Course Introduction

Machine Learning

Daniele Loiacono

POLITECNICO
MILANO 1863

Information

- **Daniele Loiacono** (Instructor)
  - Contact: daniele.loiacono@polimi.it - +39 02 2399 **3615**
  - Office: DEIB, room 150

- Teaching Assistant: Alberto Maria Metelli

- Exam
  - Written test (closed-book)
    - Questions, exercises, code
    - See examples on WeBeep
    - Late enrollment is NEVER accepted
  - Check the remote exam policy on WeBeep

WeeBeep page

Information (2)

- Weekly schedule
  - Tue, 14.15 – 16.15, T11
  - Thu, 12.15 – 14.15, T21

- There is no streaming but lectures will be recorded
- Check the syllabus for (tentative) info about the topics of each lecture

- Practical classes
  - will cover exam-like exercises and practical examples
  - will present practical examples using **Python** (bring your laptop!)

- Interact
  - Feel free to ask questions
  - Use the forum of the course

References

- You will have access to all the materials used in classroom but slides are not an alternative to textbooks!

- Supervised Learning
  - **Bishop, "Pattern Recognition and Machine Learning", Springer, 2006.**
  - Hastie, Tibshirani, Friedman, "The Elements of Statistical Learning: Data Mining, Inference, and Prediction", Springer, 2009.
  - Mitchell, "Machine Learning", McGraw Hill, 1997.
  - Murphy, "Probabilistic Machine Learning: An Introduction", MIT Press, 2022

- Reinforcement Learning
  - **Sutton and Barto, "Reinforcement Learning: an Introduction", MIT Press, 1998.** New draft available at: http://www.incompleteideas.net/book/the-book-2nd.html

What will you learn?

Goals

- Learn to correctly **model** machine learning problems
  - Can I apply machine learning to this problem?
  - What type of model is best suited for this task?
  - How do I define the inputs and outputs effectively?
- Learn the **principles** and the **main techniques** of ML
  - What are the main differences between supervised, unsupervised, and reinforcement learning?
  - How do domain constraints and data availability influence the choice of model?
  - What are the typical steps involved in an end-to-end machine learning workflow?
  - How models' parameters are optimized and how does it influence the outcome?

Goals (cont.)

- Learn how to **assess** the performances of ML models
  - Which performance metrics are suitable for classification, regression, or clustering tasks?
  - How do I accurately evaluate the performance of a model?
  - How do I detect and manage common issues such as overfitting or underfitting?
- Learn **limitations** of ML techniques and how to **choose** the most appropriate one for your problem
  - Under what circumstances might a simpler model outperform a more complex one?
  - How does data availability affect the choice of the model?
- Provide the basic background to understand the latest developments in this field
  - How do recent trends in machine learning fit into the broader field and build on existing foundations?
  - Which core principles drive these new developments, and what limitations or challenges might they face?

Topics

- Linear Regression
- Linear Classification
- Bias-Variance
- Model Selection
- PAC-Learning and VC dimension
- Kernel Methods
- Support Vector Machines
- Markov Decision Processes
- Dynamic Programming
- RL in finite MDPs
- Multi-armed bandit

What you will <u>not</u> learn?

Prerequisites

- Linear Algebra
  - Operations with matrix and vectors, eigenvalues, eigenvectors, etc.
- Probability and statistics
  - Distributions, confidence intervals, hypothesis test, bayesian statistics
- Optimization (basics)

- Basics understanding of Python (for practical classes)

- Where to find this?
  - Read Chapter 1-2 and Appendix B,C,E of textbook (Bishop, “Pattern Recognition and Machine Learning”)
  - Check the recap lectures on Python, Linear Algebra and Probability
  - Chapter 1 of “Probabilistic Machine Learning: An Introduction”

Other courses

- A course of 5 credits is **not enough** to cover Machine Learning

- Fortunately, there are **other courses** that deal with other machine learning topics not covered in this course:
  - Data Mining
  - Uncertainty in Artificial Intelligence
  - Artificial Neural Networks and Deep Learning
  - Applied Statistics
  - Numerical Analysis for Machine Learning
  - …

What is Machine Learning?

Why and when to apply it?

What is Machine Learning

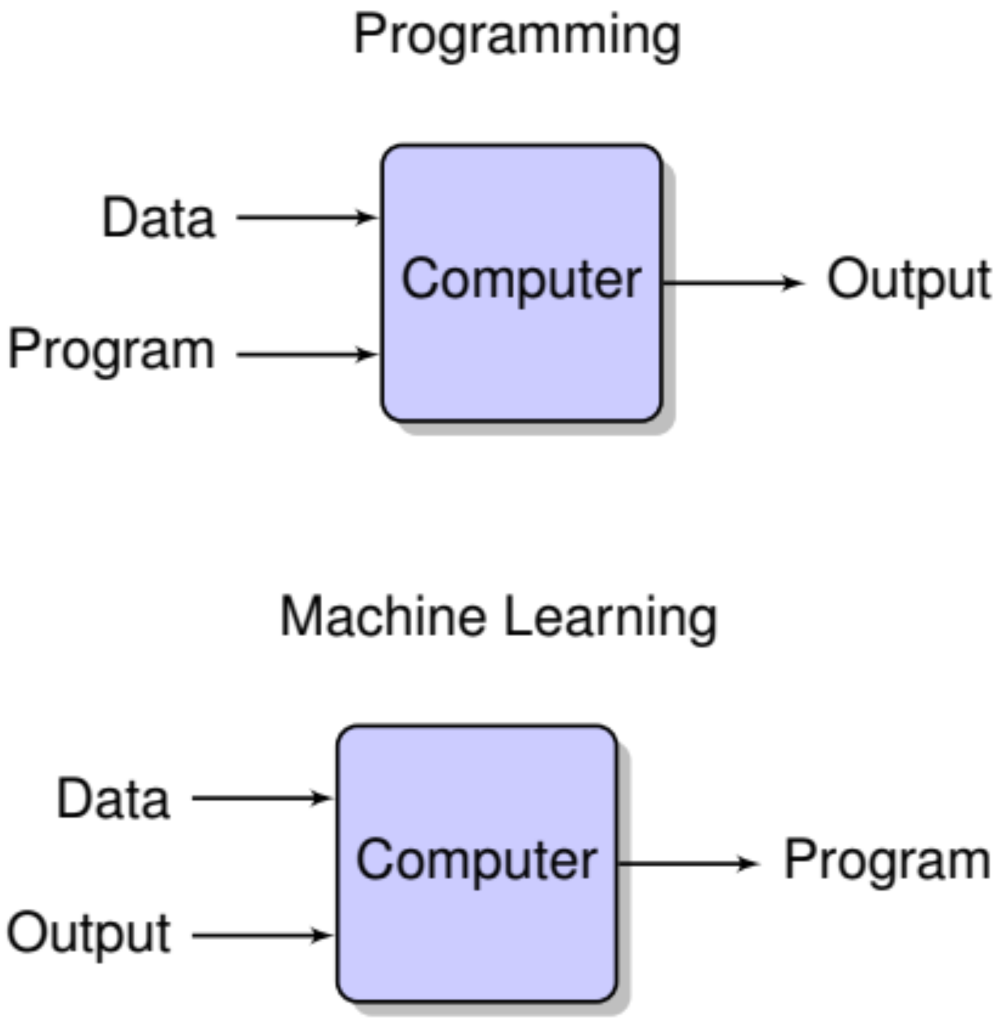
"A computer program is said to learn from **experience E** with respect to some class of **tasks T** and **performance measure P, improves with experience E**" *Mitchell (1997)*

