

Curso de Decision Trees y Random Forest con Python y Scikit-learn

Layla Scheli



Layla Scheli



Analista de BI, Big Data y Data Science.



Experiencia en múltiples áreas de la analítica de datos.



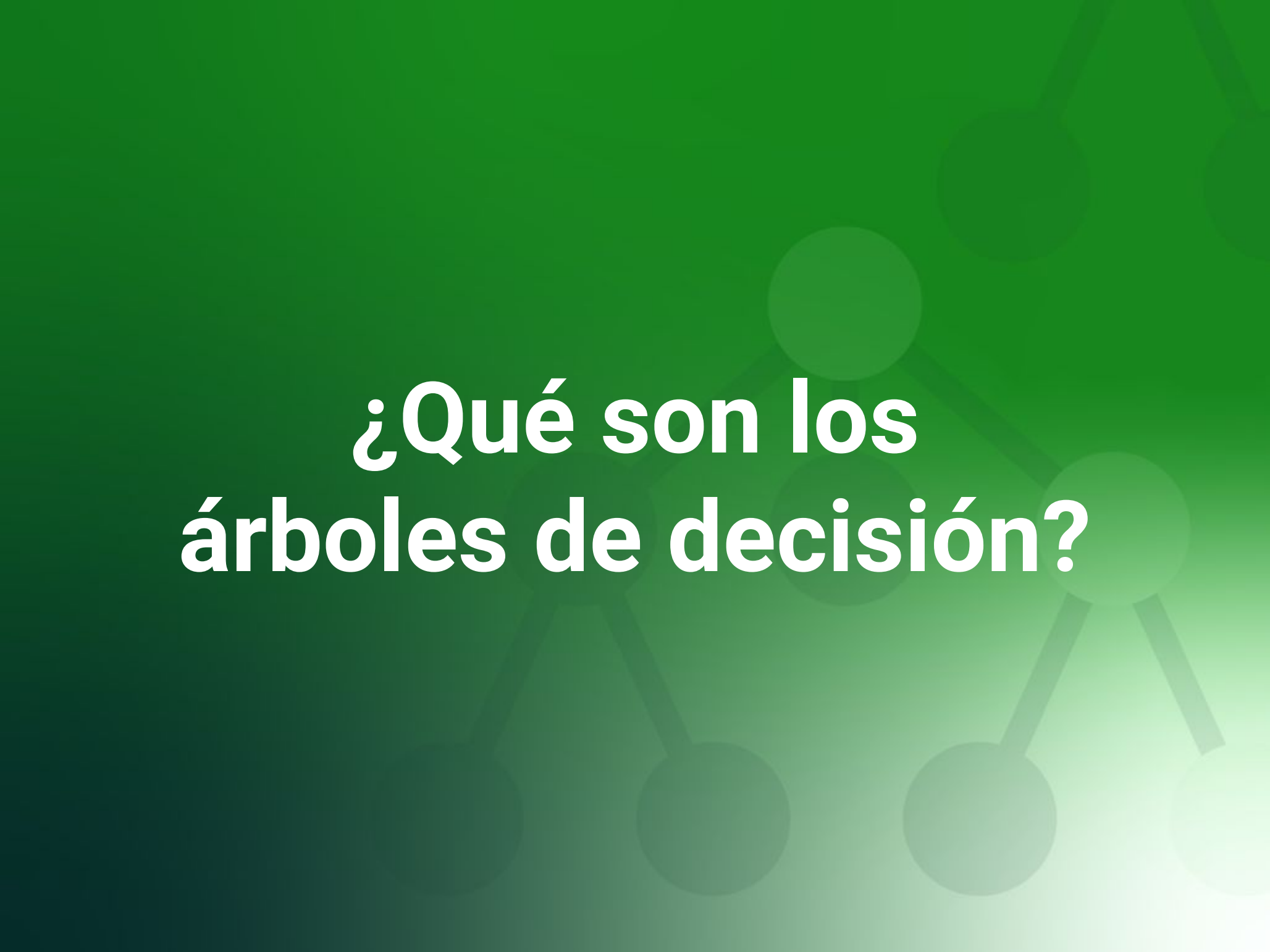
Profesora e investigadora.



Conocimientos previos

- Matemáticas para machine learning.
- Visualización de datos.
- Análisis exploratorio de datos con Python.
- Regresiones lineales y logísticas.





**¿Qué son los
árboles de decisión?**

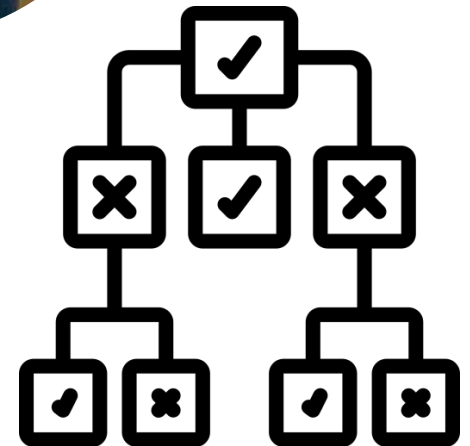
Árboles de decisión

- Aprendizaje supervisado.
- Ampliamente extendido.
- Diferentes algoritmos derivan de los árboles de decisión.



Árboles de decisión

- Primeras versiones por **Leo Breiman**.
- Utilizados para **clasificación y regresión**.



