

INTRODUCTION TO DEEP LEARNING

Winter School at UPC TelecomBCN Barcelona. 22-30 January 2018.



Instructors



Xavier
Giró-i-Nieto

Marta R.
Costa-jussà

Noé
Casas

Elisa
Sayrol

Antonio
Bonafonte

Verónica
Vilaplana

Ramon
Morros

Javier
Ruiz

Organizers



Supporters



aws educate

GitHub Education

+ info: <https://telecombcn-dl.github.io/2018-idl/>

<http://bit.ly/idl2018>



#DLUPC

Day 4 Lecture 2

Beyond Supervised Learning



Xavier Giro-i-Nieto
xavier.giro@upc.edu

Associate Professor
Universitat Politècnica de Catalunya
Technical University of Catalonia



Motivation

Yann Lecun's Black Forest cake



■ "Pure" Reinforcement Learning (cherry)

- ▶ The machine predicts a scalar reward given once in a while.
- ▶ **A few bits for some samples**



■ Supervised Learning (icing)

- ▶ The machine predicts a category or a few numbers for each input
- ▶ Predicting human-supplied data
- ▶ **10→10,000 bits per sample**

■ Unsupervised/Predictive Learning (cake)

- ▶ The machine predicts any part of its input for any observed part.
- ▶ Predicts future frames in videos
- ▶ **Millions of bits per sample**

■ (Yes, I know, this picture is slightly offensive to RL folks. But I'll make it up)

Motivation

We can categorize three types of learning procedures:

1. Supervised Learning:

$$\mathbf{y} = f(\mathbf{x})$$

2. Unsupervised Learning:

$$f(\mathbf{x})$$

3. Reinforcement Learning (RL):

$$\mathbf{y} = f(\mathbf{x})$$

$$\mathbf{z}$$



Acknowledgments

Hierarchical Object Detection with Deep Reinforcement Learning

NIPS 2016 Workshop on Reinforcement Learning
[github] [arXiv]

Míriam Bellver, Xavier Giró i Nieto, Ferran Marqués, Jordi Torres



Bellver M, Giró-i-Nieto X, Marqués F, Torres J. [Hierarchical Object Detection with Deep Reinforcement Learning](#). In Deep Reinforcement Learning Workshop, NIPS 2016. 2016.

Acknowledgments

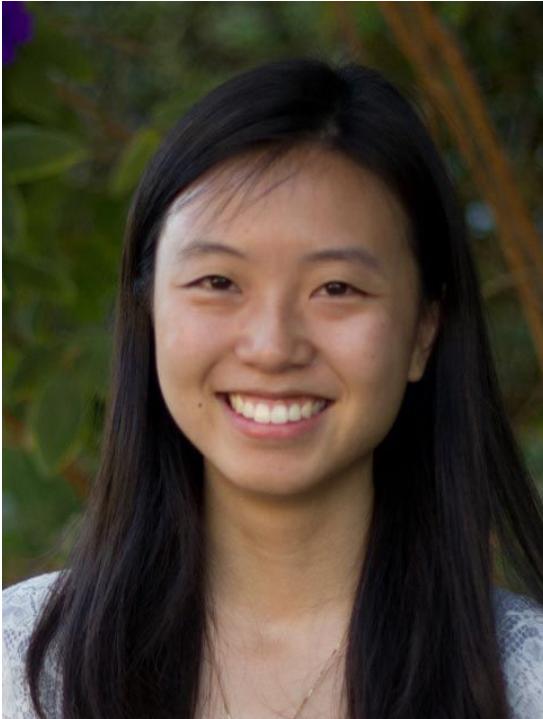


Lecture 14: Reinforcement Learning

Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 14 - 1

May 23, 2017

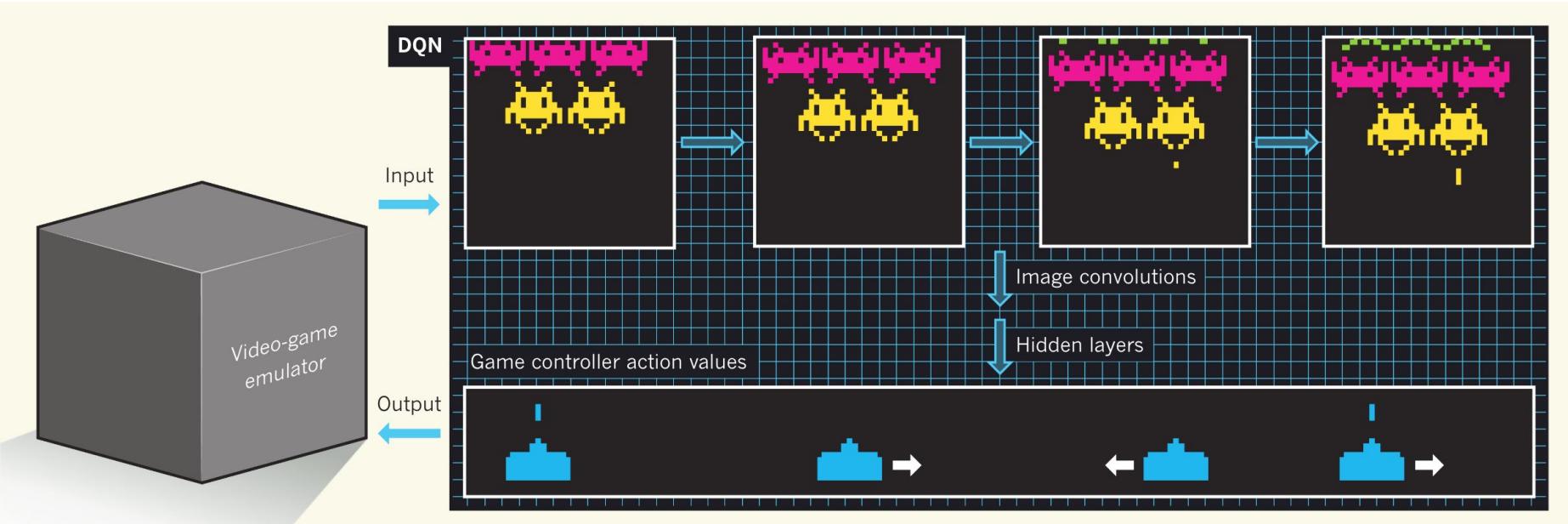


049 2 1



Mnih, Volodymyr, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller.
"Playing atari with deep reinforcement learning." arXiv preprint arXiv:1312.5602 (2013).

Motivation



Motivation



Google DeepMind

Artificial
intelligence
(AI)

Google buys UK artificial intelligence startup Deepmind for £400m

Google makes its biggest EU purchase yet with the technology that aims to make computers think like humans

Samuel Gibbs

Monday 27 January 2014
13.23 GMT



This article is 2 years
old

1046 186



<https://www.theguardian.com/technology/2014/jan/27/google-acquires-uk-artificial-intelligence-startup-deepmind>

Motivation



Silver, D., Huang, A., Maddison, C.J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M. and Dieleman, S., 2016. [Mastering the game of Go with deep neural networks and tree search](#). *Nature*, 529(7587), pp.484-489

Motivation



Greg Kohs, [“AlphaGo”](#) (2017)

Motivation



Vinyals, O., Ewalds, T., Bartunov, S., Georgiev, P., Vezhnevets, A.S., Yeo, M., Makhzani, A., Küttler, H., Agapiou, J., Schrittwieser, J. and Quan, J., 2017. [Starcraft II: A new challenge for reinforcement learning](#). arXiv preprint arXiv:1708.04782. [\[Press release\]](#)

Architecture



Figure: [UCL Course on RL by David Silver](#)

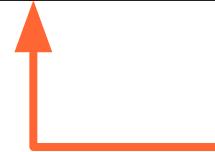
Architecture



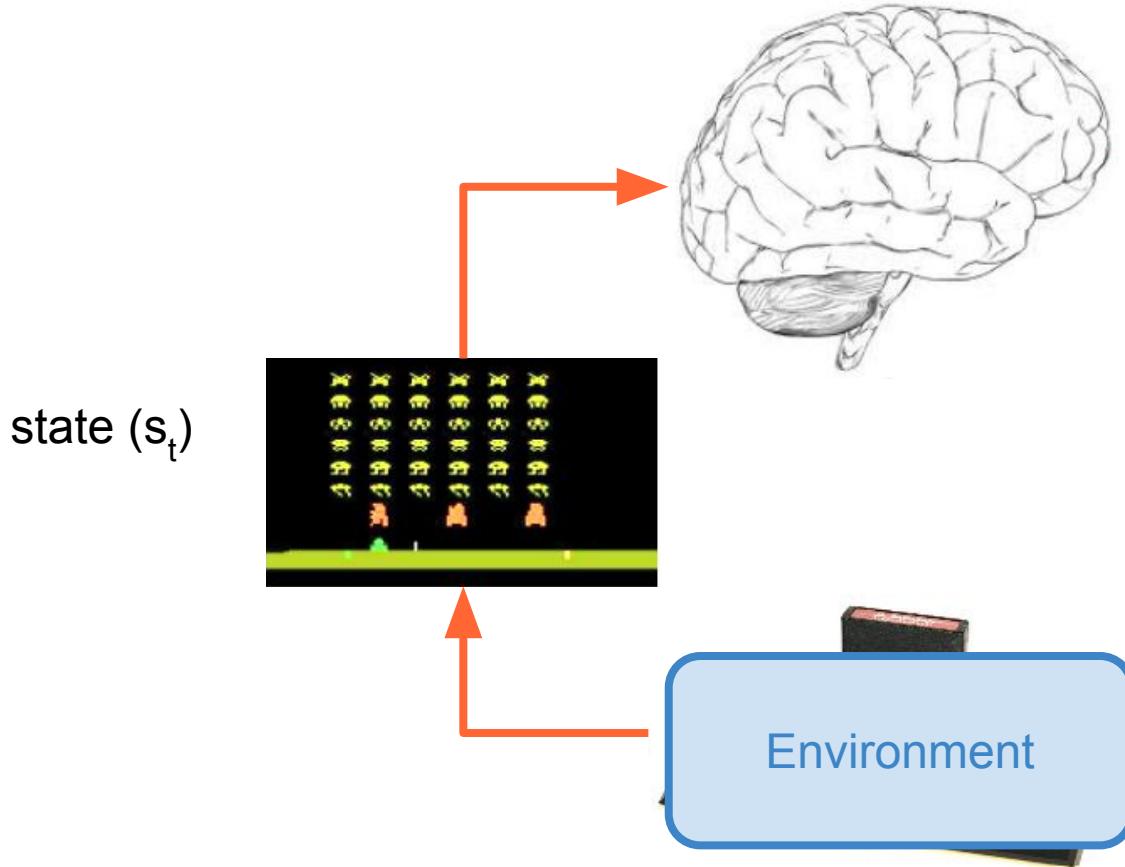
Figure: [UCL Course on RL by David Silver](#)

Architecture

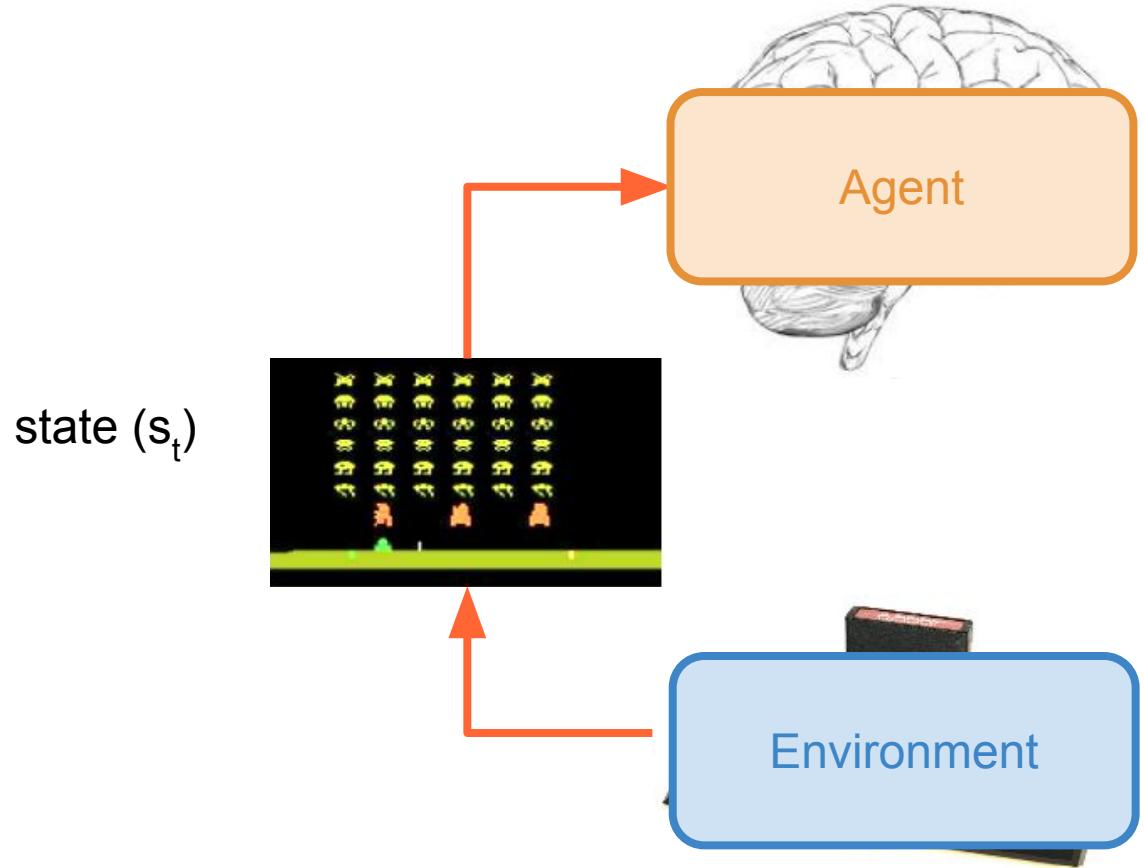
state (s_t)



