Machine Learning and Physics: A Faustian Bargain?

Jeff Byers, NRL/UniPd

Machine Learning at GGI

September 14, 2022

ML: DALL-E 2/OpenAI Text2Image Generator

INPUT_STRING="A photo of physicists discussing machine learning in Florence Italy"

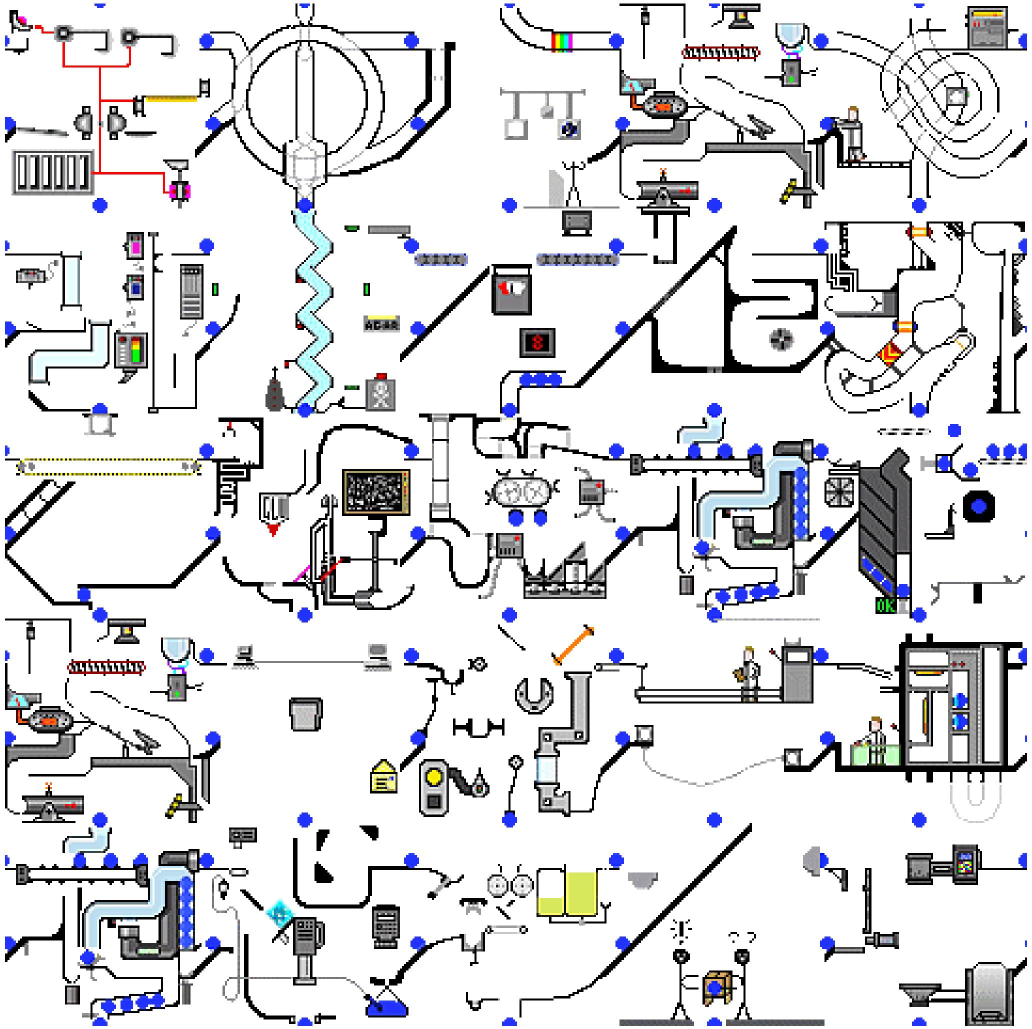
ML: DALL-E 2/OpenAI Text2Image Generator

`INPUT_STRING=`“Impressionist painting of physicists discussing machine learning in Florence Italy”

ML: DALL-E 2/OpenAI Text2Image Generator

INPUT_STRING="Dali painting of physicists discussing machine learning in a bowl of pasta"

Is Machine Learning a Faustian Bargain?



Is Machine Learning a Faustian Bargain?

https://arxiv.org/pdf/2204.06125.pdf

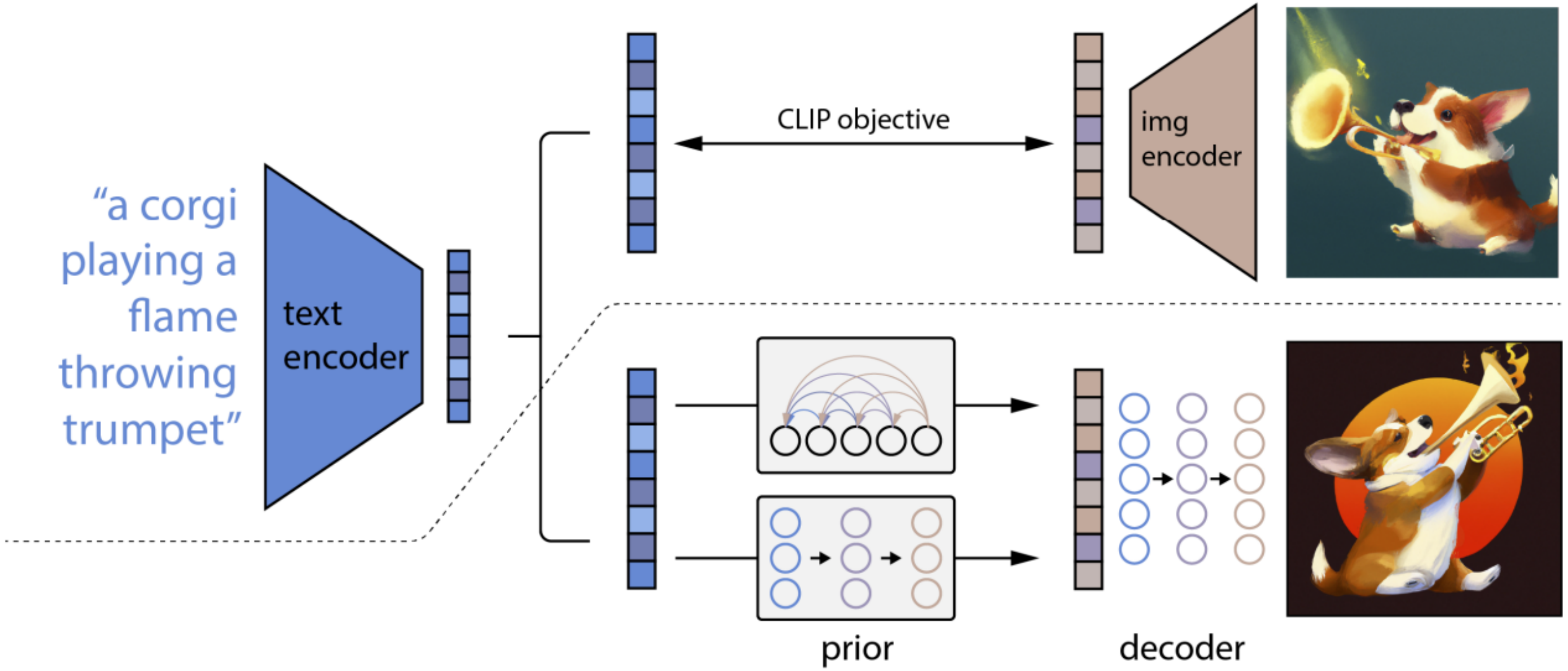


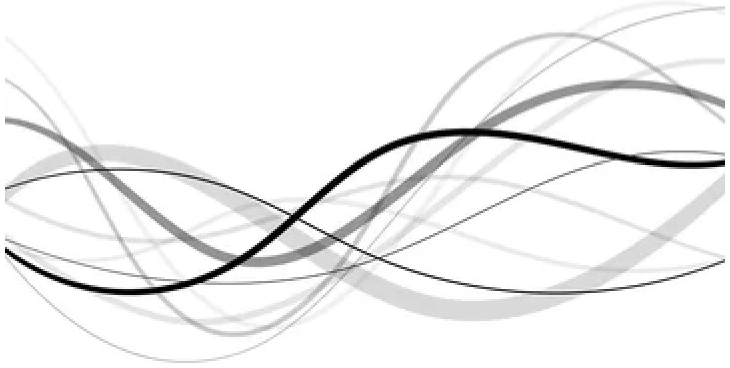
Figure 2: A high-level overview of unCLIP. Above the dotted line, we depict the CLIP training process, through which we learn a joint representation space for text and images. Below the dotted line, we depict our text-to-image generation process: a CLIP text embedding is first fed to an autoregressive or diffusion prior to produce an image embedding, and then this embedding is used to condition a diffusion decoder which produces a final image. Note that the CLIP model is frozen during training of the prior and decoder.

Is Machine Learning a Faustian Bargain?

"Algebra is the offer made by the devil to the mathematicians. The devil says: I will give you this powerful machine, it will answer any question you like. All you need to do is give me your soul: give up geometry and you will have this marvelous machine." – Michael Atiyah

GEOMETRY

"curves"



ALGEBRA

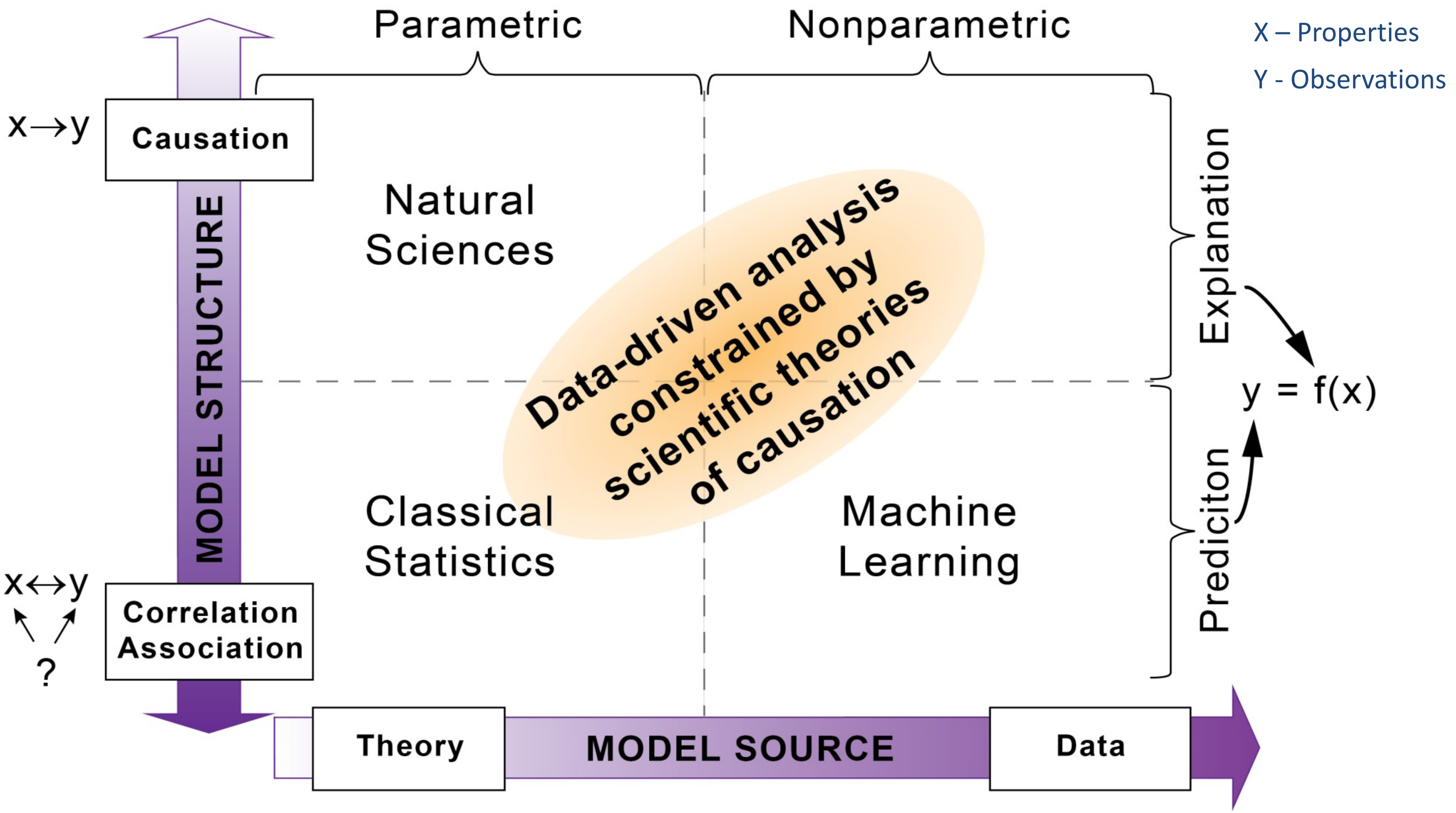
"curves"

