accuracy: 0.8665 -
val_loss: 0.3157 - val_accuracy: 0.8649
Epoch 89/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3012 - accuracy: 0.8764 -
val_loss: 0.3118 - val_accuracy: 0.8563
Epoch 90/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3084 - accuracy: 0.8702 -
val_loss: 0.3073 - val_accuracy: 0.8649
Epoch 91/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3182 - accuracy: 0.8591 -
val_loss: 0.3057 - val_accuracy: 0.8621
Epoch 92/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3002 - accuracy: 0.8739 -
val_loss: 0.3029 - val_accuracy: 0.8649
Epoch 93/100
26/26 [==============================] - 0s 15ms/step - loss: 0.2803 - accuracy: 0.8727 -
val_loss: 0.3001 - val_accuracy: 0.8621
Epoch 94/100
26/26 [==============================] - 0s 15ms/step - loss: 0.2868 - accuracy: 0.8764 -
val_loss: 0.3000 - val_accuracy: 0.8764
Epoch 95/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3082 - accuracy: 0.8739 -
val_loss: 0.3007 - val_accuracy: 0.8764
Epoch 96/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3032 - accuracy: 0.8789 -
val_loss: 0.2989 - val_accuracy: 0.8764
Epoch 97/100
26/26 [==============================] - 0s 15ms/step - loss: 0.2789 - accuracy: 0.8875 -
val_loss: 0.2941 - val_accuracy: 0.8764
Epoch 98/100
26/26 [==============================] - 0s 15ms/step - loss: 0.2755 - accuracy: 0.8912 -
val_loss: 0.2925 - val_accuracy: 0.8736
Epoch 99/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3003 - accuracy: 0.8714 -
val_loss: 0.2955 - val_accuracy: 0.8707
Epoch 100/100
26/26 [==============================] - 0s 15ms/step - loss: 0.2815 - accuracy: 0.8714 -
```

```
val_loss: 0.2915 - val_accuracy: 0.8736
Epoch 1/100
26/26 [==============================] - 2s 23ms/step - loss: 1.5059 - accuracy: 0.2546 -
val_loss: 1.0833 - val_accuracy: 0.6667
Epoch 2/100
26/26 [==============================] - 0s 15ms/step - loss: 1.0002 - accuracy: 0.6020 -
val_loss: 0.9385 - val_accuracy: 0.6667
Epoch 3/100
26/26 [==============================] - 0s 15ms/step - loss: 0.7982 - accuracy: 0.7083 -
val_loss: 0.9369 - val_accuracy: 0.6667
Epoch 4/100
26/26 [==============================] - 0s 15ms/step - loss: 0.7495 - accuracy: 0.7194 -
val_loss: 0.9303 - val_accuracy: 0.6667
Epoch 5/100
26/26 [==============================] - 0s 16ms/step - loss: 0.6901 - accuracy: 0.7194 -
val_loss: 0.9016 - val_accuracy: 0.6667
Epoch 6/100
26/26 [==============================] - 0s 16ms/step - loss: 0.6409 - accuracy: 0.7553 -
val_loss: 0.8565 - val_accuracy: 0.6667
Epoch 7/100
26/26 [==============================] - 0s 15ms/step - loss: 0.6292 - accuracy: 0.7577 -
val_loss: 0.8234 - val_accuracy: 0.6667
Epoch 8/100
26/26 [==============================] - 0s 16ms/step - loss: 0.6042 - accuracy: 0.7602 -
val_loss: 0.7647 - val_accuracy: 0.6667
Epoch 9/100
26/26 [==============================] - 0s 16ms/step - loss: 0.5844 - accuracy: 0.7787 -
val_loss: 0.7102 - val_accuracy: 0.6667
Epoch 10/100
26/26 [==============================] - 0s 15ms/step - loss: 0.5713 - accuracy: 0.7701 -
val_loss: 0.6472 - val_accuracy: 0.6782
Epoch 11/100
