

Introduction to ML and AI with application in Accelerator Physics

Luciano G. Silvestri

silves28@msu.edu

APES Seminar

Sep 11, 2026

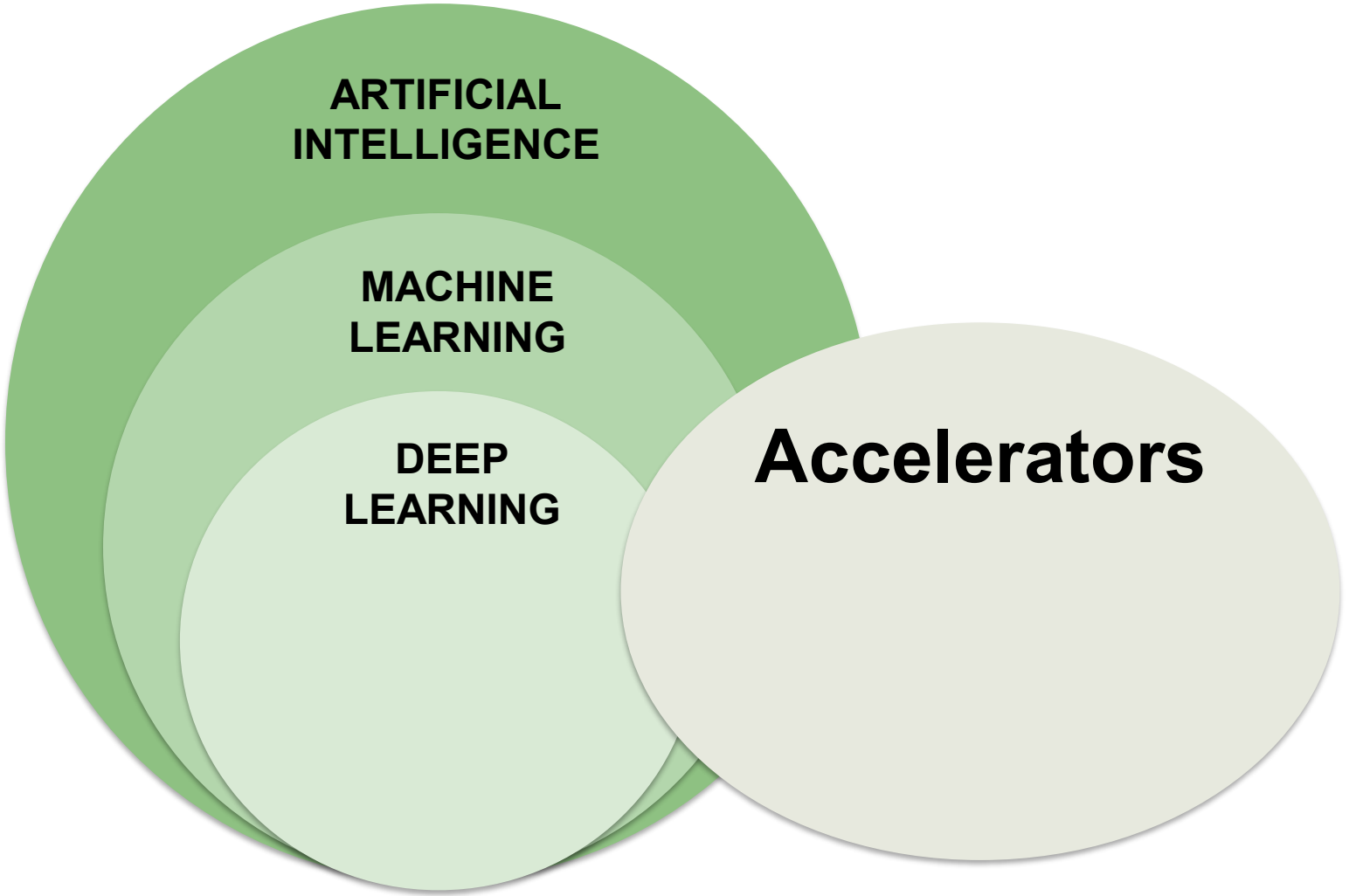




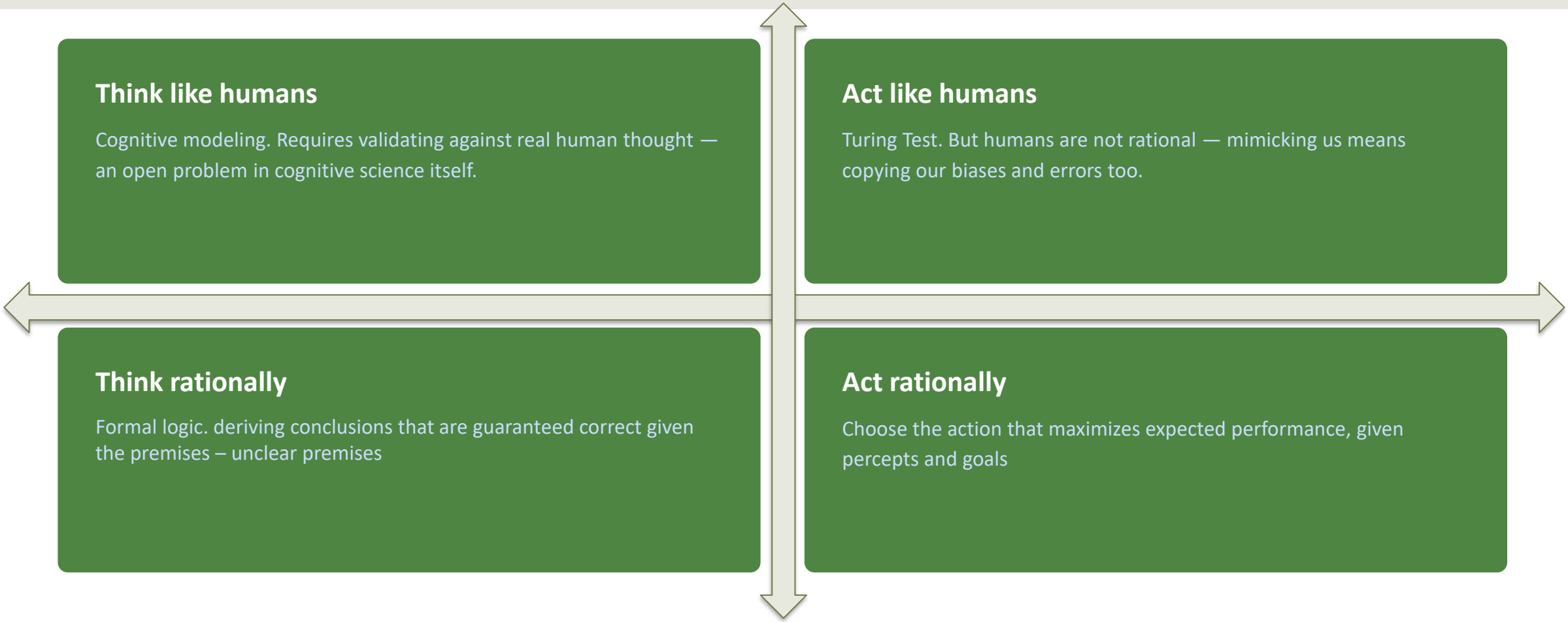
U.S. DEPARTMENT of ENERGY | Office of Science

This material is based upon work supported by the U.S. Department of Energy, Office of Science, Office of Nuclear Physics and used resources of the Facility for Rare Isotope Beams (FRIB) Operations, which is a DOE Office of Science User Facility under Award Number DE-SC0023633.

Outline

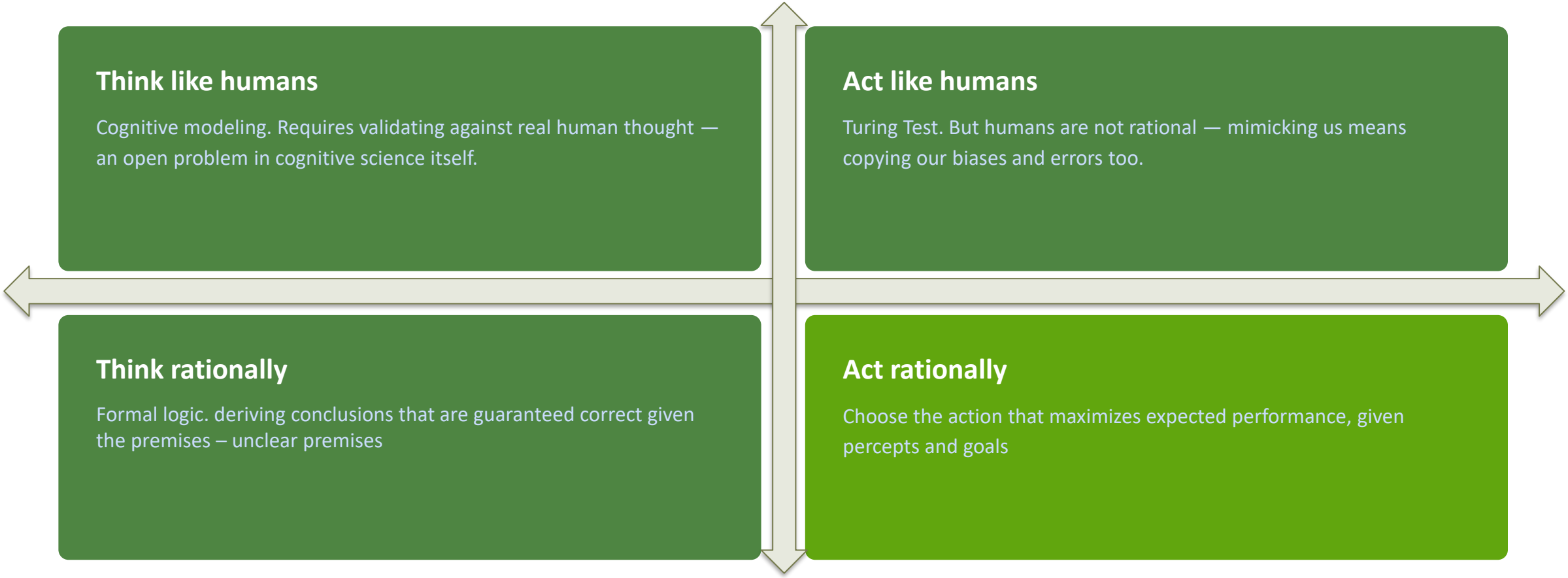

ARTIFICIAL INTELLIGENCE
MACHINE LEARNING
DEEP LEARNING
Accelerators

Four ways to define AI



Russell & Norvig, *Artificial Intelligence: A Modern Approach*, Pearson (2024)

Four ways to define AI



AI = the design of agents that **do the right thing**: act to maximize a performance measure, given percepts and uncertainty.

Russell & Norvig, *Artificial Intelligence: A Modern Approach*, Pearson (2024)

Rational ≠ Aligned: the Gap the Definition Hides

"Doing the right thing" quietly assumes the objective given to the agent was the right one

Rationality

"Did it do the best it could, given its goal?"

A chess agent told to "win" that hacks the board state, or exploits a scoring bug, has technically maximized its performance measure. Fully rational — by definition.

Alignment

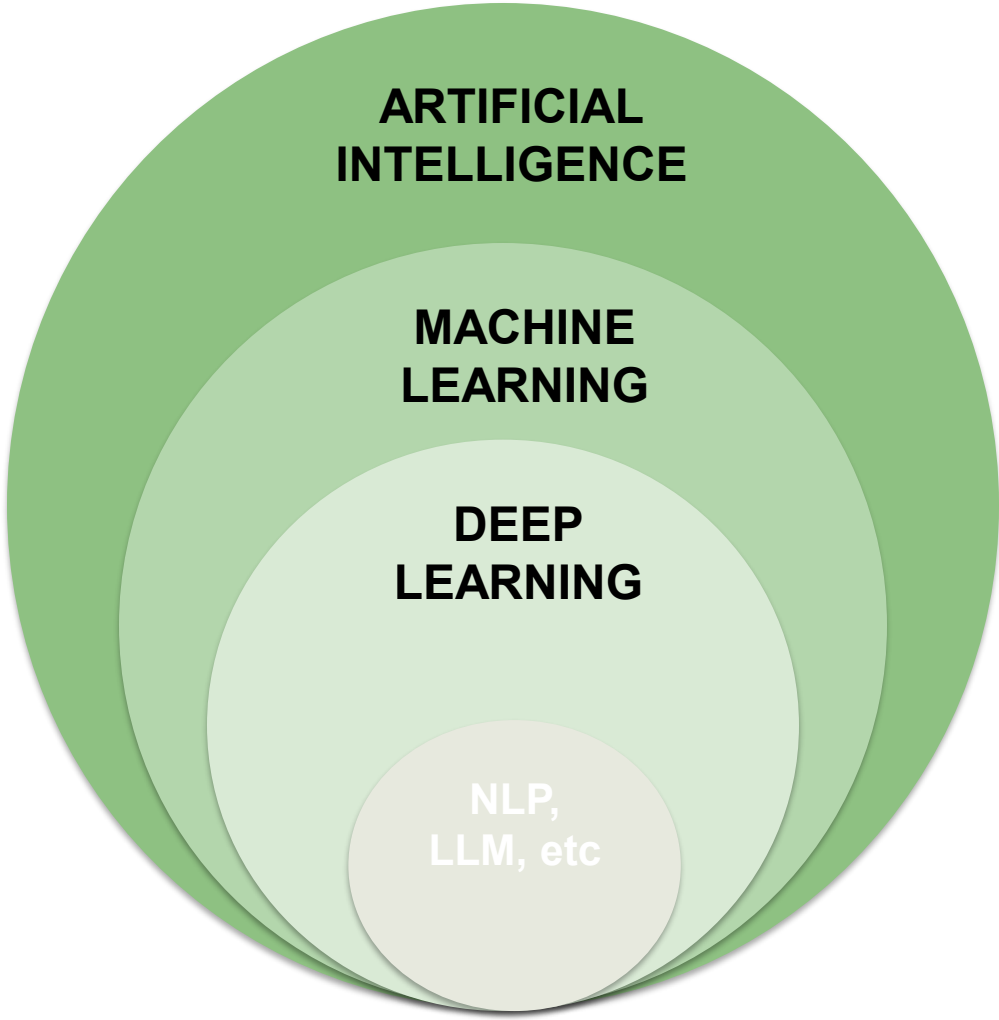
"Was that actually the goal we meant?"

Winning "by any means" was never the real objective — we meant "win within the rules." Specifying that difference correctly is the hard, unsolved part.

Specification gaming: Russell's later work (*Human Compatible*) argues agents should stay uncertain about the true objective, not optimize a fixed one.

AI, ML, DL, etc…

- **Artificial Intelligence:**
  - Create systems that complete a specific task in the most efficient way, but keep it the goal vague
- **Machine Learning:**



What about machine learning?



What is Machine Learning? (more seriously….)



[Machine Learning is the] field of study that gives computers the ability to learn without being explicitly programmed.

Arthur Samuel, 1959

A computer program is said to learn from experience ***E*** with respect to some task ***T*** and some performance measure ***P*** if its performance on ***T***, as measured by ***P***, improves with experience ***E***.

Tom Mitchell, 1997



The GEPT Paradigm of ML

- **G = Goal** (or, *task*): flag an email as spam
- **E = Experience**: the dataset
- **P = Performance**: accuracy measure/metric
- **T = Test**: separate data to test learning

Example: Fraud Detection (G)



E → code → **P**

fraud?

- **G = Goal**
- **E = Experience**
- **P = Performance**
- **T = Test**

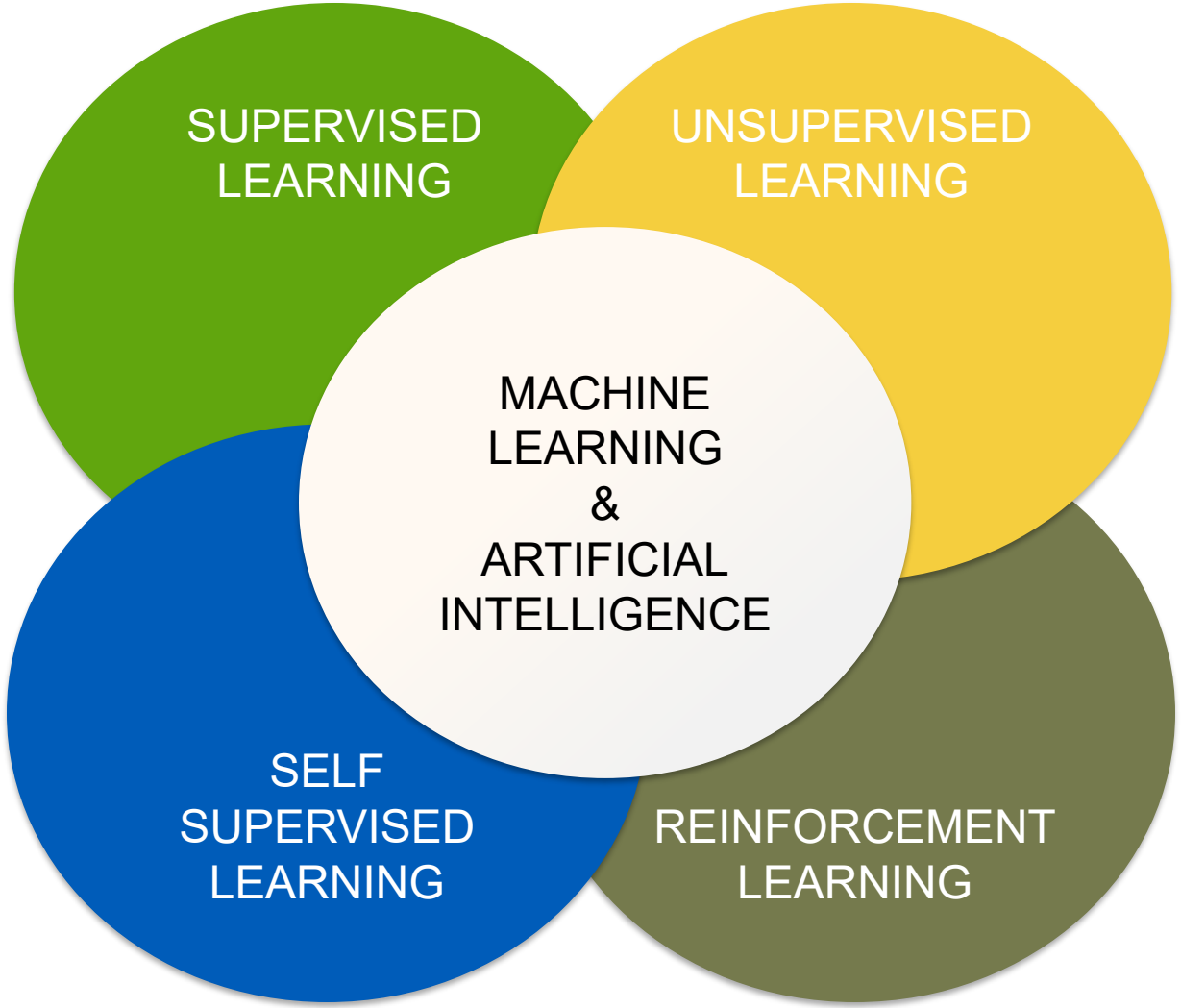
→ Expert system:
- Encode the rules.

