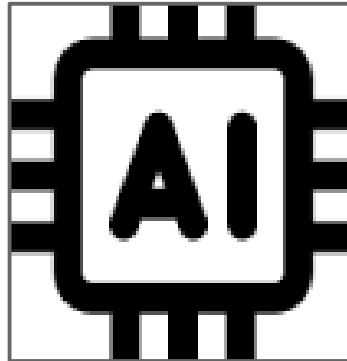


# Introducción a la Inteligencia Artificial



## Clasificadores Binarios

# En esta Presentación

## 1. Introducción a los modelos Clasificadores

- Presentación
- Evaluación

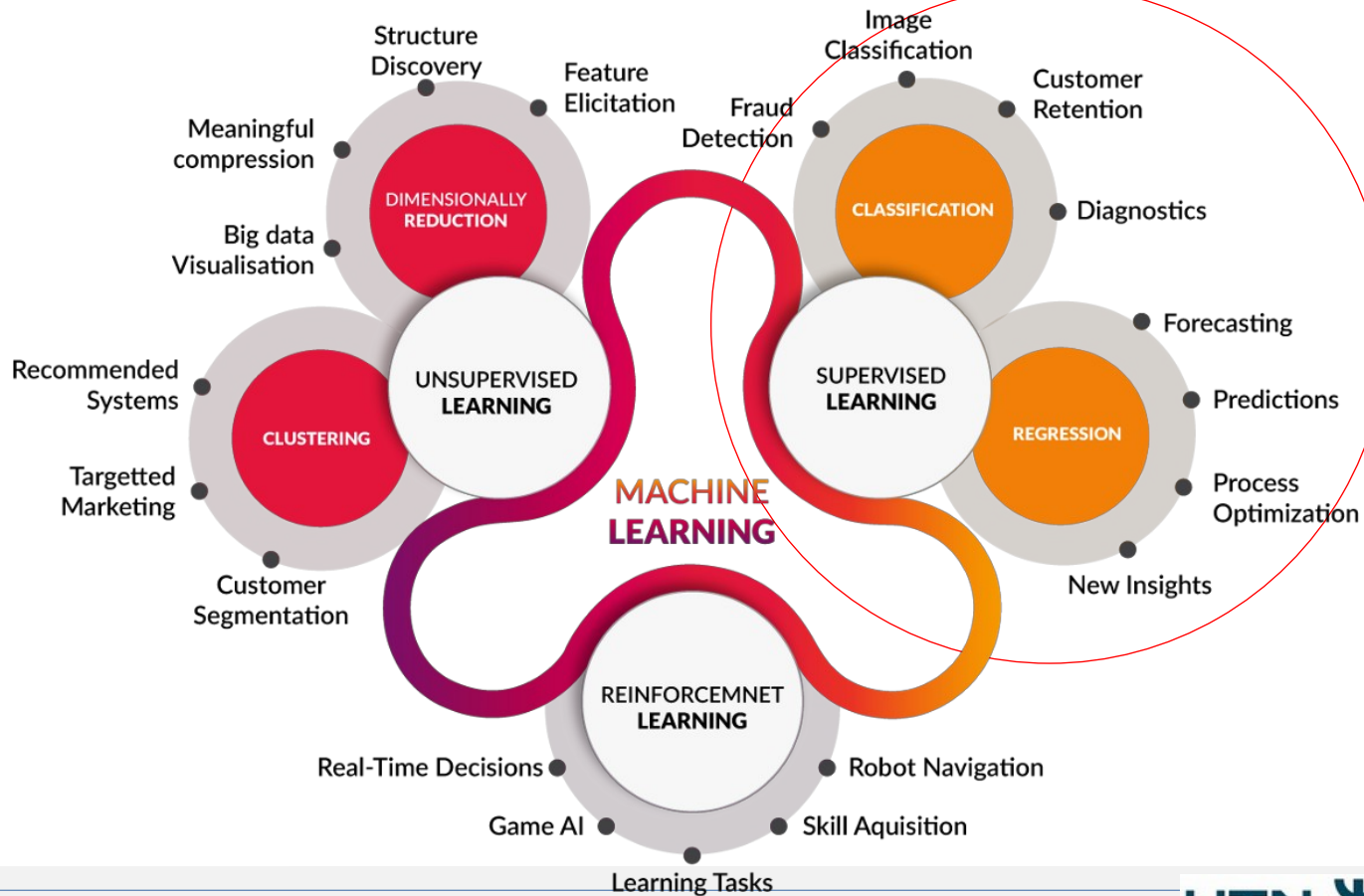
## 2. El aprendizaje supervisado

- Enfoque y abordaje

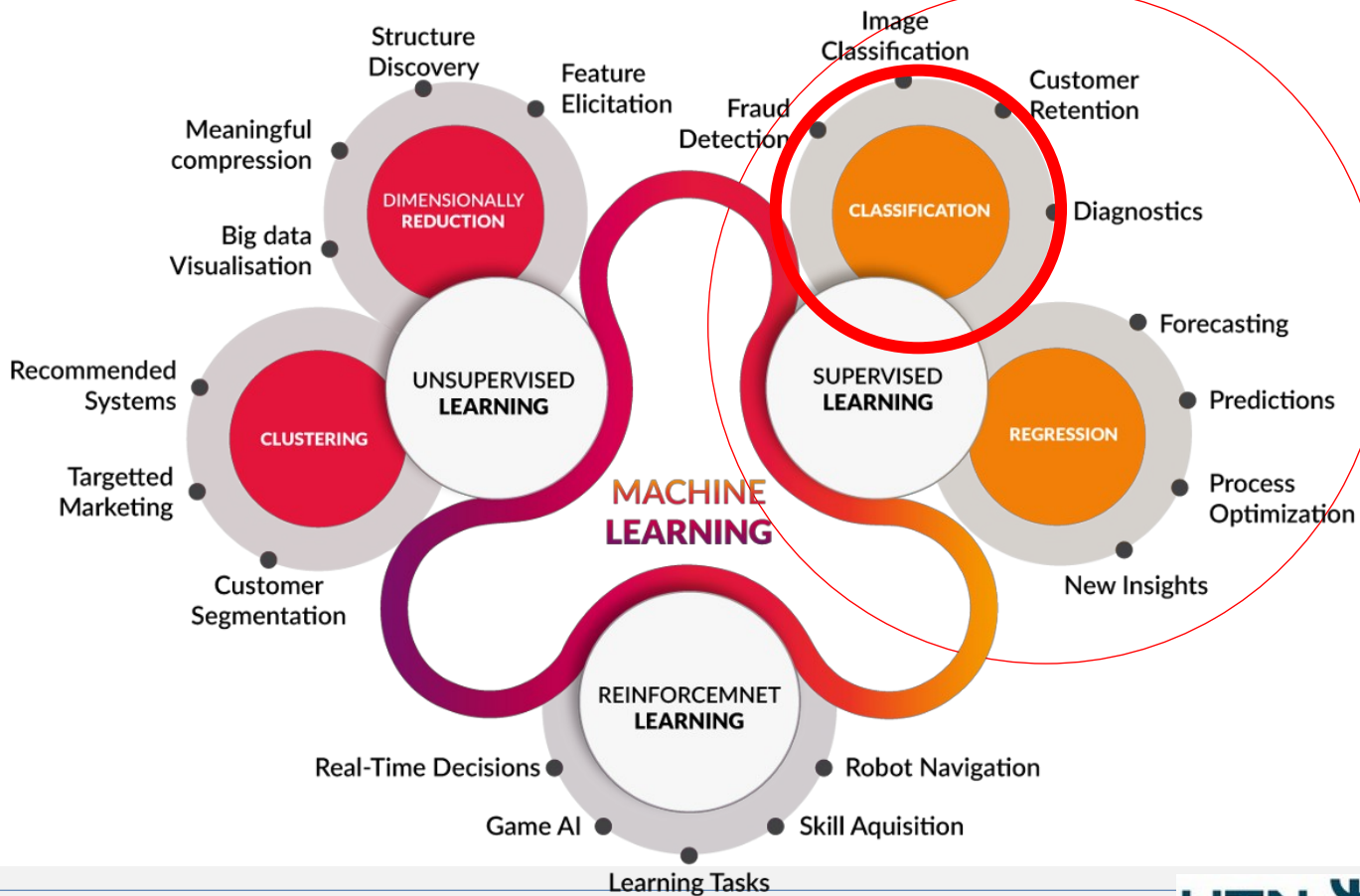
## 3. Algoritmos de Regresión

- Tipos de regresión
- Casos de uso

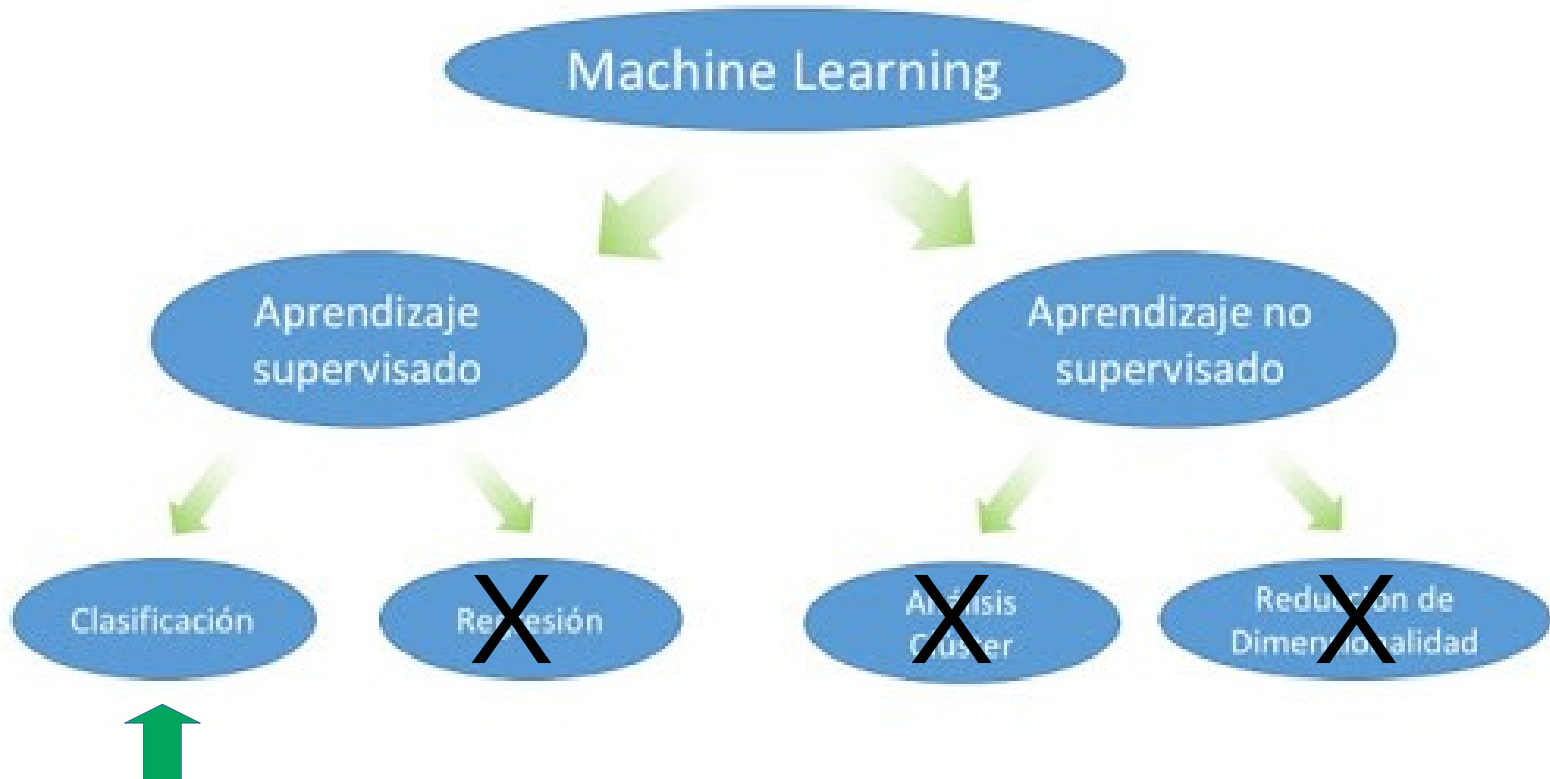
# Introducción a los Modelos Clasificadores



# Introducción a los Modelos Clasificadores



## Introducción a los Modelos Clasificadores



# Presentación

Son modelos de Aprendizaje Supervisado.

Se intenta predecir una clase (ausencia o no de un atributo) en un conjunto de datos.

# Evaluación de los modelos

- Matriz de confusión
- Principales indicadores

# Evaluación de los modelos

## Matriz de confusión

Clase real	Clasif. como	
