

Deep-Learning Do-It-Yourself



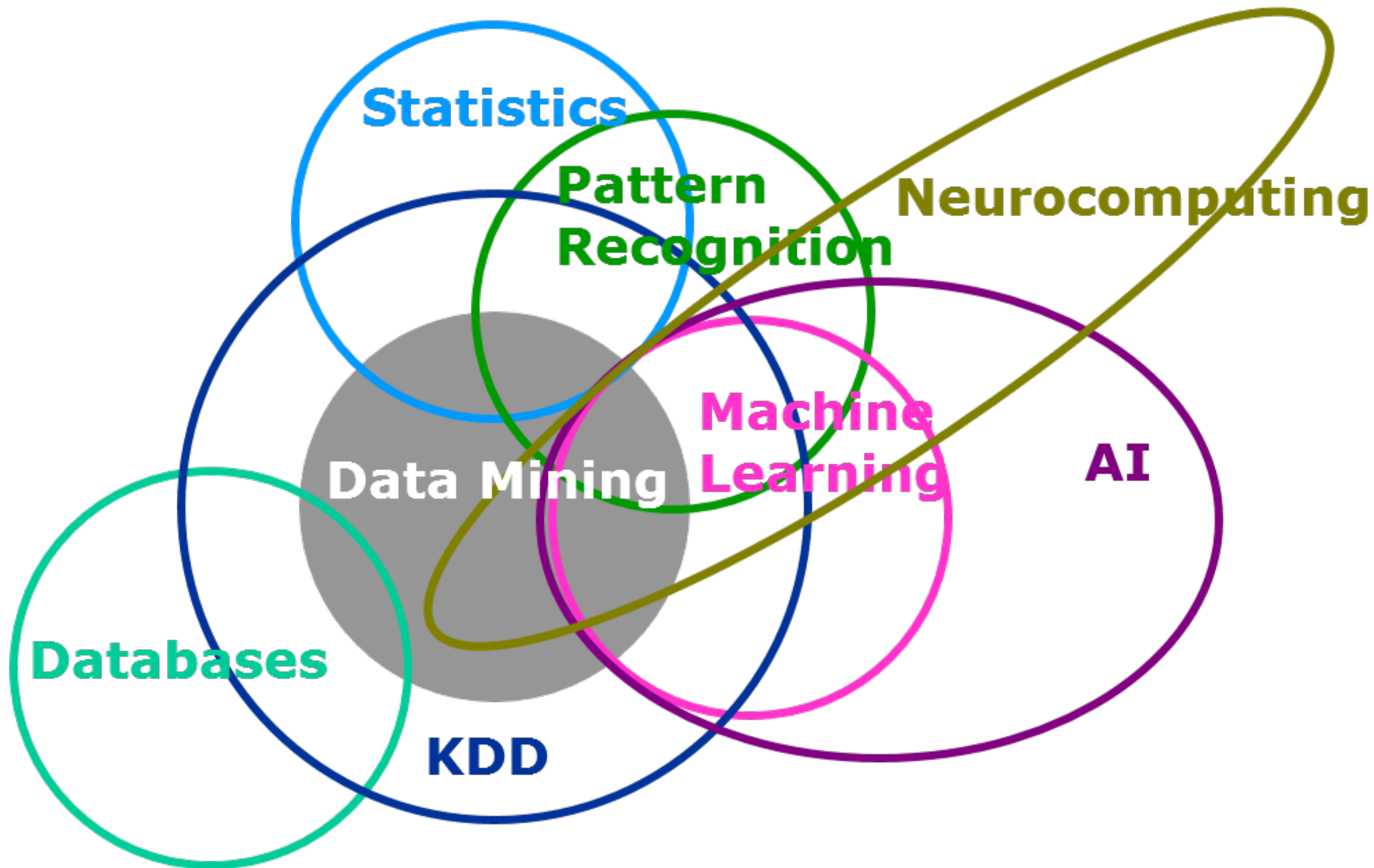
SAFRAN
AEROSPACE · DEFENCE · SECURITY



What is machine learning? What is data science?

The process of learning from data ?

What is machine learning? What is data science?



Not so clear...

Simpler question: what do machine learning people do?

Supervised



Classification



Regression

Un-Supervised



Clustering



Generative models

Supervised

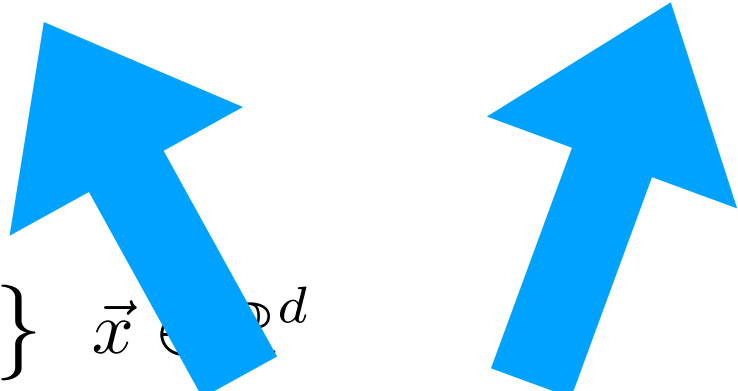
Classification & Regression

Given a dataset with label, find a function that can assign labels to new unlabelled data

Regression

Classification

Labeled data: $\begin{cases} \{\vec{x}_1, \vec{x}_2, \vec{x}_3, \dots, \vec{x}_n\} & \vec{x} \in \mathbb{R}^d \\ \{y_1, y_2, y_3, \dots, y_n\} & y \in \mathbb{R} \text{ or } y \in \mathbb{N} \end{cases}$

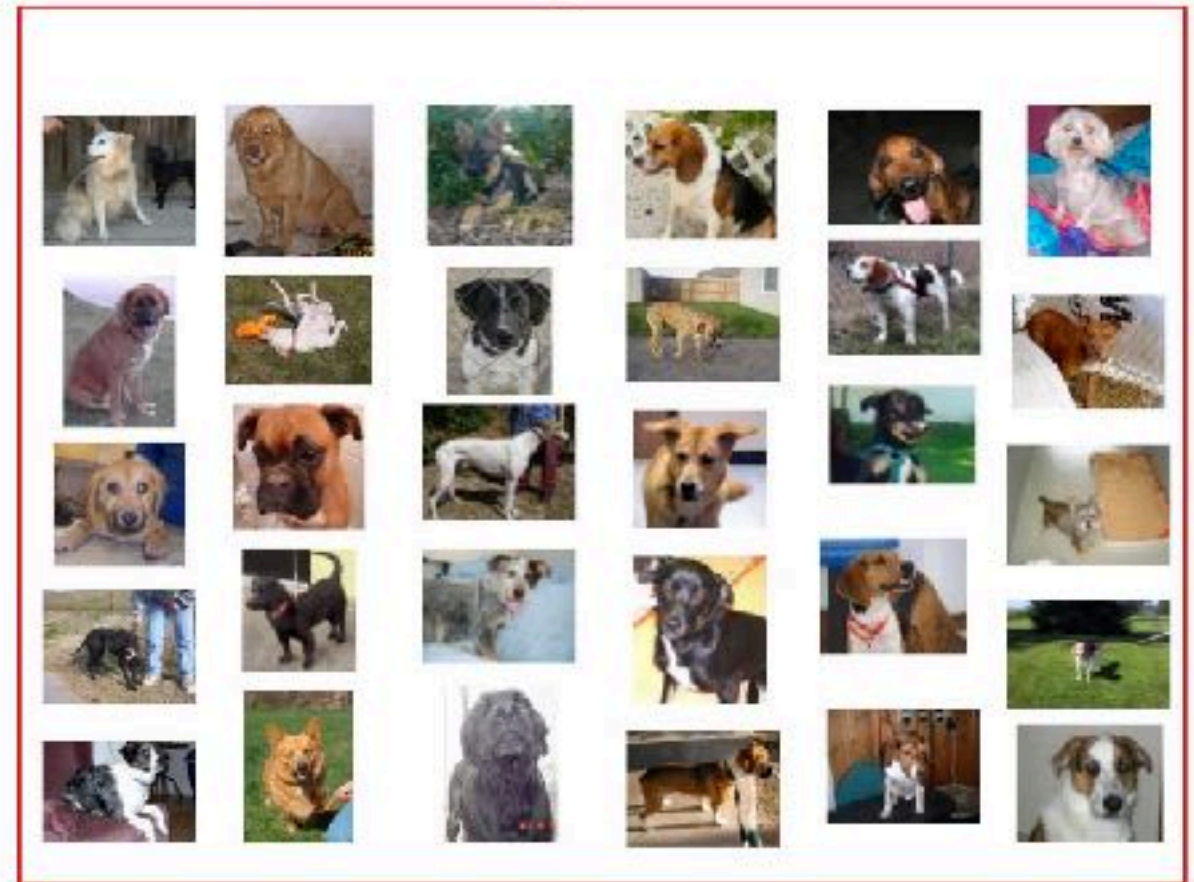


Goal: Find a function $f_W(\vec{x})$ that outputs the right class/value for an object \vec{x}

