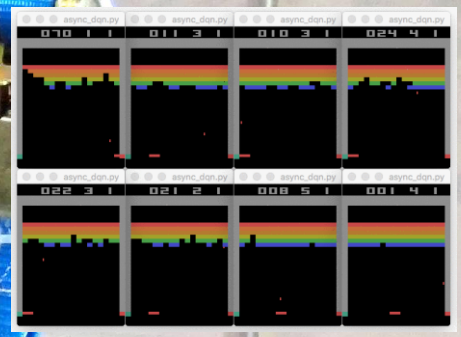
uc3m | Universidad Carlos III de Madrid

ROBOTICSLAB
Where technology happens



Introducción a la Inteligencia Artificial en Robótica

Prof. Juan G Victores

RoboticsLab – Universidad Carlos III de Madrid

@jgvictores

http://roboticslab.uc3m.es

jcgvicto@ing.uc3m.es

SUMMER SCHOOL CEA Comité Español de Automática ALMA

@jgvictores

Introducción a la Inteligencia Artificial en Robótica

- ❑ **Una taxonomía de la IA**
- ❑ **Machine Learning**

  Supervised Learning; Unsupervised Learning; Reinforcement Learning; aún más vía Deep Learning (p.ej. Selfsupervised Learning, Style Transfer, GANs…)
- ❑ **Robótica: Expert Rules hasta End-to-End Learning**
- ❑ **Conclusiones y Recursos**

@jgvictores

Artificial Intelligence

- Artificial General Intelligence (AGI)
- Narrow AI
  - Optimization (Gradient Descent, Evol. Comput...)
  - Good Old-Fashioned AI (Cognitivist/Symbolical)
  - Machine Learning (Connectionist/Statistical)

@jgvictores

Machine Learning (Tipos de Problemas)

- **Supervised Learning**
- **Unsupervised Learning**
- **Reinforcement Learning**
- **Aún más vía Deep Learning**
  - **(p.ej. Selfsupervised Learning, Style Transfer, GANs…)**

@jgvictores

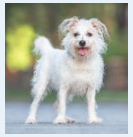
Machine Learning (Tipos de Problemas)

- **Supervised Learning**
- Unsupervised Learning
- Reinforcement Learning
- Aún más vía Deep Learning
  - (p.ej. Selfsupervised Learning, Style Transfer, GANs…)

@jgvictores

Supervised Learning (Tipo de Problema)

Supuesto: Tenemos muchos datos en formato pares {entrada, salida}

{  , “gato”}	{  , “gato”}	{  , “murciélago”}
{  , “perro”}	{  , “gato”}	{  , “perro”}

Consigue: Dada nueva entrada (p.ej. ) *estima* su correspondiente salida (p.ej. “perro”)

Supervised Learning (Tipo de Problema)

- **Classification: salida = categoría(s)**
- **Regression: salida = valor(es) numéricos**

1) Subtipo NO depende de entrada (imagen, sonido, video, datos de sensores…)

2) Vía Deep Neural Networks (2013-): realización simultánea de ambos

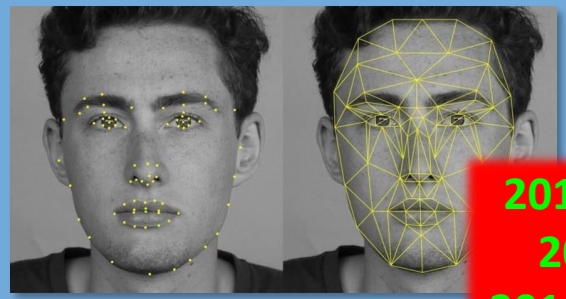
3) Vía Deep Neural Networks (2014-): muchos más tipos de entrada/salida (p.ej. modelos seq2seq admiten secuencias de tokens de lenguaje natural)

Gago, J. J., Vasco, V., Łukawski, B., Pattacini, U., Tikhanoff, V., Victores, J. G., & Balaguer, C. (2019). Sequence-to-sequence natural language to humanoid robot sign language. *arXiv preprint arXiv:1907.04198*.

@jgvictores

Supervised Learning: Classification

entrada = imagen



2012: AlexNet
2014: VGG
2014: Inception
2015: ResNet

FaceID

salida = categoría (usuario)

entrada = audio

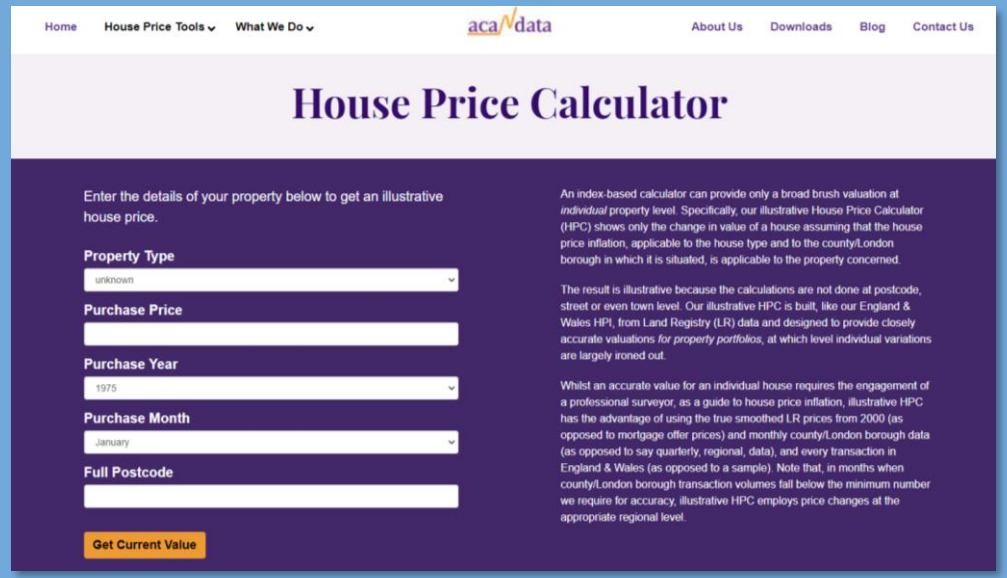
Shazam

salida = categoría (canción)

