

MNIST English handwritten numerals Classification

May 1, 2020

0.1 Dataset Reading and Visualization

```
[1]: from mlxtend.data import loadlocal_mnist # library to load ubyte files
import matplotlib.pyplot as plt
import numpy as np
import time
import cv2
import pandas as pd
```

```
[43]: from typing import Tuple, Callable
import matplotlib.pyplot as plt
import keras
from keras.utils import to_categorical
from keras import layers
from keras import models
from keras import regularizers
from keras.applications.vgg16 import VGG16
from keras.applications.resnet import ResNet50
from keras.engine.training import Model
import functools
import os
import numpy as np
```

```
[3]: # loading train and test dataset
X_train, y_train = loadlocal_mnist(images_path='Dataset/train-images.
    ↳idx3-ubyte', labels_path='Dataset/train-labels.idx1-ubyte')
X_test, y_test = loadlocal_mnist(images_path='Dataset/t10k-images.
    ↳idx3-ubyte', labels_path='Dataset/t10k-labels.idx1-ubyte')
num_classes=len(np.unique(y_train))
```

Mnist Handwritten dataset consists of 60000 training images of digits and 10000 testing images. The dataset consists of 28*28 gray scale images.

```
[4]: print('Training Dataset shape:', X_train.shape)
print('Training labels shape:', y_train.shape)

print('Testing Dataset shape:', X_test.shape)
print('Testing labels shape:', y_test.shape)
```

```
print('Unique Labels:',np.unique(y_train))
```

Training Dataset shape: (60000, 784)

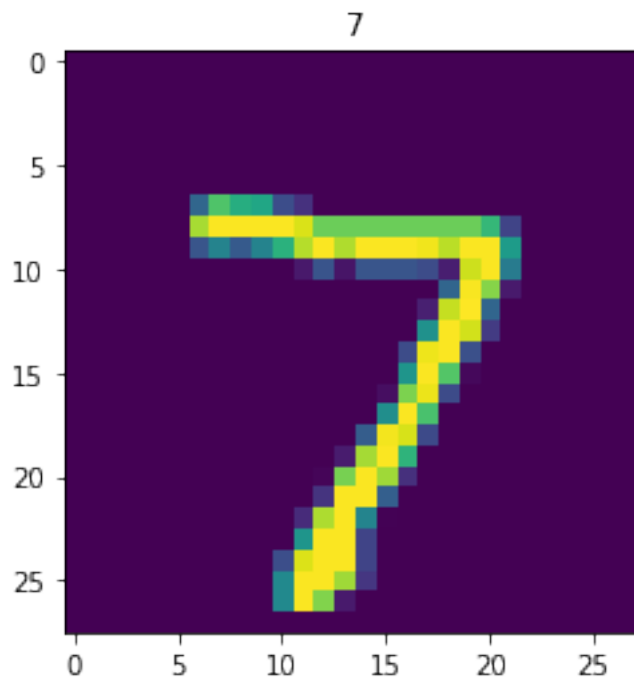
Training labels shape: (60000,)

Testing Dataset shape: (10000, 784)

Testing labels shape: (10000,)

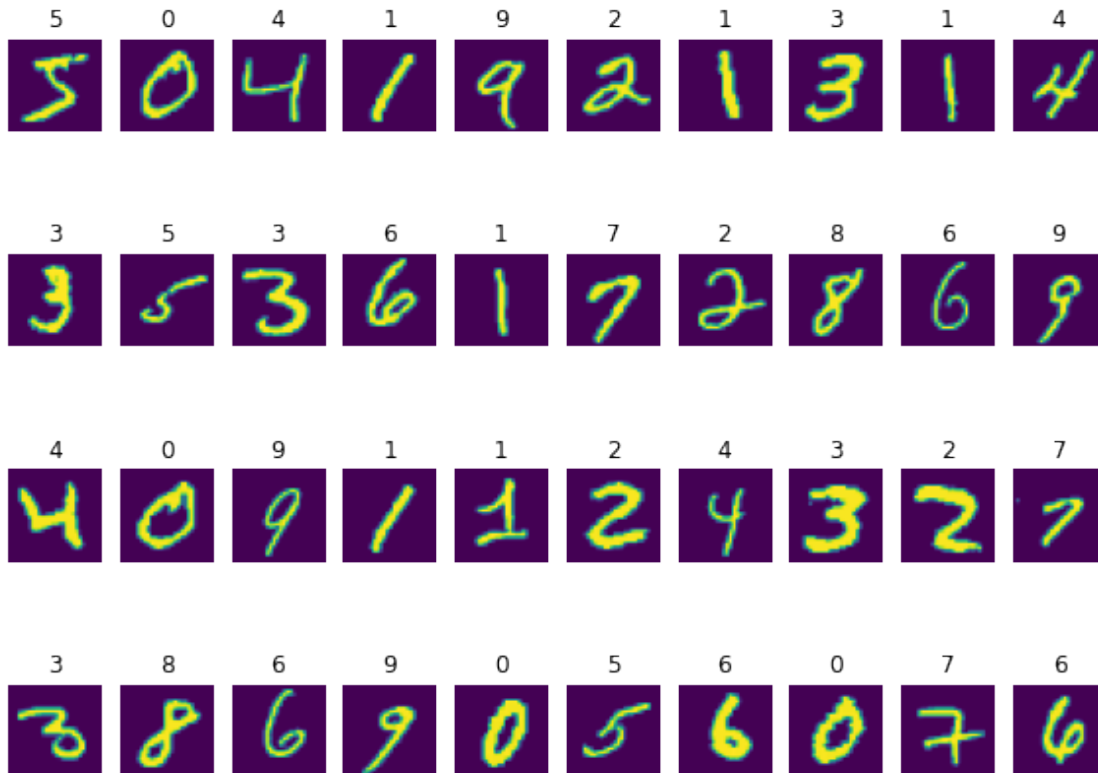
Unique Labels: [0 1 2 3 4 5 6 7 8 9]

```
[5]: # Visualizing Dataset one example.  
plt.imshow(X_test[0].reshape(28,28))  
plt.title(y_test[0]);
```



Visualizing first 40 images from training dataset and their labels. It shows some examples of style of each digit written by hand.

```
[6]: plt.figure(figsize=(10,80))  
for i in range(0,40):  
    plt.subplot(40,10,i+1)  
    img1=X_train[i].reshape(28,28)  
    plt.imshow(img1)  
    plt.title(y_train[i])  
    plt.axis('off')  
plt.show()
```



```
[7]: #function for loading dataset.
def data_resizing(old_images,img_size=(32,32)):
    old_images=old_images.reshape(old_images.shape[0],28,28)
    all_images=[]
    for img in old_images:
        image=cv2.resize(img,img_size,interpolation=cv2.INTER_CUBIC)# resizing
        →the image to 224*224*3
        all_images.append(image)
    all_images=np.array(all_images)
    return all_images.reshape(all_images.shape[0],all_images.
        →shape[1],all_images.shape[2],1)
```

```
[ ]:
```

```
[8]: train_X=data_resizing(X_train,img_size=(32,32))
test_X=data_resizing(X_test,img_size=(32,32))

print('X_train shape:',train_X.shape)
print('X_test shape:',test_X.shape)
```

```
X_train shape: (60000, 32, 32, 1)
```

```
X_test shape: (10000, 32, 32, 1)
```

```
[ ]:
```

```
[9]: train_y=to_categorical(y_train,num_classes)
test_y=to_categorical(y_test,num_classes)
```

```
print('y_train shape:',train_y.shape)
print('y_test shape:',test_y.shape)
```

```
y_train shape: (60000, 10)
```

```
y_test shape: (10000, 10)
```

0.2 Classification Method:

0.2.1 VGG:

```
[10]: def VGG_16(num_classes,img_size=(32,32,1)):
        initial_model: Model = VGG16(include_top=False,
        weights=None,input_shape=img_size)

        x = layers.Flatten()(initial_model.output)
        x = layers.Dense(256, activation='relu')(x)
        predictions = layers.Dense(num_classes, activation='softmax')(x)

        model = Model(initial_model.input, predictions)
        model.compile(loss='categorical_crossentropy', optimizer='adam',
        metrics=['acc'])
        return model
```

```
[11]: VGG_model=VGG_16(num_classes,img_size=(32,32,1))
VGG_model.summary()
```

WARNING:tensorflow:From C:\Users\afaq.ahmad\.conda\envs\tf_gpu\lib\site-packages\keras\backend\tensorflow_backend.py:4070: The name tf.nn.max_pool is deprecated. Please use tf.nn.max_pool2d instead.

