Machine Learning: A General View

Dr. Víctor Uc Cetina

Facultad de Matemáticas
Universidad Autónoma de Yucatán

cetina@informatik.uni-hamburg.de
https://sites.google.com/view/victoruccetina

Content

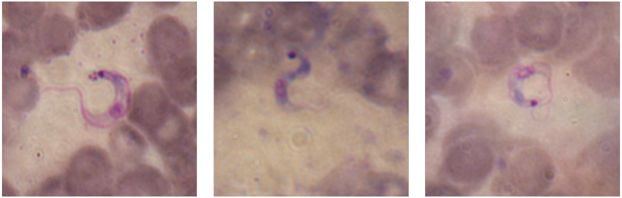
What is Machine Learning?

- 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.

Some Data	A Model	Decision Making
010101010100 010010101010 101000101010 100101001100 100101000111 0...	$y = f(x)$	*if y=a then do A* *else if y=b then do B* *else if y=c then do C* ...

What is Machine Learning?

- Example 1. 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.



What is Machine Learning?

- Example 2. 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.

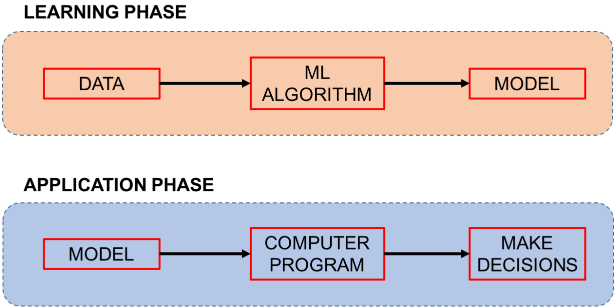
hola a todo el mundo - no me gusta decir adiós - hello people – adoro la comida - our world is wonderful - el planeta agua - ciencia ficción es ciencia - el algoritmo más rápido - la mesa es redonda - the door is black -
my chair is broken - el plato está limpio, esa escalera está muy inclinada - my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

What is Machine Learning?

- Example 3. 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.

What is Machine Learning?

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



What is Machine Learning?

- Based on the type of data available and the decisions needed, we can talk about three general kinds of machine learning algorithms:
  - Supervised Learning (SL)
  - Unsupervised Learning (UL)
  - Reinforcement Learning (RL)

What is Machine Learning?

- People involved in ML research and development comes from fields such as: Computer Science, Artificial Intelligence, Statistics, Neuroscience, Psychology, Robotics, Bioinformatics, etc.

About Learning Machine Learning

Main Human Learning Loop for ML:

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

About Learning Machine Learning

Main Human Learning Loop for ML:

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

About Learning Machine Learning

Main Human Learning Loop for ML:

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)

About Learning Machine Learning

Main Human Learning Loop for ML:

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)

About Learning Machine Learning

Main Human Learning Loop for ML:

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)
5. Experimenting (running programs and plotting graphs)

Important to know

- Course materials will be available to download from the Stine website

Important to know

- Course materials will be available to download from the Stine website
- Feedback of exercises are given in person during the exercise sessions or in my office.

Important to know

- Course materials will be available to download from the Stine website
- Feedback of exercises are given in person during the exercise sessions or in my office.
- You need to complete and deliver at least 5 exercises to have the right to write the final exam.

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 II - Unsupervised Learning

1. Expectation-Maximization (EM) algorithm

Part II - Unsupervised Learning

1. Expectation-Maximization (EM) algorithm
2. Factor analysis

Part II - Unsupervised Learning

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

Part III - Reinforcement Learning

1. Dynamic programming

Part III - Reinforcement Learning

1. Dynamic programming
2. Montecarlo methods

Part III - Reinforcement Learning

1. Dynamic programming
2. Montecarlo methods
3. Temporal difference learning

Part III - Reinforcement Learning

1. Dynamic programming
2. Montecarlo methods
3. Temporal difference learning
4. Generalization and function approximation

Sources of Information

- Presentations

Sources of Information

- Presentations
- ML notes written by Prof. Andrew Ng

Sources of Information

- Presentations
- ML notes written by Prof. Andrew Ng
- **Pattern Recognition and Machine Learning**. Christopher M. Bishop. Springer. 2006.