@jgvictores

Supervised Learning: Regression

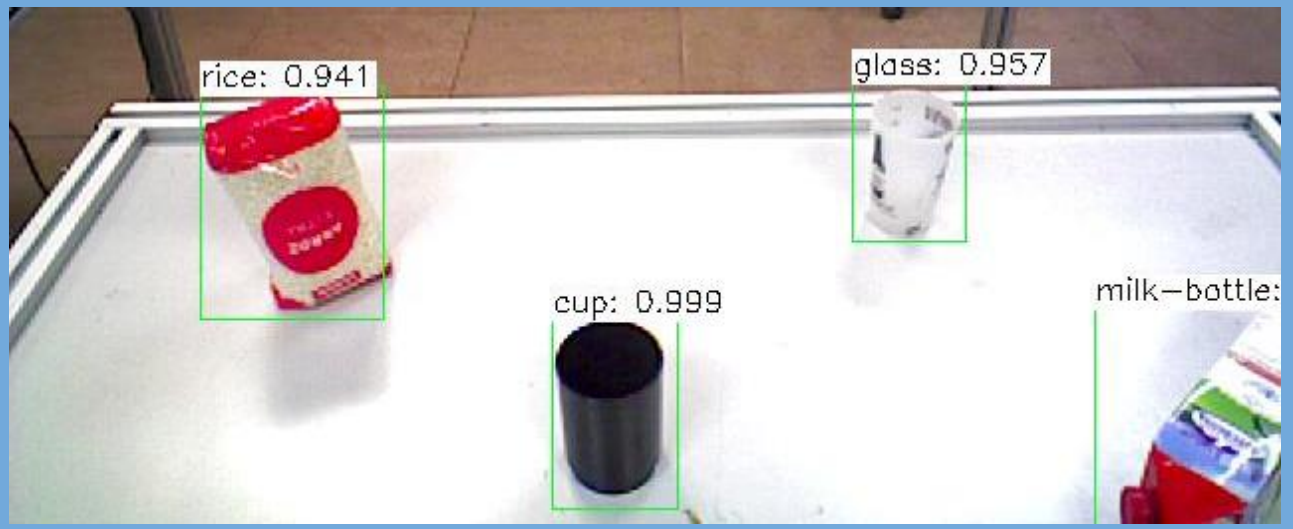
entrada = tipo vivienda, precio y año compra...



salida = valor (precio vivienda actual)

Supervised Learning: Classif.+Regress.

entrada = imagen



2013: R-CNN
2015: SSD
2015: YOLO

https://github.com/roboticslab-uc3m/vision/issues/103 : Robot object Detection

salida = categorías + valores (coordenadas)

@jgvictores

Classification: Técnicas (Modelos+Algortmos)

- Neural Networks (p.ej. Deep Learning)
  - 1949: Hebb
  - 2012: AlexNet
- Logistic Regression
- Decision Trees
- Support Vector Machines (SVM), GMM, K-NN, Naive Bayes, Algebraic Machine Learning…

@jgvictores

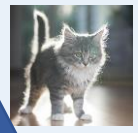
Regression: Técnicas (Modelos+Algoritmos)

- **Neural Networks (p.ej. Deep Learning)**
- **Linear Regression**
- **Alternative versions for regression of Decision Trees, Random Forest, SVM, K-NN, Algebraic Machine Learning…**

@jgvictores

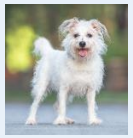
Supervised Learning: Classification

Neural Networks (muy simplificado!)

{  , “gato”}

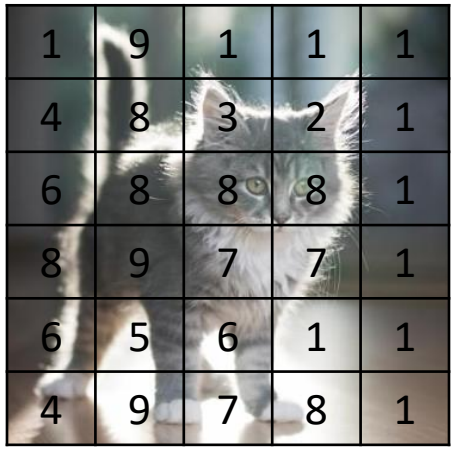
{  , “gato”}

{  , “murciélago”}

{  , “perro”}

{  , “gato”}

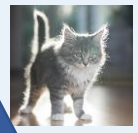
{  , “perro”}



@jgvictores

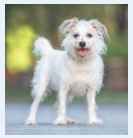
Supervised Learning: Classification

Neural Networks (muy simplificado!)

{  , “gato”}

{  , “gato”}

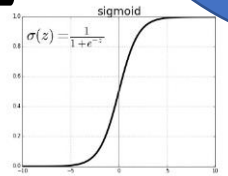
{  , “murciélago”}

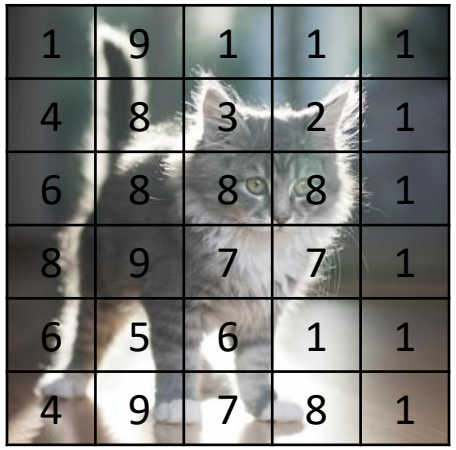
{  , “perro”}

{  , “gato”}

{  , “perro”}

f_activ(





“TRAIN” via “backpropagation”

* (? ? ?) =

model.train()

0	← “murciélago”
1	← “gato”
0	← “perro”
0	← “vaca”
0	← “lechuga”
0	← “oveja”

“one-hot” encoding

@jgvictores

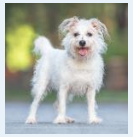
Supervised Learning: Classification

Neural Networks (muy simplificado!)

{  , “gato”}

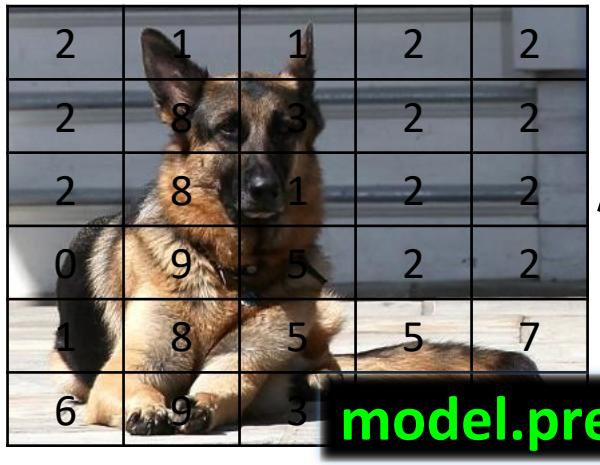
{  , “gato”}

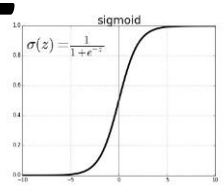
{  , “murciélago”}

{  , “perro”}

{  , “gato”}

{  , “perro”}



f_activ(

****

.1
.7
.2
-.2
-.3
.1

) **=**

	← “murciélago”
	← “gato”
	← “perro”
	← “vaca”
	← “lechuga”
	← “oveja”

model.predict()

“trained weights” do not usually change during “INFERENCE”/ “PREDICTION”

@jgvictores

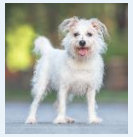
Supervised Learning: Classification

Neural Networks (muy simplificado!)

