AI for healthcare

Lecture 1/4 – Introduction

Jean Feydy
HeKA team, Inria Paris, Inserm, Université Paris-Cité

Thursday, 2pm–5pm – 4 lectures

Epita, rooms KB404 + SM15

Validation: team project + quizz

Who am I? A short CV

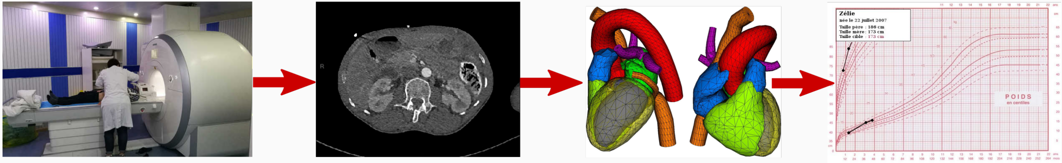
Background in **mathematics** and **data sciences**:

- **2012–2016** ENS Paris, mathematics.
- **2014–2015** M2 mathematics, vision, learning at ENS Cachan.
- **2016–2019** PhD thesis in **medical imaging** with Alain Trouvé at ENS Cachan.
- **2019–2021** **Geometric deep learning** with Michael Bronstein at Imperial College.
- **2021+** **Medical data analysis** in the HeKA INRIA team (Paris).

Close ties with **healthcare**:

- **2015** Image denoising with **Siemens Healthcare** in Princeton.
- **2019+** MasterClass AI–Imaging, for **radiology interns** in the University of Paris.
- **2020+** Colloquium on **Medical imaging in the AI era** at the Paris Brain Institute.

My motivation: medical data analysis



Three main **characteristics**:

- **Heterogeneous data**: patient history, images, etc.
- Small stratified samples: 10 – 1 000 patients per group.
- Dealing with **outliers** and the **heavy tails** of our distributions is a priority.

Two main applications – on large real-life datasets

Computational anatomy. 3D medical scans:

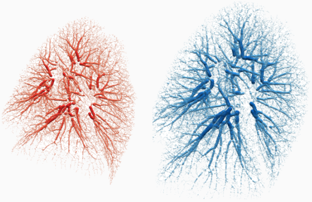
- 100k triangles to represent a brain surface.
- 512x512x512 $\simeq$ 130M voxels for a typical 3D image.

Public health. Over the last decade, medical datasets have **blown up** in size:

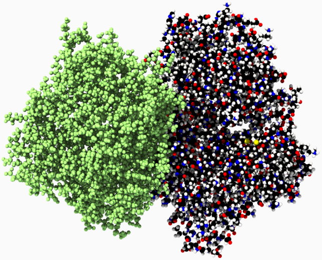
- Clinical trials: **1k patients**, controlled environment.
- UK Biobank: **500k people**, curated data.
- French Health Data Hub: **70M people**, full social security data since ~2000.

Medical doctors, pharmacists and governments need scalable methods.

Some research interests



Optimal transport
for shape registration.



Geometric deep learning
for protein docking.

Survival analysis
for pharmaco-vigilance.

Three points of view on machine learning and AI

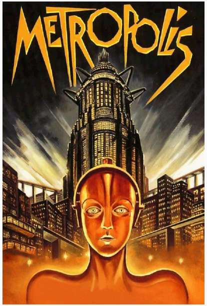
At the intersection of three communities :

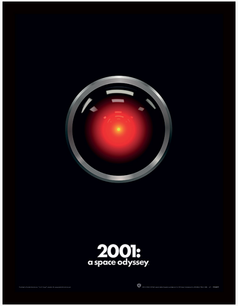
- **AI experts** in Paris, London...
- **Students** at the ENS, the MVA, Epita.
- **Medical doctors** among colleagues, friends and family.

AI in healthcare : massive gap between what we **know**,
what we **hope**,
what we **fear**.

What do **you** think?

“Artificial intelligence” is a misleading term





AI seduces, questions, protects or threatens… **But doesn’t explain much !**

Among experts, researchers always talk about **models**,
discuss their underlying **hypotheses** and study their **properties**.

The aim of this class is to give you a structured perspective on the field.

Objectives of the class

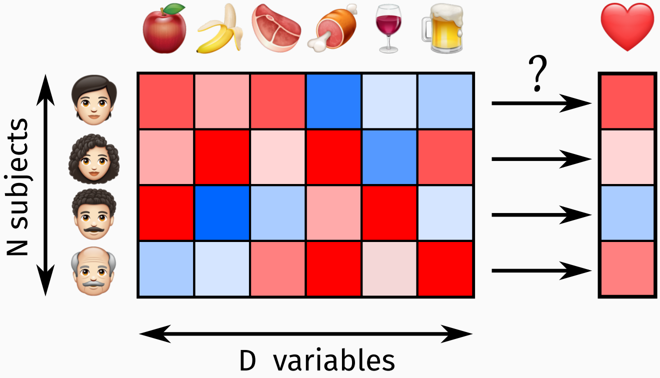
1. Present a **quick overview** of models that you are likely to encounter.
2. Highlight their underlying **hypotheses**, **strengths** and **weaknesses**.
3. Provide you with **clear guidelines** on the use of different tools and theories.
4. **Discuss** the realities of applied machine learning.

Today

1. **AI = model + data:**
    - The curse of dimensionality – or why ML is not “just statistics”.
    - Example: three levels of analysis in anatomy.
2. **How can I choose a good model?**
    - The map is not the territory.
    - Example 1: the sphere of triangles.
    - Example 2: style transfer with convolutional neural networks.
3. **Overview of the class:**
    - What’s coming next?
    - Setup on the computers.

1. AI = model + data

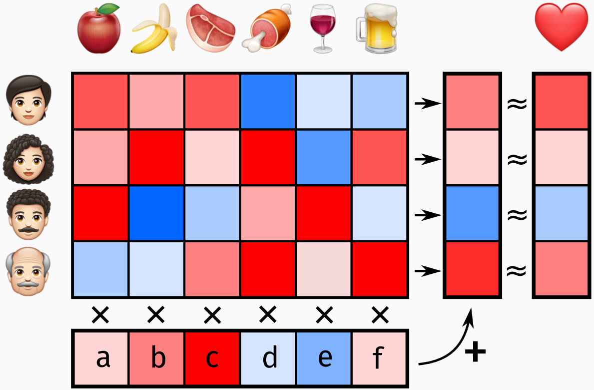
What is a dataset?



Supervised learning = **Regression**.

We look for a formula $F(x_1, \ldots, x_D)$ of the D variables that best approximates an important quantity (♡).

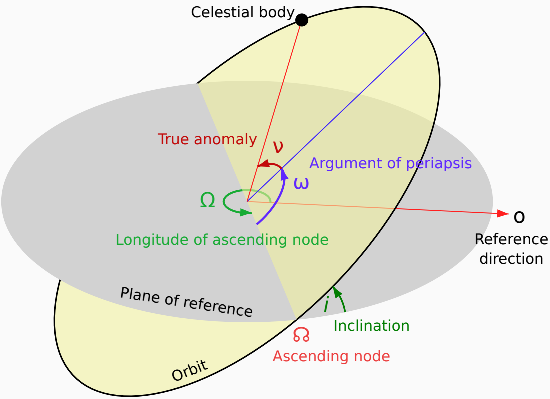
A simple model: linear regression



We choose the weights **a**, **b**, …, **f**

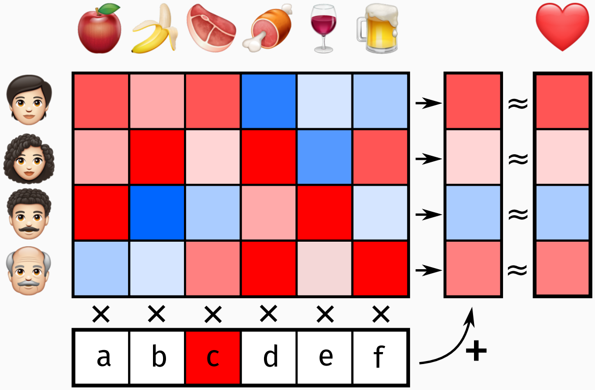
by minimizing a least squares error.

The standard setting of low-dimensional statistics [Las]



First applications to astronomy,
with **hundreds of observations**
on a **handful of variables**.

