Introduction to Deep Learning

Arijit Mondal

Dept. of Computer Science & Engineering

Indian Institute of Technology Patna

arijit@iitp.ac.in

General Information

- **Instructors**
  - Dr. Joydeep Chandra
  - Arijit Mondal
- **Teaching assistants**
  - Jyoti Kumari
  - Shruti Saxena

- **Course webpage:** `www.iitp.ac.in/~arijit/`, **then follow** `Teaching`

Course structure

- Introduction to big data problem & representation learning
- Overview of linear algebra and probability
- Basics of feature engineering
- Neural network
- Introduction to open-source tools
- Deep learning network
- Regularization
- Optimization
- Advanced topics
- Practical applications

Evaluation policy

- Two quizzes - 20%
- Midsem - 30%
- Endsem - 50%

Books

- Deep Learning - Ian Goodfellow, Yoshua Bengio, Aaron Courville
- The Elements of Statistical Learning - Jerome H Friedman, Robert Tibshirani, Trevor Hastie
- Neural Networks and Deep Learning - Charu Agarwal
- Neural Network and Learning Machines - Simon S. Haykin
- Reinforcement Learning: An Introduction - Richard S Sutton, Andrew G Barto

Introduction

Problem space

- **Problems — *a matter or situation regarded as unwelcome or harmful and needing to be dealt with and overcome***
- **Target is to solve the same on a computer**

Problem space

- Problems — *a matter or situation regarded as unwelcome or harmful and needing to be dealt with and overcome*
- Target is to solve the same on a computer

- Problems can be intellectually challenging for human being but relatively straight forward for a computer
  - Travelling salesman problem, chess
- Problems can be easy for common people but difficult for computer (even expressing it in a formal way)
  - Identifying an object, car (say), in a picture

Problem space

- **Problems — *a matter or situation regarded as unwelcome or harmful and needing to be dealt with and overcome***
- **Target is to solve the same on a computer**

- **Problems can be intellectually challenging for human being but relatively straight forward for a computer**
  - **Travelling salesman problem, chess**
- **Problems can be easy for common people but difficult for computer (even expressing it in a formal way)**
  - **Identifying an object, car (say), in a picture**

- **Primary focus will be in the *second category* problems**

Problem Solving Strategies for Big Data

- Need to solve problems efficiently and accurately when the input data is huge ($\sim$ GB, TB order)

- Finding a deterministic algorithm is difficult
  - Need to find out features
  - Requires significant effort for model building
  - Need to have domain knowledge

- Statistical inference is found to be suitable
  - Feature selection is not crucial
  - Model will learn from past data

Applications: Computer vision

- 2d to 3d conversion
- Street view generation
- Image classifications
- Image segmentation

Image source: Internet

Applications: Activity Recognition

- Recognize activities like walking, running, cooking, etc. from still image or video data