{  , “gato”}

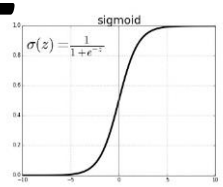
{  , “gato”}

{  , “murciélago”}

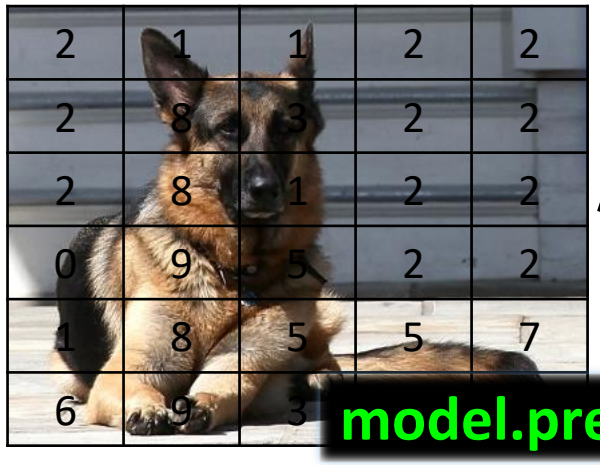
{  , “perro”}

{  , “gato”}

{  , “perro”}



f_activ(



*****

.1
.7
.2
-.2
-.3
.1

) **=**

0	← “murciélago”
.2	← “gato”
.6	← “perro”
.1	← “vaca”
0	← “lechuga”
.1	← “oveja”

model.predict()

“trained weights” do not usually change during “INFERENCE”/ “PREDICTION”

@jgvictores

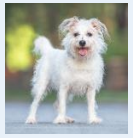
Supervised Learning: Classification

Deep Learning (Deep Neural Networks)

{  , “gato”}

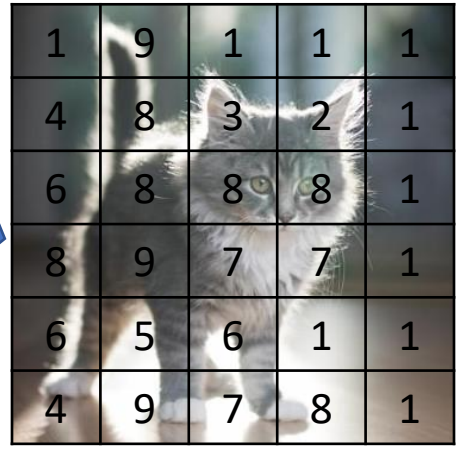
{  , “gato”}

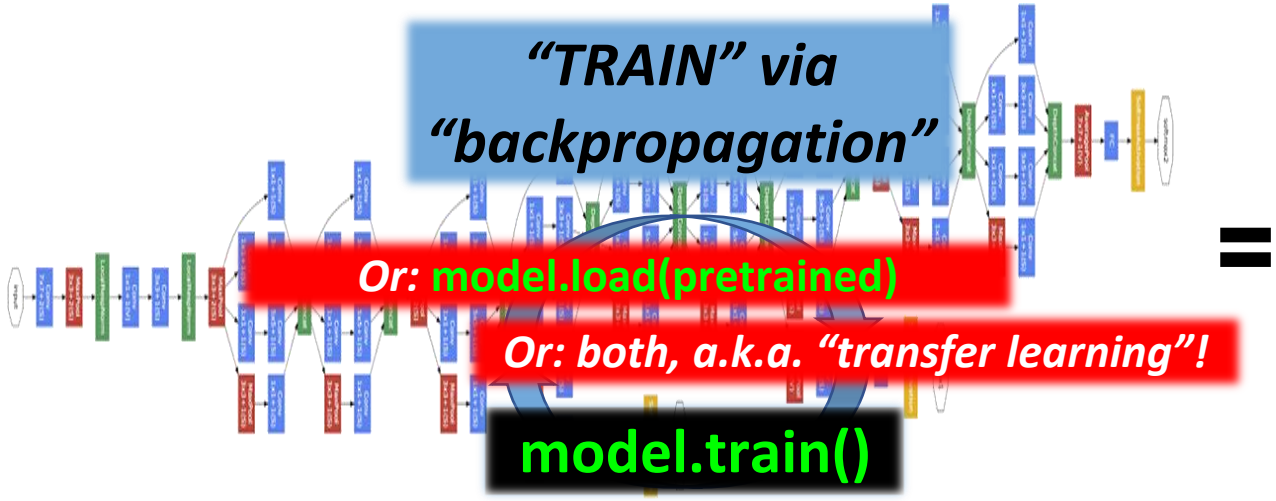
{  , “murciélago”}

{  , “perro”}

{  , “gato”}

{  , “perro”}





=

0	← “murciélago”
1	← “gato”
0	← “perro”
0	← “vaca”
0	← “lechuga”
0	← “oveja”

@jgvictores

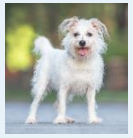
Supervised Learning: Classification

Deep Learning (Deep Neural Networks)

{  , “gato”}

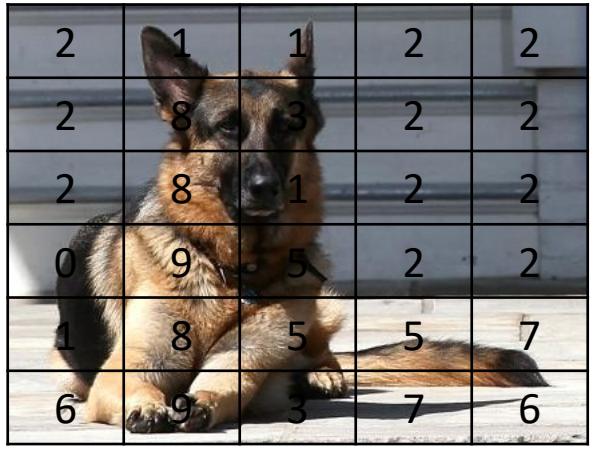
{  , “gato”}

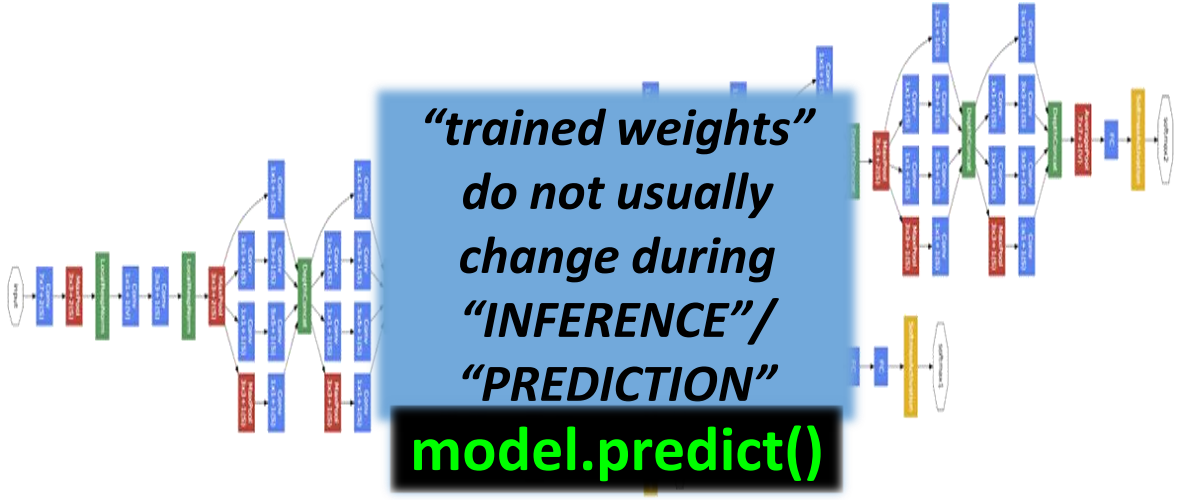
{  , “murciélago”}

{  , “perro”}

{  , “gato”}

{  , “perro”}





=

- “murciélago”
- “gato”
- “perro”
- “vaca”
- “lechuga”
- “oveja”

