ULB

UNIVERSITÉ LIBRE DE BRUXELLES

Everything you always wanted to know about ML and videogames (but were afraid to ask)

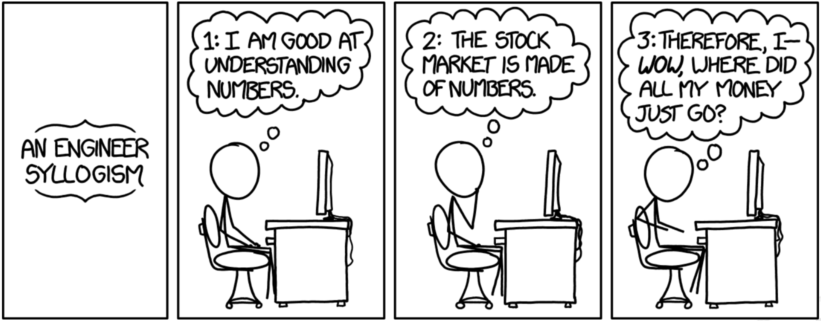
UrLab

Jacopo De Stefani - jdestefa@ulb.ac.be

Université Libre de Bruxelles

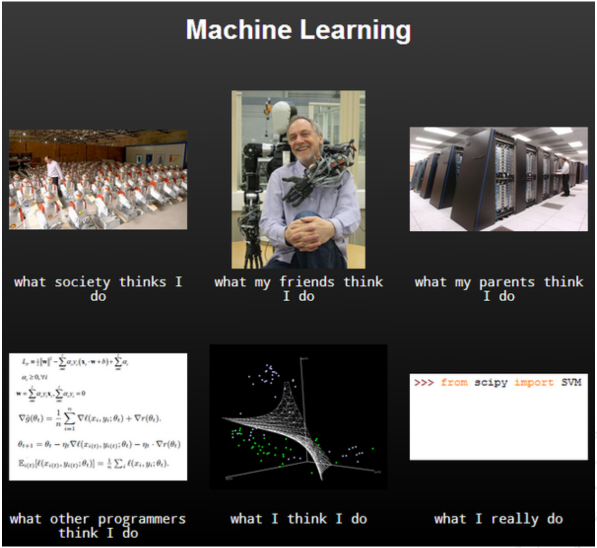
Tuesday 3rd September, 2019

Disclaimer



Relevant XCKD: 1570

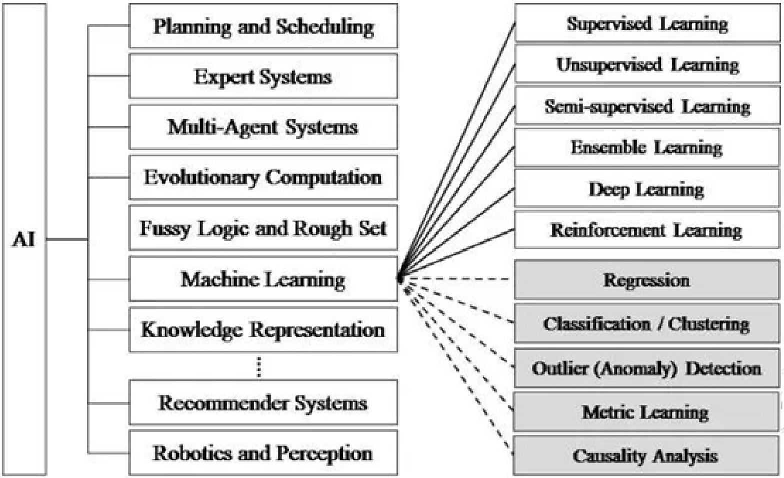
Machine Learning?



