ADVANCED ACADEMIC CENTER

A CENTER FOR INTER-DISCIPLINARY RESEARCH
2019-20

TITLE

X RAY IMAGE PROCESSING THROUGH CNN

SUPERVISED BY

LOKESH KONJETI

GOKARAJU RANGARAJU
INSTITUTE OF ENGINEERING AND TECHNOLOGY
AUTONOMOUS

Advanced Academic Center

(A Center For Inter-Discilinary Research)

This is to certify that the project titled

"X RAY IMAGE PROCESSING THROUGH CNN"

is a Bonafide work carried out by the following students in partial fulfilment of the requirements for Advanced Academic Center intern, submitted to the chair, AAC during the academic year **2019-2020**

NAME	ROLL NO.	BRANCH
KANNAN VIJAYARAGHAVAN	19241A05V9	CSE
MAHAJAN VAMSHI SAI	19241A12F8	IT
B V S NITHIN	19241A04C8	ECE
BANALA VISWA SAHITH	19241A04O9	ECE
RAGIPANI SAI RAMANA	19241A05G2	CSE

This work was not submitted or published earlier for any study

Dr. /Ms./Mr.	Dr.B.R.K.Reddy	Dr.Ramamurthy Suri
Project Supervisor	Program Coordinator	Associate Dean,AAC

ADVANCED ACADEMIC CENTER

ACKNOWLEDGEMENTS

We express our deep sense of gratitude to our respected Director , Gokaraju Rangaraju Institute of Engineering and Technology for the valuable guidance and for permitting us to carry out this project.

With immense pleasure, we record our deep sense of gratitude to our respected principal, for permitting us to carry out this project.

We are thankful to Associate Dean ,Advance Academic Centre for providing us appropriate ecosystem required for the project to complete.

We are thankful to our project supervisor who spared valuable time for us and influence novel ideas to guide us. I am indebted to all the above without whom I would not have concluded the project.

TABLES OF CONTENTS

ABSTRACT

The field of machine learning has taken a dramatic twist in recent times, with the rise of the Artificial Neural Network (ANN). These biologically inspired computational models are able to far exceed the performance of previous forms of artificial intelligence in common machine learning tasks. One of the most impressive forms of ANN architecture is that of the Convolutional Neural Network (CNN). CNNs are primarily used to solve difficult image-driven pattern recognition tasks and with their precise yet simple architecture, offers a simplified method of getting started with ANNs. This document provides a brief introduction to CNNs, discussing recently published papers and newly formed techniques in developing these brilliantly fantastic image recognition models. This introduction assumes you are familiar with the fundamentals of ANNs and machine learning.

ARTIFICAL INTILLEGENCE

In computer science, **artificial intelligence** (**AI**), sometimes called **machine intelligence**, is intelligence demonstrated by machines,in contrast to the **natural intelligence** displayed by humans.
Colloquially, the term "artificial intelligence" is often used to describe machines (or computers) that mimic "cognitive" functions that humans associate with the human mind, such as "learning“ and "problem solving".

Computer science defines AI research as the study of "intelligent agents": any device that perceives its environment and takes actions that maximize its chance of successfully achieving its goals.A more elaborate definition characterizes AI as “a system’s ability to correctly interpret external data, to learn from such data, and to use those learnings to achieve specific goals and tasks through flexible adaptation.”

Powerful Examples Of AI Applications:

1- Automated customer support
2- Healthcare
3- Smart cars and drones
4- Travel and navigation
5- Social media
6- Smart home devices
7- Creative arts

8- Security and surveillance

NEURAL NETWORKS

A neural network is a series of algorithms that endeavors to recognize underlying relationships in a set of data through a process that mimics the way the human brain operates.

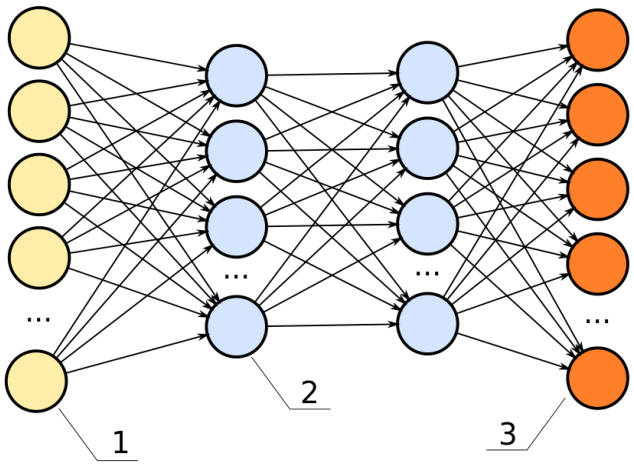
Neural networks can adapt to changing input; so the network generates the best possible result without needing to redesign the output criteria. The concept of neural networks, which has its roots in artificial intelligence, is swiftly gaining popularity in the development of trading systems.

Basics of Neural Networks:

A neural network works similarly to the human brain's neural network. A "neuron" in a neural network is a mathematical function that collects and classifies information according to a specific architecture. The network bears a strong resemblance to statistical methods such as curve fitting and regression analysis.

A neural network contains layers of interconnected nodes. Each node is a perceptron and is similar to a multiple linear regression. The perceptron feeds the signal produced by a multiple linear regression into an activation function that may be nonlinear.