@jgvictores

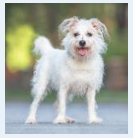
Supervised Learning: Classification

Deep Learning (Deep Neural Networks)

{  , “gato”}

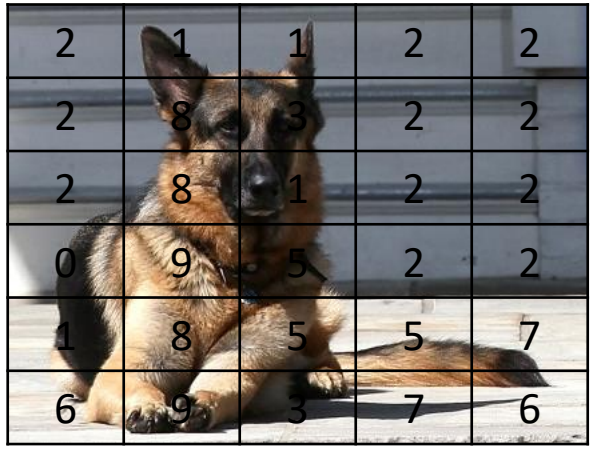
{  , “gato”}

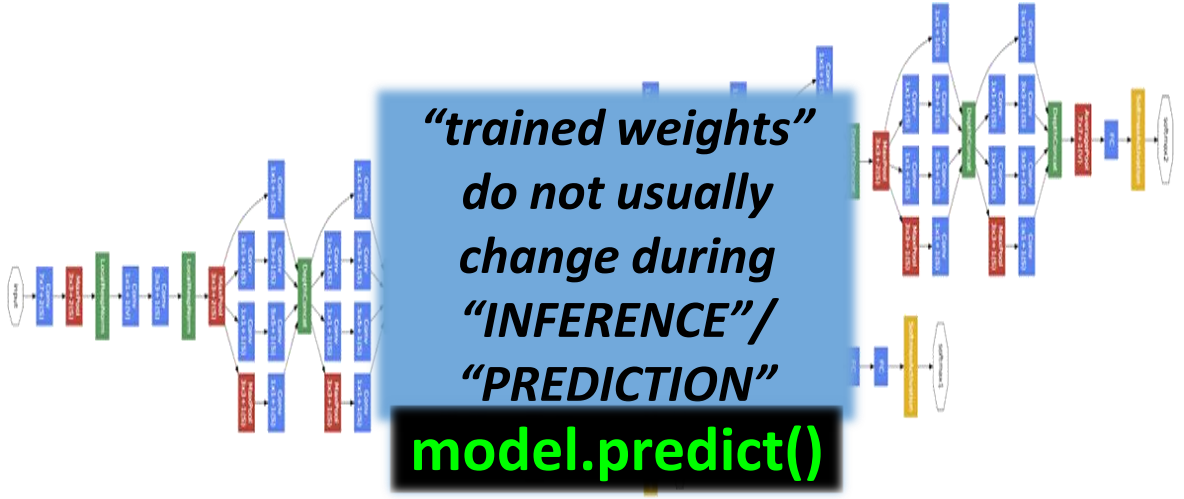
{  , “murciélago”}

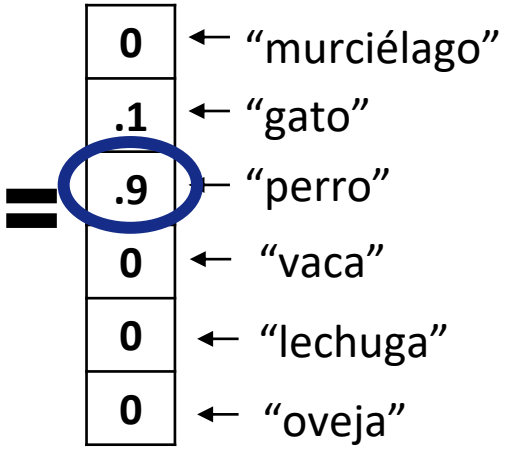
{  , “perro”}

{  , “gato”}

{  , “perro”}







@jgvictores

Machine Learning (Tipos de Problemas)

- **Supervised Learning**
- **Unsupervised Learning**
- **Reinforcement Learning**
- **Aún más vía Deep Learning**
  - **(p.ej. Selfsupervised Learning, Style Transfer, GANs…)**

@jgvictores

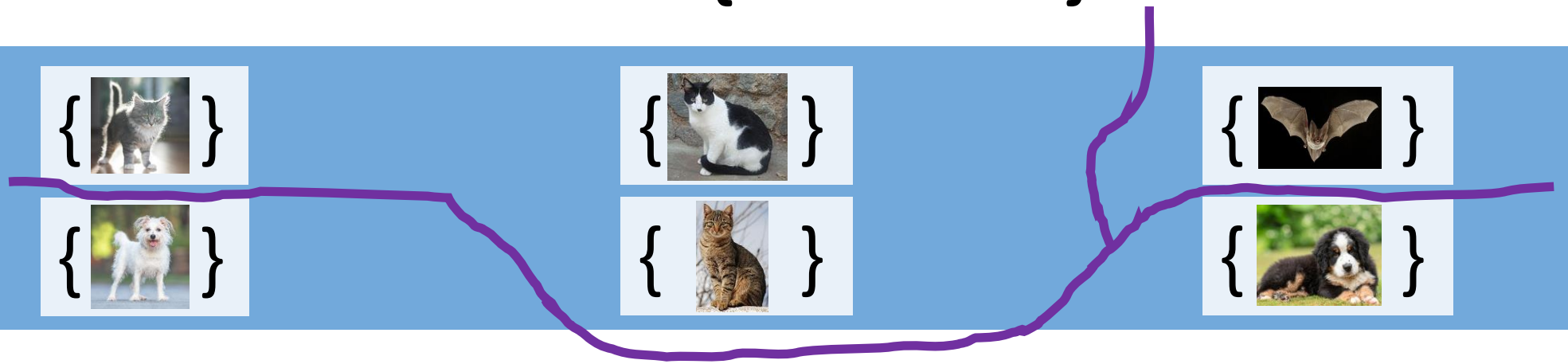
Machine Learning (Tipos de Problemas)

- Supervised Learning
- **Unsupervised Learning**
- Reinforcement Learning
- Aún más vía Deep Learning
  (p.ej. Selfsupervised Learning, Style Transfer, GANs…)

@jgvictores

Unsupervised Learning (Tipo de Problema)

Supuesto: Tenemos muchos datos en un mismo formato {entrada}



Consigue: Variedad de subtipos de problemas, p.ej. **clustering**, dim. reduct, anomaly detect...