Source: Quora

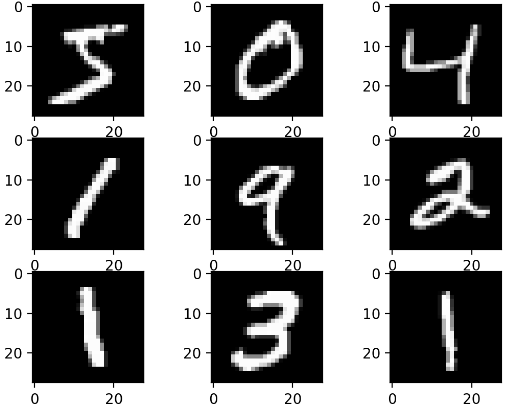
UNIVERSITÉ LIBRE DE BRUXELLES

AI vs ML?



Source: Huawei Resarch Blog

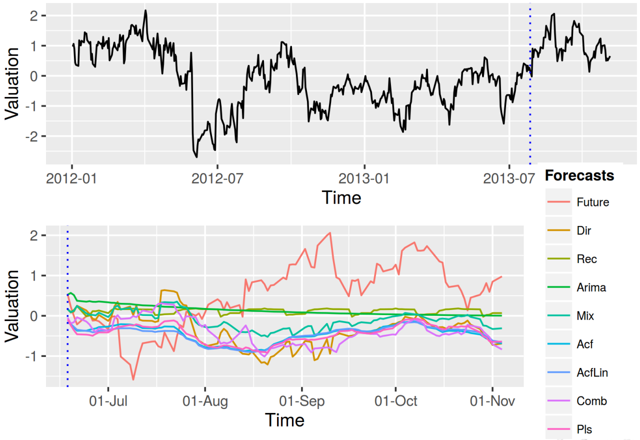
Some examples - Image classification



Some examples - Fraud Detection

Some examples - Villo availability prediction

Some examples - Time Series Analysis



What are the common points?

- Structured data
  - Often not the case in real-life problem
  - Preprocessing

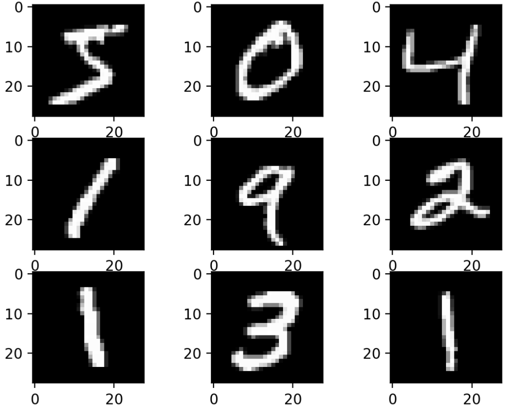
What are the common points?

- Structured data
  - Often not the case in real-life problem
  - Preprocessing
- Single output variable
  - Fraud Detection,Image classification: Discrete value ⇒ **Classification**
  - Villo,TS: Continuous value ⇒ **Regression**

What are the common points?

- Structured data
  - Often not the case in real-life problem
  - Preprocessing
- Single output variable
  - Fraud Detection,Image classification: Discrete value $\Rightarrow$ **Classification**
  - Villo,TS: Continuous value $\Rightarrow$ **Regression**
- Unknown Input/Output mapping
  - No available model
  - Data-driven

Some examples - Image classification



$$h_{IC} : \mathbf{X} \in \mathbb{R}^{32\times32} \mapsto y \in \{0, \cdots, 9\}$$