- ML is the sub-field of AI where the **knowledge** comes from:
  - **Experience**
  - **Induction**

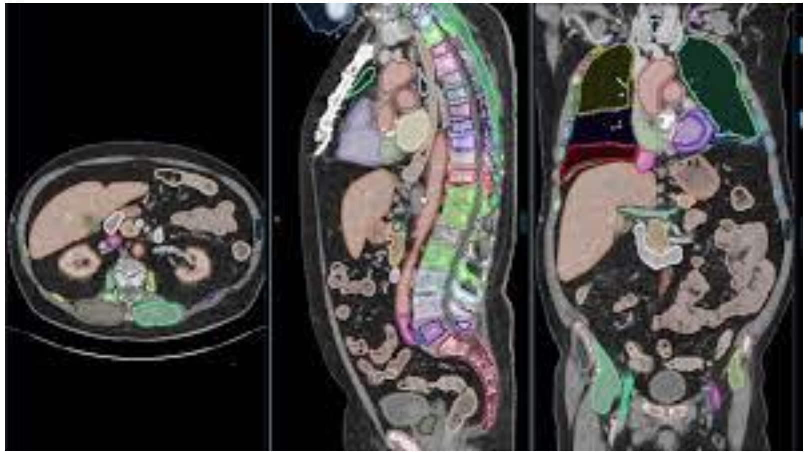
- **However,** Machine learning **is not magic!**
  - You need to know **how it works**
  - You need to understand **how to use it**
  - It can **extract** information from data, **not create** information

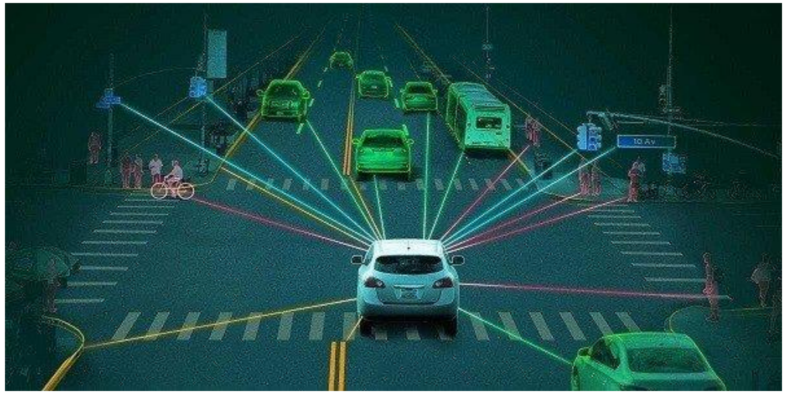
Why Machine Learning?

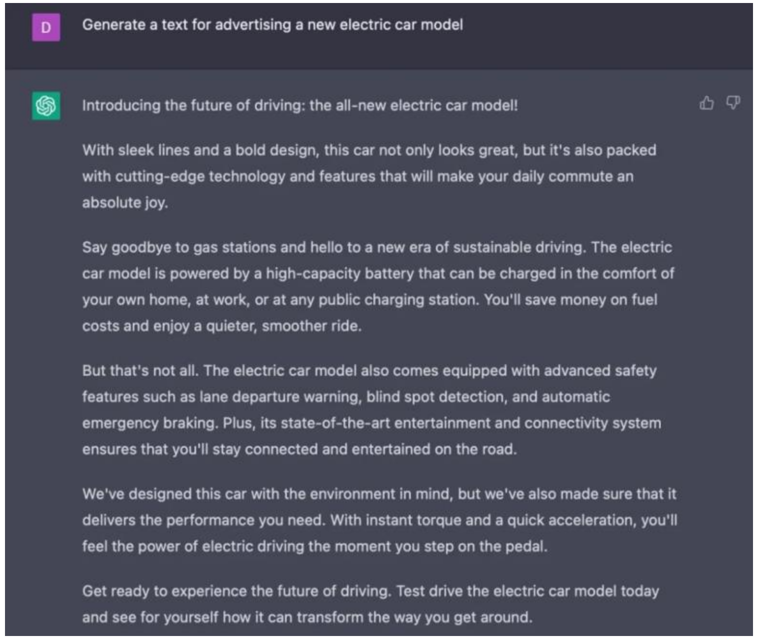
- We need computers to **make informed decisions** on new, unseen data
  - Often it is too difficult to design a set of meaningful rules
  - Machine learning allows to **automatically extract relevant information** from previous data and exploit it on new one
- Getting computers to **program themselves (automating automation)**
  - writing software is the **bottleneck**
  - let the **data** do the work instead

