Introduction to Machine Learning

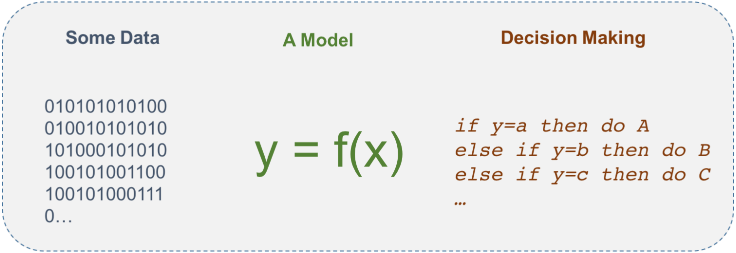
Dr. Víctor Uc Cetina

Universidad Autónoma de Yucatán

Content

What is Machine Learning?

- Machine learning (ML) refers to algorithms used to extract patterns from data and learn a mathematical model that could be used by a computer program to make intelligent decisions.



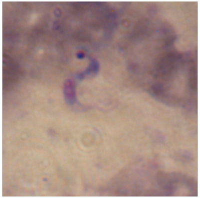
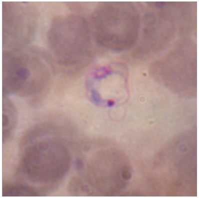
Supervised learning - Regression

- Given the characteristics and prices of several houses, develop a software for predicting the price of new houses.

Living area	Bedrooms	Price
560	2	37.000
1012	3	79.000
893	3	76.000
2196	4	130.000
⋮	⋮	⋮
936	3	72.000

Supervised learning - Classification

- Given a set of digital images of blood samples containing Chagas parasites, decide if a digital image of a new blood sample contains at least a Chagas parasite.

Unsupervised learning - Clustering

- Given a set of text phrases written in two different languages, decide if a new text phrase belongs to one of the languages existing in your set of phrases.

hola a todo el mundo - no me gusta decir adiós - hello people – adoro la comida - our world is wonderful - el planeta agua - ciencia ficción es ciencia - el algoritmo más rápido - la mesa es redonda - the door is black -
my chair is broken - el plato está limpio, esa escalera está muy inclinada - my mouse is wireless - an electronic book - a wide road is better, we were at home - …

Reinforcement learning - Control

- Given a history of the commands used to control a drone, decide which is the best command to perform in order to avoid a collision with the ground.

Three major kinds of ML problems

Based on the type of data available and the decisions needed, we can talk about three general kinds of machine learning algorithms:

Three major kinds of ML problems

Based on the type of data available and the decisions needed, we can talk about three general kinds of machine learning algorithms:

- Supervised learning: **inputs and outputs**

Three major kinds of ML problems

Based on the type of data available and the decisions needed, we can talk about three general kinds of machine learning algorithms:

- Supervised learning: **inputs and outputs**
- Unsupervised learning: **only inputs**

Three major kinds of ML problems

Based on the type of data available and the decisions needed, we can talk about three general kinds of machine learning algorithms:

- Supervised learning: **inputs and outputs**
- Unsupervised learning: **only inputs**
- Reinforcement learning: **states, actions, and rewards**

Fields of science getting involved in ML

Current machine learning developments come from fields such as:

Fields of science getting involved in ML

Current machine learning developments come from fields such as:

- Computer Science

Fields of science getting involved in ML

Current machine learning developments come from fields such as:

- Computer Science
- Artificial Intelligence

Fields of science getting involved in ML

Current machine learning developments come from fields such as:

- Computer Science
- Artificial Intelligence
- Bioinformatics

Fields of science getting involved in ML

Current machine learning developments come from fields such as:

- Computer Science
- Artificial Intelligence
- Bioinformatics
- Neuroscience

Fields of science getting involved in ML

Current machine learning developments come from fields such as:

- Computer Science
- Artificial Intelligence
- Bioinformatics
- Neuroscience
- Psychology

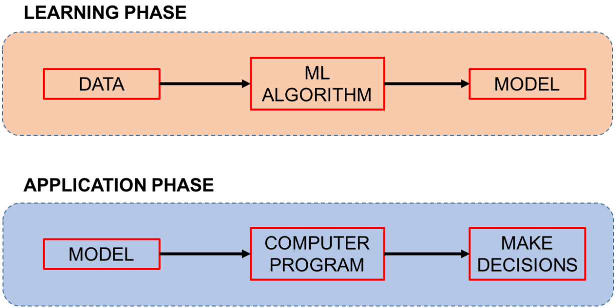
Fields of science getting involved in ML

Current machine learning developments come from fields such as:

- Computer Science
- Artificial Intelligence
- Bioinformatics
- Neuroscience
- Psychology
- Robotics