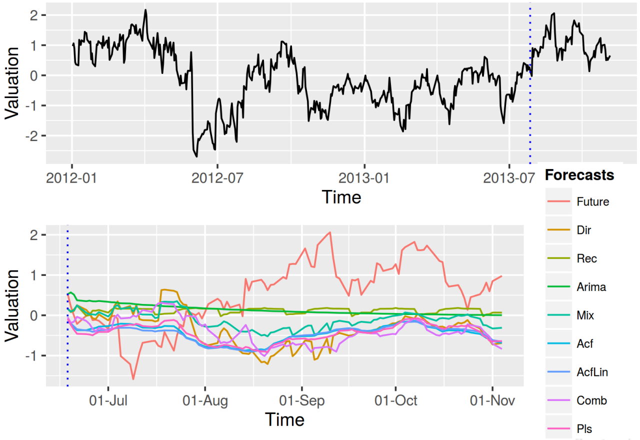
Some examples - Fraud Detection

$$h_{FD} :< ID, Country, Amount, Amount_{avg}, .. > \mapsto y \in \{0,1\}$$

Some examples - Regression

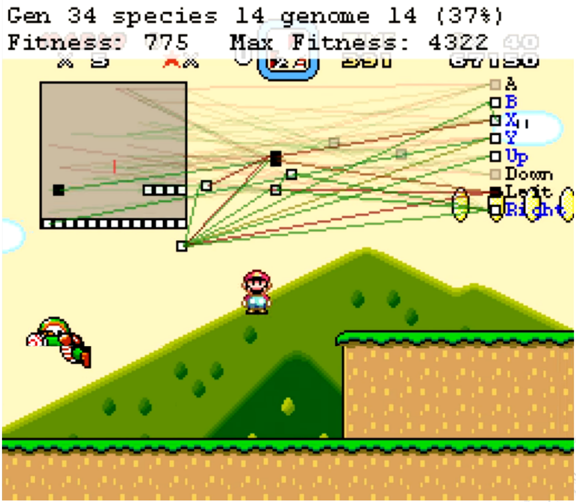
$$h_R :< Lat, Long, Weather, Day, \cdots > \mapsto y \in \mathbb{R}^+$$

Some examples - Time Series Analysis



$$h_{TS} : \mathbf{X} = [y_{t-d}, \cdots, y_{t-1}] \in \mathbb{R}^d \mapsto y = y_t \in \mathbb{R}$$

Yes, but what about videogames? - Agents

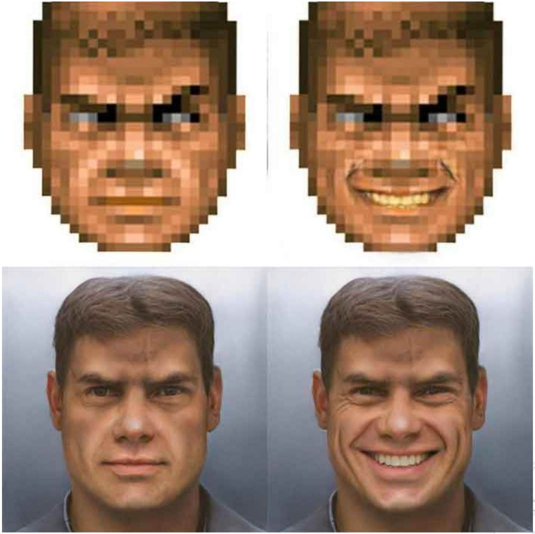


$$h_P : \mathbf{X} \in \mathbb{R}^{64\times 64} \mapsto y \in \{\uparrow, \downarrow, \rightarrow, \leftarrow, A, B, X, Y, \cdots\}$$

Yes, but what about videogames? - Agents

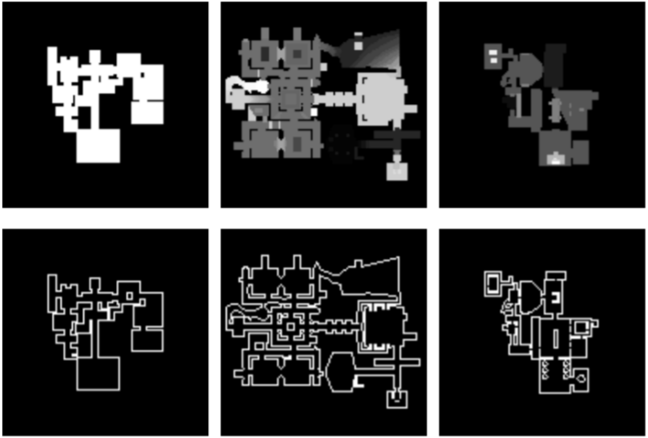
$$h_P : \mathbf{X} \in \mathbb{R}^{320\times 240} \mapsto y \in \{\uparrow, \downarrow, \rightarrow, \leftarrow, Shoot, \cdots\}$$