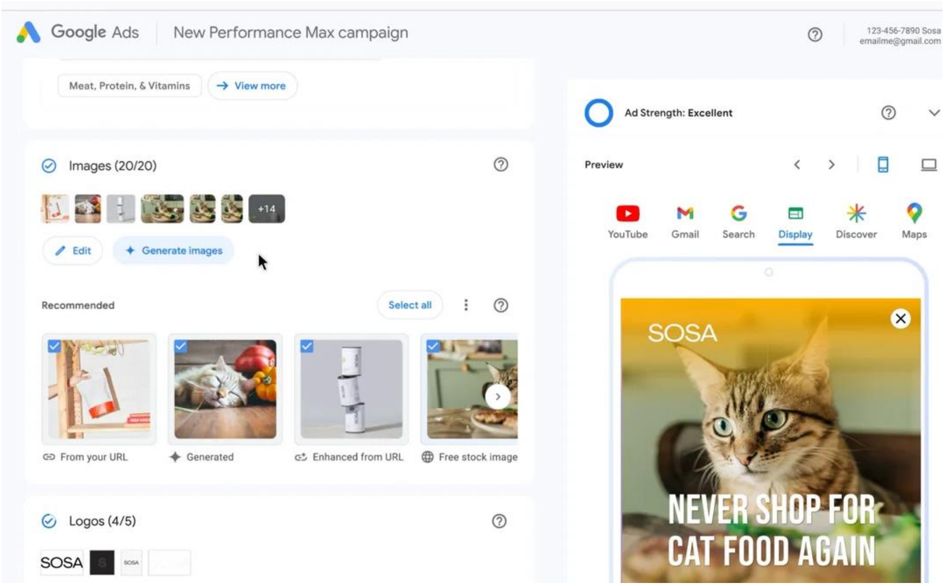


Machine Learning applications

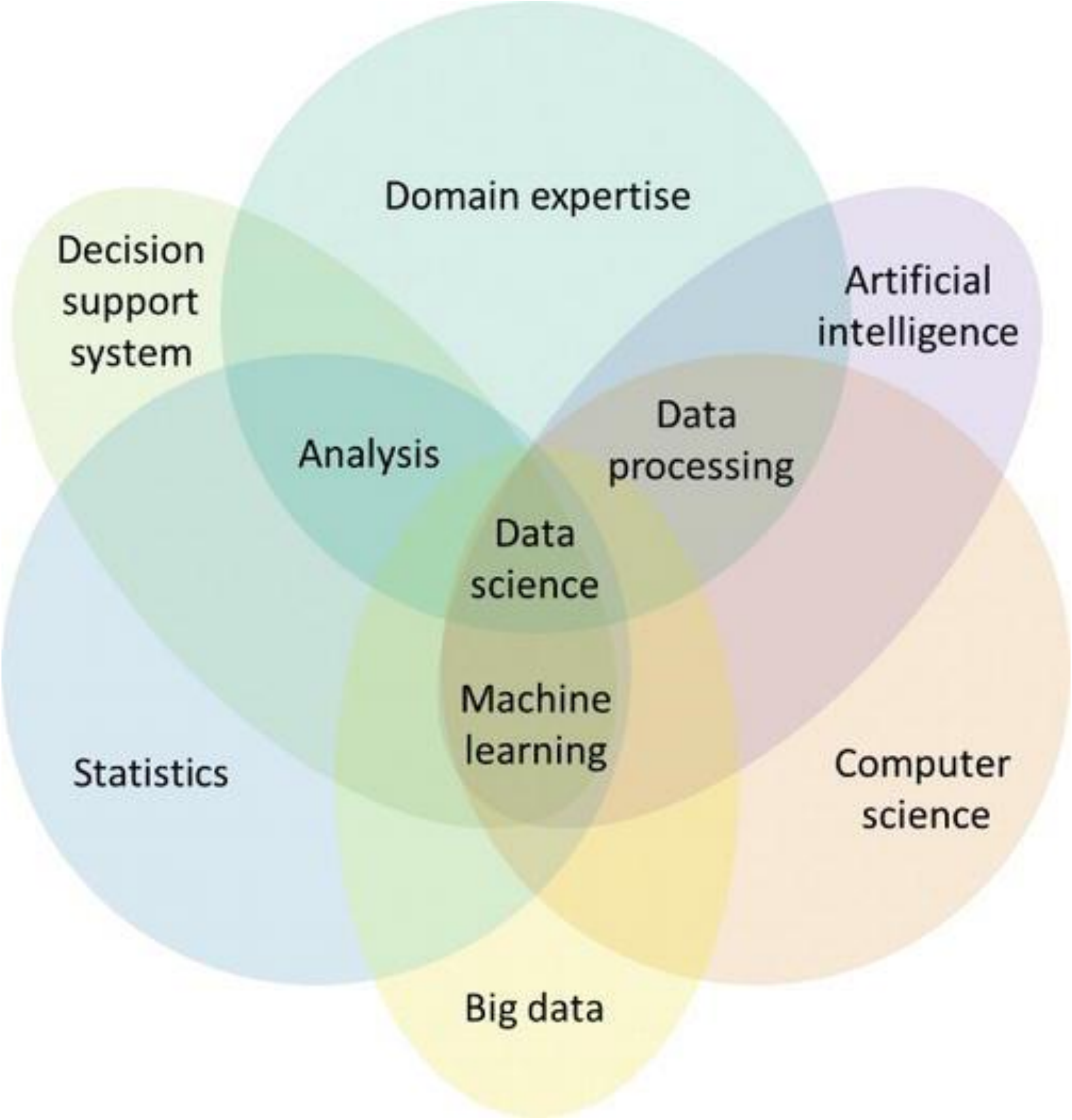




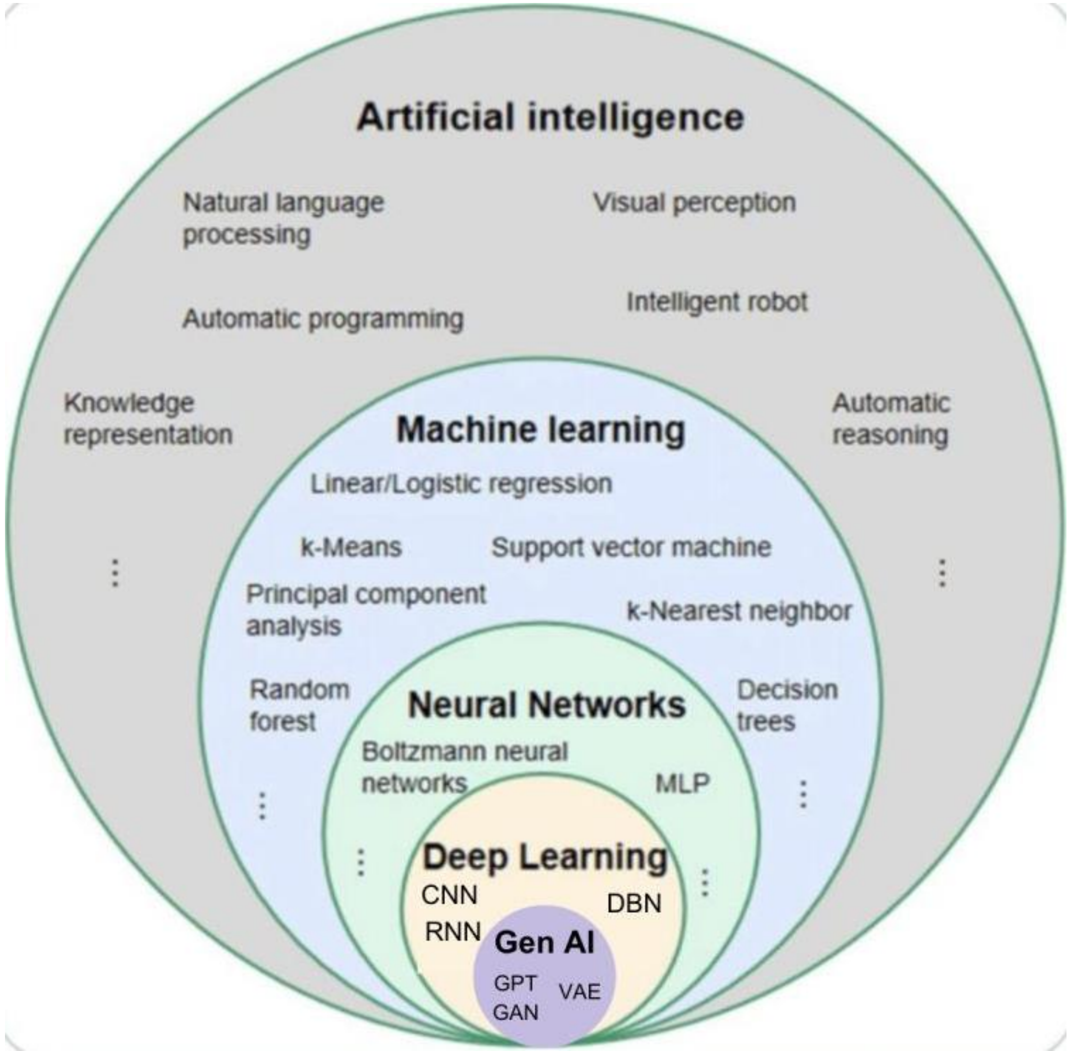




Machine Learning and other fields



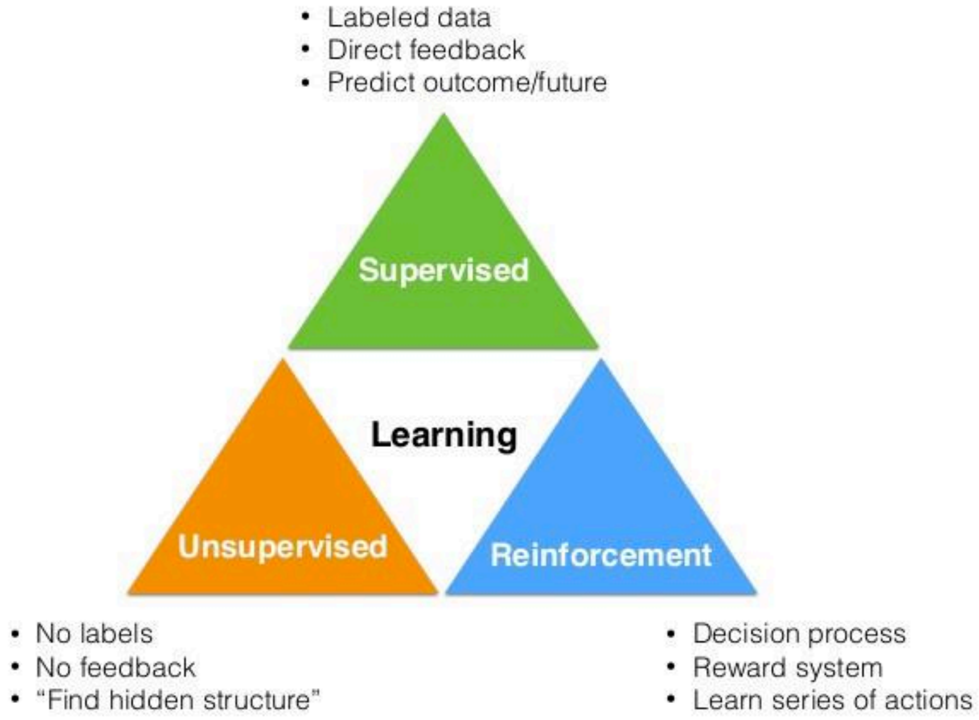
Lee et al., 2018

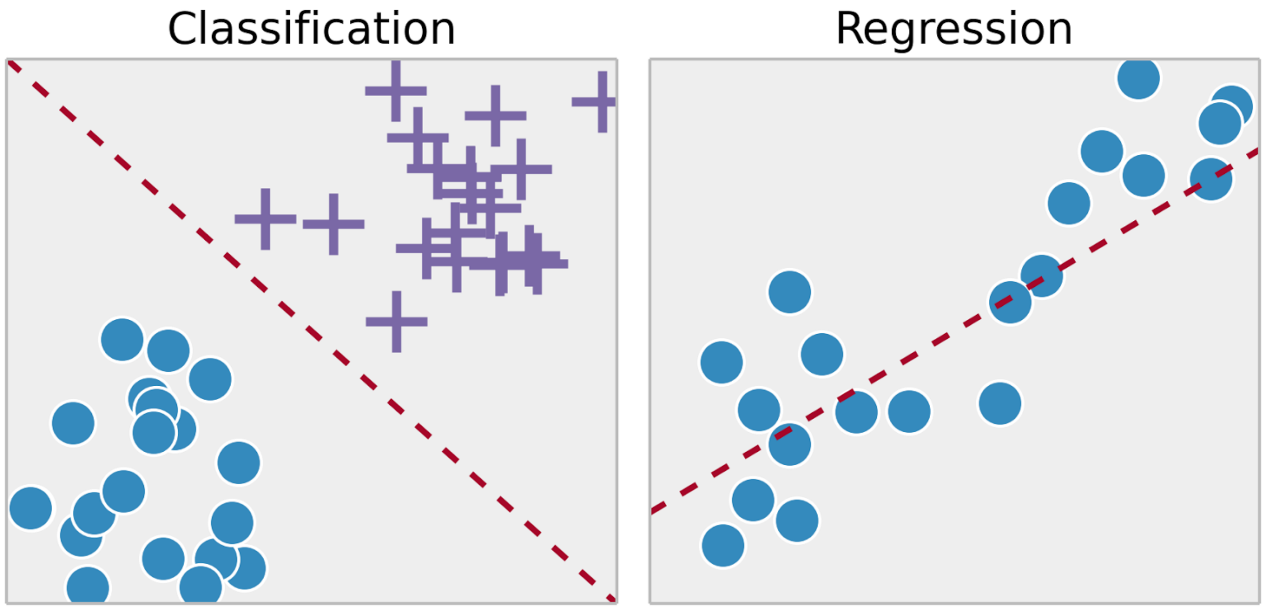


source

Learning Paradigms in ML

- Supervised Learning

