



PROGRAMACIÓN II - UNIDAD 5

Ing. Gastón Weingand (gaston.weingand@uai.edu.ar)

Agenda

1. Descripción y ejemplos de algunas de las librerías más utilizadas: Pygame, RE, Collections, NumPy, SQLAlchemy, Request, Pillow, etc.
2. Introducción al machine learning.
 - A. Introducción al Deep learning.
 - i. Algoritmos de redes neuronales.
 - ii. Práctica de generación de red neuronal sobre placa microbit y/o simulación.
 - B. Framework Anaconda y miniconda

Agenda

1. Descripción y ejemplos de algunas de las librerías más utilizadas: Pygame, RE, Collections, NumPy, SQLAlchemy, Request, Pillow, etc.
2. Introducción al machine learning.
 - A. **Introducción al Deep learning.**
 - i. Algoritmos de redes neuronales.
 - ii. Práctica de generación de red neuronal sobre placa microbit y/o simulación.
 - B. Framework Anaconda y miniconda, descripción

Artificial Intelligence

Reasoning

Natural
Language
Processing
(NLP)

Planning

Machine Learning

Supervised
Learning

Unsupervised
Learning

Reinforcement
Learning

Deep Learning
• Neural Networks

APRENDIZAJE SUPERVISADO



APRENDIZAJE POR REFUERZO

DETERMINA QUÉ ACCIONES DEBE ESCOGER UN AGENTE DE SOFTWARE EN UN ENTORNO DADO CON EL FIN DE MAXIMIZAR ALGUNA NOCIÓN DE "RECOMPENSA" O PREMIO ACUMULADO



APRENDIZAJE NO SUPERVISADO



Agenda

1. Descripción y ejemplos de algunas de las librerías más utilizadas: Pygame, RE, Collections, NumPy, SQLAlchemy, Request, Pillow, etc.
2. Introducción al machine learning.
 - A. Introducción al Deep learning.
 - i. Algoritmos de redes neuronales.
 - ii. Práctica de generación de red neuronal sobre placa microbit y/o simulación.
 - B. Framework Anaconda y miniconda, descripción y uso práctico.

Tipos y algoritmos de redes neuronales

Perceptrón (Funciones separables linealmente, binaria)

Red ADALINE/MADALINE (Similar al perceptrón pero analógica)













Perceptrón multicapa (Algoritmo de descenso de gradiente/Backpropagation)

- Codificación de información
- Traducción de texto en lenguaje hablado
- Reconocimiento óptico de caracteres (OCR)

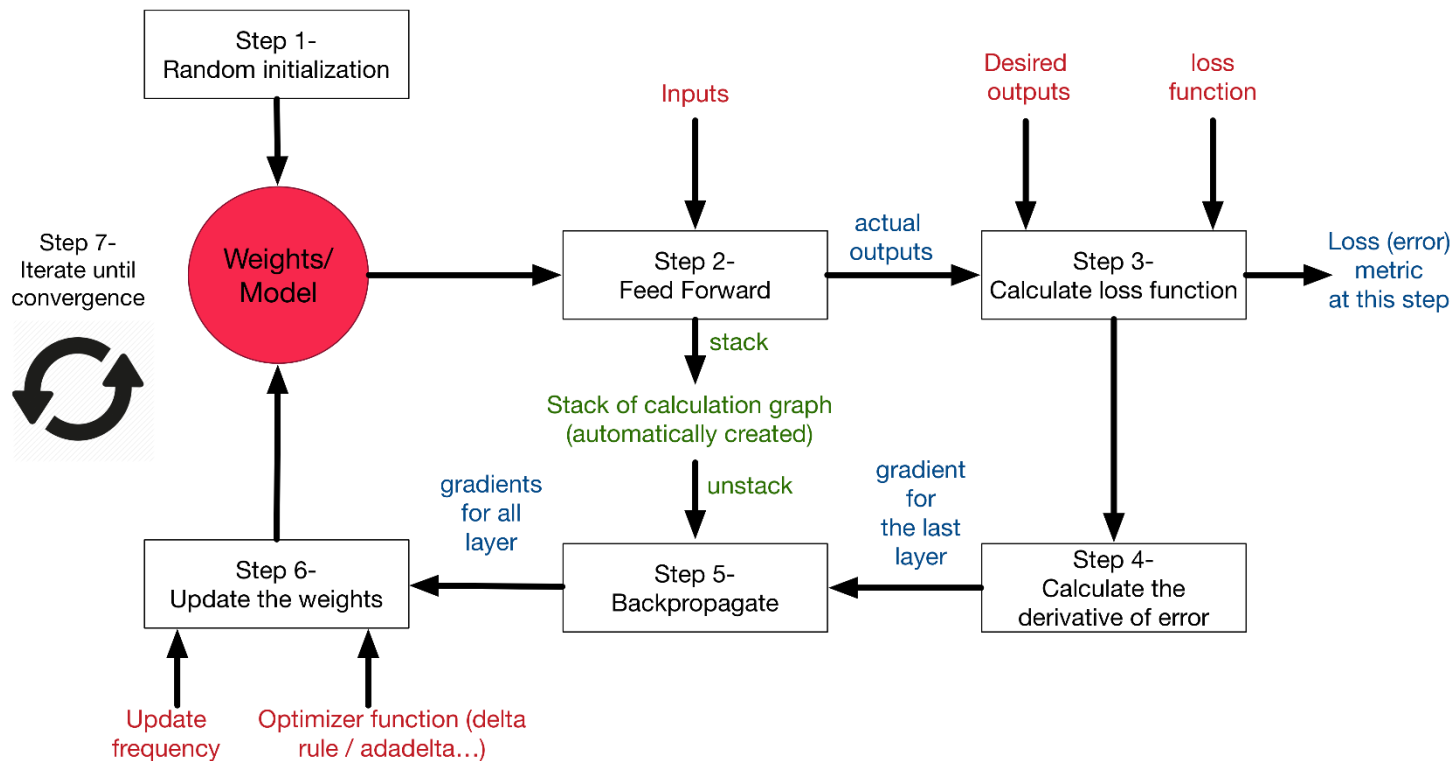
Redes neuronales convolucionales (Imagen, video)

Algoritmos genéticos (Selección, cruza, mutaciones) combinados con redes neuronales