Model: "model_1"

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	(None, 32, 32, 1)	0
block1_conv1 (Conv2D)	(None, 32, 32, 64)	640
block1_conv2 (Conv2D)	(None, 32, 32, 64)	36928
block1_pool (MaxPooling2D)	(None, 16, 16, 64)	0

block2_conv1 (Conv2D)	(None, 16, 16, 128)	73856
block2_conv2 (Conv2D)	(None, 16, 16, 128)	147584
block2_pool (MaxPooling2D)	(None, 8, 8, 128)	0
block3_conv1 (Conv2D)	(None, 8, 8, 256)	295168
block3_conv2 (Conv2D)	(None, 8, 8, 256)	590080
block3_conv3 (Conv2D)	(None, 8, 8, 256)	590080
block3_pool (MaxPooling2D)	(None, 4, 4, 256)	0
block4_conv1 (Conv2D)	(None, 4, 4, 512)	1180160
block4_conv2 (Conv2D)	(None, 4, 4, 512)	2359808
block4_conv3 (Conv2D)	(None, 4, 4, 512)	2359808
block4_pool (MaxPooling2D)	(None, 2, 2, 512)	0
block5_conv1 (Conv2D)	(None, 2, 2, 512)	2359808
block5_conv2 (Conv2D)	(None, 2, 2, 512)	2359808
block5_conv3 (Conv2D)	(None, 2, 2, 512)	2359808
block5_pool (MaxPooling2D)	(None, 1, 1, 512)	0
flatten_1 (Flatten)	(None, 512)	0
dense_1 (Dense)	(None, 256)	131328
dense_2 (Dense)	(None, 10)	2570

=====
Total params: 14,847,434
Trainable params: 14,847,434
Non-trainable params: 0
=====

```
[12]: def get_callbacks_list():
        """Get callbacks for a model"""
        return [keras.callbacks.EarlyStopping(monitor='val_acc',patience=10),
                keras.callbacks.ReduceLROnPlateau(monitor='val_loss',factor=0.
        ↪2,patience=5)]
```

[47]:

```
[13]: history_vgg = VGG_model.fit(train_X,train_y,batch_size=256,epochs =  
    ↪200,callbacks=get_callbacks_list(),validation_split=0.1)
```

WARNING:tensorflow:From C:\Users\afaq.ahmad\.conda\envs\tf_gpu\lib\site-packages\keras\backend\tensorflow_backend.py:422: The name tf.global_variables is deprecated. Please use tf.compat.v1.global_variables instead.

Train on 54000 samples, validate on 6000 samples

Epoch 1/200

54000/54000 [=====] - 57s 1ms/step - loss: 0.8895 -
acc: 0.7351 - val_loss: 0.0739 - val_acc: 0.9798

Epoch 2/200

54000/54000 [=====] - 49s 915us/step - loss: 0.0739 -
acc: 0.9797 - val_loss: 0.0520 - val_acc: 0.9848

Epoch 3/200

54000/54000 [=====] - 50s 918us/step - loss: 0.0511 -
acc: 0.9861 - val_loss: 0.0538 - val_acc: 0.9855

Epoch 4/200

54000/54000 [=====] - 50s 919us/step - loss: 0.0410 -
acc: 0.9890 - val_loss: 0.0321 - val_acc: 0.9907

Epoch 5/200

54000/54000 [=====] - 50s 920us/step - loss: 0.0373 -
acc: 0.9902 - val_loss: 0.0528 - val_acc: 0.9882

Epoch 6/200

54000/54000 [=====] - 50s 923us/step - loss: 0.0373 -
acc: 0.9910 - val_loss: 0.0378 - val_acc: 0.9908

Epoch 7/200

54000/54000 [=====] - 50s 922us/step - loss: 0.0293 -
acc: 0.9925 - val_loss: 0.0397 - val_acc: 0.9913

Epoch 8/200

54000/54000 [=====] - 50s 922us/step - loss: 0.0286 -
acc: 0.9922 - val_loss: 0.0447 - val_acc: 0.9905

Epoch 9/200

54000/54000 [=====] - 50s 922us/step - loss: 0.0243 -
acc: 0.9940 - val_loss: 0.0423 - val_acc: 0.9925

Epoch 10/200

54000/54000 [=====] - 50s 924us/step - loss: 0.0086 -
acc: 0.9977 - val_loss: 0.0232 - val_acc: 0.9950

Epoch 11/200

54000/54000 [=====] - 50s 924us/step - loss: 0.0029 -
acc: 0.9993 - val_loss: 0.0311 - val_acc: 0.9952

Epoch 12/200

54000/54000 [=====] - 50s 923us/step - loss: 0.0017 -
acc: 0.9996 - val_loss: 0.0327 - val_acc: 0.9948

Epoch 13/200

```

54000/54000 [=====] - 50s 923us/step - loss: 7.5626e-04
- acc: 0.9998 - val_loss: 0.0368 - val_acc: 0.9950
Epoch 14/200
54000/54000 [=====] - 50s 924us/step - loss: 5.1294e-04
- acc: 0.9999 - val_loss: 0.0401 - val_acc: 0.9952
Epoch 15/200
54000/54000 [=====] - 50s 923us/step - loss: 3.3193e-04
- acc: 0.9999 - val_loss: 0.0426 - val_acc: 0.9950
Epoch 16/200
54000/54000 [=====] - 50s 923us/step - loss: 1.5079e-04
- acc: 1.0000 - val_loss: 0.0441 - val_acc: 0.9947
Epoch 17/200
54000/54000 [=====] - 50s 923us/step - loss: 1.0372e-04
- acc: 1.0000 - val_loss: 0.0450 - val_acc: 0.9950
Epoch 18/200
54000/54000 [=====] - 50s 925us/step - loss: 7.9750e-05
- acc: 1.0000 - val_loss: 0.0461 - val_acc: 0.9950
Epoch 19/200
54000/54000 [=====] - 50s 924us/step - loss: 6.8676e-05
- acc: 1.0000 - val_loss: 0.0473 - val_acc: 0.9948
Epoch 20/200
54000/54000 [=====] - 50s 924us/step - loss: 5.8440e-05
- acc: 1.0000 - val_loss: 0.0484 - val_acc: 0.9948
Epoch 21/200
54000/54000 [=====] - 50s 927us/step - loss: 4.6742e-05
- acc: 1.0000 - val_loss: 0.0487 - val_acc: 0.9948

```

[1]:

[1]: 1050

```

[48]: def draw_training_info_plots(_history):
        """Draw loss graphs at the training and validation stage"""
        acc = _history.history['acc']
        val_acc = _history.history['val_acc']
        loss = _history.history['loss']
        val_loss = _history.history['val_loss']

        epochs_plot = range(1, len(acc) + 1)
        plt.plot(epochs_plot, acc, 'b--', label='Training acc')
        plt.plot(epochs_plot, val_acc, 'b', label='Validation acc')
        plt.title('Training and validation accuracy')
        plt.xlabel('epoch')
        plt.ylabel('accuracy')
        plt.legend()
        plt.figure()