In a multi-layered perceptron (MLP), perceptrons are arranged in interconnected layers. The input layer collects input patterns. The output layer has classifications or output signals to which input patterns may map.
For instance, the patterns may comprise a list of quantities for technical indicators about a security; potential outputs could be “buy,” “hold” or “sell.”



hidden layers fine-tune the input weightings until the neural network’s margin of error is minimal. It is hypothesized that hidden layers extrapolate salient features in the input data that have predictive power regarding the outputs. This describes feature extraction, which accomplishes a utility similar to statistical techniques such as principal component analysis.

Application of Neural Networks:

Neural networks are broadly used, with applications for financial operations, enterprise planning, trading, business analytics and product maintenance. Neural networks

have also gained widespread adoption in business applications such as forecasting and marketing research solutions

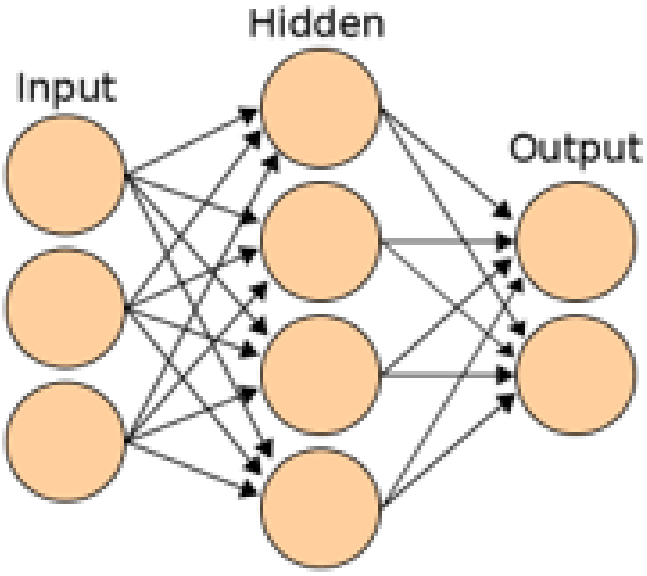
CONVOLUTION NEURAL NETWORK

In deep learning, a convolutional neural network (CNN, or ConvNet) is a class of deep neural networks, most commonly applied to analyzing visual imagery.

CNN's are used to classify image data and predict the future outcome of each of these image categories i.e If there are two categories Dogs and Cats, One has to train a CNN on the given images of Dogs and Cats then the goal is to create a convolutional neural network that learns on the image data and uses this knowledge to predict the future images. Which means that when a image of either a dog or a cat is passed to the algorithm then based on the accquired knowledge it will predict weather the image is of a dog or a cat.

A CNN is noting but a deep neural network, consisting of nodes, weights and layers of the network that work as the neurons in the brain which in turn make the algorithm function like a human brain to infer and learn from data.
A convolutional neural network consists of an input and an output layer, as well as multiple hidden layers. The hidden layers of a CNN typically consist of a series of convolutional layers that *convolve* with a multiplication or other dot product

NODES



An artificial neural network is an interconnected group of nodes, inspired by a simplification of neurons in a brain. Here, each circular node represents an artificial neuron and an arrow represents a connection from the output of one artificial neuron to the input of another.

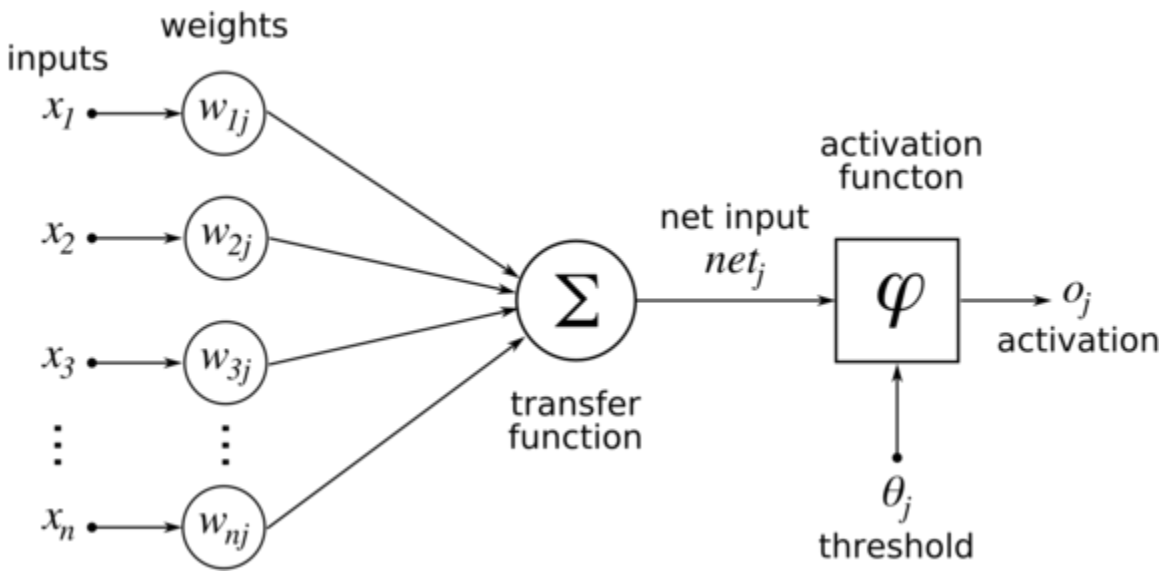
A Node ,also called a neuron,is a computational unit that has One or more weighted input connections,a transfer function that combines the input in some way,and an output connections

Nodes are then oraganized into layers to comprise a network. In general ,we cannot calculate the number of nodes to use per layer in an artifical neural network.We must specify the number of nodes that we are using in a layer.