Problem: medicine isn't XIXth century astronomy

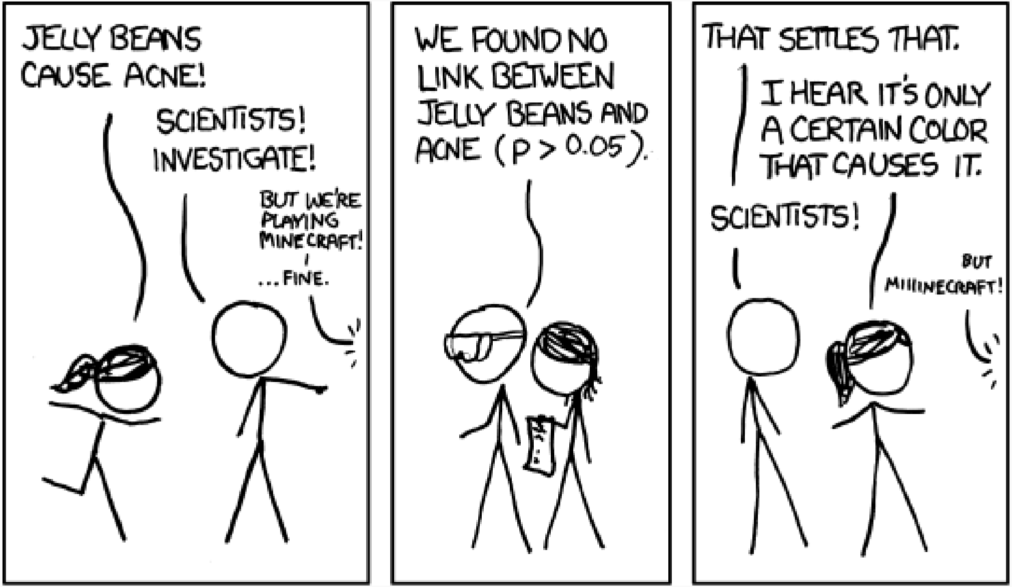


With **lots of information** about **few patients**,

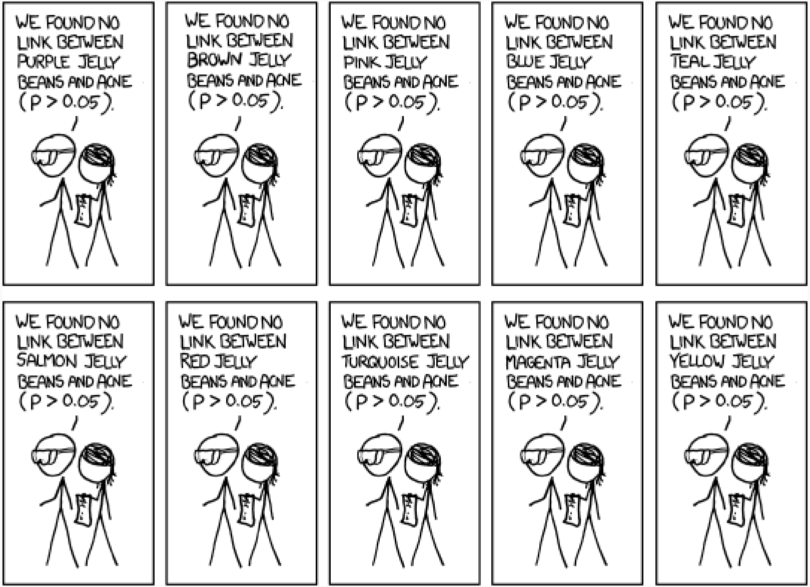
we quickly "discover" spurious correlations.

This is known as **overfitting**.

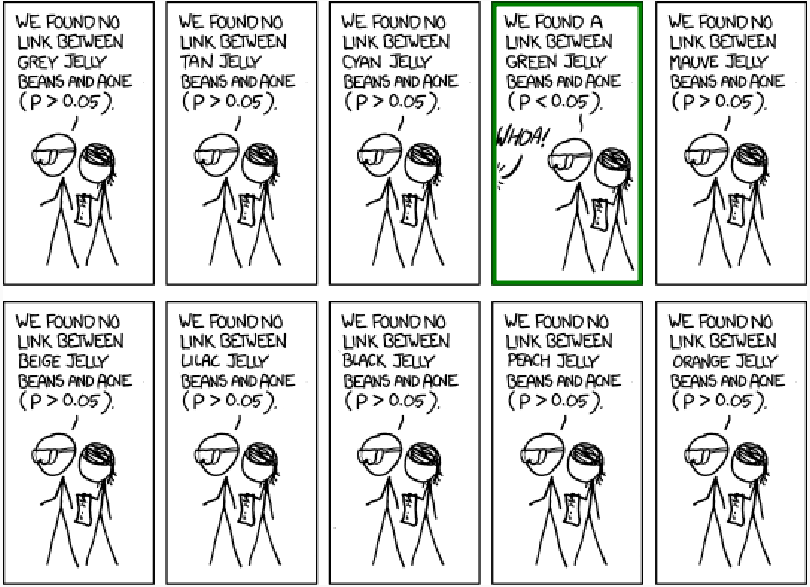
Significant (XKCD 882)



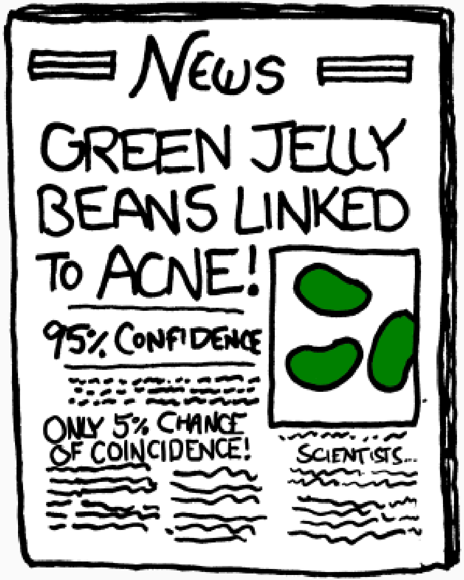
Significant (XKCD 882)



Significant (XKCD 882)



Significant (XKCD 882)



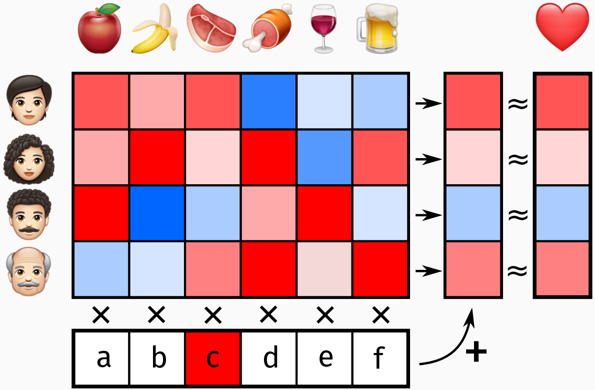
The curse of dimensionality

Having access to **more patients** is usually a **good** thing.
But getting **more information** about each patient is **very dangerous**.

In the previous example: knowing the **color** of the candy
led the (imprudent) scientists to **over-interpret** a random fluctuation.

Machine learning is about doing **reliable** statistics
in this dangerous setting.

We must regularize our decision rules – using sparsity

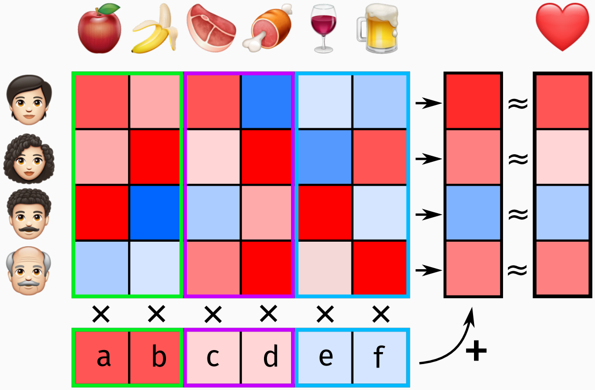


A **sparse** model will select 5 or 10 important columns.

This is useful to handle **tabular data** (XGBoost...)

or **identify sources** in signal processing (Lasso...).

We must regularize our decision rules – using a domain-specific structure



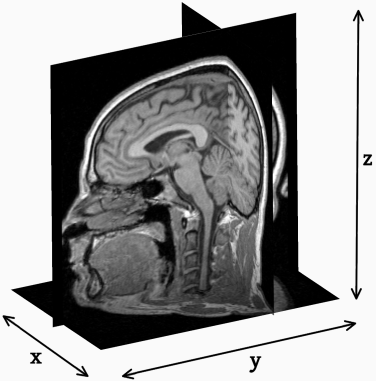
A **structured** model will leverage the **geometry of the data**.

Think about the main **food groups** or the

ATC classification for **medical drugs**.

A first example: medical imaging

A medical image is a massive lump of data



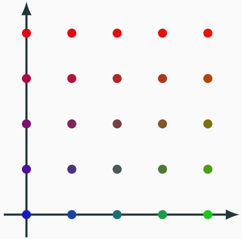
Each pixel is a **column** in our dataset!

We observe **millions to billions of variables**

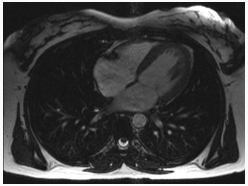
on cohorts of **a few thousand patients**.

Sampling the full space of medical images is impossible

1 number
$\rightarrow$ 5 samples



2 numbers
$\rightarrow 5^2$ samples



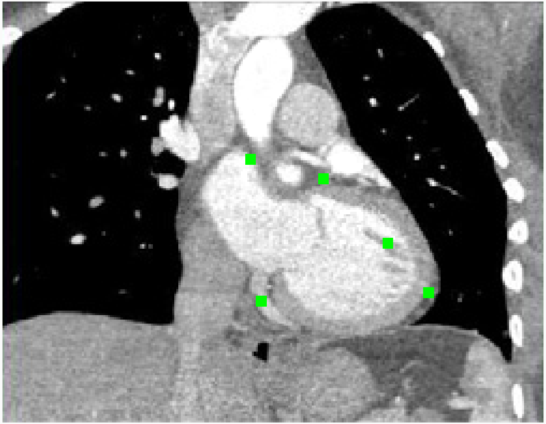
$128 \cdot 128$ numbers
$\rightarrow 5^{128 \cdot 128}$ samples

The set of all 2D/3D images is **way too large**
to be sampled with a satisfying accuracy.

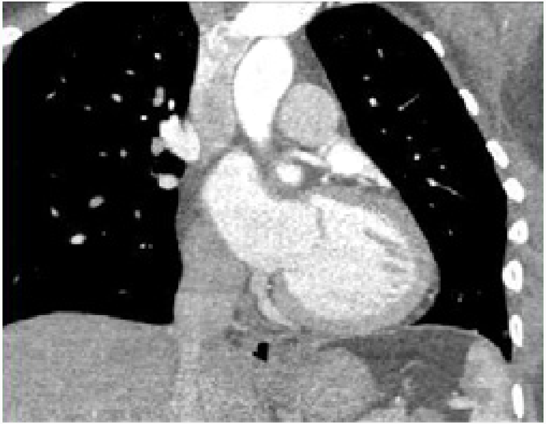
First remark: we cannot rely on sparsity

A good radiology exam does not rely exclusively on **5 or 10 pixels**.