Tipos y algoritmos de redes neuronales

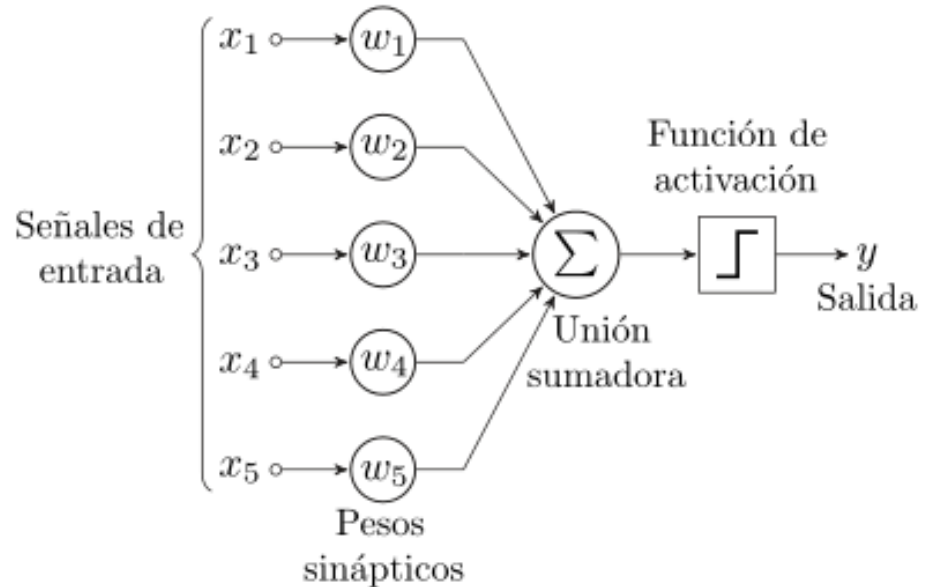
Estructura	Regiones de Decisión	Problema de la XOR	Clases con Regiones Mezcladas	Formas de Regiones más Generales
1 Capa 	Medio Plano Limitado por un Hiperplano			
2 Capas 	Regiones Cerradas o Convexas			
3 Capas 	Complejidad Arbitraria Limitada por el Número de Neuronas			

Tipos y algoritmos de redes neuronales



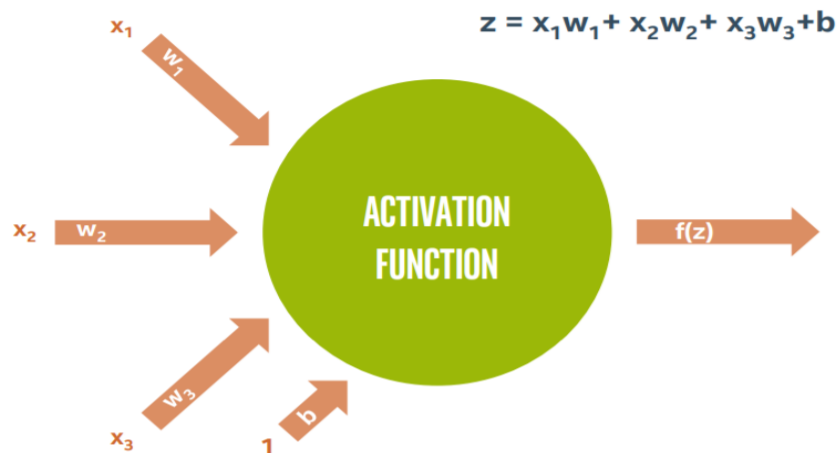
PERCEPTRON

El **Perceptron** simple, también conocido una red neuronal de una sola capa (Single-Layer Neural Network), es un algoritmo de clasificación binaria creado por [Frank Rosenblatt](#) a partir del modelo neuronal de Warren McCulloch y Walter Pitts desarrollado en 1943.



PERCEPTRON

BASIC NEURON VISUALIZATION



z = "net input"

b = "bias term"