@jgvictores

Machine Learning (Tipos de Problemas)

- **Supervised Learning**
- **Unsupervised Learning**
- **Reinforcement Learning**
- **Aún más vía Deep Learning**
  - **(p.ej. Selfsupervised Learning, Style Transfer, GANs…)**

@jgvictores

Machine Learning (Tipos de Problemas)

- Supervised Learning
- Unsupervised Learning
- **Reinforcement Learning**
- Aún más vía Deep Learning
  (p.ej. Selfsupervised Learning, Style Transfer, GANs…)

@jgvictores

Reinforcement Learning (Tipo de Problema)

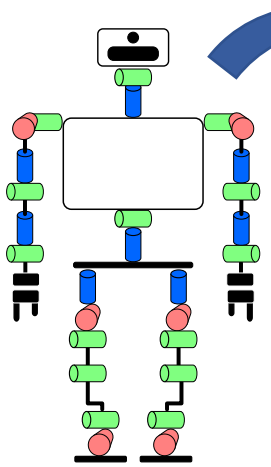
Supuesto: Un **agente** {observa estado del entorno, actúa, y obtiene un refuerzo} (bucle!)

{  , “amar”, 1 }	{  , “amar”, 1}	{  , “huir”, 1}
{  , “amar”, -1}	{  , “amar”, 1}	{  , “amar”, -1}

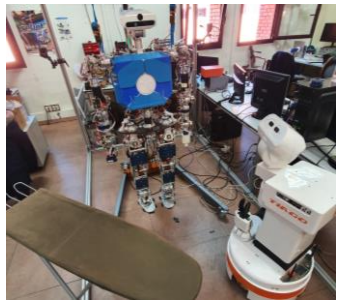
Consigue: Dado nuevo estado (p.ej. ) decide acción (p.ej. “huir”), obtiene refuerzo (1)

@jgvictores

Reinforcement Learning (Tipo de Problema)



Agente



observa estado

acción 1

nuevo estado'
y refuerzo'

acción 2

nuevo estado''
y refuerzo''

Toma de decisiones

Objetivo: ley de control

acción = π(estado)

Maximiza: f(refuerzo)

Supuestos:

Markov Assumption

Markov Decision Process

π directa o vía V/Q

@jgvictores

Reinforcement Learning: Técnicas

Watkins (1989): Q-Learning

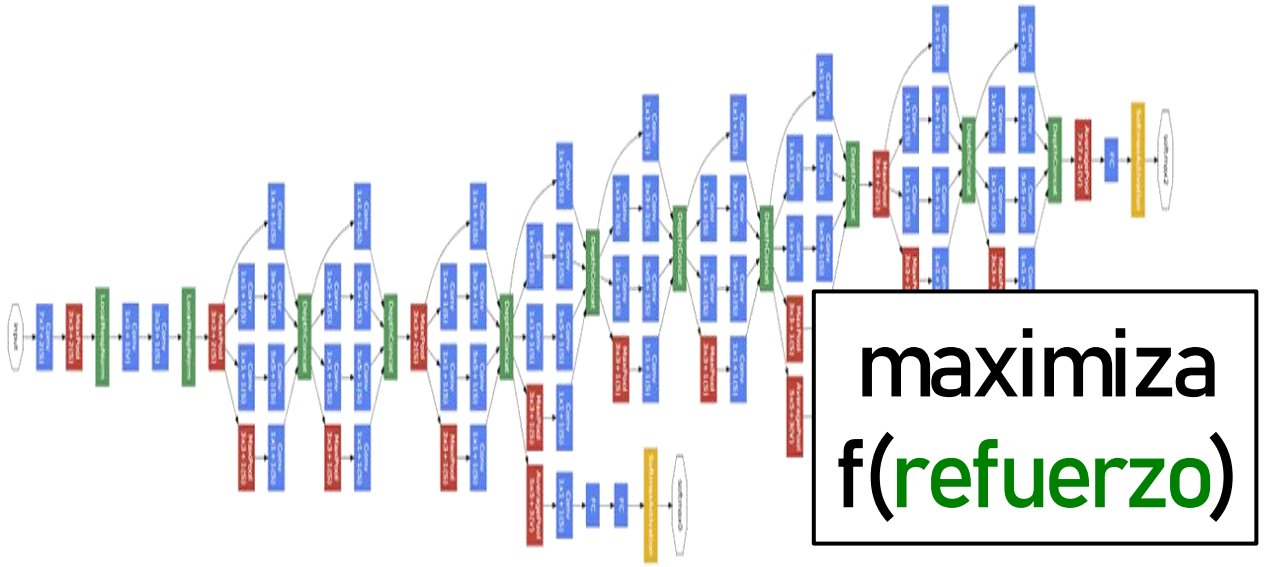
- π(s) / V(s) / Q(s,a) / Actor-Critic
- Tabular / Function Approximator
- f(reward): Average / Discounted / …
- On-policy / Off-policy
- Model-free / Model-based
- MDP / POMDP / …

@jgvictores

Deep Reinforcement Learning (Conjunto de técnicas)

Mnih (2015-): Deep Q-Network (DQN)

entrada = estado (imagen, pose...)



salida = π/V/Q

para acción = π(state)

@jgvictores

Machine Learning (Tipos de Problemas)

- **Supervised Learning**
- **Unsupervised Learning**
- **Reinforcement Learning**
- **Aún más vía Deep Learning**
  - **(p.ej. Selfsupervised Learning, Style Transfer, GANs…)**

