

# Applied Machine Learning

Introduction

Oumar Kaba

# TAs

<b>Charlotte Volk</b>	My name is Charlotte, I've just finished up the first year of my masters and am planning to fast track to PhD in the winter. I'm studying biologically plausible learning algorithms with Blake Richards. This is my 3rd time TAing for COMP 551, I'm excited to get started	Monday 2:15-3:15
<b>Rafid Saif</b>	I'm Rafid, I am also finishing the first year of my masters in Electrical Engineering. I'm currently working on skill-based learning for robotics and am advised by Hsiu-Chin Lin	Wednesday 1:00-2:00
<b>Mohamad Danesh</b>	I'm a 3rd year PhD student at CS working on robot learning under Hsiu-Chin Lin.	Friday 12:00-1:00
<b>Sebastian Sabry</b>	My name is Sebastian, I've finished my first year of my masters switching to a Thesis with Dr. Rabbany! I'm excited to TA for COMP 551.	Thursday 12:00-1:00
<b>Zahra Tehraninasab</b>	My name is Zahra and I'm a master's student working on Computer Vision for medical image analysis under professor Tal Arbel. This is my second time TAing for COMP 551. Looking forward to meeting you all!	Friday 11:00-12:00

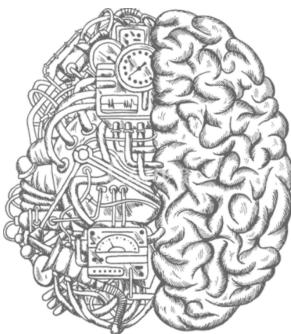
# Logistics

- Course website: <https://oumarkaba.github.io/comp551/comp551.html>
- Prerequisites quizzes in Mycourses  TODO
- Join a study group in Mycourses  TODO
- All office hours are now accessible in Zoom through Mycourses
- My office hours will be 30 minutes at the end of each class
- Be on the lookout for tutorials on Python and Math
  - Go to the TA's office hour for questions
- Any questions on the logistics?

# Outline

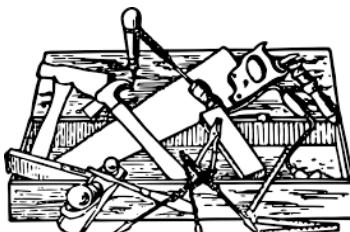
- a short history of machine learning
- understanding the scope of machine learning
  - relation to other areas
- understanding types of machine learning

# What is Machine Learning?



ML is the set of "algorithms and statistical models that computer systems use to perform a specific task without using explicit instructions"

ML is the "study of computer algorithms that improve automatically through experience"



while there are some unifying principles, machine learning may still look like a toolbox with different tools suitable for different tasks

# What is Machine Learning?

## Software 2.0



Andrej Karpathy [Follow](#)  
Nov 11, 2017 · 8 min read

## Software 1.0



## Software 2.0



Worth nuancing this perspective

# Placing Machine Learning

- **Artificial Intelligence:** it's a broader domain (includes search, planning, multiagent systems, robotics, etc.)
- **Statistics:** historically precedes ML. ML is more focused on algorithmic, practical and powerful models (e.g., neural networks) and is built around AI
- **Vision & Natural Language Processing:** use many ML algorithms and ideas
- **Optimization:** extensively used in ML
- **Data mining:** scalability, and performance comes before having theoretical foundations, more space for using heuristics, exploratory analysis, and unsupervised algorithms
- **Data science:** an umbrella term for the above mostly used in industry when the output is knowledge/information to be used for decision making

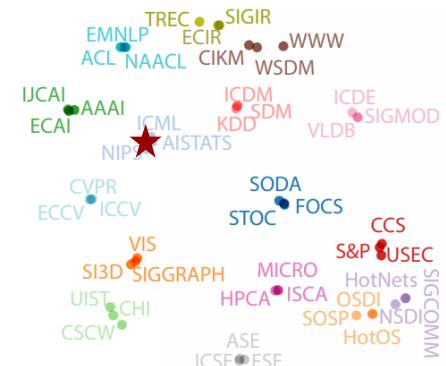
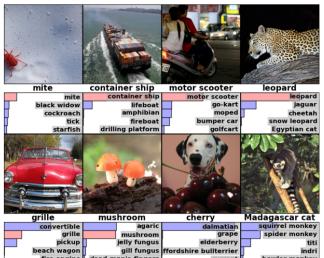


Figure from Dong et al.

# A short history of Machine Learning



- 1950: Turing test
- 1956: checker player that learned as it played (Arthur Samuel)
- 1958: first artificial neural networks called Perceptron (Frank Rosenblatt),
- 1963: support vector machines (Vapnick & Chervonenkis)
- 1969: Minsky and Pappert show the limitations of single-layer neural networks
- 1970-80s: rule-based and symbolic AI dominates (two AI winters)
- 1980's: Bayesian networks (Judea Pearl)
- 1986: Backpropagation rediscovered (Rumelhart, Hinton & Williams)
- 1980-1990s: expert systems are being replaced with general-purpose computers
- ...
- 2012: AlexNet wins Imagenet by a large margin
- 2022: ChatGPT launched, Large Language Models (LLMs) become widespread
- 2012 - now: a new AI spring around deep learning ...
- what is next?



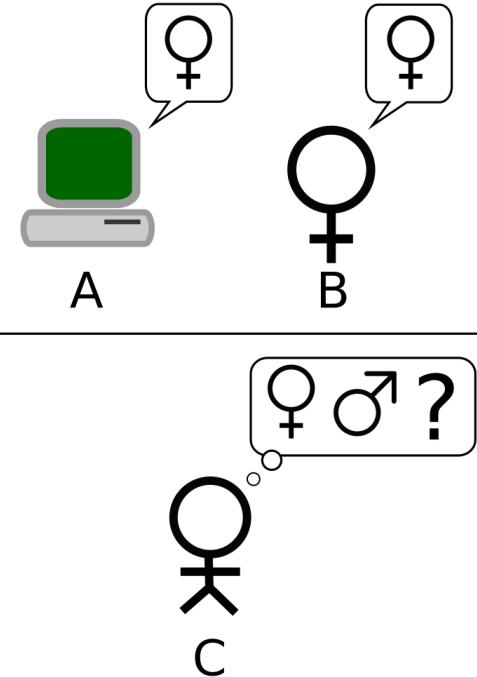
# Turing test

In his paper on "Computing Machinery and Intelligence by Alan Turing (**1950**)", Turing tried to replace the abstract question of "Can machines think?" with something more tangible, the Turing test designed based on a party game.