f = activation function

a = output to next layer

$$z = b + \sum_{i=1}^m x_i w_i$$

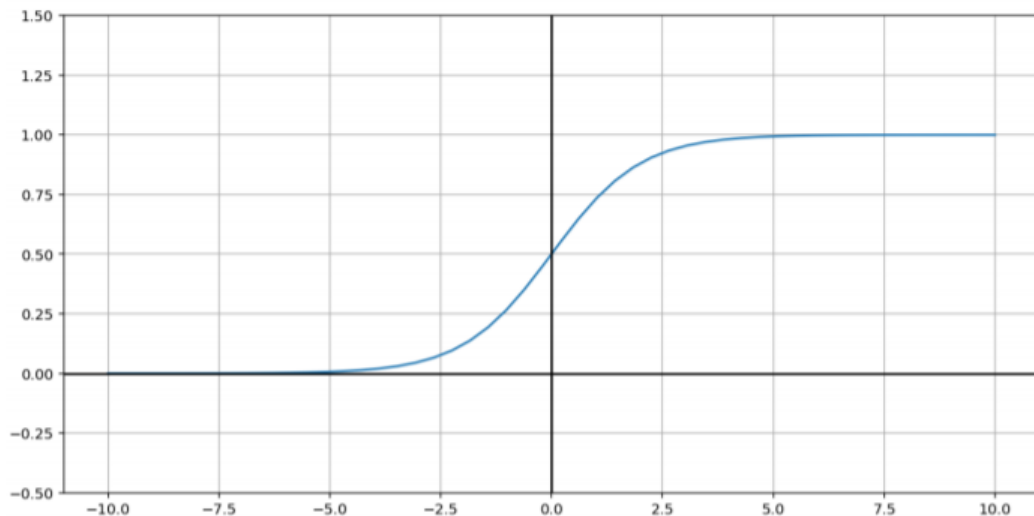
$$z = b + x^T w$$

$$a = f(z)$$

PERCEPTRON

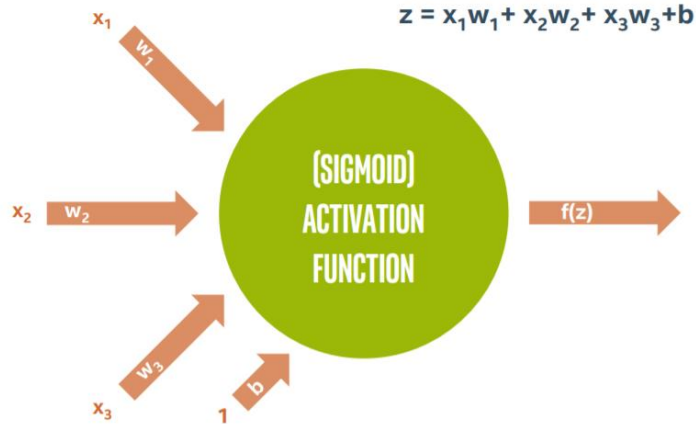
RELATION TO LOGISTIC REGRESSION

This is called the “sigmoid” function: $\sigma(z) = \frac{1}{1 + e^{-z}}$

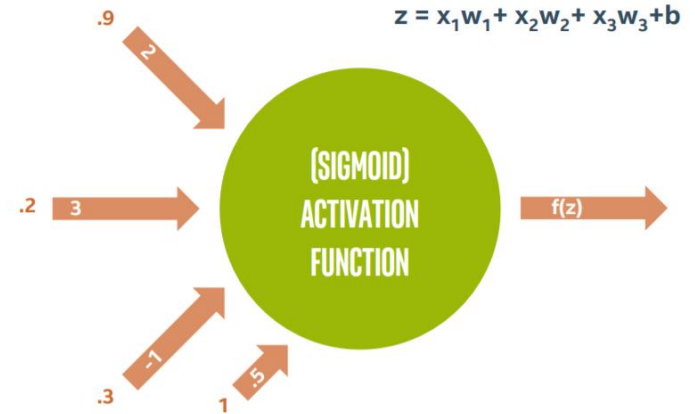


PERCEPTRON

EXAMPLE NEURON COMPUTATION

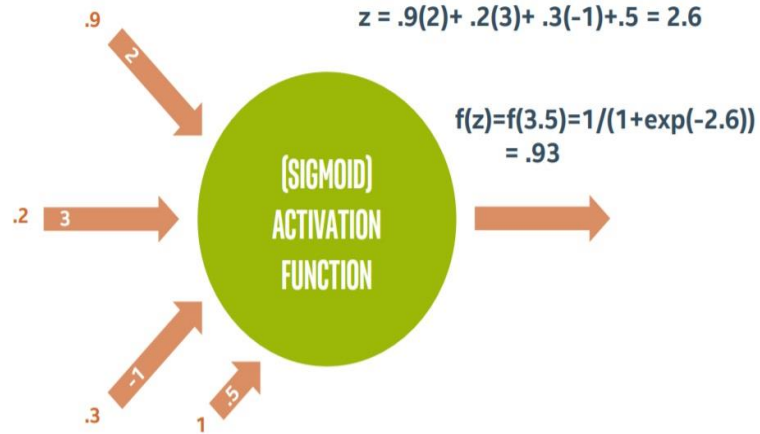


EXAMPLE NEURON COMPUTATION

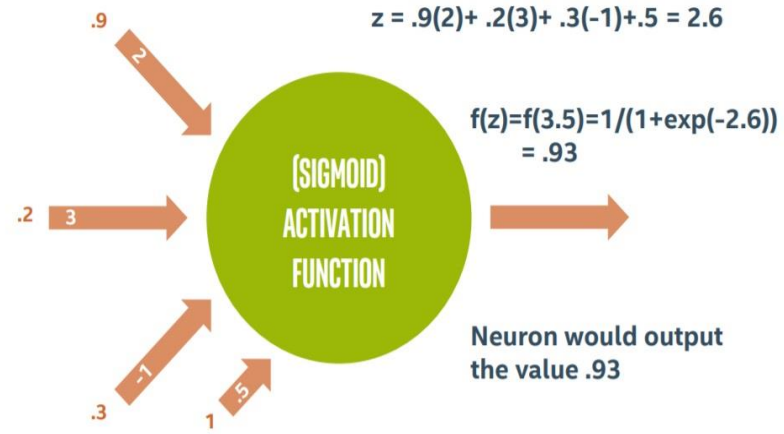


PERCEPTRON

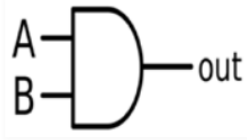
EXAMPLE NEURON COMPUTATION



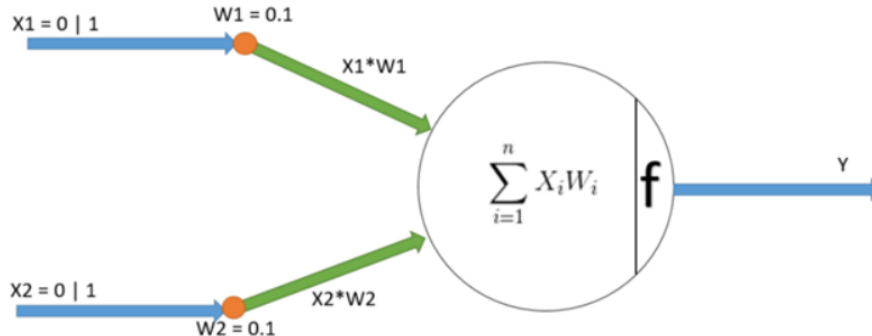
EXAMPLE NEURON COMPUTATION



“APRENDIENDO UNA COMPUERTA AND”



