Machine Intelligence

inter-disciplinary perspective

Andrey Ustyuzhanin [1,2]

[1] NRU HSE

[2] NUST MISiS

МИСиС

LAMBDA • HSE

May 19, 2020

Remarkable examples of AI technologies

- Human-level playing in computer games (Go, StarCraft, Dota 2) and winning world's Go champion Lee Sedol by Google AlphaGo;
- Understanding and generation of human-readable and understandable texts;
- Recognition and generation of images indistinguishable from photos by the naked eye;
- Simulation of complicated physics processes;
- Controlling complicated real-time systems like quantum qubits;
- Controlling autonomous vehicles in populated regions;
- … and many others.

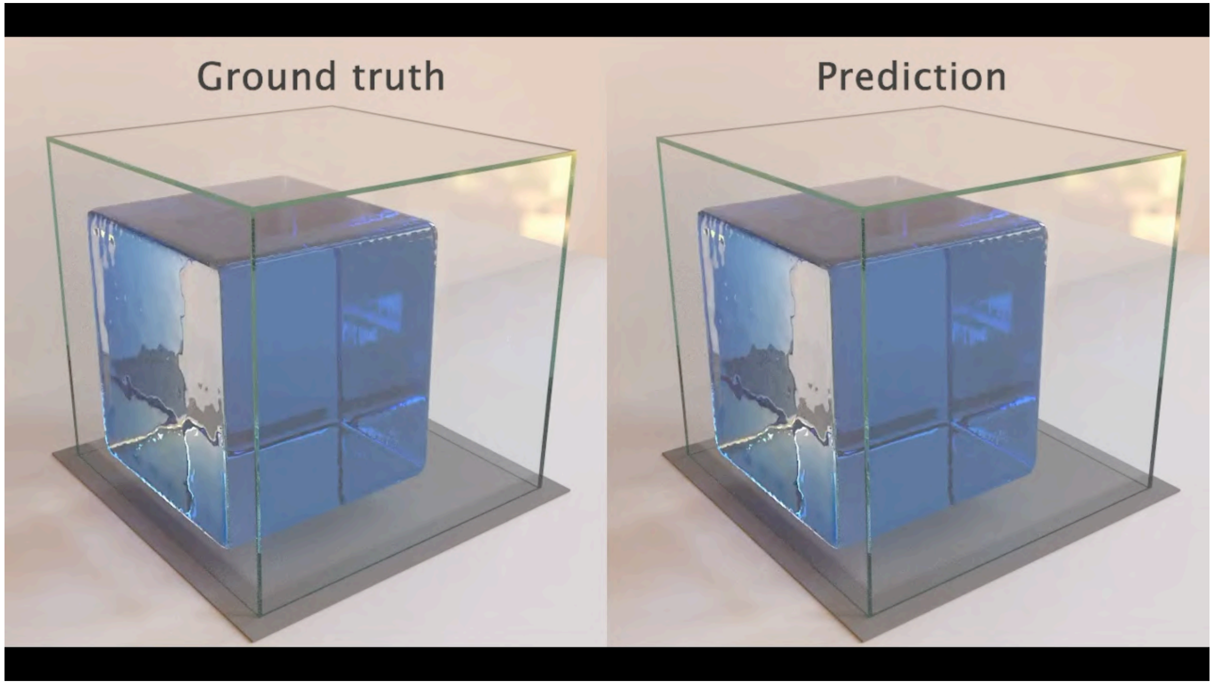
City street view simulation

This time lap shows the original scene (left), segmentation map (bottom right) and neural-network produced scene (right) by NVIDIA.

https://www.youtube.com/watch?v=ayPqjPekn7g

https://arxiv.org/abs/1808.06601

Fluid dynamics computation



- The time lap shows properly simulated water volume evolution (left) and simulated evolution by the trained neural network (right) with "Graph Network-based Simulators» (Alvaro Sanchez-Gonzalez et al.)

Tesler’s theorem:
“AI is whatever hasn’t been done yet”

Larry Tesler

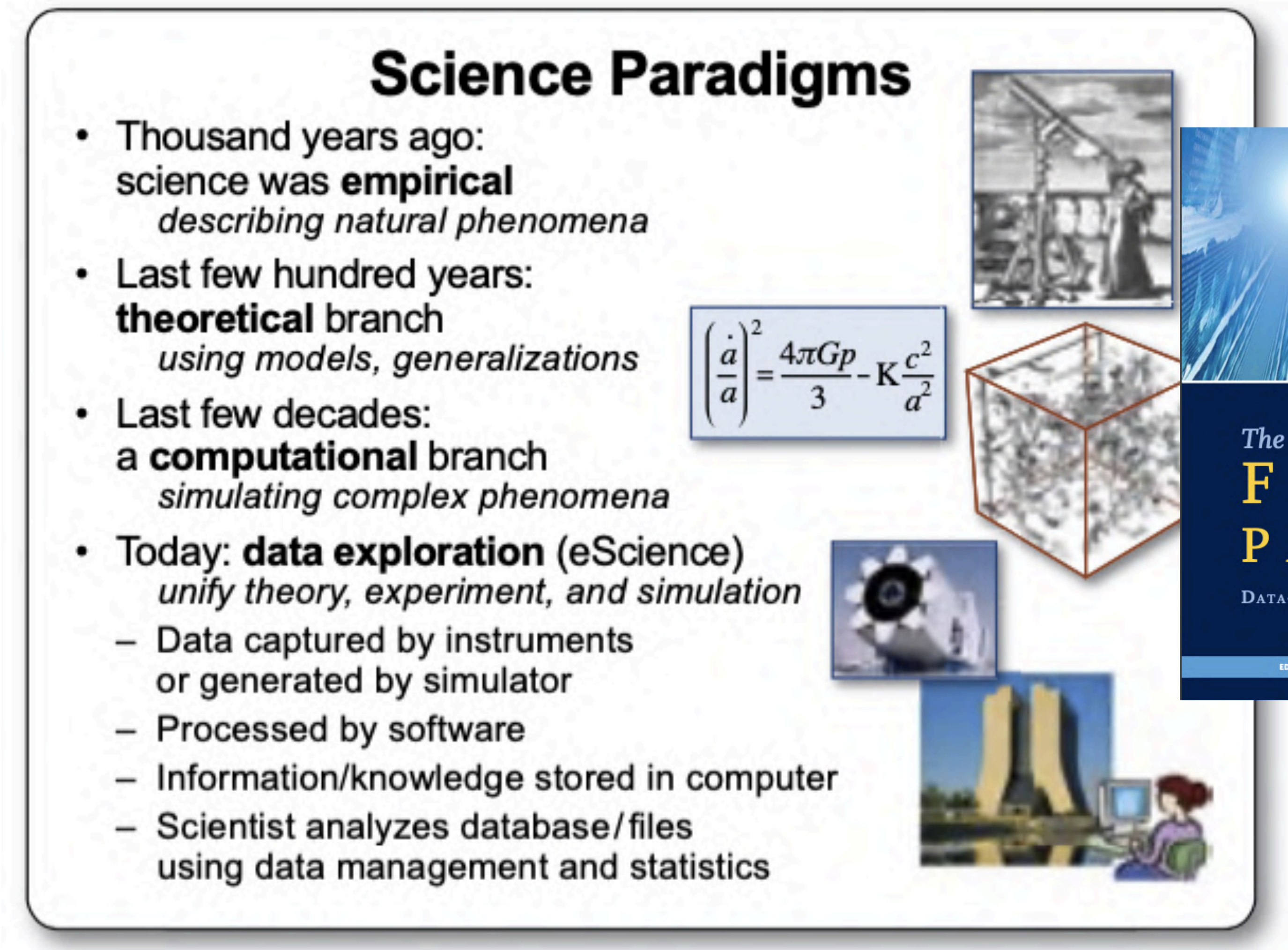
Machine Intelligence (MI)

