

INTRODUCTION TO DATA ANALYSIS, PATTERN RECOGNITION & ARTIFICIAL INTELLIGENCE

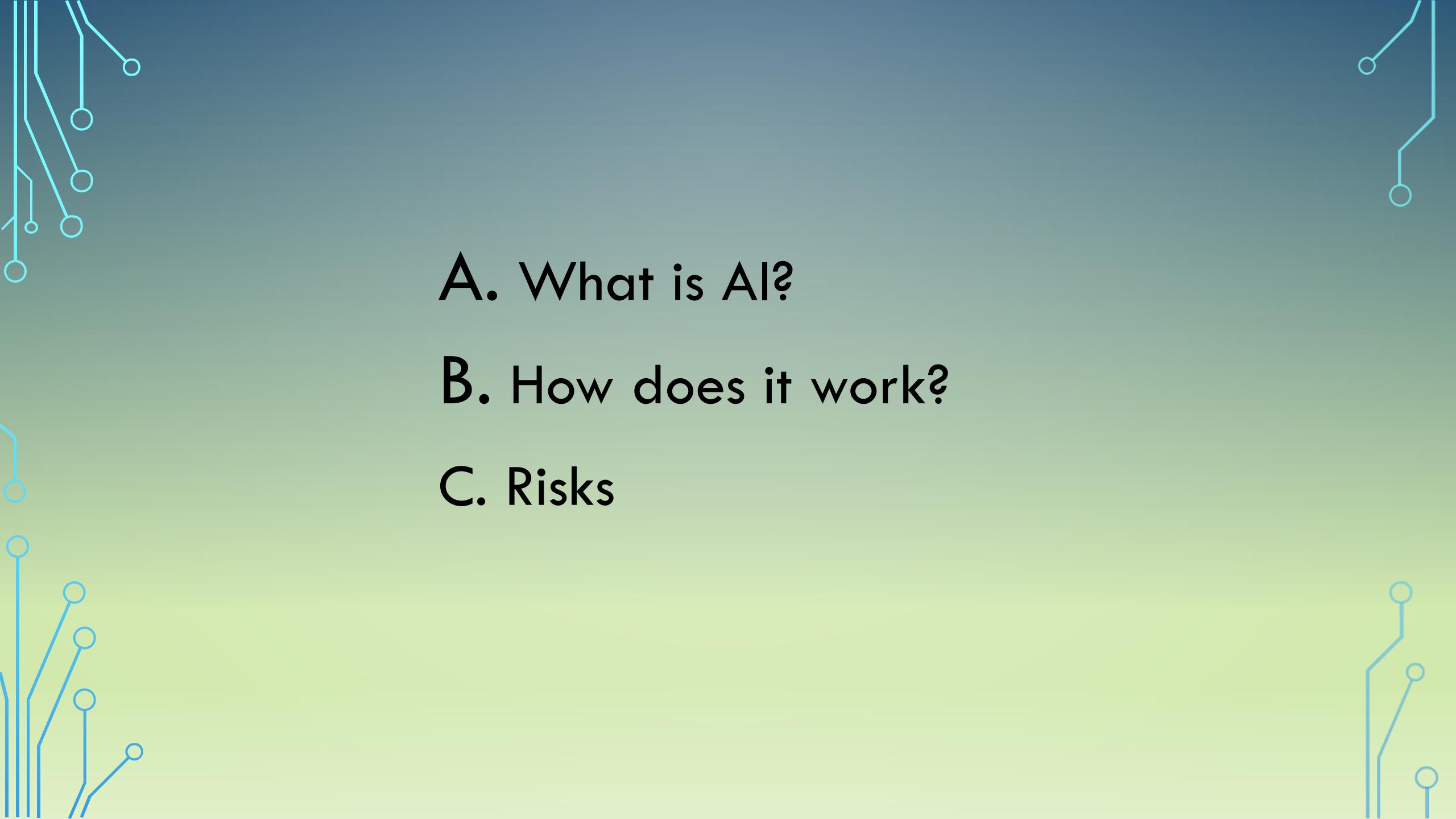
Raúl Benítez

raul.benitez@upc.edu



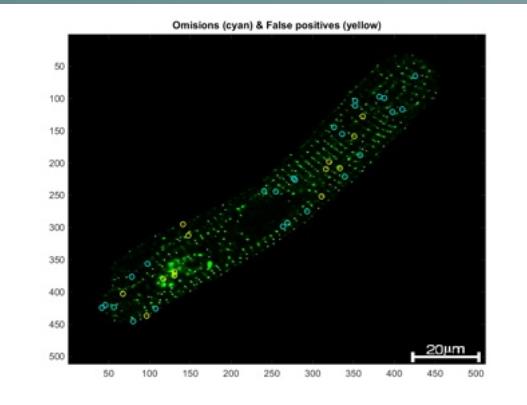
UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH

Centre de Recerca en Enginyeria Biomèdica

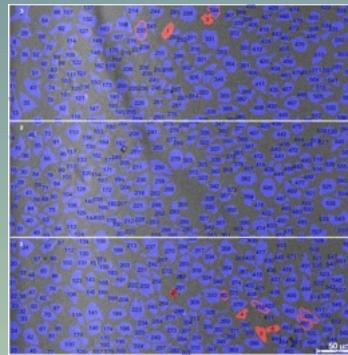
- 
- A. What is AI?**
 - B. How does it work?**
 - C. Risks**

ANALYSIS OF MEDICAL & BIOLOGICAL IMAGES

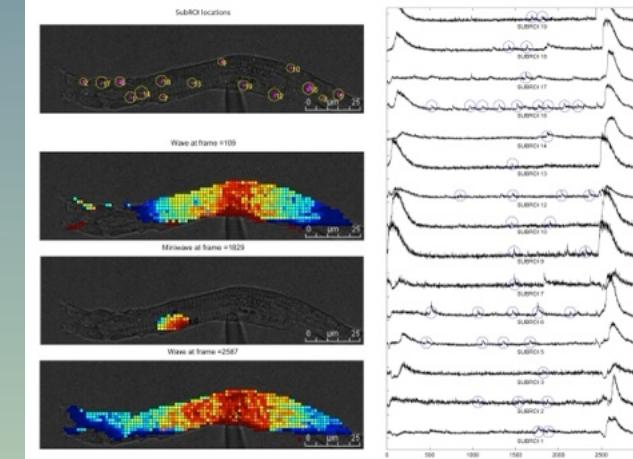
Molecular receptors



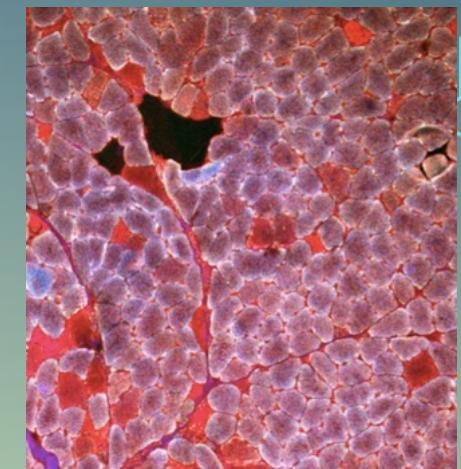
Cell cultures



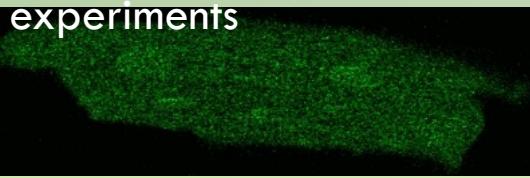
Calcium dynamics



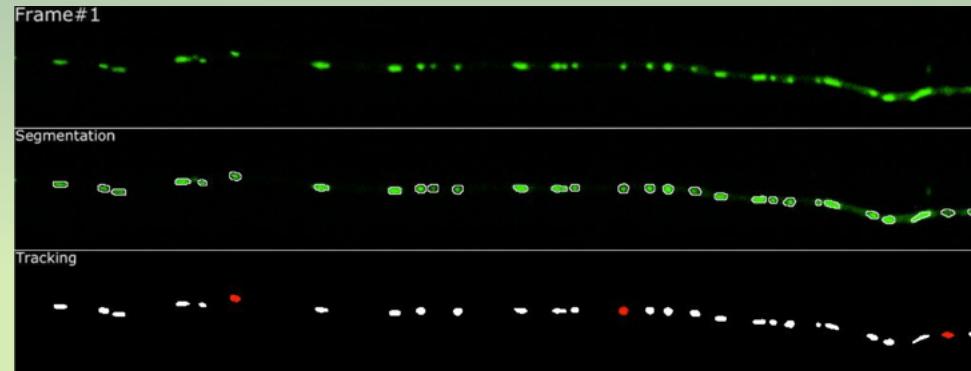
Pathology



Single-cell experiments

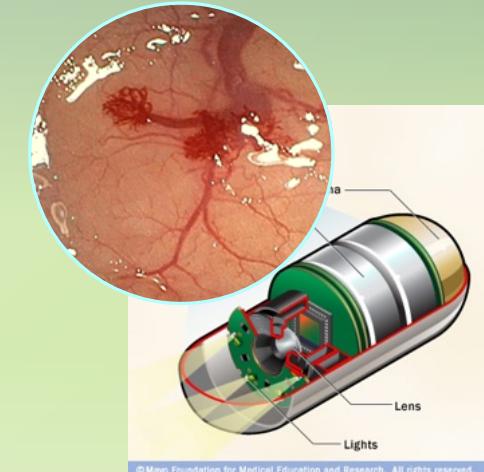


Mitochondrial transport



Extraction & analysis of multiscale biomarkers

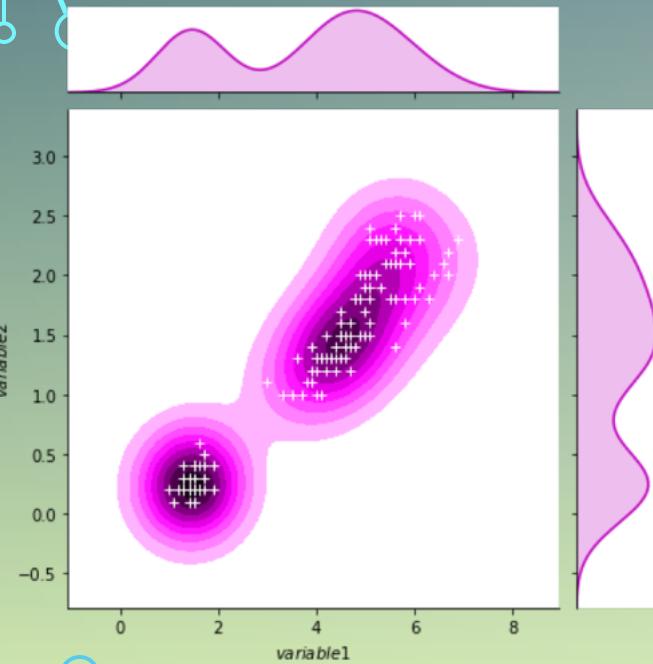
Endoscopy



A. WHAT IS AI?

artificial Intelligence
data analysis
machine learning
unsupervised learning
pattern recognition
data modeling
expert systems
statistical inference
deep learning
big data
business intelligence
knowledge retrieval
cybernetics
data analytics

