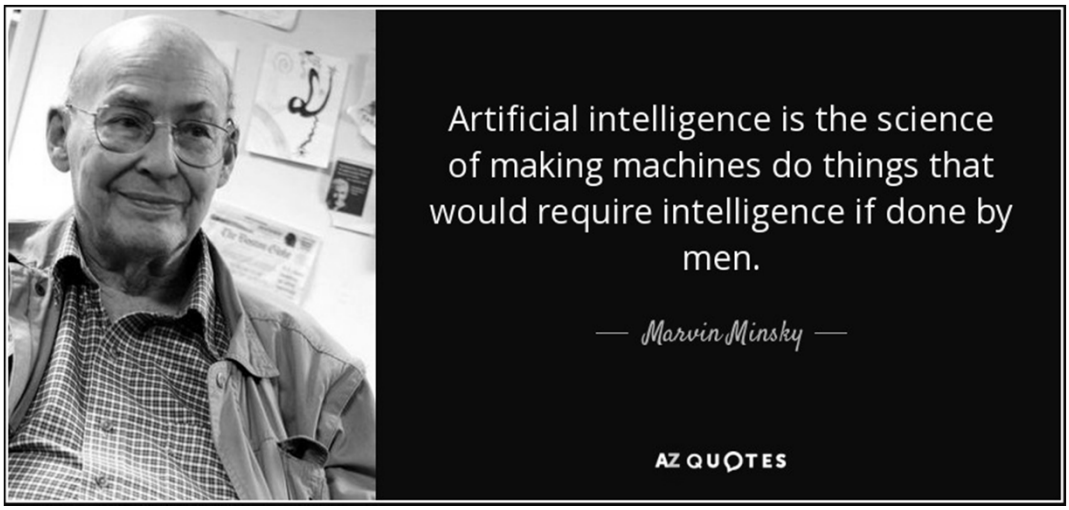
What is Artificial Intelligence?

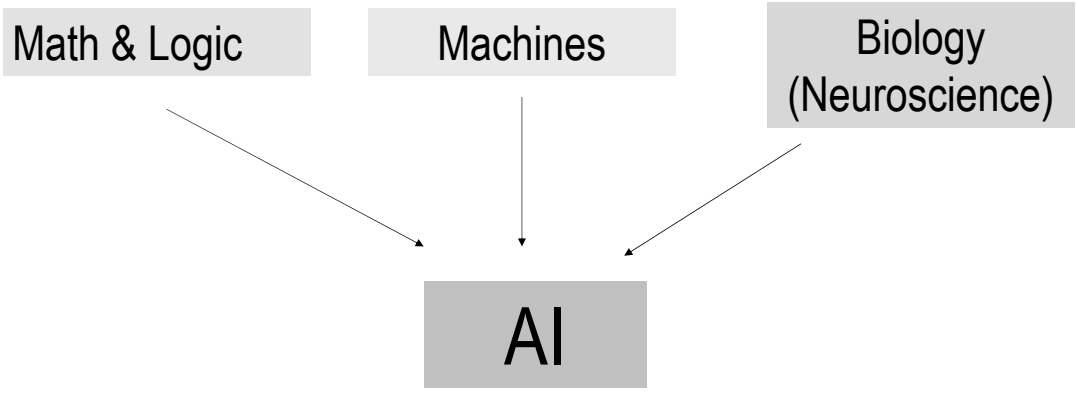
Objectives

- The Three eras of AI with key individuals, their contribution and timelines
- Role of math, machines and neuroscience in AI
- Basic understanding of;
  - Perceptrons and Neural nets
  - Classification
    - Linear regression, Logistic regression, K-means
- Machine Learning
  - Supervised vs Unsupervised vs Reinforcement Learning
  - Convolutional Neural Nets
  - Recurrent Neural nets
  - Tensor Flow

Marvin Minsky, MIT



The Three Dimensions of AI



Three Major Phases of AI

Phase I: 1600 – 1945
AI Before Digital Computers

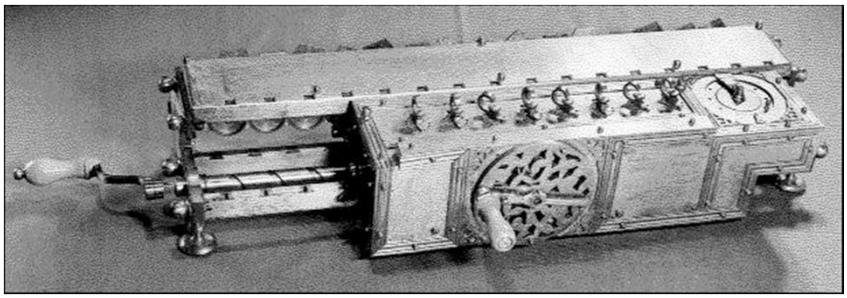
Phase 2: 1945 – 2001
AI and Computers

Phase 3: 2001 -
AI meets Big Data

Phase I

Math & Logic		Machines
1600	**Calculus**	Leibnitz (1620) - Calculator
1800	**Taylor series (sin, cos, log)**	Charles Babbage (1820) – Difference Engine
1900	**Axiomatic Logic & Set Theory**	Alan Turing (1930s) – Turing Machine
		John von Neumann (1940s)
1945		

Leibnitz – Step Reckoner – could add, subtract, multiply and divide



	Math & Logic	Machines
1600	**Calculus**	Leibnitz (1620) - Calculator
1800	**Taylor series (sin, cos, log)**	Charles Babbage (1820) – Difference Engine
1900	**Axiomatic Logic & Set Theory**	Alan Turing (1930s) - TM
		John von Neumann (1940s)
1945		

1820

Charles Babbage

Ships in the British Navy were running aground and sailors were dying because the logarithm tables used in calculations contained errors – even though the calculations were done by COMPUTERS !

What kind of computer existed in 1820?

Computers were people who did arithmetic for a living

Taylor Series – can compute sin, cos, tangent and e^x using an infinite series = A big polynomial

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \cdots$$

$$= \sum_{n=0}^{\infty} \frac{x^n}{n!}$$

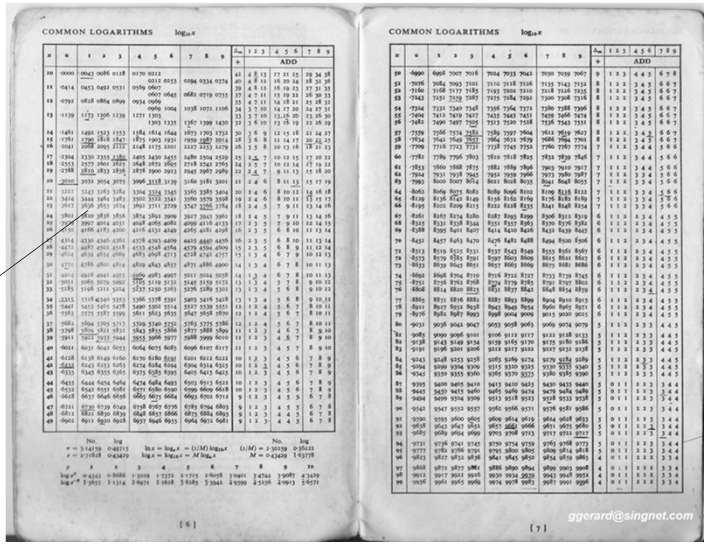
Logarithms let you multiplication and division by doing addition and subtraction

$$\log_b(MN) = \log_b(M) + \log_b(N)$$

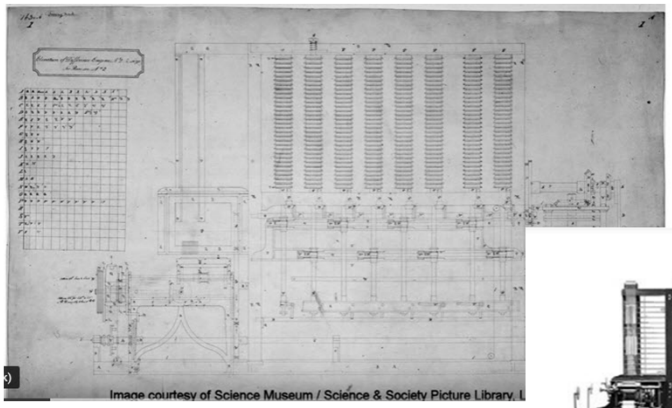
$$\log_b\left(\frac{M}{N}\right) = \log_b(M) - \log_b(N)$$

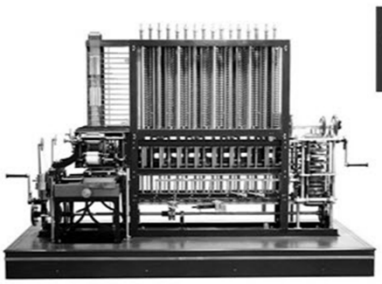
$$\log_b(M^p) = p\log_b(M)$$

BUT, errors in log tables resulted in numerous ships lost at sea.