*“What will happen when a machine takes the part of A in this [Imitation] game? Will the interrogator decide wrongly as often when the game is played like this as he does when the game is played between a man and a woman? These questions replace our original, “**Can machines think?**”*

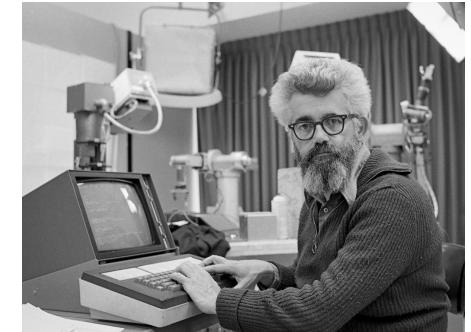
See more on: [https://en.wikipedia.org/wiki/Turing\\_test](https://en.wikipedia.org/wiki/Turing_test)



Player C, through a series of written questions, attempts to determine which of the other two players is a man, and which of the two is the woman. Player A, the man, tries to trick player C into making the wrong decision, while player B tries to help player C.

# Artificial Intelligence

John McCarthy coined the term Artificial Intelligence (AI) and organized the first AI conference in 1956 to bring together researchers to design thinking machines, read more about it [here](#)



The [Logic Theorist program](#), "the first artificial intelligence program", designed by Allen Newell, Herbert A. Simon and Cliff Shaw was presented in this conference. It was able to do automated **reasoning**, i.e. proving mathematical theorems from scratch by exploring a **search** tree, with the hypothesis as the root and branches as logical deductions, plus ad hoc rules, **heuristics**, to trim some branches and avoid exponential growth

[ brute force or exhaustive search looks at all the possible options, to find the solution: a simple but expensive approach ]

# Machine Learning

In 1959 Arthur Samuel popularized the term Machine Learning through his seminal paper on "Some Studies in Machine Learning Using the Game of Checkers"

*" a computer can learn to play a better game of checkers than its programmer given only rule of the game, a sense of direction, and a redundant and incomplete list of **parameters** that have something to do with the game but whose values are unknown and unspecified. Programming computers to **learn from experience** should eventually eliminate the need for much of this detailed programming efforts.*



**" Game as a vehicle for studying ML**

many important concepts: self-play, temporal difference learning, function approximation

[ you can read the paper [here](#) ]



How it learned/was trained?

min-max search with alpha-beta pruning,  
learning to estimate the value of a state

**" In 1954, four different IBM 704 machines working from midnight to 7 AM playing checkers with themselves and assimilating statistics that they used for the running scheme**

[ read about Samuel [here](#) ]

Psychologist Shows Embryo  
of Computer Designed to  
Read and Grow Wiser

WASHINGTON, July 7 (UPI)—The Navy revealed the embryo of an electronic computer today that it expects will be able to walk, talk, see, write, reproduce itself and be conscious of its existence.

The embryo—the Weather Bureau's \$2,000,000 "T-04" computer—learned to differentiate between right and left after attempting the "T-04's" demonstration for newsmen.

The service said it would use this principle to build the first true learning machine—a machine that will be able to read and write. It is expected to be finished in about a year at a cost of \$10,000,000.

Dr. Frank Rosenblatt, designer of the Perceptron, conducted the demonstration. He said the machine will be the first device to think as the human brain. As do human beings, Perceptrons must make mistakes at first, but will grow wiser as it gains experience, he said.

Dr. Rosenblatt, a research psychologist at the Cornell Aeronautical Laboratory, Buffalo, said Perceptrons might be fired to the planets as mechanical space explorers.

**Wires Run in Control**

The Navy said the perceptron would be the first non-living mechanism "capable of receiving, recognizing and identifying things as well as performing any human training or control."

The "brain" is designed to remember images and identify them. At present, most modern computers remember only what is fed into them on punch cards or magnetic tape.

Later, Perceptrons will be able to recognize people and call out their names and instantly translate the spoken word into speech or writing in another language, it was predicted.

Mr. Rosenblatt said in principle, Perceptrons planned to build brains that could reproduce themselves on an assembly line and which would be capable of learning a "new" skill.

In today's demonstration, the "T-04" was fed two cards, one with squares marked on the left side and the other with squares on the right side.

**Learn by Doing**

In the first fifty trials, the machine made no distinction between them. It then started requesting a "0" for the left squares and an "O" for the right squares.

Dr. Rosenblatt said he could explain why the machine learned so quickly in highly technical terms. But he said the computer has a "natural" ability to make changes in the wiring diagram.

The first Perceptron will have about 1,000 electronic components, including a scanning electrical impulse from an eye-like scanning device with 400 photo-cells. The human brain has about 100,000,000 neurons, cells, including 100,000,000 connections with the eyes.

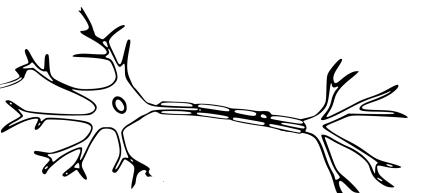
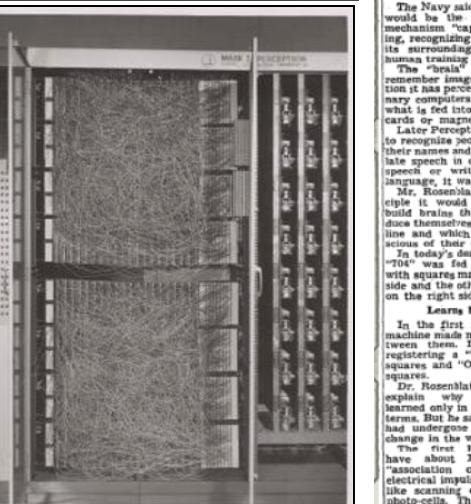
12 Books Today

# Neural Networks

The Perceptron - A Perceiving and Recognizing Automaton  
by Frank Rosenblatt (1957), [you can read the paper [here](#)]



Mark I Perceptron machine (1958)  
Custom built hardware



$$f(x) = \sigma \left( \sum_i w_i x_i \right)$$

activation function

which was in turn based on **Hebbian learning**

We will discuss Perceptron later in the course

[ read more about Perceptron [here](#), and on Rosenblatt's [here](#) ]

[ read the [article](#) from nytimes on July 8, 1958 ]

# How do humans think?



Earliest works in the 1930s and 1940s by **Donald Hebb**, a psychologist at **McGill**, Studying behaviour in terms of brain function and connections between neuron assemblies