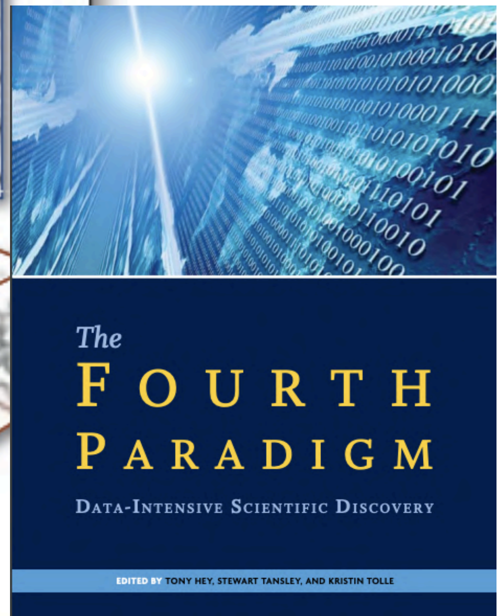
Interdisciplinary area that embraces ML and bridges the gap between ML and domain specifics by iterative procedure:

- Data collection from experiment or software simulation
- Prior choice and Hypothesis formulation
- Algorithm family selection from ML world (Decision Tree, Convolutional Neural Networks, Flows, etc.)
- Training of the algorithm using the data collected
- Validation of the trained algorithm
- Production deployment

Abridged History of science

Jim Gray vision, 2009





link

Empirical Science Questions

- How can we navigate using stars?
- Does the sun rotate around the Earth or vice versa?
- Which body does fall faster?
- What are the causes of solar eclipse?
- Can we estimate time of the next eclipse?
- How to describe motion of the moon and the planets?
- Is Earth flat?

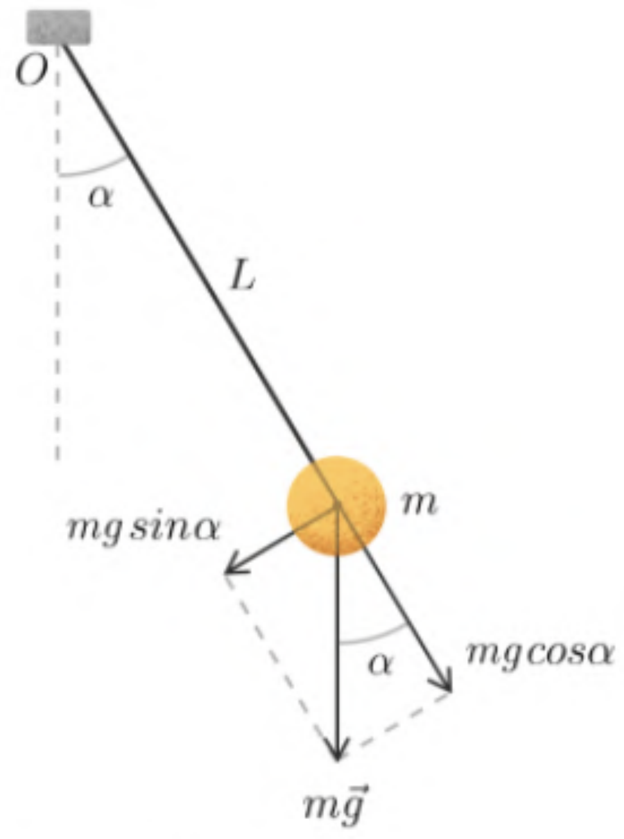
Theoretical branch: differential equations

- Developed by Newton, Leibniz
- At every moment of time t, we can express the dependency of angular acceleration ε

$$\varepsilon = \frac{d^2\alpha}{dt^2} = \frac{M}{I}$$

$$\frac{d^2\alpha}{dt^2} = \frac{mgL\sin\alpha}{mL^2} = -\frac{g\sin\alpha}{L}, \Rightarrow \frac{d^2\alpha}{dt^2} + \frac{g}{L}\sin\alpha = 0$$

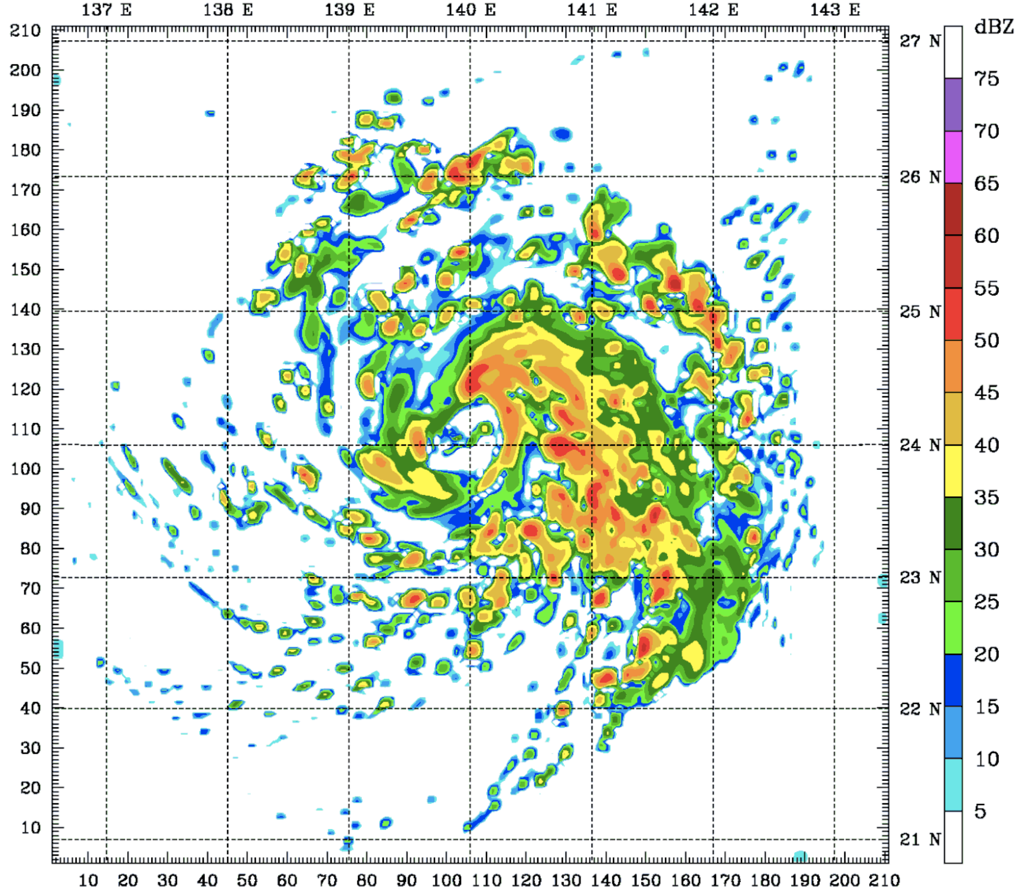
$$\frac{d^2\alpha}{dt^2} + \frac{g}{L}\alpha = 0 \text{ or } \frac{d^2\alpha}{dt^2} + \omega^2\alpha = 0, \text{ where } \quad \omega = \sqrt{\frac{g}{L}}$$