We must learn how to **group pixels** in relevant bundles.

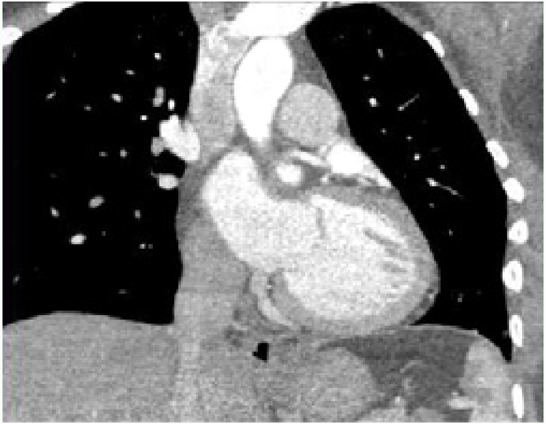


What do you see on a chest image? [EPW11, Man11]



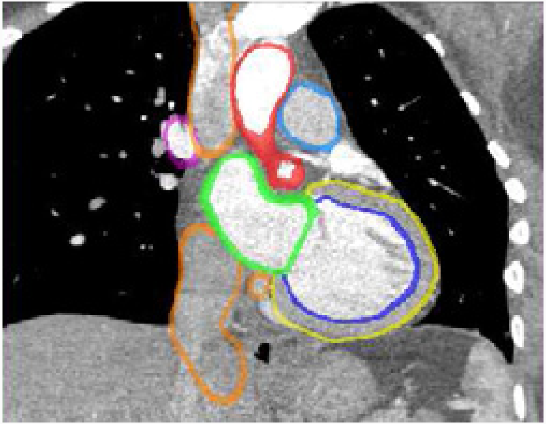
What do you see on a chest image? [EPW11, Man11]

1. Pixels



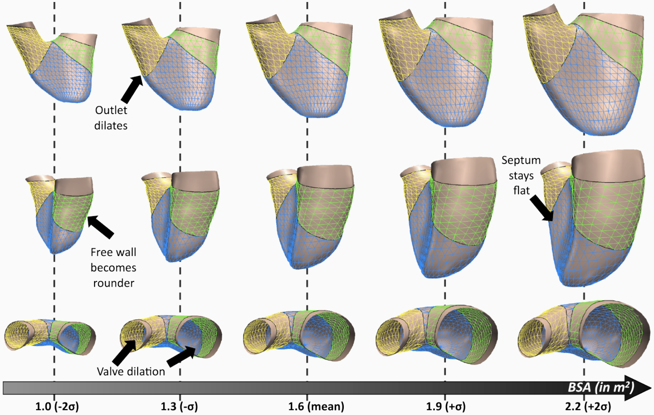
What do you see on a chest image? [EPW11, Man11]

1. Pixels
2. Anatomy



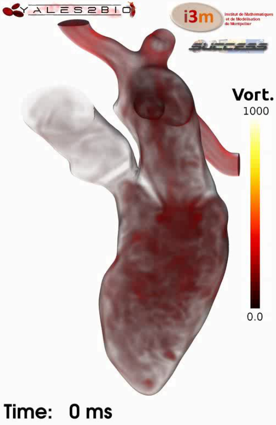
What do you see on a chest image? [EPW11, Man11]

1. Pixels
2. Anatomy



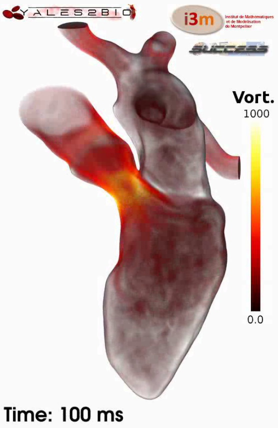
What do you see on a chest image? [EPW11, Man11]

1. Pixels 2. Anatomy 3. Function



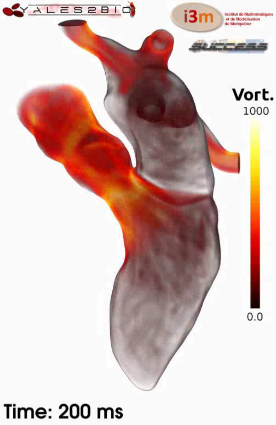
What do you see on a chest image? [EPW11, Man11]

1. Pixels 2. Anatomy 3. Function



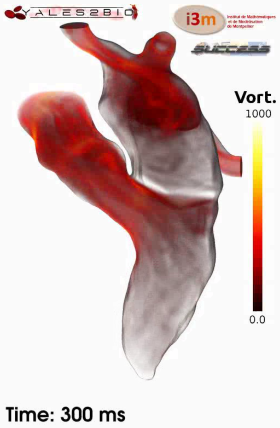
What do you see on a chest image? [EPW11, Man11]

1. Pixels
2. Anatomy
3. Function



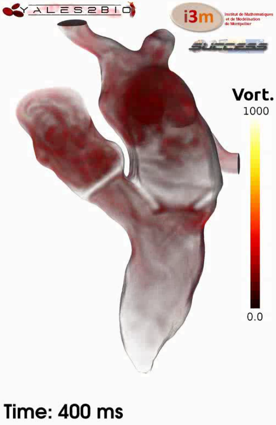
What do you see on a chest image? [EPW11, Man11]

1. Pixels
2. Anatomy
3. Function



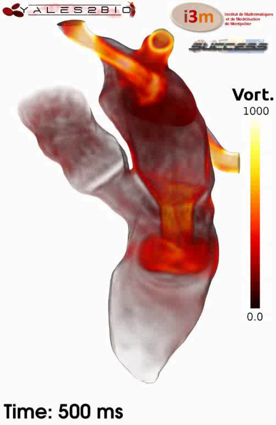
What do you see on a chest image? [EPW11, Man11]

1. Pixels
2. Anatomy
3. Function



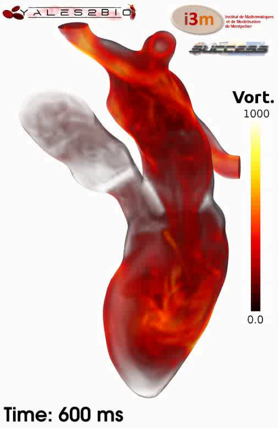
What do you see on a chest image? [EPW11, Man11]

1. Pixels
2. Anatomy
3. Function



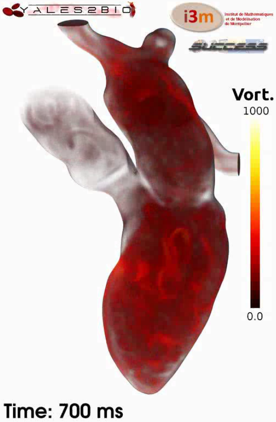
What do you see on a chest image? [EPW11, Man11]

1. Pixels
2. Anatomy
3. Function



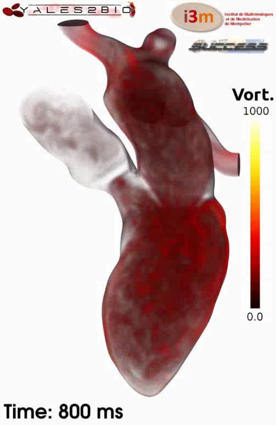
What do you see on a chest image? [EPW11, Man11]

1. Pixels 2. Anatomy 3. Function



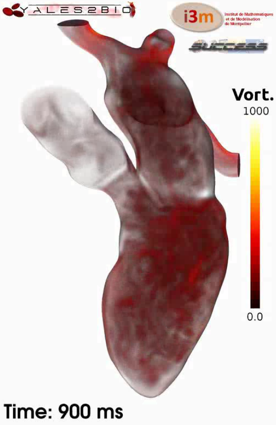
What do you see on a chest image? [EPW11, Man11]

1. Pixels 2. Anatomy 3. Function



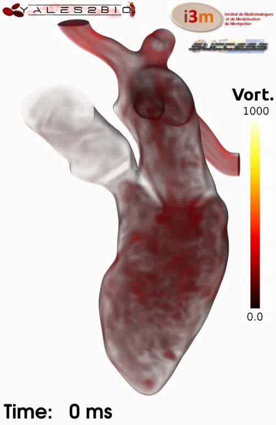
What do you see on a chest image? [EPW11, Man11]

1. Pixels 2. Anatomy 3. Function



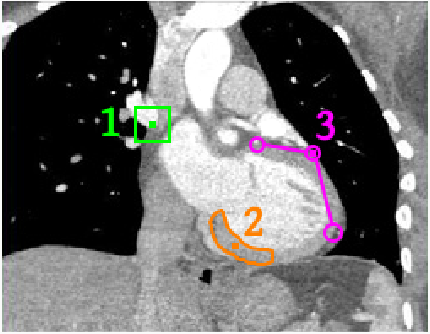
What do you see on a chest image? [EPW11, Man11]

1. Pixels
2. Anatomy
3. Function



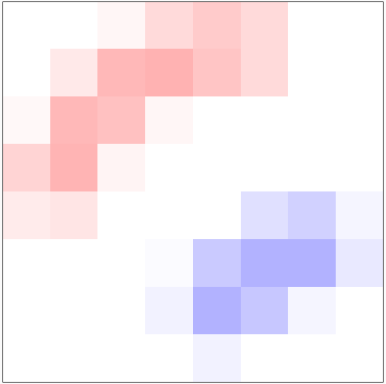
What do you see on a chest image? [EPW11, Man11]

1. Pixels   2. Anatomy   3. Function



Simplifying a bit, each level of analysis corresponds to a way of **grouping pixels** with their neighbors.

1st level: a pixel grid



$N_x \times N_y \times N_z$ array of pixels.

