UNIVERSITAT DE BARCELONA

DATA SCIENCE

DEEP LEARNING FROM SCRATCH

or how to train large and highly complex models with deeply cascaded nonlinearities by using automatic differentiation and several tricks.

Oriol Pujol, Santi Seguí, Jordi Vitrià

Our Background

DATASCIENCE

Telefónica · GE Healthcare · Fomento del Trabajo Nacional · TÜVRheinland · Ajuntament de Barcelona · 500 · Universitat de Barcelona · AOC · Invalcor Grupo · BBVA · 7ideas · Fira de Barcelona · eurecat · leadtochange · correoyvuela

PyData

DATA BEERS BCN

Oriol Pujol,
Associate Professor at UB.

MACHINE LEARNING RESEARCH

COMPUTER VISION RESEARCH

Santi Seguí,
Lecturer at UB.

Jordi Vitrià.
Full Professor at UB.

Course Agenda

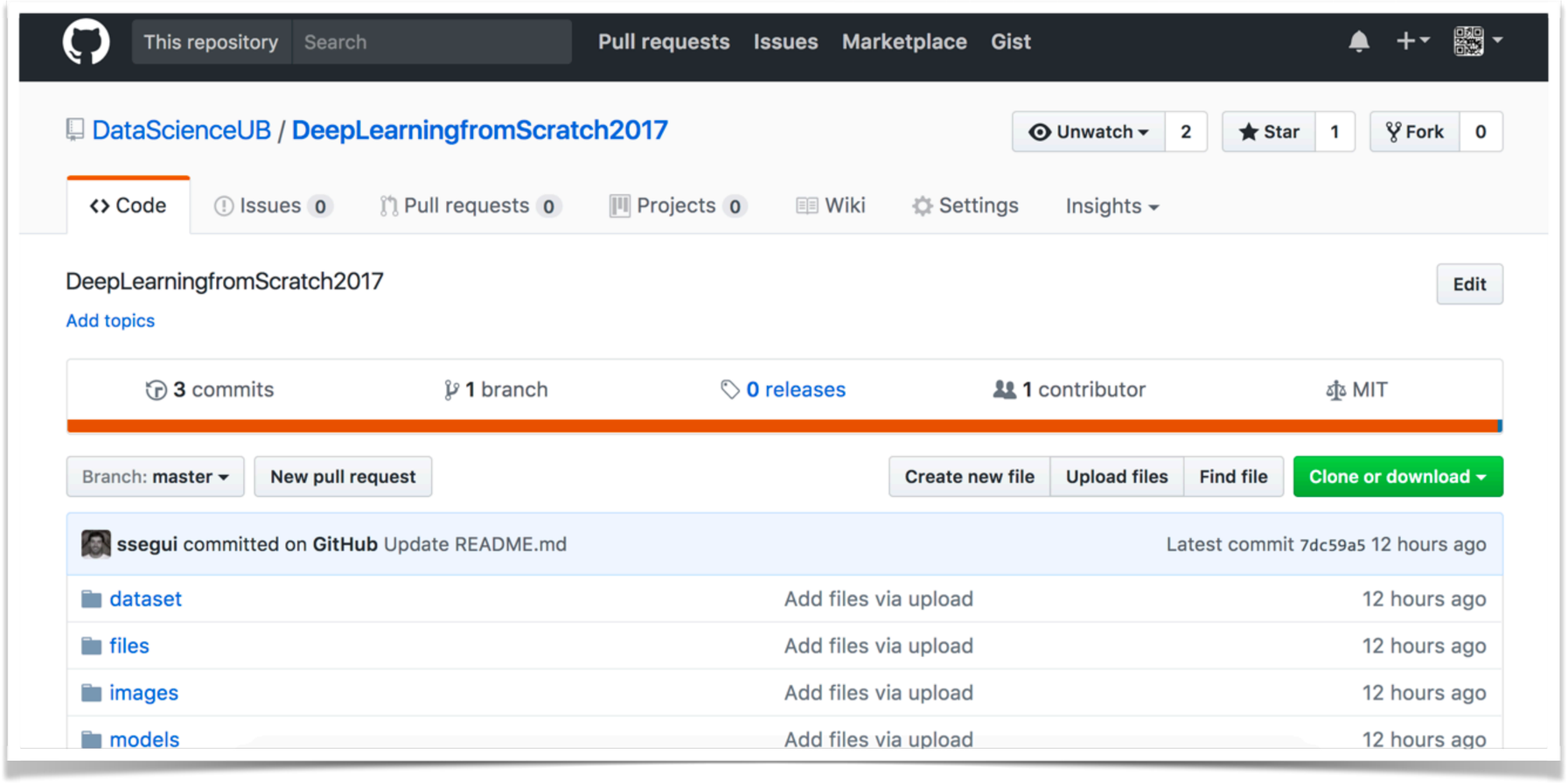
DAY 1

- 09:30-10:00, Registration
- 10:00-10:10, Welcome
- 10:10-11:15, Introduction to Deep Learning and its applications.
- 11:15-11:30, Coffee Break
- 11:30-13:30, Basic Concepts: Score & Loss functions, Optimization (SGD), Linear Regression.
- 13:30-15:00, Lunch
- 15:00-16:15, Backpropagation, Training a Neural Netwotk from Scratch.
- 16:15-16:30, Coffee Break
- 16:30-18:00, Tensorflow programming model.

DAY 2

- 09:00-11:00, Convolutions. CNN models.
- 11:00-11:15, Coffee Break
- 11:15-13:30, Recurrent Neural Netwoks.
- 13:30-15:00, Lunch
- 15:00-16:15, Unsupervised Learning.
- 16:15-16:30, Coffee Break
- 16:30-17:15, Advanced Applications: Embeddingss, Sentiment Analysis, Time Series.
- 17:15-17:30, Summary of the course and closing.

Course Repository



What

1. Describe how a (deep) neural network works and combine different types of layers and activation functions. Deep Learning is not magic.
2. Describe how these models can be applied in computer vision, text analytics, time series analysis, etc. Deep Learning is not the final machine learning method.
3. Develop your own models in **Tensorflow** and derivates. You can train (mid-size) deep models in your laptop.

How Earth Globes were made back in the 50's

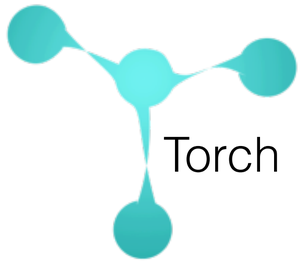
Deep Learning BackEnds

Python, C++

MultiGPU

Distributed

Google



Lua

MultiGPU

theano

Python

Large amount of sample code

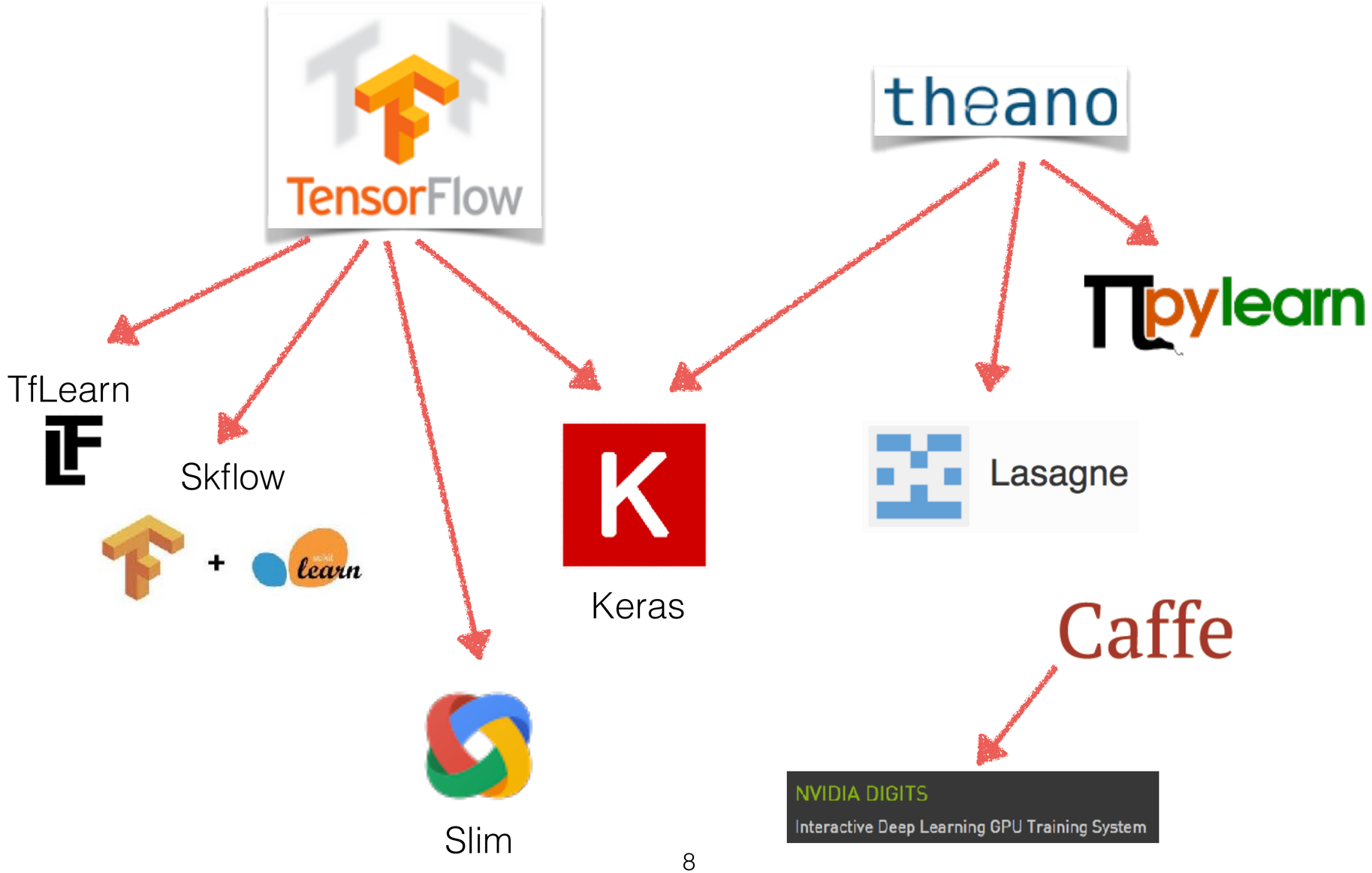
Université de Montréal

Python, C++

MultiGPU

Berkeley
UNIVERSITY OF CALIFORNIA

Tools


TensorFlow
theano
pylearn
TfLearn
Skflow
learn
Keras
Lasagne
Caffe
Slim
NVIDIA DIGITS
Interactive Deep Learning GPU Training System