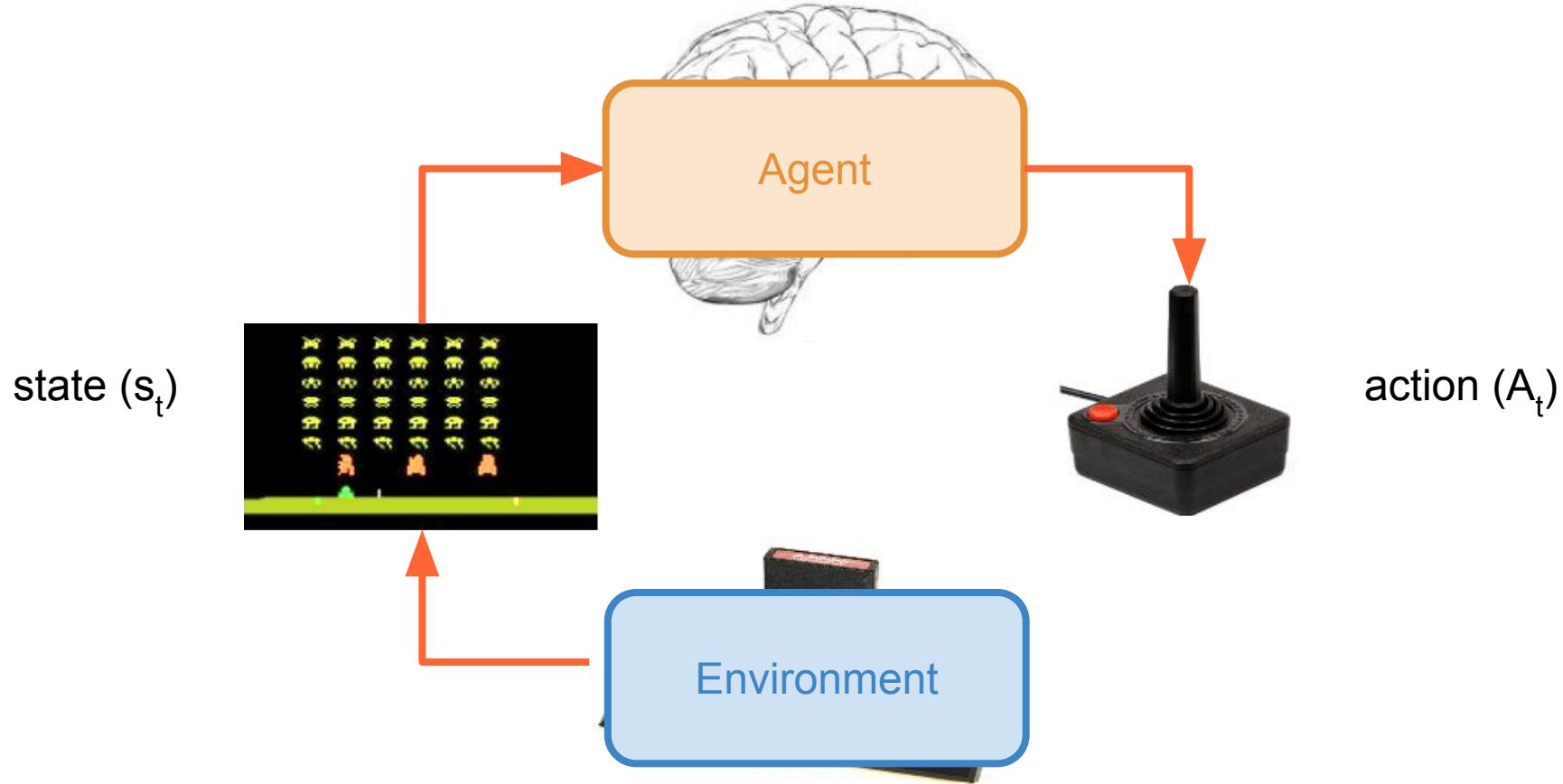
Architecture



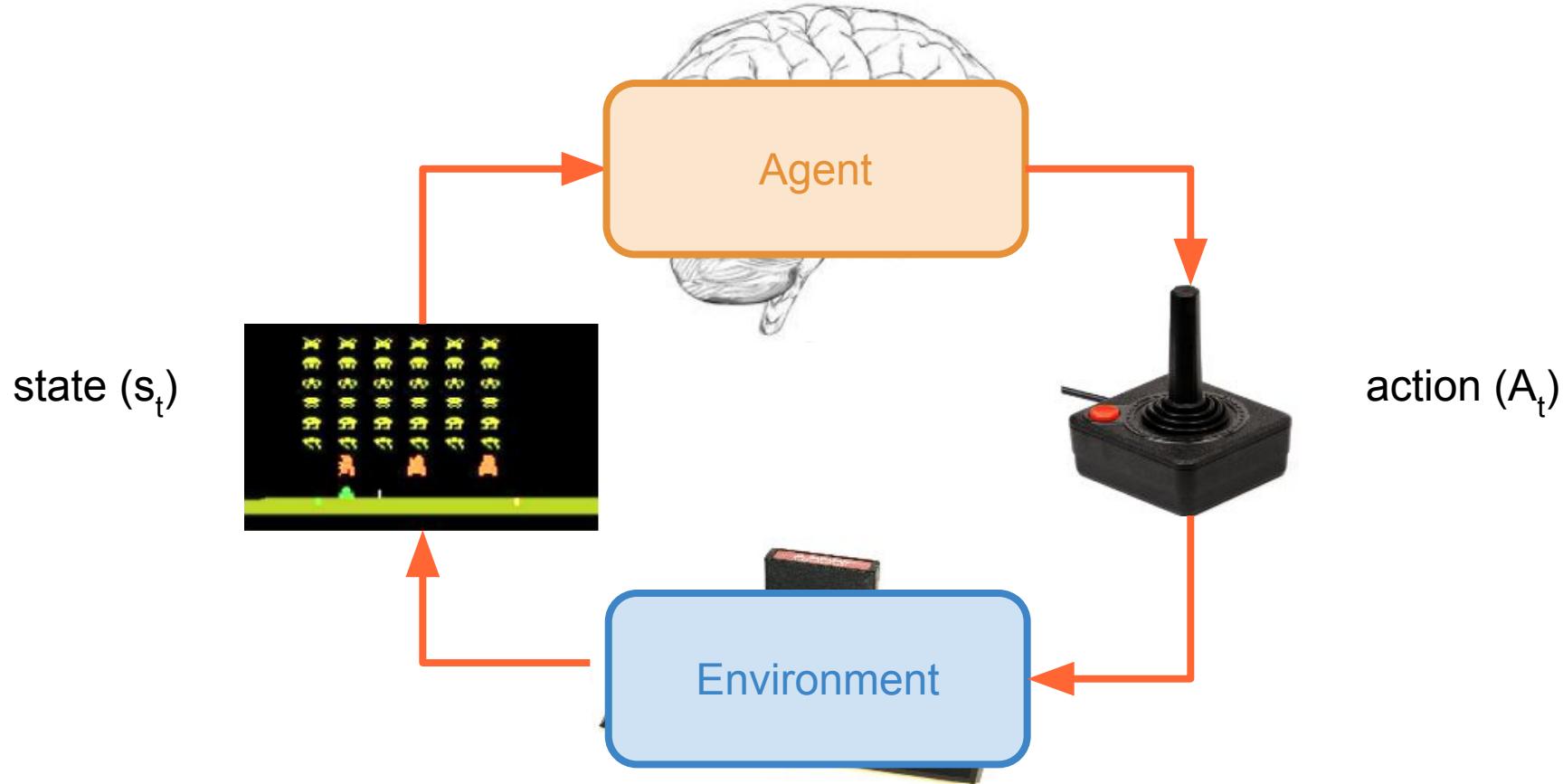
Architecture



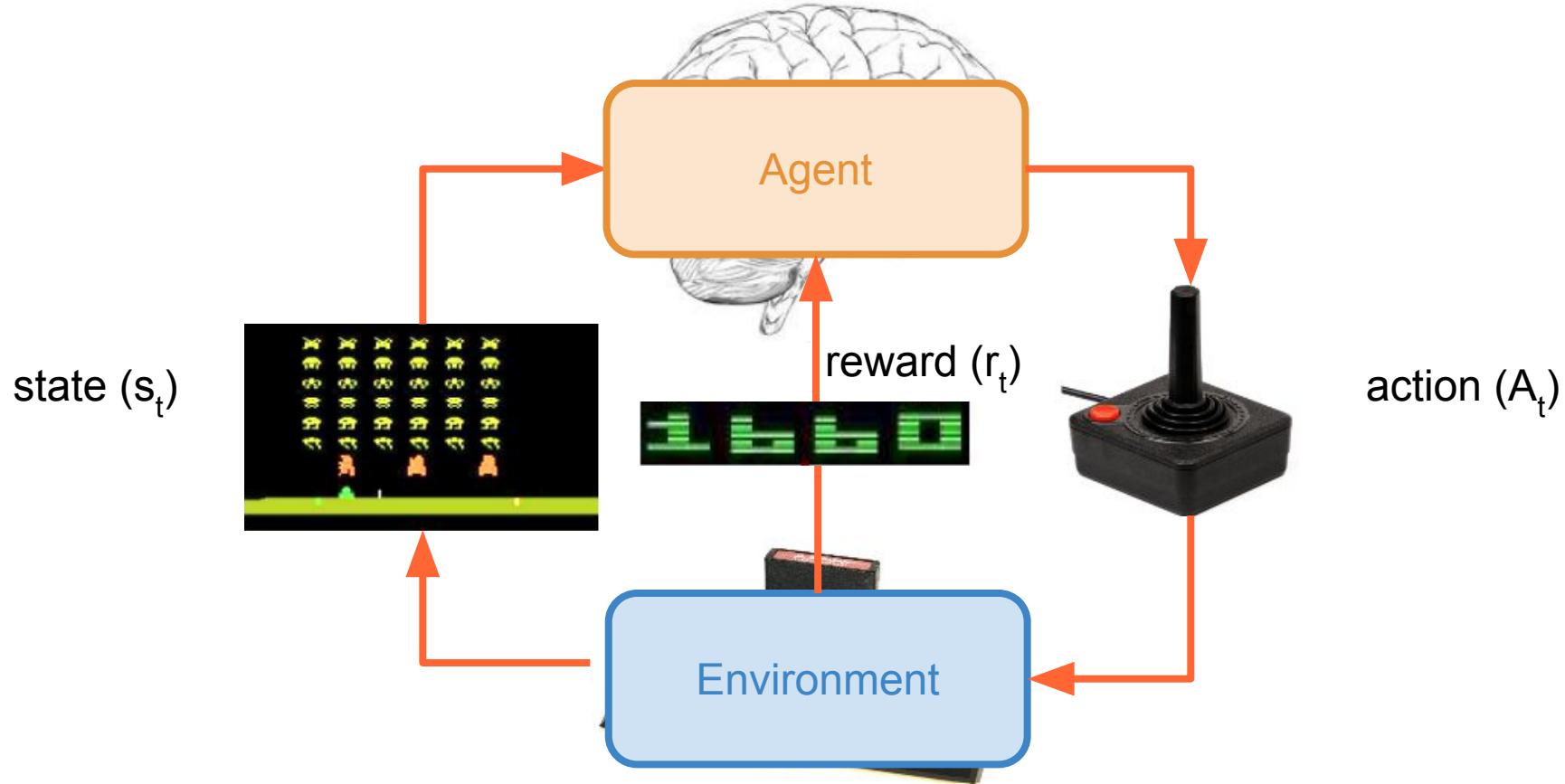
Architecture



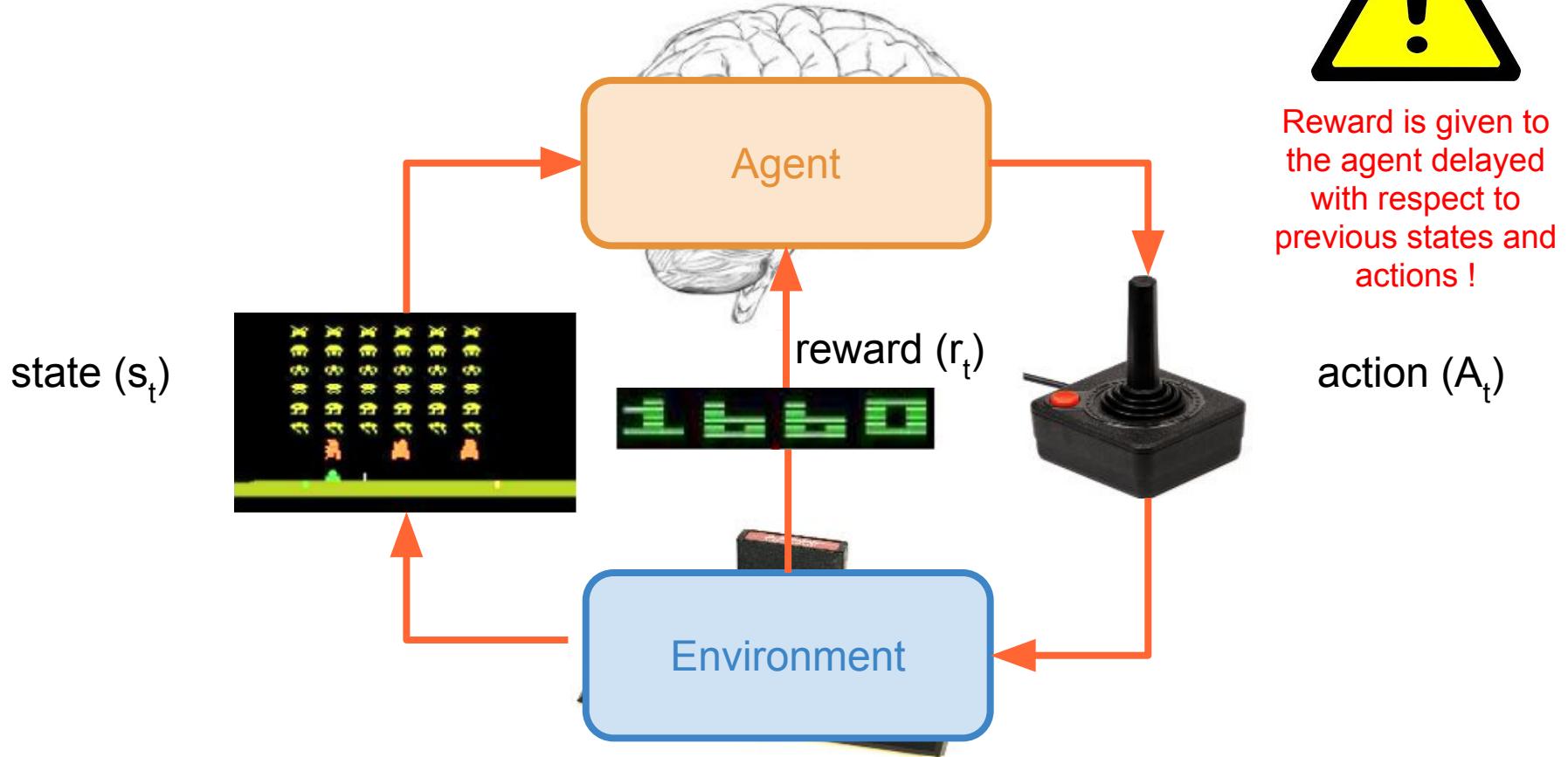
Architecture



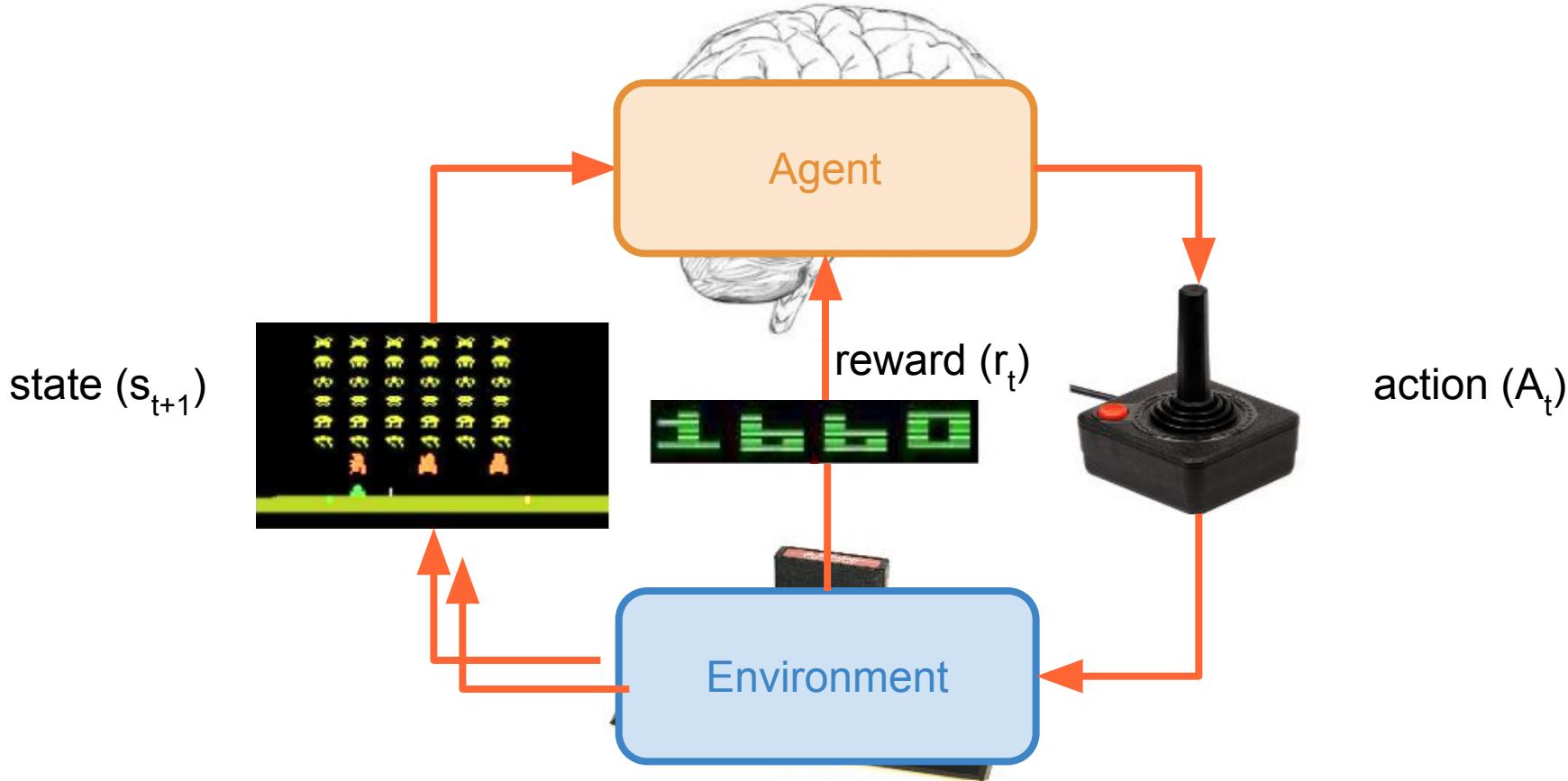
Architecture



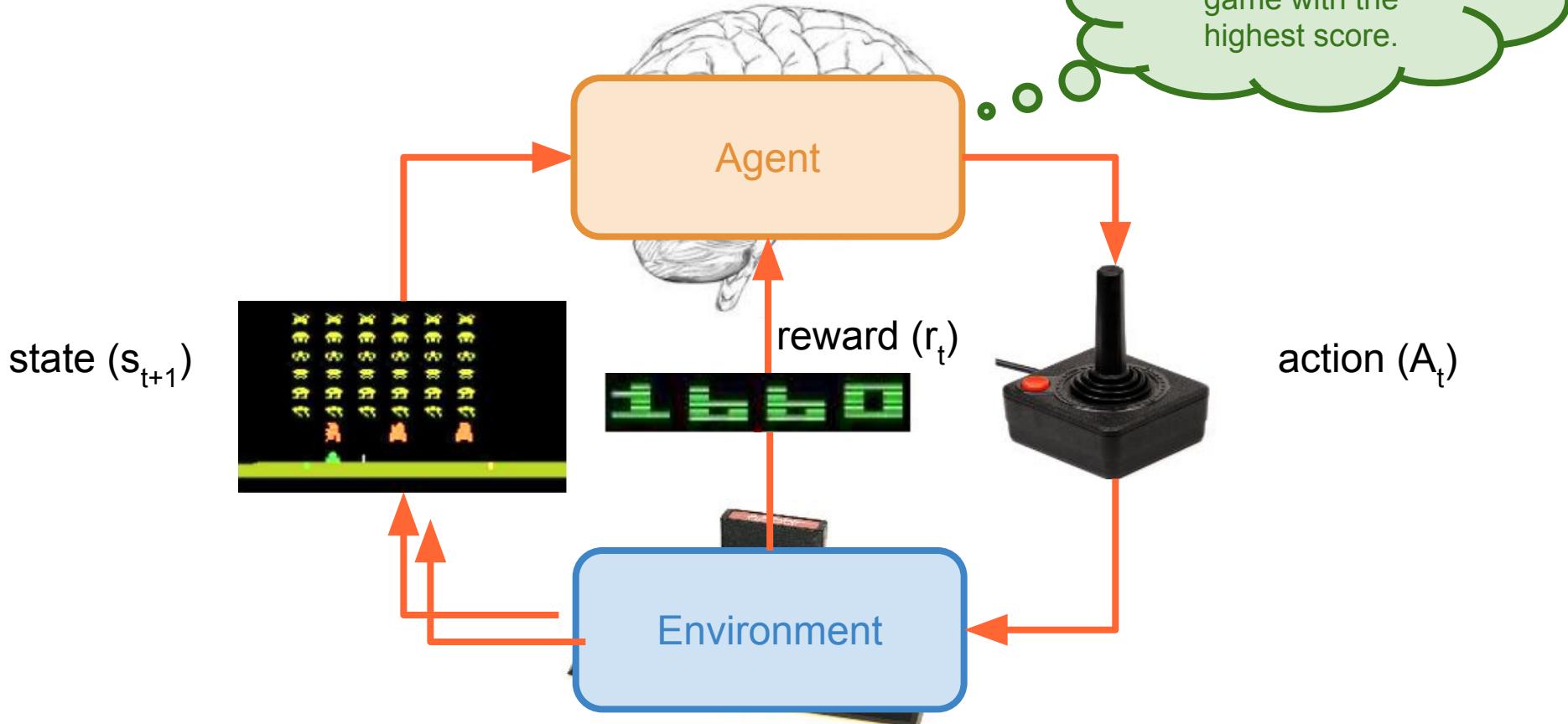
Architecture



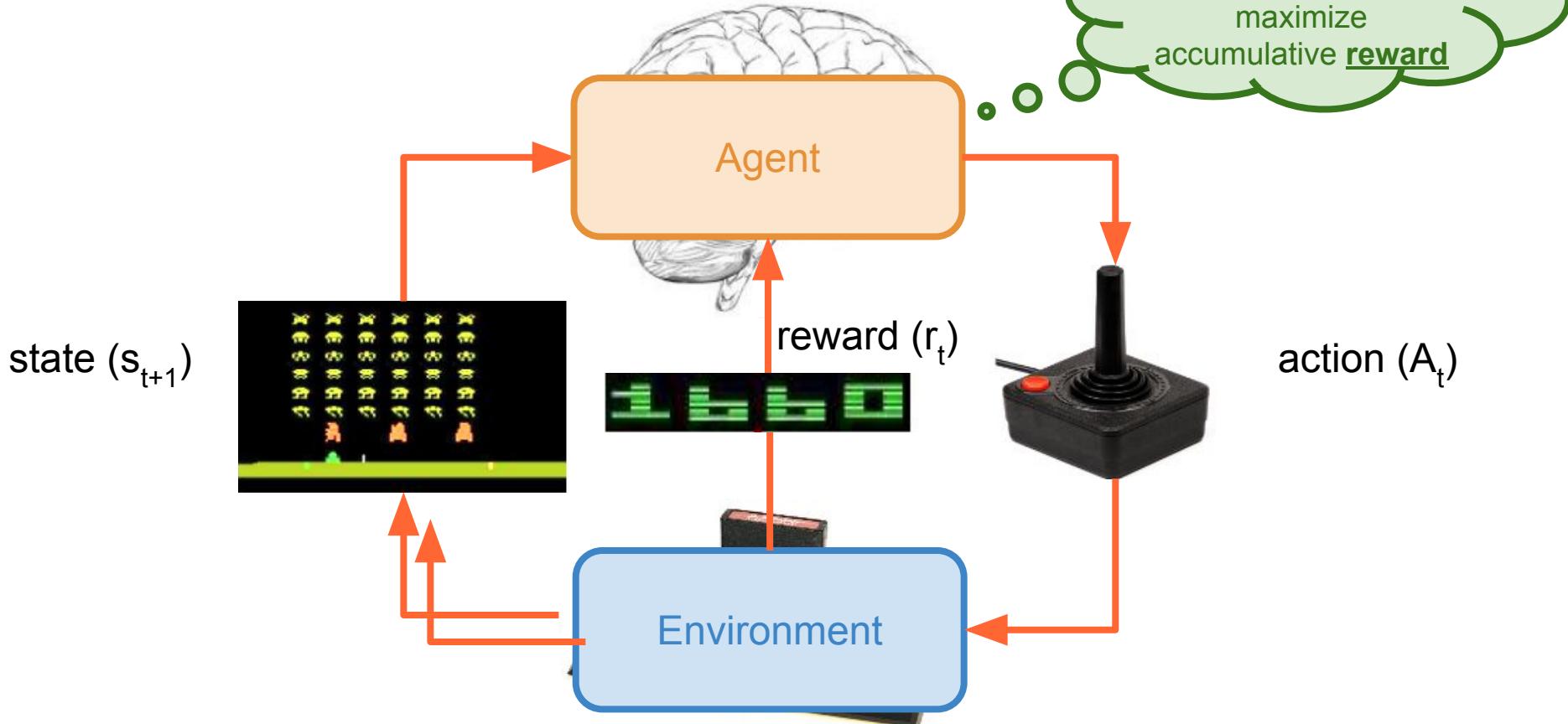
Architecture



Architecture



Architecture

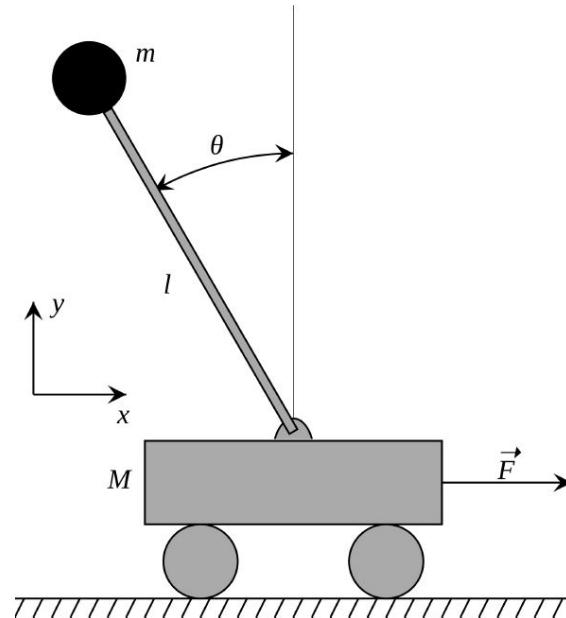


Architecture

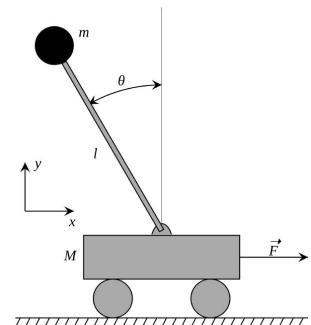
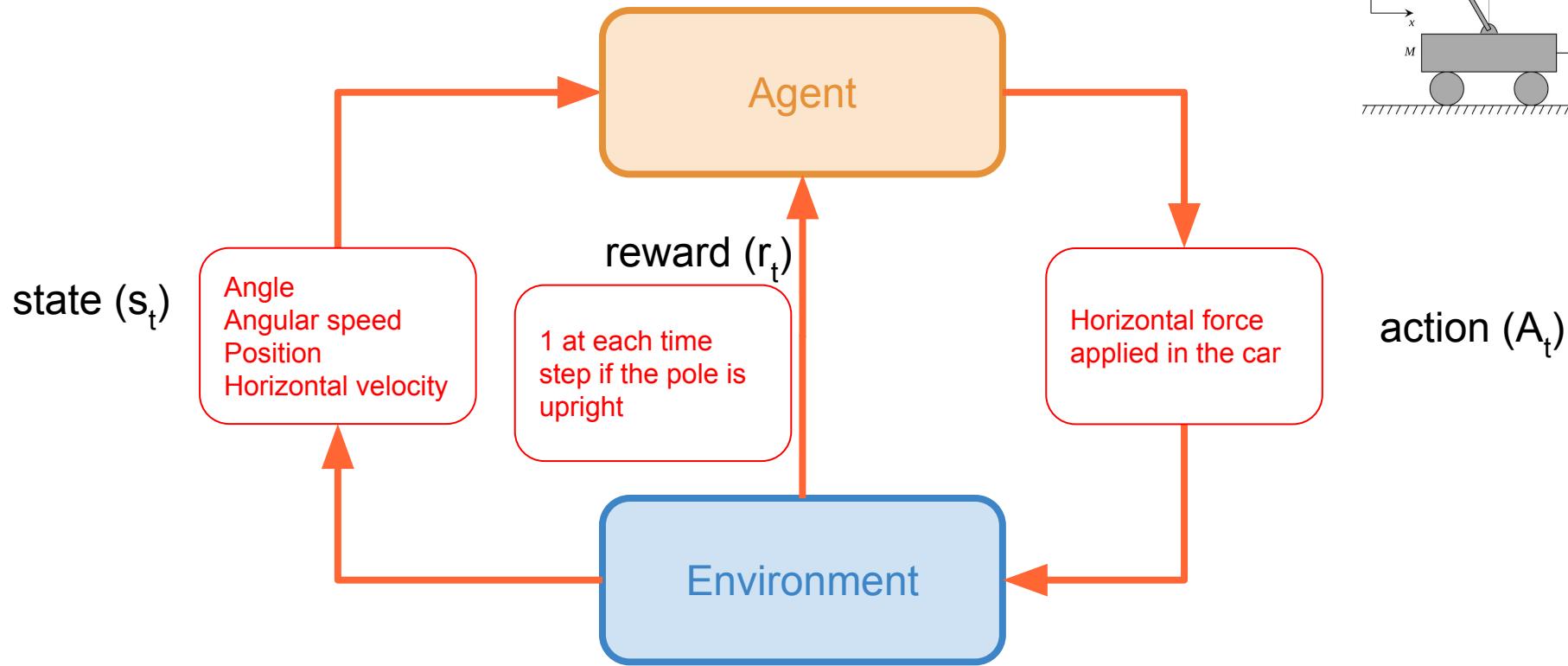
Other problems that can be formulated with a RL architecture.

Cart-Pole Problem

Objective: Balance a pole on top of a movable car



Architecture

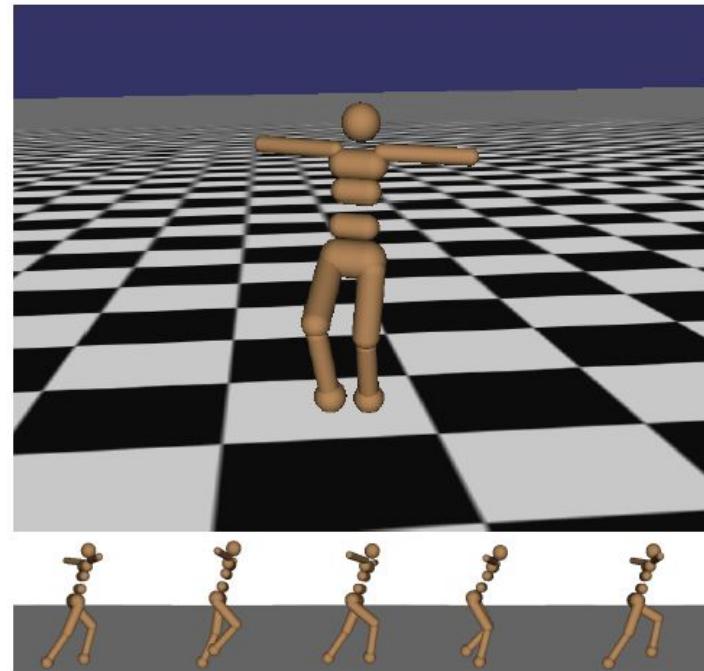


Architecture

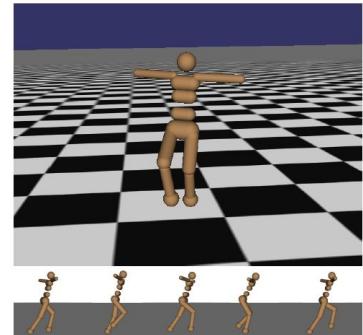
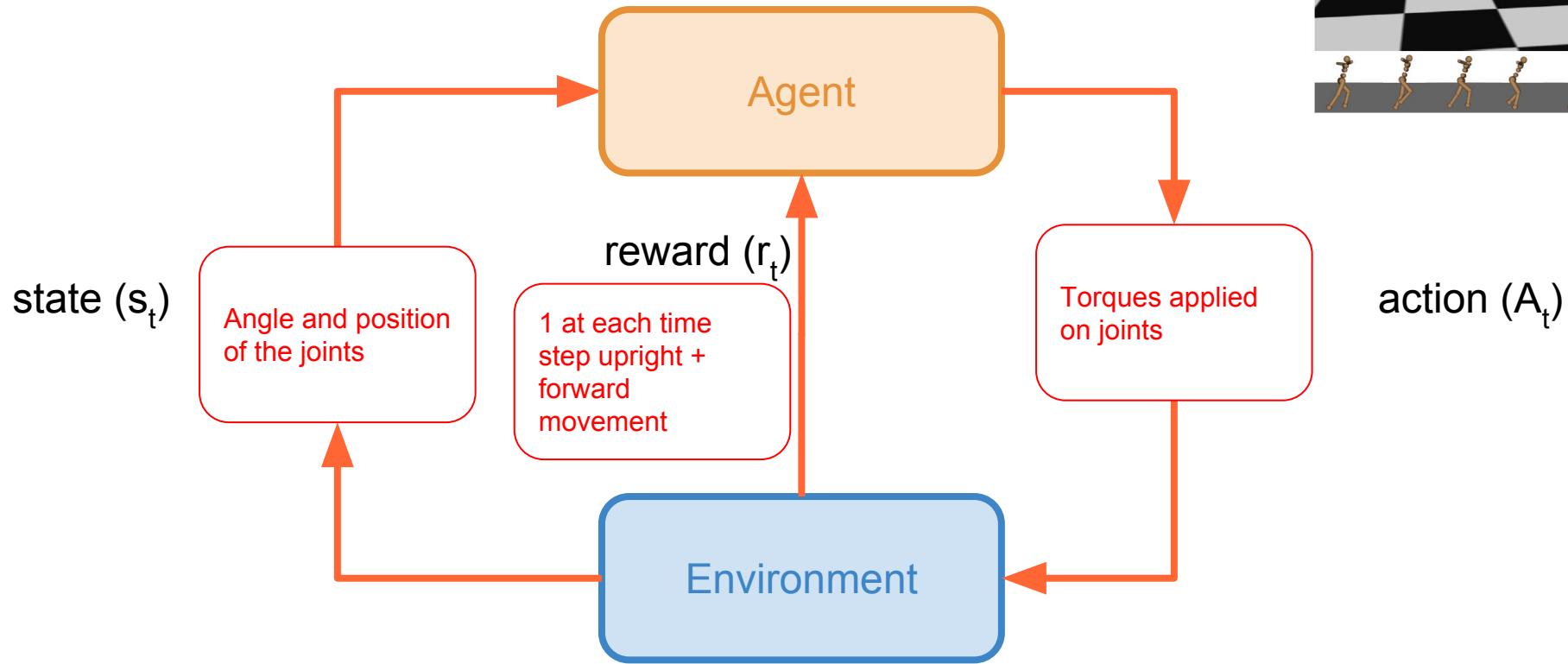
Other problems that can be formulated with a RL architecture.