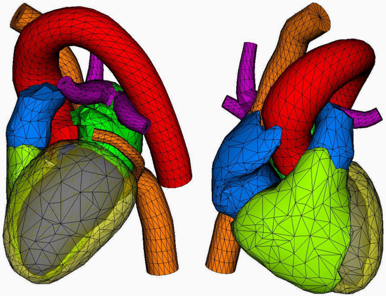
Bitmap images and volumes:

- `.bmp`, `.png`, `.jpg`
- Standard in **radiology**.

\+ Ordered memory structure.
\+ Explicit neighborhoods.
\+ Fast **convolutions**.

→ **Texture** analysis.
→ Organ **segmentation**.
→ Pattern **detection**.

2nd level: point clouds and 3D surfaces [EPW11]



$N_{points} \times 3$ array of (x, y, z) coordinates.

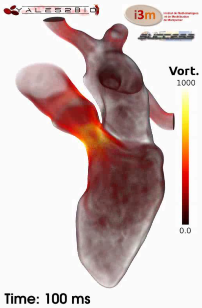
Clouds of points ($\pm$ triangles):

- `.svg`
- Standard for **video games**.

$+$ Compact representation.
$+$ High precision geometry.
$+$ **Easy to deform**.

$\rightarrow$ **3D visualization**.
$\rightarrow$ Anatomical **atlas**.
$\rightarrow$ **Shape** analysis.

3rd level: biomechanical and/or physiological model [Man11]



Volumetric mesh,
graph of interactions.

Mechanical/biological model:

- Finite elements, networks.
- Standard for **CAD**.

+ Prior **knowledge**.
+ **Robust** to noise.
+ **Realistic** behaviour.

→ **Physiological** interpretation.
→ **Infer** what cannot be seen (blood flow).
→ **Simulate** a surgery.

To summarize

We must combine a **statistical regression** method with a **relevant model**.

In medical imaging, we may work with:

1. A 2D or 3D **pixel grid**.
2. An array of (x, y, z) **coordinates**.
3. A **web** of complex interactions.
4. Everything at once!

In most cases, we will define a large **structured formula**:

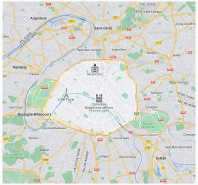
$$\text{image} \xrightarrow{\text{F}} \text{F}\,(\text{image}) \simeq \text{diagnostic}$$

F is a parametric computing **architecture**
$\simeq$ **model** to fit $\simeq$ **network** to train.

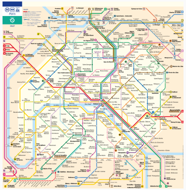
2. How can I choose a good model?

A model is like a map: a warped and partial view of the world [Duk, Str]

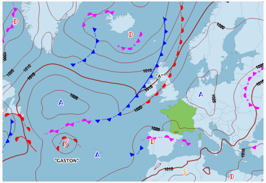
How can I trust these pictures?



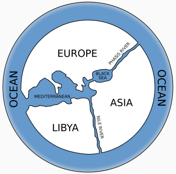
Google



RATP

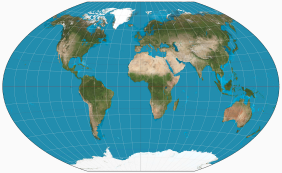


Météo France



Anaximander

Mercator



Winkel

The map is not the territory

A map is not the territory it represents, but, if correct,
it has a **similar structure** to the territory,
which accounts for its **usefulness**.

– Alfred Korzybski, 1933.

On exactitude in science – Jorge Luis Borges, 1946, translated by Andrew Hurley.

…In that empire, the art of cartography attained such **perfection** that the map of a single **province** occupied the entirety of a **city**, and the map of the **empire**, the entirety of a **province**. In time, those unconscionable maps no longer satisfied, and the cartographers guilds struck **a map of the empire whose size was that of the empire**, and which coincided point for point with it.

The following generations, who were not so fond of the study of cartography as their forebears had been, saw that **that vast map was useless**, and not without some pitilessness was it, that they delivered it up to the inclemencies of sun and winters. In the **deserts** of the West, still today, there are tattered **ruins of that map**, inhabited by animals and beggars; in all the land there is no other relic of the disciplines of geography.

– Suarez Miranda, *Viajes de varones prudentes*, Libro IV,Cap. XLV, Lerida, 1658

What is a good model?

A good map should:

- **Highlight** the relevant key points and roads.
  This is a **task-specific** objective (car, bike...).
- **Hide** unnecessary information to reduce clutter: **the lighter, the better**.
  Heavy maps *will* be discarded by the next generation.
- Be **accurate** – up to a required **tolerance**.
  There is a **tradeoff** here: think of the metro map!
- Be **transparent** about **omissions and distortions**.
  This is the main **trap** that we should not forget.

What is a good model?

All these points apply to ML models:

- **Highlight** the stuff that matters.
- **Discard** the rest.
- Be **accurate** – up to a sensible tolerance.
- Be **transparent** and **honest**.

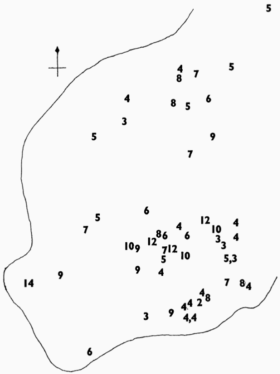
Of course, raw "performance" results do matter: **accuracy** is a real thing.

But most importantly, a good model should be **legible** and enable **creativity**.

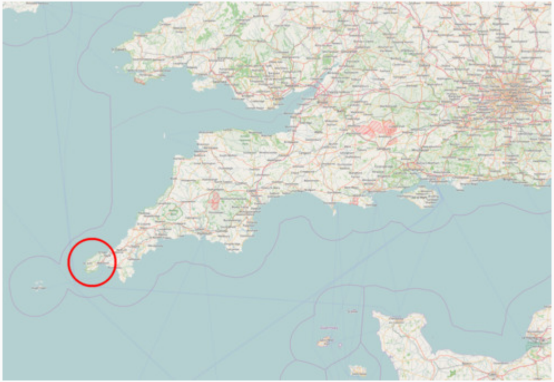
Example 1: The sphere of triangles

Surprisingly enough, our story starts with… Menhirs!

More precisely: with the distribution of megaliths in the Land’s End peninsula



52 Menhir locations.



Cornwall, in South-West England.

Can you see **alignments** here? Some people do.

Many authors have claimed that these **ley lines** demarcate “Earth energies” and/or serve as guides for alien spacecraft.

Understanding triangle shapes

Back in 1974, this problem motivated David Kendall to ask a question:

Assuming that I draw 52 points at random in a square…

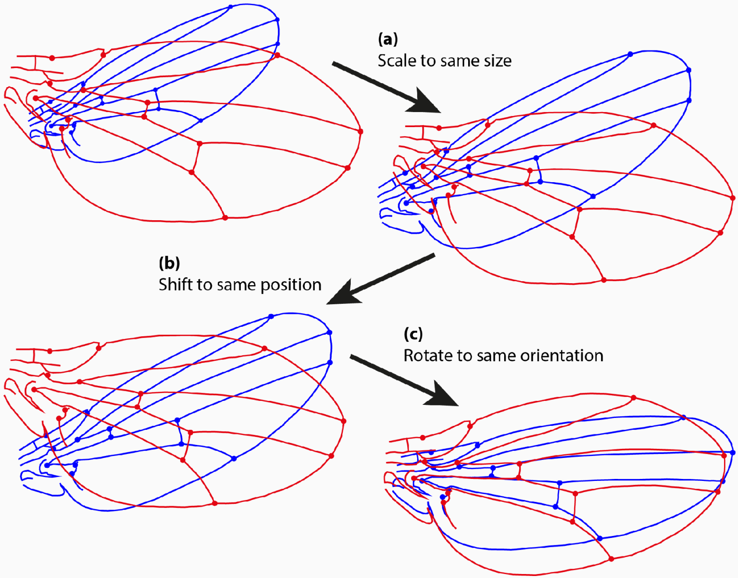
How many **flat triangles** (say, with a $180° \pm 1°$ angle) am I going to observe?

This prompted a remarkable series of papers:

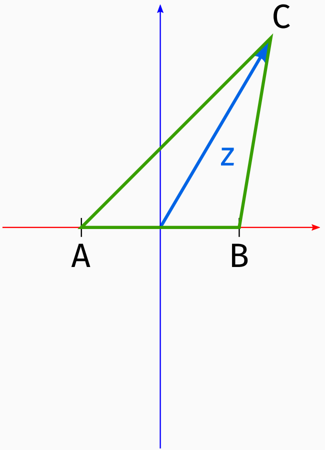
- *The diffusion of shape*, Kendall, 1977.
- *Alignments in two-dimensional **random sets of points***, Kendall and Kendall, 1980.
- *Simulating the **ley hunter***, Broadbent, 1980.
- *Shape manifolds, Procrustean metrics, and complex projective spaces*, Kendall, 1984.

And the the birth of modern shape analysis.