Yes, but what about videogames? - Visual



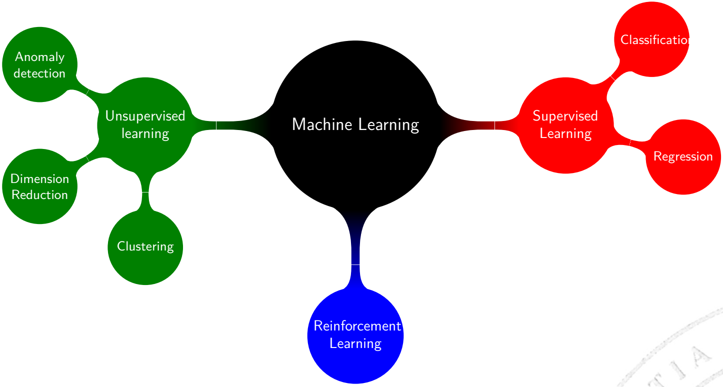
$$h_V : \mathbf{X} \in \mathbb{R}^{32\times 32} \mapsto \mathbb{R}^{1024\times 1024}$$

Yes, but what about videogames? - Level design - Giacomello et al. [2018]



$$h_{LD} : \mathbf{X} \in \mathbb{R}^{128\times128} \mapsto \mathbb{R}^{128\times128}$$

Machine Learning Overview



Machine Learning

Definition

A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E.

T. Mitchell,1997

Learning machine - [Bontempi]

- Learning machine
  - **Hypothesis/Model:** $h(\cdot,\cdot) :< \mathbf{x}, \vartheta > \mapsto h(\mathbf{x}, \vartheta) \in \mathcal{Y}$
  - **Class of hypotheses:** $h(\cdot, \vartheta), \vartheta \in \Theta$
  - **Loss function:** $L(\cdot,\cdot) :< \mathbf{x}, y > \mapsto L(\mathbf{x}, y) \in \mathbb{R}$
  - **Learning algorithm:** $\mathcal{L} :< \Theta, D_n > \mapsto h(\cdot, \vartheta_n)$

Empirical risk minimization - [Bontempi]

$$\vartheta_n = \vartheta(D_n) = \arg\min_{\vartheta \in \Theta} R_{emp}(\vartheta) \tag{1}$$

$$R_{emp}(\vartheta) = \frac{1}{n} \sum_{i=1}^{n} L(y_i, h(\mathbf{x_i}, \vartheta)) \tag{2}$$

$$\nabla J(\vartheta) = 0 \tag{3}$$

Machine Learning Process - [Bontempi]

Preliminary phase

1. Problem formulation
2. Experimental design
3. Preprocessing step
    - Missing data
    - Feature selection
    - Outlier removal

Learning phase

1. Parametric identification
2. Model selection

Model selection - [Bontempi]

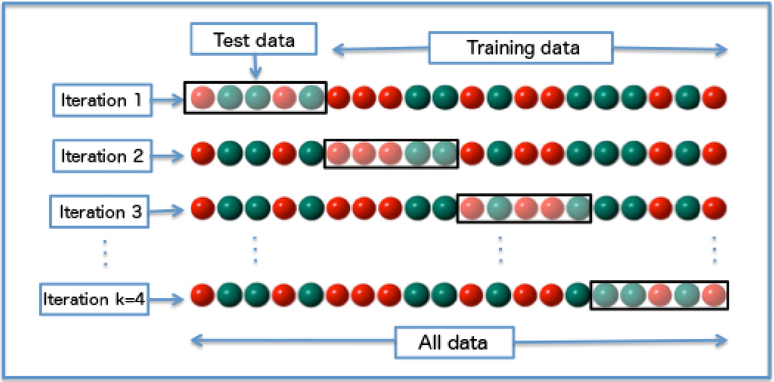
The selection of a model is usually perfomed by looking at its performance:

$$R_{ts}(\vartheta) = \frac{1}{n_{ts}} \sum_{i=1}^{n_{ts}} L(y_i, h(\mathbf{x_i}, \vartheta)) \quad (4)$$

