



# Tecnologías Aplicadas a la Mecatrónica 4.0

Introducción a Big Data y Machine Learning

## Sesión 4 – Modelos de Aprendizaje Supervisado (I): Clasificación

Antonio Saldaña: [antonio.emmanuel.saldana@upc.edu](mailto:antonio.emmanuel.saldana@upc.edu)



## Calendario

	Lunes	Martes	Miércoles	Jueves
<b>NOV</b>	28	29 S1 – Introducción a Big Data y Machine Learning	30	1 S2 – Introducción a Python
<b>DICIEMBRE</b>	5	6	7	8
	12	13 S3 – Estadística descriptiva	14	15 S4 – Modelos de aprendizaje supervisado (I): Clasificación
	19	20 S5 – Modelos de aprendizaje supervisado (II): Regresión	21	22 S6 – Introducción a Image Recognition
<b>VACACIONES</b>				
<b>ENERO</b>	9	10 S7 – Modelos de aprendizaje no supervisado y repaso	11	12 S8 – Exámen



## Objetivos de la sesión

- ¿Qué es clasificación en Machine Learning?
- Tipos de clasificación.
- Aprendizaje supervisado clasificación: aplicaciones.
- Detectar y resolver el desequilibrio de clases.
- Principales modelos de clasificación.
- Métricas de evaluación de la clasificación.
- Presentación de un ejemplo práctico de un modelo de clasificación.



### Recapitulemos...

## MACHINE LEARNING

Aprendizaje  
Supervisado

Aprendizaje **NO**  
Supervisado

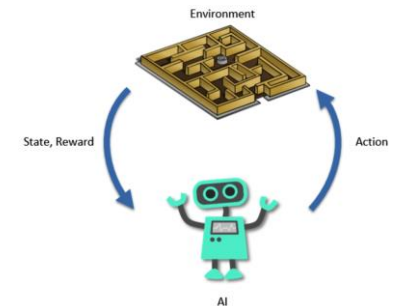
Reinforcement  
Learning

**Clasificación**

Clustering

Regresión

Reducir dimensiones



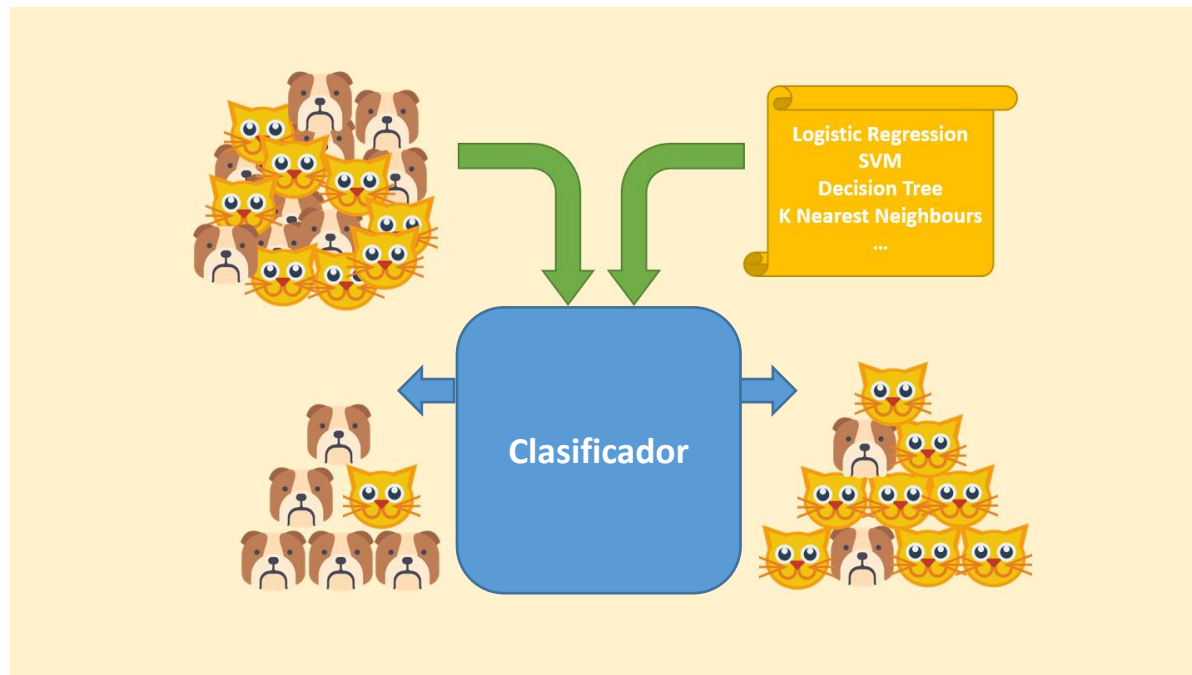
- *Aprenden iterativamente de los datos para encontrar información.*
- *Se debe dar las variables de entrada y salida para entrenar*

- *Se utiliza para explorar grandes conjuntos de datos.*
- *Solo se necesita datos de entrada*



## ¿Qué es clasificación en Machine Learning?

La **clasificación** es una subcategoría del **aprendizaje supervisado** donde el objetivo es predecir una **clase categórica** basada en un conjunto de características/atributos y observaciones pasadas.

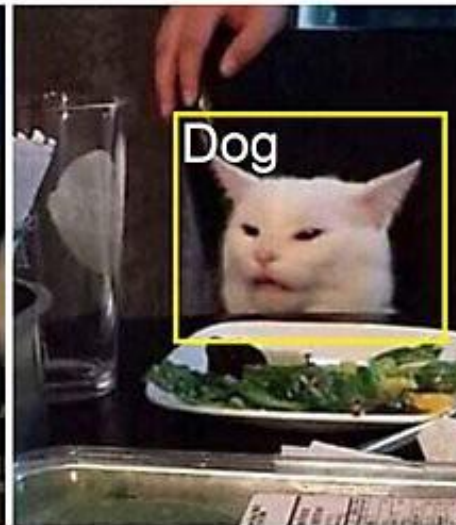




Media saying AI will  
take over the world



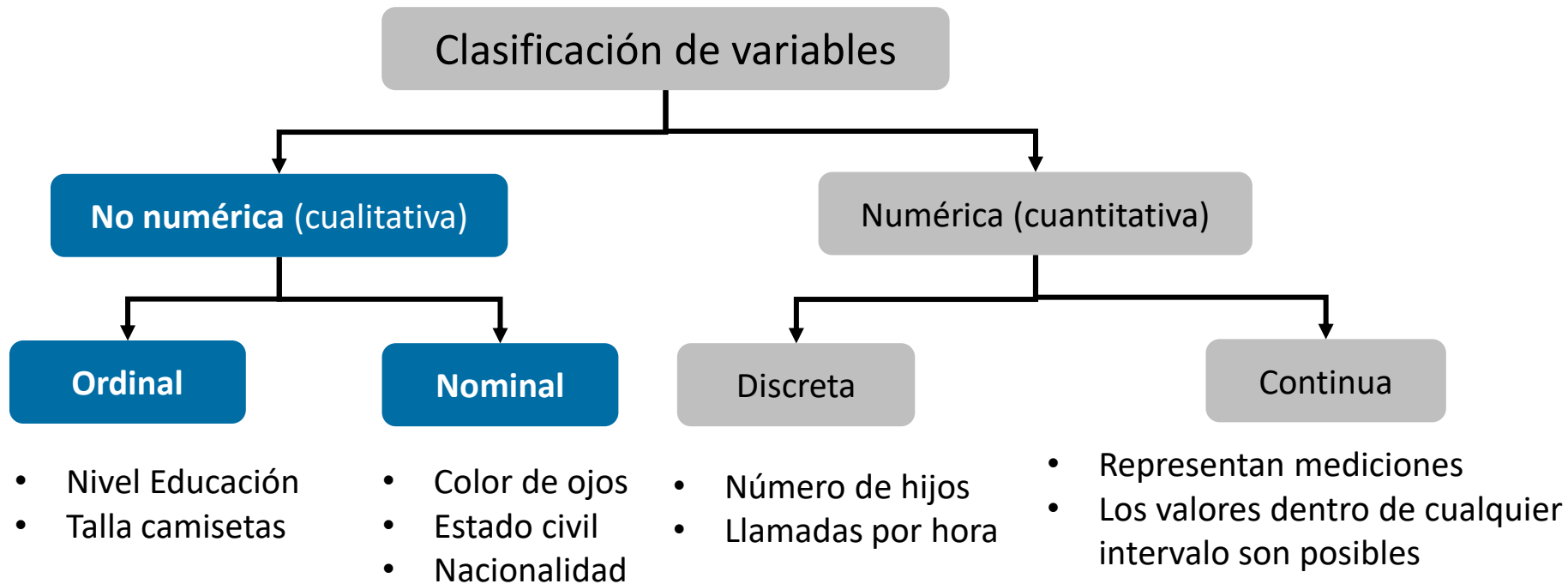
My Neural Network



AI will take over soon



## Clasificación de variables



Es necesario codificar las variables no numéricas para presentarlas como números.



## ¿Cómo pasar de variables cualitativas a numéricas?

Los modelos de machine learning y deep learning requieren que todas las variables de entrada y salida sean **numéricas**.

### Datos de entrada

ID	Country	Population
1	Japan	127185332
2	U.S	326766748
3	India	1354051854
4	China	1415045928
5	U.S	326766748
6	India	1354051854

Datos no numéricos

[Más información](#)

### LABEL ENCODER

ID	Country	Population
1	0	127185332
2	1	326766748
3	2	1354051854
4	3	1415045928
5	1	326766748
6	2	1354051854

El problema aquí es que como hay diferentes números en la misma columna, el modelo malinterpretará los datos y supondrá que existe algún tipo de orden,  $0 < 1 < 2$ .

### ONE HOT ENCODING

ID	Country_Japan	Country_U.S	Country_India	Country_China	Population
1	1	0	0	0	127185332
2	0	1	0	0	326766748
3	0	0	1	0	1354051854
4	0	0	0	1	1415045928
5	0	1	0	0	326766748
6	0	0	1	0	1354051854





Conceptos generales.