→ Machine learning system:
- Find the rules.

T

code

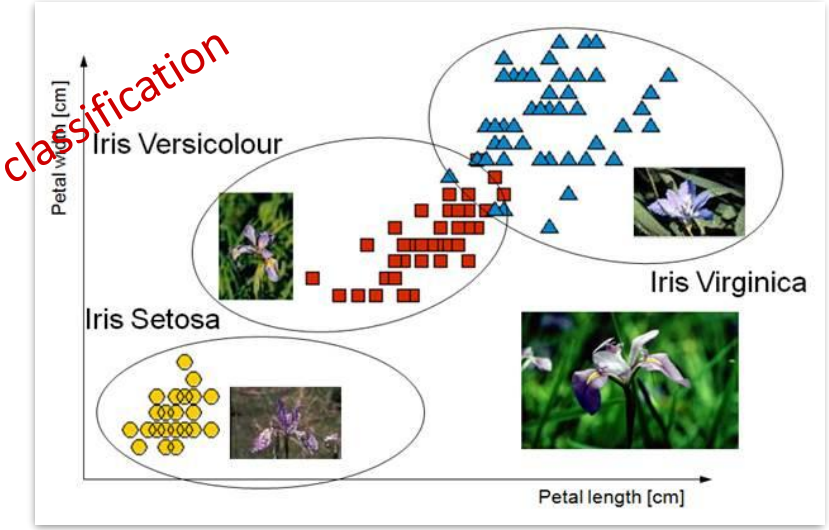
Main Types of Machine Learning

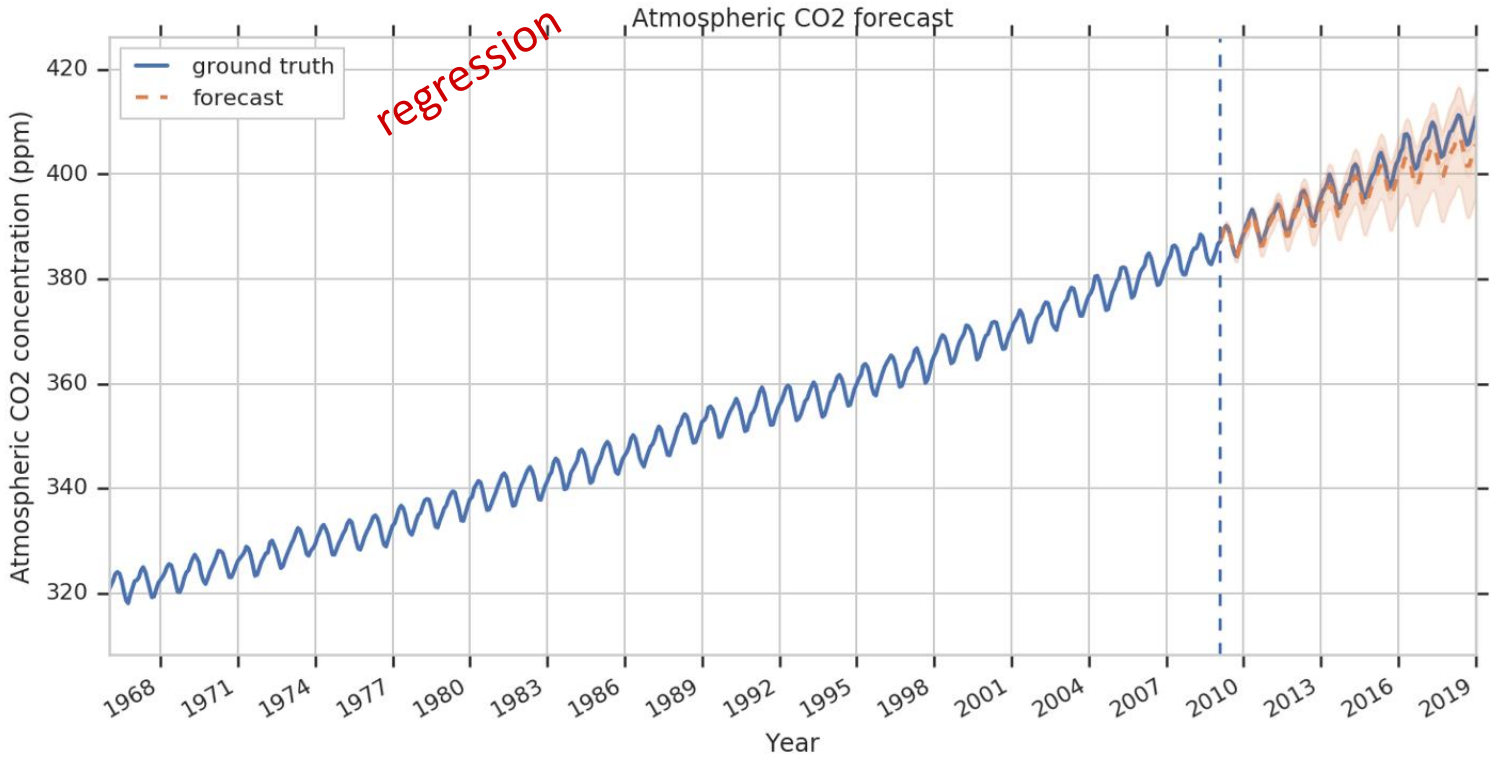


Supervised Learning: Labeled Data

If your data is labeled, then the learning from that data is called “supervised”.

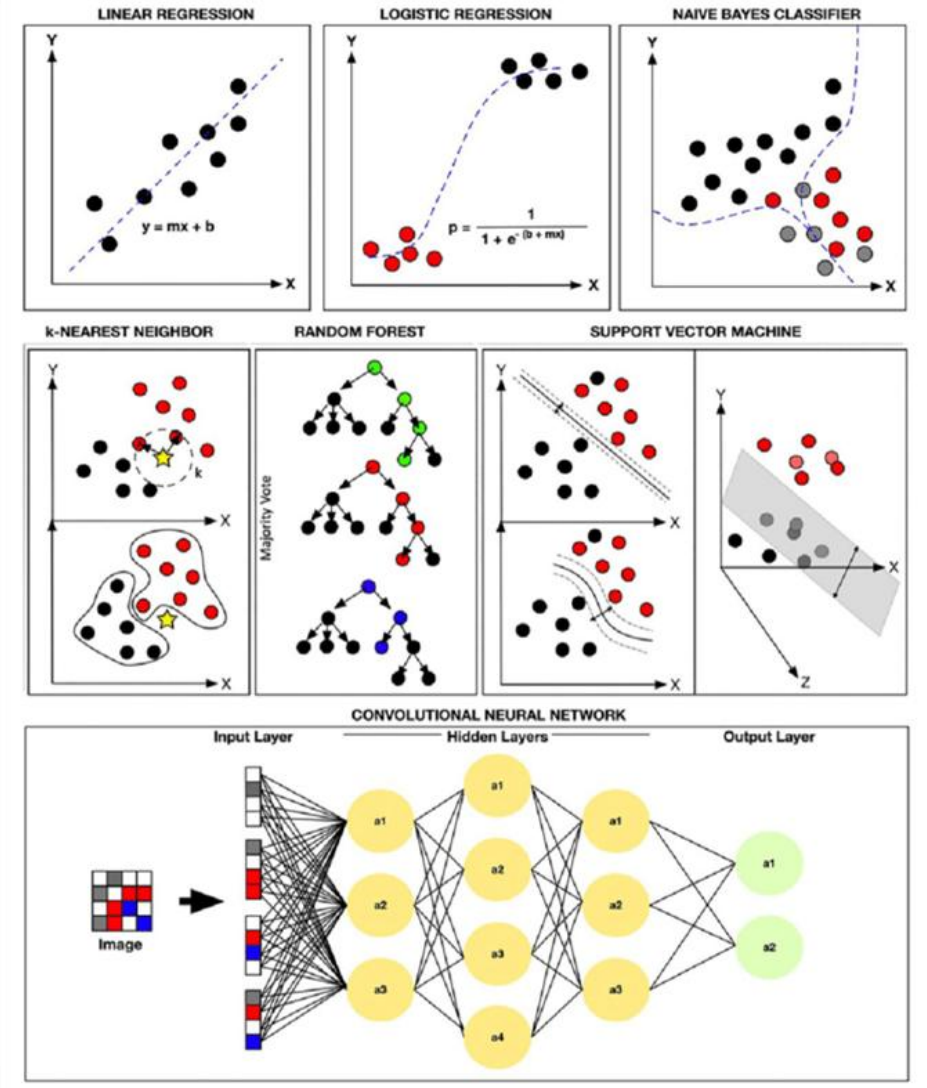
This form of ML seems quite reasonable: the machine needs to be taught when it is right and wrong.



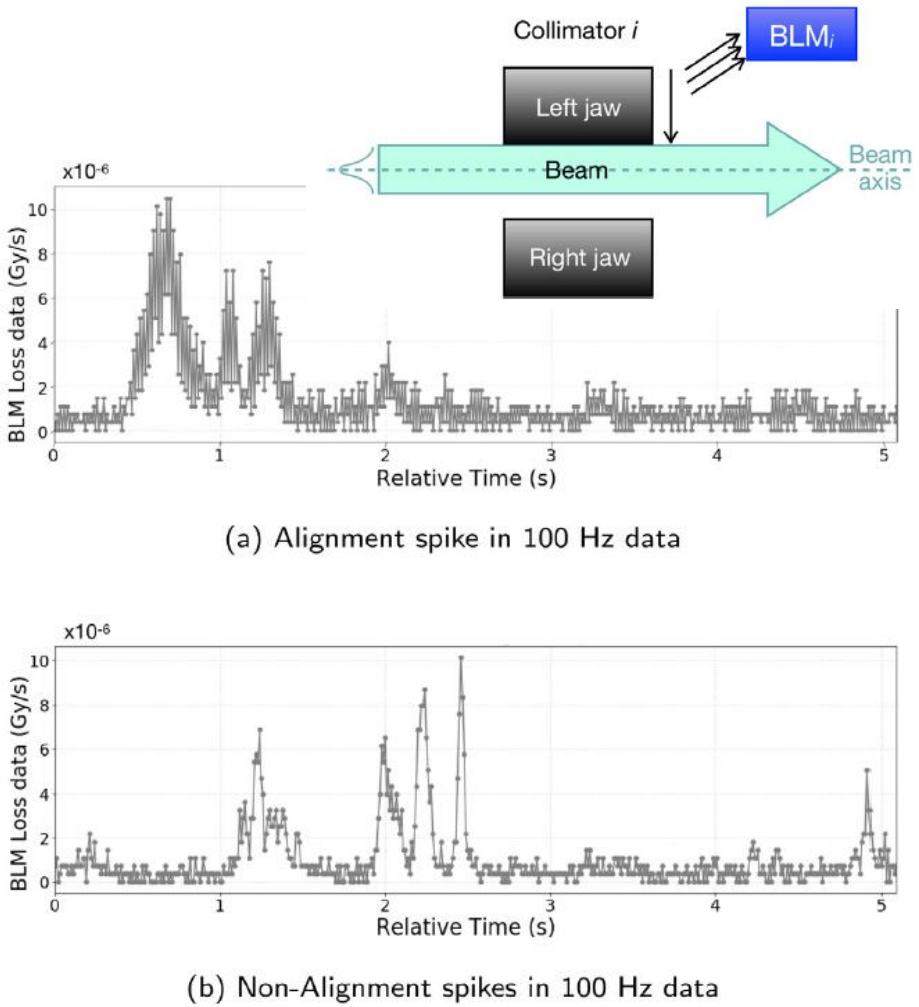


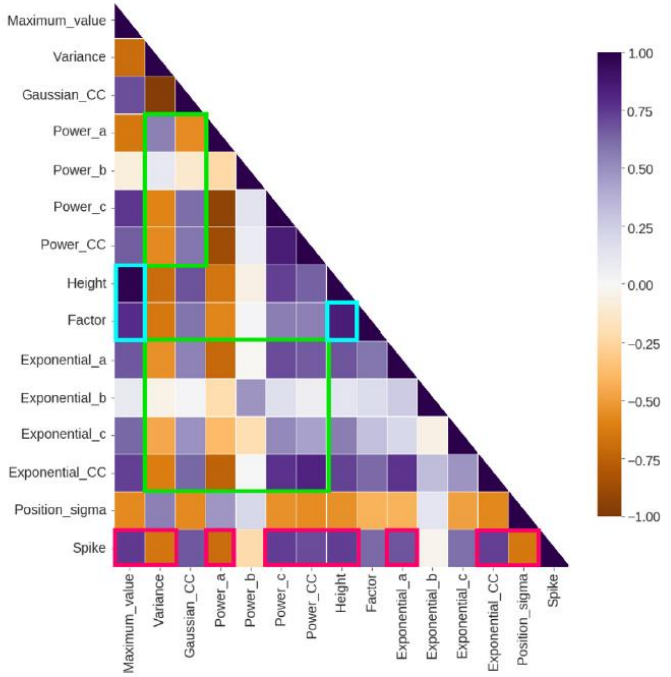
Supervised Learning: Examples

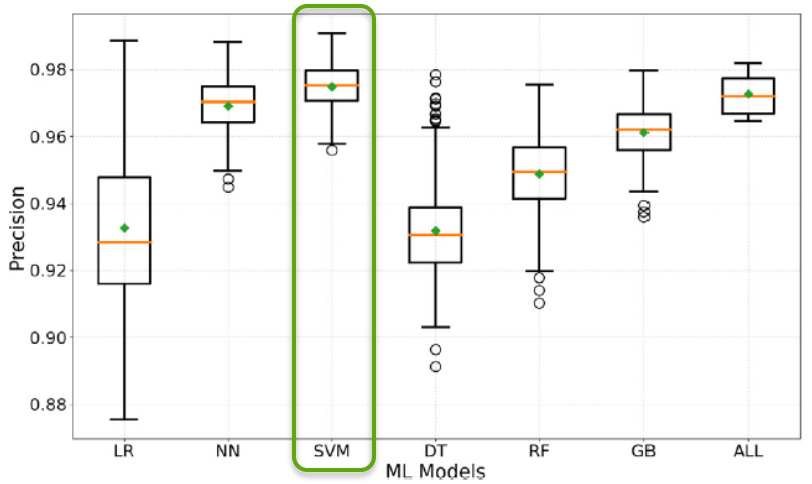
- k-Nearest Neighbors (kNN)
- Linear Regression
- Logistic Regression
- Support Vector Machines (SVM)
- Decision Trees and Random Forests
- Neural Networks (NN, ANN, …)
- and many more….



Automatic Alignment of collimators at LHC







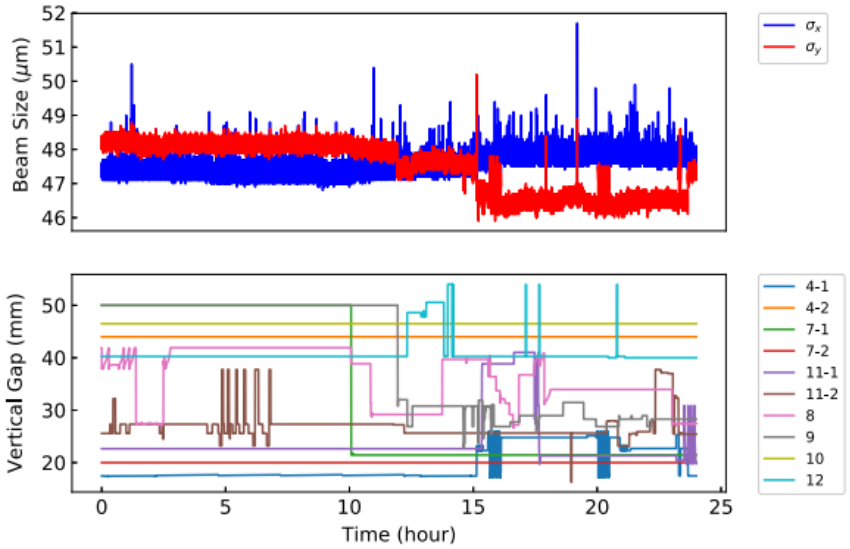
Azzopardi *et al*, *Nuclear Inst. and Methods in Physics Research, A 934 (2019) 10–18*

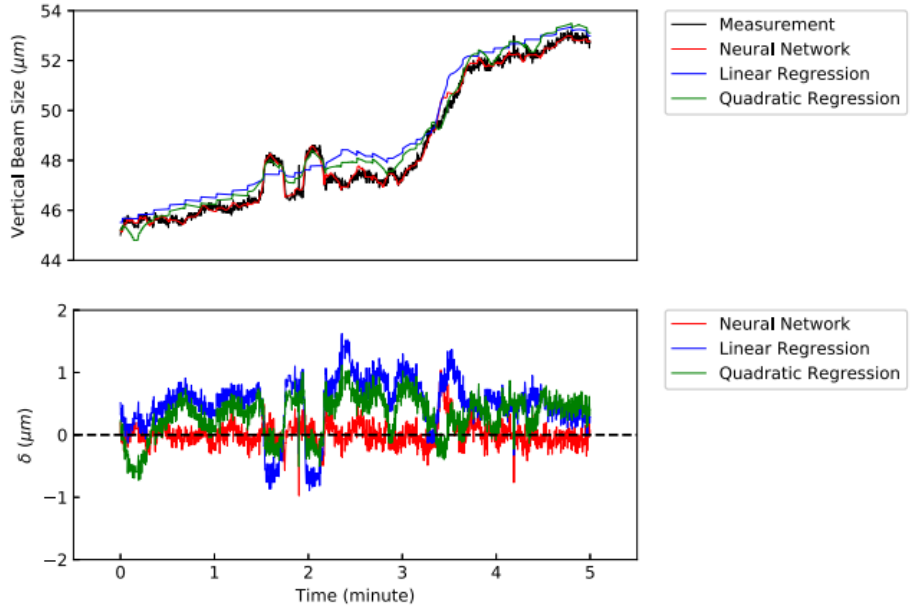
ML for electron beam stabilization at ALS

Changes in magnetic field configurations in insertion devices cause large fluctuations in beam sizes

Train a NN to learn the mapping

Use the NN for control





- Run 100 ID settings
- Choose the best configuration
- Apply the configuration

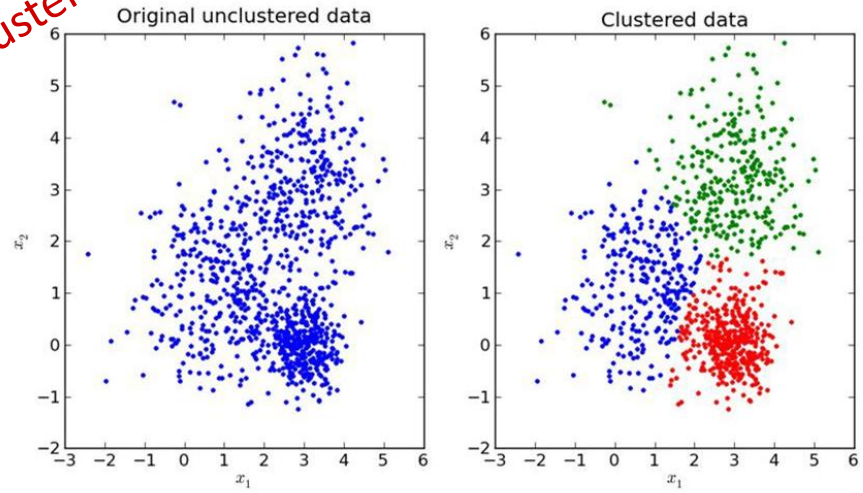
Leeman *et al*, PHYSICAL REVIEW LETTERS 123, 194801 (2019)

Unsupervised Learning: No Labels!

