



Anaconda es una distribución libre y Open Source de los lenguajes de programación Python y R muy usada en computación científica (Data Science, Machine Learning, Ciencia, Ingeniería, analítica predictiva, Big Data, etc). Es multiplataforma y contiene una gran colección de paquetes y librerías para análisis de datos, computación científica e ingeniería. Incluye (entre otras librerías y herramientas) tales como Numpy, Scipy, sk-learn, Pandas, Matplotlib entre otras, adicionalmente incluye Jupyter, Spider, R Studio y más herramientas de desarrollo para la big data, ciencia de datos, machine learning y otras disciplinas de la computación científica.

Anaconda es un ambiente muy empleado dentro de la ciencia de los datos y Big Data que habilita el funcionamiento de distintas aplicaciones. Este ambiente permite mantener el lenguaje de Python, aunque también hace posible visualizar otros lenguajes. Al mismo tiempo dispone de la instalación de diversas bibliotecas de gran utilidad para la gestión y análisis de los datos.

Anaconda dispone de una gran variedad de recursos aplicables en la ciencia de datos y aprendizaje automático. Las cuales se encuentran disponibles sin necesidad de instalar paquetes adicionales.

Contiene un gestor de paquetes conocido como Conda, el cual se encarga de la instalación de las bibliotecas y otras versiones. Las cuales permitirán aislar de manera eficiente los distintos ambientes creados.

Adicionalmente, la interacción de Anaconda con el paquete Pip, permite tener acceso a cualquier biblioteca adicional que no se encuentre dentro de este manejador.



Instalar ANACONDA

<https://www.anaconda.com/products/distribution>

ANAACONDA. Products Pricing Solutions Resources Partners Blog Company Contact Sales

Anaconda Distribution

Everything you need to get started in data science on your workstation.

- ✓ Free distribution install
- ✓ Thousands of the most fundamental DS, AI, and ML packages
- ✓ Manage packages and environments from desktop application
- ✓ Deploy across hardware and software platforms

Code in the cloud Download

Get Additional Installers

Hey there! Have any questions? Let's get you connected to an expert!

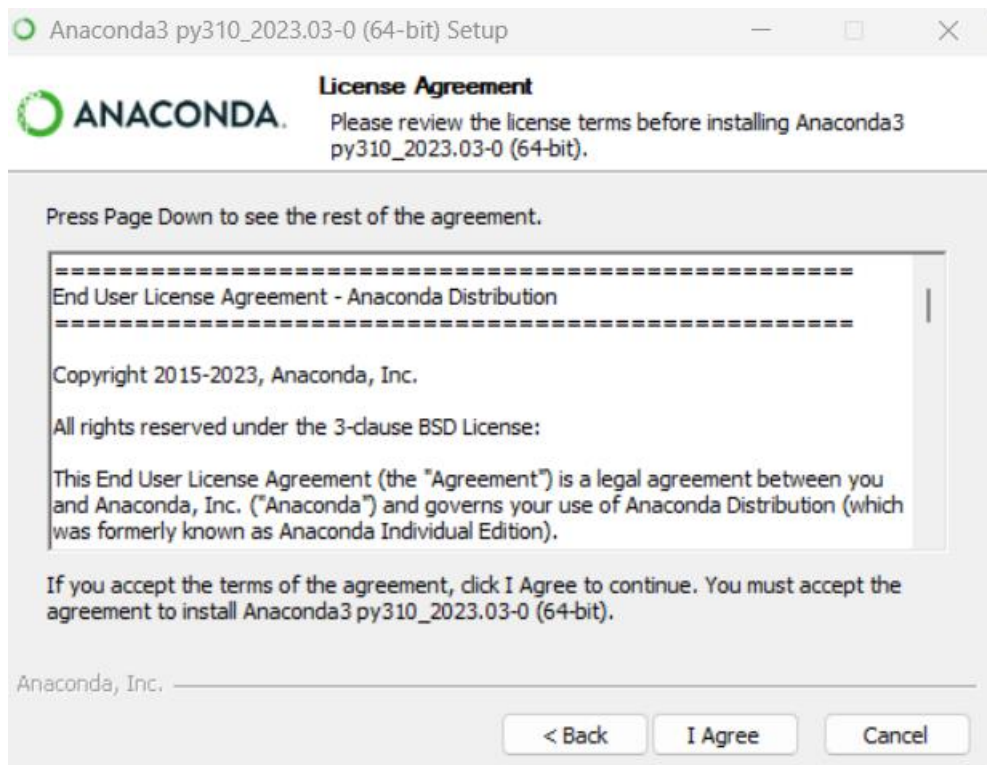
Este equipo > Data (E:) > INSTALLERS > ANACONDA				
Nombre		Fecha de modificación	Tipo	Tamaño
Anaconda3-2023.03-Windows-x86_64.exe		5/4/2023 13:43	Aplicación	804.860 KB

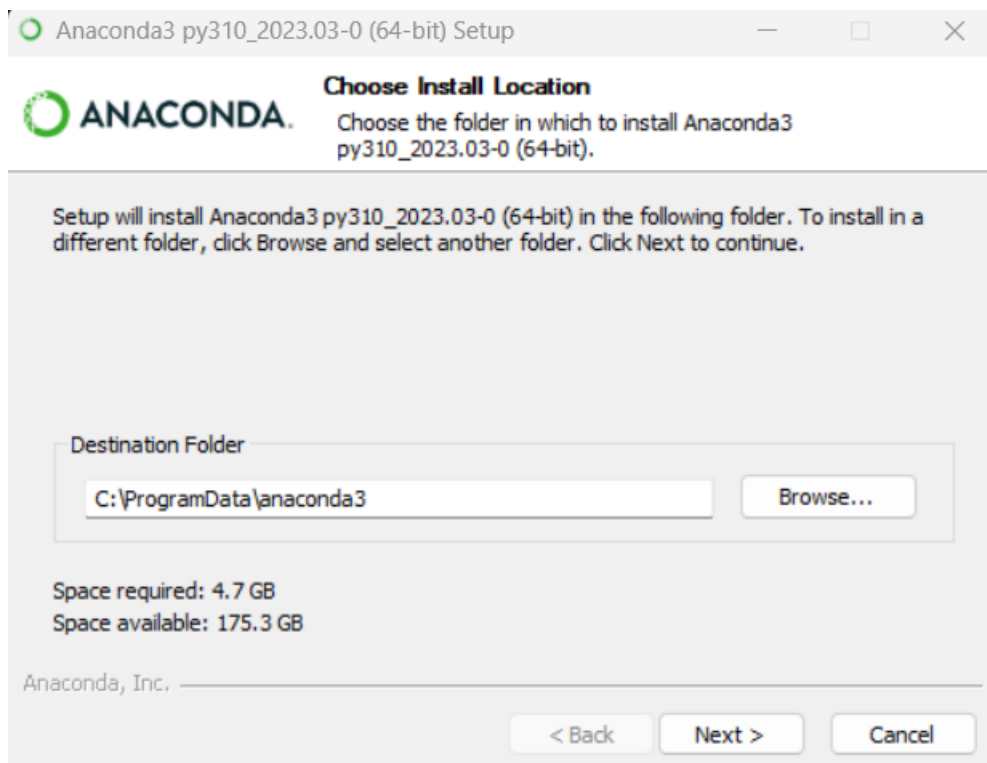
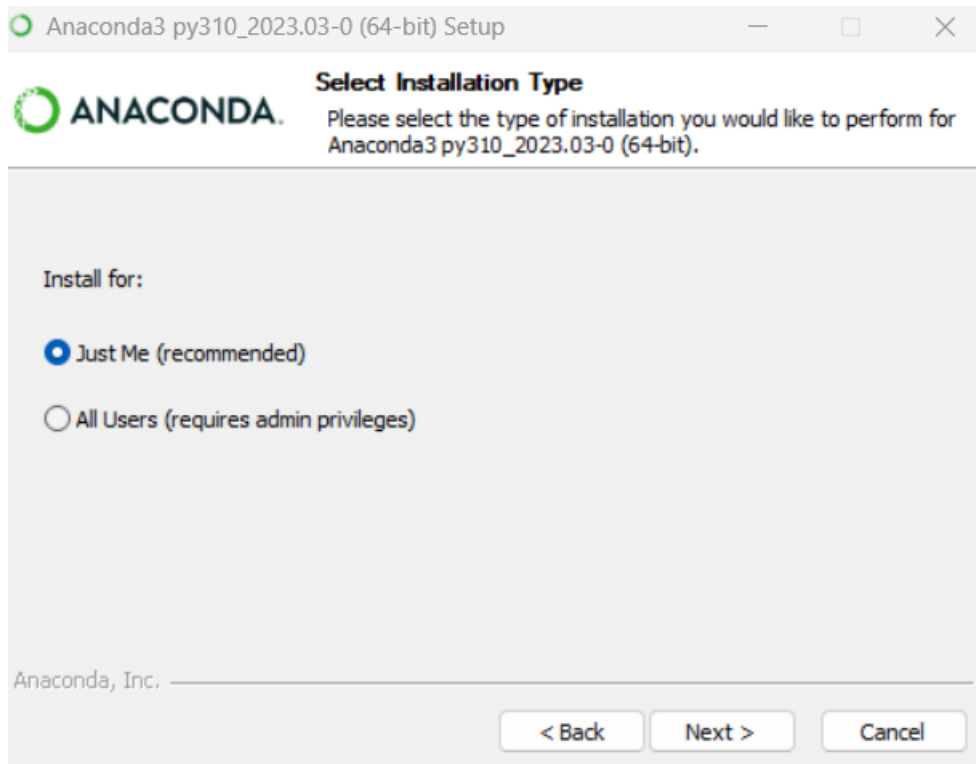
Antes:

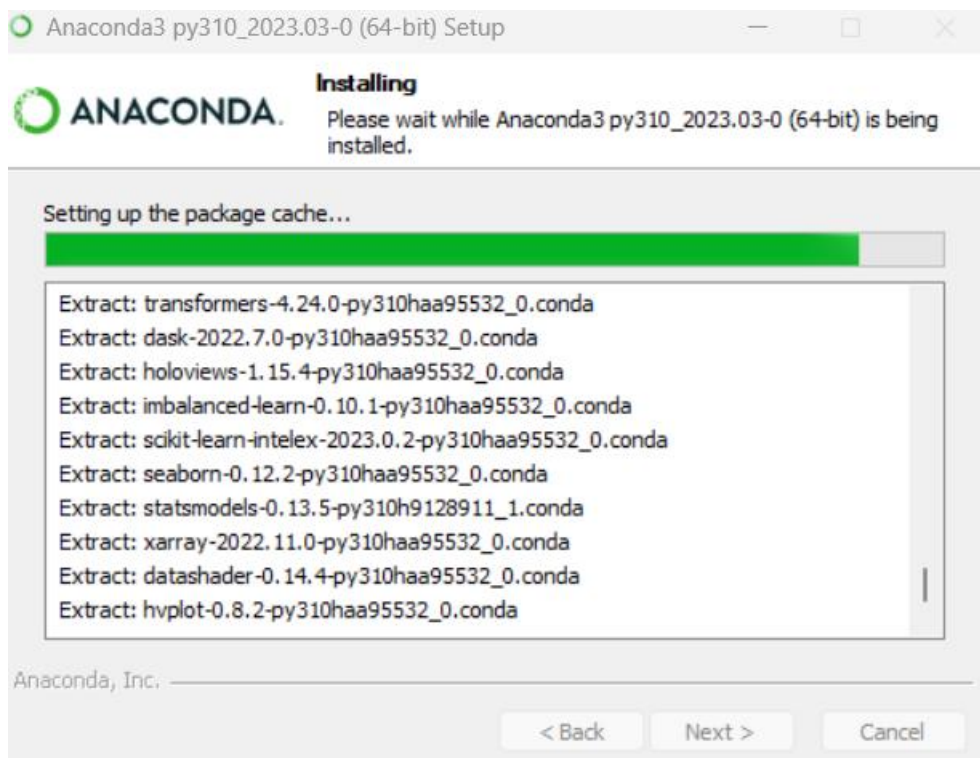
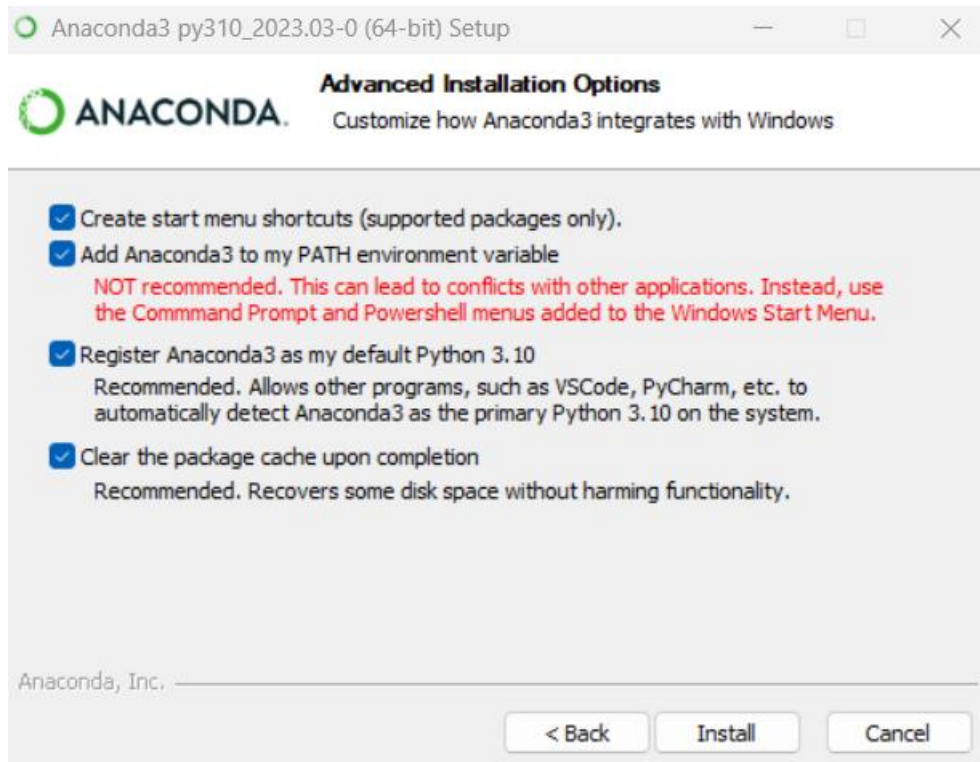
```
C:\Users\ACER>python -version
No se encontró Python; ejecuta sin argumentos para instalar desde Microsoft Store o deshabilita este acceso directo en Configuración > Administrar alias de ejecución de la aplicación.
```

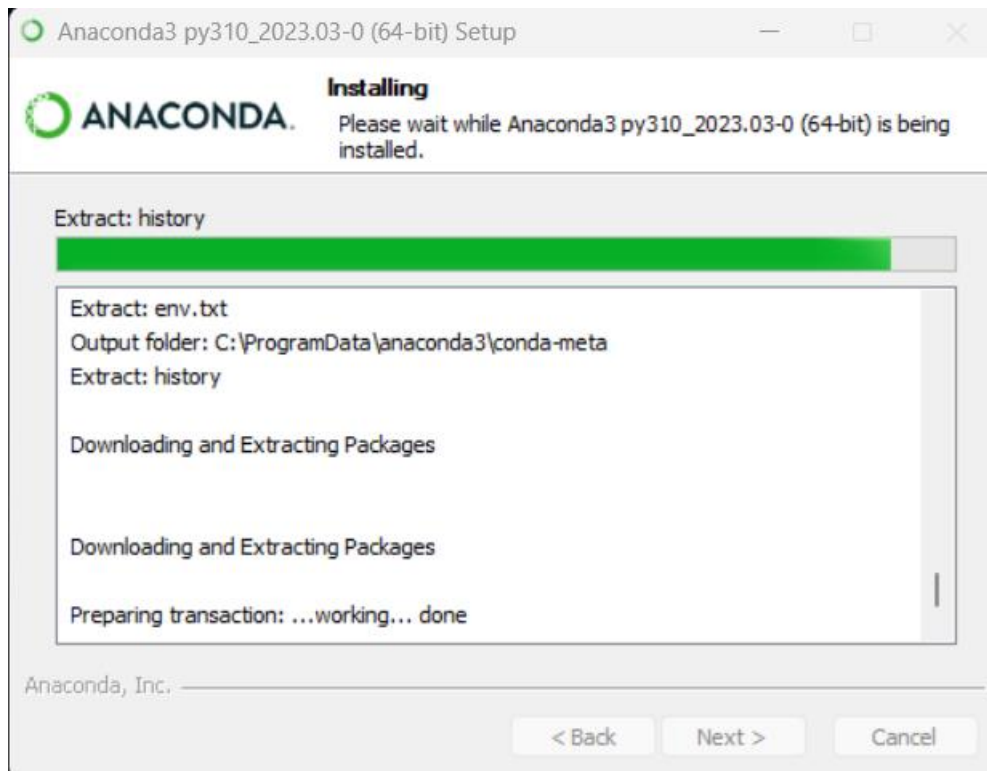


Instalación:

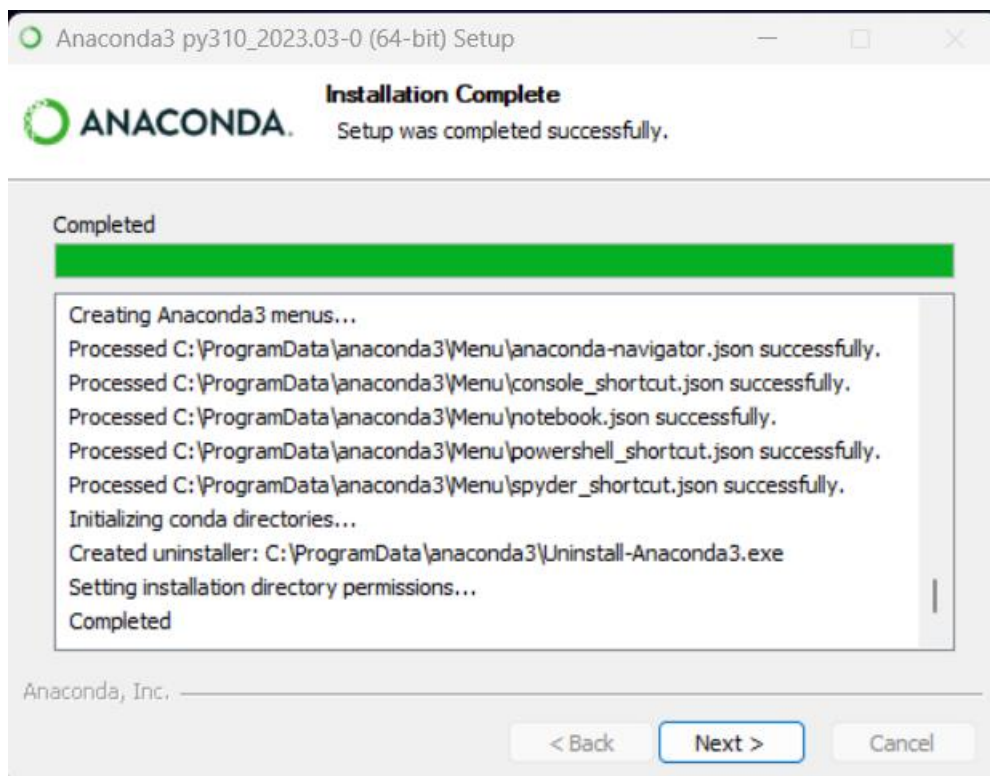


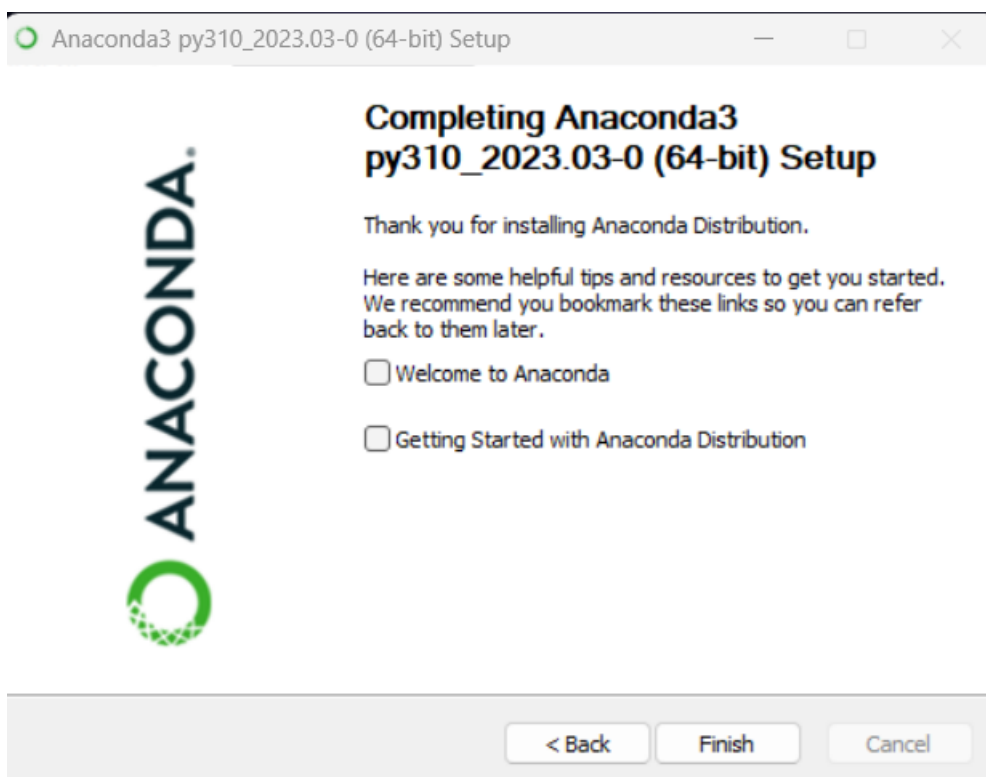
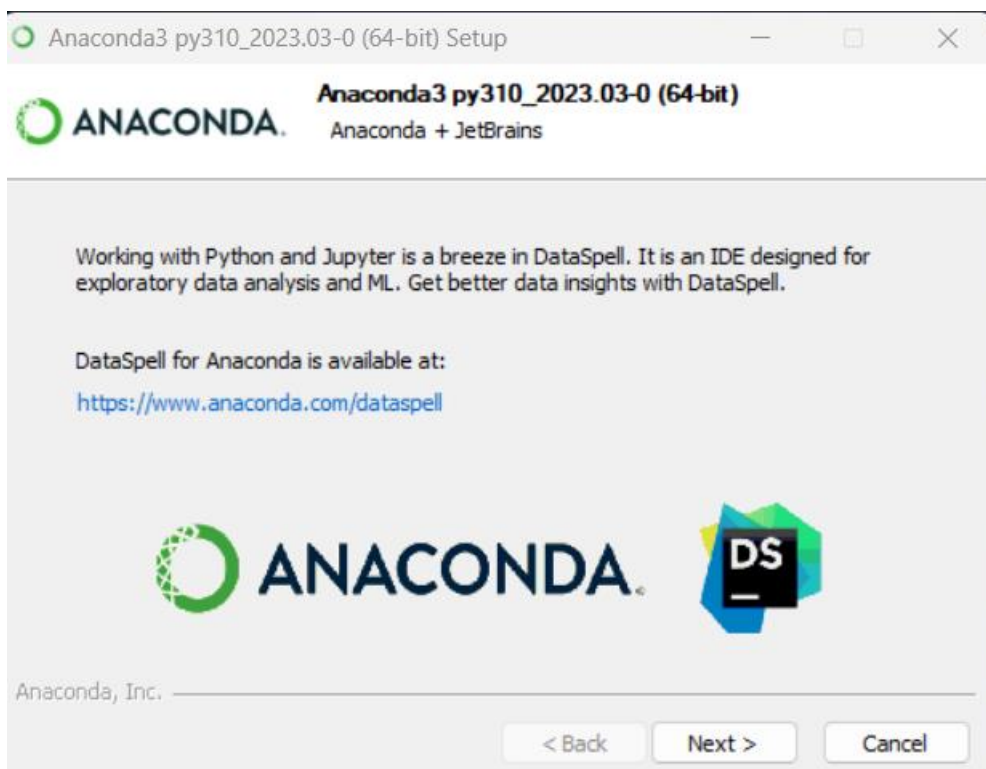






Tiempo estimado: 13:49 a 13:53= 4min







Después:

```
PS C:\Users\ACER> python --version
Python 3.10.9
PS C:\Users\ACER>
```

```
Windows PowerShell
PS C:\Users\ACER> conda --version
conda 23.1.0
PS C:\Users\ACER>
```

Notas: PIP

Pip es un gestor de paquetes, que es útil para instalar librerías adicionales en Python.

Verificar:

#pip --version

#pip3 --version

```
Windows PowerShell
PS C:\Users\ACER> pip --version
pip 22.3.1 from C:\Users\ACER\anaconda3\lib\site-packages\pip (python 3.10)
PS C:\Users\ACER> pip3 --version
pip 22.3.1 from C:\Users\ACER\anaconda3\lib\site-packages\pip (python 3.10)
PS C:\Users\ACER>
```

Ver los paquetes instalados con pip:

#pip list

#pip3 list

```
Windows PowerShell
PS C:\Users\ACER> pip list
Package Version
-----
alabaster 0.7.12
anaconda-client 1.11.1
anaconda-navigator 2.4.0
anaconda-project 0.11.1
anyio 3.5.0
appdirs 1.4.4
argon2-cffi 21.3.0
argon2-cffi-bindings 21.2.0
arrow 1.2.3
astroid 2.14.2
astropy 5.1
asttokens 2.0.5
atomicwrites 1.4.0
attrs 22.1.0
Automat 20.2.0
autopep8 1.6.0
Babel 2.11.0
backcall 0.2.0
backports.functools-lru-cache 1.6.4
backports.tempfile 1.0
backports.weakref 1.0.post1
bcrypt 3.2.0
beautifulsoup4 4.11.1
binaryornot 0.4.4
black 22.6.0
bleach 4.1.0
bokeh 2.4.3
```




Como acceder aquí:

- ✓ Mac => Terminal
- ✓ Linux => Terminal
- ✓ Windows => Anaconda Prompt (instalado junto con Anaconda)

Para instalar un paquete

```
pip install nombre_del_paquete
```

Ej:

```
pip install tensorflow
```

Para instalar un paquete de una versión específica

```
pip install nombre_del_paquete==versión
```

Ej:

```
pip install tensorflow==1.15
```

```
pip install tensorflow==2.12.*
```

Para actualizar un paquete

```
pip install --upgrade nombre_del_paquete
```

Ej:

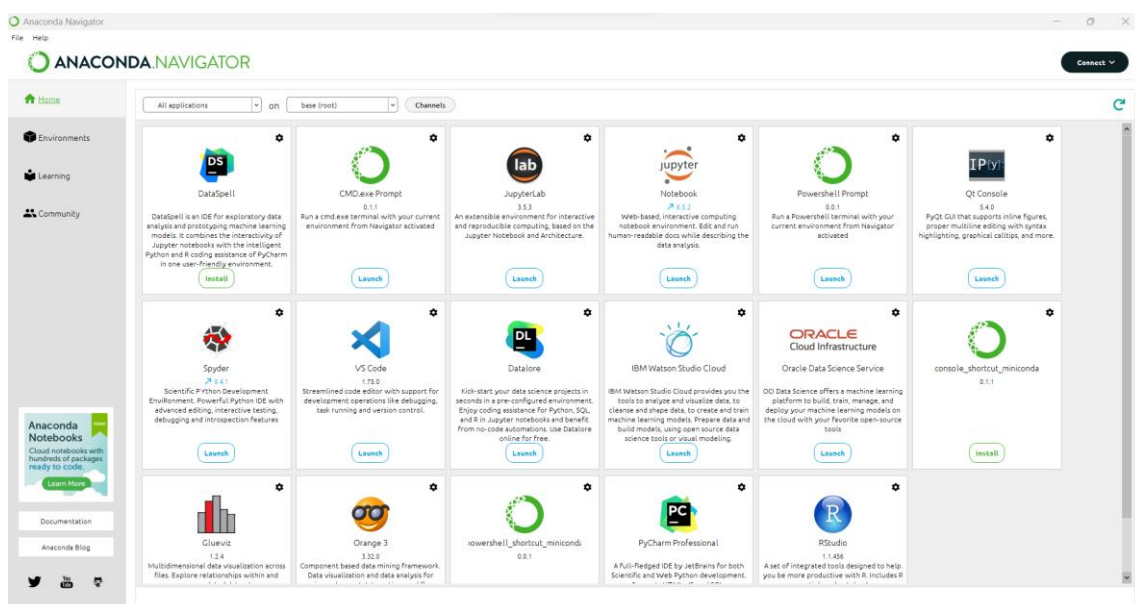
```
pip install --upgrade tensorflow
```

Para desinstalar un paquete

```
pip uninstall nombre_del_paquete
```

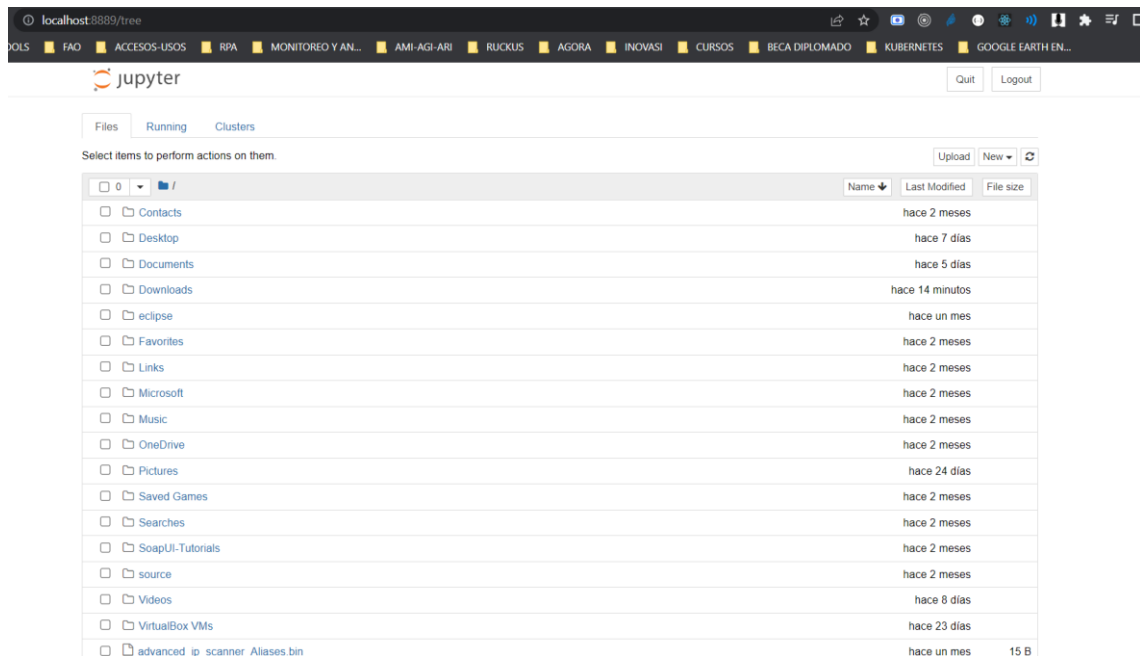
Abrir Anaconda

Al abrir (6 saltos de ventana de consola, tiempo total 2 min aproximadamente).





Abrir Jupyter Notebook



ENTORNOS DE TRABAJO

*Por regla general anaconda tiende a dividir los paquetes en lo que llamamos entorno.

*Un entorno no es más que un conjunto de paquetes de Python aglutinados para formar un entorno de trabajo.

Ej: un entorno de trabajo para big data, un entorno de trabajo para machine learning, un entorno de trabajo para un proyecto específico, se pueden crear diferentes entornos.

Es muy útil porque hay veces que se necesita trabajar con una versión de un paquete para un entorno y otra versión del mismo paquete para otro entorno.



Ver los entornos:

The screenshot shows the Anaconda Navigator interface. On the left, there's a sidebar with 'Home', 'Environments', 'Learning', and 'Community'. The 'Environments' tab is active, showing a list of installed environments. The 'base (root)' environment is selected. The main panel displays a table of installed packages for the 'base (root)' environment.

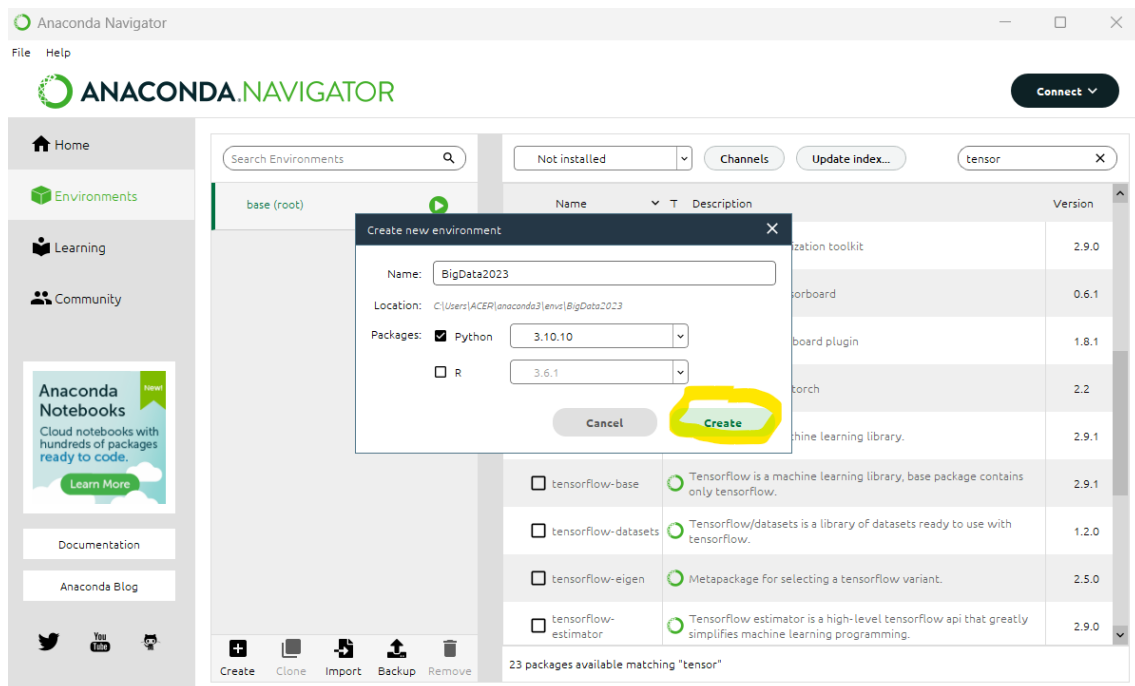
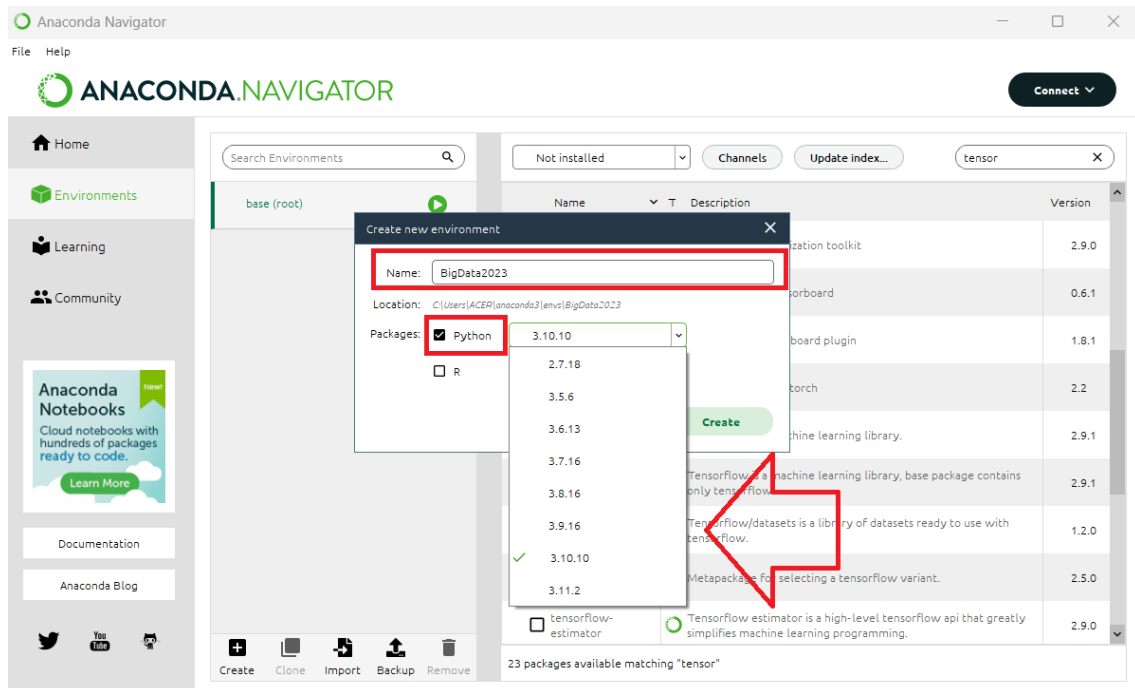
Name	Description	Version
alabaster	Configurable, python 2+3 compatible sphinx theme.	0.7.12
anaconda-client	Anaconda.org command line client library	1.11.1
anaconda-project	Tool for encapsulating, running, and reproducing data science projects	0.11.1
anyio	High level compatibility layer for multiple asynchronous event loop implementations on python	3.5.0
appdirs	A small python module for determining appropriate platform-specific dirs.	1.4.4
argon2-cffi	The secure argon2 password hashing algorithm.	21.3.0
argon2-cffi-bindings	Low-level python cffi bindings for argon2	21.2.0
arrow	Better dates & times for python	1.2.3
astroid	A abstract syntax tree for python with inference support.	2.14.2
astrophy	Community-developed python library for astronomy	5.1
asttokens	The asttokens module annotates python abstract syntax trees (ast) with the positions of tokens and text in the source code that generated them.	2.0.5
atomicwrites	Atomic file writes	1.4.0
attrs	Attrs is the python package that will bring back the joy of writing classes by relieving you from the drudgery of implementing object protocols (aka dunder methods).	22.1.0
automat	Self-service finite-state machines for the programmer on the go	20.2.0
autopep8	A tool that automatically formats python code to conform to the pep 8 style guide	1.6.0
babel	Utilities to internationalize and localize python applications.	2.11.0

