

Usando Jupyter Notebook en el aula

Edward Villegas-Pulgarin
@cosmoscalibur



Contenido

1. Jupyter Notebook
 - a. ¿Qué es?
 - b. Instalación
 - c. Funcionamiento
 - d. Anatomía de un notebook
2. Uso general de un notebook
 - a. Escritura con markdown
 - b. Ecuaciones con LaTeX
 - c. Código ejecutable
 - d. HTML y Controles
 - e. Compartir: nbviewer y mybinder
3. Notebook en clase
 - a. Documentar casos prácticos
 - b. Flujo de cálculo (básica y avanzada)
 - c. Demostraciones y ejemplos interactivos
 - d. Reportes del proceso de formación
4. Algunas extensiones útiles
 - a. nbconvert
 - b. nbsphinx
 - c. nbgrader

Jupyter Notebook

¿Qué es?

Es una aplicación web de código abierto que permite la creación de documentos interactivos.

Jupyter estados_ligados Last Checkpoint: 19/08/2017 (unsaved changes) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 2.0

Potencial armonico
El potencial armónico cumple con la descripción dada por

$$V(x) = \frac{\omega^2 x^2}{4}$$

```
In [4]: def V_arm(omega, x):  
        return omega**2 * x**2 / 4
```

```
In [5]: control_arm = fun_contenedor base()  
        agregar_control(control_arm, FloatSlider(value = 1., min = .1, max = 4., step = .1, description='$\omega$'))  
        boton_arm = Button(description='Simular potencial')  
        def click_arm(boton):  
            E_max = control_arm.children[0].value  
            L = control_arm.children[1].value  
            N = control_arm.children[2].value  
            n = control_arm.children[3].value  
            omega = control_arm.children[4].value  
            Vx = lambda x: V_arm(omega, x)  
            Solve_Schr(Vx, E_max, L, N, n)  
            clear_output(wait=True)  
  
        boton_arm.on_click(click_arm)  
        display(control_arm, boton_arm)
```

E_{max} L N n ω

Simular potencial

$E_1 = 0.48633928124996284$

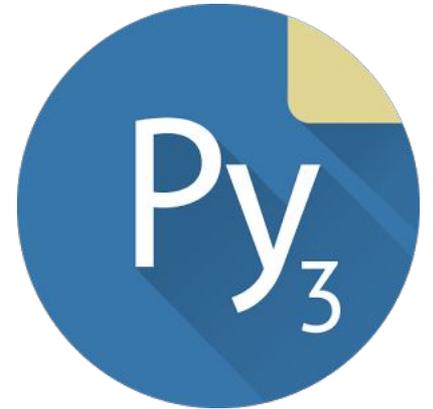
Jupyter Notebook Instalación

Gracias a la distribución Anaconda, es posible tenerlo instalado por defecto.

También se puede usar la utilidad pip:

```
pip install notebook
```

En android es posible gracias a PyDroid (python2 solamente).