*"I am referring to the general type of studies based on Donald Hebb's work at McGill. The argument goes something like this. The brain of man, like that of the animals, is made up of many cells of a certain type called neurons. These cells... react on an all-or-none basis ('fire'; ...) and transmit a pulse to other neurons through synaptic connections. Each neuron is connected to many others, and a number of input signals are, in general, required before a neuron will 'fire'. ...Learning seems to consist of alterations in the strength and even perhaps in the number of these synaptic interconnections. Now it is possible to devise a variety of mechanical, chemical, and electrical devices which simulate the behavior of individual neurons in a crude sort of way, and we can interconnect these devices in some random fashion to simulate the synaptic interconnections that exist within the brain, and, finally, we can arrange for the automatic strengthening or weakening of these interconnections using a training routine.*



# Early Real World Applications

ADALINE (Adaptive Linear Neuron) and MADALINE (Many ADALINEs) are similar to Perceptron and were proposed by Bernard Widrow et al., 1958 and 1960 [you can read the paper [here](#)]

Had many real-world applications including adaptive echo canceler for telephones, automatic equalizations for modems, speech and pattern recognition, weather forecasting etc. [read more about it, [here](#)]

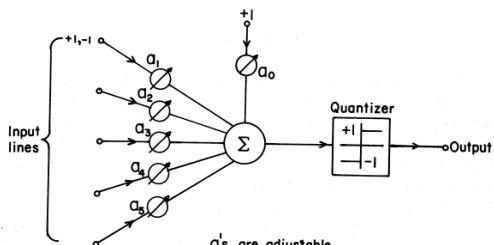


FIG. 1. -- AN ADJUSTABLE NEURON.

Trained using LMS algorithm: adjusting the weights based on the approximate gradient predecessor to backpropagation

## Letter Recognition



FIG. 2. -- ADALINE.

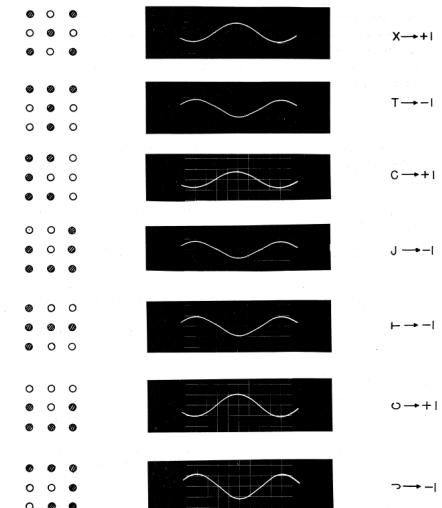
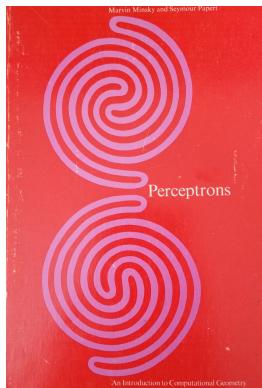


FIG. 12. WAVE-FORMS OF A MEMISTOR NEURON AFTER TRAINING EXPERIMENT.

# A short history of Machine Learning

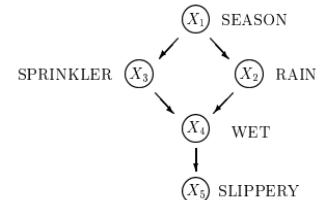


- 1950: Turing test
- 1956: checker player that learned as it played (Arthur Samuel)
- 1958: first artificial neural networks called Perceptron (Frank Rosenblatt),
  - **1963:** support vector machines (Vapnick & Chervonenkis)
    - we will discuss SVM's idea later in the course
- **1969:** Minsky and Papert show the limitations of single-layer neural networks
  - for example, it cannot learn a simple XOR function
  - the limitation does not extend to a multilayer perceptron (which was known back then)
  - one of the factors in so-called [AI winter](#)
- **1970-80s:** rule-based and **symbolic AI** dominates
  - in contrast to **connectionist AI** as in neural networks
  - expert systems find applications in industry
  - these are rule-based systems with their specialized hardware

# A short history of Machine Learning

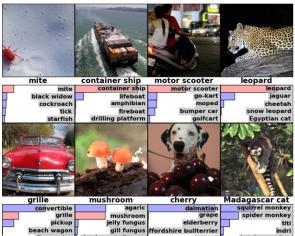


- 1950: Turing test
- 1956: checker player that learned as it played (Arthur Samuel)
- 1958: first artificial neural networks called Perceptron (Frank Rosenblatt),
- 1963: support vector machines (Vapnick & Chervonenkis)
- 1969: Minskey and Pappert show the limitations of single-layer neural networks
- 1970-80s: rule-based and symbolic AI dominates
- **1980s:** Bayesian networks (Judea Pearl)
  - combine graph structure with probabilistic (and causal) reasoning
  - related to both symbolic and connectionist approach
- **1986:** Backpropagation rediscovered (Rumelhart, Hinton & Williams)
  - an efficient method for learning the weights in neural networks using gradient descent
  - it was rediscovered many times since the 1960s
  - we discuss it later in the course
- **1980-1990s:** expert systems are being replaced with general-purpose computers



# A short history of Machine Learning

- 1950: Turing test
  - 1956: checker player that learned as it played (Arthur Samuel)
  - 1958: first artificial neural networks **Perceptron**, and **ADELINe** (1959)
  - 1963: support vector machines (Vapnik & Chervonenkis)
  - 1969: Minsky and Pappert show the limitations of **single-layer Perceptron**
  - 1970-1980: ruled based and **symbolic AI** dominates
  - 1980s: Bayesian networks (Judea Pearl)
  - 1986: Backpropagation rediscovered (Rumelhart, Hinton & Williams)
  - 1980-1990s: expert systems are being replaced with general-purpose computers
- ...
- **1997**: Deep Blue beats the world chess champion
  - **2012**: AlexNet wins Imagenet by a large margin
  - **2012 - now** a new AI spring around deep learning ...