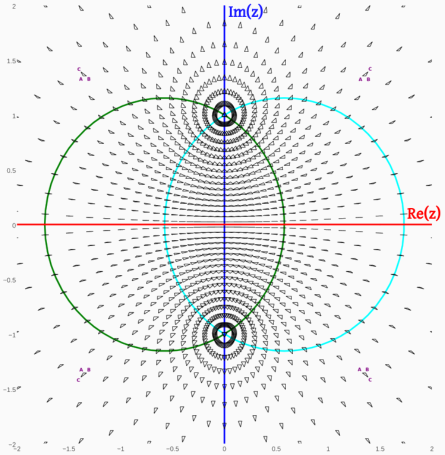
Step 1: Working with shapes up to similarities [Kli15]



Step 2: The space of triangles up to similarities is two-dimensional

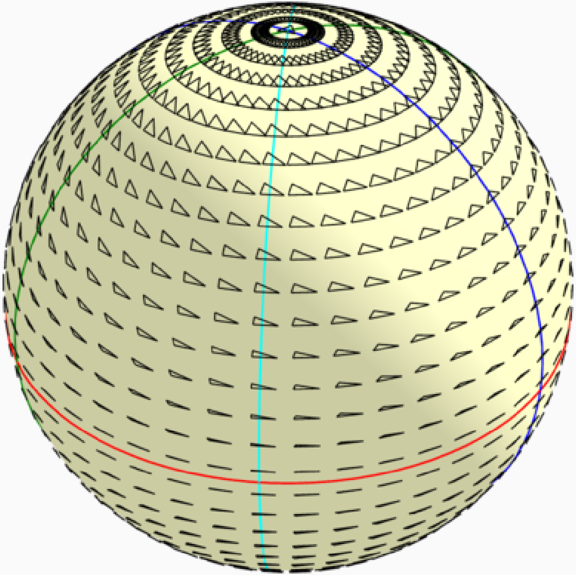


Send A to (-1, 0)
and B to (+1, 0).

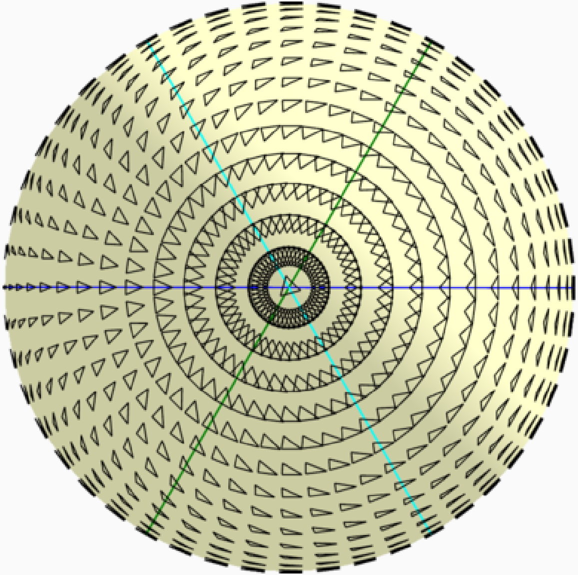


Identify $z \in \mathbb{C} \cup \{\infty\}$
with all non-degenerate triangle shapes.

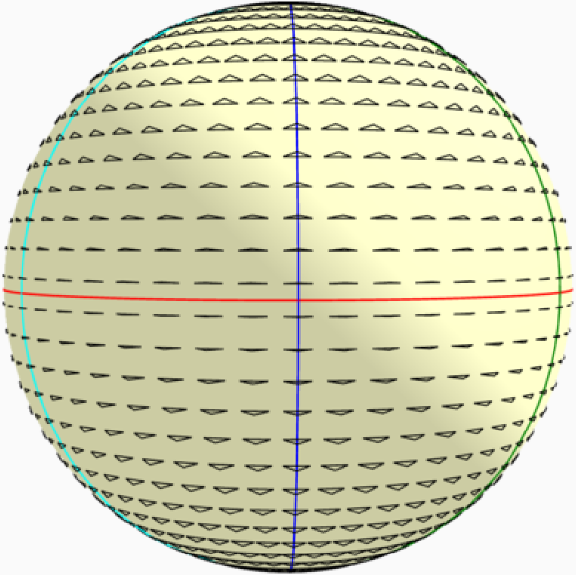
Step 3: Up to a clever change of coordinates: this is actually a sphere!



The two poles correspond to the direct and indirect equilateral triangles



The Equator corresponds to the set of flat triangles



First properties of this map

This representation respects the main **symmetries** of the set of triangles:

- The sets of **isoceles triangles** with respect to A, B and C correspond to three **great circles** that are equally spaced with each other.
- **Axial symmetries** correspond to a North-South inversion across the Equator.
- The Equator of flat triangles + the meridians of isoceles triangles cut the sphere in **12 pieces**. These exactly correspond to the 6 permutations of the vertices ABC $\times$ $\{$ the identity or an axial symmetry $\}$.

But there is more!

Metric properties of the spherical embedding

$$\mathbf{K} : (A, B, C) \in \mathbb{R}^{3\times 2} \backslash \{A = B = C\} \;\mapsto\; \mathbf{K}(A, B, C) \in \mathbb{R}^3$$

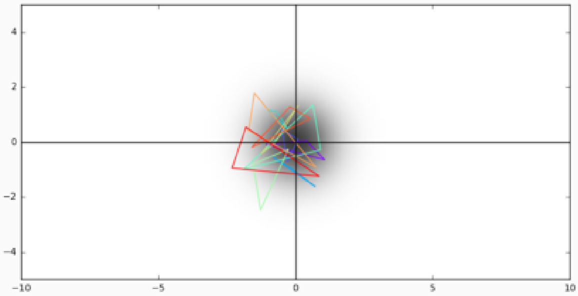
denotes the **Kendall embedding** from the set of non-degenerate triangles to the sphere of center $(0, 0, 0)$ and diameter 1. (It has an OK-ish expression using cos and sin.)

Then, straightforward computations show that:

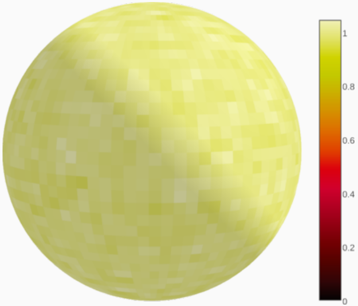
$$\min_{\text{similarity } S} \|S(A)-D\|^2_{\mathbb{R}^2} + \|S(B)-E\|^2_{\mathbb{R}^2} + \|S(C)-F\|^2_{\mathbb{R}^2} \;=\; \mathrm{Var}(D, E, F) \cdot \|\mathbf{K}(A, B, C) - \mathbf{K}(D, E, F)\|^2_{\mathbb{R}^3}$$

The **chord distance on the sphere** of Kendall corresponds to the **Euclidean distance** on triplets of points in the plane, **up to similarities**.

Statistical properties of the spherical embedding

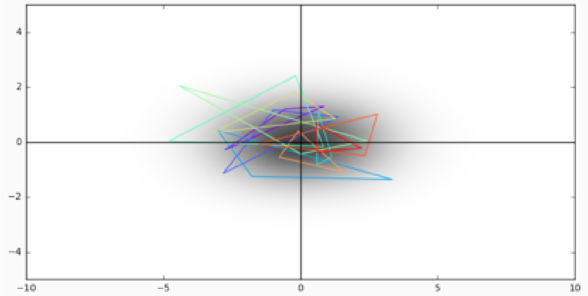


A, B, C are drawn according to an **isotropic** Gaussian distribution on the plane.

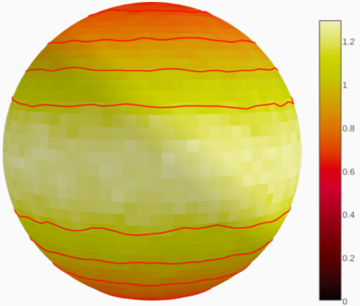


Empirical histogram on the sphere of triangle shapes.

Statistical properties of the spherical embedding

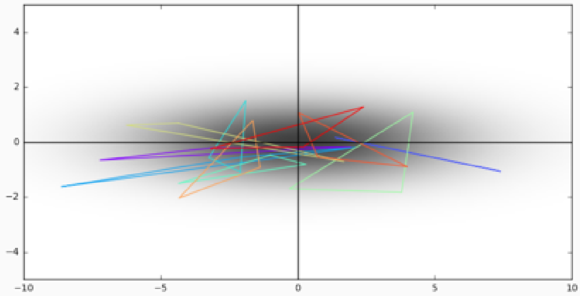


A, B, C are drawn according to a **non-isotropic** Gaussian distribution on the plane.

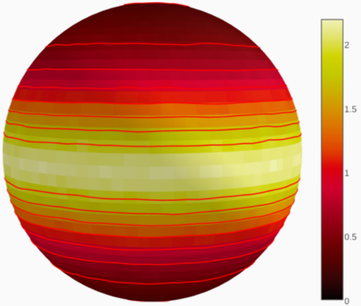


Empirical histogram on the sphere of triangle shapes.

Statistical properties of the spherical embedding



A, B, C are drawn according to a **non-isotropic** Gaussian distribution on the plane.



Empirical histogram on the sphere of triangle shapes.

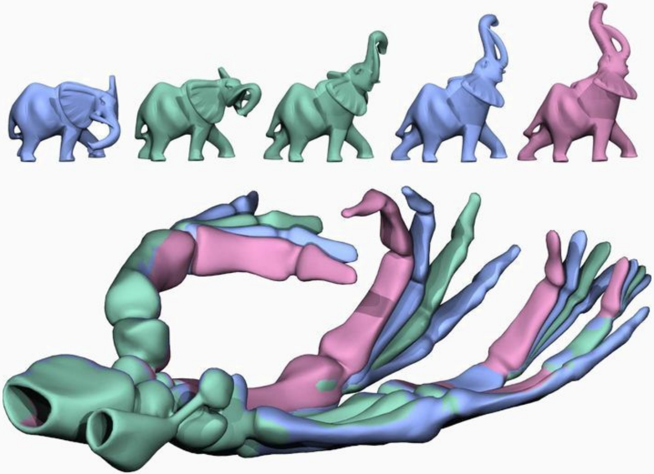
Summary

Kendall showed that the space of **triangles** is best understood as a **sphere** for **topological**, **geometric** and **statistical** reasons.

You cannot “unsee” this elegant result.

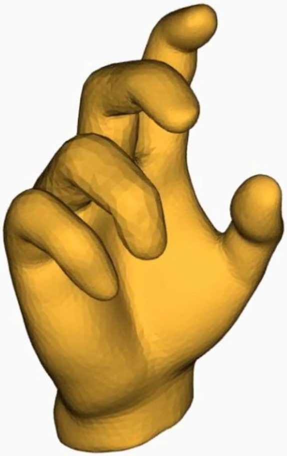
Most importantly, his theorem showed that **shapes** naturally belong to a **curved** geometric space.