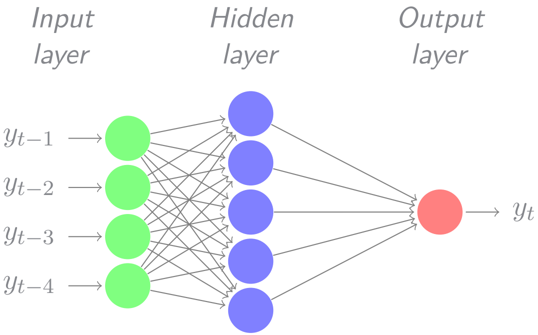
on unseen data:

$$D_{ts} = \{< \mathbf{x_{n+1}}, \mathbf{y_{n+1}} >, \cdots, < \mathbf{x_{n+n_{ts}}}, y_{n+n_{ts}} >\} \quad (5)$$

Cross Validation



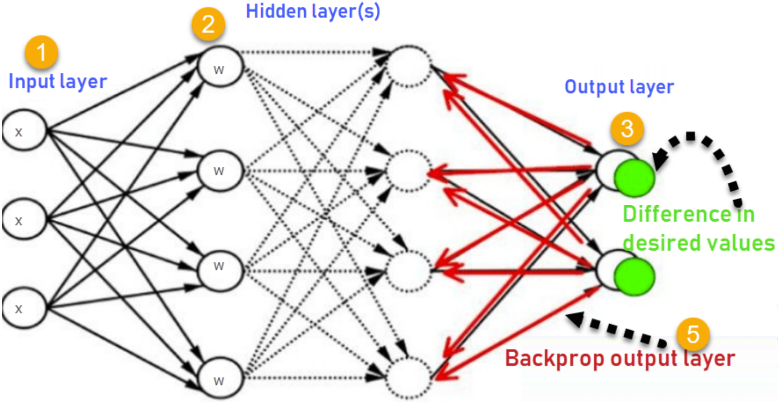
Models - ANN



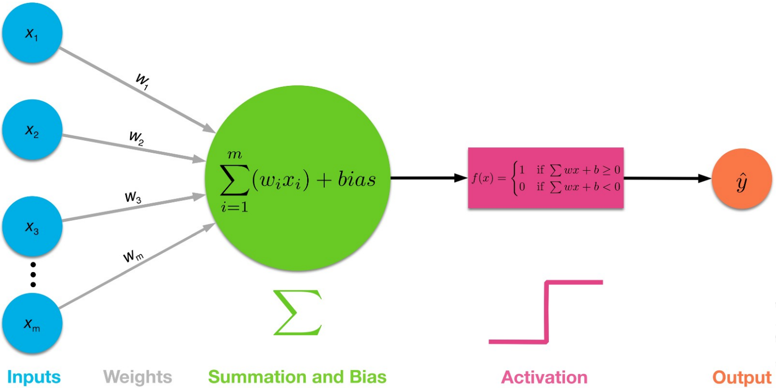
$$y = f\left(b_o + \sum_{j=1}^{|H|} w_{jo} \cdot g\left(\sum_{i=1}^{|I|} w_{ij} x_i + b_j\right)\right)$$

- **Parameters:** $\vartheta = [\mathbf{w_h}, \mathbf{w_o}]$
- **Parametric identification:** Gradient descent + Backpropagation