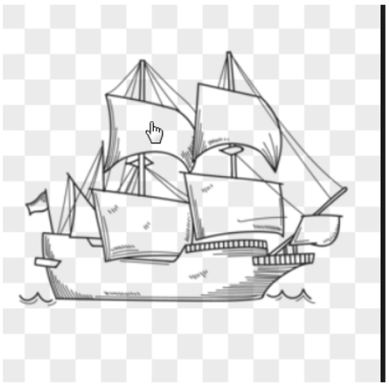


Babbage Difference Engine

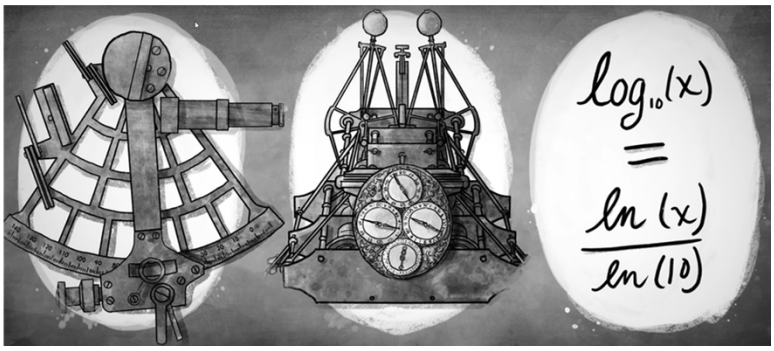




Computer History Museum



Navigation needs:
1. sextants
2. clocks
3. logarithms for math calculations



Easy way to compute logarithms using polynomials – based on difference tables

$f(x) = 2x + 5$

x	$f(x)$	Δ
1	7	
2	9	9 – 7 = 2
3	11	11 – 9 = 2
4	13	13 – 11 = 2

$f(x) = 2x^2 + 3x + 1$

x	$f(x)$	Δ_1	Δ_2
1	6		
2	15	15 – 6 = 9	
3	28	28 – 15 = 13	13 – 9 = 4
4	45	45 – 28 = 17	17 – 13 = 4
5	66	66 – 45 = 21	21 – 17 = 4

$$f(x) = 2x^2 + 3x + 1$$

x	$f(x)$	Δ_1	Δ_2
1	6	9	4
2	15	13	4
3	28	17	4
4	45	21	4

$$f(x) = x^7 + x$$

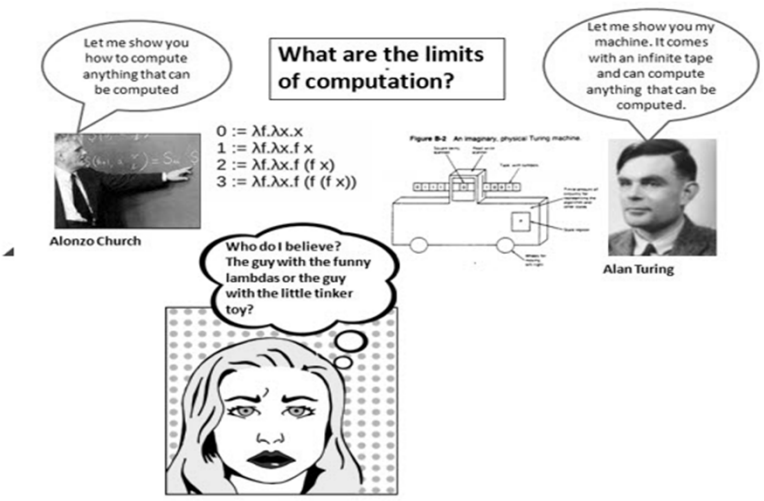
x	$f(x)$	Δ_1	Δ_2	Δ_3	Δ_4	Δ_5	Δ_6	Δ_7
3	2190	14198	47544	92526	109200	77280	30240	5040
	+	+	+	+	+	+	+	
	=	=	=	=	=	=	=	
4	16388	61742	140070	201726	186480	107520	35280	5040

$$f(x) = x^7 + x$$

x	$f(x)$	Δ_1	Δ_2	Δ_3	Δ_4	Δ_5	Δ_6	Δ_7
1	2							
2	130	128						
3	2190	2060	1932					
4	16388	14198	12138	10206				
5	78130	61742	47544	35406	25200			
6	279942	201812	140070	92526	57120	31920		
7	823550	543608	341796	201726	109200	52080	20160	
8	2097160	1273610	730002	388206	186480	77280	25200	5040
9	4782978	2685818	1412208	682206	294000	107520	30240	5040
10	10000010	5217032	2531214	1119006	436800	142800	35280	5040

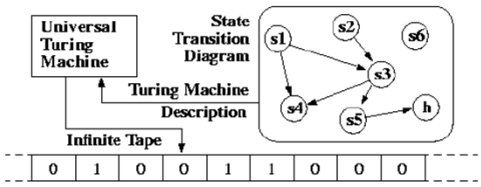
Babbage's Difference Engine

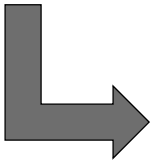
	Math & Logic	Machines
1600	**Calculus**	Leibnitz (1620) - Calculator
1800	**Taylor series (sin, cos, log)**	Charles Babbage (1820) – Difference Engine
1900	**Axiomatic Logic & Set Theory**	Alan Turing (1930s) - TM
		John von Neumann (1940s)
1945		

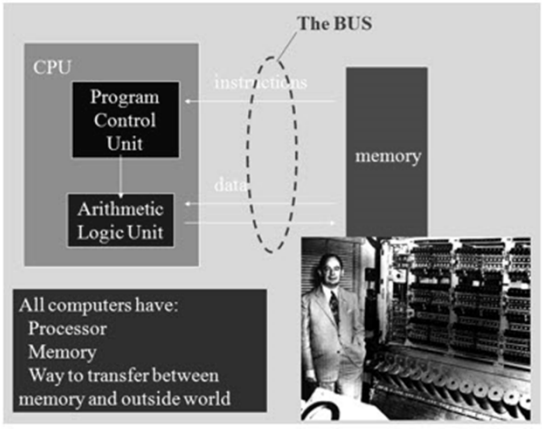


For Church it was the lambda-calculus (λ calculus)[iii] , what some have called the smallest universal programming language. It introduced nameless functions called lambda expressions, and had lots of parenthesis and recursion and took some getting accustomed to.

$$\mathsf{Y} \equiv (\lambda y.(\lambda x.y(xx))(\lambda x.y(xx)))$$

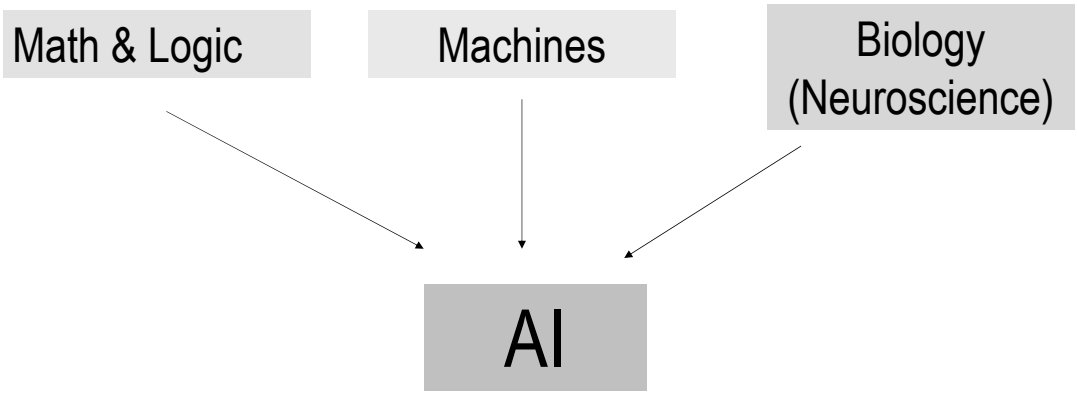




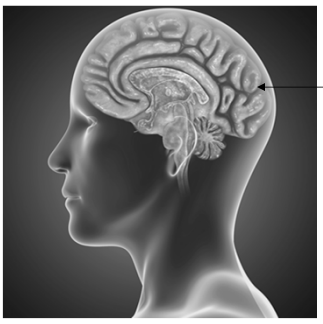


John von Neumann

The Three Dimensions of AI



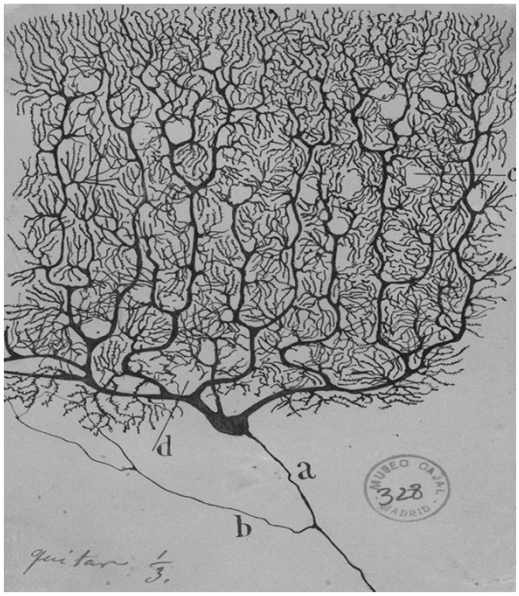
Biology (Neuroscience)

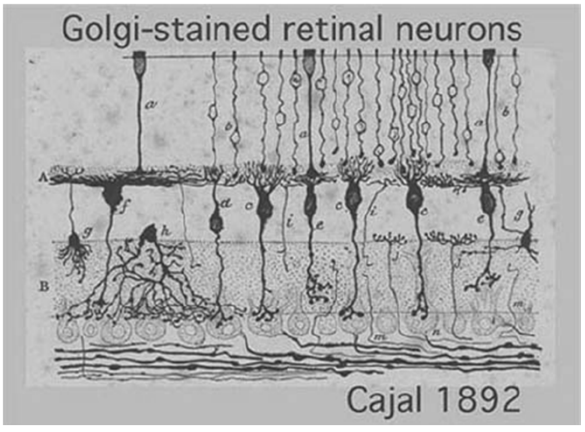


What's going on inside ??

1890s Ramon Cajal

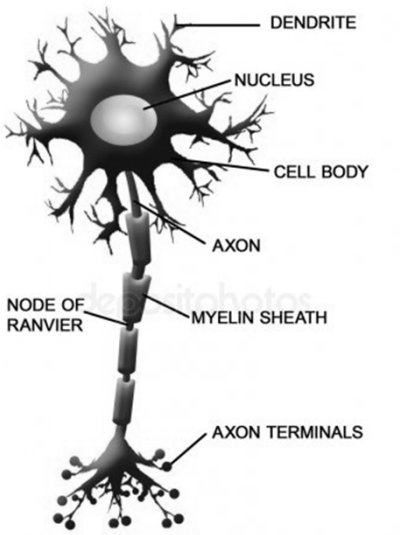
Dendrites – receptors – feed information into the cell body





BIG Question – how do electrical signals get from one cell to another ?

