**Procedure**

**Step 1: Import all the necessary packages and modules.**

**import** **keras**

**from** **keras.datasets** **import** mnist

**from** **keras.models** **import** Sequential

**from** **keras.layers** **import** Dense, Dropout, Flatten

**from** **keras.layers** **import** Conv2D, MaxPooling2D

**from** **keras** **import** backend **as** K

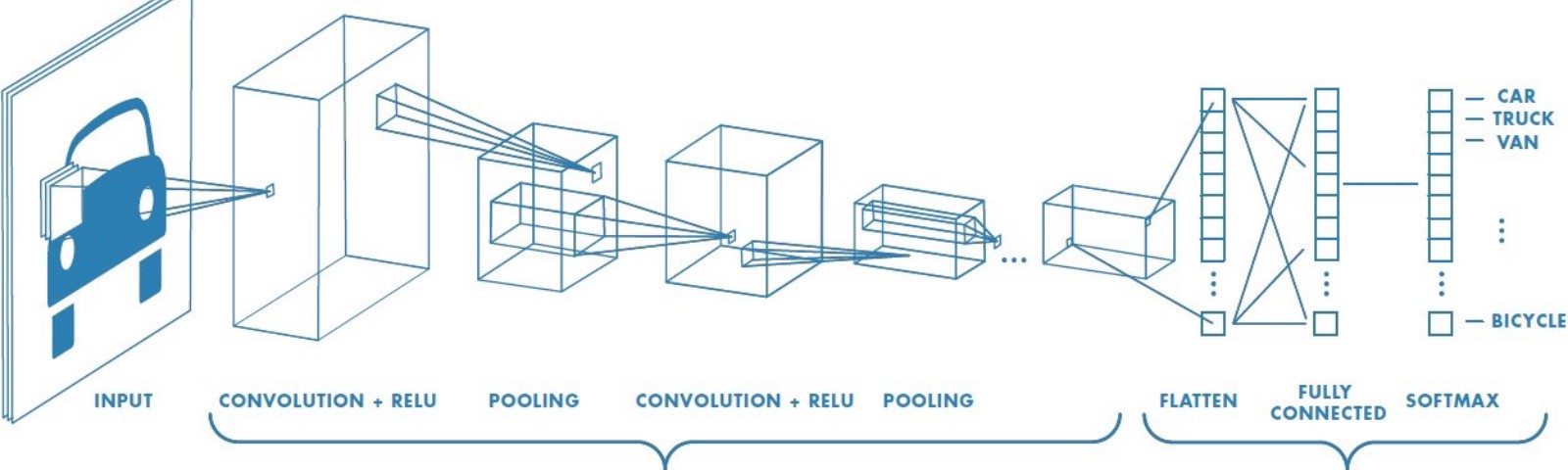
**from** **sklearn** **import** datasets, svm, metrics

**from** **sklearn.model\_selection** **import** train\_test\_split

**import** **matplotlib.pyplot** **as** **plt**

**import** **pandas** **as** **pd**

Using just 3 packages, keras, pandas, sklearn, all the modules and everything needed for this project is imported. The main things to take not of from keras are mnist, and conv2d. [Mnist](http://yann.lecun.com/exdb/mnist/) is the database from which the labeled numerical digits are imported. Conv2d is just a module which helps to build a convolutional neural network. From sklearn, the main module to be noted is train\_test\_split, which allows us to vary the amount of training data. Pandas and matplotlib are used for data plotting and making a data table internally.



**Step 2: Write a function to load the MNIST database into training and testing variables.**

**def** load\_mnist\_data(test\_size\_per):

(x\_train, y\_train), (x\_test, y\_test) = mnist.load\_data()

print(x\_train.shape, y\_train.shape)

x\_train, x\_test, y\_train, y\_test = train\_test\_split(

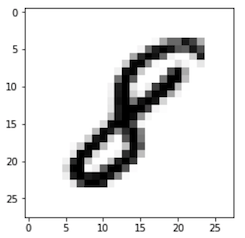
x\_train, y\_train, test\_size=test\_size\_per)

print(x\_train.shape)

**return** x\_train, y\_train, x\_test, y\_test

The name of it being "load\_minst\_data". This does exactly what the name suggests and loads the variables with training and test data. The variables being x\_train, y\_train, x\_test, and y\_test. x and y are not coordinates, but rather x is the input and y is the output. The training variables, x\_train, and y\_train are where all the training images that the computer uses to learn are kept. While x\_test, and y\_test is where the testing images are kept. These images are what the computer tests itself on to refine the algorithm. There are 60,000 images in the mnist database and by default, 50,000 are kept for training and the remaining 10,000 are used as testing images. On the 4th and 5th lines of the cell, the function train\_test\_split() is used to split the data into training and testing images by taking a percentage value which is then applied to the database and made into the test set. For example, if the percentage is given as 0.5 or 50%, 30,000 images would be used as testing and 30,000 for training. This is not a set value however, there is a parameter known as test\_size\_per which allows the percentage to be changed every time the function is called upon.

The following image shows a sample image from the mnist database.



**Step 3: Write a function to reshape and prepare all the images uniformly for input into the algorithm.**

**def** prepare\_input(x\_train, y\_train, x\_test, y\_test):

x\_train = x\_train.reshape(x\_train.shape[0], 28, 28, 1)

x\_test = x\_test.reshape(x\_test.shape[0], 28, 28, 1)

*# convert class vectors to binary class matrices*

y\_train = keras.utils.to\_categorical(y\_train, 10)

y\_test = keras.utils.to\_categorical(y\_test, 10)

x\_train = x\_train.astype('float32')

x\_test = x\_test.astype('float32')

x\_train /= 255

x\_test /= 255

print('x\_train shape:', x\_train.shape)

print(x\_train.shape[0], 'train samples')

print(x\_test.shape[0], 'test samples')

**return** x\_train, y\_train, x\_test, y\_test

This is one of the most important parts of the whole algorithm, also only to the actual machine learning part itself. All the images are now reshaped into 28 x 28, just in case some of them were not that size. Furthermore, this uniformity allows the computer to look at all of the images consistently the same. Furthermore, all the images are shuffled, so that certain patterns and exclusions of digits do not occur, therefore removing the chance that the computer's "learning" is skewed. In summation, this function allows for the uniformity of the images so that the computer sees every image equally and is not skewed by unnatural patterns such as the absence of a certain digit.

**Step 4: Writing a function to setup the model, train the model, and save the model.**

**def** setup\_model(x\_train, y\_train, x\_test, y\_test ):

batch\_size = 128

num\_classes = 10

epochs = 10

input\_shape = (28, 28, 1)

model = Sequential()

model.add(Conv2D(32, kernel\_size=(5, 5),activation='relu',input\_shape=input\_shape))

model.add(MaxPooling2D(pool\_size=(2, 2)))

model.add(Conv2D(64, (3, 3), activation='relu'))

model.add(MaxPooling2D(pool\_size=(2, 2)))

model.add(Flatten())

model.add(Dense(128, activation='relu'))

model.add(Dropout(0.3))

model.add(Dense(64, activation='relu'))

model.add(Dropout(0.5))

model.add(Dense(num\_classes, activation='softmax'))

model.compile(loss=keras.losses.categorical\_crossentropy,optimizer=keras.optimizers.Adadelta(),metrics=['accuracy'])

hist = model.fit(x\_train, y\_train,batch\_size=batch\_size,epochs=epochs,verbose=1,validation\_data=(x\_test, y\_test))

print("The model has successfully trained")

score = model.evaluate(x\_test, y\_test, verbose=0)

model.save('mnist.h5')

print("Saving the model as mnist.h5")

**return** score

This function is what sets up a model (convolutional neural network), and trains it with the training set. This is still a function, and is not executing anything, so no output is given. The way a machine learning model learns/trains itself is that it uses differential calculus. Basically, an inverse parabola is graphed internally, called the gradient descent that allows the computer to see where it is. The x axis is the answer that it is guessing. For example, an eight is given as an input, and the computer guesses that it's a 5. Now, that is the wrong answer, but the computer now knows that the eight is not a 5. The y axis is the cost. Basically, the higher the cost, the farther the model is from the answer. Graphing this would show an inverse parabola, where the answer (the number eight in the example) on the x axis, and the lowest value on the y axis (the cost) are located. The end goal of the model is to reach that low point. once it does, and it gets a problem correct, it moves on and uses the same formula that it used for the previous one. Basically, the computer at first uses random values, and then narrows it down to the correct value, and uses the formula that got it the correct values. This is all calculated in a matter of mere seconds though. This is why a larger training set, like common sense would tell us, makes the computer more accurate in its learning and testing. Below is an image of the gradient descent.