Other BackEnds

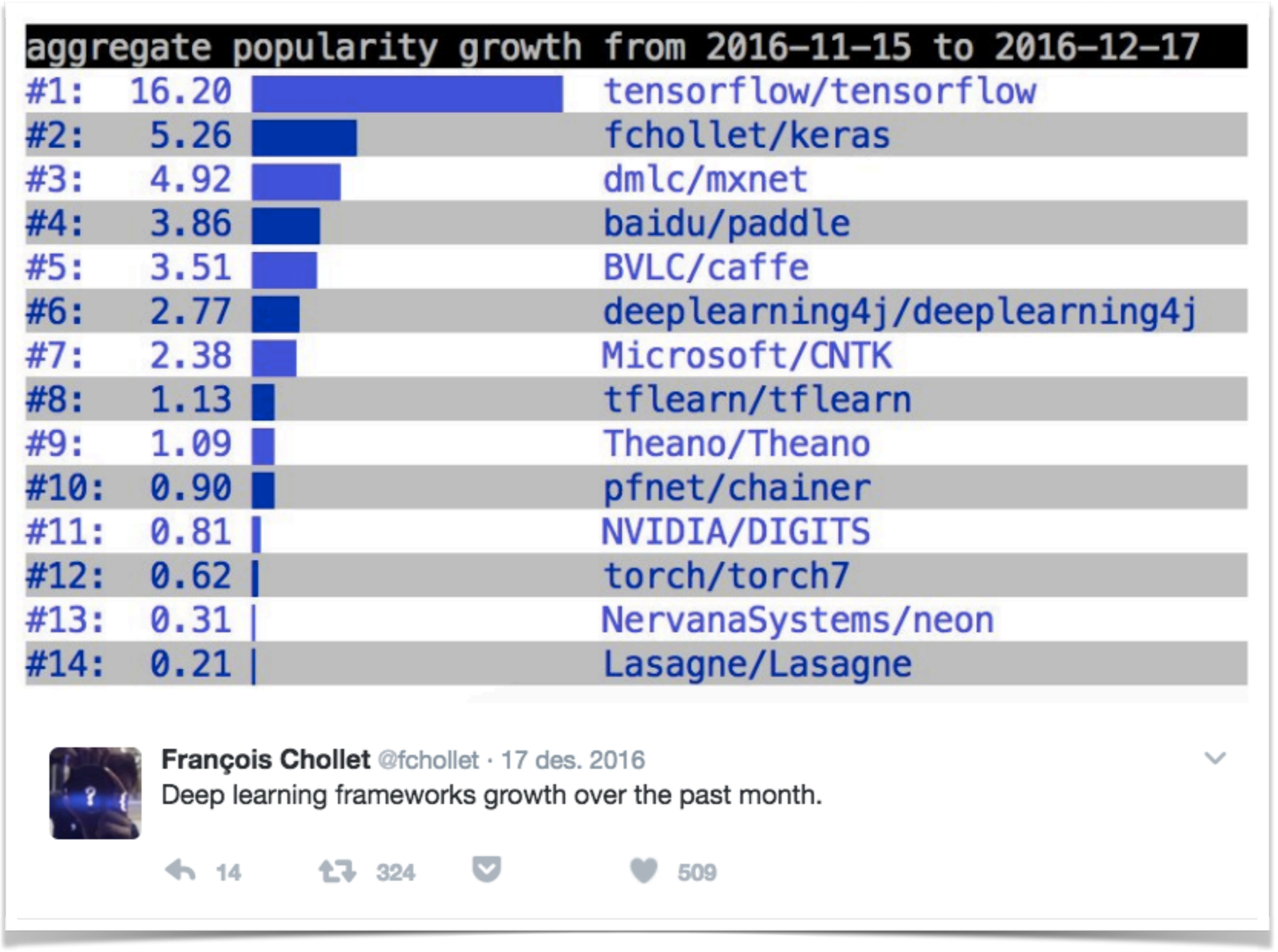
CNTK

dmlc mxnet

Neon

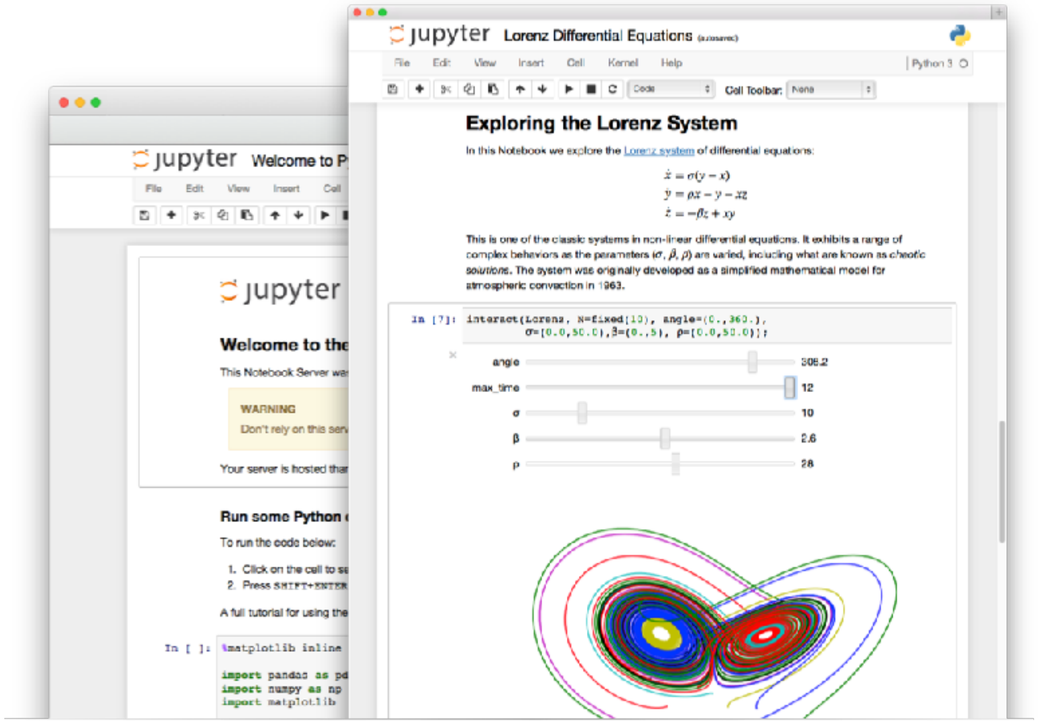
DL4J Deep Learning for Java

ConvNetJS
Deep Learning in your browser

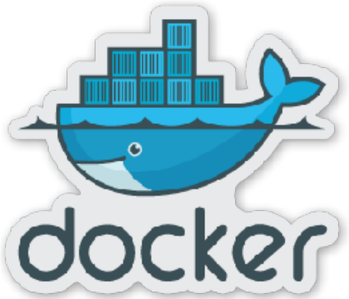

aggregate popularity growth from 2016-11-15 to 2016-12-17
#1: 16.20 tensorflow/tensorflow
#2: 5.26 fchollet/keras
#3: 4.92 dmlc/mxnet
#4: 3.86 baidu/paddle
#5: 3.51 BVLC/caffe
#6: 2.77 deeplearning4j/deeplearning4j
#7: 2.38 Microsoft/CNTK
#8: 1.13 tflearn/tflearn
#9: 1.09 Theano/Theano
#10: 0.90 pfnet/chainer
#11: 0.81 NVIDIA/DIGITS
#12: 0.62 torch/torch7
#13: 0.31 NervanaSystems/neon
#14: 0.21 Lasagne/Lasagne
François Chollet @fchollet · 17 des. 2016
Deep learning frameworks growth over the past month.
14
324
509

Approach

We will illustrate all contents with Jupyter notebooks, a web application that allows you to create and share documents that contain live code, equations, visualizations and explanatory text.



Approach



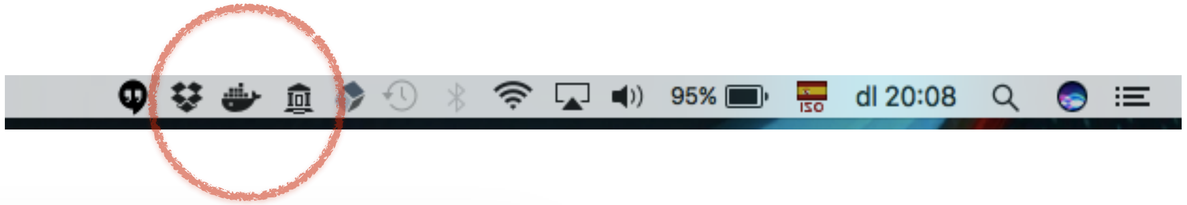
We will use a **Docker Container**.

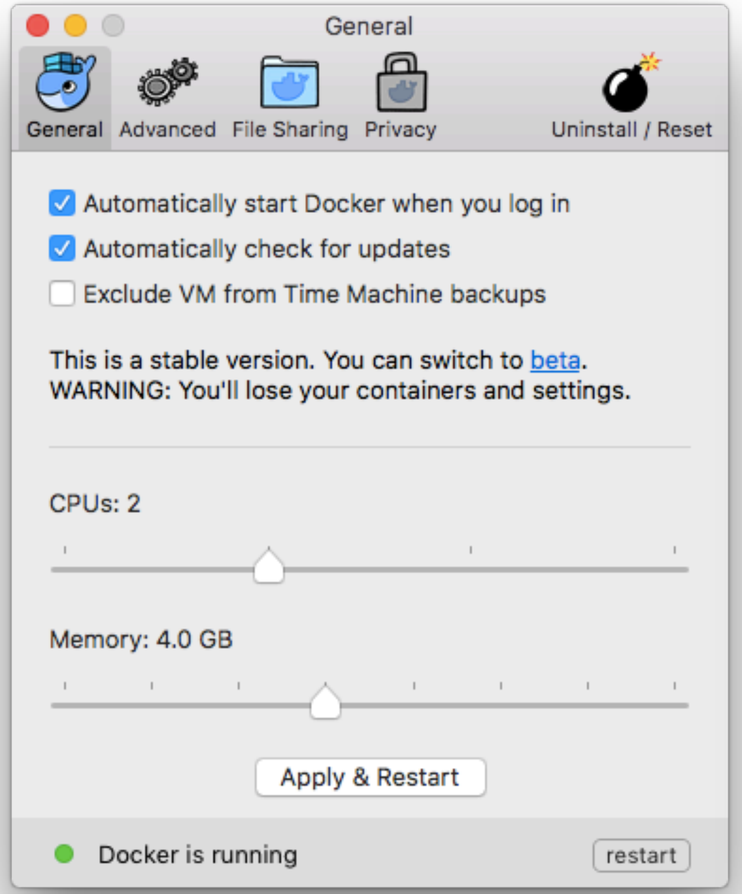
Docker provides the ability to build a runtime environment that not only remains isolated from other running containers, but also can be deployed to multiple locations in a repeatable way. Docker also uses a text document – a Dockerfile – that contains all the commands to assemble an image, which will meet our need to document the build environment. Finally, Docker's runtime options enable us to attach GPU devices when deploying on remote servers.

Course Software Installation

The best way to run the course software is to use a **Docker container**. There's full documentation on installing Docker at `docker.com`, but in a few words, the steps are:

- Go to `docs.docker.com` in your browser.
- Step one of the instructions sends you to download Docker.
- Run that downloaded file to install Docker.
- At the end of the install process a whale in the top status bar indicates that Docker is running, and accessible from a terminal.
- Click the whale to get `Preferences`, and other options.
- Open a command-line terminal, and run some Docker commands to verify that Docker is working as expected. Some good commands to try are `docker version` to check that you have the latest release installed, and `docker ps` and `docker run hello-world` to verify that Docker is running.
- By default, Docker is set to use 2 processors. You can increase processing power for the app by setting this to a higher number in `Preferences`, or lower it to have Docker for Mac use fewer computing resources.
- Memory - By default, Docker is set to use 2 GB runtime memory, allocated from the total available memory on your computer. You can increase the RAM on the app to get faster performance by setting this number higher (for example to 3) or lower (to 1) if you want Docker to use less memory.


95%
dl 20:08


General
General
Advanced
File Sharing
Privacy
Uninstall / Reset
Automatically start Docker when you log in
Automatically check for updates
Exclude VM from Time Machine backups
This is a stable version. You can switch to beta.
WARNING: You'll lose your containers and settings.
CPUs: 2
Memory: 4.0 GB
Apply & Restart
Docker is running
restart

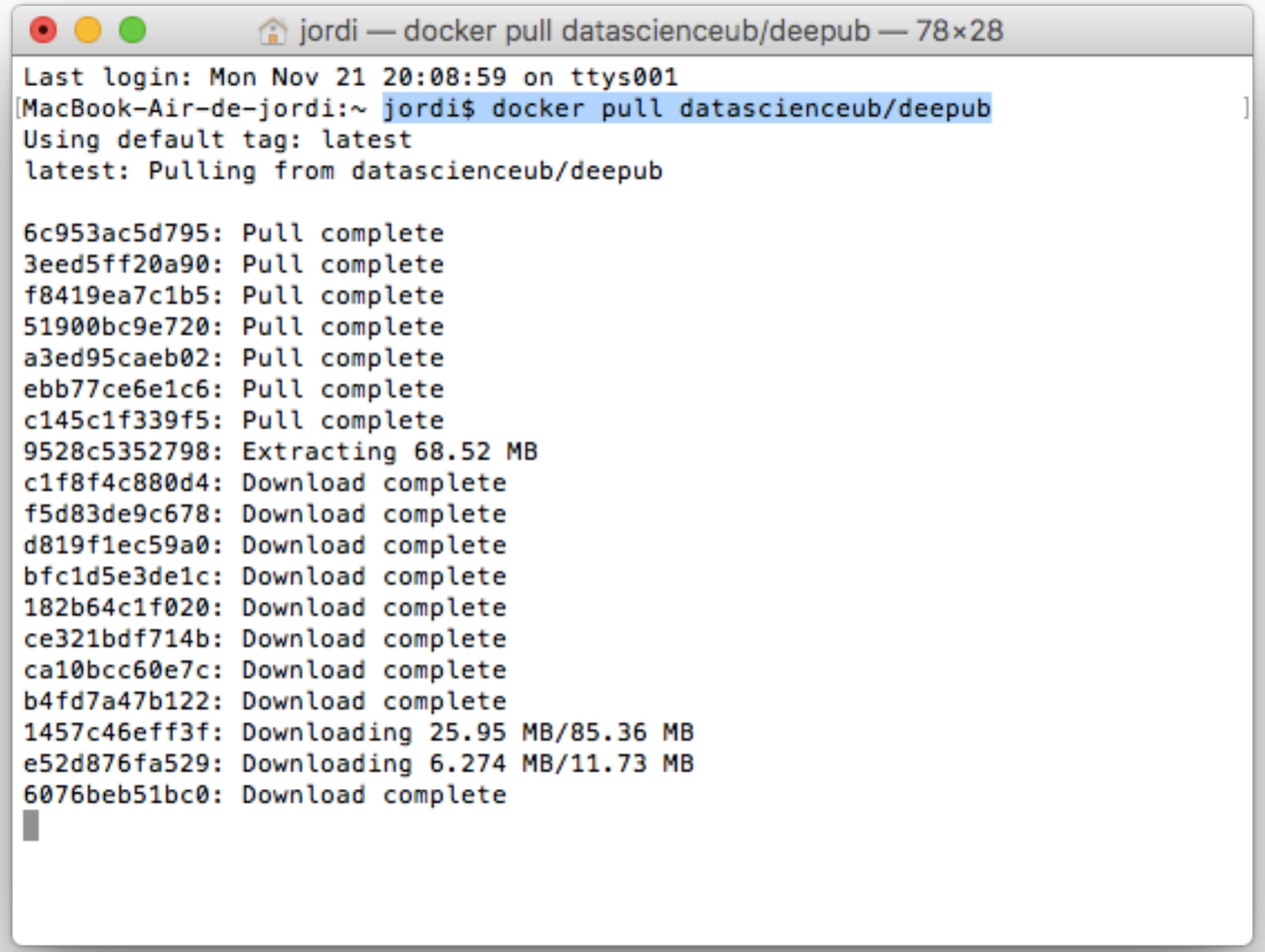

File Sharing
General
Advanced
File Sharing
Privacy
Uninstall / Reset
These OS X directories (and their subdirectories) can be bind mounted into Docker containers. These exported directories cannot overlap.
You can check the documentation for more details.
/Users
/Volumes
/tmp
/private
Apply & Restart
Docker starting...
restart

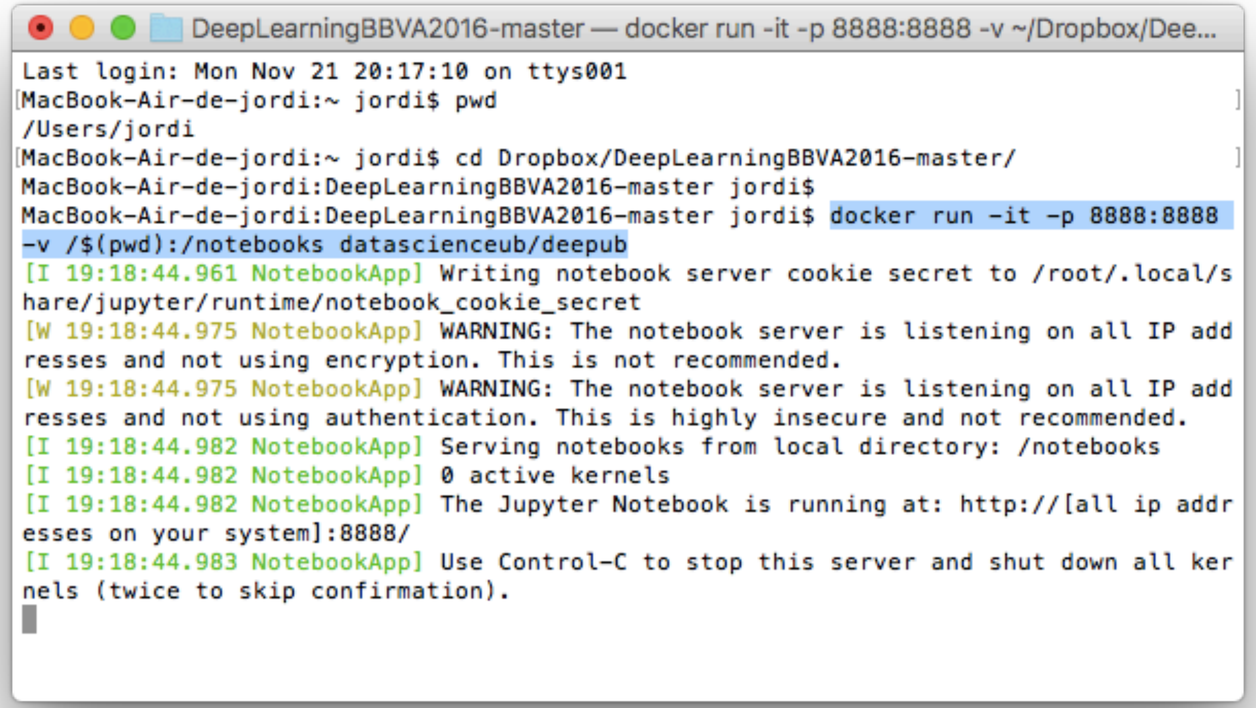
Once Docker is installed, you can dowload the image of this course:

- In a terminal, go to your course folder and run (This operation requires a good internet connection; it will take some minutes): `docker pull datascienceub/deepub`
- Run the `deepub` image on your system: `docker run -it -p 8888:8888 -p 6006:6006 -v /$(pwd):/notebooks datascienceub/deepub`
- Once these steps have been done, you can check the installation by starting your web browser and introducing this URL: `http://localhost:8888` .
- Open a new Jupyter notebook and execute this instruction in a code cell: `!git clone https://github.com/DataScienceUB/Vodafone2017`

Note:

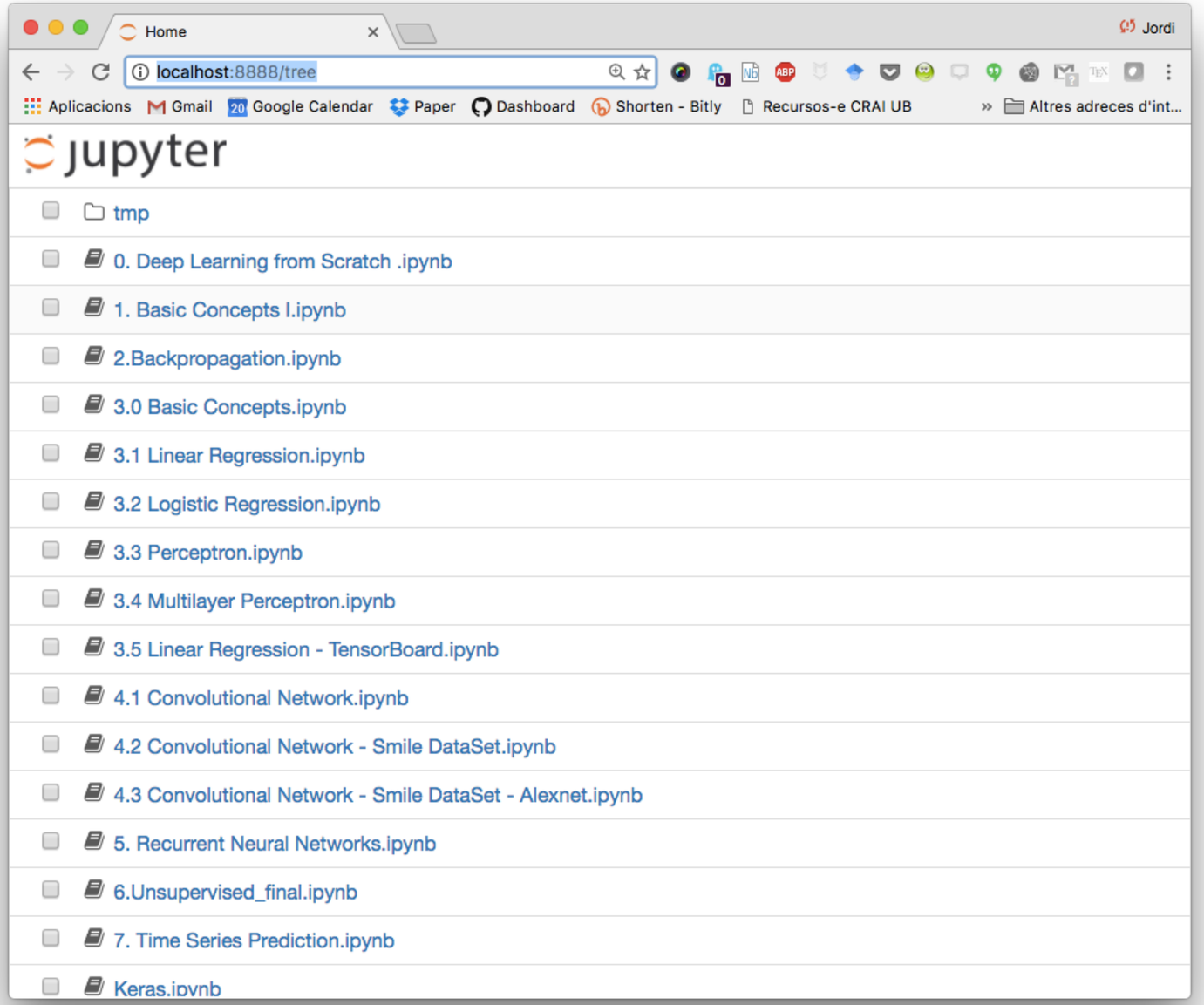
Docker for Windows requires 64bit Windows 10 Pro, Enterprise and Education. The Hyper-V package must be enabled for Docker for Windows to work. The Docker for Windows installer will enable it for you, if needed. (This requires a reboot). If your system does not satisfy these requirements, you can install Docker Toolbox, which uses Oracle Virtual Box instead of Hyper-V.


jordi — docker pull datascienceub/deepub — 78×28
Last login: Mon Nov 21 20:08:59 on ttys001
MacBook-Air-de-jordi:~ jordi$ docker pull datascienceub/deepub
Using default tag: latest
latest: Pulling from datascienceub/deepub
6c953ac5d795: Pull complete
3eed5ff20a90: Pull complete
f8419ea7c1b5: Pull complete
51900bc9e720: Pull complete
a3ed95caeb02: Pull complete
ebb77ce6e1c6: Pull complete
c145c1f339f5: Pull complete
9528c5352798: Extracting 68.52 MB
c1f8f4c880d4: Download complete
f5d83de9c678: Download complete
d819f1ec59a0: Download complete
bfc1d5e3de1c: Download complete
182b64c1f020: Download complete
ce321bdf714b: Download complete
ca10bcc60e7c: Download complete
b4fd7a47b122: Download complete
1457c46eff3f: Downloading 25.95 MB/85.36 MB
e52d876fa529: Downloading 6.274 MB/11.73 MB
6076beb51bc0: Download complete



docker run -it -p 8888:8888 -p 6006:6006 -v /$(pwd):/notebooks datascienceub/deepub

docker run -it -p 8888:8888 -p 6006:6006 -v C:\jordi\DeepLearning:/notebooks datascienceub/deepub


Home
Jordi
localhost:8888/tree
Aplicacions
Gmail
Google Calendar
Paper
Dashboard
Shorten - Bitly
Recursos-e CRAI UB
Altres adreces d'int...
jupyter
tmp
0. Deep Learning from Scratch .ipynb
1. Basic Concepts I.ipynb
2.Backpropagation.ipynb
3.0 Basic Concepts.ipynb
3.1 Linear Regression.ipynb
3.2 Logistic Regression.ipynb
3.3 Perceptron.ipynb
3.4 Multilayer Perceptron.ipynb
3.5 Linear Regression - TensorBoard.ipynb
4.1 Convolutional Network.ipynb
4.2 Convolutional Network - Smile DataSet.ipynb
4.3 Convolutional Network - Smile DataSet - Alexnet.ipynb
5. Recurrent Neural Networks.ipynb
6.Unsupervised_final.ipynb
7. Time Series Prediction.ipynb
Keras.ipynb

Why Deep Learning?


It’s funny!
It’s not rocket science!
It’s powerful!

Historical metaphors of the brain:
Hydraulic (blood cooler, spirits),
Mechanical (clock, steam machine), …

In 1943, neurophysiologist **Warren McCulloch** and mathematician **Walter Pitts** wrote a paper on how neurons might work. In order to describe how neurons in the brain might work, they modeled a simple neural network using **electrical circuits**.

In 1949, Donald **Hebb** wrote *The Organization of Behavior*, a work which pointed out the fact that **neural pathways are strengthened each time they are used**, a concept fundamentally essential to the ways in which humans **learn**.

$$f(x) = \begin{cases} 1 & \text{if } w \cdot x + b > 0 \\ 0 & \text{otherwise} \end{cases}$$

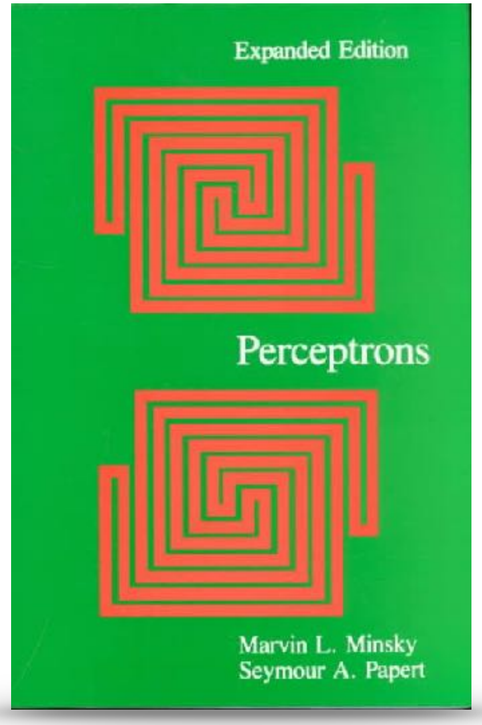
In 1957 **Frank Rosenblatt** attempted to build a kind of mechanical brain called the **Perceptron**, which was billed as "*a machine which senses, recognizes, remembers, and responds like the human mind*".

In 1962, **Widrow & Hoff** developed a **learning rule** that examines the value before the weight adjusts it (i.e. 0 or 1) according to the rule:

*Weight Change = (Pre-Weight line value) * (Error / (Number of Inputs)).*

It is based on the idea that **while one active perceptron may have a big error, one can adjust the weight values to distribute it across the network**, or at least to adjacent perceptrons.

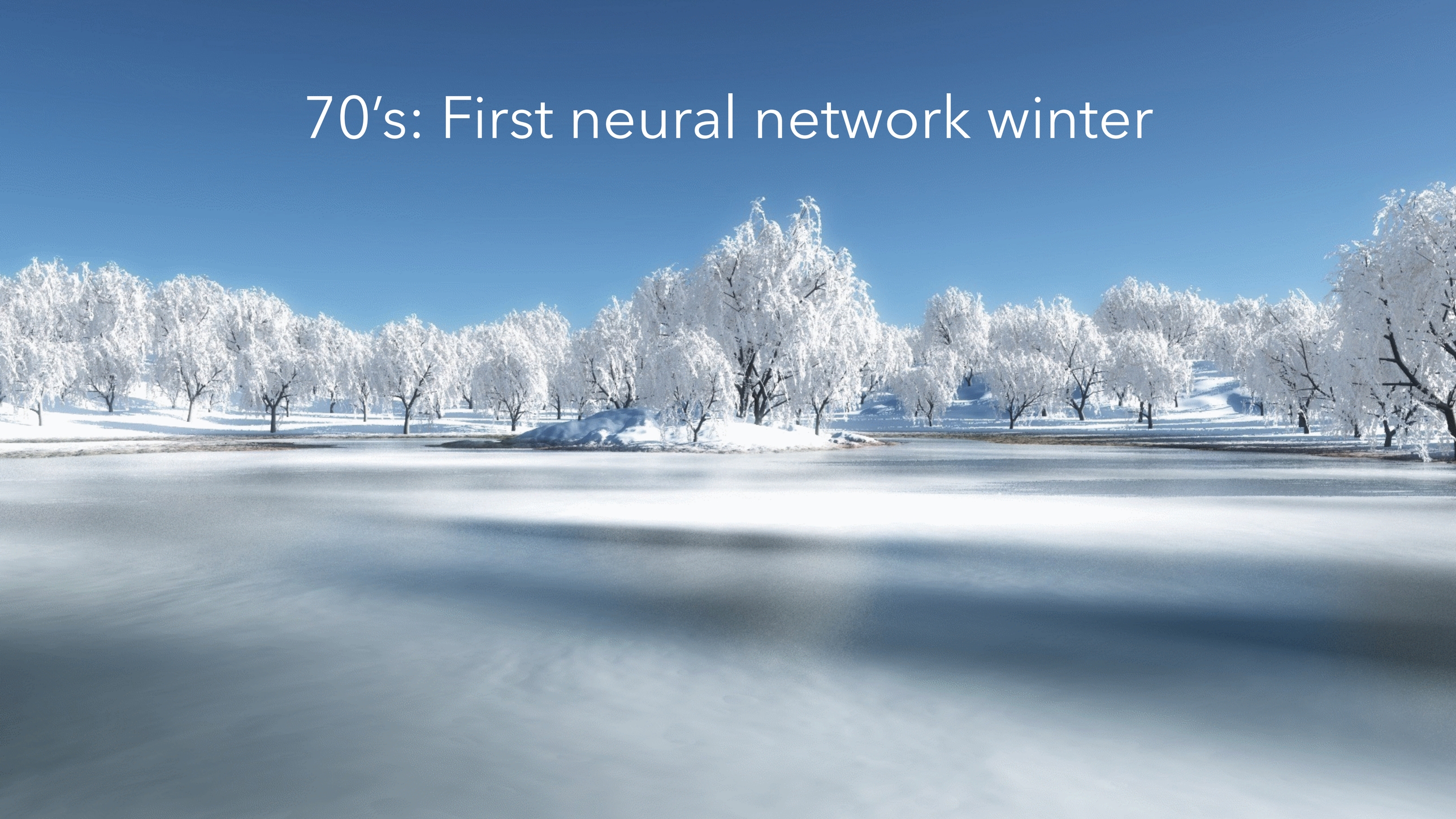
A critical book written in 1969 by **Marvin Minsky** and his collaborator **Seymour Papert** showed that Rosenblatt's original system was **painfully limited**, literally blind to some simple logical functions like "*exclusive-or*".



It is claimed that pessimistic predictions made by the authors were responsible for an erroneous change in the direction of research in AI, concentrating efforts on so-called "symbolic" systems, and contributing to the so-called AI winter. This decision, supposedly, proved to be unfortunate in the 1980s, when new discoveries showed that the prognostics in the book were wrong.

Source: Wikipedia

70’s: First neural network winter



In 1982, interest in the field was renewed. **John Hopfield** of Caltech presented a paper to the National Academy of Sciences. His approach was to create more useful machines by using **bidirectional lines**. Previously, the connections between neurons was only one way.

In 1986, the problem was how to **extend the Widrow-Hoff rule to multiple layers**. Three independent groups of researchers, which included **David E. Rumelhart, Geoffrey E. Hinton** and **Ronald J. Williams**, came up with similar ideas which are now called **back-propagation** networks because it distributes pattern recognition errors throughout the network.

From 1986 to mid 90's new developments arised: convolutional neural networks (**Y.LeCun**), unsupervised learning (**Y.Bengio**), RBM (**G.Hinton**), recurrent networks (**J.Schmidhuber**), etc.

But, by this point **new machine learning methods** had begun to also emerge, and people were again beginning to be skeptical of neural nets since they seemed so intuition-based and since computers were still barely able to meet their computational needs.