## **Tipos de Clasificación.**

Aplicaciones de Clasificación.

Detectar y resolver el desequilibrio de clases.

Principales modelos de Clasificación.

Métricas de evaluación: Clasificación.

Ejemplo práctico de un modelo de Clasificación.



## Tipos de Clasificación

### CLASIFICACIÓN BINARIA

- Solo se pueden asignar dos clases diferentes (**0 o 1**).
- Cada observación sólo puede ser etiquetada como una clase.

### CLASIFICACIÓN MULTI-CLASE

- Se asignan múltiples categorías a las observaciones.
- Cada observación sólo puede ser etiquetada como una clase.

### CLASIFICACIÓN MULTI-ETIQUETA

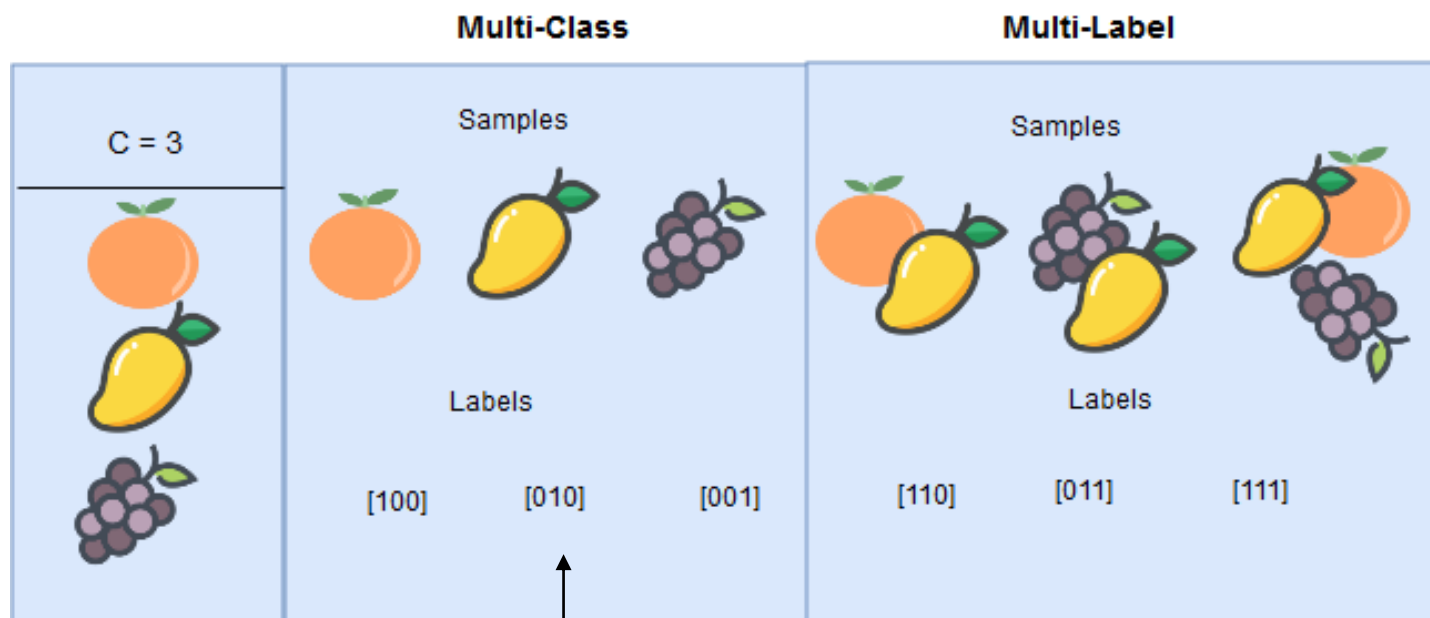
- Se pueden asignar múltiples categorías a las observaciones.
- Predice más de una etiqueta de clase para cada observación.

[Más info](#)

TIPOS DE CLASIFICACIÓN	NÚMERO DE ETIQUETAS	NÚMERO DE CLASES DIFERENTES
Clasif. binaria	1	2
Clasif. multi-clase	1	>2
Clasif. multi-etiqueta	>1	>2



## Multi-Class vs Multi-Etiqueta



One hot encoding



Conceptos generales.

Tipos de Clasificación.

## **Aplicaciones de Clasificación.**

Detectar y resolver el desequilibrio de clases.

Principales modelos de Clasificación.

Métricas de evaluación: Clasificación.

Ejemplo práctico de un modelo de Clasificación.

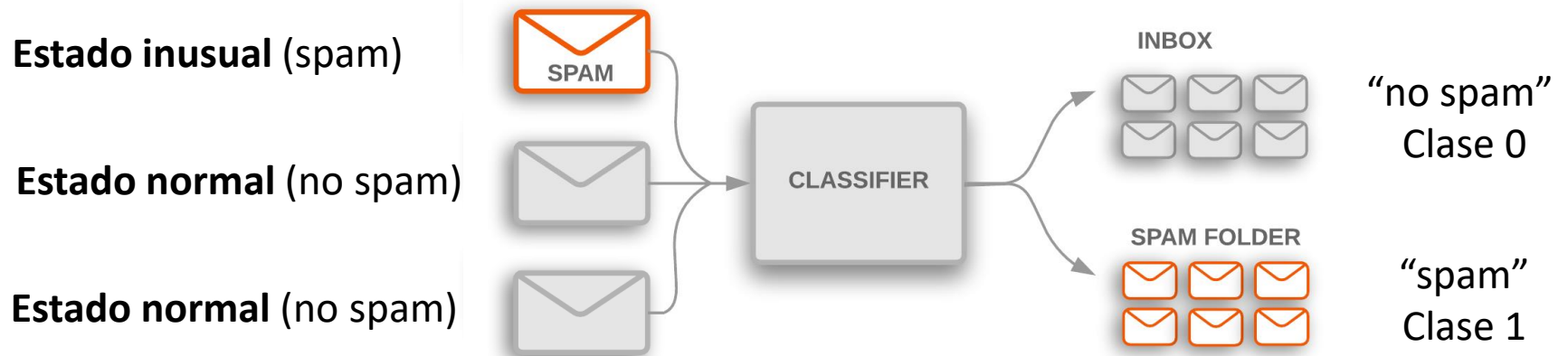


## Aplicaciones clasificación

### APLICACIONES CLASIFICACIÓN BINARIA

- Detección de spam (“spam” o “no spam”)

Las tareas de clasificación binaria implican una clase que es el **estado normal** (etiqueta de clase 0) y otra clase que es el **estado inusual** (etiqueta de clase 1).





## Aplicaciones clasificación

### APLICACIONES CLASIFICACIÓN BINARIA

- Covid detection (“COVID-19” or “non-COVID-19”)

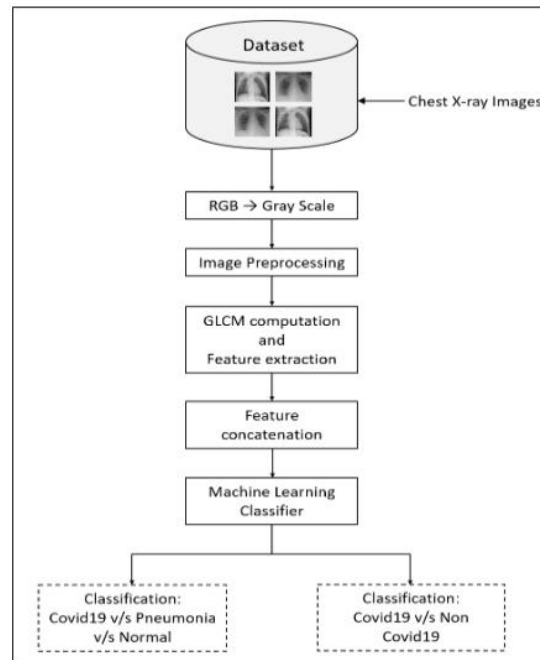


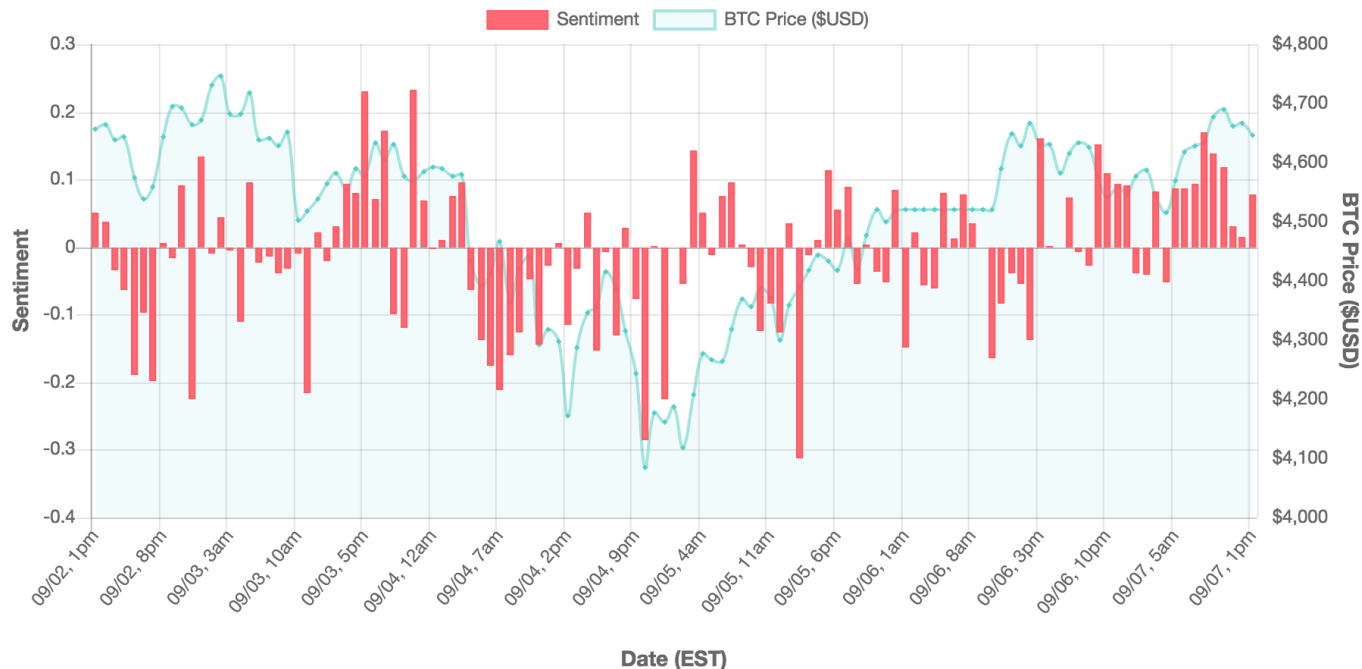
Figure 1: Block Diagram Describing Procedure of Identification of Covid19 using GLCM Feature Extraction Method