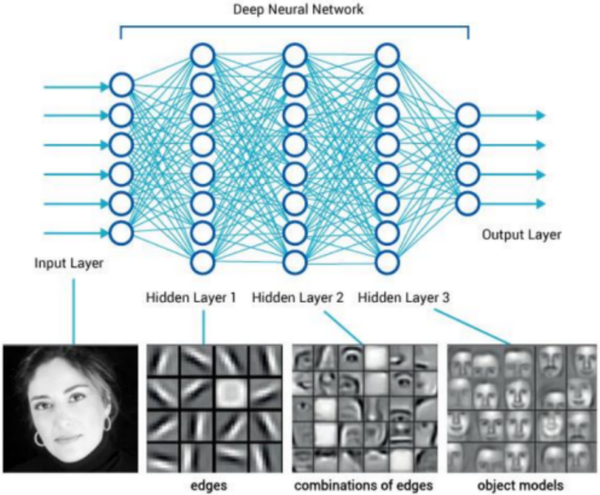
Backpropagation



Perceptron

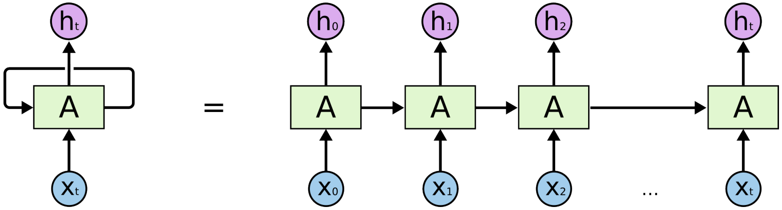

x_1
x_2
x_3
x_m
w_1
w_2
w_3
w_m
$\sum_{i=1}^{m}(w_i x_i) + bias$
$f(x) = \begin{cases} 1 & \text{if } \sum wx + b \geq 0 \\ 0 & \text{if } \sum wx + b < 0 \end{cases}$
$\hat{y}$
Inputs
Weights
Summation and Bias
Activation
Output