Robot Locomotion

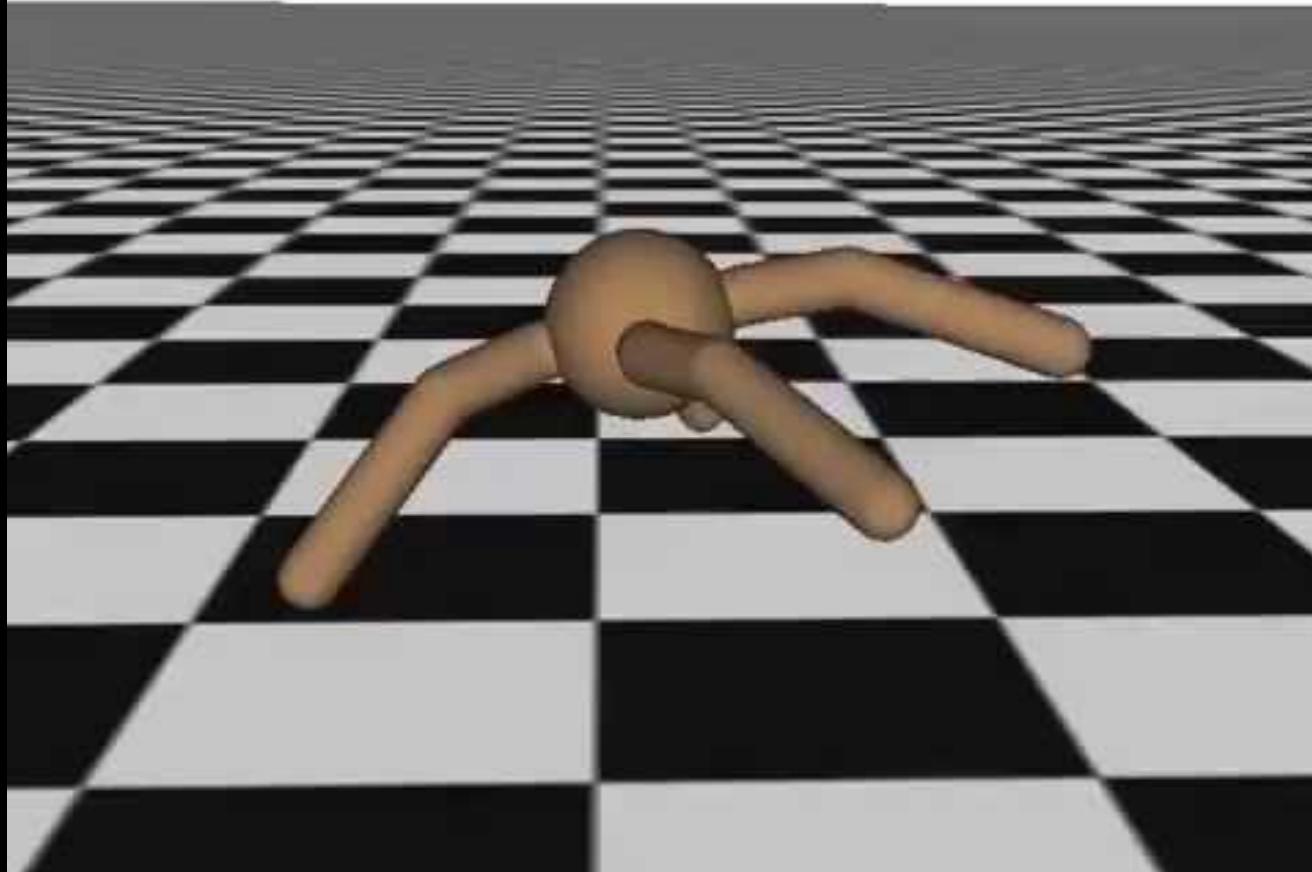
Objective: Make the robot move forward



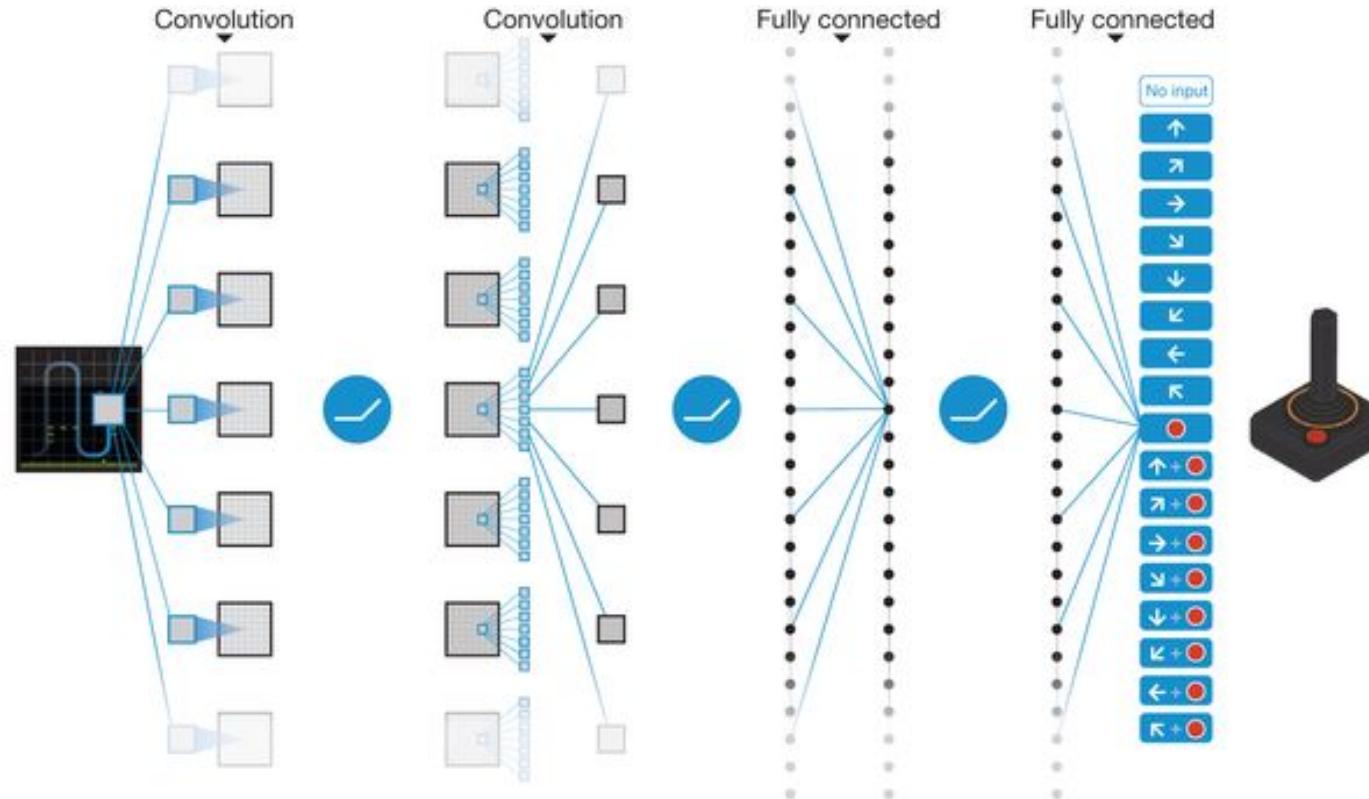
Architecture



Iteration 20

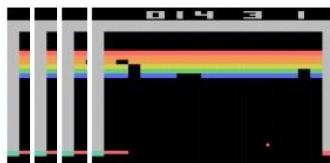
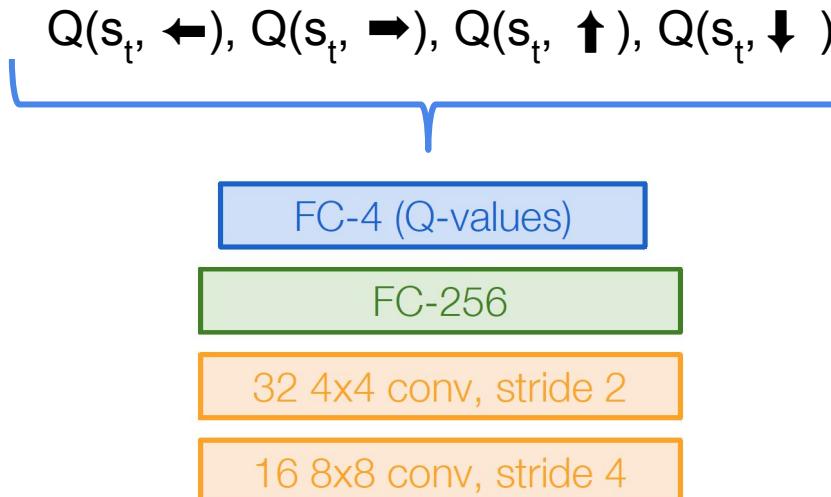


Deep Q-learning: Deep Q-Network DQN



Number of actions between 4-18, depending on the Atari game

Deep Q-learning: Deep Q-Network DQN

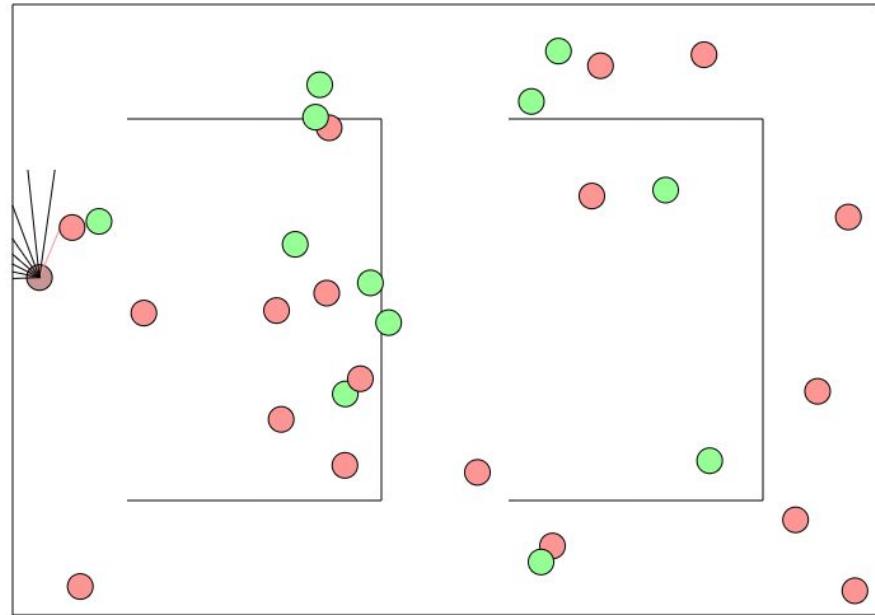


Current state s_t : 84x84x4 stack of last 4 frames
(after RGB->grayscale conversion, downsampling, and cropping)

Deep Q-learning: Demo

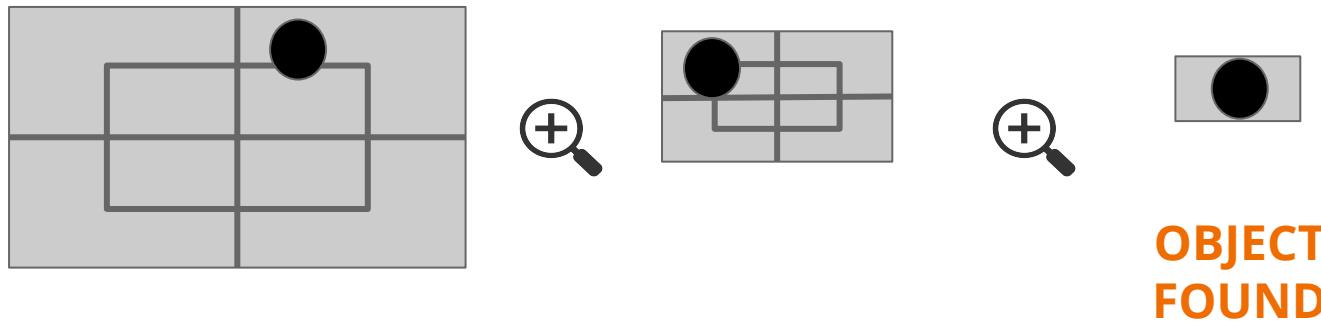


Deep Learning in your browser



Deep Q-learning: DQN: Computer Vision

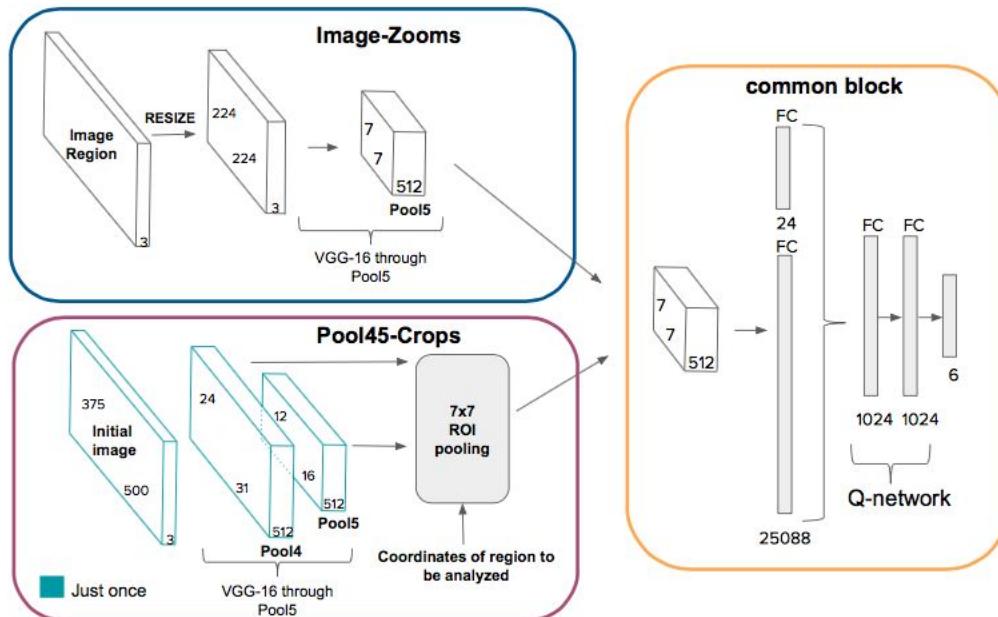
Method for performing hierarchical object detection in images guided by a **deep reinforcement learning agent**.



Deep Q-learning: DQN: Computer Vision

State: The agent will decide which action to choose based on:

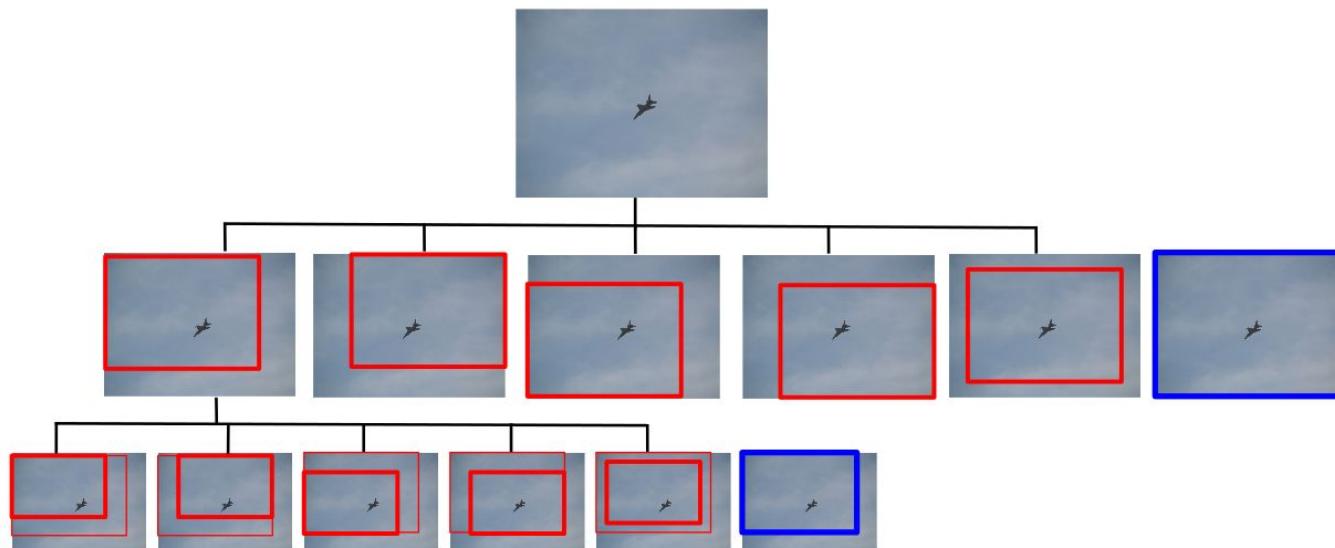
- **visual description** of the current observed region
- **history vector** that maps past actions performed



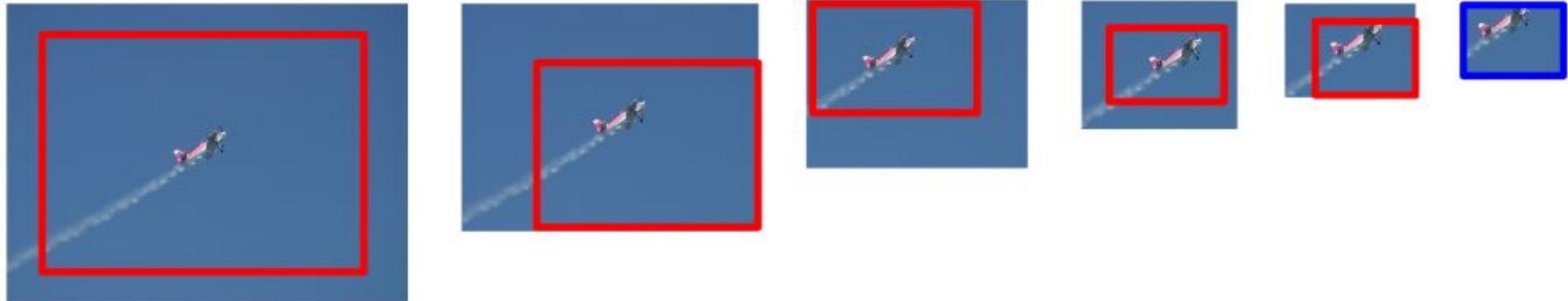
Deep Q-learning: DQN: Computer Vision

Actions: Two kind of actions:

- **movement actions**: to which of the 5 possible regions defined by the hierarchy to move
- **terminal action**: the agent indicates that the object has been found



Deep Q-learning: DQN: Computer Vision



[Miriam Bellver](#), Xavier Giro-i-Nieto, Ferran Marques, and Jordi Torres. "Hierarchical Object Detection with Deep Reinforcement Learning." Deep Reinforcement Learning Workshop NIPS 2016.

RL Frameworks

OpenAI Gym + keras-rl



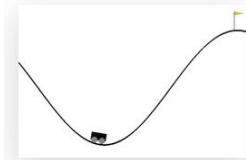
CartPole-v0
Balance a pole on a cart
(for a short time).



CartPole-v1
Balance a pole on a cart.



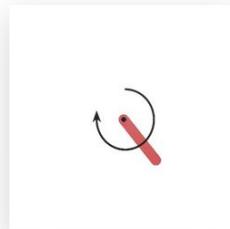
Acrobot-v1
Swing up a two-link
robot.



MountainCar-v0
Drive up a big hill.



MountainCarContinuous-v0
Drive up a big hill with
continuous control.



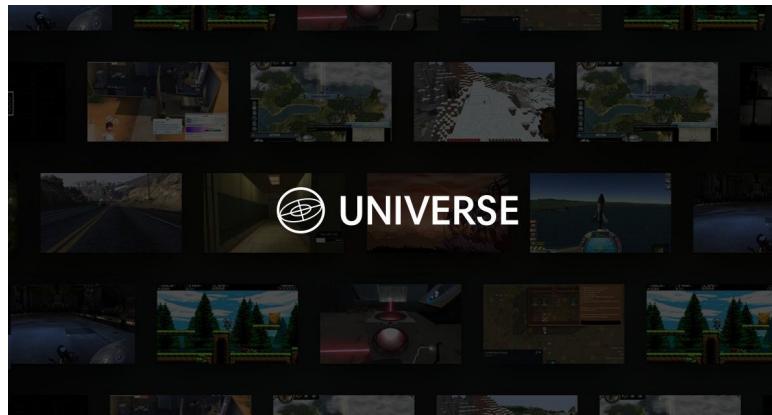
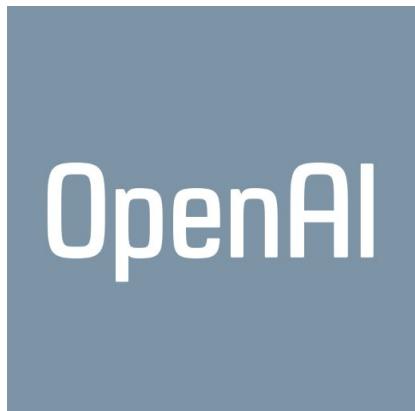
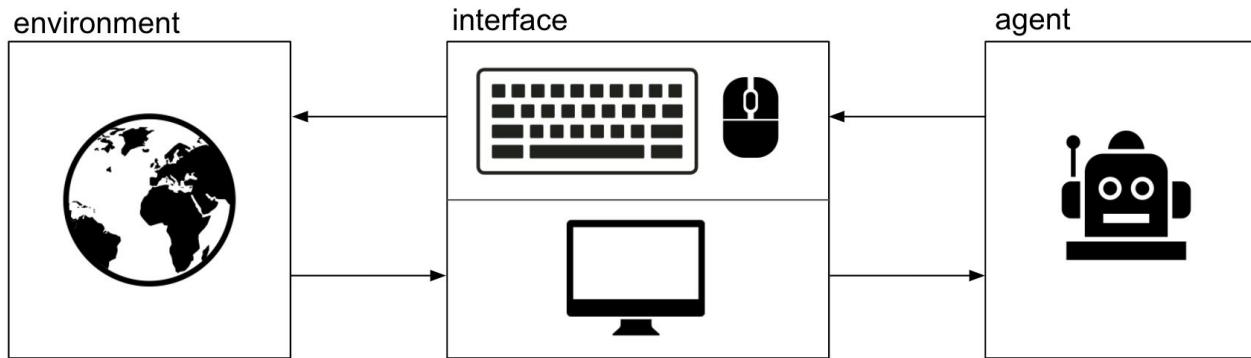
Pendulum-v0



keras-rl

keras-rl implements some state-of-the-art deep reinforcement learning algorithms in Python and seamlessly integrates with the deep learning library [Keras](#). Just like Keras, it works with either [Theano](#) or [TensorFlow](#), which means that you can train your algorithm efficiently either on CPU or GPU. Furthermore, keras-rl works with [OpenAI Gym](#) out of the box.