	0	1
0	a	b
1	c	d



# Evaluación de los modelos

## Matriz de confusión

Clase real	Clasif. como	
	0	1
0	a	b
1	c	d

a: Verdadero Negativo

b: Falso Positivo

c: Falso Negativo

d: Verdadero Positivo

# Evaluación de los modelos

## Matriz de confusión

	Clasif. como	
Clase real	0	1
0	max_0	0
1	0	max_1

Un buen clasificador tiene:

Los elementos de la diagonal tienden a los máximos.

Los elementos de la contra diagonal tienden a cero.

# Evaluación de los modelos

## Matriz de confusión

### Ejemplo: Matriz de Confusión

$\begin{bmatrix} 9 & 18 \\ 3 & 42 \end{bmatrix}$       9 Verdaderos Negativos, 18 Falsos Positivos  
3 Falsos Negativos, 42 Verdaderos Positivos

*Este clasificador identifica muy bien a los positivos (que son mayoría), pero es deficiente en detectar Negativos.*

# Evaluación de los modelos

## Principales indicadores

- Precision: Cantidad de aciertos en la clasificación.
- Recall: Tasa de verdaderos positivos o negativos.
- F1-Score: Combinación de Precision y recall.
- Accuracy (confianza): Tasa total de aciertos

# Evaluación de los modelos

## Principales indicadores

$$Presicion_0 = \frac{a}{a+c}$$

<b>a</b>		b
c		d

$$Presicion_1 = \frac{d}{b+d}$$

a		<b>b</b>
c		<b>d</b>

# Evaluación de los modelos

## Principales indicadores

$$Presicion_0 = \frac{a}{a+c}$$

a		b
+-----+		
c		d

$$Presicion_1 = \frac{d}{b+d}$$

a		b
+-----+		
c		d

$$Recall_0 = \frac{a}{a+b}$$

a		b
+-----+		
c		d

$$Recall_1 = \frac{d}{c+d}$$

a		b
+-----+		
c		d

# Evaluación de los modelos

## Principales indicadores

$$F1 - Score_0 = 2 \cdot \frac{precision_0 \cdot recall_0}{precision_0 + recall_0}$$

$$F1 - Score_1 = 2 \cdot \frac{precision_1 \cdot recall_1}{precision_1 + recall_1}$$

