IA PUCP - Diplomado de Desarrollo de Aplicaciones de Inteligencia Artificial
Python para Ciencia de Datos

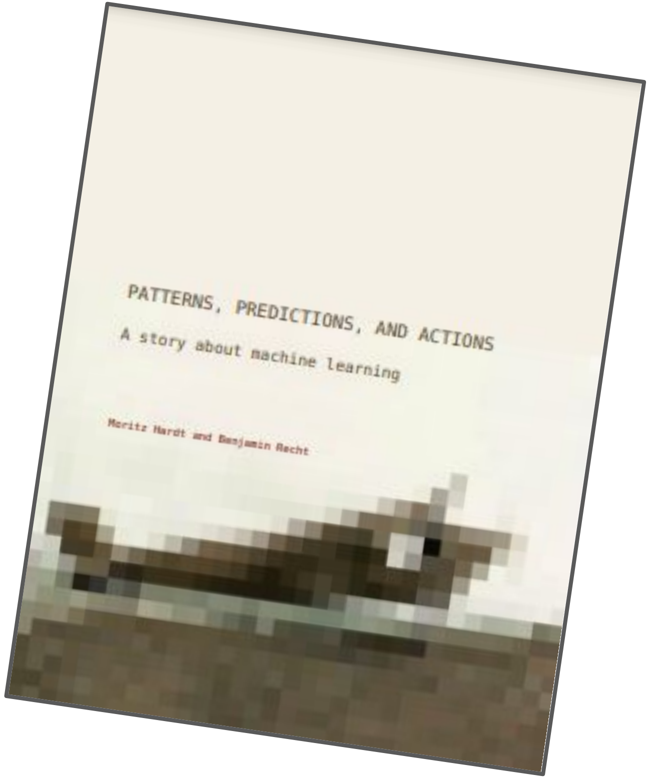
Introducción a Frameworks de Deep Learning y Redes Neuronales

Lectura Recomendada

Patterns, predictions, and actions: A story about machine learning

Moritz Hardt and Benjamin Recht

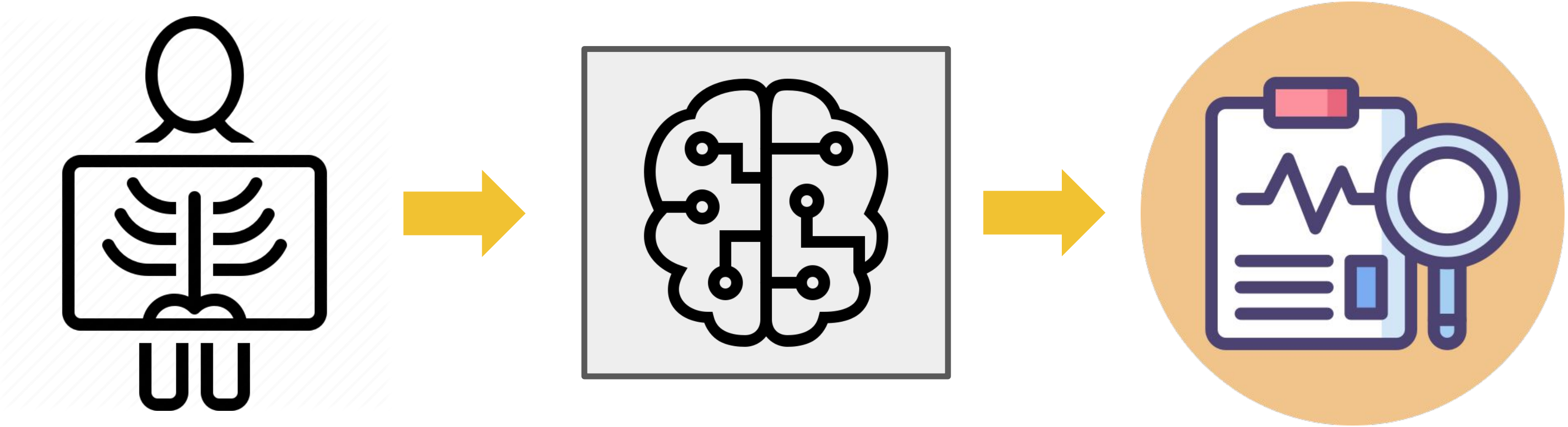
https://mlstory.org/



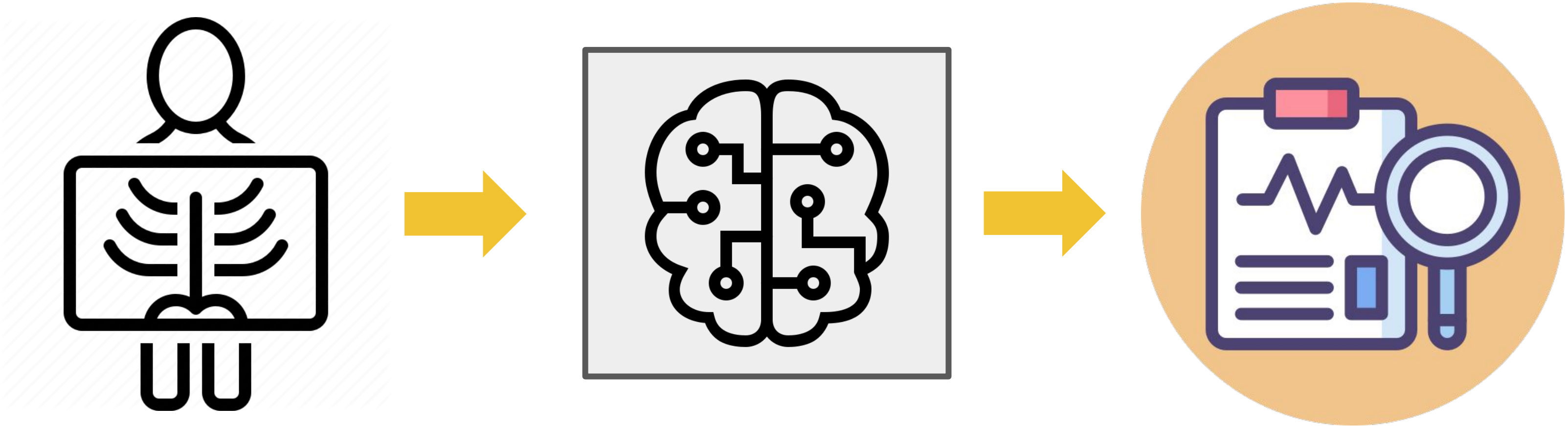
Reconocimiento de patrones

Reconocimiento de patrones

Reconocimiento de patrones



Reconocimiento de patrones



Machine Learning

¿Qué es Machine Learning?