RL Frameworks

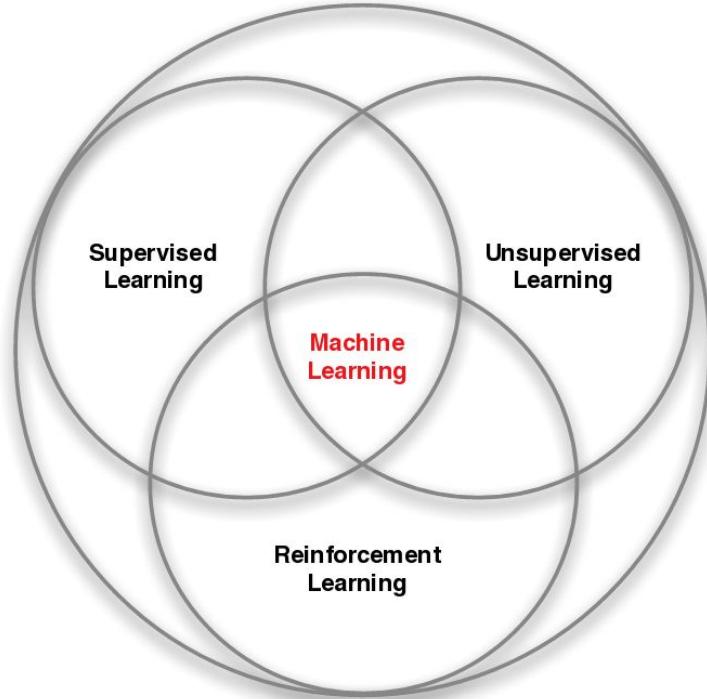


OpenAI
Universe
environment

Conclusions

Reinforcement Learning

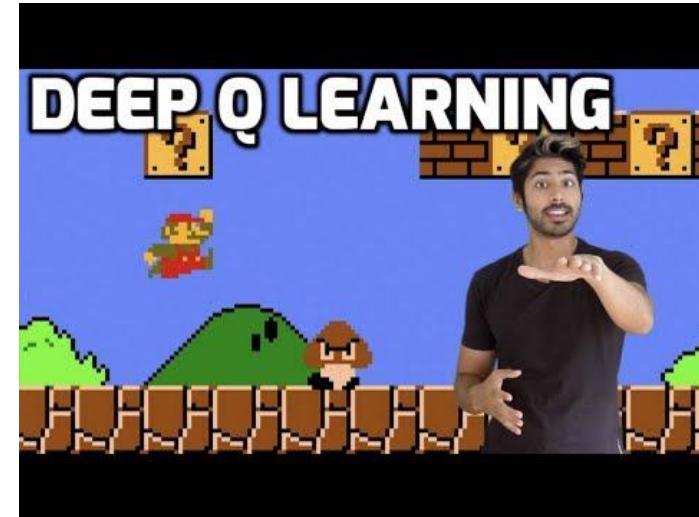
- There is no supervisor, only reward signal
- Feedback is delayed, not instantaneous
- Time really matters (sequential, non i.i.d data)



Learn more



Deep Learning TV,
[“Reinforcement learning - Ep. 30”](#)

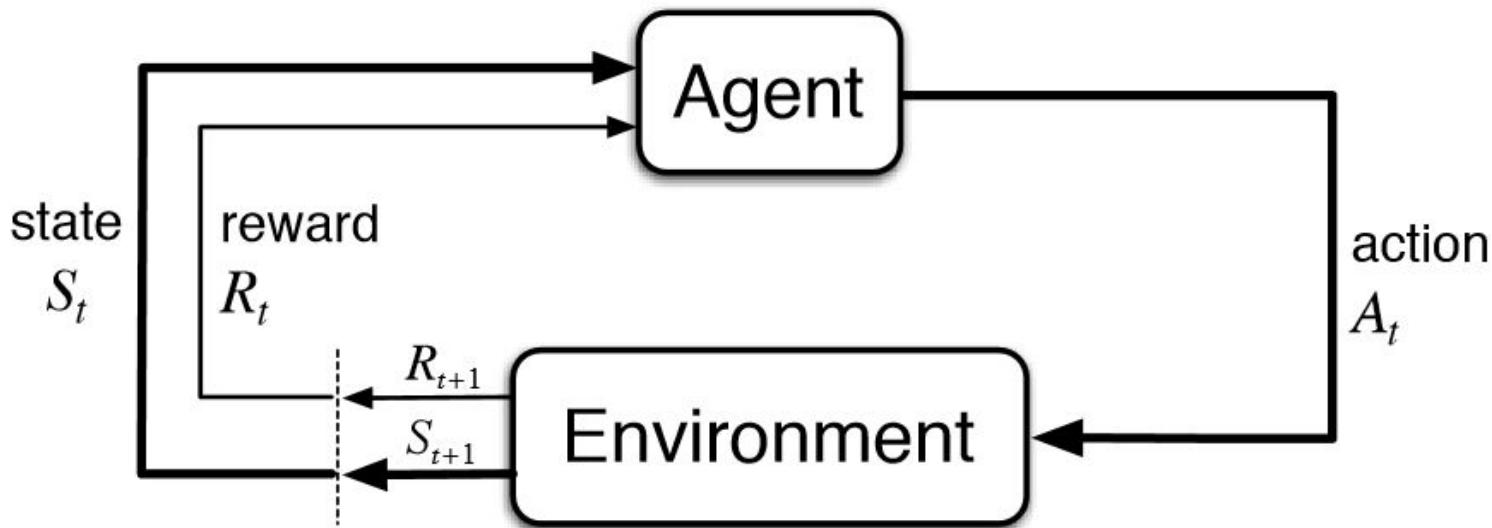


Siraj Raval,
Deep Q Learning for Video Games

Learn more



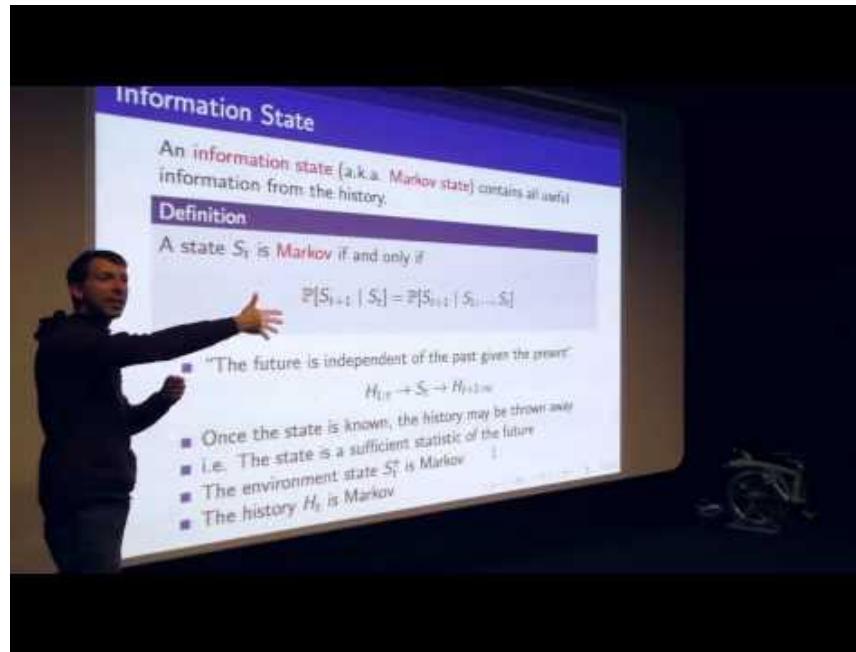
Emma Brunskill, [Stanford CS234: Reinforcement Learning](#)



Learn more



David Silver, UCL COMP050, [Reinforcement Learning](#)



Learn more



Nando de Freitas, [“Machine Learning”](#) (University of Oxford)



Attention-Based Game Agent

- Roughly the same model and training method can be used in a game-playing agent.
- The agent learns to track a ball without being told to do so.

The diagram illustrates an attention-based game agent. It shows a green rectangle labeled "Glimpse Coverage" containing a blue shape. To its right is a "Glimpse History" section showing a sequence of small frames, and a "Game Play" section showing a larger frame with a red border. Below these is a "Individual Glimpses" section showing three small frames. To the right of the diagram is a presentation slide with the University of Oxford logo, the text "DEPARTMENT OF COMPUTER SCIENCE", and the date "12 March 2015". A video frame of Nando de Freitas speaking is overlaid on the top right of the slide.

Learn more

Pieter Abbeel and John Schulman, [CS 294-112 Deep Reinforcement Learning](#), Berkeley.

Slides: [“Reinforcement Learning - Policy Optimization” OpenAI / UC Berkeley \(2017\)](#)



Motivation

We can categorize three types of learning procedures:

1. Supervised Learning:

$$\mathbf{y} = f(\mathbf{x})$$

2. Unsupervised Learning:

$$f(\mathbf{x})$$

3. Reinforcement Learning (RL):

$$\mathbf{y} = f(\mathbf{x})$$

$$\mathbf{z}$$



Acknowledgments



The slide title is "The manifold hypothesis". It states: "The data distribution lie close to a low-dimensional manifold". An example is given: "Consider image data". The list includes:

- Very high dimensional (1,000,000D)
- A randomly generated image will almost certainly not look like any real world scene
 - The space of images that occur in nature is almost completely empty
- Hypothesis: real world images lie on a smooth, low-dimensional manifold
 - Manhattan distance is a great measure of similarity

Similar for audio and text

On the right side of the slide, there are three images: a stack of four grey rectangles, a collage of three images showing a bridge, a person, and a motorcycle, and a small image of a presentation slide with a grid pattern.

Below the slide, a video frame shows a person standing at a podium, presenting. At the bottom right is the UPC logo and text: "UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH Departament de Teoria del Sinyal i Comunicacions".

[Kevin McGuinness](#), “Unsupervised Learning” Deep Learning for Computer Vision.
[Slides 2016] [Slides 2017]

Motivation



Yann LeCun

Monday at 10:15 · Edited ·

Statement from a Slashdot post about the AlphaGo victory: "We know now that we don't need any big new breakthroughs to get to true AI"

That is completely, utterly, ridiculously wrong.

As I've said in previous statements: most of human and animal learning is unsupervised learning. If intelligence was a cake, unsupervised learning would be the cake, supervised learning would be the icing on the cake, and reinforcement learning would be the cherry on the cake. We know how to make the icing and the cherry, but we don't know how to make the cake.

We need to solve the unsupervised learning problem before we can even think of getting to true AI. And that's just an obstacle we know about. What about all the ones we don't know about?



Greff, Klaus, Antti Rasmus, Mathias Berglund, Tele Hao, Harri Valpola, and Juergen Schmidhuber. "[Tagger: Deep unsupervised perceptual grouping](#)." NIPS 2016 [[video](#)] [[code](#)]

Unsupervised Learning

Why Unsupervised Learning?

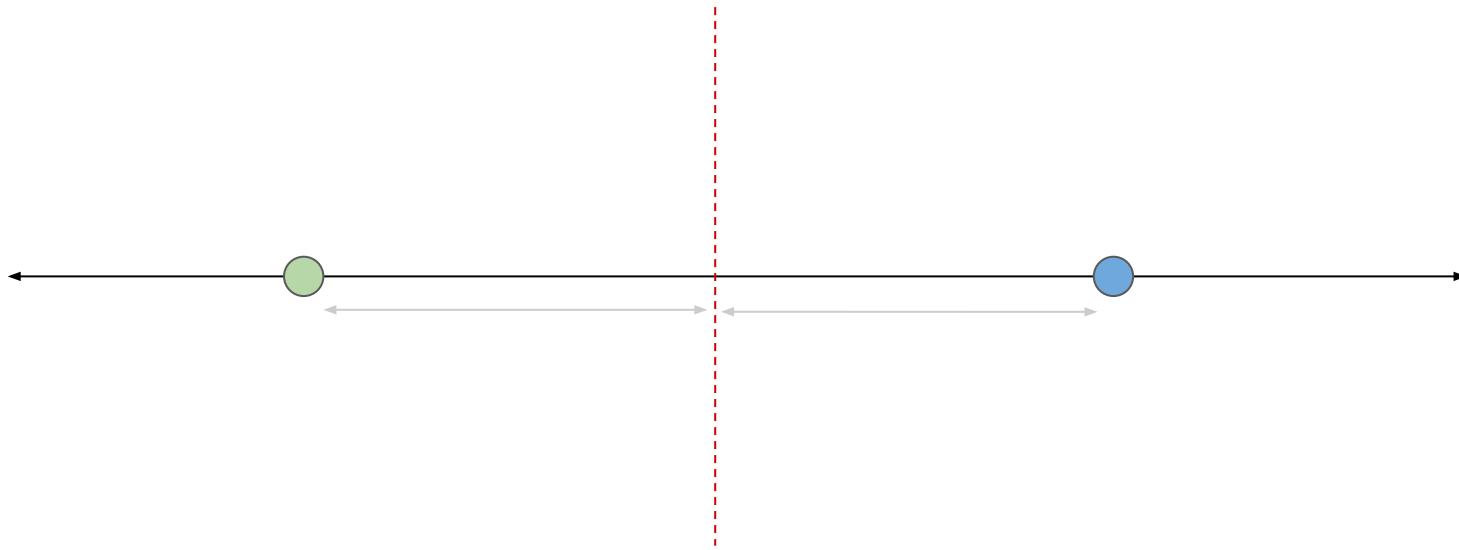
- It is the nature of how intelligent beings percept the world.
- It can save us tons of efforts to build a human-alike intelligent agent compared to a totally supervised fashion.
- Vast amounts of unlabelled data.

WHY?

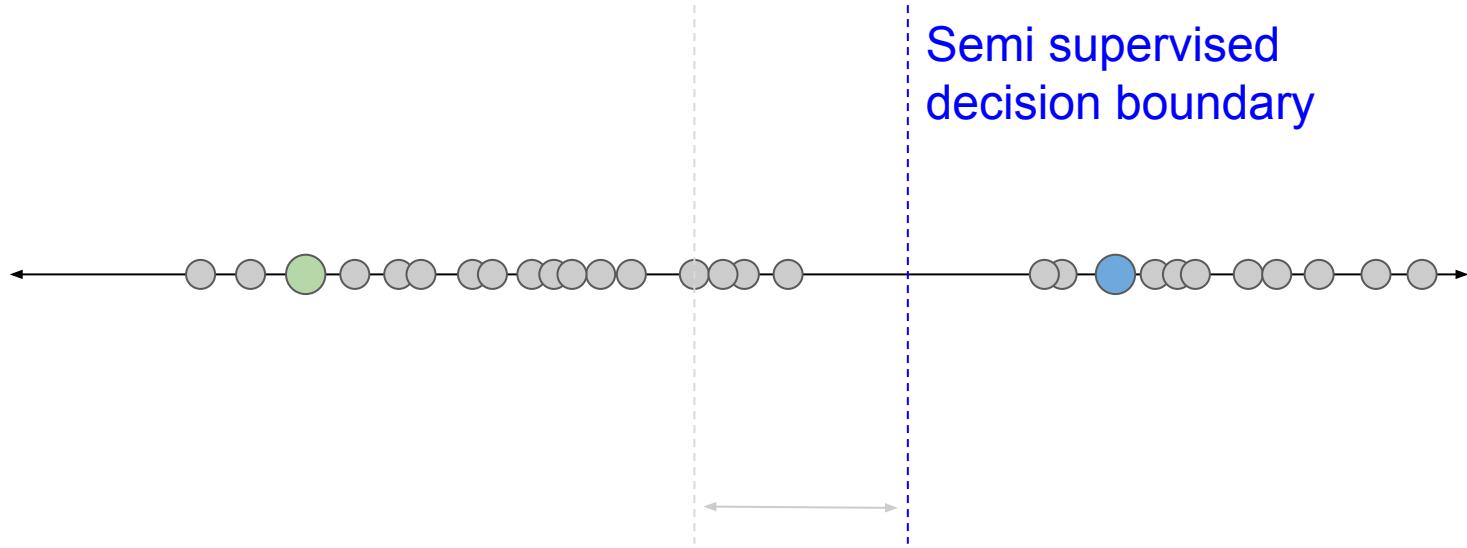


How data distribution $P(x)$ influences decisions (1D)

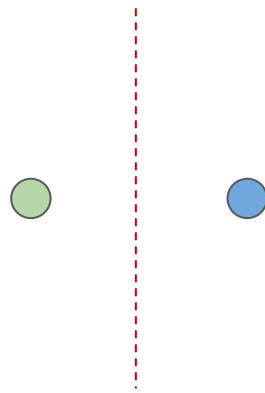
Max margin decision boundary



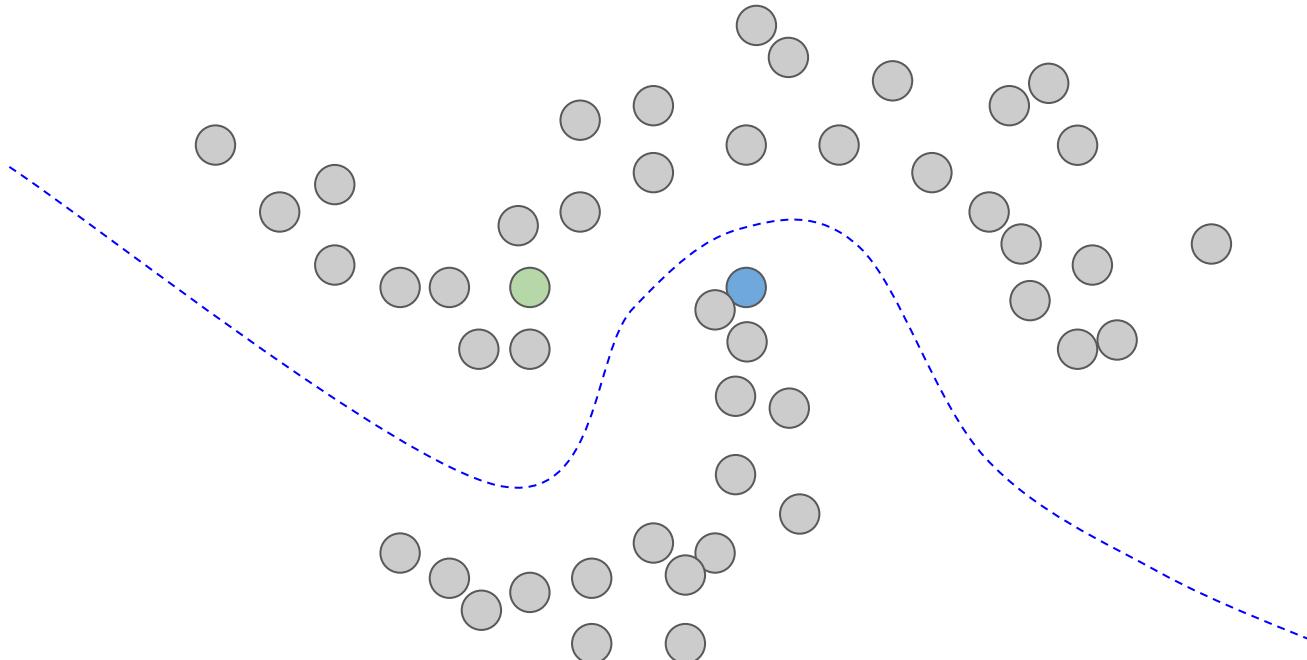
How data distribution $P(x)$ influences decisions (1D)



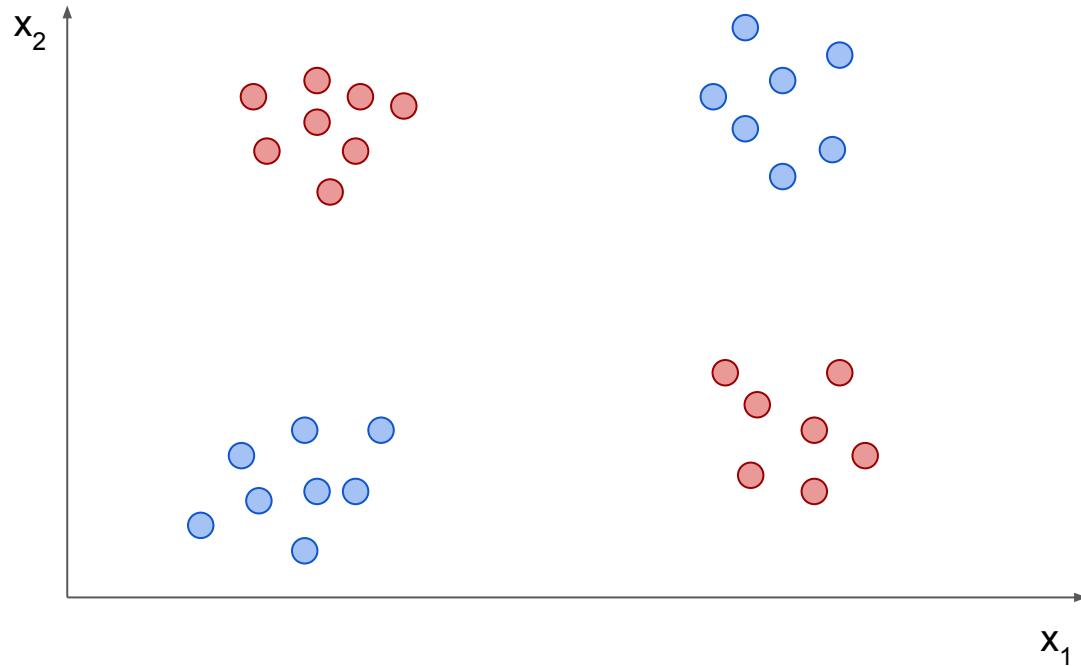
How data distribution $P(x)$ influences decisions (2D)



How data distribution $P(x)$ influences decisions (2D)