# Evaluación de los modelos

## Principales indicadores

$$F1 - Score_0 = 2 \cdot \frac{\text{precisión}_0 \cdot \text{recall}_0}{\text{precisión}_0 + \text{recall}_0}$$

$$F1 - Score_1 = 2 \cdot \frac{\text{precisión}_1 \cdot \text{recall}_1}{\text{precisión}_1 + \text{recall}_1}$$

$$\text{Accuracy} = \frac{a+d}{a+b+c+d}$$

a		b
=====+		
c		d



# Evaluación de los modelos

## Principales indicadores

### Casos de interés:

- Alta precisión y alto recall: el modelo maneja perfectamente esa clase
- Alta precisión y bajo recall: el modelo no detecta la clase muy bien, pero cuando lo hace es altamente confiable.
- Baja precisión y alto recall: El modelo detecta bien la clase pero también incluye muestras de otras clases.
- Baja precisión y bajo recall: El modelo no logra clasificar la clase correctamente.

# Evaluación de los modelos

## Principales indicadores

Ejemplo de Reporte:

### Reporte de las Clasificaciones

	precision	recall	f1-score	support
0	0.75	0.33	0.46	27
1	0.70	0.93	0.80	45
accuracy			0.71	72
macro avg	0.72	0.63	0.63	72
weighted avg	0.72	0.71	0.67	72

# Regresión Logística

La Regresión Logística es un algoritmo de Machine Learning que permite tomar una serie de datos y clasificarlos en una de dos posibles categorías o clases.

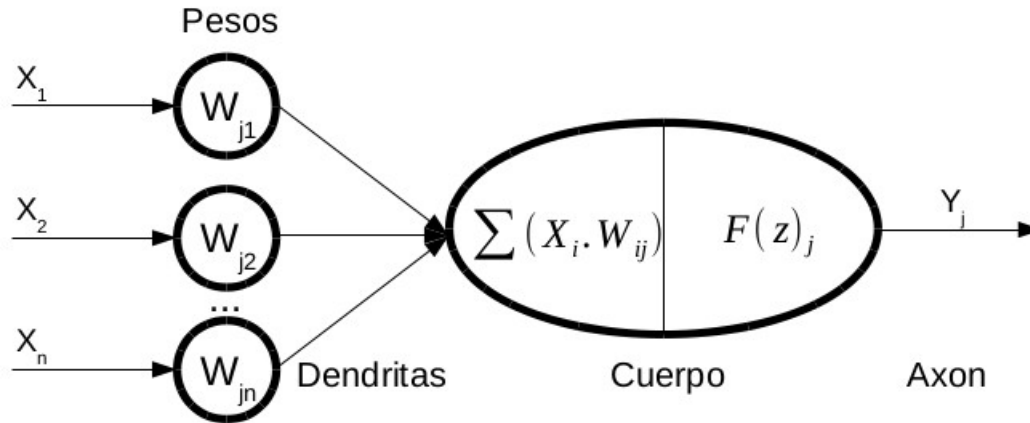
A diferencia de otras regresiones, los resultados de la Regresión logística son sólo binarios.

Su funcionamiento puede modelarse con un perceptrón: el modelo artificial de una neurona.

Un regresor logístico es una red neuronal de una neurona.

# Regresión Logística

## Perceptron<sup>1</sup>: Modelo de Neurona artificial



1. Modelo presentado en: F Rosenblatt. Perceptron simulation experiments, Proceedings of the IRE, 1960 - [ieeexplore.ieee.org](http://ieeexplore.ieee.org)

# Regresión Logística

## Función de activación: Sigmoide

$$\sigma(z) = \frac{1}{1 + e^{-z}} \quad \sigma(z) \in [0, 1]$$

donde:

$e$ : Número de Euler (aproximadamente 2,71)

$$z = \beta_0 + \beta_1 \cdot x_1 + \beta_2 \cdot x_2 + \dots + \beta_n \cdot x_n$$

con:

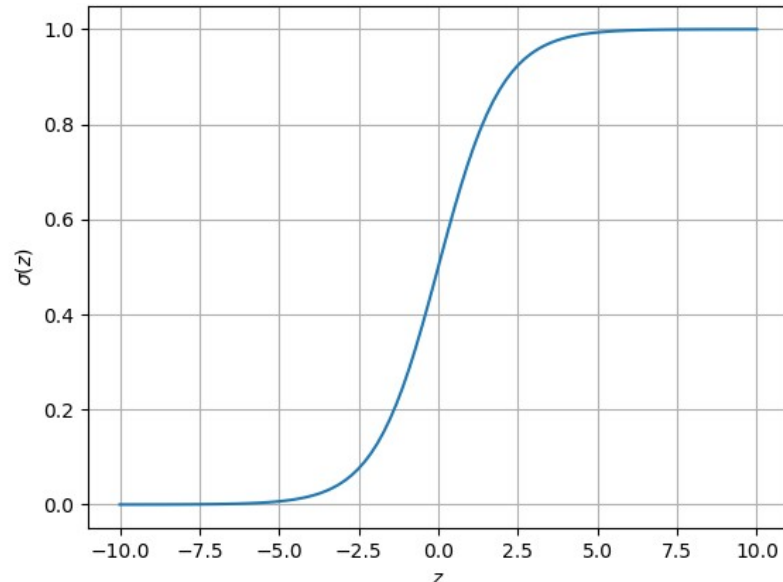
$\beta_0$ : término independiente (Bias o intercept)

