



XUNTA
DE GALICIA

CENTRO DE
FORMACIÓN E
RECURSOS DE FERROL

Inteligencia Artificial para la Sociedad



Sara Guerreiro (UDC)

Francisco Bellas (UDC)

Alejandro Romero (UDC)

Octubre de 2025



Organización del curso

1. Sesión 1: introducción a la IA y aspectos ético-legales
2. Sesión 2: percepción y actuación
3. Sesión 3: representación y razonamiento
4. Sesión 4: aprendizaje automático 1
- 5. Sesión 5: aprendizaje automático 2**
6. Sesión 6: herramientas de IA para la educación (IA generativa)



Sesión 5:

Aprendizaje automático 2



Actividad: Entrenamiento de modelos

**Data Mining
Fruitful and Fun**

Open source machine learning and data visualization.

[Download Orange 3.38.0](#)

<https://orangedatamining.com/download/>



Actividad: Entrenamiento de modelos

orange
DATA MINING

[Screenshots](#) [Download](#) [Blog](#) [Docs](#) [Workshops](#) [Donate](#)

Search



Documentation

Visual Programming

[Getting started](#)
[YouTube tutorials](#)
[Example workflows](#)
[Widget catalog](#)
[Loading your data](#)

Development

[Widget development](#)
[Example addon](#)

Python Library

[Tutorial](#)
[Reference](#)
[Orange 2.7 documentation](#)

For a list of frequently asked questions, see [FAQ](#). Also, feel free to reach out to us in our [Discord chatroom](#).

<https://orangedatamining.com/docs/>



Actividad: Entrenamiento de modelos

orange
DATA MINING

[Screenshots](#) [Download](#) [Blog](#) [Docs](#) [Workshops](#) [Donate](#)

Search



Examples

[Text Mining](#)

[Bioinformatics](#)

[Fairness](#)

[Survival Analysis](#)

[Classification](#)

[Clustering](#)

[Hierarchical Clustering](#)

[Cox Regression](#)

[Scatter Plot](#)

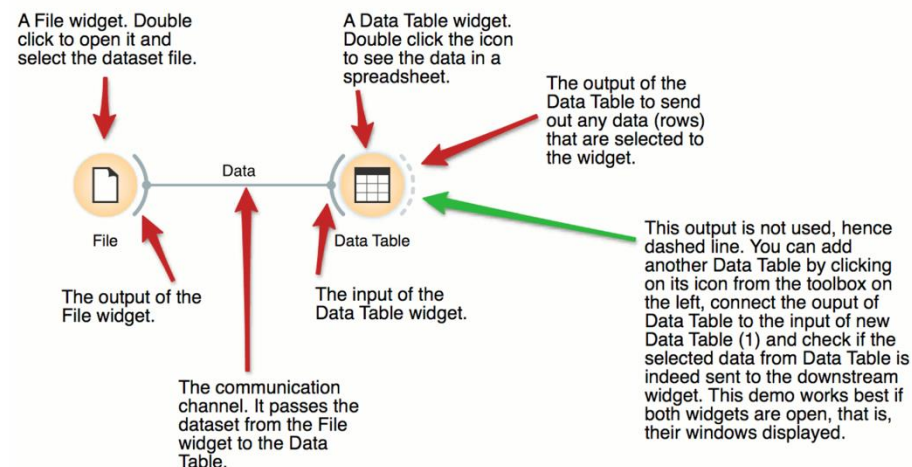
[Visualization](#)

[Data Table, Data Loading](#)

File and Data Table

The basic data mining units in Orange are called widgets. In this workflow, the File widget reads the data. File widget communicates this data to Data Table widget that shows the data in a spreadsheet. The output of File is connected to the input of Data Table.