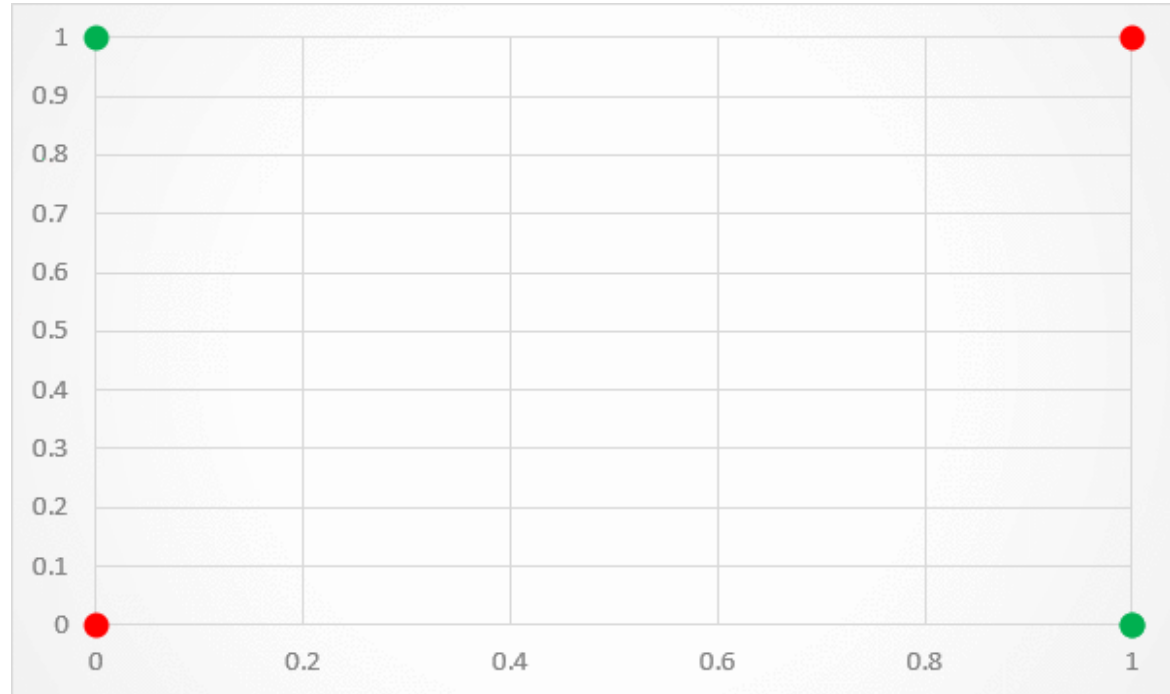
X1	X2	Y
0	0	0
0	1	0
1	0	0
1	1	1



Redes neuronales

El problema XOR

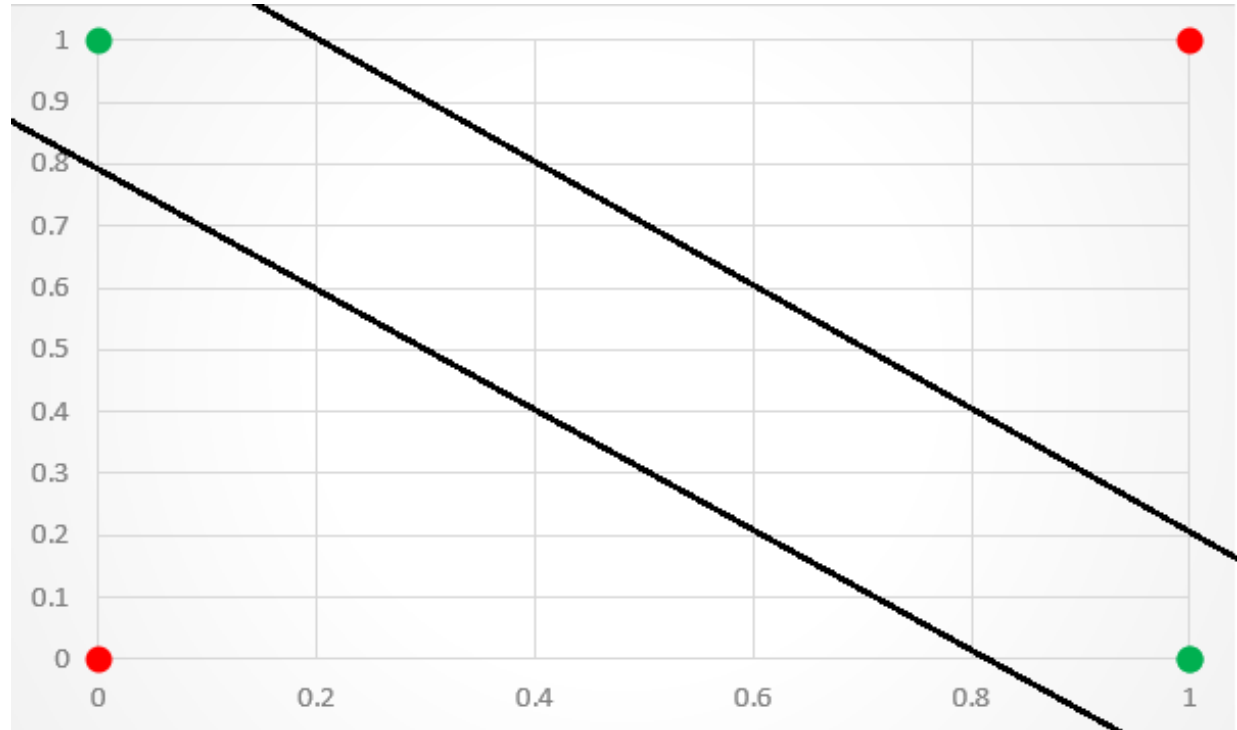
Input 1	Input 2	Output
0	0	0
0	1	1
1	1	0
1	0	1