**Step 6: Executing the functions in a for loop, so that each iteration, a bigger training set is used.**

summary\_values = {}

*# Takes increasingly more images to increase the accuracy*

training\_sets = [100,200 ,300 ,400 , 500, 600, 700, 800, 900, 1000, 1250,1500,1750,2000,2250,2500,2750,3000,3250,3500,3750,4000,4250,4500,4750,5000]

j = 0

column\_names = ["Amount of Training images", "Accuracy"]

df = pd.DataFrame(columns = column\_names, index = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26])

**for** test\_per **in** training\_sets:

training\_per = 1- (test\_per/60000) *#calculation is giving the test percentage*

print(training\_per)

x\_train, y\_train, x\_test, y\_test = load\_mnist\_data(training\_per)

print('Testing with train set size of: ', x\_train.shape)

x\_train, y\_train, x\_test, y\_test = prepare\_input(x\_train, y\_train, x\_test, y\_test)

score = setup\_model(x\_train, y\_train, x\_test, y\_test)

summary\_values[x\_train.shape] = {test\_per/100, score[0], score[1]}

i = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26]

new\_data\_list = [test\_per, score[1]]

df.loc[i[j]] = new\_data\_list

j += 1

print(summary\_values)

0.9983333333333333

(60000, 28, 28) (60000,)

(100, 28, 28)

Testing with train set size of: (100, 28, 28)

x\_train shape: (100, 28, 28, 1)

100 train samples

59900 test samples

Train on 100 samples, validate on 59900 samples

Epoch 1/10

100/100 [==============================] - 12s 122ms/step - loss: 2.3117 - accuracy: 0.0900 - val\_loss: 2.2862 - val\_accuracy: 0.2059

Epoch 2/10

100/100 [==============================] - 11s 106ms/step - loss: 2.2679 - accuracy: 0.2100 - val\_loss: 2.2642 - val\_accuracy: 0.2679

Epoch 3/10

100/100 [==============================] - 11s 114ms/step - loss: 2.2501 - accuracy: 0.1600 - val\_loss: 2.2493 - val\_accuracy: 0.1748

Epoch 4/10

100/100 [==============================] - 11s 106ms/step - loss: 2.2122 - accuracy: 0.2200 - val\_loss: 2.2274 - val\_accuracy: 0.1498

Epoch 5/10

100/100 [==============================] - 16s 162ms/step - loss: 2.1858 - accuracy: 0.2800 - val\_loss: 2.2027 - val\_accuracy: 0.1829

Epoch 6/10

100/100 [==============================] - 12s 121ms/step - loss: 2.1288 - accuracy: 0.2400 - val\_loss: 2.1618 - val\_accuracy: 0.2567

Epoch 7/10

100/100 [==============================] - 12s 121ms/step - loss: 2.1128 - accuracy: 0.2600 - val\_loss: 2.1396 - val\_accuracy: 0.3057

Epoch 8/10

100/100 [==============================] - 12s 119ms/step - loss: 2.0578 - accuracy: 0.2600 - val\_loss: 2.1106 - val\_accuracy: 0.1816

Epoch 9/10

100/100 [==============================] - 13s 127ms/step - loss: 2.0516 - accuracy: 0.2400 - val\_loss: 1.9946 - val\_accuracy: 0.4640

Epoch 10/10

100/100 [==============================] - 15s 146ms/step - loss: 1.9776 - accuracy: 0.3200 - val\_loss: 1.9474 - val\_accuracy: 0.4136

The model has successfully trained

Saving the model as mnist.h5

0.9966666666666667

(60000, 28, 28) (60000,)

(200, 28, 28)

Testing with train set size of: (200, 28, 28)

x\_train shape: (200, 28, 28, 1)

200 train samples

59800 test samples

Train on 200 samples, validate on 59800 samples

Epoch 1/10

200/200 [==============================] - 10s 52ms/step - loss: 2.3065 - accuracy: 0.0800 - val\_loss: 2.2781 - val\_accuracy: 0.1150

Epoch 2/10

200/200 [==============================] - 11s 54ms/step - loss: 2.2574 - accuracy: 0.1500 - val\_loss: 2.2518 - val\_accuracy: 0.1520

Epoch 3/10

200/200 [==============================] - 11s 57ms/step - loss: 2.2570 - accuracy: 0.0950 - val\_loss: 2.2066 - val\_accuracy: 0.1192

Epoch 4/10

200/200 [==============================] - 13s 67ms/step - loss: 2.2060 - accuracy: 0.1650 - val\_loss: 2.1518 - val\_accuracy: 0.4252

Epoch 5/10

200/200 [==============================] - 12s 60ms/step - loss: 2.1723 - accuracy: 0.1850 - val\_loss: 2.1369 - val\_accuracy: 0.2132

Epoch 6/10

200/200 [==============================] - 11s 55ms/step - loss: 2.1272 - accuracy: 0.2550 - val\_loss: 2.0190 - val\_accuracy: 0.3236

Epoch 7/10

200/200 [==============================] - 11s 56ms/step - loss: 2.0867 - accuracy: 0.2700 - val\_loss: 2.0356 - val\_accuracy: 0.3126

Epoch 8/10

200/200 [==============================] - 18s 88ms/step - loss: 2.0230 - accuracy: 0.3000 - val\_loss: 1.7765 - val\_accuracy: 0.5902

Epoch 9/10

200/200 [==============================] - 16s 82ms/step - loss: 1.8218 - accuracy: 0.3850 - val\_loss: 1.7845 - val\_accuracy: 0.3602

Epoch 10/10

200/200 [==============================] - 12s 59ms/step - loss: 1.8762 - accuracy: 0.3400 - val\_loss: 1.7486 - val\_accuracy: 0.5124

The model has successfully trained

Saving the model as mnist.h5

0.995

(60000, 28, 28) (60000,)

(300, 28, 28)

Testing with train set size of: (300, 28, 28)

x\_train shape: (300, 28, 28, 1)

300 train samples

59700 test samples

Train on 300 samples, validate on 59700 samples

Epoch 1/10

300/300 [==============================] - 11s 36ms/step - loss: 2.3023 - accuracy: 0.1267 - val\_loss: 2.2708 - val\_accuracy: 0.1019

Epoch 2/10

300/300 [==============================] - 11s 36ms/step - loss: 2.2497 - accuracy: 0.1800 - val\_loss: 2.2178 - val\_accuracy: 0.1244

Epoch 3/10

300/300 [==============================] - 11s 35ms/step - loss: 2.1949 - accuracy: 0.2100 - val\_loss: 2.1362 - val\_accuracy: 0.2035

Epoch 4/10

300/300 [==============================] - 10s 35ms/step - loss: 2.1263 - accuracy: 0.2233 - val\_loss: 2.0077 - val\_accuracy: 0.5505

Epoch 5/10

300/300 [==============================] - 11s 35ms/step - loss: 2.0001 - accuracy: 0.2967 - val\_loss: 1.9212 - val\_accuracy: 0.2624

Epoch 6/10

300/300 [==============================] - 13s 42ms/step - loss: 1.9954 - accuracy: 0.2867 - val\_loss: 1.6869 - val\_accuracy: 0.6276

Epoch 7/10

300/300 [==============================] - 11s 36ms/step - loss: 1.7400 - accuracy: 0.4333 - val\_loss: 1.6392 - val\_accuracy: 0.4327

Epoch 8/10

300/300 [==============================] - 11s 36ms/step - loss: 1.7066 - accuracy: 0.4333 - val\_loss: 1.2951 - val\_accuracy: 0.6281

Epoch 9/10

300/300 [==============================] - 12s 40ms/step - loss: 1.5044 - accuracy: 0.5000 - val\_loss: 1.2265 - val\_accuracy: 0.6582

Epoch 10/10

300/300 [==============================] - 11s 37ms/step - loss: 1.3611 - accuracy: 0.5300 - val\_loss: 0.9859 - val\_accuracy: 0.7883

The model has successfully trained

Saving the model as mnist.h5

0.9933333333333333

(60000, 28, 28) (60000,)

(400, 28, 28)

Testing with train set size of: (400, 28, 28)

x\_train shape: (400, 28, 28, 1)

400 train samples

59600 test samples

Train on 400 samples, validate on 59600 samples

Epoch 1/10

400/400 [==============================] - 11s 27ms/step - loss: 2.2979 - accuracy: 0.1525 - val\_loss: 2.2714 - val\_accuracy: 0.1775

Epoch 2/10

400/400 [==============================] - 11s 27ms/step - loss: 2.2163 - accuracy: 0.1700 - val\_loss: 2.1560 - val\_accuracy: 0.3127

Epoch 3/10

400/400 [==============================] - 13s 32ms/step - loss: 2.1427 - accuracy: 0.2200 - val\_loss: 2.1202 - val\_accuracy: 0.1869

Epoch 4/10

400/400 [==============================] - 11s 27ms/step - loss: 2.0840 - accuracy: 0.2425 - val\_loss: 1.8108 - val\_accuracy: 0.5712

Epoch 5/10

400/400 [==============================] - 11s 27ms/step - loss: 1.8986 - accuracy: 0.3425 - val\_loss: 1.7931 - val\_accuracy: 0.4050

Epoch 6/10

400/400 [==============================] - 11s 27ms/step - loss: 1.8151 - accuracy: 0.3625 - val\_loss: 1.5609 - val\_accuracy: 0.5302