Yoshua Bengio, Geoffrey Hinton, Yann LeCun

# A short history of Machine Learning

- 1950: Turing test
- 1956: checker player that learned as it played (Arthur Samuel)
- 1958: first artificial neural networks **Perceptron**, and **ADELINe** (1959)
- 1963: support vector machines (Vapnik & Chervonenkis)
- 1969: Minsky and Pappert show the limitations of **single-layer Perceptron**
- 1970-1980: ruled based and **symbolic AI** dominates
- 1980s: Bayesian networks (Judea Pearl)
- 1986: Backpropagation rediscovered (Rumelhart, Hinton & Williams)
- 1980-1990s: expert systems are being replaced with general-purpose computers

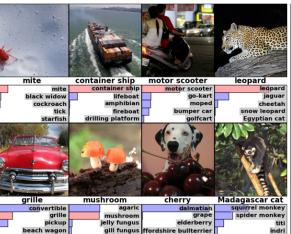
...

- 1997: Deep Blue beats the world chess champion
- 2012: AlexNet wins Imagenet by a large margin
- 2012 - now a new AI spring around deep learning ...

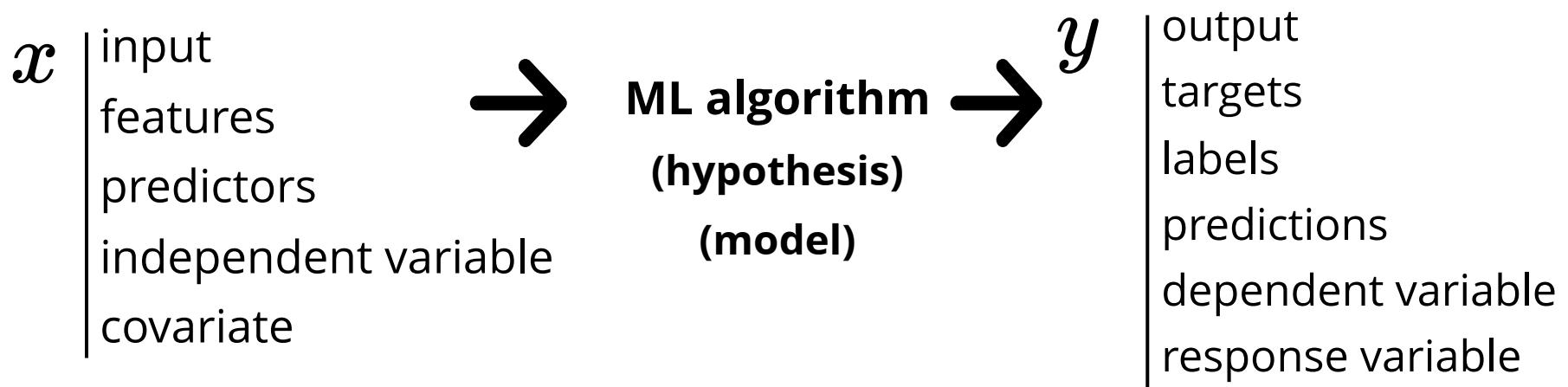
- super-human performance in many tasks
  - e.g. AlphaGo defeats Go Master (2017)
- LLMs are widely adopted by the general public

- **Future:** what is next?

- in the short term, AI will impact domain sciences, and... everything?



# Basic Terminology



example

$\langle \text{tumorsize, texture, perimeter} \rangle = \langle 18.2, 27.6, 117.5 \rangle$



cancer = No

# Basic Terminology

(labelled) **datasets**: consist of many training examples, samples or **instances**

<tumorsize, texture, perimeter> , <cancer, size change>

<18.2, 27.6, 117.5> , <No , +2 >

$x^{(1)}, y^{(1)}$

<17.9, 10.3, 122.8> , <No , -4 >

$x^{(2)}, y^{(2)}$  one instance

<20.2, 14.3, 111.2> , <Yes , +3 >

$x^{(3)}, y^{(3)}$

•

•

•

•

•

•

<15.5, 15.2, 135.5> , <No , 0 >

$x^{(N)}, y^{(N)}$

# Basic Terminology

we split the dataset into **train** and **test** sets

**Train dataset:**  
used to build the model

```
<tumorsize, texture, perimeter> , <cancer, size change>
<18.2,      27.6,      117.5> , < No ,    +2   >
<17.9,      10.3,      122.8> , < No ,    -4   >
<20.2,      14.3,      111.2> , < Yes ,   +3   >
<15.5,      15.2,      135.5> , < No ,    0    >
.
.
.
```

algorithm shouldn't have access  
to test set when being trained

**Test dataset:**  
used to evaluate the model

```
<tumorsize, texture, perimeter> , <cancer, size change>
<12.4,      15.7,      120.1> , < No ,    +5   >
<15.2,      17.2,      113.3> , < Yes ,   +1   >
<19.3,      15.9,      125.4> , < No ,    +2   >
<17.5,      11.9,      122.7> , < No ,    -3   >
.
.
```

Ground-Truth, True labels

```
<cancer, size change>
< Yes ,    +4   >
< Yes ,    +1   >
< No ,    +1   >
< Yes ,   -2   >
```

Output labels, algorithm results

algorithm shouldn't see the  
true labels when being  
evaluated (making predictions  
on test set), these true labels  
are only used to compare  
against the algorithm's results  
to measure performance

# Families of Machine Learning Methods

1. Supervised learning
2. Unsupervised & self-supervised learning
3. Semi-supervised learning
4. Reinforcement learning ...

# Families of Machine Learning Methods

**1. Supervised learning:** we have labeled data

most of this course

- classification
- regression

$\mathcal{D}$  : training set

$x$  :  $D$ -dimensional vector

$y$  : a categorical or nominal variable

$N$  : number of training instances

$n$  : index of training instance ( $n \in \{1 \dots N\}$ )

$$\mathcal{D} = \{(x^{(n)}, y^{(n)})\}_{n=1}^N$$

pairs of input vector  
and corresponding  
target or label

$x^{(1)}$	<tumorsize, texture, perimeter> , <cancer>			$y^{(1)}$	
$x^{(2)}$	<18.2,	27.6,	117.5>	, < No >	$y^{(2)}$
$x^{(3)}$	<17.9,	10.3,	122.8>	, < No >	$y^{(3)}$
$\vdots$				$\vdots$	$\vdots$
$x^{(N)}$	<15.5,	15.2,	135.5>	, < No >	$y^{(N)}$

indexes can be placed up or down based on the notation in use, or dropped all together.  
When up, not to be confused with a power