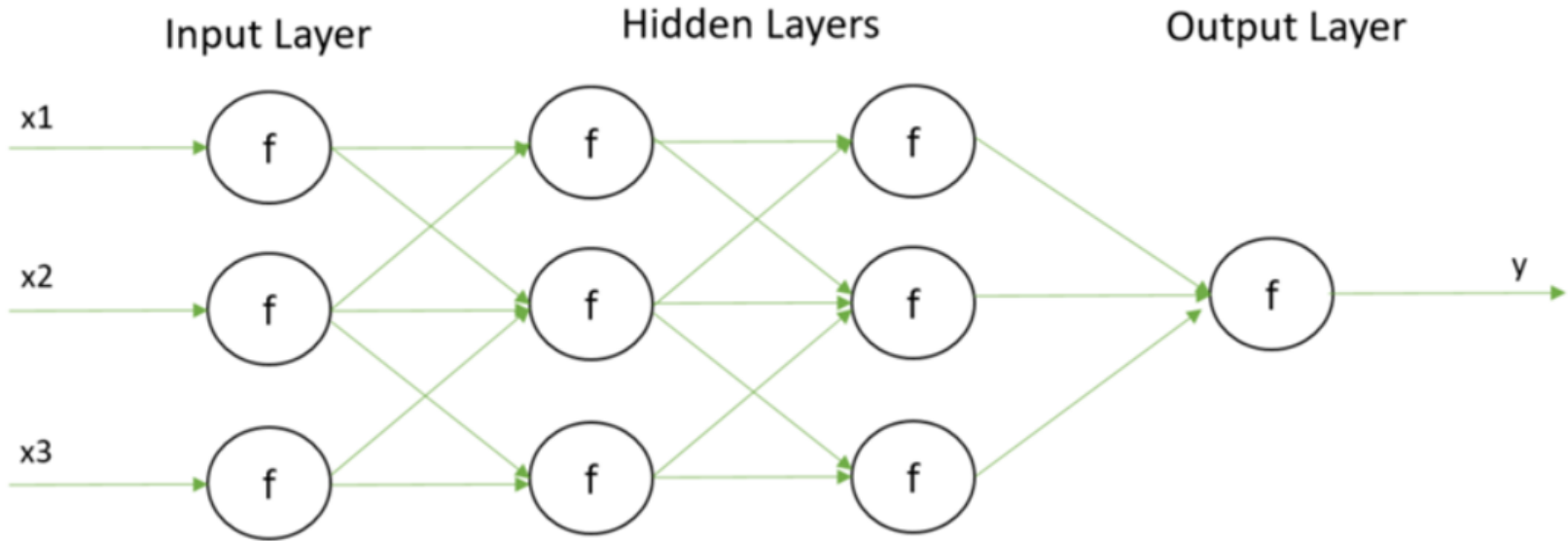
Redes neuronales

El problema XOR

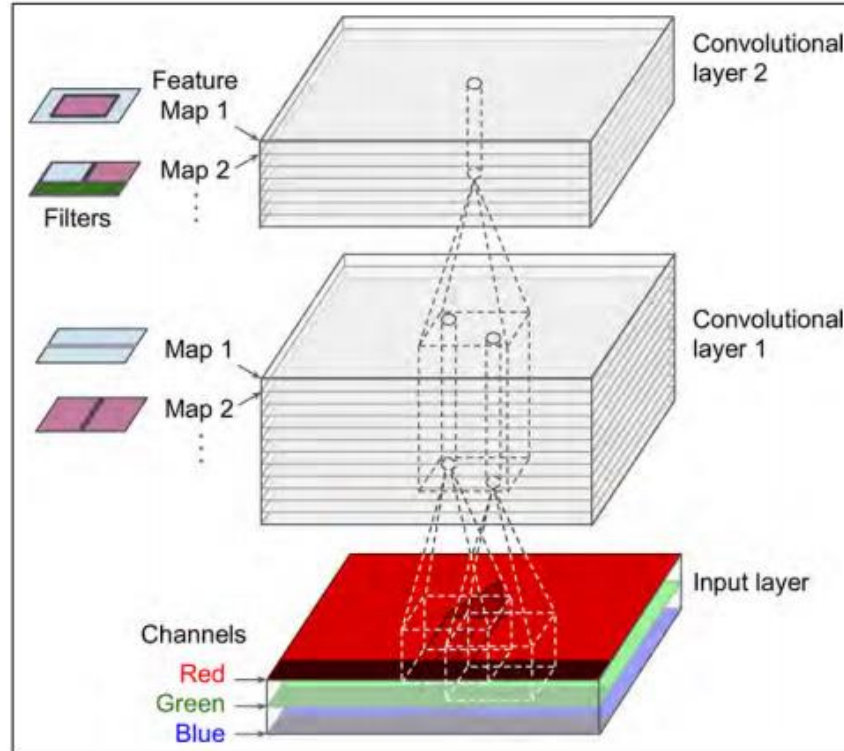
Input 1	Input 2	Output
0	0	0
0	1	1
1	1	0
1	0	1



Redes neuronales multicapa



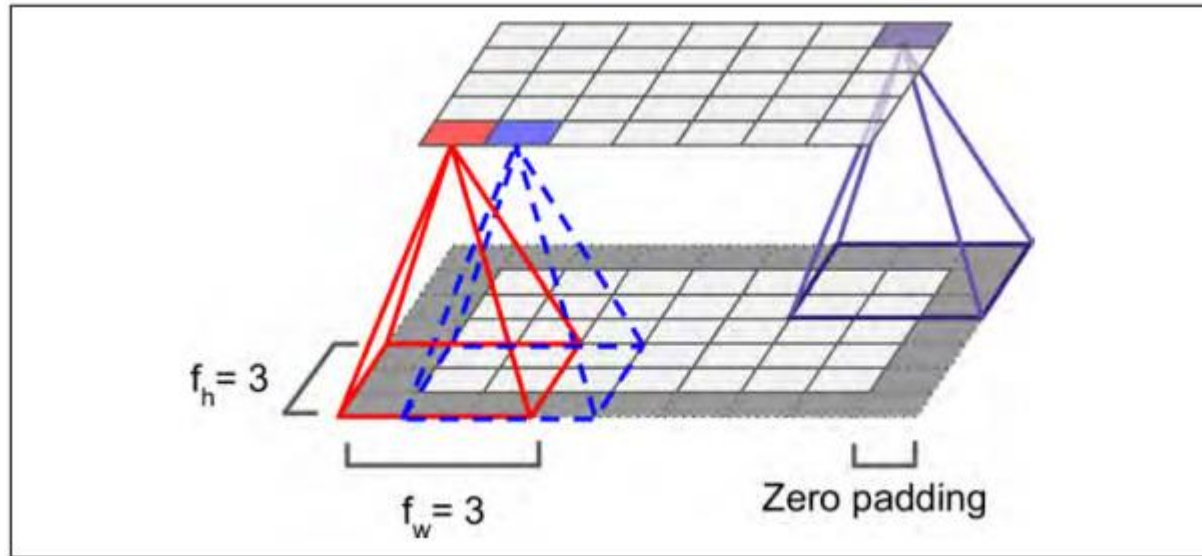
Convolutional Neural Networks



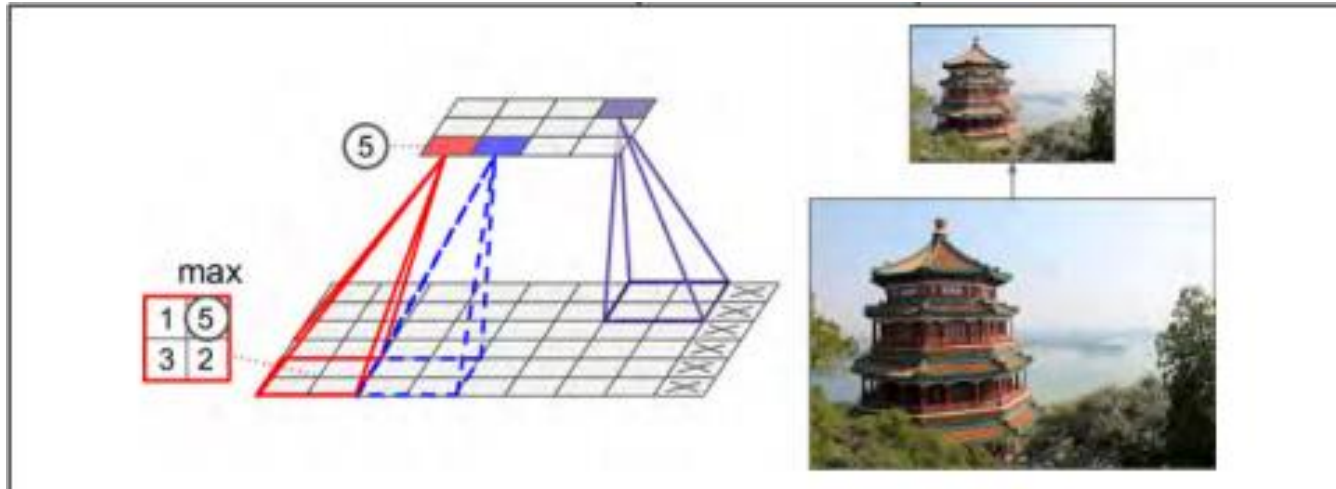
Reconociendo imágenes:
Tomado de la
Arquitectura
de la corteza visual