26/26 [==============================] - 0s 17ms/step - loss: 0.5353 - accuracy: 0.7849 -
val_loss: 0.6029 - val_accuracy: 0.6983
Epoch 12/100
26/26 [==============================] - 0s 16ms/step - loss: 0.5502 - accuracy: 0.7812 -
val_loss: 0.5644 - val_accuracy: 0.7328
Epoch 13/100
26/26 [==============================] - 0s 15ms/step - loss: 0.5131 - accuracy: 0.8121 -
val_loss: 0.5561 - val_accuracy: 0.7443
Epoch 14/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4954 - accuracy: 0.8146 -
val_loss: 0.5467 - val_accuracy: 0.7586
Epoch 15/100
26/26 [==============================] - 0s 15ms/step - loss: 0.5103 - accuracy: 0.8010 -
val_loss: 0.5228 - val_accuracy: 0.7701
Epoch 16/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4876 - accuracy: 0.8195 -
val_loss: 0.5114 - val_accuracy: 0.7672
Epoch 17/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4878 - accuracy: 0.8096 -
val_loss: 0.4990 - val_accuracy: 0.7989
Epoch 18/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4796 - accuracy: 0.8220 -
val_loss: 0.4882 - val_accuracy: 0.8075
Epoch 19/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4826 - accuracy: 0.8220 -
val_loss: 0.4862 - val_accuracy: 0.8132
Epoch 20/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4857 - accuracy: 0.8010 -
val_loss: 0.4817 - val_accuracy: 0.8190
Epoch 21/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4655 - accuracy: 0.8245 -
val_loss: 0.4804 - val_accuracy: 0.8161
Epoch 22/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4648 - accuracy: 0.8183 -
val_loss: 0.4782 - val_accuracy: 0.8190
Epoch 23/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4561 - accuracy: 0.8245 -
val_loss: 0.4724 - val_accuracy: 0.8218
Epoch 24/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4594 - accuracy: 0.8269 -
```

```
val_loss: 0.4710 - val_accuracy: 0.8161
Epoch 25/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4290 - accuracy: 0.8405 -
val_loss: 0.4650 - val_accuracy: 0.8247
Epoch 26/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4539 - accuracy: 0.8208 -
val_loss: 0.4627 - val_accuracy: 0.8218
Epoch 27/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4334 - accuracy: 0.8455 -
val_loss: 0.4561 - val_accuracy: 0.8247
Epoch 28/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4446 - accuracy: 0.8282 -
val_loss: 0.4538 - val_accuracy: 0.8305
Epoch 29/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4381 - accuracy: 0.8294 -
val_loss: 0.4502 - val_accuracy: 0.8276
Epoch 30/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4442 - accuracy: 0.8183 -
val_loss: 0.4482 - val_accuracy: 0.8218
Epoch 31/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4196 - accuracy: 0.8307 -
val_loss: 0.4454 - val_accuracy: 0.8247
Epoch 32/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4376 - accuracy: 0.8430 -
val_loss: 0.4485 - val_accuracy: 0.8247
Epoch 33/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4398 - accuracy: 0.8245 -
val_loss: 0.4431 - val_accuracy: 0.8247
Epoch 34/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4400 - accuracy: 0.8257 -
val_loss: 0.4393 - val_accuracy: 0.8305