Download



<https://orangedatamining.com/examples/>



Actividad: Entrenamiento de modelos

Clasificación: evaluación

- Matriz de confusión
- Exactitud
- Sensibilidad
- Especificidad
- Precisión

Matriz de confusión		Estimado por el modelo			
		Negativo (N)	Positivo (P)		
Real	Negativo	a: (TN)	b: (FP)	Precisión ("precision") Porcentaje predicciones positivas correctas:	d/(b+d)
	Positivo	c: (FN)	d: (TP)		
		Sensibilidad, exhaustividad ("Recall") Porcentaje casos positivos detectados	Especificidad ("Specificity") Porcentaje casos negativos detectados	Exactitud ("accuracy") Porcentaje de predicciones correctas (No sirve en datasets poco equilibrados)	
		d/(d+c)	a/(a+b)	(a+d)/(a+b+c+d)	

<https://telefonicatech.com/blog/ml-a-tu-alcance-matriz-confusion>

Actividad: Entrenamiento de modelos

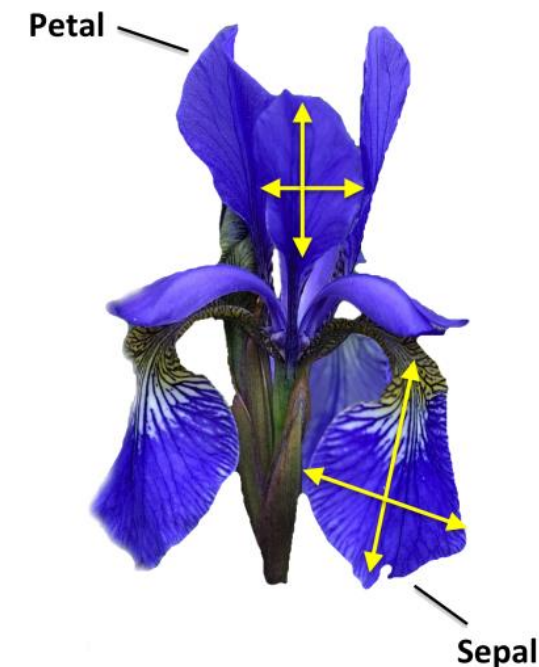
• Actividad 1: Clasificación supervisada

• Conjunto de datos: Iris

- <https://archive.ics.uci.edu/dataset/53/iris> (ya incluido en Orange, no necesaria descarga)

• Información del conjunto de datos:

Número de muestras	Valores perdidos	Atributos de entrada	Clases
150	No	4	3

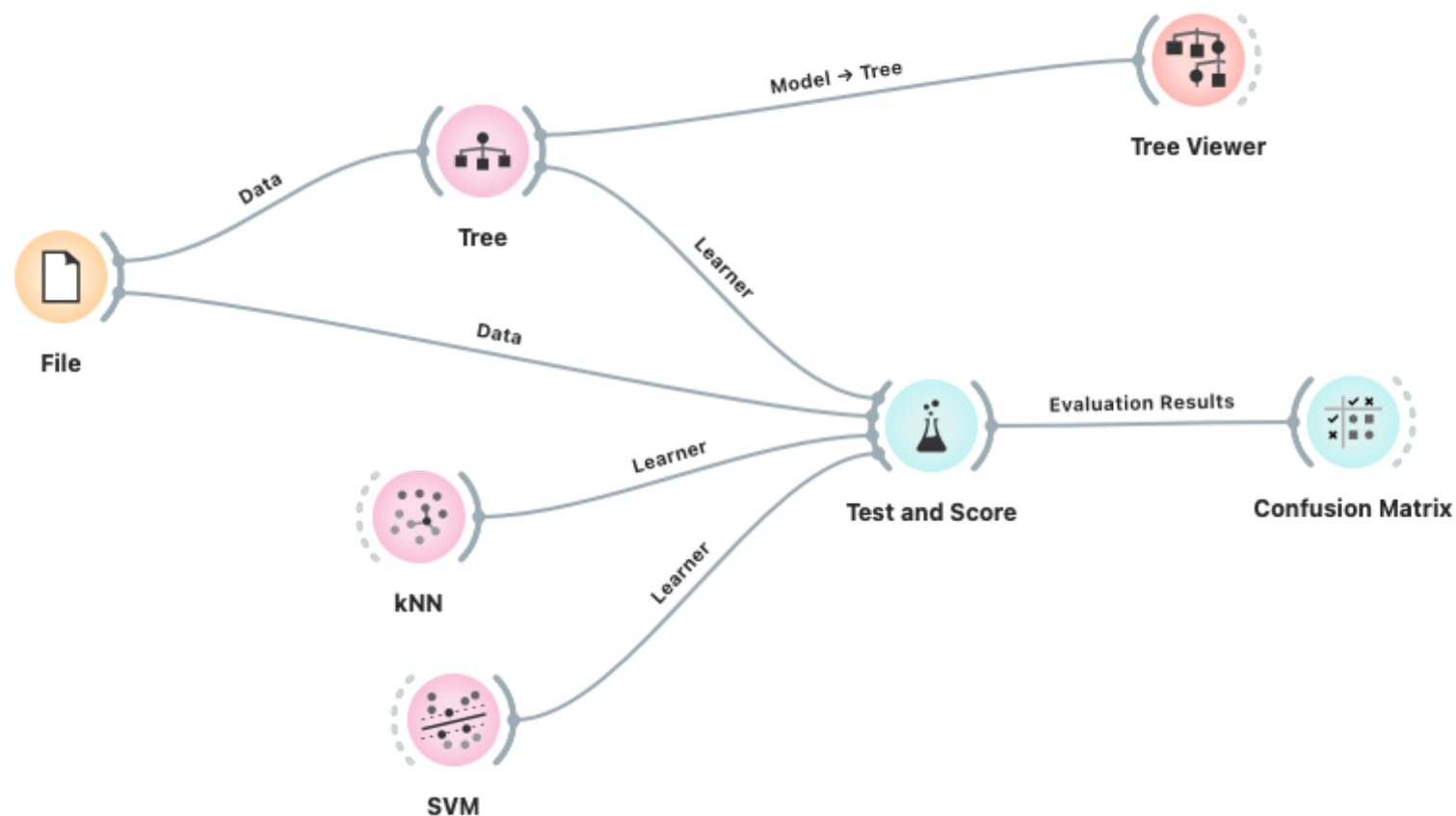




Actividad: Entrenamiento de modelos

- **Actividad 1: Clasificación supervisada**
 - Objetivo de la actividad:
 - Resolver un problema de aprendizaje supervisado.
 - Aprender a cargar un conjunto de datos.
 - Probar diferentes algoritmos de clasificación supervisada:
 - Árboles de decisión.
 - Vecinos más cercanos (kNN).
 - Máquinas de vectores de soporte(SVM).
 - Redes de neuronas.
 - Visualizar resultados.

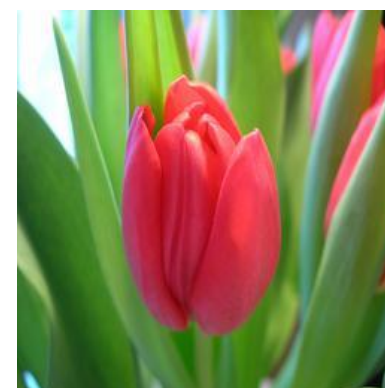
Actividad: Entrenamiento de modelos





Actividad: Entrenamiento de modelos

- **Actividad 2: Clasificación supervisada de imágenes**
 - Utilizaremos un subconjunto de imágenes, extraídas de:
 - <https://www.kaggle.com/datasets/marquis03/flower-classification>
 - Las imágenes están disponibles para descargar en el aula virtual.