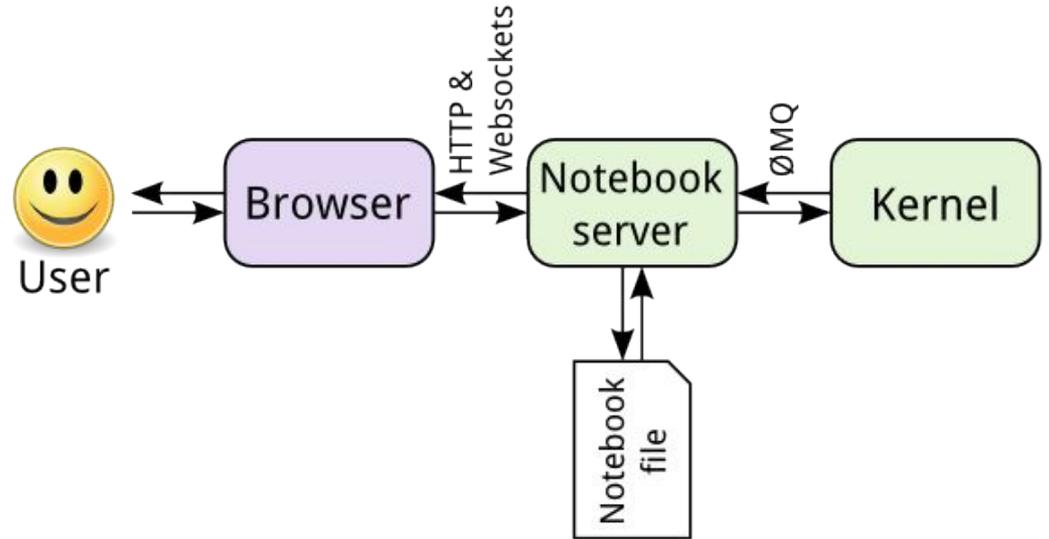


Jupyter Notebook

Funcionamiento

Aplicación servidor cliente compuesta de:

1. *Kernel*
2. Servidor de *notebooks*
3. Interfaz web
4. Archivo *notebook*



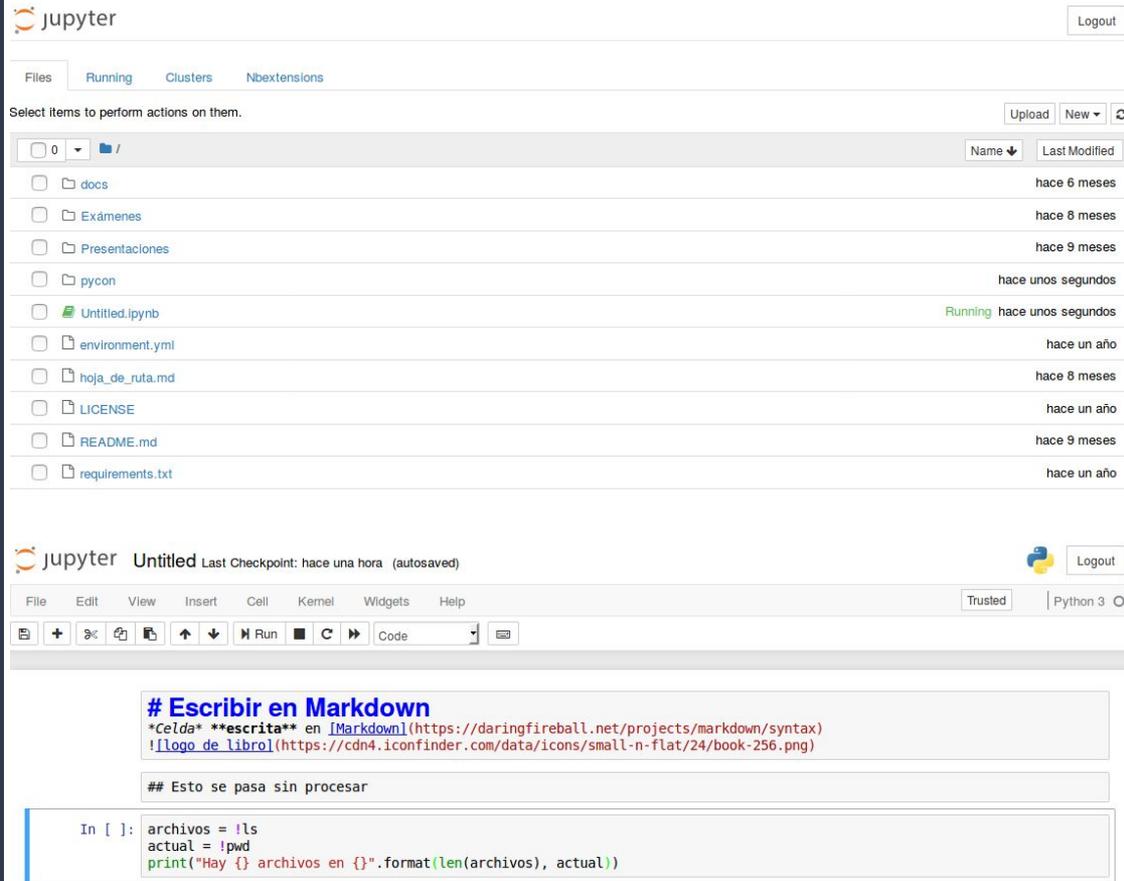
Jupyter Notebook

Anatomía de un notebook

Al iniciar la aplicación, nos encontramos el tablero (*dashboard*).