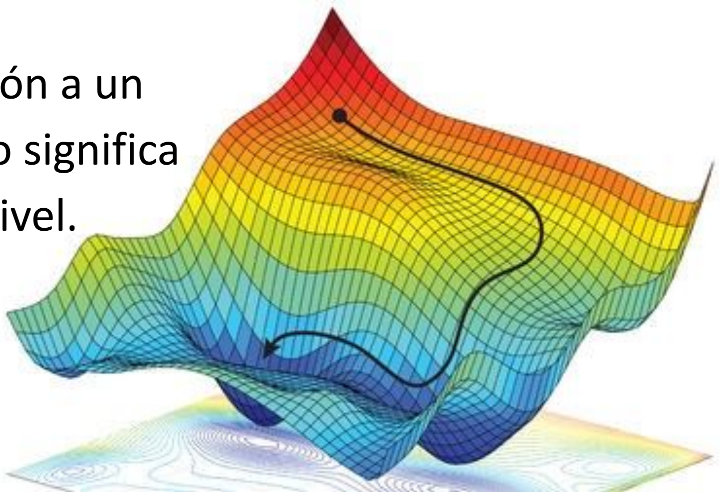
“Se puede decir que un programa aprende de una experiencia **E** con respecto a cierto tipo de tareas **T** con una medida de performance **P** si la perfomance en las tareas del tipo **T**, medidas por **P**, mejora con la experiencia **E**.”

Tom Mitchell

Convertir un problema de aprendizaje en uno de optimización

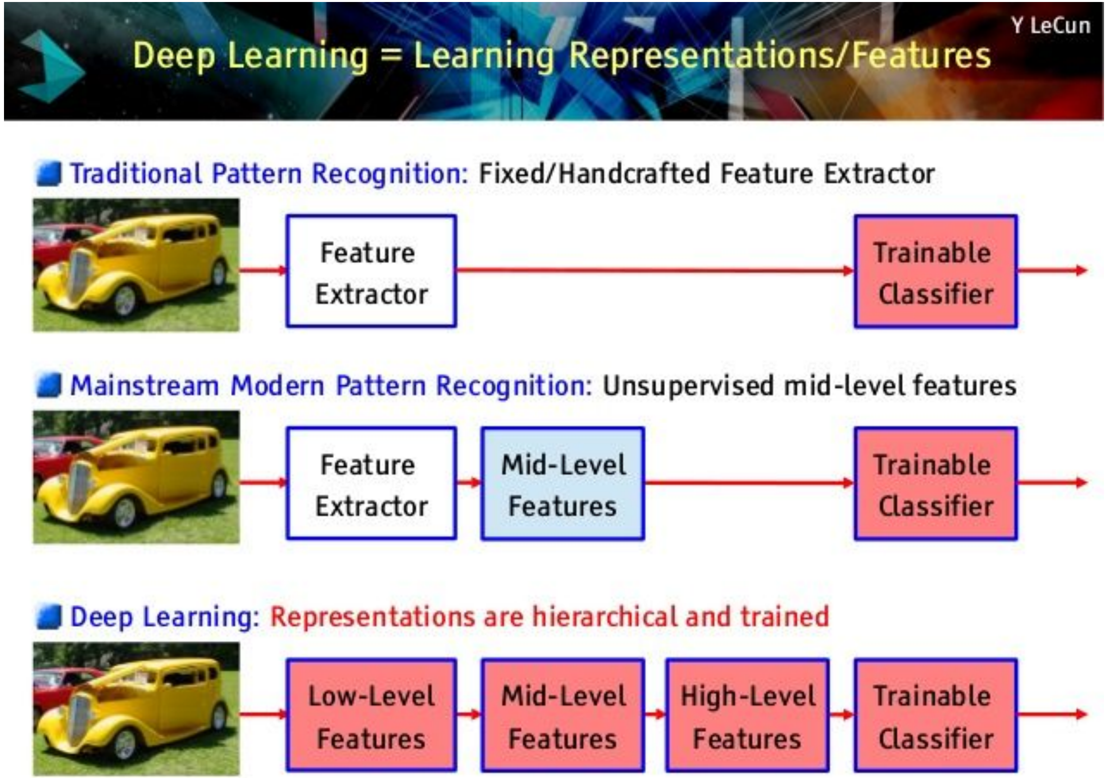
https://www.sciencemag.org/sites/default/files/styles/inline__699w__no_aspect/public/ma_0504_NID_alchemy_WEB.jpg?itok=YufMqkHI

- No debemos olvidar que los modelos de machine learning son construidos al optimizar un objetivo **(usualmente minimizar una función de pérdida)**
- Existen múltiples formas en las que un objetivo puede satisfacerse
- Podemos medir empíricamente la generalización a un conjunto de evaluación. Incluso si sale bien, no significa que se estén entendiendo conceptos de alto nivel.



Aproximaciones a la clasificación de imágenes

https://www.slideshare.net/yandex/yann-le-cun (2013)



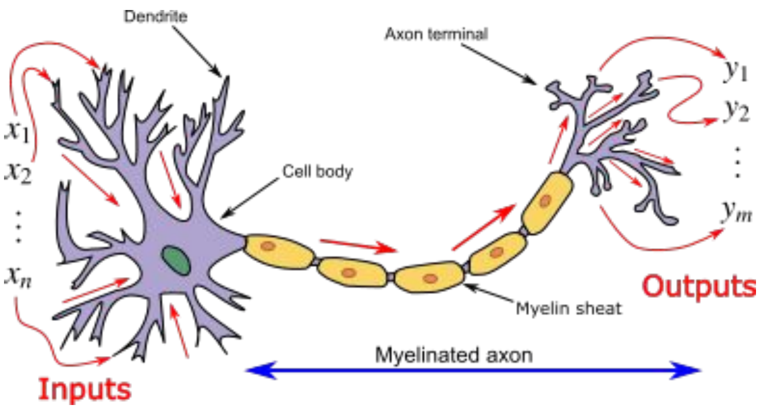
Redes Neuronales: Funciones Parametrizables

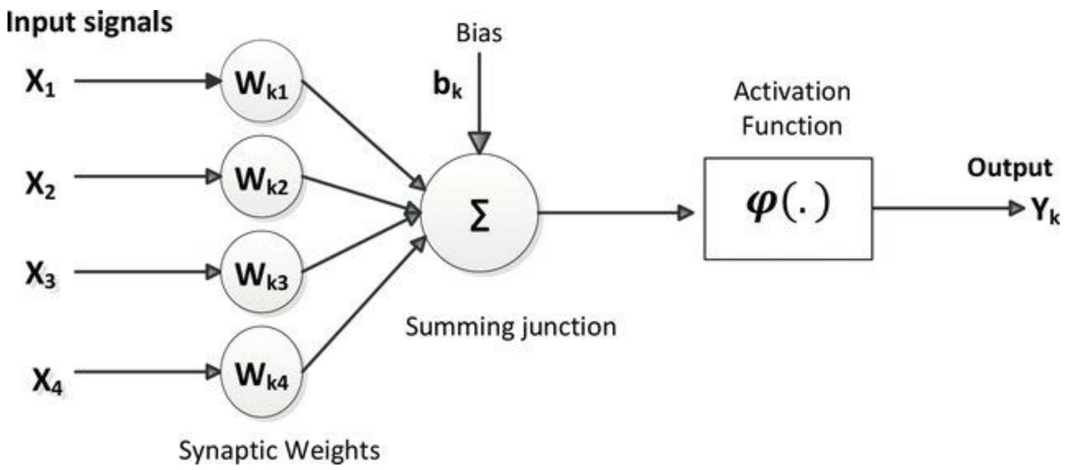
...o programas sub-especificados

Las redes neuronales pueden ser vistas como “familias de funciones”