$\beta_i$ : coeficientes de la combinación lineal (o pesos)

$x_i$ : valores (normalizados) que toman las variables

# Regresión Logística

## Función de activación: Sigmoid



Fuente: <https://www.codificandobits.com/blog/regresion-logistica-y-neurona-artificial/>

# Regresión Logística

## Función de activación: Sigmoid

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

Si  $\sigma(z) \geq 0,5 \Rightarrow y(\sigma(z)) = 1$

Si  $\sigma(z) < 0,5 \Rightarrow y(\sigma(z)) = 0$

# Regresión Logística

## Ejemplo: dataset Diabetes.

Tras aplicar Regresión Logística, el algoritmo arroja:

Intercept= [-0.22678745]

Coefs= [[ 0.40580204 1.10347521 -0.16017721 -0.01966148 -0.08252006  
0.62776504 0.32711709 0.32981005]]

$\beta_i$



# Regresión Logística

## Ejemplo: dataset Diabetes.

Aplicamos la Sigmoide sobre tres líneas del dataset:

linea	emb	glu	pre	g_piel	insu	imc	pedi	edad	Label	Pred
64	4	97	60	23	0	28.2	443.0	22	0	0
65	4	142	86	0	0	44.0	645.0	22	1	1
66	4	136	70	0	0	31.2	1182.0	22	1	1

Con los valores Normalizados:

64	0,041	-0,811	-1,005	0,1025	-0,72	-0,56	0,028	-0,96	0
65	0,041	0,656	1,096	-1,361	-0,72	1,61	0,617	-0,96	1
66	0,041	0,461	-0,197	-1,361	-0,72	-0,15	2,183	-0,96	1

$X_i$

$\sigma(z)$

0	0,1754357
1	0,7887775
1	0,6721742

Predicción

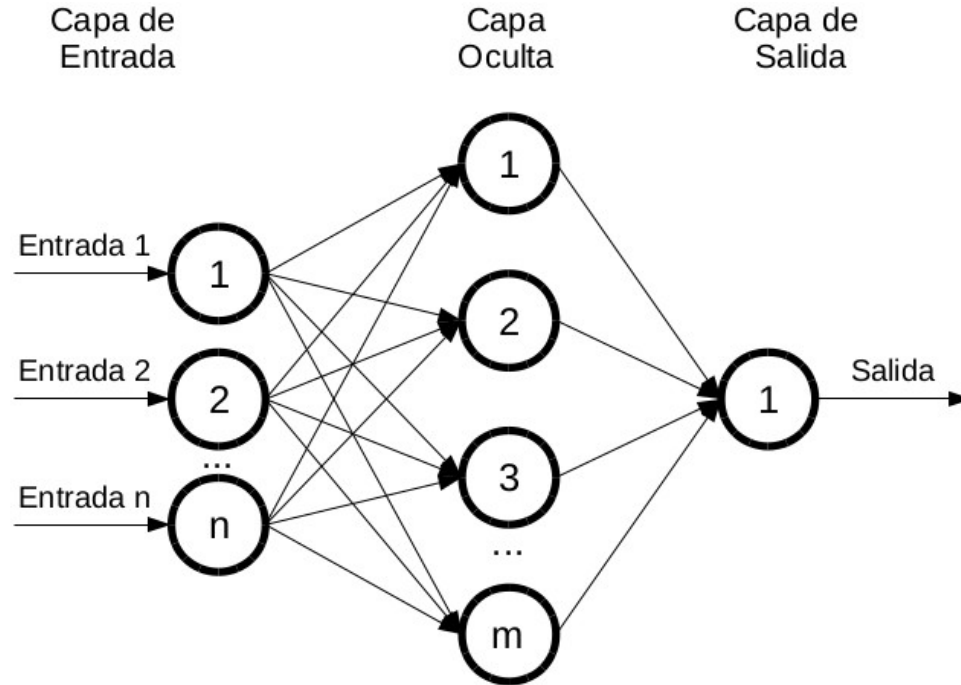
# Redes Neuronales

## Presentación

Una red neuronal no es más que muchos regresores logísticos conectados de diversas maneras. Se le conoce como “Perceptrón Multicapa”, o “Feedforward Neural Network”.

