Add Dense Layers

Dense layers

The name suggests that layers are fully connected (dense) by the neurons in a network layer. Each neuron in a layer receives input from all the neurons present in the previous layer. Dense is used to add the layers.

Adding Hidden layers

This step is to add a dense layer (hidden layer). We flatten the feature map and convert it into a vector or single dimensional array in the Flatten layer. This vector array is fed it as an input to the neural network and applies an activation function, such as sigmoid or other, and returns the output.

- init is the weight initialization; function which sets all the weights and biases of a network to values suitable as a starting point for training.
- units/ output_dim, which denote is the number of neurons in the hidden layer.
- The activation function basically decides to deactivate neurons or activate them to get the desired output. It also performs a nonlinear transformation on the input to get better results on a complex neural network.
- You can add many hidden layers, in our project we are added two hidden layers. The 1st hidden layer with 40 neurons and 2nd hidden layer with 20neurons.

Adding the output layer

This step is to add a dense layer (output layer) where you will be specifying the number of classes your dependent variable has, activation function, and weight initializer as the arguments. We use the add () method to add dense layers. The output dimensions here is 6.

model.add(Dense(40, 'relu'))

model.add(Dense(20, 'relu'))

model.add(Dense(6, 'softmax',))

If you have only one or two classes in the output layer, assign “units= 1” and “activation = sigmoid”. If you have more than two classes (let's assume 3) then assign “units / output_dim = 3” and “activation = softmax”.In this project we are using **softmax.**

In [8]:
```
x_train= train_datagen.flow_from_directory('/content/drive/MyDrive/fruit-dataset/fruit-dataset/train',batch_size=32,target_size=(128,128),
                                           color_mode='rgb',class_mode='categorical')
x_test = test_datagen.flow_from_directory('/content/drive/MyDrive/fruit-dataset/fruit-dataset/test',batch_size=32,target_size=(128,128),
                                           color_mode='rgb',class_mode='categorical')
```

```
Found 5384 images belonging to 6 classes.
Found 1686 images belonging to 6 classes.
```

In [9]:
```
from tensorflow.keras.utils import Sequence
```

Initializing The Model

In [10]:
```
model=Sequential()
```

ADD CNNLayers

In [11]:
```
model.add(Convolution2D(32,(3,3),input_shape = (128,128,3),activation = 'relu'))
model.add(MaxPooling2D(pool_size = (2,2)))
model.add(Flatten())
```

Add Dense Layers

In [12]:
```
model.add(Dense(40,  'relu'))
model.add(Dense(20, 'relu'))
model.add(Dense(6,  'softmax', ))
```