



Máster Smart Energy

INTRODUCCIÓN A MACHINE LEARNING

Postgrado de DIGITAL ENERGY

Curso 2022/2023

Sesión 2 – Introducción a Python



Información Personal

- **Nombre:** Marc Jené Vinuesa



<https://www.linkedin.com/in/marcjene/>

- **Centro:** CITCEA-UPC (ETSEIB)
- **Email:** marc.jene@upc.edu
- Oficina 23.08 Edificio G, ETSEIB – Departamento de Ingeniería Eléctrica.
- **Campos de investigación:** Inteligencia artificial aplicada a los sistemas energéticos, detección de fraude en redes de distribución, mantenimiento predictivo, gestión de la demanda.



Calendario

	Lunes	Martes	Miércoles	Jueves
ABRIL	10	11	12 S1 – Introducción a Machine Learning	13
	17 S2 – Introducción a Python	18	19 S3 – Estadística descriptiva	20
	24 S4 – Modelos de aprendizaje supervisado (I): Clasificación	25	26 S5 – Modelos de aprendizaje supervisado (II): Regresión	27
MAYO	1	2	3 S6 – Aplicación de AI en el sector eléctrico: Odit-e	4
	8 S7 – Modelos de aprendizaje no supervisado	9	10 S8 – Examen final	

Sara Barja : sara.barja@upc.edu
 Marc Jené: marc.jene@upc.edu



Objetivos de la sesión

1. Introducción a la programación en Python con enfoque en data science
2. Familiarizarse con los conceptos básicos de Machine Learning
3. Enfoque generalista y práctico al Machine Learning
4. Proporcionar conocimientos para crear un modelo de Machine Learning, desde la obtención de los datos hasta escoger el modelo y los mecanismos de validación según el tipo de problema
5. Suministrar herramientas y información adicional para profundizar en la temática sin necesidad de empezar de cero



Contenidos de la sesión

- Introducción
- Configuración python, entornos, IDEs...
 - Módulos, packages y librerías
- Variables y tipos de variables
 - Floats, Ints, Strings, Booleans, Complejos
 - Listas, diccionarios y tuplas
 - Operadores
 - Slicing
- Listas y loops
 - Crear y modificar listas
 - Loops “for” (*enumerate*)
 - Loops *while*
 - List comprehensions
- Estructuras condicionales
 - if , elif





Contenidos de la sesión

- Funciones avanzadas
 - Lambda functions
 - Map, filter, reduce
- Diccionarios y tablas de frecuencia
 - Crear un diccionario
 - Actualizar un diccionario
 - Crear tablas de frecuencia
- Funciones
 - Funciones con más de un parámetro de entrada/salida
 - Argumentos por defecto
- Lenguaje orientado a objetos
 - Classes
 - Objetos
 - Métodos



Contenidos de la sesión

- Introducción a NumPy
 - Arrays operation
 - Boolean indexing
- Introducción a Pandas
 - Abrir archivos y crear DataFrames
 - Combinar y transformar datos en pandas
 - Lectura/escritura de archivos de texto (e.g. csv, excel)
- Creación de gráficos
 - Matplotlib
 - Line charts, Bar charts, histograms, Scatter Plots, Box Plots, etc.



Introducción

¿Por qué Python?

- Intuitivo y simple
- Portable a cualquier plataforma
- Open source
- Gran cantidad de bibliotecas disponibles

“...Python’s popularity is driven in no small part by the vast number of specialized libraries available for it, particularly in the domain of artificial intelligence...”

Ranking anual de IEEE Spectrum 2021

Rank	Language	Type	Score
1	Python	🌐 🖥️ ⚙️	100.0
2	Java	🌐 📱 🖥️	95.4
3	C	📱 🖥️ ⚙️	94.7
4	C++	📱 🖥️ ⚙️	92.4
5	JavaScript	🌐	88.1
6	C#	🌐 📱 🖥️ ⚙️	82.4
7	R	🖥️	81.7
8	Go	🌐 🖥️	77.7
9	HTML	🌐	75.4
10	Swift	📱 🖥️	70.4
11	Arduino	⚙️	68.4
12	Matlab	🖥️	68.3
13	PHP	🌐	68.0
14	Dart	🌐 📱	67.7
15	SQL	🖥️	65.0



¿Por qué Python?

The 7 Most In-Demand Programming Languages of 2019

March 15, 2019

Aspiring developers need to know what language to learn, the right education and work on a skill set that will get them the dream job. So what are the top programming languages to learn? We've compiled a list for you that highlights the top languages based off current job postings on GitHub.