El funcionamiento va a depender de los parámetros que escojamos

Modelo de Neurona McCulloch & Pitts (1943)

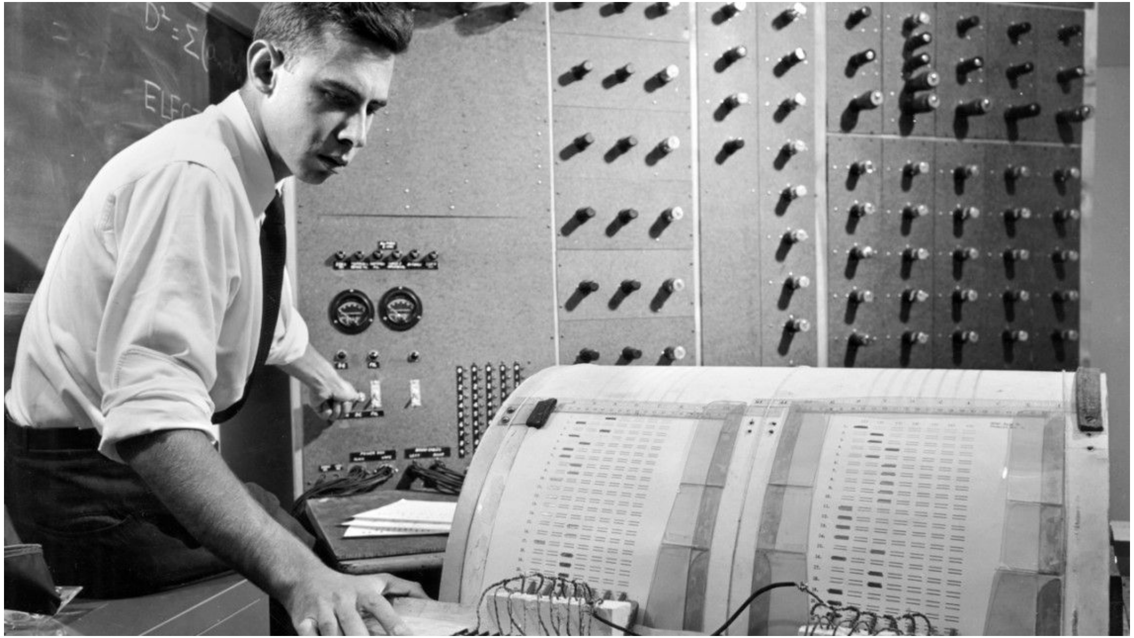




https://upload.wikimedia.org/wikipedia/commons/4/44/Neuron3.png

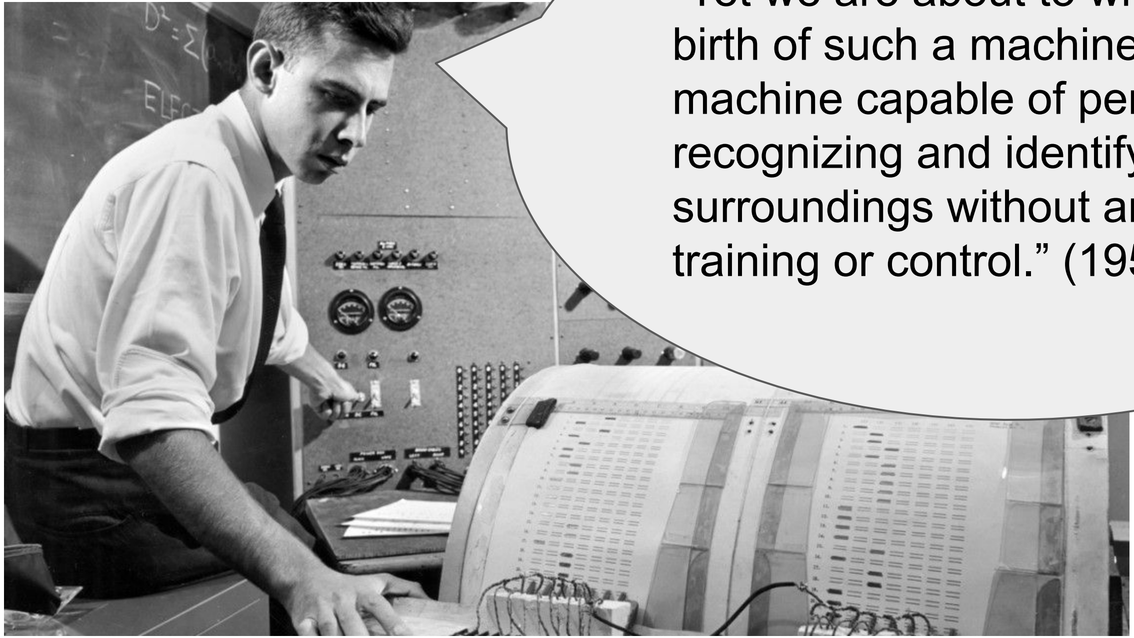
https://www.researchgate.net/publication/323465059/figure/fig2/AS:599207769554946@1519873673906/McCulloch-Pitts-computational-model-of-a-neuron_W640.jpg

Perceptrons



Division of Rare and Manuscript Collections. Frank Rosenblatt ’50, Ph.D. ’56, works on the “perceptron” – what he described as the first machine “capable of having an original idea.”
https://news.cornell.edu/stories/2019/09/professors-perceptron-paved-way-ai-60-years-too-soon

Perceptrons



"Yet we are about to witness the birth of such a machine – a machine capable of perceiving, recognizing and identifying its surroundings without any human training or control." (1958)

Division of Rare and Manuscript Collections. Frank Rosenblatt '50, Ph.D. '56, works on the "perceptron" – what he described as the first machine "capable of having an original idea."
https://news.cornell.edu/stories/2019/09/professors-perceptron-paved-way-ai-60-years-too-soon

Perceptrons

Vol. VI, No. 2, Summer 1958

research trends

CORNELL AERONAUTICAL LABORATORY, INC., BUFFALO 21, NEW YORK

The Design of an Intelligent AUTOMATON

by FRANK ROSENBLATT

Introducing the perceptron — A machine which senses, recognizes, remembers, and responds like the human mind.

STORIES about the creation of machines having human qualities have long been a fascinating province in the realm of science fiction. Yet we are now about to witness the birth of such a machine — a machine capable of perceiving, recognizing, and identifying its surroundings without any human training or control.

Development of that machine has stemmed from a search for an understanding of the physical mechanisms which underlie human experience and intelligence. The question of the nature of these processes is at least as ancient as any other question in western science and philosophy, and, indeed, ranks as one of the greatest scientific challenges of our time.

Our understanding of this problem has gone perhaps as far as had the development of physics before Newton. We have some excellent descriptions of the phenomena to be explained, a number of interesting hypotheses, and a little detailed knowledge about events in the nervous system. But we lack agreement on any integrated set of principles by which the functioning of the nervous system can be understood.

We believe now that this ancient problem is about to yield to our theoretical investigation for three reasons:

First, in recent years our knowledge of the functioning of individual cells in the central nervous system has vastly increased.