$$T = \frac{2\pi}{\omega} = 2\pi\sqrt{\frac{L}{g}}$$

Computational branch: computer simulation

- Computes the evolution of mathematical models using machines;
- Especially useful when closed-form solution is not available
  - weather forecasting, earth simulator, flight simulator, molecular protein folding, and so on.
- Requires special math methods;
- Blooms with computing power availability.



Forward and Inverse problems

- Forward: from given initial system parameters, get the observable state
- Inverse: from the observable state, get hidden parameters
  - No single solution
  - No straightforward way to compute
  - But if one can approximate evolution of a system by some differentiable surrogate, it might profit from methods of Machine Learning
  - Systems for probabilistic programming: Stan, PyMC3, pyro, Tensorflow Probability (ex Edward) or pyprob.

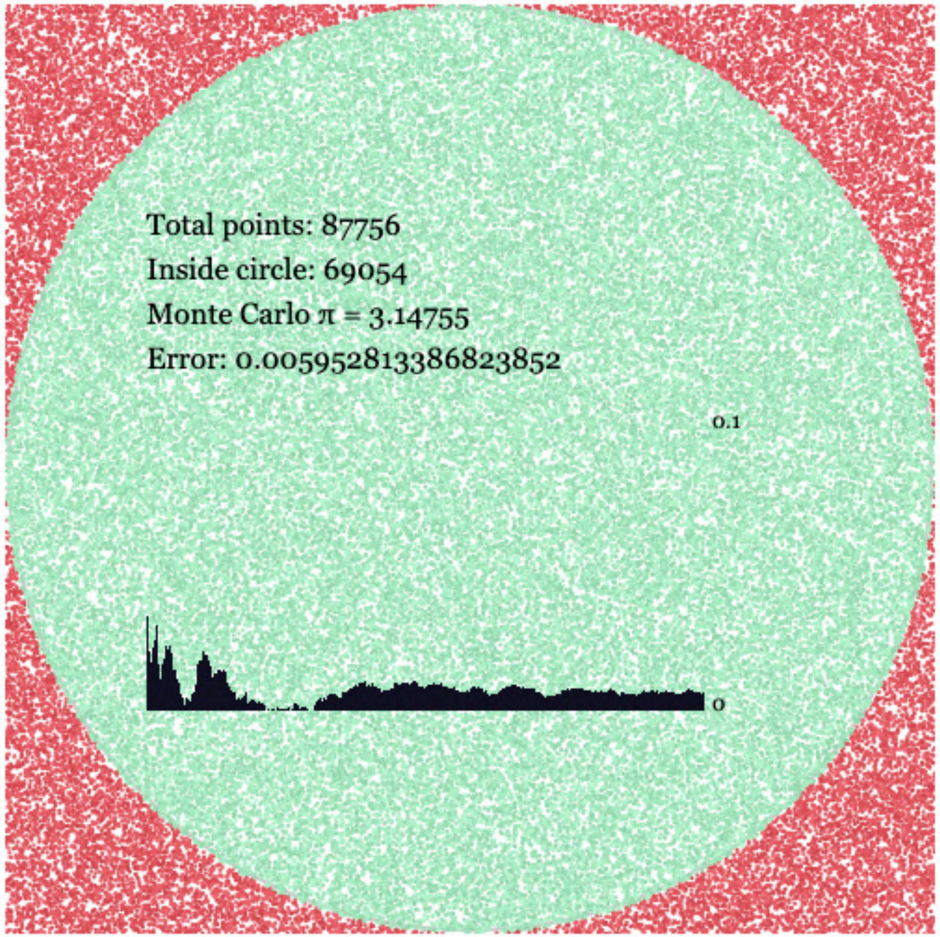
Monte Carlo Method

- Instead of accurately computing all the outcome probabilities, one can combine the randomness from different sources to replicate the overall system dynamics
- E.g. compute the area of a circle:

```
import numpy as np

def pi_MC(n):
 assert n > 0, "argument should be positive"
 x = np.random.rand(n)
 y = np.random.rand(n)
 n_c = np.count_nonzero(x**2 + y**2 <= 1)
 return 4 * n_c / n

print pi_MC(1000)
```