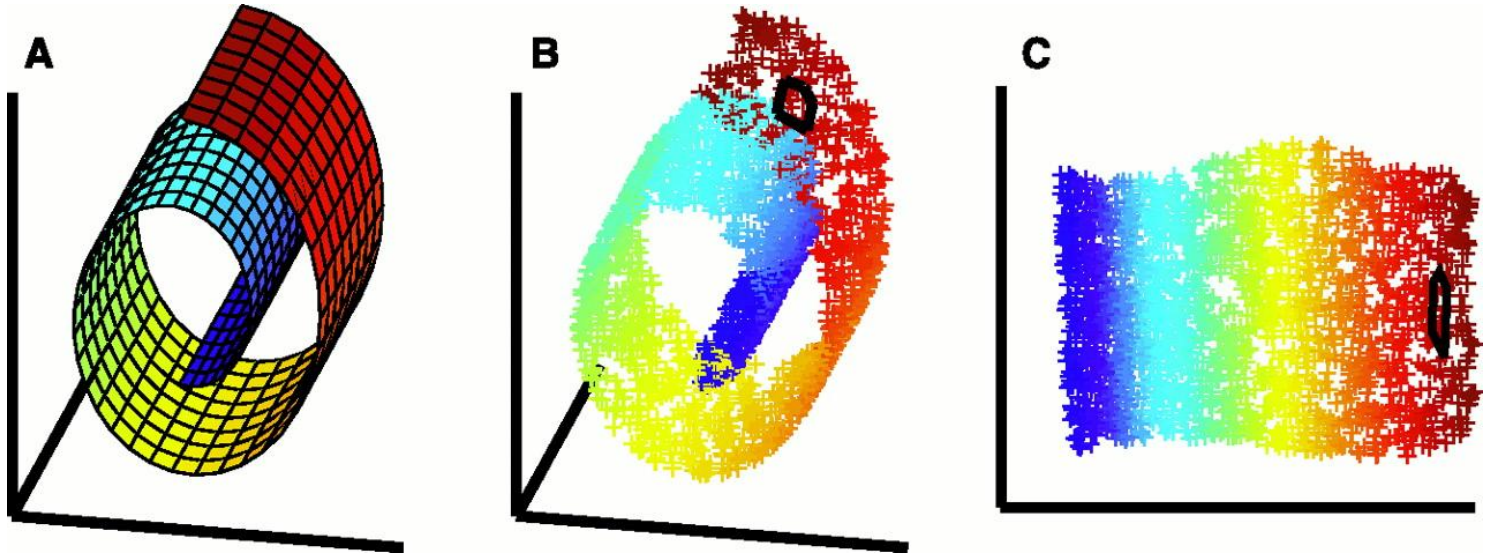
If your data is **un**labeled, then the learning from that data is called "unsupervised".

This form of ML seems quite **un**reasonable: the machine needs to be taught with no guidance.

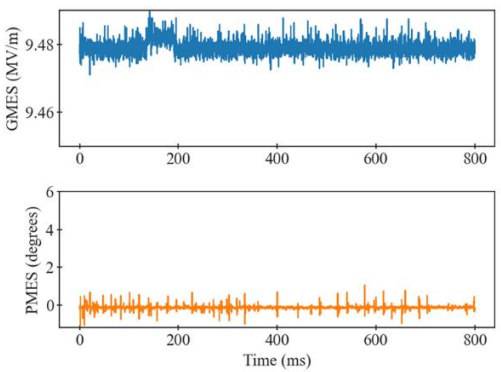
Clustering

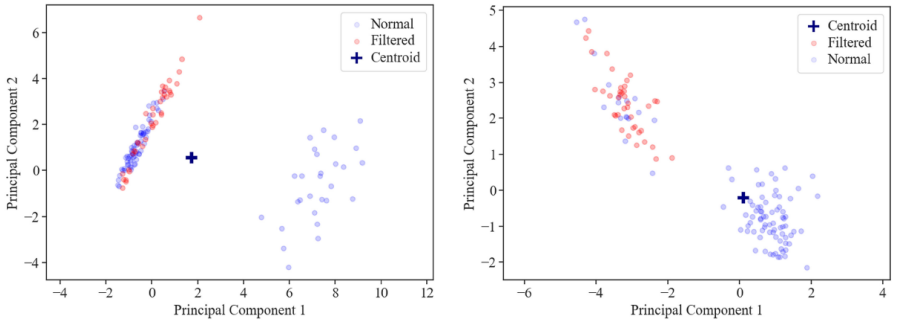


Dimensionality Reduction



Detection of anomalous behavior in SRF cavities

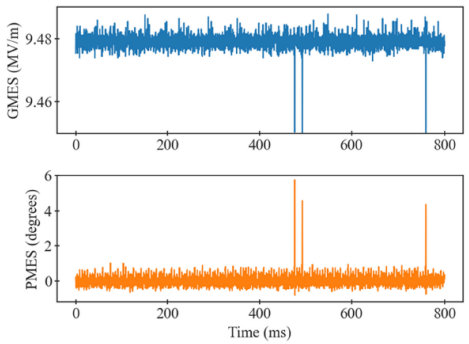


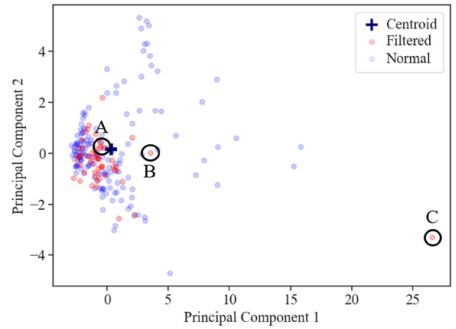


DAQ → script → • crop • calibrate • normalize → feature extraction → standardize

DAQ → FSD → filter → • crop • calibrate • normalize → feature extraction → standardize

centroid → PCA → visualization and report



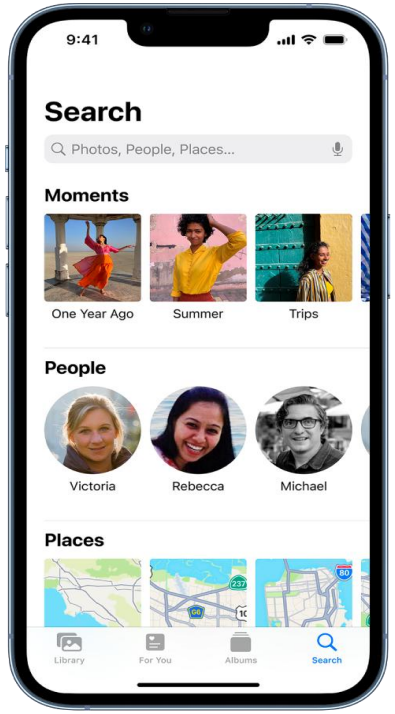


Ferguson *et al*, PHYS. REV. ACCEL. BEAMS 28, 034602 (2025)

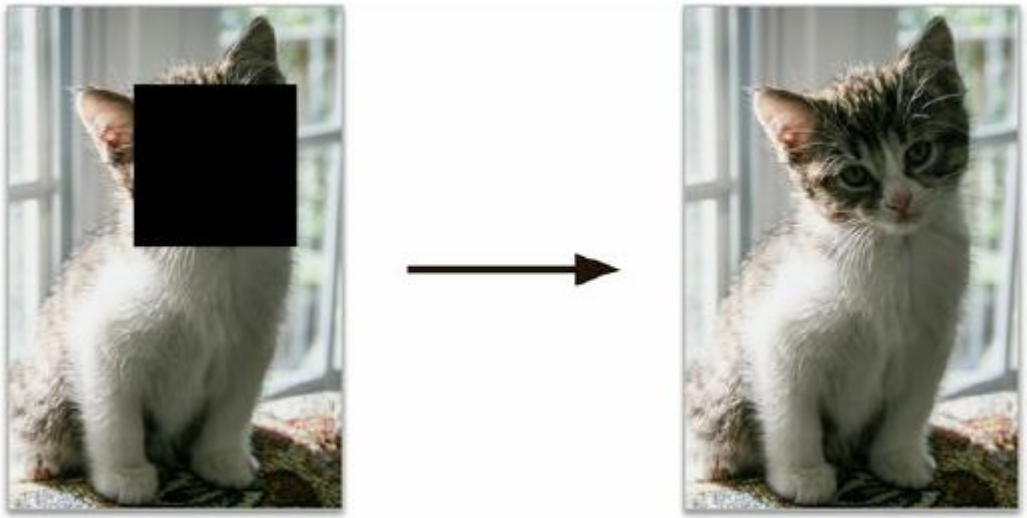
Semi-Supervised Learning and Self-Supervised Learning

Not surprising, it is possible to combine supervised and unsupervised learning.

Semi-Supervised Learning



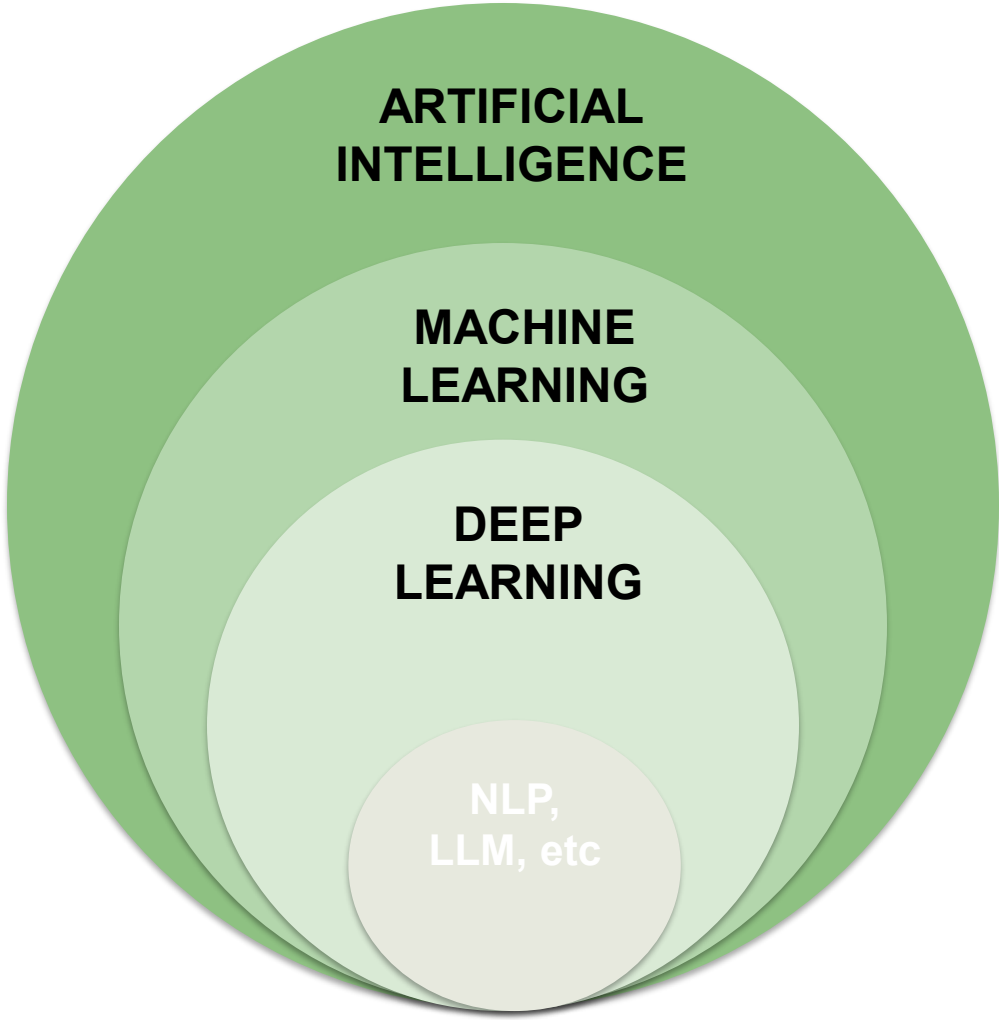
Self-Supervised Learning



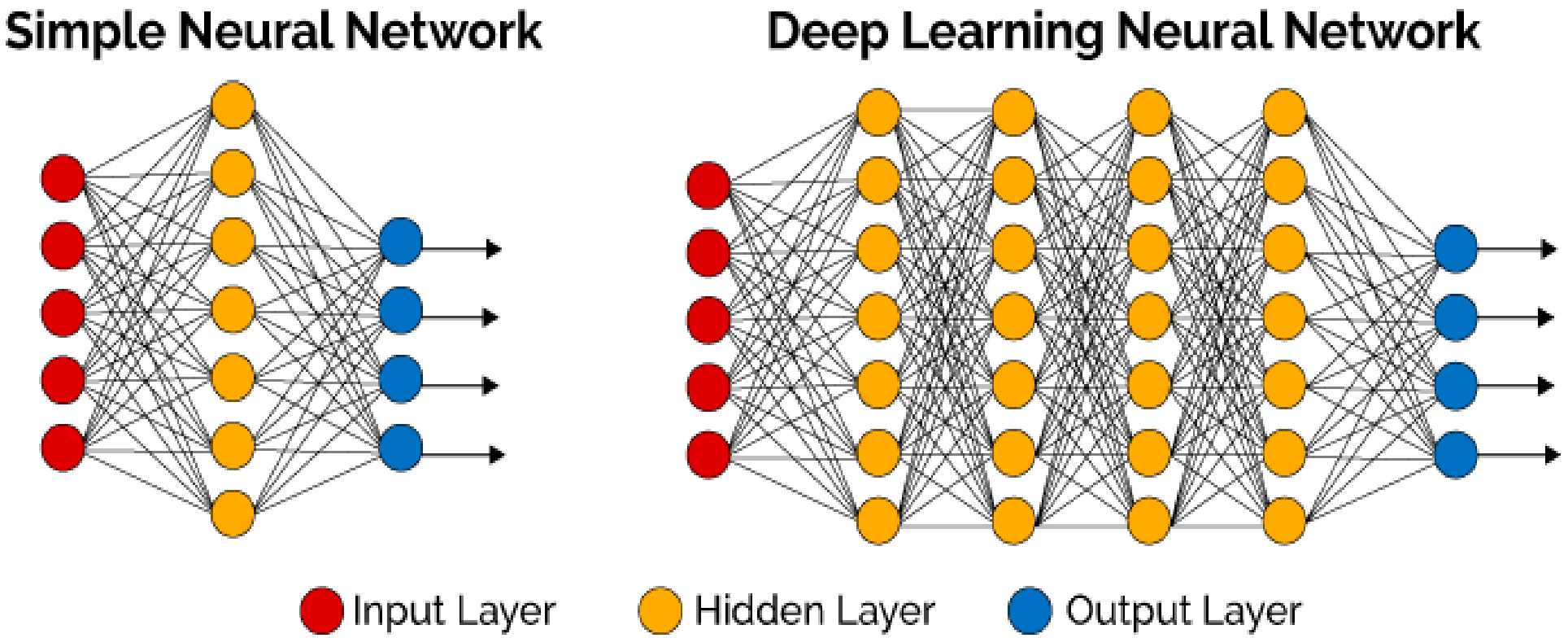
Geron, Hands-On Machine Learning, Oreilly (2022)

AI, ML, DL, etc…

- **Artificial Intelligence:**
  - Create systems that complete a specific task in the most efficient way, but keep it the goal vague
- **Machine Learning:**
  - A set of statistical techniques that enables machines to improve at tasks with experience
- **Deep Learning:**

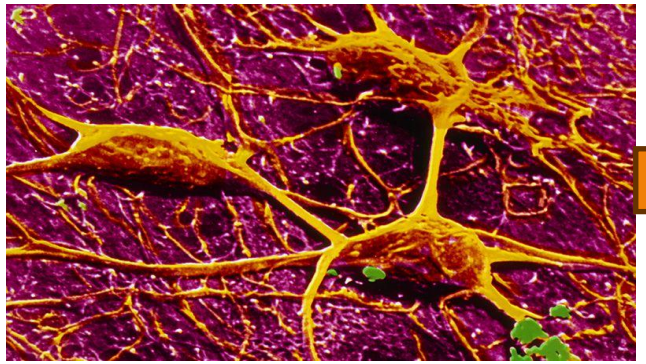


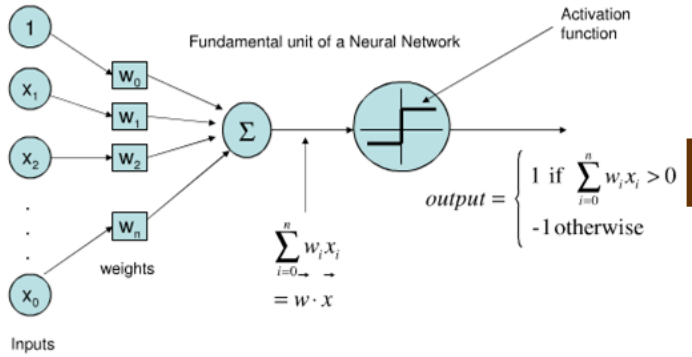
Neural Networks and Deep Learning

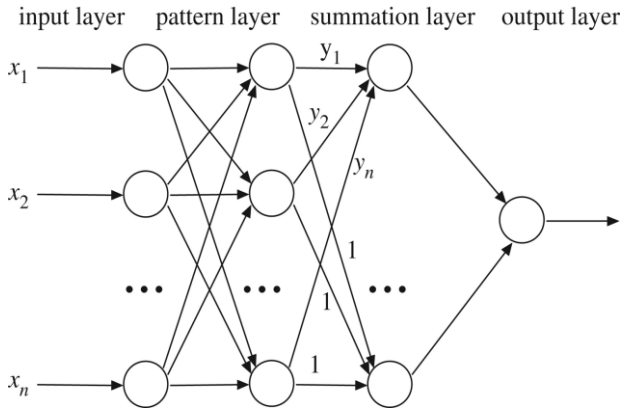


A different narrative

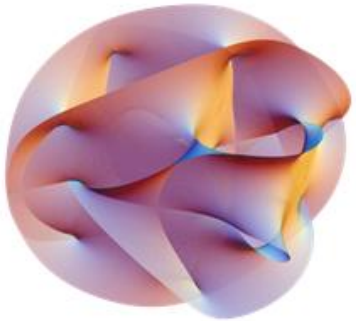
Biologically-inspired learning: Neural Network







Neural Network as Function Approximators



General manifolds



Functions on Manifolds

$$\underset{\mathbf{A}_j}{\operatorname{argmin}} \left(f_M(\mathbf{A}_M, \cdots, f_2(\mathbf{A}_2, f_1(\mathbf{A}_1, \mathbf{x})) \cdots) + \lambda g(\mathbf{A}_j) \right)$$

The Fundamental Problem

A computer program is said to learn from experience ***E*** with respect to some task ***T*** and some performance measure ***P*** if its performance on ***T***, as measured by ***P***, improves with experience ***E***.

