

```
model.add(Conv2D(64,(3,3),activation='relu',input_shape=(28,28,1)))
model.add(MaxPool2D((2,2)))
model.add(Conv2D(64,(3,3),activation='relu',input_shape=(28,28,1)))
model.add(MaxPool2D((2,2)))
```

In [10]:

```
model.summary()
```

rohit58405840@gmail.com

Layer (type)	Output Shape	Param #
conv2d_5 (Conv2D)	(None, 26, 26, 32)	320
max_pooling2d_5 (MaxPooling2)	(None, 13, 13, 32)	0
conv2d_6 (Conv2D)	(None, 11, 11, 64)	18496
max_pooling2d_6 (MaxPooling2)	(None, 5, 5, 64)	0
conv2d_7 (Conv2D)	(None, 3, 3, 64)	36928
max_pooling2d_7 (MaxPooling2)	(None, 1, 1, 64)	0

Total params: 55,744

General Trend

M, W ↓
C ↑

Simpler Data

↓
[1055 Params]

$5 - 3 + 1$
1

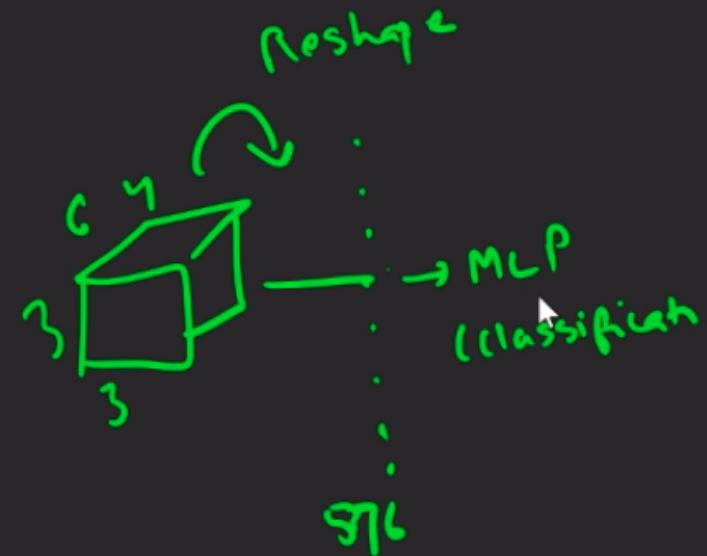
128

55K

```
model.add(MaxPool2D((2,2)))
model.add(Conv2D(64,(3,3),activation='relu',input_shape=(28,28,1)))
model.add(Flatten())
```

```
In [14]: model.summary()
```

Layer (type)	Output Shape	Param #
=====		
conv2d_11 (Conv2D)	(None, 26, 26, 32)	320
<hr/>		
max_pooling2d_10 (MaxPooling)	(None, 13, 13, 32)	0
<hr/>		
conv2d_12 (Conv2D)	(None, 11, 11, 64)	18496
<hr/>		
max_pooling2d_11 (MaxPooling)	(None, 5, 5, 64)	0
<hr/>		
conv2d_13 (Conv2D)	(None, 3, 3, 64)	36928
<hr/>		
flatten_1 (Flatten)	(None, 576)	0
=====		
Total params: 55,744		
Trainable params: 55,744		
Non-trainable params: 0		



```

X = X/255.0
Y = to_categorical(Y)
return X,Y

```

```

In [21]: XTrain,YTrain = preprocess_data(XTrain,YTrain)
print(XTrain.shape,YTrain.shape)

```

(60000, 28, 28, 1) (60000, 10)

```

In [22]: XTest,YTest = preprocess_data(XTest,YTest)
print(XTest.shape,YTest.shape)

```

(10000, 28, 28, 1) (10000, 10)

```

In [24]: # Compile
model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])

```

In []:

$$w = w - \eta \Delta w$$

Diagram illustrating the optimization process:

Initial weights w are updated to w' using the learning rate η and the gradient Δw . The loss function is shown as $Loss = 0.2 \cdot 0 + 0.5 \cdot 1 + 0.3 \cdot 0$. The optimization process is labeled "optimizer" and "90%".