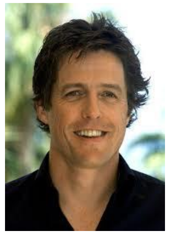




An example of classification

Training Set

0



1

1

0

Testing

0

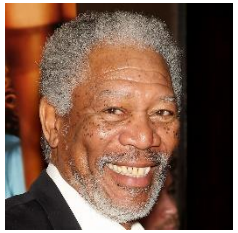
0

1

0

An example of regression

Training Set



78

34

25

Testing

78

81

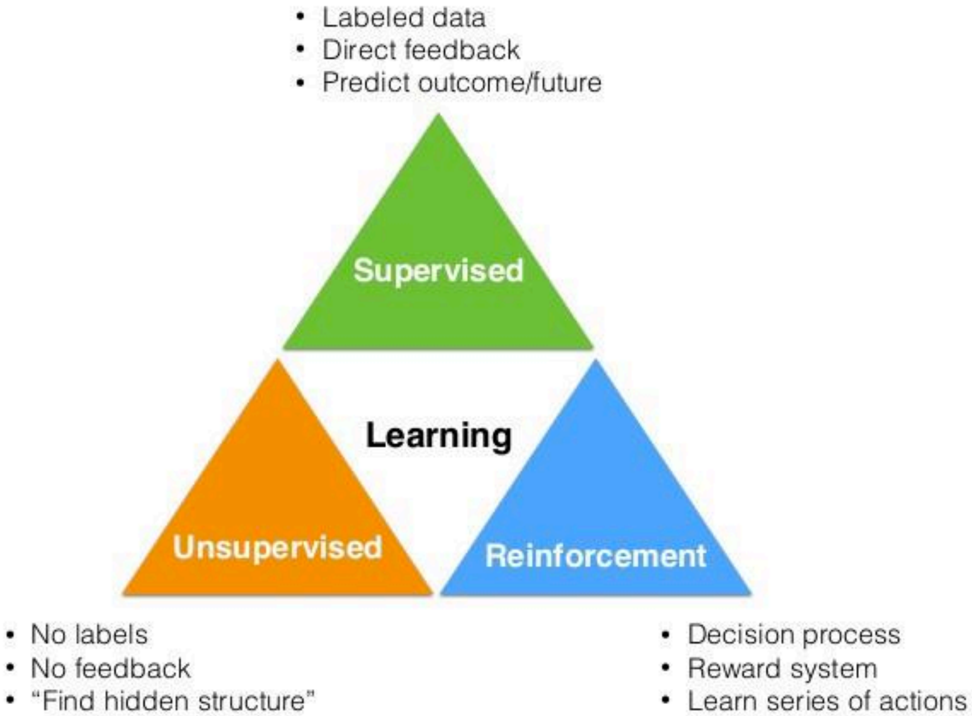
41

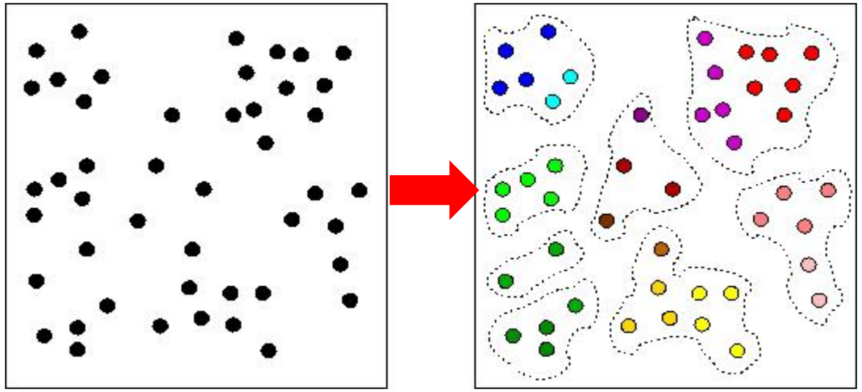
Supervised Learning

- Goal
  - Learn from **data** a **model** that maps **known inputs** to **known outputs**
  - Training set: $\mathcal{D} = \{\langle x, t \rangle\} \Rightarrow t = f(x)$
- Tasks
  - Classification
  - Regression
  - Probability estimation
- Techniques
  - Linear Models
  - Artificial Neural Networks
  - Support Vector Machines
  - Decision trees
  - etc.

