



---

# Jupyter: um caderno computacional

1ª Semana de Workshops, ICMC - USP, 03/06/2021

Por Breno Lívio

---

---

# O que pretendo passar

- História e introdução dando diversos exemplos de como pode ser usado;
  - As diversas formas de se usar o sistema Jupyter;
  - Instalação;
  - Aplicações com o Jupyter: Introdução às células, geração de imagens e widgets;
  - Usando nbextensions;
  - Usando nbconvert;
  - JupyterLab.
-

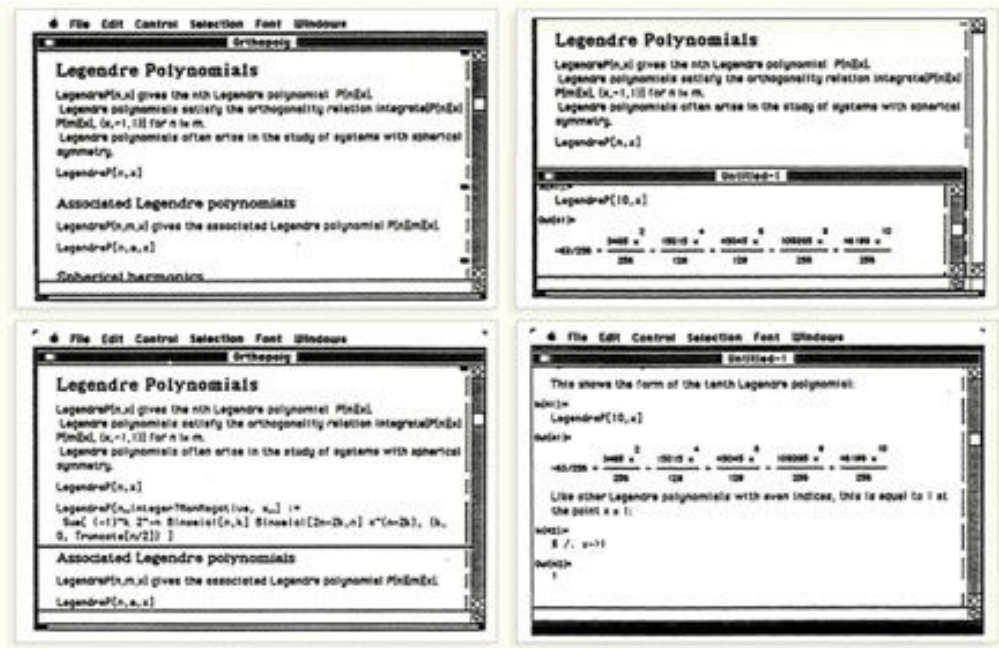
---

# História dos cadernos computacionais

Em 1987, foi desenvolvido uma espécie de caderno computacional pensando no software Mathematica.

Quando você observa como esses cadernos foram estruturados, você percebe que eles dependiam de uma hierarquia de células que permitia o esboço e a seção de documentos, que você agora também encontra nos cadernos Jupyter.

---



Retirado de <https://www.wolfram.com/mathematica/scrapbook/>

---

---

# O que é o Jupyter?

Jupyter é uma ferramenta interativa de código aberto e conhecida como um caderno de anotações computacional, que os pesquisadores podem usar para combinar códigos de software, saídas computacionais, textos explicativos e recursos multimídia em um único documento [1].

---

---

---

# Project Jupyter

O Projeto Jupyter é um projeto sem fins lucrativos e de código aberto, nascido do projeto IPython em 2014, à medida que evoluiu para apoiar a ciência dos dados interativos e a computação científica em todas as linguagens de programação [2].

---



**JULia + PYThon + R**



---

# IPython

No final de 2001, cerca de vinte anos depois que Guido van Rossum começou a desenvolver o Python no Instituto Nacional de Pesquisa em Matemática e Ciência da Computação na Holanda, Fernando Pérez começou a desenvolver o IPython.

IPython tem agora apenas dois papéis a cumprir: ser o backend Python para o Jupyter Notebook, que também é conhecido como o kernel, e um shell Python interativo [3].

---



```

Python 2.7.3 (default, Jul 10 2012, 18:48:25)
Type "copyright", "credits" or "license" for more information.

IPython 0.13.1 -- An enhanced Interactive Python.
?      -> Introduction and overview of IPython's features.
%quickref -> Quick reference.
help    -> Python's own help system.
object? -> Details about 'object', use 'object??' for extra details.

In [1]: import numpy as np

In [2]: N = 3000

In [3]: a = np.random.randn(N,N)

In [4]: b = np.random.randn(N,N)

In [5]: np.dot
np.dot   np.double

In [5]: np.dot(a, b)
Out[5]:
array([[ 65.45670189,  64.86918252, -120.2955101 , ...,  46.52919413,
         1.62384273, -117.27453077],
       [ 103.8332094 , -63.19741333,  25.63850851, ...,  10.43730591,
        -98.22728902, -9.16795735],
       [-36.45095805,  44.32128353, -17.58969917, ..., -125.12907291,
        -70.58206964, -32.85757429],
       ...,
       [-42.46168724,  36.45522834,  28.8765628 , ...,  39.40943867,
        -16.43199427, -63.08194364],
       [-84.46717927,  28.06738004,  32.09026395, ..., -42.127647 ,
        -116.20291034,  32.02266900],
       [ 56.79843374,  23.60837948,  52.24793136, ..., -35.53881726,
        -21.19119431, -151.71414646]])

In [6]: %timeit np.dot(a,b)
1 loops, best of 3: 2.17 s per loop

In [7]: np.
Display all 551 possibilities? (y or n)
np.ALLOW_THREADS      np.convolve      np.iscomplex      np.ravel
np.BUFSIZE            np.copy       np.iscomplexobj   np.ravel_multi_index