Ramon Cajal discovered that nerve cells were not continuous but contiguous



Input from thousands of cells

Cell will FIRE or NOT : All or none principle
Release chemical - neurotransmitter

Three Major Phases of AI

Phase I: 1600 – 1945
AI Before Digital Computers

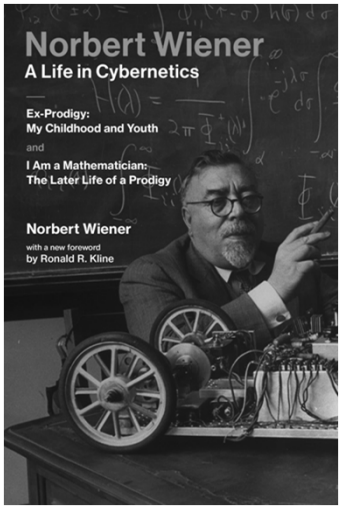
Phase 2: 1945 – 2001
AI and Computers

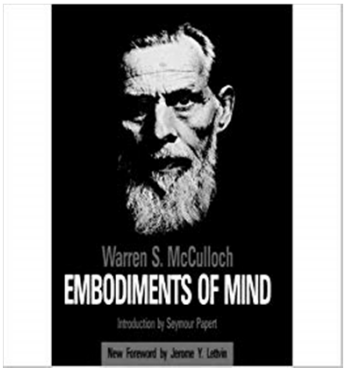
Phase 3: 2001 -
AI meets Big Data

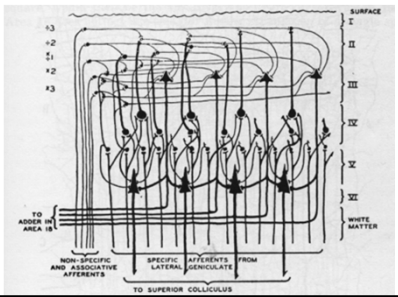
Cybernetics

Cybernetics is a transdisciplinary for exploring regulatory systems—their structures, constraints, and possibilities. Norbert Wiener defined cybernetics in 1948 as "the scientific study of control and communication in the animal and the machine."[2] In the 21st century, the term is often used in a rather loose way to imply "control of any system using technology.

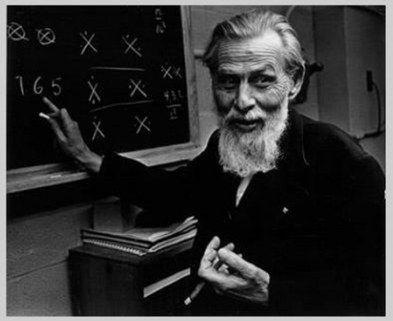
In other words, it is the scientific study of how humans, animals and machines control and communicate with each other.





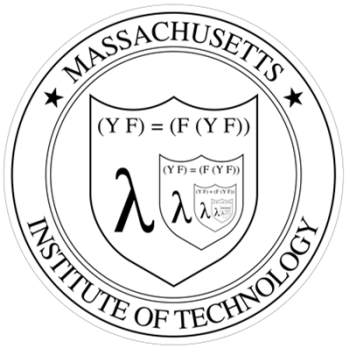


Warren McCulloch (16.11.1898 – 24.10.1969)



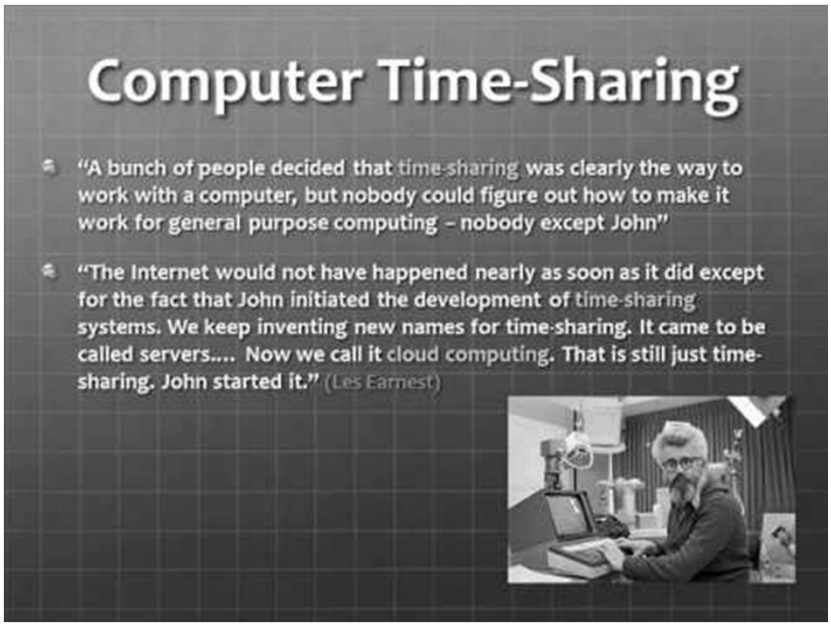
Warren Sturgis McCulloch was an American neurophysiologist and cybernetician, known for his work on the foundation for certain brain theories and his contribution to the movement cybernetics.

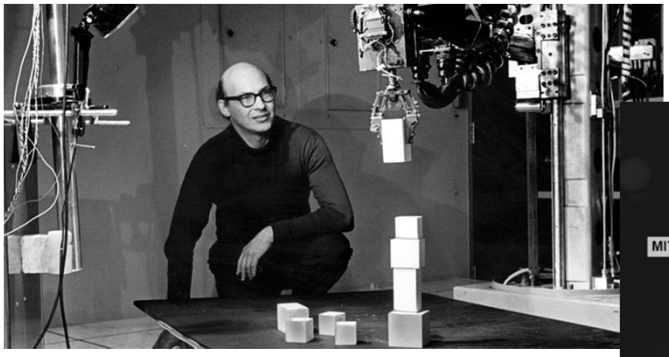
He is remembered for his work with Joannes Gregorius Dusser de Barenne from Yale and later with Walter Pitts from the University of Chicago. There he provided the foundation for certain brain theories in a number of classic papers, including "A Logical Calculus of the Ideas Immanent in Nervous Activity"



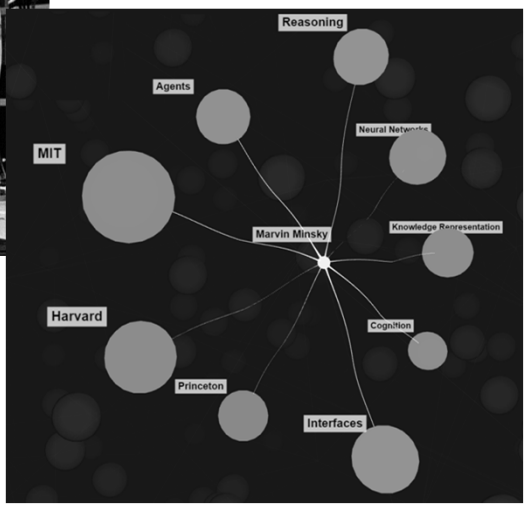
MIT 1950s

John McCarthy (MIT) : Invented the LISP programming language based on the lambda calculus to code chess programs





Marvin Minksy(MIT)



Perceptrons – Frank Rosenblatt

Background on Perceptrons

In 1958 *Psychological Review* published "The Perceptron: A Probabilistic Model For Information Storage and Organization in the Brain", by Cornell researcher Frank Rosenblatt.

> ***Perceptron is a single layer neural network*** *and a multi-layer perceptron is called Neural Networks.*

Perceptron is a linear classifier (binary). Also, it is used in supervised learning. It helps to classify the given input data. But how the heck it works ?

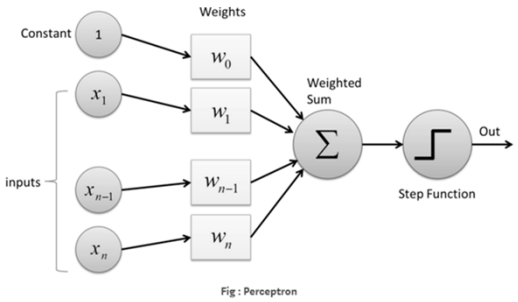
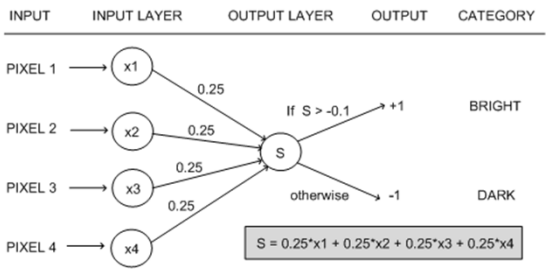


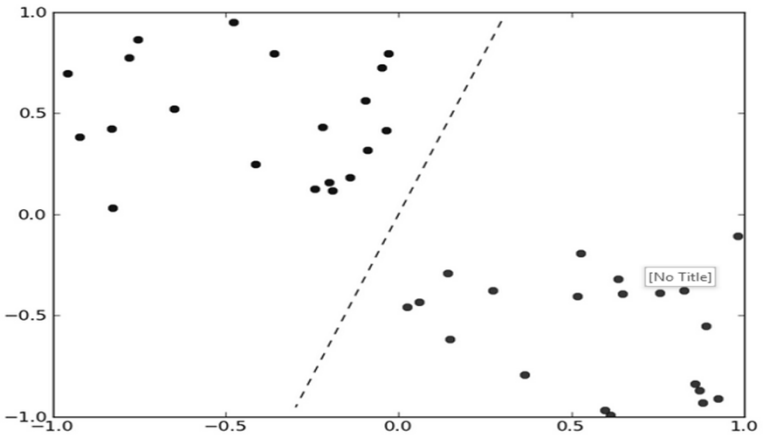
Fig : Perceptron



Why do we need Activation Function?