## Aplicaciones clasificación

### APLICACIONES CLASIFICACIÓN BINARIA

- **Detección de clientes descontentos** (“contento” o “no contento”)
  - Conocido también como como **Sentiment Analysis**.
  - Se utiliza el método **Natural Language Processing (NLP)**





## Aplicaciones clasificación

### APLICACIONES CLASIFICACIÓN MULTI-CLASE

- Reconocimiento de imágenes: identificar si se trata de un perro, gato, elefante o serpiente.
- Clasificación de caras (Image recognition).
- **Clasificación de las especies de plantas**
- Reconocimiento óptico de caracteres

Más de **10.000** especies de plantas diferentes



**PICTURE THIS**

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



**Diagnóstico automático de problemas**

Haz una foto de las partes enfermas de una planta para conseguir las causas del problemas y sugerencias de tratamiento.



**Identifica plantas con una foto**

Solo tienes que hacer o subir una foto de una planta para conseguir resultados de identificación de la planta precisos e instantáneos gracias a nuestra revolucionaria tecnología de inteligencia artificial.

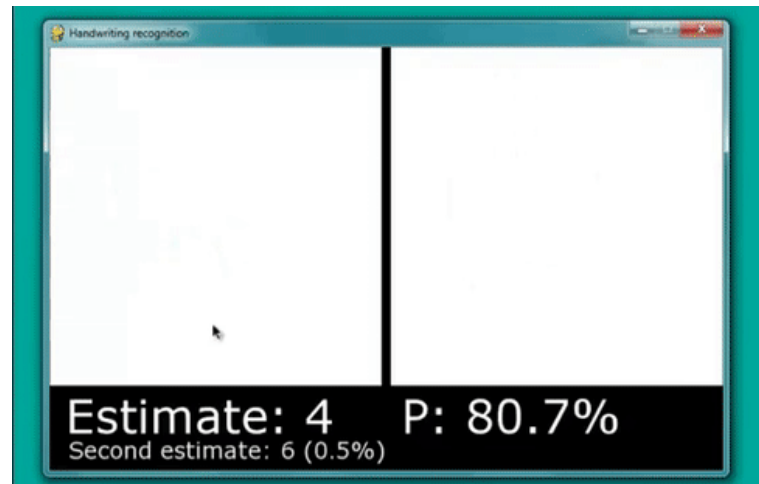
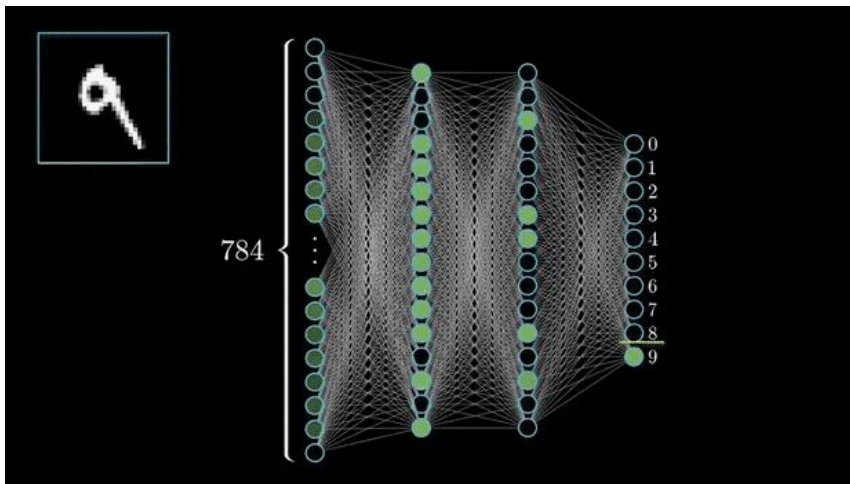




## Aplicaciones clasificación

### APLICACIONES CLASIFICACIÓN MULTI-CLASE

- Reconocimiento de imágenes: identificar si se trata de un perro, gato, elefante o serpiente.
- Clasificación de caras (Image recognition).
- Clasificación de las especies de plantas
- **Reconocimiento óptico de caracteres**



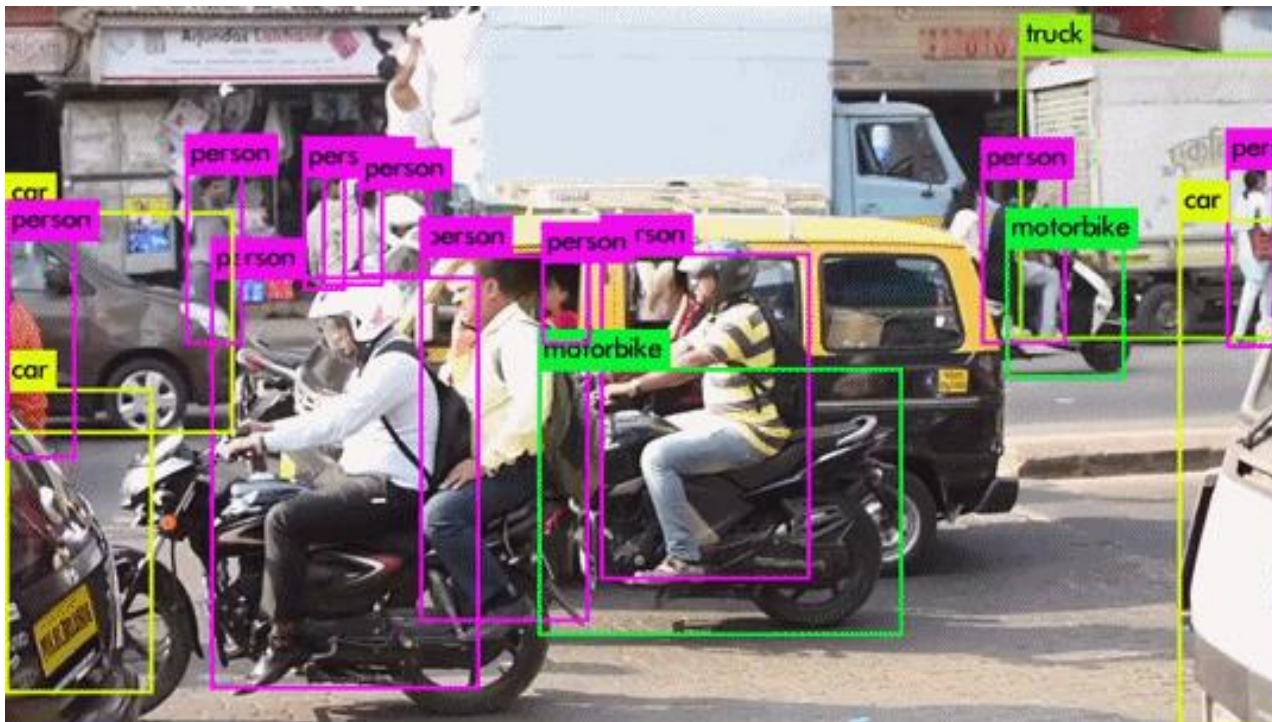
Reconocimiento óptico de caracteres



## Aplicaciones clasificación

### APLICACIONES CLASIFICACIÓN MULTI-ETIQUETA

- **Reconocimiento** de múltiples objetos conocidos en una fotografía (bicicleta, manzana, persona...).





## Aplicaciones clasificación

### APLICACIONES CLASIFICACIÓN MULTI-ETIQUETA

- **Clasificación de noticias:** un artículo puede ser sobre **deportes**, una **persona** y un **lugar** al mismo tiempo.

**Table 1.** Class label of Indonesian new

Class Label	Explanation
1	Politic
2	Law
3	Economy
4	Social
5	Culture
6	Technology
7	Life Style
8	Sport
9	Entertainment
10	Education
11	Defense
12	Health
13	Others

**Table 2.** Dataset of Indonesian news and label.

No	News Article	Class Label
1	The dynamics of dismantling the 2019 presidential candidate pair continues. Whoever is a floating figure can be juxtaposed. Moreover, a number of continually emerging are considered alternative candidates.	1
2	Today (30/11) Butet Kartaredjasa will present 138 visual works at the exhibition at the National Gallery Jakarta. The visual work takes the painted media. This way of art is deliberately done by Butet to conduct social criticism of the community and the government.	5,9
3	The Vice President of the Republic of Indonesia, Jusuf Kalla delivered a written warning to the Minister of Youth and Sports Imam Nahrawi to allocate more funds to prepare for a number of sports ahead of the 2018 Asian Games.	1,3,8



Conceptos generales.

Tipos de Clasificación.

Aplicaciones de Clasificación.

**Detectar y resolver el desequilibrio de clases.**

Principales modelos de Clasificación.

Métricas de evaluación: Clasificación.

Ejemplo práctico de un modelo de Clasificación.



## Desequilibrio de clases

**DESEQUILIBRIO ENTRE CLASES:** sucede cuando las clases (etiquetas) no se encuentran balanceadas. El modelo predecirá peor la clase con menos observaciones al tener menos instancias con las que entrenar.

