Deep Learning Using TensorFlow



1.2: Deep Learning

Autonomous Car Solves a Real World Problem

A person dies every 23 seconds on roads



Artificial Intelligence

Artificial Intelligence

Machine Learning

Deep Learning

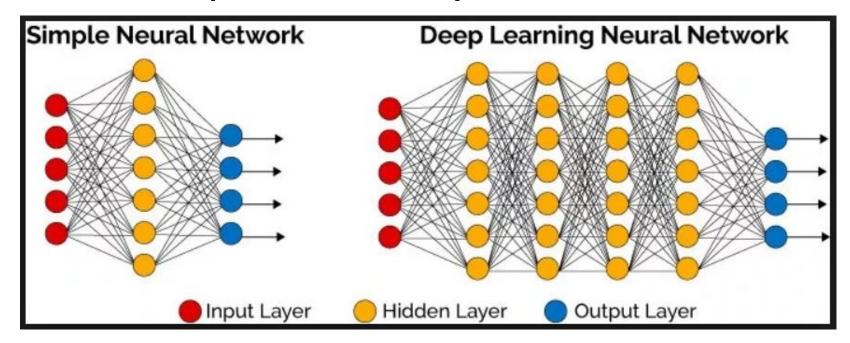
The subset of machine learning composed of algorithms that permit software to train itself to perform tasks, like speech and image recognition, by exposing multilayered neural networks to vast amounts of data.

A subset of AI that includes abstruse statistical techniques that enable machines to improve at tasks with experience. The category includes deep learning

Any technique that enables computers to mimic human intelligence, using logic, if-then rules, decision trees, and machine learning (including deep learning)

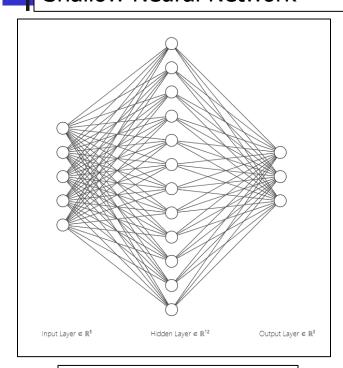
Deep Learning

 A Deep Neural Network (DNN) has multiple hidden layers



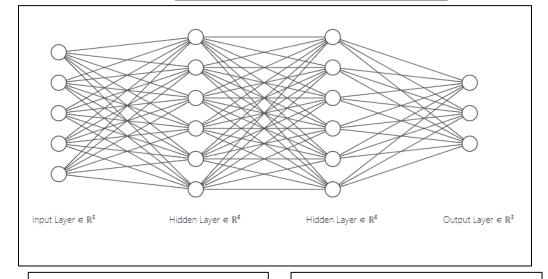
Shallow and Deep NN

Shallow Neural Network



Input layer: 5 nodes Hidden layer: 12 nodes Output layer: 3 nodes

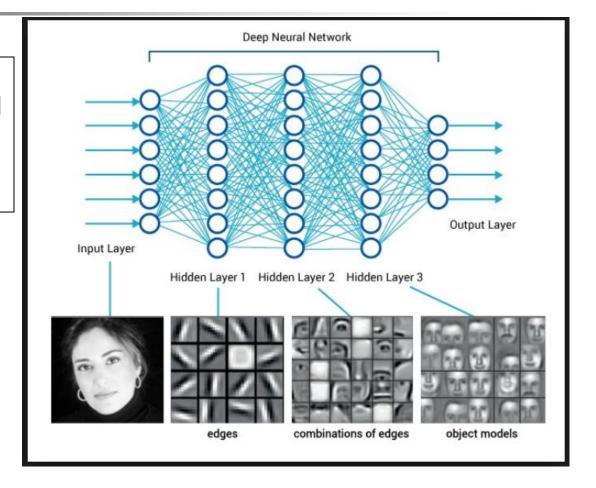
Deep Neural Network



Input layer: 5 nodes Hidden layer 1: 6 nodes Hidden layer 2: 6 nodes Output layer: 3 nodes More layers usually provides more efficient and accurate representation of data

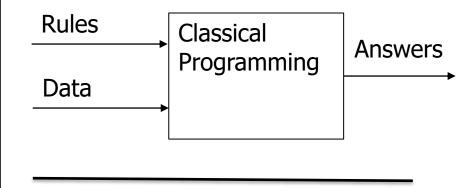


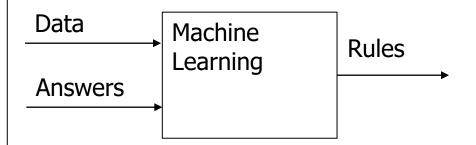
- Every layer of the DNN allows a more sophisticated build-up
 - From simple elements
 - To more complex ones



Problems that can use Neural Networks

- For simple problems we can define the rules
 - We can automate the process
 - Write software
- For complex problems
 - We cannot define the rules
 - Object recognition in an image
- To solve these types of problems
 - We provide data and the answers
 - System will create the rules





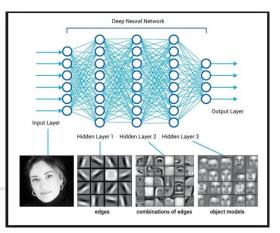


Main Applications of Deep Learning Neural Networks

- Image Recognition
 - Convolution Neural Networks
- Image Classification
 - Convolution Neural Networks
- Hand Writing Identification
- Speech Recognition
 - Long Short-Term Memory Networks



Backpropagation Algorithm:



- Conceptually the backpropagation algorithm is very simple
- Algorithm
 - Assign random values to all the weights of the NN
 - Take the first observed data
 - Forward Propagation: Compute Output
 - Compute error = $(Computed\ Output\ Observed\ Output)^2$
 - Backpropagation: adjust weights to reduce error
 - Repeat forward, backward propagation, till error is minimized
 - Repeat the previous step for the next sample till all samples are processed
 - The final weights of the NN will be used for prediction