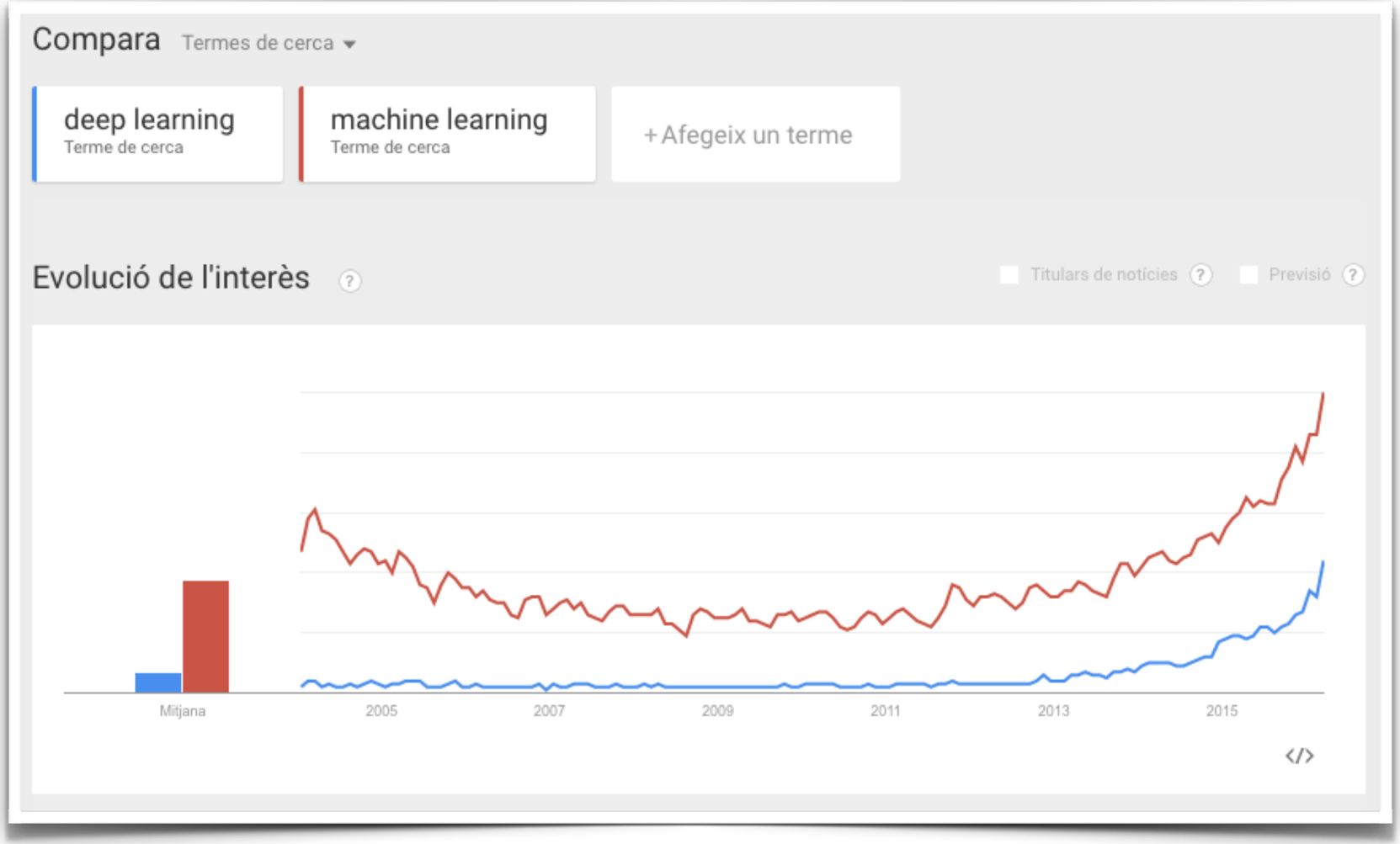
90’s-00’s: Second neural network winter

With the ascent of Support Vector Machines and the **failure of backpropagation**, the early 2000s were a dark time for neural net research.

Then, what every researcher must dream of actually happened: G.Hinton, S.Osindero, and Y.W.Teh published a paper in 2006 that was seen as a breakthrough, a breakthrough significant enough to rekindle interest in neural nets: *A fast learning algorithm for **deep** belief nets.*

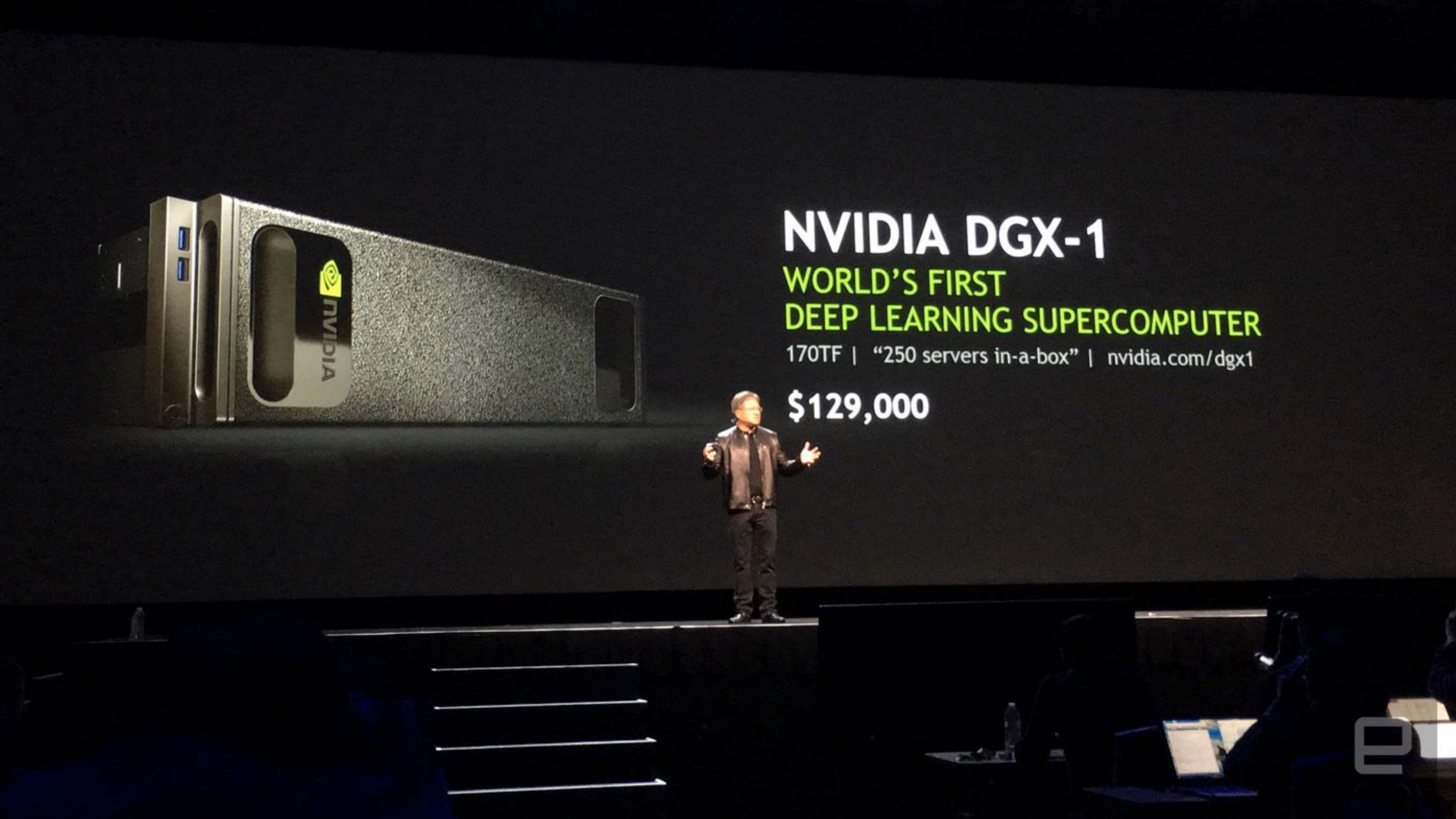
After that, following Moore's law, computers got dozens of times faster (GPUs) since the slow days of the 90s, making learning with large datasets and many layers much more tractable.

Neural Networks Reborn



Google Trends


Jetson TX1 Developer Kit
$599 retail
$299 edu
Pre-order Nov 12
Shipping Nov 16 (US)
Intl to follow
amazon
newegg.com
MICRO CENTER
computers & electronics
NVIDIA

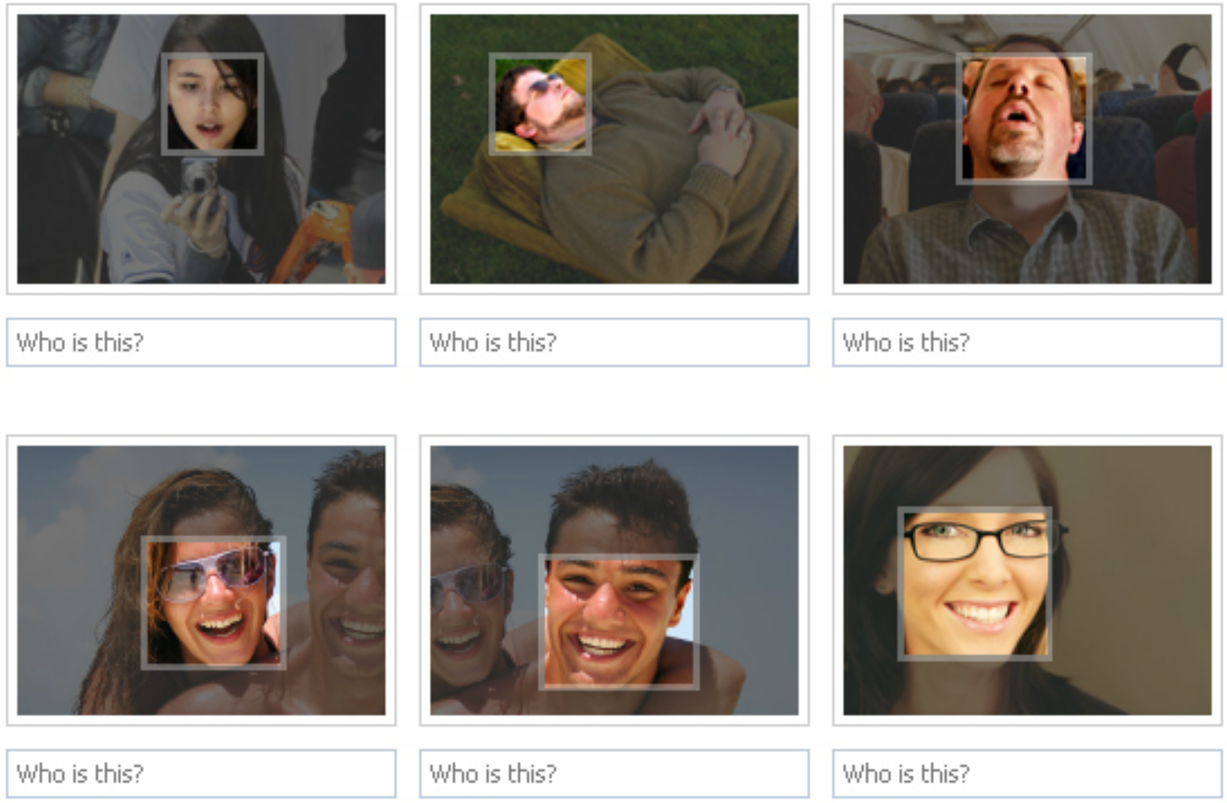

NVIDIA DGX-1
WORLD'S FIRST
DEEP LEARNING SUPERCOMPUTER
170TF | "250 servers in-a-box" | nvidia.com/dgx1
$129,000
NVIDIA

Definitions

- **Neural Networks (NN)** is a beautiful biologically-inspired programming paradigm which enables a computer to learn from observational data.
- **Deep Learning (DL)** is a powerful set of techniques for learning in neural networks.
- NN and DL currently provide the best solutions to many problems in image recognition, speech recognition, and natural language processing.

“Classical” applications:
object classification, detection and segmentation.

Face recognition.



DeepFace (Facebook): Accuracy of 97.35%

New applications: navigation and mapping.

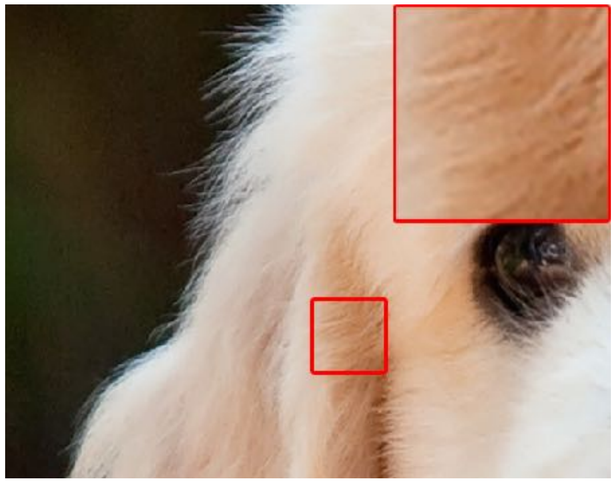
New applications: Image Upscaling (Flipboard)

http://engineering.flipboard.com/2015/05/scaling-convnets/

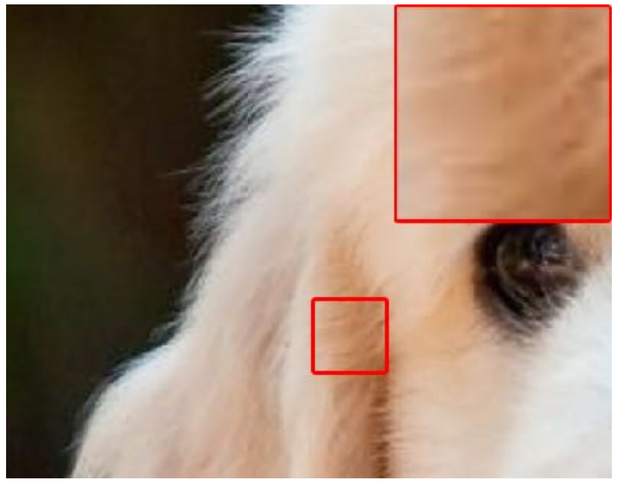
New applications: Image Upscaling (Flipboard)

http://engineering.flipboard.com/2015/05/scaling-convnets/

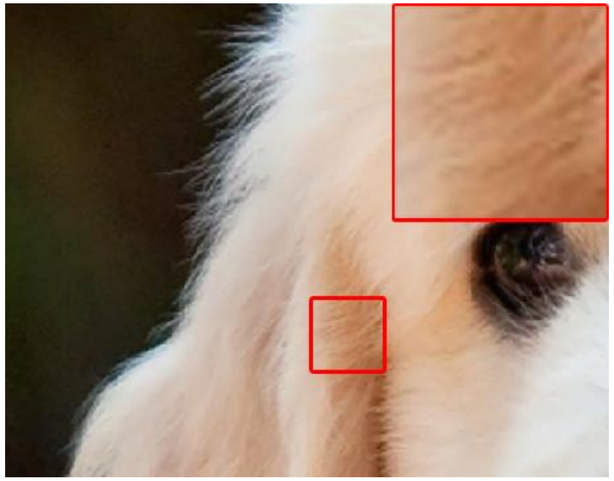
New applications: Image Upscaling (Flipboard)