Epoch 35/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4292 - accuracy: 0.8294 -
val_loss: 0.4365 - val_accuracy: 0.8305
Epoch 36/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4146 - accuracy: 0.8344 -
val_loss: 0.4279 - val_accuracy: 0.8333
Epoch 37/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4255 - accuracy: 0.8381 -
val_loss: 0.4222 - val_accuracy: 0.8305
Epoch 38/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4021 - accuracy: 0.8331 -
val_loss: 0.4195 - val_accuracy: 0.8391
Epoch 39/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4158 - accuracy: 0.8245 -
val_loss: 0.4168 - val_accuracy: 0.8362
Epoch 40/100
26/26 [==============================] - 0s 15ms/step - loss: 0.4245 - accuracy: 0.8294 -
val_loss: 0.4161 - val_accuracy: 0.8333
Epoch 41/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3795 - accuracy: 0.8541 -
val_loss: 0.4108 - val_accuracy: 0.8276
Epoch 42/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4149 - accuracy: 0.8319 -
val_loss: 0.4090 - val_accuracy: 0.8362
Epoch 43/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3985 - accuracy: 0.8418 -
val_loss: 0.4039 - val_accuracy: 0.8362
Epoch 44/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4020 - accuracy: 0.8455 -
val_loss: 0.4023 - val_accuracy: 0.8362
Epoch 45/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3978 - accuracy: 0.8418 -
val_loss: 0.4004 - val_accuracy: 0.8362
Epoch 46/100
26/26 [==============================] - 0s 16ms/step - loss: 0.4032 - accuracy: 0.8319 -
val_loss: 0.3926 - val_accuracy: 0.8420
Epoch 47/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3853 - accuracy: 0.8405 -
val_loss: 0.3866 - val_accuracy: 0.8362
Epoch 48/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3697 - accuracy: 0.8492 -
```

```
val_loss: 0.3875 - val_accuracy: 0.8391
Epoch 49/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3854 - accuracy: 0.8418 -
val_loss: 0.3853 - val_accuracy: 0.8362
Epoch 50/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3763 - accuracy: 0.8554 -
val_loss: 0.3868 - val_accuracy: 0.8362
Epoch 51/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3946 - accuracy: 0.8393 -
val_loss: 0.3809 - val_accuracy: 0.8333
Epoch 52/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3647 - accuracy: 0.8517 -
val_loss: 0.3776 - val_accuracy: 0.8362
Epoch 53/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3601 - accuracy: 0.8480 -
val_loss: 0.3758 - val_accuracy: 0.8362
Epoch 54/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3867 - accuracy: 0.8480 -
val_loss: 0.3709 - val_accuracy: 0.8391
Epoch 55/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3555 - accuracy: 0.8566 -
val_loss: 0.3682 - val_accuracy: 0.8362
Epoch 56/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3510 - accuracy: 0.8578 -
val_loss: 0.3648 - val_accuracy: 0.8420
Epoch 57/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3477 - accuracy: 0.8591 -
val_loss: 0.3592 - val_accuracy: 0.8420
Epoch 58/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3447 - accuracy: 0.8640 -
val_loss: 0.3581 - val_accuracy: 0.8391
Epoch 59/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3673 - accuracy: 0.8443 -