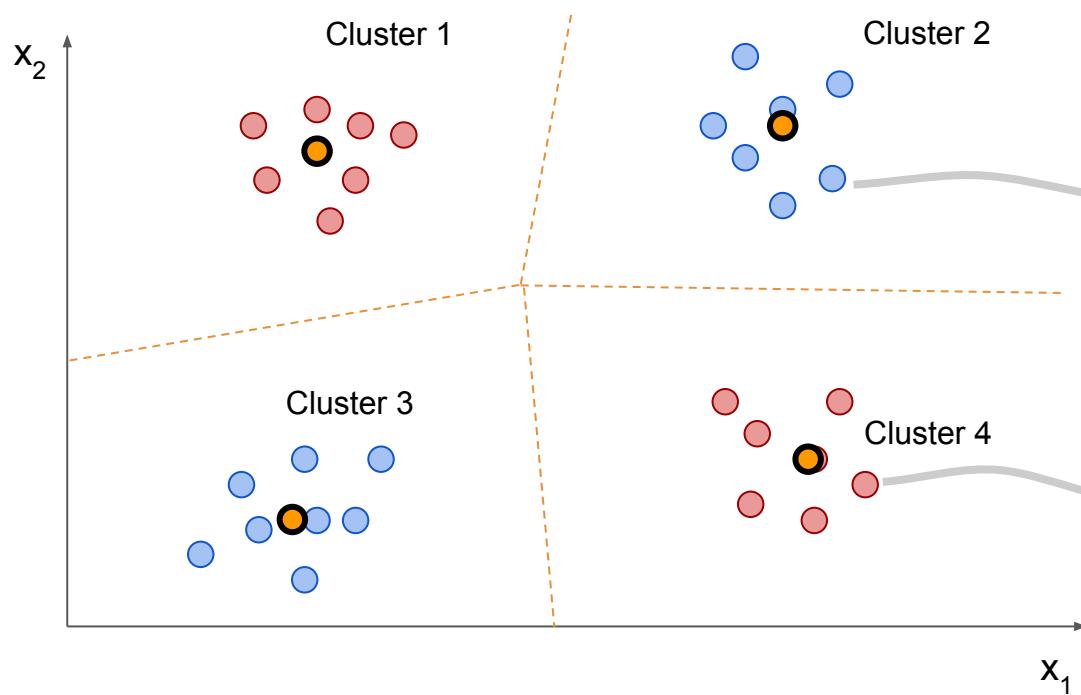


How clustering is valuable for linear classifiers



Red / Blue classes are nt
linearly separable in this
2D space :(

How clustering is valuable for linear classifiers

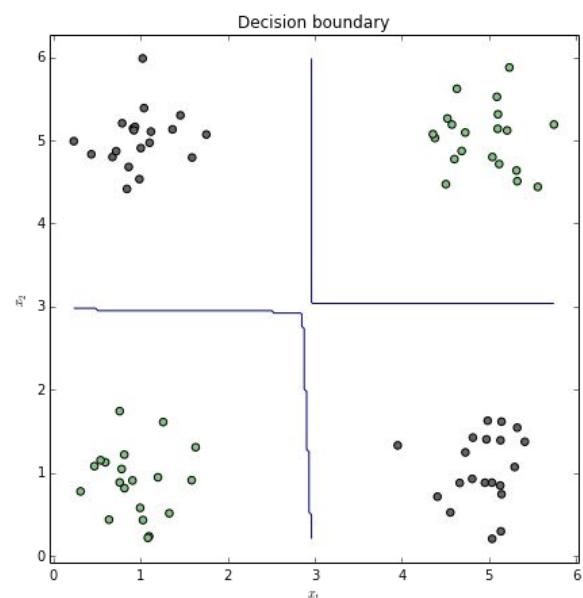
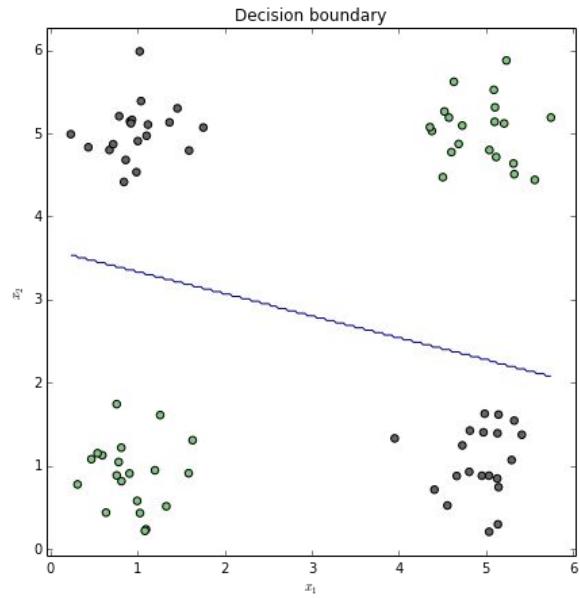
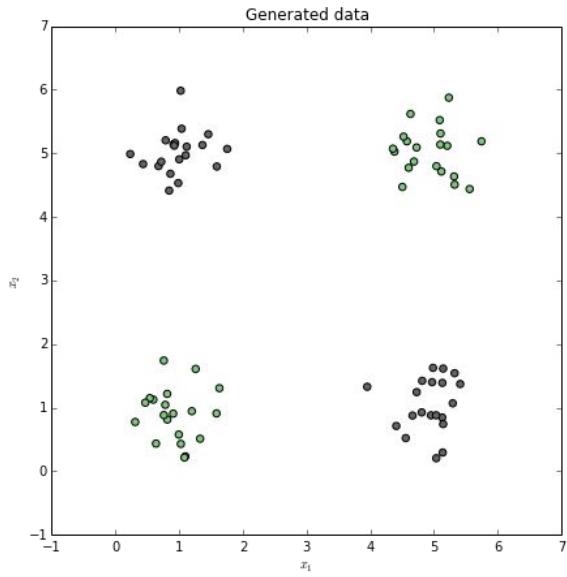


4D BoW representation
("histogram")



Red / Blue classes are now separable with a linear classifiers in a 4D space :)

How clustering is valuable for linear classifiers



Assumptions for unsupervised learning

To model $P(X)$ given data, it is necessary to make some assumptions

“You can’t do inference without making assumptions”

-- David MacKay, Information Theory, Inference, and Learning Algorithms

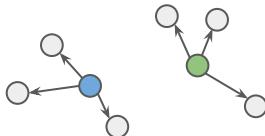
Typical assumptions:

- Smoothness assumption
 - Points which are close to each other are more likely to share a label.
- Cluster assumption
 - The data form discrete clusters; points in the same cluster are likely to share a label
- **Manifold assumption**
 - The data lie approximately on a manifold of much lower dimension than the input space.

Assumptions for unsupervised learning

Smoothness assumption

- Label propagation
 - Recursively propagate labels to nearby points
 - Problem: in high-D, your nearest neighbour may be very far away!

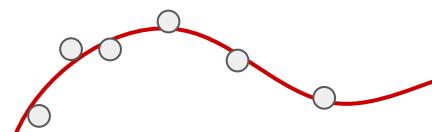


Cluster assumption

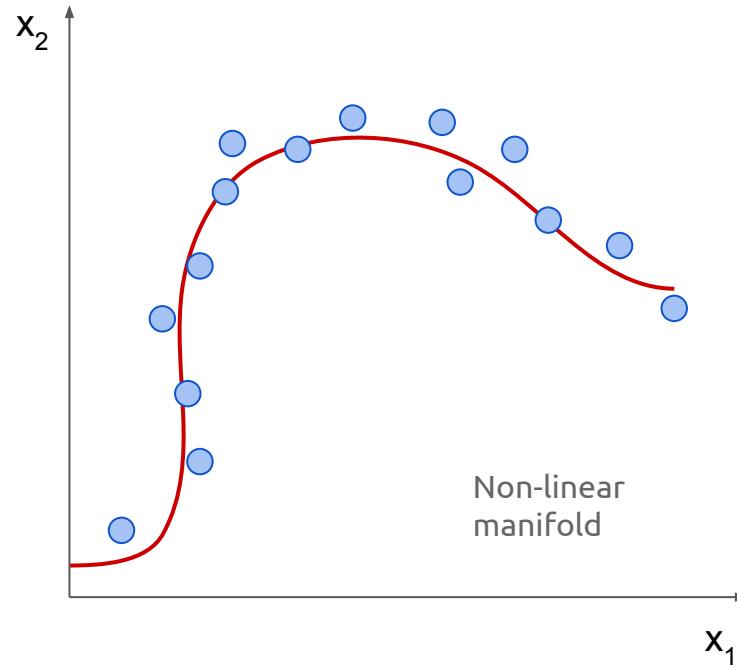
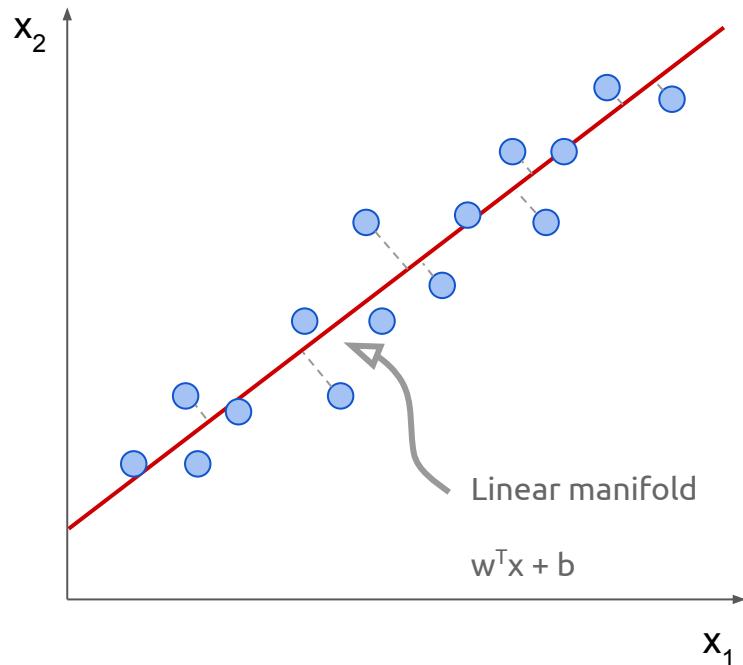
- Bag of words models
 - K-means, etc.
 - Represent points by cluster centers
 - Soft assignment
 - VLAD
- Gaussian mixture models
 - Fisher vectors

Manifold assumption

- Linear manifolds
 - PCA
 - Linear autoencoders
 - Random projections
 - ICA
- Non-linear manifolds:
 - Non-linear autoencoders
 - Deep autoencoders
 - Restricted Boltzmann machines
 - Deep belief nets



The manifold hypothesis



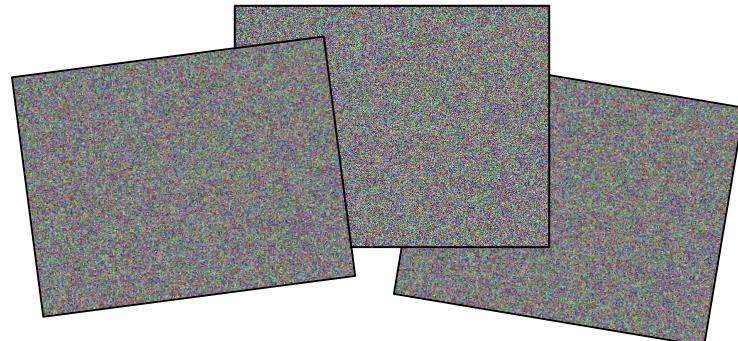
The manifold hypothesis

The data distribution lies close to a low-dimensional manifold

Example: **consider image data**

- Very high dimensional (1,000,000D)
- A randomly generated image will almost certainly not look like any real world scene
 - The space of images that occur in nature is almost completely empty
- Hypothesis: real world images lie on a smooth, low-dimensional manifold
 - Manifold distance is a good measure of similarity

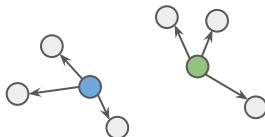
Similar for audio and text



Assumptions for unsupervised learning

Smoothness assumption

- Label propagation
 - Recursively propagate labels to nearby points
 - Problem: in high-D, your nearest neighbour may be very far away!

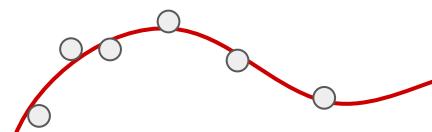


Cluster assumption

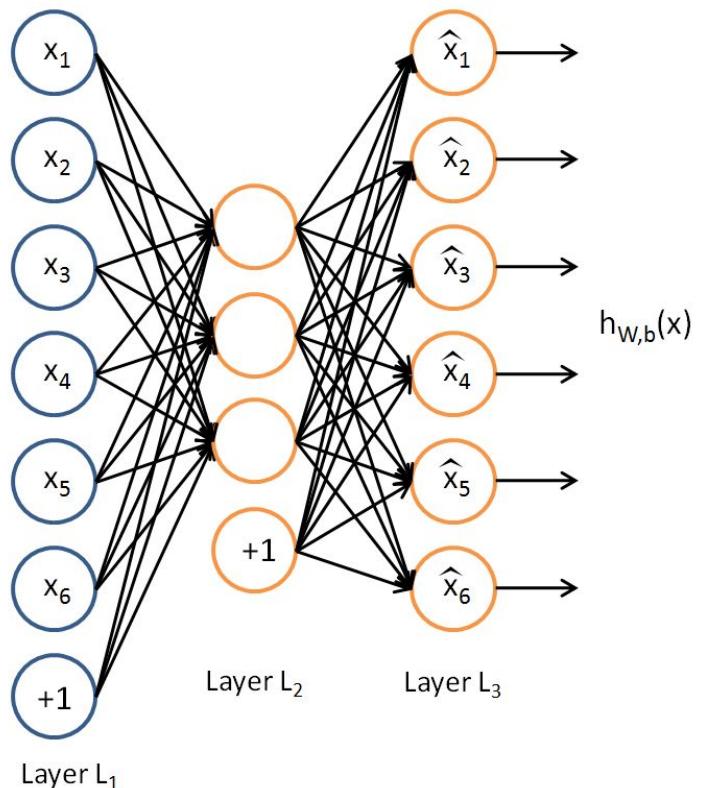
- Bag of words models
 - K-means, etc.
 - Represent points by cluster centers
 - Soft assignment
 - VLAD
- Gaussian mixture models
 - Fisher vectors

Manifold assumption

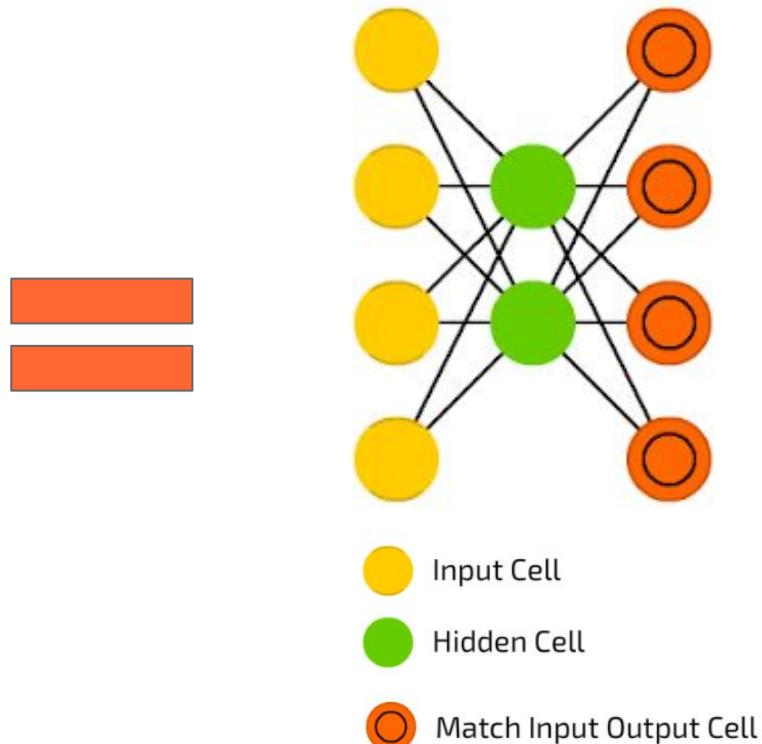
- Linear manifolds
 - PCA
 - Linear autoencoders
 - Random projections
 - ICA
- Non-linear manifolds:
 - Non-linear autoencoders
 - Deep autoencoders
 - Restricted Boltzmann machines
 - Deep belief nets



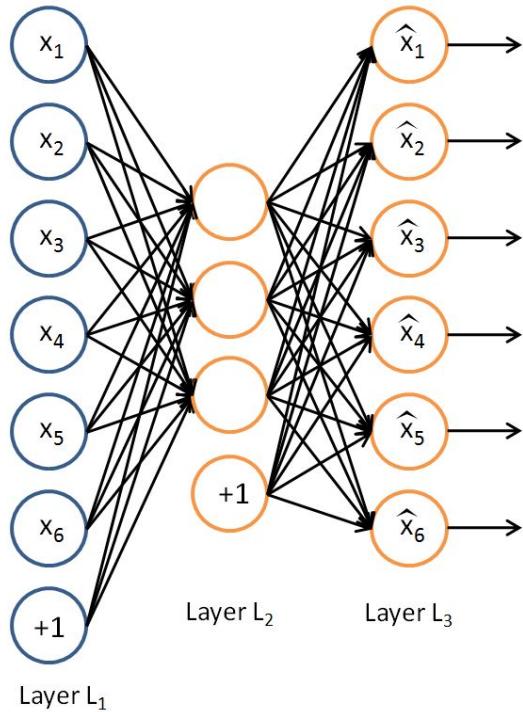
Autoencoder (AE)



Auto Encoder (AE)



Autoencoder (AE)

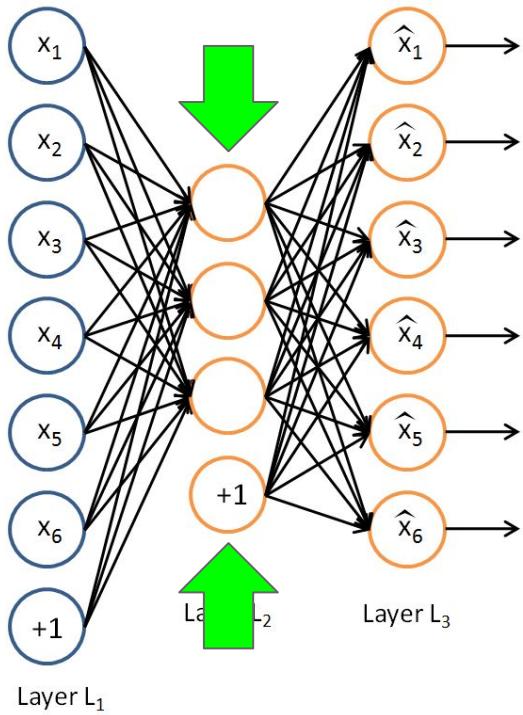


Autoencoders:

- Predict at the output the same input data.
- Do not need labels:

Autoencoder (AE)

WHY?



Application #1

Dimensionality reduction:

- Use hidden layer as a feature extractor of any desired size.

Autoencoder (AE)

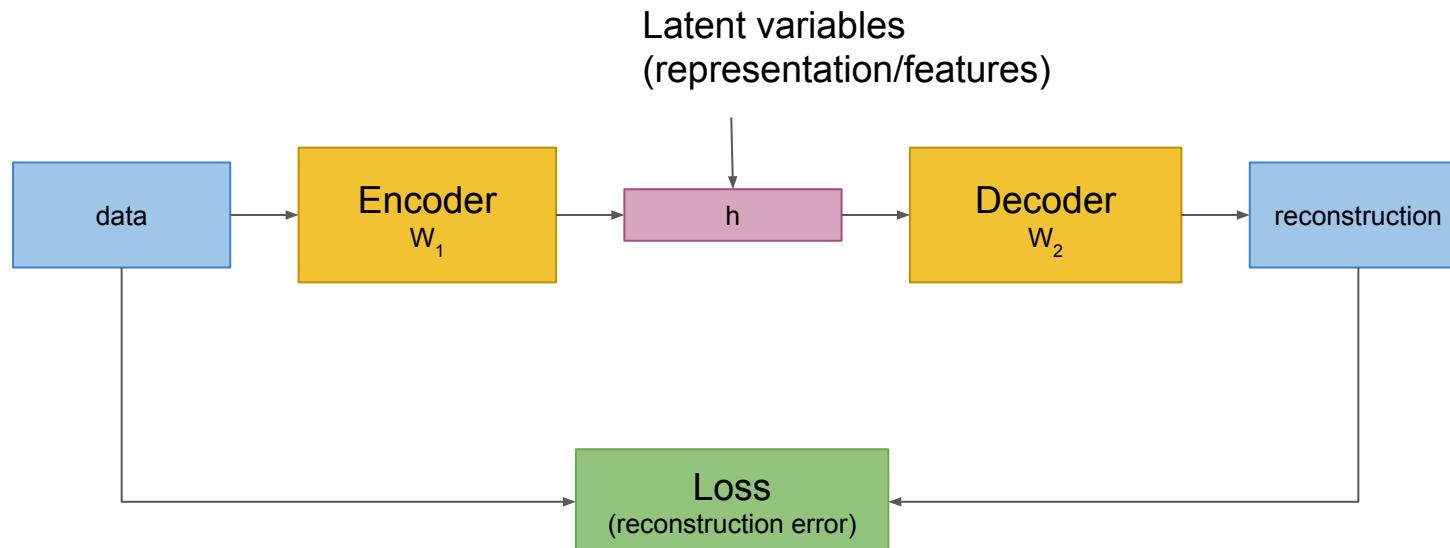
Application #2

WHY?



Pretraining:

1. Initialize a NN solving an autoencoding problem.



Autoencoder (AE)

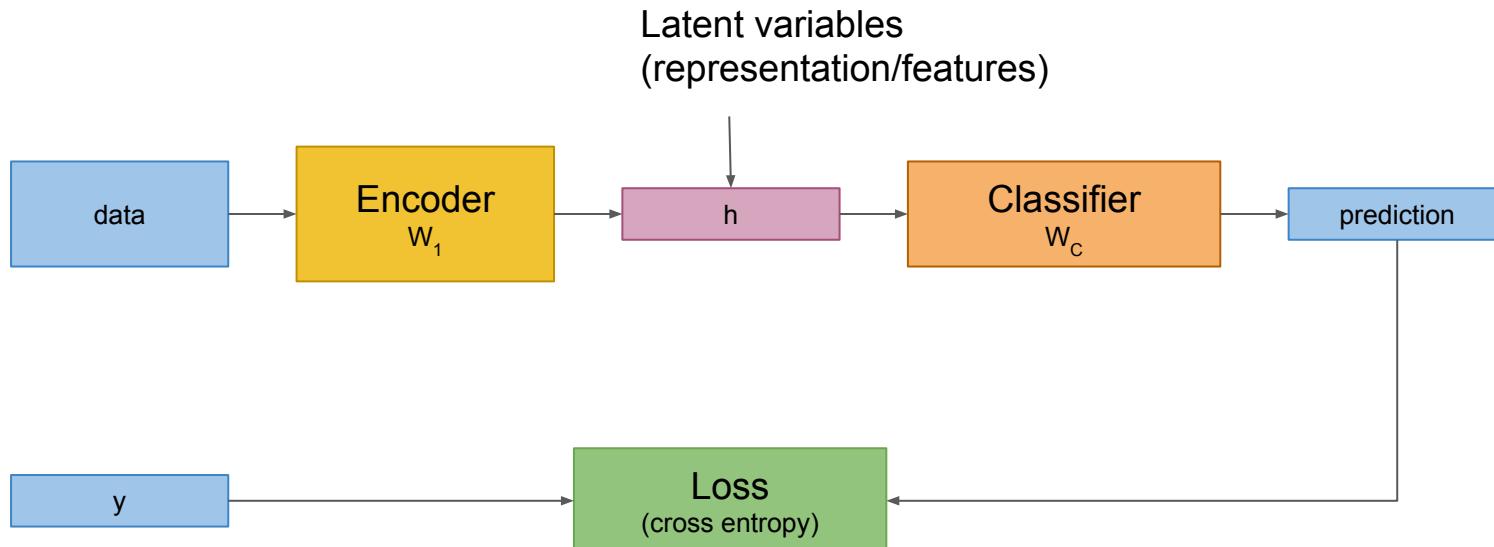
Application #2

WHY?