$$f(x, y) = 0$$

Algorithmic procedure implemented on a computer

= **MACHINE LEARNING**

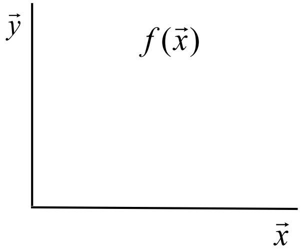
Physics and Machine Learning



Supervised vs. Unsupervised Learning

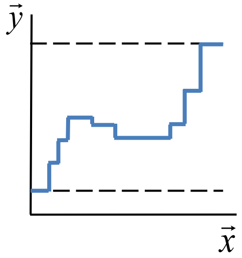
Supervised Learning

Regression



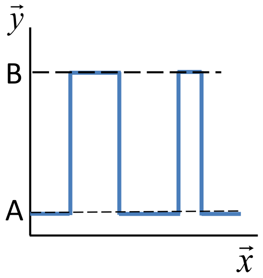
Continuum

Quantization



Multi-class

Classification

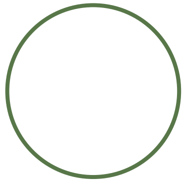


Binary class

Unsupervised Learning

Cluster

d=0

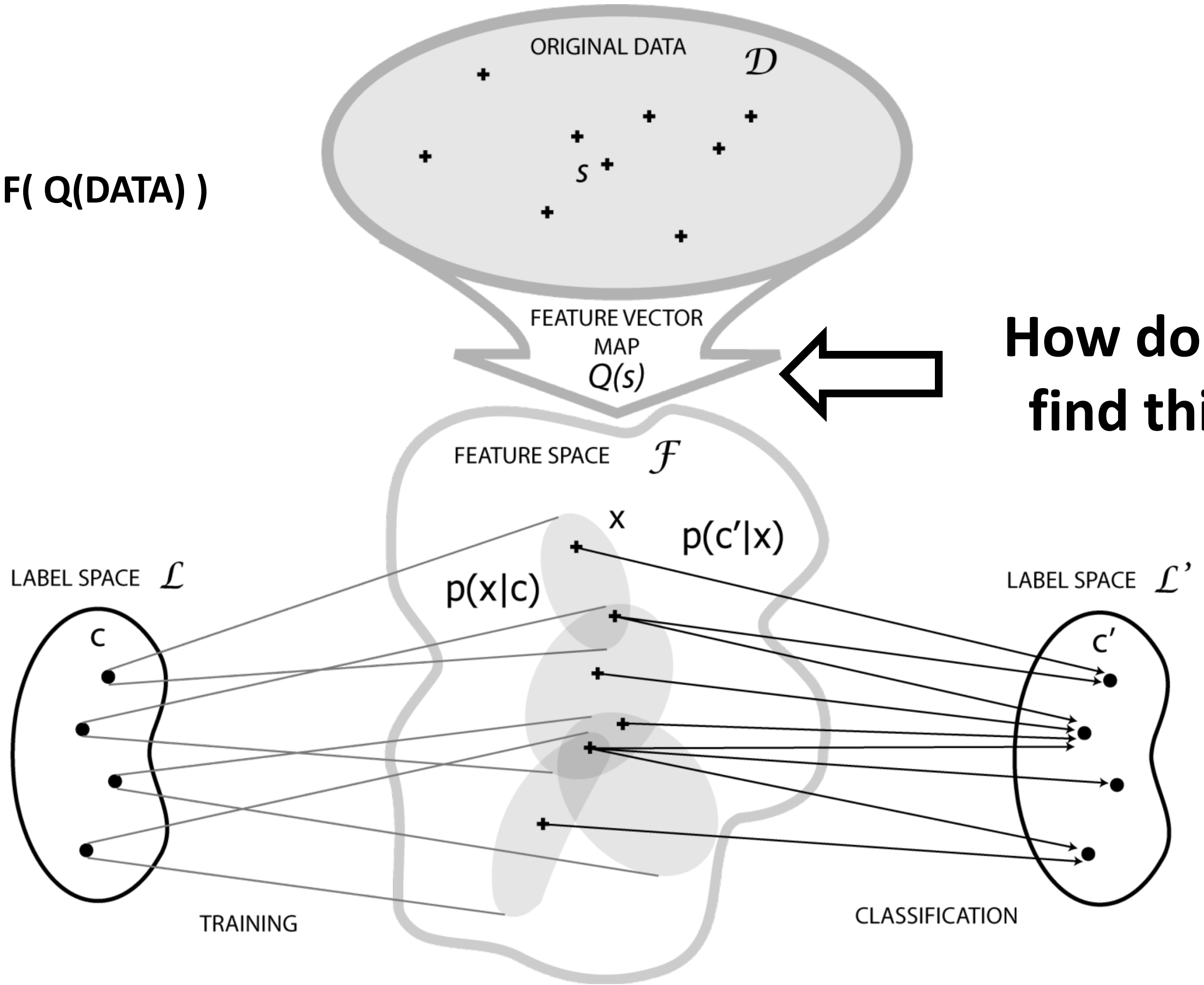


Manifold

d<<D

Supervised Learning

LABEL = F(Q(DATA))

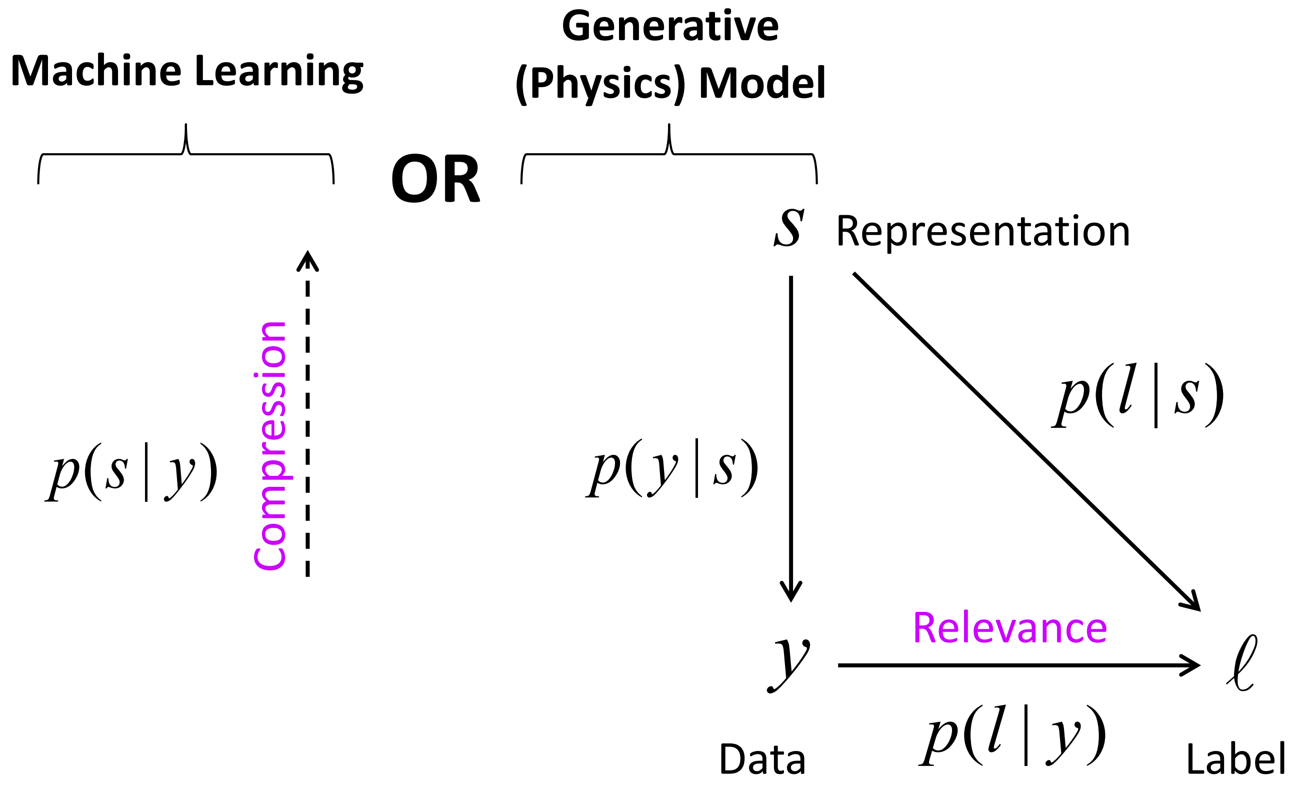


How do we find this?

Discriminative Supervised Learning uses geometric objects as boundaries: Hyper-planes, hyper-boxes, hyper-spheres

Information Bottleneck

Goal: *Compress* the data as much as possible while retaining the *relevant* information.



Formal description:

$$p^*(s|y) = \arg\max_{p(s|y)} \underbrace{\overbrace{I(S,L)}^{\text{Relevance}} - \lambda\,\overbrace{I(Y,S)}^{\text{Compression}}}_{=\mathcal{F}[p(s|y)]}$$

We want to ***maximize*** $I(S,L)$ to retain ***relevance***.

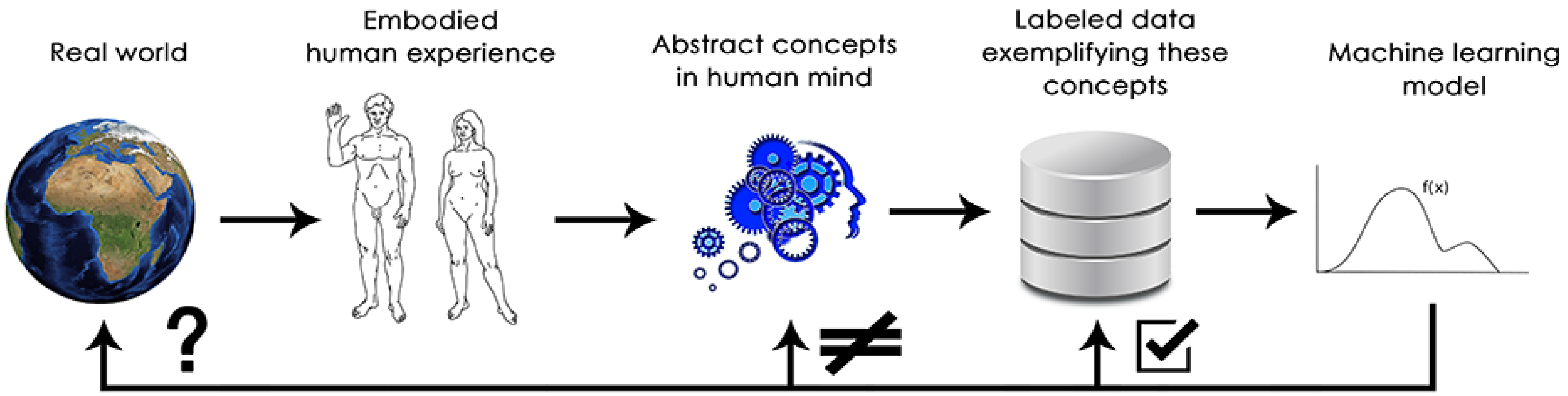
We want to ***minimize*** $I(Y,S)$,
to ***compress*** the data transferred
by the Generative model

The parameter λ is the Lagrange multiplier that tunes between these two competing objectives.

Supervised Learning

What are the issues with labels:

- Labels are **NOT** independent of the context that created them.
- Labels are **EXTREME** data compression preserving relevance in that context.
- Labels are a **SUPERFICIAL** transfer of intelligence into machine learning.



Brittle in Applications: May not always transfer well to the real world.

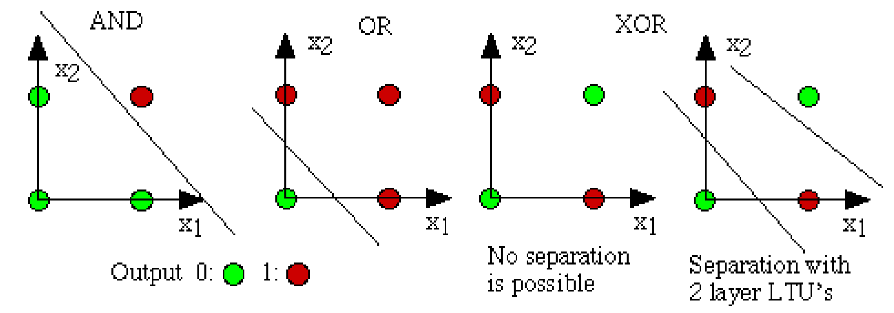
Not Explanatory: Does not match the human mental model.