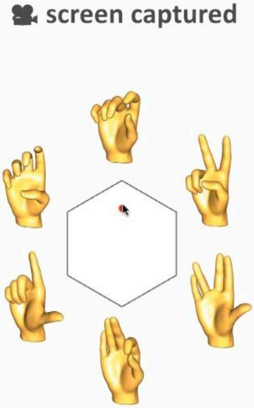
This idea is at the heart of modern shape analysis software [KMP07]



Geodesics in spaces of elephants and skeletons.

This idea is at the heart of modern shape analysis software [vRESH16]



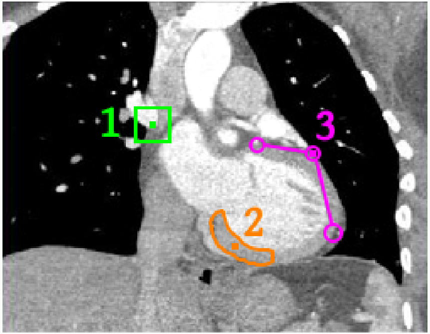


Barycentric interpolation in a space of hands.

Example 2: Style transfer with convolutional neural networks

Remember that picture? [EPW11]

1. Pixels
2. Anatomy
3. Function



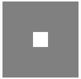
Let's talk about the **first way** of **grouping pixels** with their neighbors.

Filtering, also known as the “convolution product”

Convolution (i.e. weighted average of the neighboring pixels) :

Cheap generalization of the **product** “$a \cdot x$”,

parameterized by the coefficients of a **small filter** φ.



$\star$

$=$

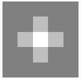
φ $\quad$ x $\quad$ $\varphi \star x$

Filtering, also known as the "convolution product"

Convolution (i.e. weighted average of the neighboring pixels) :

Cheap generalization of the **product** "a · x",

parameterized by the coefficients of a **small filter** φ.

 $\star$  $=$ 

φ x $\varphi \star x$

Filtering, also known as the “convolution product”

Convolution (i.e. weighted average of the neighboring pixels) :

Cheap generalization of the **product** “a · x”,

parameterized by the coefficients of a **small filter** φ.

φ $\quad$ x $\quad$ $\varphi \star x$

Filtering, also known as the “convolution product”

Convolution (i.e. weighted average of the neighboring pixels) :

Cheap generalization of the **product** “a · x”,
parameterized by the coefficients of a **small filter** φ.

φ x $\varphi \star x$

Filtering, also known as the "convolution product"

Convolution (i.e. weighted average of the neighboring pixels) :

Cheap generalization of the **product** "a · x",
parameterized by the coefficients of a **small filter** φ.

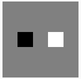
φ x $\varphi \star x$

Filtering, also known as the “convolution product”

Convolution (i.e. weighted average of the neighboring pixels) :

Cheap generalization of the **product** “$a \cdot x$”,

parameterized by the coefficients of a **small filter** φ.



$\star$

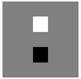
$=$

φ $\quad$ x $\quad$ $|\varphi \star x|$

Convolution (i.e. weighted average of the neighboring pixels) :

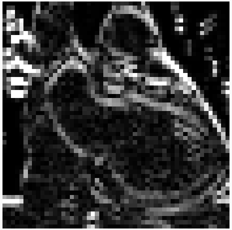
Cheap generalization of the **product** “$a \cdot x$”,

parameterized by the coefficients of a **small filter** φ.



$\star$

$=$



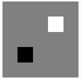
φ x $|\varphi \star x|$

Filtering, also known as the "convolution product"

Convolution (i.e. weighted average of the neighboring pixels) :

Cheap generalization of the **product** "$a \cdot x$",

parameterized by the coefficients of a **small filter** φ.



$\star$

$=$

φ $\quad$ x $\quad$ $|\varphi \star x|$

A multi-scale prior on images

Wavelet theory (1990∼2010 ; Meyer, Mallat, Daubechies...) :

Small filters + cascading zoom-out operations [Mal16]:

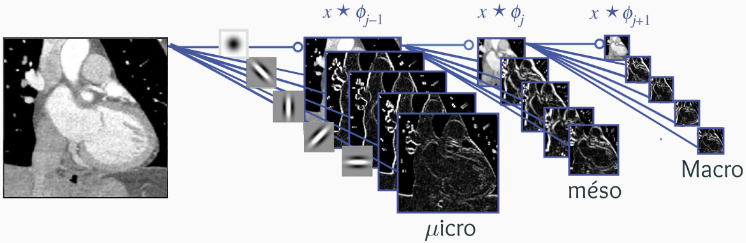
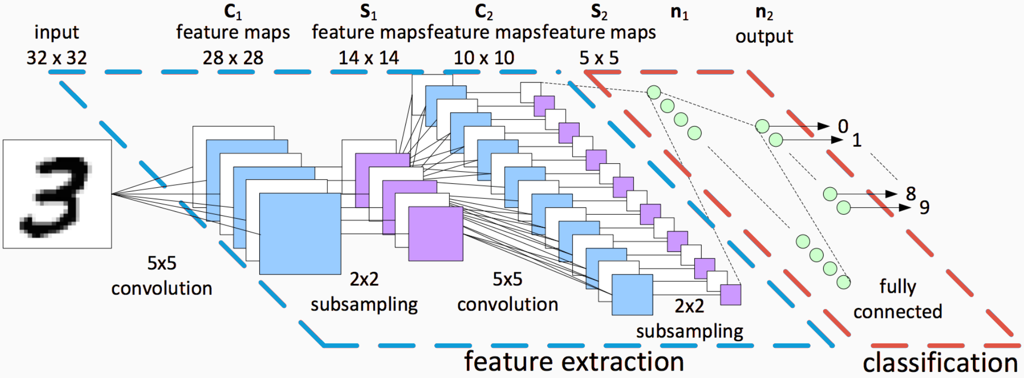


Image	$\longrightarrow$	Relevant coefficients
$\simeq$ ".wav" Audio	$\longrightarrow$	Music score

$\Longrightarrow$ **JPEG2000** format, standard of the movie industry.

Convolutional neural networks [PMC11]

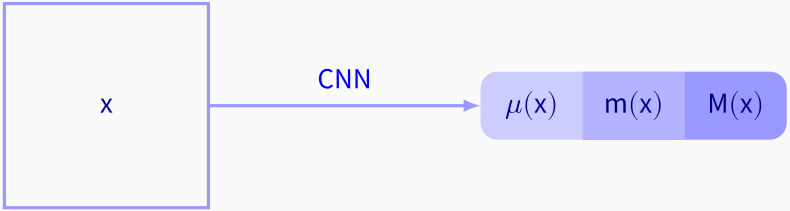


Convolutional Neural Networks as a data-driven "codec" for your data

JPEG2000 relies on a model for natural images that is:

- Computationally cheap.
- Translation-equivariant.
- Encodes a **multi-scale** prior on natural images.

By **tuning its parameters** on a labeled database,
we get a **CNN** = domain-specific "JPEG2020".

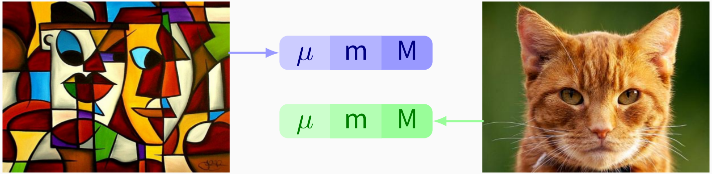


An iconic application: Deep Art [NN16]

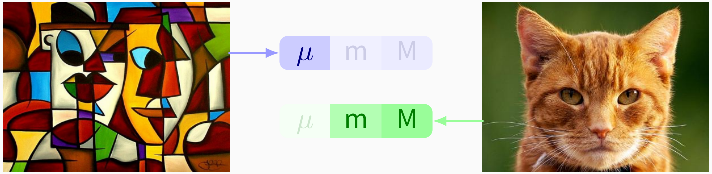
An iconic application: Deep Art [NN16]

An iconic application: Deep Art [NN16]

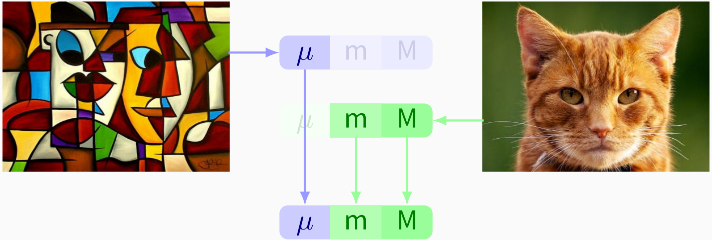
An iconic application: Deep Art [NN16]



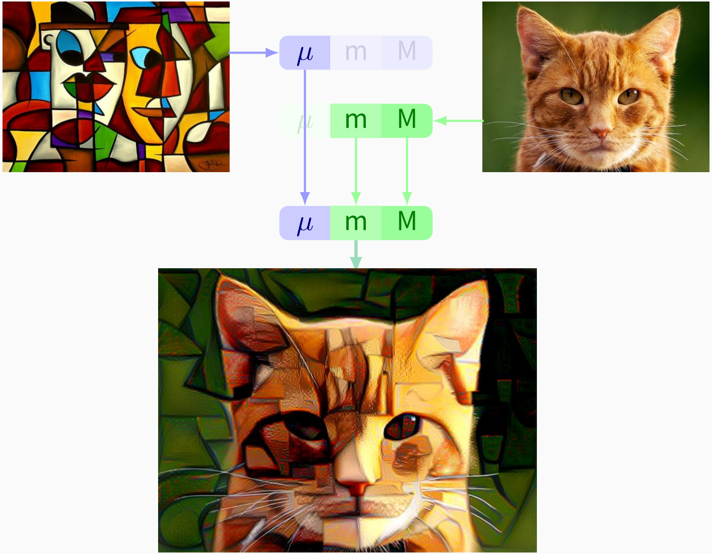
An iconic application: Deep Art [NN16]