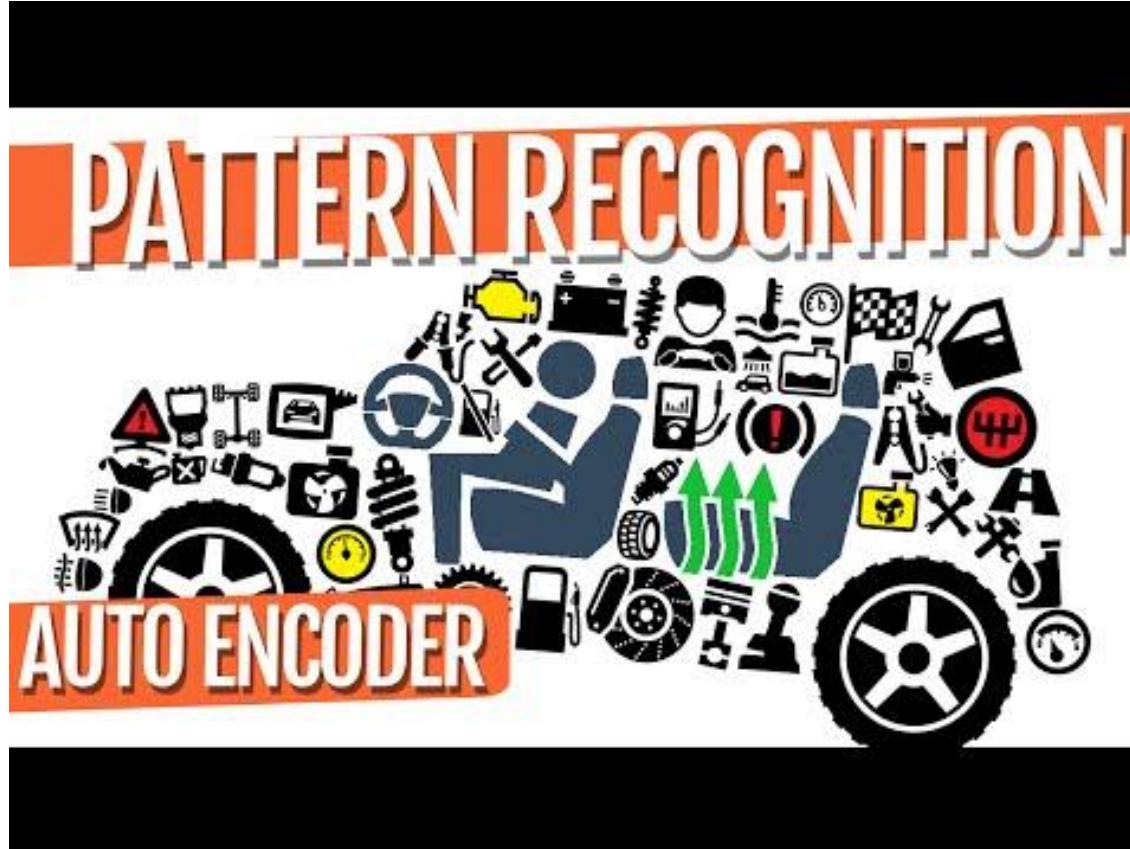


Pretraining:

1. Initialize a NN solving an autoencoding problem.
2. Train for final task with “few” labels.

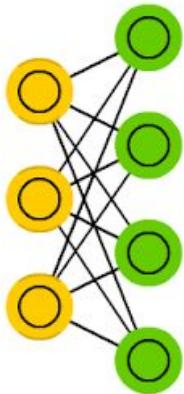


Autoencoder (AE)



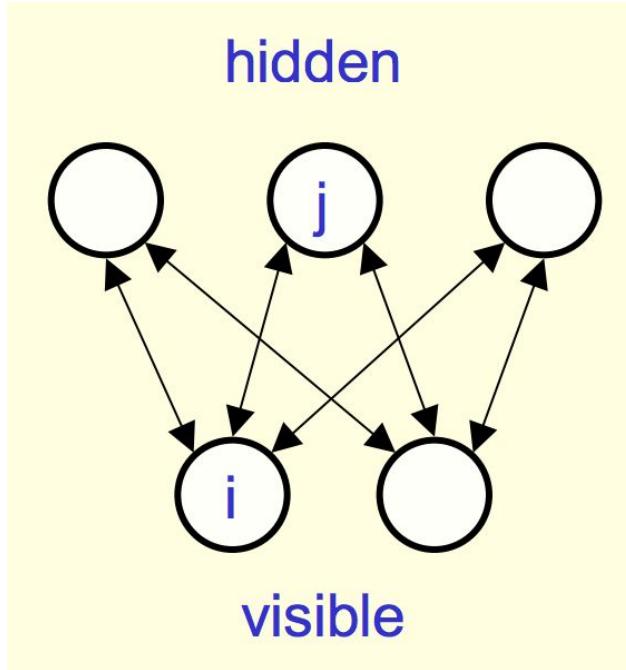
Restricted Boltzmann Machine (RBM)

Restricted BM (RBM)



- Backfed Input Cell
- Probabilistic Hidden Cell
- Hidden Cell
- Match Input Output Cell

Restricted Boltzmann Machine (RBM)

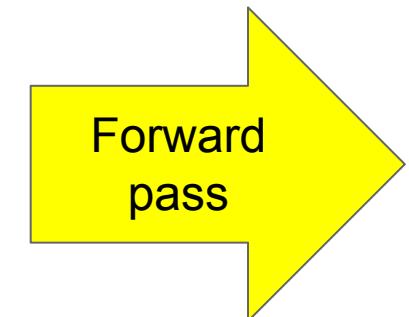
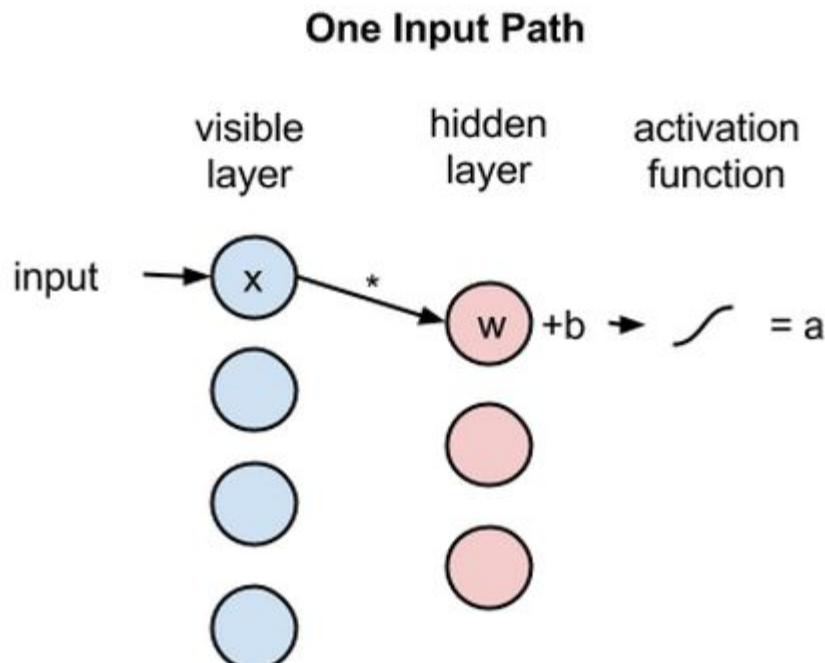


- Shallow two-layer net.
- Restricted = No two nodes in a layer share a connection
- Bipartite graph.
- Bidirectional graph
 - Shared weights.
 - Different biases.

Figure: Geoffrey Hinton (2013)

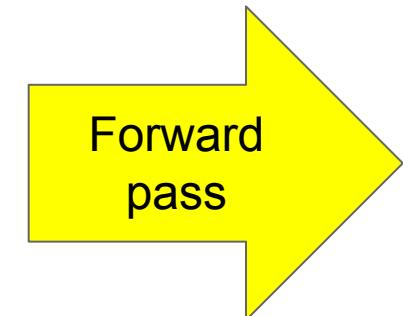
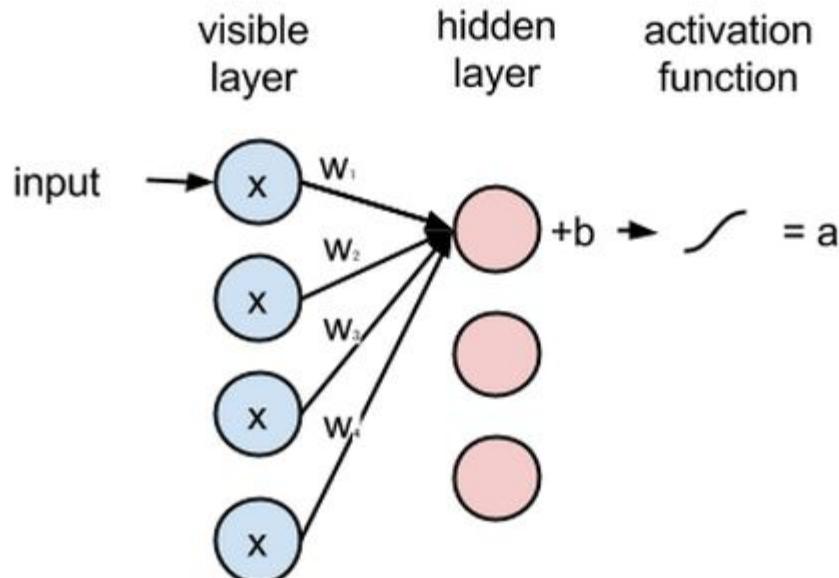
Salakhutdinov, Ruslan, Andriy Mnih, and Geoffrey Hinton. "[Restricted Boltzmann machines for collaborative filtering.](#)" Proceedings of the 24th international conference on Machine learning. ACM, 2007.

Restricted Boltzmann Machine (RBM)

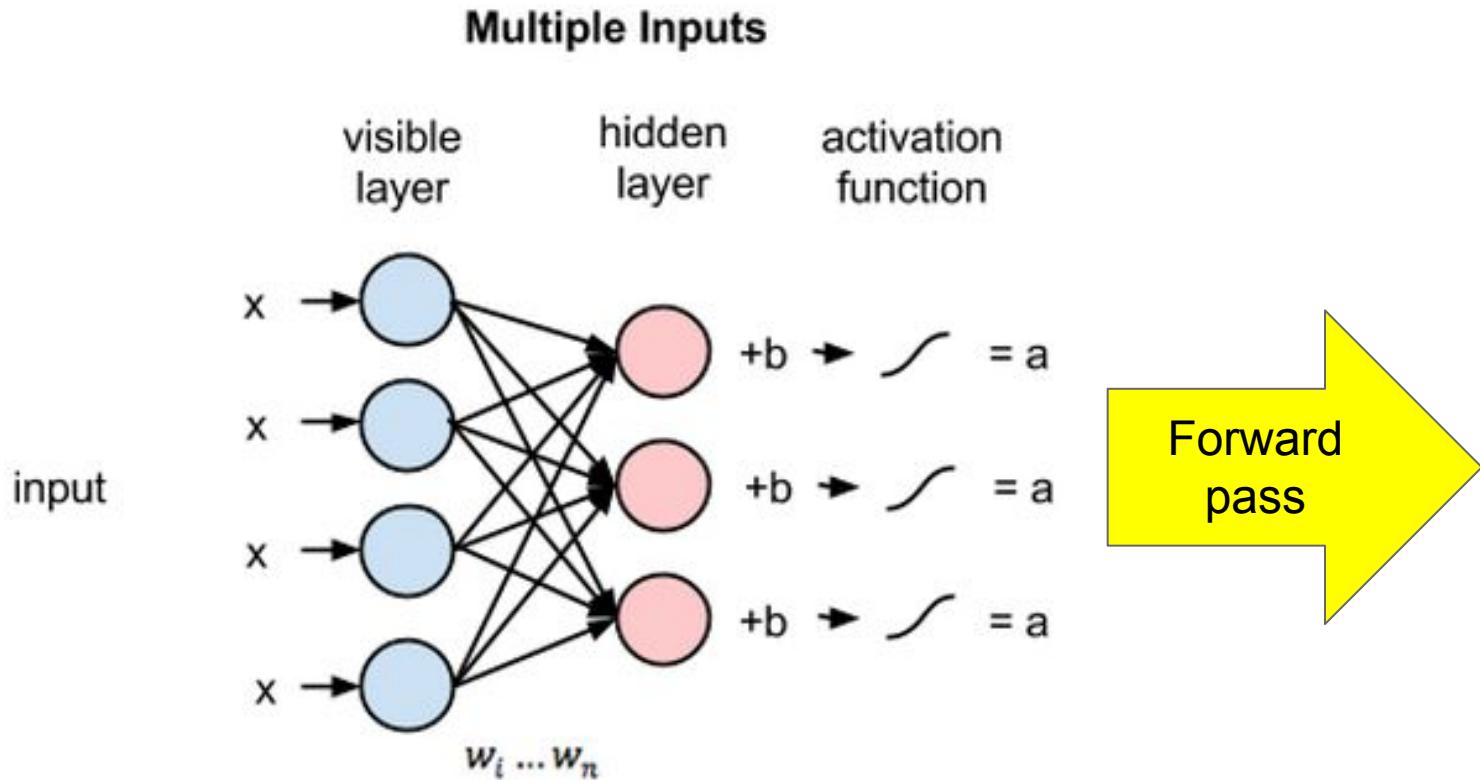


Restricted Boltzmann Machine (RBM)

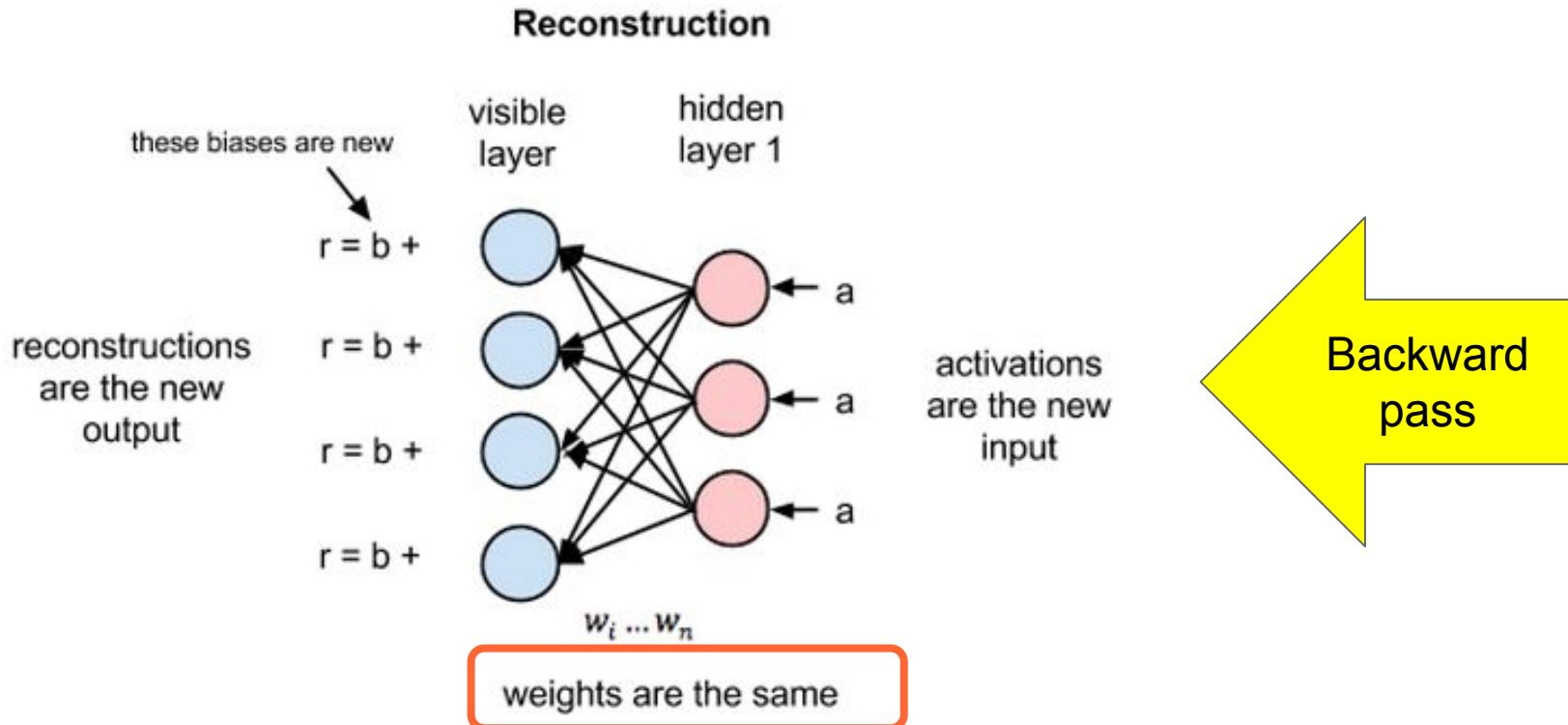
Weighted Inputs Combine @Hidden Node



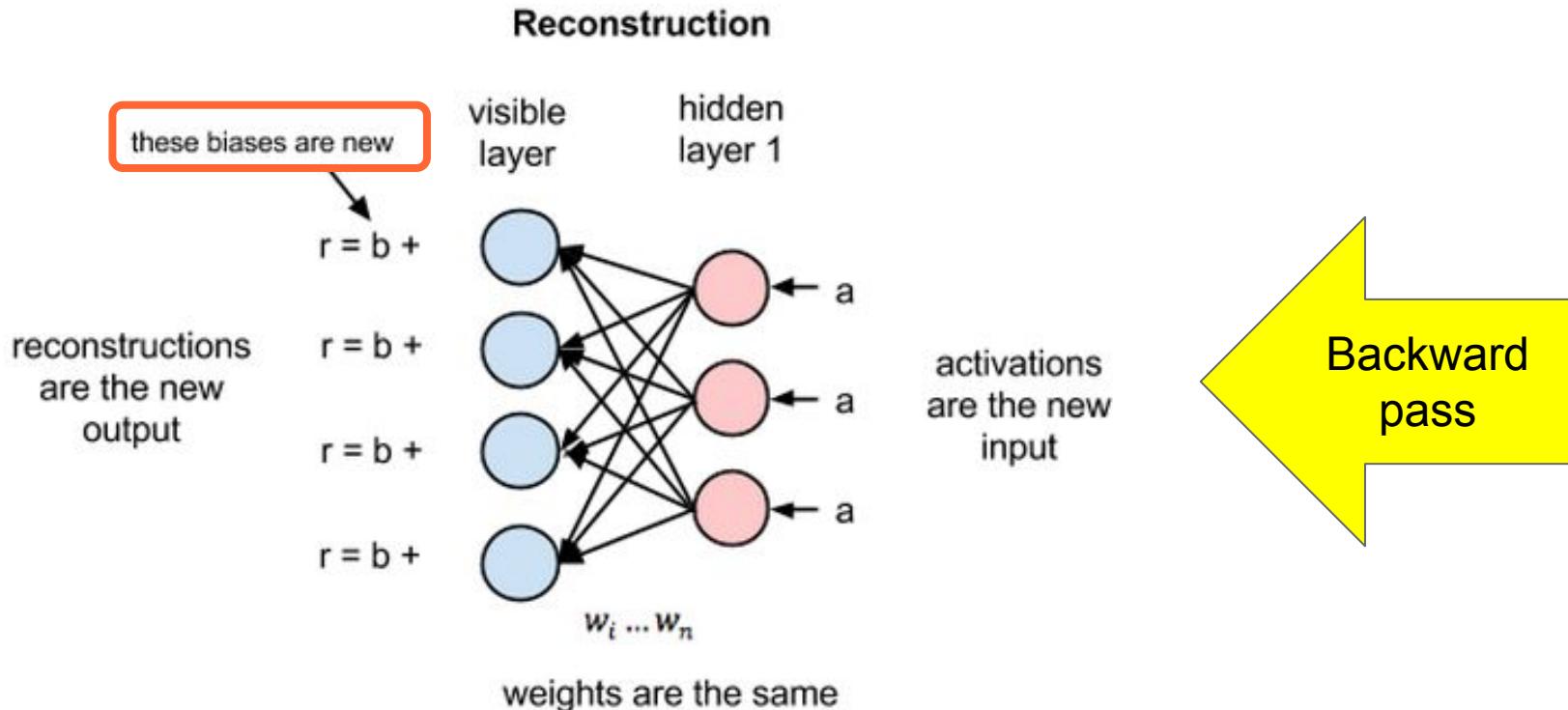
Restricted Boltzmann Machine (RBM)



Restricted Boltzmann Machine (RBM)



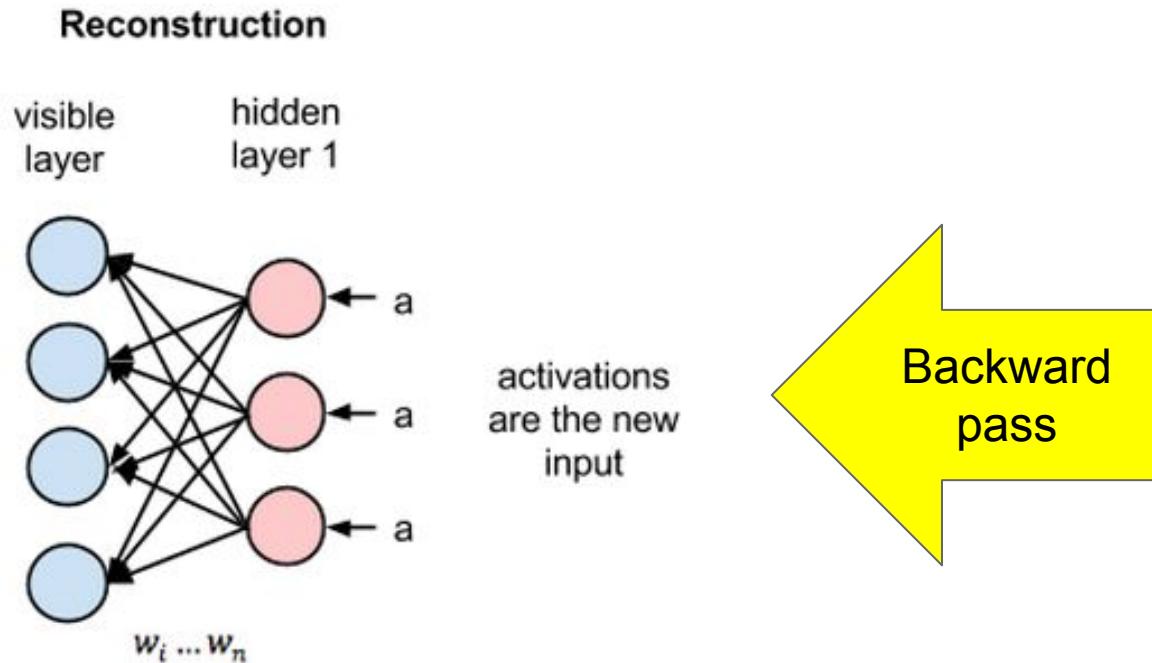
Restricted Boltzmann Machine (RBM)



Restricted Boltzmann Machine (RBM)

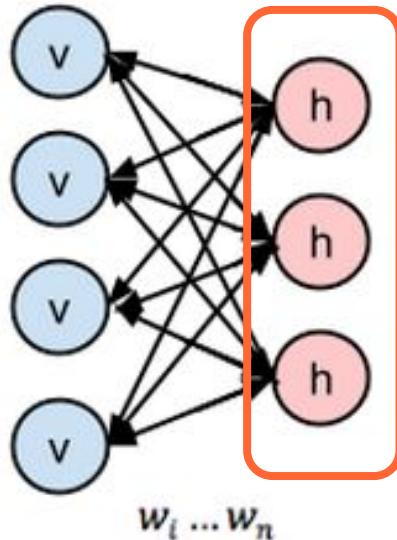
The reconstructed values at the visible layer are compared with the actual ones with the [KL Divergence](#).

$$D_{\text{KL}}(P\|Q) = - \sum_i P(i) \log \frac{Q(i)}{P(i)},$$

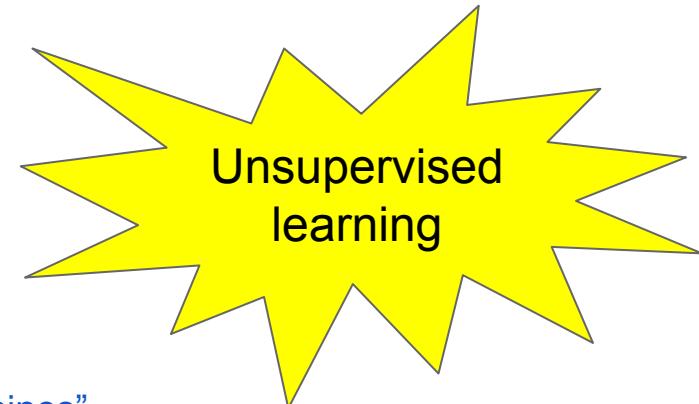


Restricted Boltzmann Machine (RBM)

WHY?