Fuente: Ronchetti, Franco. Curso de Aprendizaje Automático, UNLP, 2018

# Redes Neuronales



Modelo de red neuronal artificial

# Redes Neuronales

## Redes Neuronales Multicapa

Las unidades de procesamiento se organizan en capas. Normalmente, una red neuronal presenta tres partes:

- Una capa de entrada, donde se presentan los campos de entrada.
- Una o varias capas ocultas.
- Una capa de salida que presenta el campo de destino.

Fuente: <https://www.ibm.com/docs/es/spss-modeler/SaaS?topic=networks-neural-model>

# Redes Neuronales

## Redes Neuronales Multicapa

La red aprende examinando los registros individuales, generando una predicción para cada registro y realizando ajustes a las ponderaciones cuando realiza una predicción incorrecta.

Este proceso se repite muchas veces y la red sigue mejorando sus predicciones hasta haber alcanzado uno o varios criterios de parada.

Fuente: <https://www.ibm.com/docs/es/spss-modeler/SaaS?topic=networks-neural-model>

# Redes Neuronales

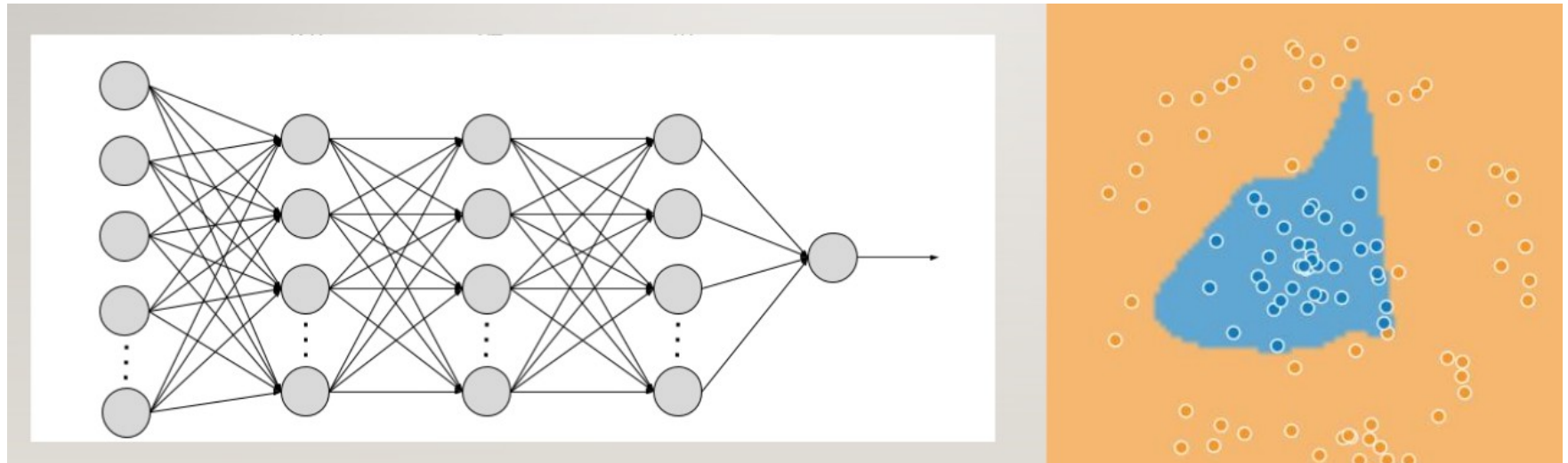
## Redes Neuronales Multicapa

- Cuanto más grande es la red (más capas y más neuronas por capa) permite modelar problemas más complejos.
- A veces ayuda, pero también podemos caer en overfitting.
- *Requiere que el dataset esté normalizado, y es sensible al desbalanceo.*

Fuente: Ronchetti, Franco. Curso de Aprendizaje Automático, UNLP, 2018

# Redes Neuronales

## Redes Neuronales Multicapa:



Fuente: Ronchetti, Franco. Curso de Aprendizaje Automático, UNLP, 2018

# Random Forest

## Presentación

- Es un conjunto de árboles de decisión combinados entre sí mediante una técnica denominada Bagging.
- La primera propuesta de Random decision forests, fue hecha por Tin Kam Ho de Bell Labs en 1995.

Fuentes: [https://es.wikipedia.org/wiki/Random\\_forest](https://es.wikipedia.org/wiki/Random_forest)



