

Jupyter Notebook

Notas por: Linda López

Compartir nuestro código con otros siempre puede ser un problema. Especialmente porque no importa que tan bien podamos comentar el código siempre habrán partes difusas que otra persona no logren entender.

```
In [1]: # es importante dejar en claro que queremos expresar, a no ser que quieras confundir a los chicos
print ('Hola chicos')
```

Hola chicos

Sobre todo la reproducibilidad del código (que otra persona pueda correrlo sin obtener errores es muy difícil de lograr). Para esto, podemos encontrar herramientas donde pareciera que pudiéramos armar una especie de "cuaderno" escribiendo los detalles de cada pedazo de código y ver los resultados del mismo sin tener que ejecutarlo estrictamente, este es el caso de Jupyter Notebook.

A primera vista Jupyter puede verse como un IDE (Integrated Development Environment) sin embargo más que como IDE es mejor verlo como un SAAS (Software as a Service) utilizado ampliamente en análisis numérico, estadística y machine learning, entre otros campos de la informática y las matemáticas. Una ventaja de este, es a que como programa nos permite a nosotros armar con nuestro código una especie de cuaderno interactivo en Python (por defecto, aunque si quieren veremos más adelante cómo permitir otros lenguajes) de manera dinámica, otra de las ventajas es que todo funciona desde un navegador y es muy facil adaptarse a su modo de uso.

Primer vistazo

Una vez que abrimos Jupyter desde Conda Navigator (y un environment decidido) o la terminal, lo primero que vemos es que se despliega en nuestro navegador predeterminado la Home o Workspace del mismo. Donde podemos observar las carpetas de nuestra computadora. Es conveniente saber que no permite navegar a carpetas superiores a la inicial.

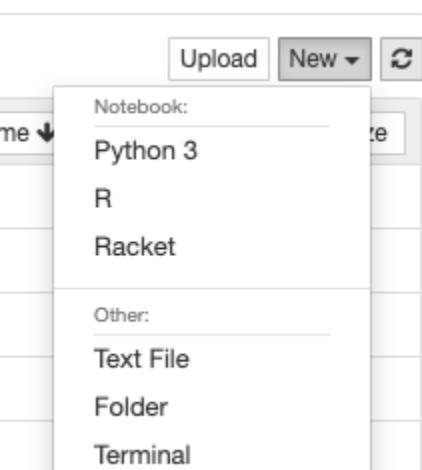


Con las opciones de la esquina superior derecha podemos cerrar Jupyter una vez terminado nuestro trabajo. La pestaña files nos muestra las carpetas disponibles, running los archivos que se estan corriendo en el programa actualmente (Hay que cerrar de vez en cuando los que ya no usaremos), y una tercera pestaña que para no complicarnos la vida, no entraremos en detalles pero es importante tener en cuenta su existencia.

Ahora, con el botón New, podemos crear:

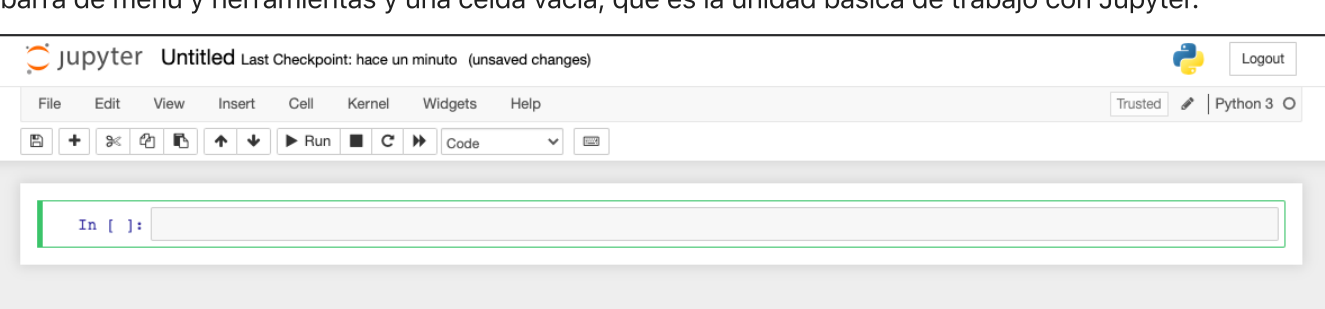
- Carpetas (recomendable para organizar todo).
- Archivos de texto (Jupyter aqui funciona como un editor simple).
- Terminales.
- Cuadernos (que es la opción que usaremos en el taller y en si la que a mi parecer más usaran en la vida científica)

Los cuadernos (a veces les diremos notebooks por costumbre) funcionan con kernels y vienen de base en Python, sin embargo podemos meter más kernels a Jupyter para generarlos con más lenguajes (en mi caso lo tengo también con un kernel para que me genera cuadernos de Racket y otro para R).



Para los alcances de este taller basta usar el kernel default con Python que debe estar disponible si la instalación fue exitosa.