In a fully connected network each neuron will be associated with many different weights. If there are n0 inputs (i.e. n0neurons in the previous layer) to a layer with n1 neurons in a fully connected network, that layer will have n0*n1 weights, not counting any bias term.
Every edge you see represents a different trainable weight

Convolutional layers are different in that they have a fixed number of weights governed by the choice of filter size and number of filters, but independent of the input size.

Connections in a artifical neural network:



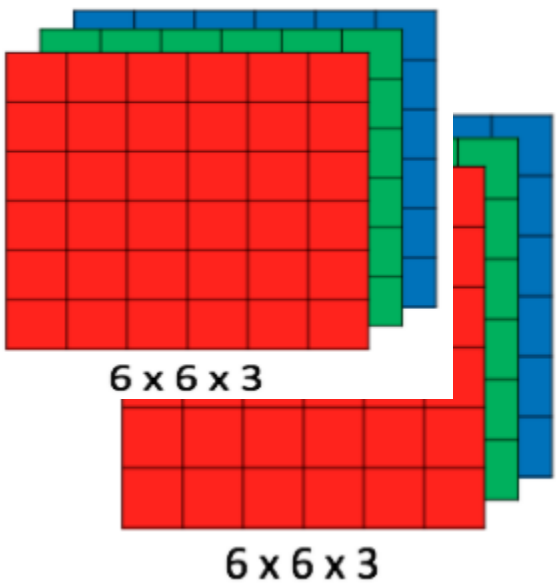
Summation of our inputs multiplied with weights give us a transfer function .These neural network forms layers in

CNN.

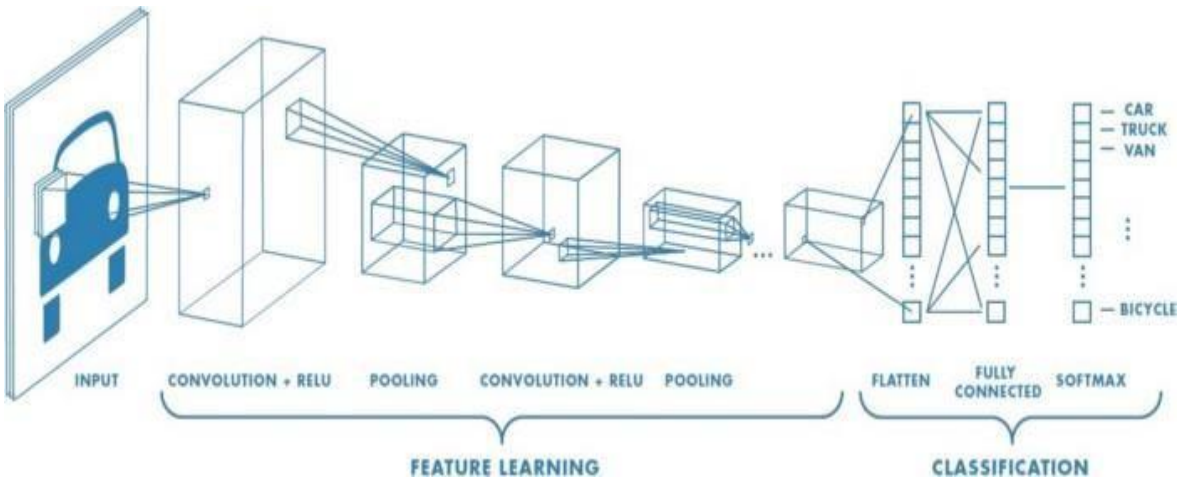
Layers in CNN

CNN image classifications takes an input image, process it and classify it under certain categories (Eg., Dog, Cat, Tiger, Lion). Computers sees an input image as array of pixels and it depends on the image resolution. Based on the image resolution, it will see h x w x d(h = Height, w = Width, d = Dimension).

Eg., An image of 6 x 6 x 3 array of matrix of RGB (3 refers to RGB values) and an image of 4 x 4 x 1 array of matrix of grayscale image.



Technically, deep learning CNN models to train and test, each input image will pass it through a series of convolution layers with filters (Kernals), Pooling, fully connected layers (FC) and apply Softmax function to classify an object with probabilistic values between 0 and 1. The below figure is a complete flow of CNN to process an input image and classifies the objects based on values.



Neural network with many convolutional layers

Convolution Layer

Convolution is the first layer to extract features from an input image. Convolution preserves the relationship between pixels by learning image features using small squares of input data. It is a mathematical operation that takes two inputs such as image matrix and a filter or kernal

Consider a 5 x 5 whose image pixel values are 0, 1 and filter matrix 3 x 3 as shown in below

1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

5 x 5 – Image Matrix

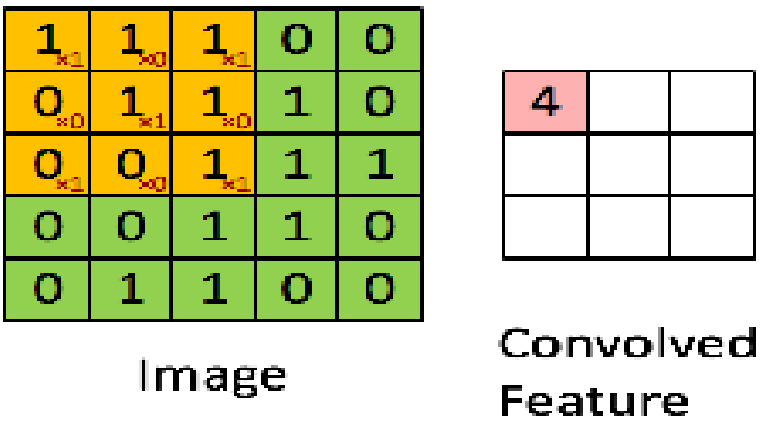
*

1	0	1
0	1	0
1	0	1

3 x 3 – Filter Matrix

Image matrix multiplies kernel or filter matrix

Then the convolution of 5 x 5 image matrix multiplies with 3 x 3 filter matrix which is called **"Feature Map"** as output shown in below



The ouput of convolution layer acts as input for next layer, that is pooling layer.

