Mechanistic Machine Learning for Engineering and Applied Science

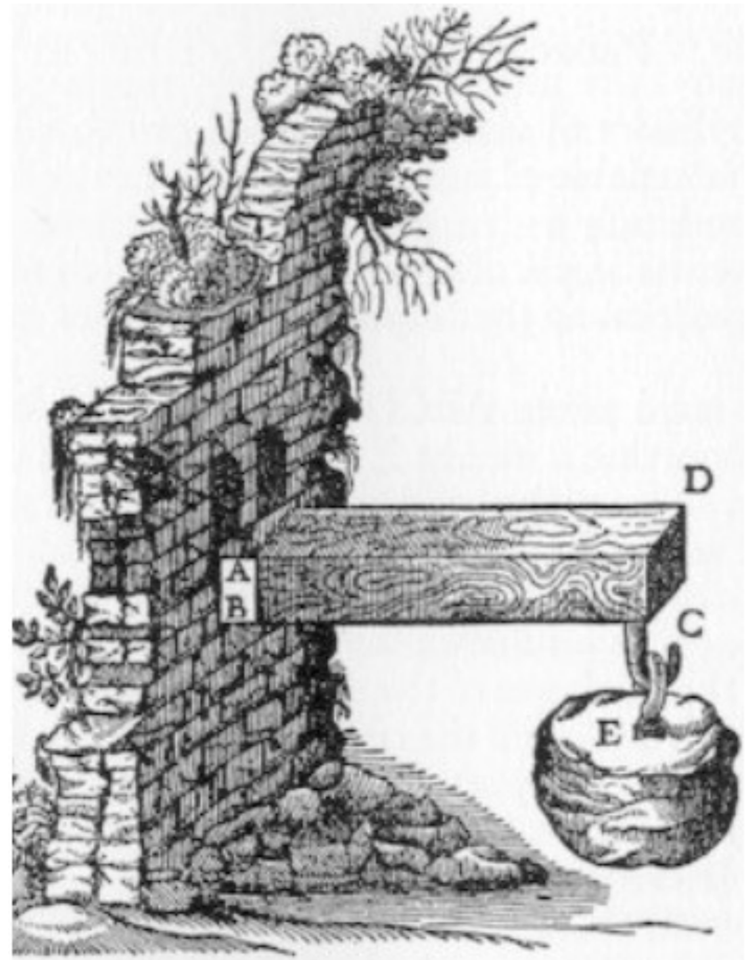
Part I: Introduction and overview

Jiun-Shyan "JS" Chen, William Prager Chair Professor, UC San Diego
WaiChing "Steve" Sun, Associate Professor, Columbia University

What is learning?

"Learning is any process by which a entity improves performance from experience." - Herbert Simon

Example – The ability to deduce F=ma by observing the trajectory of the apple falling from the tree.



What is machine learning?

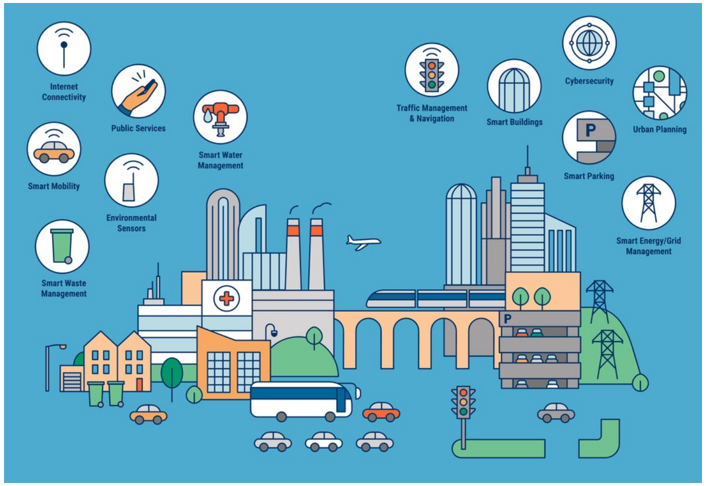
Machine Learning is the study of algorithms that
- improve their performance P
- at some tasks T
- with experience E.

A well-defined learning task is given by <P, T, E> .

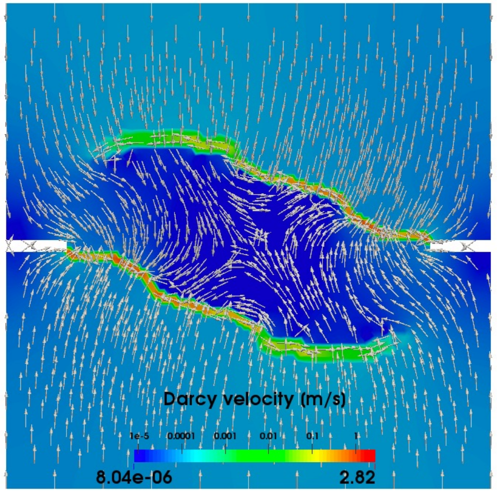
Why Machine Learning?

- Automating automation
- Getting computers to program themselves
- Writing software could be the bottleneck
- Let the data do the work instead!