Cats vs dogs classification

Cats

Dogs

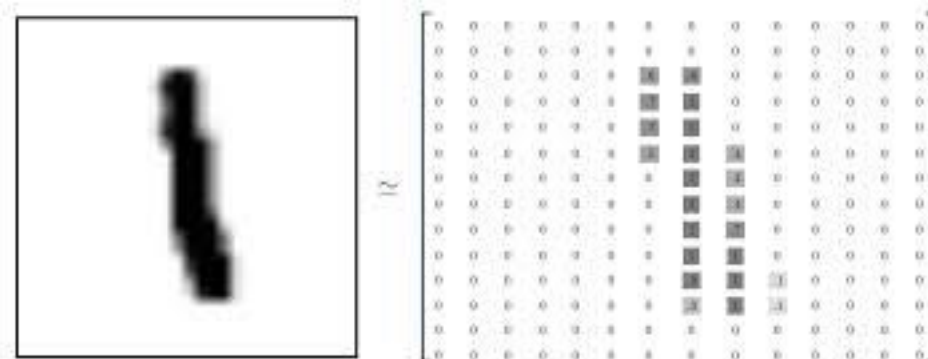


Sample of cats & dogs images from Kaggle Dataset



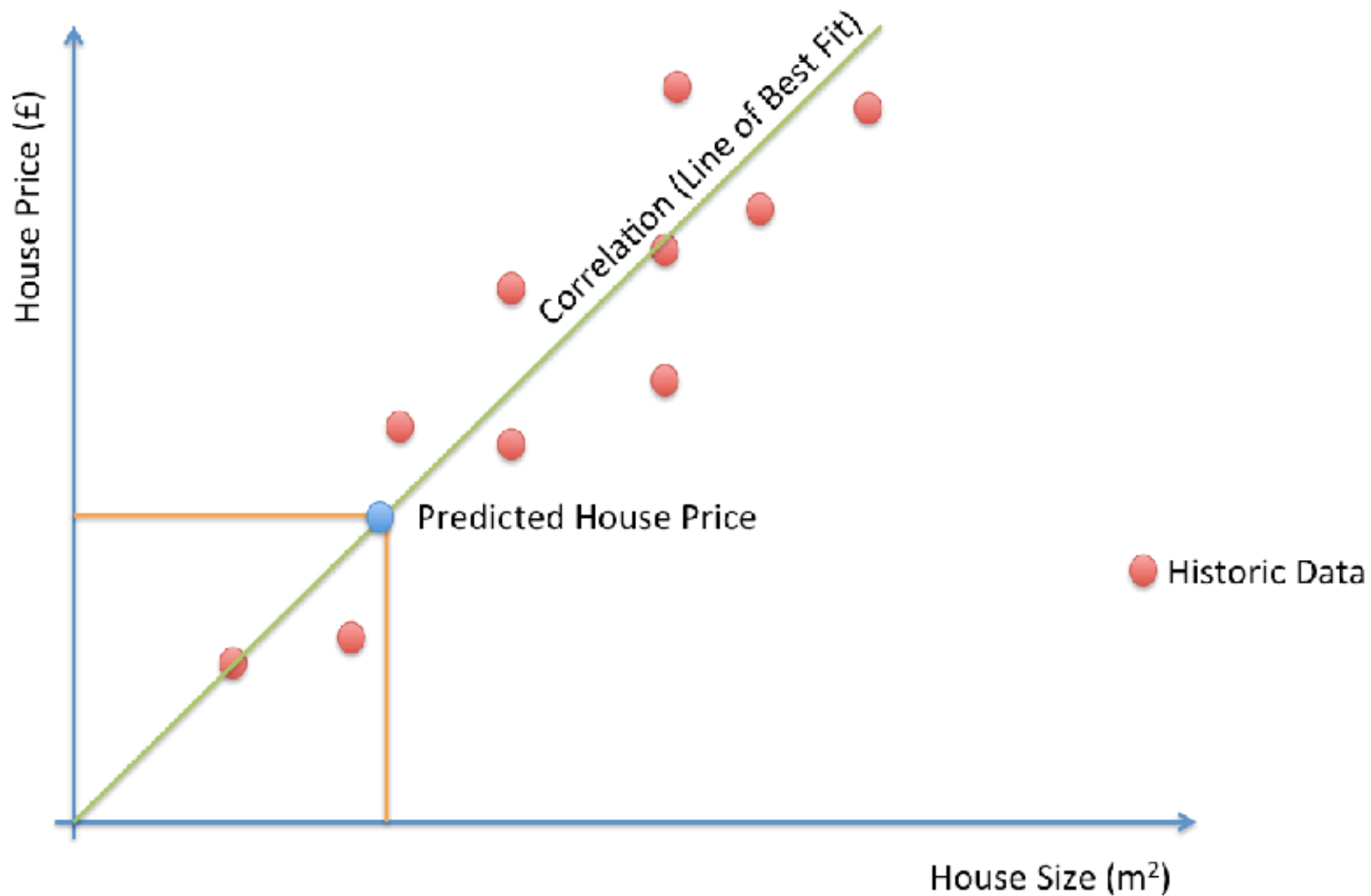
? 0 = dog
1 = cat

MNIST dataset classification



? 0,1,2,3,4,5,6,7,8,9

Regression: predicting house price

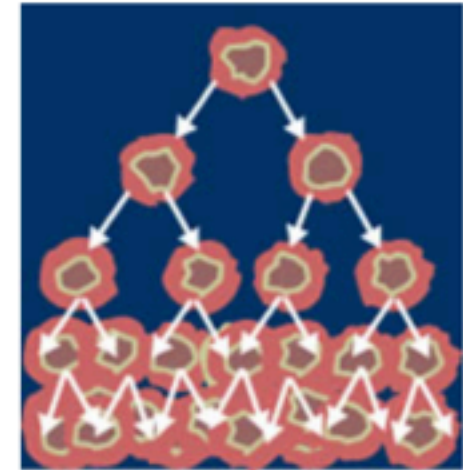


Medical applications for regression

Breast Cancer Wisconsin (Prognostic) Data Set

Download: [Data Folder](#), [Data Set Description](#)

Abstract: Prognostic Wisconsin Breast Cancer Database



Data Set Characteristics:	Multivariate	Number of Instances:	198	Area:	Life
Attribute Characteristics:	Real	Number of Attributes:	34	Date Donated	1995-12-01
Associated Tasks:	Classification, Regression	Missing Values?	Yes	Number of Web Hits:	124725

4-33) Ten real-valued features are computed for each cell nucleus:

- a) radius (mean of distances from center to points on the perimeter)
- b) texture (standard deviation of gray-scale values)
- c) perimeter
- d) area
- e) smoothness (local variation in radius lengths)
- f) compactness ($\text{perimeter}^2 / \text{area} - 1.0$)
- g) concavity (severity of concave portions of the contour)
- h) concave points (number of concave portions of the contour)
- i) symmetry
- j) fractal dimension ("coastline approximation" - 1)

Several of the papers listed above contain detailed descriptions of how these features are computed.

The mean, standard error, and "worst" or largest (mean of the three largest values) of these features were computed for each image, resulting in 30 features. For instance, field 4 is Mean Radius, field 14 is Radius SE, field 24 is Worst Radius.

Values for features 4-33 are recoded with four significant digits.

34) Tumor size - diameter of the excised tumor in centimeters
35) Lymph node status - number of positive axillary lymph nodes observed at time of surgery

8. Missing attribute values:

Lymph node status is missing in 4 cases.

2) Predicting Time To Recur (field 3 in recurrent records)
- Estimated mean error 13.9 months using Recurrence Surface Approximation. (See references (i) and (ii) above)

Medical applications for classification



Dermatologist-level classification of skin cancer

An artificial intelligence trained to classify images of skin lesions as benign lesions or malignant skin cancers achieves the accuracy of board-certified dermatologists.

In this work, we pretrain a deep neural network at general object recognition, then fine-tune it on a dataset of ~130,000 skin lesion images comprised of over 2000 diseases.