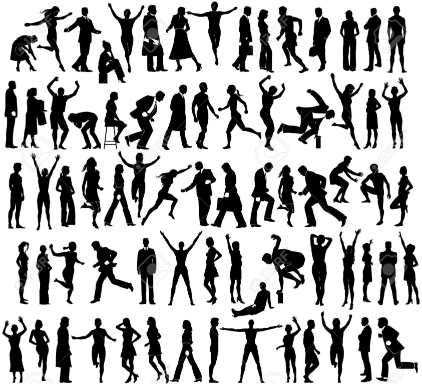


Image source: Internet

Applications: Image Captioning

- **Automated caption generation for a given image**

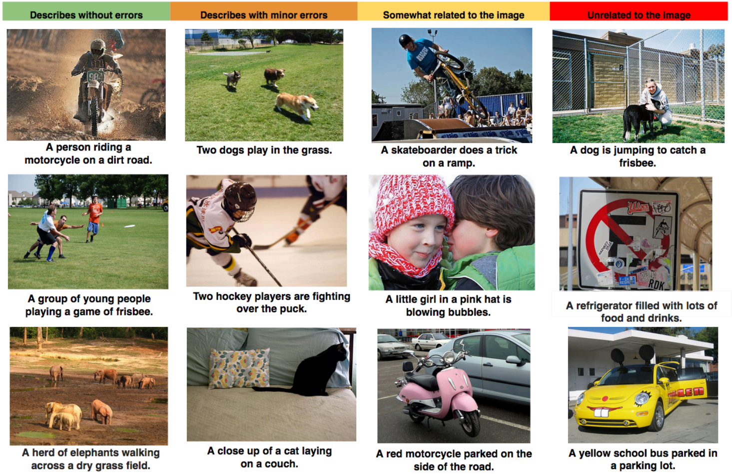


Image source: Internet

Applications: Object Identification

- **Identify objects in still image or in video stream**

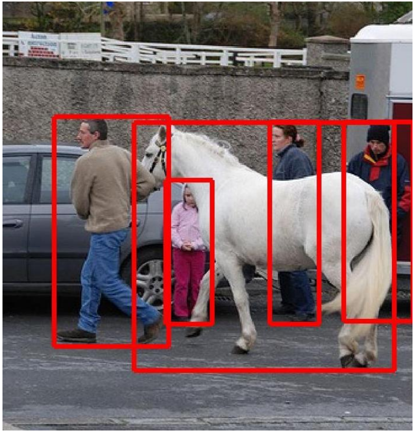


Image source: Internet

Applications: Automated Car

- **Self driving car**

Image source: Internet

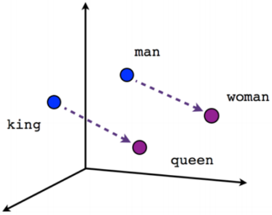
Applications: Drones & Robots

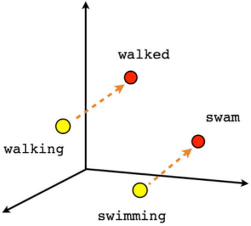
- **Managing movement of robot or drones**

Image source: Internet

Applications: Natural Language Processing

- Recommender system
- Sentiment analysis
- Question answering
- Information extraction from website
- Automated email reply





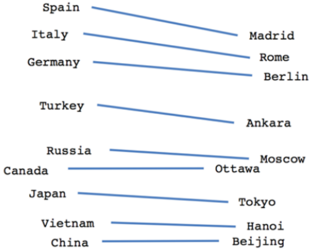


Image source: Internet

Applications: Speech processing

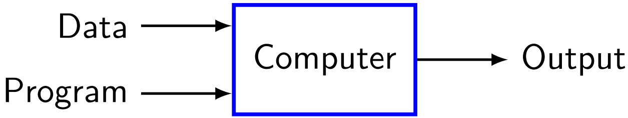
- Conversion of speech into text
- Generation of particular voice for a given text

Image source: Internet

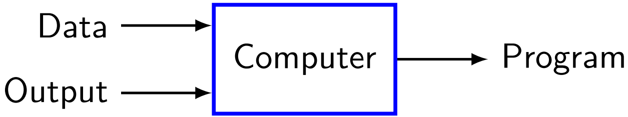
Other possible applications

- Language translation
- Weather prediction
- Genomics
- Drug discovery
- Particle physics
- Surveillance
- Cryptography and many more.

Traditional Programming vs ML/DL

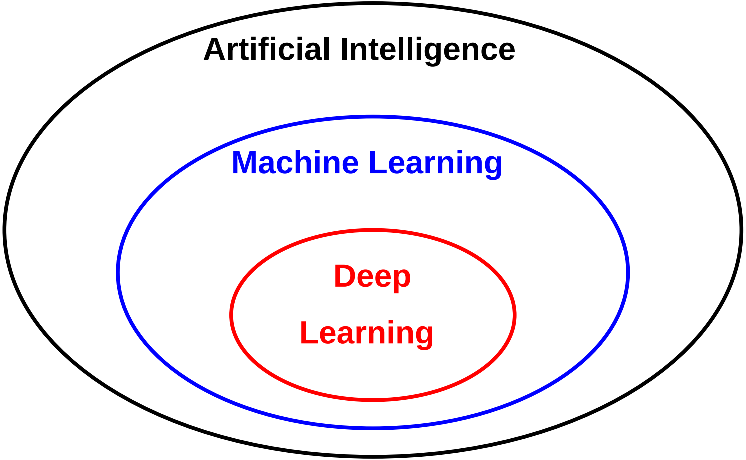


Traditional programming



Machine learning

AI Hierarchy



Issue of Representation

- Representation of data in an efficient/structured manner is crucial for solving problems more effectively
  - Searching of a set of elements in a given list (sorted/unsorted)
  - Arithmetic operations on Arabic and Roman numerals
  - Primality test of n when n is represented as $11111\ldots111$ (n-number of one)

- Structured representation can help in predicting future values

Choice of Representation

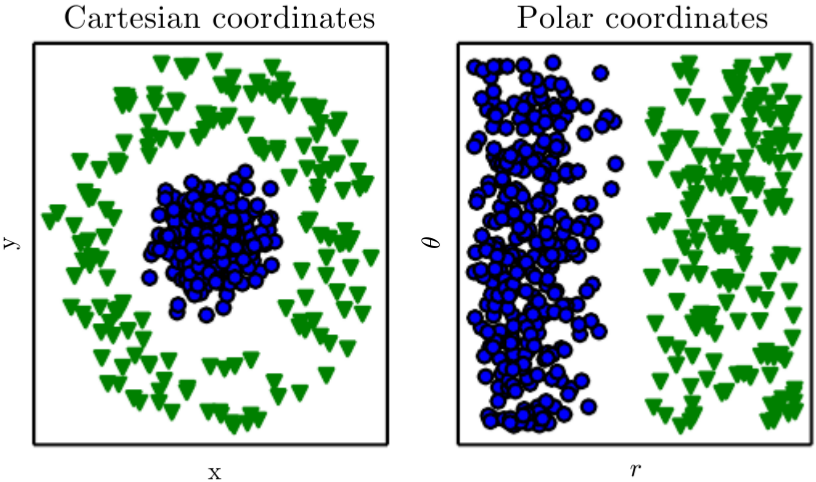


Image source: Deep Learning book

Learning representation/feature

- Traditional approaches
  - Pattern recognition
    - Input, output of the problem