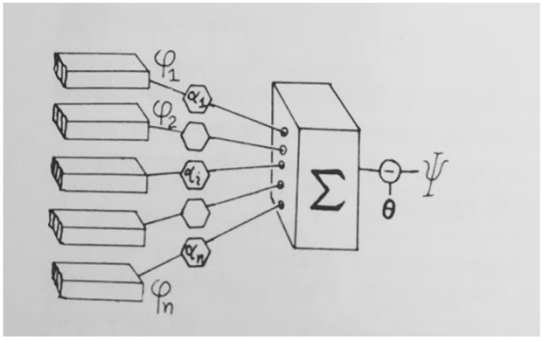
In short, the activation functions are used to map the input between the required values like (0, 1) or (-1, 1).

Perceptron is usually used to classify the data into two parts. Therefore, it is also known as a Linear Binary Classifier.



Soon after, media outlets including the New York Times and the New Yorker published articles about Rosenblatt's work. Interestingly, some of Rosenblatt's predictions regarding future of perceptrons proved to be surprisingly accurate.

> *"Later Perceptrons will be able to recognize people and call out their names and instantly translate speech in one language to speech or writing in another language, it was predicted."*
>
> *NY Times, "New Navy Device Learns By Doing" 7/7/1958*



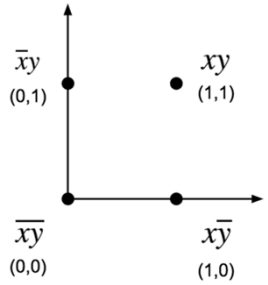
A perceptron is an approximator of linear functions (with an attached threshold function). Image source: "Perceptrons" Minsky, Papert.

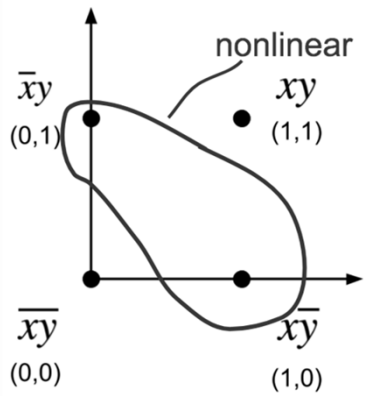
In 1969, Marvin Minsky and Seymour Papert published Perceptrons—a historic text that would alter the course of artificial intelligence research for decades.

In the text, Minsky and Papert proved that a single perceptron—a grandparent to the computational units which compose modern neural networks—was incapable of learning the exclusive-or (aka XOR) function.

Binary Pairs	XOR Output
0, 0	0
0, 1	1
1, 0	1
1, 1	0

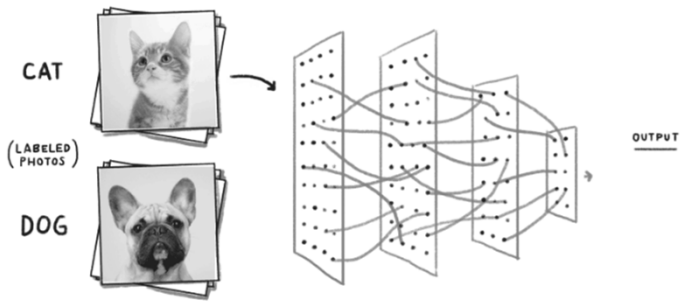
The Exclusive–Or Boolean Function.





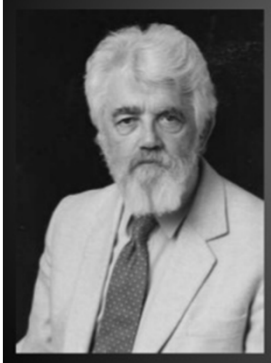
Because of Minsky's paper, researchers quit looking at Perceptrons.

Neural nets (perceptrons with multiple layers) were ignored until the 2000s with the availability of BIG DATA and GOOGLE.



HISTORY OF AI

JOHN MCCARTHY



- Precursors
- The birth of AI(1952-1956)
- The golden years(1956-1974)
- The first AI winter(1974-1980)
- Boom(1980-1987)
- Bust: second AI winter(1987-1993)
- AI(1993-2001)
- Deep learning. big data & artificial general intelligence(2001-current).

"Artificial Intelligence": 1956

A.I. term was coined during a conference at Dartmouth College by 5 leading researchers