Data



Text
Images
Multivariate numerical data
Genetics
Audio, video
HETEROGENEOUS

Mathematics

$$\begin{aligned} p(\mathcal{D}|\theta) &= p(x_1, x_2, \dots, x_n | \mu, \sigma^2) \\ &= \prod_{i=1}^n p(x_i | \theta) \\ &= \prod_{i=1}^n \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x_i - \mu)^2}{2\sigma^2}\right) \\ &= \left(\frac{1}{2\pi\sigma^2}\right)^{\frac{n}{2}} \exp\left(-\frac{\sum_{i=1}^n (x_i - \mu)^2 + n(\frac{1}{n} \sum_{i=1}^n x_i - \mu)^2}{2\sigma^2}\right) \end{aligned}$$

Statistics
Geometry
Optimization
Stochastic processes

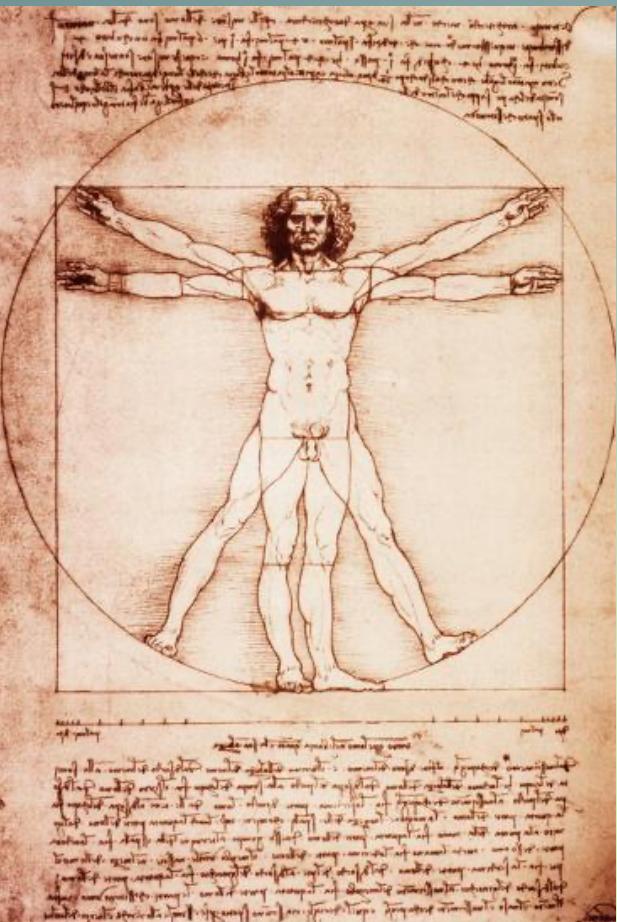
Computer Science

```
1 import numpy as np
2 from sklearn import decomposition
3 from sklearn import datasets
4
5 iris = datasets.load_iris()
6 X = iris.data
7 y = iris.target
8
9 pca = decomposition.PCA(n_components=2)
10 pca.fit(X)
11 Xproj = pca.transform(X)
```

Algorithms
Computational complexity
Information theory
Network analysis

We emulate human body

Biomechanics



Humanoid robotics

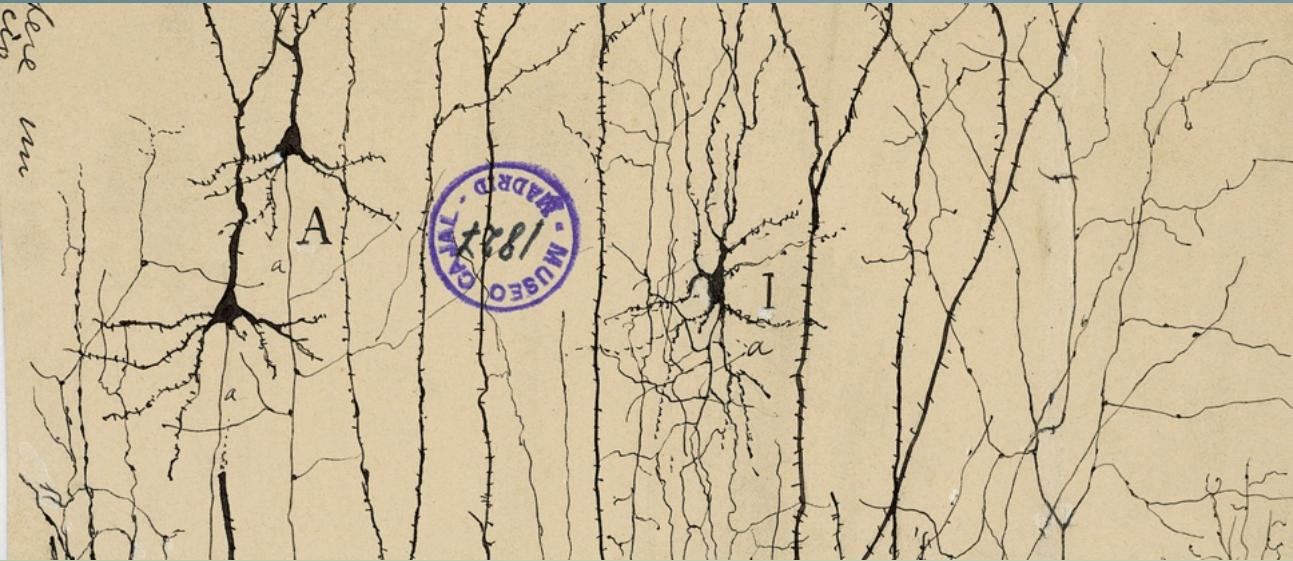


imgflip.com

Robotics



WE EMULATE HUMAN MIND



Cajal & Golgi **Nobel 1906**

Hebbian learning 1949

Hodkin & Huxley 1952 **Nobel 1963**

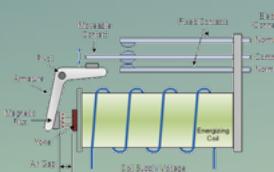
Hubel & Wiesel 1959 Visual Cortex **Nobel 1981**

Techology & IA

Mechanical calculators
(s.XVII)



Electromechanical relay
(s. XIX)



Telegraph
1830



Enigma
machine
1918

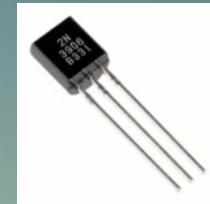
Vacuum tube(1904)



ENIAC 1946



The transistor (1947)
Nobel 1956



IBM 608
1957



GPUs (2018)



IA today



RESEARCH

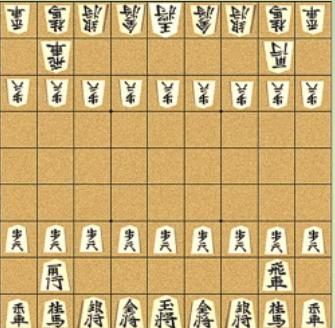
COMPUTER SCIENCE

A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play

David Silver^{1,2,3,†}, Thomas Hubert^{1*}, Julian Schrittwieser^{1,4}, Ioannis Antonoglou¹, Matthew Lai¹, Arthur Guez¹, Marc Lanctot¹, Laurent Sifre¹, Dharshan Kumaran¹, Thore Graepel¹, Timothy Lillicrap¹, Karen Simonyan¹, Demis Hassabis^{1,†}

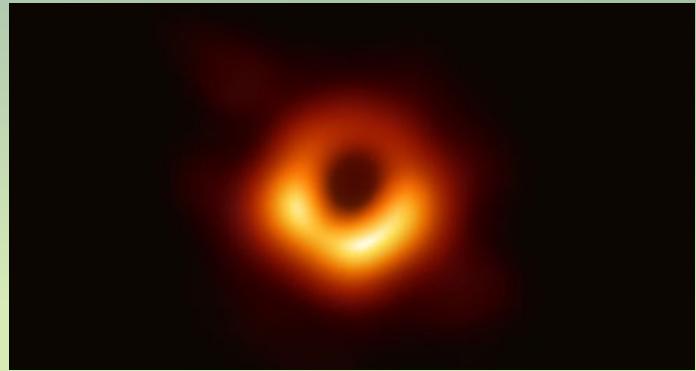
The game of chess is the longest-studied domain in the history of artificial intelligence. The strongest programs are based on a combination of sophisticated search techniques, domain-specific adaptations, and handcrafted evaluation functions that have been refined by human experts over several decades. By contrast, the AlphaGo Zero program recently achieved superhuman performance in the game of Go by reinforcement learning from self-play. In this paper, we generalize this approach into a single AlphaZero algorithm that can achieve superhuman performance in many challenging games. Starting from random play and given no domain knowledge except the game rules, AlphaZero convincingly defeated a world champion program in the games of chess and shogi (Japanese chess), as well as Go.