Original



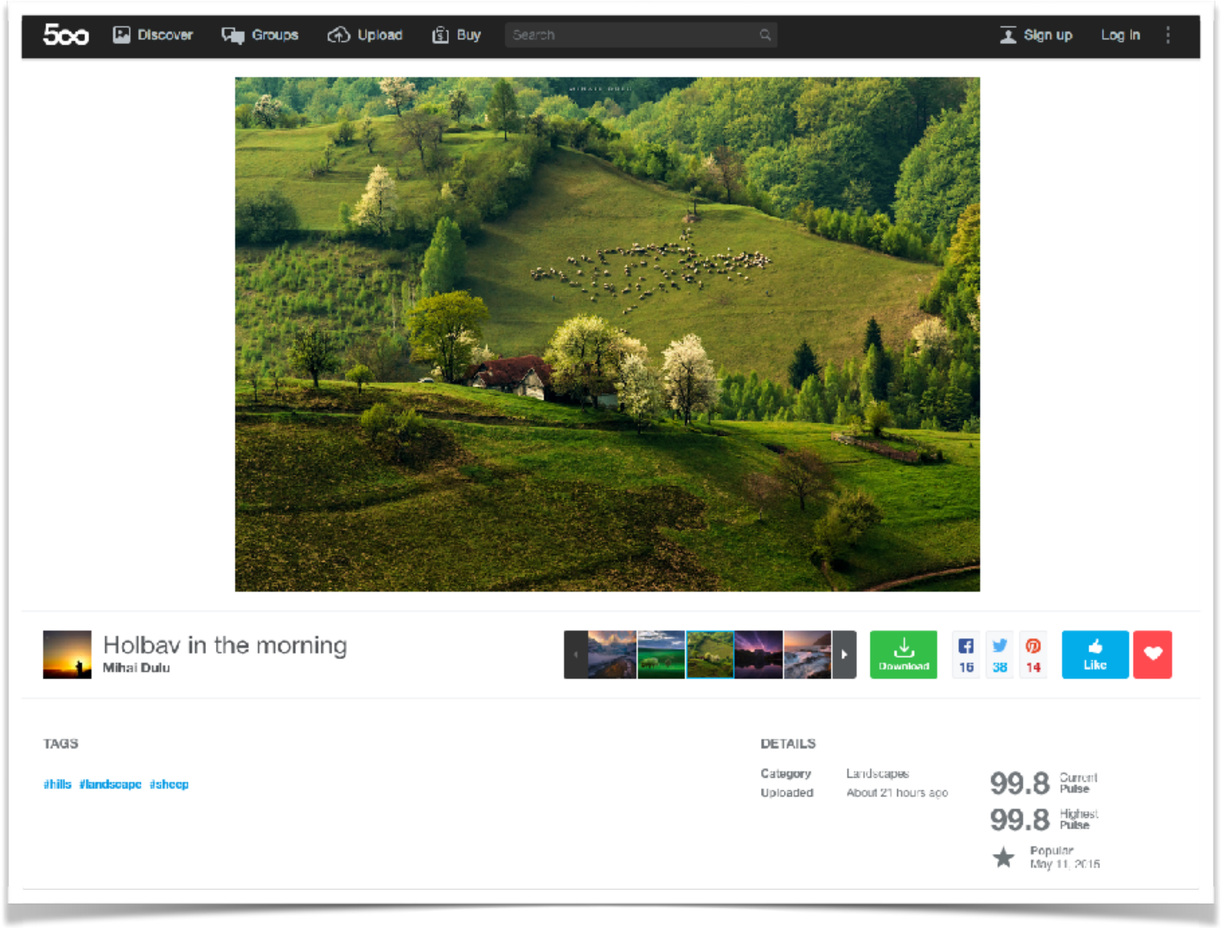
Bicubic



Model

http://engineering.flipboard.com/2015/05/scaling-convnets/

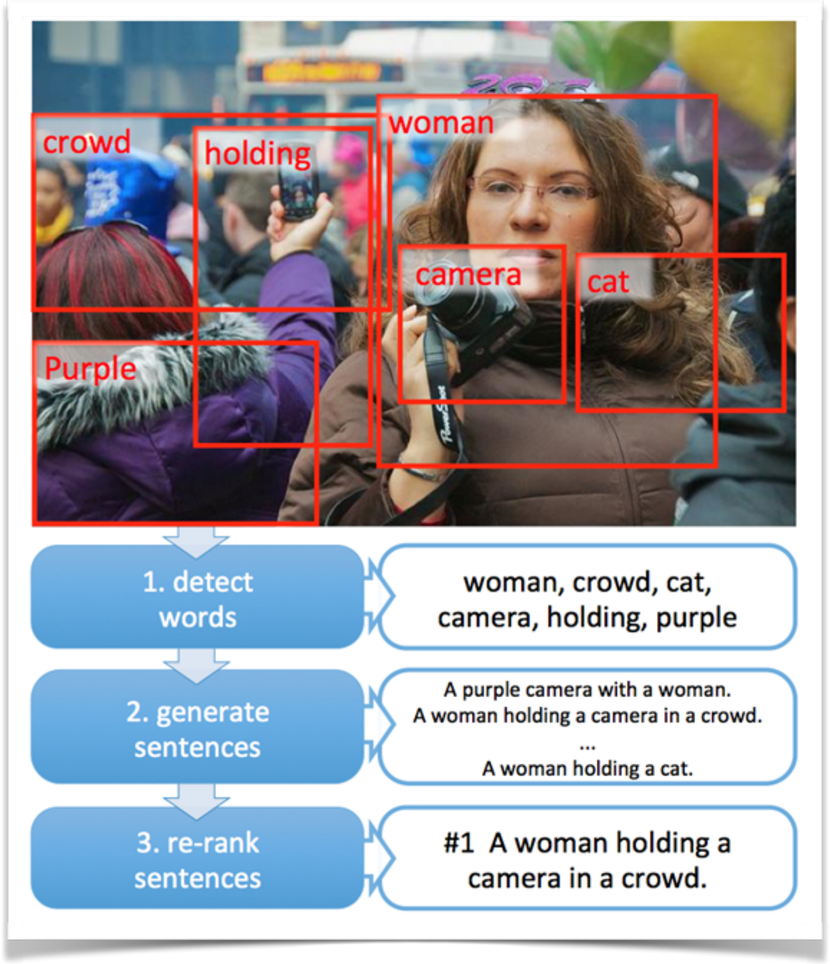
New applications: Non visual data prediction



What is Pulse?

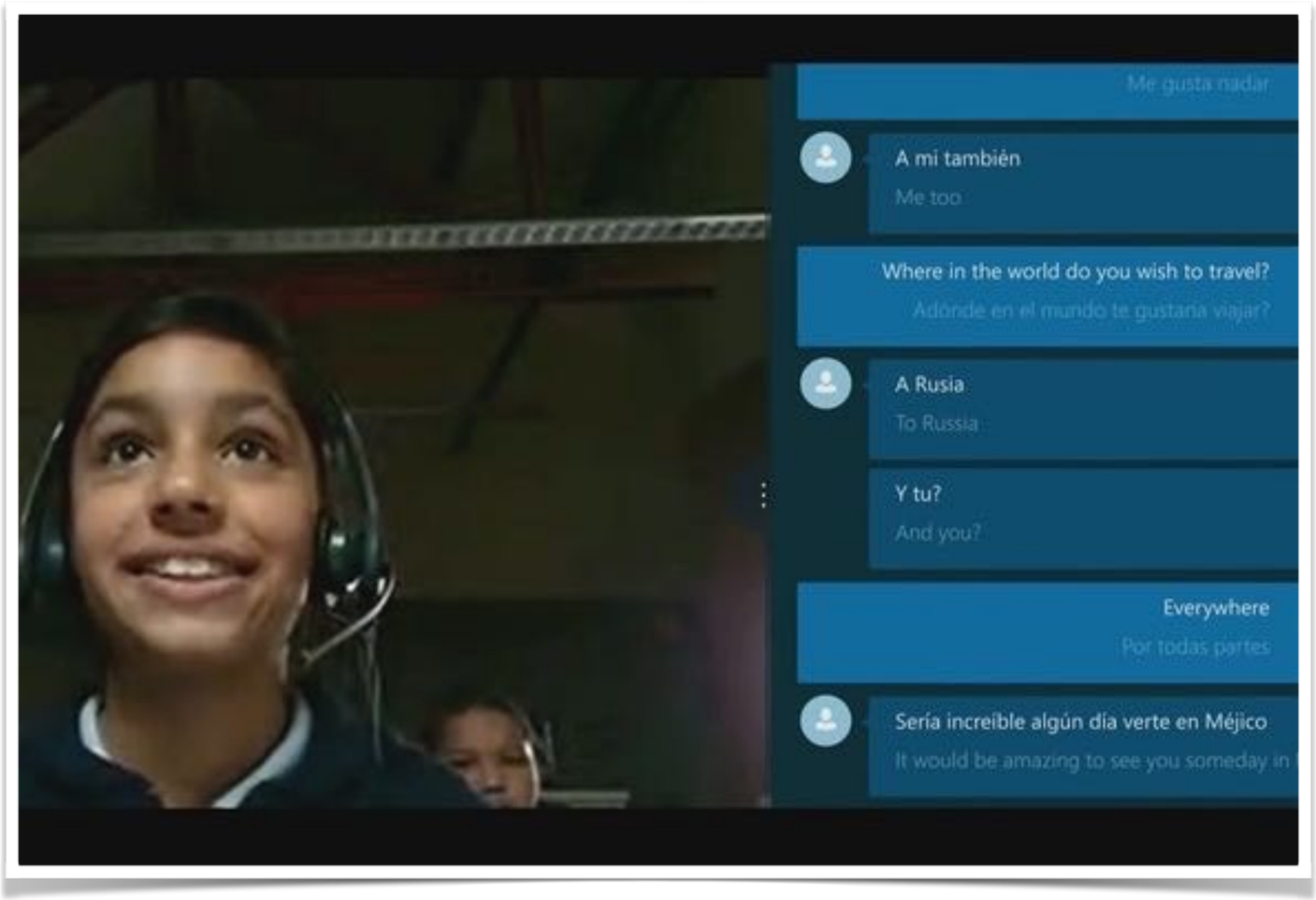
Pulse is a score out of 100 points that measures how **popular** a photo is. Pulse is calculated by an algorithm, which is unique to 500px and is based on votes (Likes & Favorites) on your photo from the community. The Pulse algorithm was designed to promote daily exposure of new photographs and photographers. It is not necessarily a measure of photograph's quality.

New applications: Automatic Image Captioning

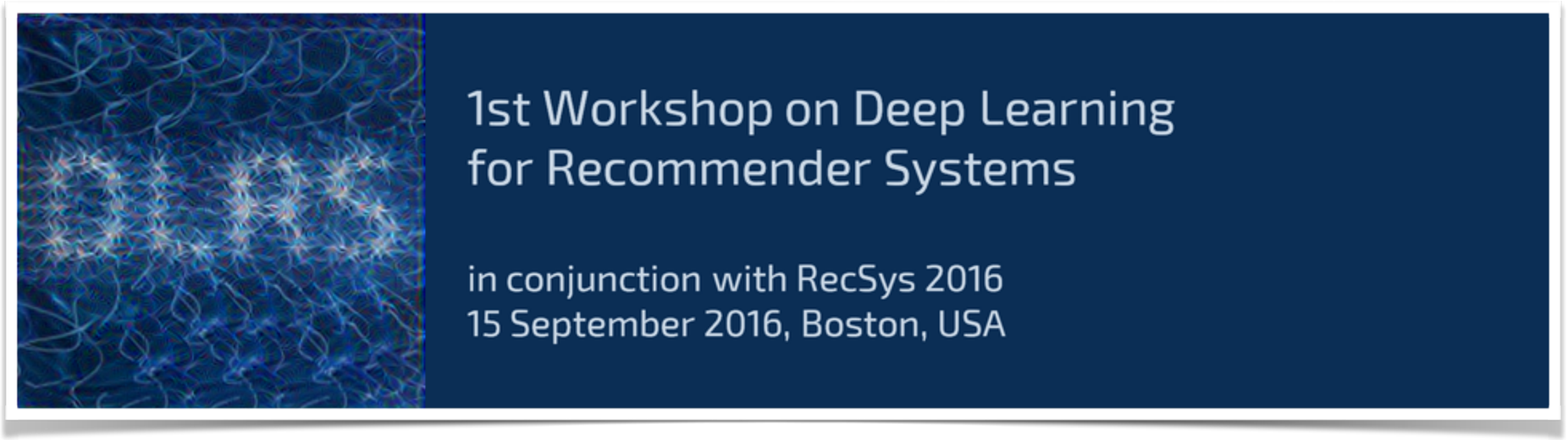


http://blogs.technet.com/b/machinelearning/archive/2014/11/18/rapid-progress-in-automatic-image-captioning.aspx