A new tool in the box!

news & views

MACHINE LEARNING

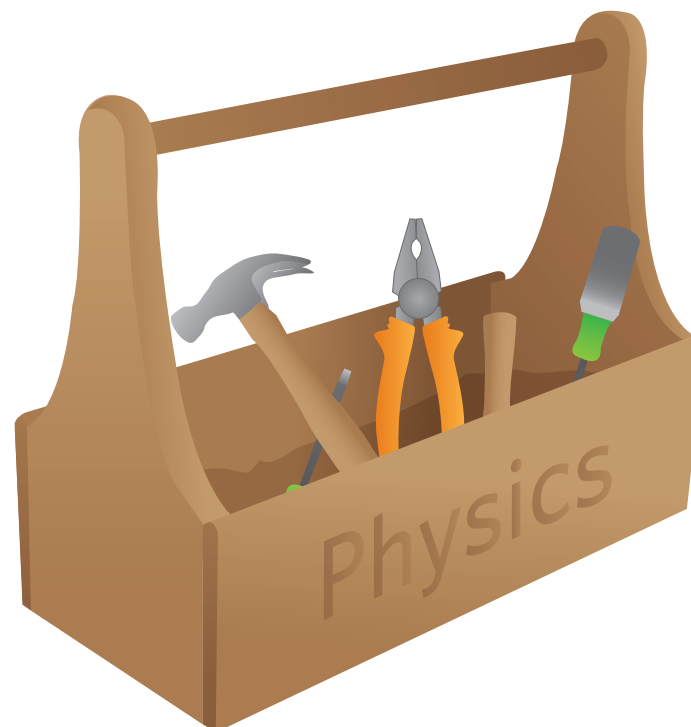
New tool in the box

A recent burst of activity in applying machine learning to tackle fundamental questions in physics suggests that associated techniques may soon become as common in physics as numerical simulations or calculus.

Lenka Zdeborová

The goal of machine learning, broadly speaking, is to design a computer code — the eponymous machine — capable of discovering meaningful structure in data. The last decade saw a game-changing revolution unfold in this field: with the development of deep neural networks¹, tasks that were considered inaccessible to automated learning became possible. This prompted fierce competition in the artificial intelligence market, but it also brought promise to many areas of data-intensive fundamental science — with physics being no exception. Current machine-learning systems are not yet able to divine the laws of general relativity from planetary data, but they are able to reliably recognize human faces, detect objects in photographs and even beat world champions of Go². And now, writing in *Nature Physics*, two groups have used artificial neural networks to recognize different phases of matter and localize associated phase transitions^{3,4}.

Isaac Gornoville and Roger Mella³



It should be stressed that saying one applied machine learning to a given problem is about as generic as saying that one used numerical simulations. It is clear to every researcher in physics that there are many kinds of numerical



learning⁵. Another group recently showed that a support vector machine can be used to classify which particles in a glassy system are susceptible to rearrangement⁶. Decision forests have been used to classify metals from insulators based on the hybridization function, combined with kernel ridge regression to predict correlation functions in many-body physics⁷. For the quantum systems



Higgs Boson Machine Learning Challenge

3.1. The Mathematical Problem

The data for the Challenge consisted of 800,000 fully simulated events provided by the ATLAS Collaboration corresponding to the signal process with Higgs decaying to $\tau^+\tau^-$ and background events from top-antitop and $Z \rightarrow \tau^+\tau^-$. Details of the ATLAS Experiment can be found in Ref. [13]. For each event, 30 numbers were recorded, including “primitive” quantities such as the jet and lepton momenta as well as derived quantities like the missing transverse energy and visible mass.

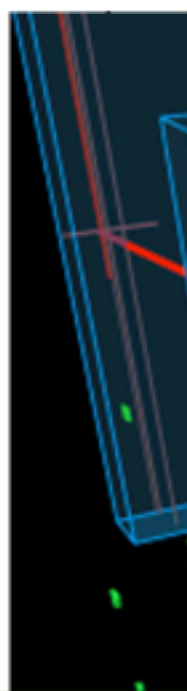
A subsample of 250,000 of these events were labeled accordingly as signal or background. This training sample was used by participants to design an algorithm that could be used to search for the signal process. The remaining 550,000 events were used for testing the performance of the algorithm.

Prizes

About The Sponsors

Timeline

Winners



G. Cowan, G. Rousseau

¹ Laboratoire de l'Accélérateur Linéaire, Orsay, France

² Department of Physics, Royal Holloway, University of London, UK

³ Laboratoire de Recherche en Informatique, Orsay, France

⁴ ChaLearn, California, USA

E-mail: g.cowan@rhul.ac.uk

Abstract. The Higgs Machine Learning Challenge was an open data analysis competition that took place between May and September 2014. Samples of simulated data from the ATLAS Experiment at the LHC corresponding to signal events with Higgs bosons decaying to $\tau^+\tau^-$ together with background events were made available to the public through the website of the data science organization Kaggle (kaggle.com). Participants attempted to identify the search region in a space of 30 kinematic variables that would maximize the expected discovery significance of the signal process. One of the primary goals of the Challenge was to promote communication of new ideas between the Machine Learning (ML) and HEP communities. In this regard it was a resounding success, with almost 2,000 participants from HEP, ML and other areas. The process of understanding and integrating the new ideas, particularly from ML into HEP, is currently underway.

A structural approach to relaxation in glassy liquids

S. S. Schoenholz^{1*†}, E. D. Cubuk^{2†}, D. M. Sussman¹, E. Kaxiras² and A. J. Liu^{1*}

In contrast with crystallization, there is no noticeable structural change at the glass transition. Characteristic features of glassy dynamics that appear below an onset temperature, T_0 (refs 1–3), are qualitatively captured by mean field theory^{4–6}, which assumes uniform local structure. Studies of more realistic systems have found only weak correlations between structure and dynamics^{7–11}. This raises the question: is structure important to glassy dynamics in three dimensions? We answer this question affirmatively, using machine learning to identify a new field, ‘softness’ which characterizes local structure and is strongly correlated with dynamics. We find that the onset of glassy dynamics at T_0 corresponds to the onset of correlations between softness (that is, structure) and dynamics. Moreover, we construct a simple model of relaxation that agrees well with our simulation results, showing that a theory of the evolution of softness in time would constitute a theory of glassy dynamics.

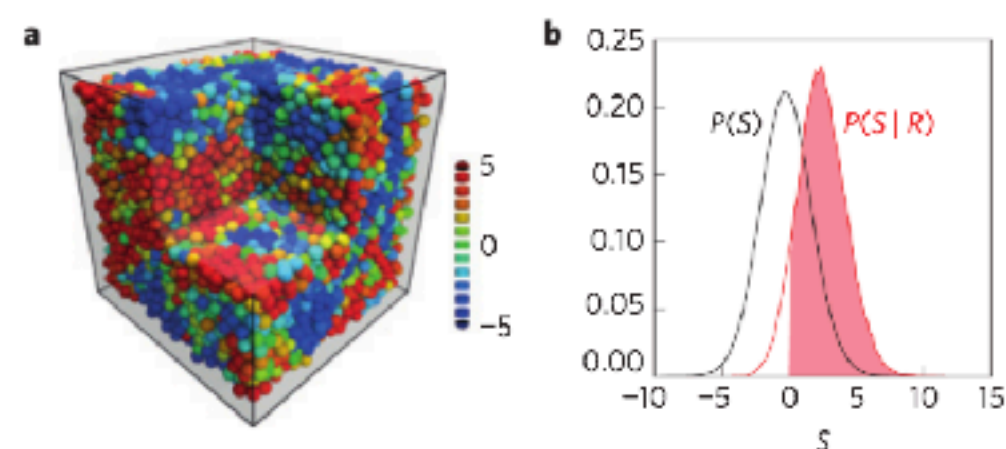


Figure 1 | The characteristics of the softness field. a, A snapshot of the system at $T=0.47$ and $\rho=1.20$ with particles coloured according to their softness from red (soft) to blue (hard). **b**, The distribution of softness of all particles in the system (black) and of those particles that are about to rearrange (red). 90% of the particles that are about to rearrange have $S > 0$ (shaded region). None of the data included in this plot were in the training set.

What are the task solved by machine learning people?

Supervised



Classification



Regression

Un-Supervised



Clustering



Generative models

Most data are actually unlabelled!

Un-Supervised Clustering

Clustering

Find « groups » in unlabelled datas

[illegible]

Un-Supervised

Clustering

<https://challengedata.ens.fr/en/home>



HOME

CHALLENGES

PRESENTATIONS

FAQ

SIGN UP

LOGIN

CONTACT

21

Challenge providers

21

Projects

2428

Participants

Supported by the CFM chair on Data Sciences at ENS



Announcement: New Season in 2018



remini7®

Cluster actor faces from TV show
In this challenge, we provide you with faces extracted from 20 episodes of a TV show. The goal is to gather, for each movie, all the faces that belong to the

PHYSICAL REVIEW X

Highlights Recent Subjects Accepted Authors

Open Access

Hierarchical Block Structures and Hig in Large Networks

Tiago P. Peixoto

Phys. Rev. X **4**, 011047 – Published 24 March 2014

IMDb

Find Movies, TV shows, Celebrities and more...