Science, 362 (6419), 1140-1144 (2018)

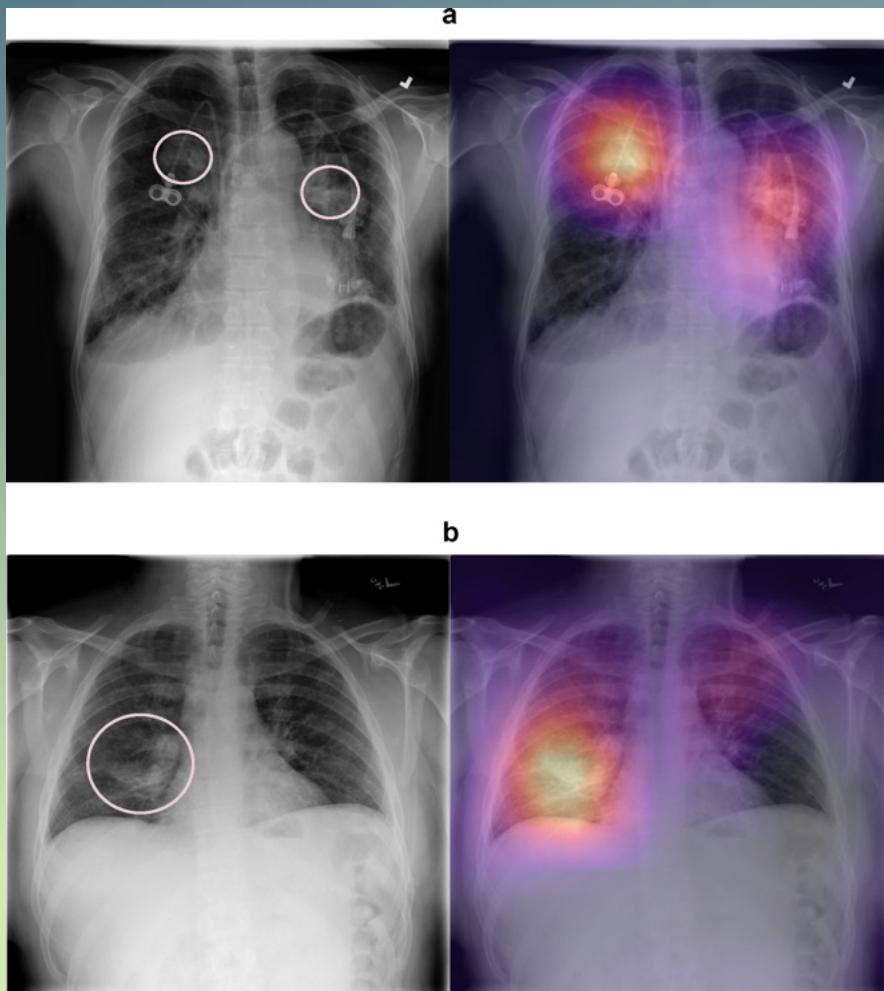


COMPUTATIONAL COMPLEXITY

First black hole image 2019
Event Horizon Telescope (EHT)
The Astrophysical Journal Letters



Applications



14 pathologies 69,682,060 parameters

PLOS Medicine 15(11): e1002686 (2018)

PLOS MEDICINE

RESEARCH ARTICLE

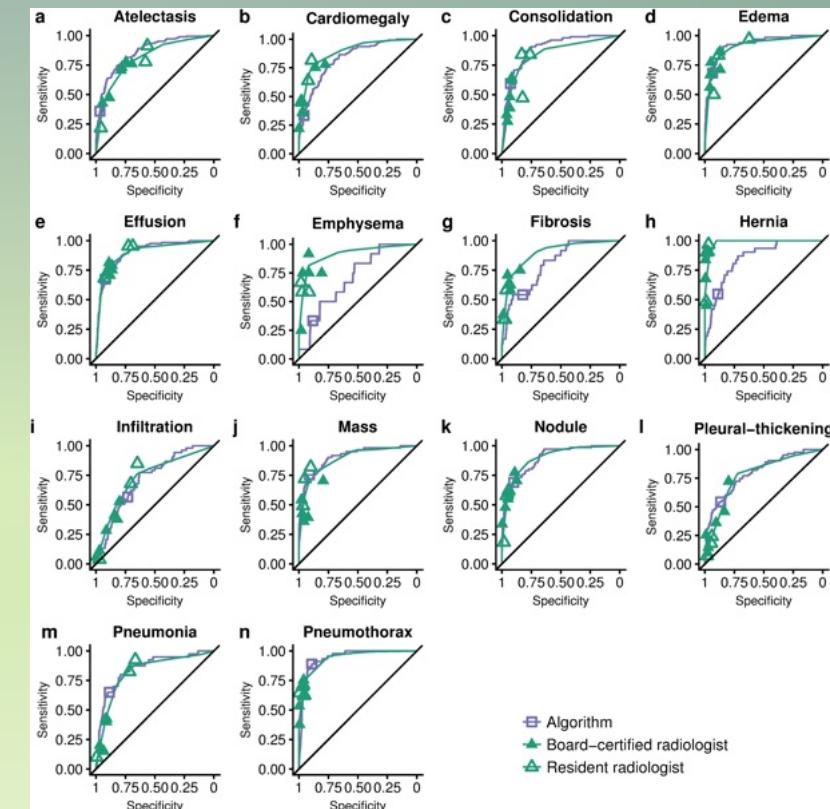
Deep learning for chest radiograph diagnosis: A retrospective comparison of the CheXNeXt algorithm to practicing radiologists

Pranav Rajpurkar^{1,2*}, Jeremy Irvin^{2†}, Robyn L. Ball², Kaylie Zhu¹, Brandon Yang¹, Hershel Mehta¹, Tony Duan¹, Daisy Ding¹, Aarti Bagul¹, Curtis P. Langlotz³, Bhavik N. Patel², Kristen W. Yeom², Katie Shpanskaya^{2,4}, Francis G. Blankenberg², Jayne Seekins², Timothy J. Arthein², David A. Mong², Suhwan S. Halabi², Evan J. Zucker², Andrew Y. Ng^{1,2}, Matthew P. Lungren²

¹ Department of Computer Science, Stanford University, Stanford, California, United States of America, ² Department of Medicine, Quantitative Sciences Unit, Stanford University, Stanford, California, United States of America, ³ Department of Radiology, Stanford University, Stanford, California, United States of America, ⁴ Department of Radiology, Duke University, Durham, North Carolina, United States of America, ⁵ Department of Radiology, University of Colorado, Denver, Colorado, United States of America

* These authors contributed equally to this work.
† These authors share first authorship on, and contributed equally to, this work.
✉ pranavar@cs.stanford.edu

Check for updates

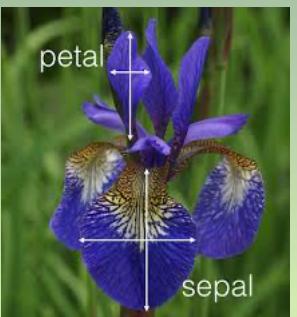


B. HOW DOES IT WORK?

MATHEMATICAL REPRESENTATION OF DATA

A: m observations \times n features

$$A_{m \times n} = \begin{pmatrix} x_1^1 & x_2^1 & \cdots & x_n^1 \\ x_1^2 & x_2^2 & \cdots & x_n^2 \\ x_1^3 & x_2^3 & \cdots & x_n^3 \\ \vdots & & & \\ x_1^m & x_2^m & \cdots & x_n^m \end{pmatrix}$$



	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa

Class labels $w(i)$

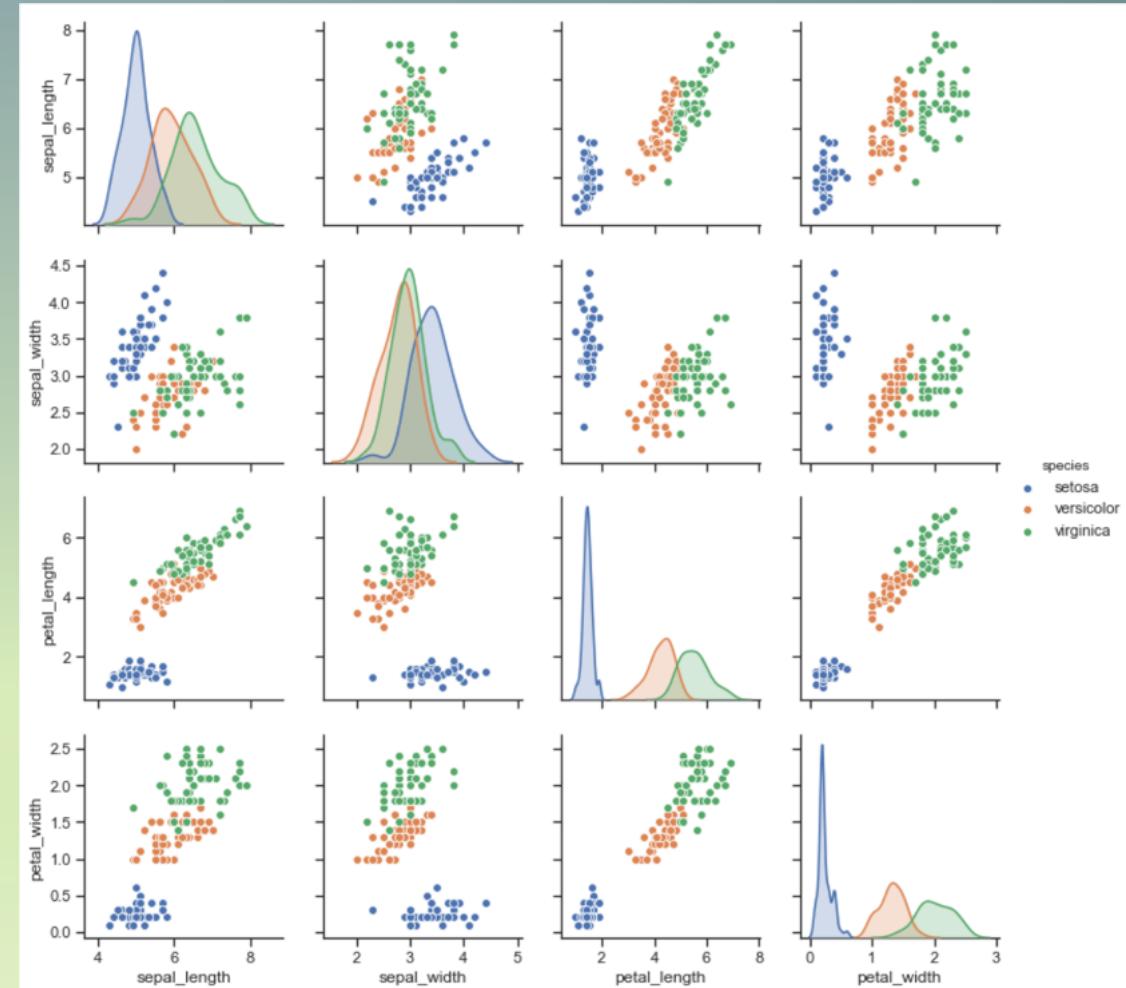
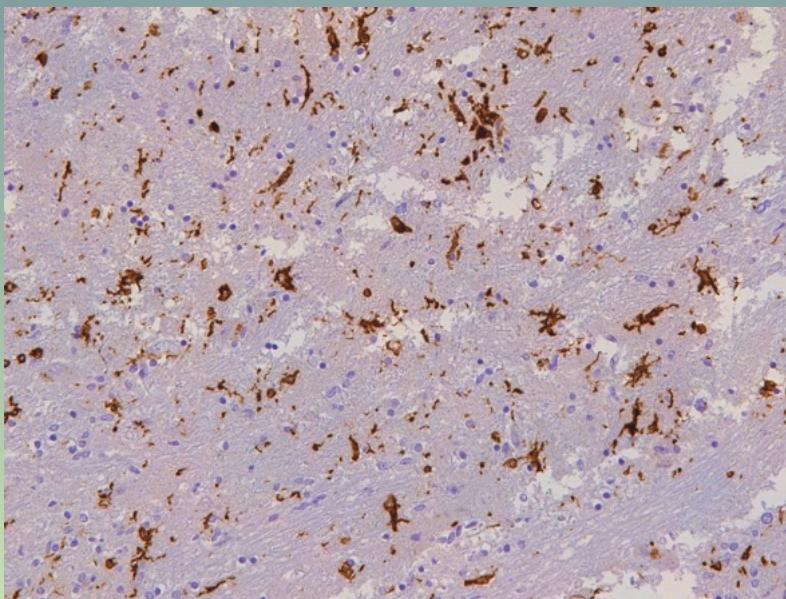