https://en.pelican.study/static/bundles/demonstrations/pi/index.html

Data-driven science

Main boosting factors

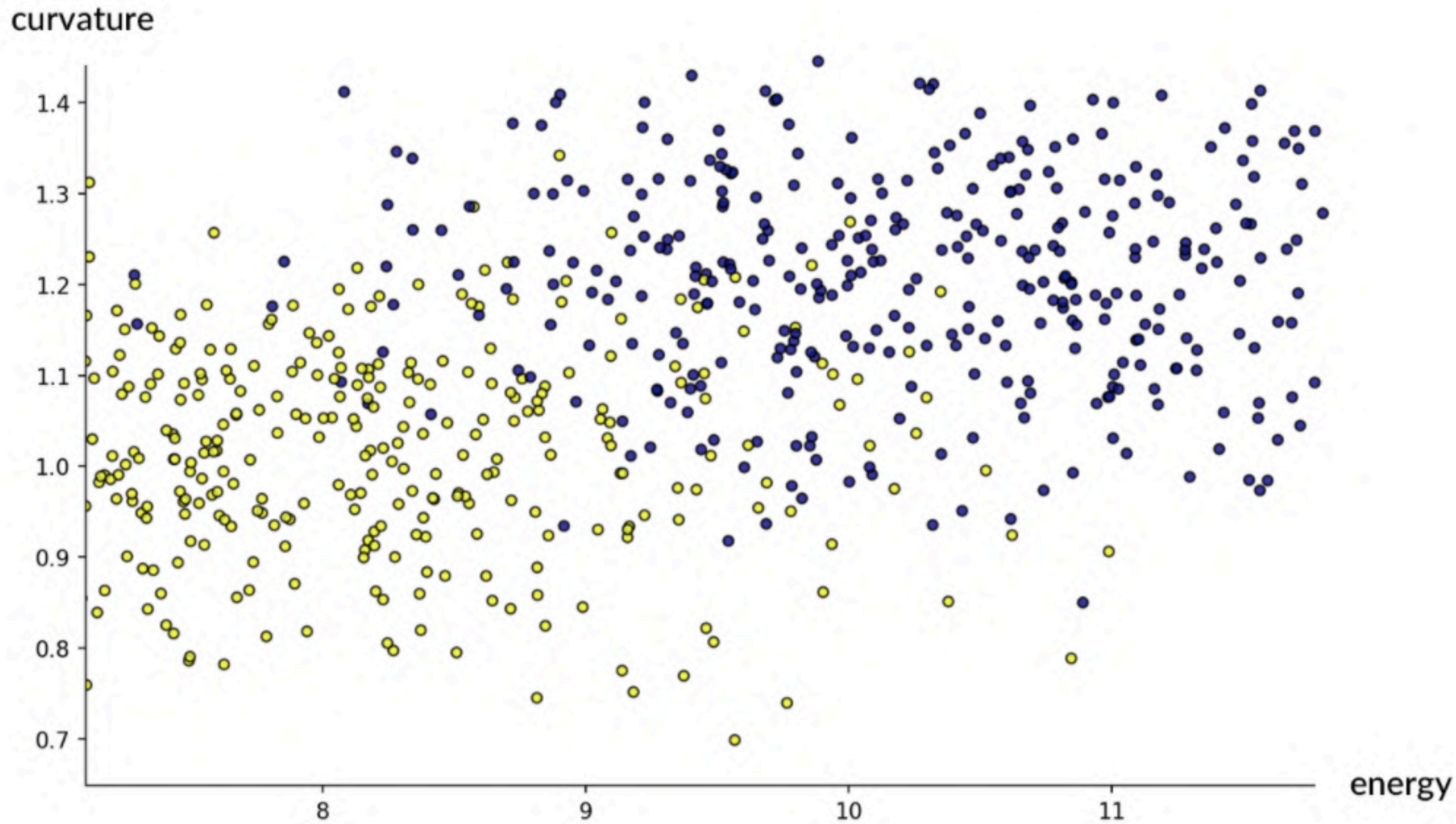
- Data deluge
  - Experiments
  - Industry
  - Simulation
- Computational power
  - Moore's law
- Sophisticated (meta—level) algorithms

A Dataset for particle classification

- Data sample
  - Features
  - Labels
- Features
  - Type: **float**, integer, …
  - Distributions
- Labels, given from outside
  - **Boolean**, integers, real vectors, …

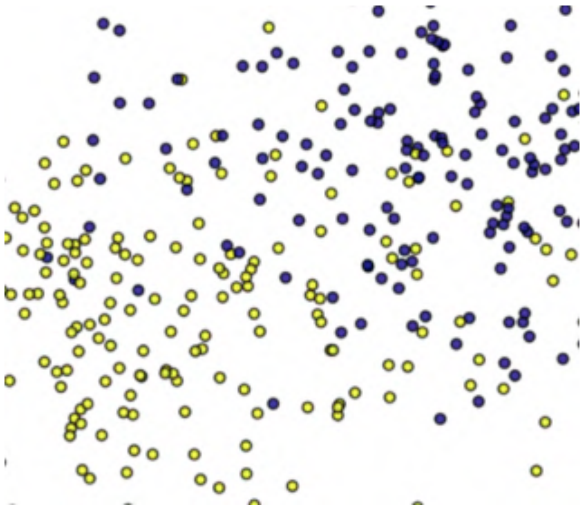
	Energy	Curvature	Class label
0	10.122077	1.226283	0.0
1	7.760199	1.062012	1.0
2	10.989290	0.906222	1.0
3	8.759292	1.096391	1.0
4	10.778759	1.182548	0.0

Two types of particles: muons and electrons

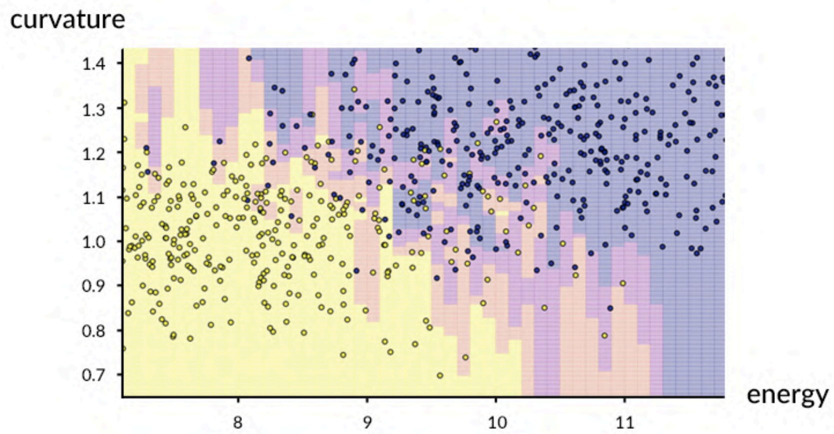


K-Nearest neighbors

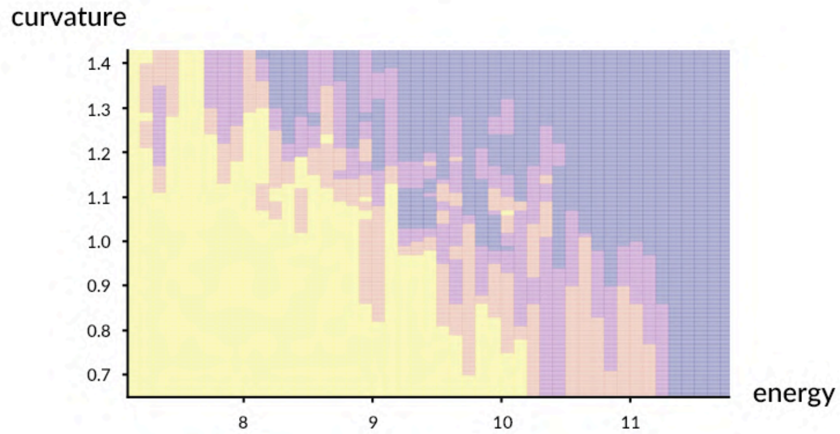
- K-nearest neighbors (or k−NN for short) is a classification algorithm that decides by
  - For every given feature vector F representing unknown particle
  - k−NN looks at the K nearest neighbors from the training sample and
  - counts the fraction of electrons and muons among the neighbors.