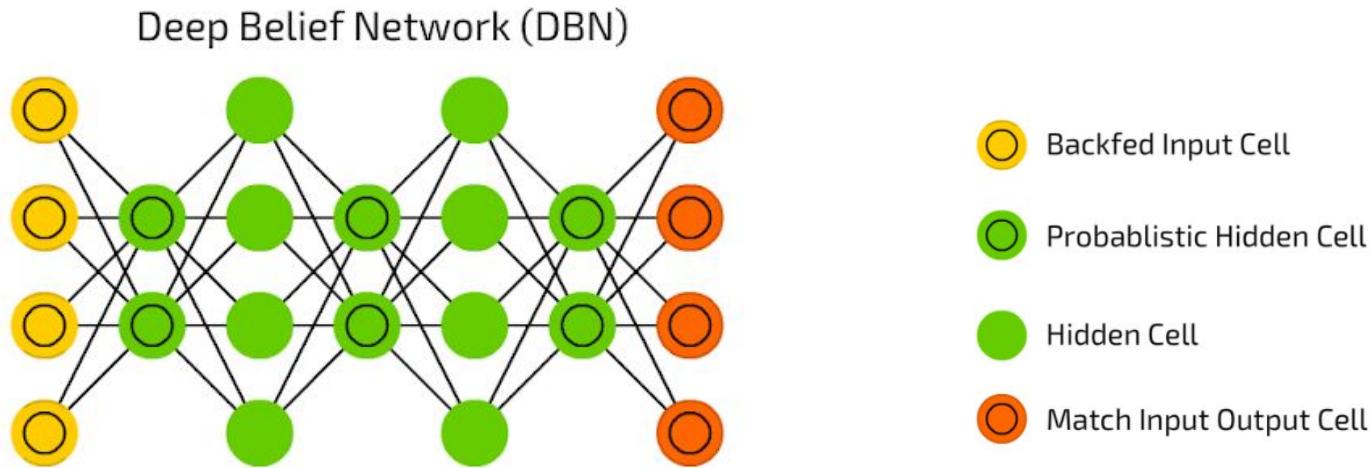
RBMs are a specific type of autoencoder.



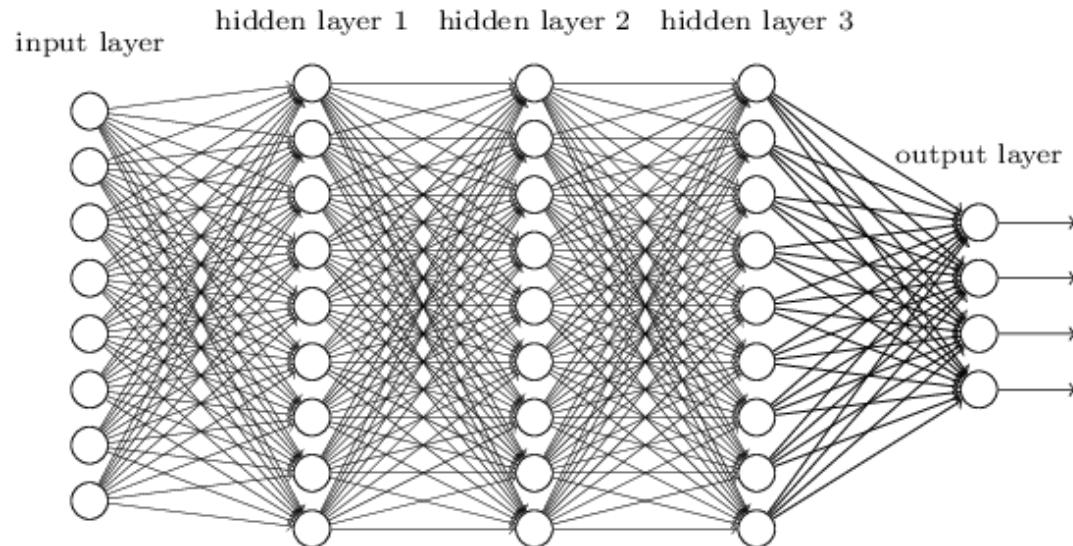
Restricted Boltzmann Machine (RBM)



Deep Belief Networks (DBN)



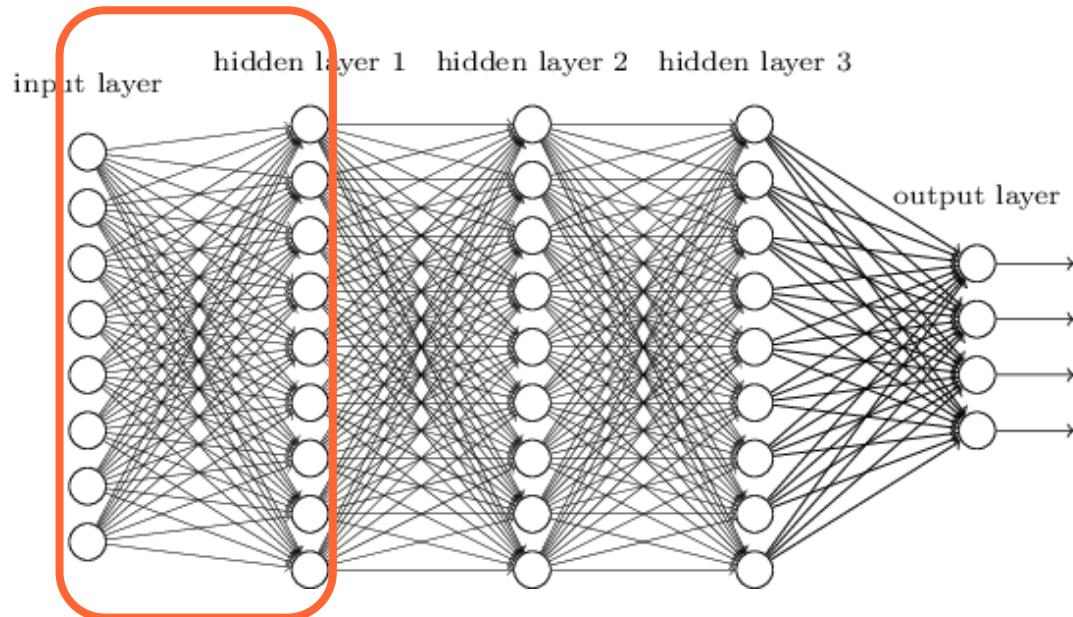
Deep Belief Networks (DBN)



- Architecture like an MLP.
- Training as a stack of RBMs.

Hinton, Geoffrey E., Simon Osindero, and Yee-Whye Teh. "[A fast learning algorithm for deep belief nets.](#)" Neural computation 18, no. 7 (2006): 1527-1554.

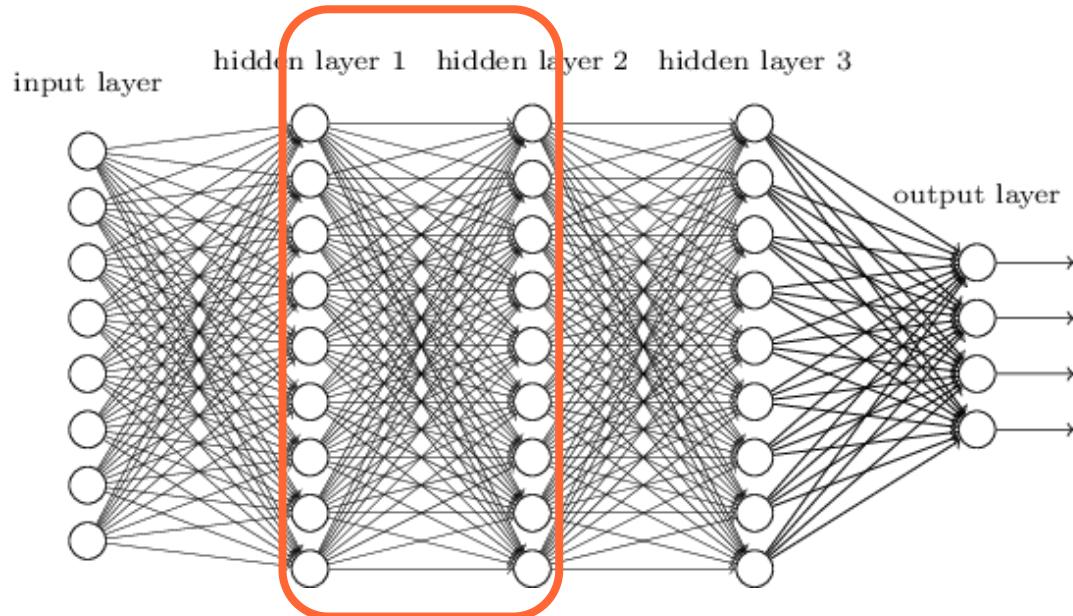
Deep Belief Networks (DBN)



- Architecture like an MLP.
- Training as a stack of RBMs.

Hinton, Geoffrey E., Simon Osindero, and Yee-Whye Teh. "[A fast learning algorithm for deep belief nets.](#)" Neural computation 18, no. 7 (2006): 1527-1554.

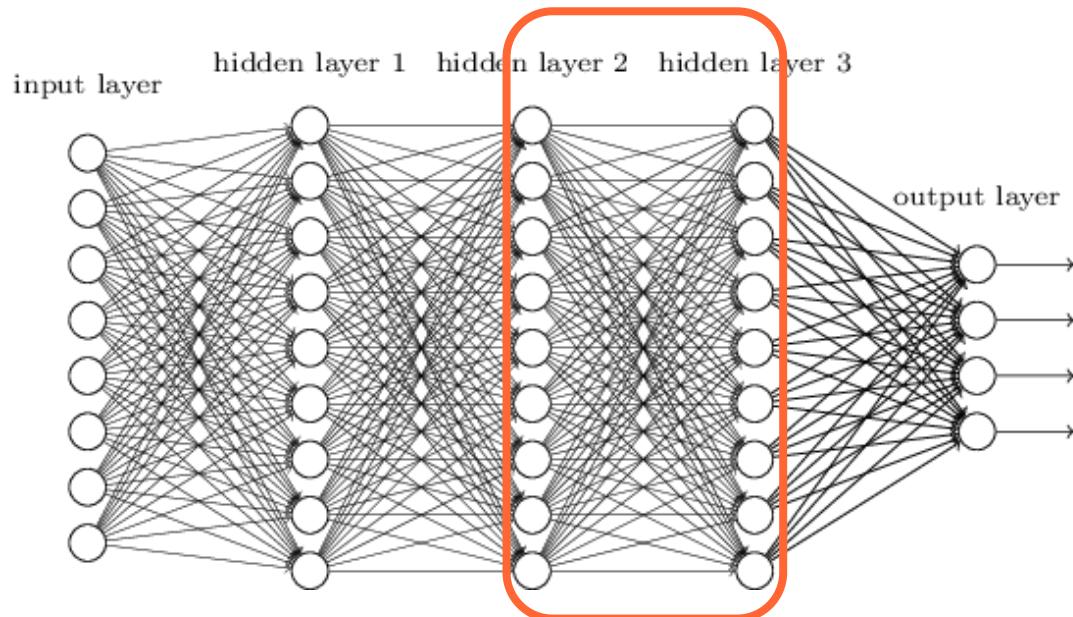
Deep Belief Networks (DBN)



- Architecture like an MLP.
- Training as a stack of RBMs.

Hinton, Geoffrey E., Simon Osindero, and Yee-Whye Teh. ["A fast learning algorithm for deep belief nets."](#) Neural computation 18, no. 7 (2006): 1527-1554.

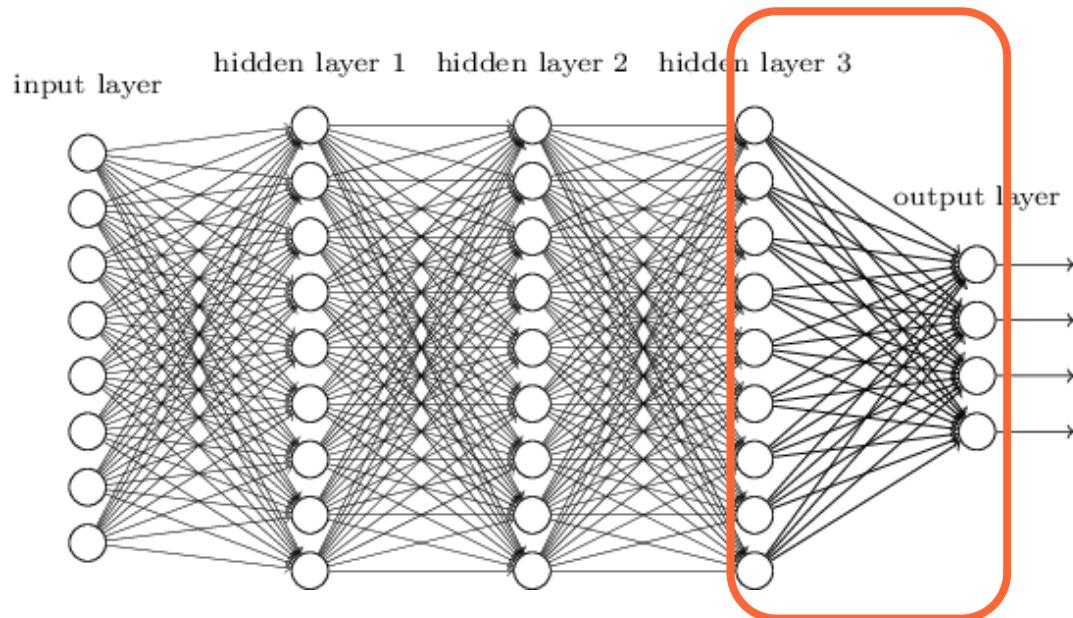
Deep Belief Networks (DBN)



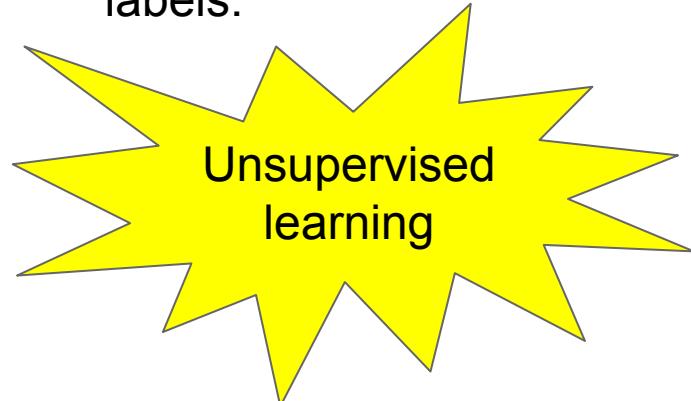
- Architecture like an MLP.
- Training as a stack of RBMs.

Hinton, Geoffrey E., Simon Osindero, and Yee-Whye Teh. "[A fast learning algorithm for deep belief nets.](#)" Neural computation 18, no. 7 (2006): 1527-1554.

Deep Belief Networks (DBN)

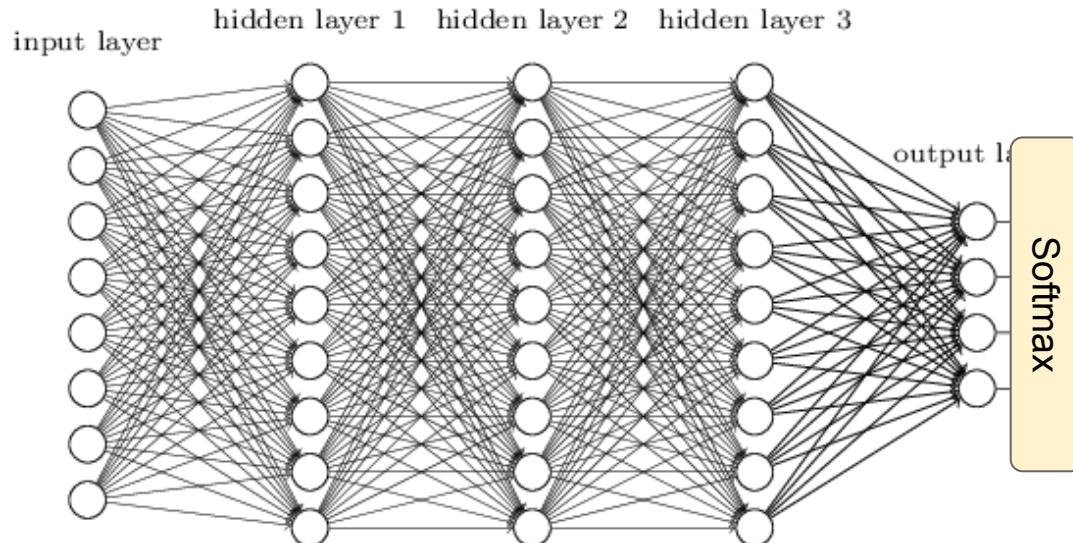


- Architecture like an MLP.
- Training as a stack of RBMs...
- ...so they do not need labels:



Hinton, Geoffrey E., Simon Osindero, and Yee-Whye Teh. "[A fast learning algorithm for deep belief nets.](#)" Neural computation 18, no. 7 (2006): 1527-1554.

Deep Belief Networks (DBN)

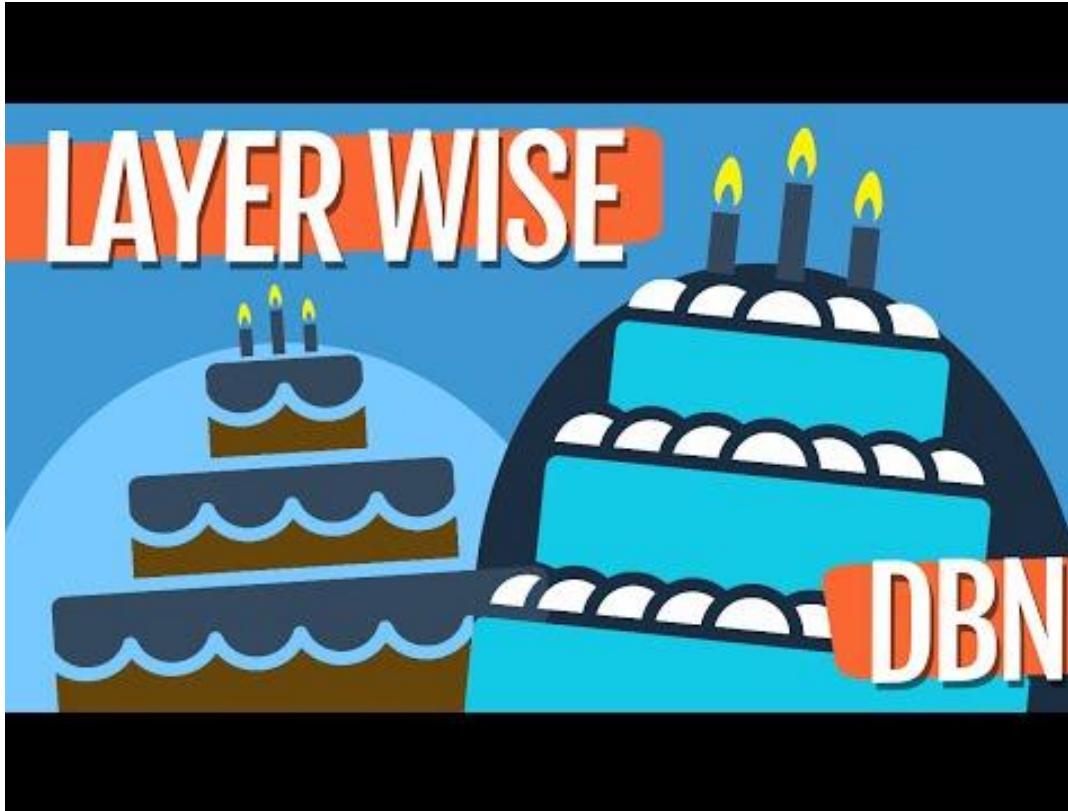


After the DBN is trained, it can be fine-tuned with a reduced amount of labels to solve a supervised task with superior performance.



Hinton, Geoffrey E., Simon Osindero, and Yee-Whye Teh. "[A fast learning algorithm for deep belief nets.](#)" Neural computation 18, no. 7 (2006): 1527-1554.

Deep Belief Networks (DBN)



Deep Belief Networks (DBN)



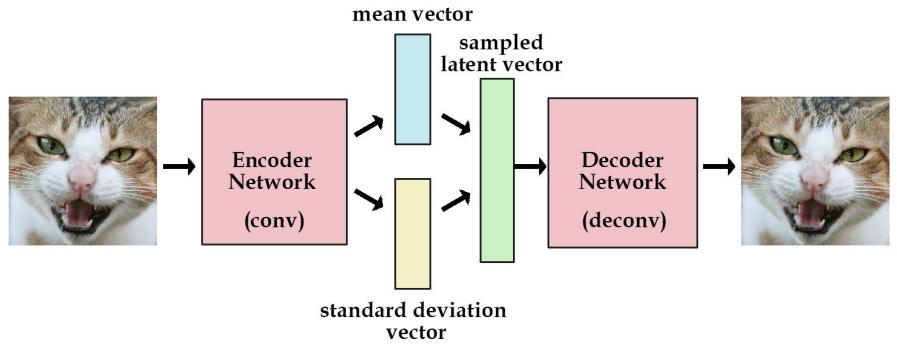
Geoffrey Hinton, "[Introduction to Deep Learning & Deep Belief Nets](#)" (2012)
Geoffrey Hinton, "[Tutorial on Deep Belief Networks](#)". NIPS 2007.