Convolutional Neural Networks

Reducción de capas usando strides

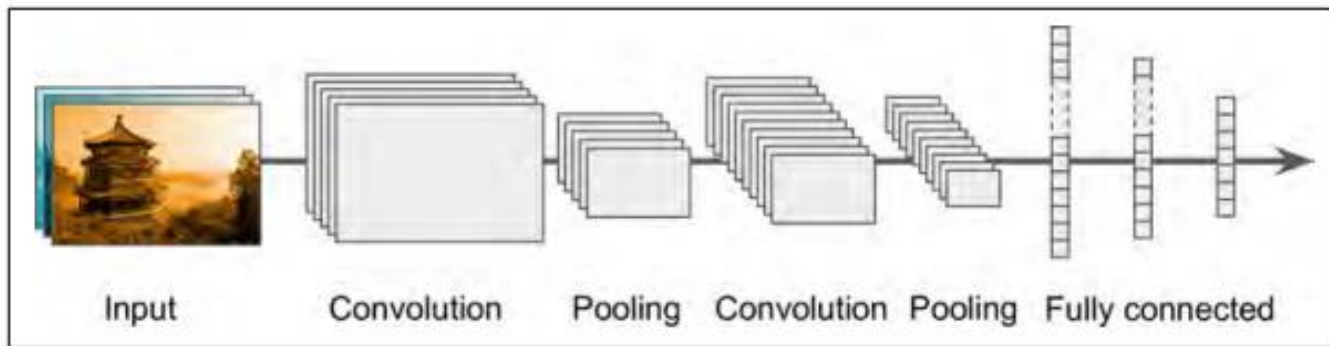


Convolutional Neural Networks



Pooling layer (Achicar imágenes)

Convolutional Neural Networks



Arquitectura CNN tradicional

LeNet-5

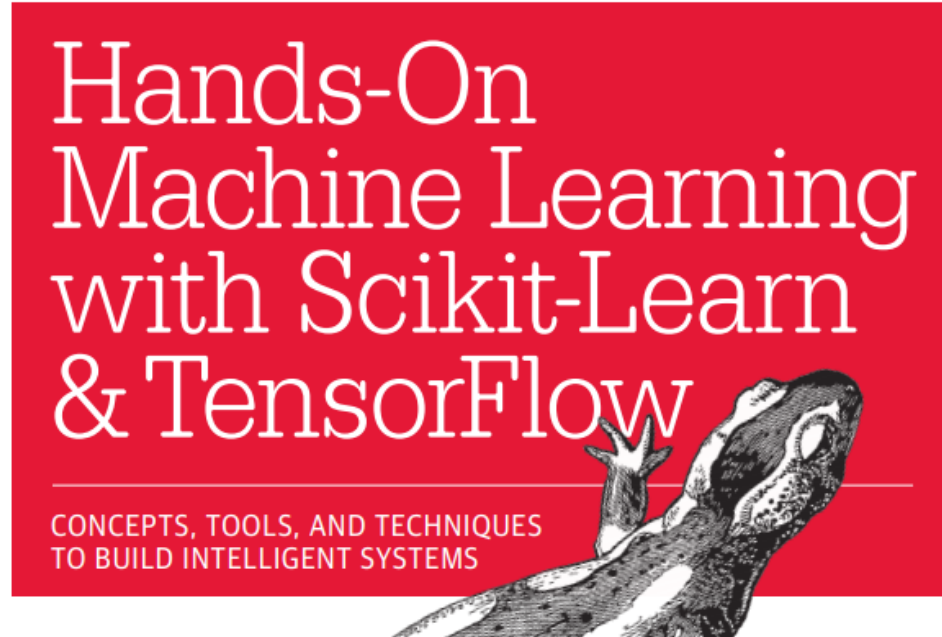
AlexNet

GoogLeNet

ResNet

Convolutional Neural Networks

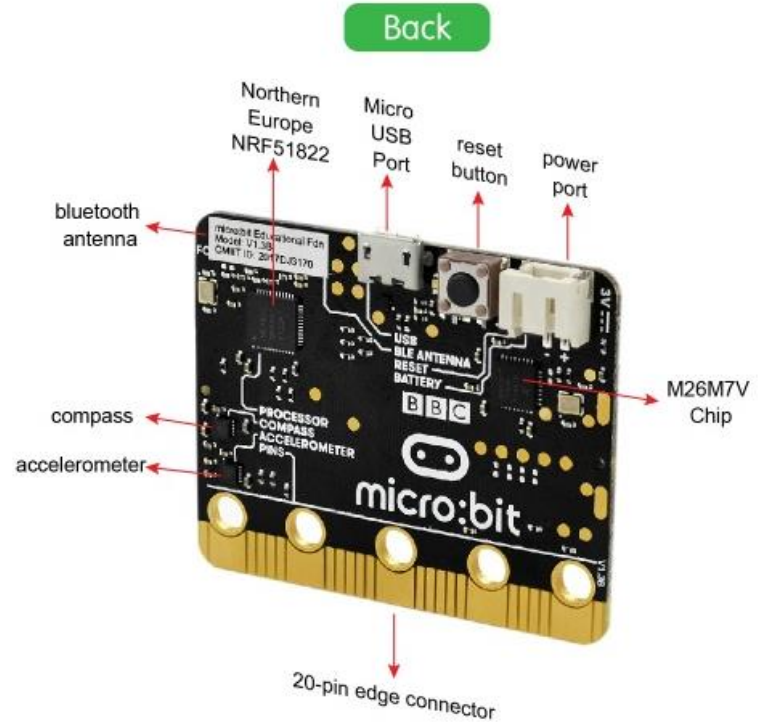
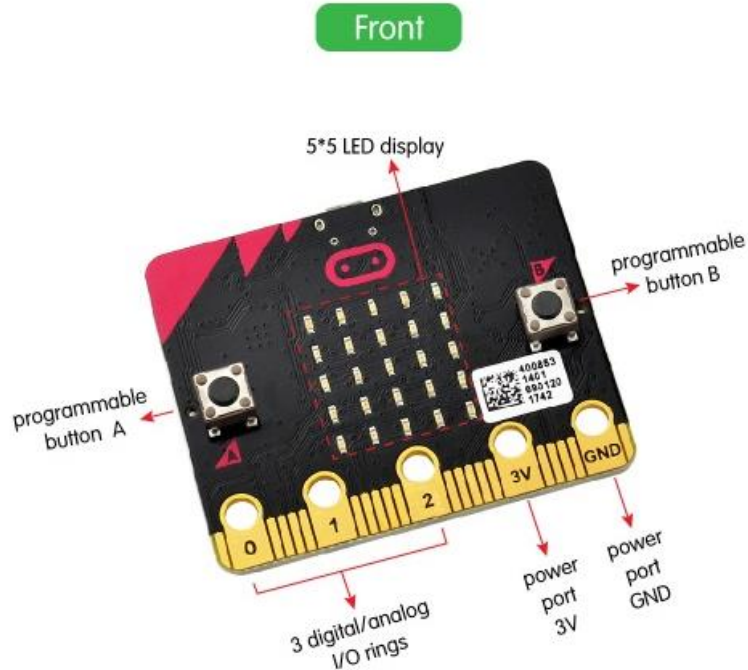
Libro recomendado