- End to end learning
  - System automatically learns internal representation

AI-ML Tasks

- Heavily depends on features
- Requires good domain knowledge
- Feature extraction is not easy job
  - Identify a car
    - How to describe wheel
    - Shadow/brightness
    - Obscuring element

Representation Learning

- Learned representation often result in better performance compared to hand design
- Allows the system to rapidly adapt to new task
- Need to discover a good set of features
- Manual design of features is nearly impossible

Design of Features

- Goal is to separate out variation factors
- These factors are separate sources of influence
- It may exist as unobserved object or unobserved forces that affect observable quantity
  - Speech - Factors are age, sex, accent, etc
  - Image - Position, color, brightness, etc.

Deep Learning

- Try to address the problem of representation learning
- Representation are expressed in terms of other simpler representation
- Develop complex concept using simpler concept

Simple to Complex Features

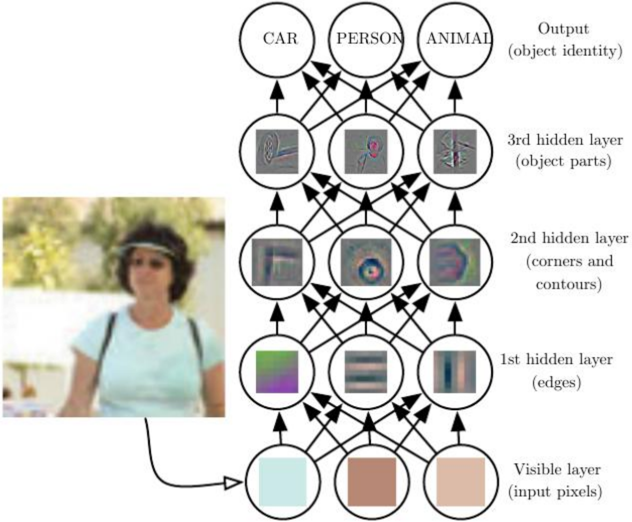


Image source: Deep Learning Book

Simple to Complex Features

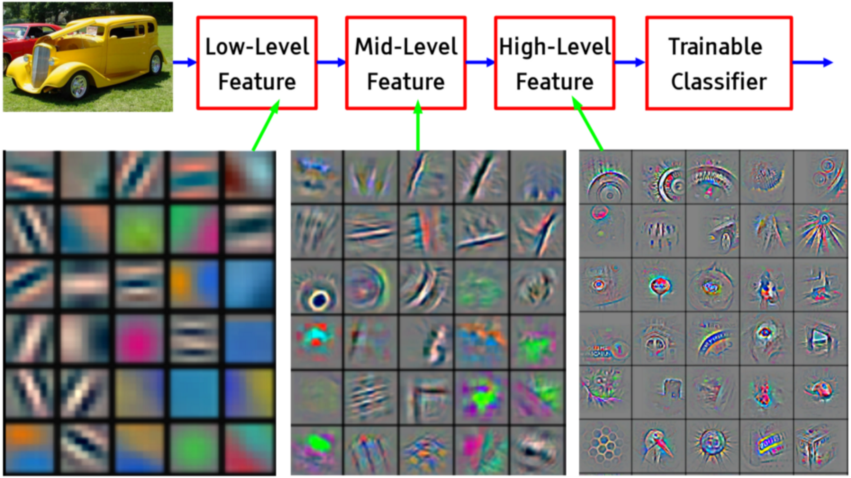


Image source: Deep Learning Tutorial by Yann LeCun Marc'Aurelio Ranzato, ICML, 2013