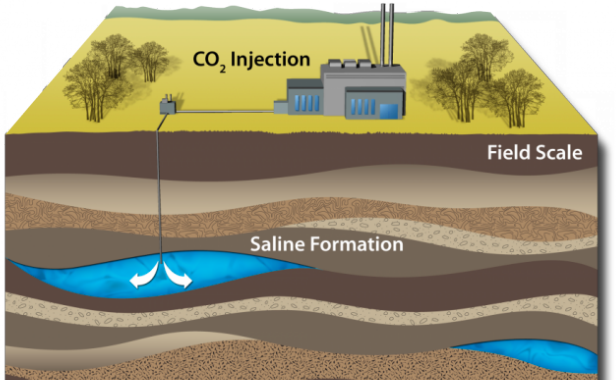
Example Applications in Engineering



Smart City



Material modeling



CO2 geological storage

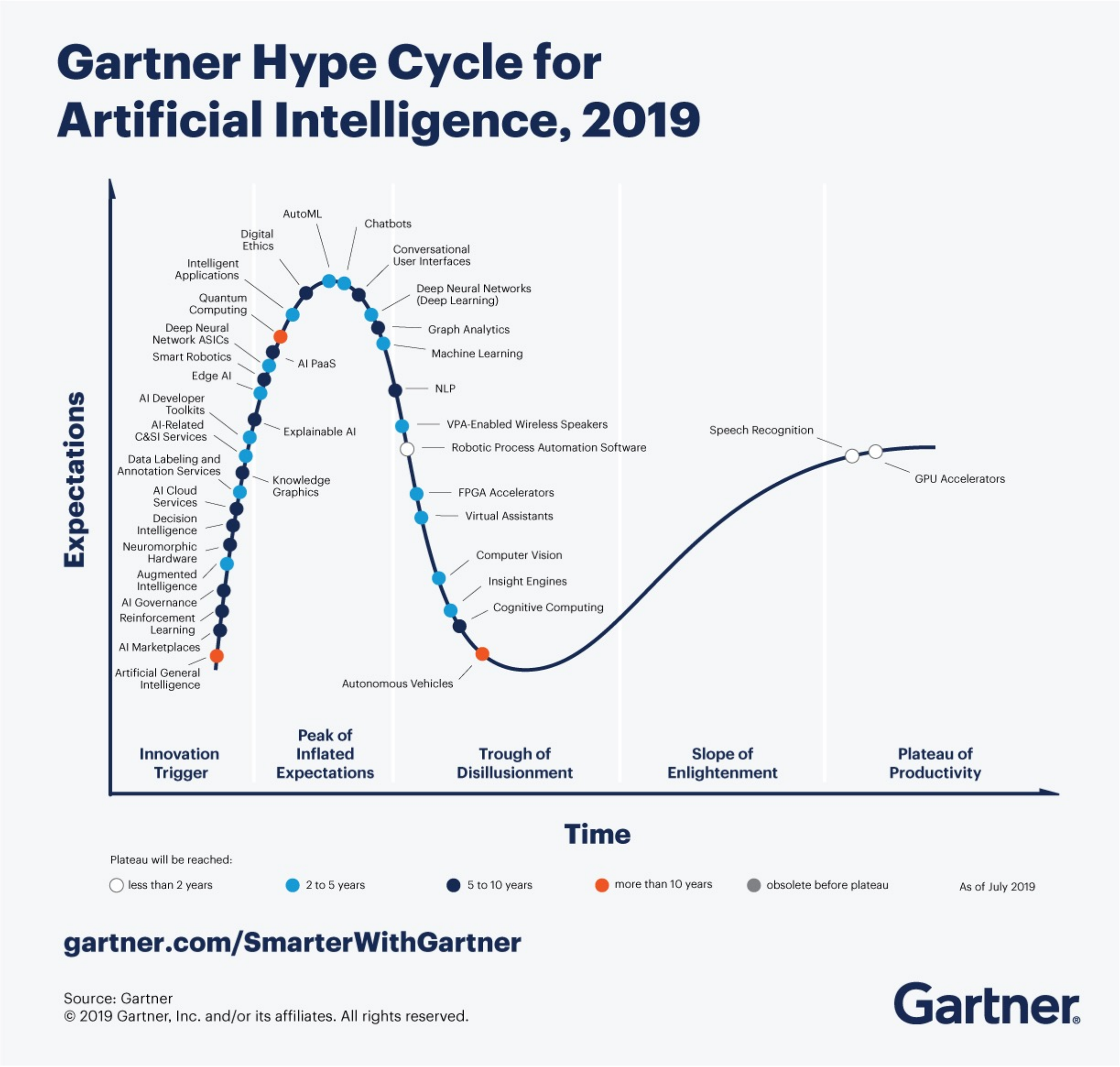
Earthquake predictions

Promises?

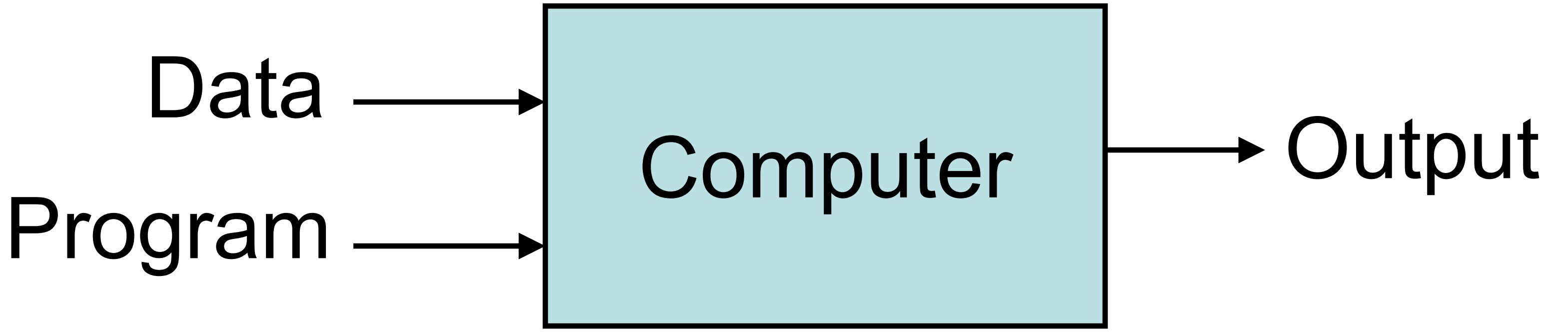
- “A breakthrough in machine learning would be worth ten Microsofts” (Bill Gates, Chairman, Microsoft)
- “Machine learning is the next Internet” (Tony Tether, Director, DARPA)
- Machine learning is the hot new thing” (John Hennessy, President, Stanford)
- “Machine learning is going to result in a real revolution” (Greg Papadopoulos, CTO, Sun)