```

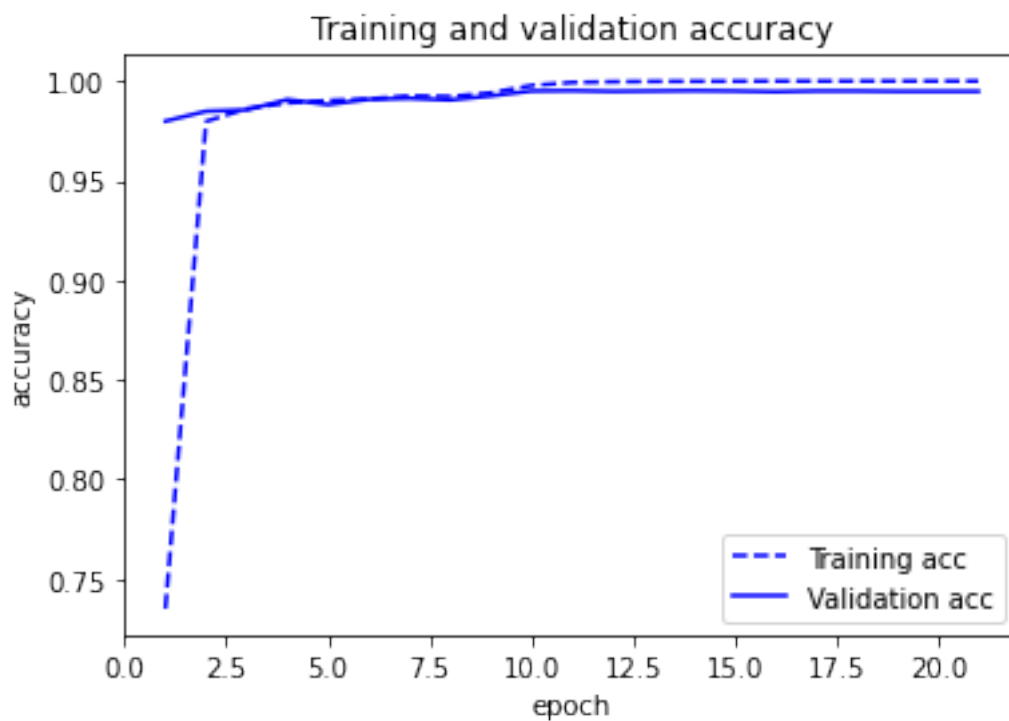
```

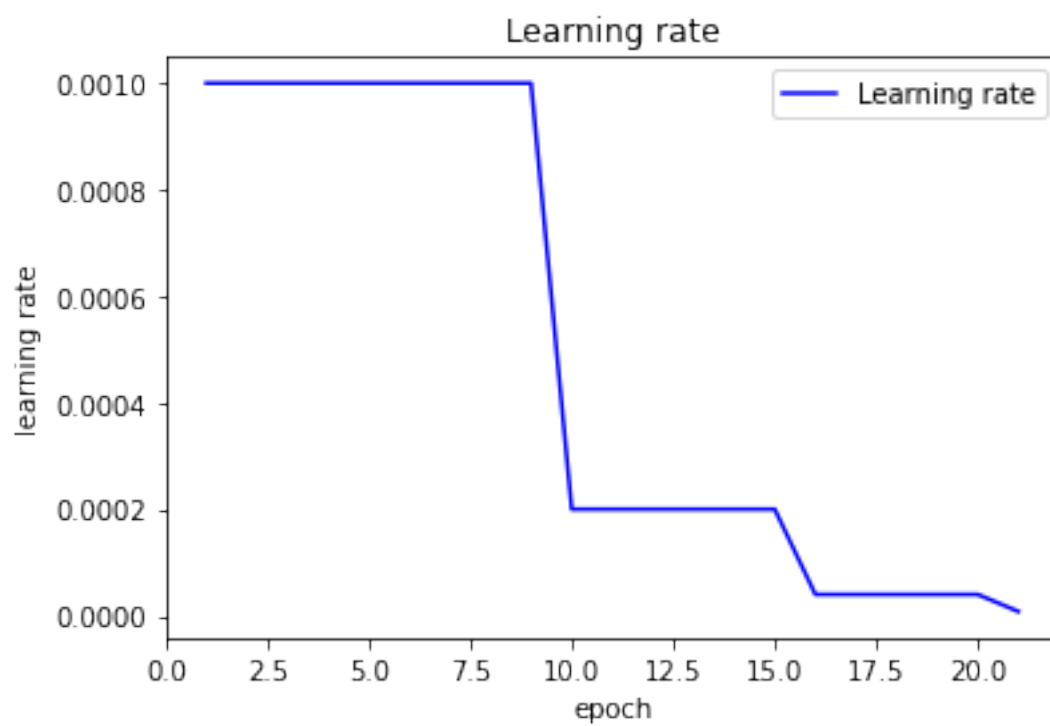
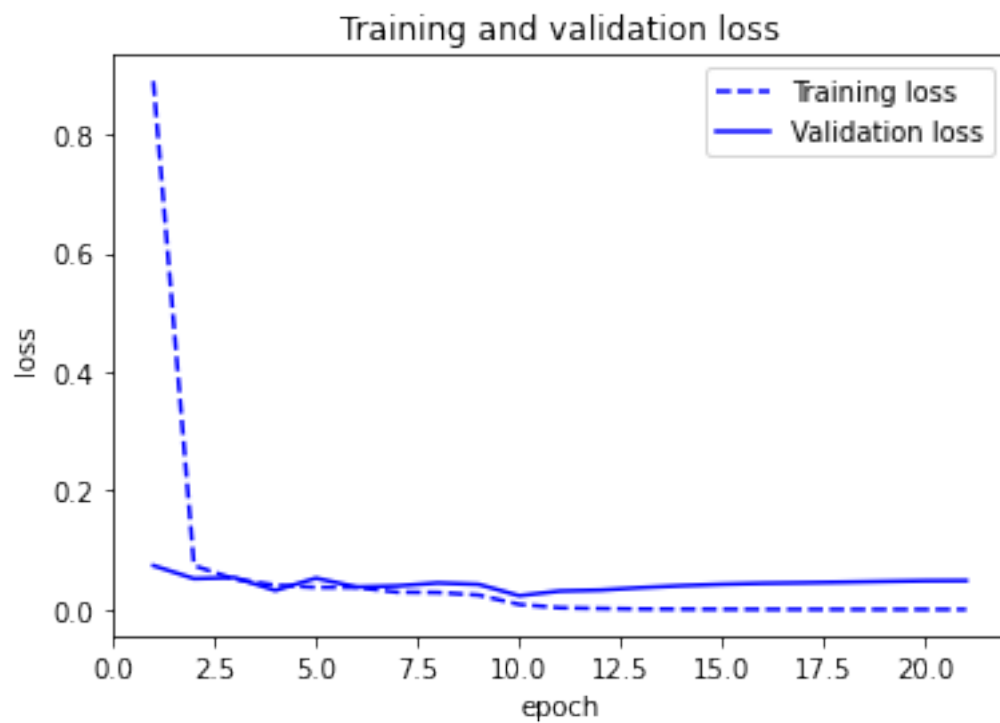
plt.plot(epochs_plot, loss, 'b--', label='Training loss')
plt.plot(epochs_plot, val_loss, 'b', label='Validation loss')
plt.title('Training and validation loss')
plt.xlabel('epoch')
plt.ylabel('loss')
plt.legend()
plt.show()

if 'lr' in _history.history:
    learning_rate = _history.history['lr']
    plt.plot(epochs_plot, learning_rate, 'b', label='Learning rate')
    plt.title('Learning rate')
    plt.xlabel('epoch')
    plt.ylabel('learning rate')
    plt.legend()
    plt.show()
return

draw_training_info_plots(history_vgg)

```





```
[37]: print('Accuracy:',VGG_model.evaluate(test_X,test_y,verbose=0)[1])
```

Accuracy: 0.9943000078201294

Classification Score and Confusion Metric

```
[38]: predictions = VGG_model.predict(test_X)

from sklearn.metrics import classification_report
print("EVALUATION ON TESTING DATA")
print(classification_report(y_test, np.argmax(predictions,axis=1)))
```

EVALUATION ON TESTING DATA

	precision	recall	f1-score	support
0	0.99	0.99	0.99	980
1	1.00	1.00	1.00	1135
2	1.00	1.00	1.00	1032
3	0.99	1.00	0.99	1010
4	0.99	0.99	0.99	982
5	0.99	0.99	0.99	892
6	0.99	0.99	0.99	958
7	0.99	0.99	0.99	1028
8	1.00	1.00	1.00	974
9	0.99	0.99	0.99	1009
accuracy				0.99 10000
macro avg	0.99	0.99	0.99	10000
weighted avg	0.99	0.99	0.99	10000

```
[39]: from sklearn.metrics import confusion_matrix
import pandas as pd
print ("Confusion matrix")
pd.DataFrame(confusion_matrix(y_test,np.argmax(predictions,axis=1)),columns=np.
    ↳unique(y_test),index=np.unique(y_test))
```

Confusion matrix

```
[39]:
```

	0	1	2	3	4	5	6	7	8	9
0	975	0	1	0	0	0	2	1	1	0
1	0	1131	2	1	0	0	1	0	0	0
2	1	0	1029	1	0	0	0	1	0	0
3	0	0	0	1007	0	3	0	0	0	0
4	0	1	0	0	973	0	2	0	0	6
5	0	0	0	4	0	887	1	0	0	0
6	2	1	0	0	0	2	952	0	1	0
7	0	1	2	1	0	0	0	1022	0	2