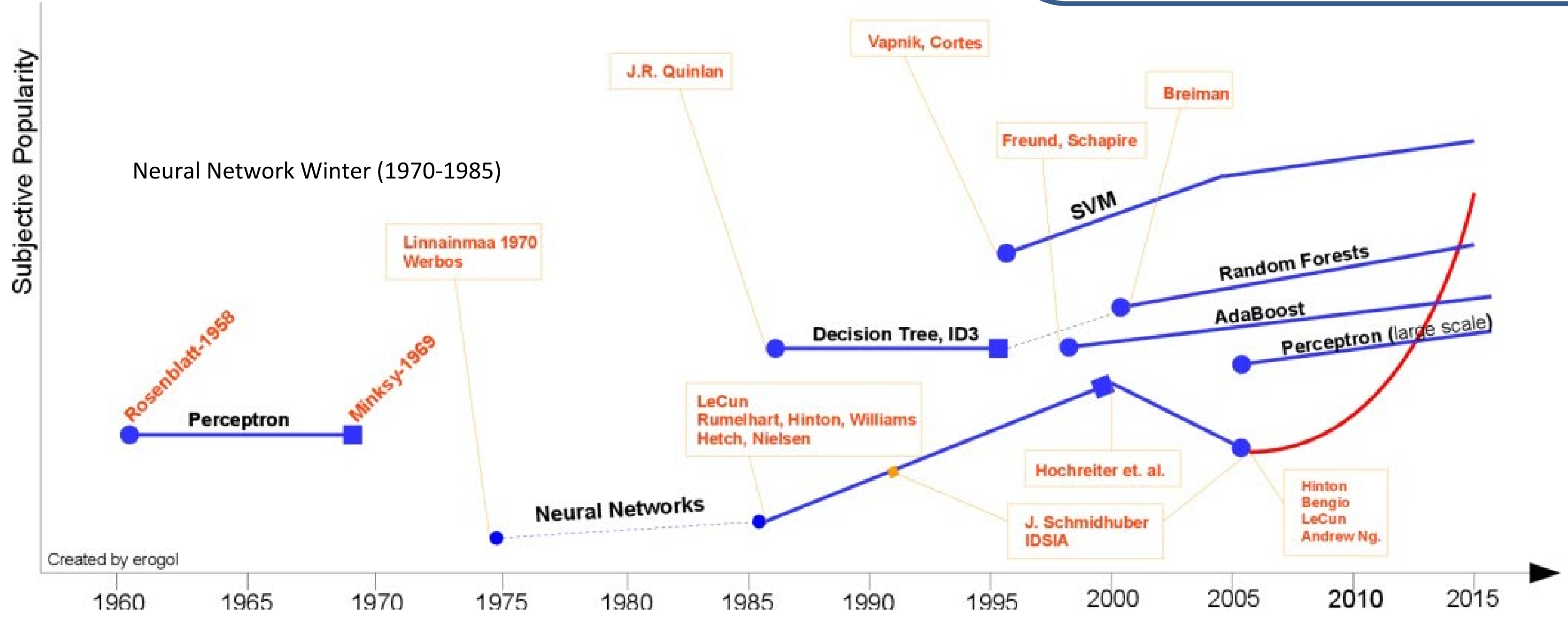
Narrow focus on performance: Matches the training data.

History of Machine Learning

Missing Bayesian techniques, PGM's, statistical techniques (e.g., Naïve Bayes, linear and logistic regression), PCA, *k*-NN, bootstrap, LASSO, Ridge regression), NLDR, logic and relational techniques, sampling techniques like MCMC, etc.   Advancements from signal processing … sparse dictionaries, regularization techniques, ICA, matrix factorizations (NNMF)
Also, reinforcement learning, active learning,

Minsky's **XOR** problem (*Perceptrons*, 1969) and MLP solution:





Backpropagation enables training of
multi-layer perceptrons (MLP), ~1985

http://www.erogol.com/brief-history-machine-learning/

Learning Dynamics of Neural Networks

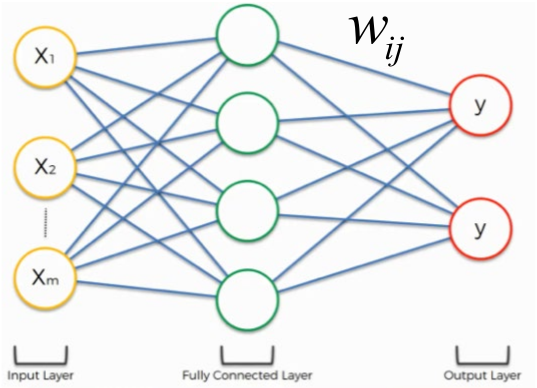
The Illusion of an Optimization Problem

Nonlinear transform of each coordinate of an affine map

$$y^i = \sigma_\bullet\left(w^i_j x^j + b^j\right)$$

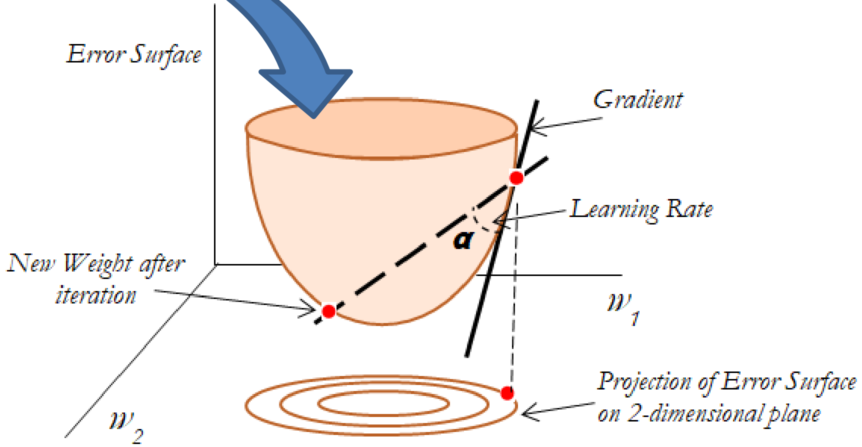
Neural Network

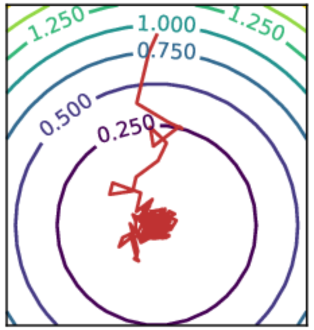
$$\mathbf{y} = f_\mathbf{W}(\mathbf{x})$$



Magic: "Inductive Bias" imposed on the network.

Cost Function: $C(\mathbf{W}) \simeq \frac{1}{N}\sum_{n=1}^{N}\left|\mathbf{y}_n - f_\mathbf{W}(\mathbf{x}_n)\right|^2$



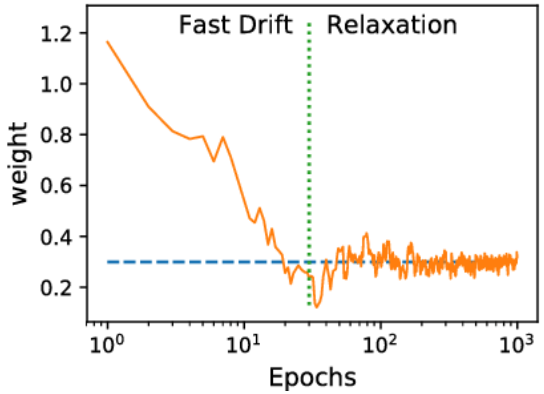


Gradient descent:

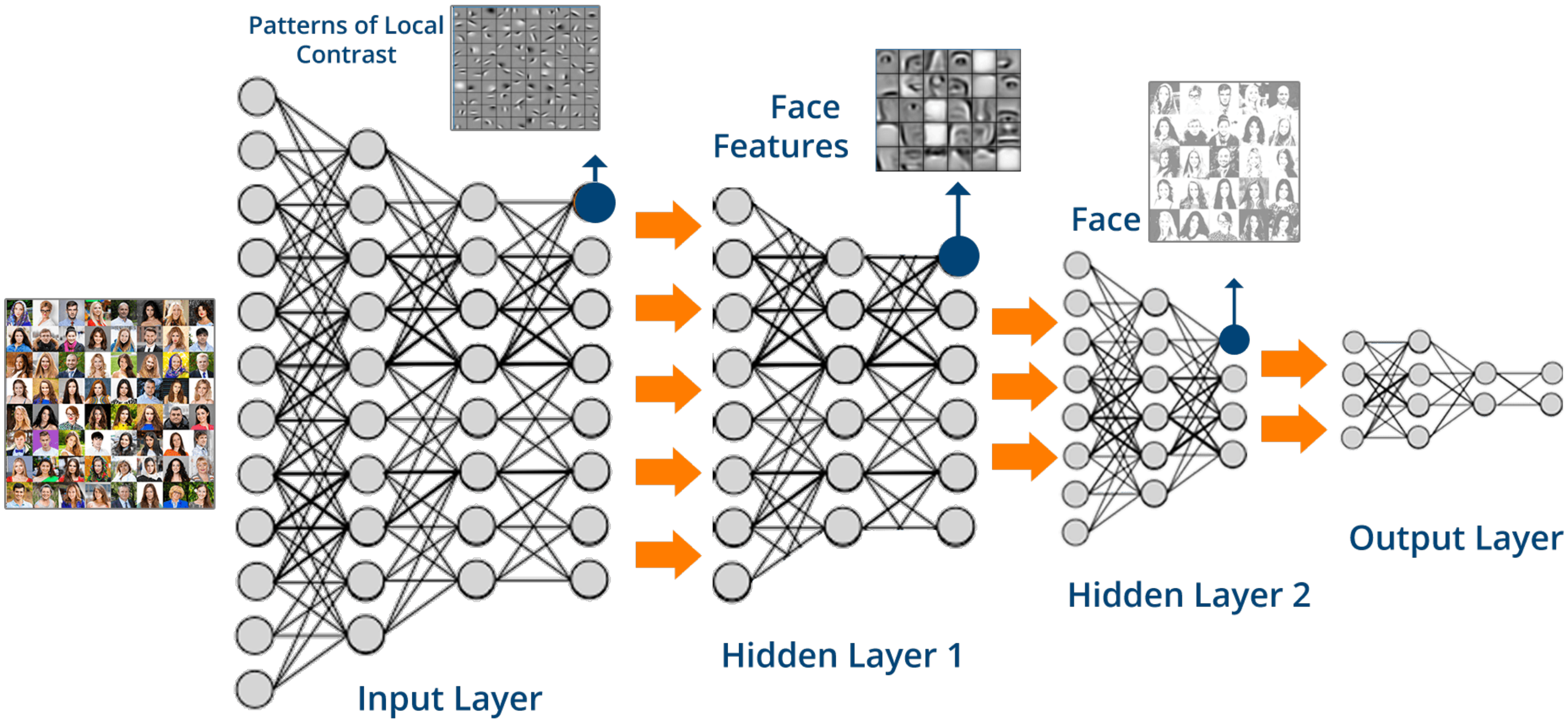
$$w_{ij}(t+1) = w_{ij}(t) - \underset{\uparrow\,\text{Learning Rate}}{\alpha}\,\overbrace{\frac{\partial C}{\partial w_{ij}}}^{\text{Automatic Differentiation}}$$

More sophisticated choices (e.g., ADAM) are possible.

Stochastic Gradient Descent: Backpropagation on $\underbrace{\text{mini-batches}}_{\textit{Random sub-sampling of data}}$



Deep Convolutional Neural Networks



How intelligent are neural networks?

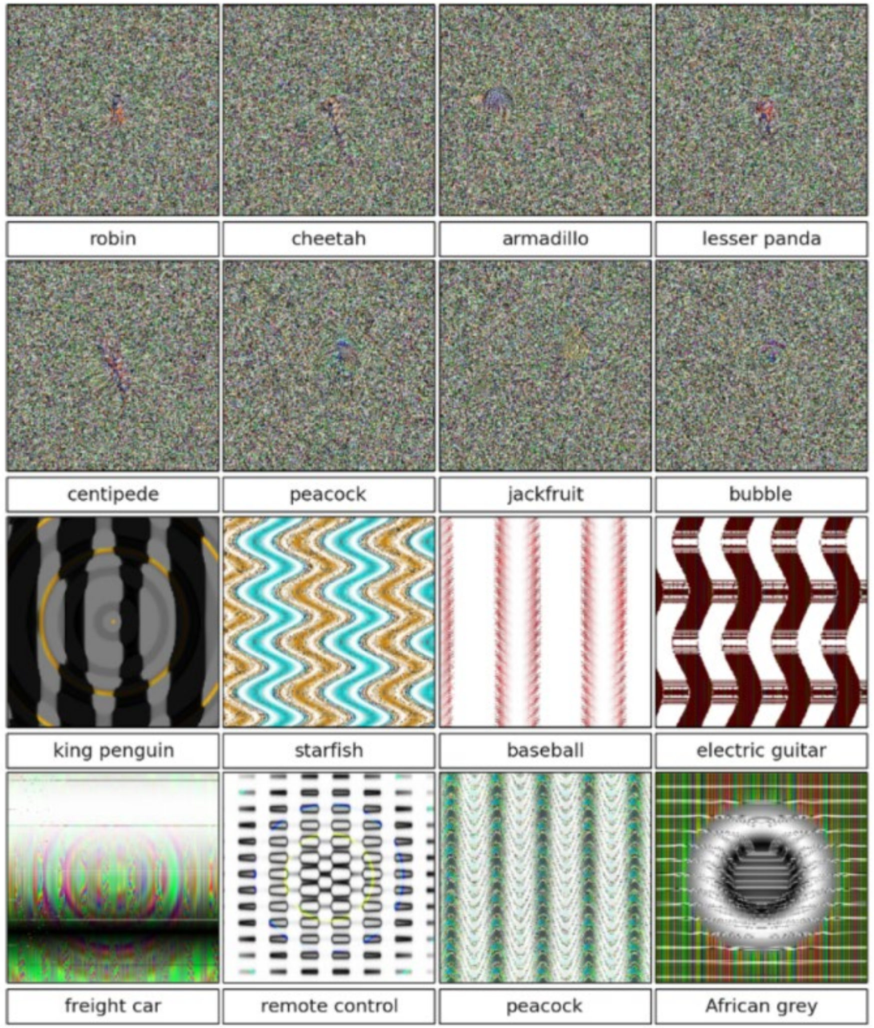
Intriguing properties of ~~neural networks~~ airliners