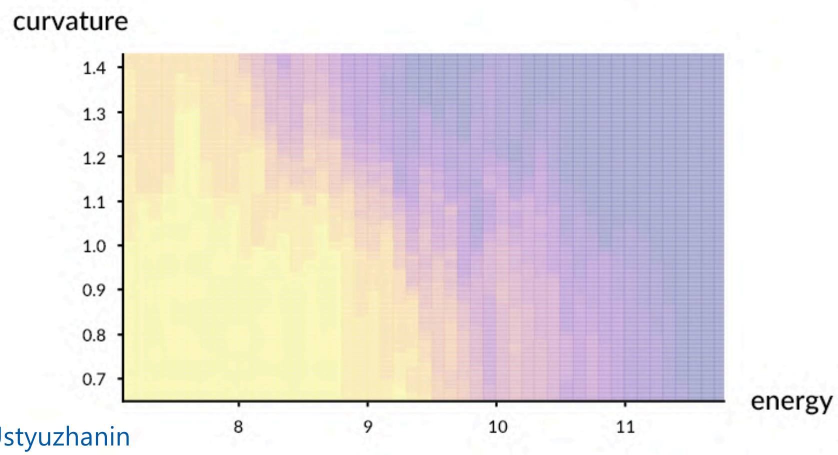
K-Nearest Neighbors (K = 3)



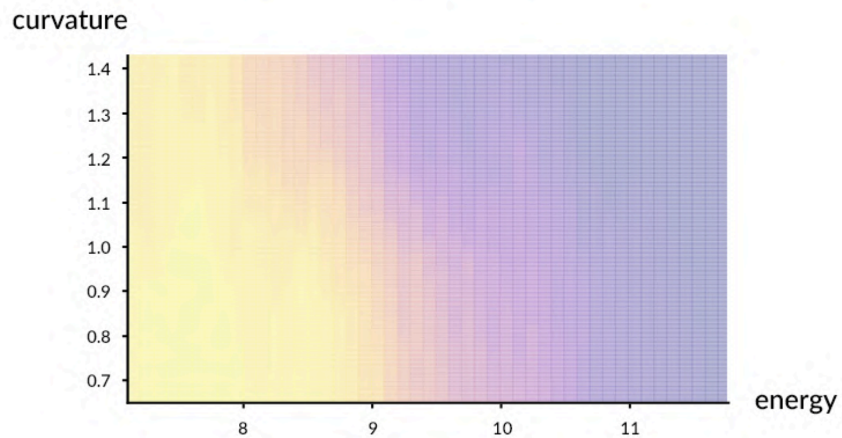
K-Nearest Neighbors (K = 3)



K-Nearest Neighbors (K = 15)



K-Nearest Neighbors (K = 50)



Taxonomies of machine learning algorithms

- By the information needed to make decision:
  - Supervised - uses labelled sample for training;
  - Unsupervised - exploits distance between different objects and doesn't need labels;
  - Reinforced learning - the ground truth becomes available to the algorithm during interaction with environment. Examples include playing computer games, controlling robots or more complicated mechanisms.
- By the type of decision:
  - Classification - attributing an object to one of given classes;
  - Regression - estimation of real number (or vector) from given features;
  - Policy search - building a control policy, e.g. strategy of moves for a game or actuator control;
  - Segmentation - selection of a region belonging to specific class object inside input data;
  - Generative - learn how to generate objects of certain kind, e.g. images of human faces or cars or even painting style.

Taxonomies of machine learning algorithms (2)

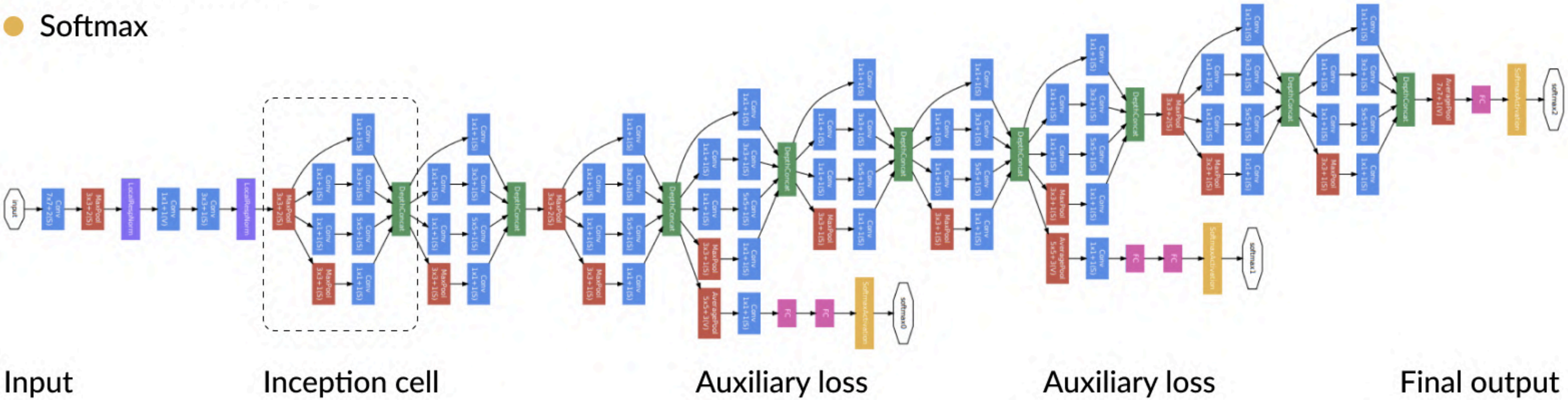
By the type of input objects (features) an algorithm can deal with:

- tabular representation;
- 2D or 3D image;
- text;
- time series;
- graphical data.

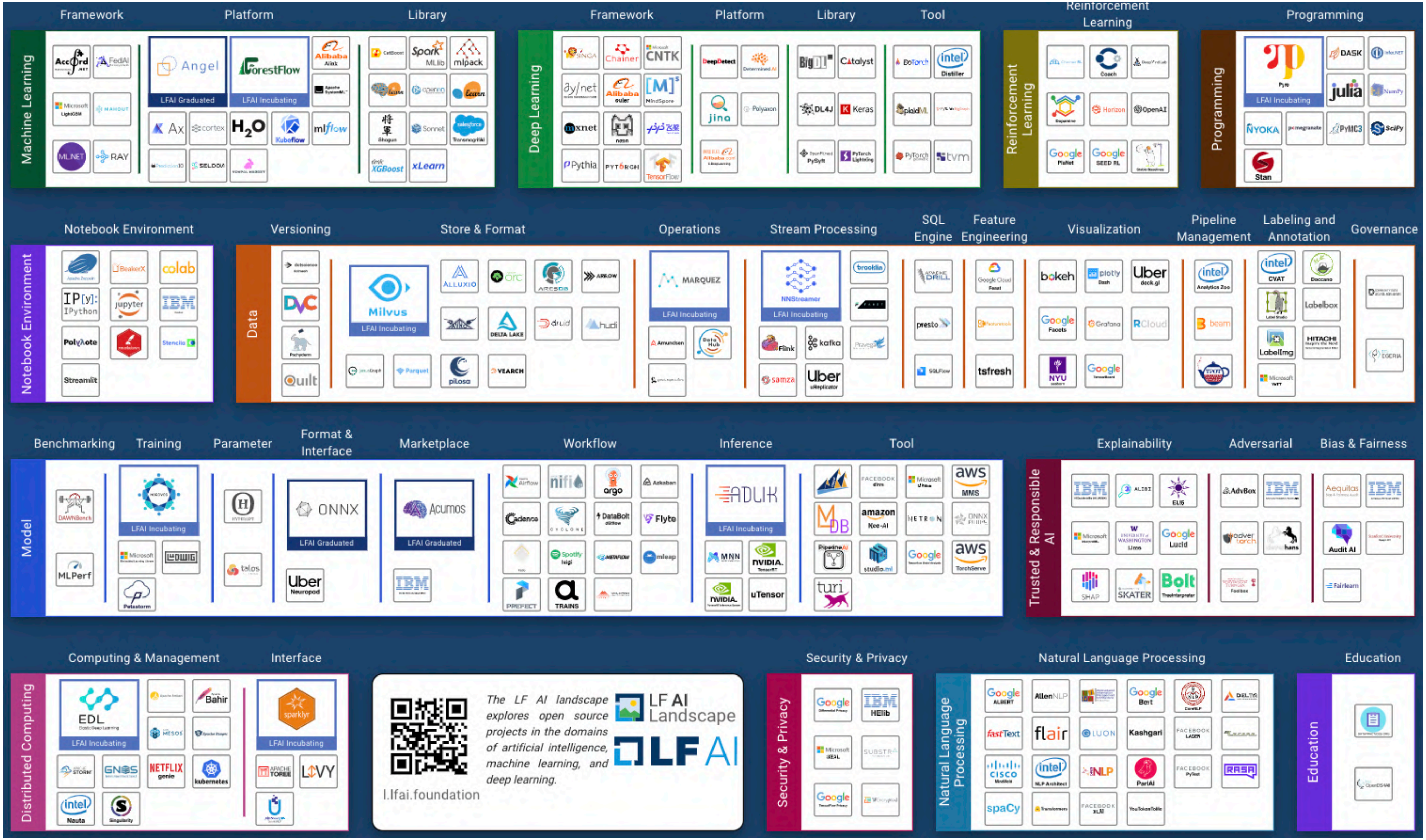
For example, Naive Bayes algorithm is supervised classification algorithm that easily deals with tabular data. Interestingly to note that the algorithm builds a generative model $P(F|C)$ behind the scenes for estimation of likelihood for different classes.