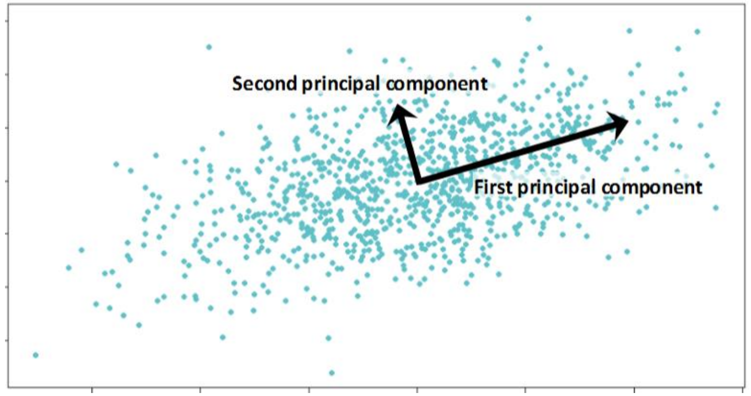
Learning Paradigms in ML

- Unsupervised Learning



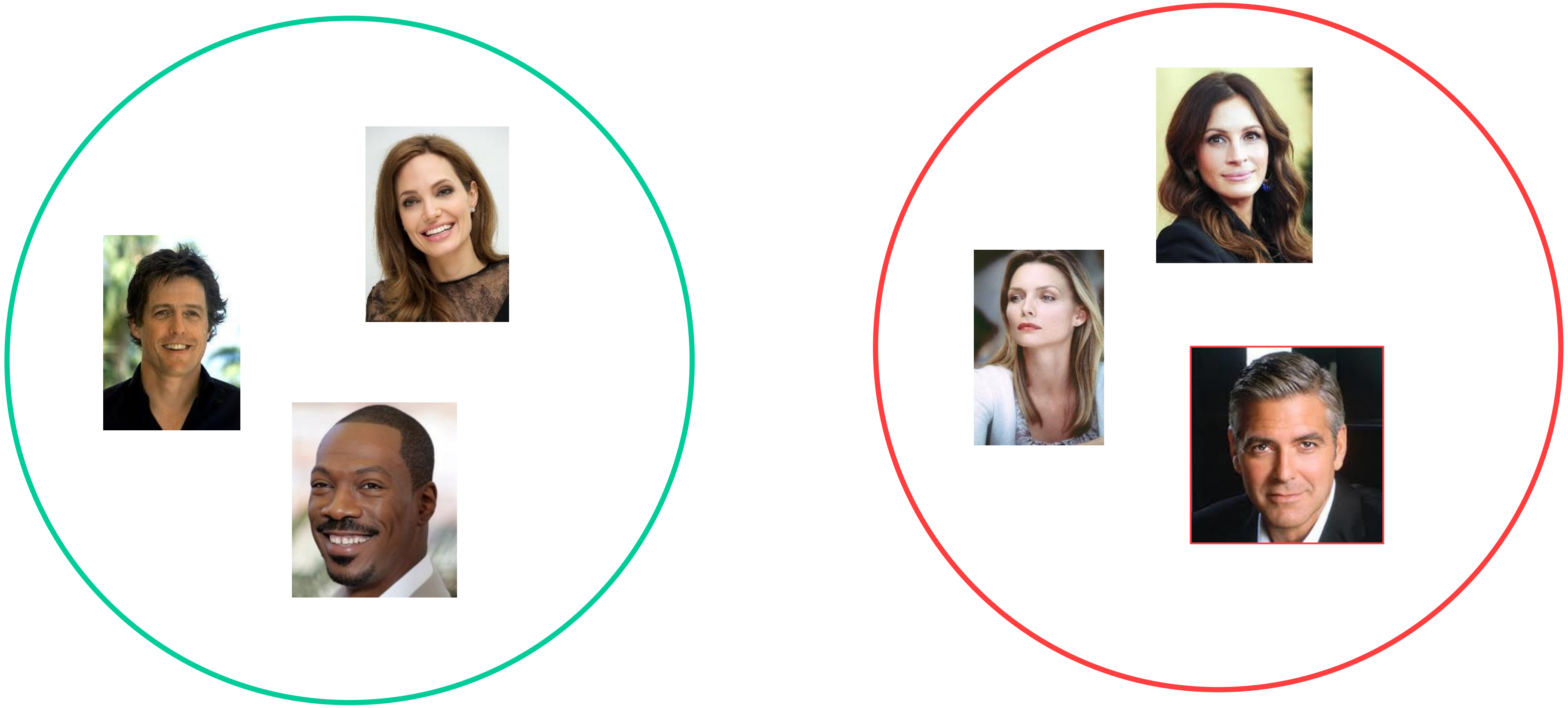


Clustering

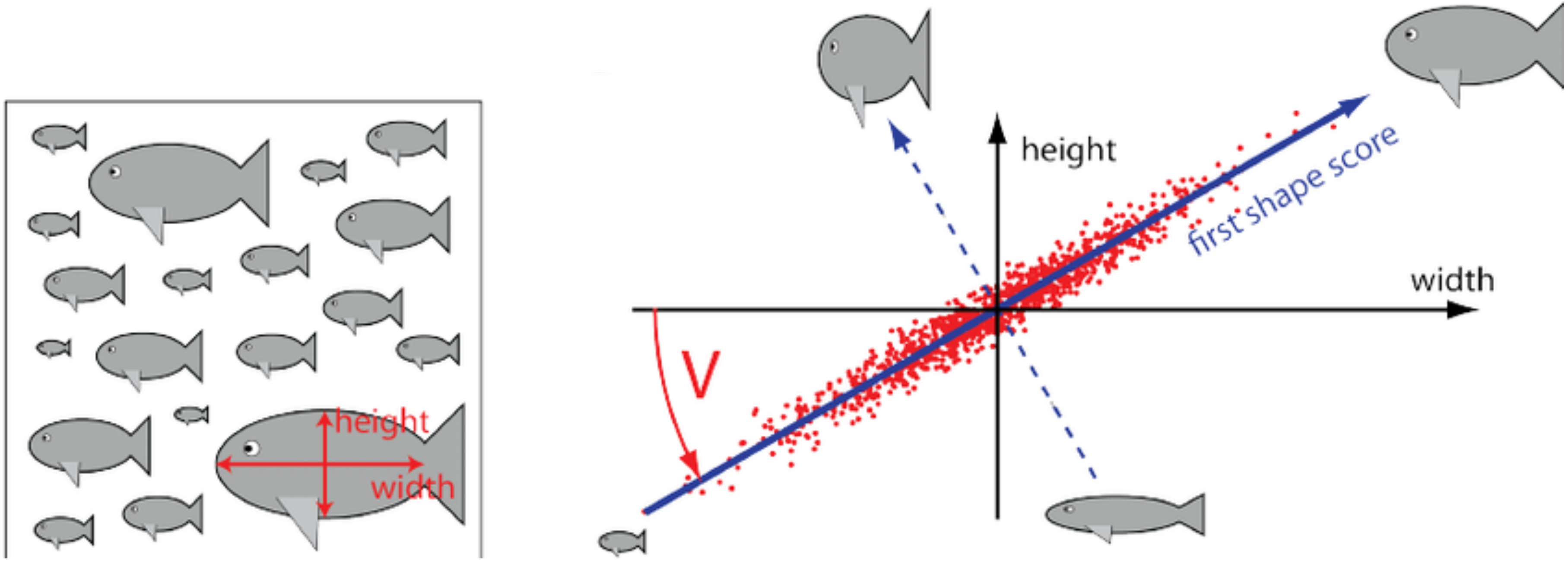


Dimensionality Reduction

An example of clustering



An example of dimensionality reduction

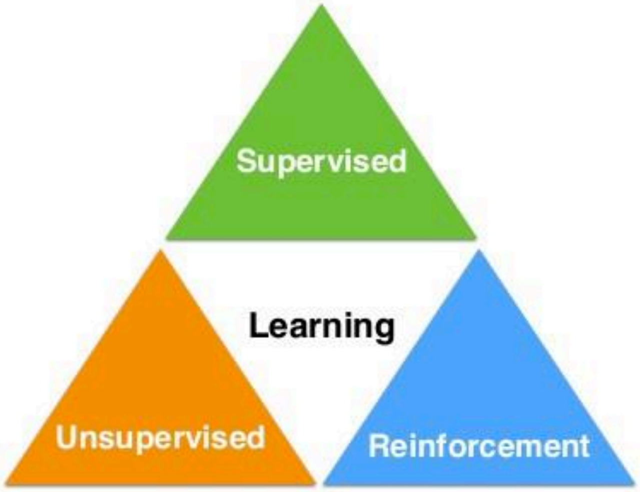


Unsupervised Learning

- Goal
  - Learn **previously unknown patterns** and **efficient data representation**
  - Training set: $\mathcal{D} = \{x\} \Rightarrow f(x)$
- Tasks
  - Dimensionality Reduction
  - Clustering
- Techniques
  - K-means
  - Self-organizing maps
  - Principal Component Analysis
  - etc.