Movies, TV
& Showtimes

Celebs, Events
& Photos

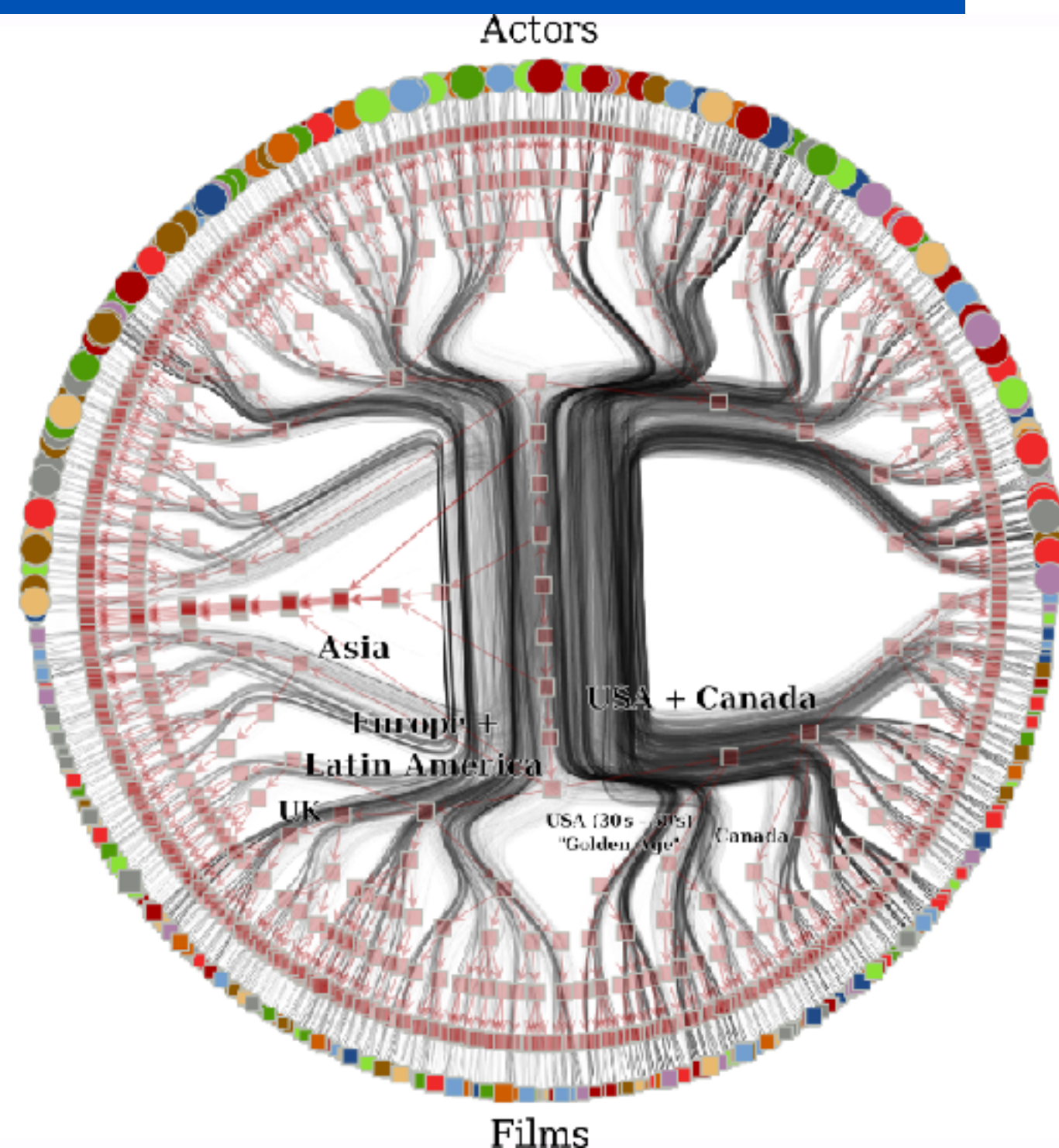
New
Com

IMDb Datasets

Subsets of IMDb data are available for access to customers for person use. You can hold local copies of this data, and it is subject to our terms of use. For more information, please refer to the [Non-Commercial Licensing](#) and [copyright/license](#) and verify the data.

Data Location and Access Requirements

Files are located in the AWS S3 bucket named **imdb-datasets** and can be accessed programmatically. The data is refreshed daily. This is a Requester-Pays S3 bucket, and the requester accessing data from this bucket is responsible for the data transfer and request costs. For details on the charges, please refer to <https://aws.amazon.com/s3/pricing/>.