Deep Learning

- Convolution
- Max pooling
- Channel concatenation
- Channel-wise normalization
- Fully-connected layer
- Softmax

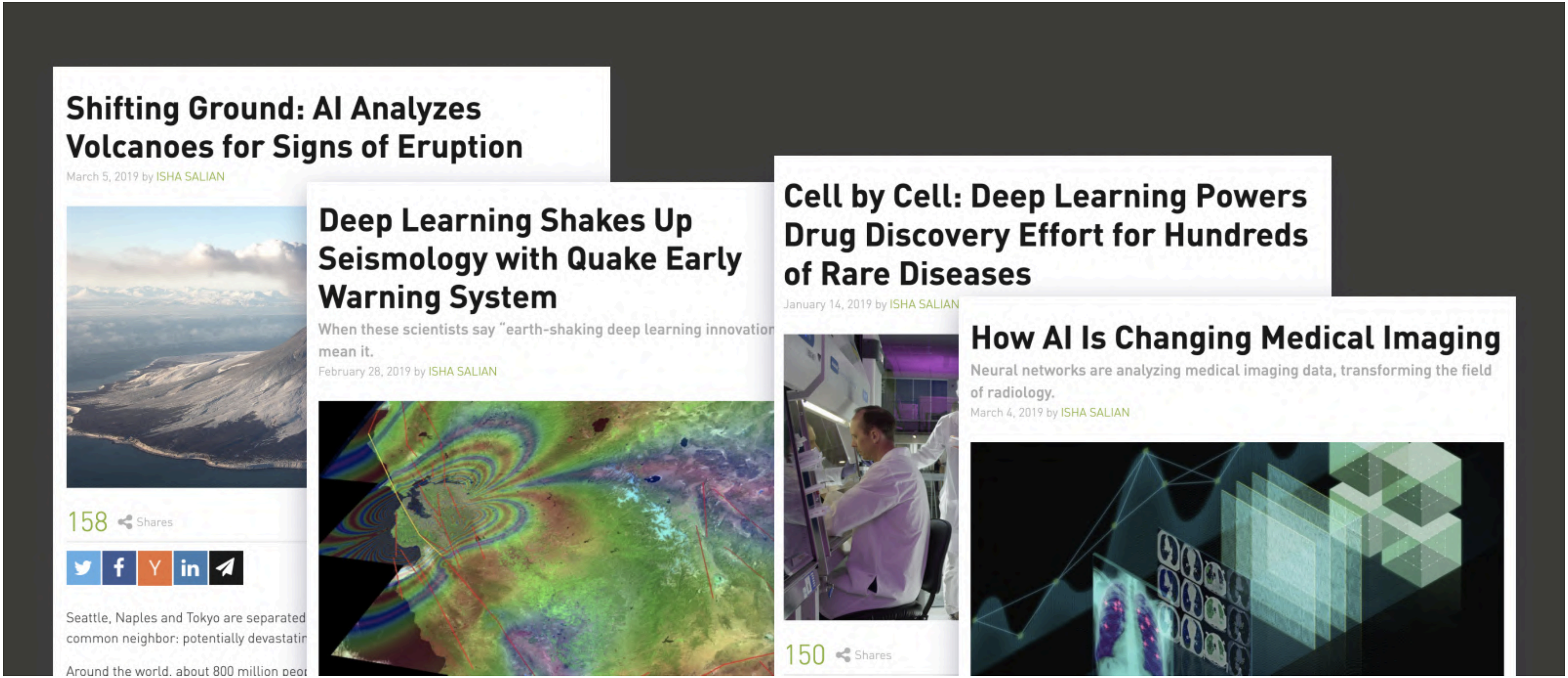


InceptionNet, 5 million parameters [Link]



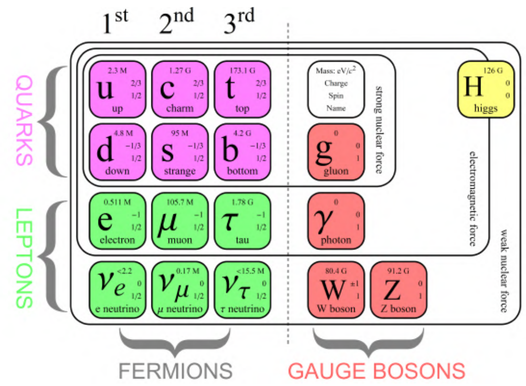
Machine Learning
Framework
Platform
Library
Deep Learning
Framework
Platform
Library
Tool
Reinforcement Learning
Programming
Notebook Environment
Data
Versioning
Store & Format
Operations
Stream Processing
SQL Engine
Feature Engineering
Visualization
Pipeline Management
Labeling and Annotation
Governance
Model
Benchmarking
Training
Parameter
Format & Interface
Marketplace
Workflow
Inference
Tool
Trusted & Responsible AI
Explainability
Adversarial
Bias & Fairness
Distributed Computing
Computing & Management
Interface
Security & Privacy
Natural Language Processing
Education
LFAI Graduated
LFAI Incubating
The LF AI landscape explores open source projects in the domains of artificial intelligence, machine learning, and deep learning.
LF AI Landscape
LF AI
l.lfai.foundation

Deep Learning Applications



Large Hadron Collider
CMS
p
p
LHCb
ATLAS
ALICE

Particle production probabilities



Particles of interest are produced rarely.

