

Capítulo 1

Software e Instalación

El lenguaje de programación Python fue creado alrededor de 1990 por el científico en computación holandés Guido van Rossum y su nombre es un tributo al grupo cómico Monty Python del cual Guido es admirador. Es un lenguaje interpretado o de script multiplataforma con una sintaxis muy clara y orientado a objetos, que favorece crear código fácilmente legible y reutilizable. La página web de Python es <http://www.python.org>.

El programa Python (como programa, no como lenguaje) posee un ambiente interactivo que nos permite ejecutar instrucciones del lenguaje Python directamente. Para ello, en la terminal basta con digitar el comando:

```
gprieto > python
Python 3.5.3 (default, Jan 21 2017, 15:44:58)
[GCC 4.2.1 Compatible Apple LLVM 8.0.0 (clang-800.0.42.1)] ...
Type "help", "copyright", "credits" or "license" ...
>>>
```

El programa ofrece un *prompt* (`>>>`), esperando instrucciones del usuario, similar a lo que sucede con *Matlab*. Las instrucciones son interpretadas y ejecutadas de manera inmediata. Esta forma de interactuar con Python es útil, ya que permite obtener respuestas inmediatas, y se puede determinar si el comando fue introducido correctamente. Sin embargo, tiene la desventaja que el código y la secuencia de comandos que se hayan utilizados no se guardan, y por lo tanto al siguiente día se debe repetir toda la operación.

Una segunda opción para interactuar con Python, es a través de archivos de texto o *scripts*, los cuales pueden tener una serie de comandos que Python va a leer, interpretar y ejecutar en el orden dado por el *script*. Este archivo puede ser reutilizado posteriormente, sin embargo la retroalimentación de Python sólo se da cuando se ejecute el *script*.

1.1. Por qué Python?

Por que elegimos Python, en vez de Matlab o lenguajes como Fortran o C? Python es un lenguaje sencillo de leer y aprender, y ha ganado en popularidad recientemente. Adicionalmente Python es gratuito para todos los usuarios y se puede instalar en cualquier sistema operativo de uso común en los computadores del hogar y la oficina. Adicionalmente, existen gran cantidad de librerías, también gratuitas, en Python para geociencias, sismología, geofísica, generación de mapas, análisis de datos, etc. además de librerías más generales para análisis estadístico y numérico como Pandas, SciPy, NumPy.

La distribución de Python de *Anaconda* permite instalarlo en cualquier sistema operativo, incluye las librerías más comunes para computación científica y generación de gráficas de alta calidad y permite la administración de *virtual environments*. Anaconda además incluye *Jupyter Notebooks*

1.1.1. Jupyter Notebooks

Para el programador experto (o de vieja guardia como yo) la interacción con cualquier lenguaje de programación se hace a través de *scripts* o código fuente que luego es ejecutado en Python (o compilado por Fortran, C, etc.).

Sin embargo hoy, los códigos de Python (R, y otros lenguajes) pueden ser ejecutados, visualizados y compartidos con otras personas a través de *Jupyter Notebooks*. En varios lugares (incluso acá en el Departamento) se usa *Jupyter Notebooks* para enseñar a los alumnos Python y en general la programación científica.

Jupyter es una aplicación web de código abierto (open-source) que permite crear y compartir código Python, permite visualizar figuras dentro del programa e incluir Notas de texto, comentarios, etc. Los *Notebooks* son fáciles de guardar y compartir y son hoy en día aceptadas y usadas por la comunidad científica global (Shen, 2014). La ventaja de *Jupyter Notebooks* es que se usa a través de un navegador de internet (Explorer, Chrome, Firefox, etc.). Esto permite que sin importar el sistema operativo (Windows, Mac, Linux), los programas de Python van a funcionar de igual manera en cualquiera de ellos.

1.2. Instalación Python a través de *Anaconda*

La instalación de Python a través de Anaconda es muy sencilla en cualquier sistema operativo. La página de Anaconda es [anaconda.org](https://www.anaconda.org) o para descargar directamente <https://www.anaconda.com/download/>. La página web generalmente detecta el sistema operativo, y provee el link para descargar el programa en su última versión estable. Se recomienda instalar la versión para Python 3.6 (Figura 1.1), no la versión 2.7.

Después de descargar el programa de instalación (para cualquier sistema operativo), se siguen los comandos para instalar la versión de Anaconda (Figura 1.2).

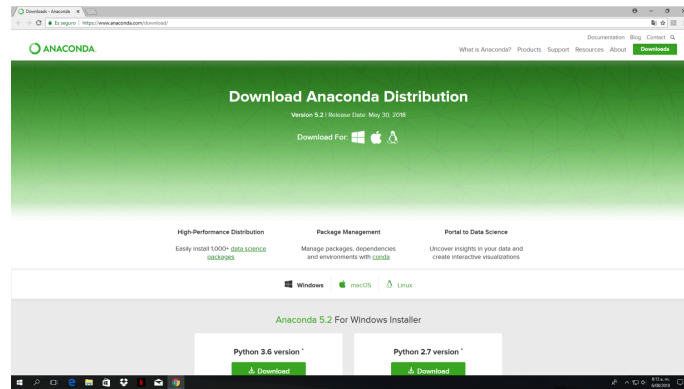


Figura 1.1: Página web para descargar Anaconda. El sistema operativo del computador es detectado automáticamente, en este caso Windows 10.

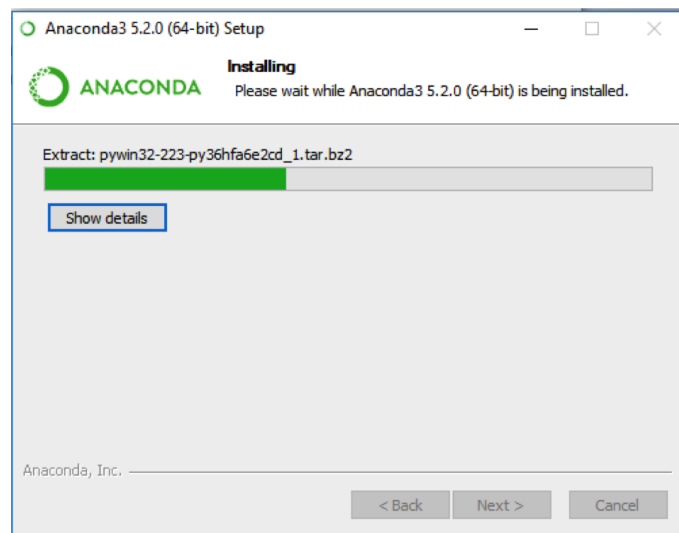


Figura 1.2: En proceso de instalación de Anaconda para Windows 10.

1.3. Cómo usar Python?

Tanto en Linux como en Mac, la correcta instalación de Anaconda permite llamar a Python directamente desde la terminal. Para abrir un *Notebook* basta con digitar

```
> jupyter notebook
```

o en su defecto (macs)

```
> jupyter-notebook
```

En Windows esto es un poco diferente, ya que Windows no está basado en un sistema tipo-unix, como Mac o Linux. Se puede acceder a Python o a Jupyter con el `cmd`. Sin embargo, la instalación de Anaconda en Windows permite correr el Jupyter Notebooks directamente, e incluso tiene una pantalla de terminal para administrar los paquetes de Python y correr los *Notebooks*.