Top Machine Learning Languages on GitHub

- 1 Python
- 2 C++
- 3 JavaScript

Here are the top 7 programming languages in January 2019:

- Java
- Python
- JavaScript
- C++
- C#
- PHP
- PERL

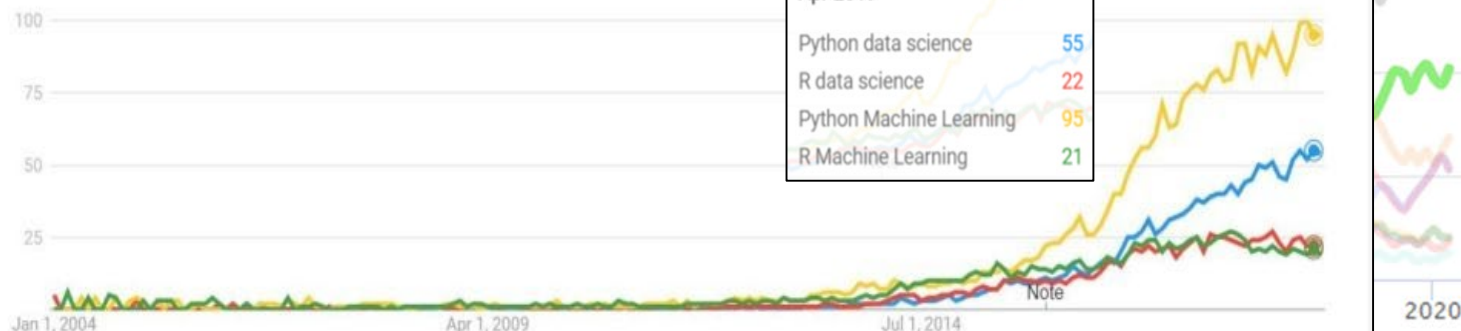


Image Source – [Google Trends](https://www.google.com/trends/)



Python IDEs

- Herramienta para crear, testear y debugar un código en Python

PyCharm



Visual
Studio Code



Sublime Text



Vim



GNU Emacs



IDLE



Atom



Spyder



JuPyter



Eclipse



Eric Python



Wing



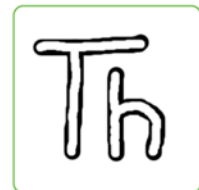
PyScripter



Pyzo



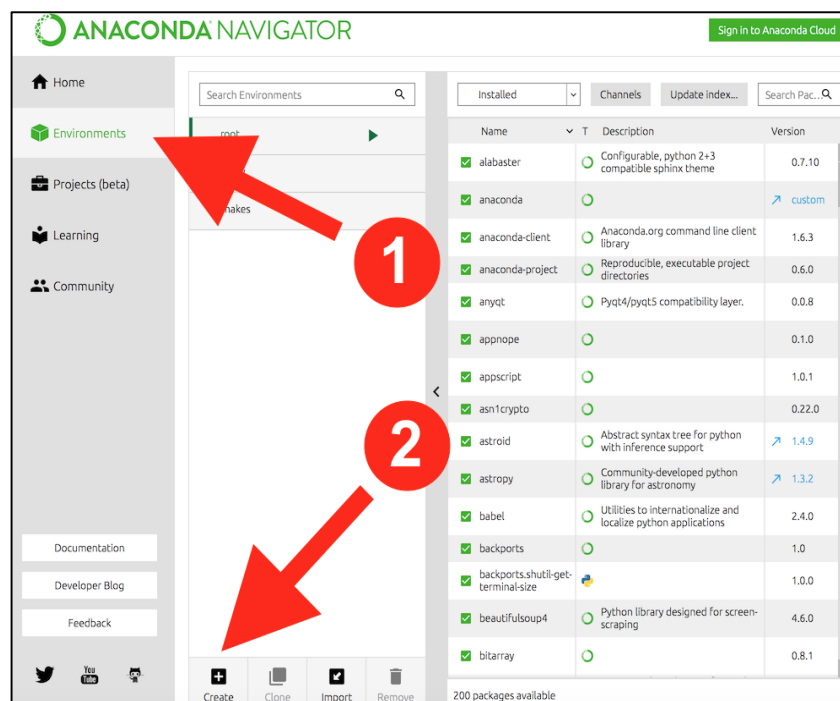
Thonny





Environments

- La creación de enviroments en Anaconda es útil para organizar las librerías instaladas.