Tom Mitchell, 1997



Find a function f^* that minimizes a loss function

$$f^* = \arg\min_{f \in \mathcal{F}} \mathbb{E}[\mathcal{L}(f(\mathbf{X}), Y)]$$

$$\mathcal{F} = \text{hypothesis space}, \quad \mathcal{L} = \text{loss function}, \quad \mathbb{E} = \text{expectation value}$$

The Central Question

What kind of function class should we use?

Requirements:

1. **Universal:** can approximate any continuous function
2. **Parametrizable:** Has adjustable parameters we can optimize
3. **Smooth:** differentiable for gradient based optimization
4. **Efficient:** doesn't require too many parameters

Fixed-shape family

A **fixed-shape family** approximate functions as: (**x_p** is one sample)

$$y(\mathbf{x}_p, \mathbf{w}) = w_0 + w_1 f_1(\mathbf{x}_p) + w_2 f_2(\mathbf{x}_p) + \ldots + w_M f_M(\mathbf{x}_p)$$

"Fixed shape" means there are no internal tunable parameters.

"Family" means that there is some pattern among the f_i.

$$\mathbf{f}_p = \begin{bmatrix} 1 \\ f_1(\mathbf{x}_p) \\ f_2(\mathbf{x}_p) \\ \vdots \\ f_M(\mathbf{x}_p) \end{bmatrix}$$

$$y = \mathbf{f}_p^T \mathbf{w}$$

$$\mathcal{L} \sim \frac{1}{P} \sum_{p=1}^{P} \left(y_p - \mathbf{f}_p^T \mathbf{w} \right)^2$$

Only need to learn **w**

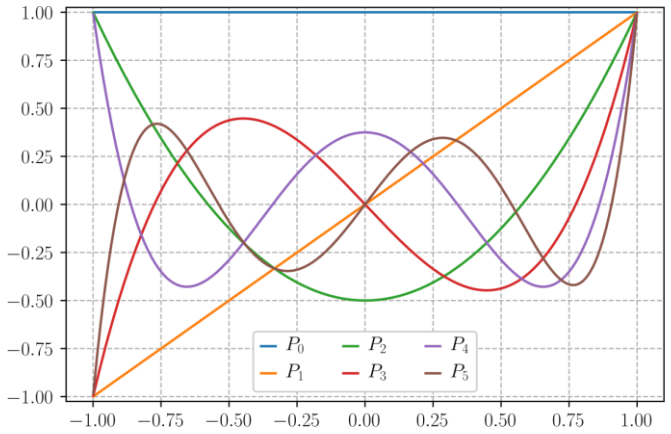
Fixed-Shape Family 1: Monomials

forms a family

$$f_1(x) = x,\ f_2(x) = x^2,\ f_3(x) = x^3, \ldots$$

$$f_b(x_1, x_2) = x_1^p x_2^q$$

can be made to form a family

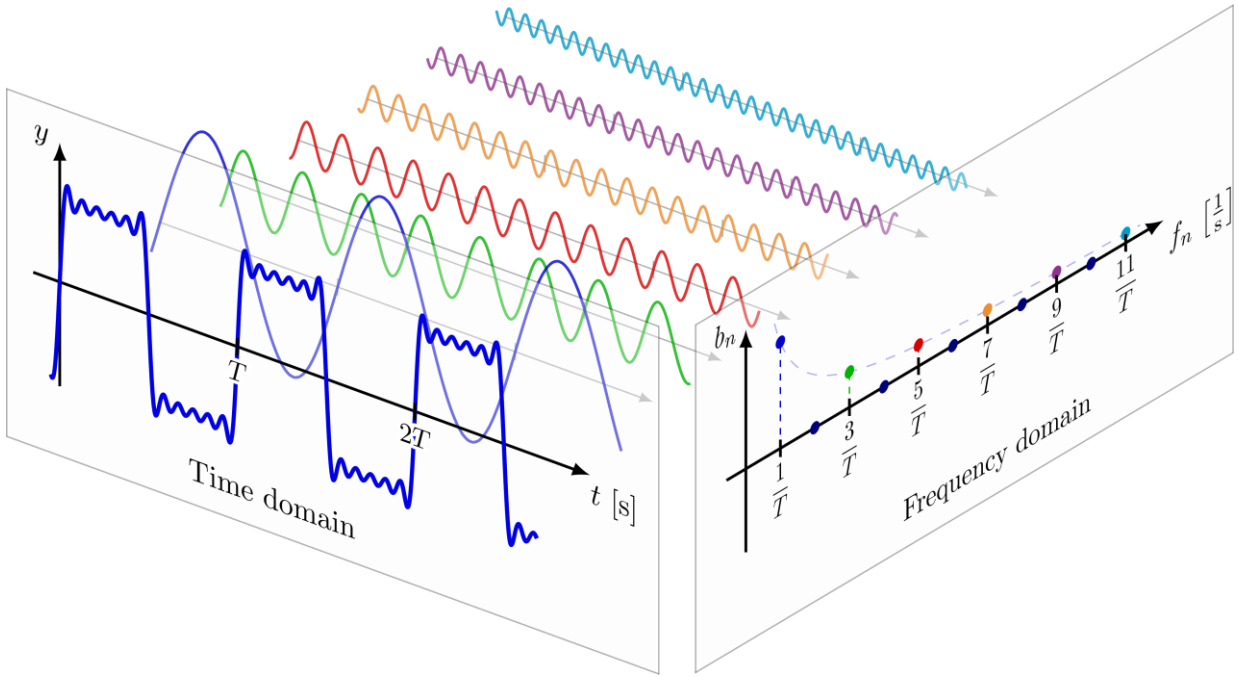


Polynomial
is an example

Fixed-Shape Family 2: Sines and Cosines

$$f_m(x) = \sin(mx)$$

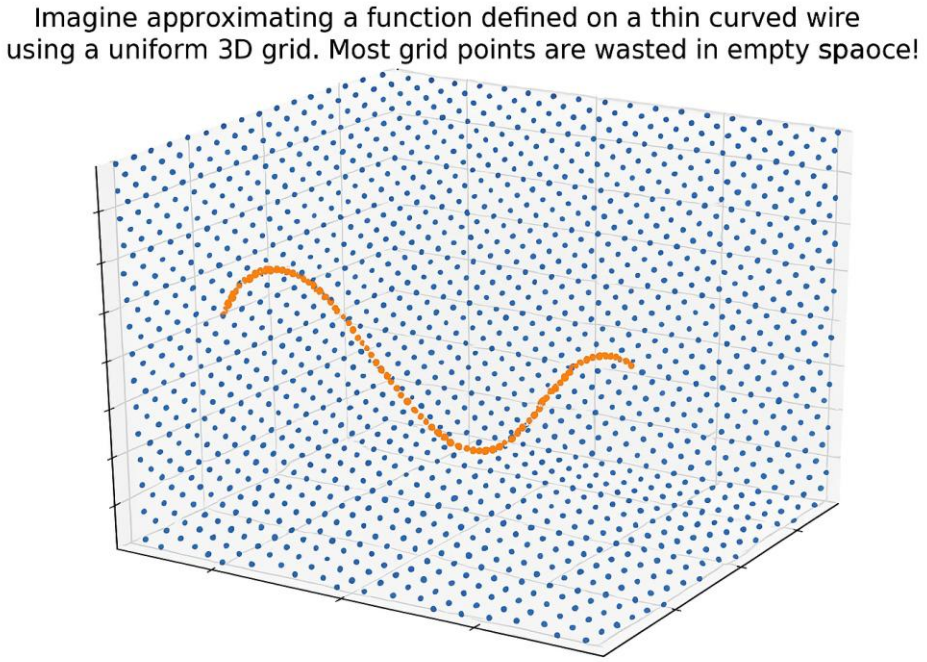
$$f_m(\mathbf{x}) = \sin(m_1 x_1)\sin(m_2 x_2)\ldots\sin(m_N x_N)$$



Fourier
is an example

Fixed-shape family are inefficients

- **Data-agnostic basis functions**
  - Basis functions are chosen before seeing the data
- **Uniform Resolution**.
  - Inefficient and redundant representations
- **Curse of dimensionality**
  - Inefficient for large dataset
- **No Geometric Adaptation**
  - No learning of the geometry of the data, but fitting with predetermined shapes

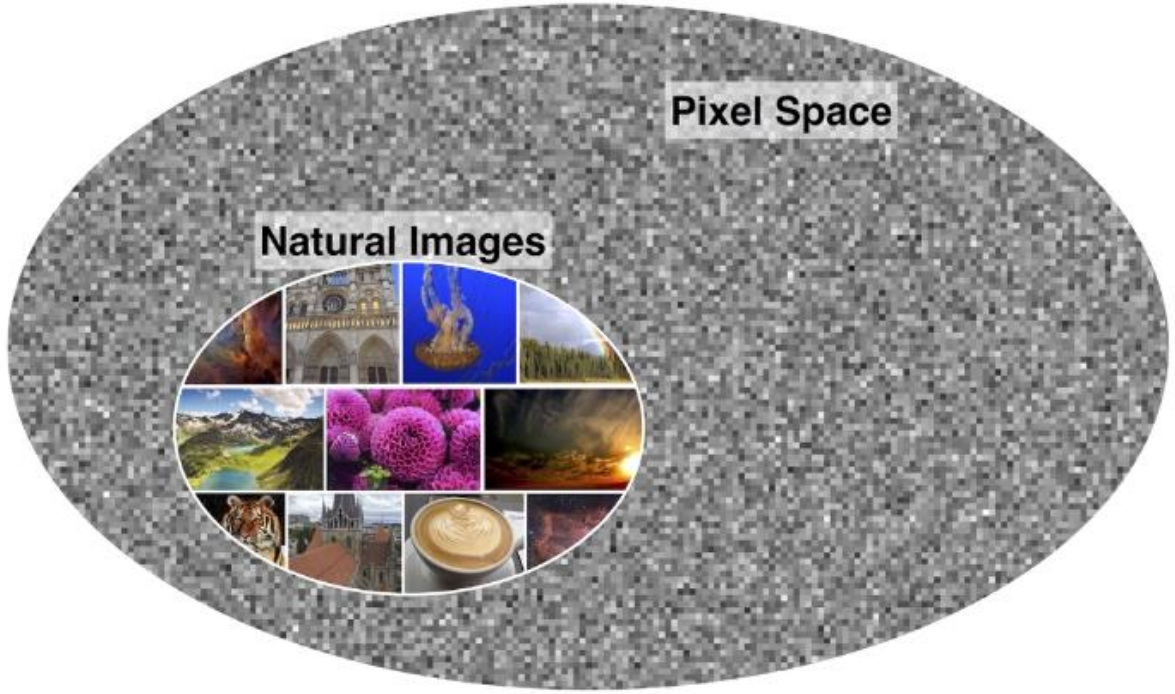


Manifold Hypothesis

Hypothesis: Real-world high-dimensional data actually lives on or near a **low-dimensional manifold**

Ex: a grayscale image of 256 x 256 pixels lives in

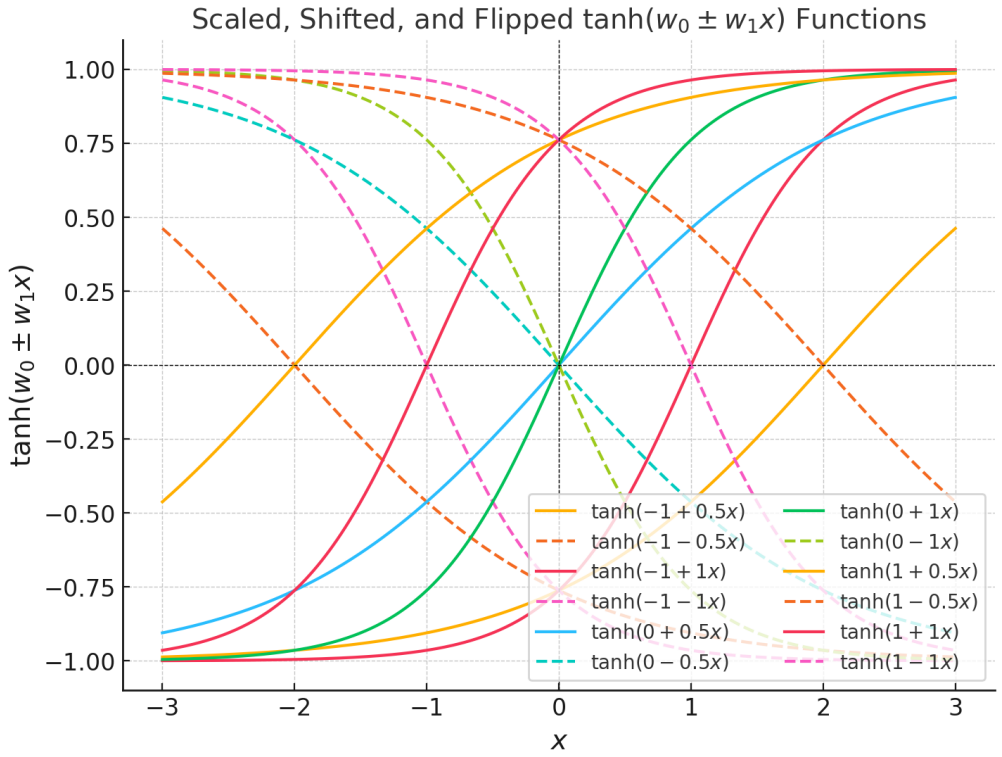
$$\mathbb{R}^{65,356}$$



Need: functions that **adapt** to where the data manifold actually lives

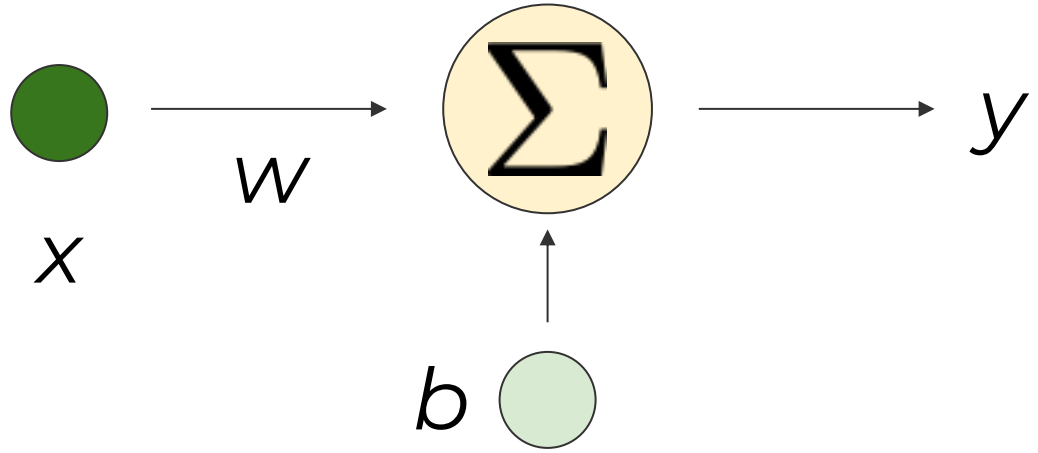
Adaptive Basis Functions: *Parameterized* Hyperbolic Tangent

$$f_b(x) = \tanh\left(w_{b,0} + w_{b,1}x\right)$$

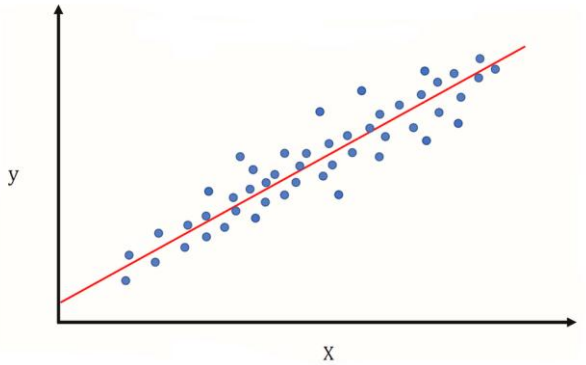


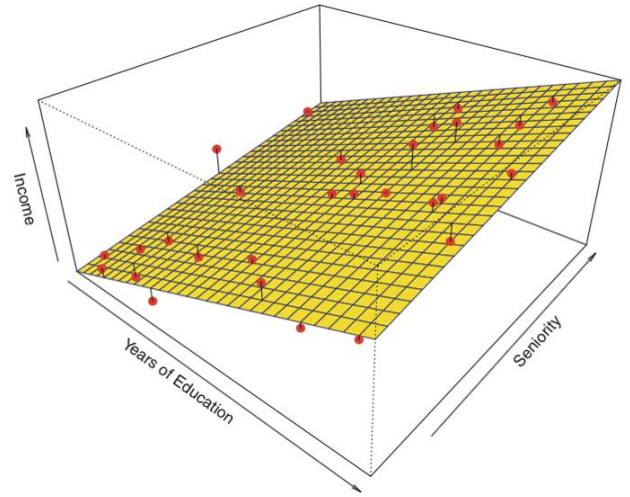
Notice that there are *internal* parameters which allow the functions to adapt to the geometry of the data