Deep Belief Networks (DBN)

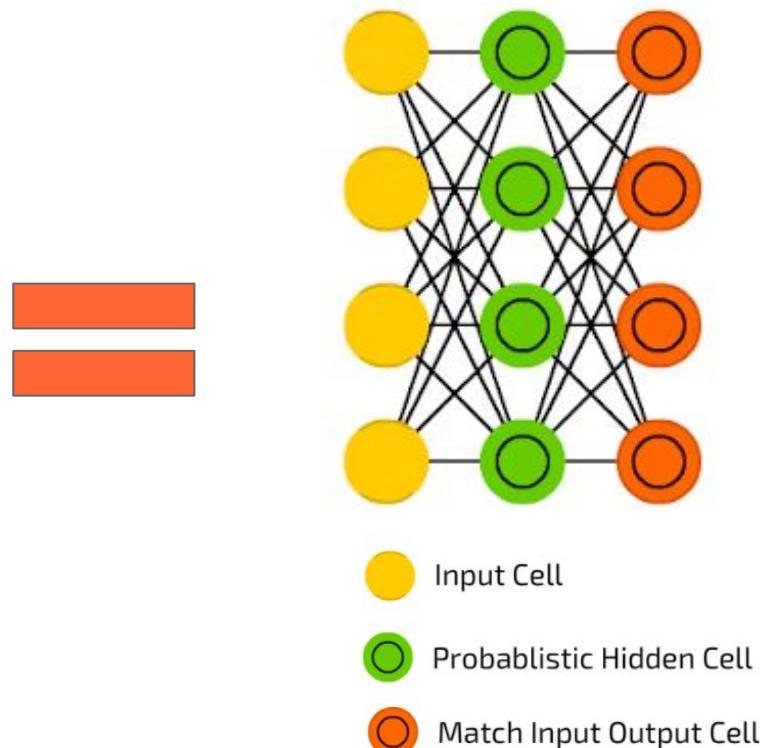
Geoffrey Hinton



Variational Autoencoder (VAE)

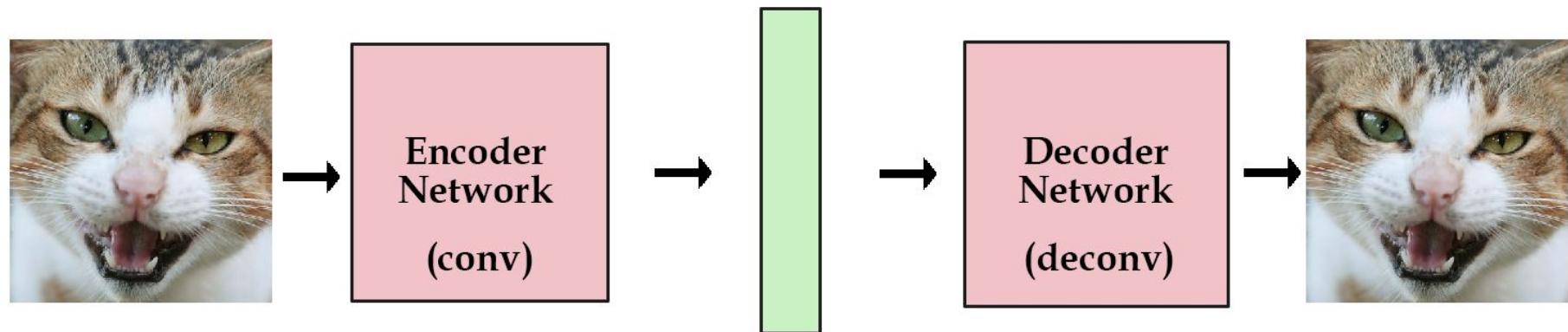


Variational AE (VAE)



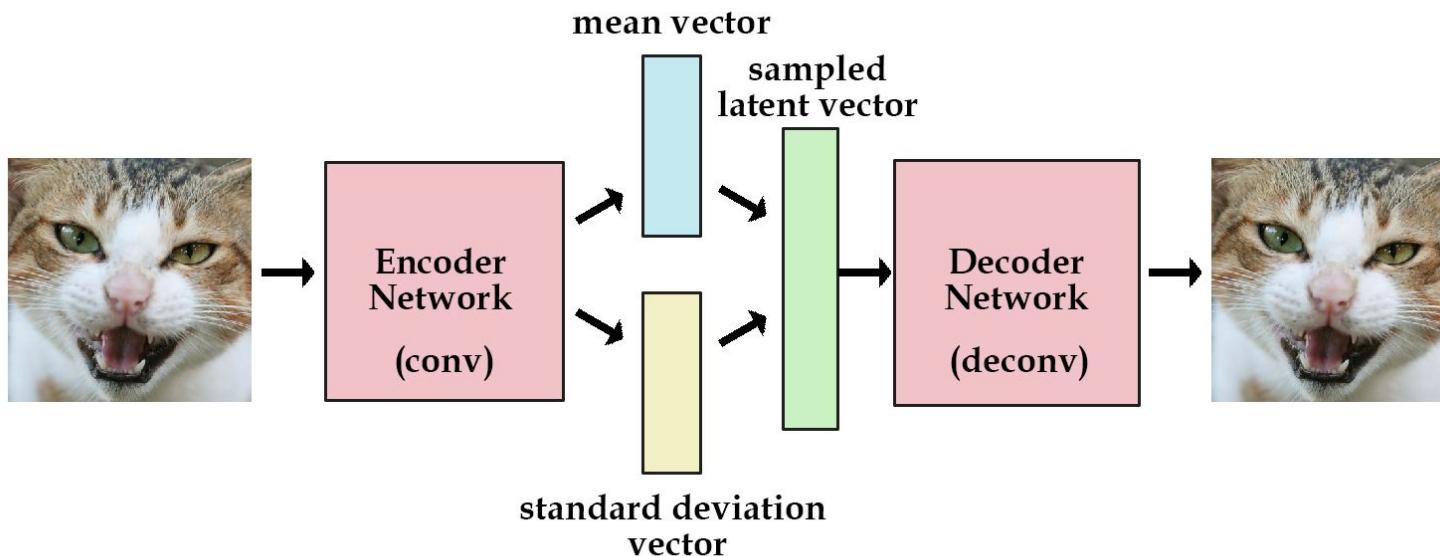
Variational Autoencoder (VAE)

The latent vector learned in the hidden layer of the basic autoencoder (in green)...



Variational Autoencoder (VAE)

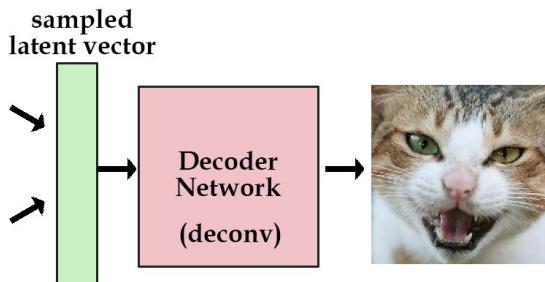
The latent vector learned in the hidden layer of the basic autoencoder (in green)...
...is forced to follow a unit Gaussian distribution in VAEs.



Variational Autoencoder (VAE)

Application #3

WHY?



Generative model:

- Create new samples by drawing from a Gaussian distribution.

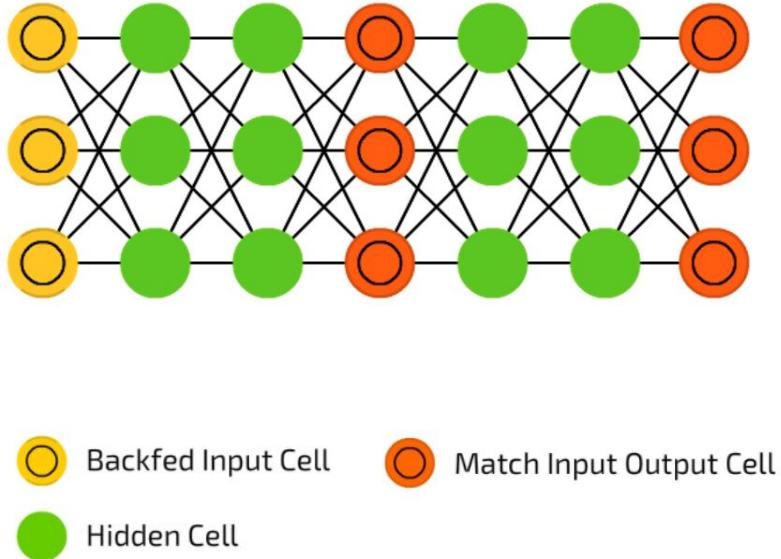
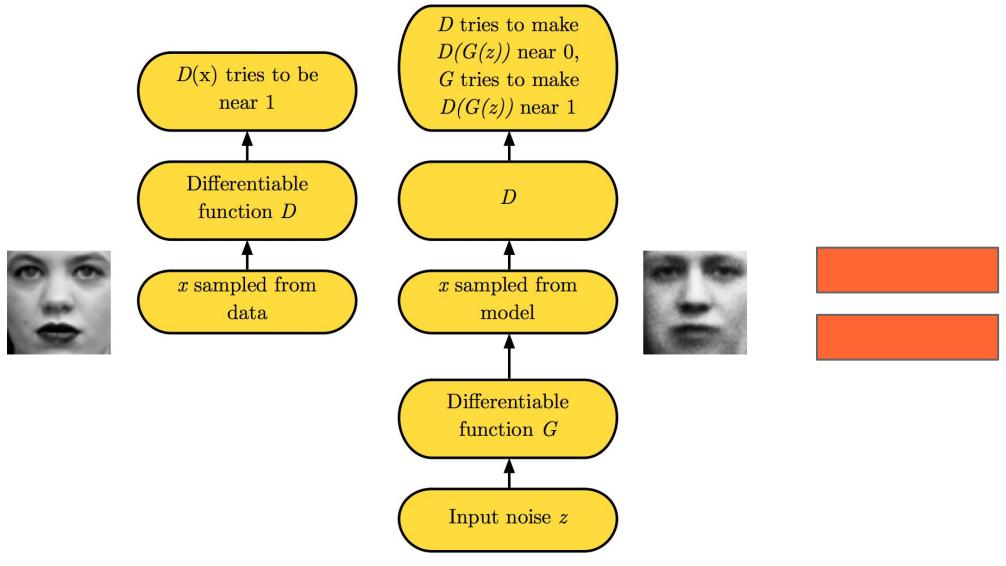
Unsupervised
learning

A large yellow starburst shape contains the text "Unsupervised learning".



Alec Radford, “[Face manifold from conv/deconv variational autoencoder](#)” (2015)

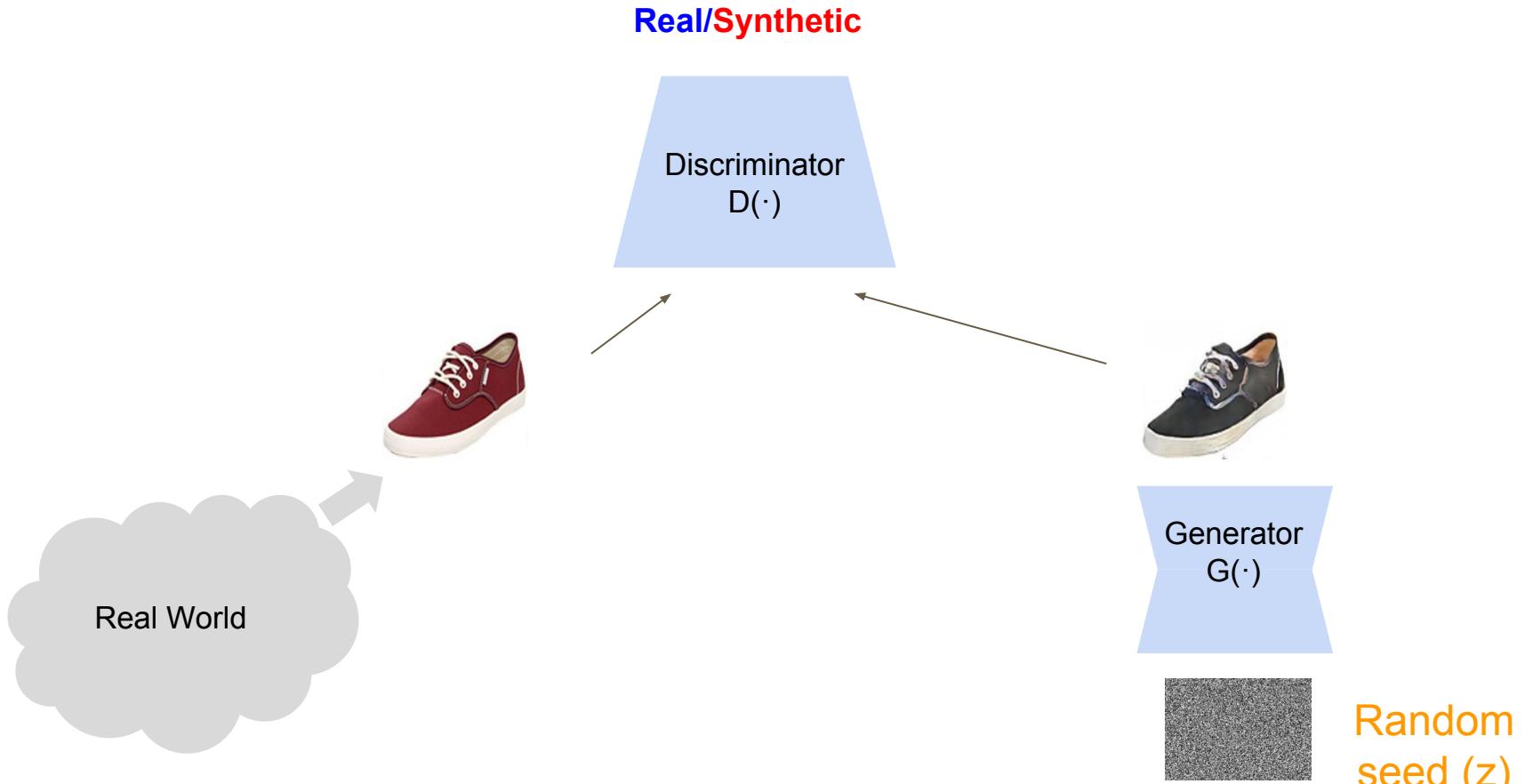
Adversarial Networks



Goodfellow, Ian, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. ["Generative adversarial nets."](#) NIPS 2014

Goodfellow, Ian. ["NIPS 2016 Tutorial: Generative Adversarial Networks."](#) arXiv preprint arXiv:1701.00160 (2016).

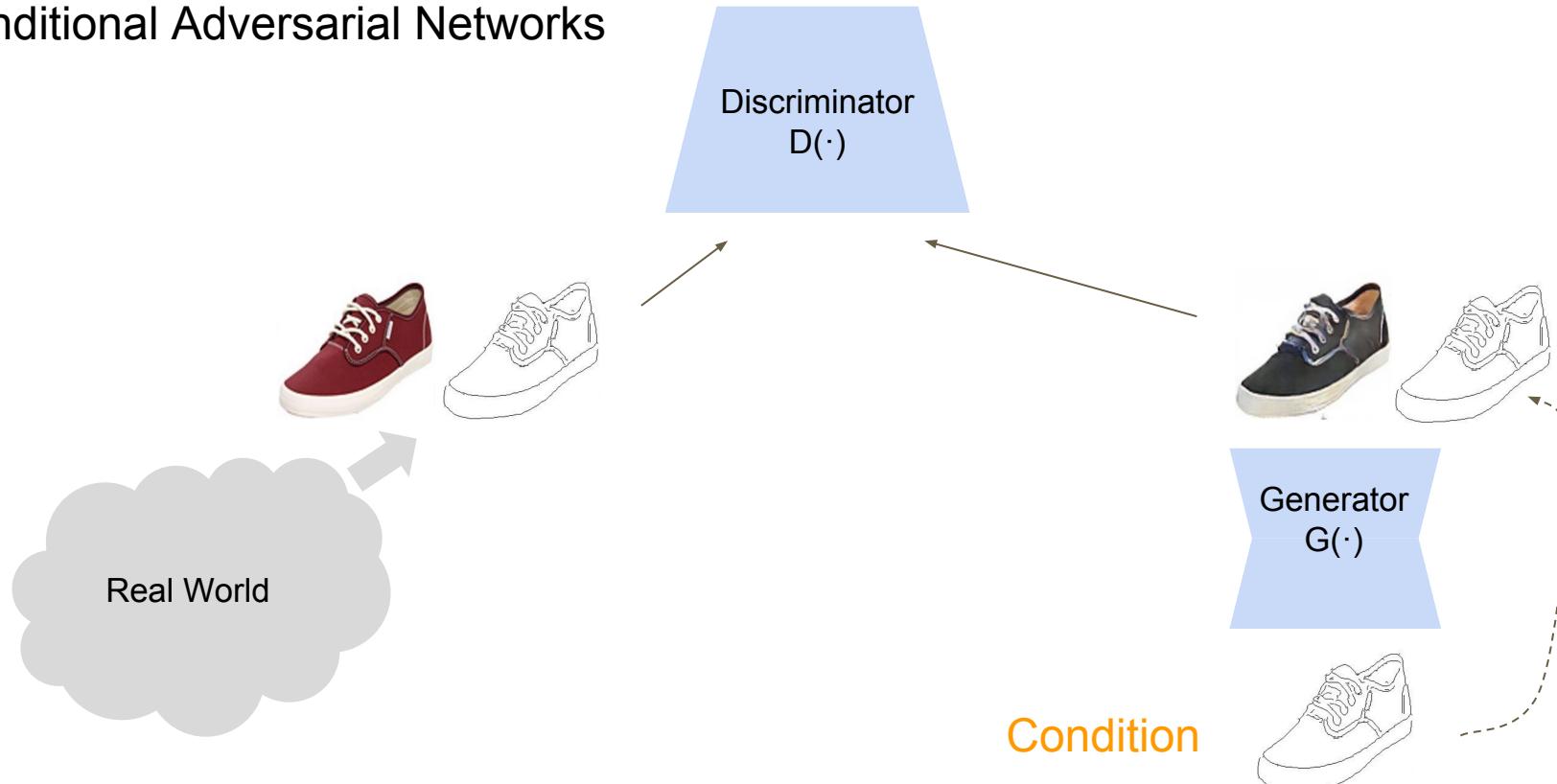
Adversarial Networks



Adversarial Networks

Real/Synthetic

Conditional Adversarial Networks



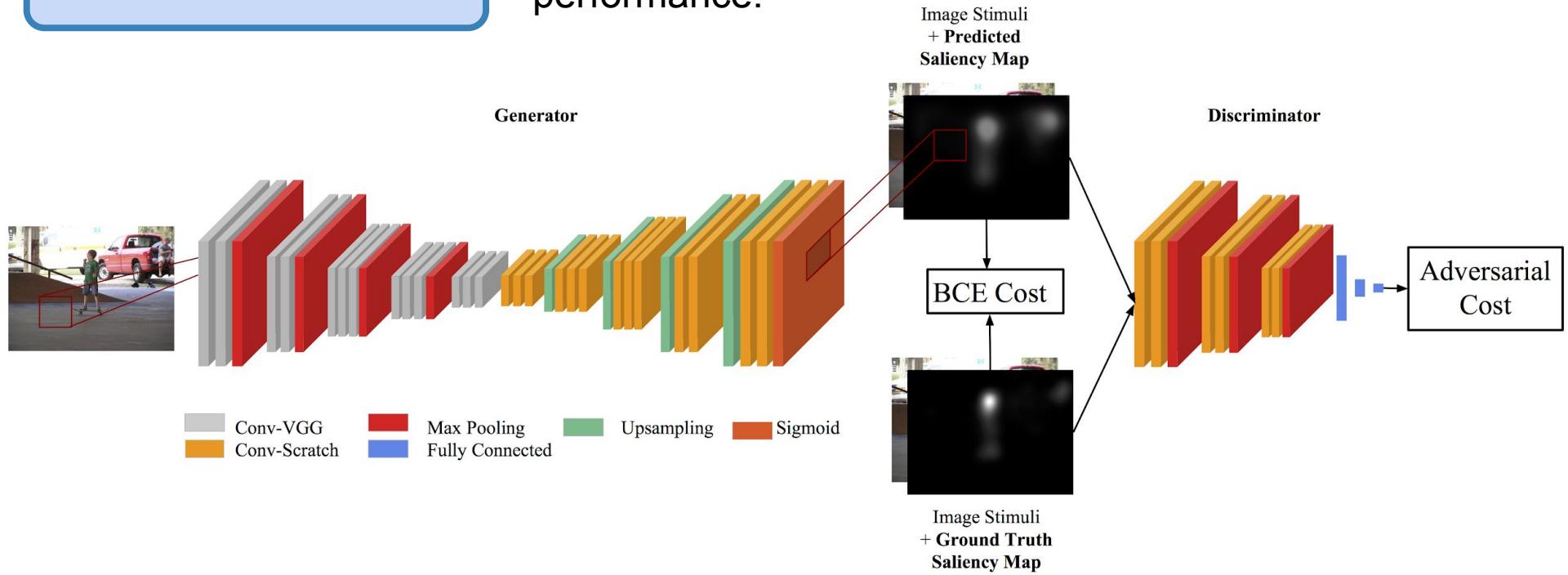


Tero Karras, Timo Aila, Samuli Laine, Jaakko Lehtinen, ["Progressive Growing of GANs for Improved Quality, Stability, and Variation"](#) (submitted to ICLR 2018)

Adversarial Networks

Application #4

Introduce an additional term to the loss to improve performance.

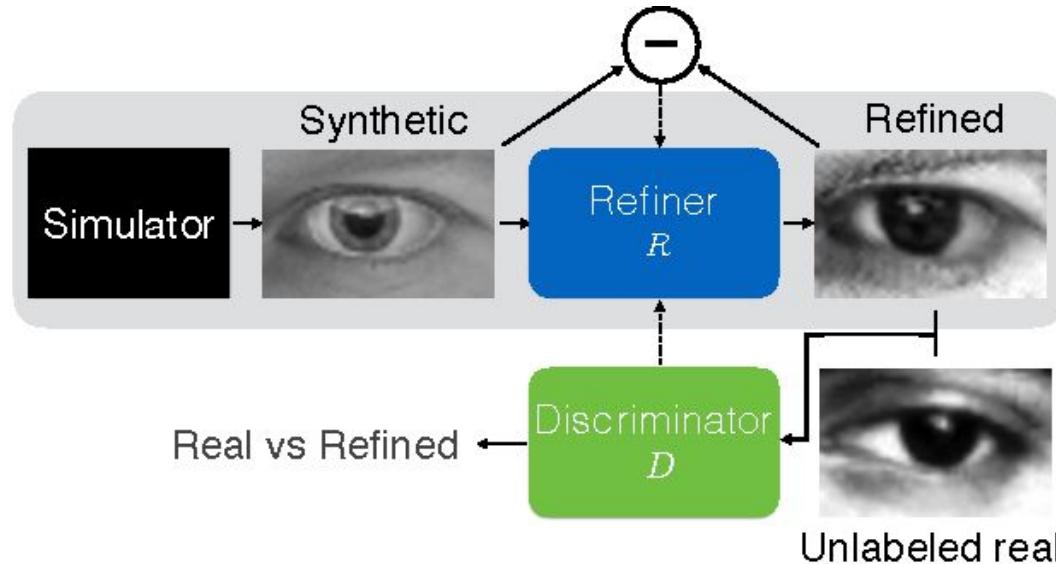


Junting Pan, Cristian Canton, Kevin McGuinness, Noel E. O'Connor, Jordi Torres, Elisa Sayrol and Xavier Giro-i-Nieto. "[SalGAN: Visual Saliency Prediction with Generative Adversarial Networks.](#)" CVPRW 2017.

Adversarial Networks

Application #5

Generate realistic data for training.



Shrivastava, Ashish, Tomas Pfister, Oncel Tuzel, Josh Susskind, Wenda Wang, and Russ Webb. ["Learning from simulated and unsupervised images through adversarial training."](#) CVPR 2017 (best paper) [\[video\]](#)

Adversarial Networks



Victòria Miró / Oriol Esteve, "La meitat de les notícies que consumirem el 2022 seran falses". TN Vespre, TV3 (26 de Novembre, 2017)



Conclusions

We can categorize three types of learning procedures:

1. Supervised Learning:

$$\mathbf{y} = f(\mathbf{x})$$

Predict label y corresponding to observation x

2. Unsupervised Learning:

$$f(\mathbf{x})$$

Estimate the distribution of observation x

3. Reinforcement Learning (RL):

$$\mathbf{y} = f(\mathbf{x})$$

$$\mathbf{z}$$

Predict action y based on observation x , to maximize a future reward z



Questions ?

Undergradese

What undergrads ask vs. what they're REALLY asking

"Is it going to be an open book exam?"

Translation: "I don't have to actually memorize anything, do I?"

"Hmm, what do you mean by that?"

Translation: "What's the answer so we can all go home."

"Are you going to have office hours today?"

Translation: "Can I do my homework in your office?"

"Can i get an extension?"

Translation: "Can you re-arrange your life around mine?"

"Is this going to be on the test?"

Translation: "Tell us what's going to be on the test."

"Is grading going to be curved?"

Translation: "Can I do a mediocre job and still get an A?"