~1 second in a human lifetime

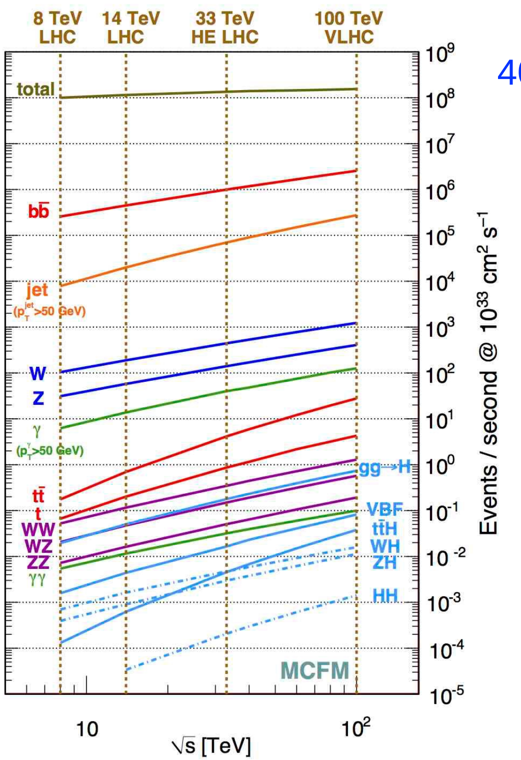
That is why we need to:

look through as many collisions as possible

〉 to maximise the probability of having interesting event

Do not store uninteresting events

〉 Decision time ~10^{-7} seconds



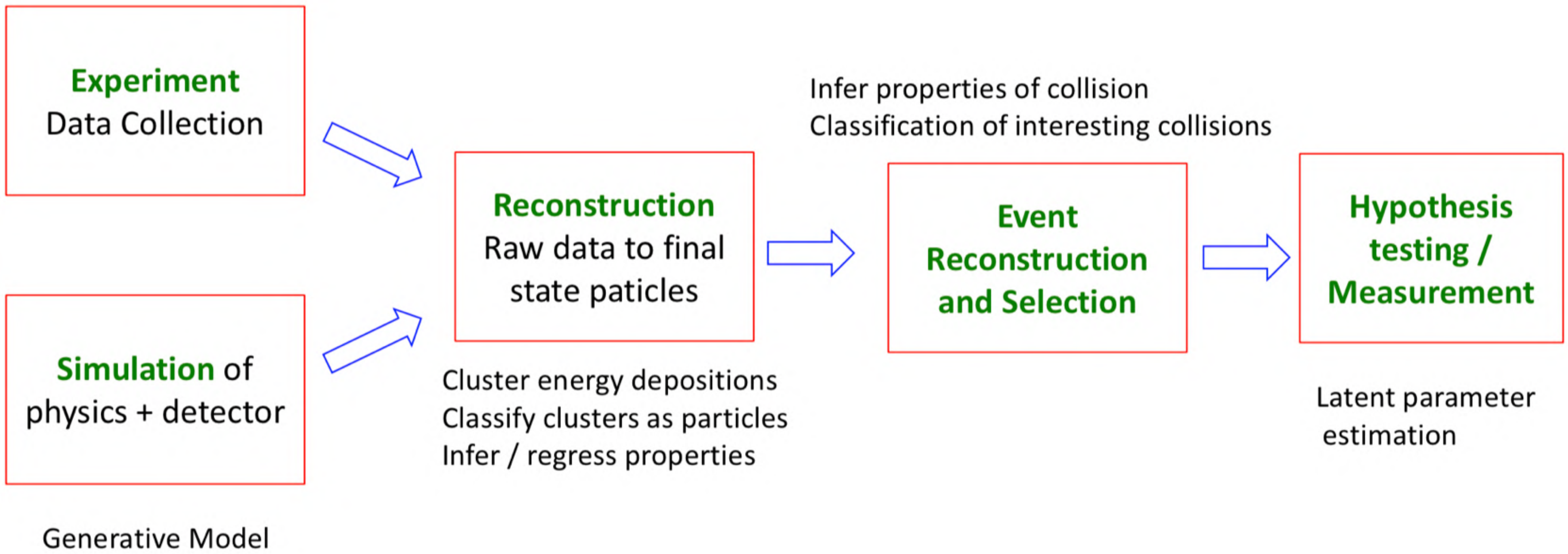
noise:

40 000 000 Hz

signal:

1..10/hour X

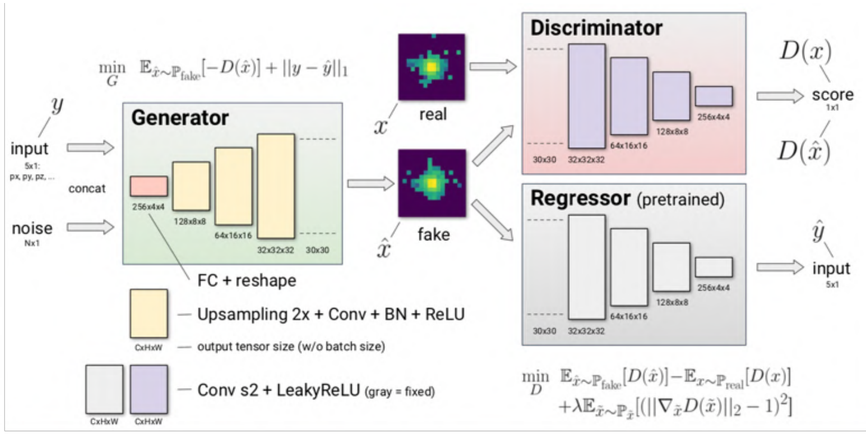
Important ML application areas in HEP

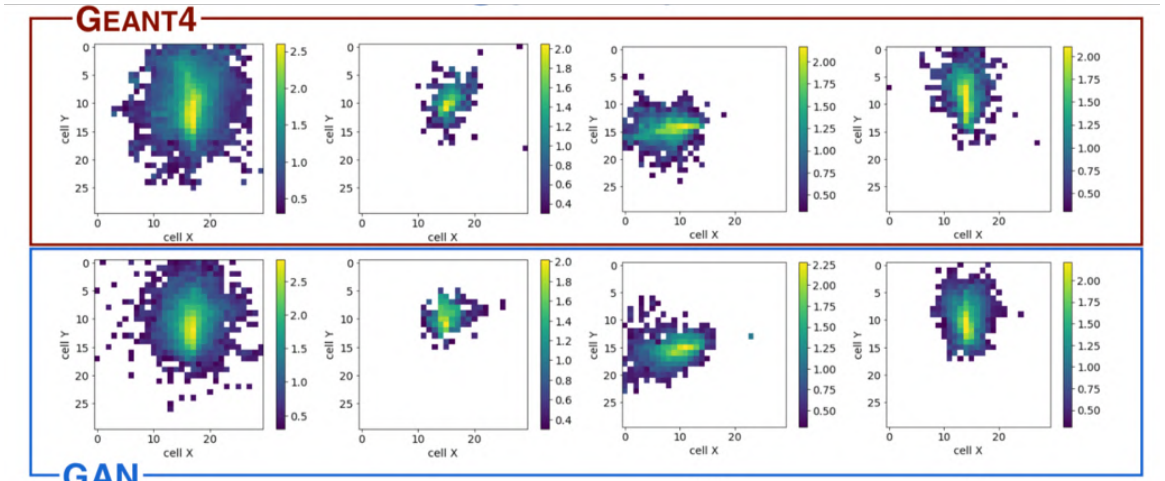


http://bit.ly/2FHWTZ4

Example: Fast Calorimetry Simulation

- LHCb-like calorimeter 30x30
- 5 conditional parameters per particle (3D momentum, 2D coordinate)
- Electrons from particle gun shot at 1x1 cm square at the center of the calorimeter face
- Approach: use GANs
- 10^5 x speed-up!





(a) $E_0 = 63.7$ GeV (b) $E_0 = 6.5$ GeV (c) $E_0 = 15.6$ GeV (d) $E_0 = 15.9$ GeV

https://doi.org/10.1051/epjconf/201921402034

Modern MI Challenges in Physics

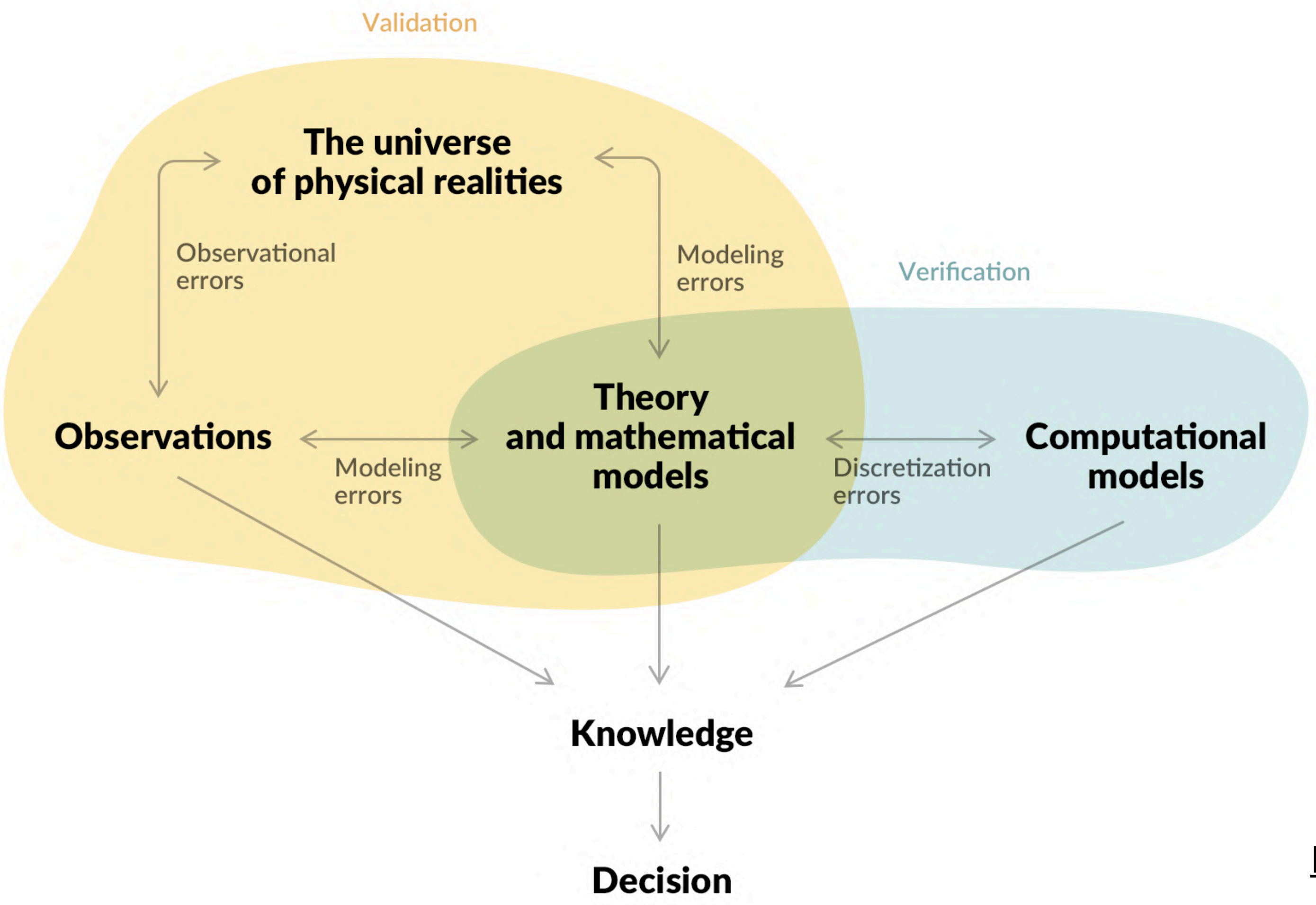
- Apply computer vision and natural language processing techniques to Physics-motivated data structures, like jets, flows, showers, fields;
- Can MI help with our most computationally costly problems, like simulation or the combinatorial challenge?
- Can fast O(ns-μs) MI inference be done with FPGAs to put MI early in the trigger / data acquisition process?
- Can we make MI models robust to data change?
- Can we encode physics-motivated reasoning into MI computations?
- How can we make the best use of simulation engine for inference (of latent variables) given that observation likelihood is intractable?

https://nature.com/articles/s41586-018-0361-2 https://doi.org/10.1073/pnas.1912789117

Mache Learning research

- Relies on the developed background,
- Works best in cooperation with domain science expertise,
- Meta-reasoning,
- Embeds learning patterns into a trainable algorithm
  - Convolutional neural network
  - Langevin Gradient Descent
  - Generative Adversarial Network
  - Neural [Ordinary] Differential Equations
  - Monte Carlo Tree Search
  - …and many others

Wholistic picture



Link

MISiS Spring semester course

- Introduction into Machine Learning
- Introduction into Deep Learning
- Generative models
- Optimization methods
- Advanced topics

Conclusion

- “AI” is for PR/politics, “ML” - for theory, “MI” - for practice
- MI is deeply grounded in all known scientific paradigms
  - Empirical, theoretical, probabilistic and computational
- MI transforms ways of reasoning into usable tools
  - Logic, common sense
  - Statistical inference
  - Rational reasoning and awareness of thinking patterns
  - Computational reasoning
- Biggest challenges for MI:
  - Improve our understanding why and how Deep Learning works
  - Close the loop: Experiment -> hypothesis -> theory -> experiment
- Welcome to the spring semester course!

Thank you!

anaderiRu

hse_lambda

Andrey Ustyuzhanin