Epoch 7/10

400/400 [==============================] - 11s 26ms/step - loss: 1.6738 - accuracy: 0.4300 - val\_loss: 1.3448 - val\_accuracy: 0.5663

Epoch 8/10

400/400 [==============================] - 11s 27ms/step - loss: 1.4808 - accuracy: 0.5125 - val\_loss: 0.9635 - val\_accuracy: 0.7289

Epoch 9/10

400/400 [==============================] - 11s 27ms/step - loss: 1.2246 - accuracy: 0.6075 - val\_loss: 1.8114 - val\_accuracy: 0.3914

Epoch 10/10

400/400 [==============================] - 11s 26ms/step - loss: 1.6187 - accuracy: 0.4925 - val\_loss: 0.8299 - val\_accuracy: 0.7593

The model has successfully trained

Saving the model as mnist.h5

0.9916666666666667

(60000, 28, 28) (60000,)

(500, 28, 28)

Testing with train set size of: (500, 28, 28)

x\_train shape: (500, 28, 28, 1)

500 train samples

59500 test samples

Train on 500 samples, validate on 59500 samples

Epoch 1/10

500/500 [==============================] - 11s 21ms/step - loss: 2.2915 - accuracy: 0.1520 - val\_loss: 2.2344 - val\_accuracy: 0.2735

Epoch 2/10

500/500 [==============================] - 15s 29ms/step - loss: 2.2184 - accuracy: 0.2100 - val\_loss: 2.1304 - val\_accuracy: 0.2635

Epoch 3/10

500/500 [==============================] - 12s 24ms/step - loss: 2.0686 - accuracy: 0.3140 - val\_loss: 1.9462 - val\_accuracy: 0.3147

Epoch 4/10

500/500 [==============================] - 11s 21ms/step - loss: 2.0143 - accuracy: 0.2840 - val\_loss: 1.6752 - val\_accuracy: 0.5093

Epoch 5/10

500/500 [==============================] - 11s 22ms/step - loss: 1.8996 - accuracy: 0.3340 - val\_loss: 1.5989 - val\_accuracy: 0.5398

Epoch 6/10

500/500 [==============================] - 11s 22ms/step - loss: 1.6635 - accuracy: 0.4560 - val\_loss: 1.1175 - val\_accuracy: 0.7032

Epoch 7/10

500/500 [==============================] - 11s 22ms/step - loss: 1.4251 - accuracy: 0.5040 - val\_loss: 1.2945 - val\_accuracy: 0.5572

Epoch 8/10

500/500 [==============================] - 11s 23ms/step - loss: 1.3066 - accuracy: 0.5520 - val\_loss: 1.0419 - val\_accuracy: 0.6540

Epoch 9/10

500/500 [==============================] - 13s 25ms/step - loss: 1.0993 - accuracy: 0.6320 - val\_loss: 0.8825 - val\_accuracy: 0.6981

Epoch 10/10

500/500 [==============================] - 11s 21ms/step - loss: 1.1477 - accuracy: 0.5940 - val\_loss: 0.6113 - val\_accuracy: 0.8489

The model has successfully trained

Saving the model as mnist.h5

0.99

(60000, 28, 28) (60000,)

(600, 28, 28)

Testing with train set size of: (600, 28, 28)

x\_train shape: (600, 28, 28, 1)

600 train samples

59400 test samples

Train on 600 samples, validate on 59400 samples

Epoch 1/10

600/600 [==============================] - 11s 18ms/step - loss: 2.2921 - accuracy: 0.1133 - val\_loss: 2.2331 - val\_accuracy: 0.2438

Epoch 2/10

600/600 [==============================] - 11s 18ms/step - loss: 2.2131 - accuracy: 0.1750 - val\_loss: 2.1035 - val\_accuracy: 0.3580

Epoch 3/10

600/600 [==============================] - 11s 18ms/step - loss: 2.1091 - accuracy: 0.2850 - val\_loss: 1.8949 - val\_accuracy: 0.5338

Epoch 4/10

600/600 [==============================] - 11s 18ms/step - loss: 1.9377 - accuracy: 0.3150 - val\_loss: 1.7653 - val\_accuracy: 0.3502

Epoch 5/10

600/600 [==============================] - 11s 18ms/step - loss: 1.7313 - accuracy: 0.3983 - val\_loss: 1.2287 - val\_accuracy: 0.6453

Epoch 6/10

600/600 [==============================] - 11s 18ms/step - loss: 1.4136 - accuracy: 0.5217 - val\_loss: 0.8218 - val\_accuracy: 0.7969

Epoch 7/10

600/600 [==============================] - 11s 18ms/step - loss: 1.2522 - accuracy: 0.5917 - val\_loss: 0.7405 - val\_accuracy: 0.8733

Epoch 8/10

600/600 [==============================] - 12s 21ms/step - loss: 0.9721 - accuracy: 0.6750 - val\_loss: 0.5883 - val\_accuracy: 0.8552

Epoch 9/10

600/600 [==============================] - 11s 18ms/step - loss: 1.0473 - accuracy: 0.6417 - val\_loss: 0.5580 - val\_accuracy: 0.8728

Epoch 10/10

600/600 [==============================] - 11s 18ms/step - loss: 0.7741 - accuracy: 0.7350 - val\_loss: 0.3970 - val\_accuracy: 0.9013

The model has successfully trained

Saving the model as mnist.h5

0.9883333333333333

(60000, 28, 28) (60000,)

(700, 28, 28)

Testing with train set size of: (700, 28, 28)

x\_train shape: (700, 28, 28, 1)

700 train samples

59300 test samples

Train on 700 samples, validate on 59300 samples

Epoch 1/10

700/700 [==============================] - 11s 16ms/step - loss: 2.2778 - accuracy: 0.1243 - val\_loss: 2.1848 - val\_accuracy: 0.4624

Epoch 2/10

700/700 [==============================] - 11s 16ms/step - loss: 2.1219 - accuracy: 0.2529 - val\_loss: 1.9712 - val\_accuracy: 0.3544

Epoch 3/10

700/700 [==============================] - 11s 15ms/step - loss: 1.9779 - accuracy: 0.2800 - val\_loss: 1.6921 - val\_accuracy: 0.4866

Epoch 4/10

700/700 [==============================] - 11s 15ms/step - loss: 1.7726 - accuracy: 0.3886 - val\_loss: 1.4623 - val\_accuracy: 0.6864

Epoch 5/10

700/700 [==============================] - 11s 15ms/step - loss: 1.3912 - accuracy: 0.5371 - val\_loss: 0.8085 - val\_accuracy: 0.7360

Epoch 6/10

700/700 [==============================] - 13s 18ms/step - loss: 1.1790 - accuracy: 0.6014 - val\_loss: 0.7642 - val\_accuracy: 0.7909

Epoch 7/10

700/700 [==============================] - 12s 17ms/step - loss: 0.9696 - accuracy: 0.6900 - val\_loss: 0.4888 - val\_accuracy: 0.8726

Epoch 8/10

700/700 [==============================] - 11s 15ms/step - loss: 0.7049 - accuracy: 0.7771 - val\_loss: 0.4045 - val\_accuracy: 0.8720

Epoch 9/10

700/700 [==============================] - 11s 15ms/step - loss: 0.6965 - accuracy: 0.7614 - val\_loss: 0.5663 - val\_accuracy: 0.8145

Epoch 10/10

700/700 [==============================] - 11s 15ms/step - loss: 0.6449 - accuracy: 0.8114 - val\_loss: 0.3194 - val\_accuracy: 0.9036

The model has successfully trained

Saving the model as mnist.h5

0.9866666666666667

(60000, 28, 28) (60000,)

(800, 28, 28)

Testing with train set size of: (800, 28, 28)

x\_train shape: (800, 28, 28, 1)

800 train samples

59200 test samples

Train on 800 samples, validate on 59200 samples

Epoch 1/10

800/800 [==============================] - 11s 14ms/step - loss: 2.2741 - accuracy: 0.1412 - val\_loss: 2.2143 - val\_accuracy: 0.0990

Epoch 2/10

800/800 [==============================] - 11s 14ms/step - loss: 2.1362 - accuracy: 0.2375 - val\_loss: 1.9732 - val\_accuracy: 0.3168

Epoch 3/10

800/800 [==============================] - 13s 16ms/step - loss: 1.9027 - accuracy: 0.3487 - val\_loss: 1.6722 - val\_accuracy: 0.4162

Epoch 4/10

800/800 [==============================] - 11s 14ms/step - loss: 1.5240 - accuracy: 0.4988 - val\_loss: 1.2136 - val\_accuracy: 0.5897

Epoch 5/10

800/800 [==============================] - 11s 14ms/step - loss: 1.2767 - accuracy: 0.5800 - val\_loss: 0.8748 - val\_accuracy: 0.7820

Epoch 6/10

800/800 [==============================] - 11s 14ms/step - loss: 0.9594 - accuracy: 0.6900 - val\_loss: 0.5812 - val\_accuracy: 0.8404