# Families of Machine Learning Methods

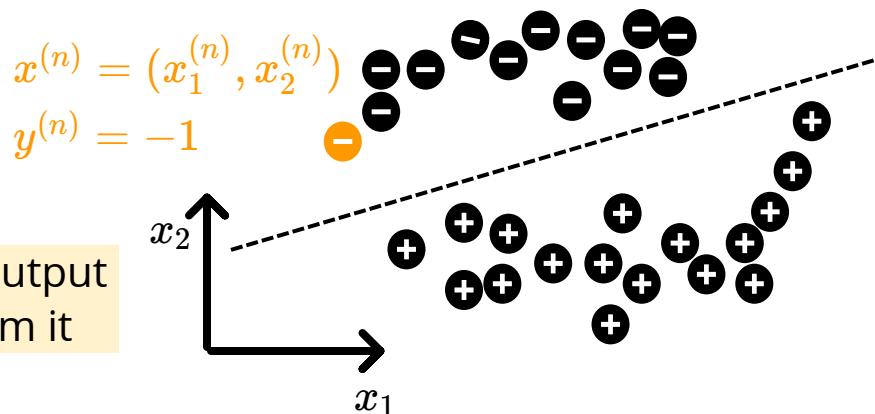
**1. Supervised learning:** we have labeled data

most of this course

- classification
- regression

$$\mathcal{D} = \{(x^{(n)}, y^{(n)})\}_{n=1}^N$$

shows samples with desired output  
to the algorithm to learn from it



# Families of Machine Learning Methods

**1. Supervised learning:** we have labeled data

**Classification:** categorical/discrete output

<tumorsize, texture, perimeter>	,	<cancer>
<18.2,	27.6,	117.5>
<17.9,	10.3,	122.8>
<20.2,	14.3,	111.2>
<15.5,	15.2,	135.5>
,		
< No >		
< Yes >		
< No >		

target

**Regression:** continuous output

<tumorsize, texture, perimeter>	,	<size change>
<18.2,	27.6,	117.5>
<17.9,	10.3,	122.8>
<20.2,	14.3,	111.2>
<15.5,	15.2,	135.5>
,		
< +2 >		
< -4 >		
< +3 >		
< 0 >		

target

# Families of Machine Learning Methods

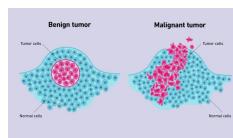
## 1. Supervised learning: we have labeled data

### Classification: categorical/discrete output

#### binary classification

classifying benign (noncancerous) vs malignant (cancerous) tumors

```
<tumorsize, texture, perimeter> , <cancer>
<18.2,      27.6,      117.5> , < No >
<17.9,      10.3,      122.8> , < No >
<20.2,      14.3,      111.2> , < Yes >
<15.5,      15.2,      135.5> , < No >
```



#### multiclass classification classifying Iris flowers

N = 150 instances of flowers

D=4 features {the length and the width of the sepals and petals}

C=3 classes {setosa, versicolor, virginica} : 50 samples of each

index	sl	sw	pl	pw	label
0	5.1	3.5	1.4	0.2	Setosa
1	4.9	3.0	1.4	0.2	Setosa
	...				
50	7.0	3.2	4.7	1.4	Versicolor
	...				
149	5.9	3.0	5.1	1.8	Virginica



# Families of Machine Learning Methods

## 1. Supervised learning: we have labeled data

classifying Iris flowers

N = 150 instances of flowers

D=4 features {the length and the width of the sepals and petals}

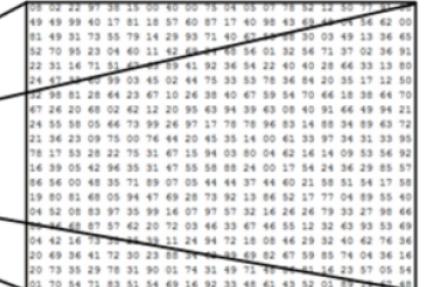
C=3 classes {setosa, versicolor, virginica} : 50 samples of each

index	sl	sw	pl	pw	label
0	5.1	3.5	1.4	0.2	Setosa
1	4.9	3.0	1.4	0.2	Setosa
...					
50	7.0	3.2	4.7	1.4	Versicolor
...					
149	5.9	3.0	5.1	1.8	Virginica

N x D design matrix:  
tabular data with



## Image classification



82% cat  
15% dog  
2% hat  
1% mug

# Supervised Learning: Example

MIT  
Technology  
Review

Topics+      The Download

---

Intelligent Machines

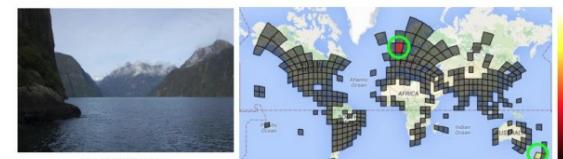
## Google Unveils Neural Network with “Superhuman” Ability to Determine the Location of Almost Any Image

Guessing the location of a randomly chosen Street View image is hard, even for well-traveled humans. But Google’s latest artificial-intelligence machine manages it with relative ease.

by Emerging Technology from the arXiv    February 24, 2016



(a)



trained on a database of geolocated images from the Web

[ read about it [here](#) ]

# Supervised Learning: Example



trained using billions of high-quality translation segments from reliable sources such as the European Parliament, Unesco patents, and literary works, bilingual sentences collected by Linguee's web crawler on the Internet

[ read about it [here](#), try it out [here](#) ]

**Machine Translation:** data consists of input-output sentence pairs  $(x,y)$ , similarly we may consider **text-to-speech**, with text and voice as input and target  $(x,y)$ , or **speech recognition** where input and output above are swapped.

# Supervised Learning: Example

Supervised methods are powered by large datasets often crawled from the web or curated with crowdsourcing

## What is COCO?



COCO is a large-scale object detection, segmentation, and captioning dataset. COCO has several features:

- ✓ Object segmentation
- ✓ Recognition in context
- ✓ Superpixel stuff segmentation
- ✓ 330K images (>200K labeled)
- ✓ 1.5 million object instances
- ✓ 80 object categories
- ✓ 91 stuff categories
- ✓ 5 captions per image
- ✓ 250,000 people with keypoints