El documento *notebook* lo forman celdas de diferente naturaleza.

- Celdas markdown.
- Celdas de código.
- Celdas no procesadas (*raw*).



The image shows the Jupyter Notebook interface. At the top, there's a 'jupyter' logo and a 'Logout' button. Below that, there are tabs for 'Files', 'Running', 'Clusters', and 'Nbextensions'. A message says 'Select items to perform actions on them.' with 'Upload', 'New', and a refresh icon. A file browser shows a directory structure with files like 'docs', 'Exámenes', 'Presentaciones', 'pycon', 'Untitled.ipynb' (marked as 'Running'), 'environment.yml', 'hoja_de_ruta.md', 'LICENSE', 'README.md', and 'requirements.txt'. Below this is the notebook editor for 'Untitled', showing the last checkpoint as 'hace una hora (autosaved)'. The menu includes 'File', 'Edit', 'View', 'Insert', 'Cell', 'Kernel', 'Widgets', and 'Help'. The toolbar has icons for file operations, running, and code execution. The notebook content shows a markdown cell with a heading '# Escribir en Markdown' and a code cell with the following code:

```
In [ ]: archivos = !ls
actual = !pwd
print("Hay {} archivos en {}".format(len(archivos), actual))
```

Usando Notebook Markdown

Se usa un lenguaje de marcado simple como markdown para la escritura de texto.

Escribir en Markdown

Celda escrita en [Markdown](#)

Podemos incorporar imágenes al estilo de enlaces `![[Texto Alternativo](ruta o url de la imagen)]`.



Bloque de código

```
a = len('casa')
print(a)
```

- Mandar presentación.
 - Preparar.
 - Ajustes finales.
- Ir a las charlas del viernes.
- Exponer el sábado.
- Ir a los talleres del domingo.
 1. Tanda 1.
 2. Tanda 2.

```
# Escribir en Markdown
*Celda* **escrita** en [Markdown](https://daringfireball.net/projects/markdown/syntax)
Podemos incorporar imágenes al estilo de enlaces ![[Texto Alternativo](ruta o url de la imagen)].
![[logo de libro](https://cdn4.iconfinder.com/data/icons/small-n-flat/24/book-256.png)]
```

Bloque de código

```
```python
a = len('casa')
print(a)
```
```

- + Mandar presentación.
 - Preparar.
 - Ajustes finales.
- + Ir a las charlas del viernes.
- + Exponer el sábado.
- + Ir a los talleres del domingo.
 1. Tanda 1.
 1. Tanda 2.

Usando Notebook LaTeX

Gracias a la biblioteca de javascript MathJax podemos realizar la escritura de ecuaciones adecuadamente formateadas en *notebook* usando sintaxis LaTeX.



Esto es una ecuación en modo texto $E=mc^2$.

Esto es una ecuación en modo matemático $\int_0^{\infty} \frac{\alpha}{\pi} T(x) dx$

Esto es una ecuación en modo texto $E = mc^2$.