Epoch 7/10

800/800 [==============================] - 11s 14ms/step - loss: 0.8518 - accuracy: 0.7212 - val\_loss: 0.6028 - val\_accuracy: 0.8376

Epoch 8/10

800/800 [==============================] - 11s 14ms/step - loss: 0.7121 - accuracy: 0.7775 - val\_loss: 0.4995 - val\_accuracy: 0.8399

Epoch 9/10

800/800 [==============================] - 11s 14ms/step - loss: 0.6012 - accuracy: 0.8125 - val\_loss: 0.3811 - val\_accuracy: 0.8890

Epoch 10/10

800/800 [==============================] - 11s 14ms/step - loss: 0.5690 - accuracy: 0.8313 - val\_loss: 0.3718 - val\_accuracy: 0.8869

The model has successfully trained

Saving the model as mnist.h5

0.985

(60000, 28, 28) (60000,)

(900, 28, 28)

Testing with train set size of: (900, 28, 28)

x\_train shape: (900, 28, 28, 1)

900 train samples

59100 test samples

Train on 900 samples, validate on 59100 samples

Epoch 1/10

900/900 [==============================] - 13s 15ms/step - loss: 2.2669 - accuracy: 0.1522 - val\_loss: 2.2453 - val\_accuracy: 0.0974

Epoch 2/10

900/900 [==============================] - 18s 20ms/step - loss: 2.1561 - accuracy: 0.2333 - val\_loss: 2.1198 - val\_accuracy: 0.2436

Epoch 3/10

900/900 [==============================] - 14s 16ms/step - loss: 1.9353 - accuracy: 0.3222 - val\_loss: 1.8313 - val\_accuracy: 0.5116

Epoch 4/10

900/900 [==============================] - 15s 17ms/step - loss: 1.7530 - accuracy: 0.4089 - val\_loss: 1.5850 - val\_accuracy: 0.4725

Epoch 5/10

900/900 [==============================] - 12s 13ms/step - loss: 1.4358 - accuracy: 0.5144 - val\_loss: 1.3141 - val\_accuracy: 0.6562

Epoch 6/10

900/900 [==============================] - 11s 12ms/step - loss: 1.2634 - accuracy: 0.5767 - val\_loss: 1.3984 - val\_accuracy: 0.5030

Epoch 7/10

900/900 [==============================] - 11s 12ms/step - loss: 1.2360 - accuracy: 0.5833 - val\_loss: 0.8538 - val\_accuracy: 0.7406

Epoch 8/10

900/900 [==============================] - 11s 12ms/step - loss: 0.9265 - accuracy: 0.6922 - val\_loss: 0.7182 - val\_accuracy: 0.7610

Epoch 9/10

900/900 [==============================] - 13s 15ms/step - loss: 0.8379 - accuracy: 0.7411 - val\_loss: 0.4640 - val\_accuracy: 0.8635

Epoch 10/10

900/900 [==============================] - 11s 12ms/step - loss: 0.6782 - accuracy: 0.7667 - val\_loss: 0.4426 - val\_accuracy: 0.8651

The model has successfully trained

Saving the model as mnist.h5

0.9833333333333333

(60000, 28, 28) (60000,)

(1000, 28, 28)

Testing with train set size of: (1000, 28, 28)

x\_train shape: (1000, 28, 28, 1)

1000 train samples

59000 test samples

Train on 1000 samples, validate on 59000 samples

Epoch 1/10

1000/1000 [==============================] - 14s 14ms/step - loss: 2.2613 - accuracy: 0.1410 - val\_loss: 2.1464 - val\_accuracy: 0.3585

Epoch 2/10

1000/1000 [==============================] - 14s 14ms/step - loss: 2.0430 - accuracy: 0.3060 - val\_loss: 1.5540 - val\_accuracy: 0.5526

Epoch 3/10

1000/1000 [==============================] - 17s 17ms/step - loss: 1.7320 - accuracy: 0.4110 - val\_loss: 1.4262 - val\_accuracy: 0.5954

Epoch 4/10

1000/1000 [==============================] - 15s 15ms/step - loss: 1.3067 - accuracy: 0.5660 - val\_loss: 0.8959 - val\_accuracy: 0.7408

Epoch 5/10

1000/1000 [==============================] - 14s 14ms/step - loss: 1.1117 - accuracy: 0.6290 - val\_loss: 0.6011 - val\_accuracy: 0.8192

Epoch 6/10

1000/1000 [==============================] - 15s 15ms/step - loss: 0.8548 - accuracy: 0.7130 - val\_loss: 0.4457 - val\_accuracy: 0.8622

Epoch 7/10

1000/1000 [==============================] - 14s 14ms/step - loss: 0.6544 - accuracy: 0.7880 - val\_loss: 0.3308 - val\_accuracy: 0.9074

Epoch 8/10

1000/1000 [==============================] - 15s 15ms/step - loss: 0.5503 - accuracy: 0.8120 - val\_loss: 0.3417 - val\_accuracy: 0.8958

Epoch 9/10

1000/1000 [==============================] - 15s 15ms/step - loss: 0.4747 - accuracy: 0.8530 - val\_loss: 0.2913 - val\_accuracy: 0.9123

Epoch 10/10

1000/1000 [==============================] - 13s 13ms/step - loss: 0.3956 - accuracy: 0.8720 - val\_loss: 0.2783 - val\_accuracy: 0.9149

The model has successfully trained

Saving the model as mnist.h5

0.9791666666666666

(60000, 28, 28) (60000,)

(1250, 28, 28)

Testing with train set size of: (1250, 28, 28)

x\_train shape: (1250, 28, 28, 1)

1250 train samples

58750 test samples

Train on 1250 samples, validate on 58750 samples

Epoch 1/10

1250/1250 [==============================] - 11s 9ms/step - loss: 2.2592 - accuracy: 0.1728 - val\_loss: 2.1093 - val\_accuracy: 0.4477

Epoch 2/10

1250/1250 [==============================] - 11s 9ms/step - loss: 2.0263 - accuracy: 0.2792 - val\_loss: 1.5063 - val\_accuracy: 0.5375

Epoch 3/10

1250/1250 [==============================] - 11s 9ms/step - loss: 1.5958 - accuracy: 0.4568 - val\_loss: 1.3172 - val\_accuracy: 0.6141

Epoch 4/10

1250/1250 [==============================] - 13s 11ms/step - loss: 1.1585 - accuracy: 0.6368 - val\_loss: 0.6477 - val\_accuracy: 0.8657

Epoch 5/10

1250/1250 [==============================] - 11s 9ms/step - loss: 0.9153 - accuracy: 0.7128 - val\_loss: 0.4402 - val\_accuracy: 0.9015

Epoch 6/10

1250/1250 [==============================] - 11s 9ms/step - loss: 0.6712 - accuracy: 0.7864 - val\_loss: 0.3313 - val\_accuracy: 0.9028

Epoch 7/10

1250/1250 [==============================] - 11s 9ms/step - loss: 0.6167 - accuracy: 0.8096 - val\_loss: 0.3466 - val\_accuracy: 0.9031

Epoch 8/10

1250/1250 [==============================] - 11s 9ms/step - loss: 0.4525 - accuracy: 0.8736 - val\_loss: 0.2356 - val\_accuracy: 0.9320

Epoch 9/10

1250/1250 [==============================] - 11s 9ms/step - loss: 0.4126 - accuracy: 0.8744 - val\_loss: 0.2538 - val\_accuracy: 0.9260

Epoch 10/10

1250/1250 [==============================] - 11s 9ms/step - loss: 0.3648 - accuracy: 0.9048 - val\_loss: 0.2020 - val\_accuracy: 0.9416

The model has successfully trained

Saving the model as mnist.h5

0.975

(60000, 28, 28) (60000,)

(1500, 28, 28)

Testing with train set size of: (1500, 28, 28)

x\_train shape: (1500, 28, 28, 1)

1500 train samples

58500 test samples

Train on 1500 samples, validate on 58500 samples

Epoch 1/10

1500/1500 [==============================] - 11s 8ms/step - loss: 2.1906 - accuracy: 0.2193 - val\_loss: 1.8545 - val\_accuracy: 0.4542

Epoch 2/10

1500/1500 [==============================] - 11s 7ms/step - loss: 1.7587 - accuracy: 0.4093 - val\_loss: 1.1351 - val\_accuracy: 0.7062

Epoch 3/10

1500/1500 [==============================] - 13s 9ms/step - loss: 1.2275 - accuracy: 0.5827 - val\_loss: 0.9144 - val\_accuracy: 0.7110

Epoch 4/10

1500/1500 [==============================] - 11s 8ms/step - loss: 0.8793 - accuracy: 0.7060 - val\_loss: 0.5965 - val\_accuracy: 0.7951

Epoch 5/10

1500/1500 [==============================] - 12s 8ms/step - loss: 0.7160 - accuracy: 0.7753 - val\_loss: 0.3531 - val\_accuracy: 0.8991

Epoch 6/10

1500/1500 [==============================] - 11s 8ms/step - loss: 0.5515 - accuracy: 0.8340 - val\_loss: 0.3380 - val\_accuracy: 0.8953