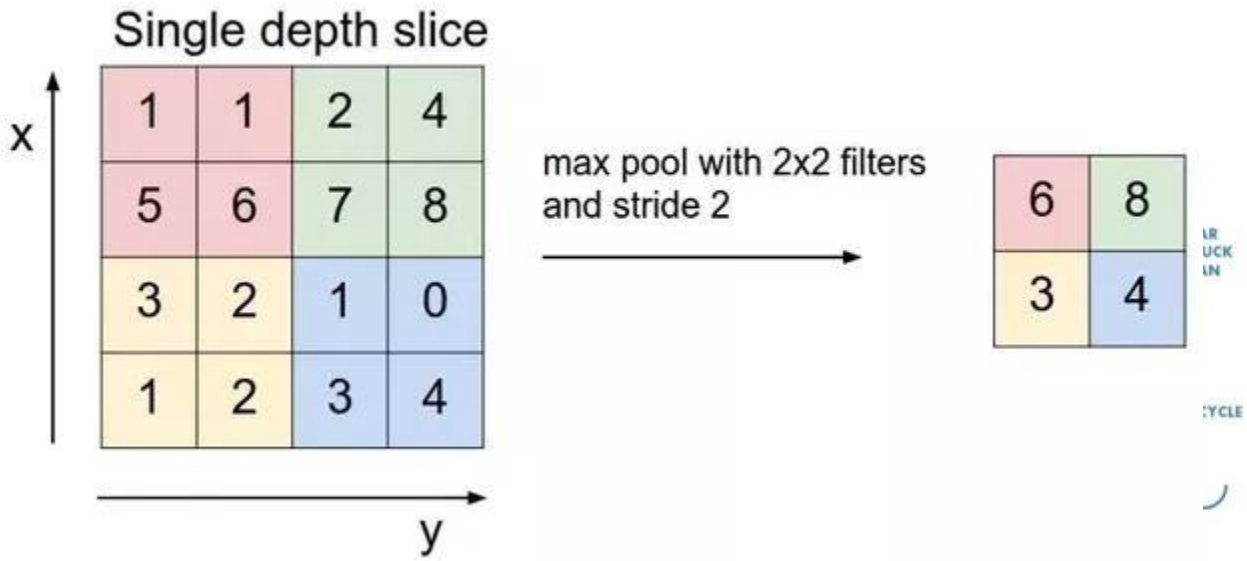
Pooling Layer:

Pooling layers section would reduce the number of parameters when the images are too large. Spatial pooling also called subsampling or downsampling which reduces the dimensionality of each map but retains the important information.

Spatial pooling can be of different types:

Max Pooling
Average Pooling
Sum Pooling

Max pooling take the largest element from the rectified feature map. Taking the largest element could also take the average pooling. Sum of all elements in the feature map call as sum pooling.