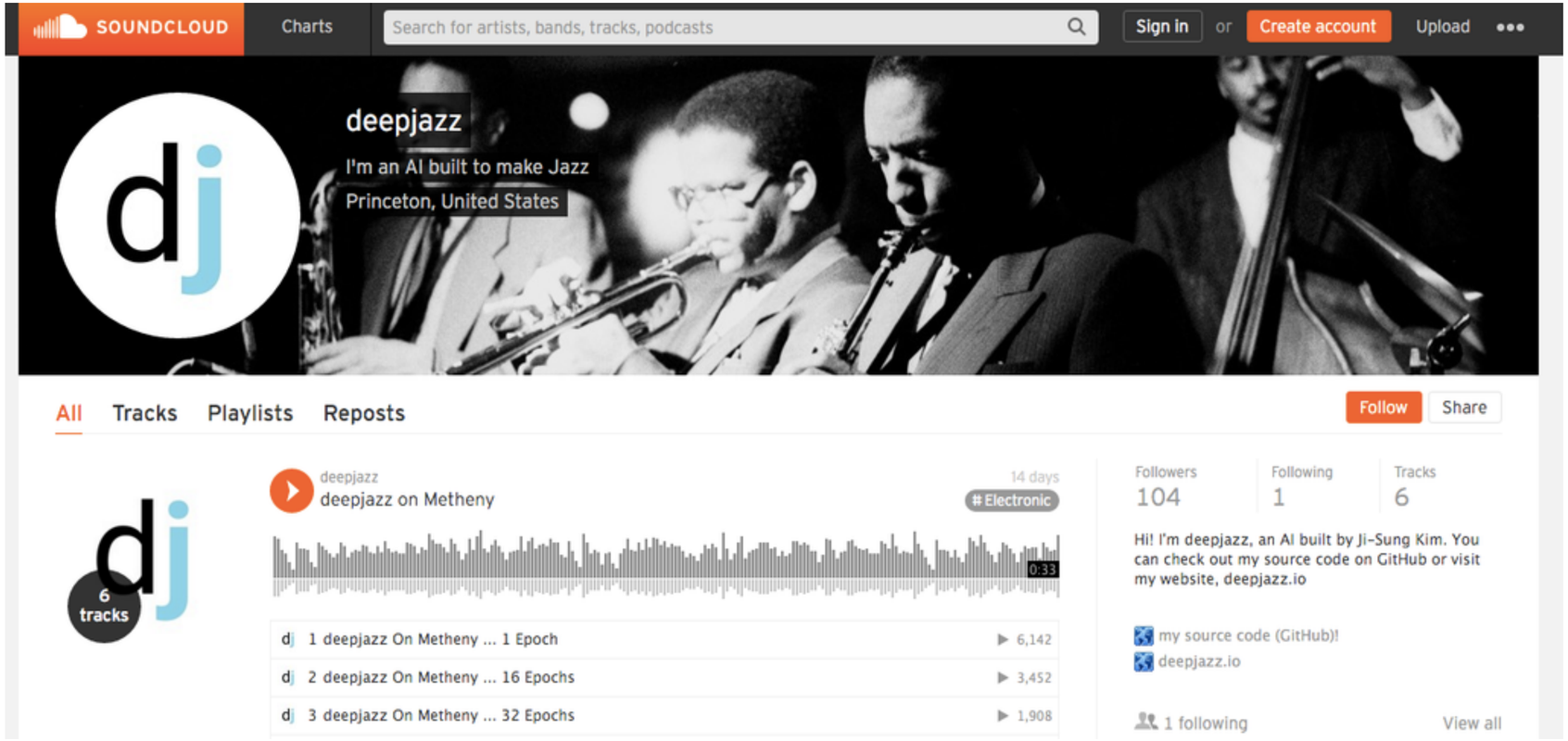
Speech translation



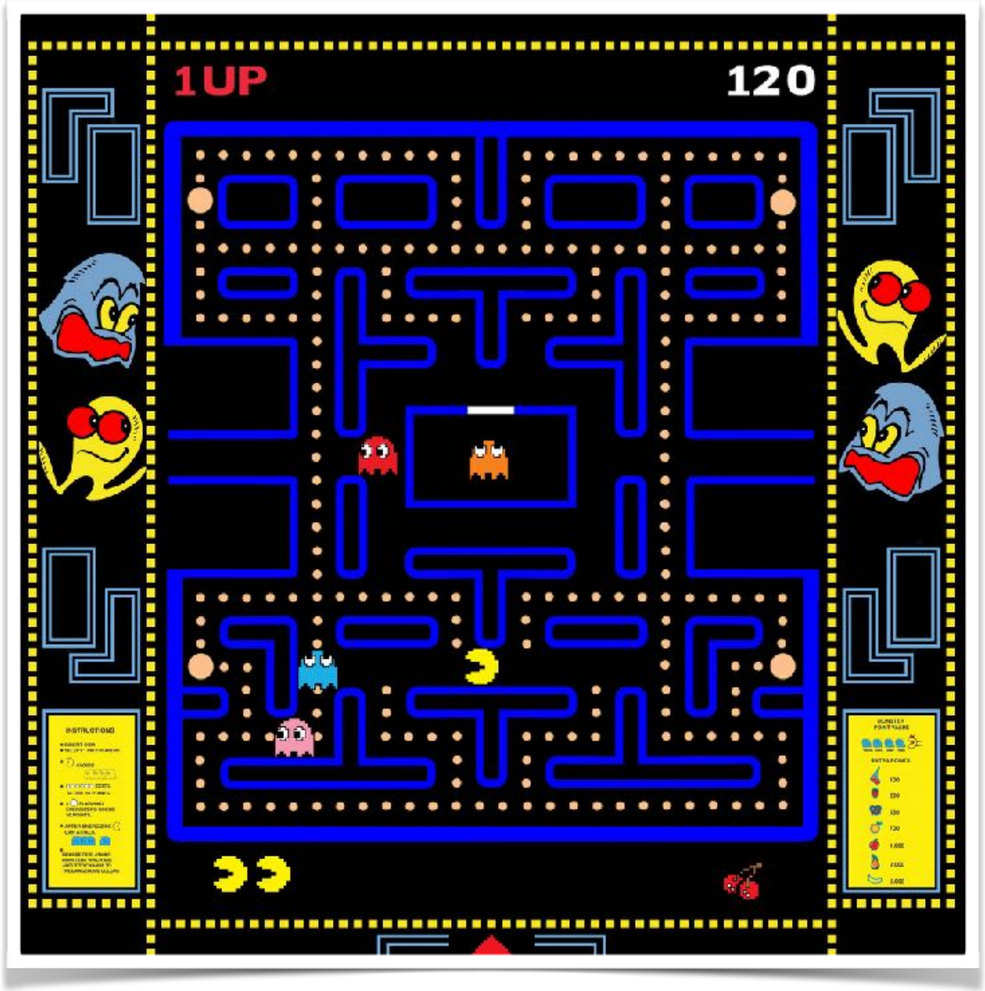
Recommenders



Music Generation



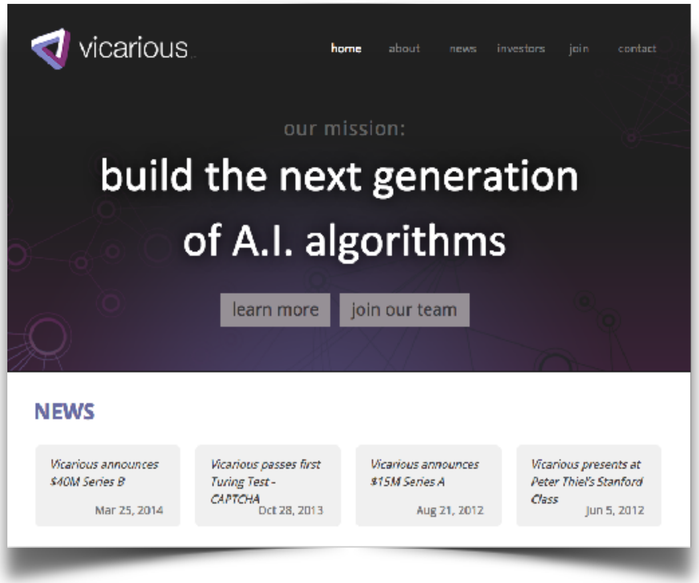
Reinforcement learning.



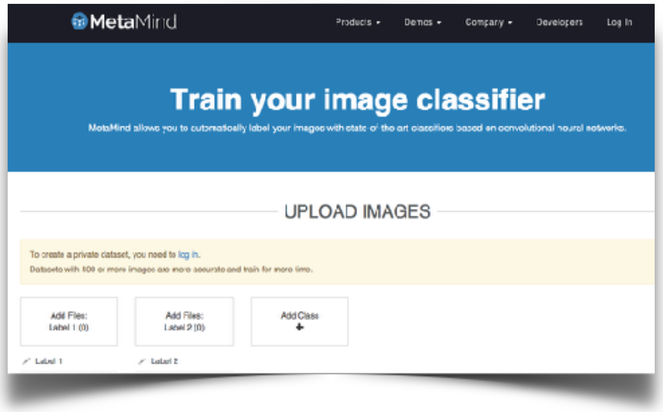
Go

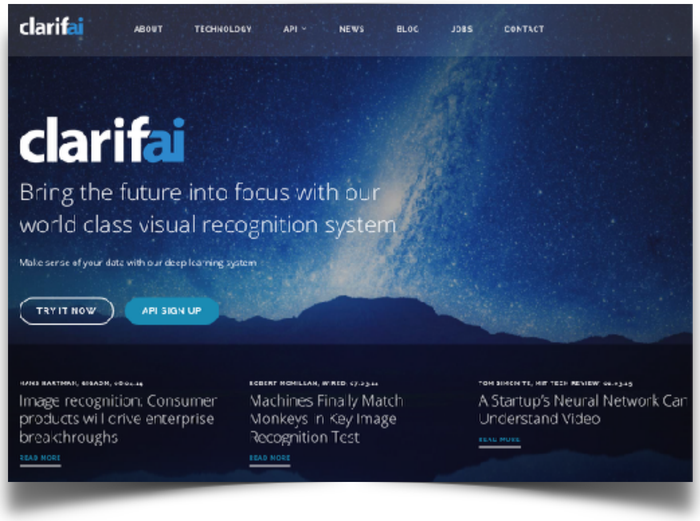

nature
THE INTERNATIONAL WEEKLY JOURNAL OF SCIENCE
At last — a computer program that can beat a champion Go player PAGE 484
ALL SYSTEMS GO
CONSERVATION
SONGBIRDS À LA CARTE
Illegal harvest of millions of Mediterranean birds
PAGE 452
RESEARCH ETHICS
SAFEGUARD TRANSPARENCY
Don't let openness backfire on individuals
PAGE 459
POPULAR SCIENCE
WHEN GENES GOT 'SELFISH'
Dawkins's calling card forty years on
PAGE 462
NATURE.COM/NATURE
28 January 2016 £10
Vol. 529, No. 7587

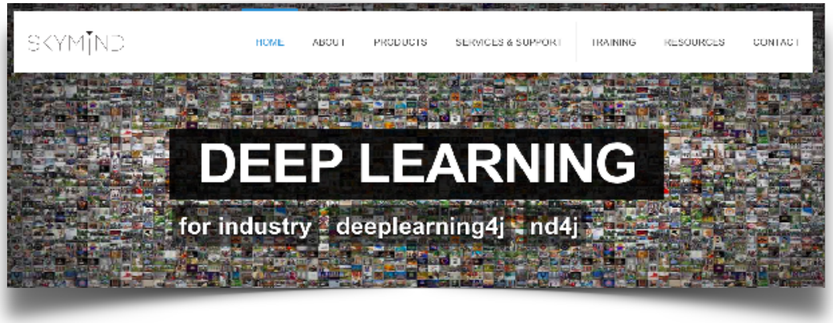
Start Ups

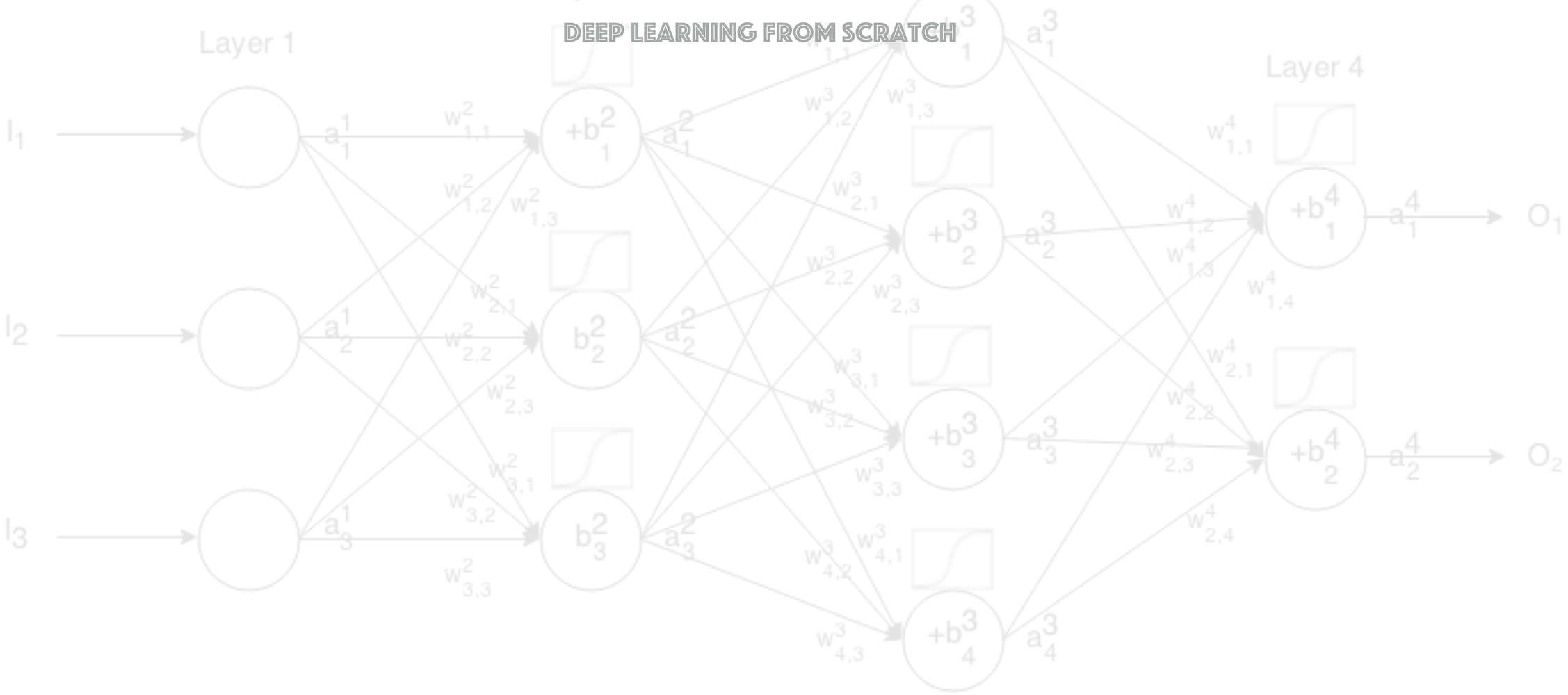

vicarious
home about news investors join contact
our mission:
build the next generation of A.I. algorithms
learn more
join our team
NEWS
Vicarious announces $40M Series B
Mar 25, 2014
Vicarious passes first Turing Test - CAPTCHA
Oct 28, 2013
Vicarious announces $15M Series A
Aug 21, 2012
Vicarious presents at Peter Thiel's Stanford Class
Jun 5, 2012


Dato
PRODUCTS
USES
LEARN
EVENTS
COMPANY
BLOG
DOWNLOAD
MACHINE LEARNING
that scales with your business
DOWNLOAD GRAPHLAB CREATE
Ultra-Fast Data Analytics
Best-In-Class Predictive Modeling
Production-Ready Data Science


MetaMind
Products
Demos
Company
Developers
Log In
Train your image classifier
UPLOAD IMAGES
Add Class


clarifai
ABOUT
TECHNOLOGY
API
NEWS
BLOG
JOBS
CONTACT
clarifai
Bring the future into focus with our world class visual recognition system
TRY IT NOW
API SIGN UP
Image recognition: Consumer products will drive enterprise breakthroughs
Machines Finally Match Monkeys in Key Image Recognition Test
A Startup's Neural Network Can Understand Video


SKYMIND
HOME
ABOUT
PRODUCTS
SERVICES & SUPPORT
TRAINING
RESOURCES
CONTACT
DEEP LEARNING
for industry : deeplearning4j : nd4j



What is Deep Learning?