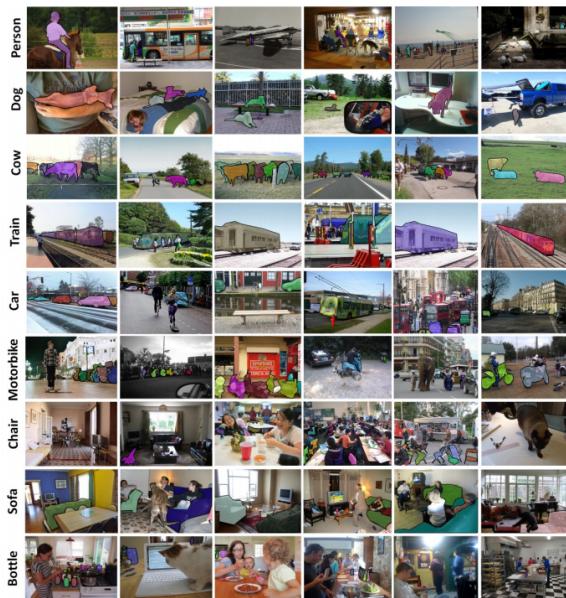
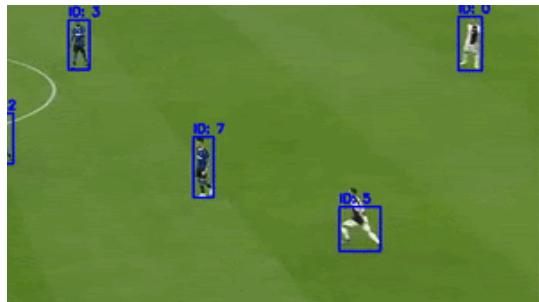


Fig. 6: Samples of annotated images in the MS COCO dataset.

## Object Recognition

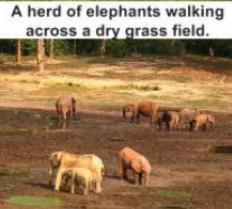
<https://medium.com/data-science/football-games-analysis-from-video-stream-with-machine-learning-745e62b36295>



**input:** image

**output:** a set of bounding box coordinates

# Supervised Learning: Example



Describes without errors

Describes with minor errors

Somewhat related to the image

Unrelated to the image

read about it [here](#)

## Image Captioning

**input:** image

**output:** text



Creating Images from Text  
[paper [here](#)]

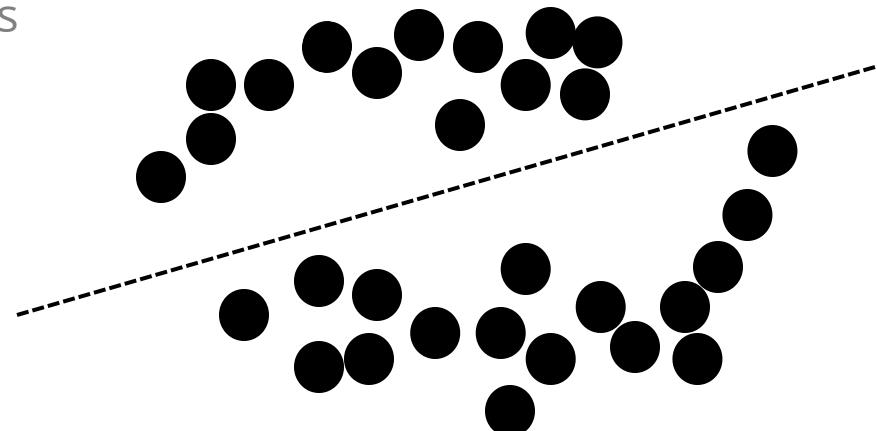
# Families of Machine Learning Methods

## 2. Unsupervised Learning: only unlabeled data

- clustering
- dimensionality reduction
- density estimation / generative modeling
- anomaly detection
- discovering latent factors and structures

The algorithm doesn't see the desired outputs, mines the patterns in the input data

- helps explore and understand the data
- closer to data mining
- we have much more unlabeled data
- more open challenges