Sources of Information

- Presentations
- ML notes written by Prof. Andrew Ng
- **Pattern Recognition and Machine Learning**. Christopher M. Bishop. Springer. 2006.
- **Foundations of Machine Learning**. Mehryar Mohri, Afshin Rostamizadeh and Ameet Talwalkar. The MIT Press. 2012.

Sources of Information

- Presentations
- ML notes written by Prof. Andrew Ng
- **Pattern Recognition and Machine Learning**. Christopher M. Bishop. Springer. 2006.
- **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.

Sources of Information

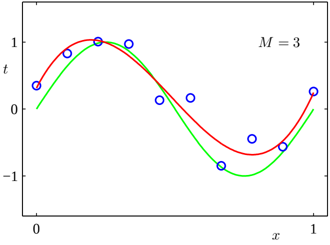
- Presentations
- ML notes written by Prof. Andrew Ng
- **Pattern Recognition and Machine Learning**. Christopher M. Bishop. Springer. 2006.
- **Foundations of Machine Learning**. Mehryar Mohri, Afshin Rostamizadeh and Ameet Talwalkar. The MIT Press. 2012.
- **Understanding Machine Learning: from Theory to Algorithms**. Shai Shalev-Shwartz and Shai Ben-David. Cambridge University Press. 2014.
- **Reinforcement Learning: An Introduction (second edition)**. Richard Sutton and Andrew Barto. The MIT Press. 2018.

Regression

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

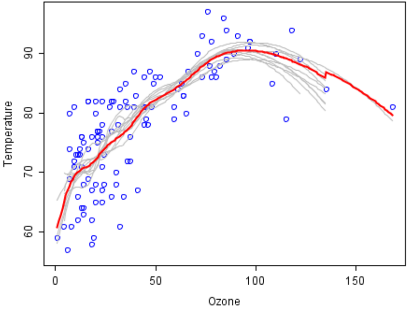
Regression

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



Regression

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

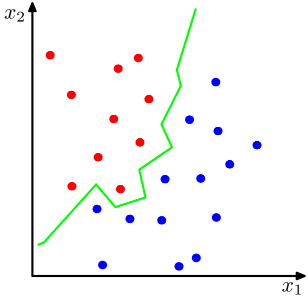


Classification

- 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

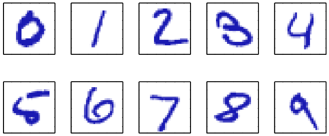
- In the classification problem we have $C : X \to Y$, $X \subseteq \mathbb{R}^d$, $Y = \{y_1, y_2, \ldots, y_n\}$. (Discrete)
  Example:
  $y = \frac{1}{1+e^{-(w_0+w_1x_1+w_2x_2)}}$



(b)

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

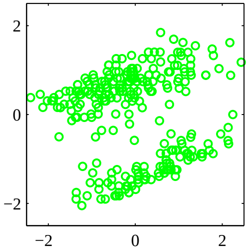
The unsupervised learning problem is similar to the classification one, but the data is not labeled this means that we do not know the category of each example.

- In the clustering problem we have $C : X \to T$, $X \subseteq \mathbb{R}^d$, $T = \{t_1, t_2, \dots, t_n\}$, 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 this means that 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 the agent must learn to perform a task. It means going from one start state s_{start} to one final state s_{final}.

Reinforcement Learning

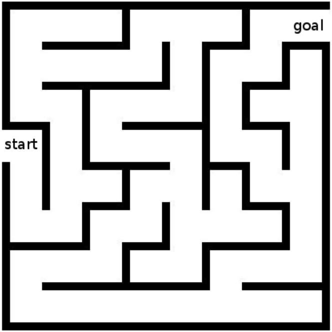
- In reinforcement learning the agent must learn to perform a task. It means going from one start state s_{start} to one final state s_{final}.
- The execution of the task can be seen as a sequence of actions $a_1, a_2, \ldots, a_n$.

Reinforcement Learning

- In reinforcement learning the agent must learn to perform a task. It means going from one start state s_{start} to one final state s_{final}.
- The execution of the task can be seen as a sequence of actions $a_1, a_2, \ldots, a_n$.
- In order to learn the right sequence of actions the agent must interact with its environment selecting at each moment one action a_{t+1} as a function of the current environment state s_t. This is $a_{t+1} = F(s_t)$.