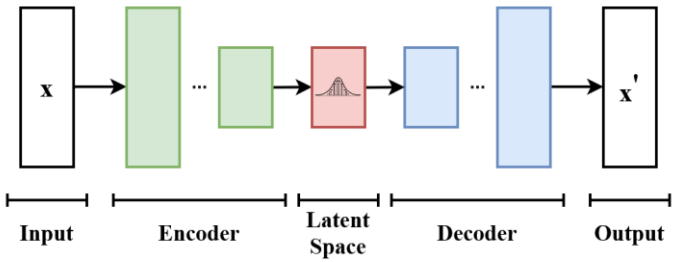
@jgvictores

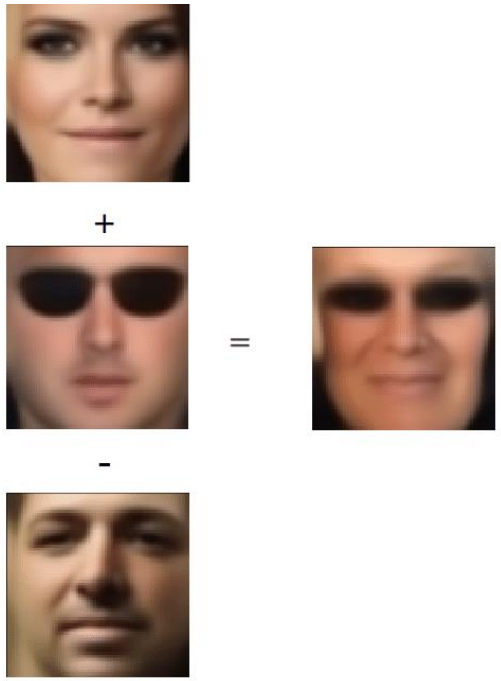
Machine Learning (Tipos de Problemas)

- Supervised Learning
- Unsupervised Learning
- Reinforcement Learning
- **Aún más vía Deep Learning**
  (p.ej. Selfsupervised Learning, Style Transfer, GANs…)

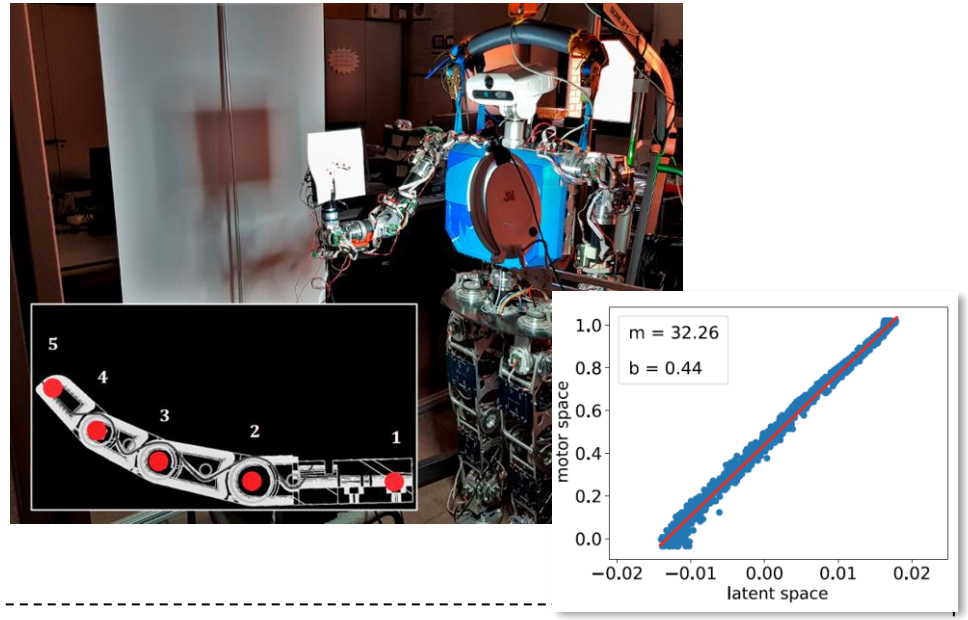
Aún más vía Deep Learning

Selfsupervised Learning: Auto-Encoders, Variational Auto-Encoders (VAE)





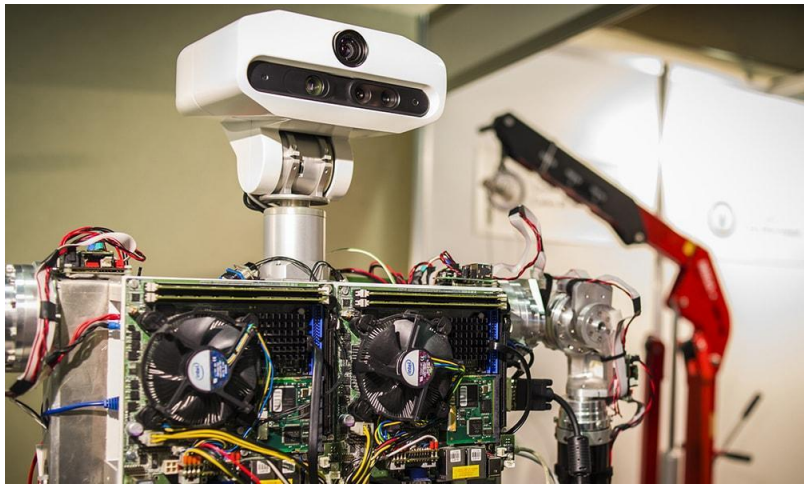
Li, D., & Wang, J. (2018). Image Semantic Transformation: Faster, Lighter and Stronger. *arXiv preprint arXiv:1803.09932*.



Gago, J. J., Łukawski, B., Victores, J. G., & Balaguer, C. (2020, November). Under-Actuation Modelling in Robotic Hands via Neural Networks for Sign Language Representation with End-User Validation. In *Int. Conf. Intelligent Data Engineering and Automated Learning (IDEAL)* (pp. 239-251).

Aún más vía Deep Learning

Style Transfer



Content

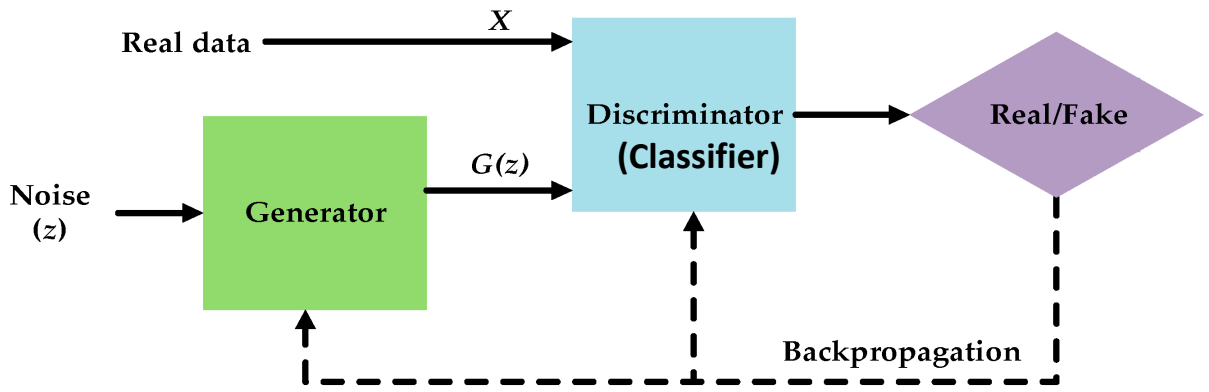
Generated

Style

Raul Fernández Fernández (2021). Action Generalization in Humanoid Robots Through Artificial Intelligence With Learning From Demonstration. Advisors: Carlos Balaguer, Juan G Victores. PhD Thesis, UC3M.