Esto es una ecuación en modo matemático

$$\int_0^{\infty} \frac{\alpha}{\pi} T(x) dx$$

Usando Notebook Código

A través de los diferentes núcleos (*kernels*) disponibles es posible usar distintos lenguajes de programación para interactuar con sus resultados.

```
In [1]: archivos_lista = !ls
archivos_total = len(archivos_lista)
directorio_actual = !pwd
print("Usted tiene {} archivos en el directorio {}".format(archivos_total, directorio_actual))
print("Lista de archivos")
print(archivos_lista)
```



Usted tiene 10 archivos en el directorio ['/home/cosmoscalibur/Dropbox/git/cosmoscalibur/hcomputacionales']
Lista de archivos
['docs', 'environment.yml', 'Exámenes', 'hoja_de_ruta.md', 'LICENSE', 'Presentaciones', 'pycon', 'README.md', 'requiremen
ts.txt', 'Untitled.ipynb']

```
In [12]: d = lm(a ~ poly(b, 2, raw=TRUE))
# poly crea el modelo polinómico
# variable, grado, y polinomio tradicional
summary(d)
```

Call:
lm(formula = a ~ poly(b, 2, raw = TRUE))

Residuals:

| 1 | 2 | 3 | 4 | 5 |
|---------|--------|---------|---------|--------|
| -0.4425 | 0.7364 | -0.0339 | -0.5677 | 0.3077 |

Coefficients:

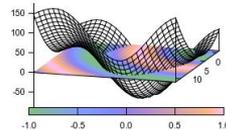
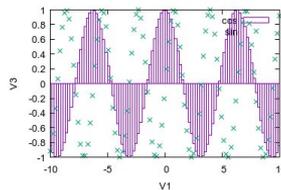
| | Estimate | Std. Error | t value | Pr(> t) |
|-------------------------|----------|------------|---------|----------|
| (Intercept) | 0.8418 | 0.8352 | 1.008 | 0.4196 |
| poly(b, 2, raw = TRUE)1 | -1.8041 | 1.0315 | -1.749 | 0.2224 |
| poly(b, 2, raw = TRUE)2 | 1.5376 | 0.2465 | 6.237 | 0.0248 * |

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.7603 on 2 degrees of freedom
Multiple R-squared: 0.9943, Adjusted R-squared: 0.9885
F-statistic: 173.6 on 2 and 2 DF, p-value: 0.005727



```
In [3]: funcion_cos(x) = cos(a*x) # Definición de funciones
funcion_sin(x) = sin(a*x)
set ytics
set size nosquare # Grafico sin conservacion de aspecto
plot a=1, funcion_cos(x) t "cos" w boxes, \
a=2, funcion_sin(x) t "sin" w p # Grafico sobrepuestos y parametricos
```



```
Out[3]: funcion_cos(x) = cos(a*x) # Definición de funciones
funcion_sin(x) = sin(a*x)
set ytics
set size nosquare # Grafico sin conservacion de aspecto
set output '/tmp/gnuplot-inline-1518060188.3310366.244417406904.png'
plot a=1, funcion_cos(x) t "cos" w boxes, a=2, funcion_sin(x) t "sin" w p # Grafico sobrepuestos y parametricos
unset output
```

Usando Notebook HTML y controles

Los *notebooks* pueden personalizarse con ayuda de HTML y CSS, y diseñar controles web y javascript (con ipywidgets).

¡Funciona! HTML en notebook.

```
<div class="alert alert-success">  
<strong>¡Funciona!</strong> HTML en notebook.  
</div>
```

```
In [3]: from ipywidgets import Text, VBox, IntSlider  
from IPython.display import display  
a = Text(value="Predefinido", description="Escriba")  
b = IntSlider(value=10, description="Edad", min=0, max=120, step=1)  
gui = VBox(children=[a, b])  
display(gui)
```