BUT

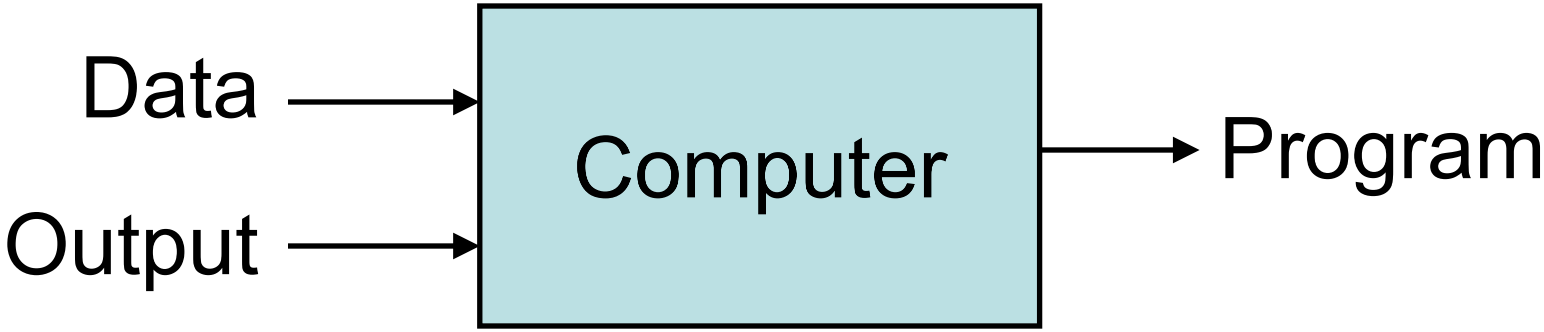
Hype?


Gartner Hype Cycle for Artificial Intelligence, 2019
Expectations
AutoML
Chatbots
Digital Ethics
Conversational User Interfaces
Intelligent Applications
Deep Neural Networks (Deep Learning)
Quantum Computing
Graph Analytics
Deep Neural Network ASICs
Machine Learning
Smart Robotics
AI PaaS
Edge AI
NLP
AI Developer Toolkits
AI-Related C&SI Services
Explainable AI
VPA-Enabled Wireless Speakers
Robotic Process Automation Software
Speech Recognition
Data Labeling and Annotation Services
Knowledge Graphics
GPU Accelerators
AI Cloud Services
FPGA Accelerators
Decision Intelligence
Virtual Assistants
Neuromorphic Hardware
Computer Vision
Augmented Intelligence
Insight Engines
AI Governance
Cognitive Computing
Reinforcement Learning
AI Marketplaces
Artificial General Intelligence
Autonomous Vehicles
Innovation Trigger
Peak of Inflated Expectations
Trough of Disillusionment
Slope of Enlightenment
Plateau of Productivity
Time
Plateau will be reached:
less than 2 years
2 to 5 years
5 to 10 years
more than 10 years
obsolete before plateau
As of July 2019
gartner.com/SmarterWithGartner
Source: Gartner
© 2019 Gartner, Inc. and/or its affiliates. All rights reserved.
Gartner

Traditional Programming



Machine Learning



Example: Rock Scissors Paper Game

Traditional Approach – explicitly programming the rules

```
switch (action)
{
   case “rock” :
      reaction =  “paper”; break;
   case “scissors”:
      reaction = “rock”; break;
   default:
      reaction = “scissors”
}
```

ML Approach – implicitly ”learning” the rules from the patterns

Given: History of game played, e.g.

Game 1: Input = rock-scissors, output = loss
Game 2: Input = scissors-paper, output = loss
Game 3: Input = rock-paper, output = win
…etc

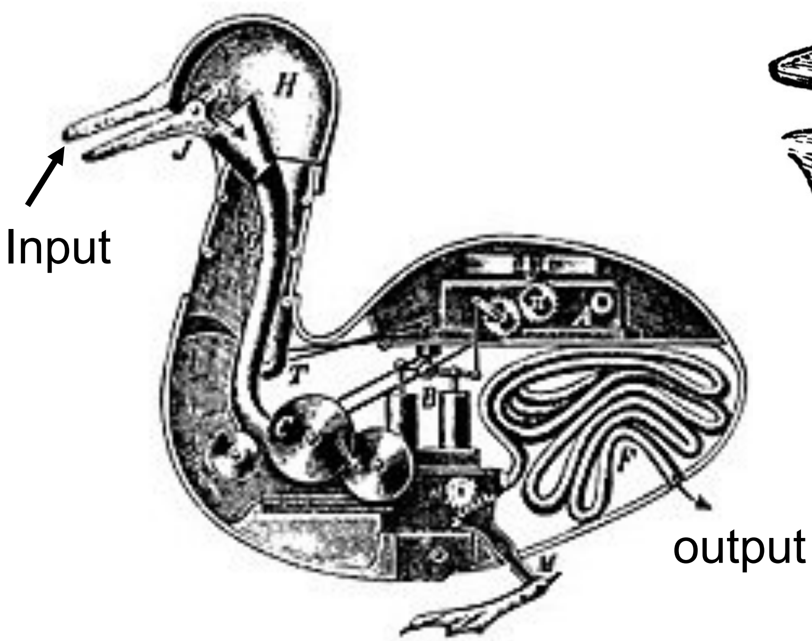
Computer then ”training” the AI by providing the data to the AI