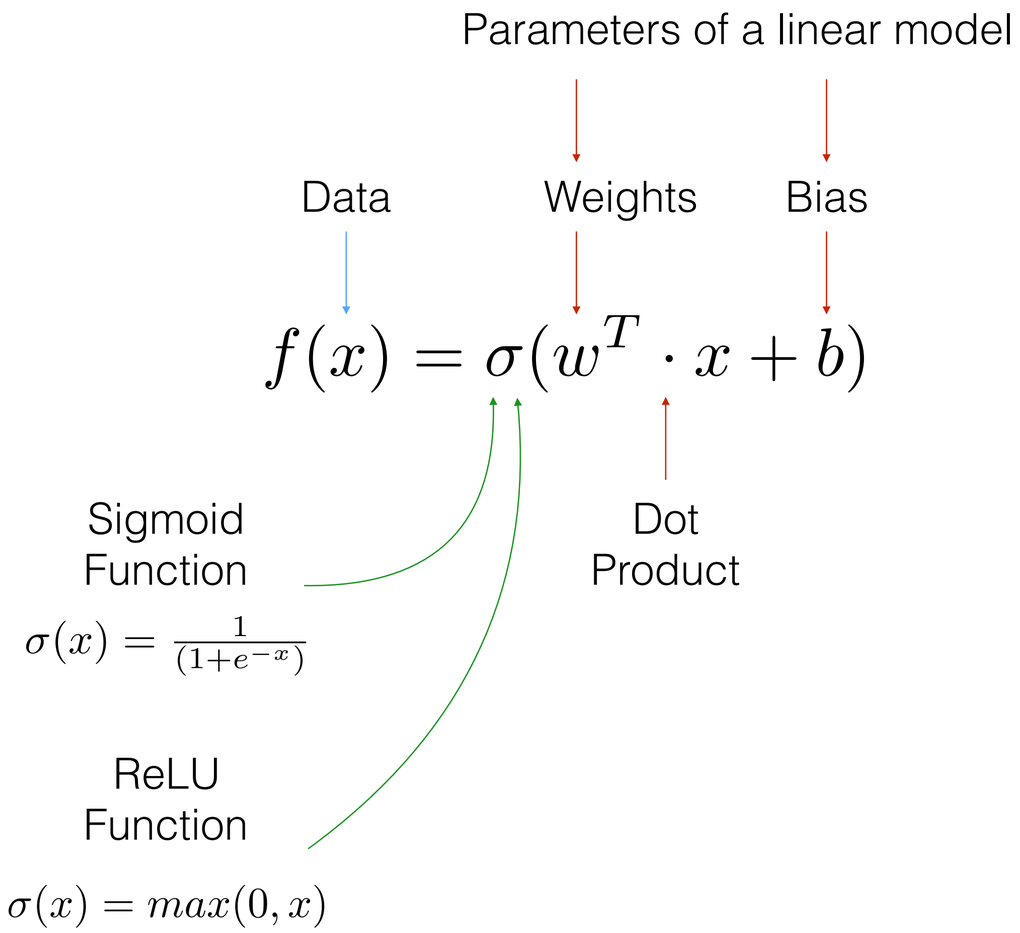
Learning from Data

Training data: a set of $(x^{(m)}, y^{(m)})$ pairs.
Learn a function $f_w : x \to y$ to predict on new inputs x.

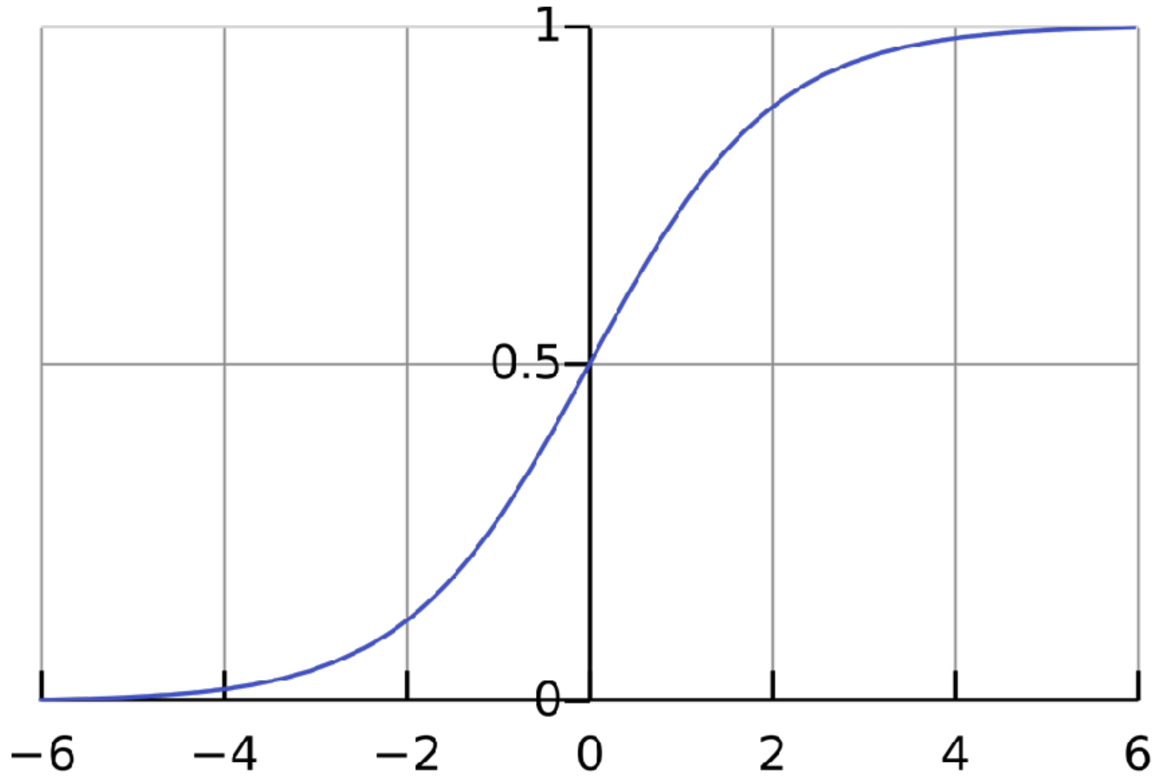
1. Choose a model function family f_w.

2. Optimize parameters w.

1-layer neural net model

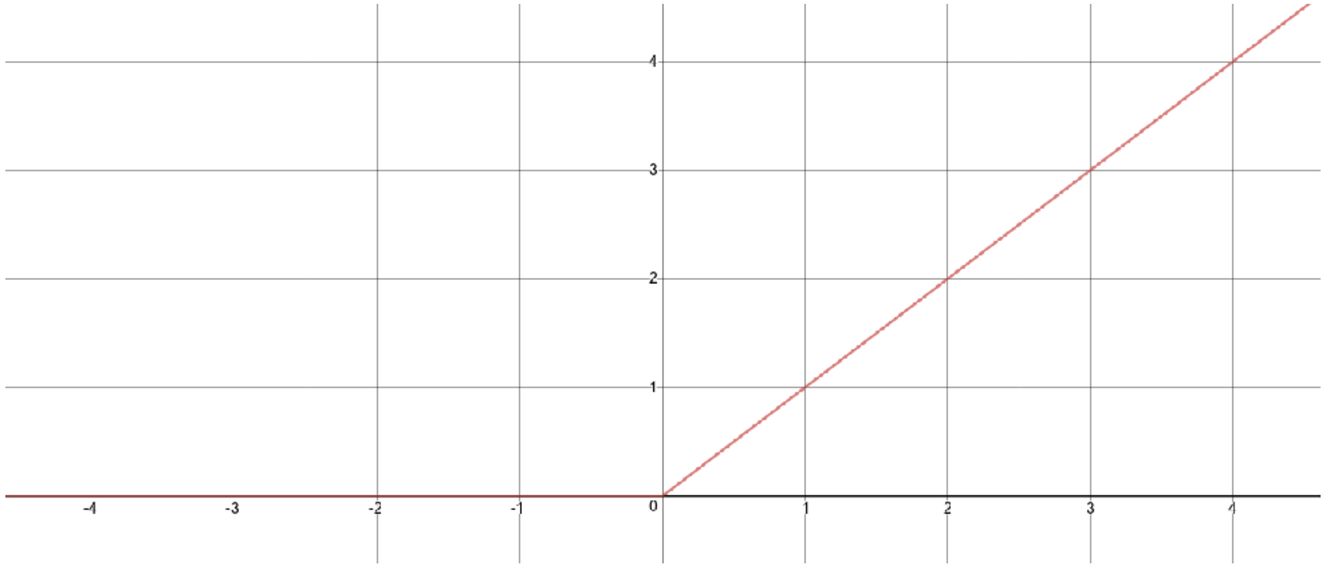


Non-linearity



$$\sigma(x) = \frac{1}{(1+e^{-x})}$$

Non-linearity



$$\sigma(x) = max(0, x)$$

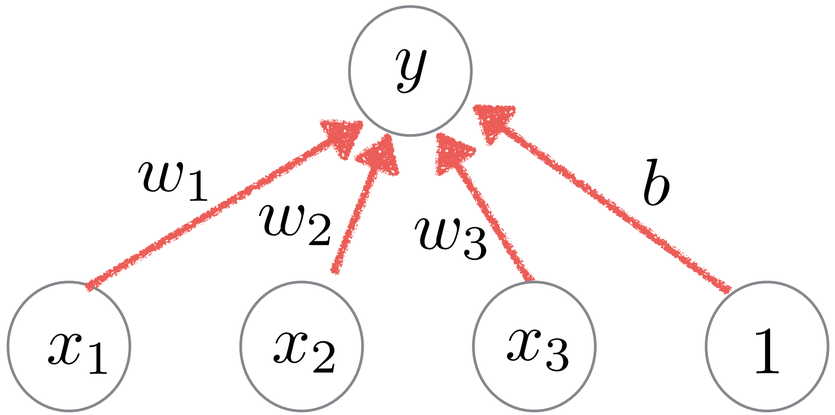
Non-linearity

Table 3: Non-linearities tested.

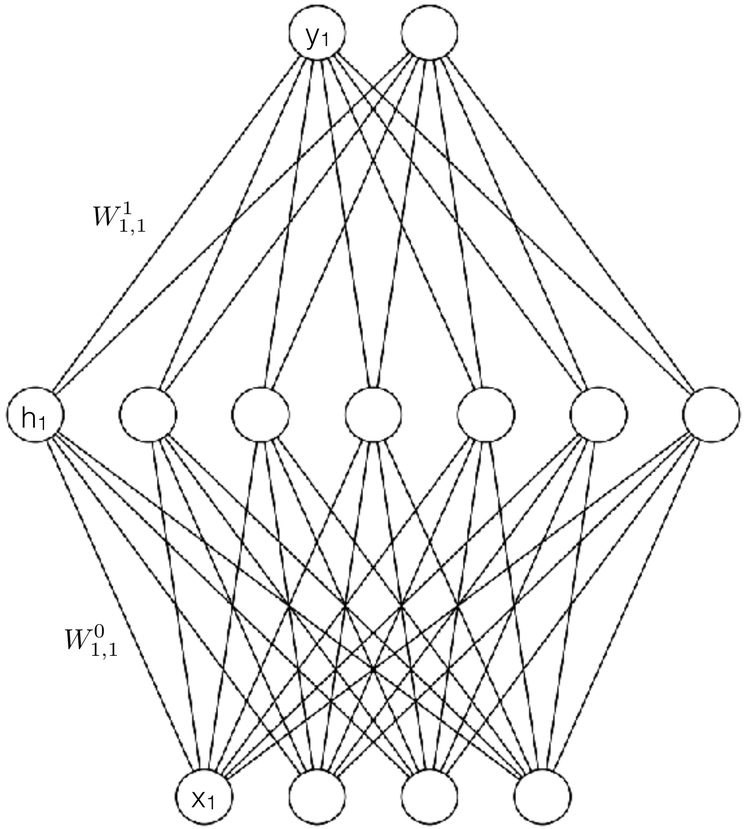
Name	Formula	Year
none	$y = x$	-
sigmoid	$y = \frac{1}{1+e^{-x}}$	1986
tanh	$y = \frac{e^{2x}-1}{e^{2x}+1}$	1986
ReLU	$y = \max(x, 0)$	2010
(centered) SoftPlus	$y = \ln(e^x + 1) - \ln 2$	2011
LReLU	$y = \max(x, \alpha x)$, $\alpha \approx 0.01$	2011
maxout	$y = \max(W_1 x + b_1, W_2 x + b_2)$	2013
APL	$y = \max(x,0) + \sum_{s=1}^{S} a_i^s \max(0, -x + b_i^s)$	2014
VLReLU	$y = \max(x, \alpha x)$, $\alpha \in 0.1, 0.5$	2014
RReLU	$y = \max(x, \alpha x)$, $\alpha = \text{random}(0.1, 0.5)$	2015
PReLU	$y = \max(x, \alpha x)$, α is learnable	2015
ELU	$y = x$, if $x \geq 0$, else $\alpha(e^x - 1)$	2015

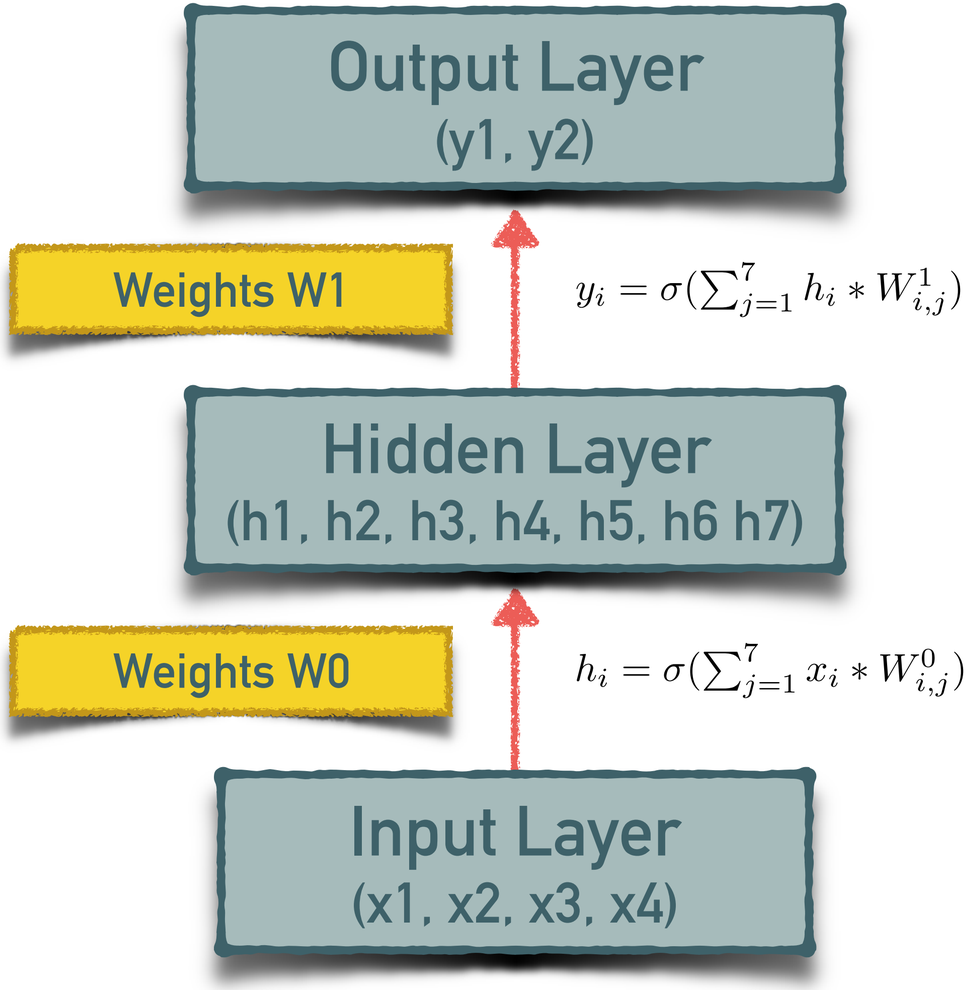
Graphical representation of 1-layer neural net model