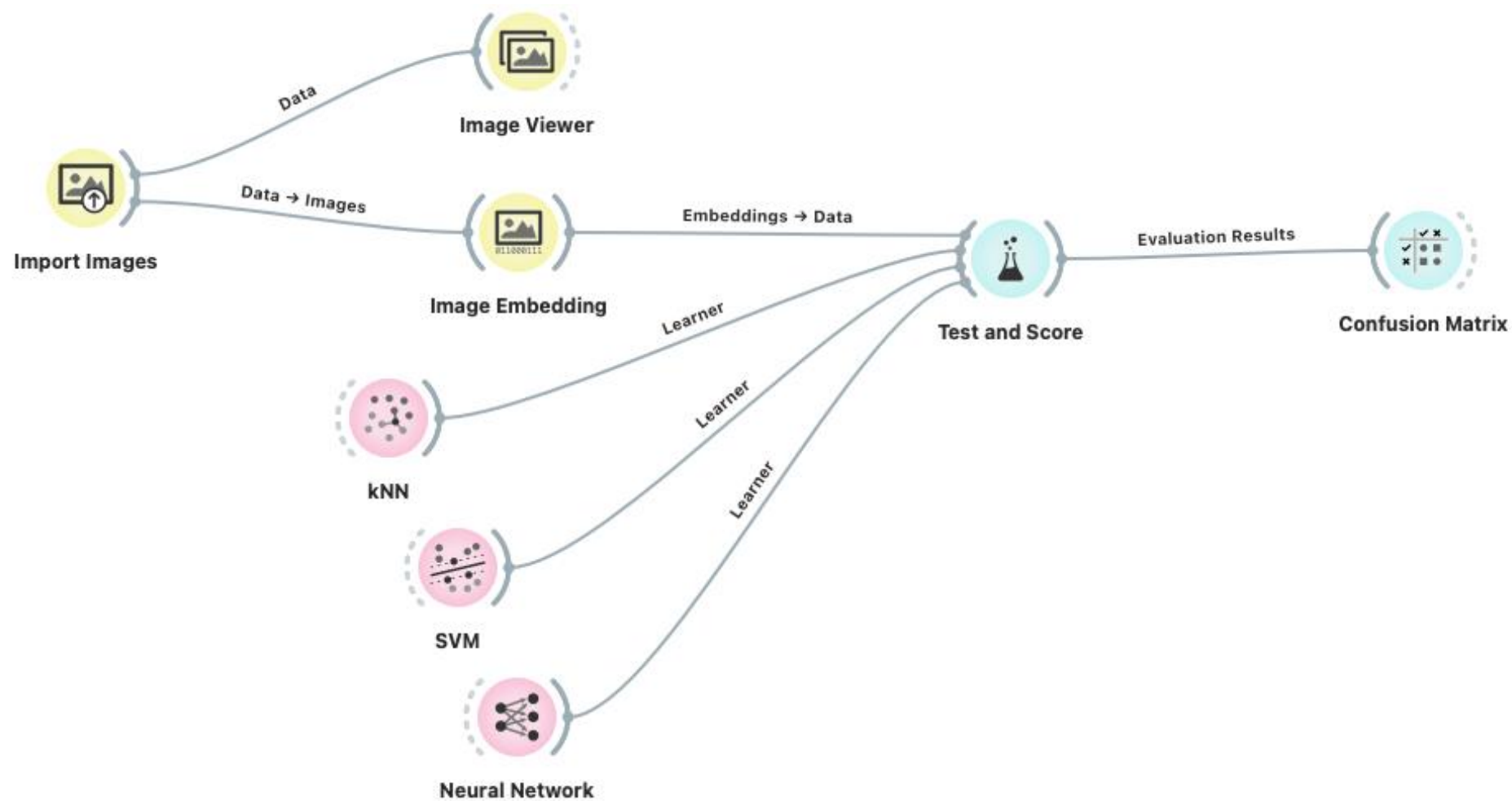




Actividad: Entrenamiento de modelos

- **Actividad 2: Clasificación supervisada de imágenes**
 - **Objetivo de la actividad:**
 - Resolver un problema de aprendizaje supervisado.
 - **Aprender a cargar un conjunto de datos en formato imagen.**
 - **Preparar los datos de entrada.**
 - Probar diferentes algoritmos de clasificación supervisada:
 - Vecinos más cercanos (kNN).
 - Máquinas de vectores de soporte(SVM).
 - Redes de neuronas.
 - Visualizar resultados.

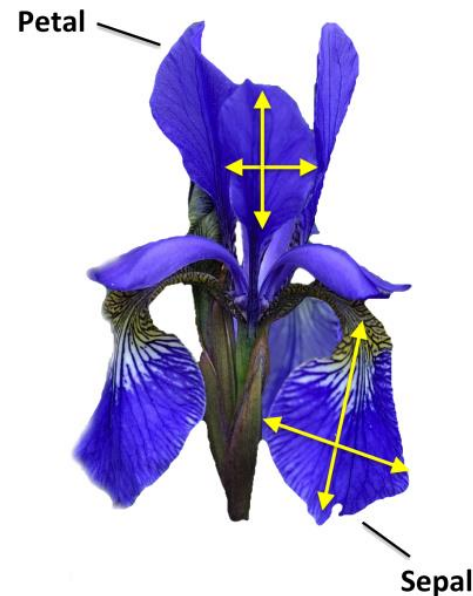
Actividad: Entrenamiento de modelos





Actividad: Entrenamiento de modelos

- **Actividad 3: Clasificación no supervisada**
 - Conjunto de datos: Iris
 - <https://archive.ics.uci.edu/dataset/53/iris> (ya incluido en Orange, no necesaria descarga)
 - Información del conjunto de datos:



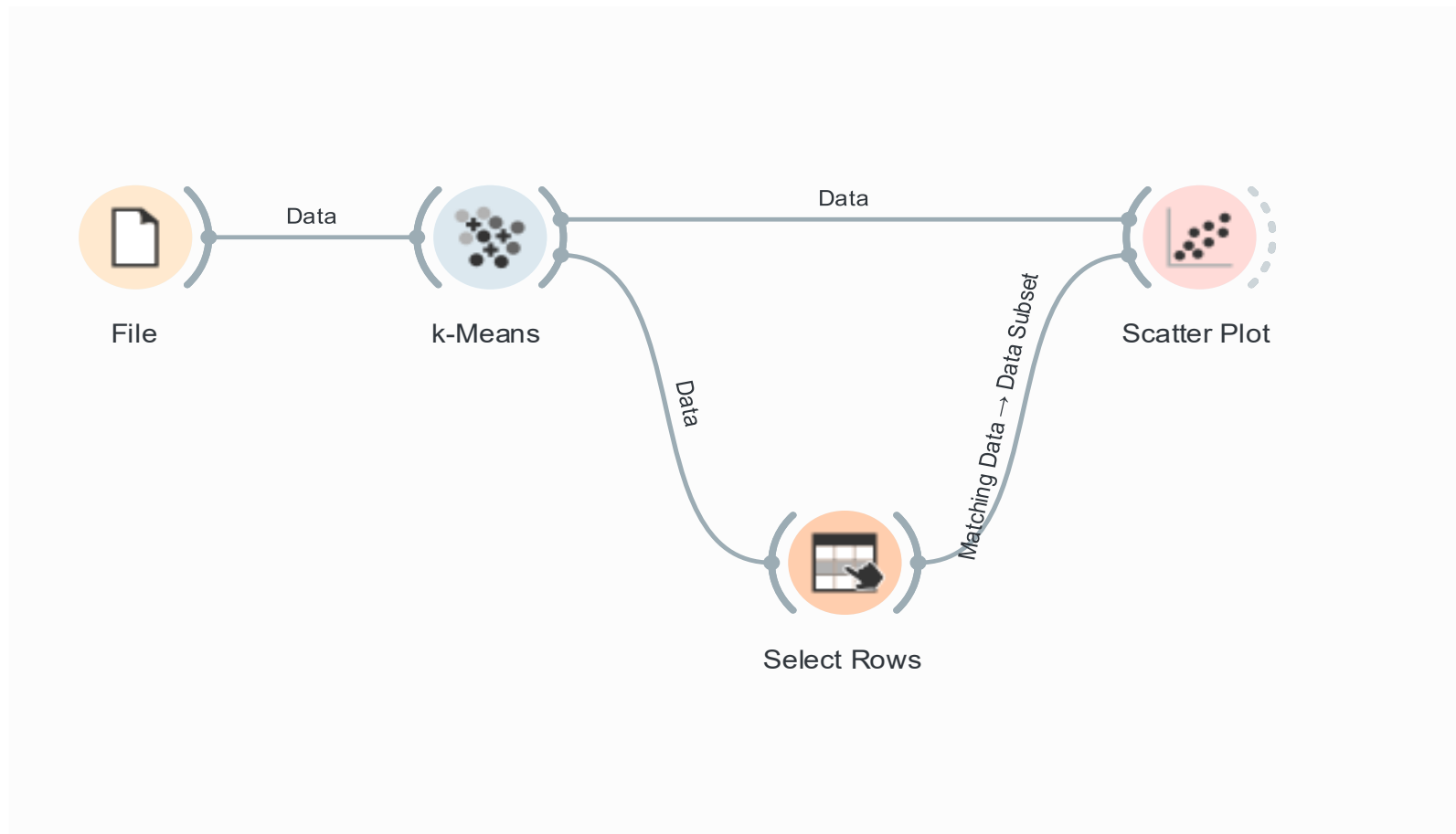


Actividad: Entrenamiento de modelos

- **Actividad 3: Clasificación no supervisada**
 - Objetivo de la actividad
 - Resolver un problema de aprendizaje no supervisado.
 - Aprender a cargar un conjunto de datos.
 - Probar algoritmos de clasificación no supervisada:
 - Algoritmo de clustering K-means.
 - Visualizar resultados.