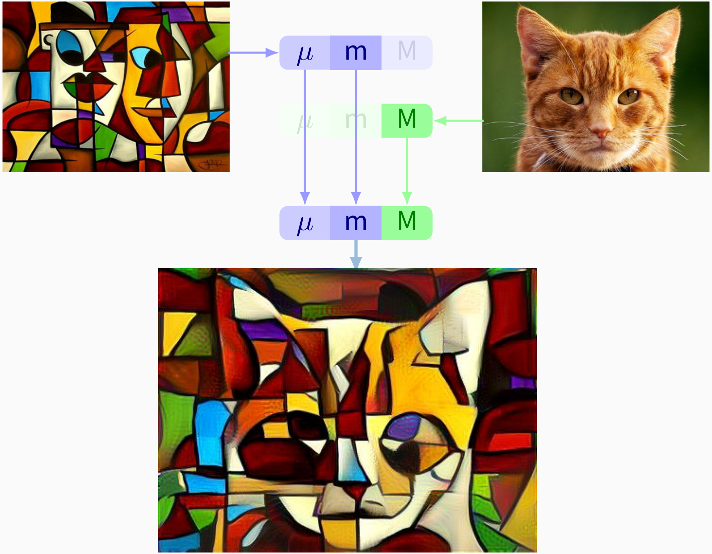
An iconic application: Deep Art [NN16]



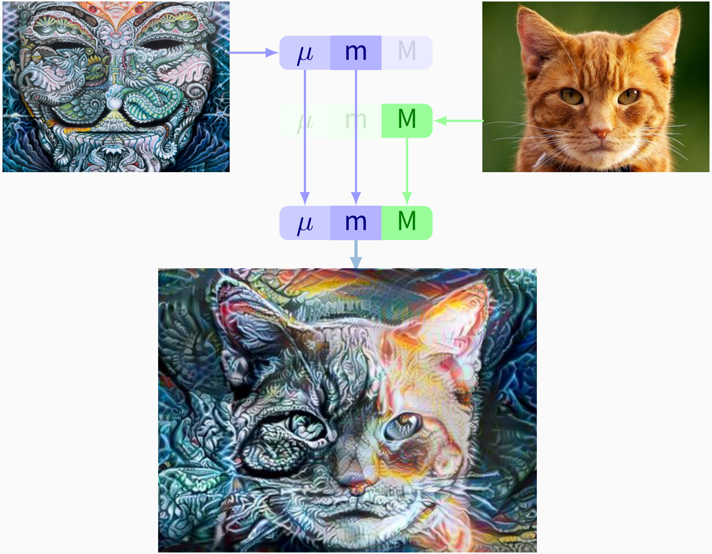
An iconic application: Deep Art [NN16]



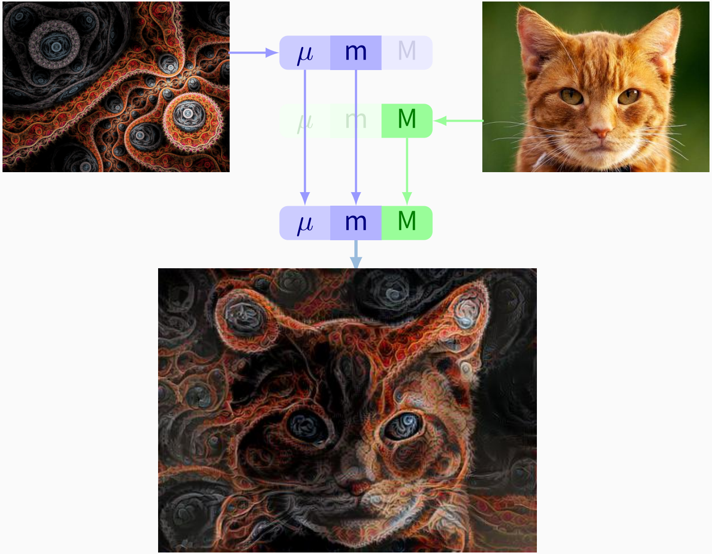
An iconic application: Deep Art [NN16]



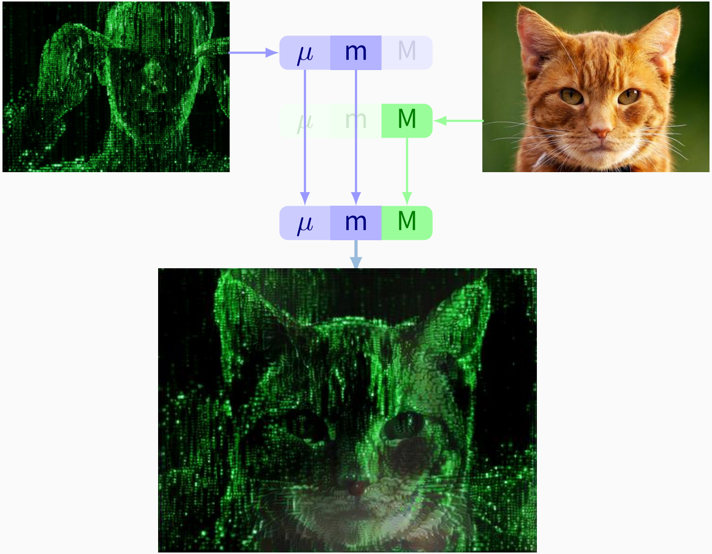
An iconic application: Deep Art [NN16]



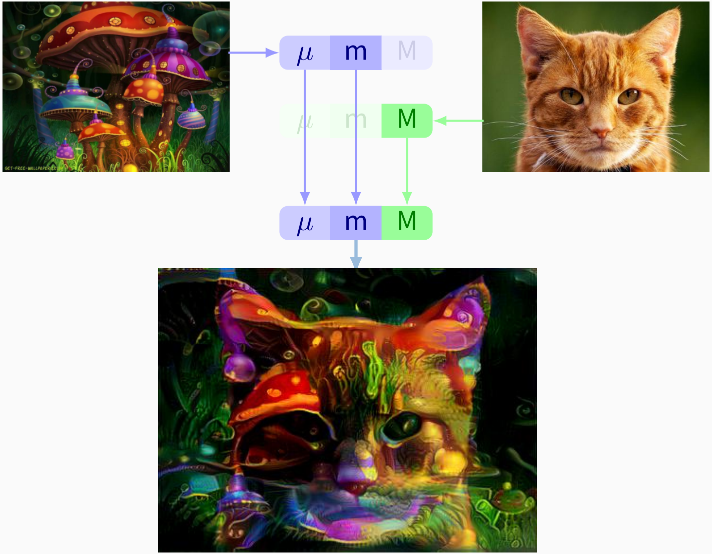
An iconic application: Deep Art [NN16]



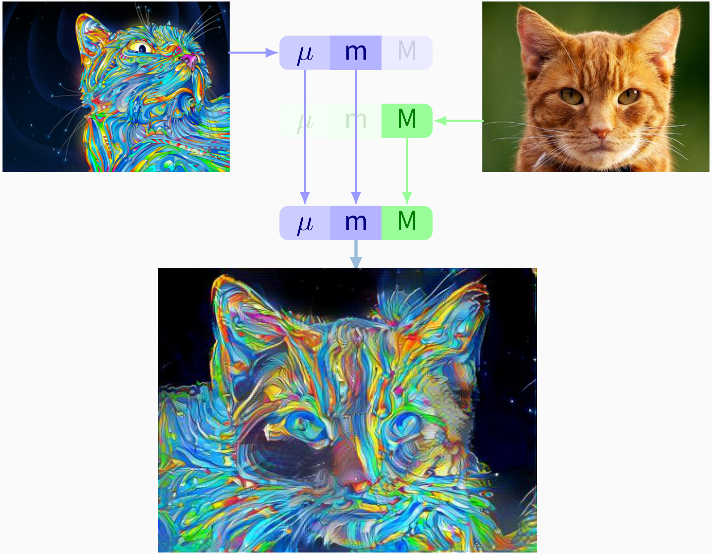
An iconic application: Deep Art [NN16]



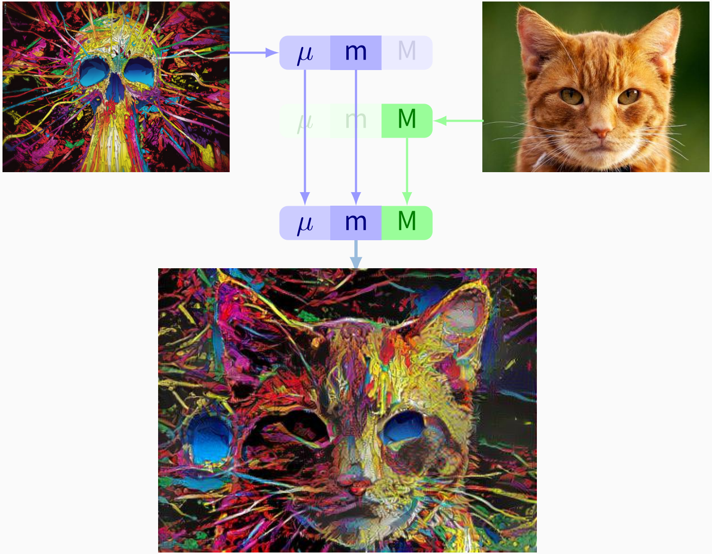
An iconic application: Deep Art [NN16]