Second, large numbers of engineers and mathematicians are, for the first time, undertaking serious study of the mathematical basis for thinking, perception, and the handling of information by the central nervous system, thus providing the hope that these problems may be within our intellectual grasp.

Third, recent developments in probability theory and in the mathematics of random processes provide new tools for the study of events in the nervous system, where only the gross statistical organization is known and the precise cell-by-cell "wiring diagram" may never be obtained.

Receives Navy Support

In July, 1957, Project PARA (Perceiving and Recognizing Automaton), an internal research program which had been in progress for over a year at Cornell Aeronautical Laboratory, received the support of the Office of Naval Research. The program had been concerned primarily with the application of probability theory to

Division of Rare and Manuscript Collections. The first page of Rosenblatt's article, "The Design of an Intelligent Automaton," in Research Trends, a Cornell Aeronautical Laboratory publication, Summer 1958. https://news.cornell.edu/stories/2019/09/professors-perceptron-paved-way-ai-60-years-too-soon

Perceptrons (1958)

From the New York Times archives:

NEW NAVY DEVICE LEARNS BY DOING

Psychologist Shows Embryo of Computer Designed to Read and Grow Wiser

WASHINGTON, July 7 (UPI) —The Navy revealed the embryo of an electronic computer today that it expects will be able to walk, talk, see, write, reproduce itself and be conscious of its existence.

The embryo—the Weather Bureau's $2,000,000 "704" computer—learned to differentiate between right and left after fifty attempts in the Navy's demonstration for newsmen.,

The service said it would use this principle to build the first of its Perceptron thinking machines that will be able to read and write. It is expected to be finished in about a year at a cost of $100,000.

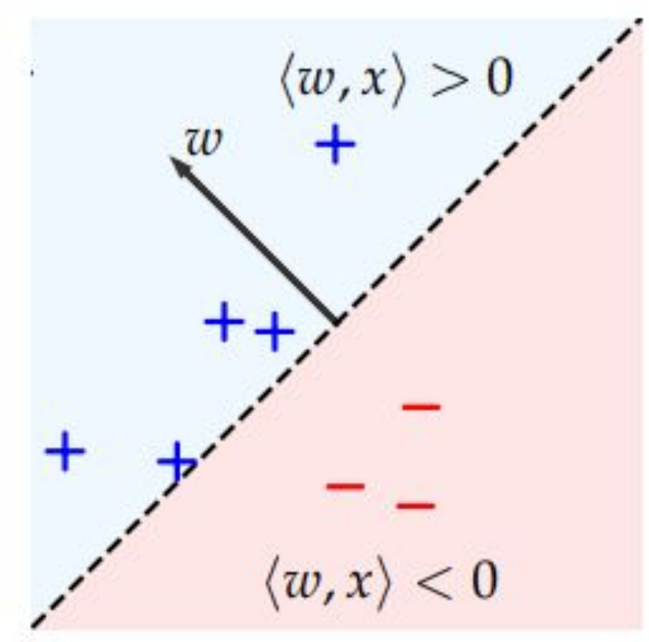
https://mlstory.org/
https://www.nytimes.com/1958/07/08/archives/new-navy-device-learns-by-doing-psychologist-shows-embryo-of.html

Perceptrons

“...be able to walk, talk, see, write, reproduce itself and be conscious of its existence.”

Perceptron: Separador Lineal

- En el caso de clasificación binaria, consideremos las clases {−1, 1}
- El algoritmo perceptrón busca encontrar un separador lineal de la data (o un hiperplano)



https://mlstory.org/

Perceptron: Separador Lineal

Division of Rare and Manuscript Collections. An image of the perceptron from Rosenblatt's “The Design of an Intelligent Automaton,” Summer 1958. https://news.cornell.edu/stories/2019/09/professors-perceptron-paved-way-ai-60-years-too-soon

FIG. 1 — Organization of a biological brain. (Red areas indicate active cells, responding to the letter X.)

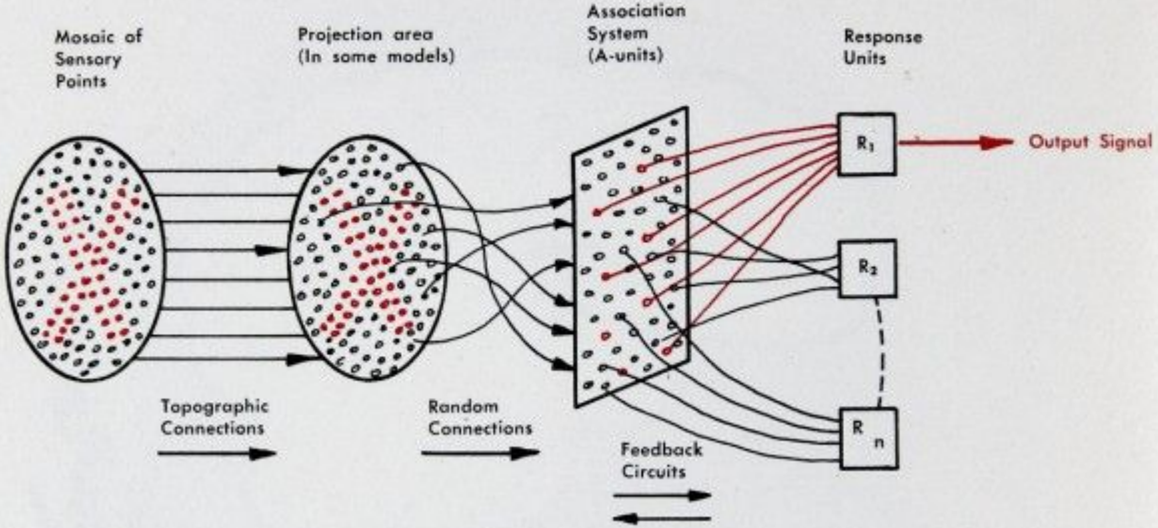
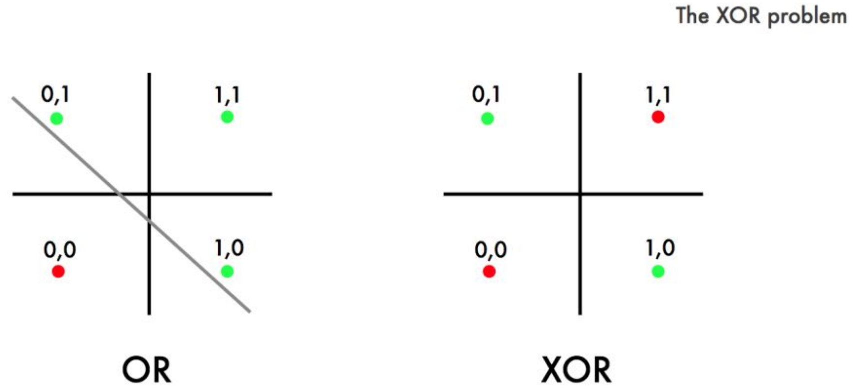