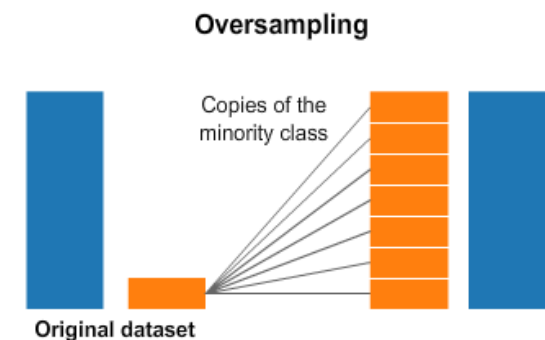
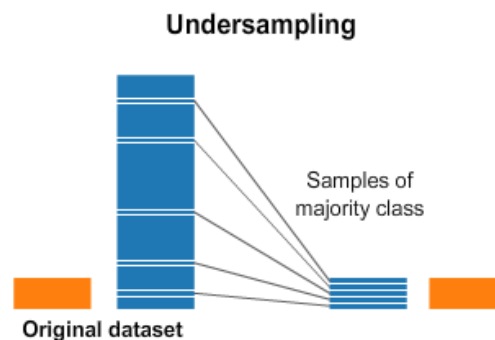
### SOLUCIONES

- **Conseguir más datos** (si es posible).
- **Sobre-muestreo (over-sampling):** duplicación de las observaciones de la clase minoritaria.
- **Sub-muestreo (under-sampling):** eliminación de datos de clase mayoritaria.
- Cambiar la **métrica de evaluación**.
- Generar **datos sintéticos** de la clase minoritaria.
- **StratifiedShuffleSplit()**

### EJEMPLOS

- Detección de fraude energético.

Referencia: [8 Tactics to Combat Imbalanced Classes in Your Machine Learning Dataset](#)



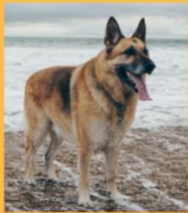




## Ejemplo

### Underfitting

Entreno al modelo con  
1 sólo raza de perro



Muestra nueva:  
¿Es perro?



**NO**



La máquina fallará en reconocer al perro por falta de suficientes muestras. No puede generalizar el conocimiento.

### Overfitting

Entreno al modelo con  
10 razas de perro color marrón



Muestra nueva:  
¿Es perro?



**NO**



La máquina fallará en reconocer un perro nuevo porque no tiene estrictamente los mismos valores de las muestras de entrenamiento.

### Prevenir el Sobre-ajuste de datos:

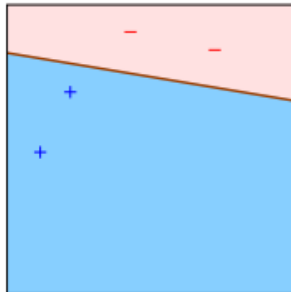
- Cantidad mínima de muestras para entrenar y validar el modelo
- Clases variadas y equilibradas en cantidad, es importante que los datos de entrenamiento estén balanceados.
- Ajuste de parámetros, deberemos experimentar sobre todo dando más/menos “tiempo/iteraciones” al entrenamiento.



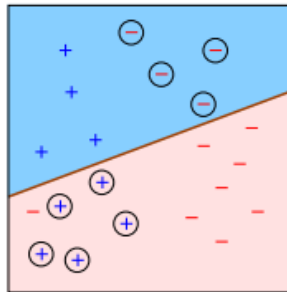
## Construir un clasificador preciso

Para un buen rendimiento del **test**, es necesario:

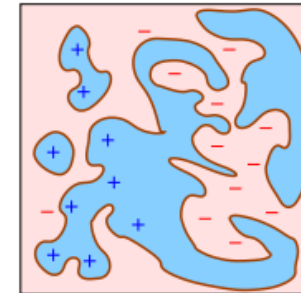
- Suficientes datos de **entreno** (así se evita el *underfitting*).
- Obtener un buen resultado de los datos de **entrenamiento**.
- El clasificador no debe ser demasiado "complejo" (así se evita el *overfitting*).



Insuficientes  
datos de entreno



Error de entreno  
demasiado alto



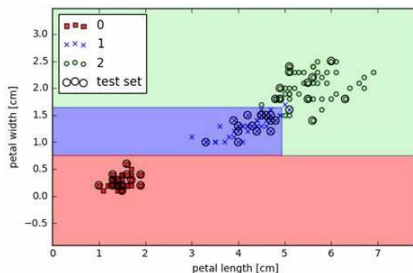
Clasificador  
demasiado complejo.

## Ejemplo 1\_IRIS: Jupyter Notebook

- El conjunto de datos contiene 3 clases, donde cada clase se refiere a un tipo de planta de iris. El numero total de muestras son 150 (divididas en 50 para cada clases), las cuales contienen 4 atributos diferentes.

Objetivo: Entrenar un modelos con los inputs (atributos) y los outputs (clases), para que aprenda a reconocer el tipo de planta.

Planta	Longitud Sépalo	Anchura del Sépalo	Longitud Pétalo	Ancho de Pétalo	Tipo de Iris
1	4.3	2	1	0.1	Setosa
2	7.9	4.4	6.9	2.5	Versicolor
3	5.84	3.05	3.76	1.2	Verginica
4	0.83	0.43	1.76	0.76	Setosa
...	...	...	...	...	...
150	3.5	3.95	1.56	2.61	Versicolor



Iris - Setosa



Iris - Versicolor



Iris - Verginica





Conceptos generales.

Tipos de Clasificación.

Aplicaciones de Clasificación.

Detectar y resolver el desequilibrio de clases.

**Principales modelos de Clasificación.**

Métricas de evaluación: Clasificación.

Ejemplo práctico de un modelo de Clasificación.



## Modelos supervisados de Clasificación

Algunos de los modelos de clasificación son:

- Support Vector Machines (SVM)
- K-nearest neighbors (k-NN)
- Árboles de decision
- Ensemble Random Forest
- Neural Network (MLP)

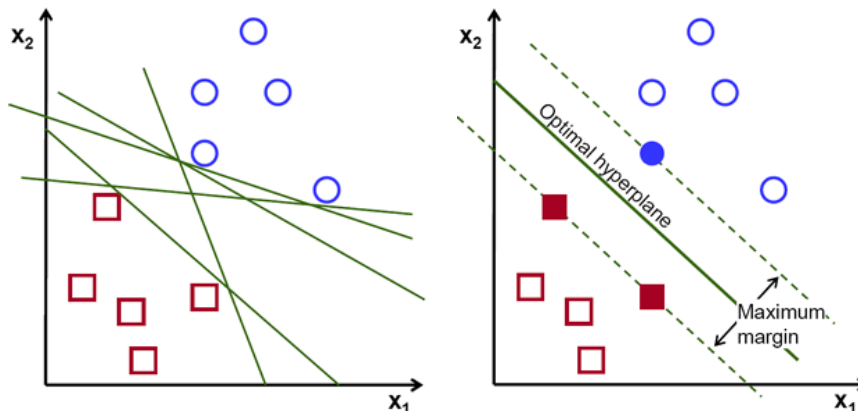
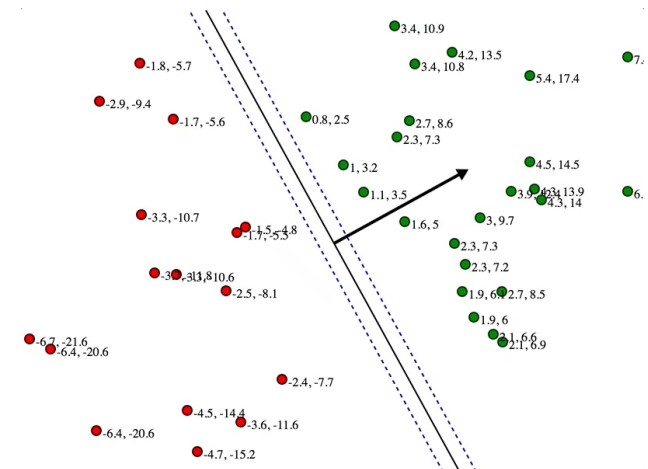
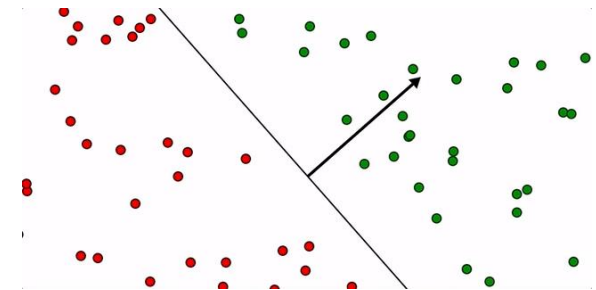


## Support Vector Machines

**DEFINICIÓN:** el objetivo de SVM es encontrar un hiperplano óptimo de decisión que clasifique los puntos de datos, maximizando el margen entre esta línea y los puntos de muestra cercanos a este hiperplano.