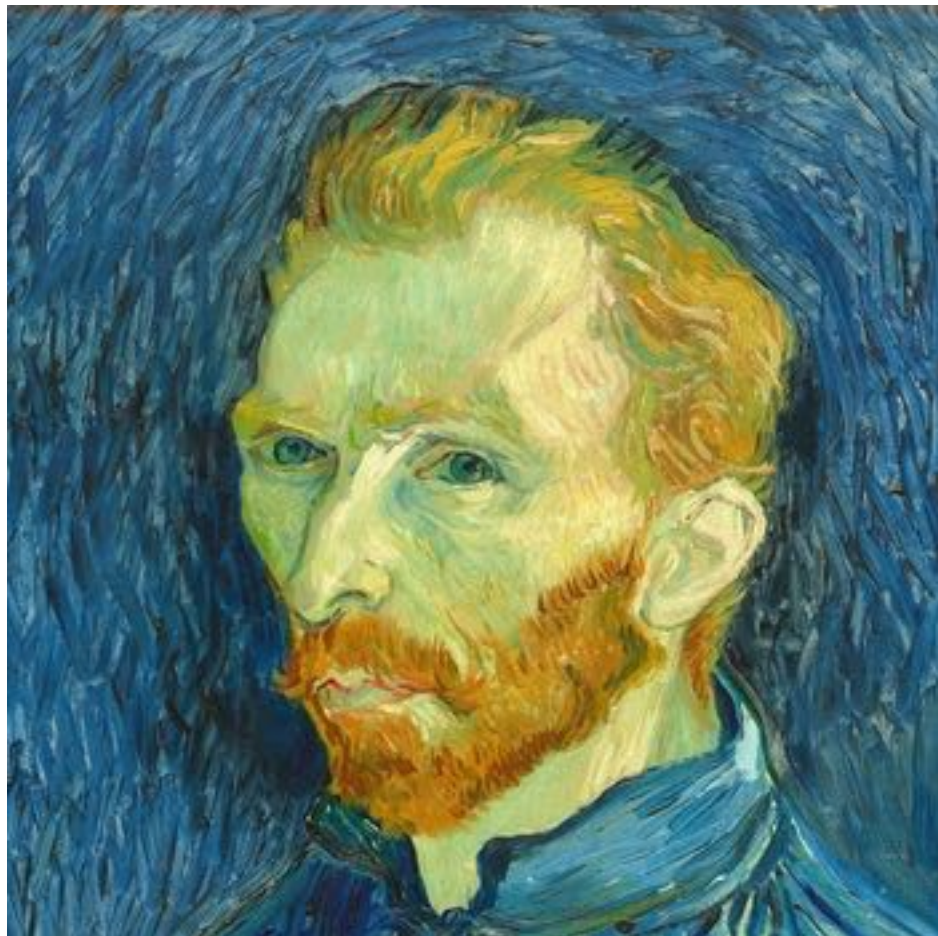
Un-Supervised

Generative models





=



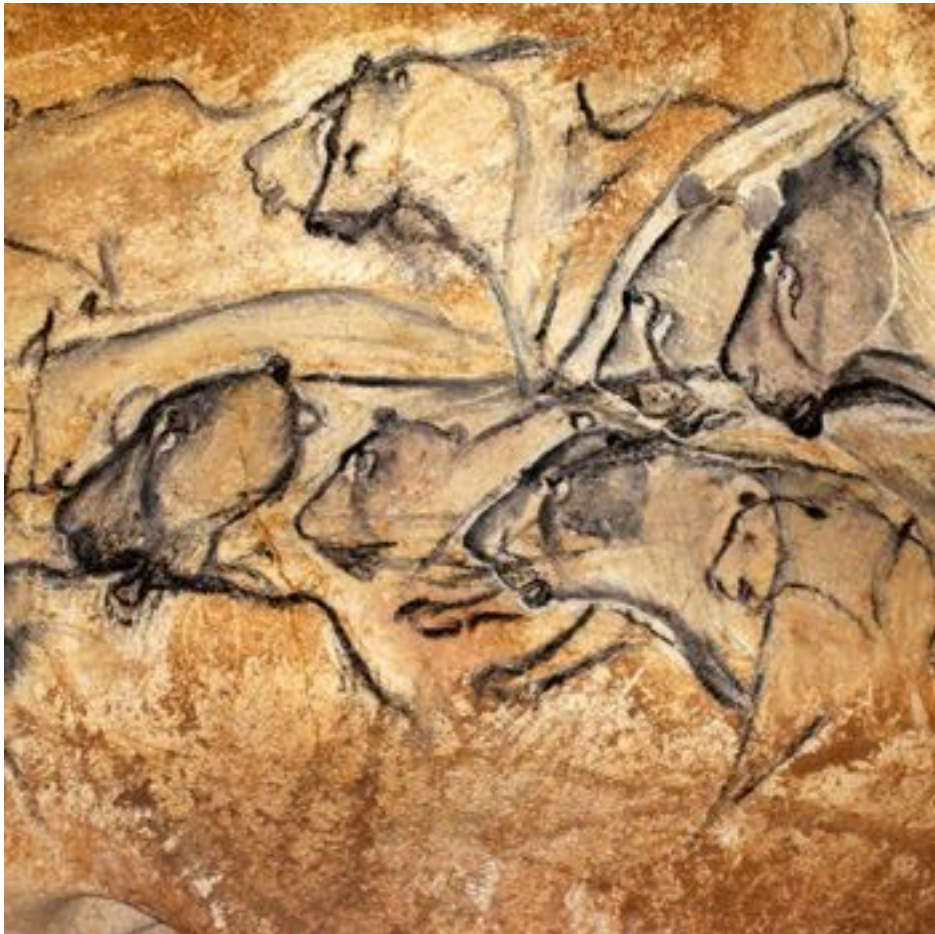
+







=



+





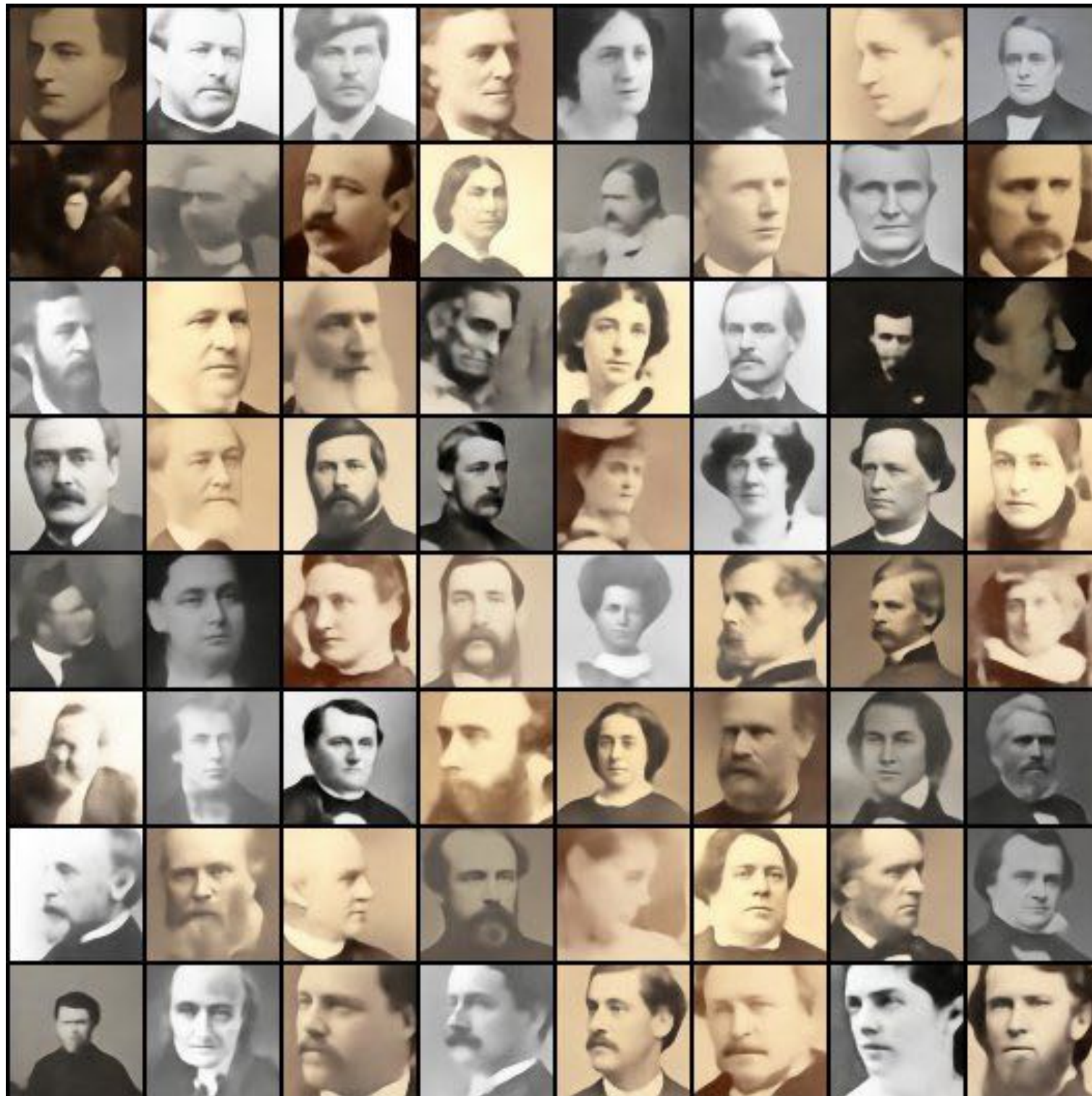
Peter Leonard

@pleonard

Suivre

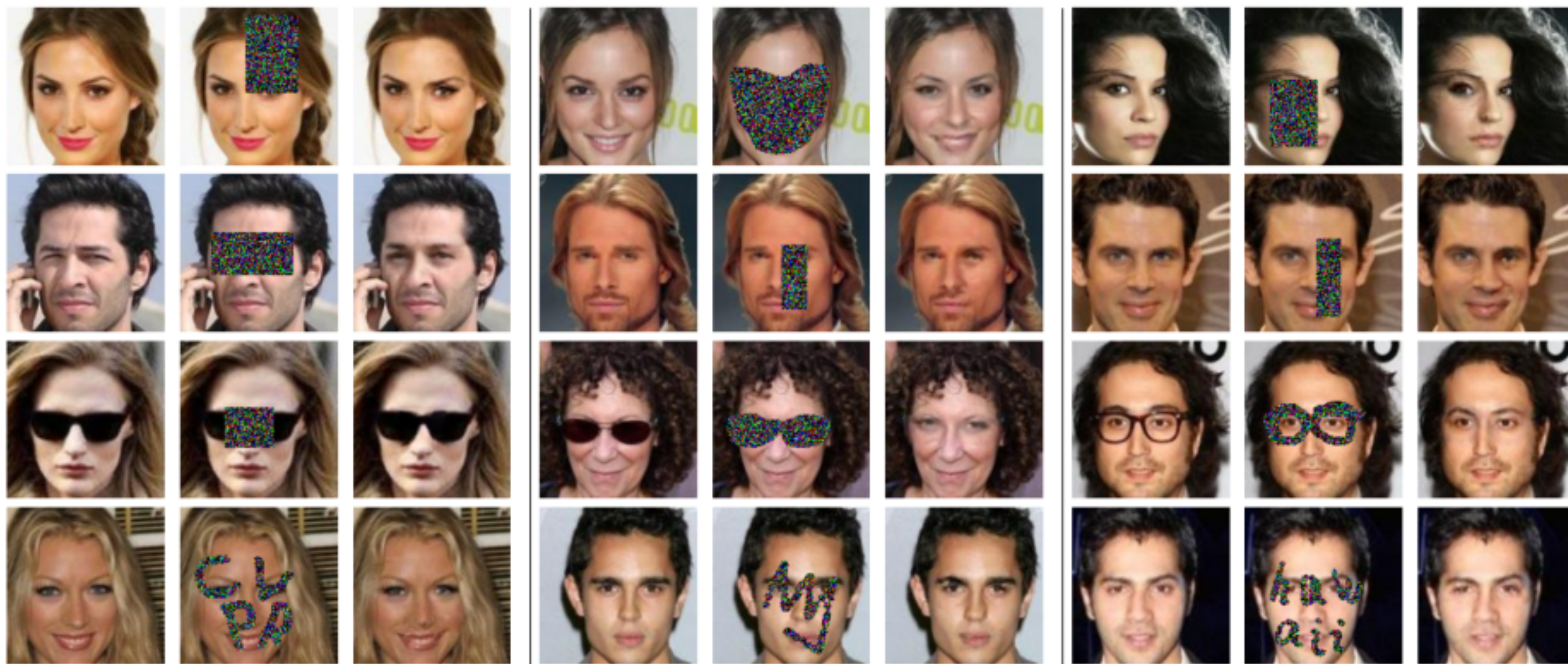


Spent a weekend training a Generative Adversarial Network on 25k 19th century portraits; results seem good. These people have never existed:



Many applications to generative models

Cleaning data: image completion



<https://medium.com/@Synced/generative-face-completion-cfa85cc4e835>

Many applications to generative models

Science

Home News Journals Topics Careers

Log in | My account

SHARE

RESEARCH ARTICLE



Solving the quantum many-body problem with artificial neural networks

Giuseppe Carleo^{1,*}, Matthias Troyer^{1,2}

+ See all authors and affiliations

Science 10 Feb 2017:
Vol. 355, Issue 6325, pp. 602-606
DOI: 10.1126/science.aag2302



Peer Reviewed
← see details

Today's lesson

Supervised problems

Let's introduce some of the machine learning vernacular...