IMAGE DATA

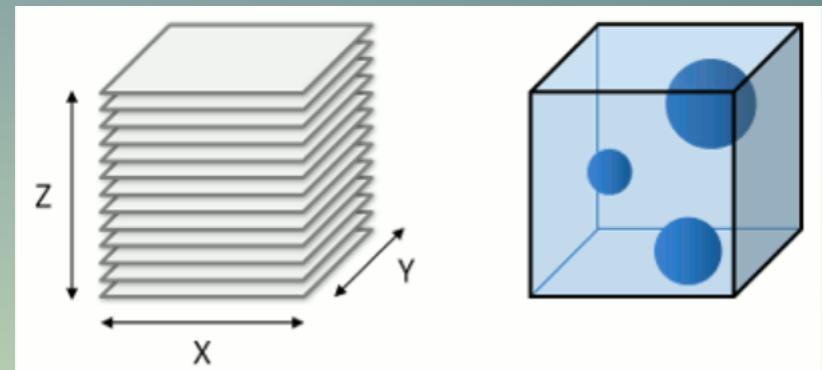
grayscale



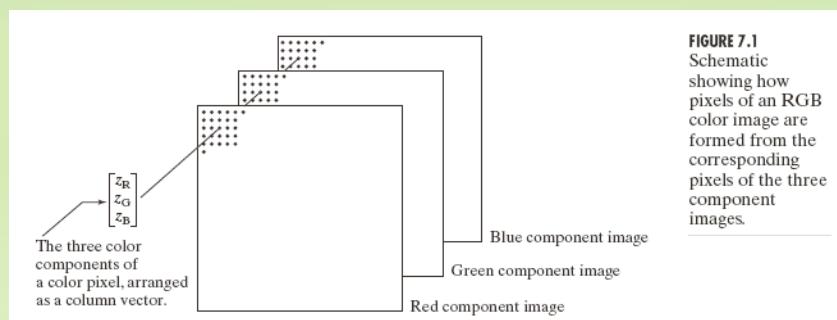
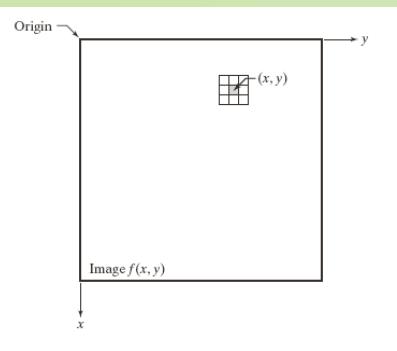
color images



3D stacks

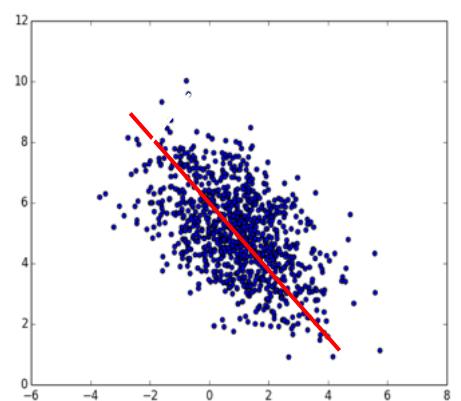


video sequences

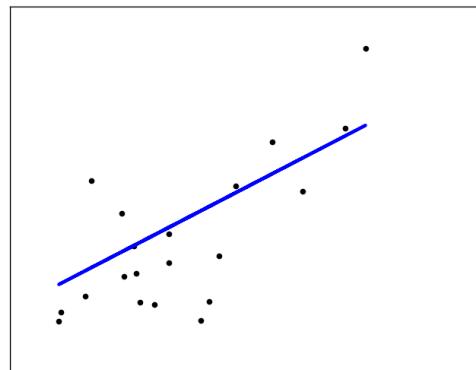


PATTERN RECOGNITION

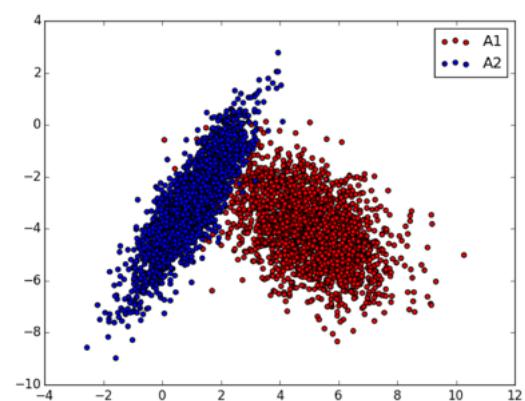
DIMENSIONALITY REDUCTION



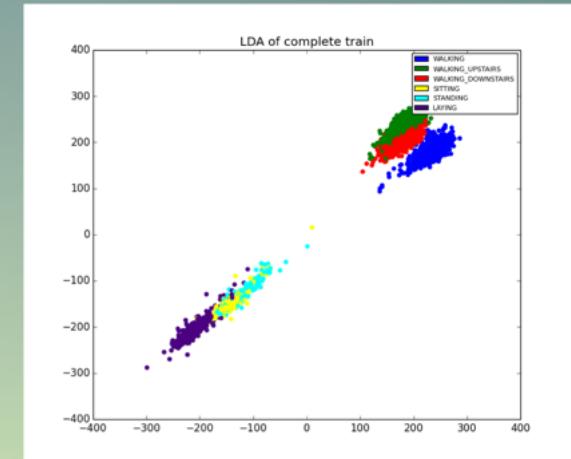
MULTIVARIATE REGRESSION



SUPERVISED CLASSIFICATION



CLUSTER ANALYSIS



Principal Component Analysis
Singular Value Decomposition
Independent Component Analysis

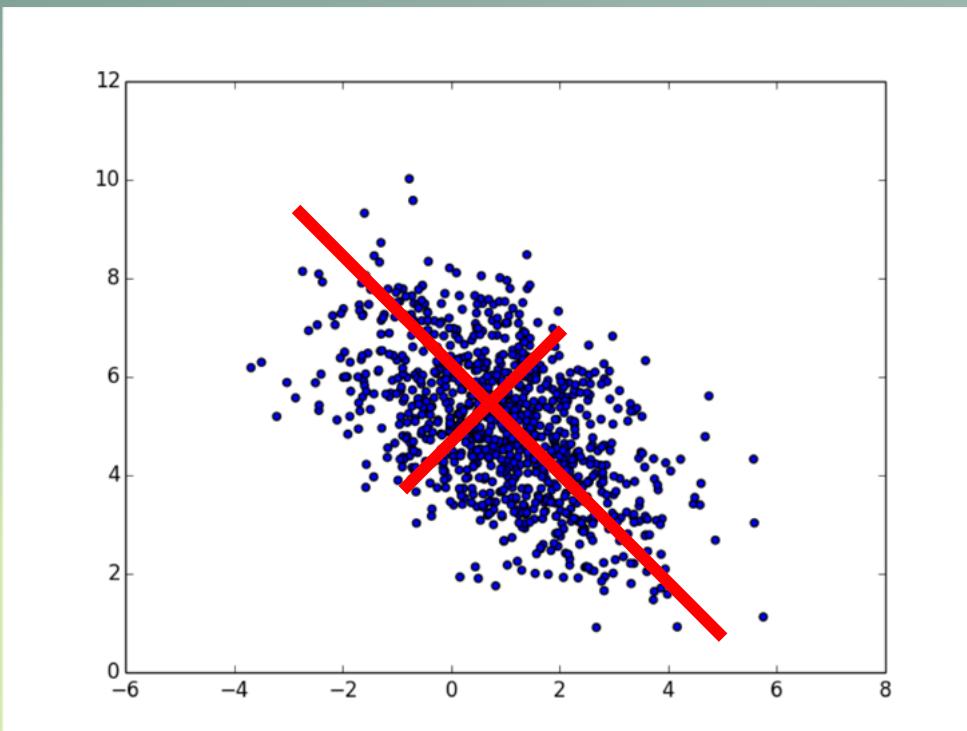
Multivariate linear regression
Regression using decision trees
Support Vector Machines

Discriminant Analysis
Artificial Neural Networks
Support Vector Machines
Bayesian methods
Decision Trees
Ensemble classifiers
Deep Learning

K-means
Hierarchical clustering
Gaussian Mixture Methods

DIMENSIONALY REDUCTION PRINCIPAL COMPONENT ANALYSIS (PCA)

Reduce dimensionality of data



Mathematics

$$C_{n \times n} = (A - \bar{A})^T (A - \bar{A})$$

$$C \cdot \vec{v}_i = \lambda_i \cdot \vec{v}_i, \quad i = 1 \dots n$$

Computer Science

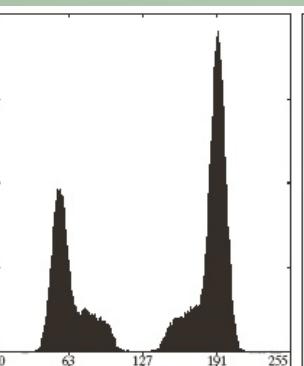


```
21 # Obtain covariance matrix:  
22 A1 = A - A.mean(0)  
23 matcov = dot(A1.transpose(),A1)  
24  
25 # Diagonalization of covariance matrix:  
26 valp,vecp = linalg.eig(mtcov)  
27  
28 ind_creciente = argsort(valp) # sort eigenvalues
```