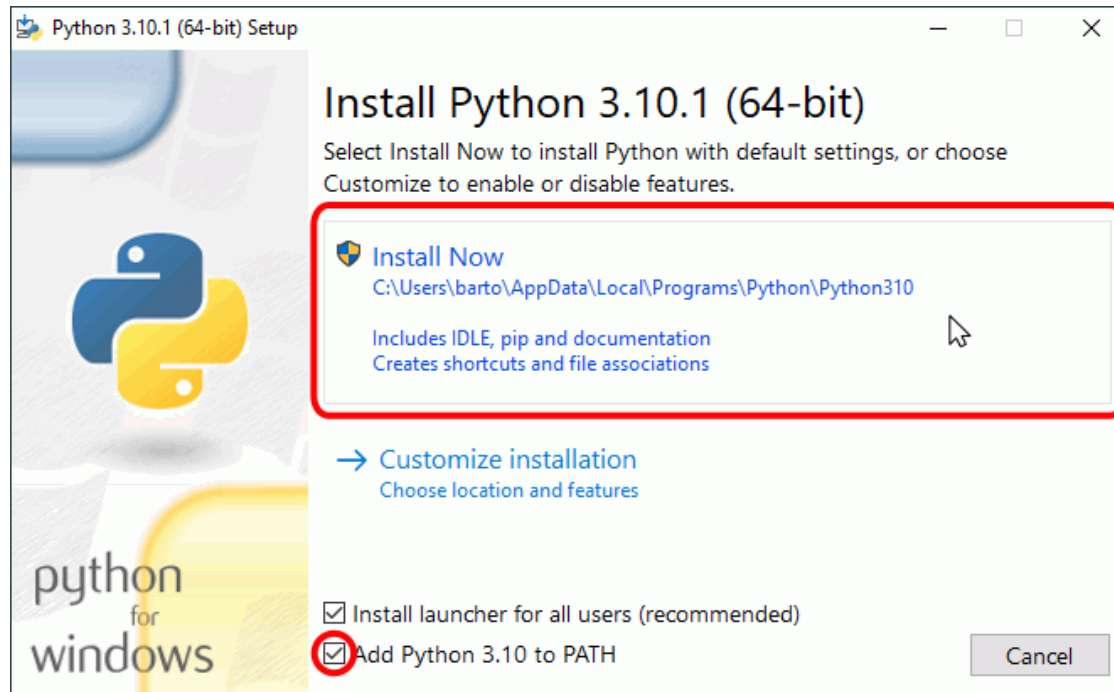




Instalar Python

En Windows:

- www.python.org/downloads





Repositorio de GitHub del Módulo Big Data y Machine Learning


<https://github.com/sbarja/smart-energy-22-23>





Repositorio de GitHub del Módulo Big Data y Machine Learning

MJ How can I clone a github repository?

 To clone a GitHub repository, you can follow these steps:

1. Go to the GitHub website and navigate to the repository you want to clone.
2. Click on the green "Code" button located on the right side of the repository page.
3. You will see a dropdown menu with two options, "HTTPS" and "SSH". Select the option you prefer to use for cloning.
4. Click on the copy icon to copy the URL of the repository.
5. Open your terminal or command prompt and navigate to the directory where you want to clone the repository.
6. Type the command "git clone" followed by the URL you copied in step 4.
For example:

```
bash  
git clone https://github.com/username/repository.git
```

7. Press Enter and Git will create a local copy of the repository in the current directory.

That's it! You have successfully cloned a GitHub repository.



Repositorio de GitHub del Módulo Big Data y Machine Learning

MJ

How can I pull changes from github repository to update my local folder?



To pull changes from a GitHub repository to update your local folder, you can follow these steps:



1. Open your terminal or command prompt and navigate to the local repository that you want to update.
2. Use the ``git pull`` command to fetch and merge the changes from the remote repository. For example, ``git pull origin main``.
3. If there are no conflicts between your local changes and the changes in the remote repository, Git will automatically merge the changes and update your local repository.
4. If there are conflicts between your local changes and the changes in the remote repository, Git will prompt you to resolve the conflicts before merging the changes. You will need to manually resolve the conflicts and then commit the changes using the ``git commit`` command.
5. After the pull is complete, your local repository should be updated with the latest changes from the remote repository.

That's it! You have successfully pulled changes from a GitHub repository to update your local folder.



Para saber más...

- <http://python.org>
- www.kaggle.com/learn/python
- www.learnpython.org
- <https://practice.geeksforgeeks.org/courses/Python-Foundation>