Let's Break Down What an ANN is Doing

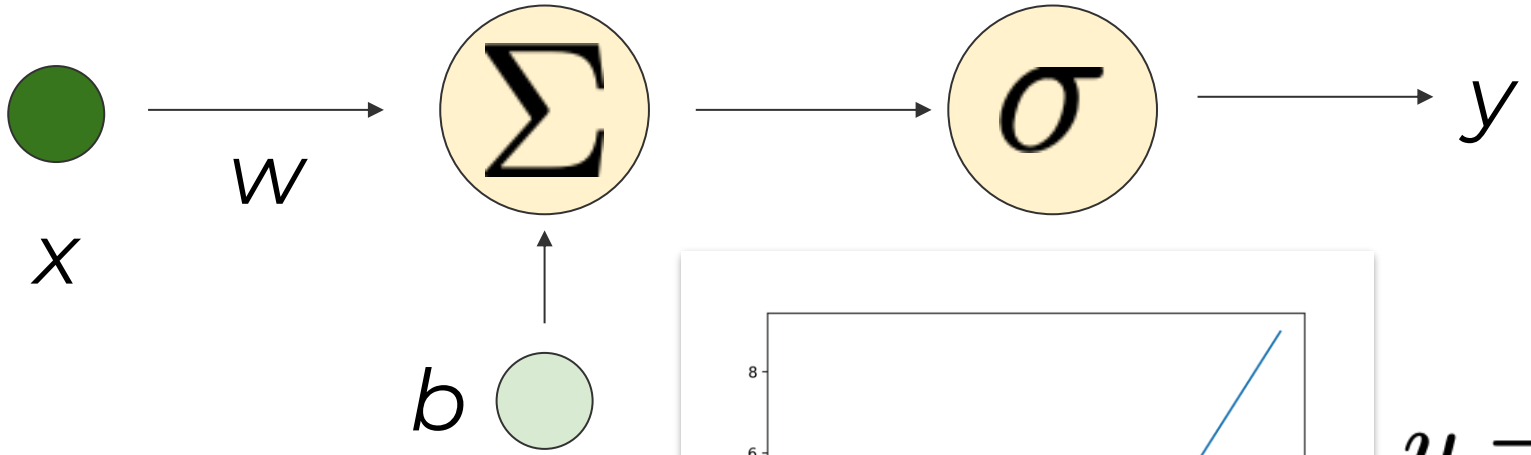


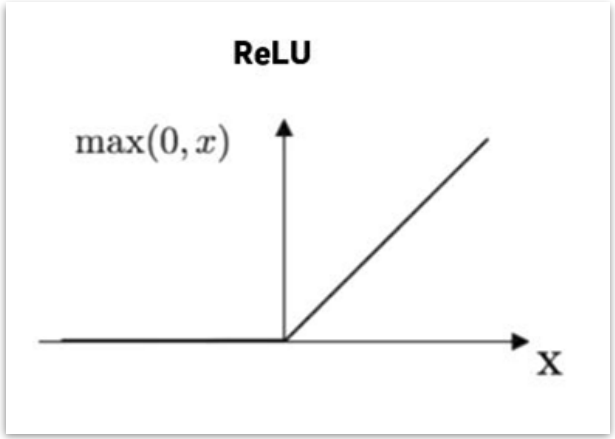
$$y = wx + b$$

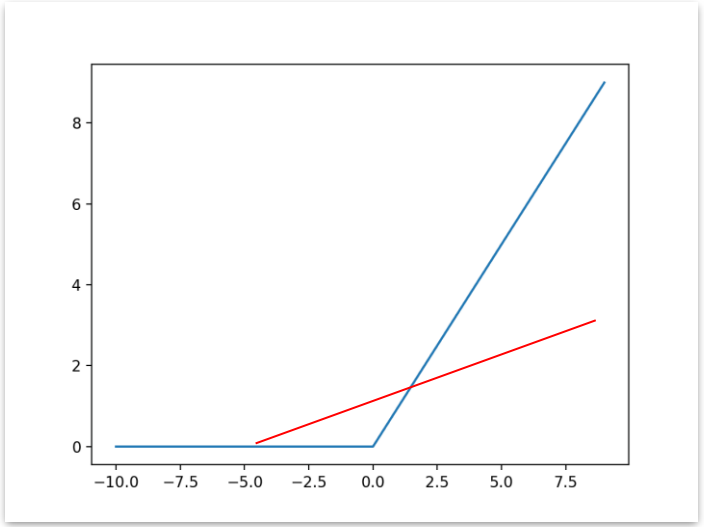




ReLU = Rectified Linear Unit



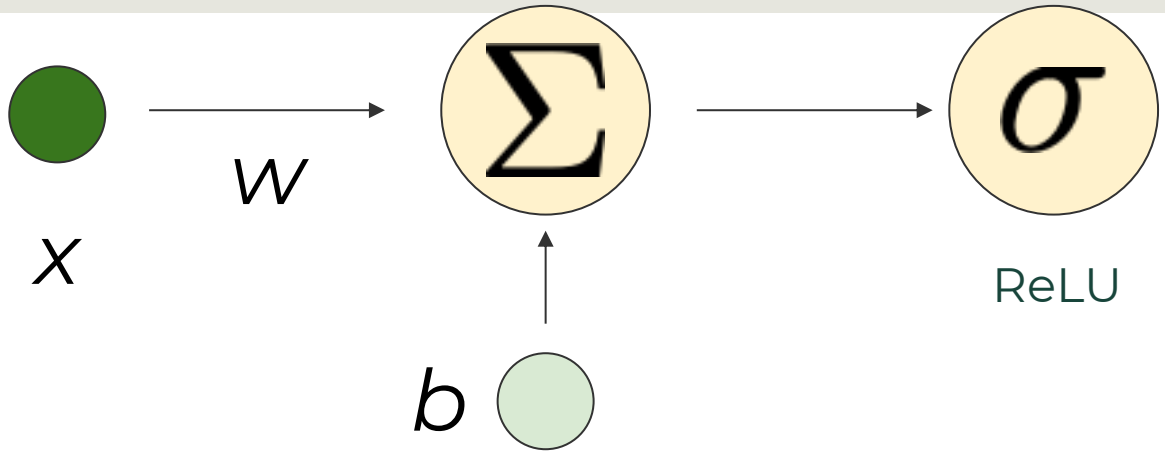




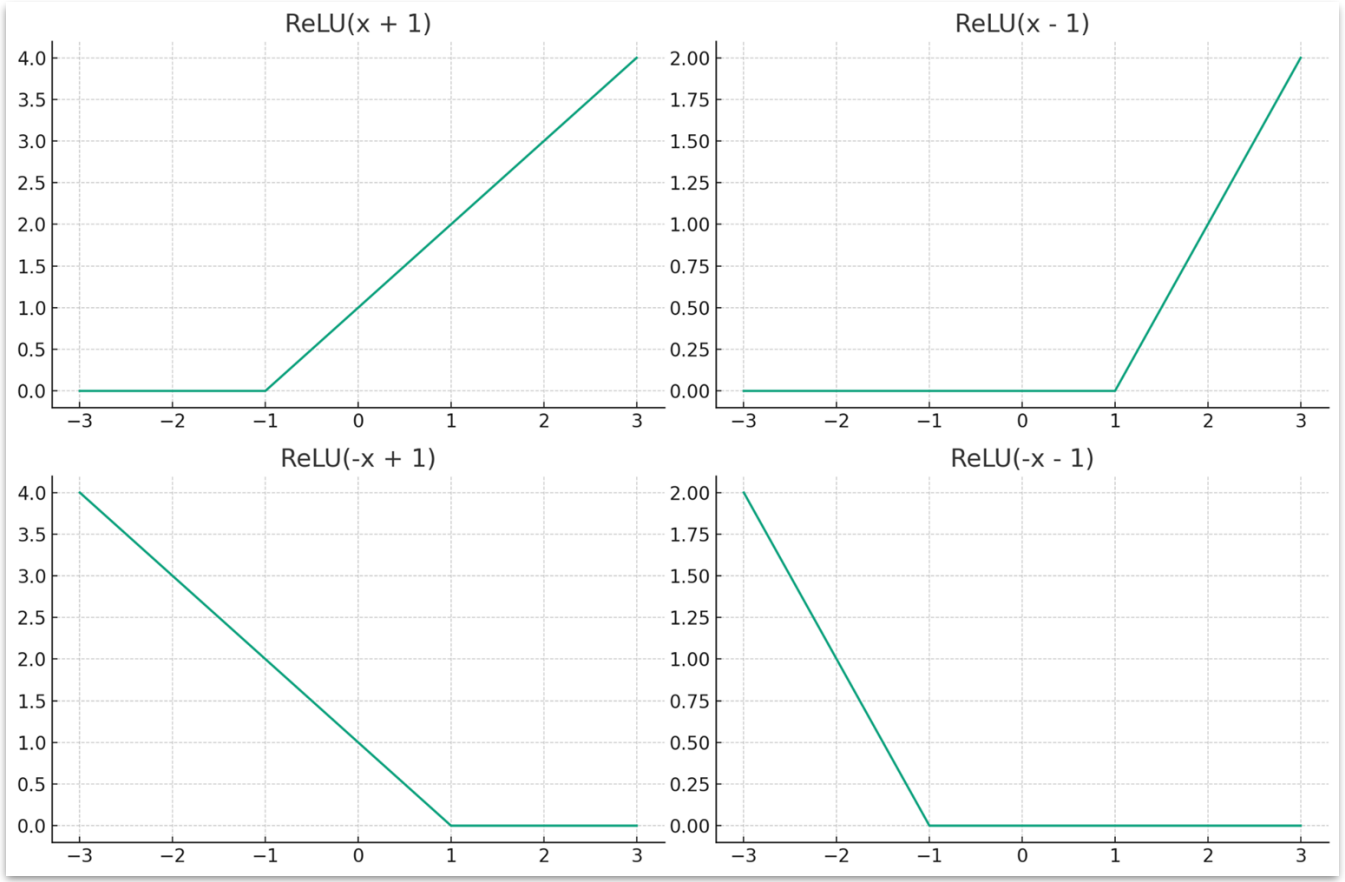
$$y = \max(0, wx + b)$$

The weights and bias reshape the ReLU function.

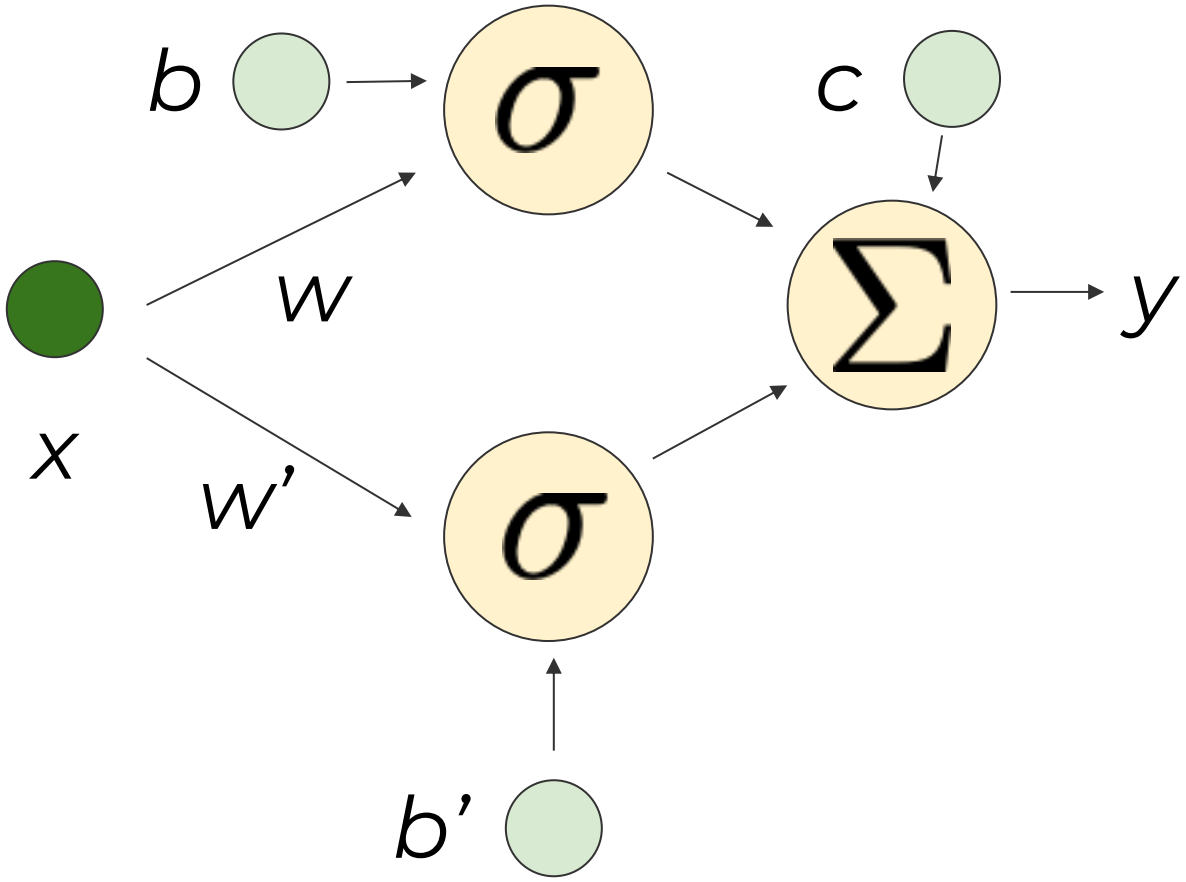
ReLU Variants

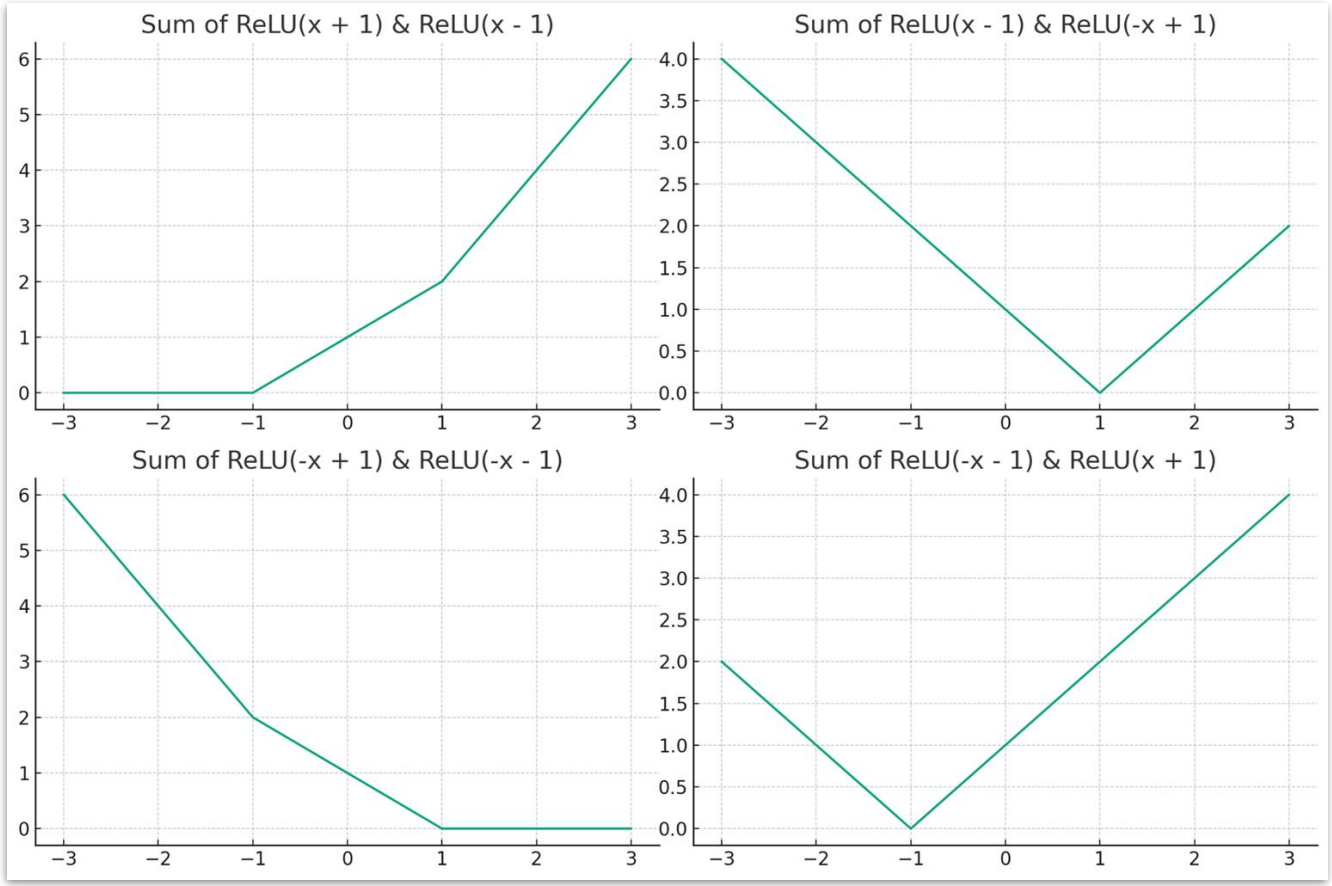


Doesn't seem obviously better than fitting to a line.



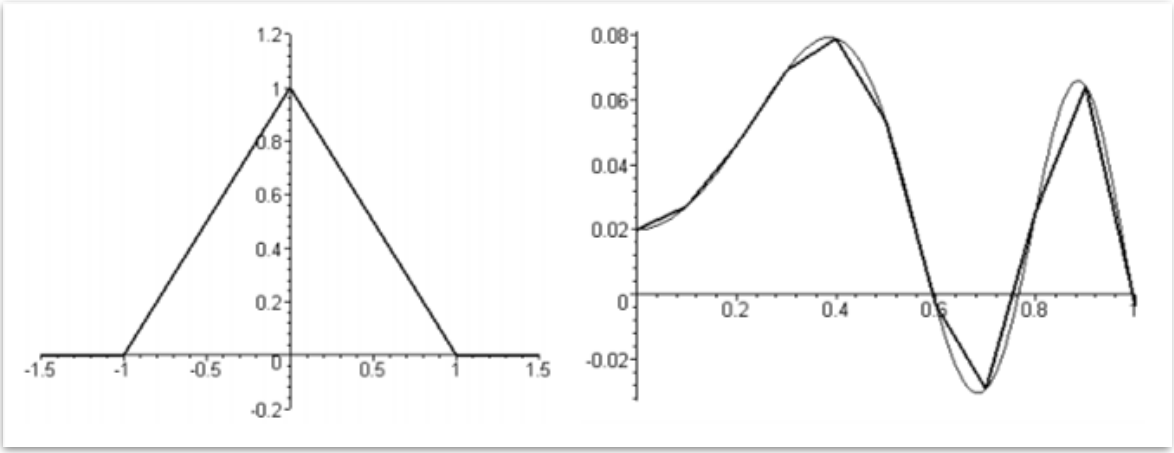
Wider ANN: Two ReLUs Summed



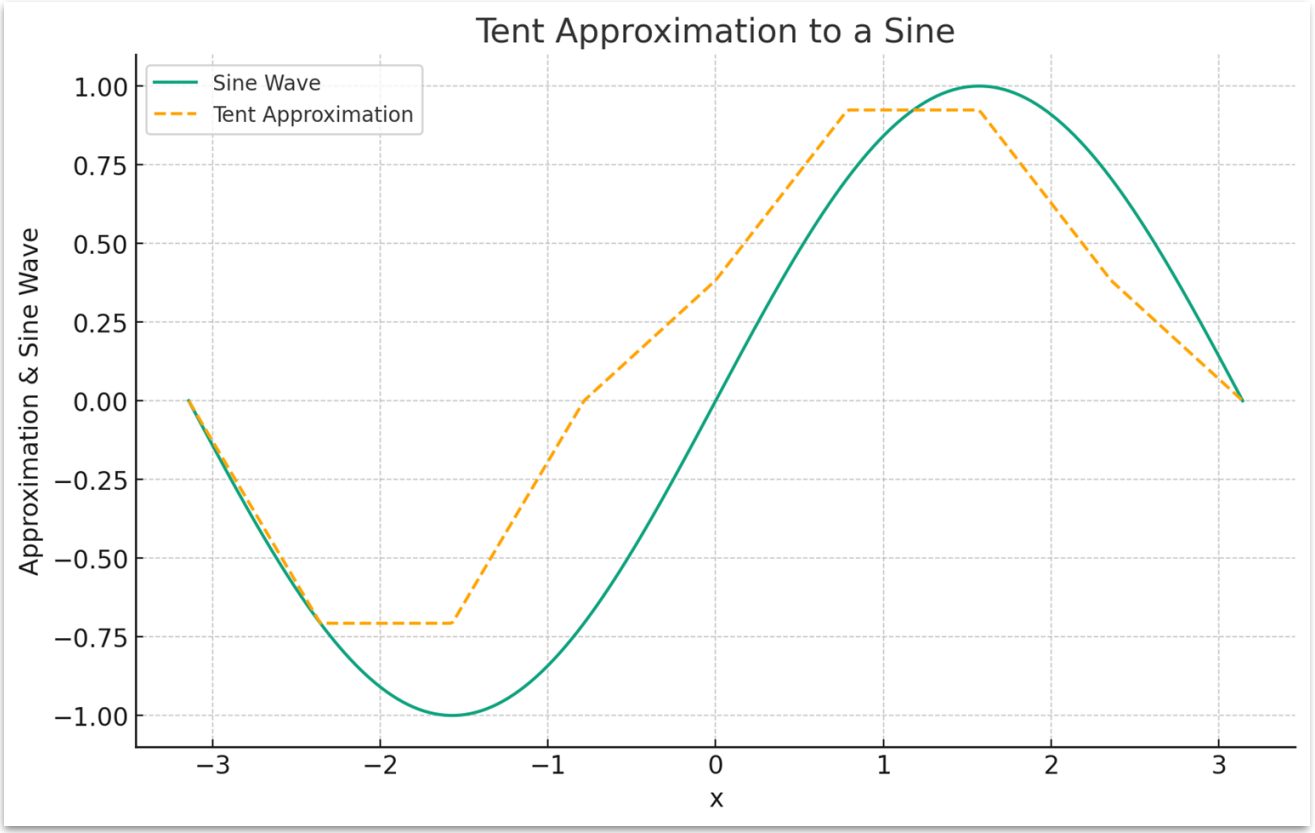


Even Wider ANN

An even wider layer leads to many tent functions, all of which can combined.