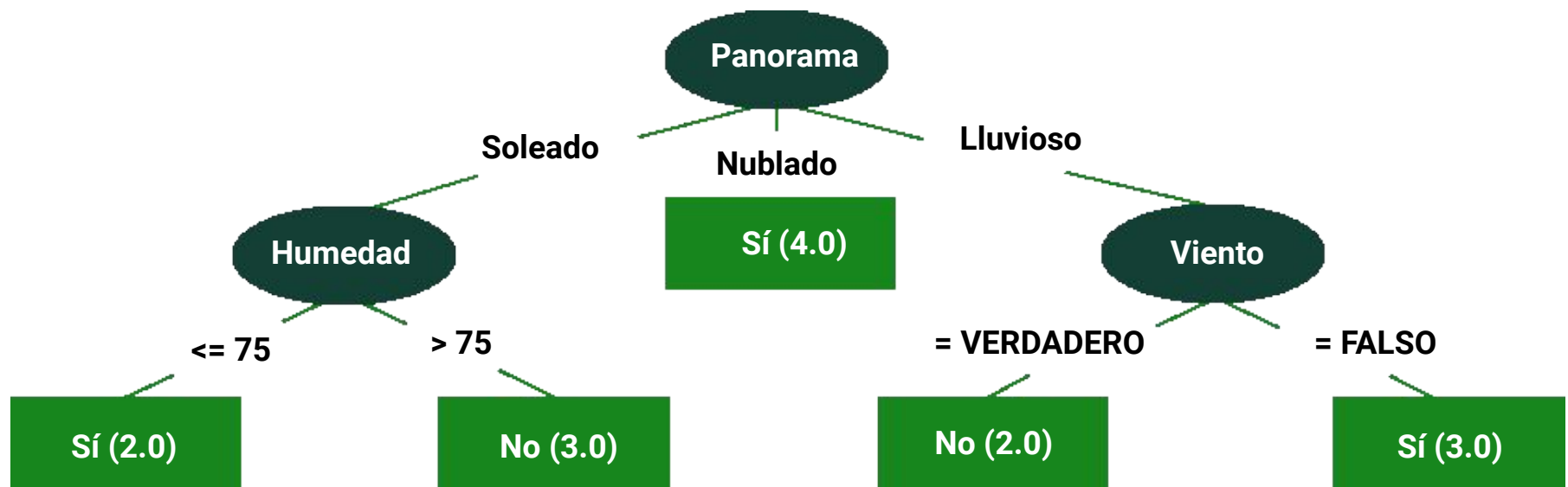
Árboles de decisión

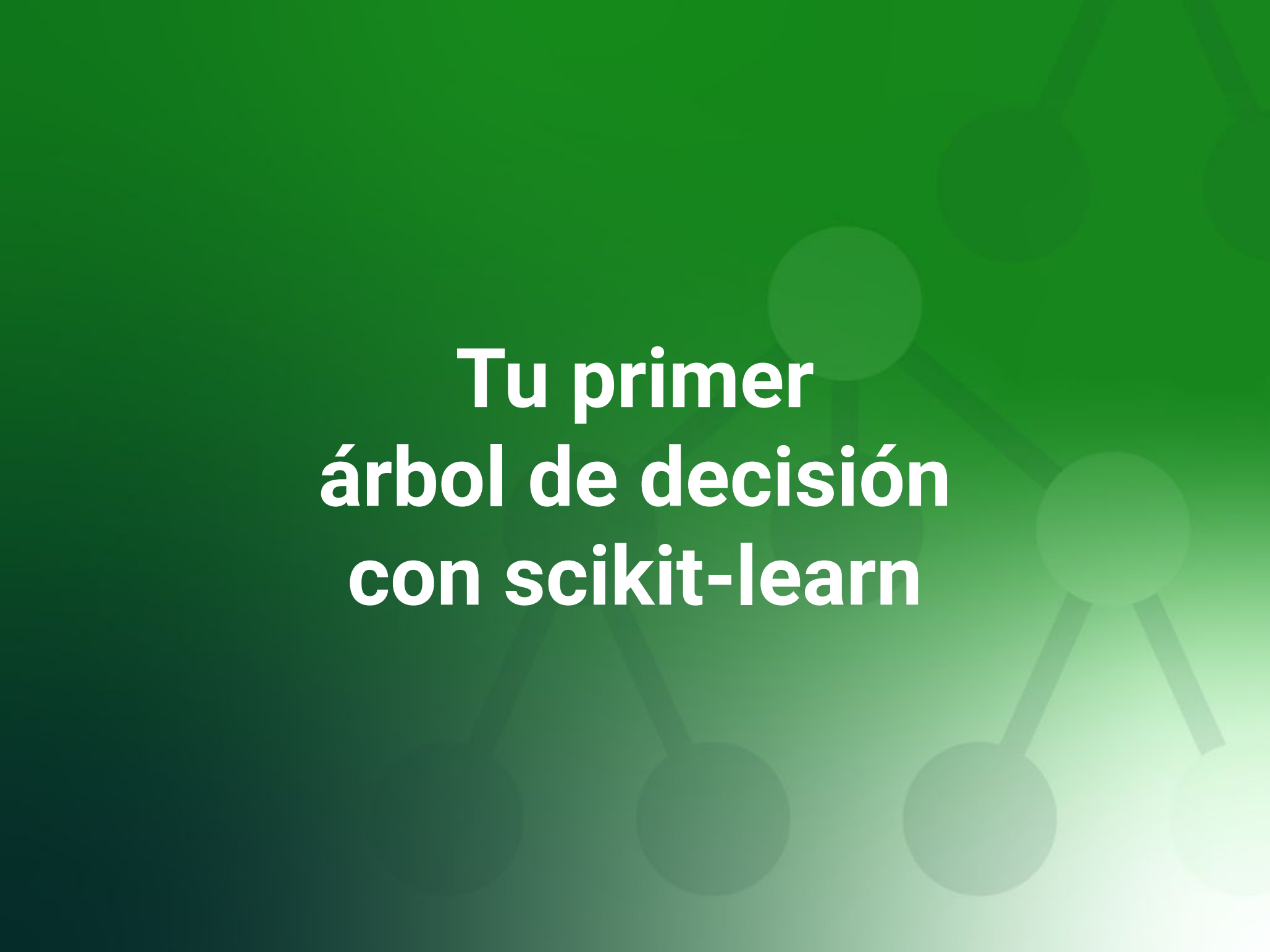
- Aprenden de los datos generando **reglas de tipo if-else**.
- Divisiones conocidas como **nodos**.
- Cuando un nodo no conduce a nuevas divisiones, se le denomina **hoja**.