Practical aspects of machine learning

- Using machine learning algorithms usually involves at least two phases:



Learning machine learning

Recipe for an optimal learning of machine learning:

1. Understanding the kind of machine learning problem (SL, UL, RL)

Learning machine learning

Recipe for an optimal learning of machine learning:

1. Understanding the kind of machine learning problem (SL, UL, RL)
2. Understanding the mathematics (basic statistics, probability, calculus, linear algebra)

Learning machine learning

Recipe for an optimal learning of machine learning:

1. Understanding the kind of machine learning problem (SL, UL, RL)
2. Understanding the mathematics (basic statistics, probability, calculus, linear algebra)
3. Understanding the algorithms (data structures, complexity)

Learning machine learning

Recipe for an optimal learning of machine learning:

1. Understanding the kind of machine learning problem (SL, UL, RL)
2. Understanding the mathematics (basic statistics, probability, calculus, linear algebra)
3. Understanding the algorithms (data structures, complexity)
4. Coding (Matlab/Octave/Python)

Learning machine learning

Recipe for an optimal learning of machine learning:

1. Understanding the kind of machine learning problem (SL, UL, RL)
2. Understanding the mathematics (basic statistics, probability, calculus, linear algebra)
3. Understanding the algorithms (data structures, complexity)
4. Coding (Matlab/Octave/Python)
5. Experimenting (running programs and plotting graphs)

Part I - Supervised Learning

1. Linear regression

Part I - Supervised Learning

1. Linear regression
2. Classification

Part I - Supervised Learning

1. Linear regression
2. Classification
3. Generative learning models

Part I - Supervised Learning

1. Linear regression
2. Classification
3. Generative learning models
4. Support vector machines

Part I - Supervised Learning

1. Linear regression
2. Classification
3. Generative learning models
4. Support vector machines
5. AdaBoost

Part I - Supervised Learning

1. Linear regression
2. Classification
3. Generative learning models
4. Support vector machines
5. AdaBoost
6. Learning theory

Part I - Supervised Learning

1. Linear regression
2. Classification
3. Generative learning models
4. Support vector machines
5. AdaBoost
6. Learning theory
7. Regularization and model selection

Part I - Supervised Learning

1. Linear regression
2. Classification
3. Generative learning models
4. Support vector machines
5. AdaBoost
6. Learning theory
7. Regularization and model selection
8. Neural networks

Part II - Unsupervised Learning

1. Expectation-Maximization (EM) algorithm

Part II - Unsupervised Learning

1. Expectation-Maximization (EM) algorithm
2. Principal component analysis

Part II - Unsupervised Learning

1. Expectation-Maximization (EM) algorithm
2. Principal component analysis
3. Self organizing maps

Part II - Unsupervised Learning

1. Expectation-Maximization (EM) algorithm
2. Principal component analysis
3. Self organizing maps
4. Spectral clustering

Part III - Reinforcement Learning

1. Markov decision processes

Part III - Reinforcement Learning

1. Markov decision processes
2. Dynamic programming

Part III - Reinforcement Learning

1. Markov decision processes
2. Dynamic programming
3. Monte Carlo methods

Part III - Reinforcement Learning

1. Markov decision processes
2. Dynamic programming
3. Monte Carlo methods
4. Temporal difference learning

Sources of Information

- **Presentations** (Videos and slides)

Sources of Information

- **Presentations** (Videos and slides)
- **Lecture notes**

Sources of Information

- **Presentations** (Videos and slides)
- **Lecture notes**
- **Exercises**

Sources of Information

- **Presentations** (Videos and slides)
- **Lecture notes**
- **Exercises**

- **Pattern Recognition and Machine Learning**. Christopher M. Bishop. Springer. 2006.

Sources of Information

- **Presentations** (Videos and slides)
- **Lecture notes**
- **Exercises**

- **Pattern Recognition and Machine Learning**. Christopher M. Bishop. Springer. 2006.
- **Reinforcement Learning: An Introduction (second edition)**. Richard Sutton and Andrew Barto. The MIT Press. 2018.

Sources of Information

- **Presentations** (Videos and slides)
- **Lecture notes**
- **Exercises**

- **Pattern Recognition and Machine Learning**. Christopher M. Bishop. Springer. 2006.
- **Reinforcement Learning: An Introduction (second edition)**. Richard Sutton and Andrew Barto. The MIT Press. 2018.

- **Foundations of Machine Learning**. Mehryar Mohri, Afshin Rostamizadeh and Ameet Talwalkar. The MIT Press. 2012.

Sources of Information

- **Presentations** (Videos and slides)
- **Lecture notes**
- **Exercises**

- **Pattern Recognition and Machine Learning**. Christopher M. Bishop. Springer. 2006.
- **Reinforcement Learning: An Introduction (second edition)**. Richard Sutton and Andrew Barto. The MIT Press. 2018.

