



UNIVERSIDAD  
TORCUATO DI TELLA

Licenciatura en Tecnología Digital

**Tecnología Digital VI: Inteligencia Artificial**

# **scikit-learn, árboles y taller**

Clase práctica 3

# Motivación

---

- ¿Cómo podemos implementar concretamente un **árbol de decisión**?
- ¿Cómo determinamos el valor de sus **hiperparámetros**?

# Organización

Etapa	Inicio		Duración
	Sección 1	Sección 2	
Introducción	08:00	09:45	05'
scikit-learn	08:05	09:50	20'
Árboles	08:25	10:10	20'
Taller   Intervalo de práctica	08:45	10:30	30'
Taller   Puesta en común	09:15	11:00	20'
Cierre	09:30	11:15	05'
Fin de la clase	09:35	11:20	-

# scikit-learn | Qué

---

## ¿Qué es scikit-learn?

scikit-learn es la librería más útil y robusta para **machine learning en Python**.

En gran parte escrita en Python, se construye **sobre** las bases de NumPy, SciPy y Matplotlib.

Proporciona implementaciones de **algoritmos** de aprendizaje automático y herramientas para el **preprocesamiento** de datos y la **evaluación** de modelos.

Desarrollada por David Cournapeau como un proyecto de Google Summer of Code en 2007, fue lanzada públicamente como una librería **open-source** en 2010.

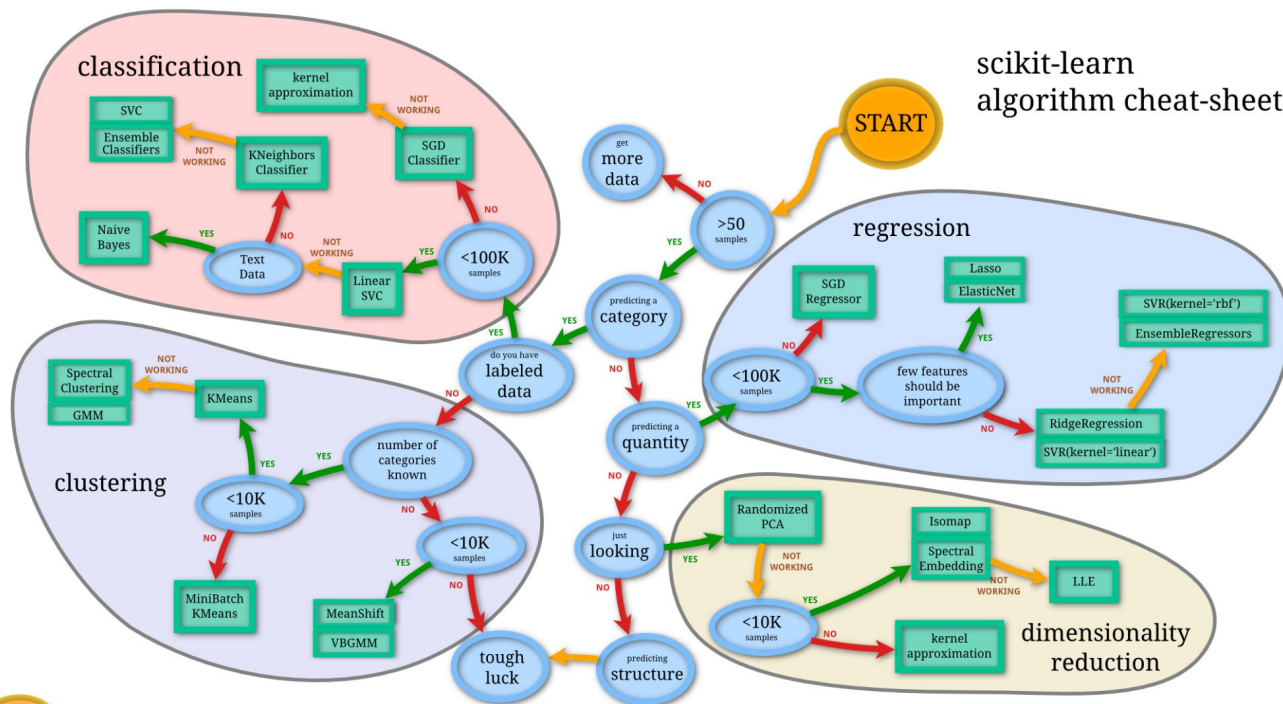
# scikit-learn | Por qué

---

## ¿Por qué seleccionamos scikit-learn?

- **Varios algoritmos** supervisados, no supervisados y semi-supervisados.