Ejemplo de árbol de decisión

Mejor clima para jugar tenis





Tu primer árbol de decisión con scikit-learn

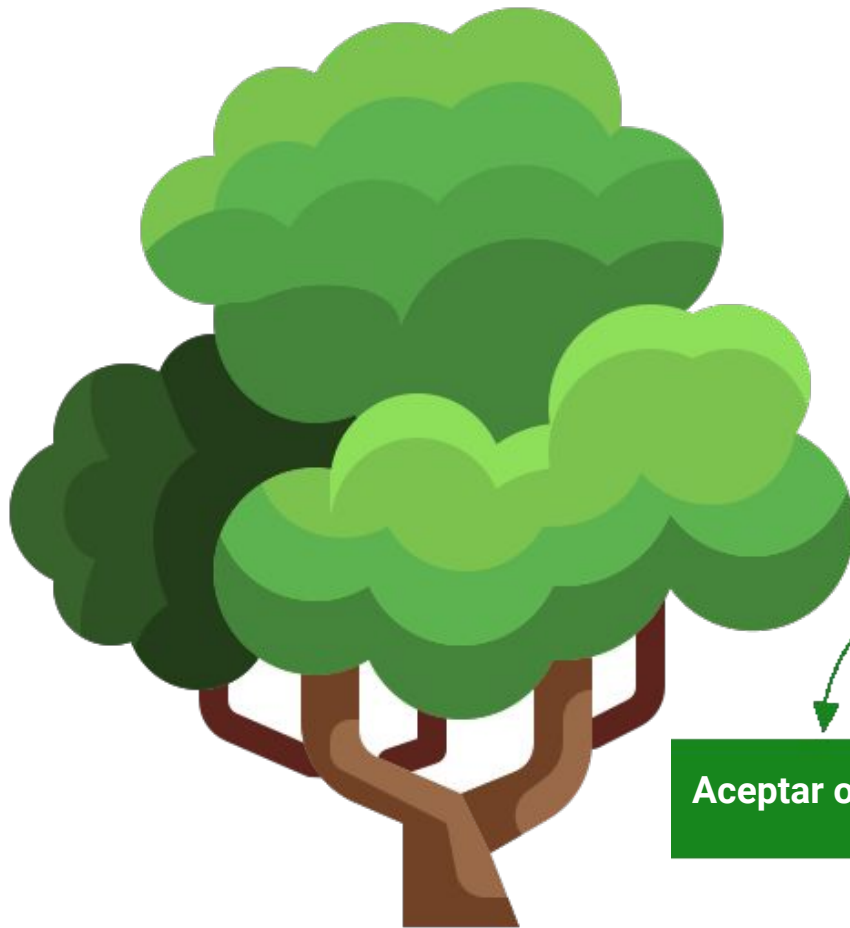
Análisis de datos para tu primer árbol de decisión