Como crear un entorno:

The screenshot shows the Anaconda Navigator interface with the 'Environments' tab active. The 'base (root)' environment is selected. The main panel displays a table of packages that are not installed. The 'Create' button in the bottom left corner is highlighted with a yellow circle.

Name	Description	Version
tensorboard	Tensorflow's visualization toolkit	2.9.0
tensorboard-data-server	Data server for tensorboard	0.6.1
tensorboard-plugin-wit	What-if tool tensorboard plugin	1.8.1
tensorboardx	Tensorboard for pytorch	2.2
tensorflow	Tensorflow is a machine learning library.	2.9.1
tensorflow-base	Tensorflow is a machine learning library, base package contains only tensorflow.	2.9.1
tensorflow-datasets	Tensorflow/datasets is a library of datasets ready to use with tensorflow.	1.2.0
tensorflow-eigen	Metapackage for selecting a tensorflow variant.	2.5.0
tensorflow-estimator	Tensorflow estimator is a high-level tensorflow api that greatly simplifies machine learning programming.	2.9.0

23 packages available matching "tensor"





Anaconda Navigator

File Help

ANACONDA.NAVIGATOR

Connect

Home

Environments

Learning

Community

Anaconda Notebooks

Cloud notebooks with hundreds of packages ready to code.

Learn More

Documentation

Anaconda Blog

Search Environments

base (root)

BigData2023

Created

pytorch

transformers

Pytorch is an optimized tensor library for deep learning using gpus and cpus.

State-of-the-art machine learning for jax, pytorch and tensorflow

1.12.1

4.24.0

Version

Loading packages of C:\Users\ACER\anaci

Create Clone Import Backup Remove

Cancel

Anaconda Navigator

File Help

ANACONDA.NAVIGATOR

Connect

Home

Environments

Learning

Community

Anaconda Notebooks

Cloud notebooks with hundreds of packages ready to code.

Learn More

Documentation

Anaconda Blog

Search Environments

base (root)

BigData2023

Created

Installed

Channels

Update index...

tensor

Name

Description

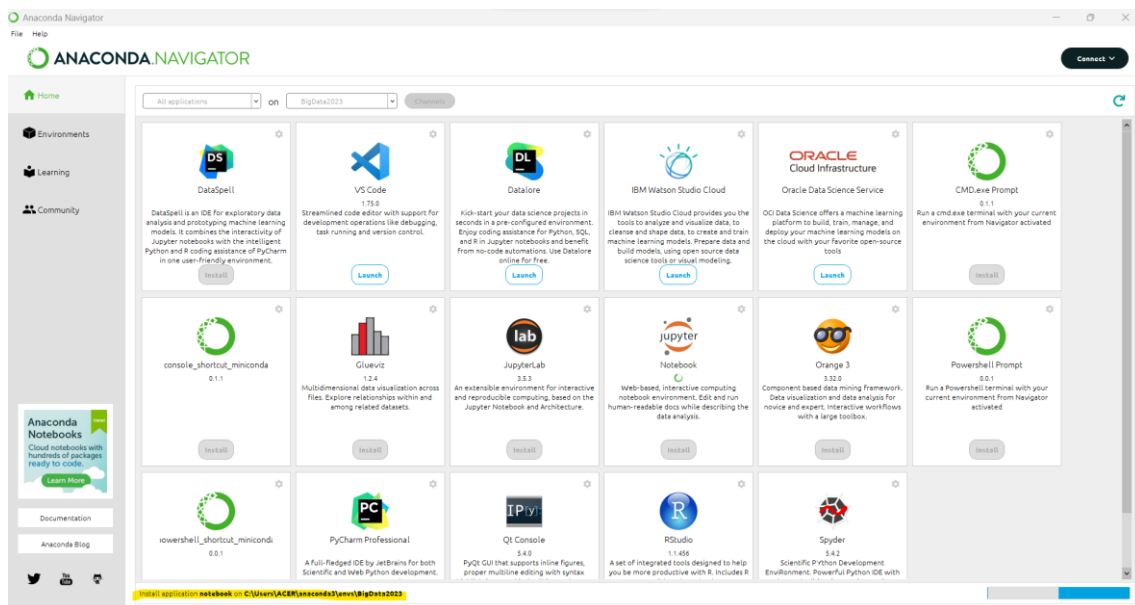
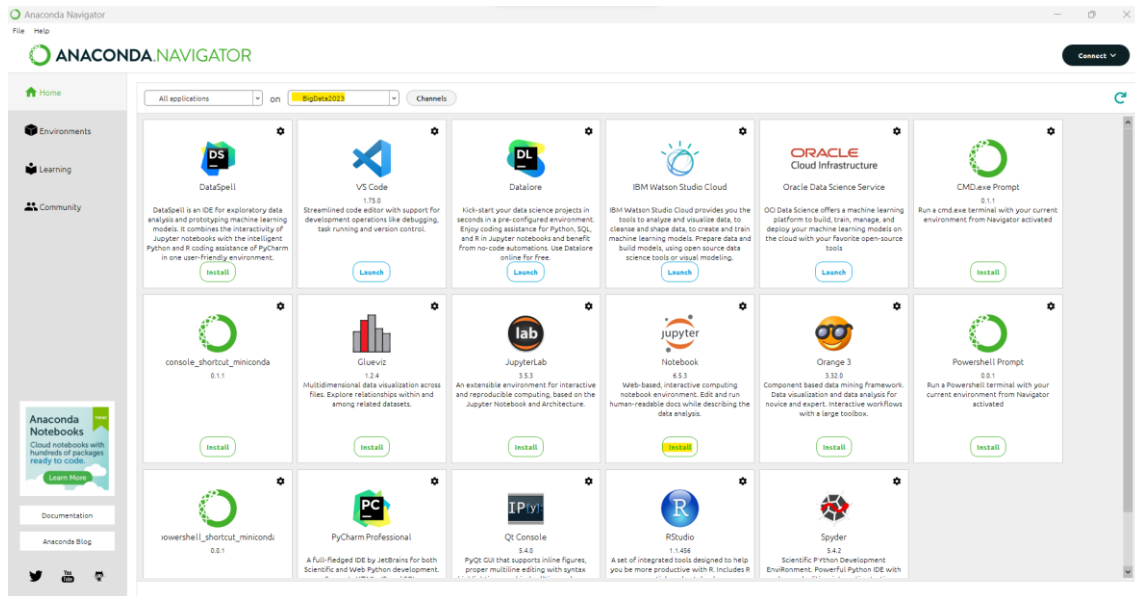
Version

0 packages available matching "tensor"