TRADITIONAL MACHINE LEARNING

Processament d'imatges

Filtering
Artefact removal



Segmentation
Feature extraction



FIGURE 11.13 (a) Noisy fingerprint. (b) Histogram. (c) Segmented result using a global threshold (the border was added manually for clarity). (Original courtesy of the National Institute of Standards and Technology.)

features
→

- Shape
- Size
- Color
- Texture
- Orientation
- Contours
- Intensity

Pattern Recognition

Supervised classification

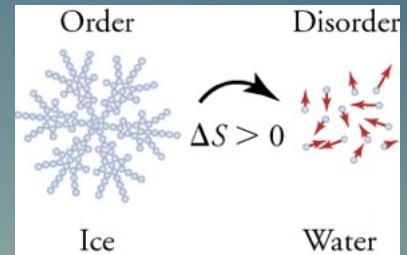
$$A_{m \times n} = \begin{pmatrix} x_1^1 & x_2^1 & \cdots & x_n^1 \\ x_1^2 & x_2^2 & \cdots & x_n^2 \\ x_1^3 & x_2^3 & \cdots & x_n^3 \\ \vdots & & & \\ x_1^m & x_2^m & \cdots & x_n^m \end{pmatrix} \quad w(i)$$

Fase 1: TRAINING
Fase 2: TEST

SHANNON'S ENTROPY

Physics: Measure of the disorder of a system

$$S = k \log W$$



Information theory: Information content

Set of symbols (alphabet): $a_1, a_2, a_3, \dots, a_n$

Probability of occurrence of each symbol in a message $p_1, p_2, p_3, \dots, p_n$

$$S = - \sum_{i=1}^n p_i \log p_i$$

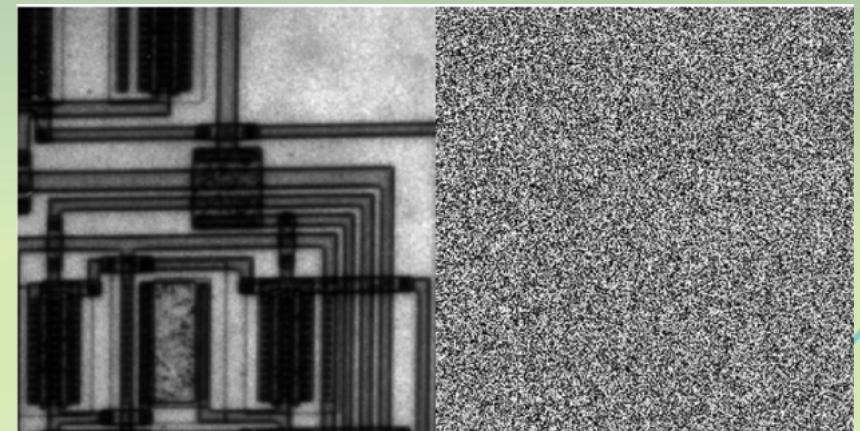
Example: Grayscale image

Alphabet: grayscale intensity levels $0, 1, 2, \dots, 255$

Probability of occurrence: Histogram $p_1, p_2, p_3, \dots, p_n$

$S = 6.9439$ bits/pixel

$S = 8$ bits/pixel



Low entropy



less bits needed to encode image



image compression

IMAGE RECOGNITION



Database of labelled images

- **Images** $I(i)$, $i = 1, \dots N_{im}$
- **Class labels** (name, happy/sad, healthy/ii, etc.) $w(i)$

Represent intra and inter-class variability (100-1000 images)
70% training set / 30% test set