val_loss: 0.3585 - val_accuracy: 0.8391
Epoch 60/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3851 - accuracy: 0.8430 -
val_loss: 0.3626 - val_accuracy: 0.8333
Epoch 61/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3499 - accuracy: 0.8578 -
val_loss: 0.3580 - val_accuracy: 0.8391
Epoch 62/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3483 - accuracy: 0.8578 -
val_loss: 0.3521 - val_accuracy: 0.8391
Epoch 63/100
26/26 [==============================] - 0s 18ms/step - loss: 0.3358 - accuracy: 0.8640 -
val_loss: 0.3516 - val_accuracy: 0.8362
Epoch 64/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3511 - accuracy: 0.8504 -
val_loss: 0.3458 - val_accuracy: 0.8391
Epoch 65/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3474 - accuracy: 0.8603 -
val_loss: 0.3434 - val_accuracy: 0.8420
Epoch 66/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3396 - accuracy: 0.8566 -
val_loss: 0.3399 - val_accuracy: 0.8391
Epoch 67/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3420 - accuracy: 0.8653 -
val_loss: 0.3383 - val_accuracy: 0.8391
Epoch 68/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3308 - accuracy: 0.8665 -
val_loss: 0.3380 - val_accuracy: 0.8362
Epoch 69/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3484 - accuracy: 0.8504 -
val_loss: 0.3346 - val_accuracy: 0.8391
Epoch 70/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3424 - accuracy: 0.8665 -
val_loss: 0.3324 - val_accuracy: 0.8391
Epoch 71/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3391 - accuracy: 0.8566 -
val_loss: 0.3348 - val_accuracy: 0.8420
Epoch 72/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3002 - accuracy: 0.8789 -
```

```
val_loss: 0.3324 - val_accuracy: 0.8477
Epoch 73/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3221 - accuracy: 0.8677 -
val_loss: 0.3292 - val_accuracy: 0.8506
Epoch 74/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3085 - accuracy: 0.8727 -
val_loss: 0.3285 - val_accuracy: 0.8506
Epoch 75/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3406 - accuracy: 0.8640 -
val_loss: 0.3265 - val_accuracy: 0.8506
Epoch 76/100
26/26 [==============================] - 0s 15ms/step - loss: 0.2959 - accuracy: 0.8838 -
val_loss: 0.3236 - val_accuracy: 0.8534
Epoch 77/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3200 - accuracy: 0.8665 -
val_loss: 0.3254 - val_accuracy: 0.8534
Epoch 78/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3368 - accuracy: 0.8603 -
val_loss: 0.3198 - val_accuracy: 0.8621
Epoch 79/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3071 - accuracy: 0.8789 -
val_loss: 0.3170 - val_accuracy: 0.8621
Epoch 80/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3106 - accuracy: 0.8764 -
val_loss: 0.3175 - val_accuracy: 0.8592
Epoch 81/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3105 - accuracy: 0.8789 -
val_loss: 0.3168 - val_accuracy: 0.8592
Epoch 82/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3300 - accuracy: 0.8727 -
val_loss: 0.3158 - val_accuracy: 0.8592
Epoch 83/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3228 - accuracy: 0.8467 -