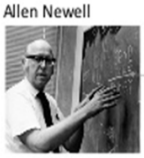
John McCarthy

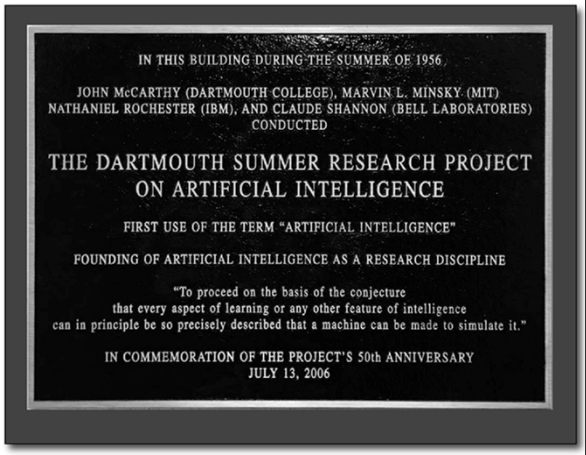
Marvin Minsky

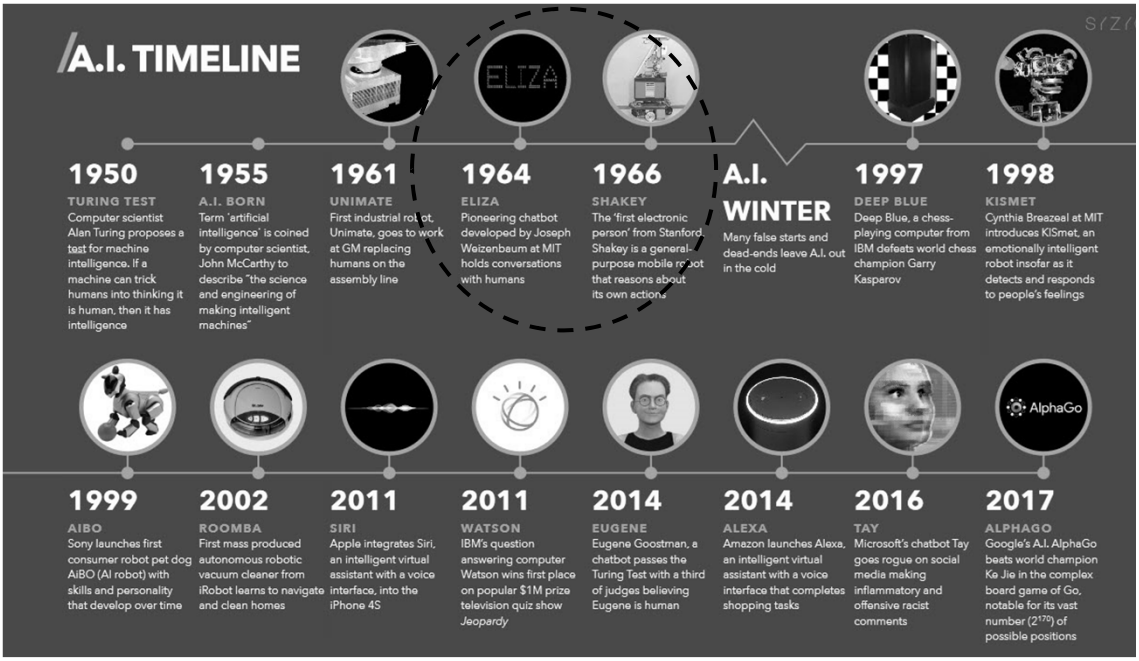
Herbert Simon

Arthur Samuel

Allen Newell

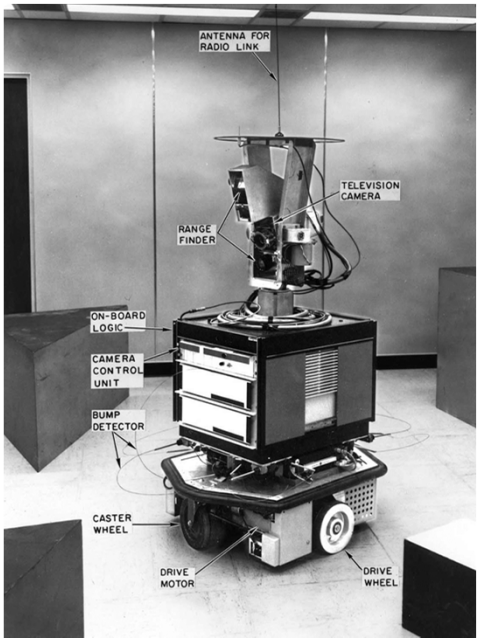




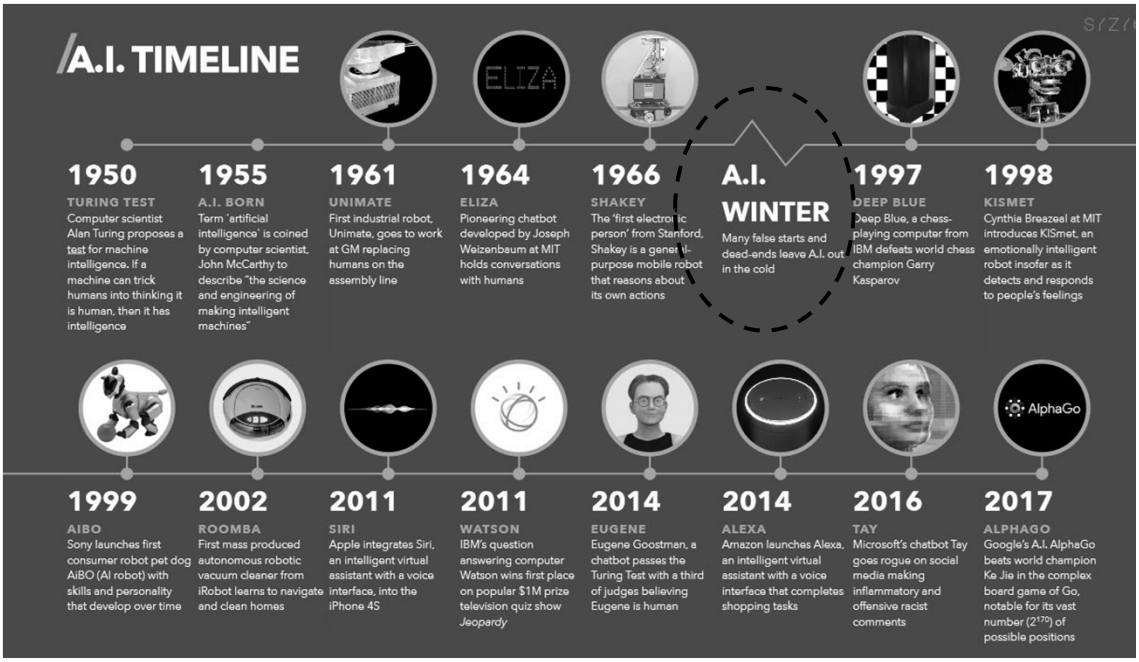


ELIZA

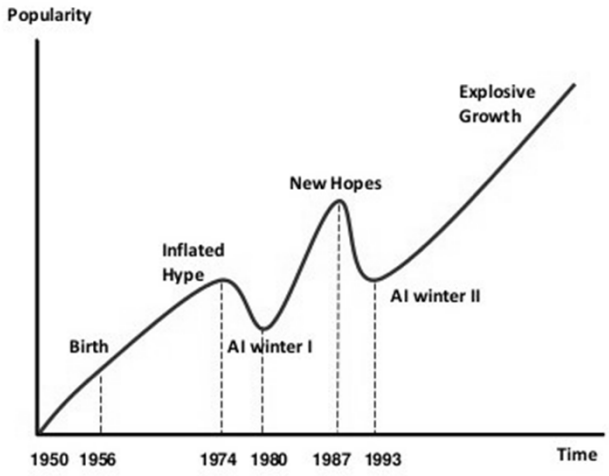
```
 Welcome to
              EEEEEE  LL      IIII  ZZZZZZZ   AAAAA
              EE      LL       II       ZZ   AA   AA
              EEEEE   LL       II     ZZZ    AAAAAAA
              EE      LL       II    ZZ      AA   AA
              EEEEEE  LLLLLL  IIII  ZZZZZZZ  AA   AA

 Eliza is a mock Rogerian psychotherapist.
 The original program was described by Joseph Weizenbaum in 1966.
 This implementation by Norbert Landsteiner 2005.


ELIZA: Is something troubling you ?
YOU:   Men are all alike.
ELIZA: What is the connection, do you suppose ?
YOU:   They're always bugging us about something or other.
ELIZA: Can you think of a specific example ?
YOU:   Well, my boyfriend made me come here.
ELIZA: Is it important to you that your boyfriend made you come here ?
YOU:   He says I'm depressed much of the time.
ELIZA: I am sorry to hear that you are depressed.
YOU:   It's true. I am unhappy.
ELIZA: Can you explain what made you unhappy ?
YOU:
```



SHAKEY
the Robot

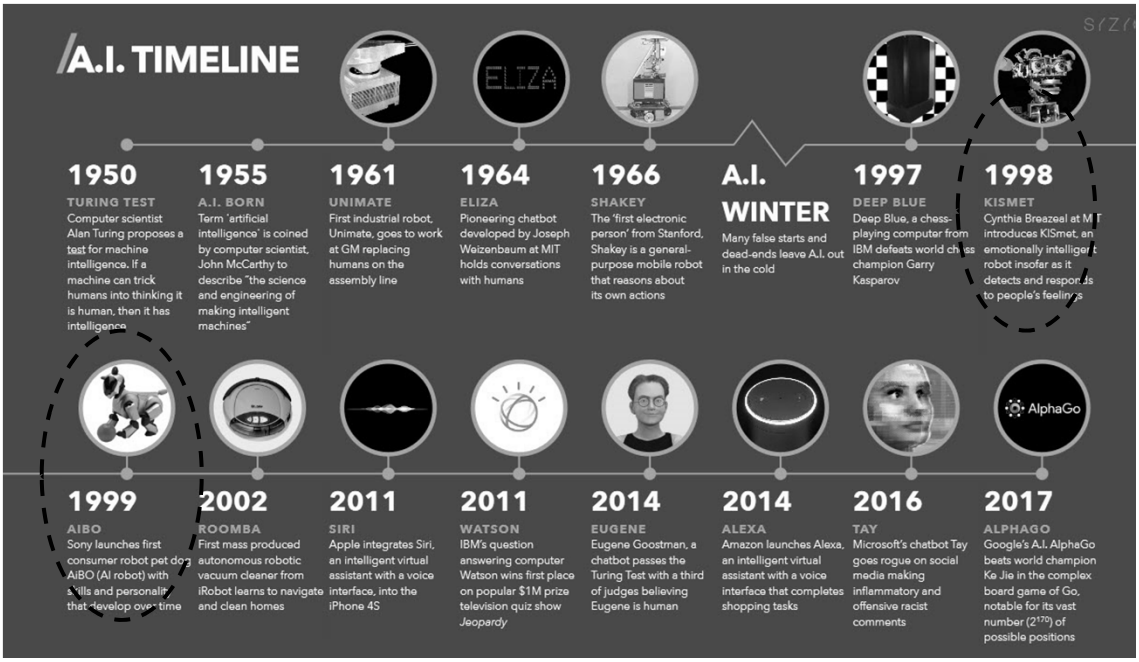


AI HAS A LONG HISTORY OF BEING "THE NEXT BIG THING"...

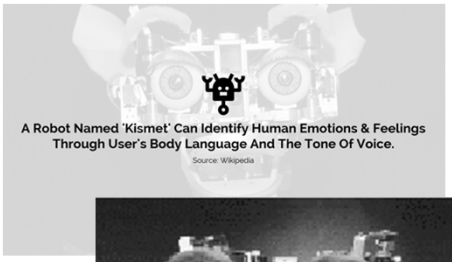


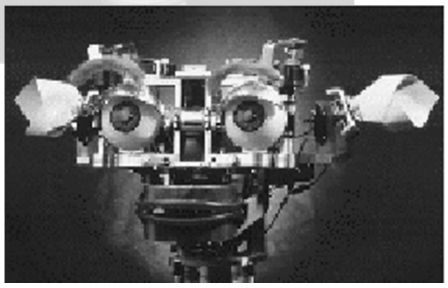
Timeline of AI Development

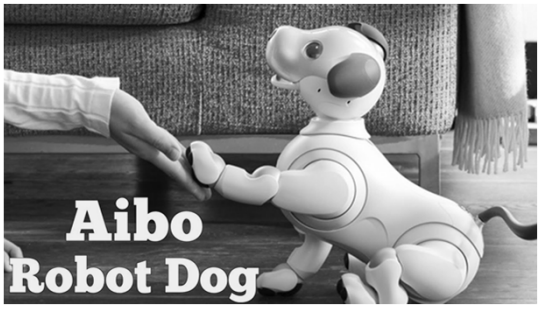
- **1950s-1960s**: First AI boom - the age of reasoning, prototype AI developed
- **1970s**: AI winter I
- **1980s-1990s**: Second AI boom: the age of Knowledge representation (appearance of expert systems capable of reproducing human decision-making)
- **1990s**: AI winter II
- **1997**: Deep Blue beats Gary Kasparov
- **2006**: University of Toronto develops Deep Learning
- **2011**: IBM's Watson won Jeopardy
- **2016**: Go software based on Deep Learning beats world's champions