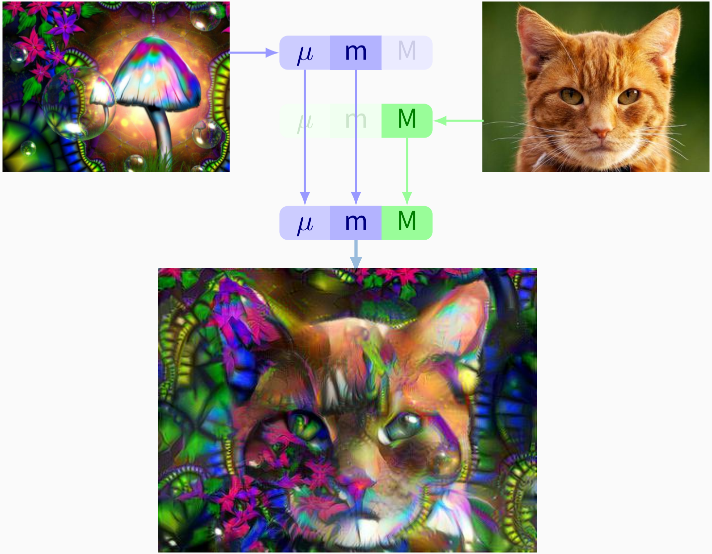
An iconic application: Deep Art [NN16]



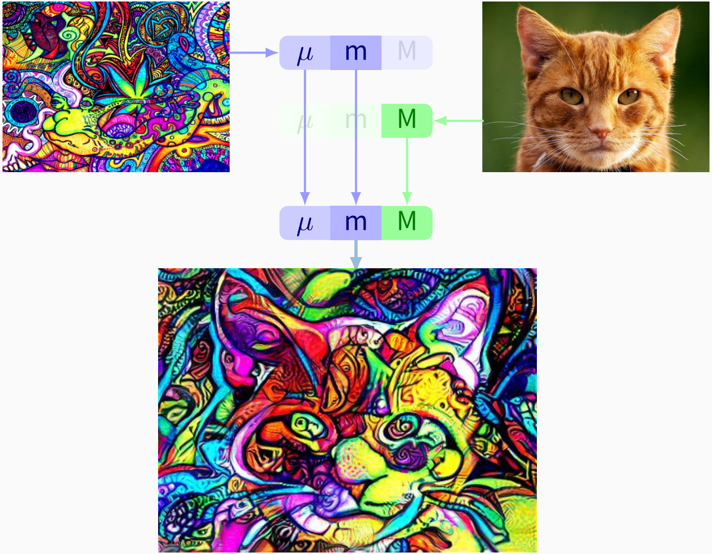
An iconic application: Deep Art [NN16]



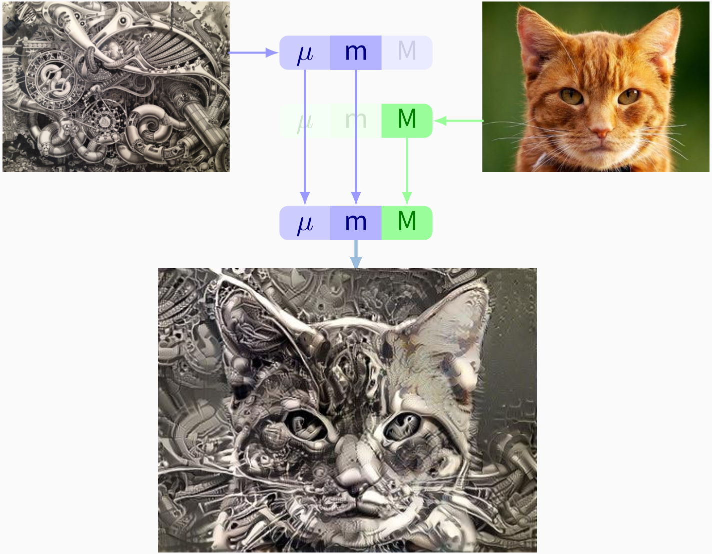
An iconic application: Deep Art [NN16]



An iconic application: Deep Art [NN16]



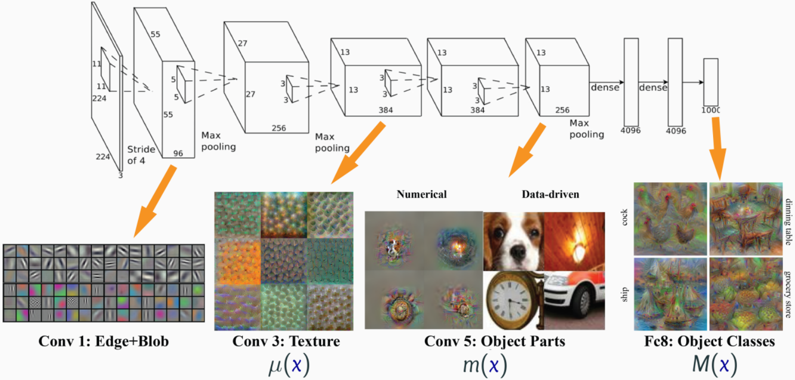
An iconic application: Deep Art [NN16]



The dream application: image classification [WZTF17]

Looking at $\mathrm{CNN}(x) = [\,\mu(x)\,,\; m(x)\,,\; M(x)\,]$,
can we **distinguish** seagulls from pandas?

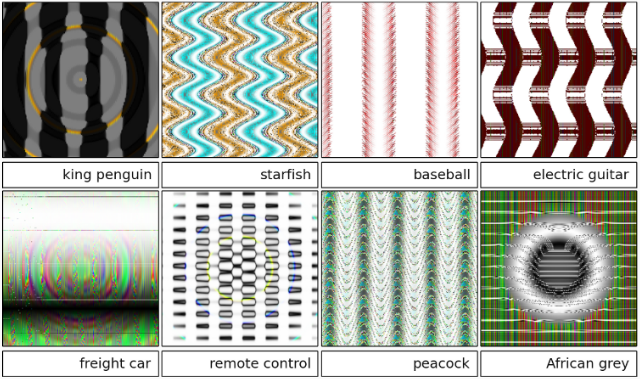
What researchers have in mind:



The limits of multiscale filtering [NYC15]

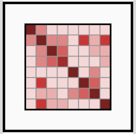
Standard CNNs perform **pattern detection** – little more, little less:

« $\mu(\mathbf{x})$ is reliable ; M$(\mathbf{x})$ really isn't. »



Overview of the class

A geometric perspective on data sciences

Domain-specific observations on a population of N subjects	N-by-N matrix of similarities	General machine learning methods
MRI/CT images		Clustering (K-Means...)
Cognitive scores		Classification (hierarchy...)
Blood samples		Regression (kernels...)
Drug consumption history		Visualization (UMAP...)

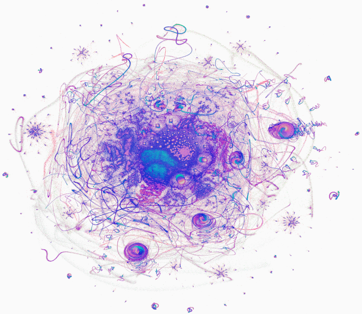
This class is about understanding **similarity metrics**.

What are the implicit **priors** that they reflect?

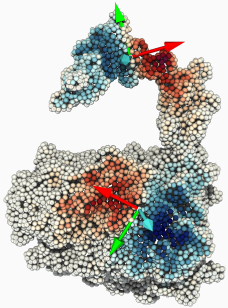
How can we manipulate them **efficiently**?

Overview of the class [Wil]

Vectors, linear models
trees and kernels.



Graphs, curvature
and embeddings.



Deep learning: convolutions,
geometry and attention.

References

References i

Dukesy68.

Man scratching head.

Wikipedia, CC BY-SA v4.0.

Olivier Ecabert, Jochen Peters, and Matthew Walker.

Segmentation of the heart and great vessels in ct images using a model-based adaptation framework.

Medical Image Analysis, (15):863–876, 2011.

Christian Peter Klingenberg.

Analyzing fluctuating asymmetry with geometric morphometrics: concepts, methods, and applications.

Symmetry, 7(2):843–934, 2015.

Martin Kilian, Niloy J Mitra, and Helmut Pottmann.

Geometric modeling in shape space.

In *ACM Transactions on Graphics (TOG)*, volume 26, page 64. ACM, 2007.

Lasunncty.

Diagram illustrating and explaining various terms in relation to orbits of celestial bodies.

Wikipedia, CC BY-SA v3.0.

Stéphane Mallat.

Understanding deep convolutional networks.

Phil. Trans. R. Soc. A, 374(2065):20150203, 2016.

Tomaso Mansi.

A statistical model for quantification and prediction of cardiac remodelling: Application to tetralogy of fallot.

IEEE transactions on medical imaging, 2011.

Yaroslav Nikulin and Roman Novak.

Exploring the neural algorithm of artistic style.

arXiv preprint arXiv:1602.07188, 2016.

Anh Nguyen, Jason Yosinski, and Jeff Clune.

Deep neural networks are easily fooled: High confidence predictions for unrecognizable images.

In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 427–436, 2015.

Maurice Peemen, Bart Mesman, and Henk Corporaal.

Speed sign detection and recognition by convolutional neural networks.

In *Proceedings of the 8th International Automotive Congress*, pages 162–170, 2011.

Strebe.

Earth maps.

Wikipedia, CC BY-SA v3.0.

Philipp von Radziewsky, Elmar Eisemann, Hans-Peter Seidel, and Klaus Hildebrandt.

Optimized subspaces for deformation-based modeling and shape interpolation.

Computers & Graphics, 58:128–138, 2016.

John Williamson.

What do numbers look like?

https://johnhw.github.io/umap_primes/index.md.html.

Donglai Wei, Bolei Zhou, Antonio Torralba, and William T Freeman.

mNeuron: A Matlab plugin to visualize neurons from deep models.

Massachusetts Institute of Technology, 2017.