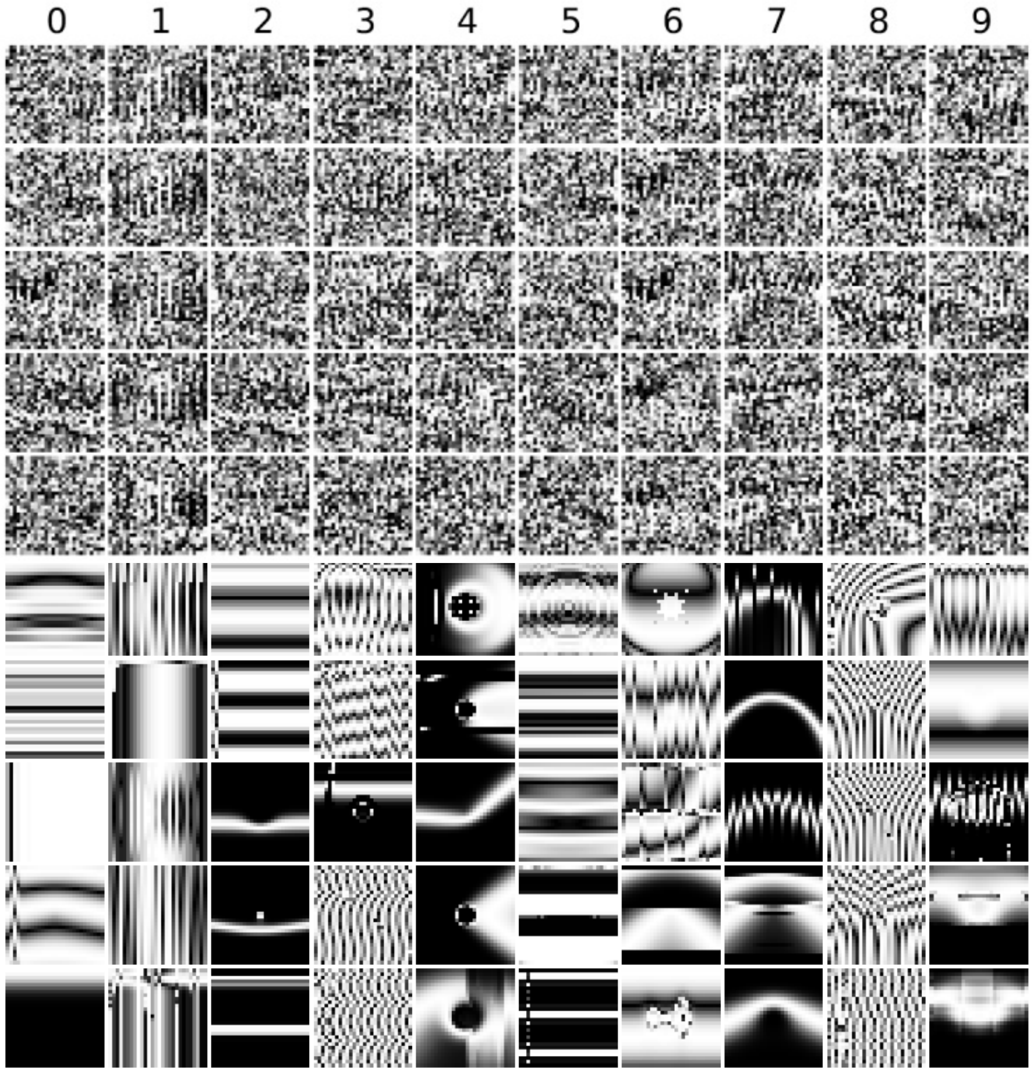
2014

Christian Szegedy Google Inc.
Wojciech Zaremba New York University
Ilya Sutskever Google Inc.
Joan Bruna New York University
Dumitru Erhan Google Inc.
Ian Goodfellow University of Montreal
Rob Fergus New York University, Facebook Inc.

Deep Learning: Adversarial Examples

The algorithm is >99.6% confident of these labels



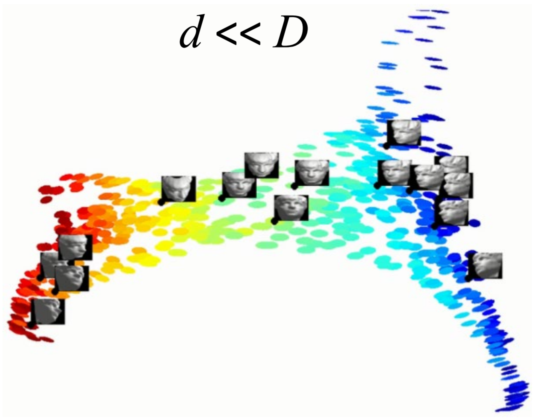


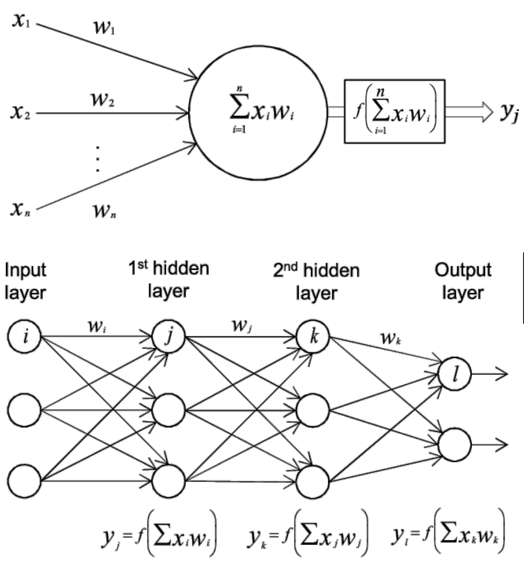
“Deep Neural Networks are Easily Fooled: High Confidence Predictions for Unrecognizable Images”, Anh Nguyen, Jason Yosinski and Jeff Clune, *CVPR 2015*, p.427-436.

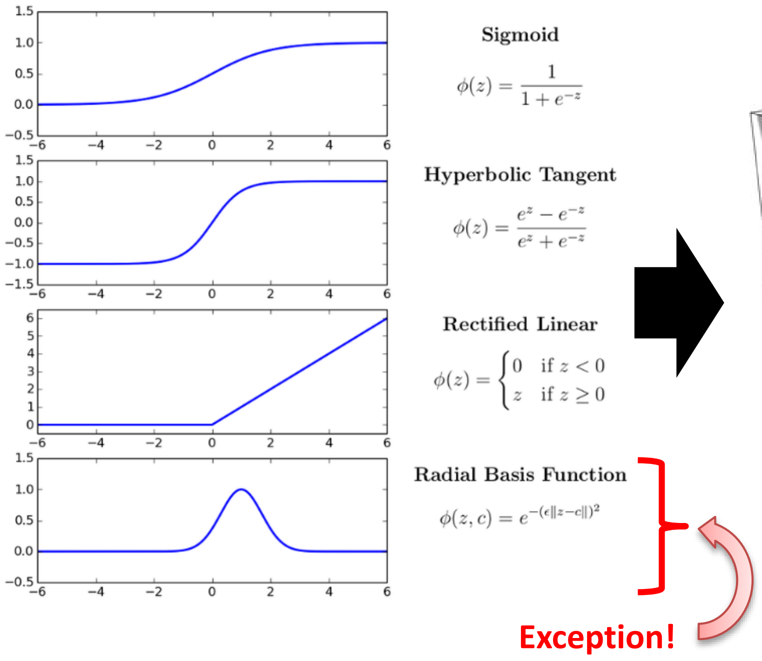
Geometry of Neural Networks

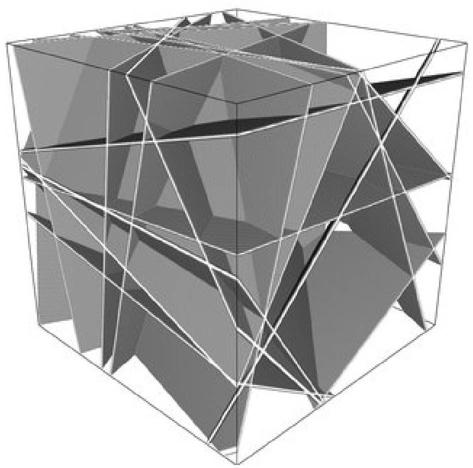
The data lies along a manifold constrained to low-dimensions by the generative mechanism.

However, the Neural Network typically chops up the space with hyperplanes to form non-local decision boundaries for classes.

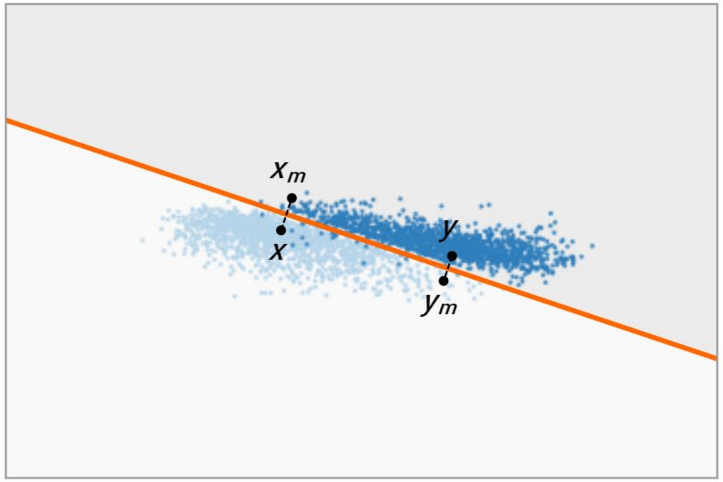


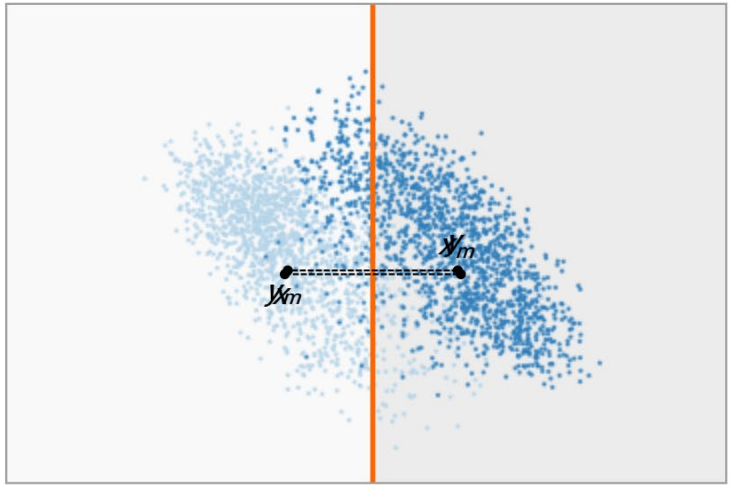




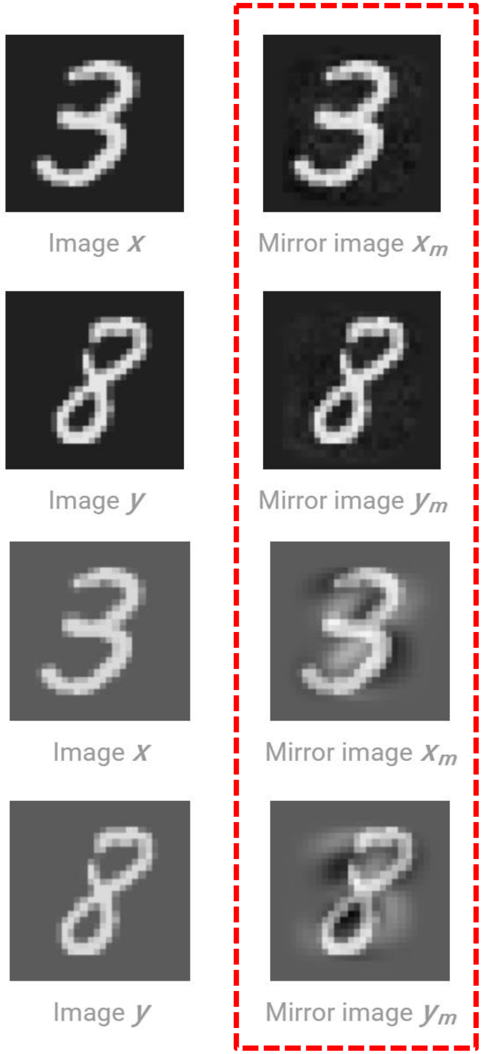


Deep Learning: Adversarial Examples



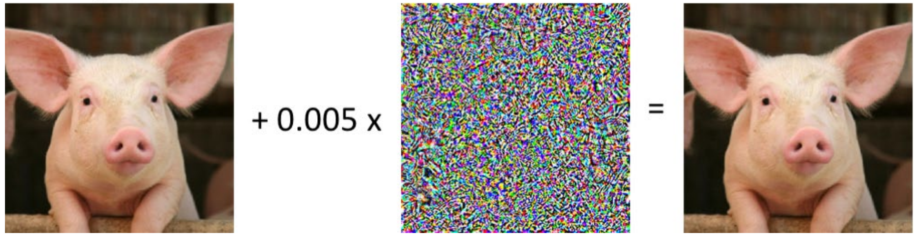


Real data **Adversarial data**



Nudging images in high-dimensional spaces

"pig" "airliner"