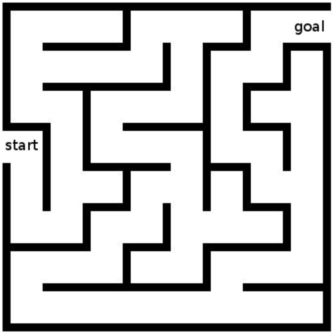
Reinforcement Learning

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



Reinforcement Learning

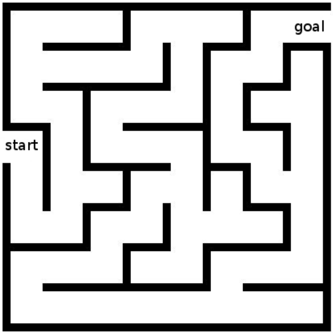
Markov decision process. MDP $= (S, A, T, R)$



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

Reinforcement Learning

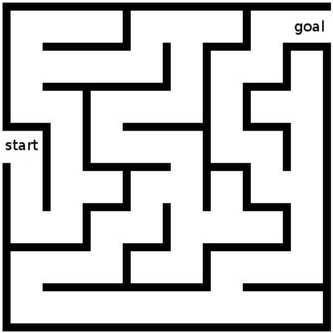
Markov decision process. MDP $= (S, A, T, R)$



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

Reinforcement Learning

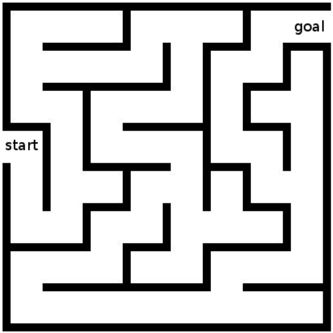
Markov decision process. $\text{MDP} = (S, A, T, R)$



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

Reinforcement Learning

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



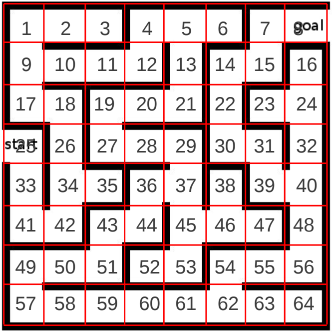
- Set of states $S = \{s_1, s_2, \ldots\}$
- Set of actions $A(s) = \{a_1, a_2, \ldots\}$
- Transition function (dynamics of the environment) $T : S \times A \to S$
- Reward function $R : S \to \mathbb{R}$

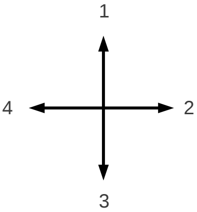
Possible RL solution to the maze problem

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

Possible RL solution to the maze problem

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





Reward Function:

$r = -1$ if state is not terminal

$r = 10$ if state is terminal

Function $V(s)$

S	$V(s)$
1	0
2	0
3	0
4	0
$\ldots$	$\ldots$
64	0

Function $V(s)$

S	$V(s)$
1	0
2	0
3	0
4	0
$\dots$	$\dots$
64	0

S	$V(s)$
1	-20
2	-20
3	-5
4	-22
$\dots$	$\dots$
64	-24

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

Q-learning Algorithm

Initialize $Q(s, a)$ arbitrarily

Q-learning Algorithm

Initialize $Q(s, a)$ arbitrarily
Repeat (for each episode):

Q-learning Algorithm

Initialize $Q(s, a)$ arbitrarily
Repeat (for each episode):
    Initialize s

Q-learning Algorithm

Initialize $Q(s, a)$ arbitrarily
Repeat (for each episode):
    Initialize s
    Repeat (for each step of episode):

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*)

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

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

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'
        $Q(s, a) \leftarrow Q(s, a) + \alpha\left[r + \gamma \max_{a'} Q(s', a') - Q(s, a)\right]$
        $s \leftarrow s'$
    Until s is terminal

Some videos

- Inverted pendulum
- Object handling

Thank you!

Dr. Víctor Uc Cetina
cetina@informatik.uni-hamburg.de