Supervised machine learning

Labeled data:
$$\begin{cases} \{\vec{x}_1, \vec{x}_2, \vec{x}_3, \dots, \vec{x}_n\} & \vec{x} \in \mathbb{R}^d \\ \{y_1, y_2, y_3, \dots, y_n\} & y \in \mathbb{R} \quad \text{or} \quad y \in \mathbb{N} \end{cases}$$

Goal: Find a function $f_W(\vec{x})$ that outputs the right class/value for an object \vec{x}

$f_W(\vec{x})$ has many parameters, denoted $W \in \mathbb{R}^p$

Ideal : Find $f_W(\vec{x})$ such that the prediction error is minimal among all unseen vectors

Supervised machine learning

Labeled data:
$$\begin{cases} \{\vec{x}_1, \vec{x}_2, \vec{x}_3, \dots, \vec{x}_n\} & \vec{x} \in \mathbb{R}^d \\ \{y_1, y_2, y_3, \dots, y_n\} & y \in \mathbb{R} \quad \text{or} \quad y \in \mathbb{N} \end{cases}$$

Goal: Find a function $f_W(\vec{x})$ that outputs the right class/value for an object \vec{x}

$f_W(\vec{x})$ has many parameters, denoted $W \in \mathbb{R}^p$

Ideal : Find $f_W(\vec{x})$ such that the prediction error is minimal among all unseen vectors

Instead : Find $f_W(\vec{x})$ such that the prediction error is minimal among all vectors in the dataset

Example for cats and dog classification



Ideally: find a function

$$f_W(\vec{x})$$

That minimise the error on
all possible images of cats
and dogs!

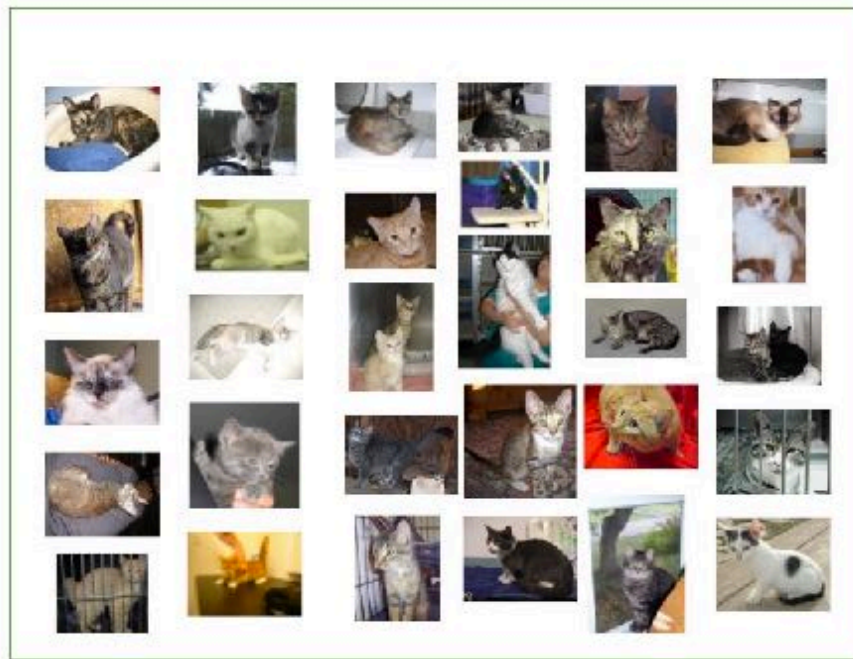
Define the loss as $\text{loss}(F_W(.), \vec{x}_i, y_i) = (F_W(\vec{x}_i) - y_i)^2$

We want to minimise the
population risk defined as $\mathcal{R}_{\text{pop}}(F_W(.)) = \mathbb{E}_{\text{population}}(F_W(\vec{x}) - y)^2$

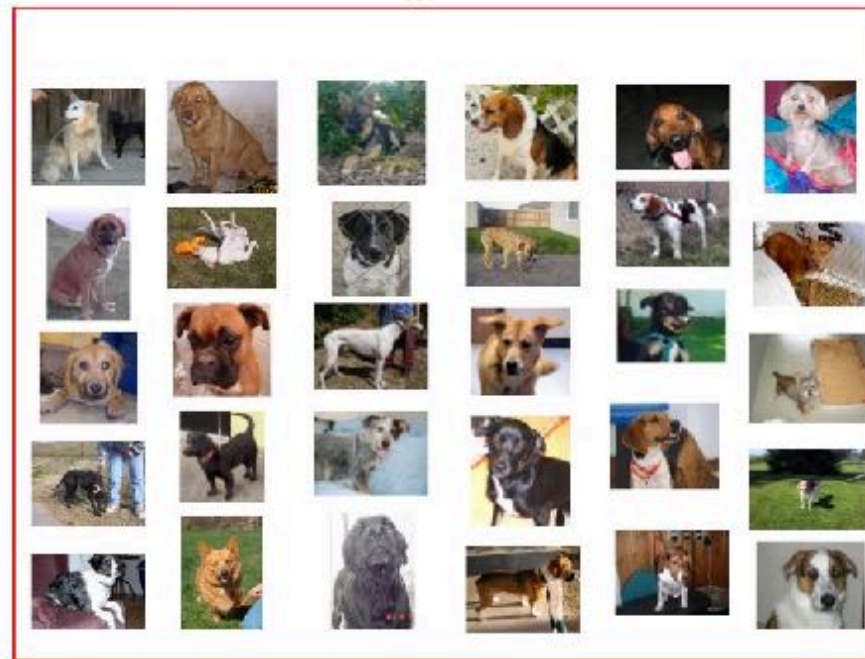
Cannot do this: I do not have access to ALL images in the
universe, and most of them are not labeled anyway

Example for cats and dog classification

Cats



Dogs



Sample of cats & dogs images from Kaggle Dataset

Instead: find a function

$$f_W(\vec{x})$$

That minimise the error on the images on the dataset

Define the loss as

$$\text{loss}(F_W(.), \vec{x}_i, y_i) = (F_W(\vec{x}_i) - y_i)^2$$

We want to minimise the empirical risk defined as

$$\mathcal{R}_{\text{empirical}}(F_W(.)) = \frac{1}{N} \sum_i^{\text{dataset}} (F_W(\vec{x}_i) - y_i)^2$$

We are actually minimizing the wrong function (but we have no choice)

The workhorse: Empirical Risk Minimisation

Minimize

$$\mathcal{R}_{\text{empirical}}(W) = \frac{1}{N} \sum_i^{\text{dataset}} \ell(W, (\vec{x}_i), y_i)$$

Rationale: it should be close to

$$\mathcal{R}_{\text{population}}(W) = \mathbb{E} \ell(W, (\vec{x}), y)$$

Questions?

Statistics: How close is the empirical risk to the population risk ?

Computational: How do we minimise the empirical risk?

Statistics: How close is the empirical risk to the population risk ?

Generalization bound

Theorem (Vapnik, Chervonenkis, 1968; . . .)

Under conditions [omitted], with high probability

$$\sup_{W \in \theta} |\mathcal{R}_n^{\text{emp}}(W) - \mathcal{R}^{\text{pop}}(W)| \leq C_{\text{st}} \sqrt{\frac{VC \log n}{n}}$$

VCd dimension = capacity of your classifier

Additional term

Error on unseen data	=	Error on training data	+	Decay with number of training sample Increase with complexity of function f
				
Generalisation error		Training error		

Statistics: How close is the empirical risk to the population risk ?