Escriba

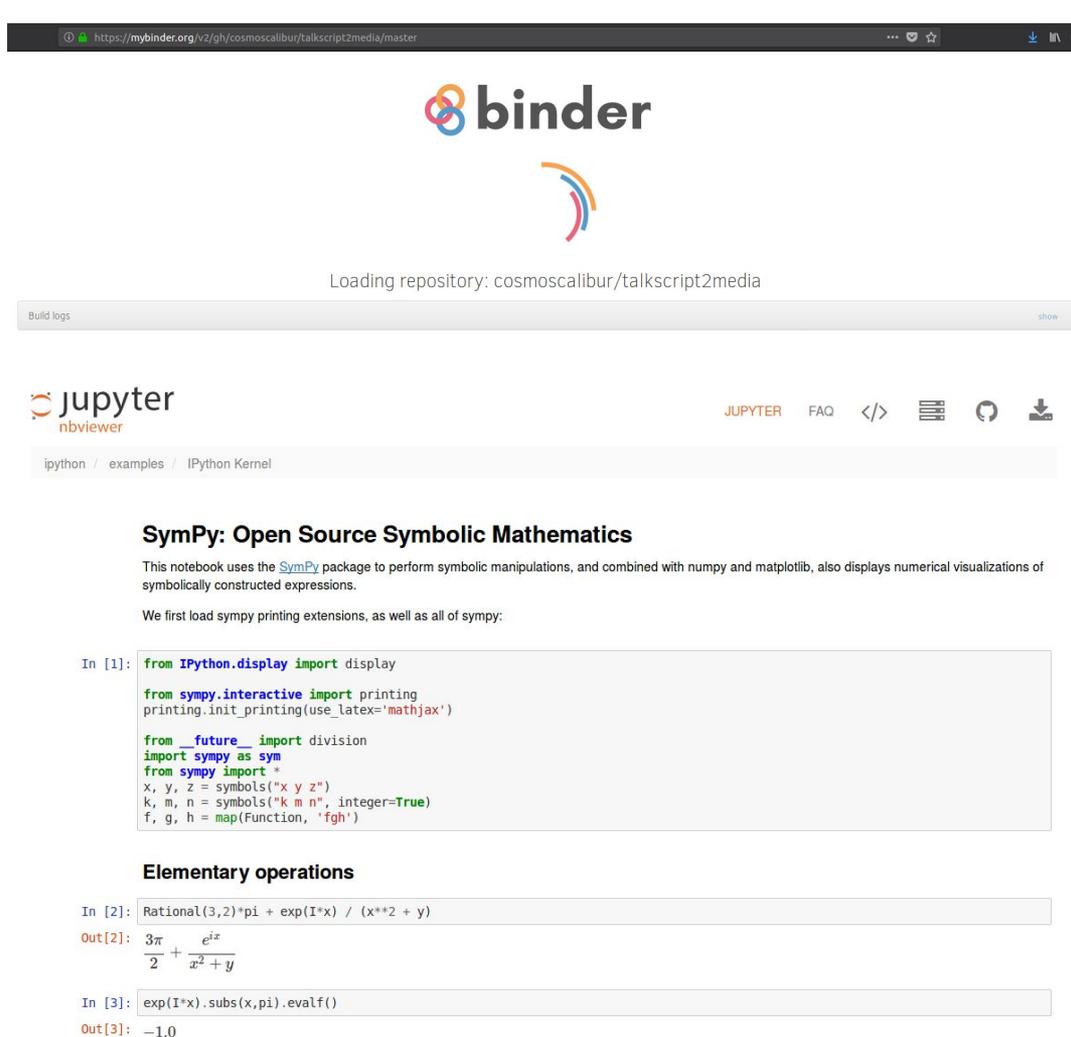
Edad 10

Usando Notebook Compartir

La rápida aceptación del formato por distintas comunidades ha permitido la aparición de infraestructura de acceso gratuito para compartir *notebooks*.

Servicios:

- MyBinder.
- NBViewer.
- Authorea (existe versión paga).
- Cocalc (existe versión paga).



https://mybinder.org/v2/gh/cosmoscalibur/talkscript2media/master

binder

Loading repository: cosmoscalibur/talkscript2media

Build logs show

jupyter
nbviewer

JUPYTER FAQ </> ☰ 🔄 ⬇️

ipython / examples / IPython Kernel

SymPy: Open Source Symbolic Mathematics

This notebook uses the [SymPy](#) package to perform symbolic manipulations, and combined with numpy and matplotlib, also displays numerical visualizations of symbolically constructed expressions.