Create Clone Import Backup Remove

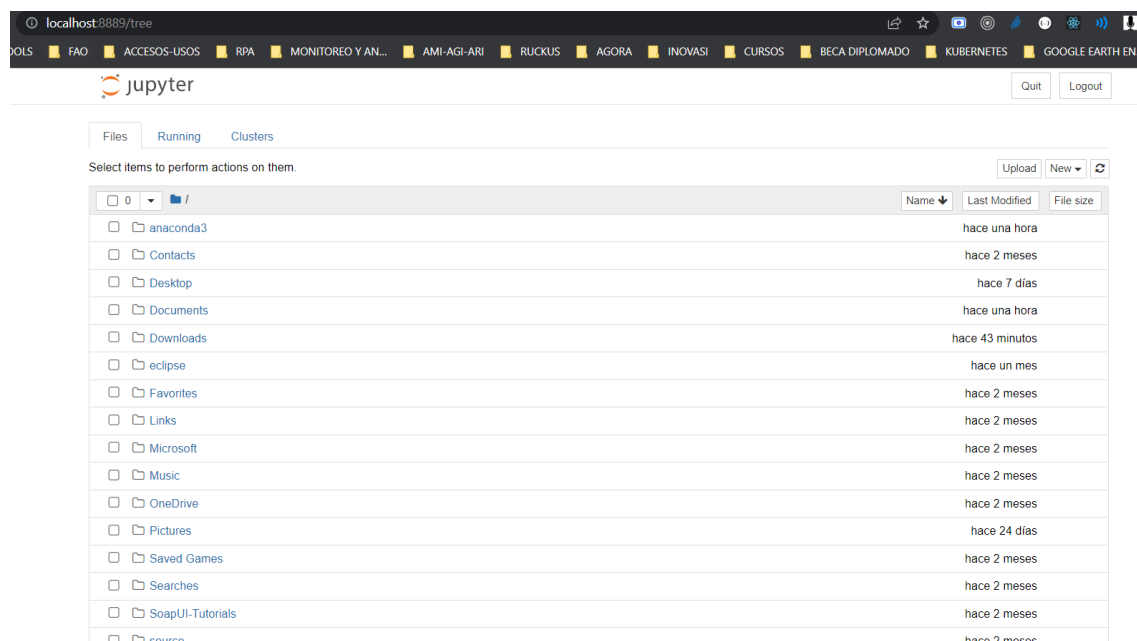
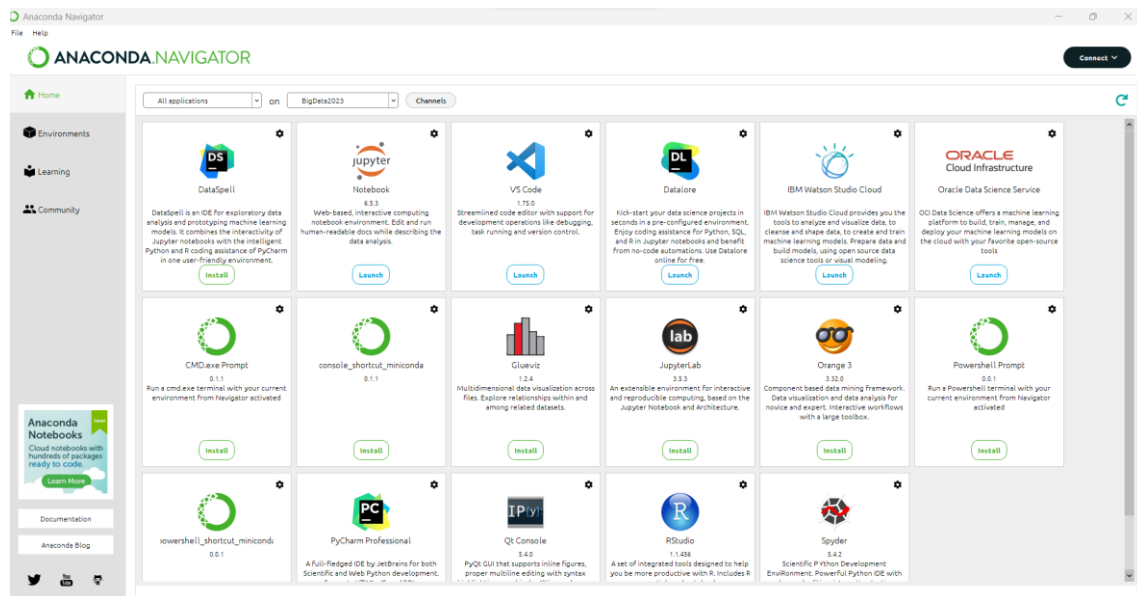


Universidad Nacional de Asunción
Facultad Politécnica
Ingeniería en Informática
Electiva – Big Data – Apuntes de Clases Prácticas

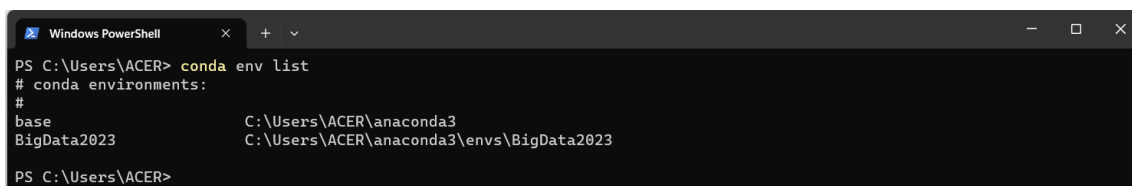




Universidad Nacional de Asunción
Facultad Politécnica
Ingeniería en Informática
Electiva – Big Data – Apuntes de Clases Prácticas



Ver listado de entornos existentes:
`#conda env list`





```
Administrador: Símbolo del sistema
Microsoft Windows [Versión 10.0.22621.1413]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Windows\System32>conda env list
# conda environments:
#
base                  C:\Users\ACER\anaconda3
BigData2023           C:\Users\ACER\anaconda3\envs\BigData2023

C:\Windows\System32>conda activate BigData2023

(BigData2023) C:\Windows\System32>conda env list
# conda environments:
#
base                  C:\Users\ACER\anaconda3
BigData2023           * C:\Users\ACER\anaconda3\envs\BigData2023

(BigData2023) C:\Windows\System32>
```

Ver los paquetes y la versión instalados en un entorno

#conda list -n nombre_entorno

Ej:

conda list -n BigData2023

```
Windows PowerShell
PS C:\Users\ACER> conda list -n BigData2023
# packages in environment at C:\Users\ACER\anaconda3\envs\BigData2023:
#
# Name                        Version      Build      Channel
anyio                         3.5.0        py310haa95532_0
argon2-cffi                   21.3.0       pyhd3eb1b0_0
argon2-cffi-bindings         21.2.0       py310h2bbff1b_0
asttokens                     2.0.5        pyhd3eb1b0_0
attrs                         22.1.0       py310haa95532_0
backcall                      0.2.0        pyhd3eb1b0_0
beautifulsoup4                4.12.0       py310haa95532_0
bleach                        4.1.0        pyhd3eb1b0_0
bzip2                         1.0.8        he774522_0
ca-certificates               2023.01.10   haa95532_0
certifi                       2022.12.7    py310haa95532_0
cffi                           1.15.1       py310h2bbff1b_3
colorama                      0.4.6        py310haa95532_0
comm                           0.1.2        py310haa95532_0
debugpy                       1.5.1        py310hd77b12b_0
decorator                     5.1.1        pyhd3eb1b0_0
defusedxml                    0.7.1        pyhd3eb1b0_0
entrypoints                    0.4          py310haa95532_0
executing                     0.8.3        pyhd3eb1b0_0
flit-core                     3.8.0        py310haa95532_0
idna                          3.4          py310haa95532_0
ipykernel                     6.19.2       py310h9909e9c_0
ipython                       8.10.0       py310haa95532_0
ipython_genutils              0.2.0        pyhd3eb1b0_1
jedi                          0.18.1       py310haa95532_1
jinja2                        3.1.2        py310haa95532_0
```

Activar entorno

#conda activate nombre_entorno

Ej:

conda activate BigData2023



```
Administrador: Símbolo del sistema
Microsoft Windows [Versión 10.0.22621.1413]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Windows\System32>conda env list
# conda environments:
#
base                  C:\Users\ACER\anaconda3
BigData2023           C:\Users\ACER\anaconda3\envs\BigData2023

C:\Windows\System32>conda activate BigData2023

(BigData2023) C:\Windows\System32>conda env list
# conda environments:
#
base                  C:\Users\ACER\anaconda3
BigData2023           * C:\Users\ACER\anaconda3\envs\BigData2023

(BigData2023) C:\Windows\System32>
```

Documentación gestión de entornos:

<https://conda.io/projects/conda/en/latest/user-guide/tasks/manage-environments.html>

DIRECTORIO DE TRABAJO JUPYTER NOTEBOOK

Método 1:

```
cd "E:\Cp_Dev\BigData2023"
```

```
jupyter notebook
```

Nota: se podría crear archivo batch para automatizar el inicio en un directorio específico como ejemplo: abrirJupyterNotebook.bat