MIT CSAIL

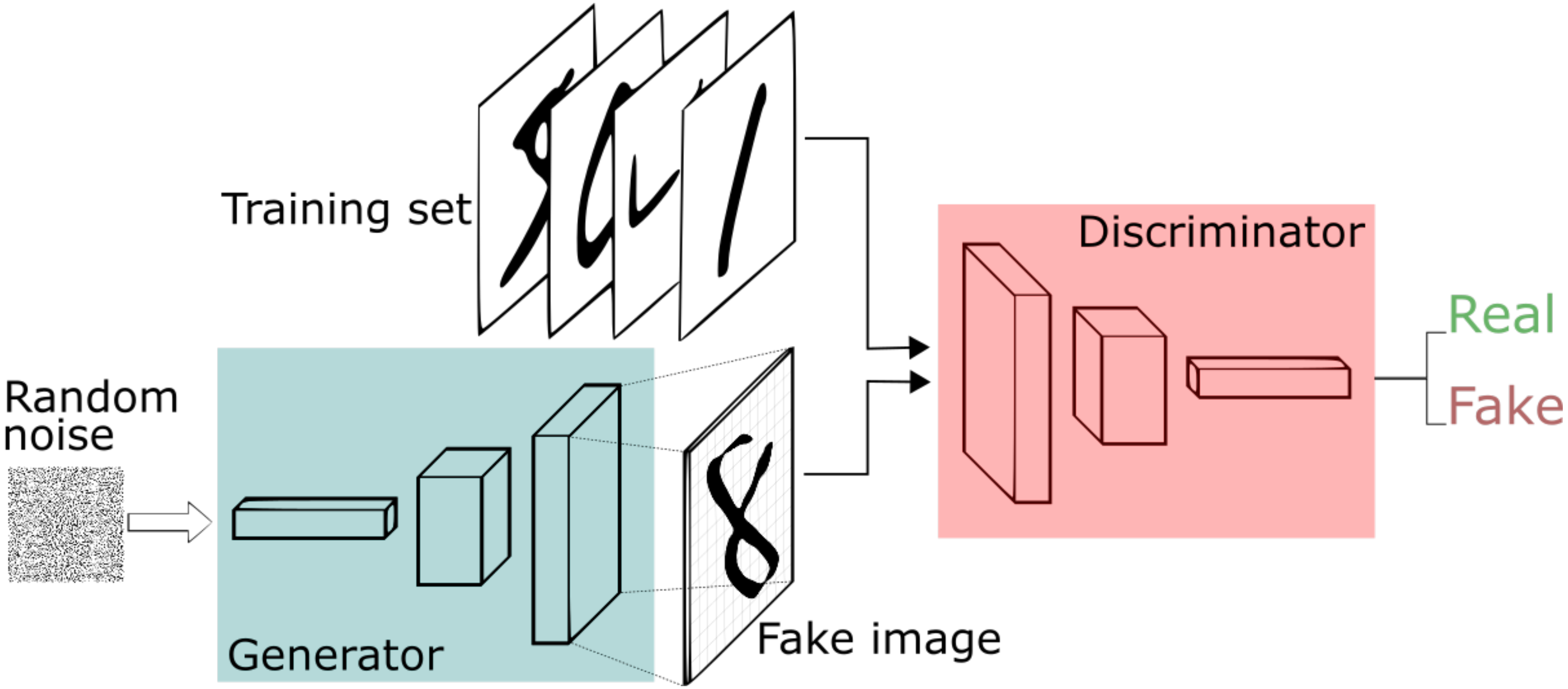
Use of ShapeShifter by Shang-Tse Chen, Ga.Tech

https://thomas-tanay.github.io/post--L2-regularization/

Generative Adversarial Network (GAN): Bug as Feature

> "This, and the variations that are now being proposed is the most interesting idea in the last 10 years in ML, in my opinion."
> -Yann LeCun (2016)

More Supervised Learning to the rescue … **LABEL** = { *Real*, *Fake* }

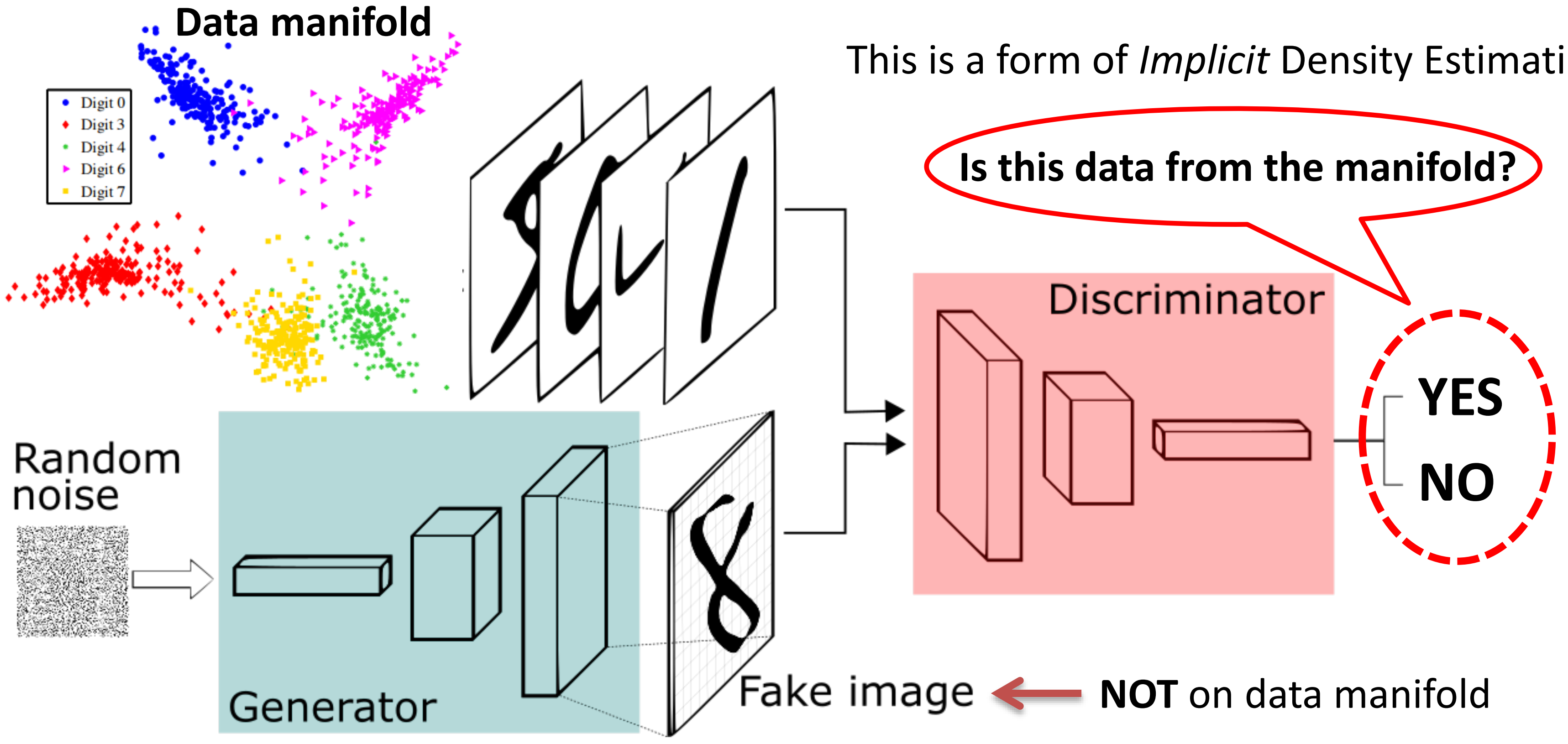


Generative Adversarial Network (GAN): Bug as Feature

> "This, and the variations that are now being proposed is the most interesting idea in the last 10 years in ML, in my opinion."
> -Yann LeCun (2016)

More Supervised Learning to the rescue …   **LABEL** = { *Real*, *Fake* }

This is a form of *Implicit* Density Estimation

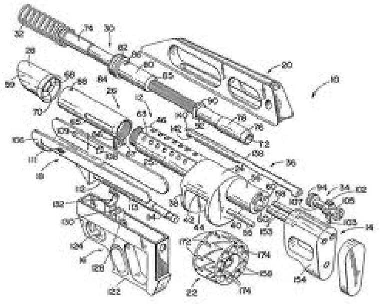


To be fair, a bit of a caricature of …

ML Work Flow leads to Gamification

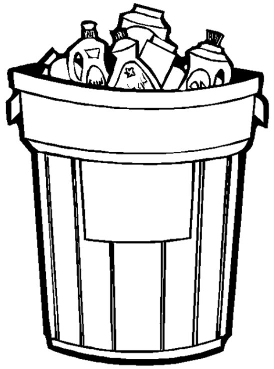
Invent new algorithm

Analyze the algorithm



Design choices smashed together, difficult to see sub-component interactions (Ablation studies?)

Apply to standardized data



Unknown intrinsic complexity

?

Place in archival literature

REPEAT

My algorithm is like your brain …

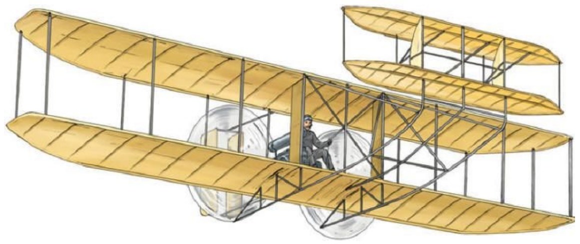
Bio-inspiration!

Biomimetic solution?

Neuromorphic computing
Neural networks

“Flapping wings” = “Firing neurons”

Or not?



$$\frac{\text{Propeller}}{\text{Flapping wings}} = \frac{?}{\text{Firing neurons}}$$

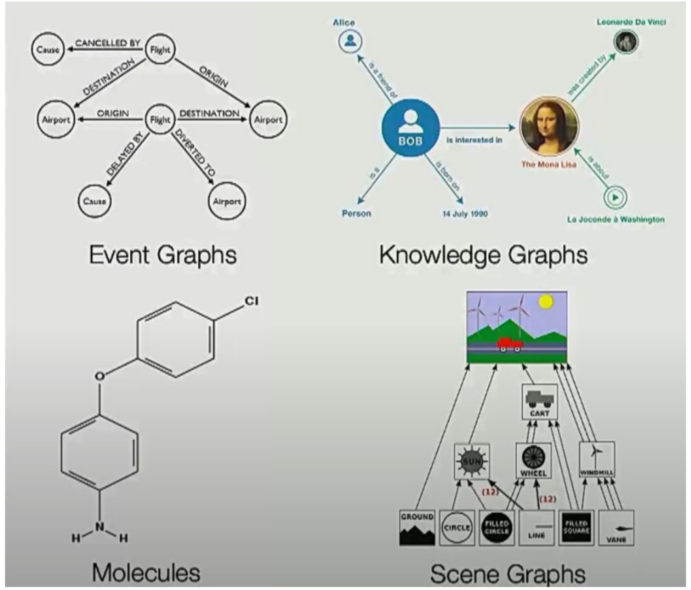
- Birds are a solution to an engineering problem with physical constraints. The Wright brothers understood the physical problem and then found an appropriate engineering solution different from birds.
- Brains are an evolutionary solution to the statistical constraints of inference from experience. What are those statistical constraints?

"It sort of makes you stop and think, doesn't it."
Ti fa solo fermare e pensare, vero?

Geometry of Machine Learning

	Physics	Machine Learning
Sets or Spaces:	MODEL $$\begin{tikzcd} \mathcal{M} \arrow[d, "f"'] \arrow[dr, "\tilde{L}"] & \\ \mathcal{D} \arrow[r, "L"] & \mathcal{L} \end{tikzcd}$$ DATA RELEVANCE	FEATURES $$\begin{tikzcd} & \mathcal{F} \arrow[d, "\tilde{L}"] \\ \mathcal{D} \arrow[ur, "G"] \arrow[r, "L"] & \mathcal{L} \end{tikzcd}$$ DATA RELEVANCE
Elements or Variables:	MODEL $$\begin{tikzcd} x \arrow[r, "\tilde{L}"] \arrow[d, "f"'] & \ell' \arrow[d, dash, dashed, "\Delta"] \\ y \arrow[r, "L"] & \ell \end{tikzcd}$$ DATA RELEVANCE	FEATURES $$\begin{tikzcd} z \arrow[r, "\tilde{L}"] & \ell' \arrow[d, dash, dashed, "\Delta"] \\ y \arrow[u, "G"] \arrow[r, "L"] & \ell \end{tikzcd}$$ DATA RELEVANCE

Diverse and complex data structures



But we want to use Linear Algebra and Multivariable Calculus!

So just work in a huge vector space! $\mathbb{R}^D$

Functions acting on these vectors should have:
Continuity → Smoothness → Differentiability } ***manifold***

Model regularization (prior) *Learning via gradient descent*

Geometry of Machine Learning

STEPS:

1. Form a DATA space, $\mathcal{D}$, as a high-dimensional vector space, $\mathbb{R}^D$.
2. Identify transformations, S, on the DATA space that leave the similarity measure invariant.
3. Learn the underlying MODEL space, $\mathcal{M}$, or "embedding" that preserves the invariances of the similarity measure.