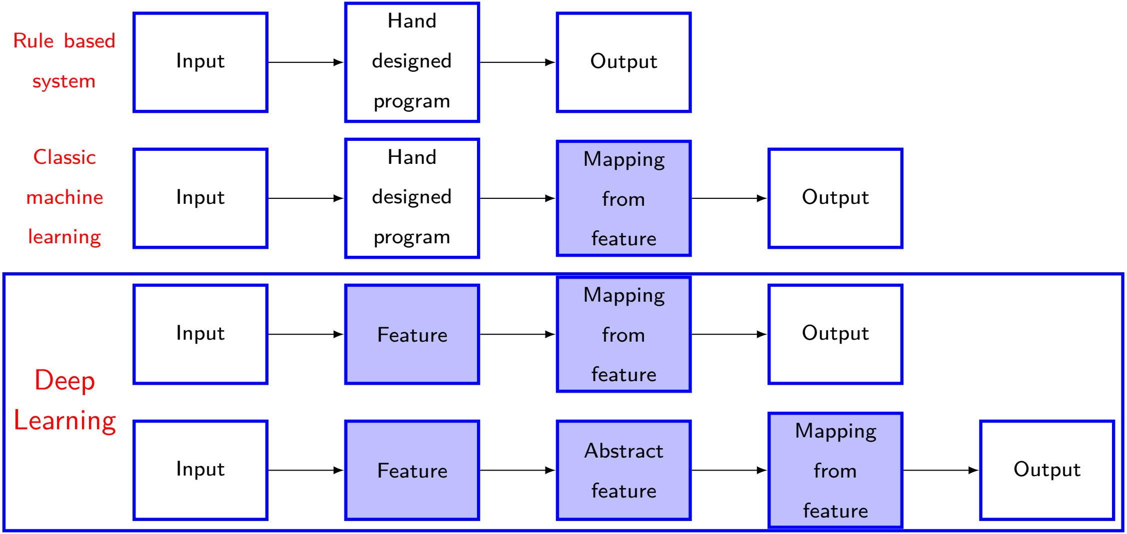
Conventional Machine Learning

Image source: Deep Learning by Yann LeCun, Yoshua Bengio & Geoffrey Hinton

Deep Learning Model

- Feed-forward deep network or multilayer perceptron
- Mathematical functions that map input to output
- Composed of simpler functions
- Each layer provides a new representation
- Learning right representation

Representation learning



History

- Has many names and view point
  - Cybernetics (1940-1960)
  - Connectionism (1980-1990) (neural net)
  - Deep learning (2006+)

- More useful as the amount of data is increased
- Models have grown in size as increase in computing resources
- Solving complex problem with increasing accuracy

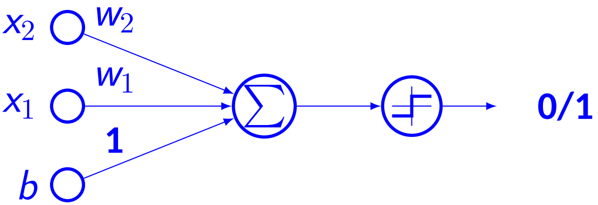
Learning Algorithm

- Early learning algorithm
  - How learning happen in brain?
  - Computational model of biological learning

- Neural perspective of DL
  - Brains provide a proof by example
  - Reverse engineer the computational principle behind the brain and duplicate its functionality

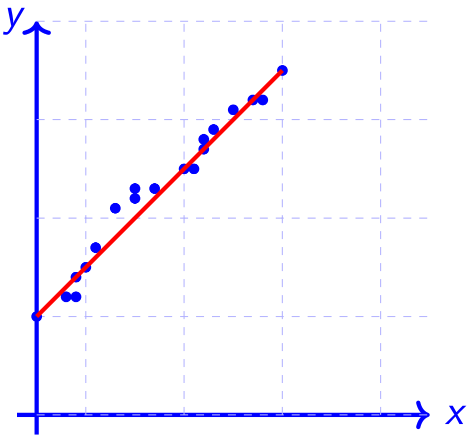
History of basic model

- The first learning machine: the Perceptron
  - Built at Cornell, 1960
- Perceptron was linear classifier on top of simple feature extractor
- Most of the practical applications of ML today use glorified linear classifiers or glorified template matching.
- Significant effort is required for identifying relevant features
- Typically it will solve $y = sign\left(\sum_{i=1}^{N}(w_i \times f_i(X) + b)\right)$