THE TRAINING PROCESS



Training set

$I(i)$



FEATURE
EXTRACTION

Update parameters

SUPERVISED
CLASSIFIER

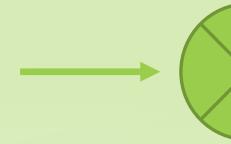
Error measure

$w(i)$

Actual class

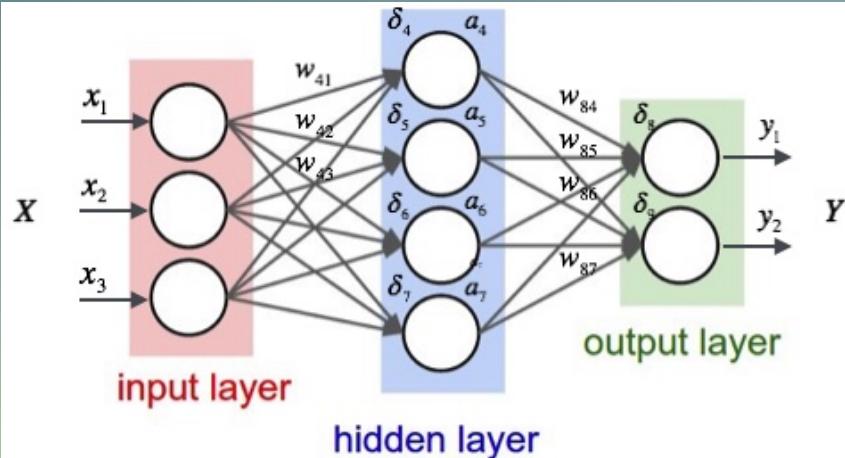
$\hat{w}(i)$

Predicted class

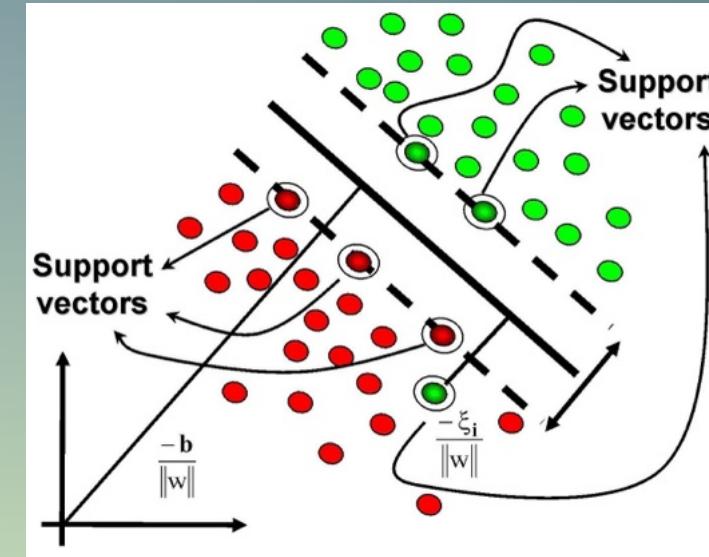


DIFFERENT SUPERVISED ALGORITHMS

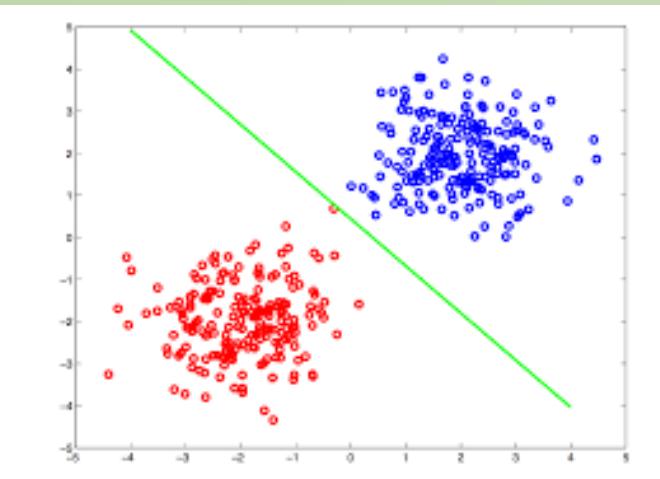
Neural Networks



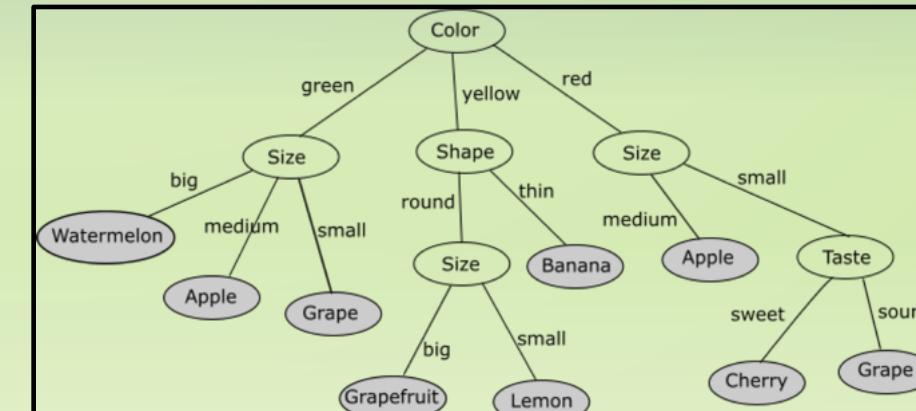
Support Vector Machines



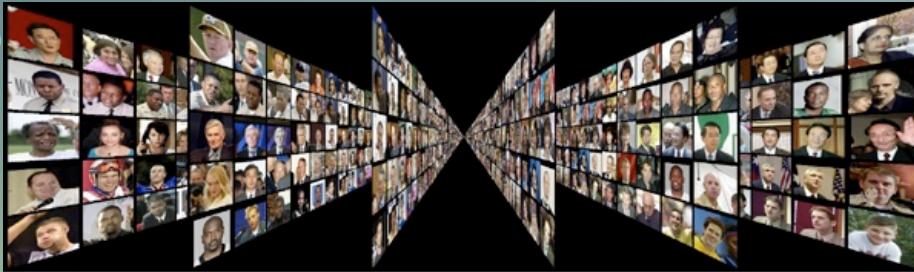
Probabilistic discriminants



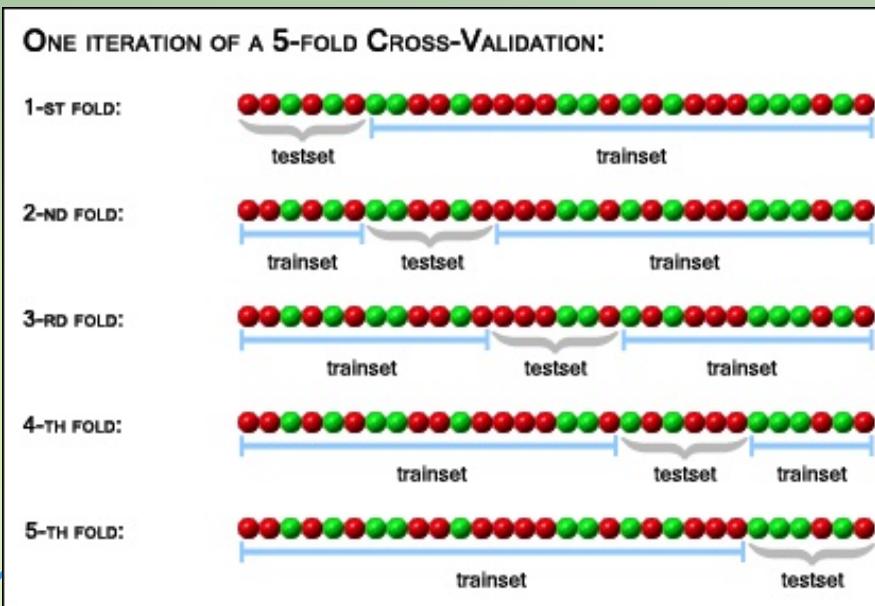
Decision trees



TEST: PERFORMANCE EVALUATION



test set



$I(i)$

FEATURE
EXTRACTION

SUPERVISED
CLASSIFIER

$w(i)$

Actual
class

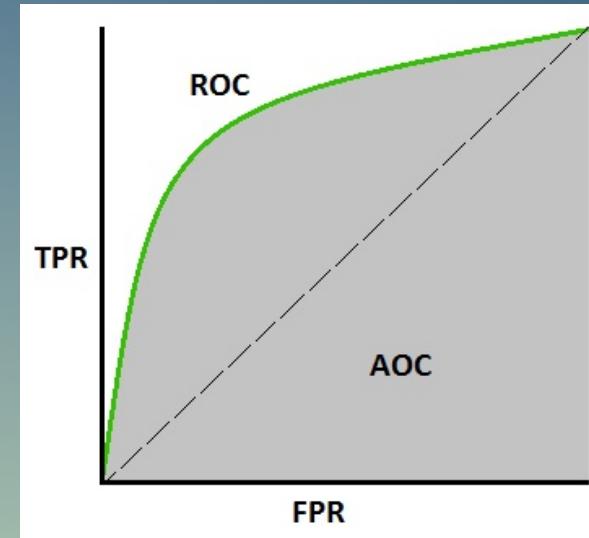
$\hat{w}(i)$

Predicted
class

Error measures

PERFORMANCE MEASURES

Confusion matrix



		True condition			
		Condition positive	Condition negative	Prevalence = $\frac{\sum \text{Condition positive}}{\sum \text{Total population}}$	Accuracy (ACC) = $\frac{\sum \text{True positive} + \sum \text{True negative}}{\sum \text{Total population}}$
Predicted condition	Predicted condition positive	True positive	False positive, Type I error	Positive predictive value (PPV), Precision = $\frac{\sum \text{True positive}}{\sum \text{Predicted condition positive}}$	False discovery rate (FDR) = $\frac{\sum \text{False positive}}{\sum \text{Predicted condition positive}}$
	Predicted condition negative	False negative, Type II error	Omission!	False omission rate (FOR) = $\frac{\sum \text{False negative}}{\sum \text{Predicted condition negative}}$	Negative predictive value (NPV) = $\frac{\sum \text{True negative}}{\sum \text{Predicted condition negative}}$
		True positive rate (TPR), Recall, Sensitivity, probability of detection, Power $= \frac{\sum \text{True positive}}{\sum \text{Condition positive}}$	False positive rate (FPR), Fall-out, probability of false alarm $= \frac{\sum \text{False positive}}{\sum \text{Condition negative}}$	Positive likelihood ratio (LR+) = $\frac{\text{TPR}}{\text{FPR}}$	Diagnostic odds ratio (DOR) = $\frac{\text{LR+}}{\text{LR-}}$
		False negative rate (FNR), Miss rate $= \frac{\sum \text{False negative}}{\sum \text{Condition positive}}$	Specificity (SPC), Selectivity, True negative rate (TNR) = $\frac{\sum \text{True negative}}{\sum \text{Condition negative}}$	Negative likelihood ratio (LR-) = $\frac{\text{FNR}}{\text{TNR}}$	

Example: Fetal plane recognition

Data:

Fetal ultrasound 1st trimester
506 RGB Images of size 960x720 pixels / class
3 class labels (fetal planes BPD, CLR, NT)



BPD - Biparietal Diameter



CLR - Crown-Rump-Length



NT - Nuchal translucency



TRAINING & TEST SETS

	BPD	CLR	NT
Training	406	406	406
Test	100	100	100
Total	506	506	506

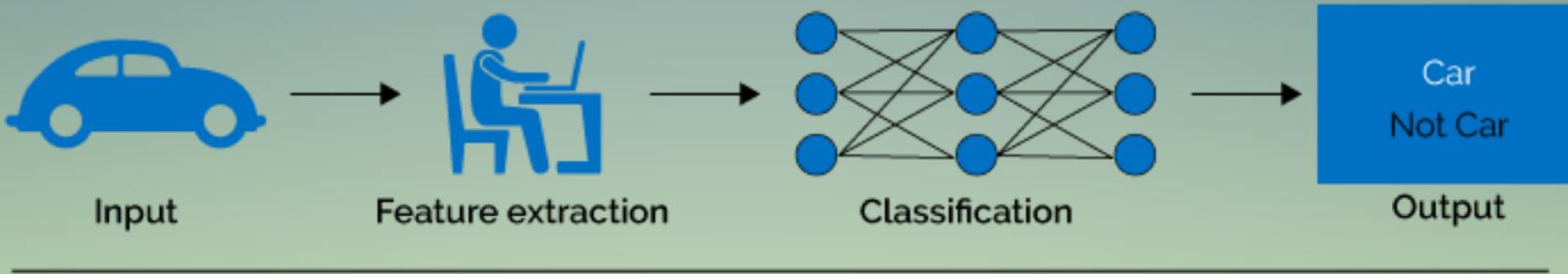
ACTUAL

	BPD	CLR	NT
BPD	95	4	1
CLR	1	97	2
NT	2	6	92

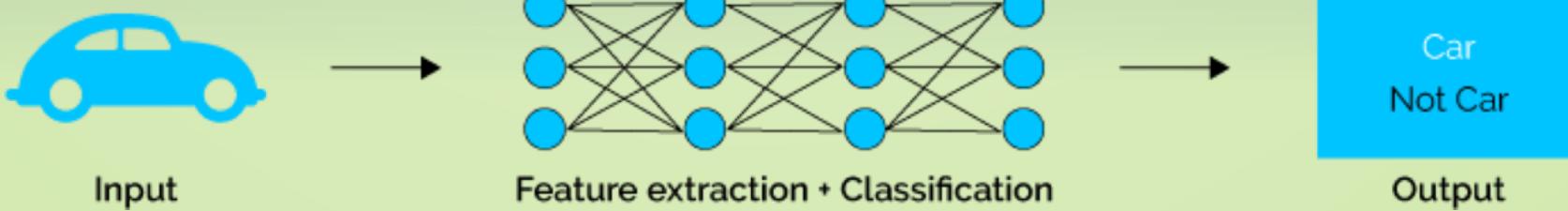
PREDICTED

CHANGE IN PARADIGM

Machine Learning



Deep Learning

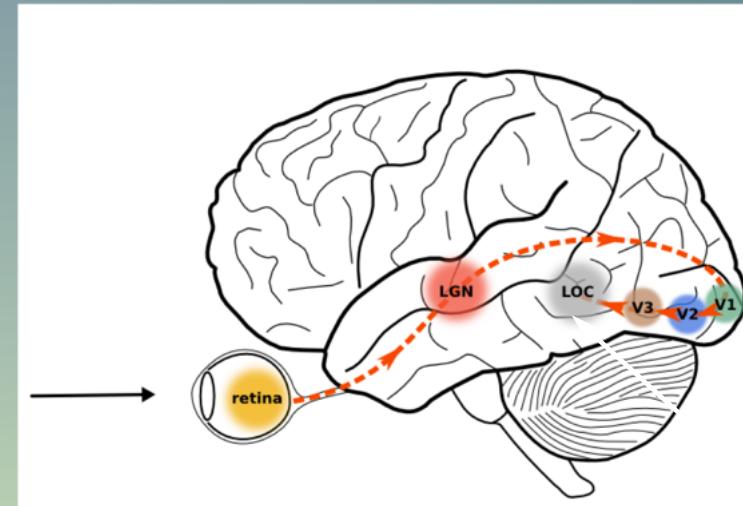


DEEP LEARNING – CONVOLUTIONAL NEURAL NETWORKS

Hubel & Wiesel: Neural basis of visual perception

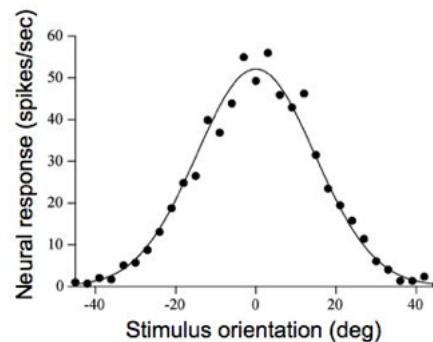
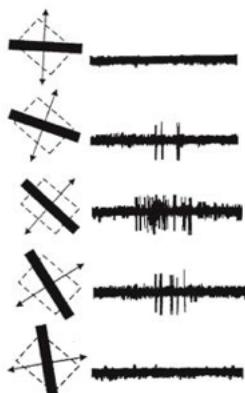
LGN: Lateral Geniculate Nucleus