Actividad: Entrenamiento de modelos





Actividad: Entrenamiento de modelos

- **Actividad 4: Regresión**

- Utilizaremos el siguiente conjunto de datos:
 - <https://www.kaggle.com/datasets/heitornunes/yacht-hydrodynamics-data-set>
- El conjunto de datos está disponible para descargar en el aula virtual.

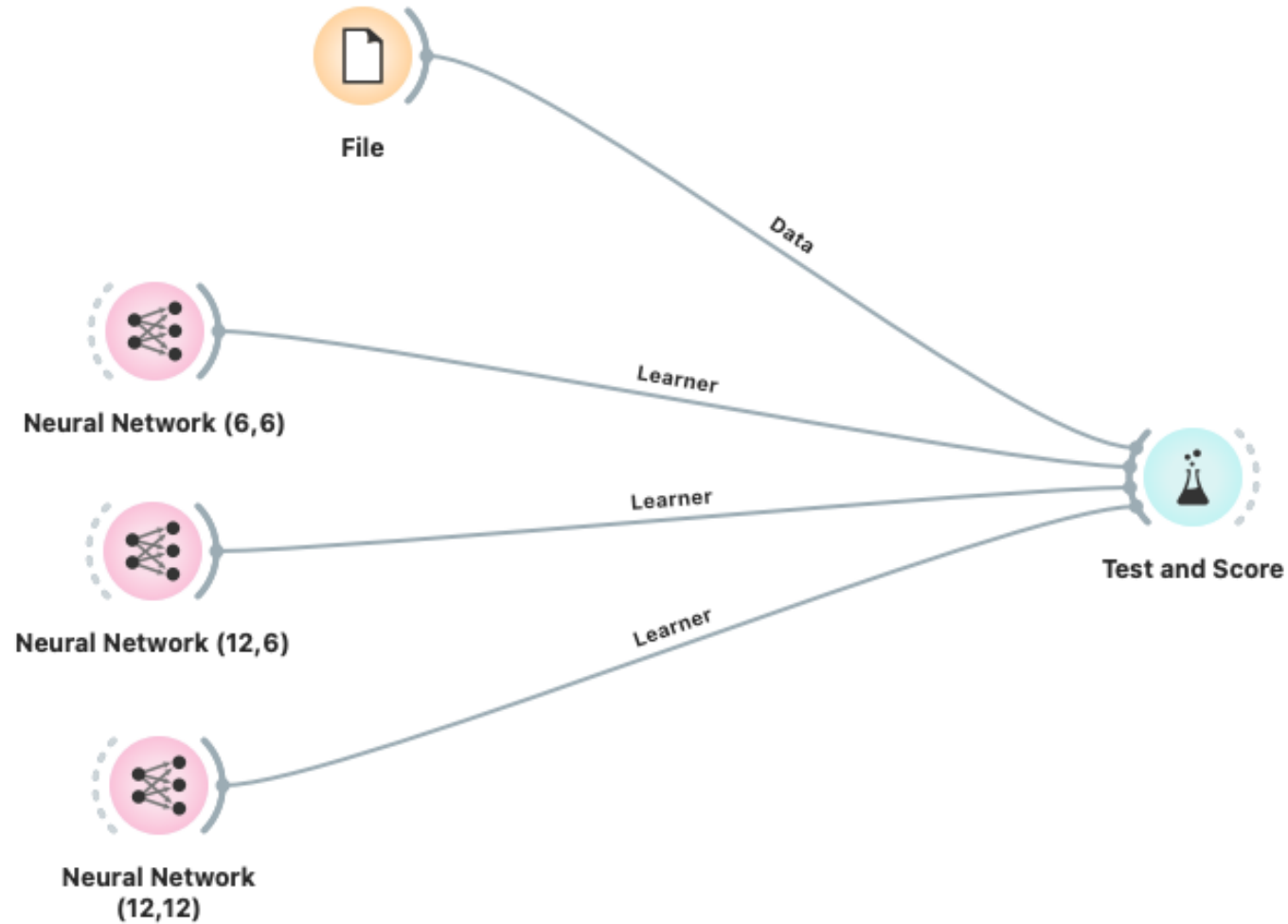


Actividad: Entrenamiento de modelos

- **Actividad 4: Regresión**
 - Objetivo de la actividad:
 - Resolver un problema de aprendizaje supervisado usando técnicas de regresión.
 - Aprender a cargar un conjunto de datos.
 - Probar diferentes algoritmos para crear modelos matemáticos de regresión:
 - Redes de neuronas.
 - Visualizar resultados.