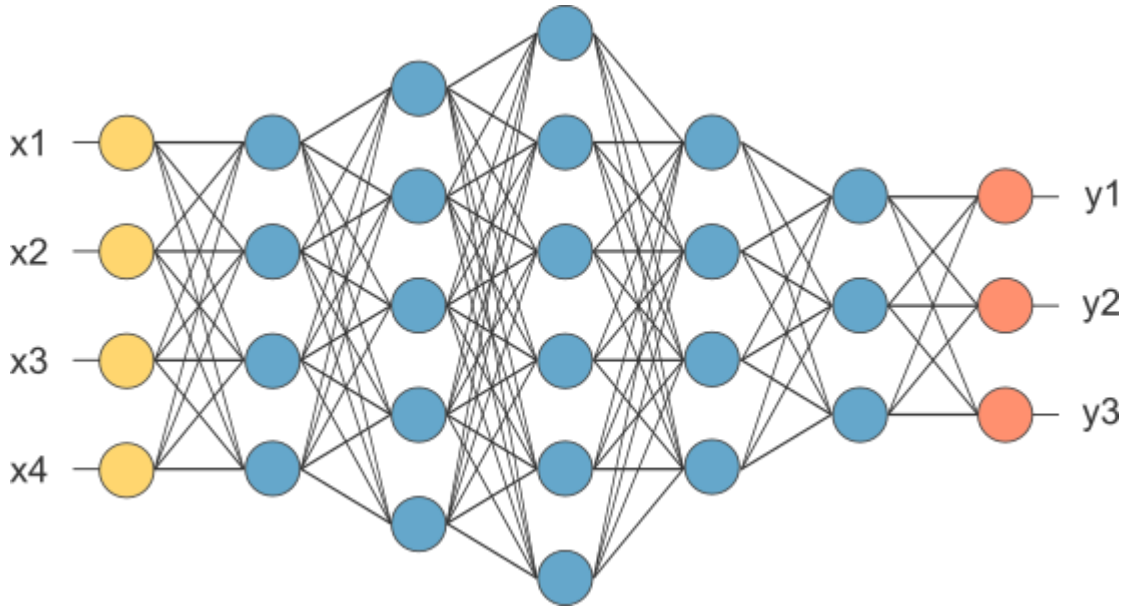


Max Pooling

Fully Connected Layer

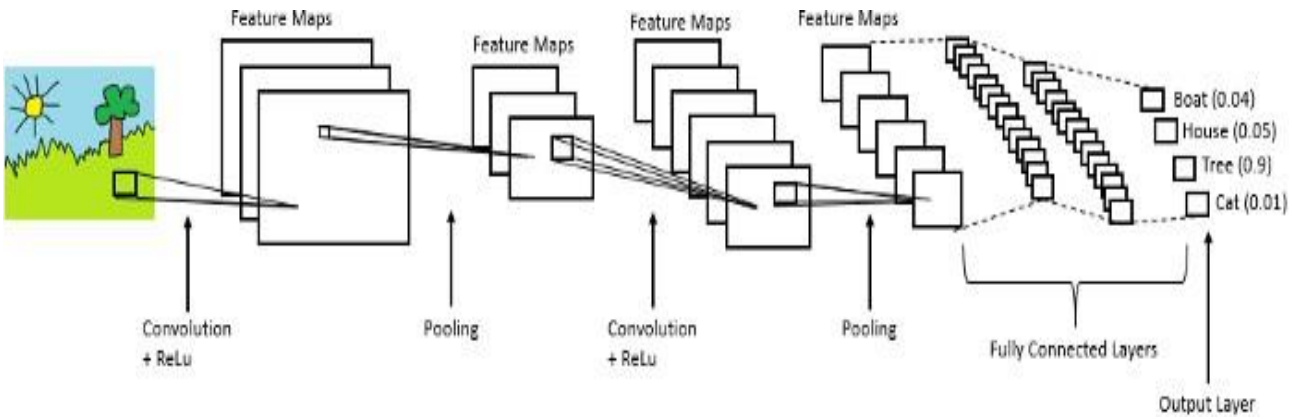
The layer we call as FC layer, we flattened our matrix into vector and feed it into a fully connected layer like neural

network.



After pooling layer, flattened as FC layer

In the above diagram, feature map matrix will be converted as vector (x1, x2, x3, …). With the fully connected layers, we combined these features together to create a model. Finally, we have an activation function such as softmax or sigmoid to classify the outputs as cat, dog, car, truck etc.



Complete CNN architecture

Activation functions

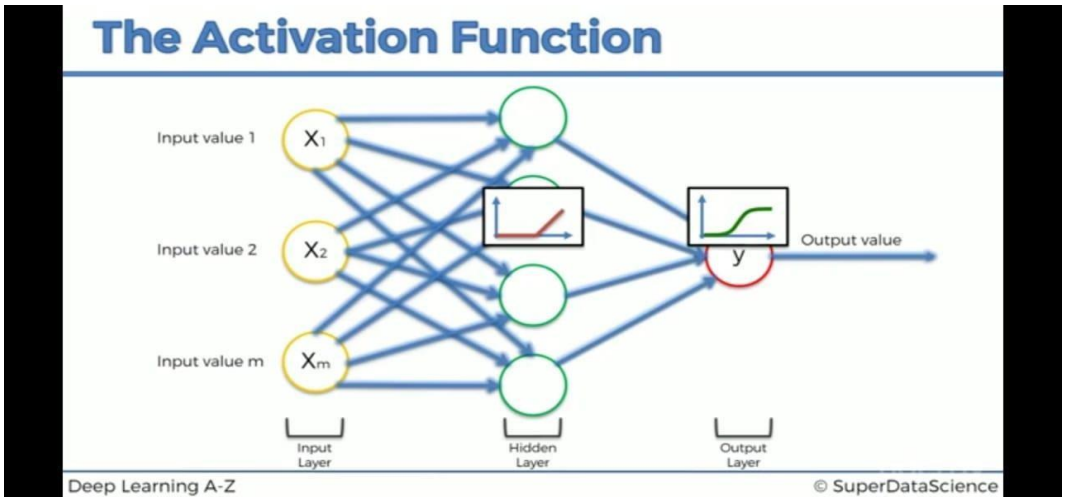
Activation function decides, whether a neuron should be activated or not by calculating weighted sum and further adding bias with it. The purpose of the activation function is to **introduce non-linearity** into the output of a neuron.