val_loss: 0.3162 - val_accuracy: 0.8649
Epoch 84/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3141 - accuracy: 0.8665 -
val_loss: 0.3150 - val_accuracy: 0.8621
Epoch 85/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3231 - accuracy: 0.8702 -
val_loss: 0.3162 - val_accuracy: 0.8592
Epoch 86/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3113 - accuracy: 0.8776 -
val_loss: 0.3136 - val_accuracy: 0.8592
Epoch 87/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3024 - accuracy: 0.8714 -
val_loss: 0.3100 - val_accuracy: 0.8621
Epoch 88/100
26/26 [==============================] - 0s 16ms/step - loss: 0.3044 - accuracy: 0.8739 -
val_loss: 0.3061 - val_accuracy: 0.8649
Epoch 89/100
26/26 [==============================] - 0s 16ms/step - loss: 0.2970 - accuracy: 0.8739 -
val_loss: 0.3044 - val_accuracy: 0.8678
Epoch 90/100
26/26 [==============================] - 0s 15ms/step - loss: 0.2995 - accuracy: 0.8702 -
val_loss: 0.3051 - val_accuracy: 0.8793
Epoch 91/100
26/26 [==============================] - 0s 16ms/step - loss: 0.2968 - accuracy: 0.8764 -
val_loss: 0.3026 - val_accuracy: 0.8764
Epoch 92/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3108 - accuracy: 0.8603 -
val_loss: 0.3010 - val_accuracy: 0.8592
Epoch 93/100
26/26 [==============================] - 0s 16ms/step - loss: 0.2754 - accuracy: 0.8888 -
val_loss: 0.2988 - val_accuracy: 0.8592
Epoch 94/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3203 - accuracy: 0.8690 -
val_loss: 0.2949 - val_accuracy: 0.8649
Epoch 95/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3152 - accuracy: 0.8752 -
val_loss: 0.2936 - val_accuracy: 0.8678
Epoch 96/100
26/26 [==============================] - 0s 17ms/step - loss: 0.3064 - accuracy: 0.8714 -
```

```
val_loss: 0.2959 - val_accuracy: 0.8736
Epoch 97/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3152 - accuracy: 0.8714 -
val_loss: 0.2967 - val_accuracy: 0.8793
Epoch 98/100
26/26 [==============================] - 0s 15ms/step - loss: 0.2893 - accuracy: 0.8875 -
val_loss: 0.2941 - val_accuracy: 0.8678
Epoch 99/100
26/26 [==============================] - 0s 15ms/step - loss: 0.3055 - accuracy: 0.8714 -
val_loss: 0.2961 - val_accuracy: 0.8649
Epoch 100/100
26/26 [==============================] - 0s 15ms/step - loss: 0.2873 - accuracy: 0.8826 -
val_loss: 0.2925 - val_accuracy: 0.8764
```

In [93]:

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

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

Test Accuracy

In [95]:

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

In [96]:

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

```
Test Accuracy is : 89.47368421052632  %
```

Confusion Matrix

In [101]:

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

In [102]:

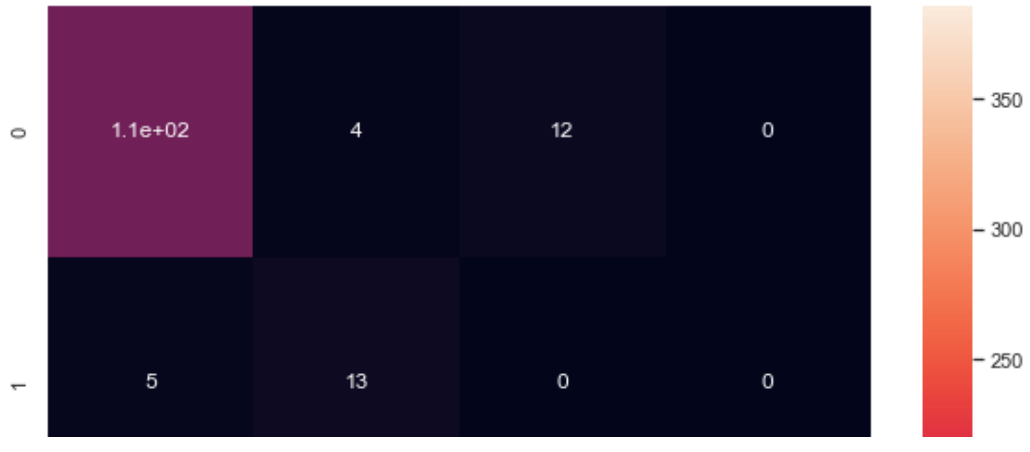
```
ecm
```

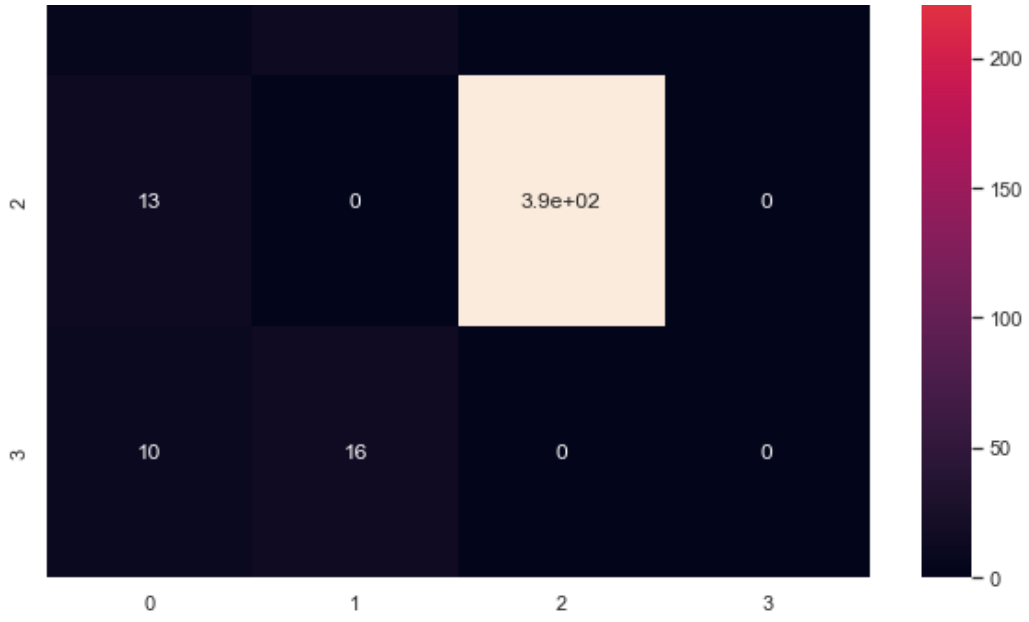
Out[102]:

```
array([[111,   4,  12,   0],
       [  5,  13,   0,   0],
       [ 13,   0, 386,   0],
       [ 10,  16,   0,   0]], dtype=int64)
```

In [103]:

```
matrix(ecm)
```





Inference:

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