8	1	0	0	1	0	1	0	0	971	0
9	1	0	0	0	5	2	0	4	1	996

```
[49]: VGG_model.save('VGG_model_mnist.h5')
```

0.2.2 Resnet:

```
[44]: def Resnet_50(num_classes, img_size=(32,32,1)):
        initial_model: Model = ResNet50(include_top=False,
        ↳weights=None, input_shape=img_size)

        x = layers.Flatten()(initial_model.output)
        x = layers.Dense(256, activation='relu')(x)
        predictions = layers.Dense(num_classes, activation='softmax')(x)

        model = Model(initial_model.input, predictions)
        model.compile(loss='categorical_crossentropy', optimizer='adam',
        ↳metrics=['acc'])
        return model
```

```
[45]: Resnet_model=Resnet_50(num_classes, img_size=(32,32,1))
Resnet_model.summary()
```

Model: "model_2"

Layer (type)	Output Shape	Param #	Connected to
input_2 (InputLayer)	(None, 32, 32, 1)	0	
conv1_pad (ZeroPadding2D)	(None, 38, 38, 1)	0	input_2[0][0]
conv1_conv (Conv2D)	(None, 16, 16, 64)	3200	conv1_pad[0][0]
conv1_bn (BatchNormalization)	(None, 16, 16, 64)	256	conv1_conv[0][0]
conv1_relu (Activation)	(None, 16, 16, 64)	0	conv1_bn[0][0]
pool1_pad (ZeroPadding2D)	(None, 18, 18, 64)	0	conv1_relu[0][0]

pool1_pool (MaxPooling2D)	(None, 8, 8, 64)	0	pool1_pad[0][0]
conv2_block1_1_conv (Conv2D)	(None, 8, 8, 64)	4160	pool1_pool[0][0]
conv2_block1_1_bn (BatchNormali	(None, 8, 8, 64)	256	conv2_block1_1_conv[0][0]
conv2_block1_1_relu (Activation	(None, 8, 8, 64)	0	conv2_block1_1_bn[0][0]
conv2_block1_2_conv (Conv2D)	(None, 8, 8, 64)	36928	conv2_block1_1_relu[0][0]
conv2_block1_2_bn (BatchNormali	(None, 8, 8, 64)	256	conv2_block1_2_conv[0][0]
conv2_block1_2_relu (Activation	(None, 8, 8, 64)	0	conv2_block1_2_bn[0][0]
conv2_block1_0_conv (Conv2D)	(None, 8, 8, 256)	16640	conv2_block1_2_relu[0][0]
conv2_block1_3_conv (Conv2D)	(None, 8, 8, 256)	16640	conv2_block1_0_conv[0][0]
conv2_block1_0_bn (BatchNormali	(None, 8, 8, 256)	1024	conv2_block1_3_conv[0][0]
conv2_block1_3_bn (BatchNormali	(None, 8, 8, 256)	1024	conv2_block1_0_bn[0][0]
conv2_block1_add (Add)	(None, 8, 8, 256)	0	conv2_block1_3_bn[0][0]

```

-----
conv2_block1_out (Activation)      (None, 8, 8, 256)      0
conv2_block1_add[0][0]
-----
conv2_block2_1_conv (Conv2D)       (None, 8, 8, 64)      16448
conv2_block1_out[0][0]
-----
conv2_block2_1_bn (BatchNormaliz (None, 8, 8, 64)      256
conv2_block2_1_conv[0][0]
-----
conv2_block2_1_relu (Activation)    (None, 8, 8, 64)      0
conv2_block2_1_bn[0][0]
-----
conv2_block2_2_conv (Conv2D)       (None, 8, 8, 64)      36928
conv2_block2_1_relu[0][0]
-----
conv2_block2_2_bn (BatchNormaliz (None, 8, 8, 64)      256
conv2_block2_2_conv[0][0]
-----
conv2_block2_2_relu (Activation)    (None, 8, 8, 64)      0
conv2_block2_2_bn[0][0]
-----
conv2_block2_3_conv (Conv2D)       (None, 8, 8, 256)     16640
conv2_block2_2_relu[0][0]
-----
conv2_block2_3_bn (BatchNormaliz (None, 8, 8, 256)     1024
conv2_block2_3_conv[0][0]
-----
conv2_block2_add (Add)              (None, 8, 8, 256)      0
conv2_block1_out[0][0]
conv2_block2_3_bn[0][0]
-----
conv2_block2_out (Activation)       (None, 8, 8, 256)      0
conv2_block2_add[0][0]
-----
conv2_block3_1_conv (Conv2D)       (None, 8, 8, 64)      16448

```

conv2_block2_out[0][0]

conv2_block3_1_bn (BatchNormali (None, 8, 8, 64) 256
conv2_block3_1_conv[0][0]

conv2_block3_1_relu (Activation (None, 8, 8, 64) 0
conv2_block3_1_bn[0][0]

conv2_block3_2_conv (Conv2D) (None, 8, 8, 64) 36928
conv2_block3_1_relu[0][0]

conv2_block3_2_bn (BatchNormali (None, 8, 8, 64) 256
conv2_block3_2_conv[0][0]

conv2_block3_2_relu (Activation (None, 8, 8, 64) 0
conv2_block3_2_bn[0][0]

conv2_block3_3_conv (Conv2D) (None, 8, 8, 256) 16640
conv2_block3_2_relu[0][0]

conv2_block3_3_bn (BatchNormali (None, 8, 8, 256) 1024
conv2_block3_3_conv[0][0]

conv2_block3_add (Add) (None, 8, 8, 256) 0
conv2_block2_out[0][0]
conv2_block3_3_bn[0][0]

conv2_block3_out (Activation) (None, 8, 8, 256) 0
conv2_block3_add[0][0]

conv3_block1_1_conv (Conv2D) (None, 4, 4, 128) 32896
conv2_block3_out[0][0]

conv3_block1_1_bn (BatchNormali (None, 4, 4, 128) 512
conv3_block1_1_conv[0][0]

conv3_block1_1_relu (Activation (None, 4, 4, 128)	0
conv3_block1_1_bn[0][0]	

conv3_block1_2_conv (Conv2D) (None, 4, 4, 128)	147584
conv3_block1_1_relu[0][0]	

conv3_block1_2_bn (BatchNormali (None, 4, 4, 128)	512
conv3_block1_2_conv[0][0]	

conv3_block1_2_relu (Activation (None, 4, 4, 128)	0
conv3_block1_2_bn[0][0]	

conv3_block1_0_conv (Conv2D) (None, 4, 4, 512)	131584
conv2_block3_out[0][0]	

conv3_block1_3_conv (Conv2D) (None, 4, 4, 512)	66048
conv3_block1_2_relu[0][0]	

conv3_block1_0_bn (BatchNormali (None, 4, 4, 512)	2048
conv3_block1_0_conv[0][0]	

conv3_block1_3_bn (BatchNormali (None, 4, 4, 512)	2048
conv3_block1_3_conv[0][0]	

conv3_block1_add (Add) (None, 4, 4, 512)	0
conv3_block1_0_bn[0][0]	
conv3_block1_3_bn[0][0]	

conv3_block1_out (Activation) (None, 4, 4, 512)	0
conv3_block1_add[0][0]	

conv3_block2_1_conv (Conv2D) (None, 4, 4, 128)	65664
conv3_block1_out[0][0]	

conv3_block2_1_bn (BatchNormali (None, 4, 4, 128)	512
conv3_block2_1_conv[0][0]	

```

-----
conv3_block2_1_relu (Activation (None, 4, 4, 128))    0
conv3_block2_1_bn[0][0]
-----

-----
conv3_block2_2_conv (Conv2D)      (None, 4, 4, 128)    147584
conv3_block2_1_relu[0][0]
-----

-----
conv3_block2_2_bn (BatchNormali (None, 4, 4, 128)    512
conv3_block2_2_conv[0][0]
-----

-----
conv3_block2_2_relu (Activation (None, 4, 4, 128))    0
conv3_block2_2_bn[0][0]
-----

-----
conv3_block2_3_conv (Conv2D)      (None, 4, 4, 512)    66048
conv3_block2_2_relu[0][0]
-----

-----
conv3_block2_3_bn (BatchNormali (None, 4, 4, 512)    2048
conv3_block2_3_conv[0][0]
-----

-----
conv3_block2_add (Add)              (None, 4, 4, 512)    0
conv3_block1_out[0][0]
conv3_block2_3_bn[0][0]
-----

-----
conv3_block2_out (Activation)      (None, 4, 4, 512)    0
conv3_block2_add[0][0]
-----

-----
conv3_block3_1_conv (Conv2D)      (None, 4, 4, 128)    65664
conv3_block2_out[0][0]
-----

-----
conv3_block3_1_bn (BatchNormali (None, 4, 4, 128)    512
conv3_block3_1_conv[0][0]
-----

-----
conv3_block3_1_relu (Activation (None, 4, 4, 128))    0
conv3_block3_1_bn[0][0]
-----

-----
conv3_block3_2_conv (Conv2D)      (None, 4, 4, 128)    147584
conv3_block3_1_relu[0][0]