AI learns the “rule” and make predictions

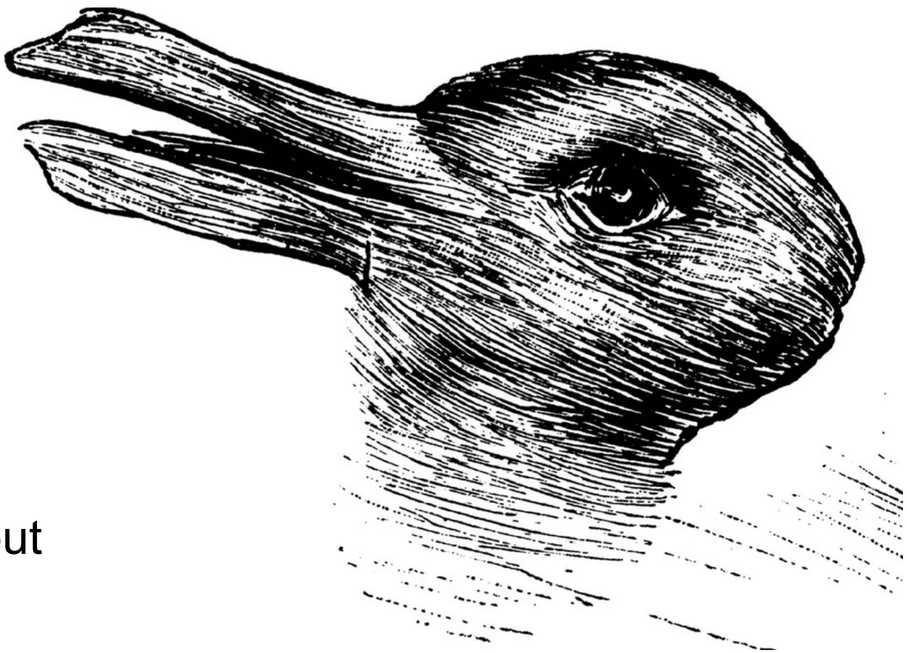
The key difference is that the rule has never been explicitly programmed

Example: “Seeing that” vs. “seeing as”

Rationale of Predictions: External behaviors vs. internal properties



Canard Digérateur (1741)



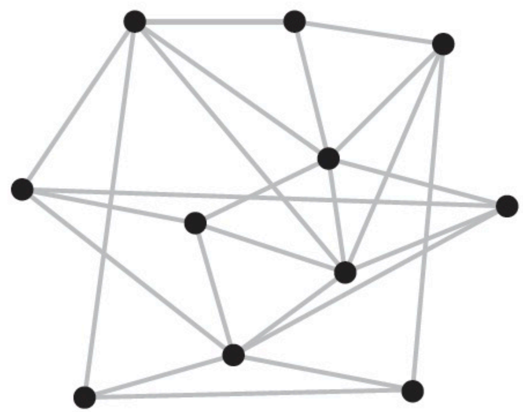
Duck-rabbit (1892)

ML in a Nutshell

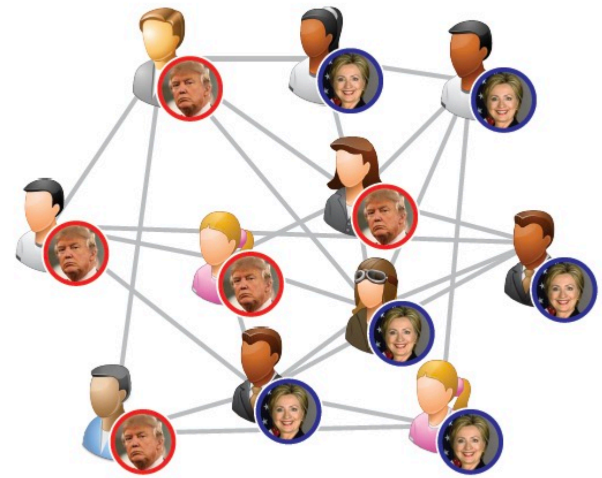
- Every machine learning algorithm has three components:
  - **Representation**
  - **Evaluation**
  - **Optimization**