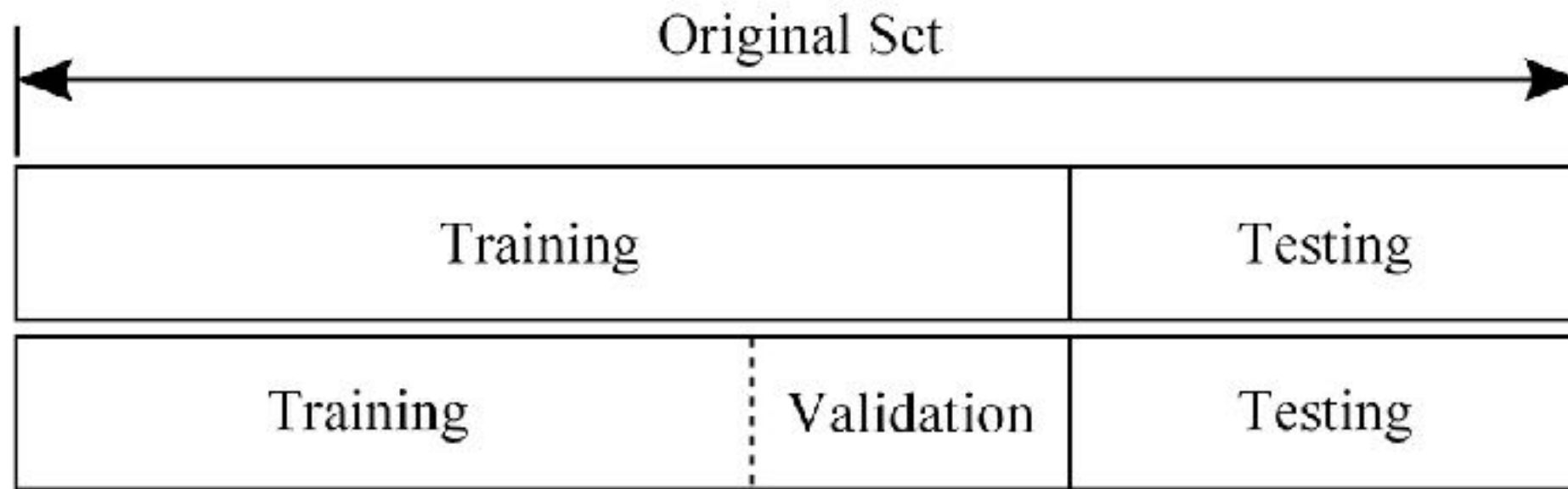
Those results are cool and beautiful, but they have no practical consequence. No one uses generalization bounds.

Yann LeCun, Facebook IA

$$\underbrace{\text{Error on unseen data}}_{\text{Generalisation error}} = \underbrace{\text{Error on training data}}_{\text{Training error}} + \underbrace{\text{Decay with number of training sample} + \text{Increase with complexity of function } f}_{\text{Additional term}}$$

In practice

Divide the labelled set into training, validation and testing sets



- * Training set: used to train the classifier
- * Validation set (optional): choose between different methods, fine-tune parameters,
- * Testing set: predict the generalization error

No cheat: do not use the test set to train your algorithm!

No cheat: do not use the test set to train your algorithm!



A View from **Tom Simonite**

Why and How Baidu Cheated an Artificial Intelligence Test

Machine learning gets its first cheating scandal.

Login

Topics+

The Download

Magazine

Quora

Search for questions, people, and topics

Quora uses cookies to improve your experience. Read more

Rewrite

Regression (statistics)

Statistics (academic discipline)

How much reliable is the paper "Stacked Approximated Regression Machine"? Are we going to rewrite DL frameworks?

1 Answer



Zhaojun Zhang, Trained as a Bayesian for two years

Answered Sep 10, 2016

From Arxiv, the paper has been withdrawn: [A Simple Deep Learning Approach](#).

So, it is unlikely that we are going to rewrite DL frameworks based on this paper.

328 Views

Promoted by QuantInsti

Become a successful algo & quant trader in 6 months.

Acquire the knowledge, tools & techniques used by traders in the real world.

Start now at quantinsti.com

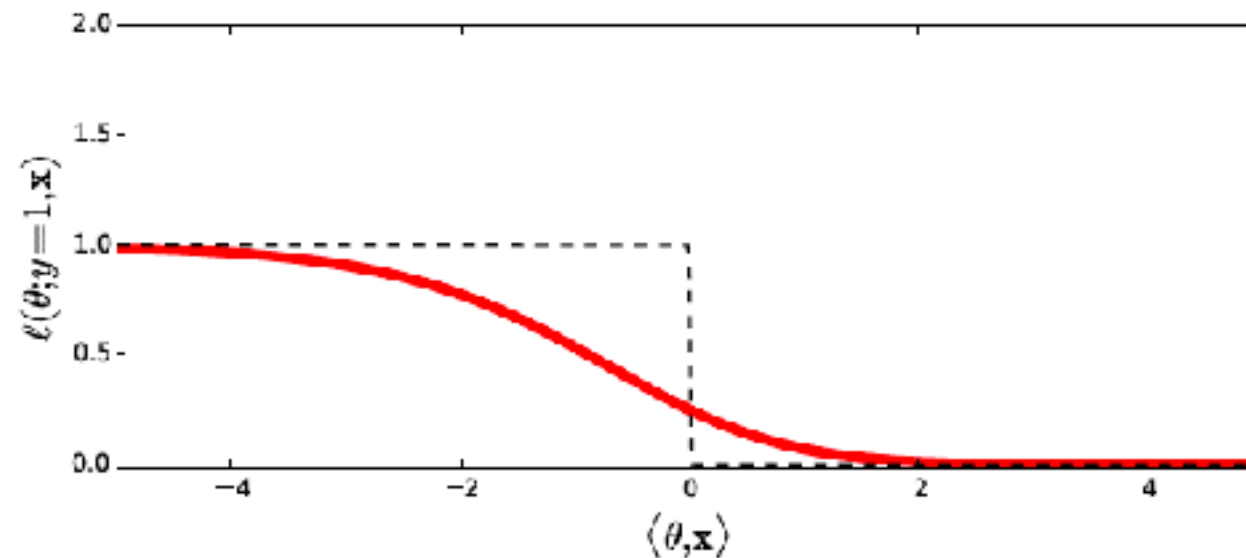
Questions?

Statistics: How close is the empirical risk to the population risk ?

Computational: How do we minimise the empirical risk?

Computational: How do we minimise the empirical risk?

Example 1: binary classification with the perceptron (Rosenblath 1958)



$$\mathbf{z}_i = (y_i, \mathbf{x}_i), \quad y_i \in \{0, 1\}, \quad \mathbf{x}_i \in \mathbb{R}^d$$

$$\hat{R}_n(\mathbf{w}) = \frac{1}{n} \sum_{i=1}^n (y_i - \sigma(\langle \mathbf{w}, \mathbf{x}_i \rangle))^2,$$

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

Computational: How do we minimise the empirical risk?

Example 2: Multi-layered networks)

1 layer neural network: $\mathbf{w} \in \mathbb{R}^d$

$$\ell(\mathbf{w}; y, x) = (y - \sigma(\langle \mathbf{w}, x \rangle))^2$$

2 layers neural network: $w_1 \in \mathbb{R}^d$, $\mathbf{W}_2 \in \mathbb{R}^{d \times d}$