Learning Paradigms in ML

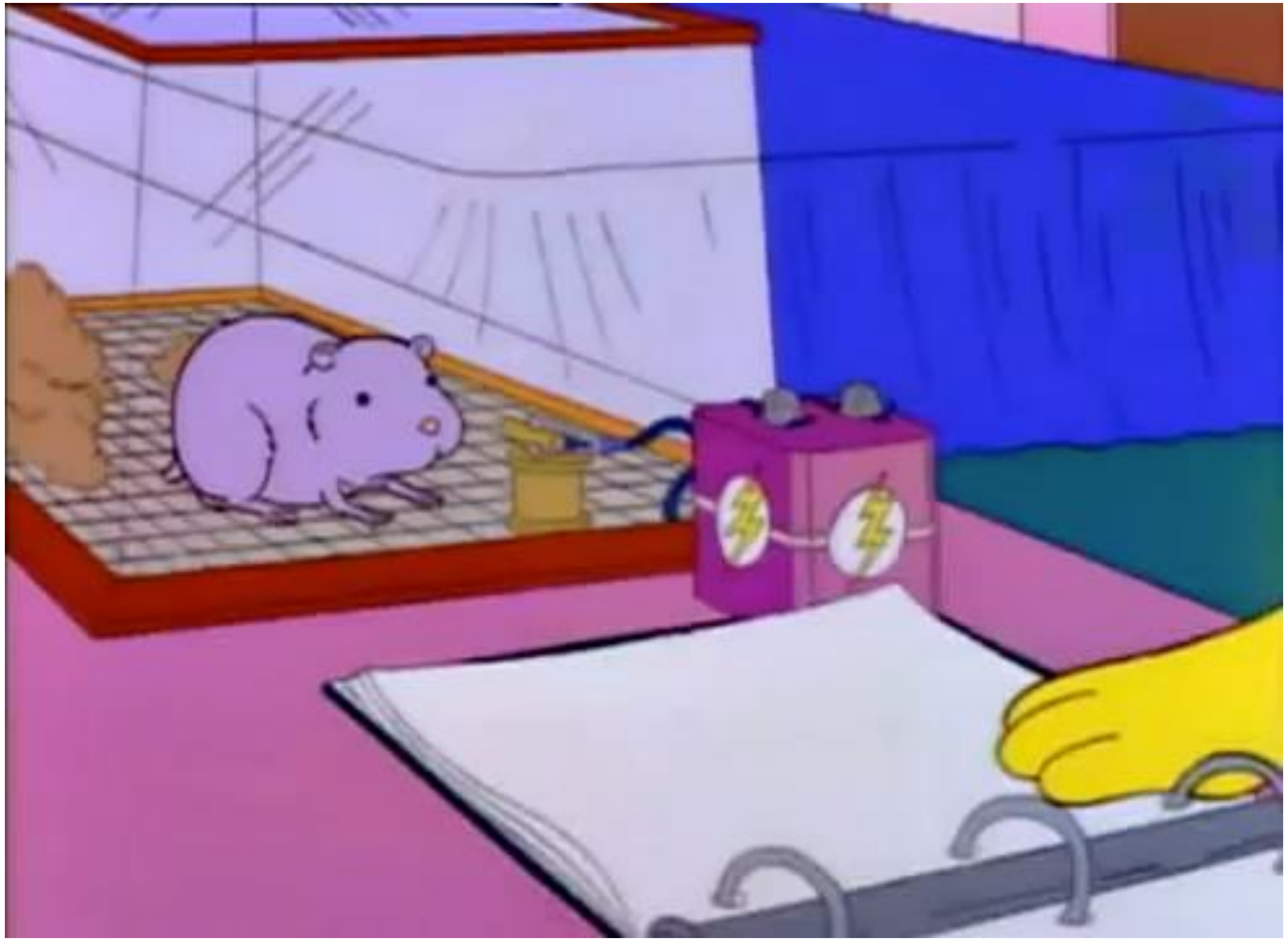
- Labeled data
- Direct feedback
- Predict outcome/future



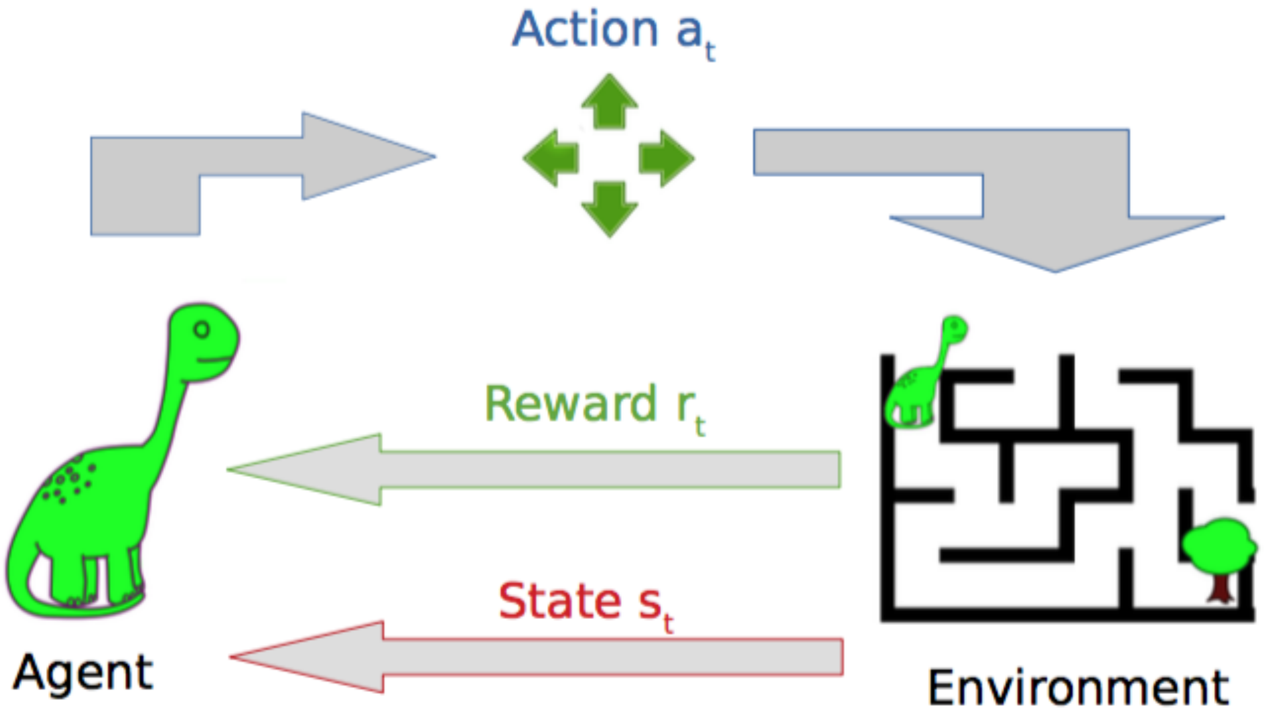
- No labels
- No feedback
- “Find hidden structure”

- Decision process
- Reward system
- Learn series of actions

- Reinforcement Learning



An example of reinforcement learning



Reinforcement Learning

- Goal
  - Learning the **optimal policy**
  - Training set: $\mathcal{D} = \{\langle x, u, x', r \rangle\} \Rightarrow \pi^*(x) = arg\,\max_u \{Q^*(x, u)\}$

to estimate

- Problems
  - Markov Decision Process (MDP)
  - Partially Observable MDP (POMDP)
  - Stochastic Games (SG)
- Techniques
  - Q-learning
  - SARSA
  - Fitted Q-iteration
  - etc.