We first load sympy printing extensions, as well as all of sympy:

```
In [1]: from IPython.display import display
        from sympy.interactive import printing
        printing.init_printing(use_latex='mathjax')

        from __future__ import division
        import sympy as sym
        from sympy import *
        x, y, z = symbols("x y z")
        k, m, n = symbols("k m n", integer=True)
        f, g, h = map(Function, 'fgh')
```

Elementary operations

```
In [2]: Rational(3,2)*pi + exp(I*x) / (x**2 + y)
```

Out[2]: $\frac{3\pi}{2} + \frac{e^{ix}}{x^2 + y}$

```
In [3]: exp(I*x).subs(x,pi).evalf()
```

Out[3]: -1.0

Aula Notebook Experiencias

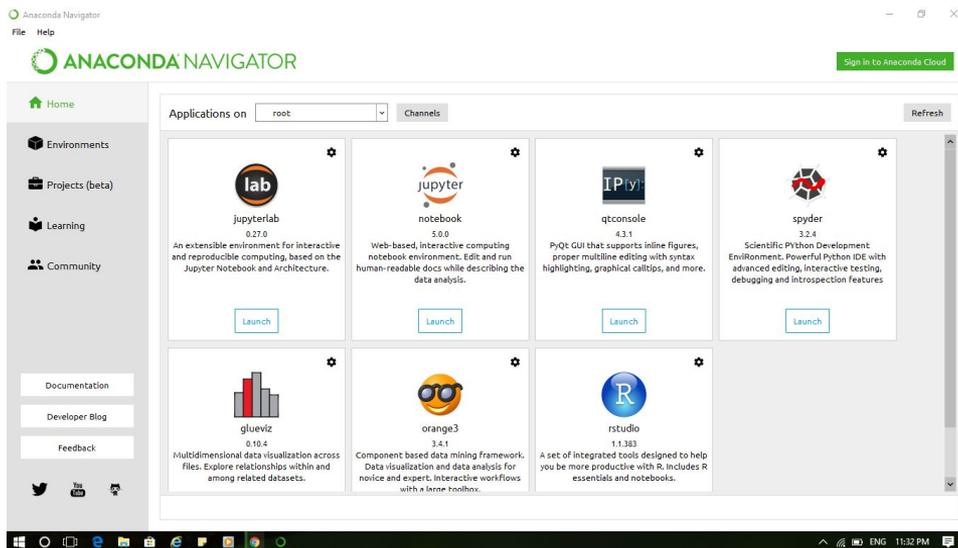
Algunos casos de uso:

- Documentar
- Calcular
- Interacción
- Reportes

En el directorio en el cual se encuentran los *notebooks*
jupyter notebook

Si usa Anaconda puede usar Anaconda Navigator para
abrir Notebook.

Veamos algunos ejemplos (disponibles en
<https://github.com/cosmoscalibur/aula-notebook>).



Extensiones Notebook Extras

Para complementar lo ya visto, es posible extender el potencial de Jupyter notebooks con extensiones.

Entorno al aula de clase, recomiendo:

- nbconvert
- nbsphinx
- nbgrader

Referencias

Material de la presentación disponible en:

<https://github.com/cosmoscalibur/aula-notebook>

How IPython and Jupyter Notebook work

http://jupyter.readthedocs.io/en/latest/architecture/how_jupyter_ipython_work.html

The Jupyter notebook

<http://jupyter-notebook.readthedocs.io/en/stable/index.html>

Jupyter Widgets

<https://ipywidgets.readthedocs.io/en/latest/>

Jupyter Notebook Quickstart

<http://jupyter.readthedocs.io/en/latest/content-quickstart.html>

Jupyter kernels

<https://github.com/jupyter/jupyter/wiki/Jupyter-kernels>

Herramientas computacionales

<https://github.com/cosmoscalibur/hcomputacionales>

Cuántica Jupyter

https://github.com/fisicatyc/Cuantica_Jupyter