Epoch 7/10

1500/1500 [==============================] - 11s 8ms/step - loss: 0.5249 - accuracy: 0.8293 - val\_loss: 0.2361 - val\_accuracy: 0.9311

Epoch 8/10

1500/1500 [==============================] - 11s 8ms/step - loss: 0.4148 - accuracy: 0.8640 - val\_loss: 0.1959 - val\_accuracy: 0.9431

Epoch 9/10

1500/1500 [==============================] - 11s 7ms/step - loss: 0.3457 - accuracy: 0.8967 - val\_loss: 0.1820 - val\_accuracy: 0.9468

Epoch 10/10

1500/1500 [==============================] - 11s 7ms/step - loss: 0.3126 - accuracy: 0.9073 - val\_loss: 0.1947 - val\_accuracy: 0.9437

The model has successfully trained

Saving the model as mnist.h5

0.9708333333333333

(60000, 28, 28) (60000,)

(1750, 28, 28)

Testing with train set size of: (1750, 28, 28)

x\_train shape: (1750, 28, 28, 1)

1750 train samples

58250 test samples

Train on 1750 samples, validate on 58250 samples

Epoch 1/10

1750/1750 [==============================] - 13s 8ms/step - loss: 2.1445 - accuracy: 0.2280 - val\_loss: 1.8178 - val\_accuracy: 0.4917

Epoch 2/10

1750/1750 [==============================] - 12s 7ms/step - loss: 1.5816 - accuracy: 0.4554 - val\_loss: 1.0556 - val\_accuracy: 0.7162

Epoch 3/10

1750/1750 [==============================] - 11s 6ms/step - loss: 1.0835 - accuracy: 0.6406 - val\_loss: 0.6808 - val\_accuracy: 0.8553

Epoch 4/10

1750/1750 [==============================] - 11s 7ms/step - loss: 0.7415 - accuracy: 0.7520 - val\_loss: 0.4008 - val\_accuracy: 0.8880

Epoch 5/10

1750/1750 [==============================] - 12s 7ms/step - loss: 0.5698 - accuracy: 0.8183 - val\_loss: 0.2825 - val\_accuracy: 0.9226

Epoch 6/10

1750/1750 [==============================] - 11s 6ms/step - loss: 0.4523 - accuracy: 0.8543 - val\_loss: 0.2431 - val\_accuracy: 0.9306

Epoch 7/10

1750/1750 [==============================] - 11s 6ms/step - loss: 0.4026 - accuracy: 0.8823 - val\_loss: 0.2365 - val\_accuracy: 0.9317

Epoch 8/10

1750/1750 [==============================] - 11s 6ms/step - loss: 0.3410 - accuracy: 0.9046 - val\_loss: 0.1931 - val\_accuracy: 0.9449

Epoch 9/10

1750/1750 [==============================] - 11s 6ms/step - loss: 0.2691 - accuracy: 0.9189 - val\_loss: 0.2160 - val\_accuracy: 0.9352

Epoch 10/10

1750/1750 [==============================] - 11s 6ms/step - loss: 0.2721 - accuracy: 0.9160 - val\_loss: 0.1944 - val\_accuracy: 0.9456

The model has successfully trained

Saving the model as mnist.h5

0.9666666666666667

(60000, 28, 28) (60000,)

(2000, 28, 28)

Testing with train set size of: (2000, 28, 28)

x\_train shape: (2000, 28, 28, 1)

2000 train samples

58000 test samples

Train on 2000 samples, validate on 58000 samples

Epoch 1/10

2000/2000 [==============================] - 12s 6ms/step - loss: 2.1956 - accuracy: 0.2075 - val\_loss: 1.9066 - val\_accuracy: 0.2585

Epoch 2/10

2000/2000 [==============================] - 12s 6ms/step - loss: 1.6209 - accuracy: 0.4570 - val\_loss: 0.8691 - val\_accuracy: 0.7522

Epoch 3/10

2000/2000 [==============================] - 12s 6ms/step - loss: 1.0078 - accuracy: 0.6675 - val\_loss: 0.4458 - val\_accuracy: 0.8822

Epoch 4/10

2000/2000 [==============================] - 11s 6ms/step - loss: 0.6751 - accuracy: 0.7725 - val\_loss: 0.3530 - val\_accuracy: 0.8985

Epoch 5/10

2000/2000 [==============================] - 11s 6ms/step - loss: 0.5387 - accuracy: 0.8360 - val\_loss: 0.3306 - val\_accuracy: 0.8962

Epoch 6/10

2000/2000 [==============================] - 11s 6ms/step - loss: 0.4131 - accuracy: 0.8740 - val\_loss: 0.2254 - val\_accuracy: 0.9338

Epoch 7/10

2000/2000 [==============================] - 14s 7ms/step - loss: 0.3188 - accuracy: 0.8995 - val\_loss: 0.1970 - val\_accuracy: 0.9372

Epoch 8/10

2000/2000 [==============================] - 13s 6ms/step - loss: 0.2794 - accuracy: 0.9230 - val\_loss: 0.2063 - val\_accuracy: 0.9388

Epoch 9/10

2000/2000 [==============================] - 12s 6ms/step - loss: 0.2488 - accuracy: 0.9265 - val\_loss: 0.1733 - val\_accuracy: 0.9487

Epoch 10/10

2000/2000 [==============================] - 12s 6ms/step - loss: 0.2203 - accuracy: 0.9325 - val\_loss: 0.1617 - val\_accuracy: 0.9504

The model has successfully trained

Saving the model as mnist.h5

0.9625

(60000, 28, 28) (60000,)

(2250, 28, 28)

Testing with train set size of: (2250, 28, 28)

x\_train shape: (2250, 28, 28, 1)

2250 train samples

57750 test samples

Train on 2250 samples, validate on 57750 samples

Epoch 1/10

2250/2250 [==============================] - 12s 5ms/step - loss: 2.0784 - accuracy: 0.2591 - val\_loss: 1.3878 - val\_accuracy: 0.6773

Epoch 2/10

2250/2250 [==============================] - 12s 5ms/step - loss: 1.2639 - accuracy: 0.5764 - val\_loss: 0.6113 - val\_accuracy: 0.8645

Epoch 3/10

2250/2250 [==============================] - 15s 7ms/step - loss: 0.7735 - accuracy: 0.7462 - val\_loss: 0.4079 - val\_accuracy: 0.8948

Epoch 4/10

2250/2250 [==============================] - 13s 6ms/step - loss: 0.5514 - accuracy: 0.8267 - val\_loss: 0.3666 - val\_accuracy: 0.8918

Epoch 5/10

2250/2250 [==============================] - 15s 7ms/step - loss: 0.4531 - accuracy: 0.8551 - val\_loss: 0.2139 - val\_accuracy: 0.9367

Epoch 6/10

2250/2250 [==============================] - 12s 5ms/step - loss: 0.3345 - accuracy: 0.9036 - val\_loss: 0.1892 - val\_accuracy: 0.9477

Epoch 7/10

2250/2250 [==============================] - 12s 5ms/step - loss: 0.2858 - accuracy: 0.9173 - val\_loss: 0.2192 - val\_accuracy: 0.9341

Epoch 8/10

2250/2250 [==============================] - 12s 5ms/step - loss: 0.2401 - accuracy: 0.9271 - val\_loss: 0.1622 - val\_accuracy: 0.9513

Epoch 9/10

2250/2250 [==============================] - 12s 5ms/step - loss: 0.1986 - accuracy: 0.9444 - val\_loss: 0.1767 - val\_accuracy: 0.9511

Epoch 10/10

2250/2250 [==============================] - 12s 5ms/step - loss: 0.1787 - accuracy: 0.9533 - val\_loss: 0.1419 - val\_accuracy: 0.9594

The model has successfully trained

Saving the model as mnist.h5

0.9583333333333334

(60000, 28, 28) (60000,)

(2500, 28, 28)

Testing with train set size of: (2500, 28, 28)

x\_train shape: (2500, 28, 28, 1)

2500 train samples

57500 test samples

Train on 2500 samples, validate on 57500 samples

Epoch 1/10

2500/2500 [==============================] - 12s 5ms/step - loss: 2.0904 - accuracy: 0.2496 - val\_loss: 1.5259 - val\_accuracy: 0.5297

Epoch 2/10

2500/2500 [==============================] - 14s 6ms/step - loss: 1.2405 - accuracy: 0.5864 - val\_loss: 0.5859 - val\_accuracy: 0.8416

Epoch 3/10

2500/2500 [==============================] - 12s 5ms/step - loss: 0.6781 - accuracy: 0.7816 - val\_loss: 0.4614 - val\_accuracy: 0.8475

Epoch 4/10

2500/2500 [==============================] - 12s 5ms/step - loss: 0.5001 - accuracy: 0.8464 - val\_loss: 0.2488 - val\_accuracy: 0.9262

Epoch 5/10

2500/2500 [==============================] - 12s 5ms/step - loss: 0.3420 - accuracy: 0.8928 - val\_loss: 0.1832 - val\_accuracy: 0.9468