Kismet





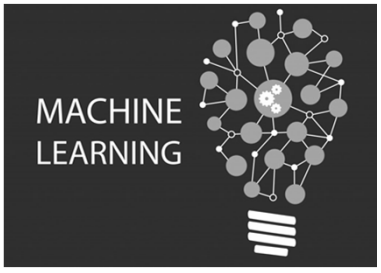


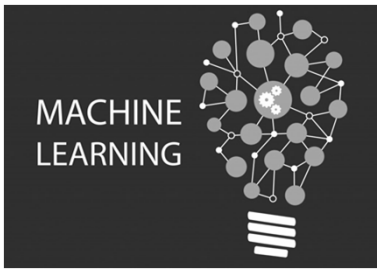
Three Major Phases of AI

Phase I: 1600 – 1945
AI Before Digital Computers

Phase 2: 1945 – 2001
AI and Computers

Phase 3: 2001 -
AI meets Big Data





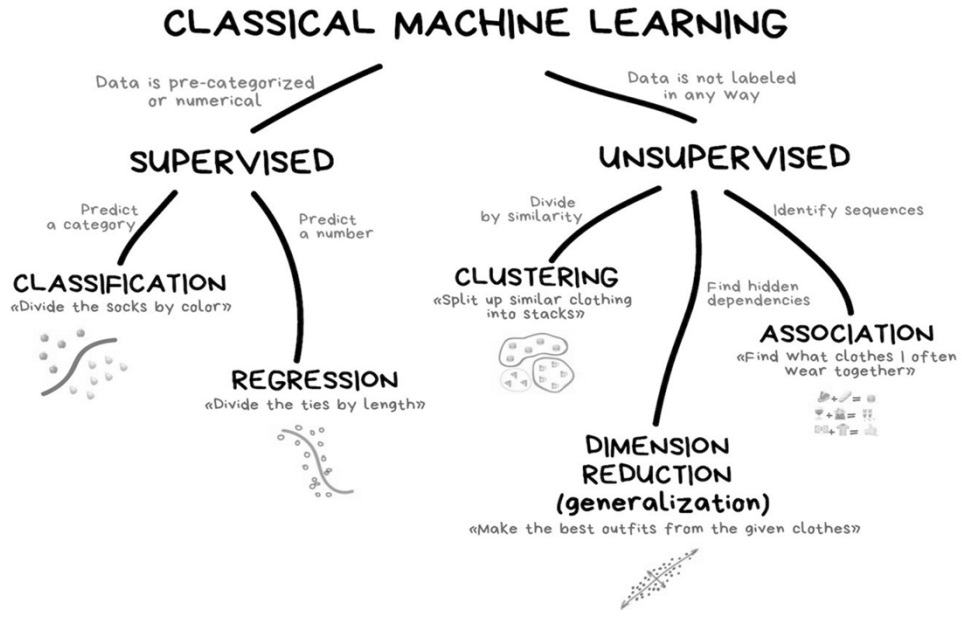
- ML – the ability to learn and improve without explicitly being programmed
- ML is part of data analytics

Business Analytics

- Data Mining
- Reports
- SQL Queries

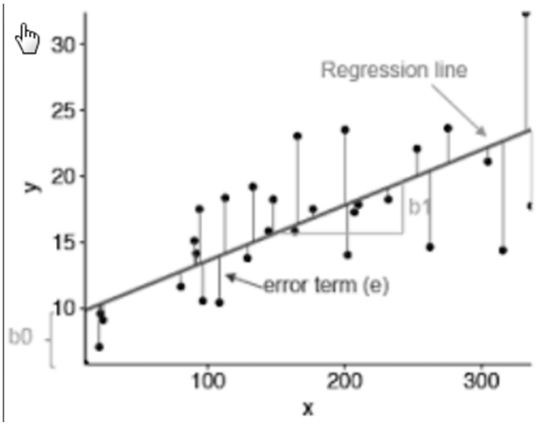
Predictive Analytics

- Likelihood answers
- Probable results
- ML with trained data sets



Linear Regression –

Use existing data to predict a value

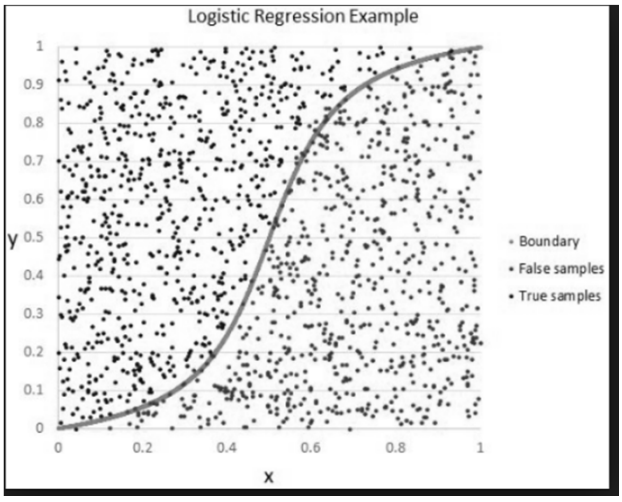


Compute a line (y = mx + b) such that you get the best fit for all data points

BUT, what if the data is not linear ??

Logistic Regression

Used to predict categories not values



Introduction

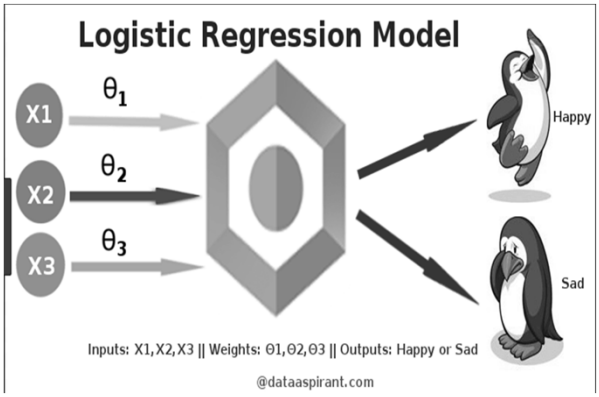
Logistic regression is a classification algorithm used to assign observations to a discrete set of classes. Unlike linear regression which outputs continuous number values, logistic regression transforms its output using the logistic sigmoid function to return a probability value which can then be mapped to two or more discrete classes.

Comparison to linear regression

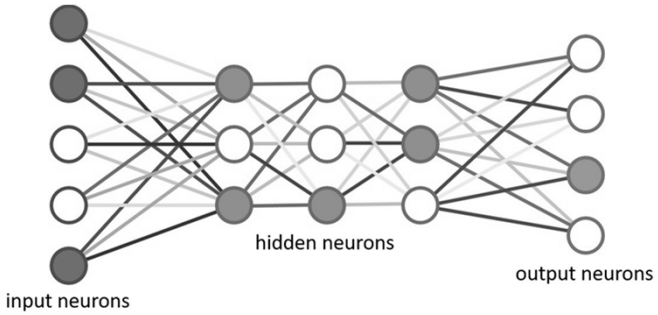
Given data on time spent studying and exam scores. Linear Regression and logistic regression can predict different things:

Linear Regression could help us predict the student's test score on a scale of 0 - 100. Linear regression predictions are continuous (numbers in a range).

Logistic Regression could help use predict whether the student passed or failed. Logistic regression predictions are discrete (only specific values or categories are allowed). We can also view probability scores underlying the model's classifications.



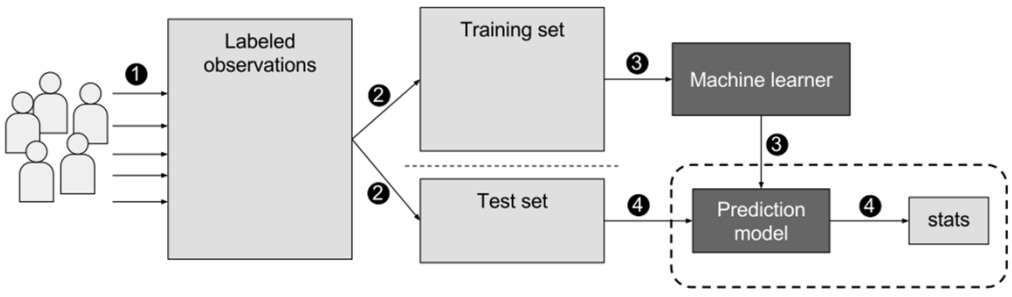
Neural Networks



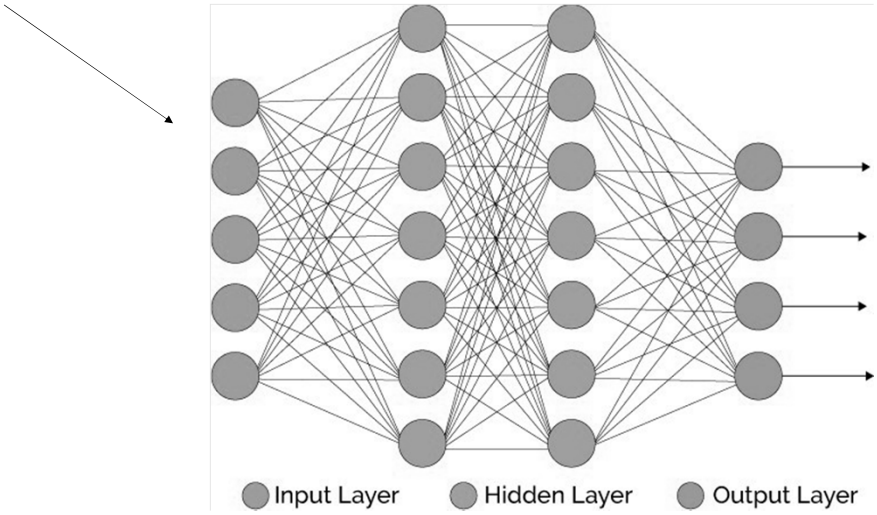
Neural Networks are just Perceptrons with hidden layers

Supervised Machine Learning

Supervised ML



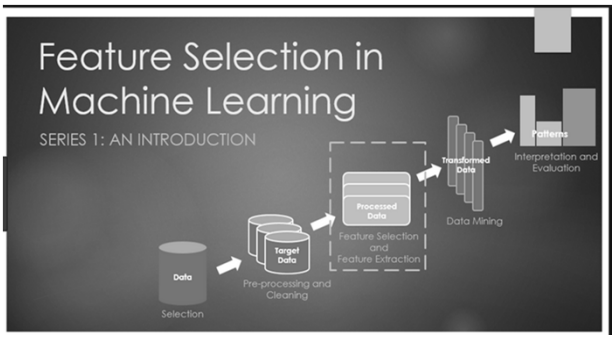
Problem: What FEATURES do feed the NN as input??