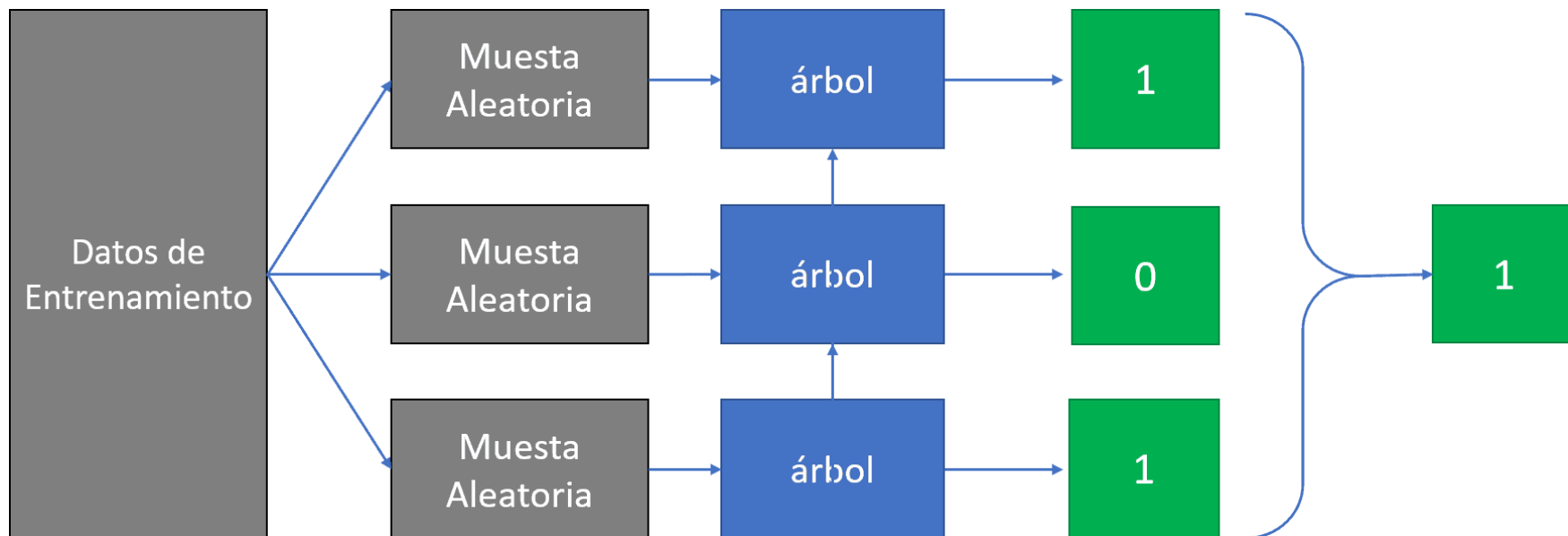
# Random Forest

## Fundamentos

- Un bosque está compuesto por árboles. Se dice que cuantos más árboles tiene, más robusto es un bosque.
- Cada uno de los bosques aleatorios crean árboles de decisión sobre muestras de datos seleccionadas al azar, obtienen predicciones de cada árbol y seleccionan la mejor solución mediante votación.

Fuentes: [https://es.wikipedia.org/wiki/Random\\_forest](https://es.wikipedia.org/wiki/Random_forest)

# Random Forest



Fuentes: <https://www.iartificial.net/random-forest-bosque-aleatorio/>

# Random Forest

## Ventajas

- Es uno de los algoritmos clasificadores más certero, más aún en grandes datasets
- Puede manejar cientos de variables de entrada sin excluir ninguna.
- Da estimaciones de qué variables son importantes en la clasificación.
- No requiere normalización ni balanceo.

Fuente: [https://es.wikipedia.org/wiki/Random\\_forest](https://es.wikipedia.org/wiki/Random_forest)

Contacto: [ia@frh.utn.edu.ar](mailto:ia@frh.utn.edu.ar)

# Algunos casos de Uso y ejemplos

## Regresión Logística: Diagnósticos médicos

Diagnósticos de enfermedades a partir de datos relacionados con historia clínica y hábitos del paciente.

### Ventajas:

- Expresión que explique el fenómeno.

### Inconvenientes:

- Sensible al desbalanceo de datos.
- Requiere normalización de datos.

# Algunos casos de Uso y ejemplos

## **Regresión Logística: Estudios socio-económicos**

Estudios de factores de crecimiento, educación y estudios universitarios, calidad de producción, siniestralidad y seguros, elección o no de una marca (marketing), entre otros.

# Algunos casos de Uso y ejemplos

## Redes Neuronales: OCR



Fuente: Ronchetti, Franco. Curso de Aprendizaje Automático, UNLP, 2018

Contacto: [ia@frh.utn.edu.ar](mailto:ia@frh.utn.edu.ar)

# Algunos casos de Uso y ejemplos

## Redes Neuronales: OCR



Fuente: Ronchetti, Franco. Curso de Aprendizaje Automático, UNLP, 2018

Contacto: [ia@frh.utn.edu.ar](mailto:ia@frh.utn.edu.ar)