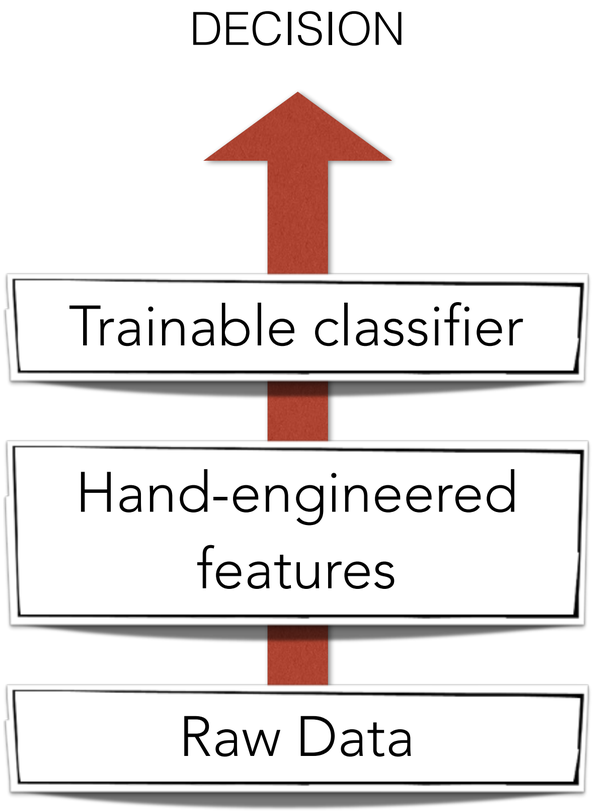
$$f(x) = \sigma(w^T \cdot x + b)$$



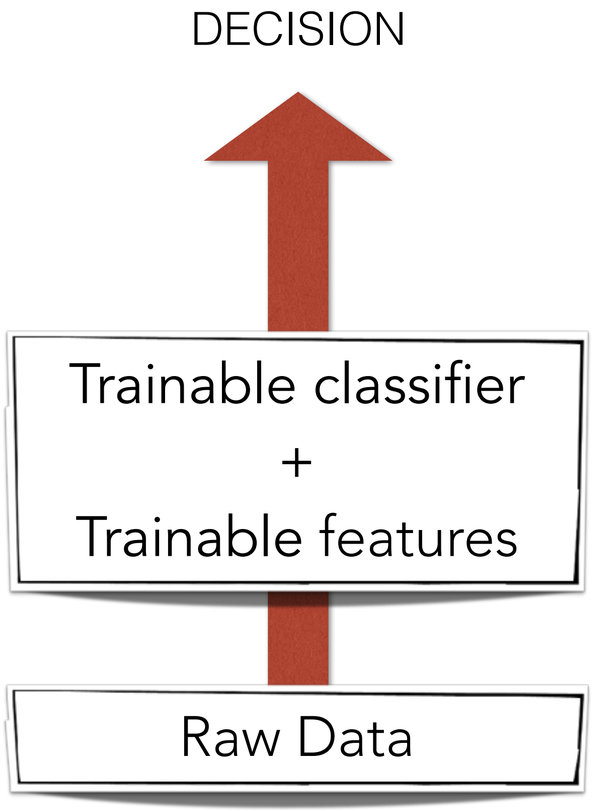
2-layer neural net model





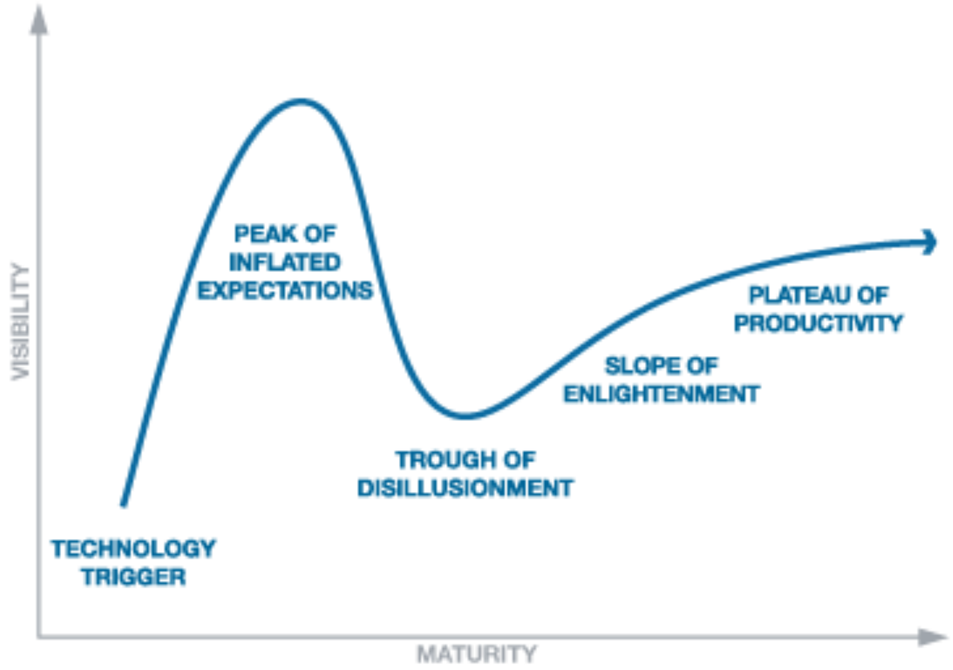


STANDARD MACHINE LEARNING

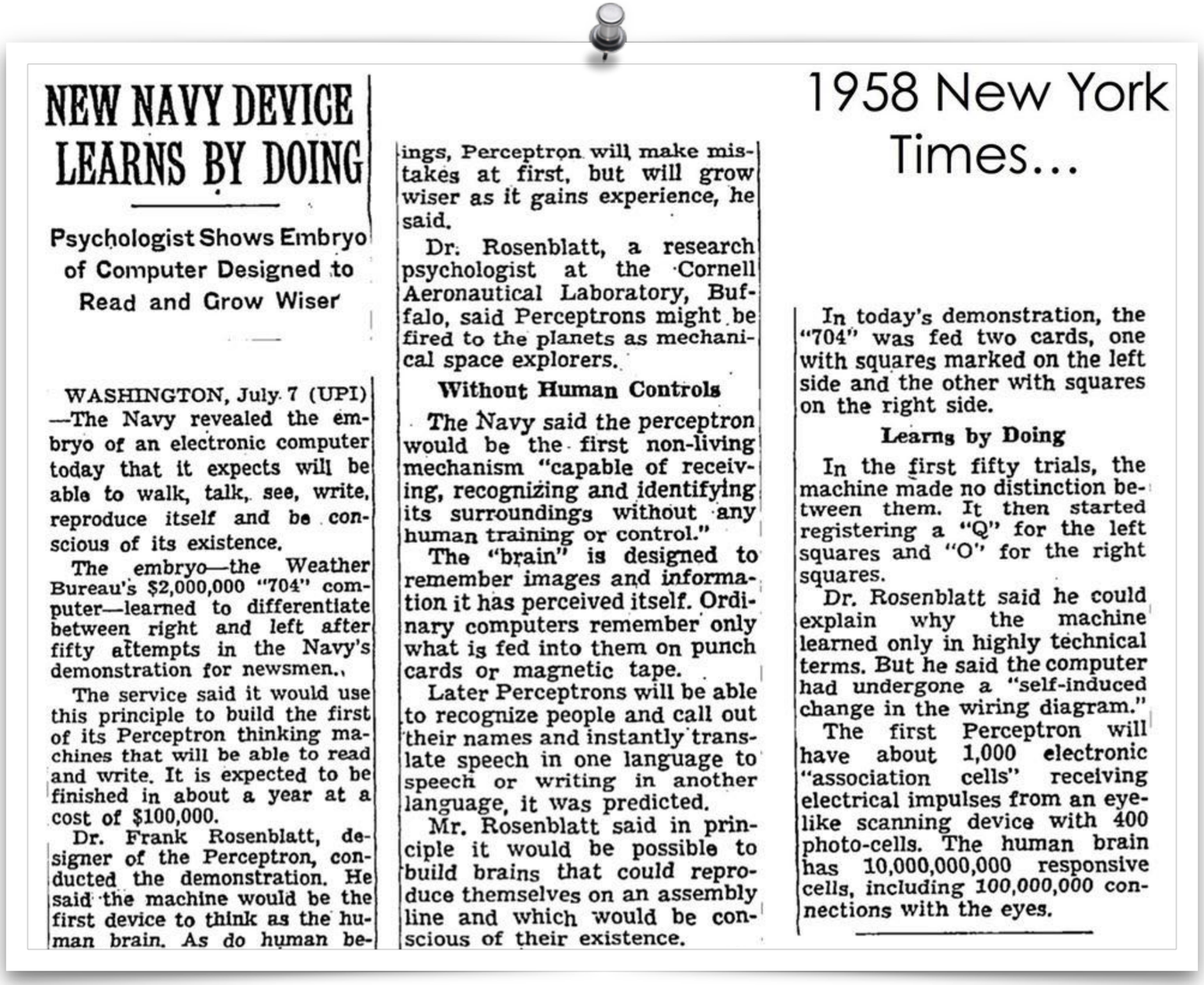


DEEP LEARNING

Is Deep Learning Overhyped?



Hype is not new



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.

Dr. Frank Rosenblatt, designer of the Perceptron, conducted the demonstration. He said the machine would be the first device to think as the human brain. As do human beings, Perceptron will make mistakes at first, but will grow wiser as it gains experience, he said.

Dr. Rosenblatt, a research psychologist at the Cornell Aeronautical Laboratory, Buffalo, said Perceptrons might be fired to the planets as mechanical space explorers.

Without Human Controls

The Navy said the perceptron would be the first non-living mechanism "capable of receiving, recognizing and identifying its surroundings without any human training or control."

The "brain" is designed to remember images and information it has perceived itself. Ordinary computers remember only what is fed into them on punch cards or magnetic tape.

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.

Mr. Rosenblatt said in principle it would be possible to build brains that could reproduce themselves on an assembly line and which would be conscious of their existence.

In today's demonstration, the "704" was fed two cards, one with squares marked on the left side and the other with squares on the right side.

Learns by Doing

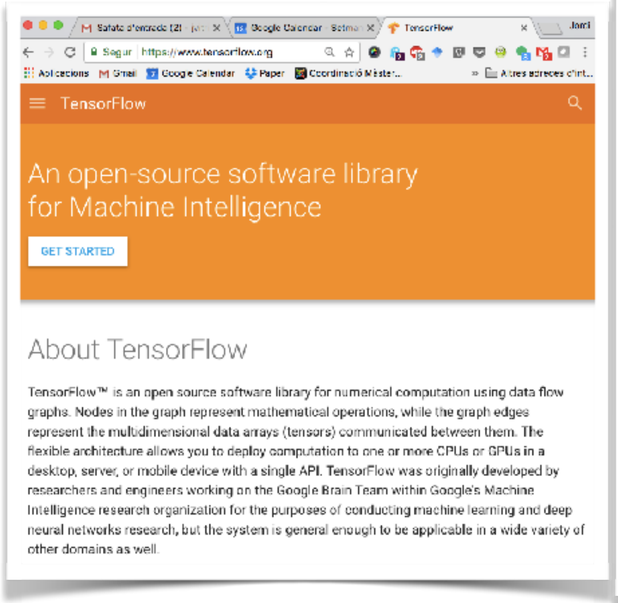
In the first fifty trials, the machine made no distinction between them. It then started registering a "Q" for the left squares and "O" for the right squares.

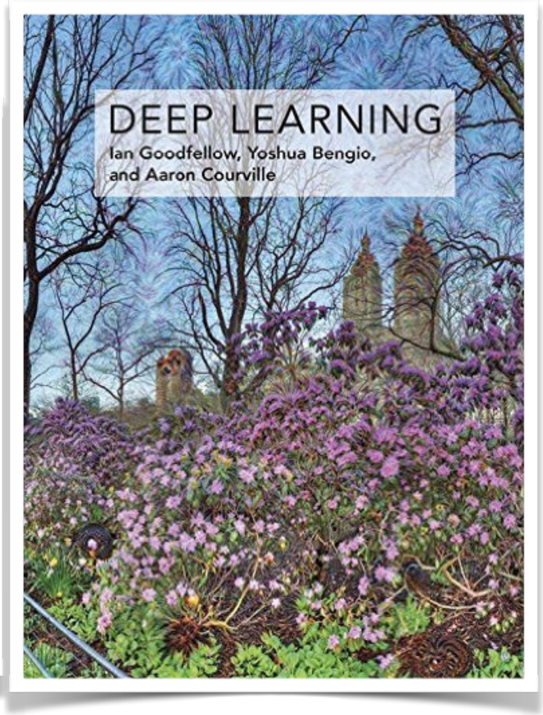
Dr. Rosenblatt said he could explain why the machine learned only in highly technical terms. But he said the computer had undergone a "self-induced change in the wiring diagram."

The first Perceptron will have about 1,000 electronic "association cells" receiving electrical impulses from an eye-like scanning device with 400 photo-cells. The human brain has 10,000,000,000 responsive cells, including 100,000,000 connections with the eyes.

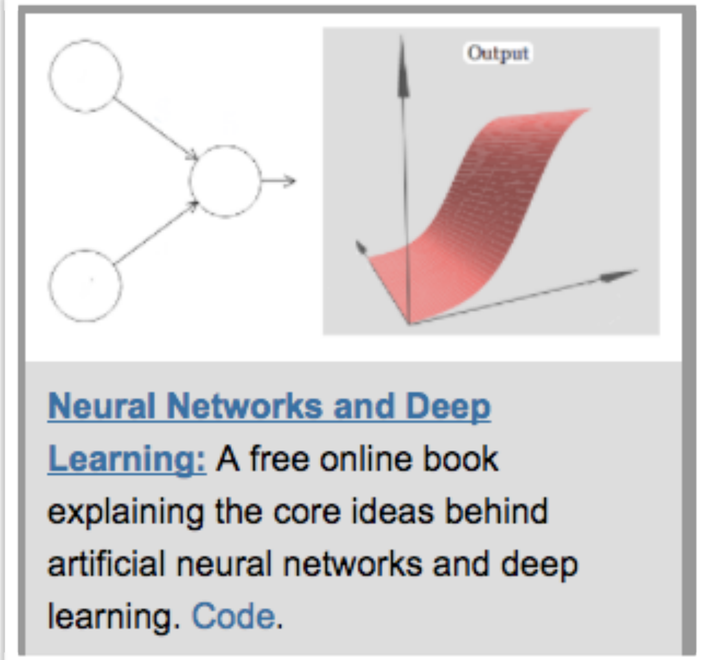
1958 New York Times...

Bibliography





Deep Learning, Yoshua Bengio, Ian Goodfellow, Aaron Courville, MIT Press, 2017
http://www.deeplearningbook.org/



By Michael Nielsen / Jan 2017
http://neuralnetworksanddeeplearning.com/