```



```

-----
-----
conv3_block3_2_bn (BatchNormali (None, 4, 4, 128)    512
conv3_block3_2_conv[0][0]
-----
-----
conv3_block3_2_relu (Activation (None, 4, 4, 128)    0
conv3_block3_2_bn[0][0]
-----
-----
conv3_block3_3_conv (Conv2D)      (None, 4, 4, 512)    66048
conv3_block3_2_relu[0][0]
-----
-----
conv3_block3_3_bn (BatchNormali (None, 4, 4, 512)    2048
conv3_block3_3_conv[0][0]
-----
-----
conv3_block3_add (Add)              (None, 4, 4, 512)    0
conv3_block2_out[0][0]
conv3_block3_3_bn[0][0]
-----
-----
conv3_block3_out (Activation)      (None, 4, 4, 512)    0
conv3_block3_add[0][0]
-----
-----
conv3_block4_1_conv (Conv2D)      (None, 4, 4, 128)    65664
conv3_block3_out[0][0]
-----
-----
conv3_block4_1_bn (BatchNormali (None, 4, 4, 128)    512
conv3_block4_1_conv[0][0]
-----
-----
conv3_block4_1_relu (Activation (None, 4, 4, 128)    0
conv3_block4_1_bn[0][0]
-----
-----
conv3_block4_2_conv (Conv2D)      (None, 4, 4, 128)    147584
conv3_block4_1_relu[0][0]
-----
-----
conv3_block4_2_bn (BatchNormali (None, 4, 4, 128)    512
conv3_block4_2_conv[0][0]
-----
-----
conv3_block4_2_relu (Activation (None, 4, 4, 128)    0

```

conv3_block4_2_bn[0][0]

conv3_block4_3_conv (Conv2D) (None, 4, 4, 512) 66048
conv3_block4_2_relu[0][0]

conv3_block4_3_bn (BatchNormali (None, 4, 4, 512) 2048
conv3_block4_3_conv[0][0]

conv3_block4_add (Add) (None, 4, 4, 512) 0
conv3_block3_out[0][0]
conv3_block4_3_bn[0][0]

conv3_block4_out (Activation) (None, 4, 4, 512) 0
conv3_block4_add[0][0]

conv4_block1_1_conv (Conv2D) (None, 2, 2, 256) 131328
conv3_block4_out[0][0]

conv4_block1_1_bn (BatchNormali (None, 2, 2, 256) 1024
conv4_block1_1_conv[0][0]

conv4_block1_1_relu (Activation (None, 2, 2, 256) 0
conv4_block1_1_bn[0][0]

conv4_block1_2_conv (Conv2D) (None, 2, 2, 256) 590080
conv4_block1_1_relu[0][0]

conv4_block1_2_bn (BatchNormali (None, 2, 2, 256) 1024
conv4_block1_2_conv[0][0]

conv4_block1_2_relu (Activation (None, 2, 2, 256) 0
conv4_block1_2_bn[0][0]

conv4_block1_0_conv (Conv2D) (None, 2, 2, 1024) 525312
conv3_block4_out[0][0]


```
conv4_block1_3_conv (Conv2D)      (None, 2, 2, 1024)    263168
conv4_block1_2_relu[0][0]
```

```
-----
conv4_block1_0_bn (BatchNormaliz (None, 2, 2, 1024)    4096
conv4_block1_0_conv[0][0]
```

```
-----
conv4_block1_3_bn (BatchNormaliz (None, 2, 2, 1024)    4096
conv4_block1_3_conv[0][0]
```

```
-----
conv4_block1_add (Add)              (None, 2, 2, 1024)    0
conv4_block1_0_bn[0][0]
conv4_block1_3_bn[0][0]
```

```
-----
conv4_block1_out (Activation)        (None, 2, 2, 1024)    0
conv4_block1_add[0][0]
```

```
-----
conv4_block2_1_conv (Conv2D)      (None, 2, 2, 256)     262400
conv4_block1_out[0][0]
```

```
-----
conv4_block2_1_bn (BatchNormaliz (None, 2, 2, 256)     1024
conv4_block2_1_conv[0][0]
```

```
-----
conv4_block2_1_relu (Activation)    (None, 2, 2, 256)     0
conv4_block2_1_bn[0][0]
```

```
-----
conv4_block2_2_conv (Conv2D)      (None, 2, 2, 256)     590080
conv4_block2_1_relu[0][0]
```

```
-----
conv4_block2_2_bn (BatchNormaliz (None, 2, 2, 256)     1024
conv4_block2_2_conv[0][0]
```

```
-----
conv4_block2_2_relu (Activation)    (None, 2, 2, 256)     0
conv4_block2_2_bn[0][0]
```

```
-----
conv4_block2_3_conv (Conv2D)      (None, 2, 2, 1024)    263168
conv4_block2_2_relu[0][0]
```

```

-----
conv4_block2_3_bn (BatchNormali (None, 2, 2, 1024) 4096
conv4_block2_3_conv[0][0]
-----

conv4_block2_add (Add) (None, 2, 2, 1024) 0
conv4_block1_out[0][0]
conv4_block2_3_bn[0][0]
-----

conv4_block2_out (Activation) (None, 2, 2, 1024) 0
conv4_block2_add[0][0]
-----

conv4_block3_1_conv (Conv2D) (None, 2, 2, 256) 262400
conv4_block2_out[0][0]
-----

conv4_block3_1_bn (BatchNormali (None, 2, 2, 256) 1024
conv4_block3_1_conv[0][0]
-----

conv4_block3_1_relu (Activation (None, 2, 2, 256) 0
conv4_block3_1_bn[0][0]
-----

conv4_block3_2_conv (Conv2D) (None, 2, 2, 256) 590080
conv4_block3_1_relu[0][0]
-----

conv4_block3_2_bn (BatchNormali (None, 2, 2, 256) 1024
conv4_block3_2_conv[0][0]
-----

conv4_block3_2_relu (Activation (None, 2, 2, 256) 0
conv4_block3_2_bn[0][0]
-----

conv4_block3_3_conv (Conv2D) (None, 2, 2, 1024) 263168
conv4_block3_2_relu[0][0]
-----

conv4_block3_3_bn (BatchNormali (None, 2, 2, 1024) 4096
conv4_block3_3_conv[0][0]
-----

conv4_block3_add (Add) (None, 2, 2, 1024) 0
conv4_block2_out[0][0]