# scikit-learn | Por qué



# scikit-learn | Por qué

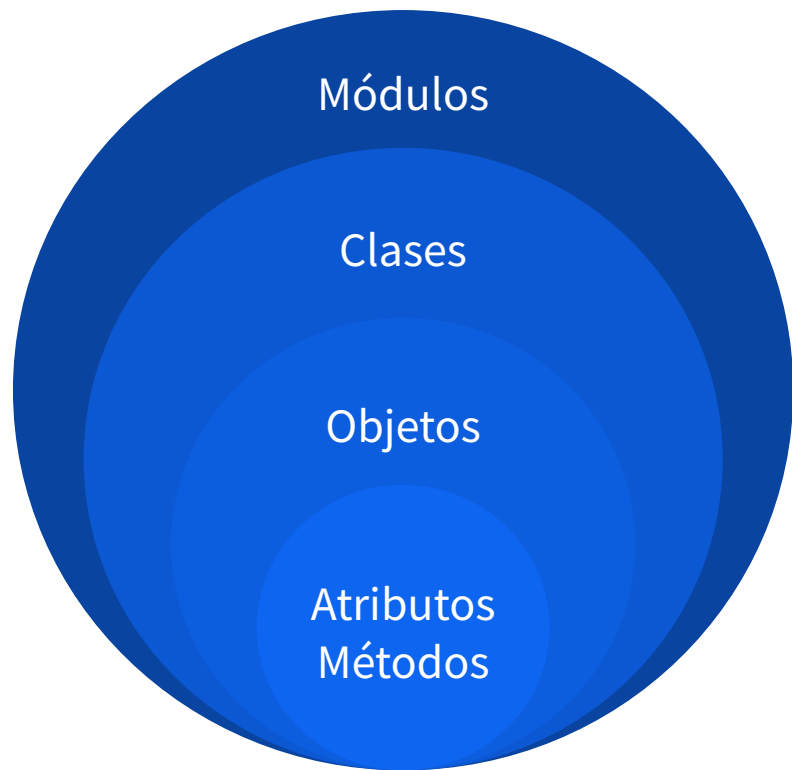
---

## ¿Por qué seleccionamos scikit-learn?

- **Varios algoritmos** supervisados, no supervisados y semi-supervisados.
- **Implementaciones eficientes** de dichos algoritmos de aprendizaje automático.
- **Herramientas adicionales** para el preprocesamiento de datos, la selección y extracción de atributos, el ajuste de hiperparámetros y la evaluación de modelos.
- Buena **integración** con NumPy, SciPy, Pandas y Matplotlib.
- Buen **rendimiento** en tiempo de ejecución.
- Extensa **documentación** oficial en <https://scikit-learn.org>.
- Tiene una **API** clara y consistente.

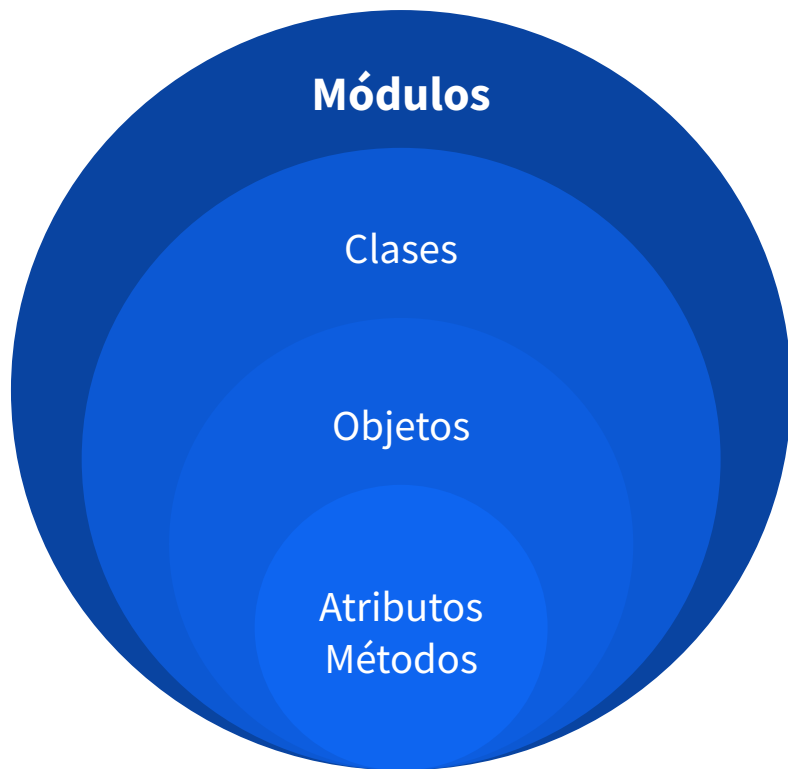
# scikit-learn | Estructura

---





# scikit-learn | Estructura



Las funcionalidades están agrupadas en **módulos**. Cada módulo está dedicado a una **tarea en particular**.