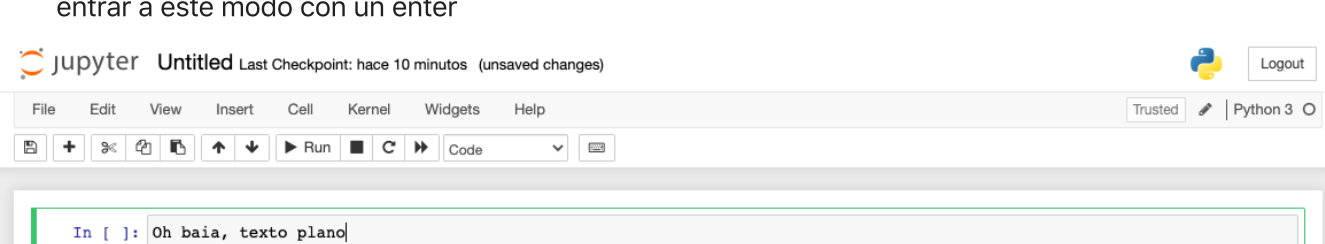
Cuando creamos un cuaderno, se nos abre en una pestaña nueva. La interfaz es muy sencilla, con una barra de menú y herramientas y una celda vacía, que es la unidad básica de trabajo con Jupyter.



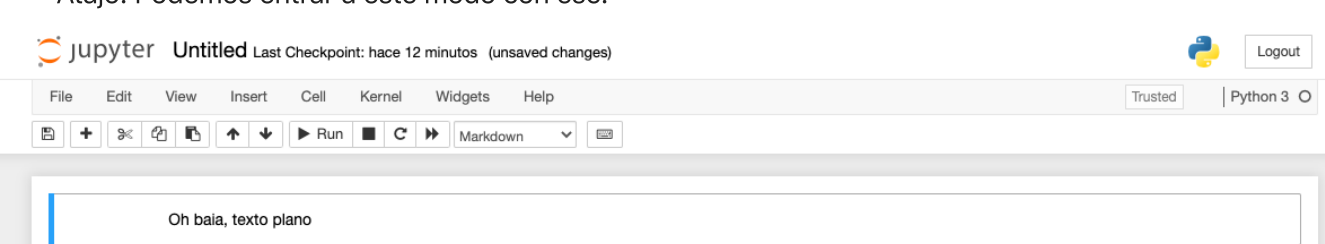
Jupyter me permite estructurar el código por diferentes bloques o celdas teniendo la posibilidad de ejecutar cada uno de estos bloques de código por separado a diferencia de por ejemplo en Python o en otros IDE donde siempre que ejecutamos el código tenemos que ejecutarlo todos desde la primera línea hasta la última línea.

Es importante remarcar que tenemos dos modos de trabajo en Jupyter:

- Edición (barra lateral izquierda verde): Permite modificar el contenido de las celdas. Atajo: Podemos entrar a este modo con un enter



- Mando (barra lateral izquierda azul): Permite ejecutar celdas o modificar el cuaderno y su estructura. Atajo: Podemos entrar a este modo con esc.

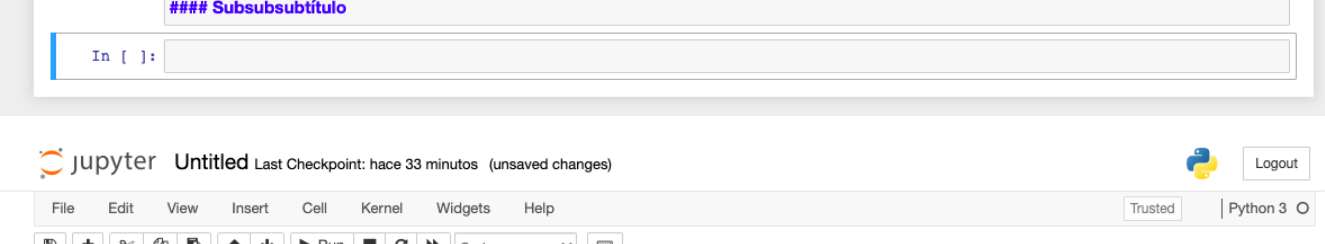
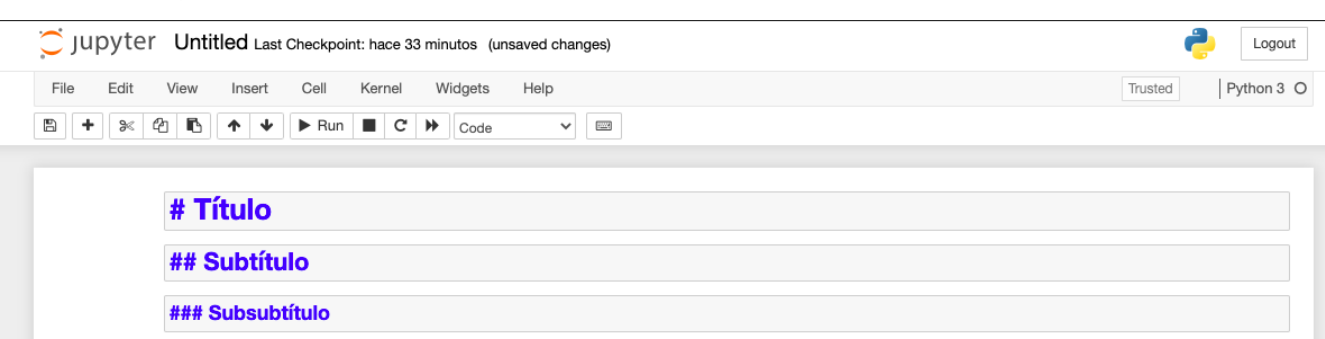