# Algunos casos de Uso y ejemplos

## Redes Neuronales: Detección de rostros



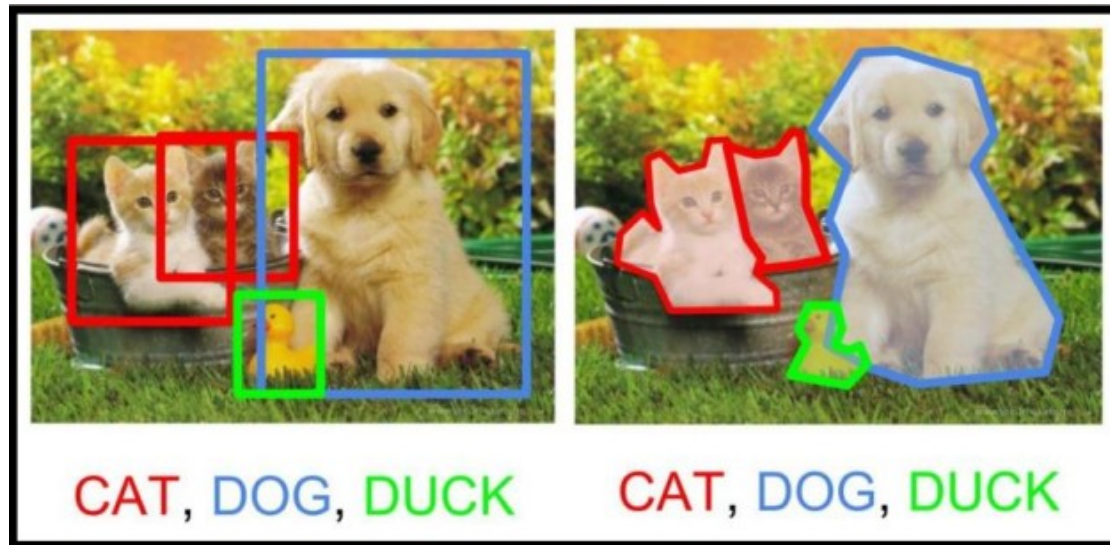
Fuente: Ronchetti, Franco. Curso de Aprendizaje Automático, UNLP, 2018

Contacto: [ia@frh.utn.edu.ar](mailto:ia@frh.utn.edu.ar)



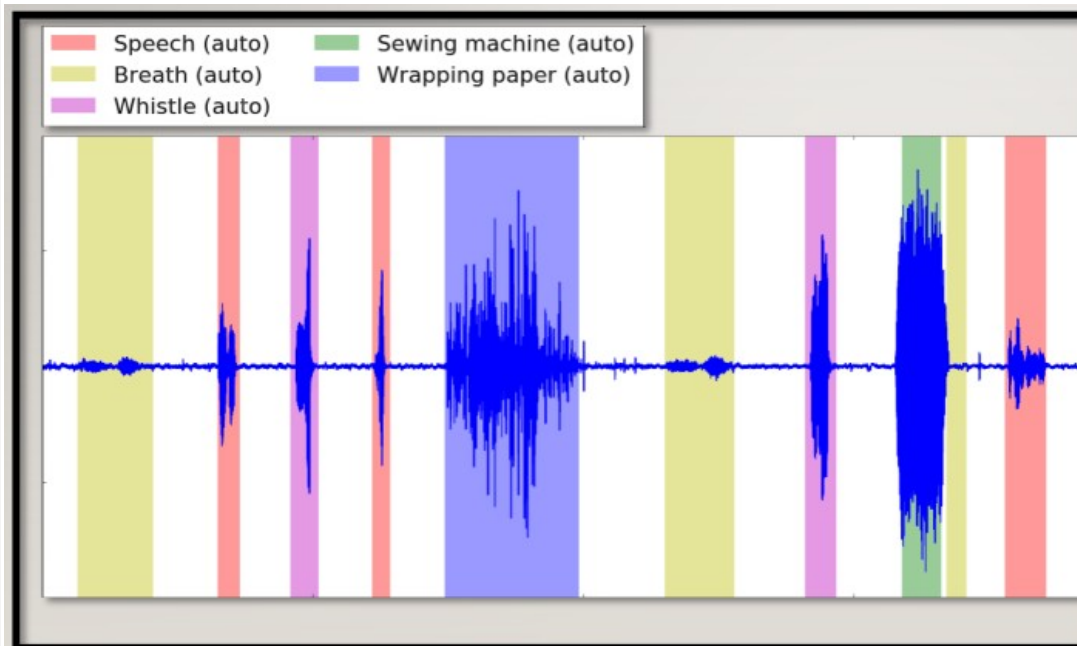
# Algunos casos de Uso y ejemplos

## Redes Neuronales: Detección Objetos / imágenes



# Algunos casos de Uso y ejemplos

## Redes Neuronales: Reconocimiento de voz



Fuente: Ronchetti, Franco. Curso de Aprendizaje Automático, UNLP, 2018

Contacto: [ia@frh.utn.edu.ar](mailto:ia@frh.utn.edu.ar)

# Algunos casos de Uso y ejemplos

## **Random Forest: Clasificaciones en general**

Al ser un algoritmo tan efectivo y versatil, Random Forest puede usarse para generar modelos clasificadores en general.

# Algunos casos de Uso y ejemplos

## **Random Forest: Detección de Fraudes y anomalías**

Tanto fraudes como anomalías se presentan en proporciones muy pequeñas en comparación con el resto de los registros.

Al manejar exitosamente conjuntos de datos desbalanceados, Random Forest se presenta como la opción ideal para este tipo de casos.

# ¿Preguntas?