Example: Data Representation

Consider descriptors of data as the ingredients for theory



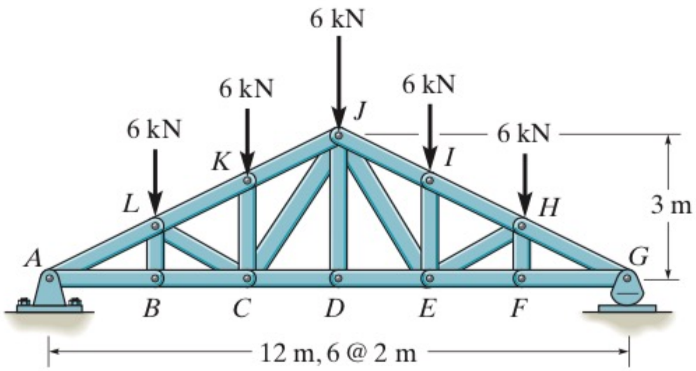
Domain structure

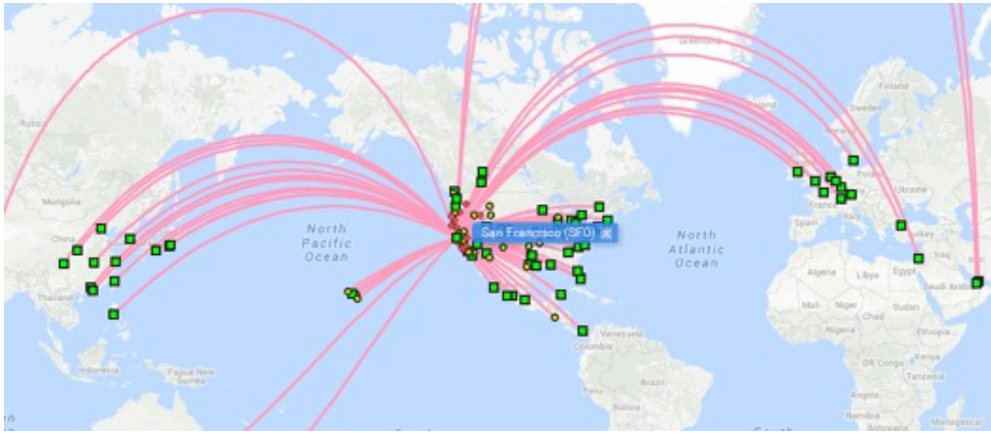


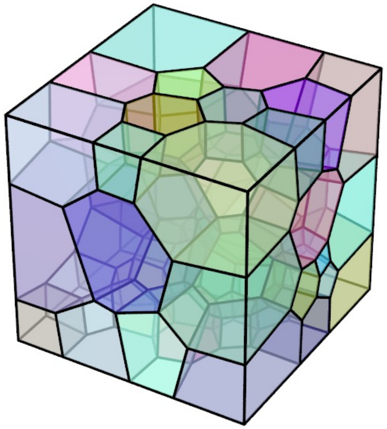
Social network

Figures from geometriclearning.com

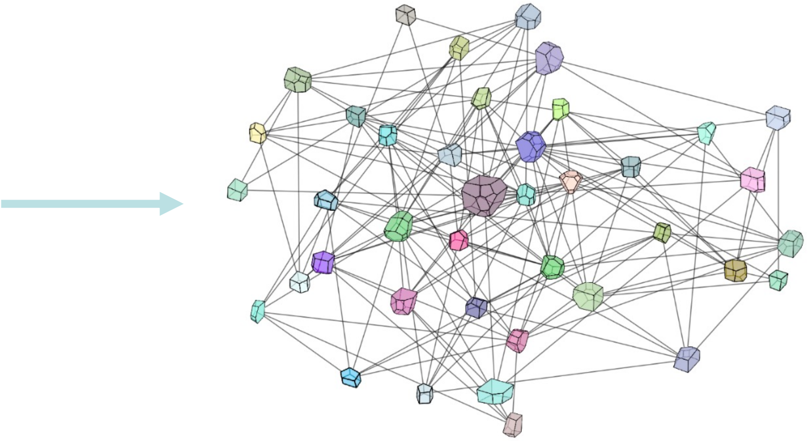
Example for civil engineering and engineering mechanics





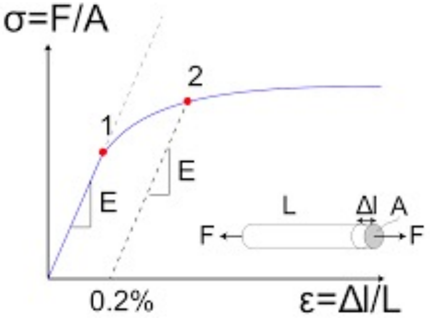


Polycrystal RVE



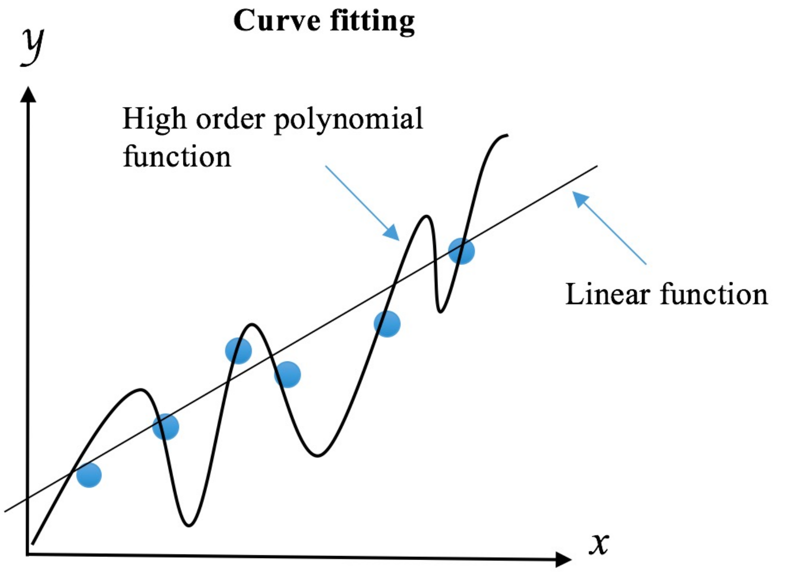
Node-weighted undirected crystal connectivity graph