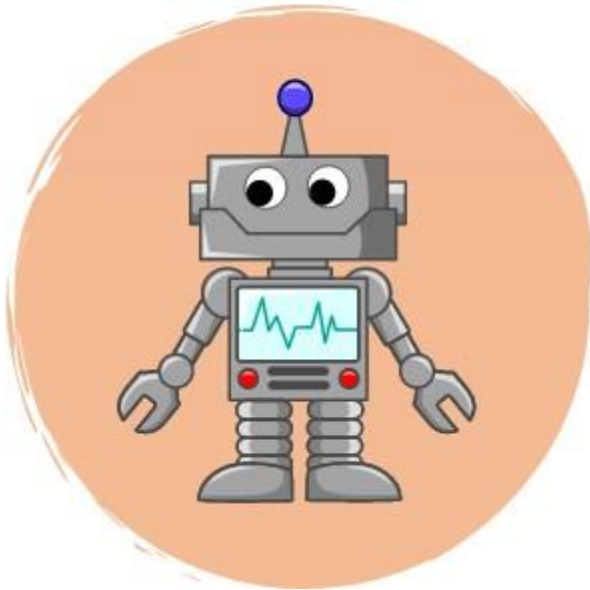
Actividad: Entrenamiento de modelos





Actividad: Aprendizaje por refuerzo

AGENTE



ACCIÓN



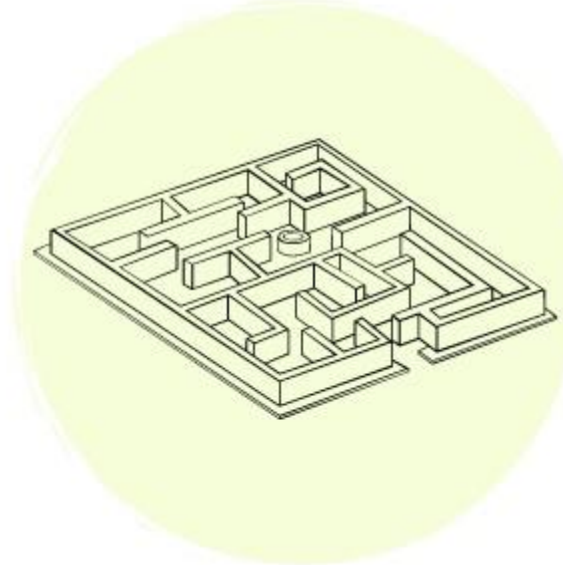
ESTADO



RECOMPENSA



ENTORNO





Aprendizaje por refuerzo: Q-learning

- Se trata de **encontrar la siguiente mejor acción**, dado un estado actual, tratando de maximizar la recompensa.
- El objetivo principal a la hora de entrenar nuestro modelo a través de las simulaciones será **"rellenar" una tabla de acciones para que las decisiones que tome nuestro agente obtengan "la mayor recompensa"** a la vez que avanzamos y no nos quedamos estancados, es decir, poder cumplir el objetivo global (o final) que queremos conseguir.



Aprendizaje por refuerzo: Q-learning

- En este tipo de modelo, es necesario conocer los siguientes conceptos:
 - **Q-table**: la tabla de acciones. Se rellena automáticamente utilizando el algoritmo de aprendizaje durante la fase de aprendizaje. El objetivo es aprender la tabla Q.
 - **Valores Q**: se utilizan para determinar lo buena que es una Acción, A, realizada en un estado concreto, S. $\rightarrow Q(A, S)$.
- Funciona en dos fases:
 - **Aprendizaje**: se actualizarán los valores de la tabla Q.
 - **Ejecución**: se selecciona la acción con mayor valor en la tabla Q en el estado en el que se encuentra.



Aprendizaje por refuerzo: Q-learning

- La tabla Q tiene la forma siguiente:

Q(s,a)	A1	A2	A3	...	An
S0					
S1					
S2					
...					
Sn					



Aprendizaje por refuerzo: Q-learning

- Durante la **fase de aprendizaje**, la selección de la acción a realizar se realiza de dos formas:
 - **Selección aleatoria**: permite al agente conocer el resultado de las acciones, ya que todavía no sabe cual es mejor. Se establece un porcentaje de exploración para esta selección aleatoria.
 - **Selección de la mejor acción**: mayor valor de la tabla Q para el estado del robot.
- Tras realizar la acción seleccionada, se calcula la recompensa y se actualiza la tabla Q.