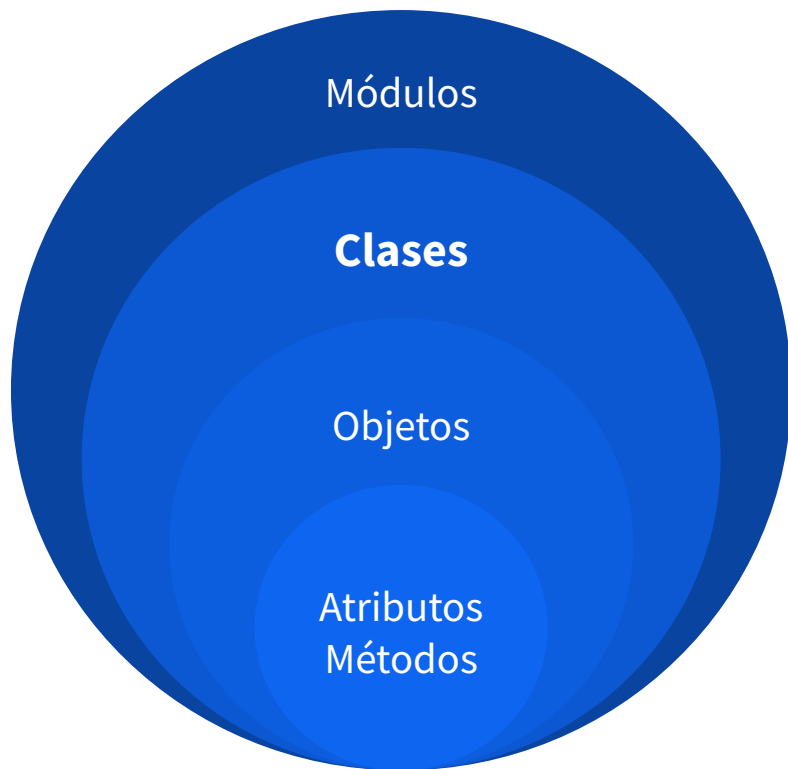
**Ejemplos** de módulos son:

- `sklearn.datasets`
- `sklearn.preprocessing`
- `sklearn.model_selection`
- `sklearn.metrics`
- `sklearn.linear_model`
- `sklearn.tree`

```
from sklearn import linear_model
```

# scikit-learn | Estructura

---



Las **clases** proporcionan una forma estructurada de **organizar** el código. Representan componentes **independientes**.

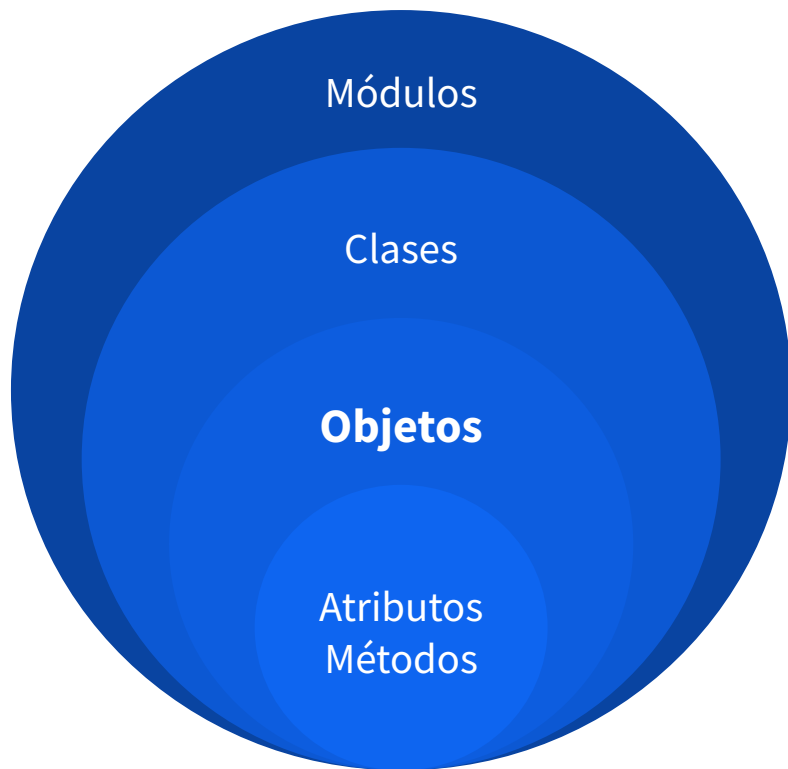
Por **ejemplo**, en el módulo `sklearn.linear_model` hay clases como:

- `LinearRegression`
- `LogisticRegression`
- `Ridge`
- `Lasso`

```
from sklearn.linear_model import  
LogisticRegression
```

# scikit-learn | Estructura

---

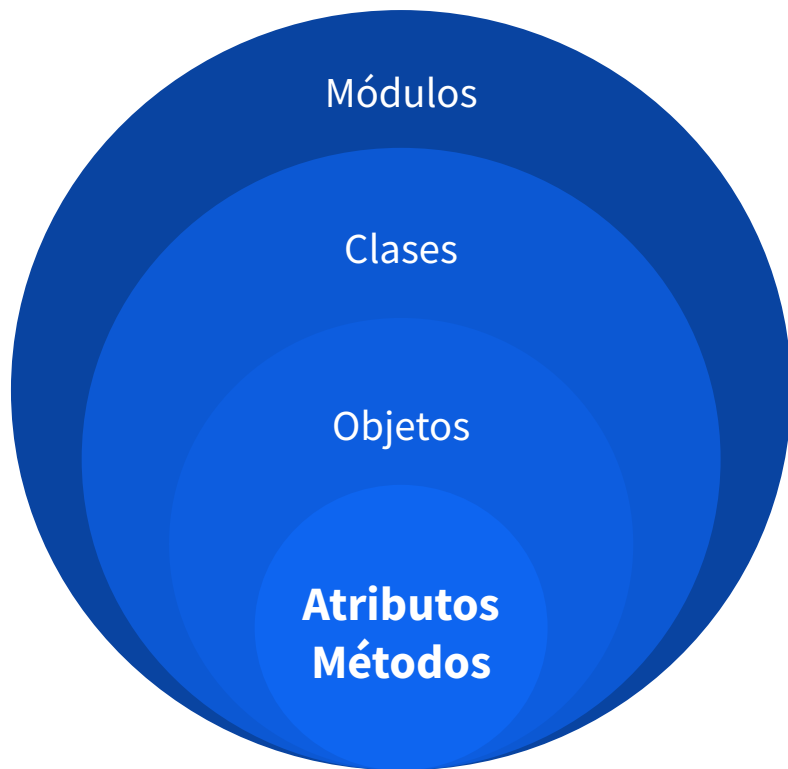


Se llama **objeto** a la **instancia** de una clase.

A modo de **ejemplo**, una instancia de la clase `LogisticRegression` y, pues, un objeto es `reg_log = LogisticRegression()`

# scikit-learn | Estructura

---



Un objeto tiene asociados atributos y métodos.

Los **atributos** representan ciertas **características** del objeto.

Mientras que los **métodos** son funciones que definen **acciones** sobre el objeto.

# scikit-learn | Modo

---

Para conocer la forma en que utilizar esta librería, veamos `td6-p03-c-scikit-learn.ipynb`.

Tengamos en cuenta que también contamos con los archivos `X_train.csv`, `X_test.csv`, `y_train.csv` y `y_test.csv`.

# scikit-learn | Dónde

---

**¿Dónde podemos encontrar recursos útiles sobre scikit-learn?**

Primera opción: [documentación oficial](#).

*Cheat sheets:* [DataCamp](#).

Tutoriales: [DataCamp](#) y [Tutorials Point](#).

# Árboles

---

Para conocer cómo implementar, específicamente, árboles de decisión con la librería scikit-learn, veamos **td6-p03-c-arboles.ipynb**.

# Taller | Consignas

---

Para conocer las consignas del taller, veamos  
`td6-p03-d-taller-consignas.ipynb`.



# Taller | Resoluciones

---

Para conocer las resoluciones del taller, veamos  
`td6-p03-e-taller-resoluciones.ipynb`.

# Cierre

---

Hoy vimos y practicamos, entre otras cuestiones,

- cómo podemos implementar concretamente un **árbol de decisión** y
- cómo determinamos el valor de sus **hiperparámetros**.

¡Recuerden **registrar su grupo** del TP1 en el Campus Virtual! El cierre es el este Viernes.

Pueden darnos *feedback* de la clase [acá](#).