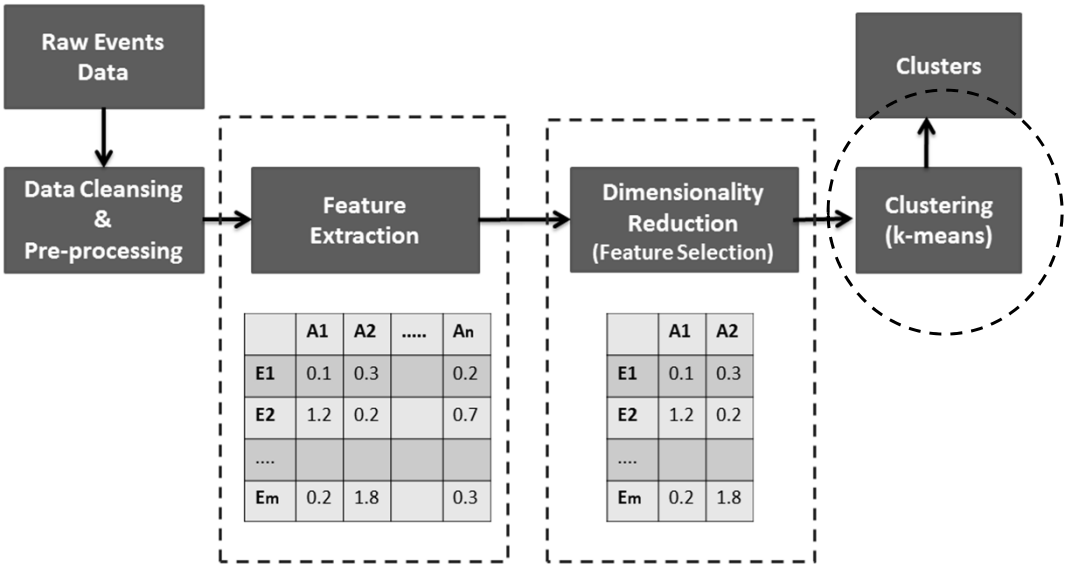


Feature Extraction & Feature Selection

Feature Selection. **Feature selection** is for filtering irrelevant or redundant **features** from your dataset.

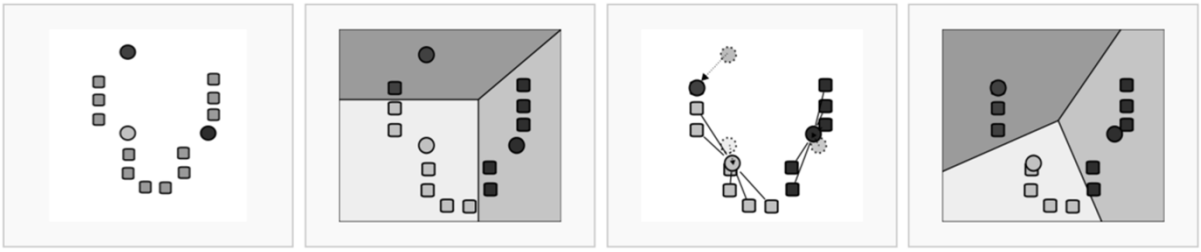
The key **difference between feature selection** and **extraction** is that **feature selection** keeps a subset of the original **features** while **feature extraction** creates brand new ones.





K-Means Clustering

Demonstration of the standard algorithm



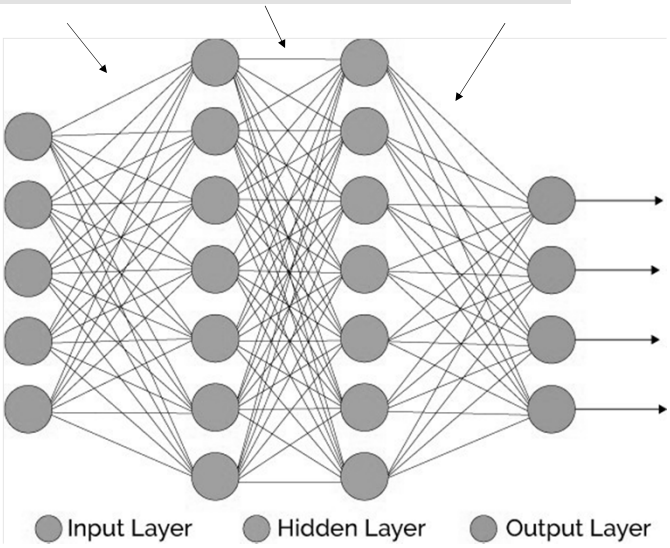
1. *k* initial "means" (in this case *k*=3) are randomly generated within the data domain (shown in color).

2. *k* clusters are created by associating every observation with the nearest mean. The partitions here represent the Voronoi diagram generated by the means.

3. The centroid of each of the *k* clusters becomes the new mean.

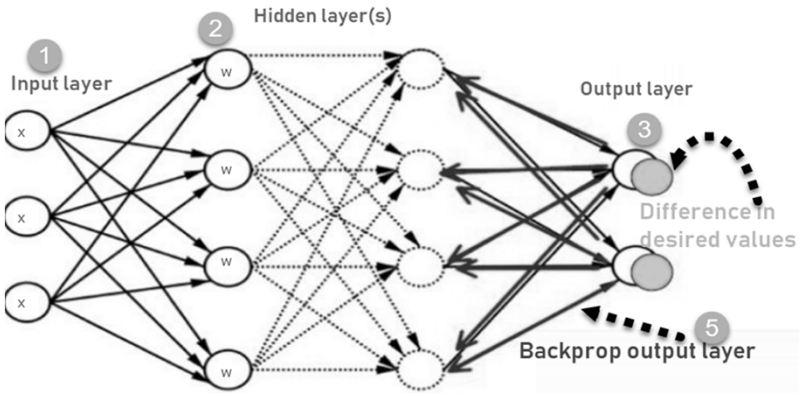
4. Steps 2 and 3 are repeated until convergence has been reached.

Problem: How do NNs determine the weights to use??



Backpropagation

Backpropagation is an **algorithm** used to calculate derivatives quickly. Artificial neural networks use **backpropagation** as a learning **algorithm** to compute a gradient descent with respect to weights. ... The **algorithm** gets its name because the weights are updated backwards, from output towards input.

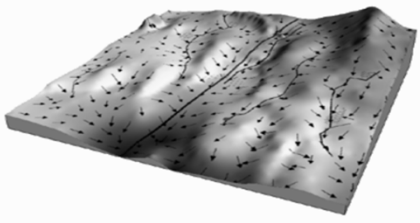


Gradient Descent

Gradient descent is an optimization algorithm used to minimize some function by iteratively moving in the direction of steepest descent as defined by the negative of the gradient. In machine learning, we use gradient descent to update the parameters of our model. Parameters refer to coefficients in Linear Regression and weights in neural networks.

Introduction

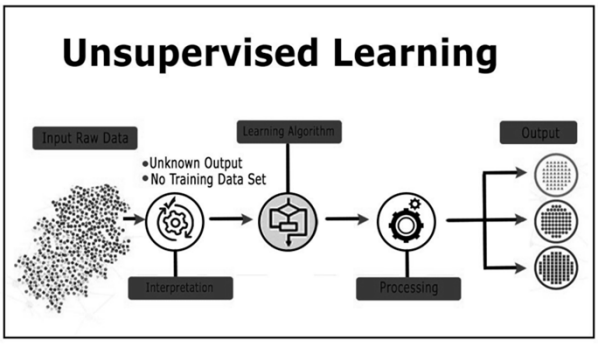
Consider the 3-dimensional graph below in the context of a cost function. Our goal is to move from the mountain in the top right corner (high cost) to the dark blue sea in the bottom left (low cost). The arrows represent the direction of steepest descent (negative gradient) from any given point–the direction that decreases the cost function as quickly as possible. Source

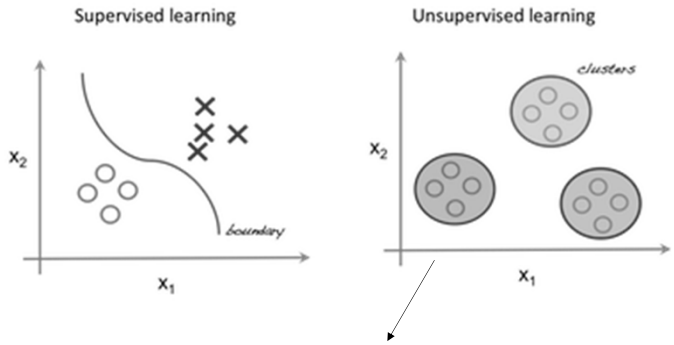


Unsupervised learning

Unsupervised learning is a type of machine **learning** algorithm used to draw inferences from datasets consisting of input data without labeled responses.

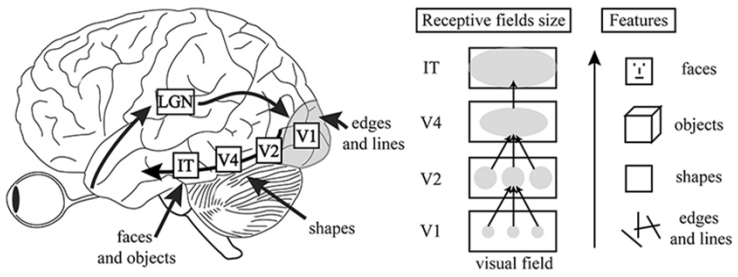
The most common **unsupervised learning** method is cluster analysis, which is used for exploratory data analysis to find hidden patterns or grouping in data.