**VENTAJAS:** Es eficaz en espacios de grandes dimensiones y es eficiente en cuanto a la memoria.

**DESVENTAJAS:** el algoritmo no da buenos resultados si *características > muestras*.



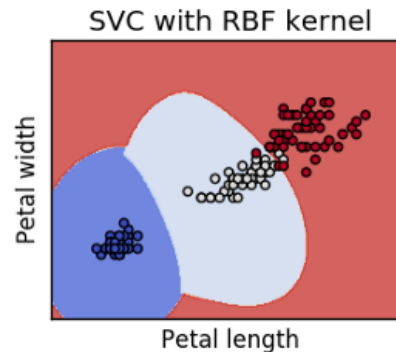
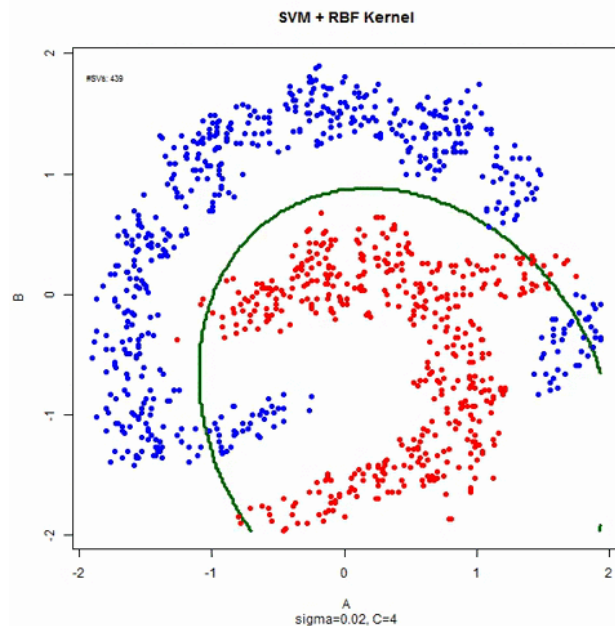


## Support Vector Machines

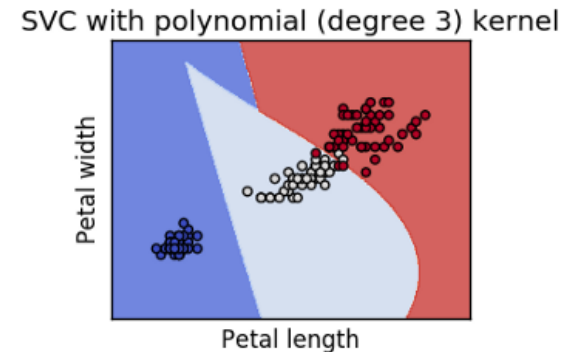
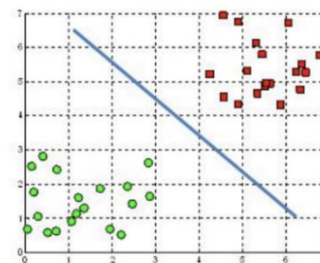
¿Pero qué pasa cuando los datos no son separables linealmente?

**SVM kernels:** método de análisis de patrones.

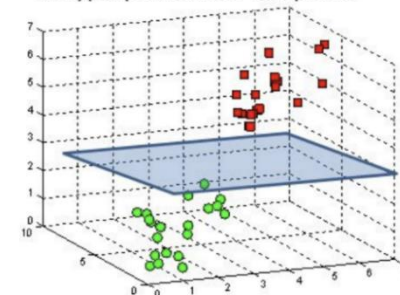
- Curvas no lineales de separación
- Casos donde los conjuntos de datos no pueden ser completamente separados.



A hyperplane in  $\mathbb{R}^2$  is a line



A hyperplane in  $\mathbb{R}^3$  is a plane



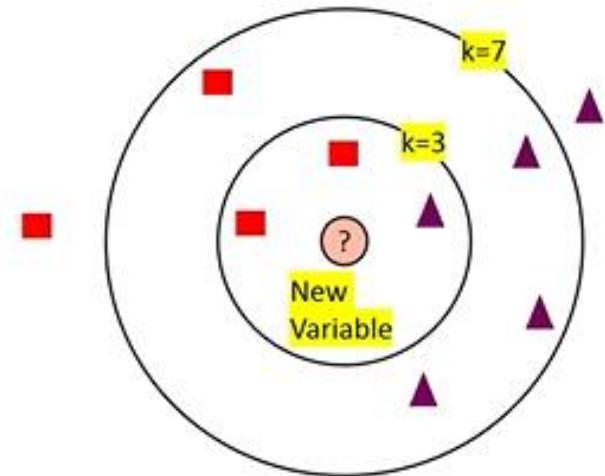


## K-NN

**DEFINICIÓN** : La clasificación se calcula a partir de la mayoría simple de los votos de los  $k$  vecinos más cercanos de cada punto, es decir, se basa en como están clasificados sus vecinos.

**VENTAJAS** : Este algoritmo es simple de implementar, robusto a los datos de entrenamiento ruidosos, y efectivo si los datos de entrenamiento son pequeños y sin muchas dimensiones.

**DESVENTAJAS** : La necesidad de determinar el valor de  $K$  y el costo del cálculo es alto ya que necesita calcular la distancia euclídea de cada instancia a todas las muestras de entrenamiento.



$K \downarrow$ , sesgo tendrá mucho ruido.  
 $K \uparrow$ , tiempo computación muy alto.



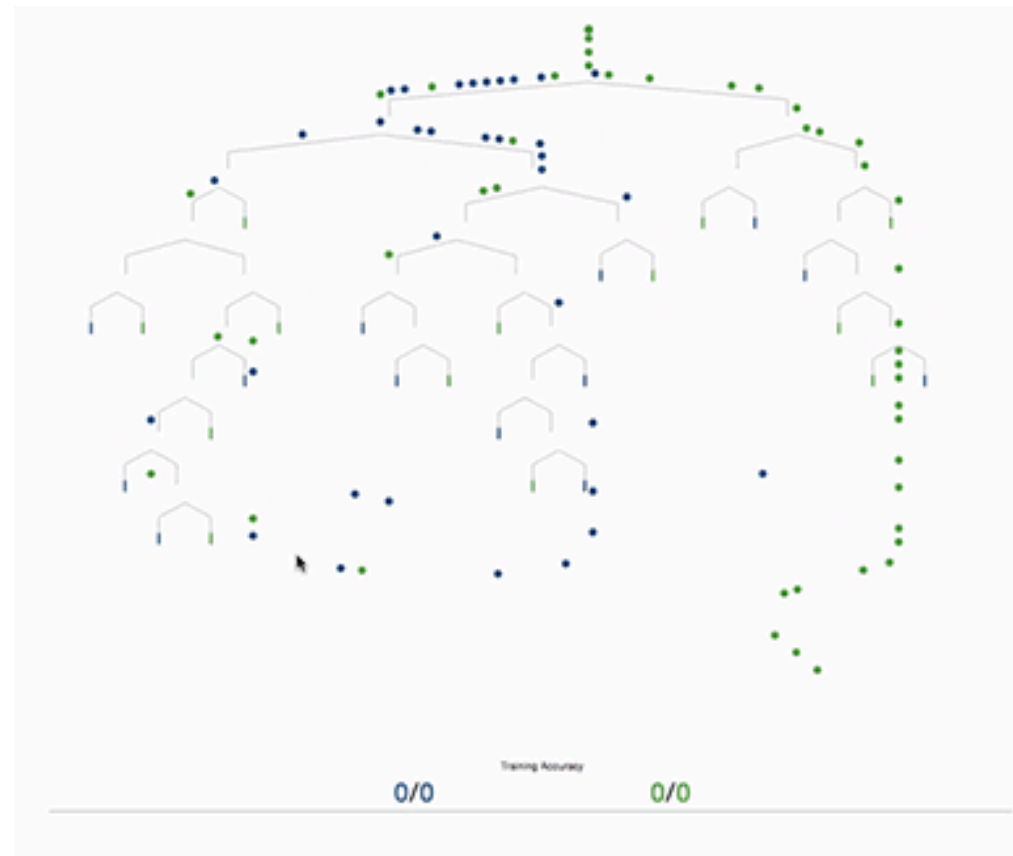
## Árboles de decisión

**DEFINICIÓN** : Dado un dato de atributos junto con sus clases, un árbol de decisión produce una secuencia de reglas que pueden utilizarse para clasificar los datos.

**VENTAJAS** : es sencillo de entender y visualizar, requiere poca preparación de datos (no es necesario normalizar) y es muy rápido de entrenar y evaluar.

### DESVENTAJAS

- Cuando hay muchas variables, riesgo de overfitting: control de la complejidad.
- Inestabilidad: un pequeño cambio en los datos puede modificar ampliamente la estructura del árbol.





## Ensemble: Random Forest

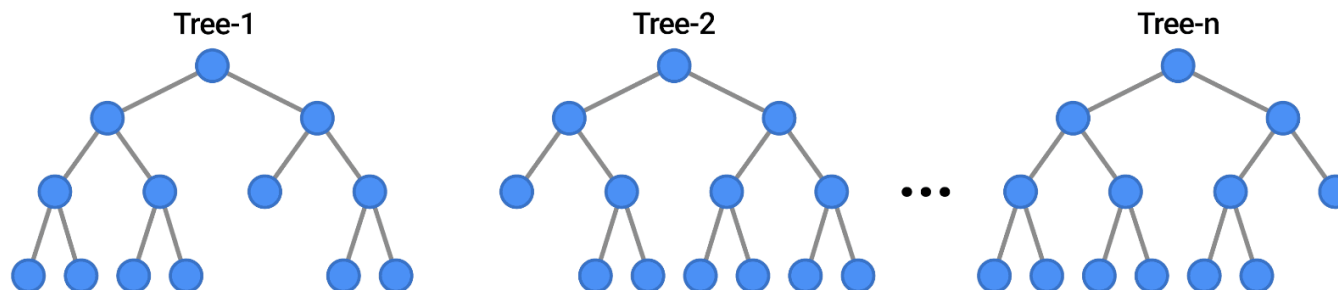
Es un **algoritmo tipo ensambladores**. Estos están formados por un **grupo de modelos predictivos** que permiten alcanzar una mejor precisión y estabilidad del modelo.

### VENTAJAS :

- Reducción del overfitting.
- Es más preciso que los árboles de decisión en la mayoría de los casos.
- Puede manejar miles de variables de entrada e identificar las más significativas (**feature selection**).