```

conv4_block3_3_bn[0][0]

conv4_block3_out (Activation) (None, 2, 2, 1024) 0
conv4_block3_add[0][0]

conv4_block4_1_conv (Conv2D) (None, 2, 2, 256) 262400
conv4_block3_out[0][0]

conv4_block4_1_bn (BatchNormali (None, 2, 2, 256) 1024
conv4_block4_1_conv[0][0]

conv4_block4_1_relu (Activation (None, 2, 2, 256) 0
conv4_block4_1_bn[0][0]

conv4_block4_2_conv (Conv2D) (None, 2, 2, 256) 590080
conv4_block4_1_relu[0][0]

conv4_block4_2_bn (BatchNormali (None, 2, 2, 256) 1024
conv4_block4_2_conv[0][0]

conv4_block4_2_relu (Activation (None, 2, 2, 256) 0
conv4_block4_2_bn[0][0]

conv4_block4_3_conv (Conv2D) (None, 2, 2, 1024) 263168
conv4_block4_2_relu[0][0]

conv4_block4_3_bn (BatchNormali (None, 2, 2, 1024) 4096
conv4_block4_3_conv[0][0]

conv4_block4_add (Add) (None, 2, 2, 1024) 0
conv4_block3_out[0][0]
conv4_block4_3_bn[0][0]

conv4_block4_out (Activation) (None, 2, 2, 1024) 0
conv4_block4_add[0][0]

conv4_block5_1_conv (Conv2D)	(None, 2, 2, 256)	262400
conv4_block4_out[0][0]		

conv4_block5_1_bn (BatchNormali	(None, 2, 2, 256)	1024
conv4_block5_1_conv[0][0]		

conv4_block5_1_relu (Activation	(None, 2, 2, 256)	0
conv4_block5_1_bn[0][0]		

conv4_block5_2_conv (Conv2D)	(None, 2, 2, 256)	590080
conv4_block5_1_relu[0][0]		

conv4_block5_2_bn (BatchNormali	(None, 2, 2, 256)	1024
conv4_block5_2_conv[0][0]		

conv4_block5_2_relu (Activation	(None, 2, 2, 256)	0
conv4_block5_2_bn[0][0]		

conv4_block5_3_conv (Conv2D)	(None, 2, 2, 1024)	263168
conv4_block5_2_relu[0][0]		

conv4_block5_3_bn (BatchNormali	(None, 2, 2, 1024)	4096
conv4_block5_3_conv[0][0]		

conv4_block5_add (Add)	(None, 2, 2, 1024)	0
conv4_block4_out[0][0]		
conv4_block5_3_bn[0][0]		

conv4_block5_out (Activation)	(None, 2, 2, 1024)	0
conv4_block5_add[0][0]		

conv4_block6_1_conv (Conv2D)	(None, 2, 2, 256)	262400
conv4_block5_out[0][0]		

conv4_block6_1_bn (BatchNormali	(None, 2, 2, 256)	1024
conv4_block6_1_conv[0][0]		

```

-----
conv4_block6_1_relu (Activation (None, 2, 2, 256)    0
conv4_block6_1_bn[0][0]
-----

-----
conv4_block6_2_conv (Conv2D)      (None, 2, 2, 256)    590080
conv4_block6_1_relu[0][0]
-----

-----
conv4_block6_2_bn (BatchNormali (None, 2, 2, 256)    1024
conv4_block6_2_conv[0][0]
-----

-----
conv4_block6_2_relu (Activation (None, 2, 2, 256)    0
conv4_block6_2_bn[0][0]
-----

-----
conv4_block6_3_conv (Conv2D)      (None, 2, 2, 1024)   263168
conv4_block6_2_relu[0][0]
-----

-----
conv4_block6_3_bn (BatchNormali (None, 2, 2, 1024)   4096
conv4_block6_3_conv[0][0]
-----

-----
conv4_block6_add (Add)             (None, 2, 2, 1024)   0
conv4_block5_out[0][0]
conv4_block6_3_bn[0][0]
-----

-----
conv4_block6_out (Activation)      (None, 2, 2, 1024)   0
conv4_block6_add[0][0]
-----

-----
conv5_block1_1_conv (Conv2D)      (None, 1, 1, 512)    524800
conv4_block6_out[0][0]
-----

-----
conv5_block1_1_bn (BatchNormali (None, 1, 1, 512)    2048
conv5_block1_1_conv[0][0]
-----

-----
conv5_block1_1_relu (Activation (None, 1, 1, 512)    0
conv5_block1_1_bn[0][0]
-----

-----
conv5_block1_2_conv (Conv2D)      (None, 1, 1, 512)    2359808
conv5_block1_1_relu[0][0]

```

```

-----
conv5_block1_2_bn (BatchNormali (None, 1, 1, 512)    2048
conv5_block1_2_conv[0][0]
-----

conv5_block1_2_relu (Activation (None, 1, 1, 512)    0
conv5_block1_2_bn[0][0]
-----

conv5_block1_0_conv (Conv2D)      (None, 1, 1, 2048)    2099200
conv4_block6_out[0][0]
-----

conv5_block1_3_conv (Conv2D)      (None, 1, 1, 2048)    1050624
conv5_block1_2_relu[0][0]
-----

conv5_block1_0_bn (BatchNormali (None, 1, 1, 2048)    8192
conv5_block1_0_conv[0][0]
-----

conv5_block1_3_bn (BatchNormali (None, 1, 1, 2048)    8192
conv5_block1_3_conv[0][0]
-----

conv5_block1_add (Add)            (None, 1, 1, 2048)    0
conv5_block1_0_bn[0][0]
conv5_block1_3_bn[0][0]
-----

conv5_block1_out (Activation)      (None, 1, 1, 2048)    0
conv5_block1_add[0][0]
-----

conv5_block2_1_conv (Conv2D)      (None, 1, 1, 512)    1049088
conv5_block1_out[0][0]
-----

conv5_block2_1_bn (BatchNormali (None, 1, 1, 512)    2048
conv5_block2_1_conv[0][0]
-----

conv5_block2_1_relu (Activation (None, 1, 1, 512)    0
conv5_block2_1_bn[0][0]
-----

conv5_block2_2_conv (Conv2D)      (None, 1, 1, 512)    2359808

```



```

conv5_block2_1_relu[0][0]
-----

-----
conv5_block2_2_bn (BatchNormali (None, 1, 1, 512)    2048
conv5_block2_2_conv[0][0]
-----

-----
conv5_block2_2_relu (Activation (None, 1, 1, 512)    0
conv5_block2_2_bn[0][0]
-----

-----
conv5_block2_3_conv (Conv2D)      (None, 1, 1, 2048)    1050624
conv5_block2_2_relu[0][0]
-----

-----
conv5_block2_3_bn (BatchNormali (None, 1, 1, 2048)    8192
conv5_block2_3_conv[0][0]
-----

-----
conv5_block2_add (Add)              (None, 1, 1, 2048)    0
conv5_block1_out[0][0]
conv5_block2_3_bn[0][0]
-----

-----
conv5_block2_out (Activation)      (None, 1, 1, 2048)    0
conv5_block2_add[0][0]
-----

-----
conv5_block3_1_conv (Conv2D)      (None, 1, 1, 512)    1049088
conv5_block2_out[0][0]
-----

-----
conv5_block3_1_bn (BatchNormali (None, 1, 1, 512)    2048
conv5_block3_1_conv[0][0]
-----

-----
conv5_block3_1_relu (Activation (None, 1, 1, 512)    0
conv5_block3_1_bn[0][0]
-----

-----
conv5_block3_2_conv (Conv2D)      (None, 1, 1, 512)    2359808
conv5_block3_1_relu[0][0]
-----

-----
conv5_block3_2_bn (BatchNormali (None, 1, 1, 512)    2048
conv5_block3_2_conv[0][0]
-----

-----

```

```

conv5_block3_2_relu (Activation (None, 1, 1, 512)    0
conv5_block3_2_bn[0][0]

-----

conv5_block3_3_conv (Conv2D)      (None, 1, 1, 2048)  1050624
conv5_block3_2_relu[0][0]

-----

conv5_block3_3_bn (BatchNormaliz (None, 1, 1, 2048)  8192
conv5_block3_3_conv[0][0]

-----

conv5_block3_add (Add)            (None, 1, 1, 2048)  0
conv5_block2_out[0][0]
conv5_block3_3_bn[0][0]

-----

conv5_block3_out (Activation)     (None, 1, 1, 2048)  0
conv5_block3_add[0][0]

-----

flatten_2 (Flatten)              (None, 2048)        0
conv5_block3_out[0][0]

-----

dense_3 (Dense)                  (None, 256)         524544    flatten_2[0][0]

-----

dense_4 (Dense)                  (None, 10)          2570      dense_3[0][0]
=====
Total params: 24,108,554
Trainable params: 24,055,434
Non-trainable params: 53,120
=====

```

```

[46]: def get_callbacks_list():
        """Get callbacks for a model"""
        return [keras.callbacks.EarlyStopping(monitor='val_acc',patience=10),
                keras.callbacks.ReduceLROnPlateau(monitor='val_loss',factor=0.
↪2,patience=5)]

```

```

[50]: history_resnet = Resnet_model.fit(train_X,train_y,batch_size=256,epochs =
↪200,callbacks=get_callbacks_list(),validation_split=0.1)