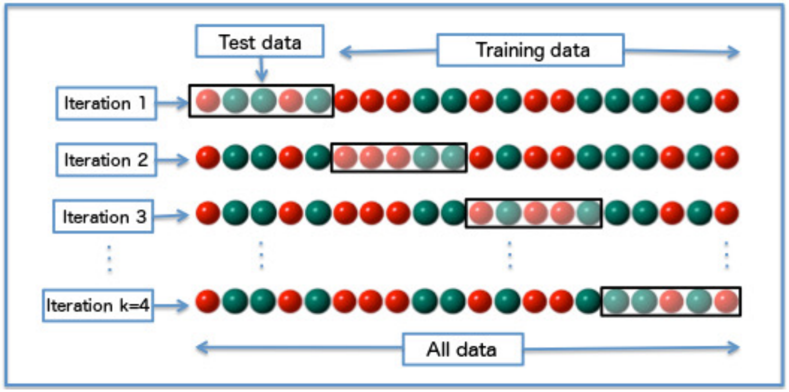
- Constitutive law generation from non-Euclidean grid data



Evaluation

- Accuracy
- Precision and recall
- Squared error
- Likelihood
- Posterior probability
- Cost / Utility
- Margin
- Entropy
- K-L divergence
- Etc.



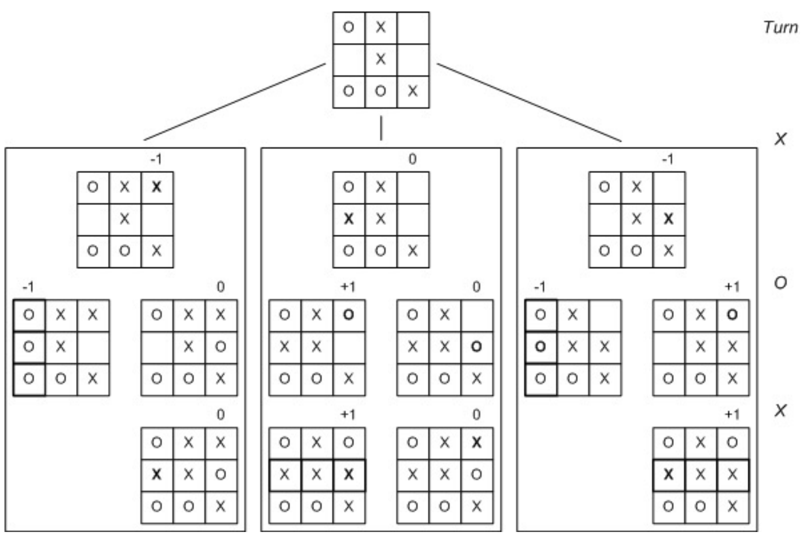


Optimization

- Combinatorial optimization
  - E.g.: Greedy search
- Convex optimization
  - E.g.: Gradient descent
- Constrained optimization
  - E.g.: Linear programming

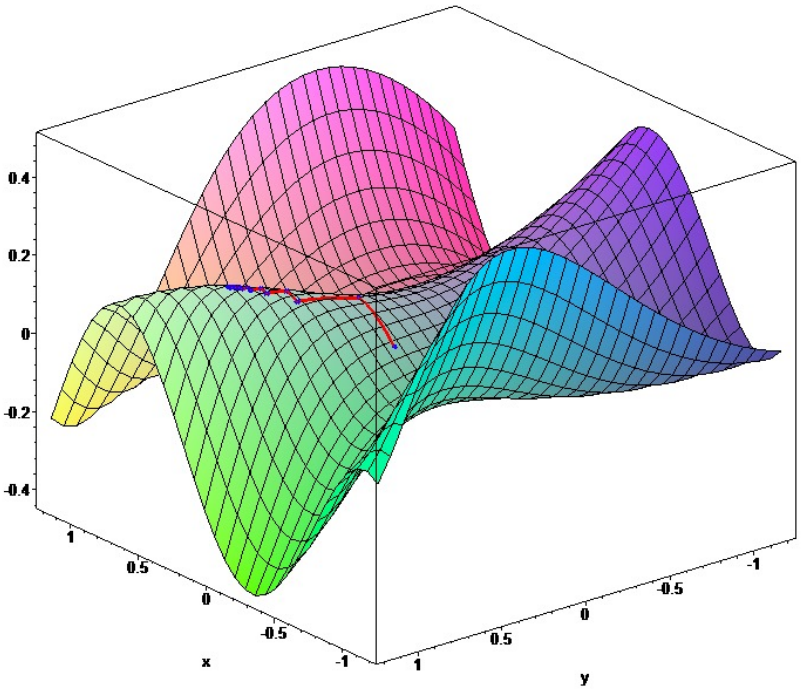
Combinatorial Optimization

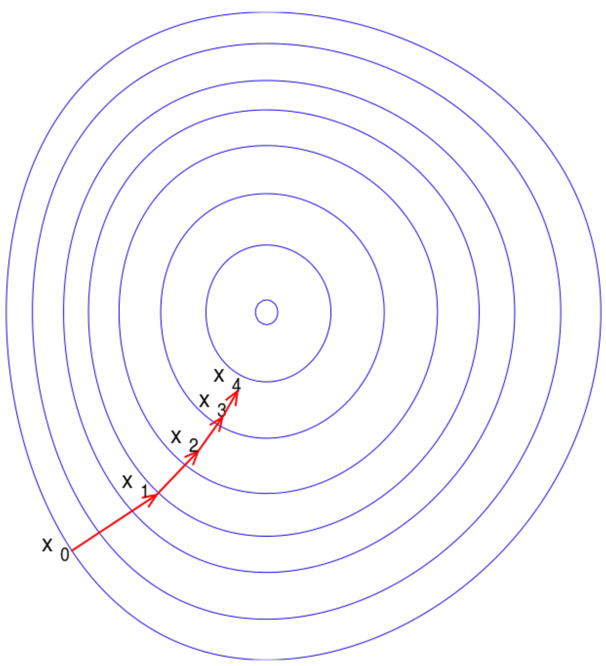
- How to determine elements of a set or n-tuple that maximize an objective function?