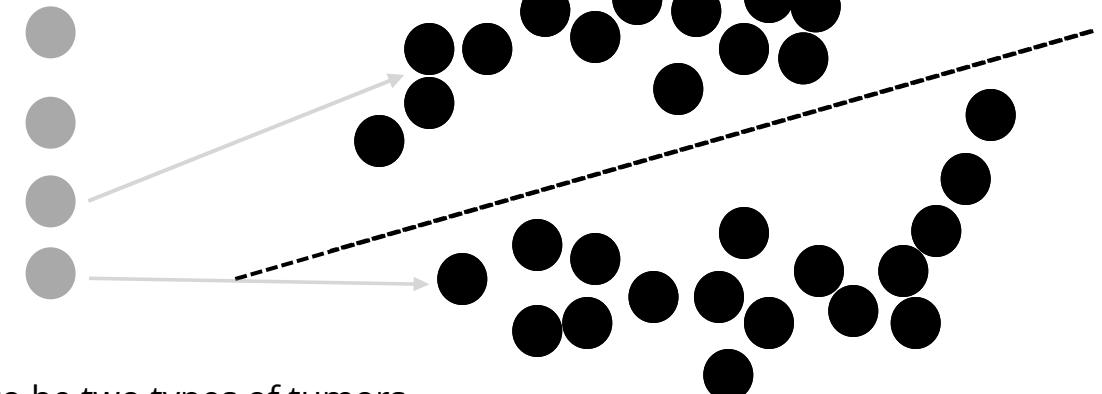


# Unsupervised Learning: Example

**Clustering:** similar to classification but labels/classes should be inferred and are not given to the algorithm

<tumorsize, texture, perimeter>

<18.2,	27.6,	117.5>
<17.9,	10.3,	122.8>
<20.2,	14.3,	111.2>
<15.5,	15.2,	135.5>



there seems to be two types of tumors,  
larger and rough vs smaller and smooth

# Unsupervised Learning: Example

**Clustering:** similar to classification but labels/classes should be inferred and are not given to the algorithm

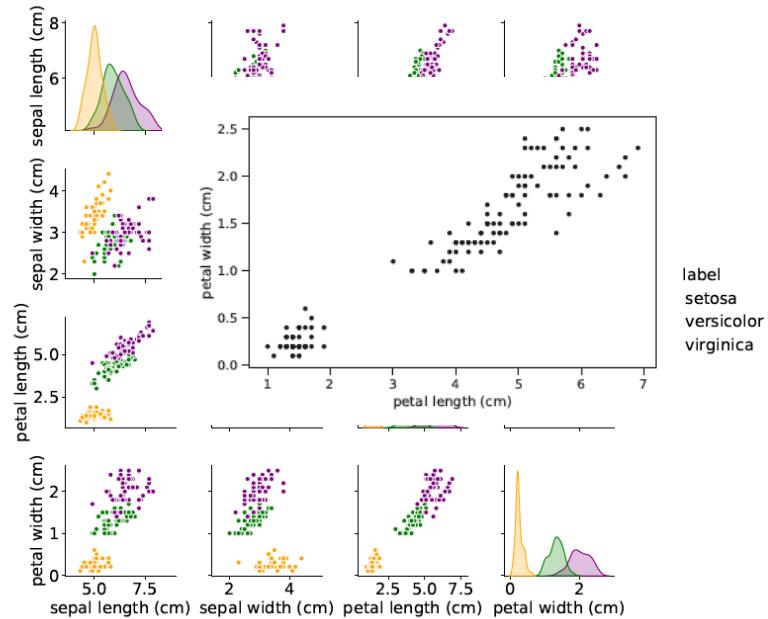
## clustering Iris flowers

N = 150 instances of flowers

D=4 features {the length and the width of the sepals and petals}

C=3 classes {setosa, versicolor, virginica} : 50 samples of each

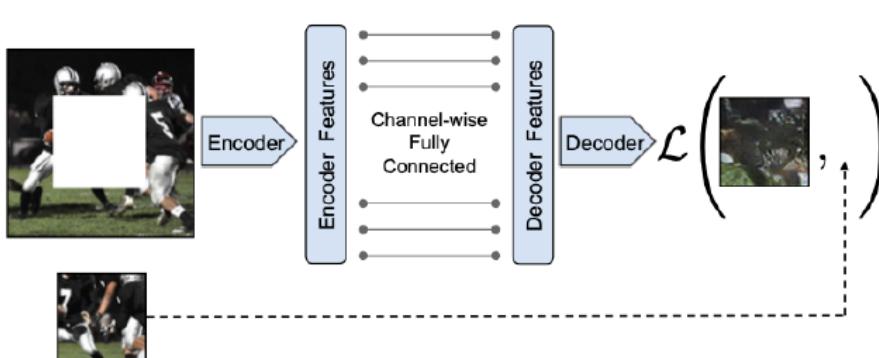
index	sl	sw	pl	pw	label
0	5.1	3.5	1.4	0.2	Setosa
1	4.9	3.0	1.4	0.2	Setosa
...					
50	7.0	3.2	4.7	1.4	Versicolor
...					
149	5.9	3.0	5.1	1.8	Virginica



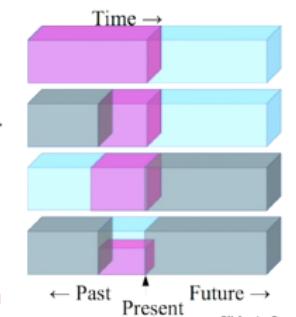
# Unsupervised Learning: Example

**Self supervised learning:** create proxy supervised tasks from unlabeled data, e.g. predict a color image from a grayscale image or mask out words in a sentence and then try to predict them given the surrounding context

Goal: learn useful features from the data, that can then be used in standard, downstream supervised tasks [see a list of relevant papers [here](#)]



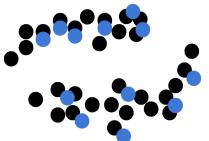
- ▶ Predict any part of the input from any other part.
- ▶ Predict the **future** from the **past**.
- ▶ Predict the **future** from the **recent past**.
- ▶ Predict the **past** from the **present**.
- ▶ Predict the **top** from the **bottom**.
- ▶ Predict the **occluded** from the **visible**
- ▶ **Pretend there is a part of the input you don't know and predict that.**



Let's make [MASK] chicken! [SEP] It [MASK] great with orange sauce.

# Unsupervised Learning: Example

**Generative Modelling:** model the distribution of the data and learns to generate the data instead of directly categorizing/discriminating the instances into different classes



Generate faces that look like celebrity images

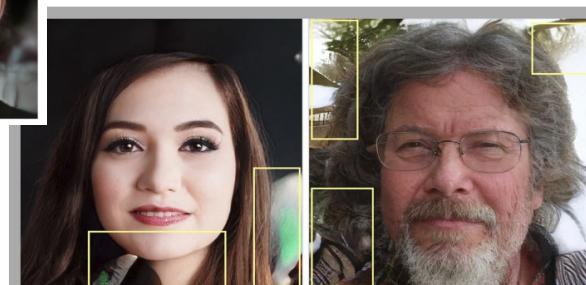
from the paper [here](#)

click [here](#) to get a random fake person

Ethical  
Challenges:  
Misuse

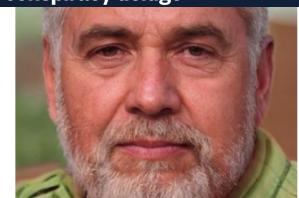
## Facebook Removes Accounts With AI-Generated Profile Photos

Researchers said it appears to be the first use of artificial intelligence to support an inauthentic social media campaign.



Profile pictures for Facebook accounts "Mary Keen" and "Jacobs Guillermo," admins on groups associated with The BL highlighted by Graphika. COURTESY OF GRAPHIKA

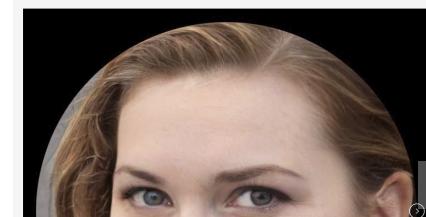
How a fake persona laid the groundwork for a Hunter Biden conspiracy deluge



A viral dossier about Hunter Biden was written by "Martin Aspen," a fake identity whose profile picture was created by artificial intelligence.  
TyphoonInvestig / via Twitter

## Experts: Spy used AI-generated face to connect with targets

By RAPHAEL SATTER June 13, 2019

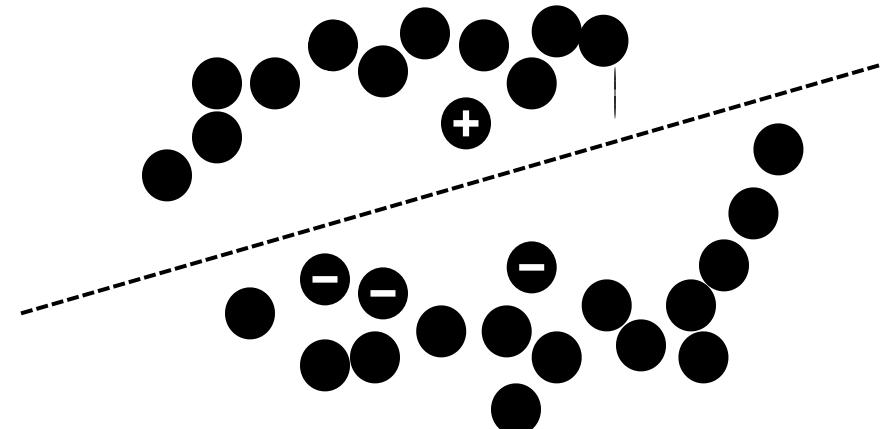


# Families of Machine Learning Methods

## 3. Semisupervised learning: a few labeled examples

- we can include structured problems such as
  - matrix completion (a few entries are observed)
  - link prediction