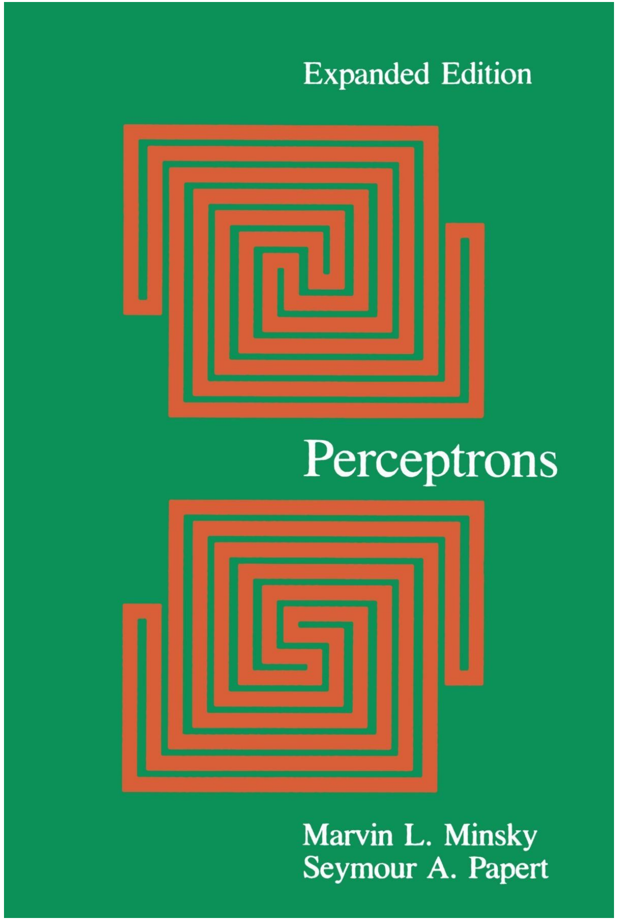
FIG. 2 — Organization of a perceptron.

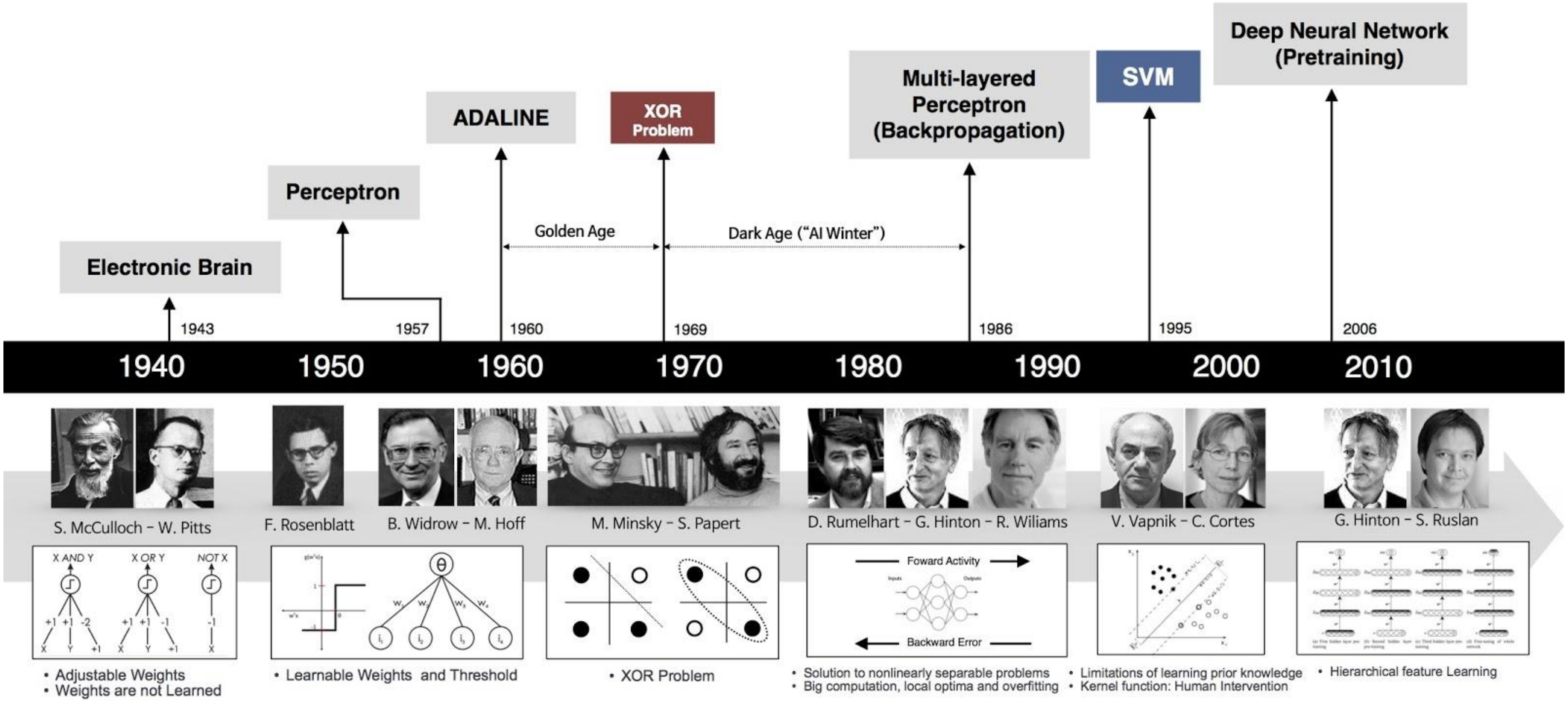
Minsky & Papert y el AI-Winter (1969)