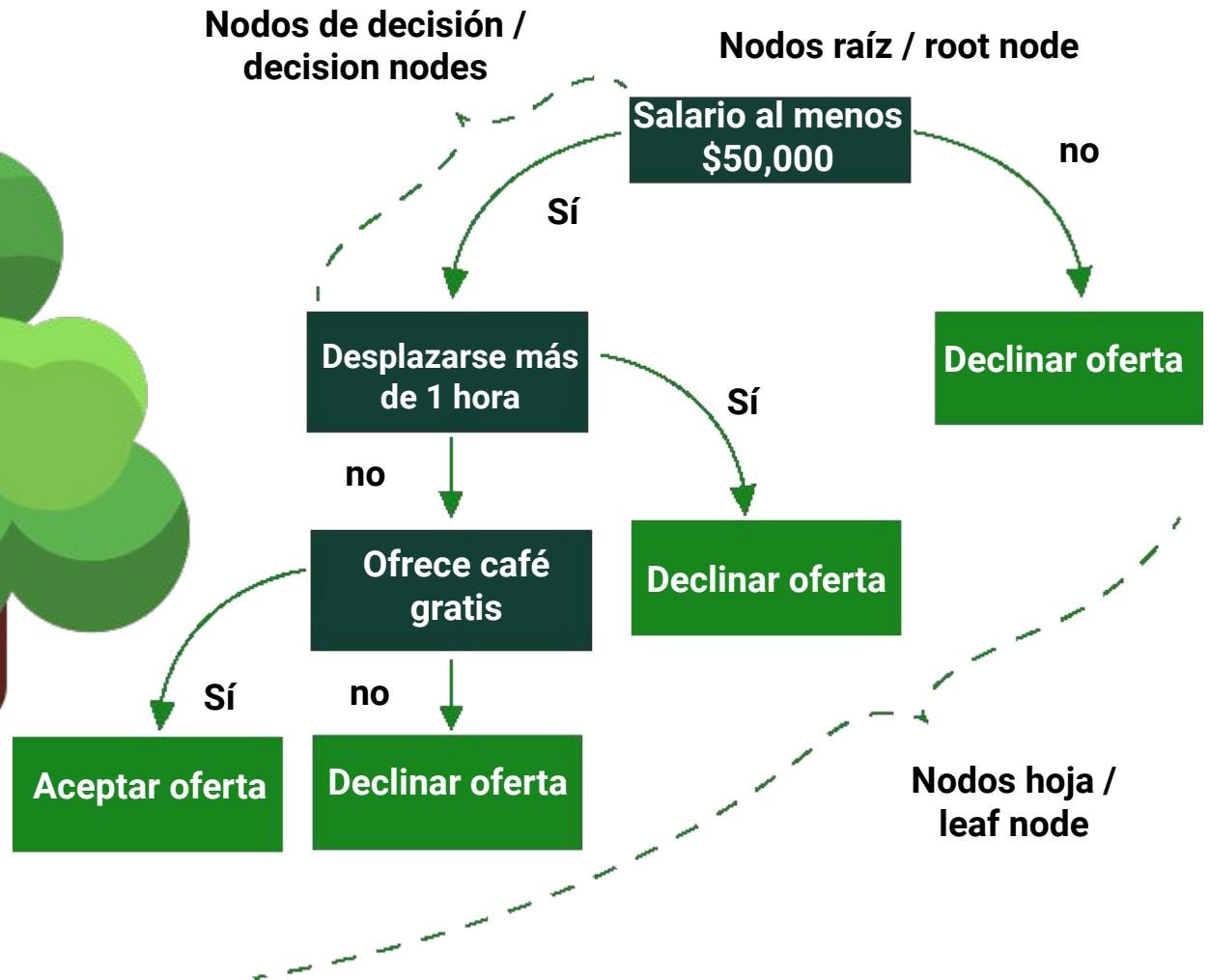
Entrenamiento y evaluación de árbol de decisión con scikit-learn



**¿Cómo funcionan los
árboles de decisión?**



**Árbol de decisión:
¿debería aceptar una
nueva propuesta laboral?**



Terminología

Nodo raíz	Poda (Pruning)
División	Rama / Subárbol
Nodo de decisión	Nodo madre/padre e hijo
Nodo de hoja o terminal	



**¿Cuándo usar
árboles de decisión?**

Ventajas

- Algoritmo de caja blanca.
- Resultados fáciles de interpretar y de entender.
- Las combinaciones de los mismos pueden dar resultados muy certeros. Por ejemplo, *random forest*.



Desventajas

- Tienden al sobreajuste u overfitting.
- Se ven influenciadas por los outliers.
- Crear árboles demasiado complejos puede conllevar que no se adapten bien a los nuevos datos.
- Se pueden crear árboles sesgados si una de las clases es más numerosa.



¿Cuándo usar árboles de decisión?

- Sencillo y fácil de entender.
- Funcionan bastante bien con grandes conjuntos de datos.
- Relativamente robusto.
- Es un método muy útil para analizar datos cuantitativos.
- Aplica para clasificación y regresión.

Ejemplo:

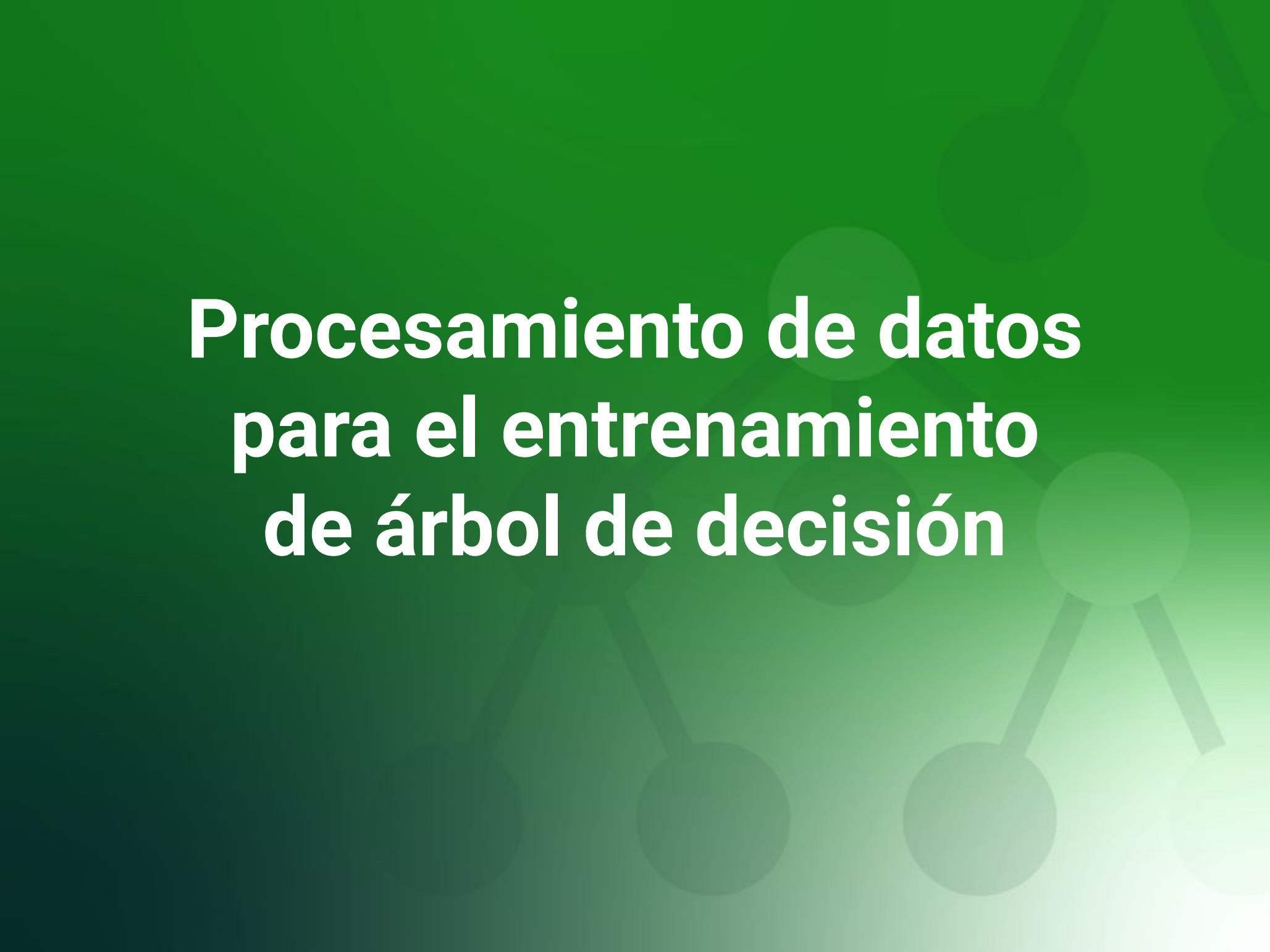
<https://economipedia.com/definiciones/arbol-de-decision-en-valoracion-de-inversiones.html>

Árbol de decisión para problemas de clasificación

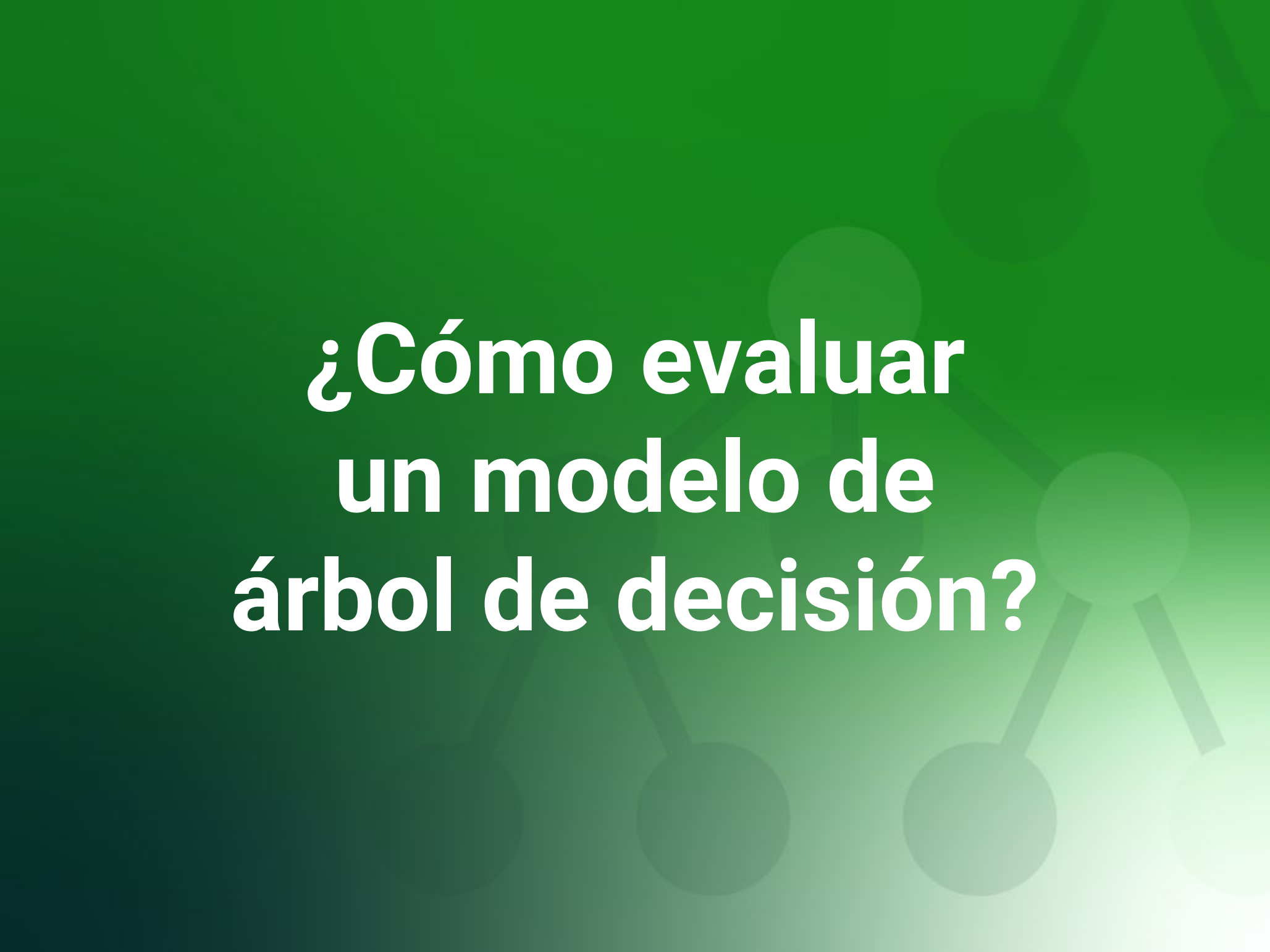
Análisis exploratorio de datos para árbol de decisión