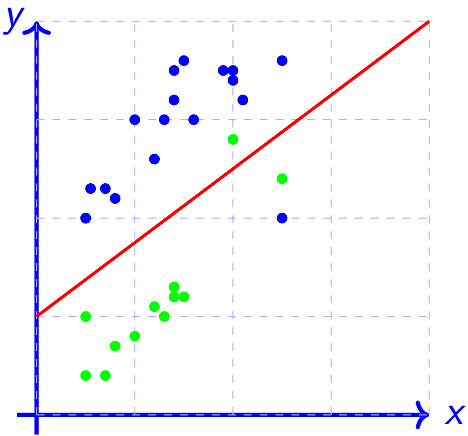


Broad Categories of Problem

- **Regression**

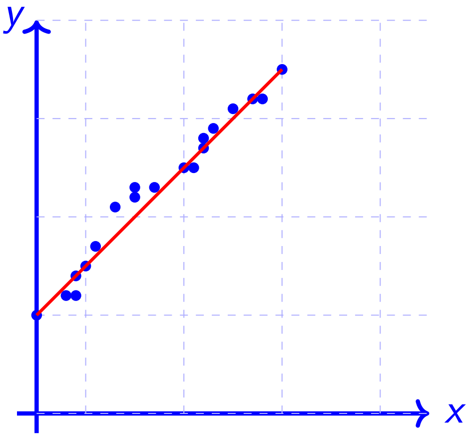


- **Classification**

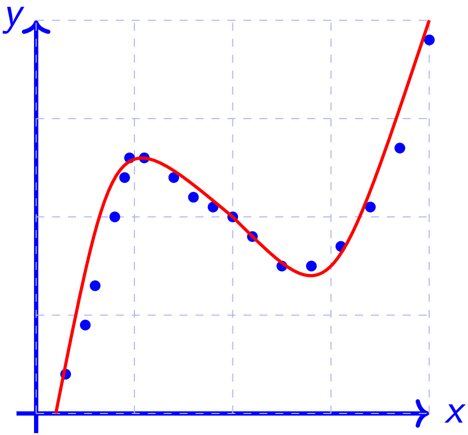


Regression

- **Regression (linear)**

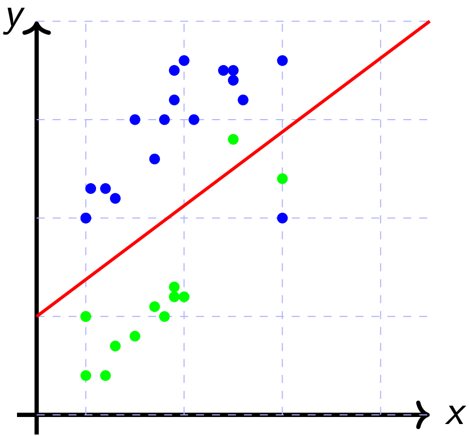


- **Regression (Non-linear)**

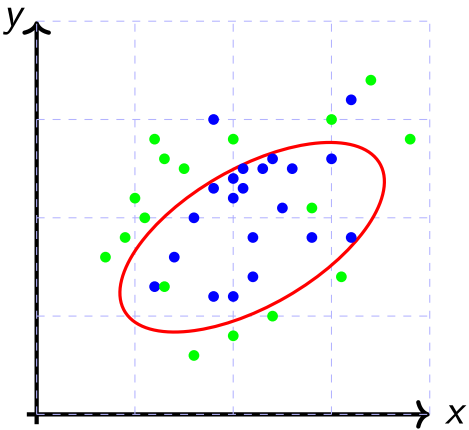


Classification

- **Linear**

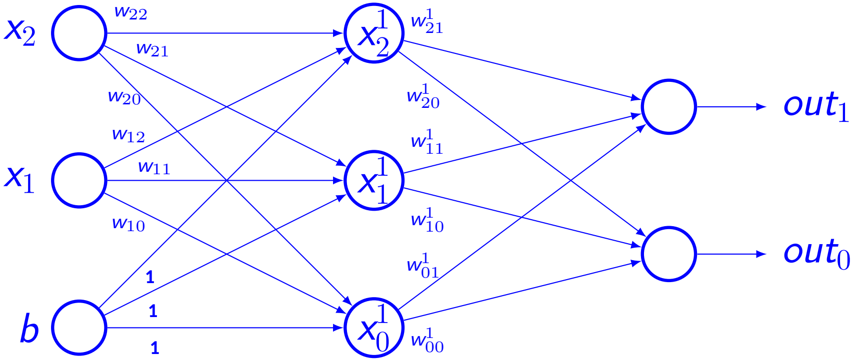


- **Non-linear**

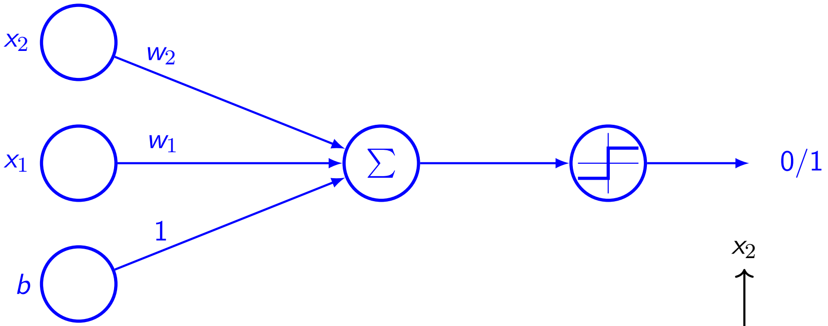


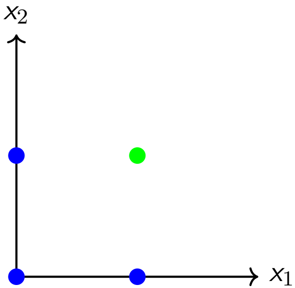
Artificial Neural Network

- **A simple model**

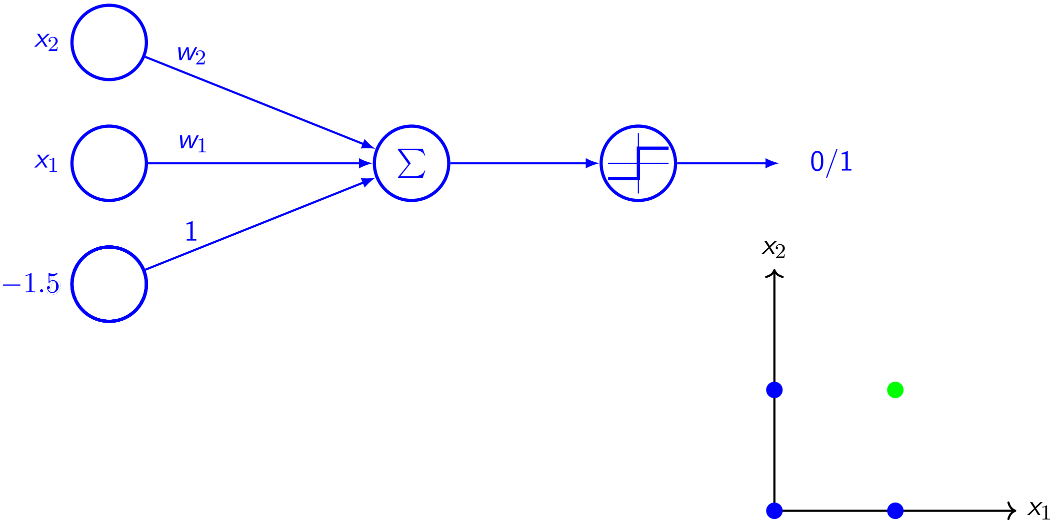


Example NN: AND gate

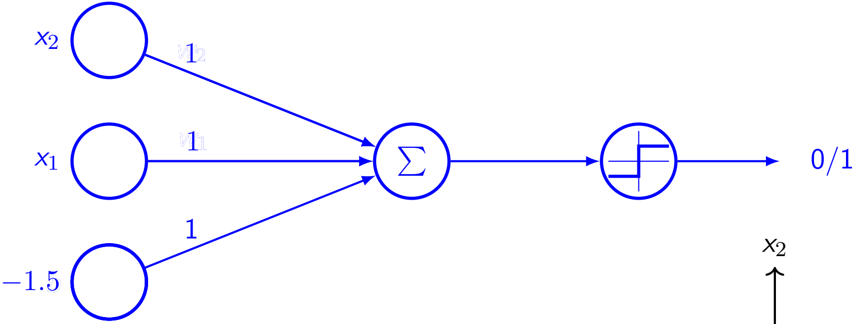