Input 1	Input 2	Output
0	0	0
0	1	1
1	1	0
1	0	1

https://dev.to/jbahire/demystifying-the-xor-problem-1blk

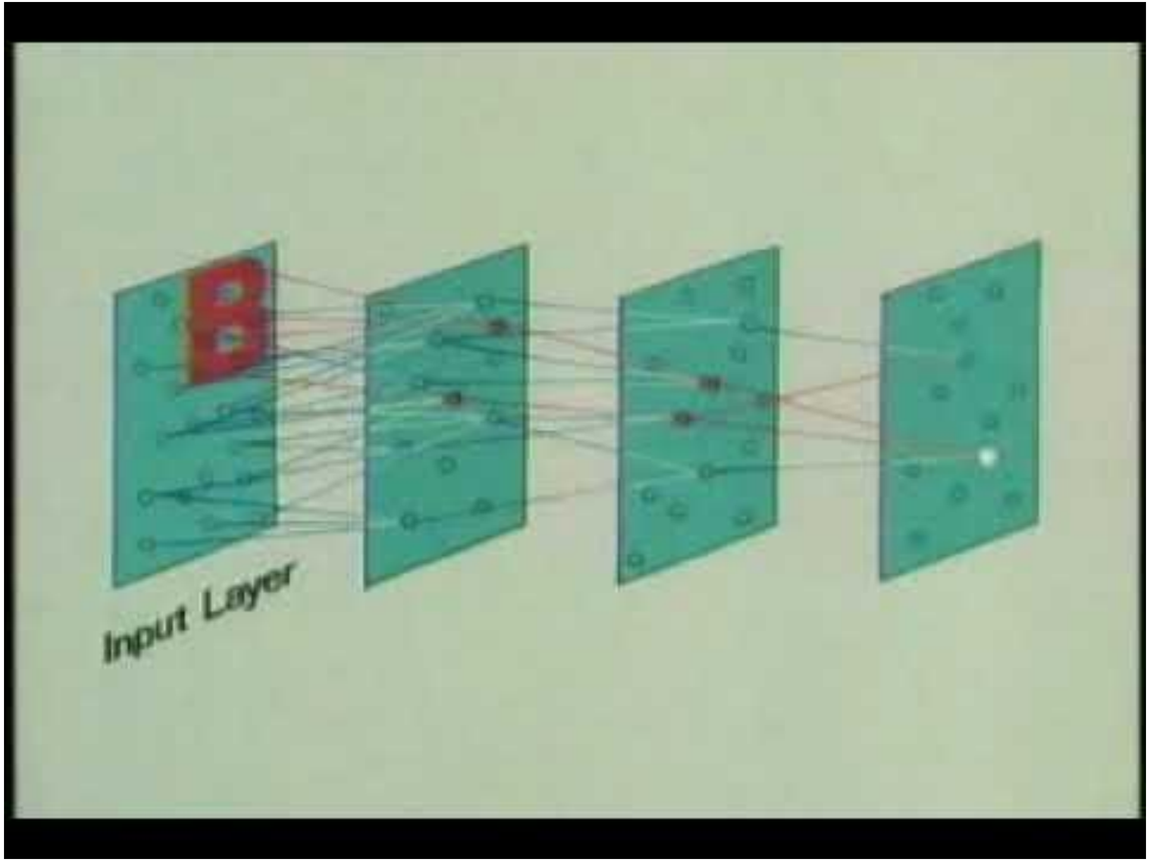



Electronic Brain
Perceptron
ADALINE
XOR Problem
Multi-layered Perceptron (Backpropagation)
SVM
Deep Neural Network (Pretraining)
Golden Age
Dark Age ("AI Winter")
1943
1957
1960
1969
1986
1995
2006
1940
1950
1960
1970
1980
1990
2000
2010
S. McCulloch – W. Pitts
F. Rosenblatt
B. Widrow – M. Hoff
M. Minsky – S. Papert
D. Rumelhart – G. Hinton – R. Wiliams
V. Vapnik – C. Cortes
G. Hinton – S. Ruslan
X AND Y
X OR Y
NOT X
Foward Activity
Backward Error
• Adjustable Weights
• Weights are not Learned
• Learnable Weights and Threshold
• XOR Problem
• Solution to nonlinearly separable problems
• Big computation, local optima and overfitting
• Limitations of learning prior knowledge
• Kernel function: Human Intervention
• Hierarchical feature Learning

Hubel & Wiesel (1959-1962)

Visual Cortex Cell Recording

Neocognitron (1980): Parte I



Neocognitron (1980): Parte II

Backpropagation (1986)

Learning representations by back-propagating errors

David E. Rumelhart*, Geoffrey E. Hinton† & Ronald J. Williams*

* Institute for Cognitive Science, C-015, University of California, San Diego, La Jolla, California 92093, USA
† Department of Computer Science, Carnegie-Mellon University, Pittsburgh, Philadelphia 15213, USA

We describe a new learning procedure, back-propagation, for networks of neurone-like units. The procedure repeatedly adjusts the weights of the connections in the network so as to minimize a measure of the difference between the actual output vector of the net and the desired output vector. As a result of the weight adjustments, internal 'hidden' units which are not part of the input or output come to represent important features of the task domain, and the regularities in the task are captured by the interactions of these units. The ability to create useful new features distinguishes back-propagation from earlier, simpler methods such as the perceptron-convergence procedure[1].

There have been many attempts to design self-organizing neural networks. The aim is to find a powerful synaptic modification rule that will allow an arbitrarily connected neural network to develop an internal structure that is appropriate for a particular task domain. The task is specified by giving the

more difficult when we introduce hidden units whose actual or desired states are not specified by the task. (In perceptrons, there are 'feature analysers' between the input and output that are not true hidden units because their input connections are fixed by hand, so their states are completely determined by the input vector: they do not learn representations.) The learning procedure must decide under what circumstances the hidden units should be active in order to help achieve the desired input–output behaviour. This amounts to deciding what these units should represent. We demonstrate that a general purpose and relatively simple procedure is powerful enough to construct appropriate internal representations.

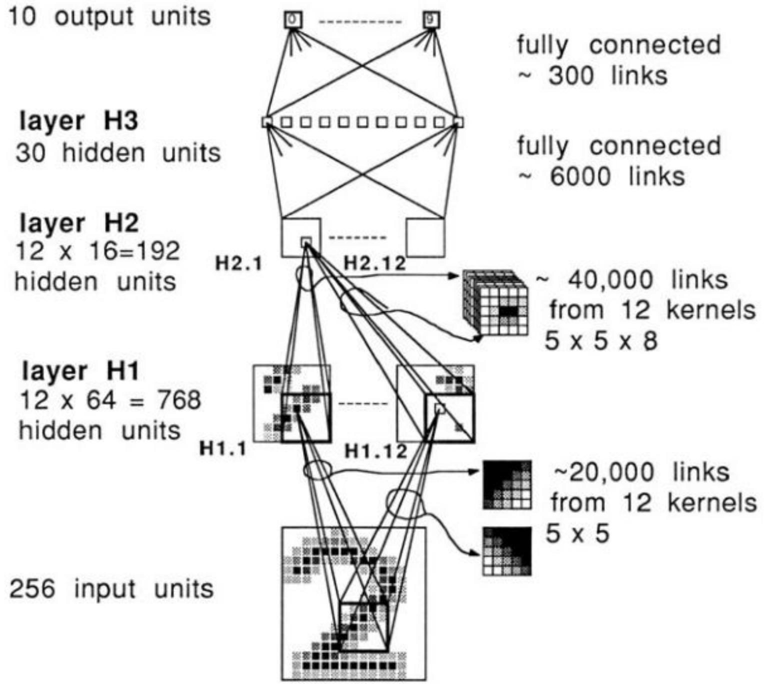
The simplest form of the learning procedure is for layered networks which have a layer of input units at the bottom; any number of intermediate layers; and a layer of output units at the top. Connections within a layer or from higher to lower layers are forbidden, but connections can skip intermediate layers. An input vector is presented to the network by setting the states of the input units. Then the states of the units in each layer are determined by applying equations (1) and (2) to the connections coming from lower layers. All units within a layer have their states set in parallel, but different layers have their states set sequentially, starting at the bottom and working upwards until the states of the output units are determined.

The total input, x_j, to unit j is a linear function of the outputs, y_i, of the units that are connected to j and of the weights, w_{ji}, on these connections

$$x_j = \sum_i y_i w_{ji} \qquad (1)$$

https://www.iro.umontreal.ca/~vincentp/ift3395/lectures/backprop_old.pdf

LeNet 5 (1988-1996)



Y. LeCun *et al*., "Backpropagation Applied to Handwritten Zip Code Recognition," in *Neural Computation*, vol. 1, no. 4, pp. 541-551, Dec. 1989, doi: 10.1162/neco.1989.1.4.541.