```

Train on 54000 samples, validate on 6000 samples
Epoch 1/200

54000/54000 [=====] - 69s 1ms/step - loss: 0.4080 -
acc: 0.8924 - val_loss: 0.5344 - val_acc: 0.8813
Epoch 2/200
54000/54000 [=====] - 61s 1ms/step - loss: 0.0749 -
acc: 0.9782 - val_loss: 0.0847 - val_acc: 0.9735
Epoch 3/200
54000/54000 [=====] - 61s 1ms/step - loss: 0.0468 -
acc: 0.9857 - val_loss: 0.0956 - val_acc: 0.9762
Epoch 4/200
54000/54000 [=====] - 61s 1ms/step - loss: 0.0322 -
acc: 0.9901 - val_loss: 0.0670 - val_acc: 0.9788
Epoch 5/200
54000/54000 [=====] - 61s 1ms/step - loss: 0.0317 -
acc: 0.9902 - val_loss: 0.0778 - val_acc: 0.9785
Epoch 6/200
54000/54000 [=====] - 61s 1ms/step - loss: 0.0247 -
acc: 0.9922 - val_loss: 0.0710 - val_acc: 0.9812
Epoch 7/200
54000/54000 [=====] - 61s 1ms/step - loss: 0.0229 -
acc: 0.9929 - val_loss: 0.0608 - val_acc: 0.9850
Epoch 8/200
54000/54000 [=====] - 61s 1ms/step - loss: 0.0184 -
acc: 0.9946 - val_loss: 0.0773 - val_acc: 0.9798
Epoch 9/200
54000/54000 [=====] - 61s 1ms/step - loss: 0.0221 -
acc: 0.9930 - val_loss: 0.0483 - val_acc: 0.9870
Epoch 10/200
54000/54000 [=====] - 61s 1ms/step - loss: 0.0203 -
acc: 0.9938 - val_loss: 0.0594 - val_acc: 0.9850
Epoch 11/200
54000/54000 [=====] - 62s 1ms/step - loss: 0.0173 -
acc: 0.9945 - val_loss: 0.0500 - val_acc: 0.9882
Epoch 12/200
54000/54000 [=====] - 61s 1ms/step - loss: 0.0141 -
acc: 0.9954 - val_loss: 0.0473 - val_acc: 0.9885
Epoch 13/200
54000/54000 [=====] - 61s 1ms/step - loss: 0.0139 -
acc: 0.9954 - val_loss: 0.0617 - val_acc: 0.9843
Epoch 14/200
54000/54000 [=====] - 62s 1ms/step - loss: 0.0151 -
acc: 0.9954 - val_loss: 0.0681 - val_acc: 0.9858
Epoch 15/200
54000/54000 [=====] - 62s 1ms/step - loss: 0.0186 -
acc: 0.9944 - val_loss: 0.0505 - val_acc: 0.9878
Epoch 16/200
54000/54000 [=====] - 62s 1ms/step - loss: 0.0136 -
acc: 0.9962 - val_loss: 0.0379 - val_acc: 0.9898
Epoch 17/200

54000/54000 [=====] - 63s 1ms/step - loss: 0.0143 -
acc: 0.9958 - val_loss: 0.0708 - val_acc: 0.9832
Epoch 18/200
54000/54000 [=====] - 63s 1ms/step - loss: 0.0118 -
acc: 0.9966 - val_loss: 0.0631 - val_acc: 0.9872
Epoch 19/200
54000/54000 [=====] - 63s 1ms/step - loss: 0.0155 -
acc: 0.9956 - val_loss: 0.0678 - val_acc: 0.9843
Epoch 20/200
54000/54000 [=====] - 63s 1ms/step - loss: 0.0163 -
acc: 0.9952 - val_loss: 0.0467 - val_acc: 0.9902
Epoch 21/200
54000/54000 [=====] - 63s 1ms/step - loss: 0.0159 -
acc: 0.9954 - val_loss: 0.0601 - val_acc: 0.9857
Epoch 22/200
54000/54000 [=====] - 63s 1ms/step - loss: 0.0038 -
acc: 0.9989 - val_loss: 0.0326 - val_acc: 0.9928
Epoch 23/200
54000/54000 [=====] - 61s 1ms/step - loss: 8.7408e-04 -
acc: 0.9998 - val_loss: 0.0306 - val_acc: 0.9943
Epoch 24/200
54000/54000 [=====] - 61s 1ms/step - loss: 3.6210e-04 -
acc: 0.9999 - val_loss: 0.0312 - val_acc: 0.9945
Epoch 25/200
54000/54000 [=====] - 61s 1ms/step - loss: 1.8320e-04 -
acc: 1.0000 - val_loss: 0.0324 - val_acc: 0.9942
Epoch 26/200
54000/54000 [=====] - 61s 1ms/step - loss: 1.2057e-04 -
acc: 1.0000 - val_loss: 0.0334 - val_acc: 0.9938
Epoch 27/200
54000/54000 [=====] - 61s 1ms/step - loss: 8.8838e-05 -
acc: 1.0000 - val_loss: 0.0348 - val_acc: 0.9937
Epoch 28/200
54000/54000 [=====] - 61s 1ms/step - loss: 5.9450e-05 -
acc: 1.0000 - val_loss: 0.0355 - val_acc: 0.9938
Epoch 29/200
54000/54000 [=====] - 61s 1ms/step - loss: 4.6951e-05 -
acc: 1.0000 - val_loss: 0.0358 - val_acc: 0.9938
Epoch 30/200
54000/54000 [=====] - 61s 1ms/step - loss: 4.4265e-05 -
acc: 1.0000 - val_loss: 0.0359 - val_acc: 0.9940
Epoch 31/200
54000/54000 [=====] - 62s 1ms/step - loss: 4.8221e-05 -
acc: 1.0000 - val_loss: 0.0362 - val_acc: 0.9942
Epoch 32/200
54000/54000 [=====] - 62s 1ms/step - loss: 5.1690e-05 -
acc: 1.0000 - val_loss: 0.0365 - val_acc: 0.9940
Epoch 33/200

```

54000/54000 [=====] - 61s 1ms/step - loss: 3.3616e-05 -
acc: 1.0000 - val_loss: 0.0367 - val_acc: 0.9940
Epoch 34/200
54000/54000 [=====] - 61s 1ms/step - loss: 3.3328e-05 -
acc: 1.0000 - val_loss: 0.0368 - val_acc: 0.9940