Aprendizaje por refuerzo: Q-learning

- La actualización de la tabla Q se realiza aplicando la fórmula:

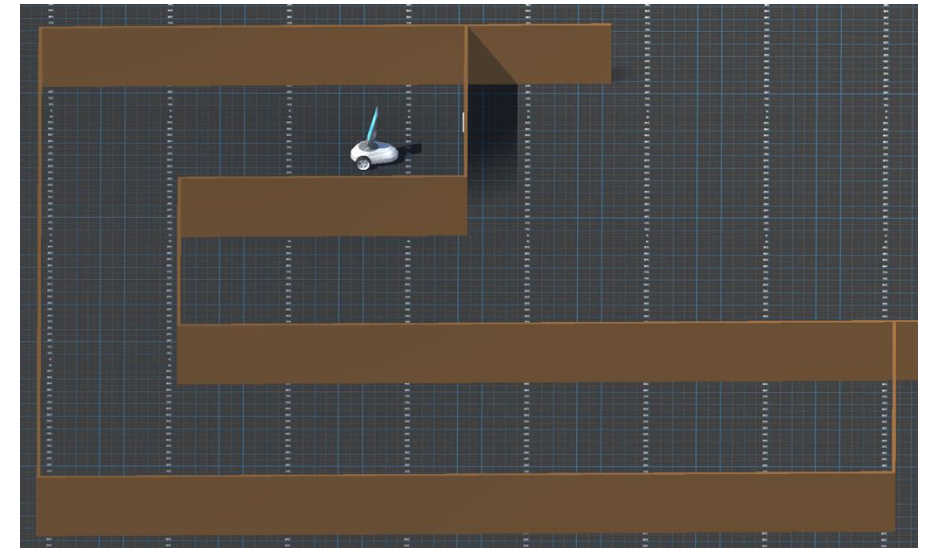
$$Q(s_t, a) = (1 - \alpha) \cdot Q(s_t, a) + \alpha \cdot [R(s_t, a) + \gamma \max Q(s_{t+1}, a)]$$

- $Q(s_t, a)$: Antiguo valor almacenado en la tabla Q para ese estado y acción.
- α : tasa de aprendizaje. Valor entre 0 y 1. Valores cercanos a 1 tiene mucho más en cuenta las nuevas acciones.
- γ : factor de descuento. Valor entre 0 y 1. Valores cercanos a 0, tiene en cuenta los valores inmediatos.
- $R(s_t, a)$: recompensa obtenida por ejecutar la acción a en el estado s .
- $\max Q(s_{t+1}, a)$: máximo valor de recompensa que se obtendrá al transitar al estado s . Es el valor máximo de la tabla para el estado s .



Actividad: Aprendizaje por refuerzo

- Actividad 5: Aprendizaje tabla Q-learning
- Objetivo de la actividad:
 - Crear una tabla Q que permita a un Robobo aprender de forma autónoma a moverse en un laberinto.
 - Aprender a definir los estados en los que puede estar el robot y las acciones que podría realizar en base a ellos.





Actividad: Aprendizaje por refuerzo

- En este ejemplo podemos definir **tres estados**, en función de la mayor distancia detectada por el robot respecto a los obstáculos con sus sensores IR delanteros.
- También definimos **tres acciones** que el robot puede realizar:

Estado	Condición
S1	Mayor distancia de frente
S2	Mayor distancia a la derecha
S3	Mayor distancia a la izquierda

Acción	Comportamiento
A1	Avanzar recto
A2	Girar a la derecha
A3	Girar a la izquierda



Actividad: Aprendizaje por refuerzo

- Una vez determinados los estados y acciones, hay que rellenar la tabla para que el robot sepa en qué estado se encuentra y cómo realizar cada una de las acciones.

Q(s,a)	A1	A2	A3
S0			
S1			
S2			



Actividad: Aprendizaje por refuerzo

- Ejemplo de cómo funciona el algoritmo de Q-learning:

<https://www.mladdict.com/q-learning-simulator>



XUNTA
DE GALICIA

CENTRO DE
FORMACIÓN E
RECURSOS DE FERROL

Inteligencia Artificial para la Sociedad

Alejandro Romero

alejandro.romero.montero@udc.es

Octubre de 2025