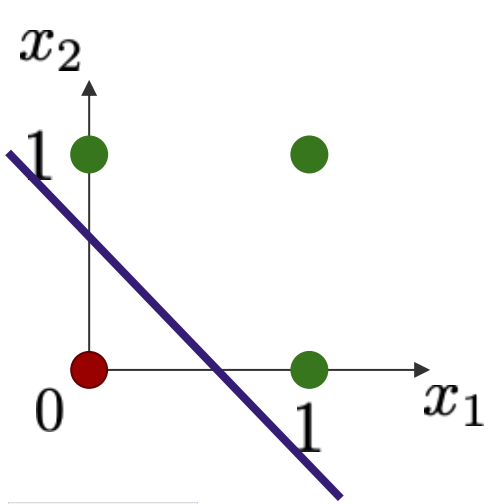
sum of tents

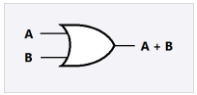


The XOR Problem

An obvious test of ANNs is to the basic building blocks of logic: OR, AND, NOT, etc.

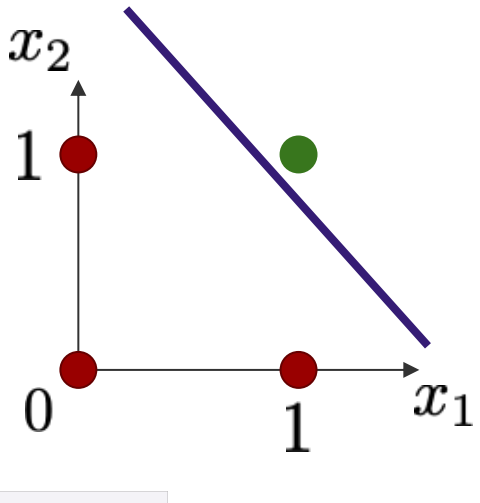
For simplicity, let's just consider the four corners.

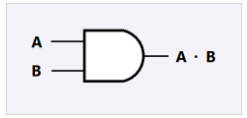




OR

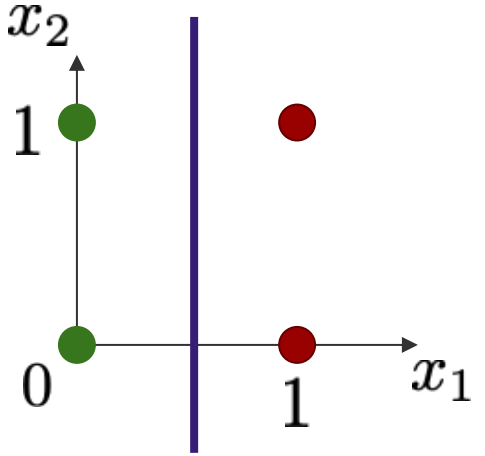
Reverse colors to get **NOR**.

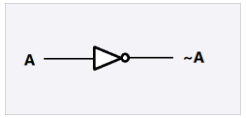




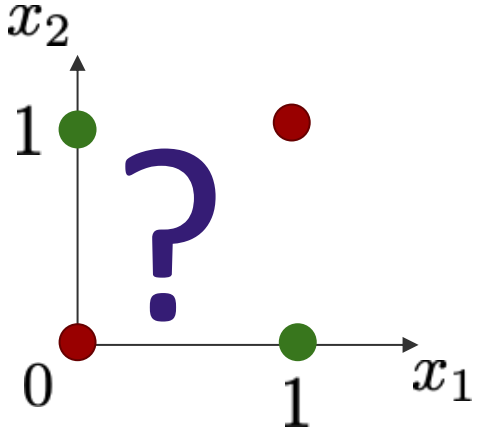
AND

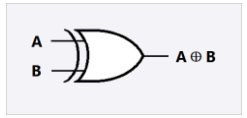
Reverse colors to get **NAND**.





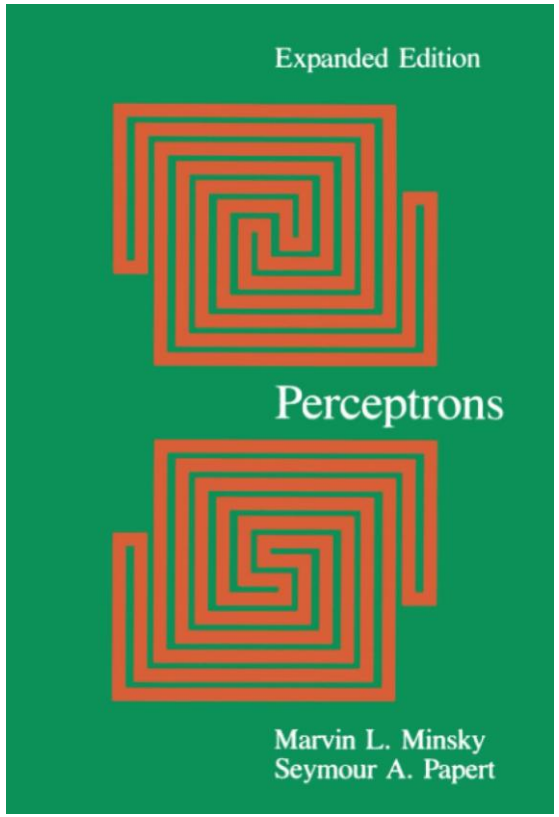
NOT





XOR

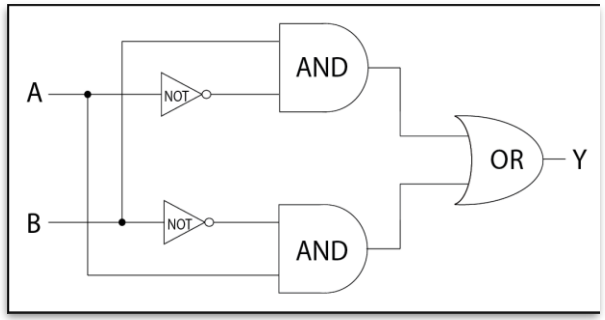
The AI Winter 1970s—1980s

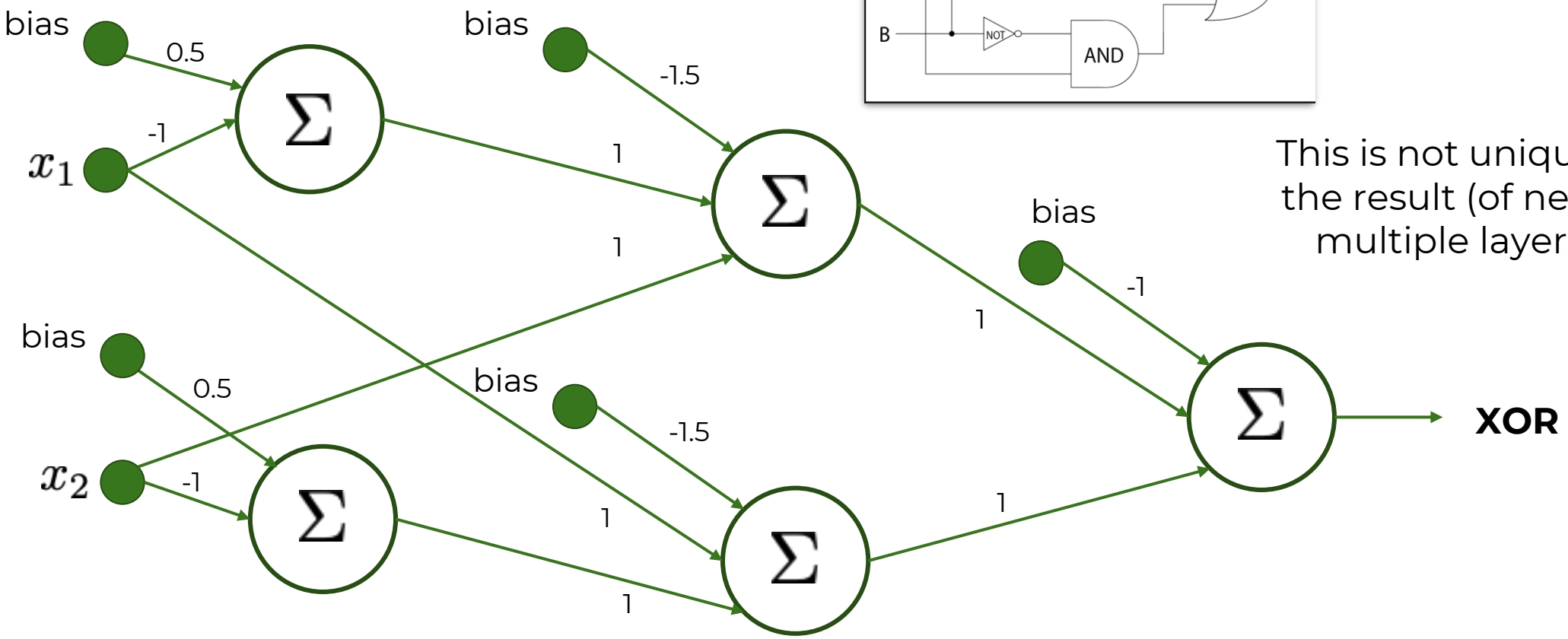




This Photo by Unknown Author is licensed under CC BY-NC

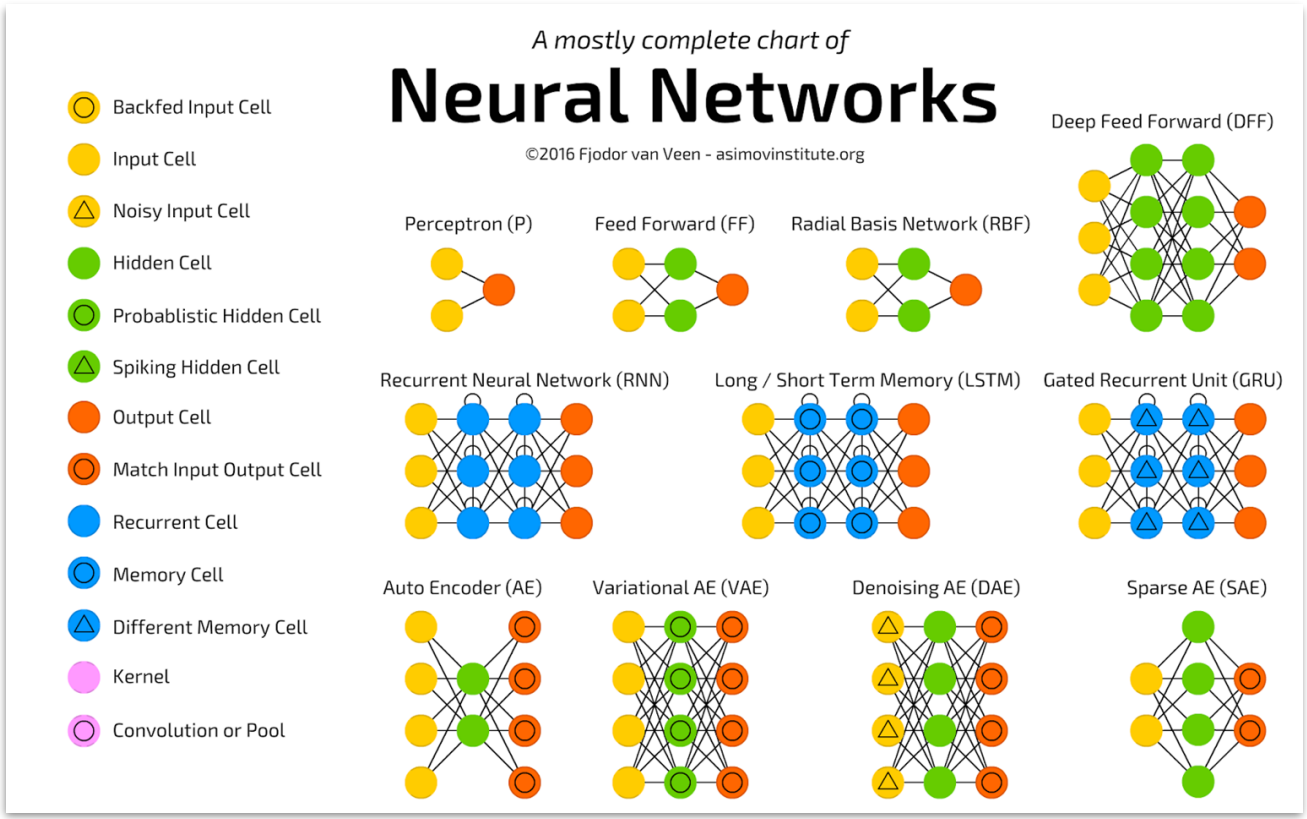
Add more layers

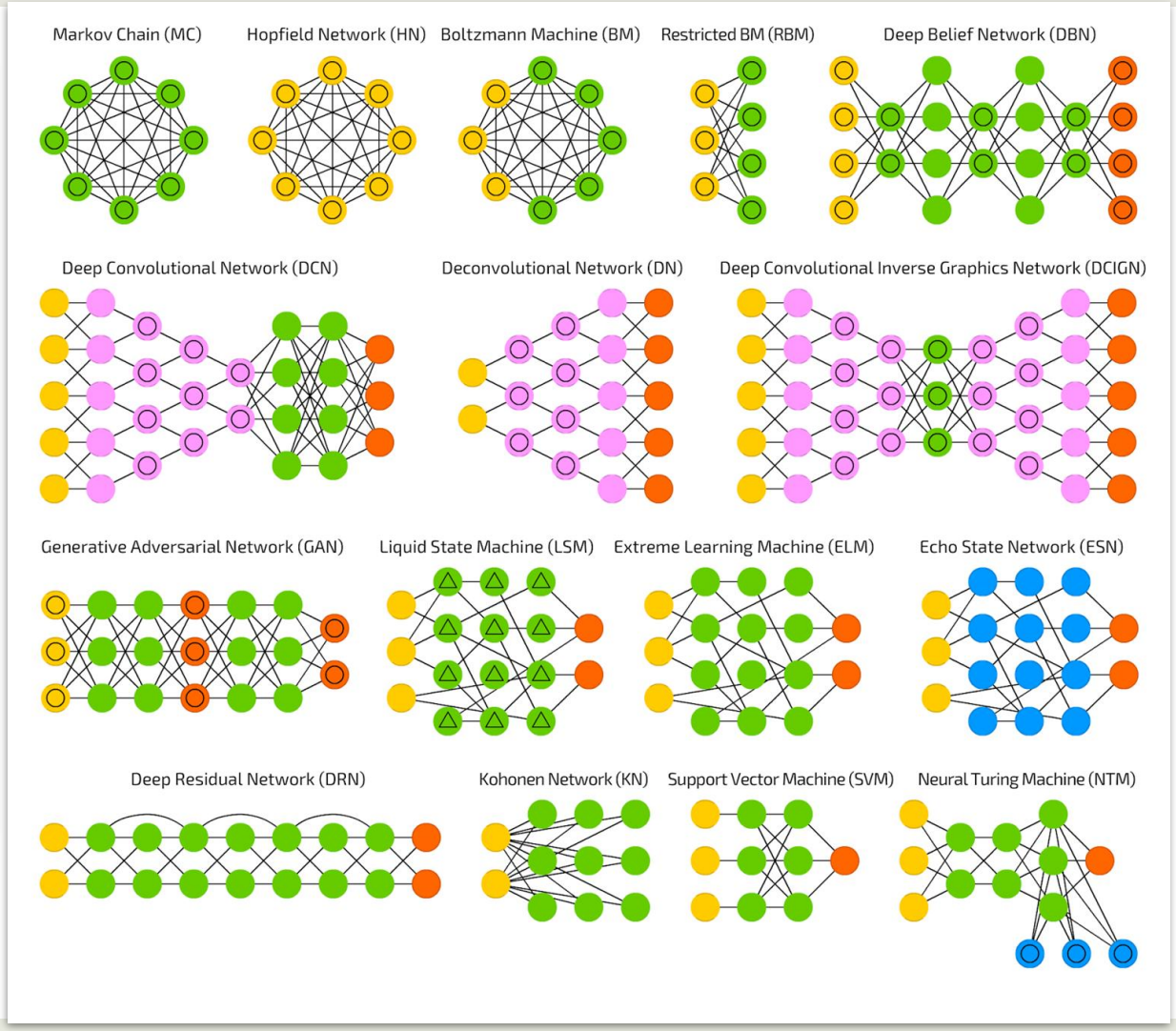




This is not unique, but the result (of needing multiple layers) is.

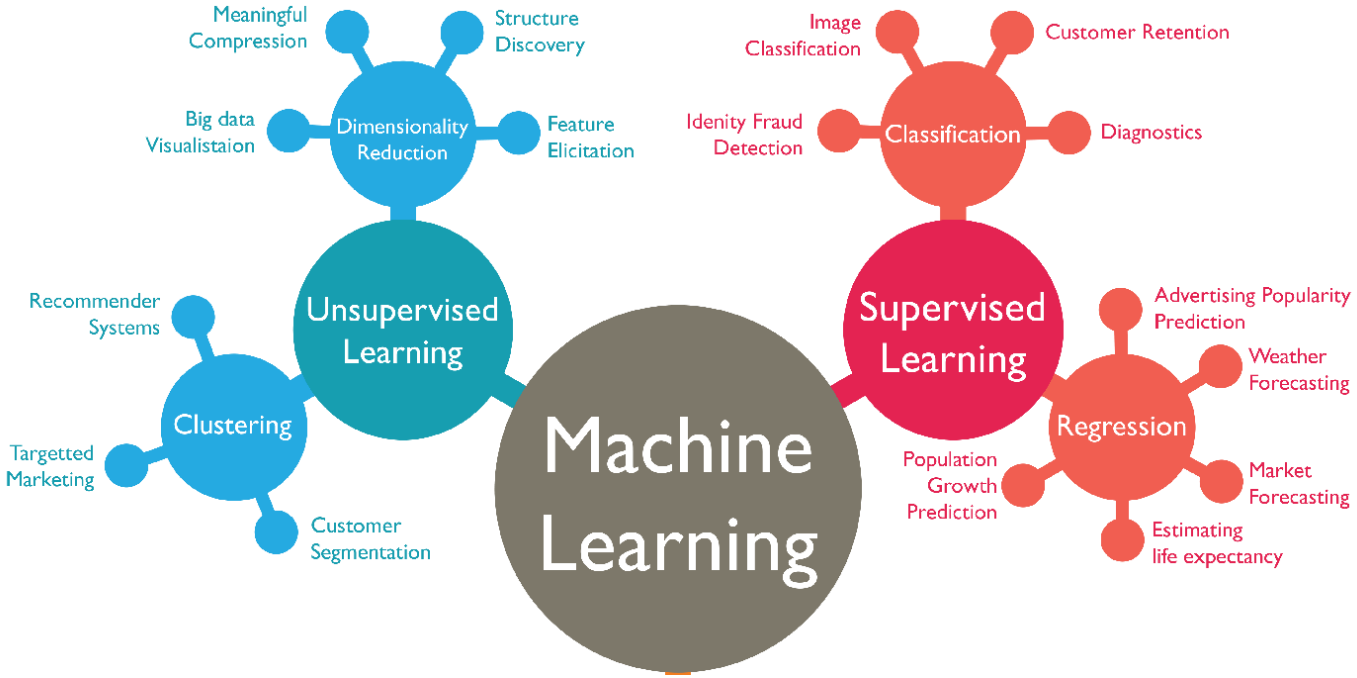
A Neural Network Zoo





Credit

Where are we now?