Procesamiento de datos para el entrenamiento de árbol de decisión



Entrenamiento de modelo de clasificación con árbol de decisión



**¿Cómo evaluar
un modelo de
árbol de decisión?**

Matriz de confusión

- Permite visualizar el desempeño de un algoritmo de aprendizaje supervisado.
- Cada **columna** representa el número de predicciones de cada clase.
- Cada **fila** representa a las instancias en la clase real.

VALORES PREDICCIÓN	VALORES REALES	
	Verdaderos positivos	Falsos positivos
Falsos negativos		
Verdaderos negativos		

Matriz de confusión

- En términos prácticos nos permite ver **qué tipos de aciertos y errores** está teniendo nuestro modelo.

VALORES PREDICCIÓN	VALORES REALES	
	Positivo	Negativo
Positivo	Verdaderos positivos	Falsos positivos
Negativo	Falsos negativos	Verdaderos negativos

Interpretación de matriz de confusión

VALORES PREDICCIÓN	VALORES REALES	
	Positivo	Negativo
Positivo	Verdaderos positivos	Falsos positivos
Negativo	Falsos negativos	Verdaderos negativos

- **Verdadero Positivo (TP):** predije que era positivo y lo era.
- **Verdadero Negativo (TN):** predije que era falso y lo era.
- **Falso Positivo (FP):** predije que era positivo, pero resultó ser negativo.
- **Falso Negativo (FN):** predije que era negativo, pero resultó siendo positivo.

Interpretación de matriz de confusión

VALORES PREDICCIÓN	VALORES REALES	
	Positivo	Negativo
Positivo	Verdaderos positivos	Falsos positivos
Negativo	Falsos negativos	Verdaderos negativos

- **Verdaderos positivos como negativos son aciertos.**
- **Falsos negativos como positivos son errores.**

Exactitud o accuracy

- Cercanía al resultado de una medición del valor verdadero.
- En términos estadísticos, la exactitud está relacionada con el sesgo de una estimación.



Exactitud o accuracy

- Proporción entre los positivos reales predichos y todos los casos positivos.
- En forma práctica, la exactitud es el % **total de elementos clasificados correctamente**.



Fórmula accuracy

$$\frac{(VP + VN)}{(VP + FP + FN + VN)} * 100$$

Precisión

- Dispersión del conjunto de valores obtenidos a partir de mediciones repetidas de una magnitud.
- Cuanto menor es la dispersión, mayor la precisión.
- Proporción entre el número de predicciones correctas (tanto positivas como negativas) y el total de predicciones.



Precisión

- En forma práctica, es el **porcentaje de casos positivos detectados**.
- Sirve para medir la **calidad del modelo** de machine learning en **clasificación**.



Fórmula precisión

$$\frac{(VP)}{(VP+FP)}$$

Sensibilidad

- *Recall, sensitivity* o **tasa de verdaderos positivos**.
- Proporción de **casos positivos** que fueron correctamente identificados.



Fórmula sensibilidad

$$\frac{VP}{(VP + FN)}$$

Especificidad

- Tasa de verdaderos negativos.
- Proporción de **casos negativos** que fueron correctamente identificados.

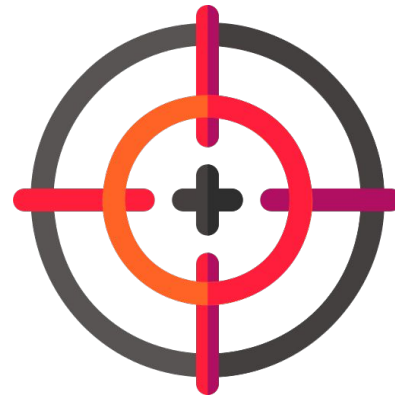


Fórmula especificidad

$$\frac{VN}{VN + FP}$$

F1-score

- Resume la **precisión** y **sensibilidad** en una sola métrica.