```

```
[ ]:
```

```

[51]: def draw_training_info_plots(_history):
        """Draw loss graphs at the training and validation stage"""
        acc = _history.history['acc']
        val_acc = _history.history['val_acc']
        loss = _history.history['loss']
        val_loss = _history.history['val_loss']

        epochs_plot = range(1, len(acc) + 1)
        plt.plot(epochs_plot, acc, 'b--', label='Training acc')
        plt.plot(epochs_plot, val_acc, 'b', label='Validation acc')
        plt.title('Training and validation accuracy')
        plt.xlabel('epoch')
        plt.ylabel('accuracy')
        plt.legend()
        plt.figure()

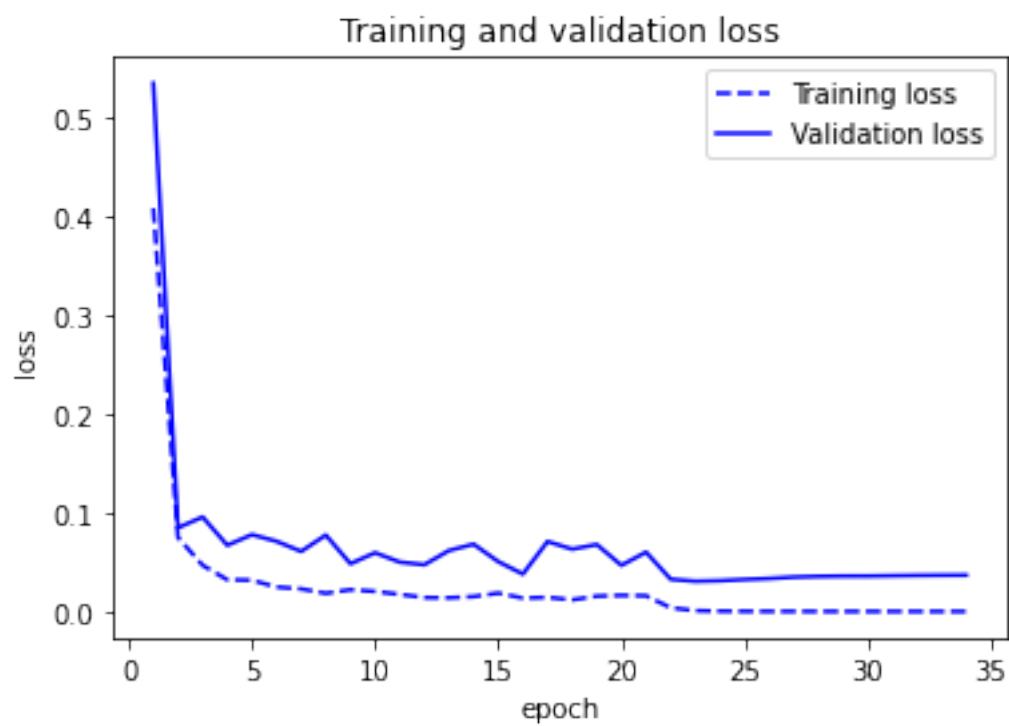
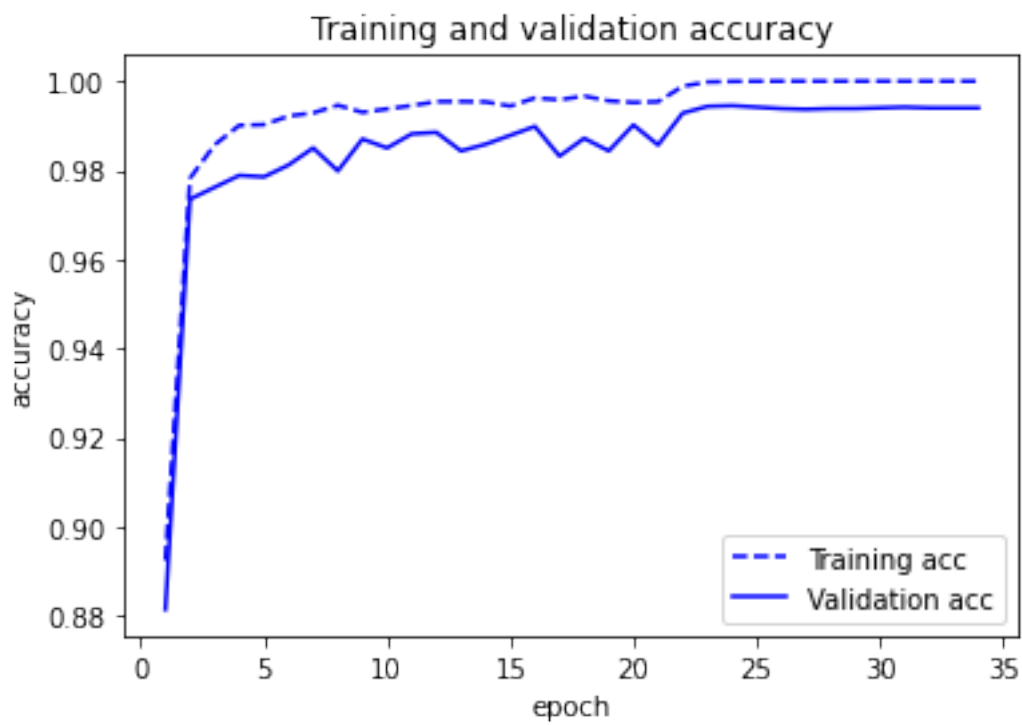
        plt.plot(epochs_plot, loss, 'b--', label='Training loss')
        plt.plot(epochs_plot, val_loss, 'b', label='Validation loss')
        plt.title('Training and validation loss')
        plt.xlabel('epoch')
        plt.ylabel('loss')
        plt.legend()
        plt.show()

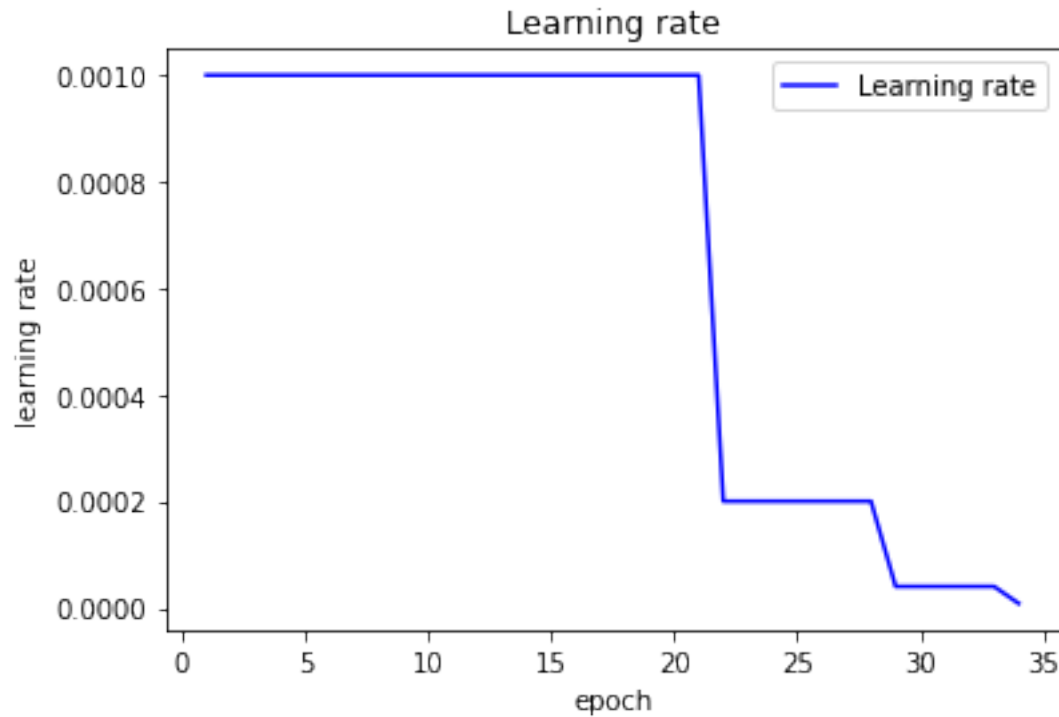
        if 'lr' in _history.history:
            learning_rate = _history.history['lr']
            plt.plot(epochs_plot, learning_rate, 'b', label='Learning rate')
            plt.title('Learning rate')
            plt.xlabel('epoch')
            plt.ylabel('learning rate')
            plt.legend()
            plt.show()

        return

draw_training_info_plots(history_resnet)

```





```
[52]: print('Accuracy:',Resnet_model.evaluate(test_X,test_y,verbose=0)[1])
```

Accuracy: 0.9934999942779541

Classification Score and Confusion Metric

```
[53]: predictions = Resnet_model.predict(test_X)

from sklearn.metrics import classification_report
print("EVALUATION ON TESTING DATA")
print(classification_report(y_test, np.argmax(predictions,axis=1)))
```

EVALUATION ON TESTING DATA

	precision	recall	f1-score	support
0	0.99	1.00	1.00	980
1	0.99	1.00	1.00	1135
2	1.00	1.00	1.00	1032
3	0.99	1.00	0.99	1010
4	0.99	0.99	0.99	982
5	0.99	0.99	0.99	892
6	1.00	0.99	0.99	958
7	0.99	0.99	0.99	1028
8	0.99	1.00	0.99	974
9	0.99	0.99	0.99	1009

accuracy			0.99	10000
macro avg	0.99	0.99	0.99	10000
weighted avg	0.99	0.99	0.99	10000

```
[54]: from sklearn.metrics import confusion_matrix
import pandas as pd
print ("Confusion matrix")
pd.DataFrame(confusion_matrix(y_test,np.argmax(predictions,axis=1)),columns=np.
    ↳unique(y_test),index=np.unique(y_test))
```

Confusion matrix

```
[54]:
```

	0	1	2	3	4	5	6	7	8	9
0	978	0	1	0	0	0	0	1	0	0
1	0	1131	0	0	0	2	0	1	0	1
2	0	1	1027	0	1	0	0	1	2	0
3	0	1	0	1005	0	2	0	0	2	0
4	0	0	0	0	974	0	3	0	0	5
5	0	0	0	7	0	883	1	1	0	0
6	4	2	0	0	1	0	950	0	1	0
7	0	2	2	0	0	0	0	1022	1	1
8	0	1	0	0	0	0	0	1	970	2
9	1	0	0	0	5	4	0	2	2	995

```
[56]: Resnet_model.save('Resnet_model_mnist.h5')
```

```
[ ]:
```