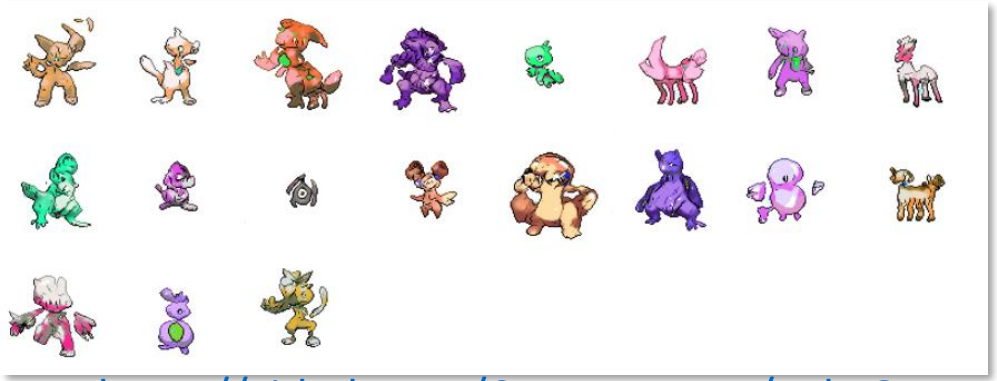
Aún más vía Deep Learning

Generative Adversarial Networks (GANs)



https://bytes860770954.wordpress.com/2020/08/22/what-are-generative-adversarial-networks-gans/

StyleGAN, Karras et al.



https://github.com/ConorLazarou/PokeGAN

@jgvictores

Introducción a la Inteligencia Artificial en Robótica

- ❑ **Una taxonomía de la IA**
- ❑ **Machine Learning**

  Supervised Learning; Unsupervised Learning; Reinforcement Learning; aún más vía Deep Learning (p.ej. Selfsupervised Learning, Style Transfer, GANs…)
- ❑ **Robótica: Expert Rules hasta End-to-End Learning**
- ❑ **Conclusiones y Recursos**

@jgvictores

Introducción a la Inteligencia Artificial en Robótica

- ❑ Una taxonomía de la IA
- ❑ Machine Learning

  Supervised Learning; Unsupervised Learning; Reinforcement Learning; aún más vía Deep Learning (p.ej. Selfsupervised Learning, Style Transfer, GANs…)
- ❑ **Robótica: Expert Rules hasta End-to-End Learning**
- ❑ **Conclusiones y Recursos**

@jgvictores

Robótica: Expert Rules hasta End-to-End

- Component Based Software Engineering (CBSE)
  - Sistemas Expertos (Rule-Based: IF-ELSE)
  - Planificación, Cinemática, Control
  - Learning w/hand-crafted Features
  - All Learned Features
- End-to-End (p.ej. Via DRL)
  - Hyperparameters
    - Hand-crafted
    - Automated search

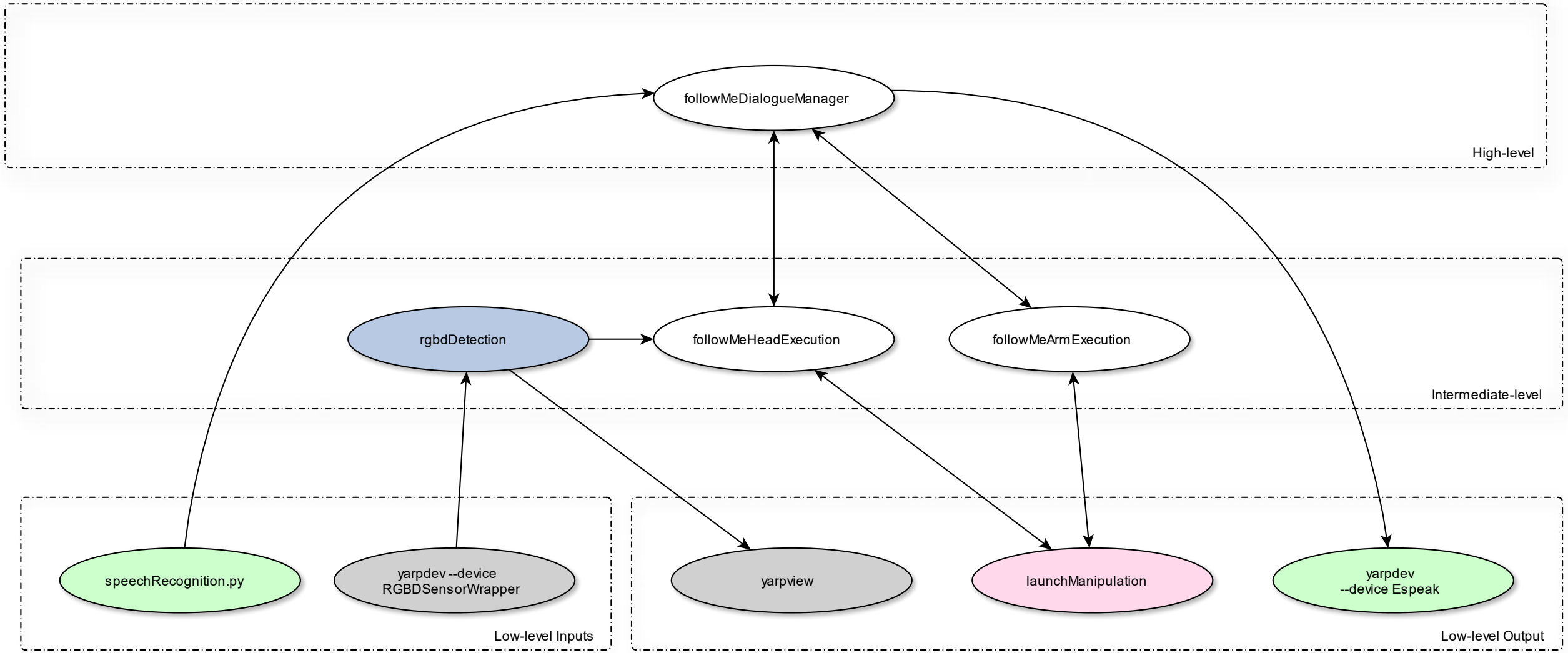
Expert (Coding)

"generic/ agnostic" Algorithms

Learning (Data+Model +Algorithms)