Fórmula F1-score

$$2 * \frac{precision * recall}{precision + recall}$$

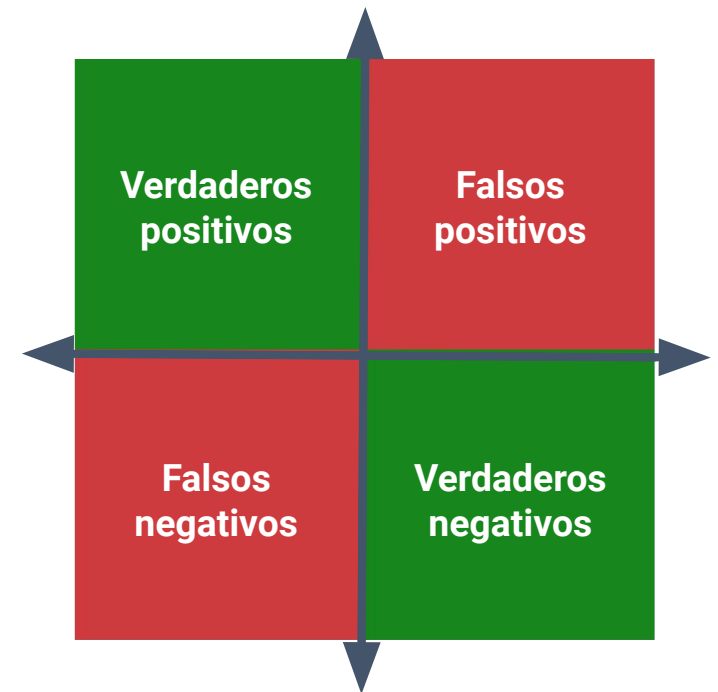
En resumen

$$Precision = \frac{TruePositive}{ActualResults} \text{ or } \frac{TruePositive}{TruePositive + FalsePositive}$$

$$Recall = \frac{TruePositive}{PredictedResults} \text{ or } \frac{TruePositive}{TruePositive + FalseNegative}$$

$$Accuracy = \frac{TruePositive + TrueNegative}{Total}$$

$$Specificity = \frac{TrueNegatives}{TrueNegatives + FalsePositives}$$



Evaluación del modelo de árbol de decisión



**¿Qué son los
random forest
o bosques aleatorios?**

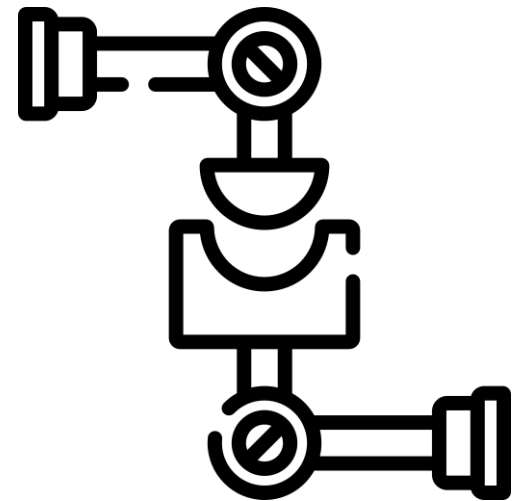
Random forest

- Bosques aleatorios.
- **Ensamble** en machine learning en donde se **combinan árboles de decisión**.



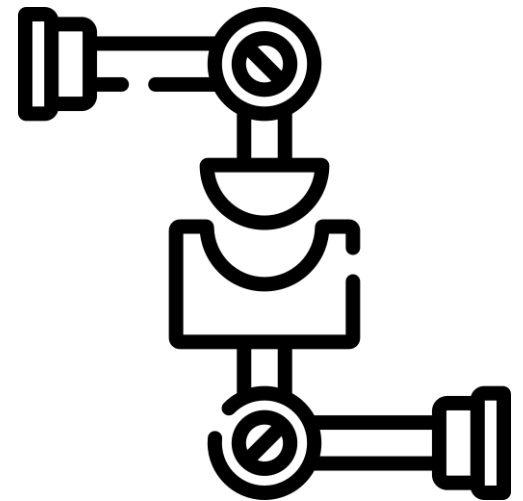
¿Qué es un ensamble?

- También conocidos como métodos combinados.
- Intentan ayudar a mejorar el **rendimiento** de los modelos de machine learning.



¿Qué es un ensamble?

- Proceso mediante el cual se construyen estratégicamente varios modelos de machine learning para resolver un problema particular.



Random forest


- Al igual que el árbol de decisión, es un algoritmo de aprendizaje supervisado.
- Utilizados en problemas de clasificación.
- También puede usarse para regresión.