LeNet 5 (1988-1996)

LeNet 5 (1988-1996)

Timeline

1989	Yann LeCun et al. proposed the original form of LeNet	LeCun, Y.; Boser, B.; Denker, J. S.; Henderson, D.; Howard, R. E.; Hubbard, W. & Jackel, L. D. (1989). Backpropagation applied to handwritten zip code recognition. Neural Computation, 1(4):541-551.[1]
1989	Yann LeCun proves that minimizing the number of free parameters in neural networks can enhance the generalization ability of neural networks.	LeCun, Y.(1989). Generalization and network design strategies. Technical Report CRG-TR-89-4, Department of Computer Science, University of Toronto.[2]
1990	Their paper describes the application of backpropagation networks in handwritten digit recognition once again	LeCun, Y.; Boser, B.; Denker, J. S.; Henderson, D.; Howard, R. E.; Hubbard, W. & Jackel, L. D. (1990). Handwritten digit recognition with a back-propagation network. Advances in Neural Information Processing Systems 2 (NIPS*89).[3]
1998	They reviewed various methods applied to handwritten character recognition and compared them with standard handwritten digit recognition benchmarks. The results show that convolutional neural networks outperform all other models.	LeCun, Y.; Bottou, L.; Bengio, Y. & Haffner, P. (1998). Gradient-based learning applied to document recognition.Proceedings of the IEEE. 86(11): 2278 - 2324.[4]

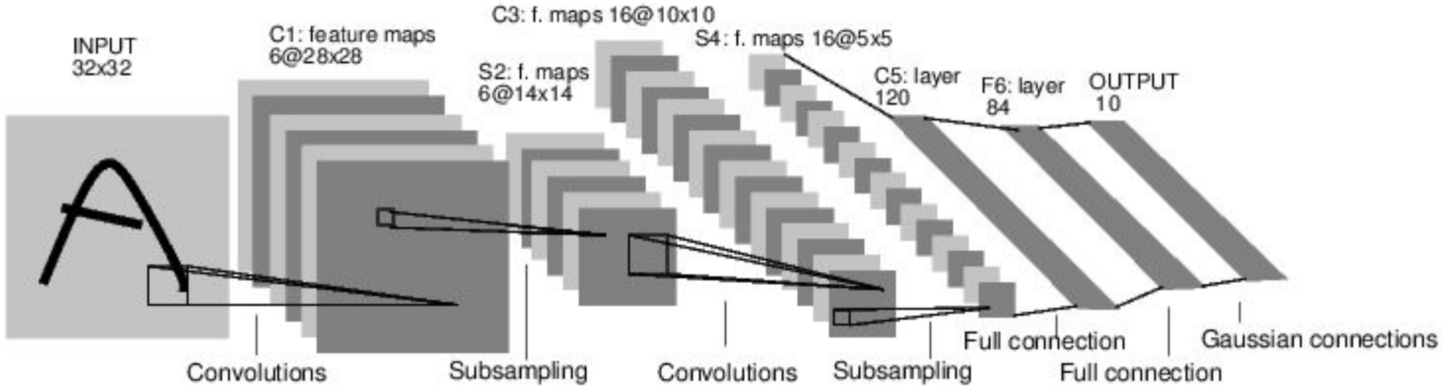
https://en.wikipedia.org/wiki/LeNet

LeNet 5 (1988-1996)

Layer		Feature Map	Size	Kernel Size	Stride	Activation
Input	Image	1	32x32	-	-	-
1	Convolution	6	28x28	5x5	1	tanh
2	Average Pooling	6	14x14	2x2	2	tanh
3	Convolution	16	10x10	5x5	1	tanh
4	Average Pooling	16	5x5	2x2	2	tanh
5	Convolution	120	1x1	5x5	1	tanh
6	FC	-	84	-	-	tanh
Output	FC	-	10	-	-	softmax

https://www.datasciencecentral.com/profiles/blogs/lenet-5-a-classic-cnn-architecture#:~:text=The%20LeNet%2D5%20architecture%20consists,and%20finally%20a%20softmax%20classifier .

LeNet 5 (1988-1996)



Y. Le Cun et. al, 1998

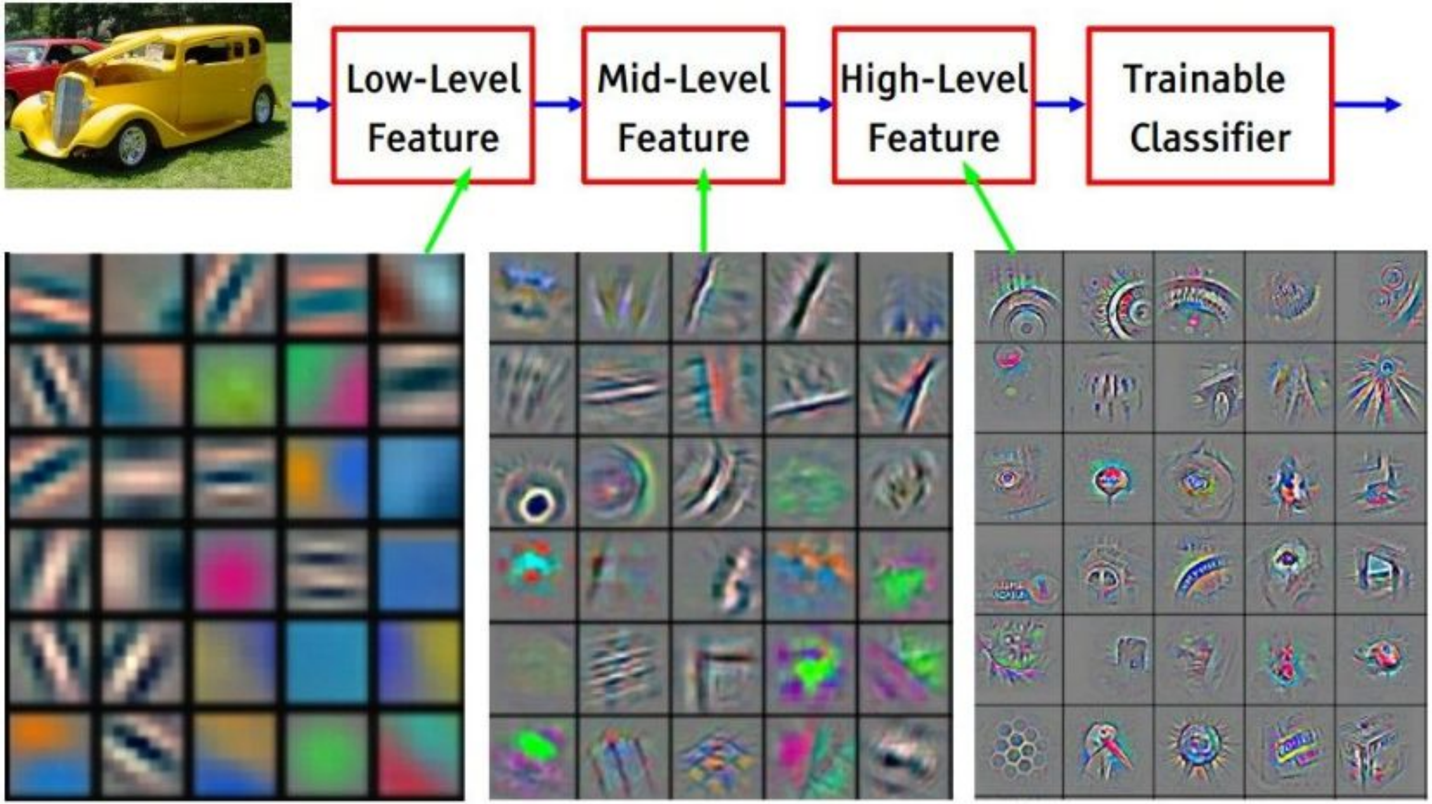
AlexNet (2012)