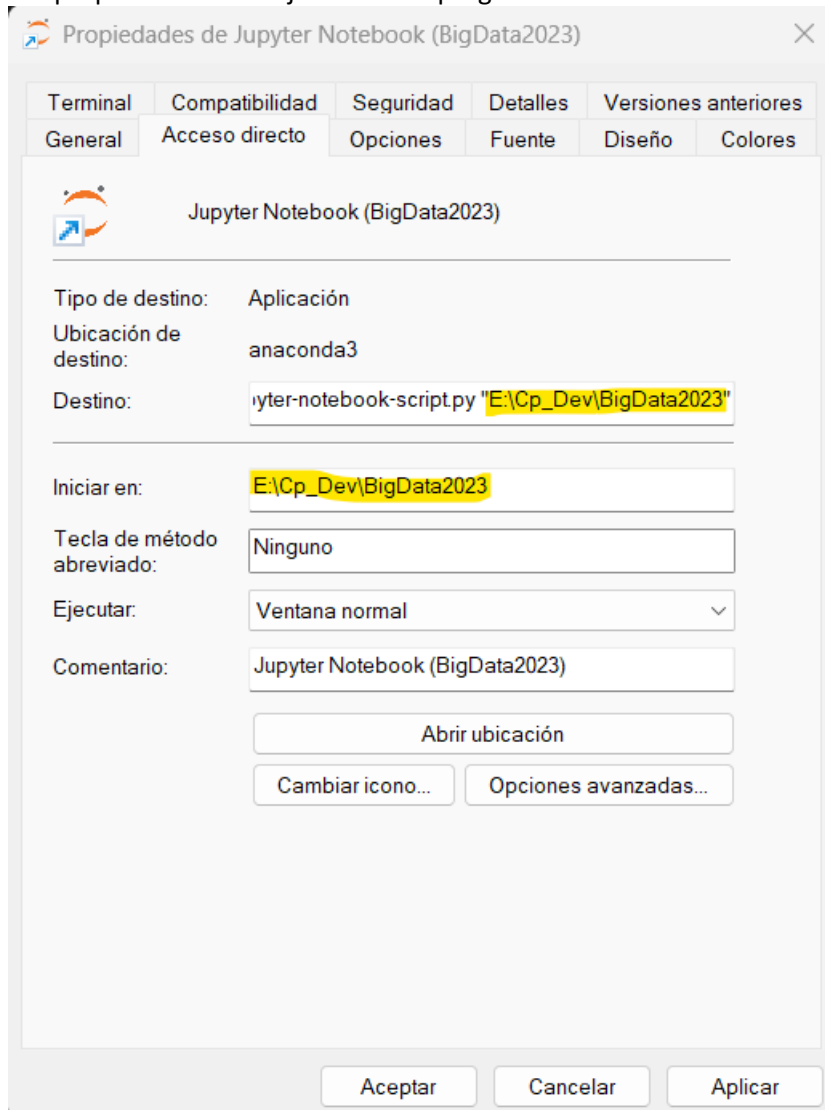
```
abrirJupyterNotebook.bat
1 cd "E:\Cp_Dev\BigData2023"
2 jupyter notebook
```

Método 2:

Cómo cambiar la ruta de trabajo por defecto de Jupyter Notebook en Windows.



Ir a propiedades de la ejecución del programa:



Nota:

- 1- Lo importante del método 2 es que ya está asociado al entorno.
- 2- Se puede ejecutar directo el jupyter sin necesidad de abrir Anaconda

RESUMEN

- ✓ Instalar última versión de Anaconda (ojo con 32 vs 64 bits en Windows, en MAC o Linux; no se debería tener ese problema).
- ✓ (Opcional) Cuando se instale Anaconda que escriba en el Path (solo se habilita en la instalación modo **Just Me**, por seguridad está bloqueada en la opción **All Users**).
- ✓ (Opcional) Instalar paquetes por consola con pip (considerar los cuidados mencionados)



Recomendaciones:

- ✓ Crear un entorno para trabajar dependiendo del propósito, con las versiones adecuadas de los paquetes.
- ✓ Seleccionar ese entorno desde el desplegable de Anaconda
- ✓ Lanzar Jupyter Notebook desde el desplegable de Anaconda en el entorno apropiado
- ✓ Disfrutar y aprender mucho del curso

Lanzar Jupyter Notebook desde un Entorno Específico

Directorio y Entorno

```
cd "E:\Cp_Dev\BigData2023"  
conda env list  
conda activate BigData2023  
jupyter notebook
```

```
Administrador: Símbolo del sistema  
Microsoft Windows [Versión 10.0.22621.1413]  
(c) Microsoft Corporation. Todos los derechos reservados.  
  
C:\Windows\System32>cd "E:\Cp_Dev\BigData2023"  
  
C:\Windows\System32>E:  
  
E:\Cp_Dev\BigData2023>conda env list  
# conda environments:  
#  
base                  C:\Users\ACER\anaconda3  
BigData2023           C:\Users\ACER\anaconda3\envs\BigData2023  
  
E:\Cp_Dev\BigData2023>conda activate BigData2023  
  
(BigData2023) E:\Cp_Dev\BigData2023>
```

Instalar PySpark

Chequear si ya está instalado

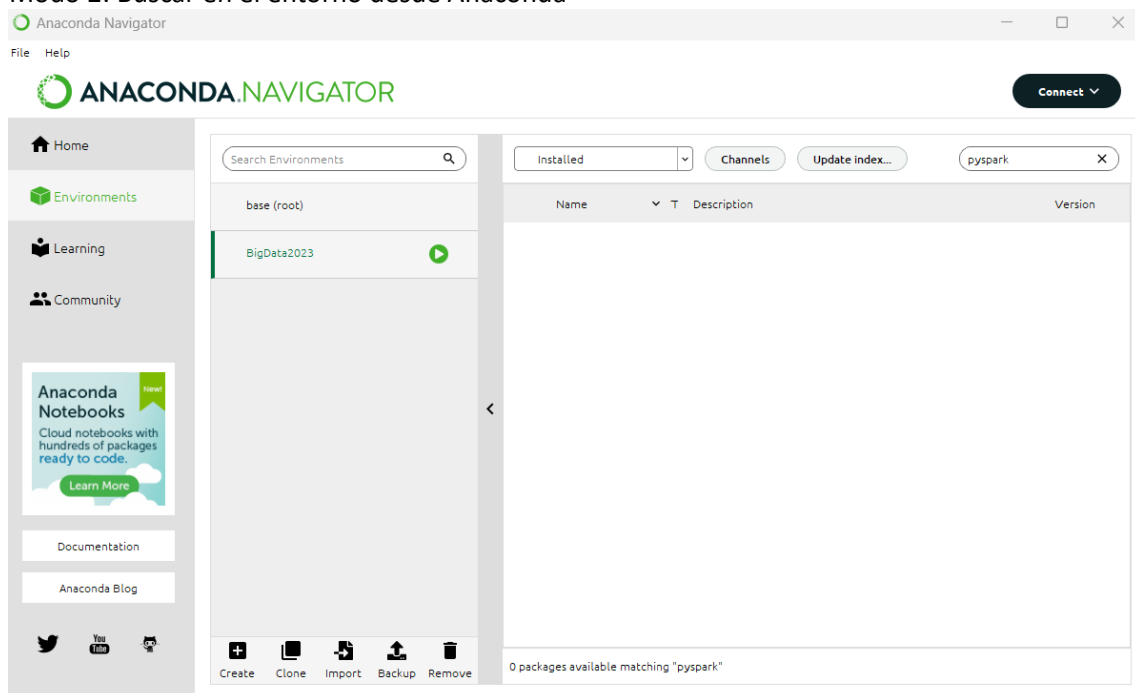
Modo 1: Buscar en el entorno desde terminal

```
conda list -n BigData2023
```



```
Símbolo del sistema
C:\Users\ACER>conda list -n BigData2023
# packages in environment at C:\Users\ACER\anaconda3\envs\BigData2023:
#
# Name                        Version      Build                Channel
anyio                         3.5.0        py310haa95532_0      pyhd3eb1b0_0
argon2-cffi                   21.3.0       pyhd3eb1b0_0         pyhd3eb1b0_0
argon2-cffi-bindings         21.2.0       py310h2bbff1b_0      pyhd3eb1b0_0
asttokens                     2.0.5        pyhd3eb1b0_0         pyhd3eb1b0_0
attrs                         22.1.0       py310haa95532_0      pyhd3eb1b0_0
backcall                      0.2.0        pyhd3eb1b0_0         pyhd3eb1b0_0
beautifulsoup4                4.12.0       py310haa95532_0      pyhd3eb1b0_0
bleach                         4.1.0        pyhd3eb1b0_0         pyhd3eb1b0_0
bzip2                         1.0.8        he774522_0           haa95532_0
ca-certificates               2023.01.10   haa95532_0           py310haa95532_0
certifi                       2022.12.7    py310haa95532_0      py310h2bbff1b_3
cffi                          1.15.1       py310h2bbff1b_3      py310haa95532_0
colorama                      0.4.6        py310haa95532_0      py310haa95532_0
comm                          0.1.2        py310haa95532_0      py310hd77b12b_0
debugpy                       1.5.1        py310hd77b12b_0      pyhd3eb1b0_0
decorator                     5.1.1        pyhd3eb1b0_0         pyhd3eb1b0_0
defusedxml                    0.7.1        pyhd3eb1b0_0         py310haa95532_0
entrypoints                   0.4          py310haa95532_0      pyhd3eb1b0_0
executing                     0.8.3        pyhd3eb1b0_0         py310haa95532_0
flit-core                     3.8.0        py310haa95532_0      py310haa95532_0
idna                          3.4          py310haa95532_0      py310h9909e9c_0
ipykernel                     6.19.2       py310haa95532_0      py310haa95532_0
ipython                       8.10.0       py310haa95532_0      pyhd3eb1b0_1
ipython_genutils              0.2.0        pyhd3eb1b0_1         py310haa95532_1
jedi                          0.18.1       py310haa95532_1      py310haa95532_0
jinja2                        3.1.2        py310haa95532_0
```

Modo 2: Buscar en el entorno desde Anaconda





Anaconda Navigator

File Help

ANACONDA.NAVIGATOR

Connect

Home

Environments

Learning

Community

Anaconda Notebooks

Cloud notebooks with hundreds of packages ready to code.