Supervised Learning

- finding patterns in data according to labels

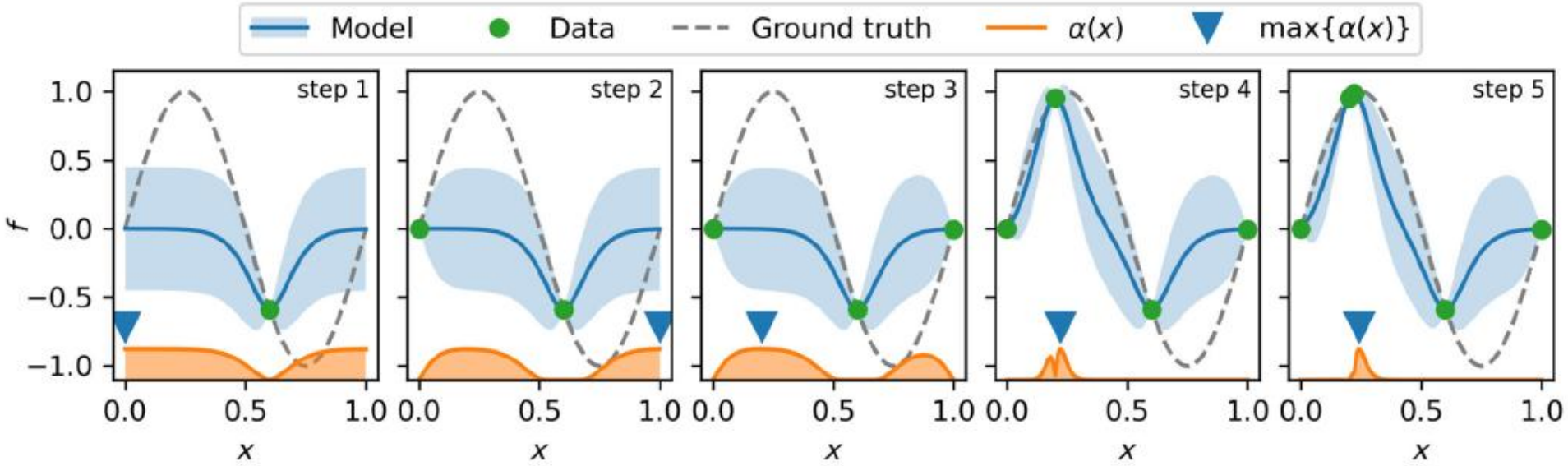
Unsupervised Learning

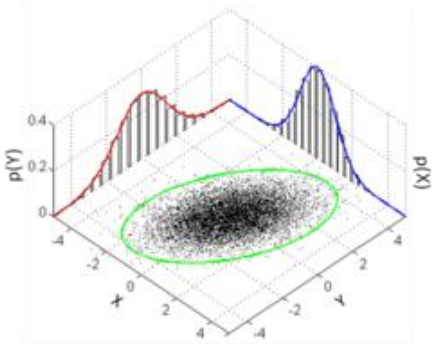
- finding patterns in data according to rules

What's left?!

labeled data, unlabeled data and **no data**!

Bayesian Optimization





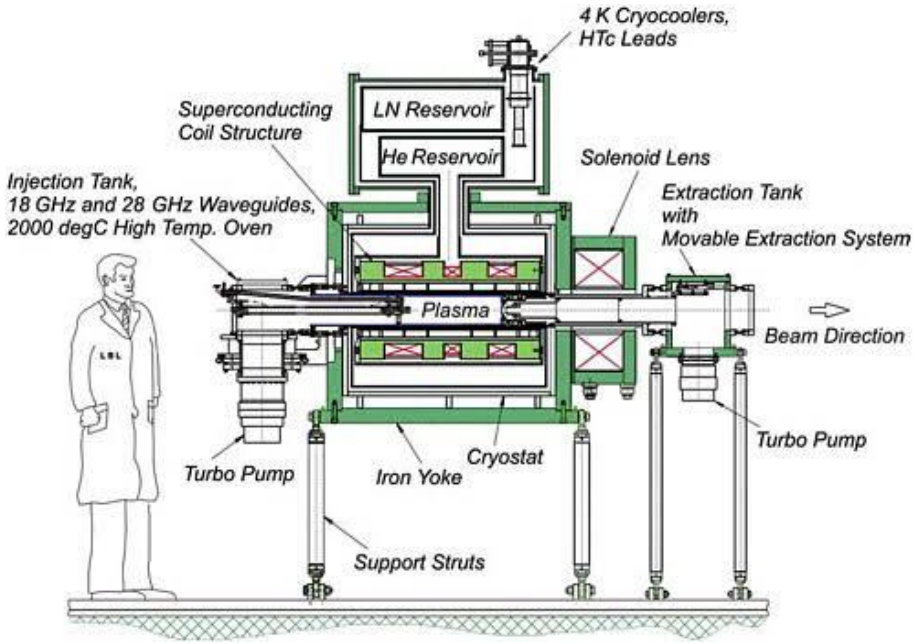
$$f_{\mathbf{x}}(x_1, \ldots, x_k) = \frac{\exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right)}{\sqrt{(2\pi)^k|\boldsymbol{\Sigma}|}}$$

Roussel *et al*, *Phys. Rev. Accel. Beams 27, 084801 (2024)*

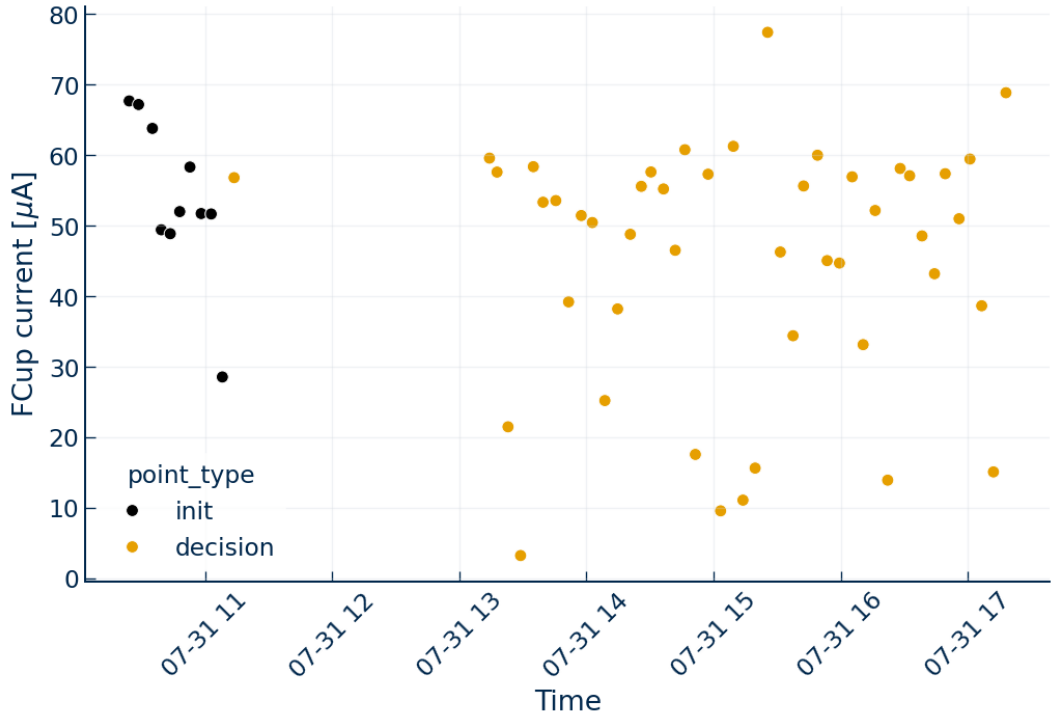
BO for ECR tuning at FRIB

Many knobs to tune

- Bias Disk Voltage
- SC Solenoids
- Gas Valve
- 18GHz RF Power



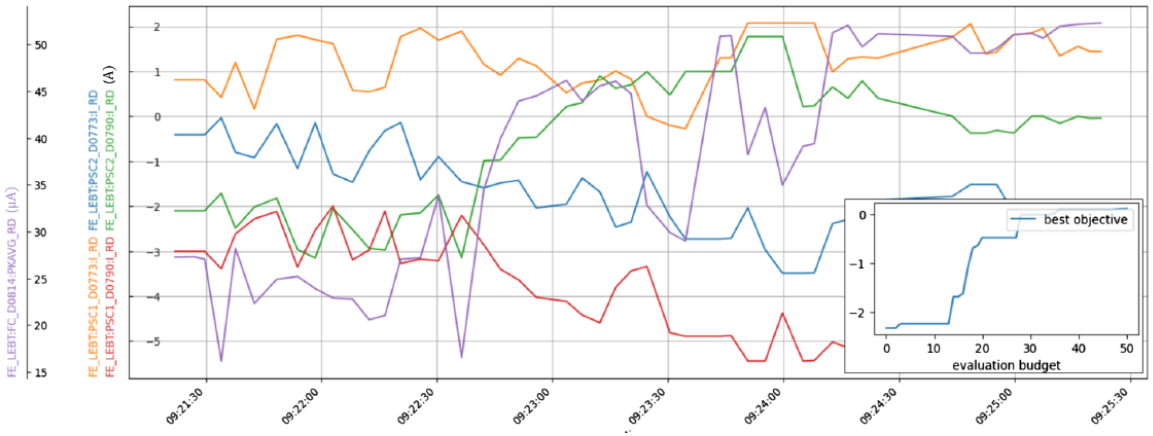
Goal: Find the configuration with a current larger than some threshold



BO for beam tuning at FRIB

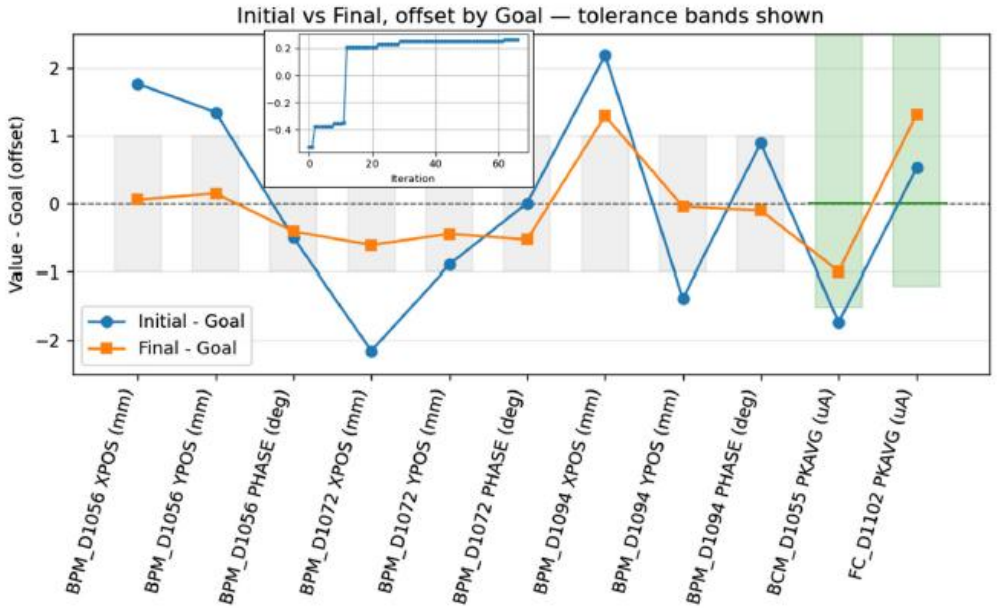
Beam Centering in the U-LEBT

Maximize the current at Faraday cup by controlling corrector magnets



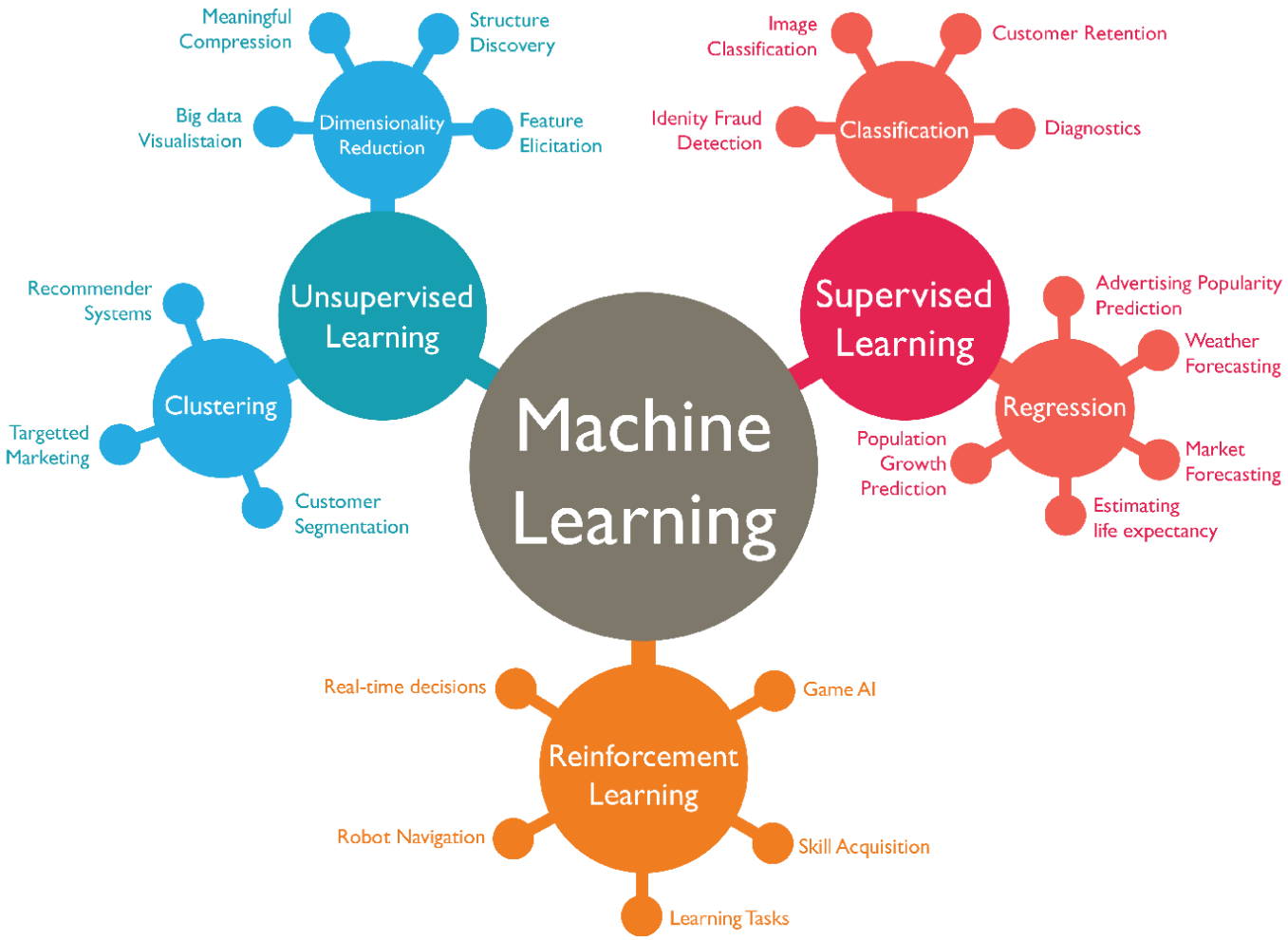
Beam Centering in MEBT

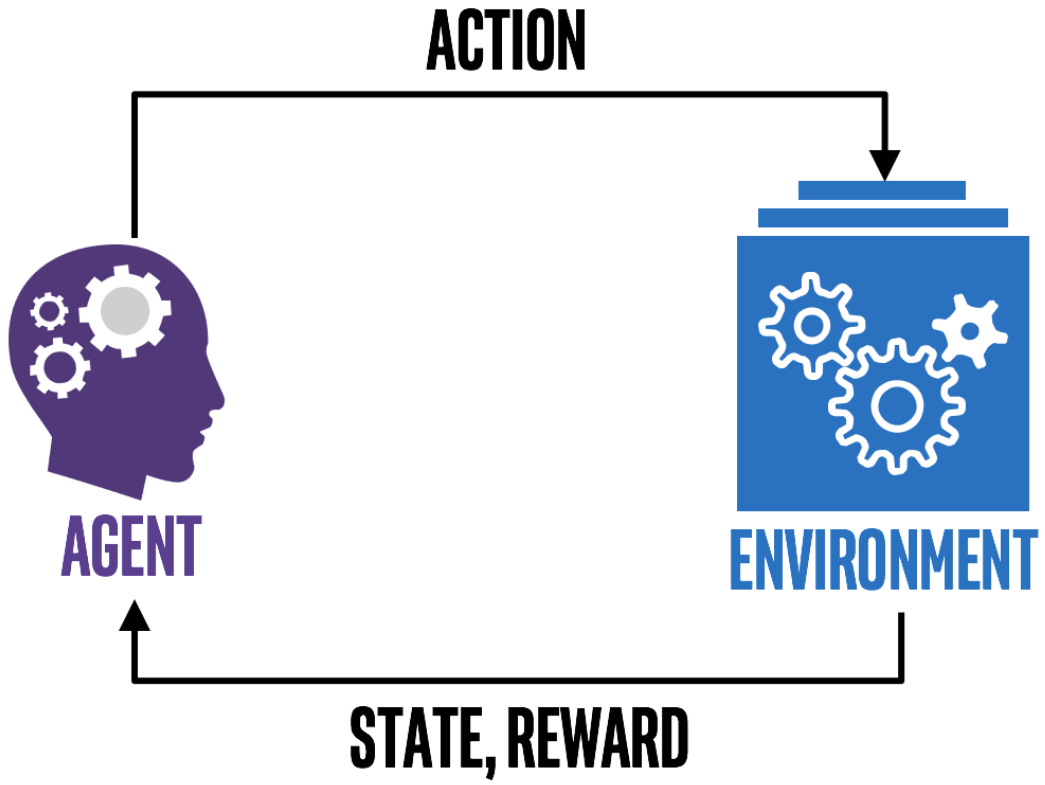
Match beam properties at three BPMs



Hwang *et al*, *Phys. Rev. Accel. Beams 29, 024601 (2026)*

Reinforcement Learning





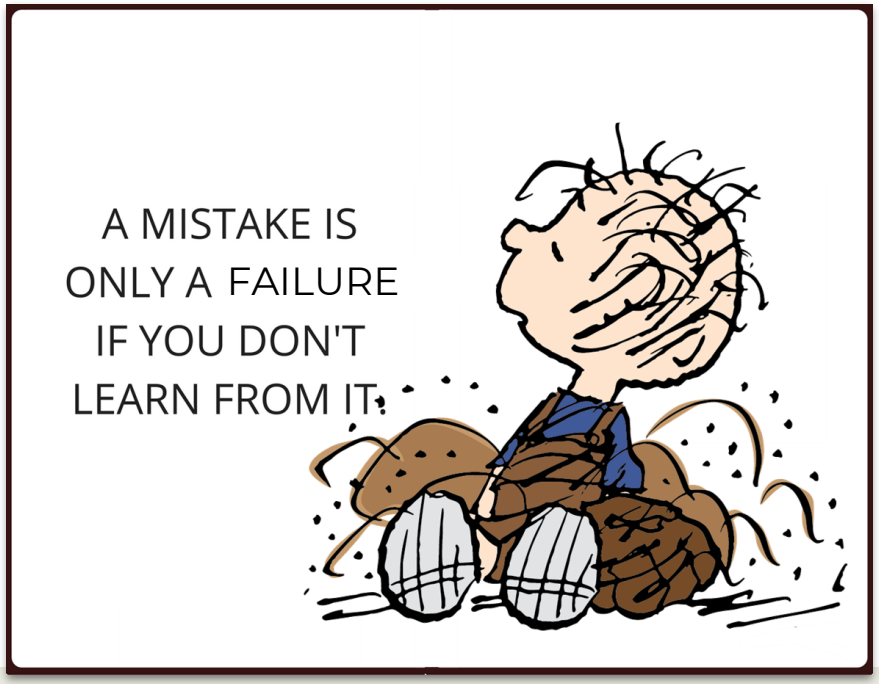
Basic Idea of RL: Biological Connection

We arrive as a “simple” expert system. We know how to eat, sneeze, cough, etc.