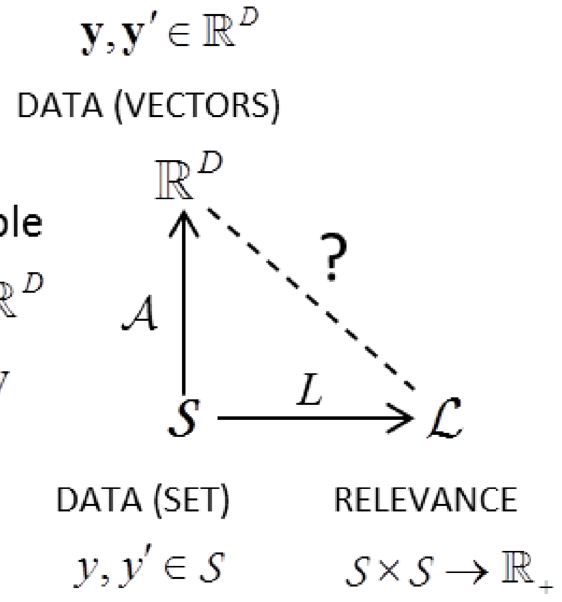
A *group*, G, acting on the *set*, $\mathcal{S}$, that leaves the similarity measure *invariant*:

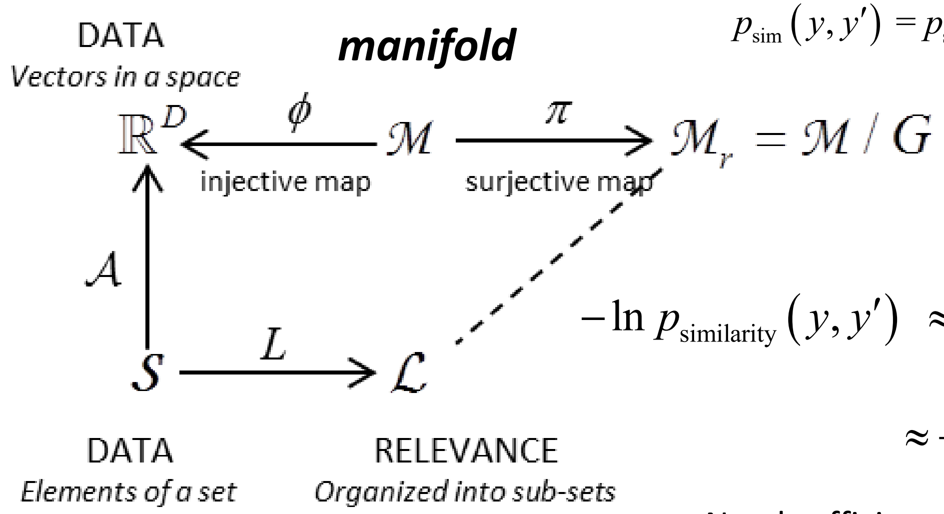
$$p_{\text{sim}}(y, y') = p_{\text{sim}}(g_\alpha y, g_\beta y') \text{ where } g_\alpha, g_\beta \in G.$$

Not learnable

$\mathcal{A} : \mathcal{S} \to \mathbb{R}^D$

$y \mapsto \mathbf{y}$





$$-\ln p_{\text{similarity}}(y, y') \approx \frac{1}{2}\text{dist}^2_{\mathcal{M}}(\mathcal{A}(y), \mathcal{A}(y'))$$

$$\approx \frac{1}{2}\|\mathbf{y} - \mathbf{y}'\|_2^2$$

Need sufficient sampling so the data is "close" enough to use this approximation.

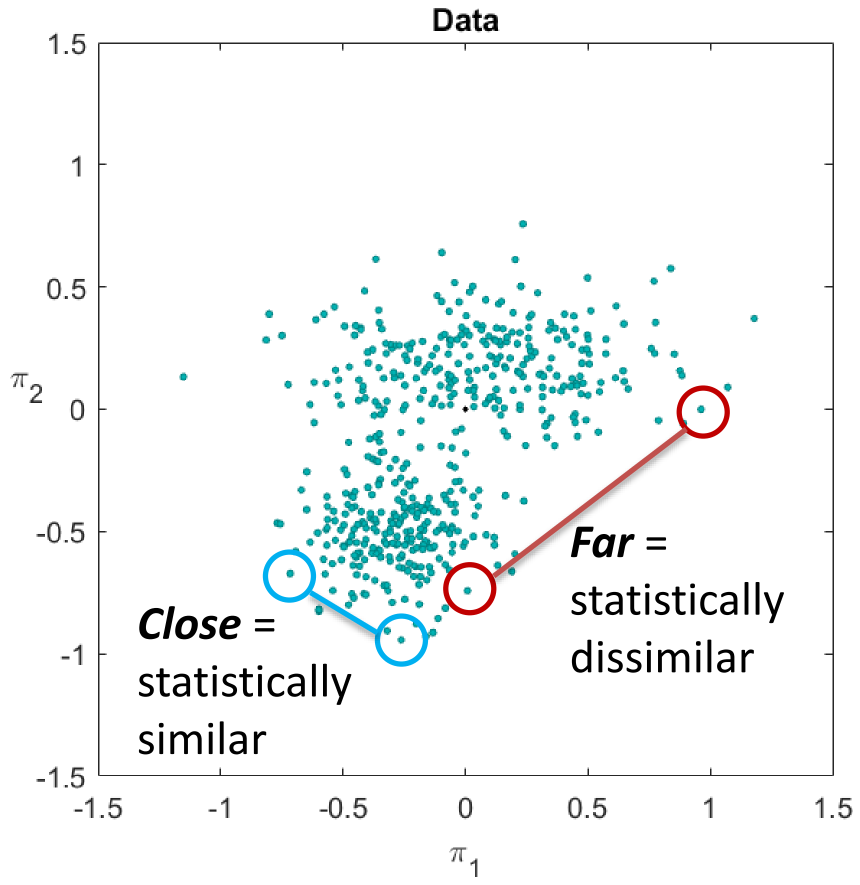
Geometry of ML: How much data is enough?

Answer: It's not about having "a lot of data", it's about having *enough* data in the *right places* to answer a particular question.

Assumption: The *distance* encodes information about *statistical similarity*.

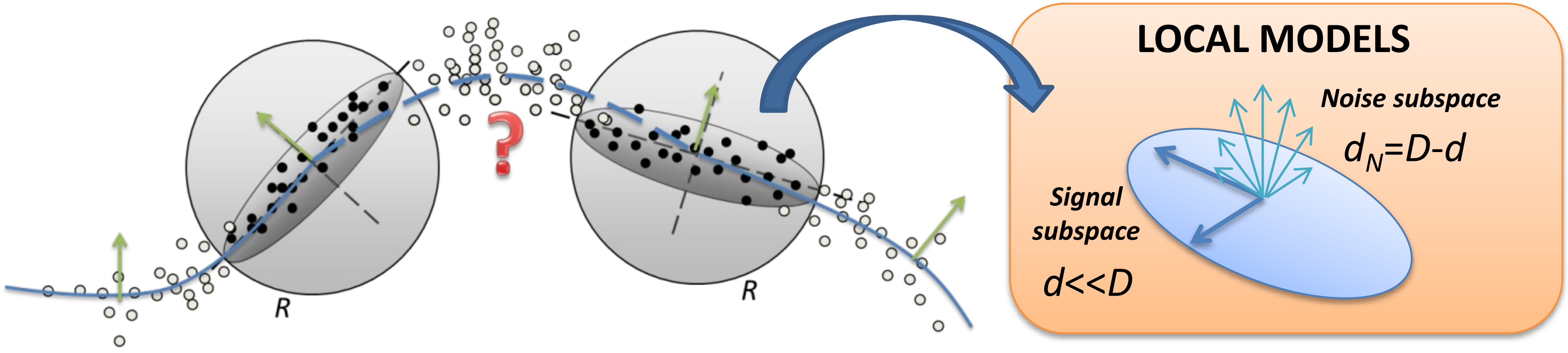
$$-\ln p_{\text{similarity}}\left(y, y'\right) \approx \frac{1}{2}\left\|\mathbf{y}-\mathbf{y}'\right\|_2^2$$

If distant data points are mapped nearby in the model space, then there has to be significant inferential evidence (more data), and likewise if local data points are mapped to large separations by the model.

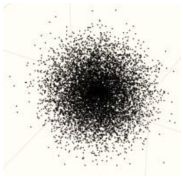


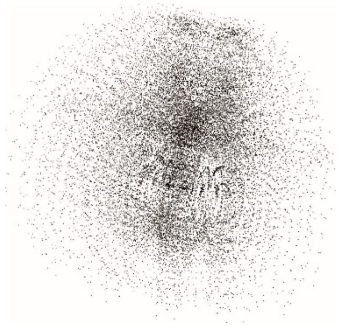
Geometry of ML: How much data is enough?

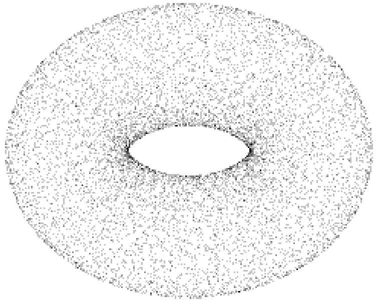
To find the best local linear model: We need enough data *locally* to distinguish curvature from finite sampling noise



Local structure	**Crossover Regime**	**Large-scale structure**
High-*D* Noise Balls	Noisy Tangent Spaces	Low-*d* Manifold
	(*with an envelope*)	







Geometry of ML: Experimental Calibration

Generative model for data:

Observations (Dependent var.)

Latent variables (Independent var.)

Instrument state (Model parameters)

$$\mathbf{Y} = f(\mathbf{X} \mid \boldsymbol{\alpha})$$

1. CALIBRATION

Regression with training data

KNOWN **KNOWN** **UNKNOWN**

$$\mathbf{Y} = f(\mathbf{X}_{\text{calib}} \mid \boldsymbol{\alpha})$$

KNOWN DATA SOURCE or more a MODEL

$$p(\boldsymbol{\alpha} \mid \mathbf{X}_{\text{calib}})$$

Estimate of the unknown state of the instrument

$$\hat{\boldsymbol{\alpha}}$$

Nuisance variable to marginalize out

2. MEASUREMENT

Inferring latent variables

KNOWN **UNKNOWN**

$$\mathbf{Y} = f(\mathbf{X} \mid \hat{\boldsymbol{\alpha}})$$

UNKNOWN DATA SOURCE (Experiment)

$$p(\mathbf{X} \mid \mathbf{X}_{\text{calib}}) = \int d\boldsymbol{\alpha}\, p(\mathbf{X} \mid \boldsymbol{\alpha}) \cdot p(\boldsymbol{\alpha} \mid \mathbf{X}_{\text{calib}})$$

Estimate of the unknown latent variables

$$\hat{\mathbf{X}}$$

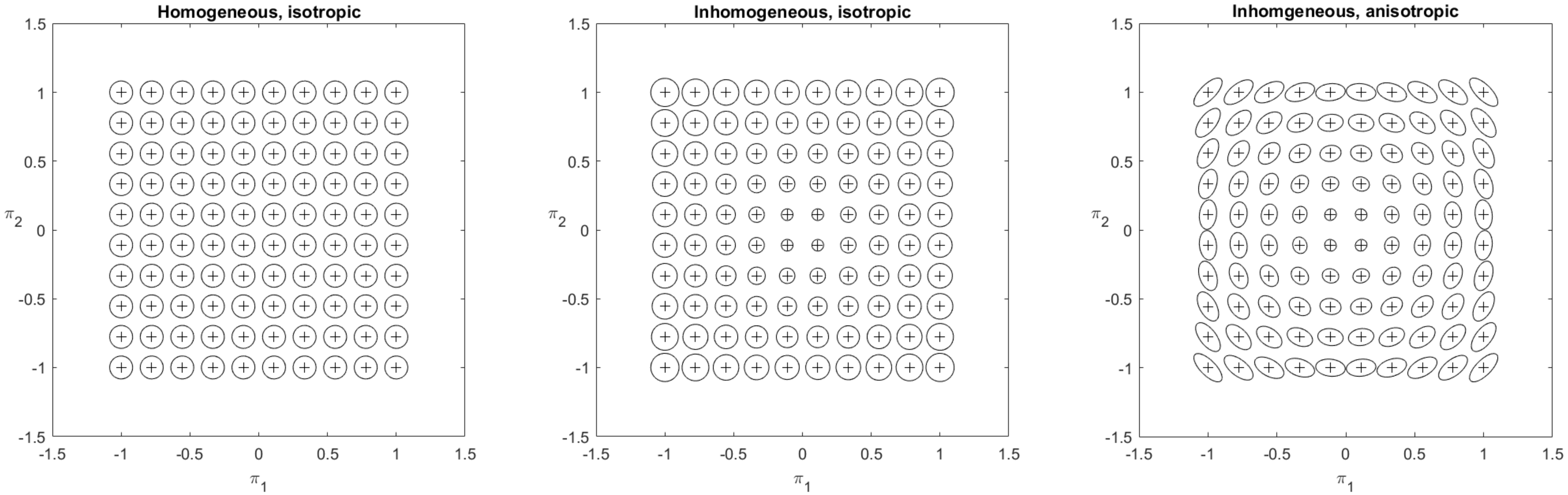
Did the calibration provide coverage?