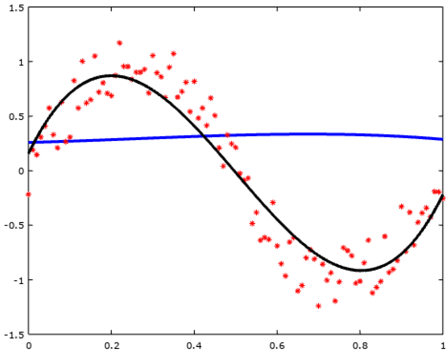
- **Foundations of Machine Learning**. Mehryar Mohri, Afshin Rostamizadeh and Ameet Talwalkar. The MIT Press. 2012.
- **Understanding Machine Learning: from Theory to Algorithms**. Shai Shalev-Shwartz and Shai Ben-David. Cambridge University Press. 2014.

Regression

- In the regression problem we have $R : X \to Y$, $X \subseteq \mathbb{R}^d$, $Y \subseteq \mathbb{R}$. (Continuous) Example: $y = w_o + w_1 x + w_2 x^2 + w_3 x^3$

Regression

- In the regression problem we have
  $R : X \to Y$,
  $X \subseteq \mathbb{R}^d$,
  $Y \subseteq \mathbb{R}$. (Continuous)
  Example:
  $y = w_o + w_1 x + w_2 x^2 + w_3 x^3$

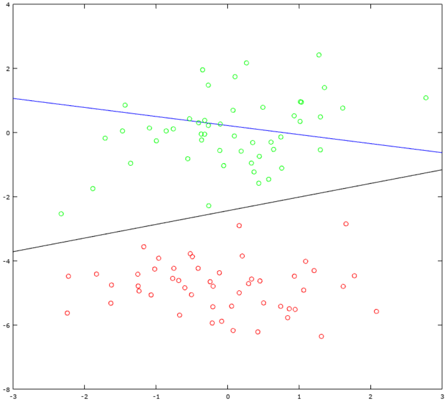


Classification

- In the classification problem we have $C : X \to Y$, $X \subseteq \mathbb{R}^d$, $Y = \{y_1, y_2, \ldots, y_n\}$. (Discrete) Example: $y = \frac{1}{1+e^{-(w_0+w_1x_1+w_2x_2)}}$

Classification

- In the classification problem we have $C : X \to Y$, $X \subseteq \mathbb{R}^d$, $Y = \{y_1, y_2, \ldots, y_n\}$. (Discrete)
  Example:
  $y = \frac{1}{1+e^{-(w_0+w_1x_1+w_2x_2)}}$



Clustering

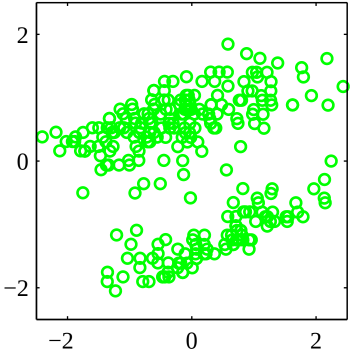
The unsupervised learning problem is similar to the classification one, but the data is not labeled In this case we do not know the category of each example.

- In the clustering problem we have $C : X \to T$, $X \subseteq \mathbb{R}^d$, $T = \{t_1, t_2, \ldots, t_n\}$, with unknown n. Example: $t = \arg\min_{t_i} \mathrm{distance}(t_i, x)$

Clustering

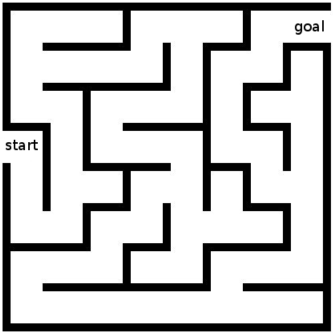
The unsupervised learning problem is similar to the classification one, but the data is not labeled In this case we do not know the category of each example.

- In the clustering problem we have
  $C : X \to T$,
  $X \subseteq \mathbb{R}^d$,
  $T = \{t_1, t_2, \ldots, t_n\}$,
  with unknown n. Example:
  $t = \arg\min_{t_i} \mathrm{distance}(t_i, x)$



Reinforcement Learning

In reinforcement learning an agent (i.e. a robot) must learn to perform a task: a sequence of actions to go from one **start** state to one **goal** state.

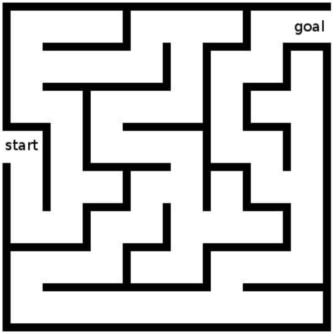


Markov decision process

$\text{MDP} = (S, A, T, R)$

Reinforcement Learning

In reinforcement learning an agent (i.e. a robot) must learn to perform a task: a sequence of actions to go from one **start** state to one **goal** state.