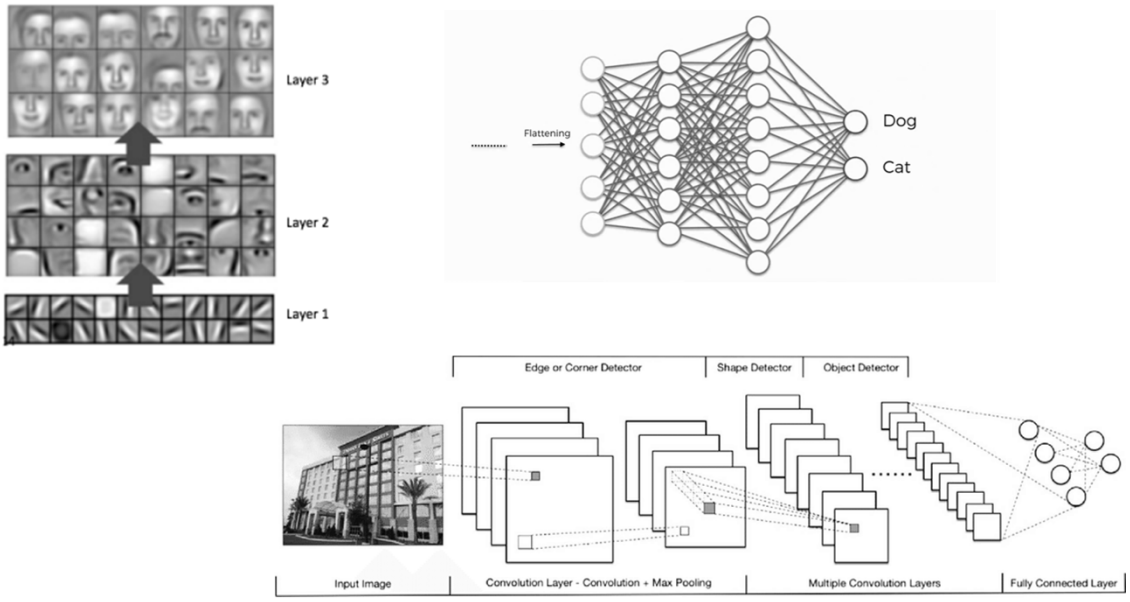




An **autoencoder** is a type of artificial neural network used to learn efficient data codings in an unsupervised manner.
The aim of an autoencoder is to learn a representation (encoding) for a set of data, typically for dimensionality reduction, by training the network to ignore signal "noise".
Recently, the autoencoder concept has become more widely used for learning generative models of data.
Some of the most powerful AI in the 2010s have involved sparse autoencoders stacked inside of deep neural networks.

Convolutional Neural Nets





Recurrent Neural Networks

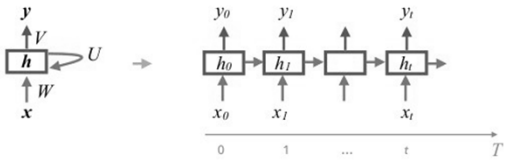
A **recurrent neural network** (**RNN**) is a class of artificial neural network where connections between nodes form a directed graph along a temporal sequence. This allows it to exhibit temporal dynamic behavior.

Unlike feedforward neural networks, RNNs can use their internal state (memory) to process sequences of inputs. This makes them applicable to tasks such as speech recognition.

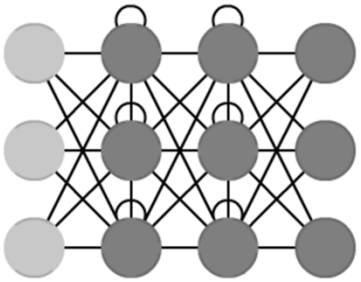
RECURRENT NEURAL NETWORKS: INTUITION

10

- **Recurrent neural network** (RNN) is a neural network model proposed in the 80's for modelling time series.
- The structure of the network is similar to feedforward neural network, with the distinction that it allows a **recurrent hidden state** whose activation at each time is dependent on that of the previous time (cycle).

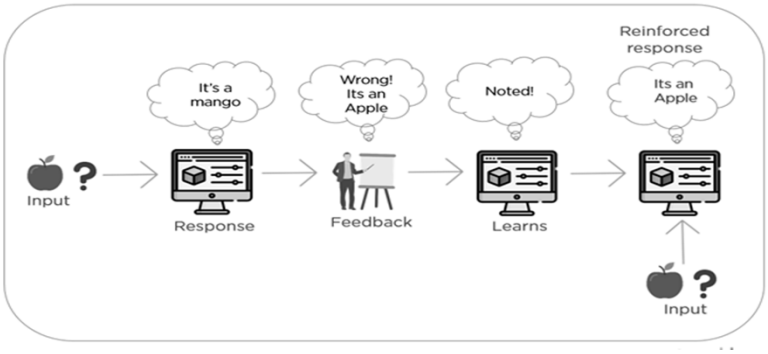


Recurrent Neural Network (RNN)

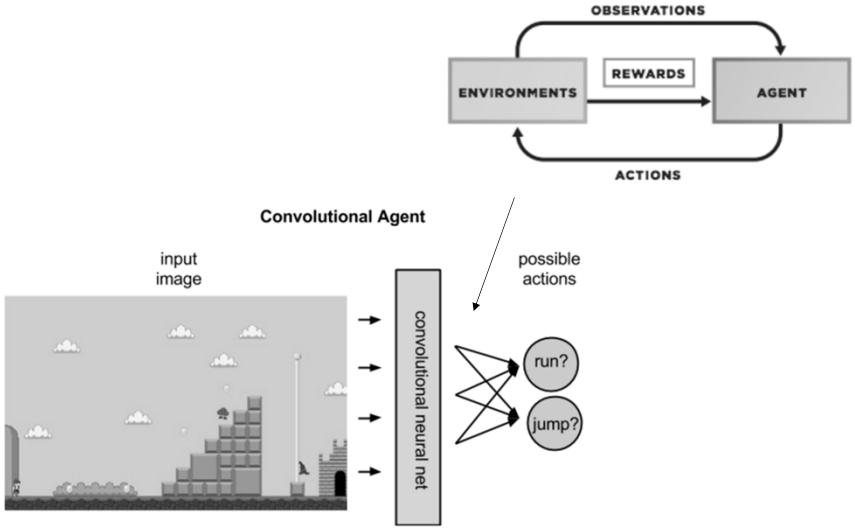


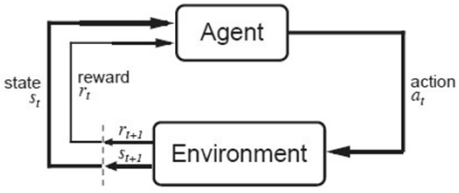
Reinforcement Learning

Reinforcement Learning(RL) is a type of machine **learning** technique that enables an agent to **learn** in an interactive environment by trial and error using feedback from its own actions and experiences.

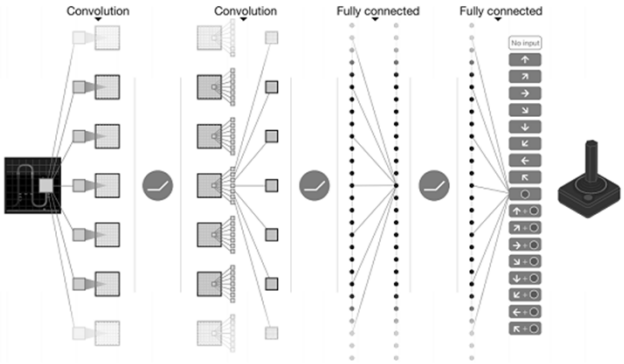


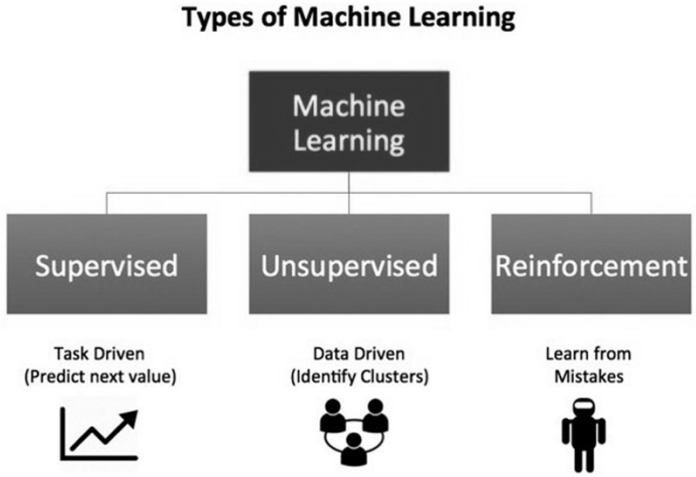
Reinforcement Learning in Games





AlphaGo

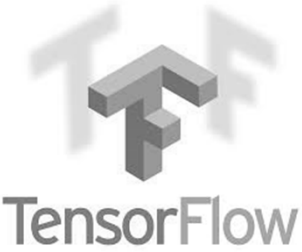




Google Brain

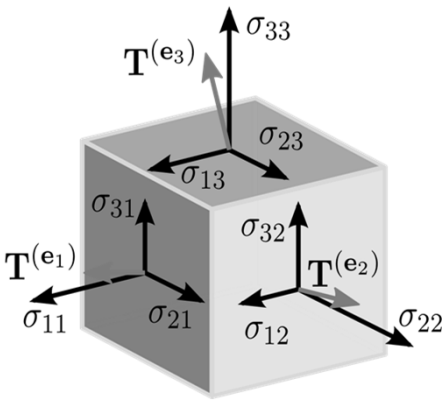
Starting in 2011, Google Brain built DistBelief as a proprietary machine learning system based on deep learning neural networks. Its use grew rapidly across diverse Alphabet companies in both research and commercial applications. Google assigned multiple computer scientists, including Jeff Dean, to simplify and refactor the codebase of DistBelief into a faster, more robust application-grade library, which became TensorFlow.

In 2009, the team, led by Geoffrey Hinton, had implemented generalized backpropagation and other improvements which allowed generation of neural networks with substantially higher accuracy, for instance a 25% reduction in errors in speech recognition.



Developed by Google

TensorFlow is a free and open-source software library for dataflow and differentiable programming across a range of tasks. It is a symbolic math library, and is also used for machine learning applications such as neural networks.



Readings:

- Wikipedia History of AI
  https://en.wikipedia.org/wiki/History_of_artificial_intelligence
- Wikipedia Machine Learning
- https://en.wikipedia.org/wiki/Machine_learning
- Wikipedia Neural Networks
- https://en.wikipedia.org/wiki/Neural_network