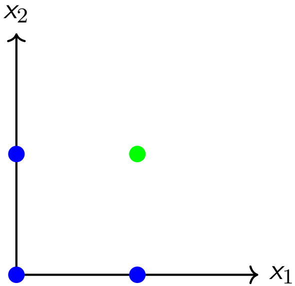


Example NN: AND gate

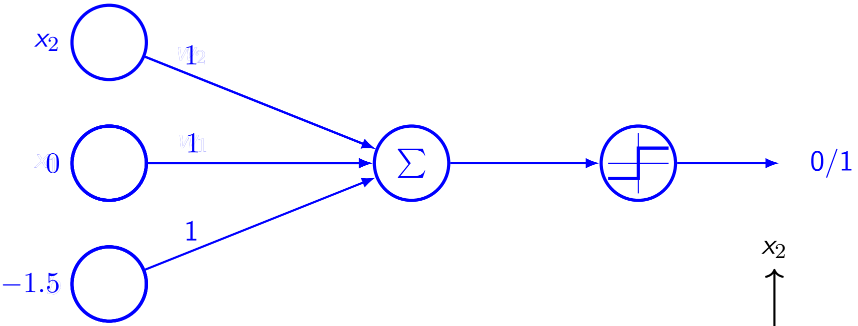


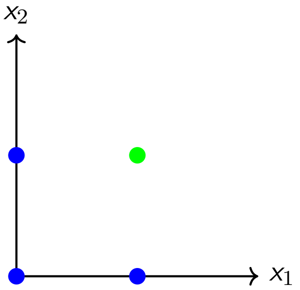
Example NN: AND gate



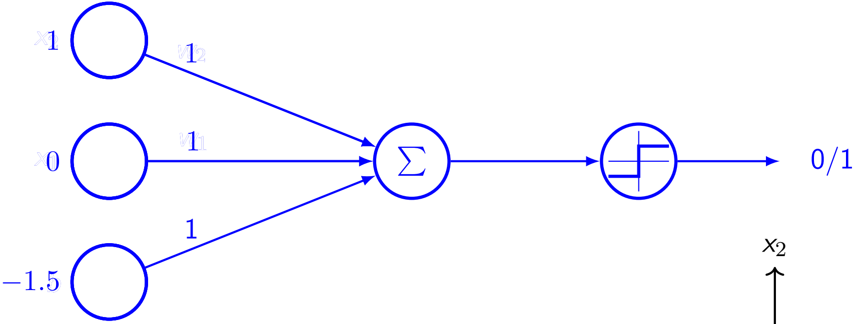


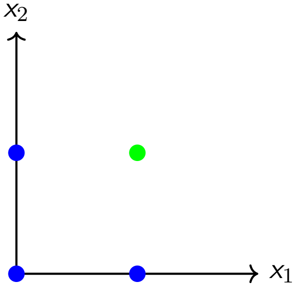
Example NN: AND gate



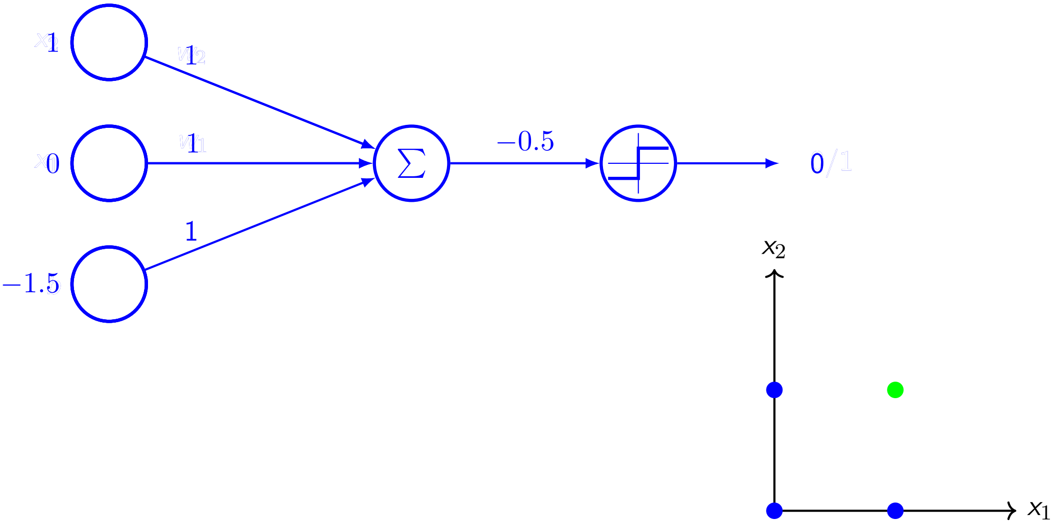


Example NN: AND gate

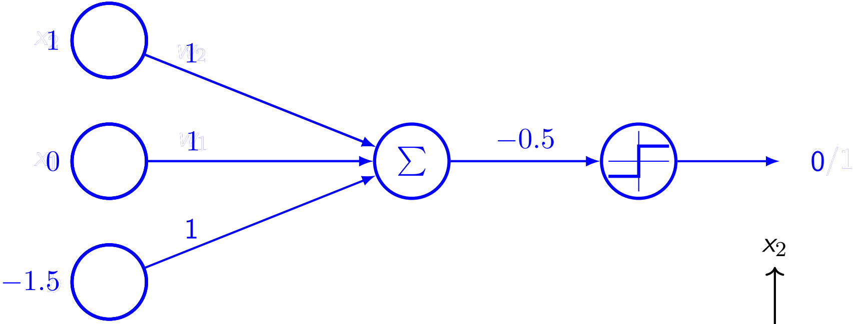


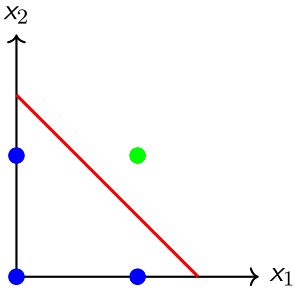


Example NN: AND gate

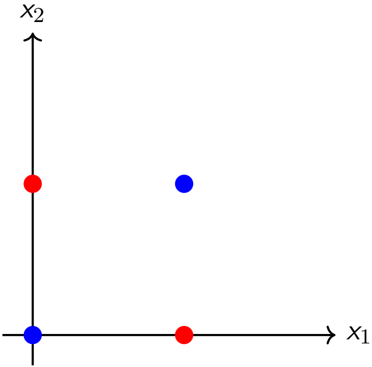


Example NN: AND gate

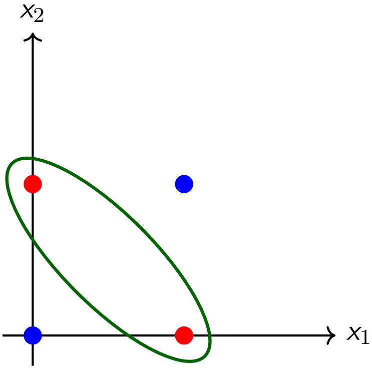




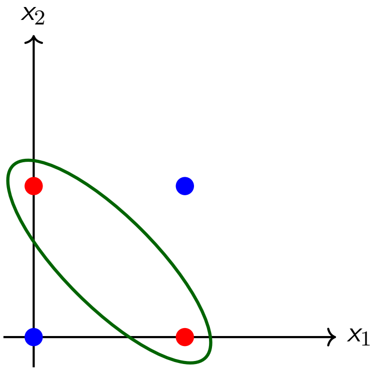
Example NN: XOR gate



Example NN: XOR gate



Example NN: XOR gate



Distributed representation

- Each input should be represented by many features
- Each feature should be involved in the representation of many possible inputs
- Example: car, flower, birds — red, green, blue
  - 9 neurons
  - For each combination of color and object
- Distributed neurons
  - 3 Neurons for color
  - 3 Neurons for object
  - Total 6 neurons

Popularization of Neural Network

- **Most of the theory of neural network was developed in the 1980s**
- **Started gaining popularity around 2012**
  - **Geoffrey Hinton and Alex Krizhevsky winning the ImageNet competition where they beat the nearest competitor by a huge margin (2012)**