Epoch 6/10

2500/2500 [==============================] - 12s 5ms/step - loss: 0.2895 - accuracy: 0.9196 - val\_loss: 0.1831 - val\_accuracy: 0.9459

Epoch 7/10

2500/2500 [==============================] - 12s 5ms/step - loss: 0.2608 - accuracy: 0.9228 - val\_loss: 0.2180 - val\_accuracy: 0.9382

Epoch 8/10

2500/2500 [==============================] - 12s 5ms/step - loss: 0.2121 - accuracy: 0.9340 - val\_loss: 0.1697 - val\_accuracy: 0.9498

Epoch 9/10

2500/2500 [==============================] - 12s 5ms/step - loss: 0.1756 - accuracy: 0.9508 - val\_loss: 0.1378 - val\_accuracy: 0.9617

Epoch 10/10

2500/2500 [==============================] - 14s 6ms/step - loss: 0.1527 - accuracy: 0.9516 - val\_loss: 0.1642 - val\_accuracy: 0.9551

The model has successfully trained

Saving the model as mnist.h5

0.9541666666666667

(60000, 28, 28) (60000,)

(2750, 28, 28)

Testing with train set size of: (2750, 28, 28)

x\_train shape: (2750, 28, 28, 1)

2750 train samples

57250 test samples

Train on 2750 samples, validate on 57250 samples

Epoch 1/10

2750/2750 [==============================] - 13s 5ms/step - loss: 2.0159 - accuracy: 0.2909 - val\_loss: 1.3250 - val\_accuracy: 0.6320

Epoch 2/10

2750/2750 [==============================] - 12s 5ms/step - loss: 1.2154 - accuracy: 0.5898 - val\_loss: 0.6223 - val\_accuracy: 0.7780

Epoch 3/10

2750/2750 [==============================] - 14s 5ms/step - loss: 0.7296 - accuracy: 0.7611 - val\_loss: 0.3515 - val\_accuracy: 0.8979

Epoch 4/10

2750/2750 [==============================] - 13s 5ms/step - loss: 0.4530 - accuracy: 0.8553 - val\_loss: 0.3681 - val\_accuracy: 0.8848

Epoch 5/10

2750/2750 [==============================] - 13s 5ms/step - loss: 0.3636 - accuracy: 0.8920 - val\_loss: 0.1918 - val\_accuracy: 0.9444

Epoch 6/10

2750/2750 [==============================] - 13s 5ms/step - loss: 0.2782 - accuracy: 0.9182 - val\_loss: 0.2141 - val\_accuracy: 0.9344

Epoch 7/10

2750/2750 [==============================] - 14s 5ms/step - loss: 0.2220 - accuracy: 0.9342 - val\_loss: 0.1676 - val\_accuracy: 0.9499

Epoch 8/10

2750/2750 [==============================] - 13s 5ms/step - loss: 0.1849 - accuracy: 0.9524 - val\_loss: 0.1383 - val\_accuracy: 0.9597

Epoch 9/10

2750/2750 [==============================] - 12s 5ms/step - loss: 0.1731 - accuracy: 0.9498 - val\_loss: 0.1360 - val\_accuracy: 0.9624

Epoch 10/10

2750/2750 [==============================] - 13s 5ms/step - loss: 0.1421 - accuracy: 0.9589 - val\_loss: 0.1401 - val\_accuracy: 0.9602

The model has successfully trained

Saving the model as mnist.h5

0.95

(60000, 28, 28) (60000,)

(3000, 28, 28)

Testing with train set size of: (3000, 28, 28)

x\_train shape: (3000, 28, 28, 1)

3000 train samples

57000 test samples

Train on 3000 samples, validate on 57000 samples

Epoch 1/10

3000/3000 [==============================] - 13s 4ms/step - loss: 2.0605 - accuracy: 0.2633 - val\_loss: 1.3640 - val\_accuracy: 0.5233

Epoch 2/10

3000/3000 [==============================] - 13s 4ms/step - loss: 1.0907 - accuracy: 0.6403 - val\_loss: 0.3936 - val\_accuracy: 0.8988

Epoch 3/10

3000/3000 [==============================] - 14s 5ms/step - loss: 0.5858 - accuracy: 0.8187 - val\_loss: 0.2448 - val\_accuracy: 0.9302

Epoch 4/10

3000/3000 [==============================] - 13s 4ms/step - loss: 0.4116 - accuracy: 0.8747 - val\_loss: 0.1936 - val\_accuracy: 0.9438

Epoch 5/10

3000/3000 [==============================] - 13s 4ms/step - loss: 0.3292 - accuracy: 0.9053 - val\_loss: 0.1569 - val\_accuracy: 0.9537

Epoch 6/10

3000/3000 [==============================] - 13s 4ms/step - loss: 0.2438 - accuracy: 0.9323 - val\_loss: 0.1289 - val\_accuracy: 0.9617

Epoch 7/10

3000/3000 [==============================] - 15s 5ms/step - loss: 0.2155 - accuracy: 0.9397 - val\_loss: 0.1342 - val\_accuracy: 0.9601

Epoch 8/10

3000/3000 [==============================] - 13s 4ms/step - loss: 0.2070 - accuracy: 0.9440 - val\_loss: 0.1304 - val\_accuracy: 0.9625

Epoch 9/10

3000/3000 [==============================] - 13s 4ms/step - loss: 0.1605 - accuracy: 0.9577 - val\_loss: 0.1507 - val\_accuracy: 0.9569

Epoch 10/10

3000/3000 [==============================] - 13s 4ms/step - loss: 0.1696 - accuracy: 0.9520 - val\_loss: 0.1177 - val\_accuracy: 0.9662

The model has successfully trained

Saving the model as mnist.h5

0.9458333333333333

(60000, 28, 28) (60000,)

(3250, 28, 28)

Testing with train set size of: (3250, 28, 28)

x\_train shape: (3250, 28, 28, 1)

3250 train samples

56750 test samples

Train on 3250 samples, validate on 56750 samples

Epoch 1/10

3250/3250 [==============================] - 13s 4ms/step - loss: 1.9252 - accuracy: 0.3169 - val\_loss: 0.9690 - val\_accuracy: 0.7250

Epoch 2/10

3250/3250 [==============================] - 13s 4ms/step - loss: 0.9199 - accuracy: 0.6849 - val\_loss: 0.4239 - val\_accuracy: 0.8675

Epoch 3/10

3250/3250 [==============================] - 13s 4ms/step - loss: 0.5430 - accuracy: 0.8283 - val\_loss: 0.2258 - val\_accuracy: 0.9342

Epoch 4/10

3250/3250 [==============================] - 14s 4ms/step - loss: 0.3588 - accuracy: 0.8948 - val\_loss: 0.2517 - val\_accuracy: 0.9245

Epoch 5/10

3250/3250 [==============================] - 13s 4ms/step - loss: 0.2946 - accuracy: 0.9135 - val\_loss: 0.1521 - val\_accuracy: 0.9559

Epoch 6/10

3250/3250 [==============================] - 13s 4ms/step - loss: 0.2352 - accuracy: 0.9308 - val\_loss: 0.1474 - val\_accuracy: 0.9570

Epoch 7/10

3250/3250 [==============================] - 15s 5ms/step - loss: 0.2024 - accuracy: 0.9425 - val\_loss: 0.1267 - val\_accuracy: 0.9639

Epoch 8/10

3250/3250 [==============================] - 13s 4ms/step - loss: 0.1763 - accuracy: 0.9526 - val\_loss: 0.1253 - val\_accuracy: 0.9647

Epoch 9/10

3250/3250 [==============================] - 13s 4ms/step - loss: 0.1485 - accuracy: 0.9594 - val\_loss: 0.1255 - val\_accuracy: 0.9651

Epoch 10/10

3250/3250 [==============================] - 13s 4ms/step - loss: 0.1437 - accuracy: 0.9594 - val\_loss: 0.1175 - val\_accuracy: 0.9665

The model has successfully trained

Saving the model as mnist.h5

0.9416666666666667

(60000, 28, 28) (60000,)

(3500, 28, 28)

Testing with train set size of: (3500, 28, 28)

x\_train shape: (3500, 28, 28, 1)

3500 train samples

56500 test samples

Train on 3500 samples, validate on 56500 samples

Epoch 1/10

3500/3500 [==============================] - 13s 4ms/step - loss: 1.8775 - accuracy: 0.3454 - val\_loss: 0.9127 - val\_accuracy: 0.7458

Epoch 2/10

3500/3500 [==============================] - 13s 4ms/step - loss: 0.8391 - accuracy: 0.7206 - val\_loss: 0.3837 - val\_accuracy: 0.8894

Epoch 3/10

3500/3500 [==============================] - 15s 4ms/step - loss: 0.4966 - accuracy: 0.8563 - val\_loss: 0.2199 - val\_accuracy: 0.9379

Epoch 4/10

3500/3500 [==============================] - 13s 4ms/step - loss: 0.3393 - accuracy: 0.9020 - val\_loss: 0.1755 - val\_accuracy: 0.9485