Learn More

Documentation

Anaconda Blog

Twitter YouTube GitHub

Search Environments

base (root)

BigData2023

Not installed

Channels

Update index...

pyspark

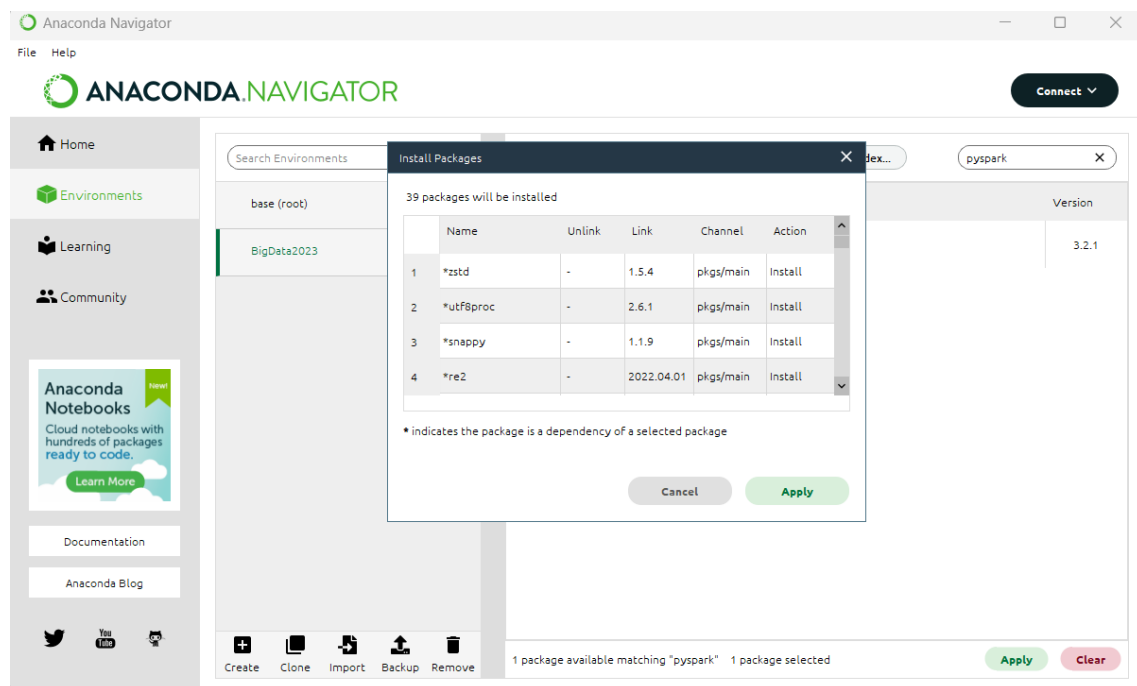
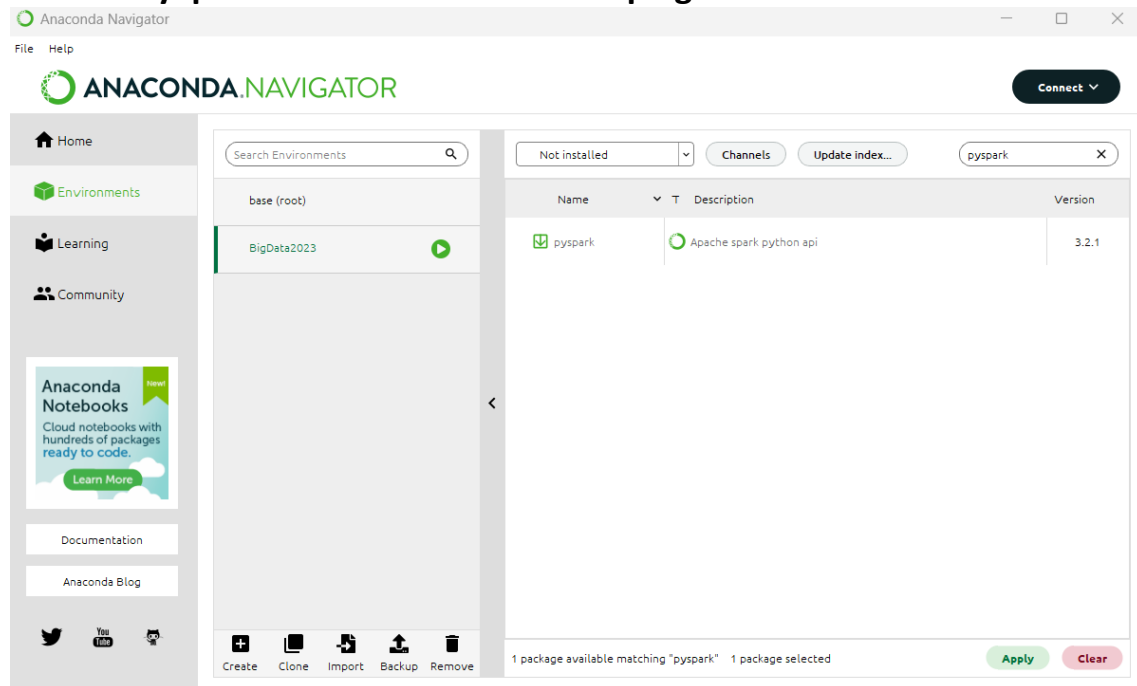
Name	Description	Version
pyspark	Apache spark python api	3.2.1

Create Clone Import Backup Remove

1 package available matching "pyspark"

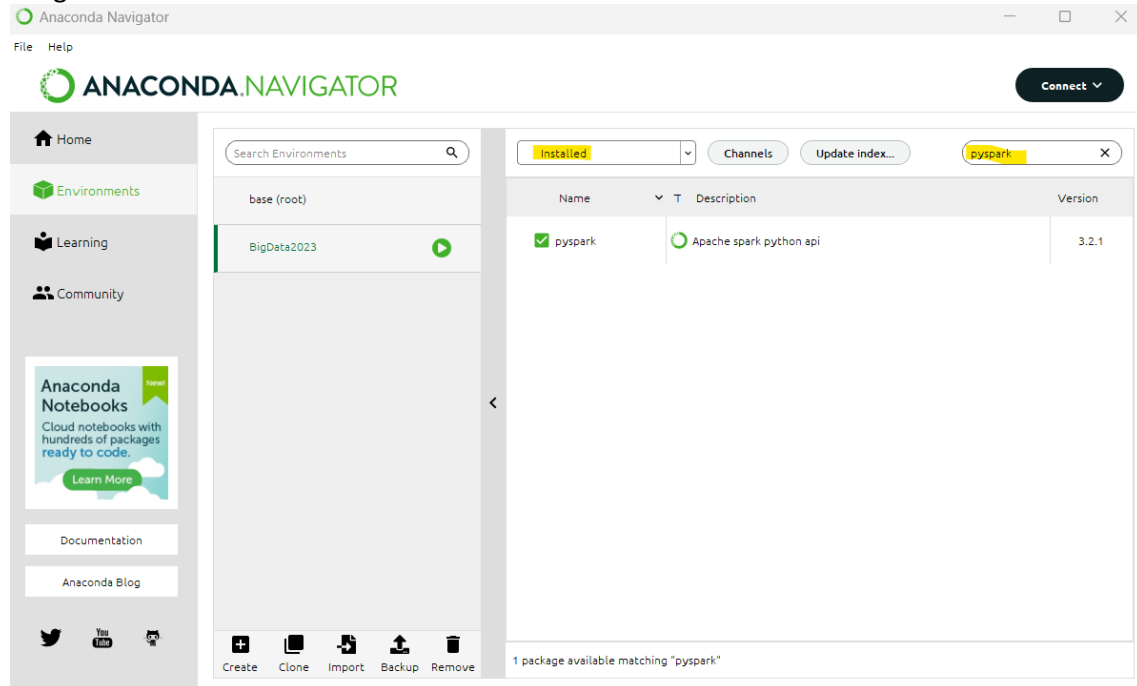


Instalar PySpark – Modo 1: desde el desplegable de Anaconda





Luego:



Instalar PySpark – Modo 2: desde el terminal

- Colocarnos en el entorno apropiado
- Instalar el paquete con pip

```
conda env list
```

```
conda activate BigData2023
```

```
pip install pyspark
```