**DESVENTAJAS** : Predicción lenta en tiempo real.

### EXAMPLES







## Neural Network (MLP)

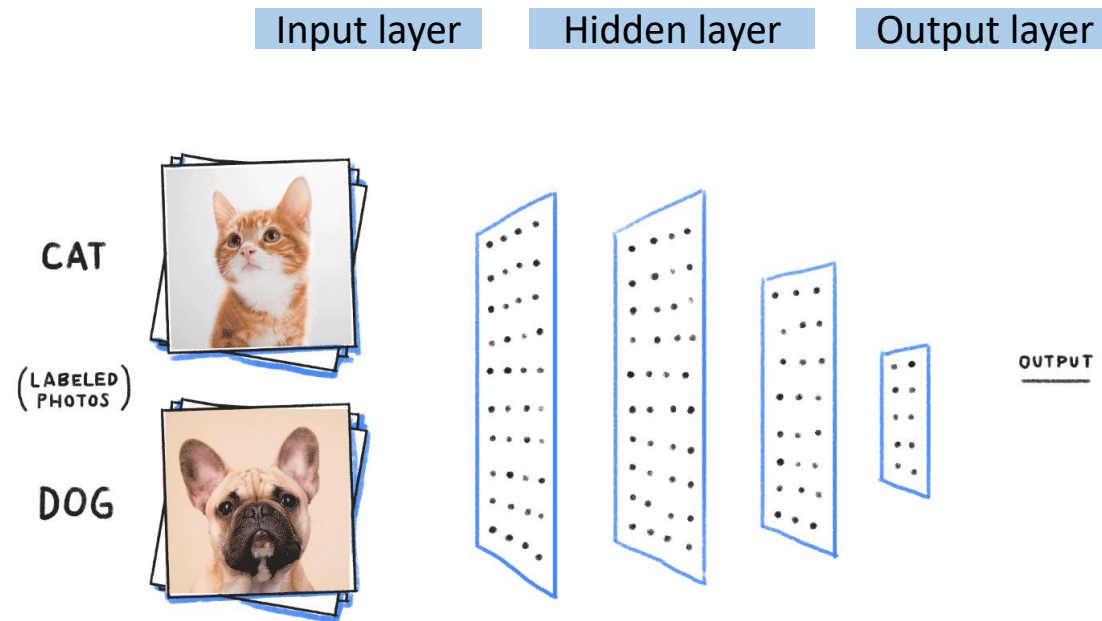
**DEFINICIÓN:** es una red neuronal artificial de tipo “feedforward” totalmente conectada.

**VENTAJAS:**

- Puede aplicarse a problemas complejos no lineales.
- Funciona bien con grandes datos de entrada.
- Predicciones rápidas tras el entrenamiento.

**DESVENTAJAS:**

Los cálculos son difíciles y requieren mucho tiempo.





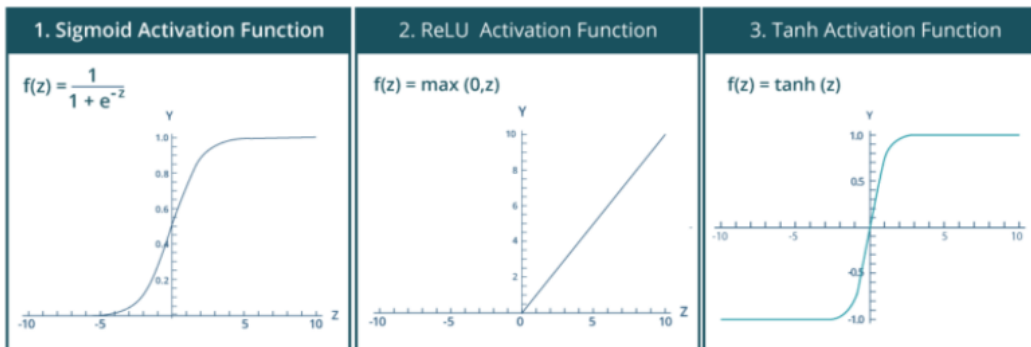


## Neural Network: Deep Learning

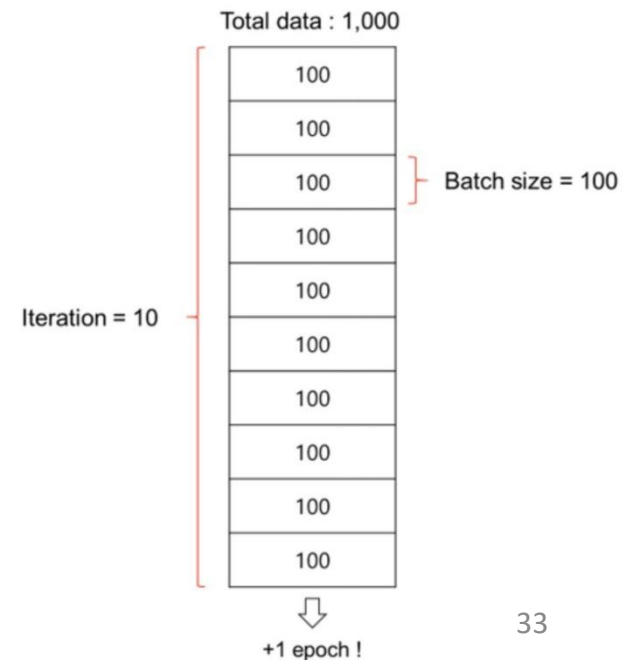
Veamos como funcionan las redes neuronales con ejemplos (deep learning)

<http://playground.tensorflow.org/>

### Funciones de activación

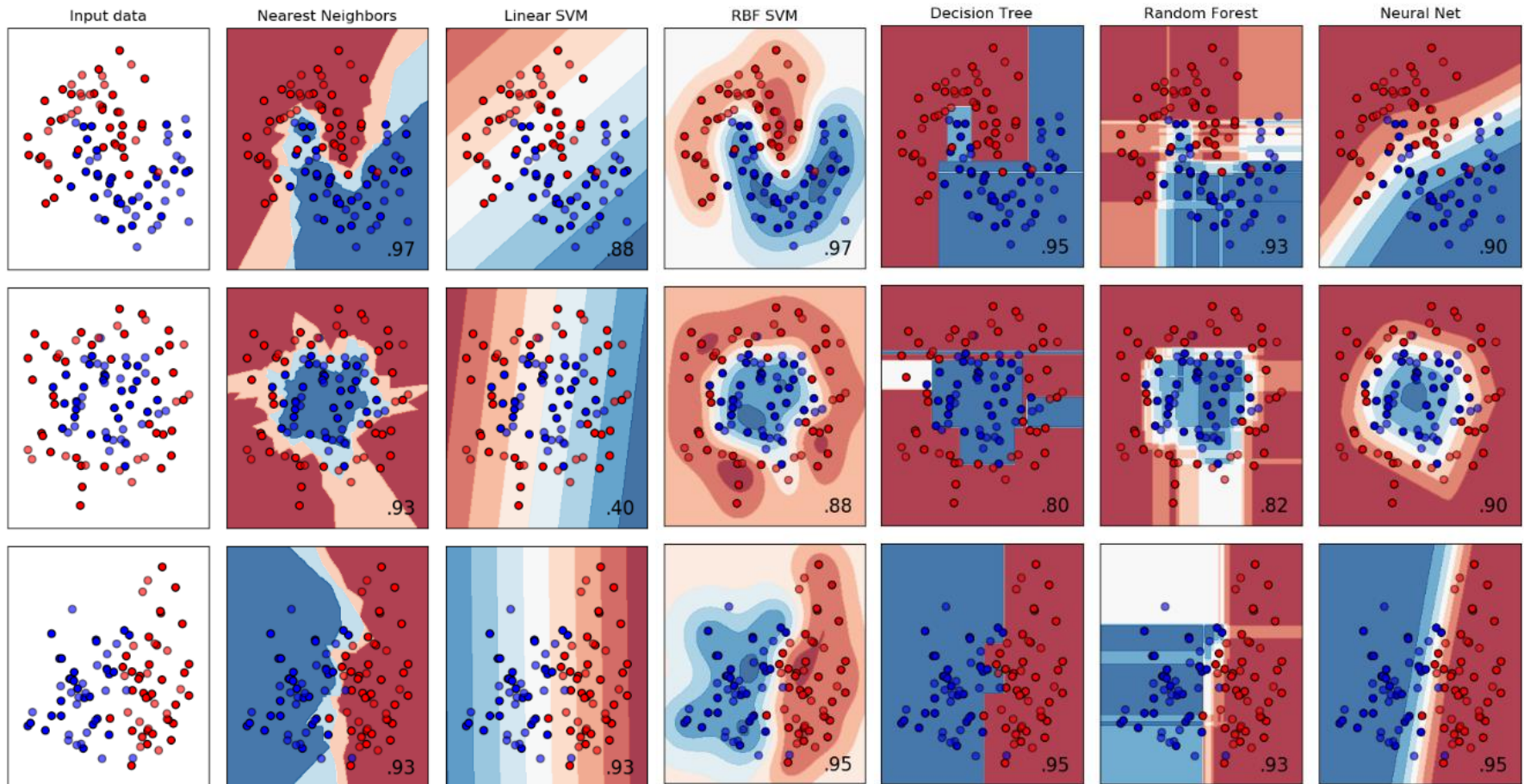


### Batch, epoch e iteraciones





## Algoritmos Supervisados de Clasificación





Conceptos generales.

Tipos de Clasificación.

Aplicaciones de Clasificación.

Detectar y resolver el desequilibrio de clases.

Principales modelos de Clasificación.

**Métricas de evaluación: Clasificación.**

Ejemplo práctico de un modelo de Clasificación.



## Evaluación de Algoritmos de Clasificación

Existen muchas formas de evaluar la precisión y desempeño del algoritmo.