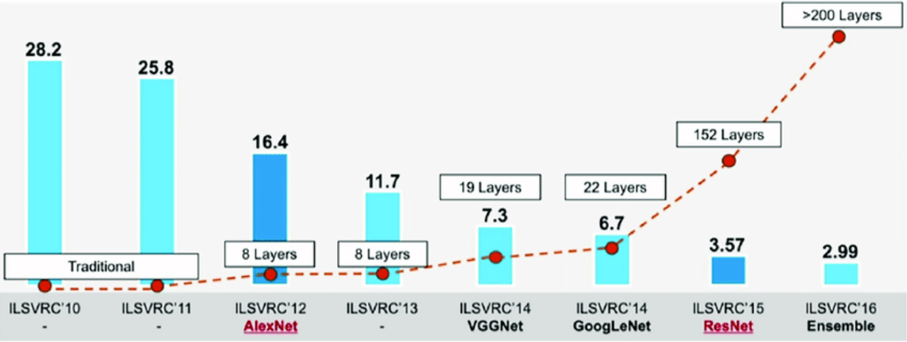


Image source: Internet

Popularity

- Increase data size
  - Computing resources are available
  - Accepting performance 5000 labeled example per category
  - 10 million for human performance
- Increasing model size
- Increasing accuracy, complexity, real world impact
- Used by many companies
  - Google, Microsoft, Facebook, IBM, Baidu, Apple, Adobe, Nvidia, NEC, etc.
- Availability of good commercial & open-source tools
  - Theano, Torch, DistBelief, Caffe, TensorFlow, Keras, etc.

DL Trend

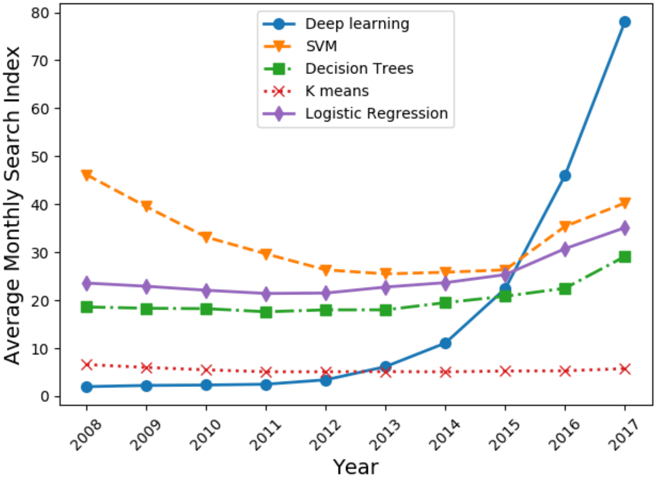


Image source: Internet

Search trend in Google

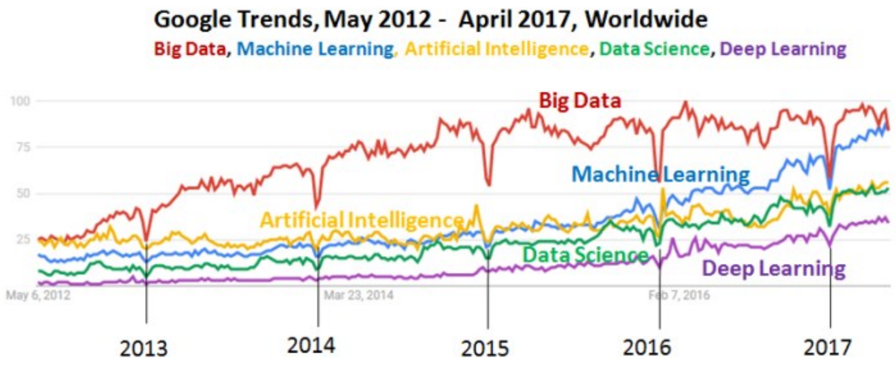
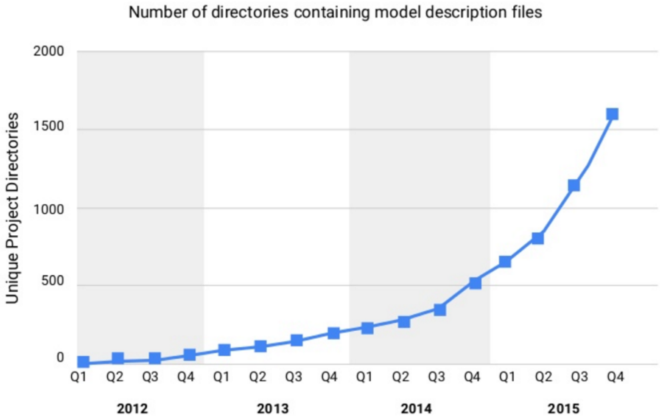


Image source: Internet

AI/DL in Google



Across many products/areas

- Apps
- Maps
- Photos
- Gmail
- Speech
- Android
- YouTube
- Translation
- Robotics Research
- Image Understanding
- Natural Language Understanding
- Drug Discovery

Google

Image source: Internet

Artificial Intelligence is the New Electricity - Andrew Ng

Artificial Intelligence is the New Electricity - Andrew Ng

Thank you!