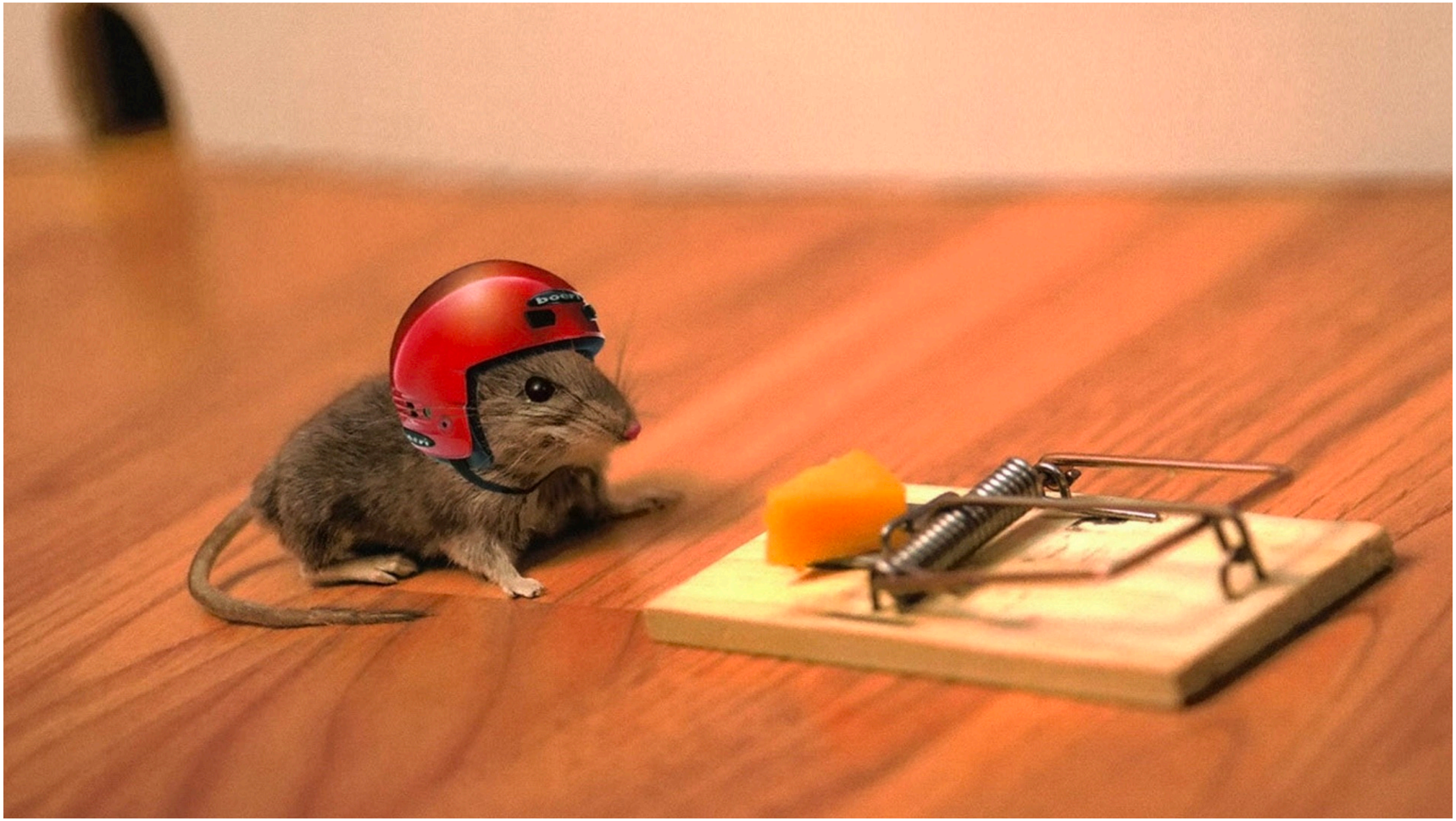
Convex and constrained Optimization

- How to a point in a parametric space that maximize one or multiple objective functions?





Exploitation vs exploration

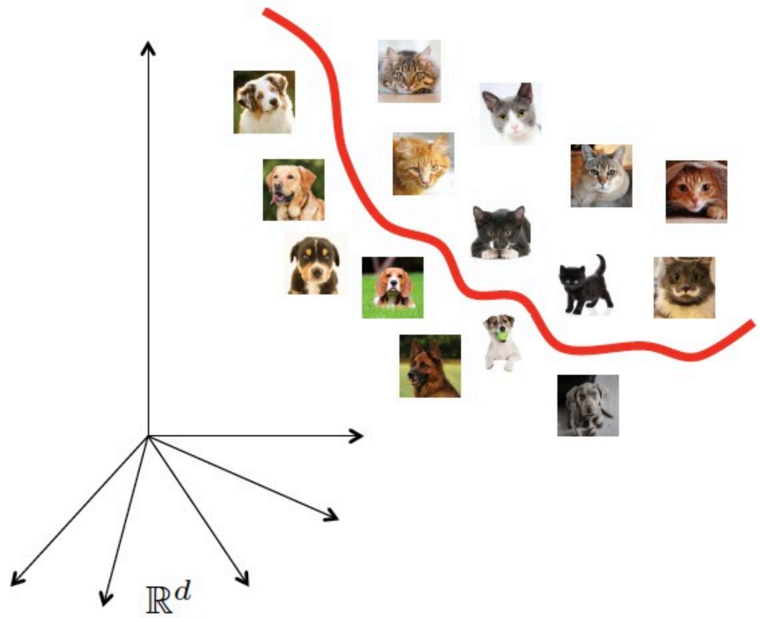


Major branches of machine learning

- **Supervised learning**
  - Rule induction from input/output pairs
  - Learn with labeled data
- **Unsupervised learning**
  - Dimensional reduction
  - Data compression
  - No output required
- **Reinforcement learning**
  - Markov decision process
  - Agent interact with environment
  - Need reward from environment influence actions