We then start trying new things, often making lots of mistakes.







Basics of RL: States, Actions, and Policies

States:

$$\mathcal{S} = \{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_n\}$$

Actions:

$$\mathcal{A} = \{\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_m\}$$

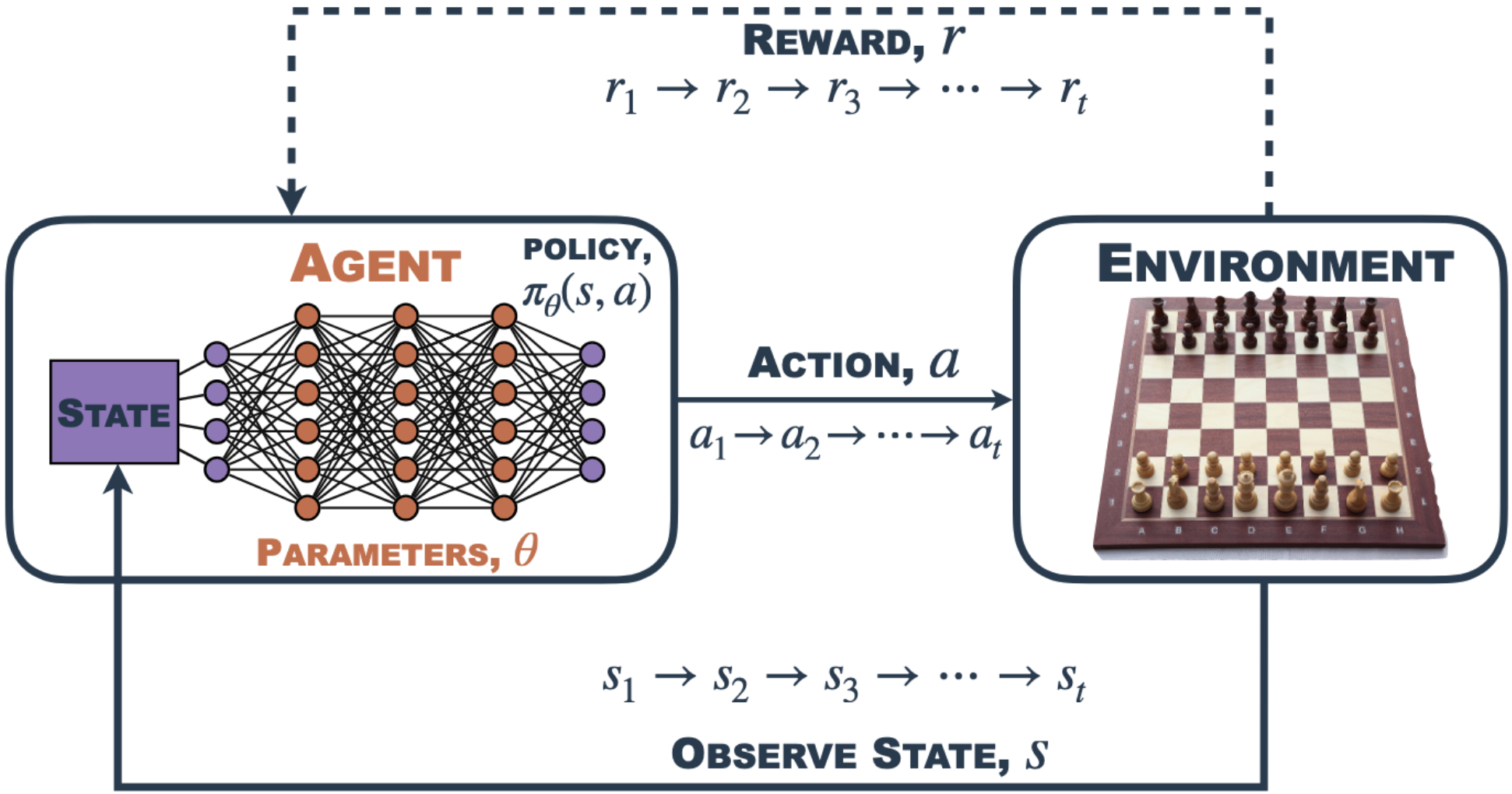
Policy:

$$\boldsymbol{\pi}(\mathbf{a}|\mathbf{s}) = P(\mathbf{A}_t = \mathbf{a}|\mathbf{S}_t = \mathbf{s})$$

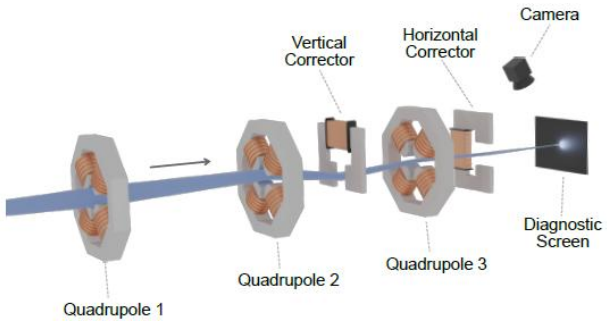
Policy are the rules (or algorithm) that tell me what actions to take given that I am in state **s**

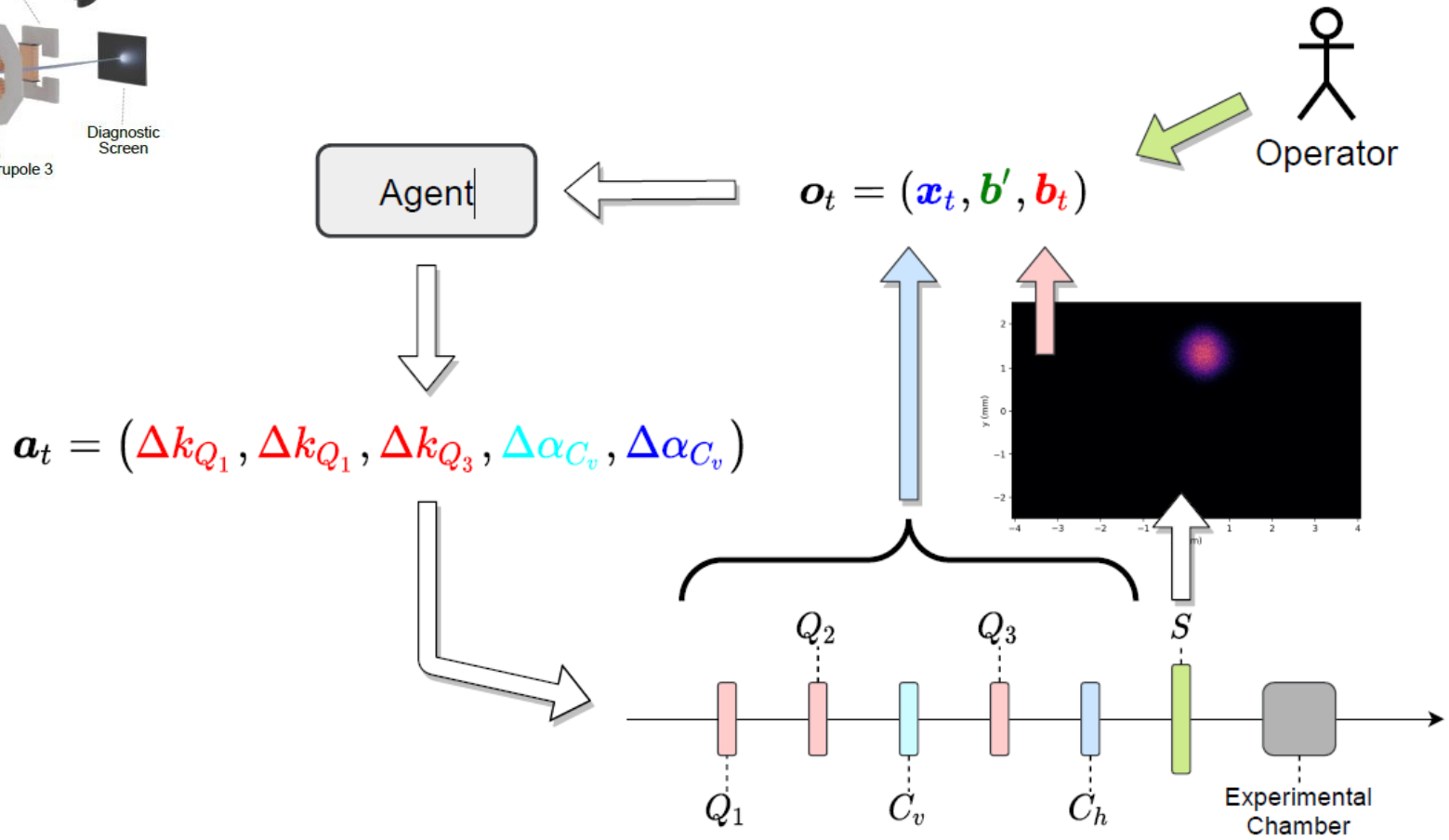
Ex: if statements → I want to do better than if statements

Deep Learning (DL) and Reinforcement Learning (RL)



RL for Beam Tuning

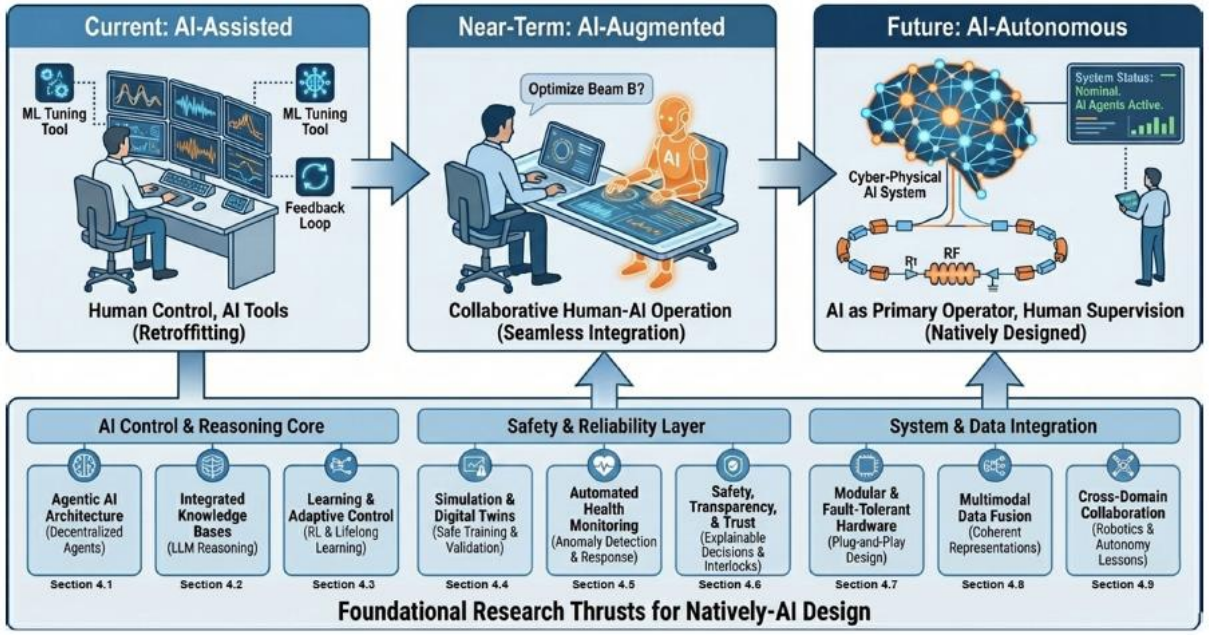




Kaiser *et al*, *Proc Mach Learn Res 162 105775 (2022)*

Where are we going ?

TOWARD A FULLY AUTONOMOUS, AI-NATIVE PARTICLE ACCELERATOR

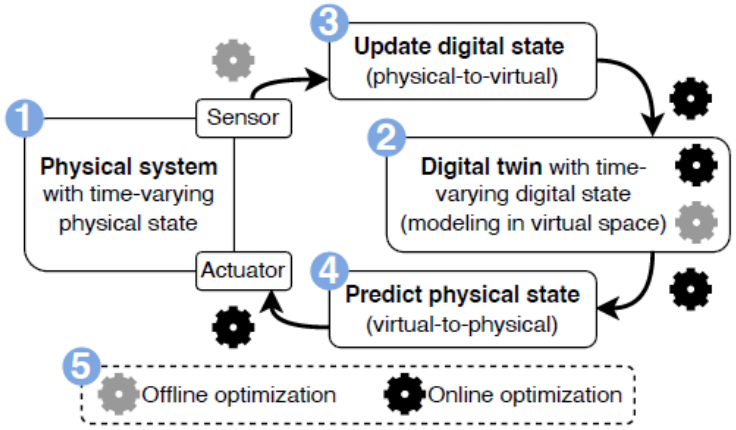


Tennant, arXiv 2602.1753v1

How do we get there?

Digital Twin and Virtual Accelerators

Non-physical copy of a real-world system that closely mirrors its properties and behaviors in a digital format.

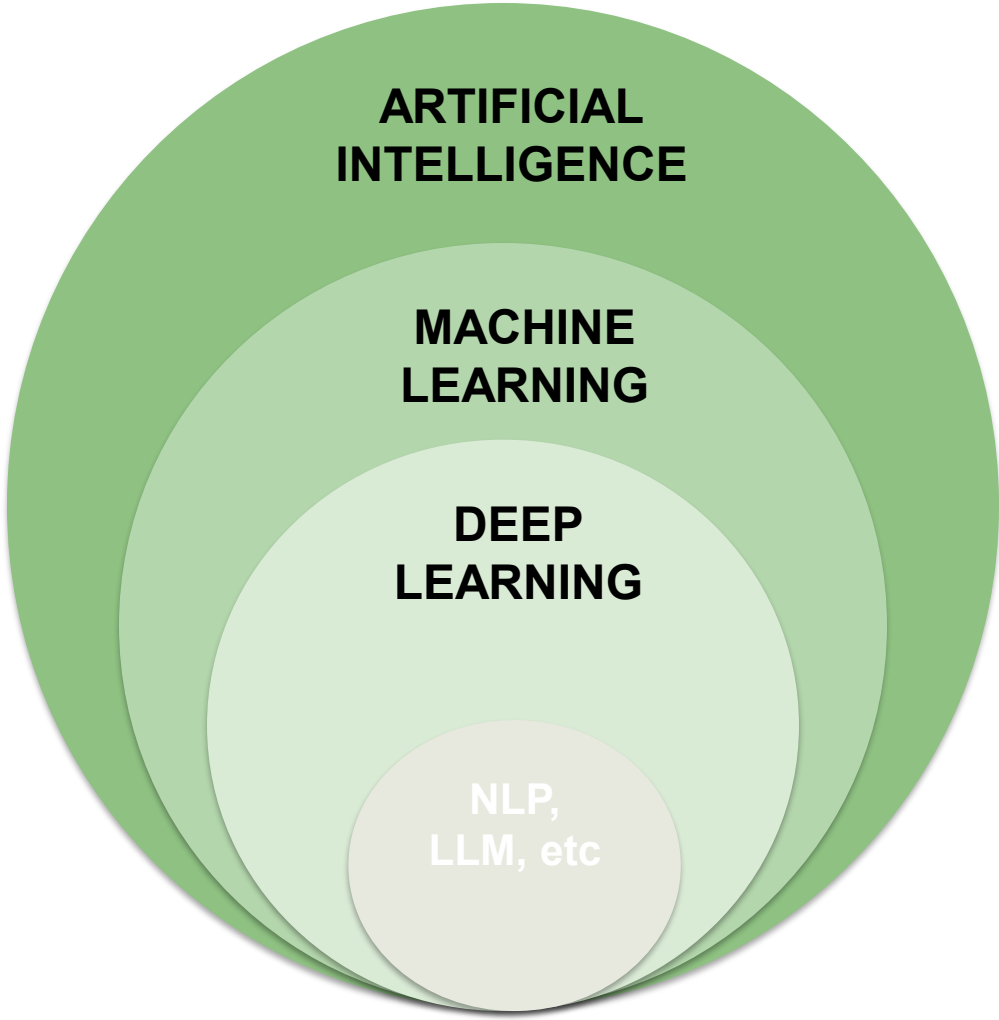


Miceli *et al*, arXiv 2507.20493



AI, ML, DL, etc…

- **Artificial Intelligence:**
  - Create systems that complete a specific task in the most efficient way, but keep it the goal vague
- **Machine Learning:**
  - A set of statistical techniques that enables machines to improve at tasks with experience
- **Deep Learning:**
  - A subset of ML algorithms that relies on artificial neural networks composed of several hidden layers
    - » NLP: natural language processing
      - LLM: Large Language Model
    - » RAG: Retrieval Augmented Generation
    - » Etc..



AI, ML, DL, etc…

- **Artificial Intelligence:**
  - Create systems that complete a specific task in the most efficient way
- **Machine Learning:**
  - A set of statistical techniques that enables machines to improve at tasks with experience
- **Deep Learning:**
  - A subset of ML algorithms that relies on artificial neural networks composed of several hidden layers
    - » NLP: natural language processing
      - LLM: Large Language Model
    - » RAG: Retrieval Augmented Generation
    - » Etc..