Geometry of ML: Experimental Calibration

Examples of different possible calibrations across the **DATA** space …

1. CALIBRATION

(1,1)-tensor field of covariance



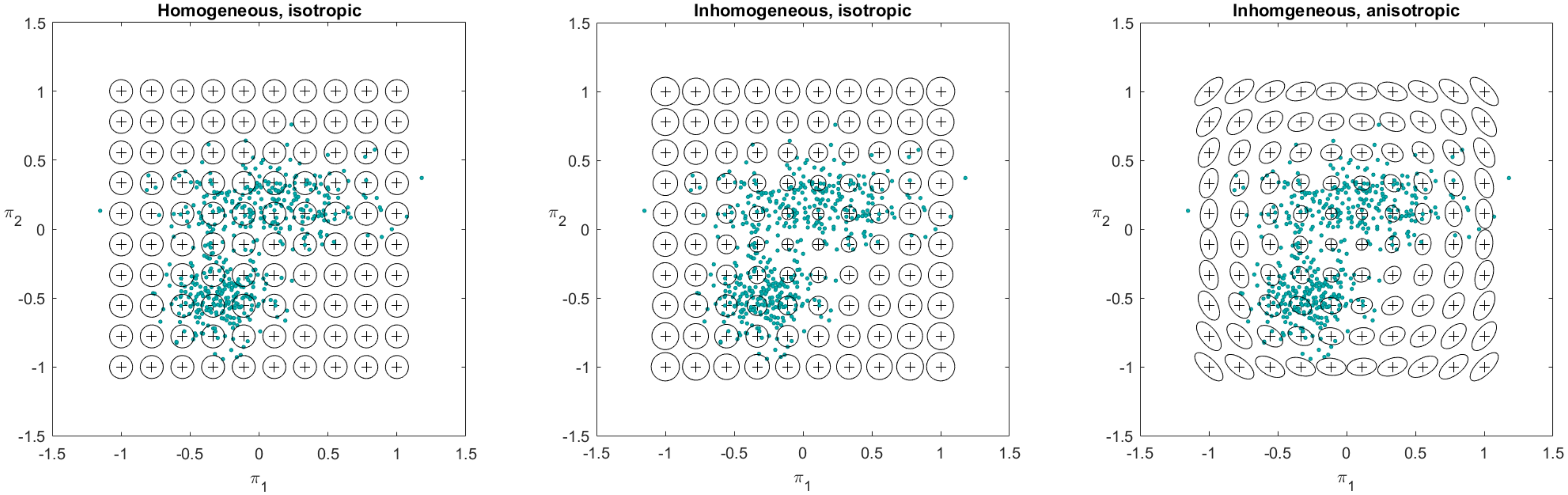
How do we use this information in Machine Learning?

Geometry of ML: Experimental Calibration

…. with the "same" data measurements.

2. MEASUREMENT

(1,1)-tensor field of covariance

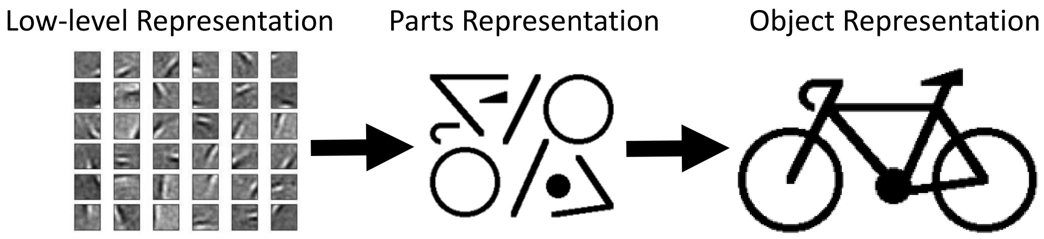


Use the covariance to estimate the metric tensor of the measurement space

$$\mathbf{g}(y) = \mathbf{\Sigma}^{-1}(y) = \mathbf{J}_y^T \mathbf{J}_y \rightarrow \tilde{\mathbf{g}} = \mathbb{I}_D$$

Now, the ML Euclidean space ansatz is true!

Geometry of ML: Manifolds and Relevance



DATA

Vectors in a space

$$
\begin{tikzcd}
\mathbb{R}^D & \mathcal{M} \arrow[l, "\phi"', "\text{injective map}"] \arrow[r, "\pi", "\text{surjective map}"'] & \mathcal{M}_r = \mathcal{M} / G \\
\mathcal{S} \arrow[u, "\mathcal{A}"] \arrow[r, "L"] & \mathcal{L} \arrow[ur, dashed] &
\end{tikzcd}
$$

DATA

Elements of a set

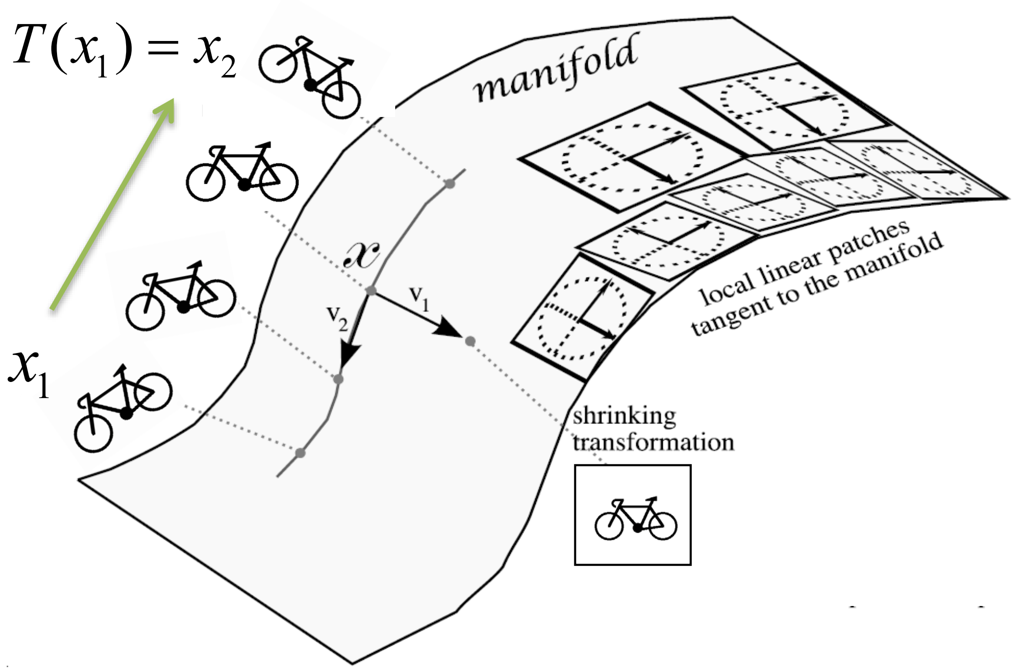
RELEVANCE

Organized into sub-sets

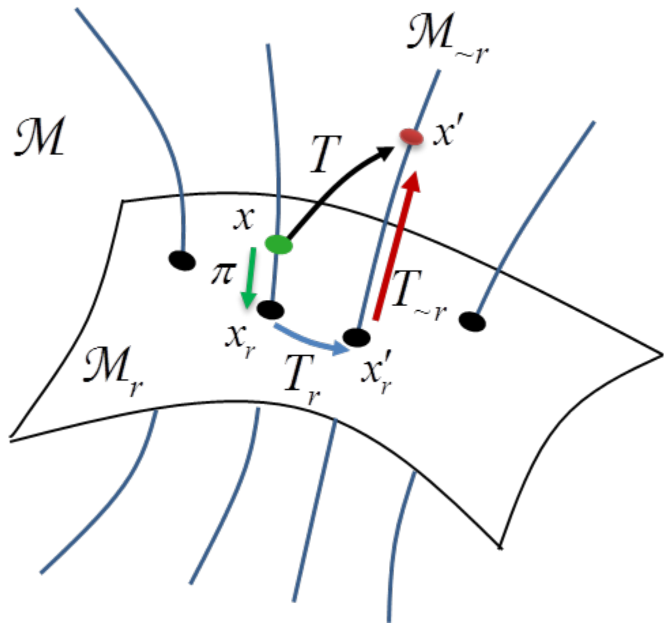
A *group*, G, acting on the *set*, $\mathcal{S}$, that leaves the similarity measure *invariant*:

$$p_{\text{sim}}(y, y') = p_{\text{sim}}(g_\alpha y, g_\beta y')$$

where $g_\alpha, g_\beta \in G$.



Re-organize the space as a *fiber bundle*

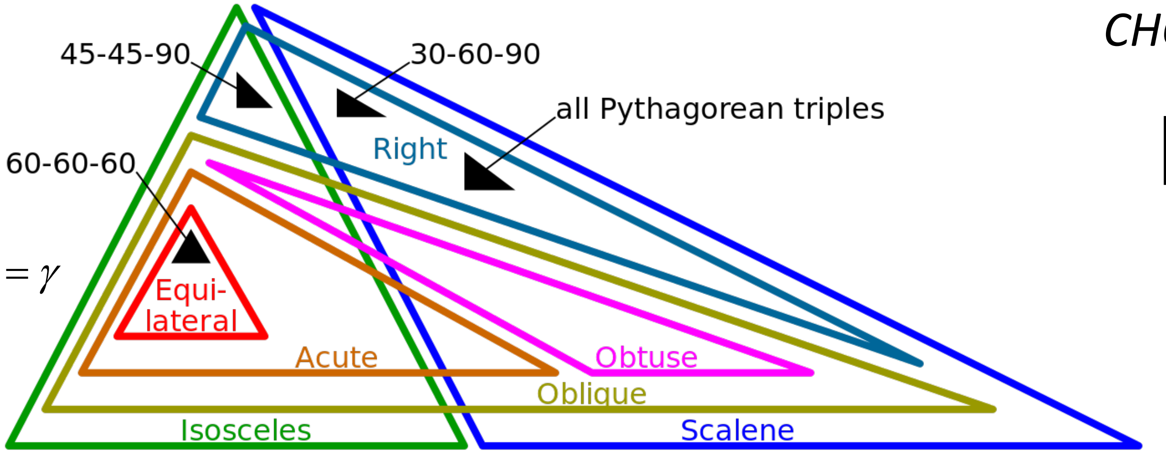


Example: Choosing the Relevance of Triangles

Types of Triangles

$$\alpha+\beta+\gamma=\pi$$

$\alpha=\frac{\pi}{2}, \beta=\frac{\pi}{2}$, or $\gamma=\frac{\pi}{2}$



$\alpha=\beta=\gamma$

$\alpha=\beta\neq\gamma, \beta=\gamma\neq\alpha$ or $\gamma=\alpha\neq\beta$

$\alpha\neq\beta\neq\gamma$

CHOOSE

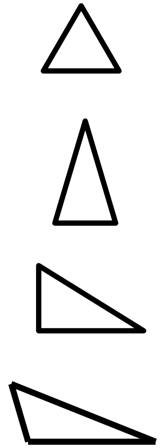
Relevance Set $\mathcal{L}$

Equilateral

Isosceles

Right

Scalene



y = IMAGE → "f^{-1}"= Vertex Detector → x, dim($\mathcal{M}$)=6 → x_r, dim($\mathcal{M}_r$)=2 → ℓ, $|\mathcal{L}|=4$