Supervised Learning

- **Given** examples of a function *(X, F(X))*
- **Predict** function *F(X)* for new examples *X*
  - Discrete *F(X)*: Classification
  - Continuous *F(X)*: Regression
  - *F(X)* = Probability(*X*): Probability estimation

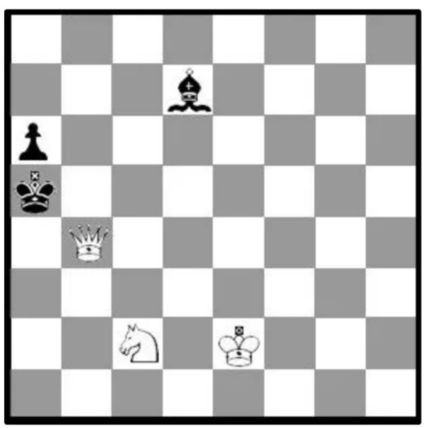


Example: Classification Problems?
Input: pictures of dogs and cats
Outpiut: The label "dog" or "cat"

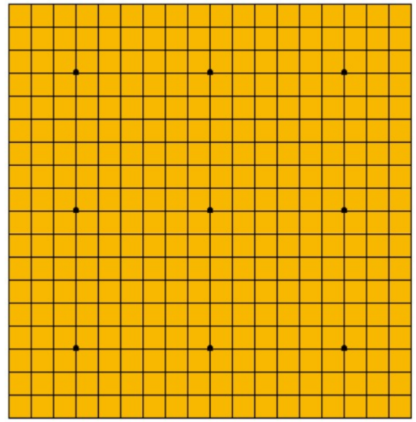
Reinforcement learning and Game

Chess

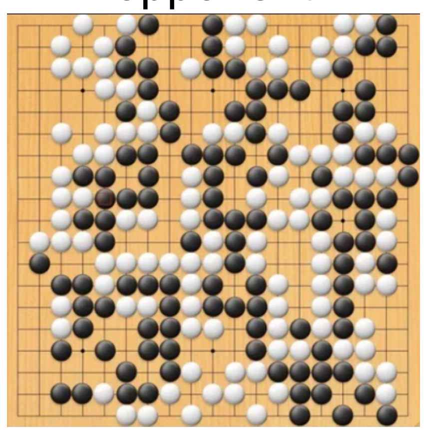
Move pieces to put the opponent's king in "checkmate"



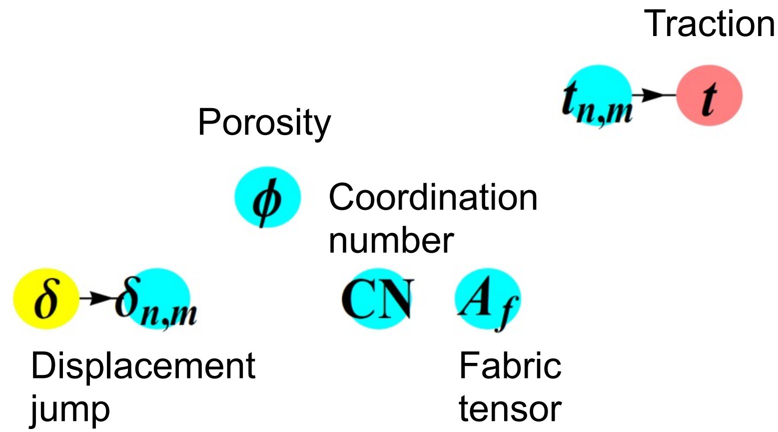
Go Game



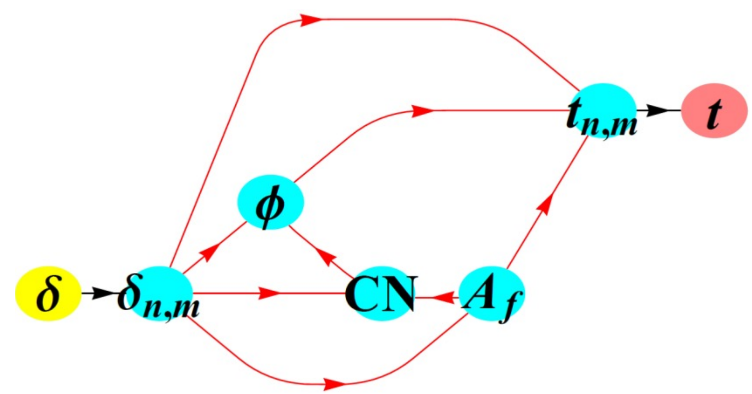
Place pieces to control more territory than your opponent



Meta-modeling Game



Connect edges to generate optimal internal information flow of constitutive models



Applications in Engineering

- Transportation Engineering
- Engineering Mechanics
- Topological optimization
- Predictive modeling
- Architecture materials

And countless more ….