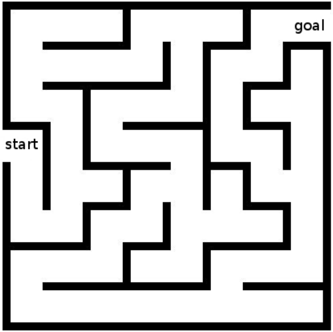


Markov decision process
$\text{MDP} = (S, A, T, R)$

- Set of states
  $S = \{s_1, s_2, \ldots\}$

Reinforcement Learning

In reinforcement learning an agent (i.e. a robot) must learn to perform a task: a sequence of actions to go from one **start** state to one **goal** state.

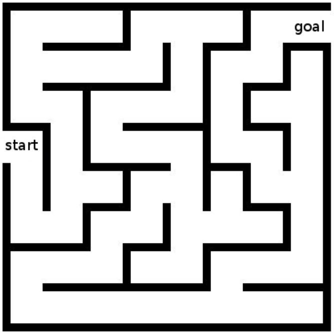


Markov decision process
$\text{MDP} = (S, A, T, R)$

- Set of states
  $S = \{s_1, s_2, \ldots\}$
- Set of actions
  $A = \{a_1, a_2, \ldots\}$

Reinforcement Learning

In reinforcement learning an agent (i.e. a robot) must learn to perform a task: a sequence of actions to go from one **start** state to one **goal** state.

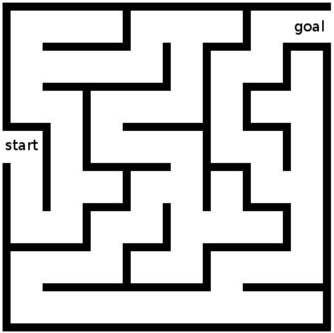


Markov decision process
$\text{MDP} = (S, A, T, R)$

- Set of states
  $S = \{s_1, s_2, \ldots\}$
- Set of actions
  $A = \{a_1, a_2, \ldots\}$
- Transition function (dynamics of the environment)
  $T : S \times A \rightarrow S$

Reinforcement Learning

In reinforcement learning an agent (i.e. a robot) must learn to perform a task: a sequence of actions to go from one **start** state to one **goal** state.

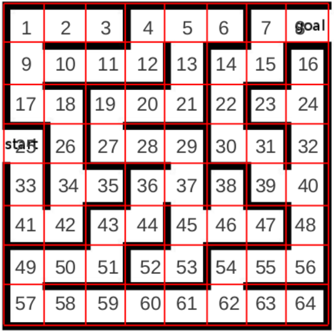


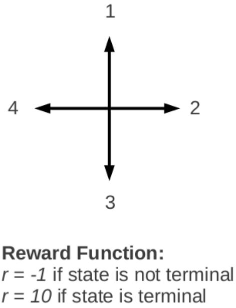
Markov decision process
$\text{MDP} = (S, A, T, R)$

- Set of states
  $S = \{s_1, s_2, \ldots\}$
- Set of actions
  $A = \{a_1, a_2, \ldots\}$
- Transition function (dynamics of the environment)
  $T : S \times A \to S$
- Reward function
  $R : S \to \mathbb{R}$

Possible RL solution to the maze problem

We discretize states and actions: 64 states and 4 actions.





The agent interacts with the environment several times and the RL algorithm estimates a control function $a_{t+1} = \pi(s_t)$.

Function *Q(s,a)*

S	A	Q(s,a)
1	1	0
1	2	0
1	3	0
1	4	0
2	1	0
2	2	0
2	3	0
...	...	...
64	4	0

Function *Q(s,a)*

S	A	Q(s,a)
1	1	0
1	2	0
1	3	0
1	4	0
2	1	0
2	2	0
2	3	0
...	...	...
64	4	0

S	A	Q(s,a)
1	1	-20
1	2	-23
1	3	-5
1	4	-22
2	1	-20
2	2	-5
2	3	-22
...	...	...
64	4	-24

Q-learning Algorithm

Initialize $Q(s, a)$ arbitrarily

Repeat (for each episode):

  Initialize s

  Repeat (for each step of episode):

    Choose a from s using policy derived from Q (ε-*greedy*)

    Take action a, observe r, s'

    $Q(s, a) \leftarrow Q(s, a) + \alpha\big[r + \gamma \max_{a'} Q(s', a') - Q(s, a)\big]$

    $s \leftarrow s'$

  Until s is terminal

Thank you!

Dr. Víctor Uc Cetina
victoruccetina@gmail.com