$$\ell = L(y)$$

Example: Shapes and Geometric Invariants

$$\mathcal{M} = \mathbb{R}^6$$

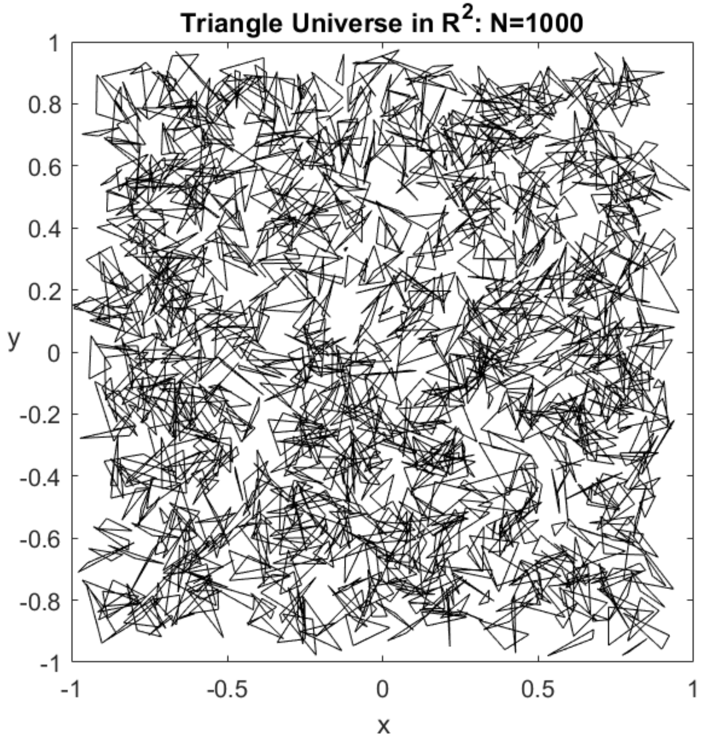
$$\dim \mathcal{M} = 6$$

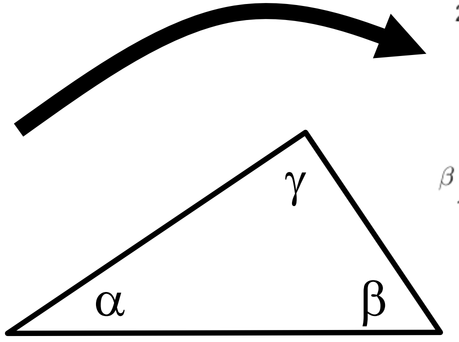
$$\dim \mathcal{M}_r = 2$$

$$x_r = \pi \circ f^{-1}(y)$$

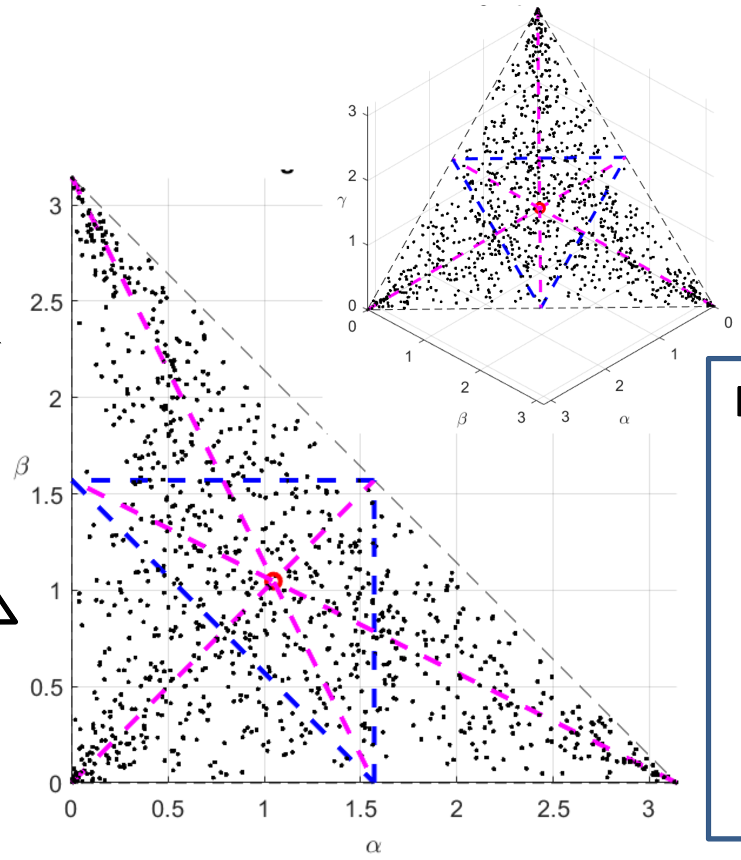
Data Space $\mathcal{D}$

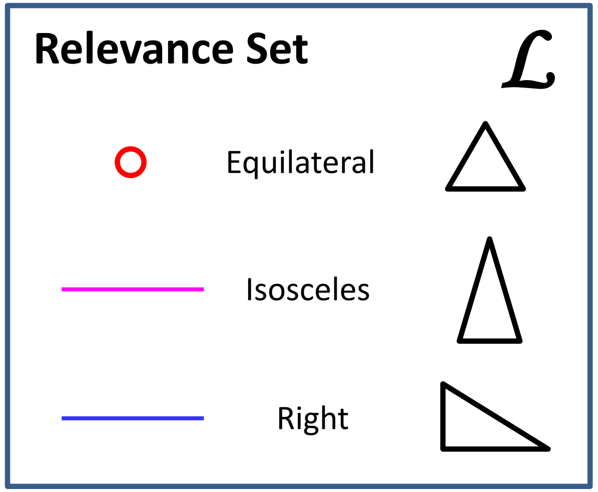
Model Relevance Space $\mathcal{M}_r$





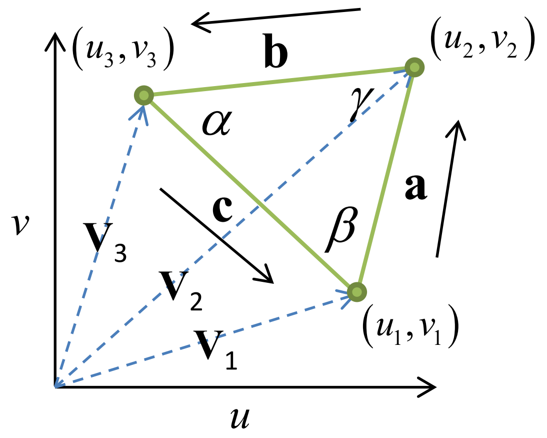
$$\alpha + \beta + \gamma = \pi$$





Transformations: From Model to Model Relevance Space

MODEL space



$$x_r = \pi(x)$$

Translation invariance

Scale invariance

Rotation invariance

$$\mathbf{x} = \begin{pmatrix} \mathbf{v}_1 \\ \mathbf{v}_2 \\ \mathbf{v}_3 \end{pmatrix} = \begin{pmatrix} u_1 \\ v_1 \\ u_2 \\ v_2 \\ u_3 \\ v_3 \end{pmatrix} \Rightarrow \mathbf{s} = \begin{pmatrix} \mathbf{a} \\ \mathbf{b} \\ \mathbf{c} \end{pmatrix} = \begin{pmatrix} u_2 - u_1 \\ v_2 - v_1 \\ u_3 - u_2 \\ v_3 - v_2 \\ u_1 - u_3 \\ v_1 - v_3 \end{pmatrix} \Rightarrow \hat{\mathbf{s}} = \begin{pmatrix} \hat{\mathbf{a}} \\ \hat{\mathbf{b}} \\ \hat{\mathbf{c}} \end{pmatrix} = \begin{pmatrix} \frac{u_2 - u_1}{a} \\ \frac{v_2 - v_1}{a} \\ \frac{u_3 - u_2}{b} \\ \frac{v_3 - v_2}{b} \\ \frac{u_1 - u_3}{c} \\ \frac{v_1 - v_3}{c} \end{pmatrix} \Rightarrow \begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix} = \begin{pmatrix} \cos^{-1}\left(-\hat{\mathbf{b}}^T \hat{\mathbf{c}}\right) \\ \cos^{-1}\left(-\hat{\mathbf{c}}^T \hat{\mathbf{a}}\right) \\ \cos^{-1}\left(-\hat{\mathbf{a}}^T \hat{\mathbf{b}}\right) \end{pmatrix}$$

Rotation invariance

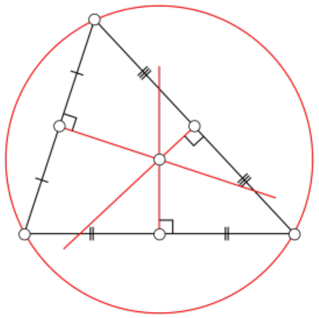
$$a = \sqrt{\mathbf{a}^T \mathbf{a}} = \sqrt{(u_2 - u_1)^2 + (v_2 - v_1)^2}$$
$$b = \sqrt{\mathbf{b}^T \mathbf{b}} = \sqrt{(u_3 - u_2)^2 + (v_3 - v_2)^2}$$
$$c = \sqrt{\mathbf{c}^T \mathbf{c}} = \sqrt{(u_1 - u_3)^2 + (v_1 - v_3)^2}$$

$$\mathbf{x}_r = \begin{pmatrix} \xi \\ \eta \\ s \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{2}}(\beta - \alpha) \\ \frac{1}{\sqrt{6}}(2\gamma - \alpha - \beta) \\ \frac{a+b+c}{\sin\alpha + \sin\beta + \sin\gamma} \end{pmatrix}$$

s is **NOT** scale invariant

Diameter of the circumscribed circle:

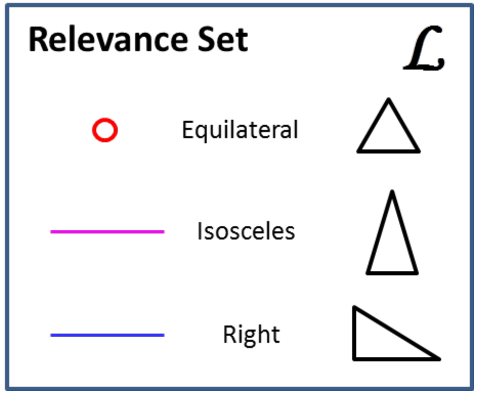
$$s = \frac{a}{\sin\alpha} = \frac{b}{\sin\beta} = \frac{c}{\sin\gamma}$$



$$s = \frac{a+b+c}{\sin\alpha + \sin\beta + \sin\gamma}$$
$$P = a + b + c = s \cdot (\sin\alpha + \sin\beta + \sin\gamma)$$
$$A = \frac{1}{2} s^2 \sin\alpha \sin\beta \sin\gamma$$

$$\begin{aligned} \sin\gamma &= \sin(\pi - \alpha - \beta) \\ &= \sin\pi \cdot \cos(\alpha + \beta) - \cos\pi \cdot \sin(\alpha + \beta) \\ &= \sin(\alpha + \beta) \end{aligned}$$

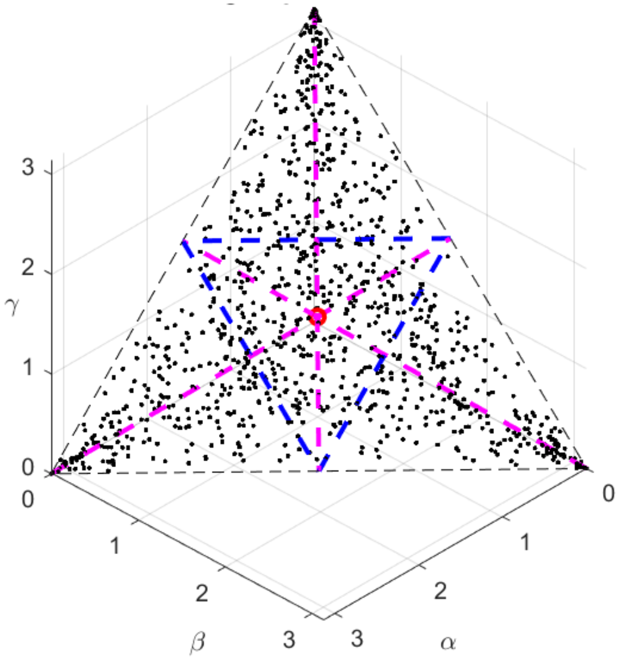
Model compression: Triangle Example

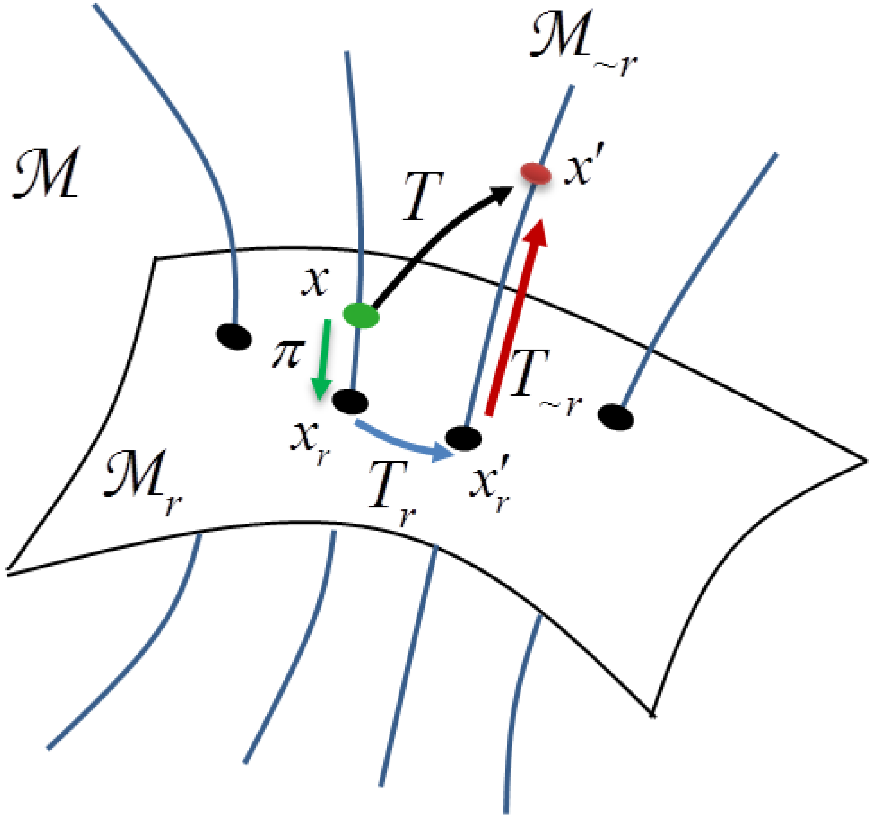


$$\mathcal{M} = \mathbb{R}^6$$
$$\dim \mathcal{M} = 6$$
$$\dim \mathcal{M}_r = 2$$
$$x_r = \pi \circ f^{-1}(y)$$

sim(2)

- **Translation invariance**
- **Scale invariance**
- **Rotation invariance**





Physics and Machine Learning