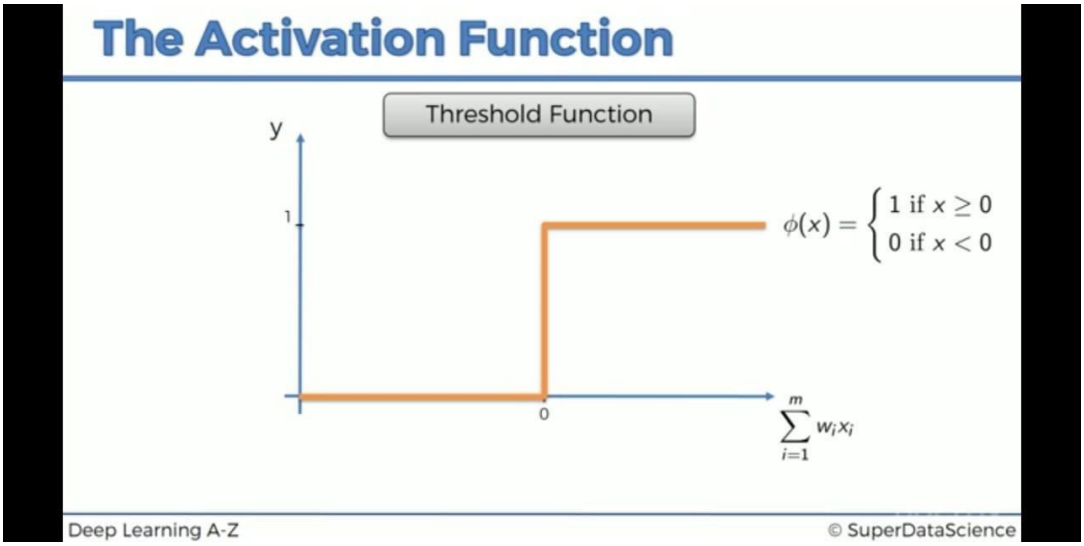
Explanation :-

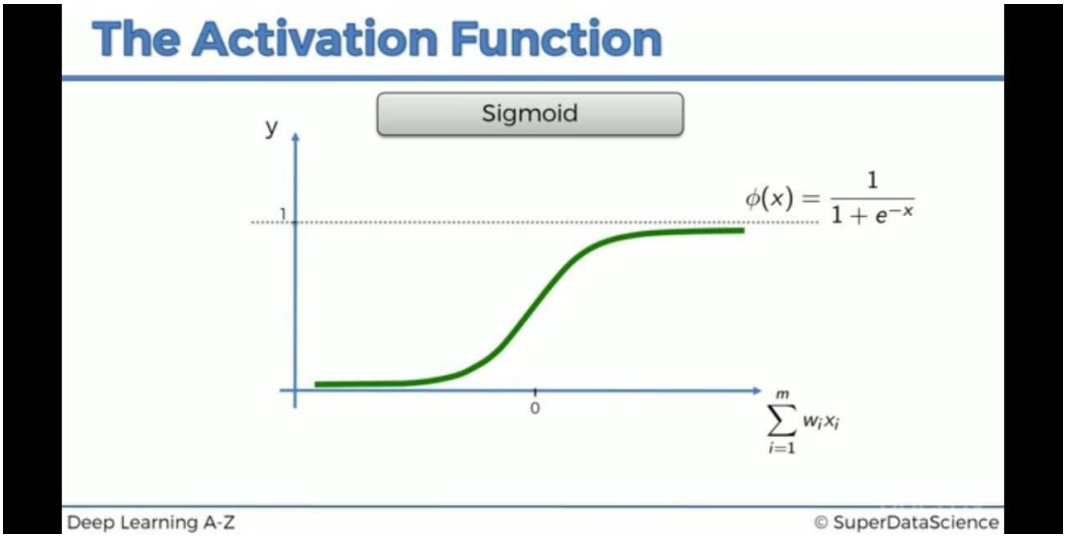
We know, neural network has neurons that work in correspondence of *weight, bias* and their respective activation function. In a neural network, we would update the weights and biases of the neurons on the basis of the error at the output. This process is known as *back-propagation*. Activation functions make the back-propagation possible since the gradients are supplied along with the error to update the weights and biases.

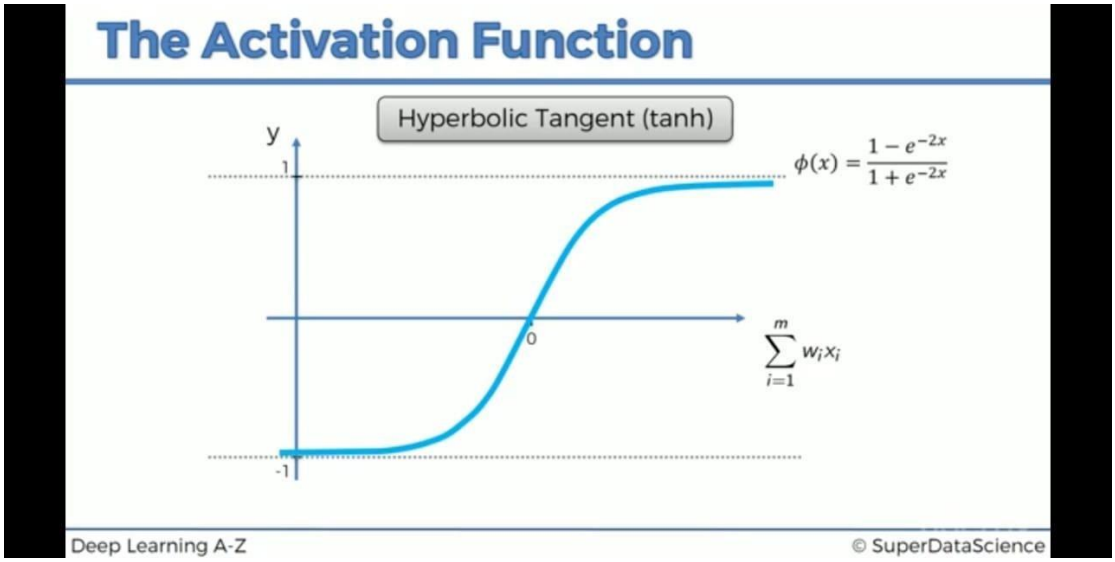
A neural network without an activation function is essentially just a linear regression model. The activation function does the non-linear transformation to the input making it capable to learn and perform more complex tasks. Activation functions classifies the outputs.