V1: Primary Visual cortex

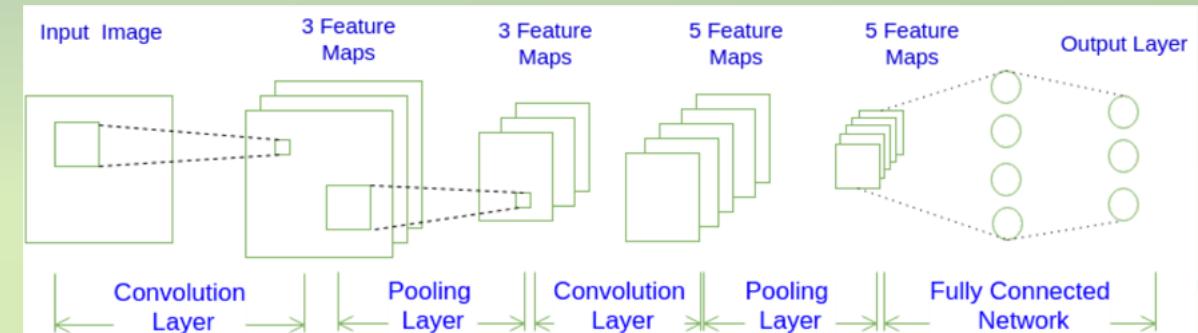


Mug

V1 physiology: orientation selectivity



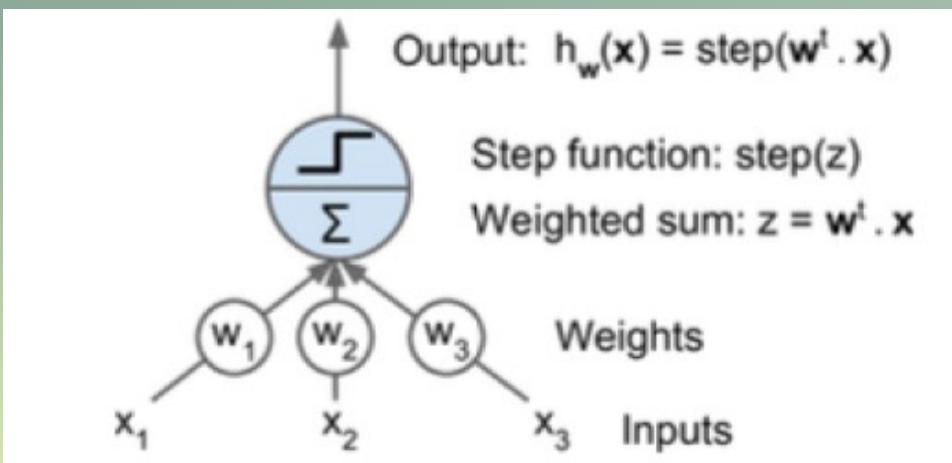
Hubel & Wiesel, 1968



CNN's: 1982 – Now become hot topic due to GPU's (gamers)
SUPERVISED LEARNING OF WEIGHTS

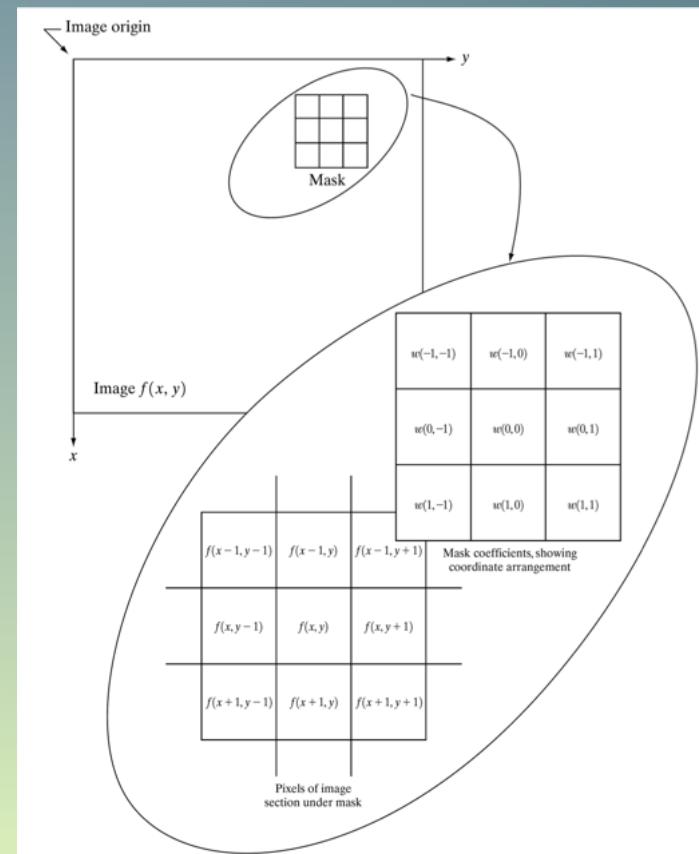
NEURAL NETWORK

Artificial neuron:
Soma, post-synaptic inputs, activation function



Training / learning weights:
Hebbian rule “wiring by firing”

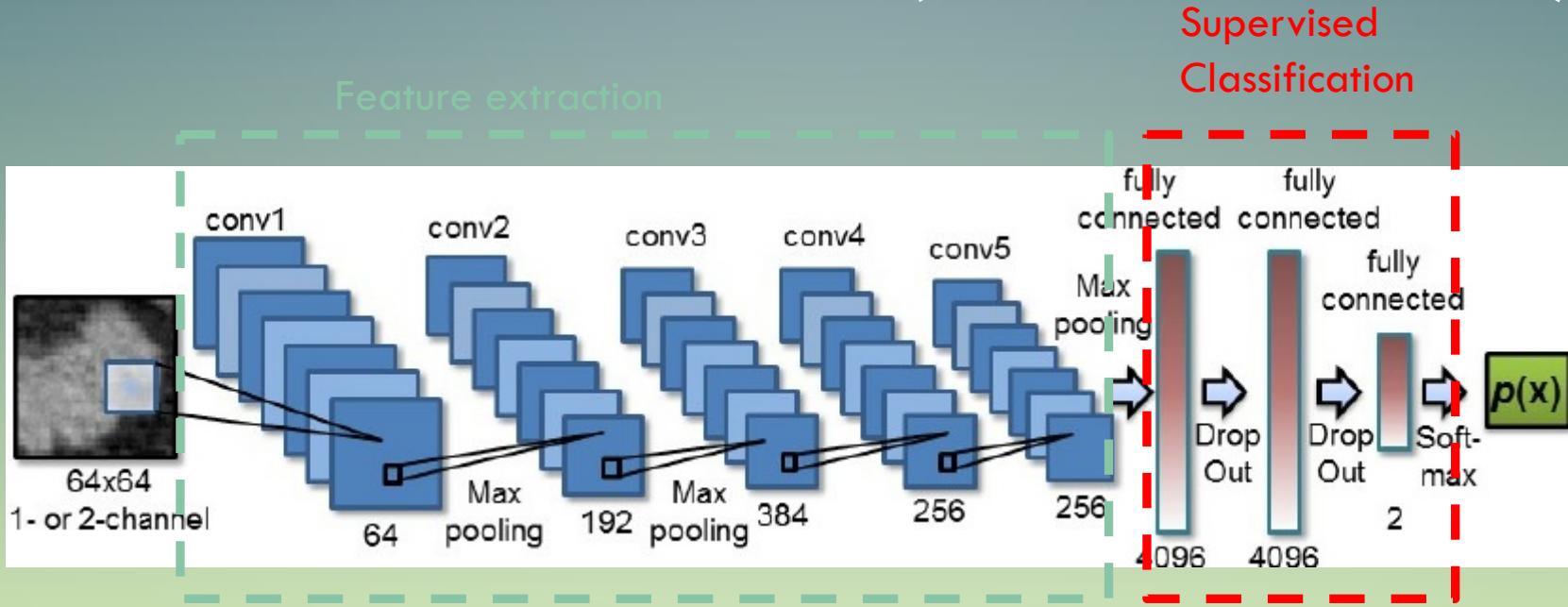
Convolutional filters



$$g(x, y) = \sum_{s=-a}^a \sum_{t=-b}^b w(s, t) \cdot f(x + s, y + t)$$

Mask Image

DL: BLACK-BOX LEARNING ! (DARK-LEARNING)



Learns both:

- Which features are more relevant
- How to classify the images

TRANSFER LEARNING: PRE-TRAINED MODELS

```
from keras.applications.inception_v3 import InceptionV3
from keras.layers import Input

# this could also be the output of a different Keras model or layer
input_tensor = Input(shape=(224, 224, 3)) # this assumes K.image_data_format() == 'channels_last'

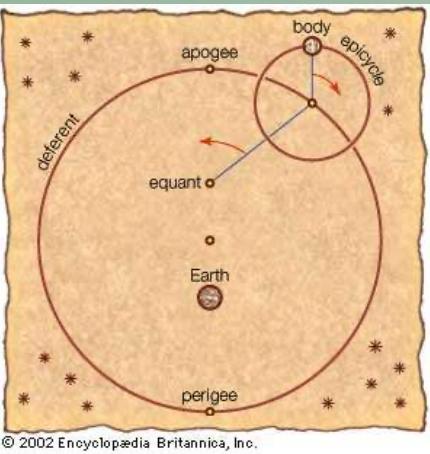
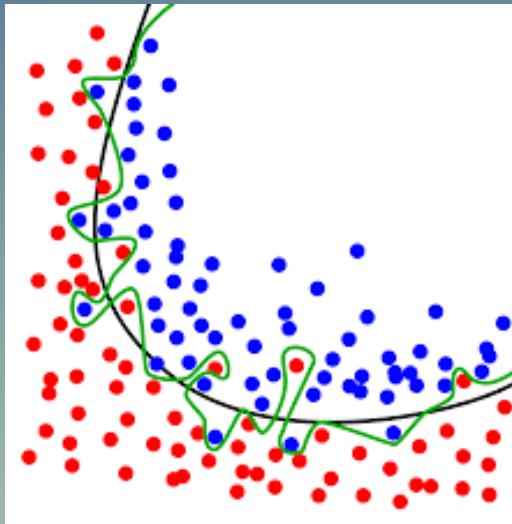
model = InceptionV3(input_tensor=input_tensor, weights='imagenet', include_top=True)
```

Documentation for individual models

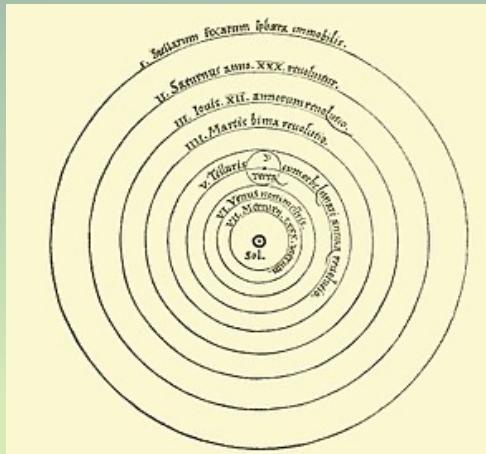
Model	Size	Top-1 Accuracy	Top-5 Accuracy	Parameters	Depth
Xception	88 MB	0.790	0.945	22,910,480	126
VGG16	528 MB	0.715	0.901	138,357,544	23
VGG19	549 MB	0.727	0.910	143,667,240	26
ResNet50	99 MB	0.759	0.929	25,636,712	168
InceptionV3	92 MB	0.788	0.944	23,851,784	159
InceptionResNetV2	215 MB	0.804	0.953	55,873,736	572
MobileNet	17 MB	0.665	0.871	4,253,864	88
DenseNet121	33 MB	0.745	0.918	8,062,504	121
DenseNet169	57 MB	0.759	0.928	14,307,880	169
DenseNet201	80 MB	0.770	0.933	20,242,984	201