$$\ell(\theta; y, x) = (y - \sigma(w_1^T \sigma(\mathbf{W}_2 x)))^2$$

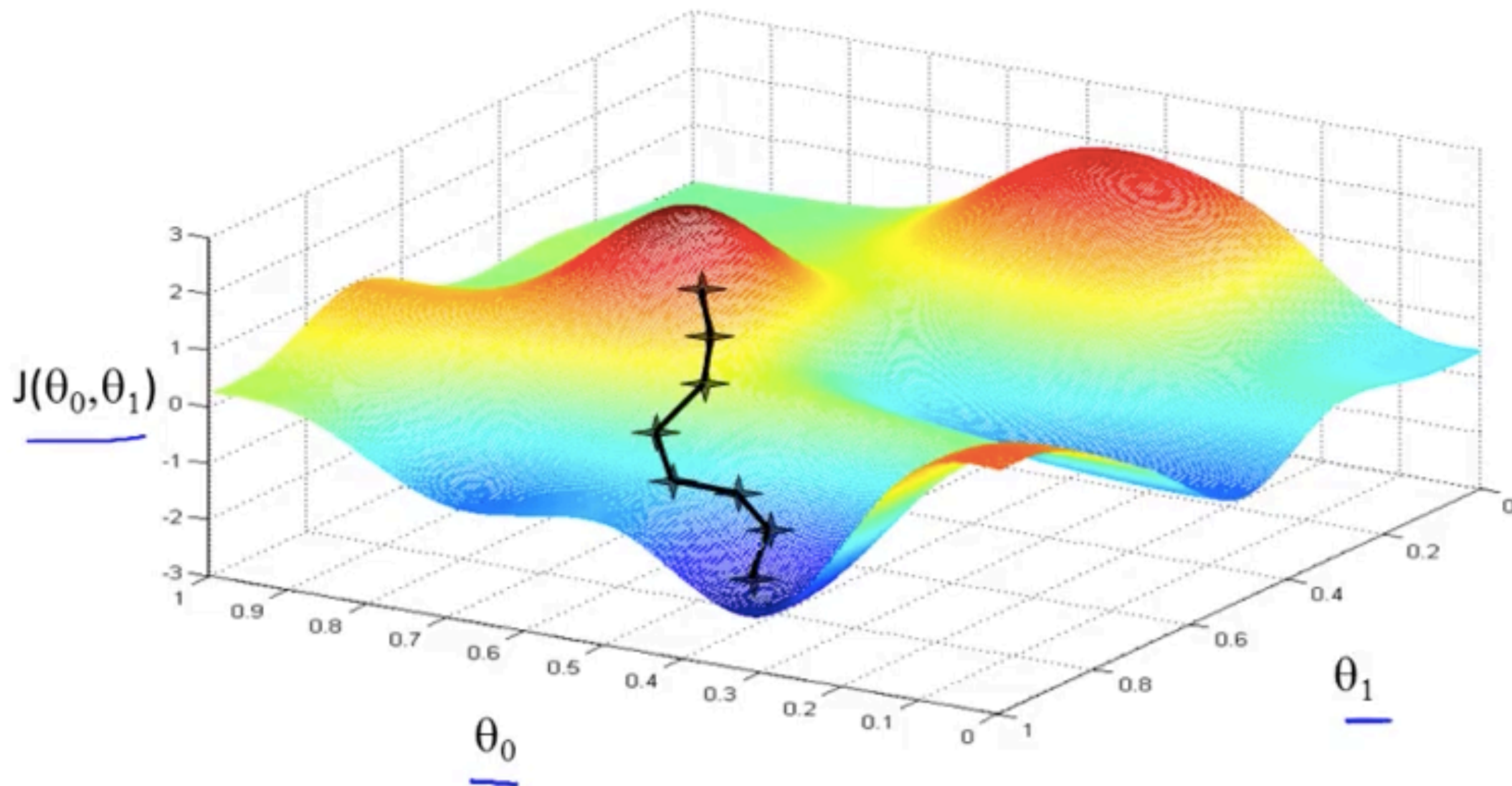
3 layers neural network: $w_1 \in \mathbb{R}^d$, $\mathbf{W}_2 \in \mathbb{R}^{d \times d}$, $\mathbf{W}_3 \in \mathbb{R}^{d \times d}$

$$\ell(\theta; y, x) = (y - \sigma(w_1^T \sigma(\mathbf{W}_2 \sigma(\mathbf{W}_3 x))))^2$$

Minimising the cost function by gradients descent

$$\vec{\theta}^{t+1} = \vec{\theta}^t - \gamma \nabla R(\vec{\theta}^t)$$

If γ small enough, converge to a (possible local) minima



Minimising the cost function by gradients descent

$$\vec{\theta}^{t+1} = \vec{\theta}^t - \gamma \nabla R(\vec{\theta}^t)$$

If γ small enough, converge to a (possible local) minima

Standard (or "batch") gradient descent

Compute the gradient by averaging the derivative of the loss is the entire training set

$$\vec{\theta}^{t+1} = \vec{\theta}^t - \gamma \sum_i \frac{1}{N} \nabla l(\vec{\theta}^t; \vec{x}_i, y_i)$$

Minimising the cost function by gradients descent

$$\vec{\theta}^{t+1} = \vec{\theta}^t - \gamma \nabla R(\vec{\theta}^t)$$

If γ small enough, converge to a (possible local) minima

Stochastic (or « mini-batch») **gradient descent**

Compute the gradient by averaging the derivative of the loss in a mini-batch

1) Divide the training set into P batch of size B

2) For each batch, do

$$\vec{\theta}^{t+\frac{1}{P}} = \vec{\theta}^t - \gamma \sum_{i \text{ in mini batch}} \frac{1}{B} \nabla l(\vec{\theta}^t; \vec{x}_i, y_i)$$

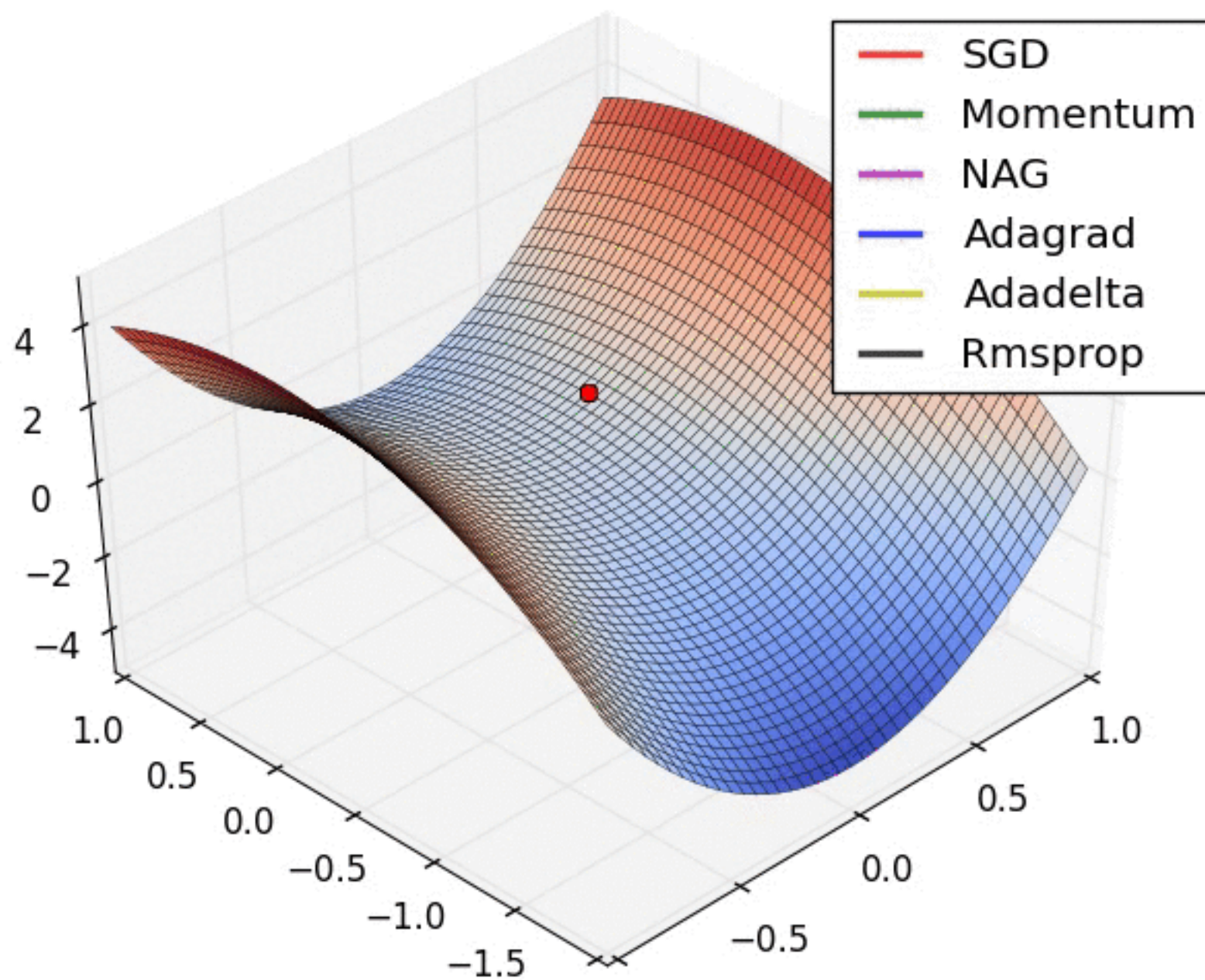
3) One « epoch » (t->t+1) means running the algorithm through all mini-batches

Why Mini-batch gradient descent?

$$\vec{\theta}^{t+\frac{1}{P}} = \vec{\theta}^t - \gamma \sum_{i \text{ in mini batch}} \frac{1}{B} \nabla l(\vec{\theta}^t; \vec{x}_i, y_i)$$

- The model update frequency is higher than batch gradient descent: **faster** and **memory efficient** (*often nothing else is actually possible*)
- Effective noise in the dynamics helps optimization/regularization: works better than full batch minimisation in practice

Many mini-batch algorithms (but we shall discuss them later)



That's all for today

Supervised learning

Empirical risk minimisation

Training, Validation, and Test sets

Gradient descent

Mini-batch Gradient descent