Tu primer random forest con scikit-learn

Análisis de datos para tu primer random forest



Entrenamiento de tu primer modelo de random forest con scikit-learn

Evaluación de tu primer modelo de random forest con scikit-learn

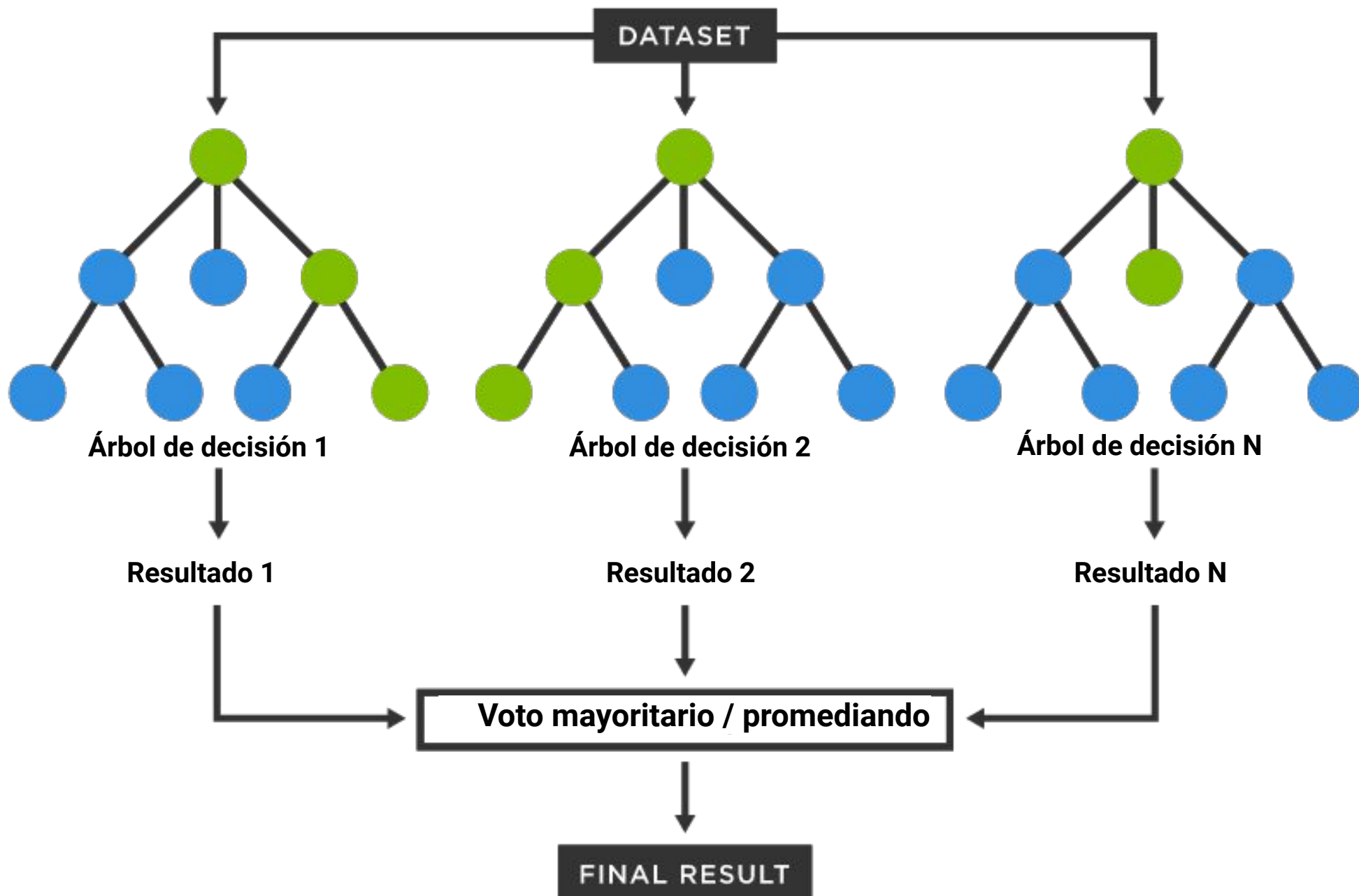


**¿Cómo funcionan los
random forest?**

Problemas de overfitting

Uno de los problemas con la creación de un árbol de decisión es que si le damos la **profundidad suficiente**, tiende a “memorizar” las soluciones en vez de generalizar.

Es decir, a tener **overfitting**. ❌





**¿Cuándo debería utilizar un
algoritmo de random forest?**

Ventajas

- Funciona bien aún sin ajuste de hiper parámetros.
- Al utilizar múltiples árboles se reduce considerablemente el riesgo de overfitting.
- Suele mantenerse estable frente a nuevas muestras de datos.

Desventajas

- Es mucho más “costoso” de crear y ejecutar que “un solo árbol” de decisión.
- No funciona bien con datasets pequeños.
- Puede requerir muchísimo tiempo de entrenamiento.
- Su interpretación a veces se vuelve compleja.

¿Cuándo usar random forest?

- Rápido y fácil de aplicar.
- En el caso de realizar técnicas de hypertuning de hiper parámetros.
- Para problemas de clasificación y también de regresión.
- Datasets grandes.
- Para evitar el overfitting mediante la aplicación de métodos de ensamble.

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

Entrenamiento de modelo de clasificación de carros con random forest

Evaluación de resultados del modelo de clasificación con random forest

Proyecto final y cierre



Proyecto final: clasificación de ingresos



**UCI**
Machine Learning Repository
Center for Machine Learning and Intelligent Systems

[About](#) [Citation Policy](#) [Donate a Data Set](#) [Contact](#)

☒ Repository ☐ Web 

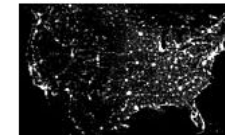
[View ALL Data Sets](#)

Check out the [beta version](#) of the new UCI Machine Learning Repository we are currently testing! [Contact us](#) if you have any issues, questions, or concerns. [Click here to try out the new site.](#) ×

Census Income Data Set

Download: [Data Folder](#), [Data Set Description](#)

Abstract: Predict whether income exceeds \$50K/yr based on census data. Also known as "Adult" dataset.



Data Set Characteristics:	Multivariate	Number of Instances:	48842	Area:	Social
Attribute Characteristics:	Categorical, Integer	Number of Attributes:	14	Date Donated	1996-05-01
Associated Tasks:	Classification	Missing Values?	Yes	Number of Web Hits:	734457

Source:

Donor:

Ronny Kohavi and Barry Becker
Data Mining and Visualization
Silicon Graphics.
e-mail: ronnyk1@sgi.com for questions.

Data Set Information:

Extraction was done by Barry Becker from the 1994 Census database. A set of reasonably clean records was extracted using the following conditions:
((AGE>16) && (AGI>100) && (AFNLWGT>1)&& (HRSWK>0))

Prediction task is to determine whether a person makes over 50K a year.

Curso de
**Decision Trees
y Random Forest
con Python
y Scikit-learn**

Layla Scheli