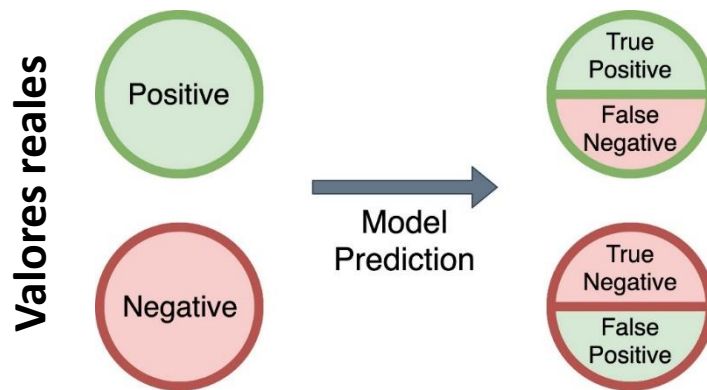
### **Métricas de Clasificación**

- Matriz de confusión
- Acierto
- Recall
- Precision
- F-1 Score
- MCC (Matthews correlation coefficient)
- AUC-ROC



## Matriz de confusión

Una matriz de confusión es una tabla que resume el rendimiento de un algoritmo de clasificación de tipo supervisado.



Acierto en la predicción

		Actual	
		Positive 1	Negative 0
Predicted	Positive 1	<b>True Positive</b>	<b>False Positive</b>
	Negative 0	<b>False Negative</b>	<b>True Negative</b>

- **Verdaderos positivos (TP):** Cuando predecimos un positivo y el resultado verdadero es positivo.
- **Falsos positivos (FP):** Cuando predecimos un positivo y el resultado verdadero es negativo.
- **Falsos negativos (FN):** Cuando predecimos un negativo y el resultado verdadero es positivo.
- **Verdaderos Negativos (TN):** Cuando predecimos un negativo y el resultado real es negativo.

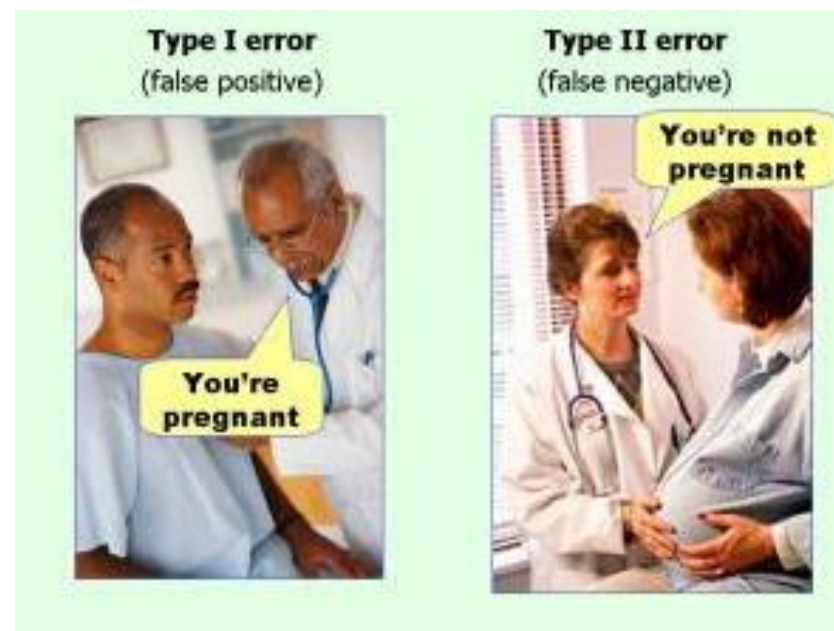




## Matriz de confusión

Una matriz de confusión es una tabla que resume el rendimiento de un algoritmo de clasificación de tipo supervisado.

		Actual	
		Positive 1	Negative 0
Predicted	Positive 1	True Positive	False Positive
	Negative 0	False Negative	True Negative



**Objetivo:** evitar falsos positivos y falsos negativos

- **Falsos positivos (FP):** Cuando predecimos un positivo y el resultado verdadero es negativo.
- **Falsos negativos (FN):** Cuando predecimos un negativo y el resultado verdadero es positivo.



## Accuracy (exactitud)

Fracción de predicciones que se realizaron correctamente en un modelo de clasificación.

$$Accuracy = \frac{\text{Total de aciertos } (TP + TN)}{\text{Total de muestras } (N)} = \frac{270 + 9}{314} = 0,88$$

Predicciones	Valor real		
	Actual - Cancer 1	Actual - NOT Cancer 0	
	Predicted - Cancer 1	TP=9 FP=5	14
	Predicted - NOT Cancer 0	FN = 30 TN = 270	300
Total	39	275	N=314

Matriz de confusion para el modelo de Hawkins

Pero... cuidado!  
(Accuracy paradox)

Es necesario utilizar otras métricas de evaluación (recall, f1, etc).

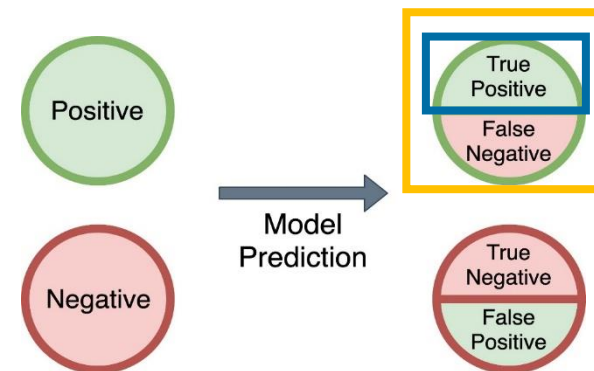


## Recall (Sensibilidad)

Número de predicciones de una clase que fueron correctamente identificadas entre el total de observaciones de esa clase.

Responde a la siguiente pregunta: **de todas las etiquetas positivas posibles, ¿cuántas identificó correctamente el modelo?**

Predicciones	Valor real		
	Actual - Cancer <b>1</b>	Actual - NOT Cancer <b>0</b>	Total
	Predicted - Cancer <b>1</b>	TP=9 FP=5	14
	Predicted - NOT Cancer <b>0</b>	FN = 30 TN = 270	300
Total	39	275	N=314



$$Recall = \frac{TP}{TP + FN} = \frac{9}{9 + 30} = 0,23$$

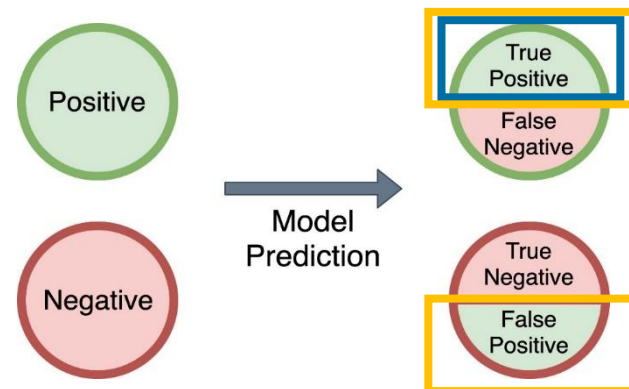




## Precisión

Identifica la frecuencia con la que un modelo predijo correctamente la clase positiva.  
 Responde a la siguiente pregunta: **de todas las predicciones que he etiquetado como cáncer, ¿cuántas eran realmente cáncer?**

		Valor real		
		Actual - Cancer <b>1</b>	Actual - NOT Cancer <b>0</b>	Total
Predicciones	Predicted - Cancer <b>1</b>	TP=9	FP=5	14
	Predicted - NOT Cancer <b>0</b>	FN = 30	TN = 270	300
	Total	39	275	N=314



$$Precision\ Clase\ 1 = \frac{TP}{TP + FP} = \frac{9}{9 + 5} = 0,64$$



## MCC (Matthews correlation coefficient)

Solo para **clasificación binaria**

MCC = 1 → Predicción perfecta  
 MCC = 0 → Predicción aleatoria  
 MCC = -1 → Predicción totalmente errónea

**Valor real**

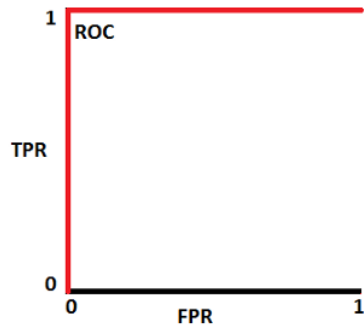
**Predicciones**

	Actual - Cancer <b>1</b>	Actual - NOT Cancer <b>0</b>	Total
Predicted - Cancer <b>1</b>	TP=9	FP=5	14
Predicted - NOT Cancer <b>0</b>	FN = 30	TN = 270	300
Total	39	275	N=314

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} = 0,34$$

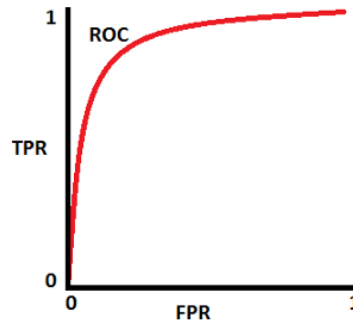


## Curva ROC / AUC



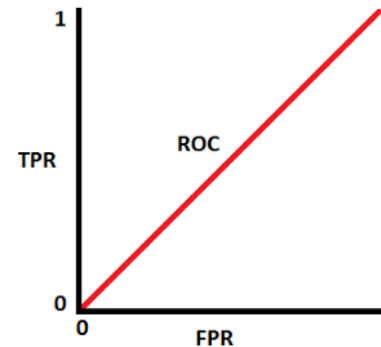
### Predicción perfecta.

- $AUC = 1$ .
- $FPR=0$ ,  $TPR=1$ .



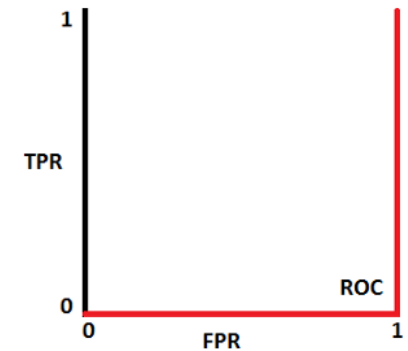
### Predicción buena/mediocre.

- $AUC = (1-0,5)$
- Si AUC es 0.85, hay un 85% de probabilidades de que el modelo distinga correctamente las clases.



### Predicción aleatoria.

- $AUC = 0,5$ .
- $FPR=TPR$ .
- Es tan exacto como voltear una moneda.
- ¡Este es el peor de los casos!