Deep Learning - Intuition

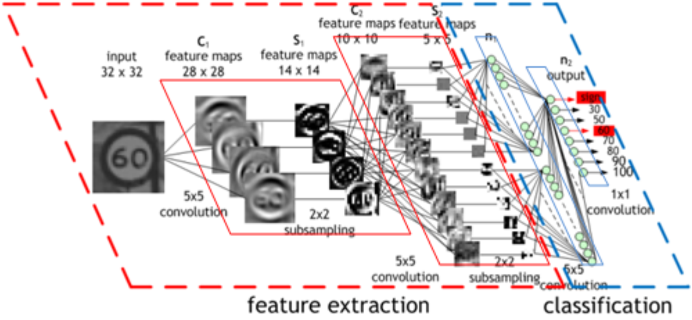


Demo : http://playground.tensorflow.org/

Deep Learning - RNN - Intuition

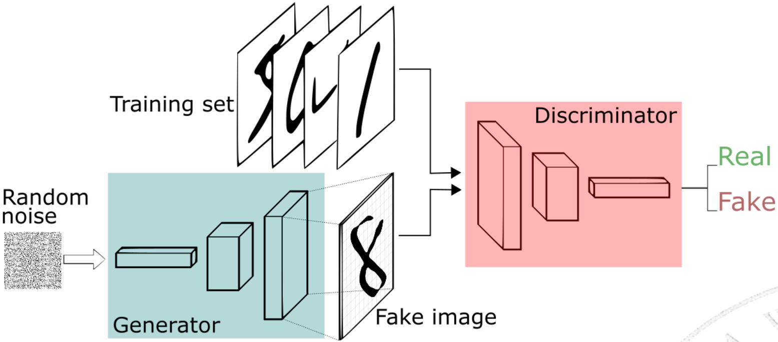


Deep Learning - CNN - Intuition



Demo: https://cs.stanford.edu/people/karpathy/convnetjs/demo/mnist.html

Models - GAN

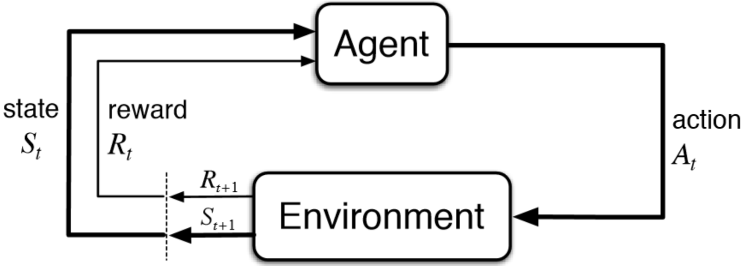


Training set
Discriminator
Real
Fake
Random noise
Generator
Fake image

And many more...

- **Non-parametric methods**
  - Decision Trees
  - K-nearest neighbors
  - Radial Basis Functions
- **Network based**
  - Restricted Boltzmann Machines
- **Ensemble techniques**
  - Random Forests
  - Gradient Boosting

Models - Reinforcement Learning



Models - Deep Reinforcement Learning

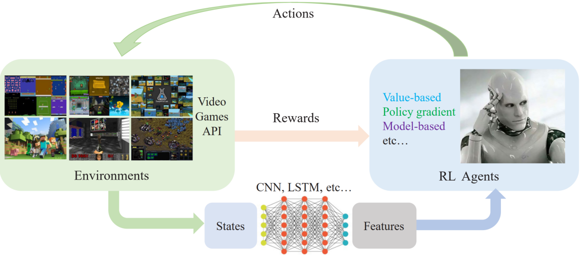


Fig. 1. The framework diagram of the typical DRL for video games. The deep learning model takes input from video games API, and extract meaningful features automatically. DRL agents produces actions based on these features, and make the environments transfer to next state.

Source: [Shao et al., 2019]

Models - Deep Reinforcement Learning Architectures

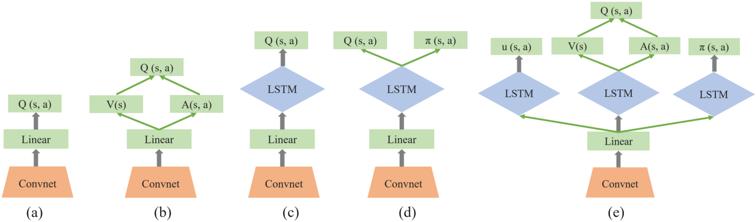
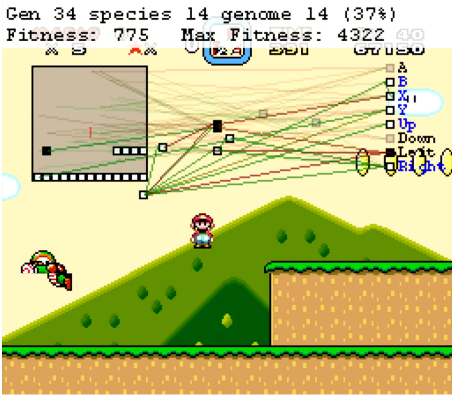


Fig. 2. The network architectures of typical DRL methods, with increased complexity and performance. (a): DQN network; (b)Dueling DQN network; (c): DRQN network; (d): Actor-critic network; (e): Reactor network.

Source: [Shao et al., 2019]

Wrap up - Review



- **Computer vision**
  - CNN
  - Specialized algorithms
- **Agent decision**
  - (Deep) Reinforcement Learning
  - MLP