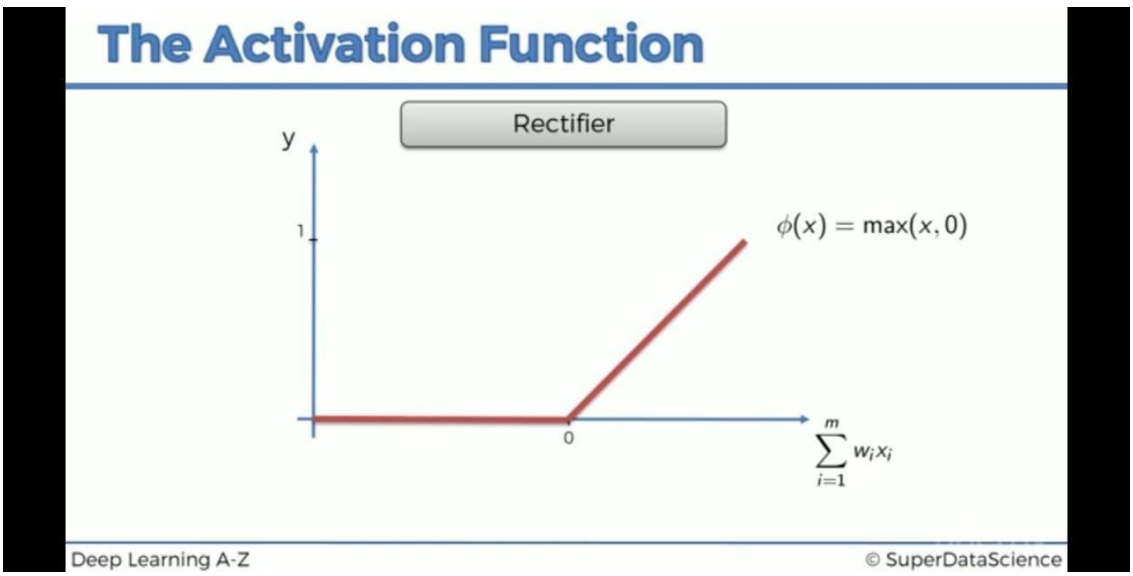
The activation function can be sigmoid function, relufunction, threshold function,hyperbolic function,rectifier.

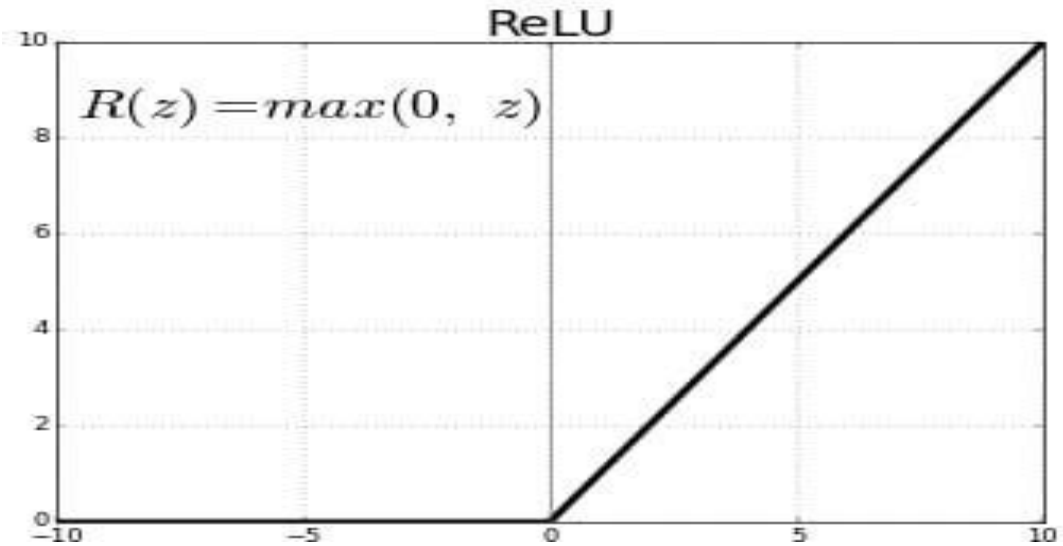

The Activation Function
Input value 1
Input value 2
Input value m
X_1
X_2
X_m
y
Output value
Input Layer
Hidden Layer
Output Layer
Deep Learning A-Z
© SuperDataScience


The Activation Function
Threshold Function
y
1
0
$\phi(x) = \begin{cases} 1 \text{ if } x \geq 0 \\ 0 \text{ if } x < 0 \end{cases}$
$\sum_{i=1}^{m} w_i x_i$
Deep Learning A-Z
© SuperDataScience


The Activation Function
Sigmoid
y
1
0
$\phi(x) = \frac{1}{1+e^{-x}}$
$\sum_{i=1}^{m} w_i x_i$
Deep Learning A-Z
© SuperDataScience

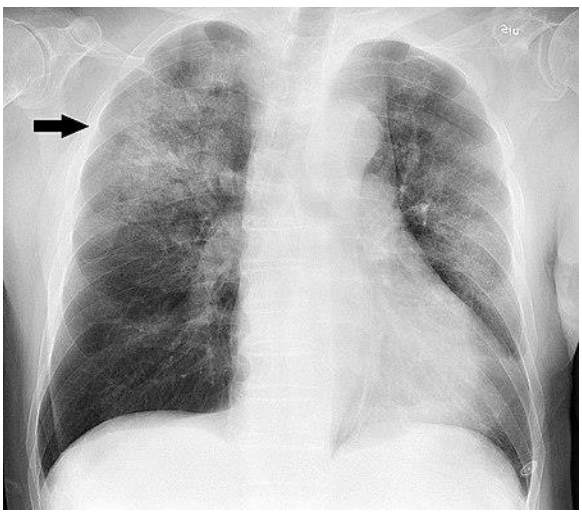

The Activation Function
Hyperbolic Tangent (tanh)
y
1
-1
0
$\phi(x) = \frac{1 - e^{-2x}}{1 + e^{-2x}}$
$\sum_{i=1}^{m} w_i x_i$
Deep Learning A-Z
© SuperDataScience