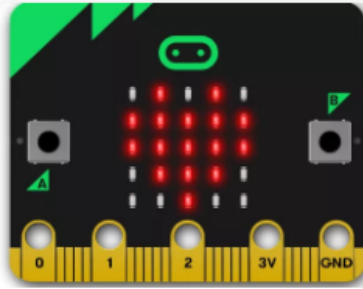
Agenda

1. Descripción y ejemplos de algunas de las librerías más utilizadas: Pygame, RE, Collections, NumPy, SQLAlchemy, Request, Pillow, etc.
2. Introducción al machine learning.
 - A. Introducción al Deep learning.
 - i. Algoritmos de redes neuronales.
 - ii. Práctica de generación de red neuronal sobre placa microbit y/o simulación.
 - B. Framework Anaconda y miniconda, descripción y uso práctico.

Microbit

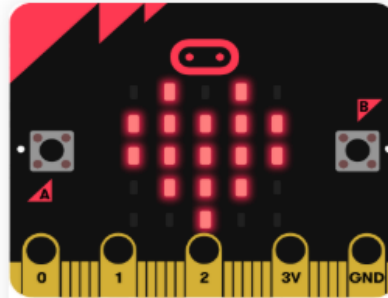


Microbit



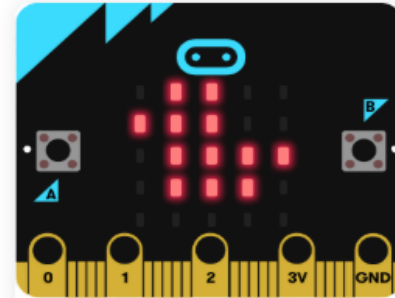
Heart

Light up your micro:bit with love by showing a heart



Beating heart

Make your micro:bit's heart beat using loops



Animated animals

Animate your own animals on the micro:bit display

Guía para microbit (Python)

<https://microbit.org/get-started/user-guide/python/>

Microbit

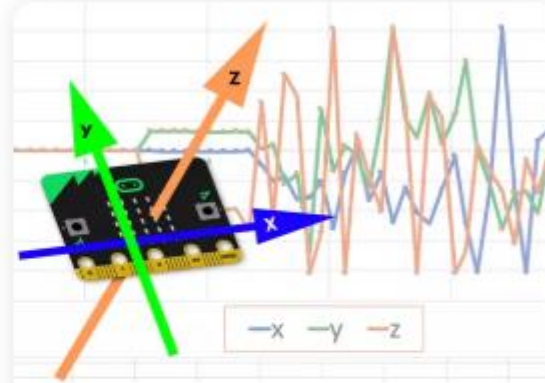


Sensitive step counter

Make a step-counter tailored to your walking style



Intermediate



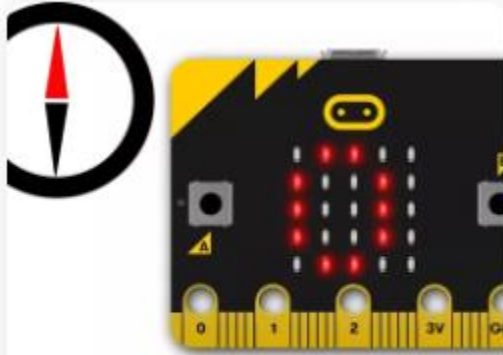
Python wireless data logger

Create a wireless data logger with Python



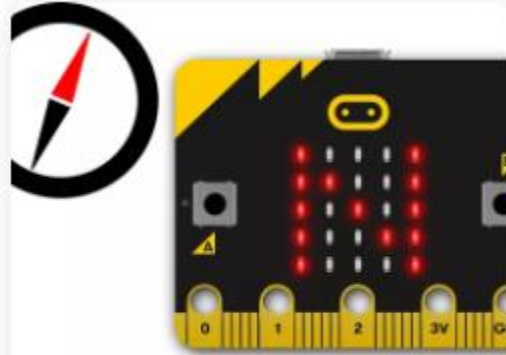
Advanced

Microbit



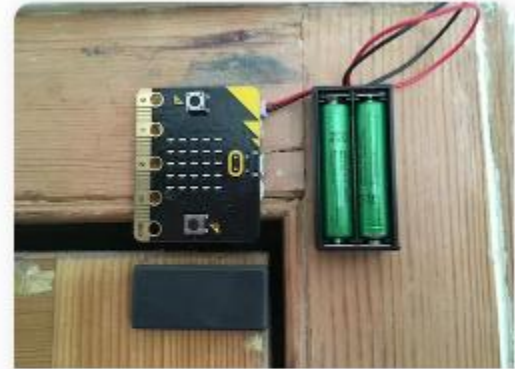
Compass bearing

Turn your micro:bit into a simple compass



Compass North

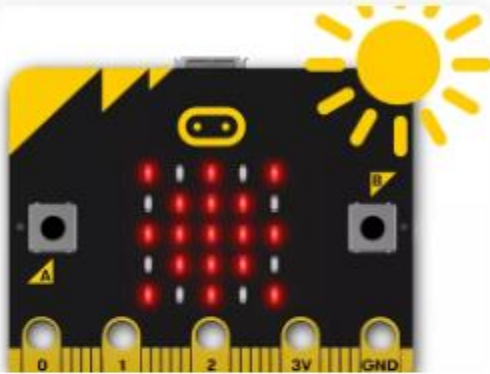
Create a simple compass to show which way is North