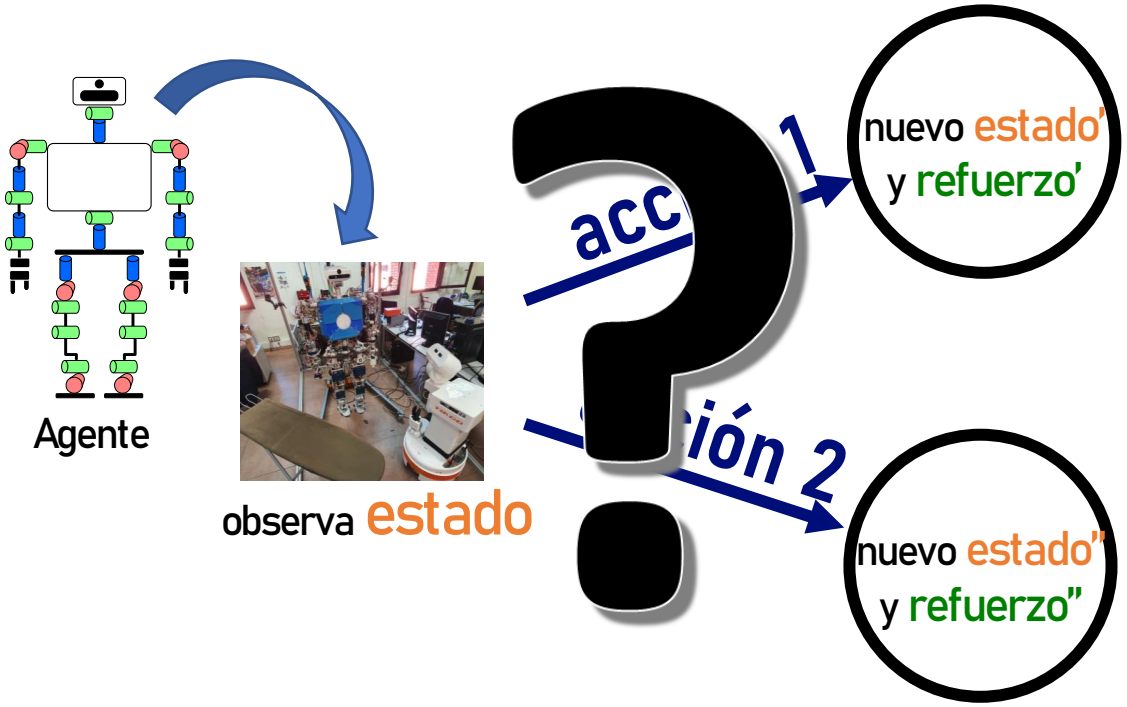
@jgvictores

Robótica: CBSE (rules & generic & learning)

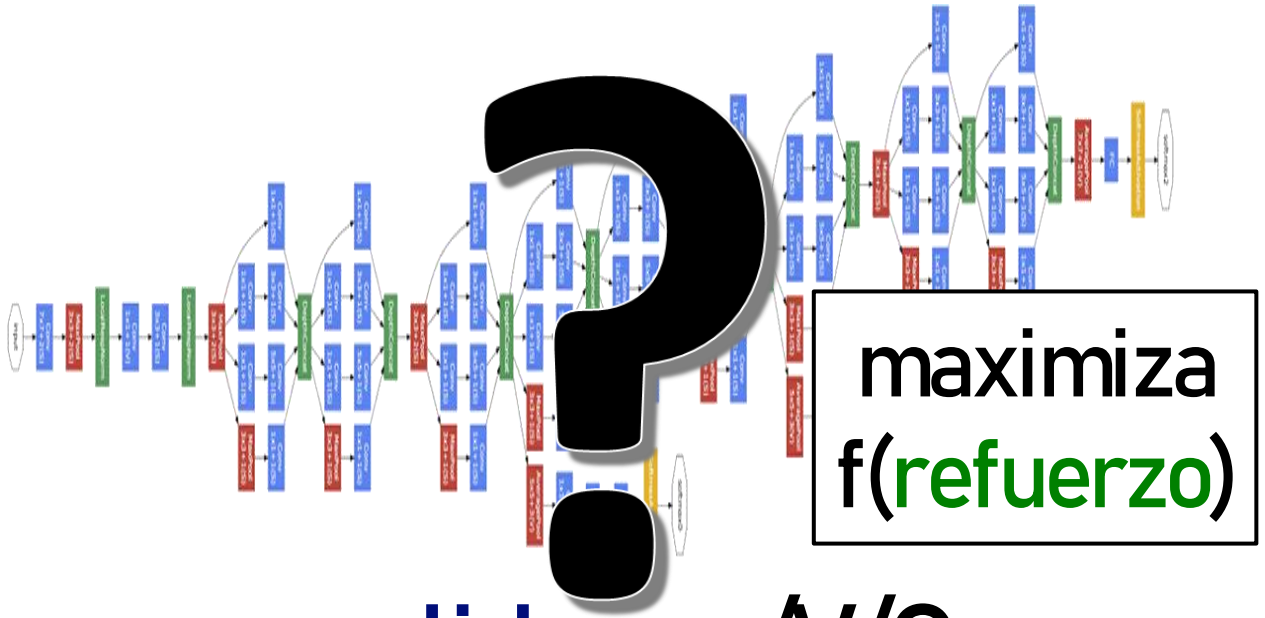


@jgvictores

Robótica: End-to-End (p.ej. Via DRL 2015-)



entrada = estado (imagen, pose...)



salida = π/V/Q

para acción = π(state)

@jgvictores

Robótica: Expert Rules hasta End-to-End

- Component Based Software Engineering (CBSE)
  - Sistemas Expertos (Rule-Based: IF-ELSE)
  - Planificación, Cinemática, Control
  - Learning w/hand-crafted Features
  - All Learned Features
- End-to-End (p.ej. Via DRL)
  - Hyperparameters
  - Hand-crafted
  - Automated search

Expert (Coding)

"generic/ agnostic" Algorithms

Learning (Data+Model +Algorithms)

@jgvictores

Recursos (libros, cursos, proyectos)

Guili, “Deep Learning with TensorFlow 2 and Keras”, 2019

https://www.coursera.org/specializations/deep-learning

Sutton, Barto, “Reinforcement Learning: An Introduction”, 2018

https://www.coursera.org/specializations/reinforcement-learning

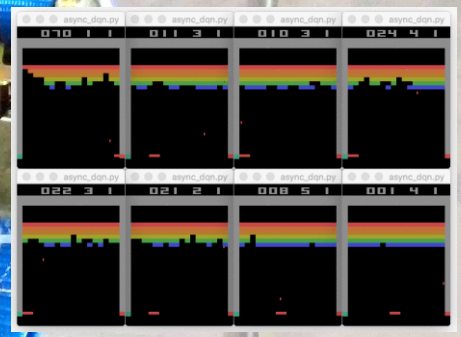
http://rail.eecs.berkeley.edu/deeprlcourse

Máster Universitario en Robótica y Automatización (UC3M)
(p.ej. Planificación, Aprendizaje, Simuladores…)

ALMA: Human Centric Algebraic Machine Learning (H2020-EIC-FETPROACT-2019)

uc3m | Universidad Carlos III de Madrid

ROBOTICSLAB
Where technology happens



Introducción a la Inteligencia Artificial en Robótica

Prof. Juan G Victores

RoboticsLab – Universidad Carlos III de Madrid

@jgvictores

http://roboticslab.uc3m.es

jcgvicto@ing.uc3m.es

SUMMER SCHOOL CEA Comité Español de Automática ALMA