The Activation Function
Rectifier
y
1
0
$\phi(x) = \max(x, 0)$
$\sum_{i=1}^{m} w_i x_i$
Deep Learning A-Z
© SuperDataScience


ReLU
$R(z) = max(0, z)$
10
8
6
4
2
0
−10
−5
0
5
10

PNEUMONIA

Pneumonia is an inflammatory condition of the lung affecting primarily the small air sacs known as alveoli. The pneumonia is usually caused by bacteria or fungi. The bacteria behind causing pneumonia are streptococcus pneumonia. These bacteria enter into the small air sacs called alveoli which are located at the end of respiratory chain causing them to fill with fluid or puss which in turn makes it hard to breathe and also the oxygen transfer to the blood is blocked to some extent.



Chest x-ray: An x-ray exam will allow your doctor to see your lungs, heart and blood vessels to help determine if you have pneumonia. When interpreting the x-ray, the radiologist will look for white spots in the lungs (called infiltrates) that identify an infection.

The normal chest X-ray (left panel) depicts clear lungs without any areas of abnormal pacification in the image. Bacterial pneumonia (middle) typically exhibits a focal lobar consolidation, in this case in the right upper lobe (white arrows), whereas viral pneumonia (right) manifests with a `more diffuse ‘‘interstitial’ pattern in both lungs.

SYMPTOMS

People with infectious pneumonia often have
a productive cough, fever accompanied by shaking chills, shortness of breath, sharp or stabbing chest pain during deep breaths, and an increased rate of breathing. In elderly people, confusion may be the most prominent sign.
The typical signs and symptoms in children under five are fever, cough, and fast or difficult breathing.] Fever is not very specific, as it occurs in many other common illnesses and may be absent in those with severe disease, malnutrition or in the elderly. In addition, a cough is frequently absent in children less than 2 months old.More severe signs and symptoms in children may include blue-tinged skin, unwillingness to drink, convulsions, ongoing vomiting, extremes of temperature, or a decreased level of consciousness

Symptoms frequency[18]	
Symptom	**Frequency**
Cough	79–91%
Fatigue	90%
Fever	71–75%
Shortness of breath	67–75%
Sputum	60–65%
Chest pain	39–49%

CODE

```
## Importing the Keras libraries and packages
from keras.models import Sequential
from keras.layers import Conv2D
from keras.layers import MaxPooling2D
from keras.layers import Flatten
from keras.layers import Dense
from keras.preprocessing.image import ImageDataGenerator
#import os

#os.chdir("\Users\apple\Desktop\chest_xray")

# Initialising the CNN
pneumonia_classifier = Sequential()

# Step 1 - Convolution
pneumonia_classifier.add(Conv2D(32, (3, 3), input_shape = (128, 128, 3),
activation = 'relu'))

# Step 2 - Pooling
pneumonia_classifier.add(MaxPooling2D(pool_size = (2, 2)))

# Adding a second convolutional layer and pooling layer
pneumonia_classifier.add(Conv2D(32, (3, 3), activation = 'relu'))
pneumonia_classifier.add(MaxPooling2D(pool_size = (2, 2)))

# Step 3 - Flattening
pneumonia_classifier.add(Flatten())

# Step 4 - Full connection
pneumonia_classifier.add(Dense(units = 128, activation = 'relu'))
pneumonia_classifier.add(Dense(units = 64, activation = 'relu'))
pneumonia_classifier.add(Dense(units = 1, activation = 'sigmoid'))

# Compiling the CNN
pneumonia_classifier.compile(optimizer = 'adam', loss =
'binary_crossentropy', metrics = ['accuracy'])
```

```
# Part 2 - Fitting the CNN to the images



train_datagen = ImageDataGenerator(rescale = 1./255,
                                   shear_range = 0.2,
                                   zoom_range = 0.2,
                                   horizontal_flip = True)

test_datagen = ImageDataGenerator(rescale = 1./255,
                                  shear_range = 0.2,
                                  zoom_range = 0.2,
                                  horizontal_flip = True)

training_set = train_datagen.flow_from_directory('train',
                                                 target_size = (128, 128),
                                                 batch_size = 32,
                                                 class_mode = 'binary')

test_set = test_datagen.flow_from_directory('test',
                                            target_size = (128, 128),
                                            batch_size = 32,
                                            class_mode = 'binary')

pneumonia_classifier.fit_generator(training_set,
                         steps_per_epoch = 100,
                         epochs = 1,
                         validation_data = test_set,
                         validation_steps = 50)



# Part 3 - Making new predictions
import numpy as np
from keras.preprocessing import image
test_image =
image.load_img(r"/Users/apple/Desktop/person1_bacteria_2.jpeg",
target_size = (128, 128))
test_image = image.img_to_array(test_image)
test_image = np.expand_dims(test_image, axis = 0)
result = pneumonia_classifier.predict(test_image)
```

```
training_set.class_indices
if result[0][0] == 1:
    print('\nEFFECTED')
else:
    print('\nUNEFFECTED')
```