The algorithm sees few examples of the desired outputs



# Semisupervised Learning: Example

## Matrix Completion in Recommendation Systems

Predict what movies you will like based on what you liked sofar and what others users liked who like similar movies to you



[ figure from [here](#) ]

.9	-1	1	1	-.9
-.2	-.8	-1	.9	1
1	.1	.88	-1.08	0.9
-1	0	-0.9	1.0	-1.0
.2	-1	0.38	0.6	1.2
.1	1	-0.11	-0.9	-0.9
		1.0	-0.7	-1.18
			1.0	0.91

Ethical Challenges:  
Privacy of Users  
Polarizing Users

## Netflix Awards \$1 Million Prize and Starts a New Contest

BY STEVE LOHR SEPTEMBER 21, 2009 10:15 AM



Jason Kempin/Getty Images Netflix prize winners, from left: Yehuda Koren, Martin Chabert, Martin Plotte, Michael Jahrer, Andreas Toscher, Chris Volinsky and Robert Bell.

**Update | 1:45 p.m.** Adding details announced Monday about the extremely close finish to the contest.

[ read about it [here](#) ]

RYAN SINGEL SECURITY 03.12.18 02:48 PM

## NetFlix Cancels Recommendation Contest After Privacy Lawsuit



Netflix is canceling its second \$1 million Netflix Prize to settle a legal challenge that it breached customer privacy as part of the first contest's race for a better movie-recommendation engine.

# Families of Machine Learning Methods

## 4. Reinforcement Learning:

- weak supervision through the reward signal
- sequential decision making
- biologically motivated

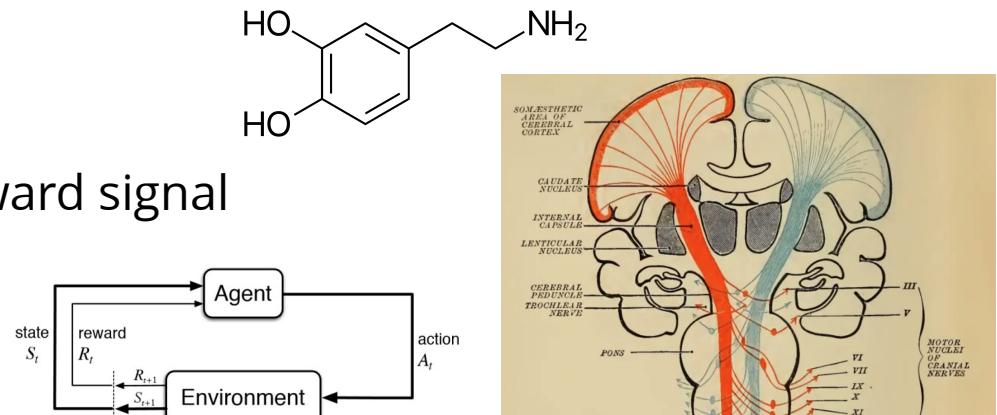


figure from [here](#)

also related:

**imitation learning:** learning from demonstrations

- behavior cloning (is supervised learning!)
- inverse reinforcement learning (learning the reward function)



figure from [here](#)

# Reinforcement Learning: Examples



## Human Level Control Through Deep Reinforcement Learning

### Abstract

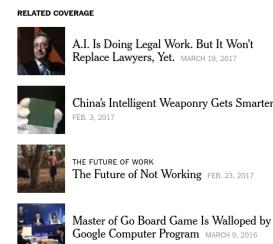
The theory of reinforcement learning provides a normative account deeply rooted in psychological and neuroscientific perspectives on animal behaviour, of how agents may optimize their control of an environment. To use reinforcement learning successfully in situations approaching real-world complexity, however, agents are confronted with a difficult task: they must derive efficient representations of the environment from high-dimensional sensory inputs, and use these to generalize

Playing Atari like a pro 2015, see [here](#)

### *Google's AlphaGo Defeats Chinese Go Master in Win for A.I.*

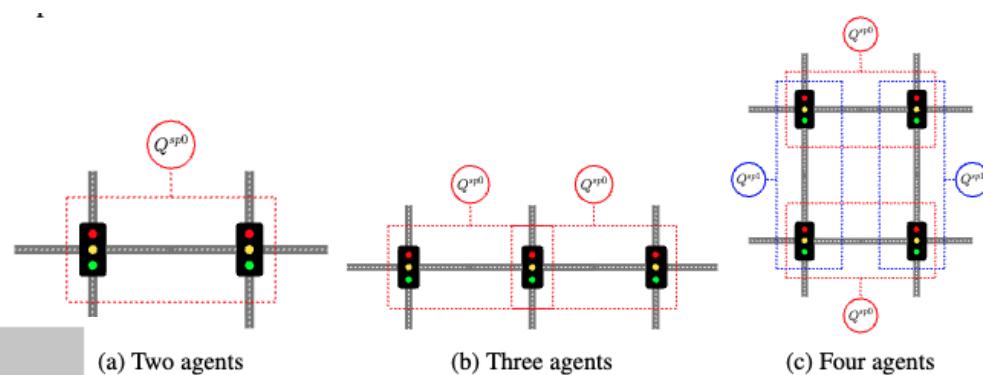
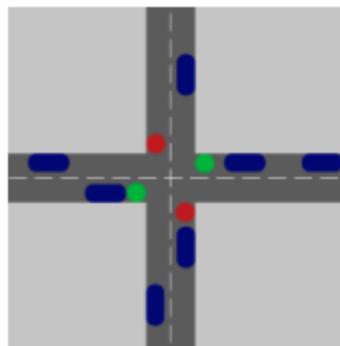
[点击查看本文中文版](#)

By PAUL MOZUR MAY 23, 2017



Playing Go like a pro 2017

# Reinforcement Learning: Examples



Traffic control via reinforcement learning, from [this paper](#)

# Questions

# Summary

**Supervised Learning:** we have labeled data

- classification
- regression

**Unsupervised Learning:** only unlabeled data

- clustering & self-supervised learning
- density estimation / generative modeling

**Semisupervised learning:** a few labeled examples

**Reinforcement Learning:** reward signal