### Predicción inversa.

- $AUC = 0$ .
- $FPR=1$ ,  $TPR=0$ .
- El algoritmo predice perfectamente el inverso de las clases
- Situación altamente improbable.



Conceptos generales.

Tipos de Clasificación.

Aplicaciones de Clasificación.

Detectar y resolver el desequilibrio de clases.

Principales modelos de Clasificación.

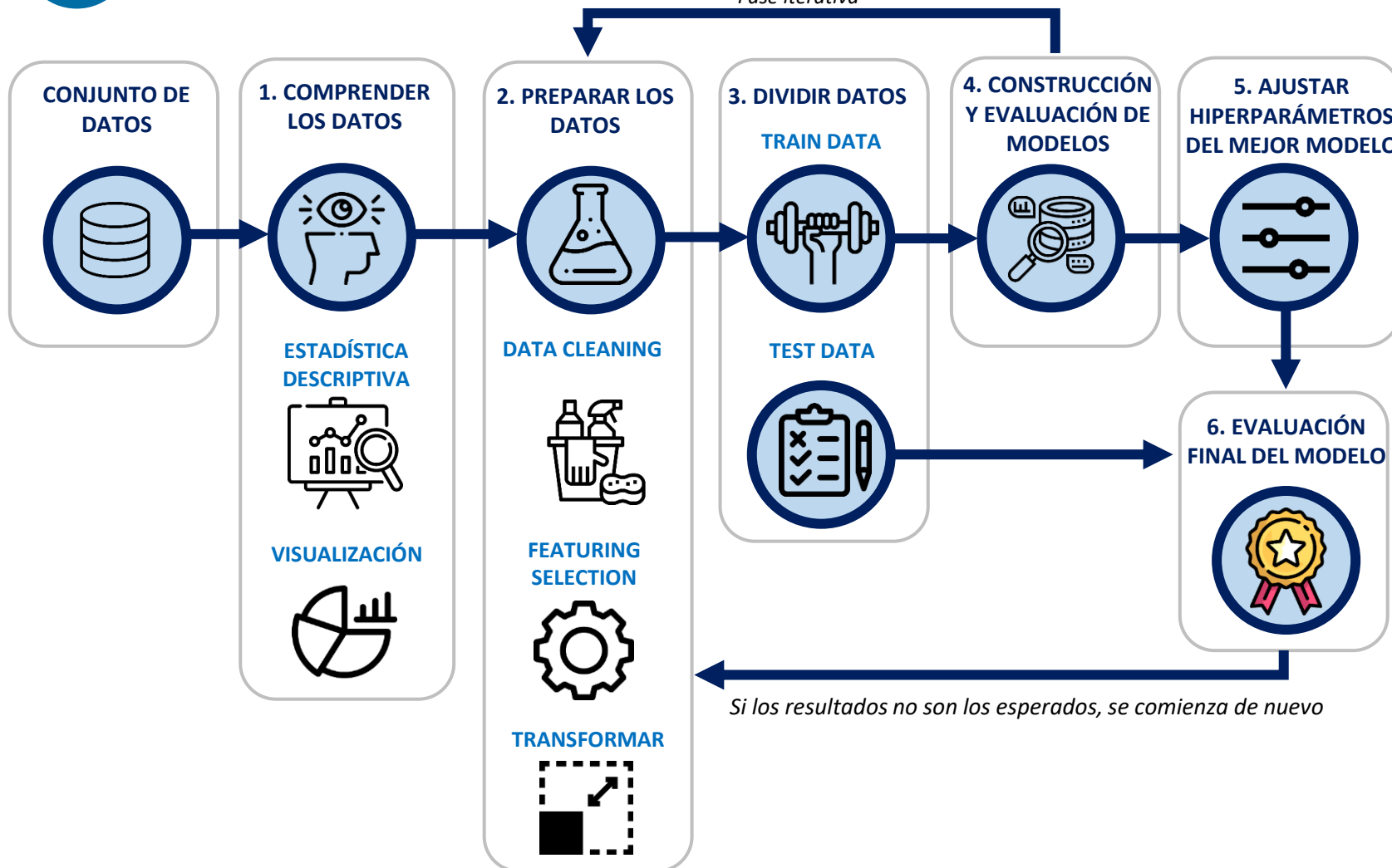
Métricas de evaluación: Clasificación.

**Ejemplo práctico de un modelo de Clasificación.**



## Recordatorio antes de la práctica

*Fase iterativa*





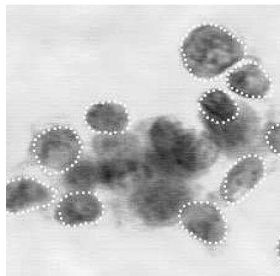
# Ejercicio 2: Detección de cáncer de mama

- **Dataset:** Las características se calculan a partir de una imagen digitalizada de una aspiración con aguja fina (FNA) de una masa mamaria. Describen características de los núcleos celulares presentes en la imagen.

## Atributos 30

radio (media):	6.981	28.11
textura (media):	9.71	39.28
perímetro (media):	43.79	188.5
área (media):	143.5	2501.0
suavidad (media):	0.053	0.163
compacidad (media):	0.019	0.345
concavidad (media):	0.0	0.427
puntos cóncavos (media):	0.0	0.201
simetría (media):	0.106	0.304
dimensión fractal (media):	0.05	0.097
radio (error estándar):	0.112	2.873
textura (error estándar):	0.36	4.885
perímetro (error estándar):	0.757	21.98
área (error estándar):	6.802	542.2
Suavidad (error estándar):	0.002	0.031
compacidad (error estándar):	0.002	0.135
concavidad (error estándar):	0.0	0.396
puntos cóncavos (error estándar):	0.0	0.053
simetría (error estándar):	0.008	0.079
dimensión fractal (error estándar):	0.001	0.03
radio (peor):	7.93	36.04
textura (peor):	12.02	49.54
perímetro (peor):	50.41	251.2
zona (peor):	185.2	4254.0
suavidad (peor):	0.071	0.223
compacidad (peor):	0.027	1.058
concavidad (peor):	0.0	1.252
puntos cóncavos (peor):	0.0	0.291
simetría (peor):	0.156	0.664
dimensión fractal (peor):	0.055	0.208

- N. Datos: 569 Datos
- N. Atributos: 30 atributos
- Clase:
  - 1. Maligno
  - 2. Benigno



•WN Street, WH Wolberg y OL Mangasarian. Extracción de características nucleares para el diagnóstico de tumores de mama. IS&T/SPIE 1993 Simposio internacional sobre imágenes electrónicas: ciencia y tecnología, volumen 1905, páginas 861-870, San José, CA, 1993.



## Machine Learning y Deep Learning con Python

- Librerías para machine learning



- Fácil de utilizar (intuitivo)
- Contiene una amplia variedad de modelos de clasificación, regression, clustering y dimensional reduction.

<https://scikit-learn.org/stable/>

- Librerías para deep learning



Keras







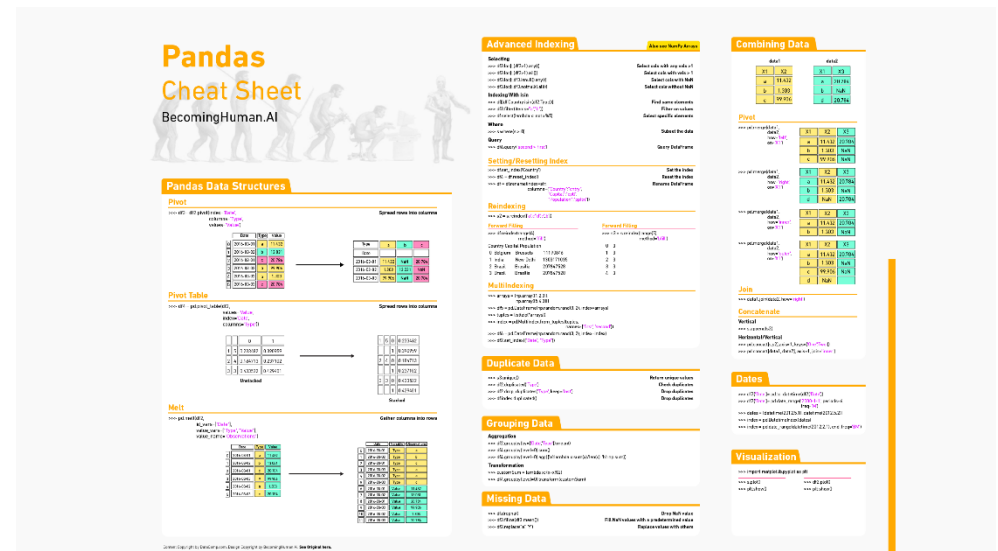
## Bibliografía

- **“Advanced Data Analytics for Power Systems”** - Ali Tajer, Samir M. Perlaza, H. Vincent Poor.
- **“Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems”** Aurelien Geron
- **“Python Data Science Handbook: Tools and Techniques for Developers: Essential Tools for working with Data”**. Jake VanderPlas



## Más información

- [Cheat-sheets de Scikit-learn Machine Learning, Pandas, Matplotlib...](#)
- [Aprendizaje automático \(Coursera\)](#)



The collage features several cheat sheets for data science tools:

- Pandas Cheat Sheet:** Titled "Becoming Human AI", it covers Pandas Data Structures, Pivot, Pivot Table, Melt, and other basic operations.
- Advanced Indexing:** Details various indexing methods like iloc, loc, ix, and iat.
- Combining Data:** Explains how to merge datasets using concat, join, and merge functions.
- Dates:** Provides shortcuts for working with time series data.
- Visualization:** Lists common plotting functions from Matplotlib and Seaborn.
- Missing Data:** Shows how to handle missing values with isnull, notnull, and dropna.
- Grouping Data:** Covers data aggregation using groupby.
- Duplicate Data:** Shows how to identify and remove duplicate rows.
- Pivot Table:** Provides a detailed guide on creating and manipulating pivot tables.