

Basically **Neural networks** are modeled after our brains. There are individual nodes that form the layers in the network, just like the neurons in our brains connect different areas.

The inputs to nodes in a single layer will have a weight assigned to them that changes the effect that parameter has on the overall prediction result. Since the weights are assigned on the links between nodes, each node maybe influenced by multiple weights.

The neural network takes all of the training data in the input layer. Then it passes the data through the hidden layers, transforming the values based on the weights at each node. Finally it returns a value in the output layer.

It can take some time to properly tune a neural network to get consistent, reliable results. Testing and training your neural network is a balancing process between deciding what features are the most important to your model.

A convolutional neural network is a specific kind of neural network with multiple layers. It processes data that has a grid-like arrangement then extracts important features. One huge **advantage** of using CNNs is that you **don't** need to do **a lot of pre-processing** on images.

✓ **CNN is Convolutional Neural Network.** It is a type of deep learning algorithm designed specifically for working with **grid-like data, such as images.**

A **big difference between a CNN and a regular neural network** is that CNNs use convolutions to handle the math behind the scenes. A convolution is used instead of matrix multiplication in at least one layer of the CNN. Convolutions take two functions and return a function.

The convolutional neural network algorithm's **main purpose** is to get data into forms that are easier to process without losing the features that are important for figuring out what the data represents. This also makes them great candidates for handling huge datasets.

Key Components of CNN:

Convolutional Layers:

Filters/Kernels: Small matrices (e.g., 3x3, 5x5) that slide over the input data to detect patterns or features.

Feature Maps: The output of convolution operations, highlighting features like edges, textures, and more.

Activation Functions:

Often use ReLU (Rectified Linear Unit) to introduce non-linearity. ReLU replaces negative values with zero. Softmax,

Pooling Layers:

Max Pooling: Downsamples the input by taking the maximum value in a region (e.g., 2x2).

Average Pooling: Downsamples by taking the average value in a region.

Fully Connected Layers:

Dense layers where each neuron is connected to all neurons in the previous layer. These layers make decisions based on the extracted features.

Flatten Layer:

Converts the 2D matrix data into a 1D vector to feed into the fully connected layers.

✓ Working Principle:

Input Layer:

An image is inputted into the CNN.

Convolutional Layers:

The image is passed through convolutional layers to extract features.

Pooling Layers:

Reduce the dimensions while retaining important features.

Flatten Layer:


Converts the pooled feature maps into a single vector.

Fully Connected Layers:

Process the vector to produce the final output (e.g., classification).

```
from keras.models import Sequential
from keras.layers import Conv2D, MaxPooling2D, Flatten, Dense

model = Sequential()
# Convolutional layer with 32 filters of size 3x3
model.add(Conv2D(32, (3, 3), activation='relu', input_shape=(64, 64, 3)))
# Max pooling layer with pool size of 2x2
model.add(MaxPooling2D(pool_size=(2, 2)))
# Flatten layer
model.add(Flatten())
# Fully connected layer
model.add(Dense(units=128, activation='relu'))
# Output layer
model.add(Dense(units=10, activation='softmax'))
```

 /usr/local/lib/python3.10/dist-packages/keras/src/layers/convolutional/base_conv.py:107: UserWarning: Do not pass an `in`
super().__init__(activity_regularizer=activity_regularizer, **kwargs)

In this example:

Conv2D: Convolutional layer with 32 filters of size 3x3.

MaxPooling2D: Pooling layer with a pool size of 2x2.

Flatten: Flattens the pooled feature maps.

Dense: Fully connected layers with 128 and 10 neurons, respectively.