Epoch 5/10

3500/3500 [==============================] - 13s 4ms/step - loss: 0.2732 - accuracy: 0.9154 - val\_loss: 0.1834 - val\_accuracy: 0.9467

Epoch 6/10

3500/3500 [==============================] - 13s 4ms/step - loss: 0.2387 - accuracy: 0.9311 - val\_loss: 0.1491 - val\_accuracy: 0.9550

Epoch 7/10

3500/3500 [==============================] - 15s 4ms/step - loss: 0.2082 - accuracy: 0.9431 - val\_loss: 0.1465 - val\_accuracy: 0.9561

Epoch 8/10

3500/3500 [==============================] - 13s 4ms/step - loss: 0.1809 - accuracy: 0.9523 - val\_loss: 0.1613 - val\_accuracy: 0.9513

Epoch 9/10

3500/3500 [==============================] - 13s 4ms/step - loss: 0.1580 - accuracy: 0.9537 - val\_loss: 0.1154 - val\_accuracy: 0.9667

Epoch 10/10

3500/3500 [==============================] - 13s 4ms/step - loss: 0.1295 - accuracy: 0.9620 - val\_loss: 0.1309 - val\_accuracy: 0.9642

The model has successfully trained

Saving the model as mnist.h5

0.9375

(60000, 28, 28) (60000,)

(3750, 28, 28)

Testing with train set size of: (3750, 28, 28)

x\_train shape: (3750, 28, 28, 1)

3750 train samples

56250 test samples

Train on 3750 samples, validate on 56250 samples

Epoch 1/10

3750/3750 [==============================] - 13s 4ms/step - loss: 1.8524 - accuracy: 0.3520 - val\_loss: 0.8080 - val\_accuracy: 0.7885

Epoch 2/10

3750/3750 [==============================] - 13s 4ms/step - loss: 0.7989 - accuracy: 0.7381 - val\_loss: 0.3112 - val\_accuracy: 0.9142

Epoch 3/10

3750/3750 [==============================] - 15s 4ms/step - loss: 0.4518 - accuracy: 0.8675 - val\_loss: 0.2962 - val\_accuracy: 0.9212

Epoch 4/10

3750/3750 [==============================] - 13s 4ms/step - loss: 0.3169 - accuracy: 0.9045 - val\_loss: 0.1884 - val\_accuracy: 0.9421

Epoch 5/10

3750/3750 [==============================] - 13s 4ms/step - loss: 0.2485 - accuracy: 0.9312 - val\_loss: 0.1679 - val\_accuracy: 0.9521

Epoch 6/10

3750/3750 [==============================] - 13s 4ms/step - loss: 0.2333 - accuracy: 0.9328 - val\_loss: 0.1643 - val\_accuracy: 0.9533

Epoch 7/10

3750/3750 [==============================] - 15s 4ms/step - loss: 0.1865 - accuracy: 0.9456 - val\_loss: 0.1217 - val\_accuracy: 0.9630

Epoch 8/10

3750/3750 [==============================] - 13s 4ms/step - loss: 0.1499 - accuracy: 0.9544 - val\_loss: 0.1115 - val\_accuracy: 0.9674

Epoch 9/10

3750/3750 [==============================] - 13s 4ms/step - loss: 0.1380 - accuracy: 0.9581 - val\_loss: 0.1078 - val\_accuracy: 0.9686

Epoch 10/10

3750/3750 [==============================] - 14s 4ms/step - loss: 0.1231 - accuracy: 0.9627 - val\_loss: 0.1088 - val\_accuracy: 0.9686

The model has successfully trained

Saving the model as mnist.h5

0.9333333333333333

(60000, 28, 28) (60000,)

(4000, 28, 28)

Testing with train set size of: (4000, 28, 28)

x\_train shape: (4000, 28, 28, 1)

4000 train samples

56000 test samples

Train on 4000 samples, validate on 56000 samples

Epoch 1/10

4000/4000 [==============================] - 13s 3ms/step - loss: 1.8822 - accuracy: 0.3277 - val\_loss: 0.9661 - val\_accuracy: 0.7822

Epoch 2/10

4000/4000 [==============================] - 13s 3ms/step - loss: 0.8389 - accuracy: 0.7297 - val\_loss: 0.4795 - val\_accuracy: 0.8451

Epoch 3/10

4000/4000 [==============================] - 15s 4ms/step - loss: 0.4317 - accuracy: 0.8685 - val\_loss: 0.3064 - val\_accuracy: 0.9028

Epoch 4/10

4000/4000 [==============================] - 13s 3ms/step - loss: 0.3220 - accuracy: 0.9028 - val\_loss: 0.1766 - val\_accuracy: 0.9444

Epoch 5/10

4000/4000 [==============================] - 13s 3ms/step - loss: 0.2371 - accuracy: 0.9277 - val\_loss: 0.1655 - val\_accuracy: 0.9486

Epoch 6/10

4000/4000 [==============================] - 13s 3ms/step - loss: 0.2168 - accuracy: 0.9383 - val\_loss: 0.1603 - val\_accuracy: 0.9526

Epoch 7/10

4000/4000 [==============================] - 15s 4ms/step - loss: 0.1896 - accuracy: 0.9442 - val\_loss: 0.1128 - val\_accuracy: 0.9672

Epoch 8/10

4000/4000 [==============================] - 13s 3ms/step - loss: 0.1604 - accuracy: 0.9550 - val\_loss: 0.1494 - val\_accuracy: 0.9560

Epoch 9/10

4000/4000 [==============================] - 14s 3ms/step - loss: 0.1468 - accuracy: 0.9575 - val\_loss: 0.1056 - val\_accuracy: 0.9705

Epoch 10/10

4000/4000 [==============================] - 15s 4ms/step - loss: 0.1318 - accuracy: 0.9613 - val\_loss: 0.1089 - val\_accuracy: 0.9689

The model has successfully trained

Saving the model as mnist.h5

0.9291666666666667

(60000, 28, 28) (60000,)

(4250, 28, 28)

Testing with train set size of: (4250, 28, 28)

x\_train shape: (4250, 28, 28, 1)

4250 train samples

55750 test samples

Train on 4250 samples, validate on 55750 samples

Epoch 1/10

4250/4250 [==============================] - 14s 3ms/step - loss: 1.8046 - accuracy: 0.3800 - val\_loss: 0.9105 - val\_accuracy: 0.6897

Epoch 2/10

4250/4250 [==============================] - 14s 3ms/step - loss: 0.7555 - accuracy: 0.7433 - val\_loss: 0.3631 - val\_accuracy: 0.8833

Epoch 3/10

4250/4250 [==============================] - 16s 4ms/step - loss: 0.4446 - accuracy: 0.8605 - val\_loss: 0.1897 - val\_accuracy: 0.9443

Epoch 4/10

4250/4250 [==============================] - 14s 3ms/step - loss: 0.3066 - accuracy: 0.9108 - val\_loss: 0.2721 - val\_accuracy: 0.9186

Epoch 5/10

4250/4250 [==============================] - 14s 3ms/step - loss: 0.2480 - accuracy: 0.9271 - val\_loss: 0.1407 - val\_accuracy: 0.9596

Epoch 6/10

4250/4250 [==============================] - 14s 3ms/step - loss: 0.1974 - accuracy: 0.9435 - val\_loss: 0.1570 - val\_accuracy: 0.9545

Epoch 7/10

4250/4250 [==============================] - 15s 4ms/step - loss: 0.1712 - accuracy: 0.9480 - val\_loss: 0.1651 - val\_accuracy: 0.9546

Epoch 8/10

4250/4250 [==============================] - 13s 3ms/step - loss: 0.1525 - accuracy: 0.9555 - val\_loss: 0.1085 - val\_accuracy: 0.9691

Epoch 9/10

4250/4250 [==============================] - 14s 3ms/step - loss: 0.1338 - accuracy: 0.9621 - val\_loss: 0.1207 - val\_accuracy: 0.9660

Epoch 10/10

4250/4250 [==============================] - 16s 4ms/step - loss: 0.1292 - accuracy: 0.9642 - val\_loss: 0.1087 - val\_accuracy: 0.9707

The model has successfully trained

Saving the model as mnist.h5

0.925

(60000, 28, 28) (60000,)

(4500, 28, 28)

Testing with train set size of: (4500, 28, 28)

x\_train shape: (4500, 28, 28, 1)

4500 train samples

55500 test samples

Train on 4500 samples, validate on 55500 samples

Epoch 1/10

4500/4500 [==============================] - 14s 3ms/step - loss: 1.8119 - accuracy: 0.3649 - val\_loss: 0.9295 - val\_accuracy: 0.7055

Epoch 2/10

4500/4500 [==============================] - 14s 3ms/step - loss: 0.7687 - accuracy: 0.7507 - val\_loss: 0.3274 - val\_accuracy: 0.9008

Epoch 3/10

4500/4500 [==============================] - 16s 3ms/step - loss: 0.4379 - accuracy: 0.8678 - val\_loss: 0.1899 - val\_accuracy: 0.9449