C. RISKS

OVERFITTING



data-driven
models



knowledge-driven
models

Model Performance ↑

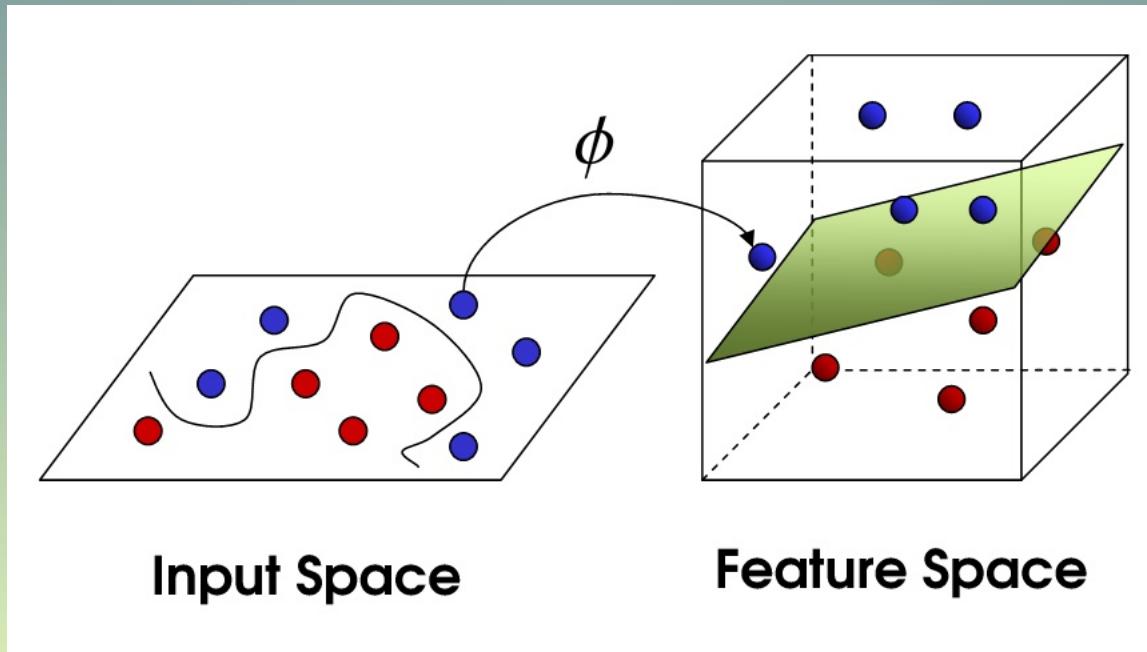


Model Complexity →

Errors due to excess of complexity

Parsimony principle

HUMAN INTERPRETABILITY



Algorithms should be interpretable by humans

OPEN AND CLOSED DOMAIN PROBLEMS



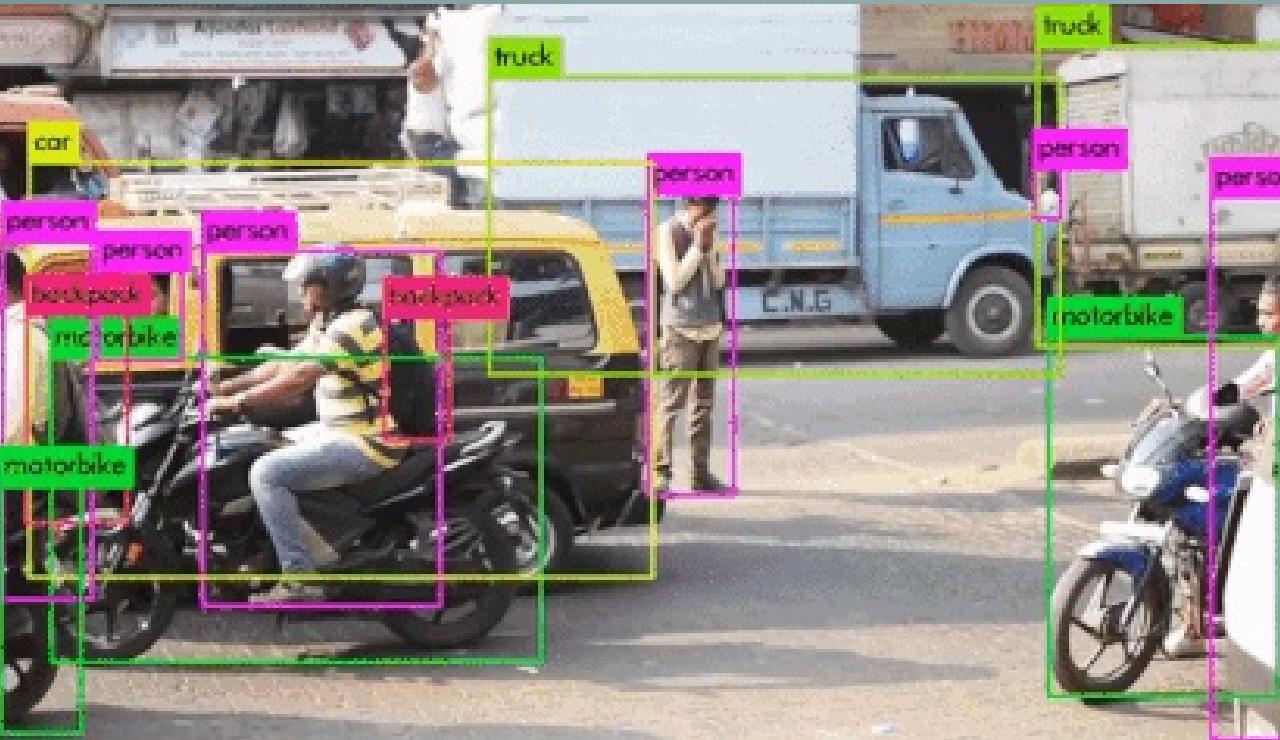
Include open, contextual and dynamic information

WHO IS RESPONSIBLE OF ERRORS?



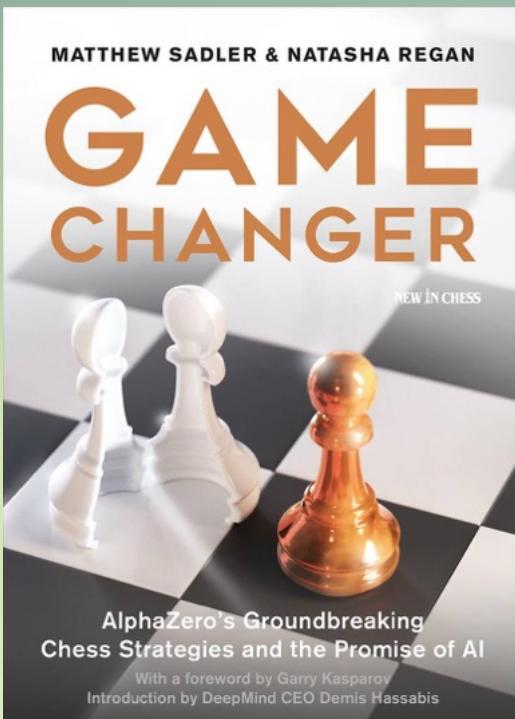
Ethical considerations

YOU ONLY LOOK ONCE (YOLO)



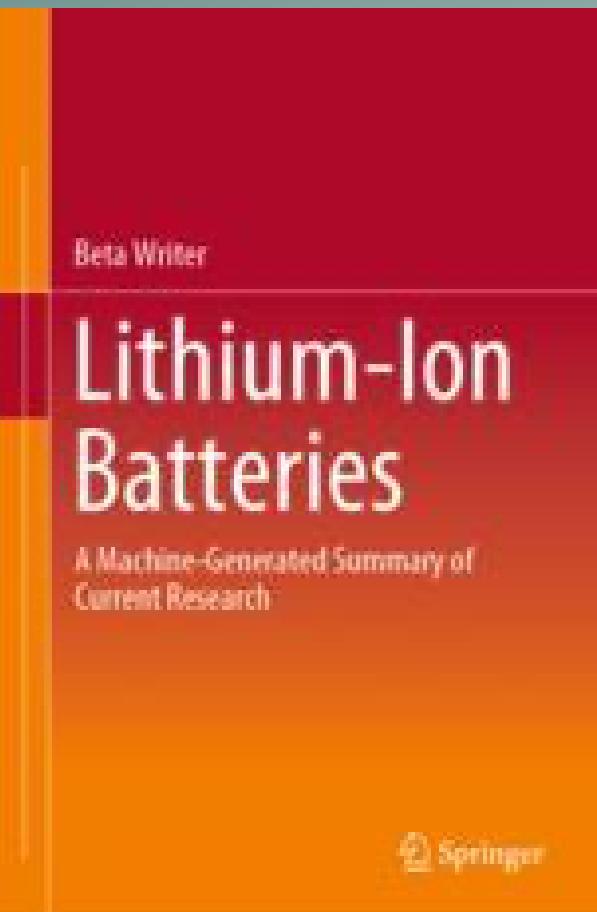
Reduce the costs of computational resources and data

HUMANS WILL BE REPROGRAMMED BY MACHINES



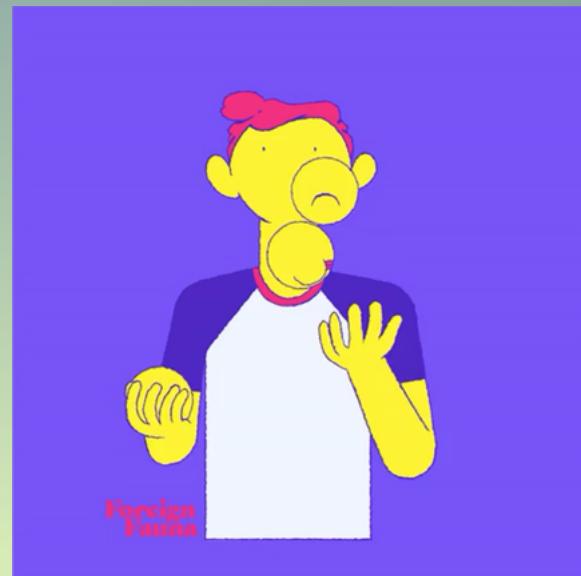
WHAT ELSE?

Creativity?



Lithium-Ion Batteries | © Springer Nature

Empathy?
Pity?
Consciousness?
Respect?
Enthusiasm?



Thanks! Let's get started