Layer		Feature Map	Size	Kernel Size	Stride	Activation
Input	Image	1	227x227x3	-	-	-
1	Convolution	96	55 x 55 x 96	11x11	4	relu
	Max Pooling	96	27 x 27 x 96	3x3	2	relu
2	Convolution	256	27 x 27 x 256	5x5	1	relu
	Max Pooling	256	13 x 13 x 256	3x3	2	relu
3	Convolution	384	13 x 13 x 384	3x3	1	relu
4	Convolution	384	13 x 13 x 384	3x3	1	relu
5	Convolution	256	13 x 13 x 256	3x3	1	relu
	Max Pooling	256	6 x 6 x 256	3x3	2	relu
6	FC	-	9216	-	-	relu
7	FC	-	4096	-	-	relu
8	FC	-	4096	-	-	relu
Output	FC	-	1000	-	-	Softmax

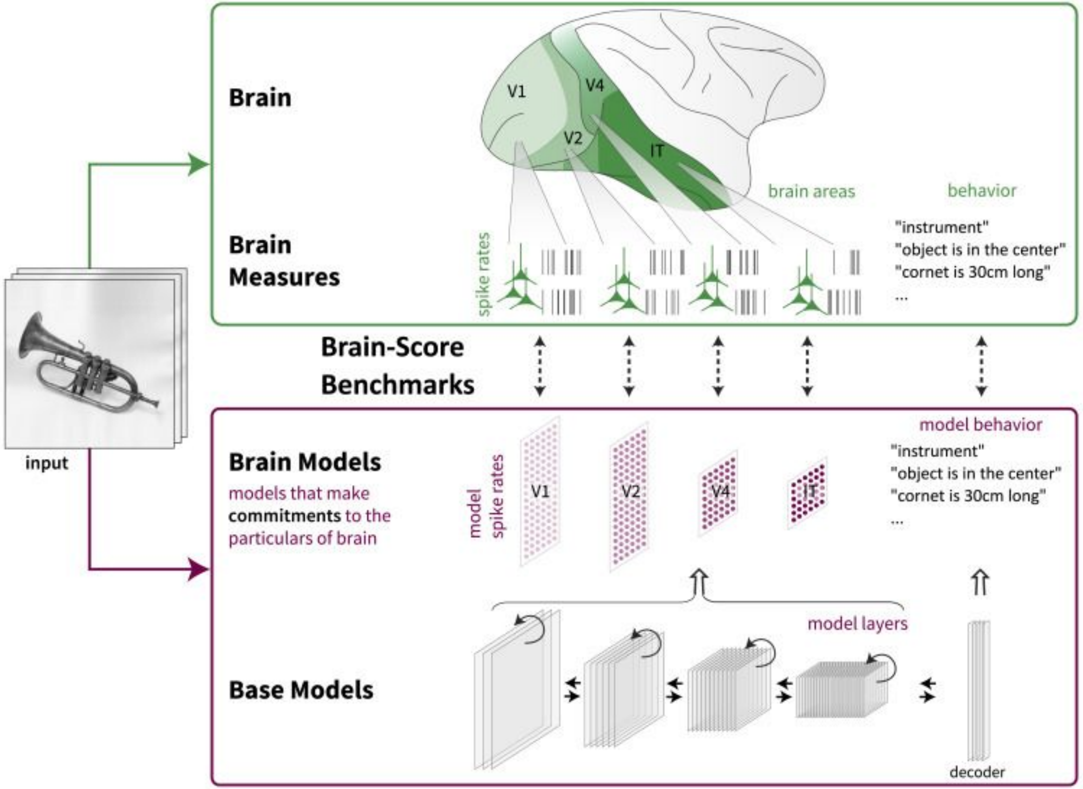
https://www.datasciencecentral.com/profiles/blogs/alexnet-implementation-using-keras

Representaciones jerárquicas

Zeiler, M. D., & Fergus, R. (2014, September). Visualizing and understanding convolutional networks. In *European conference on computer vision* (pp. 818-833). Springer, Cham.



Brainscore



Integrative Benchmarking to Advance Neurally Mechanistic Models of Human Intelligence

Martin Schrimpf, Jonas Kubilius, Michael J. Lee, N. Apurva Ratan Murty, Robert Ajemian, James J. DiCarlo

Neuron
Volume 108 Issue 3 Pages 413-423
(November 2020)
DOI: 10.1016/j.neuron.2020.07.040

Brainscore

Brain-Score   Leaderboard   About   Compare   Participate

Rank	Model *submitted by*	average	V1 1 benchmark	V2 1 benchmark	V4 1 benchmark	IT 2 benchmarks	behavior 1 benchmark	engineering 1 benchmark	Deng2009-top1 v1
1	CORnet-S *Brain-Score Team*	.417	.294	.242	.581	.423	.545	.747	.747
2	vgg-19 *Brain-Score Team*	.408	.347	.341	.610	.248	.494	.711	.711
3	resnet-50-robust *Joel Dapello*	.408	.378	.365	.537	.243	.515		
4	resnet-101_v1 *Brain-Score Team*	.407	.266	.341	.590	.274	.561	.764	.764
5	vgg-16 *Brain-Score Team*	.406	.355	.336	.620	.259	.461	.715	.715
6	resnet-152_v1 *Brain-Score Team*	.405	.282	.338	.598	.277	.533	.768	.768
7	resnet-101_v2 *Brain-Score Team*	.404	.274	.332	.599	.263	.555	.774	.774
8	resnet50-SIN_IN *Brain-Score Team*	.404	.282	.324	.599	.276	.541	.746	.746
9	densenet-169 *Brain-Score Team*	.404	.281	.322	.601	.274	.543	.759	.759
10	densenet-201 *Brain-Score Team*	.402	.277	.325	.599	.273	.537	.772	.772
11	resnet-50-pytorch *Joel Dapello*	.399	.289	.317	.600	.259	.528	.752	.752

http://www.brain-score.org/#leaderboard

Código en Keras

```
from tensorflow.keras.layers import Dense, Conv2D, MaxPool2D, Flatten

model = Sequential([
    Conv2D(16, 3, activation='relu', input_shape=(28,28,1)),
    MaxPool2D(),
    Conv2D(32, 3, activation='relu'),
    MaxPool2D(),
    Flatten(),
    Dense(10, activation='softmax')
])
```