Epoch 4/10

4500/4500 [==============================] - 14s 3ms/step - loss: 0.3295 - accuracy: 0.9013 - val\_loss: 0.2858 - val\_accuracy: 0.9091

Epoch 5/10

4500/4500 [==============================] - 14s 3ms/step - loss: 0.2615 - accuracy: 0.9300 - val\_loss: 0.1331 - val\_accuracy: 0.9606

Epoch 6/10

4500/4500 [==============================] - 14s 3ms/step - loss: 0.1834 - accuracy: 0.9456 - val\_loss: 0.1262 - val\_accuracy: 0.9627

Epoch 7/10

4500/4500 [==============================] - 15s 3ms/step - loss: 0.1727 - accuracy: 0.9500 - val\_loss: 0.1284 - val\_accuracy: 0.9615

Epoch 8/10

4500/4500 [==============================] - 14s 3ms/step - loss: 0.1571 - accuracy: 0.9540 - val\_loss: 0.2150 - val\_accuracy: 0.9403

Epoch 9/10

4500/4500 [==============================] - 13s 3ms/step - loss: 0.1340 - accuracy: 0.9638 - val\_loss: 0.1028 - val\_accuracy: 0.9721

Epoch 10/10

4500/4500 [==============================] - 15s 3ms/step - loss: 0.1184 - accuracy: 0.9649 - val\_loss: 0.1144 - val\_accuracy: 0.9668

The model has successfully trained

Saving the model as mnist.h5

0.9208333333333334

(60000, 28, 28) (60000,)

(4750, 28, 28)

Testing with train set size of: (4750, 28, 28)

x\_train shape: (4750, 28, 28, 1)

4750 train samples

55250 test samples

Train on 4750 samples, validate on 55250 samples

Epoch 1/10

4750/4750 [==============================] - 14s 3ms/step - loss: 1.7100 - accuracy: 0.3994 - val\_loss: 0.6681 - val\_accuracy: 0.7893

Epoch 2/10

4750/4750 [==============================] - 14s 3ms/step - loss: 0.6823 - accuracy: 0.7827 - val\_loss: 0.5081 - val\_accuracy: 0.8489

Epoch 3/10

4750/4750 [==============================] - 15s 3ms/step - loss: 0.4002 - accuracy: 0.8872 - val\_loss: 0.1572 - val\_accuracy: 0.9546

Epoch 4/10

4750/4750 [==============================] - 14s 3ms/step - loss: 0.2640 - accuracy: 0.9280 - val\_loss: 0.1512 - val\_accuracy: 0.9573

Epoch 5/10

4750/4750 [==============================] - 14s 3ms/step - loss: 0.2310 - accuracy: 0.9373 - val\_loss: 0.1166 - val\_accuracy: 0.9662

Epoch 6/10

4750/4750 [==============================] - 16s 3ms/step - loss: 0.1859 - accuracy: 0.9482 - val\_loss: 0.1102 - val\_accuracy: 0.9674

Epoch 7/10

4750/4750 [==============================] - 14s 3ms/step - loss: 0.1730 - accuracy: 0.9516 - val\_loss: 0.1063 - val\_accuracy: 0.9692

Epoch 8/10

4750/4750 [==============================] - 14s 3ms/step - loss: 0.1377 - accuracy: 0.9596 - val\_loss: 0.1049 - val\_accuracy: 0.9692

Epoch 9/10

4750/4750 [==============================] - 14s 3ms/step - loss: 0.1190 - accuracy: 0.9657 - val\_loss: 0.0934 - val\_accuracy: 0.9730

Epoch 10/10

4750/4750 [==============================] - 16s 3ms/step - loss: 0.1072 - accuracy: 0.9682 - val\_loss: 0.2269 - val\_accuracy: 0.9423

The model has successfully trained

Saving the model as mnist.h5

0.9166666666666666

(60000, 28, 28) (60000,)

(5000, 28, 28)

Testing with train set size of: (5000, 28, 28)

x\_train shape: (5000, 28, 28, 1)

5000 train samples

55000 test samples

Train on 5000 samples, validate on 55000 samples

Epoch 1/10

5000/5000 [==============================] - 14s 3ms/step - loss: 1.8116 - accuracy: 0.3558 - val\_loss: 1.3767 - val\_accuracy: 0.4889

Epoch 2/10

5000/5000 [==============================] - 16s 3ms/step - loss: 0.7656 - accuracy: 0.7548 - val\_loss: 0.3476 - val\_accuracy: 0.9034

Epoch 3/10

5000/5000 [==============================] - 14s 3ms/step - loss: 0.4259 - accuracy: 0.8680 - val\_loss: 0.2524 - val\_accuracy: 0.9263

Epoch 4/10

5000/5000 [==============================] - 14s 3ms/step - loss: 0.3068 - accuracy: 0.9048 - val\_loss: 0.2304 - val\_accuracy: 0.9275

Epoch 5/10

5000/5000 [==============================] - 14s 3ms/step - loss: 0.2258 - accuracy: 0.9340 - val\_loss: 0.1714 - val\_accuracy: 0.9468

Epoch 6/10

5000/5000 [==============================] - 16s 3ms/step - loss: 0.1964 - accuracy: 0.9368 - val\_loss: 0.1280 - val\_accuracy: 0.9645

Epoch 7/10

5000/5000 [==============================] - 14s 3ms/step - loss: 0.1823 - accuracy: 0.9476 - val\_loss: 0.1065 - val\_accuracy: 0.9692

Epoch 8/10

5000/5000 [==============================] - 14s 3ms/step - loss: 0.1412 - accuracy: 0.9598 - val\_loss: 0.1077 - val\_accuracy: 0.9692

Epoch 9/10

5000/5000 [==============================] - 15s 3ms/step - loss: 0.1297 - accuracy: 0.9602 - val\_loss: 0.1233 - val\_accuracy: 0.9650

Epoch 10/10

5000/5000 [==============================] - 14s 3ms/step - loss: 0.1135 - accuracy: 0.9682 - val\_loss: 0.0984 - val\_accuracy: 0.9728

The model has successfully trained

Saving the model as mnist.h5

{(100, 28, 28, 1): {0.41360601782798767, 1.0, 1.9473908741366683}, (200, 28, 28, 1): {0.5123578310012817, 1.7485677863124223, 2.0}, (300, 28, 28, 1): {0.9859188210025704, 0.7883082032203674, 3.0}, (400, 28, 28, 1): {0.8299333902173395, 0.7592952847480774, 4.0}, (500, 28, 28, 1): {0.6112979846501551, 0.8488907814025879, 5.0}, (600, 28, 28, 1): {0.3970352809035818, 0.9013131260871887, 6.0}, (700, 28, 28, 1): {0.31940235241569115, 0.903642475605011, 7.0}, (800, 28, 28, 1): {8.0, 0.3717542235915725, 0.8868749737739563}, (900, 28, 28, 1): {0.44257667893685665, 9.0, 0.8651099801063538}, (1000, 28, 28, 1): {0.2782689767591024, 0.9149321913719177, 10.0}, (1250, 28, 28, 1): {0.20204347675911924, 0.9416170120239258, 12.5}, (1500, 28, 28, 1): {0.1946935737489635, 0.9437435865402222, 15.0}, (1750, 28, 28, 1): {0.1944182147186873, 17.5, 0.9456309080123901}, (2000, 28, 28, 1): {0.1616910392809017, 0.9504482746124268, 20.0}, (2250, 28, 28, 1): {0.14191391077379645, 0.9594285488128662, 22.5}, (2500, 28, 28, 1): {0.16420171630298314, 25.0, 0.9550782442092896}, (2750, 28, 28, 1): {0.14014463707147418, 0.9602096080780029, 27.5}, (3000, 28, 28, 1): {0.11771224454435798, 0.9662280678749084, 30.0}, (3250, 28, 28, 1): {32.5, 0.966519832611084, 0.11747534622235577}, (3500, 28, 28, 1): {0.13085077676460544, 0.9642124176025391, 35.0}, (3750, 28, 28, 1): {0.10876199407554335, 0.9685510993003845, 37.5}, (4000, 28, 28, 1): {40.0, 0.10887770517852291, 0.9689106941223145}, (4250, 28, 28, 1): {0.10870982873496796, 0.9707264304161072, 42.5}, (4500, 28, 28, 1): {0.11442138385119101, 0.9667567610740662, 45.0}, (4750, 28, 28, 1): {0.22694782386888745, 0.9423348307609558, 47.5}, (5000, 28, 28, 1): {0.09841099160335619, 0.9728000164031982, 50.0}}

The three main things that are part of this step are the variables summary\_values, training\_sets and the for loop. The variable summary\_values is a dictionary variable that allows for the ending data for each training size to be collected and stored safely. This variable is printed at the end to allow for easy data collection. The variable training\_sets is used in conjunction with the for loop to iterate through each amount of training images. The variable test\_per is being iterated with each value in the array training\_sets and is converted into a percentage to be put into the function load\_mnist\_data and specify what percent of the images are training. The results of the project are displayed above, and the data table and graphs are made and printed down below