Los cuadernos tienen dos tipos de celdas, de código y de texto. La celda que estás leyendo es una celda de texto escrita con Markdown, un lenguaje de marcado parecido al que utiliza wikipedia para sus páginas o al HTML. Las celdas de código son ejecutables, es decir, se pueden correr individualmentePara ejecutar una celda, haremos clic en el botton "Run" o bien, como atajos para la ejecución tenemos Control + Enter, o si queremos pasar automáticamente a la siguiente después de ejecutarla hacemos Shift + Enter.

Todos los atajos de Jupyter estan en el botón con símbolo de teclado o bien pueden encontrarlos [AQUI](#) condensados tanto para usuarios Windows como de MacOS. Creo el mas útil que les puedo comentar es pulsar doblemente la tecla d para eliminar una celda, esto en modo mando (azul), agregar una celda debajo pulsando b y agregar una celda arriba pulsando a.

Escribiendo texto

Primero, debemos indicar que el tipo de celda es Markdown (atajo presionando m en modo mando) ya que por default nos aparecera en tipo código (atajo presionando y en modo mando). Podemos separar el contenido del documento con cabeceras para secciones, subsecciones, subsubsecciones etc. Solo tenemos que añadir de una a cinco almohadillas (#) al principio de la línea, tantas como el nivel que queramos tenga el título.



Podemos añadir imagenes (con tipo Markdown) manualmente desde Edit/ Insert image o bien con el siguiente comando indicando anchura, alineación, etc. Omitiendo los ##

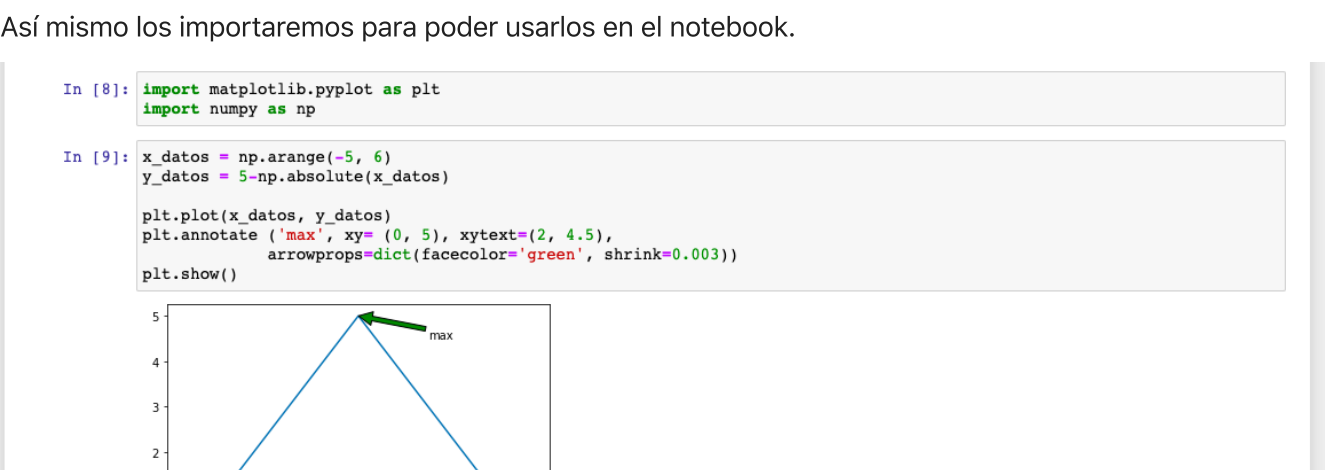
```
In [2]: # <img src='...' align='...'>#
```

Si queremos mostrar una lista, solo basta poner guión + espacio.

Código

Es importante, a la par que evidente, tener instalados los paquetes de Python que queramos utilizar. Una vez hecho, podemos importarlos al cuaderno. Para este ejemplo usare Matplotlib. La instalación de los mismos se hace en las primeras celdas del notebook. (IMPORTANTE)no mezclar instaladores, pip y conda. Aqui en el taller nos manejaremos con conda para evitar problemas de compatibilidad.

Así mismo los importaremos para poder usarlos en el notebook.



LaTeX

Para mostrar código $LaTeX$ en nuestro cuaderno, simplemente tenemos que escribir entre dólares (simples o dobles, tal cual como en un editor de .tex común), y marcar la celda en Markdown.

\$\$Prueba\ por\ \LaTeX{:}\ Si\ esta\ escrito\ en\ \LaTeX\ es\ que\ es\ cierto\$\$

Agregando una \ entre cada palabra para espacios.

Cualquier duda no duden en escribirme. Lo importante es irse soltando y practicar en esta gran herramienta.