Wrap-up - Review

Table 1 Machine learning techniques used within academic digital game research

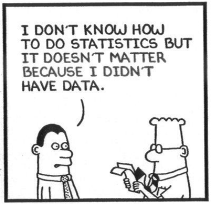
Learning technique	Game agent representation	Game environment	Reference
Backpropagation	Multi-layer perceptron	*Motocross the force*	Chaperot and Fyfe (2006)
	Multi-layer perceptron	Simulated racing	Togelius et al. (2007b)
	Multi-layer perceptron	Simulated social environment	MacNamee and Cunningham (2003)
	Multi-layer perceptron	*Soldier of fortune 2*	Geisler (2004)
	Multi-layer perceptron (ATA[a])	*Legion-I*	Bryant and Miikkulainen (2003)
	Multi-layer perceptron (ATA)	*Legion-II*	Bryant and Miikkulainen (2006a)
	Multi-layer perceptron (Ensemble)	*Quake II*	Bauckhage and Thurau (2004)
Backpropagation (LM[b])	Multi-layer perceptron	*FlatLand*	Yannakakis et al. (2003)
Backpropagation (bagging)	Multi-layer perceptron (ensemble)	*Motocross the force*	Chaperot and Fyfe (2006)
	Multi-layer perceptron (ensemble)	*Soldier of fortune 2*	Geisler (2004)
Backpropagation (boosting)	Multi-layer perceptron	*Motocross the force*	Chaperot and Fyfe (2006)
	Multi-layer perceptron (Ensemble)	*Soldier of fortune 2*	Geisler (2004)
SOM	Self-organising map	*Pong*	McGlinchey (2003)
SOM & Backpropagation (LM)	Self-organising map & multi-layer perceptron	*Quake II*	Thurau et al. (2003)
Evolutionary algorithm	Single-layer perceptron	*Cellz*	Lucas (2004)
	Multi-layer perceptron	Simulated racing	Togelius and Lucas (2005)
	Multi-layer perceptron	Simulated racing	Togelius and Lucas (2006)
	Rule-base	*Wargus*	Ponsen and Spronck (2004)
Genetic algorithm	Single-layer perceptron	*Xpilot*	Parker et al. (2005b)
	Multi-layer perceptron	*Dead end*	Yannakakis et al. (2004)
	Multi-layer perceptron	*FlatLand*	Yannakakis et al. (2003)

Source: [Galway et al., 2008]

ULB

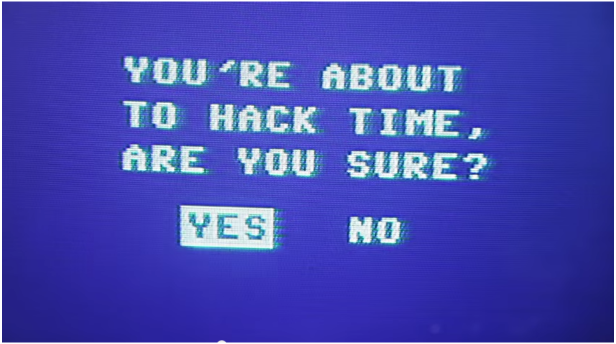
Wrap-up

- ML is not magic, but heavily relying on:
  - Linear algebra
  - Statistics
- Data, and its structure is as important (if not more) than the model
- Data preprocessing can be as time consuming as parameter estimation / model selection
- The usage of ML in videogames is not only restricted to agents



UNIVERSITÉ LIBRE DE BRUXELLES

Since we are at an hackerspace...



- Deepmind Lab
- Mario AI
- OpenAI Gym

Thank you for your attention! Any questions/comments?

Bibliography I

References

Gianluca Bontempi. Statistical foundations of machine learning.

Leo Galway, Darryl Charles, and Michaela Black. Machine learning in digital games: a survey. *Artificial Intelligence Review*, 29(2): 123–161, 2008.

Edoardo Giacomello, Pier Luca Lanzi, and Daniele Loiacono. Doom level generation using generative adversarial networks. In *2018 IEEE Games, Entertainment, Media Conference (GEM)*, pages 316–323. IEEE, 2018.

Bibliography II

Kun Shao, Zhentao Tang, Yuanheng Zhu, Nannan Li, and Dongbin Zhao. A survey of deep reinforcement learning in video games. *arXiv preprint arXiv:1912.10944*, 2019.