Door alarm

Make your own micro:bit wireless door alarm

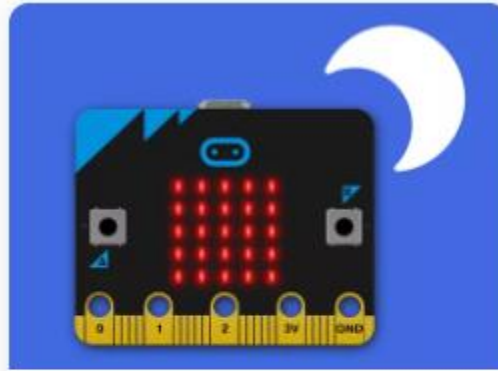
Microbit



Sunlight sensor

Make your micro:bit light up when the sun comes up

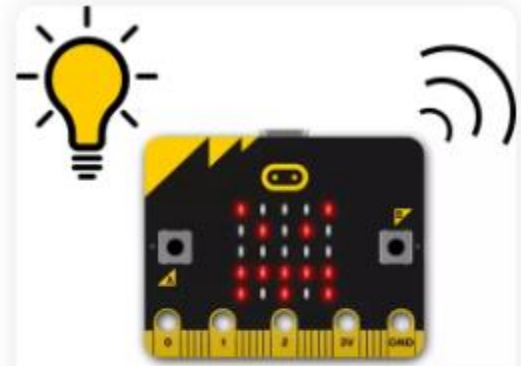
■ □ □ Beginner



Nightlight

Create a light that turns on when it's dark

■ ■ □ Intermediate

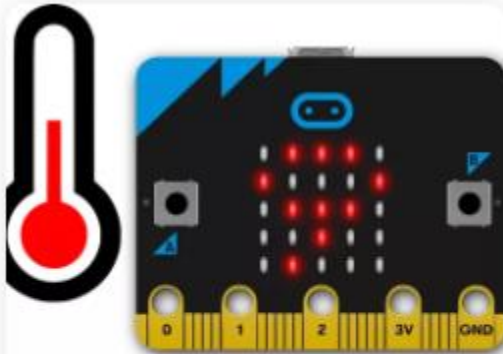


Light alarm

Make an alarm that goes off when lights go on

■ ■ ■ Advanced

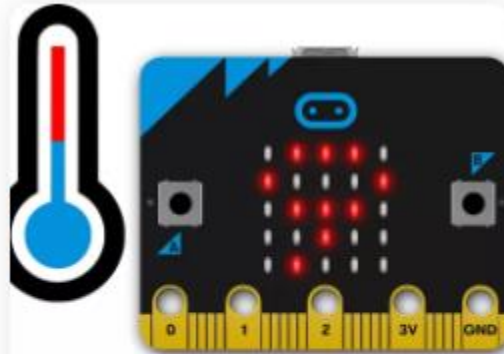
Microbit



Thermometer

Make a simple thermometer with your micro:bit

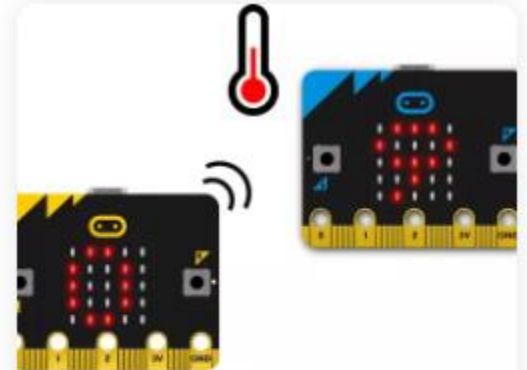
■ ■ ■ Beginner



Max-min thermometer

Track high and low temperatures with your micro:bit

■ ■ ■ Intermediate

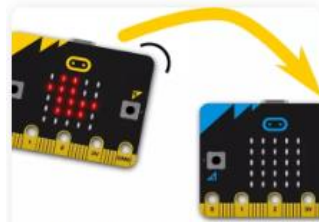


Indoor-outdoor thermometer

Sense how warm or cold it is outside

■ ■ ■ Intermediate

Microbit



Teleporting duck

Teleport a duck between micro:bits using radio

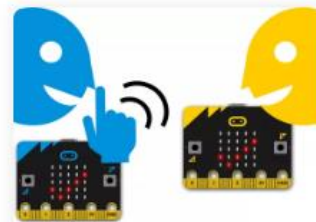
■ ■ ■ Intermediate



Group teleporting duck

Teleport ducks in a multi-player game

■ ■ ■ Intermediate



Tell me a secret

Swap secrets with a friend using radio

■ ■ ■ Intermediate

Music



Microbit



https://wiki.keyestudio.com/Ks0356_keyestudio_Micro:bit_Mini_Smart_Robot_Car

Agenda

1. Descripción y ejemplos de algunas de las librerías más utilizadas: Pygame, RE, Collections, NumPy, SQLAlchemy, Request, Pillow, etc.
2. Introducción al machine learning.
 - A. Introducción al Deep learning.
 - i. Algoritmos de redes neuronales.
 - ii. Práctica de generación de red neuronal sobre placa microbit y/o simulación.
 - B. Framework Anaconda y miniconda, descripción.

2.B Framework Anaconda y miniconda

Anaconda es una distribución libre y abierta de los lenguajes Python y R, utilizada en ciencia de datos, y aprendizaje automático (machine learning). Esto incluye procesamiento de grandes volúmenes de información, análisis predictivo y cómputos científicos. Está orientado a simplificar el despliegue y administración de los paquetes de software.

<https://www.anaconda.com/>

<https://www.anaconda.com/distribution/#download-section>

2.B Framework Anaconda y miniconda



ANACONDA NAVIGATOR

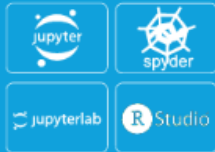
Desktop Portal to Data Science

ANACONDA PROJECT

Portable Data Science Encapsulation

DATA SCIENCE LIBRARIES

Data Science IDEs



Analytics & Scientific Computing



Visualization



Machine Learning



...and many more!



Data Science Package & Environment Manager

2.B Framework Anaconda y miniconda

En lugar de instalar Anaconda, pueden instalar **Miniconda**, que es una versión pequeña de Anaconda que solo incluye Python, pip y conda (y otros paquetes necesarios para su funcionamiento).

¿Qué son pip y conda? Son "package managers", que sirven para bajar e instalar paquetes automáticamente, entre otras cosas.

¿Por qué instalaría Miniconda en lugar de Anaconda? Hay varias razones posibles:

- Por cuestiones de espacio en disco: en lugar de instalar todo, pueden instalar solo lo que necesitan.
- Para aprovechar y aprender a instalar paquetes, por si alguna vez necesitan uno que Anaconda no incluya. Por ejemplo, [Lantz](#), que sirve para controlar instrumentos

2.B Framework Anaconda y miniconda, descripción y uso práctico.