```

Retirado de <https://ipython.org/screenshots/index.html>

---

# Importância desses cadernos computacionais

Esses cadernos podem auxiliar em diversos tipos de casos:

- Ensino de programação;
  - Produção de relatórios científicos;
  - Desenvolvimento de aplicações de estatística e ciência de dados.
-

---

# Crescimento do uso desses cadernos

Há cerca de 3 anos, havia 1.230.000 cadernos Jupyter publicados no GitHub. Em outubro de 2020, este número havia crescido 8 vezes, e conta com 9.720.000 cadernos [4].





---

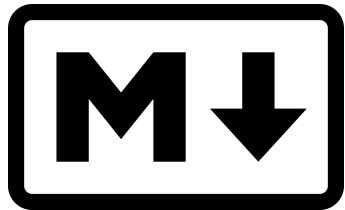
## R Markdown

Outra “alternativa” ao Jupyter pode ser o RMarkdown. Ele funciona de forma análoga ao Jupyter pensando em células de texto e código.

As diferenças notáveis é que o arquivo do RMarkdown é possível editar como texto, enquanto o Jupyter gera um código em JSON (JavaScript Object Notation), como se o caderno já fosse pensado como uma “aplicação web”.

Além disso, para reproduzir widgets como o Jupyter, recomenda-se o uso do pacote Shiny.

---



---

## Markdown no Jupyter

O Markdown é uma linguagem de marcação (*markup language*) para formatar textos, tal como HTML, XML e LaTeX, que podem ser também consideradas linguagens de marcação.

Para células de texto no Jupyter, a formatação é em Markdown.

---

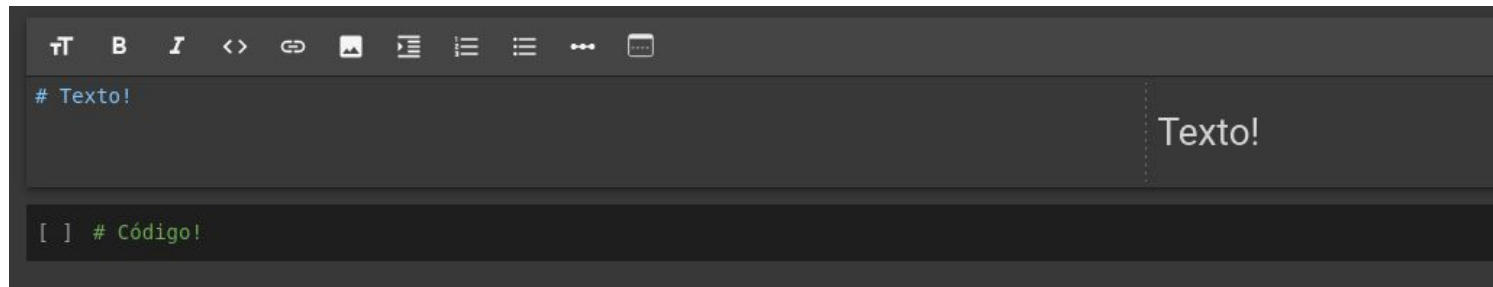
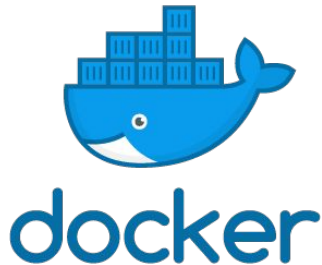


Imagem: Célula de texto e de código no Jupyter, pelo Google Colab

---



---

## Diversas formas de se usar o Jupyter

Online, por um servidor que armazena o sistema Jupyter:

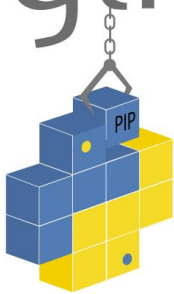
Google Colab, Binder.

Instalando no computador:

- Instalar diretamente pelo pip ou conda;
  - Instalar extensão de Python no Visual Studio Code;
  - Docker;
  - IDE PyCharm.
-

---

python™



pip  
Installation

<https://pip.pypa.io/en/stable/installing/#>

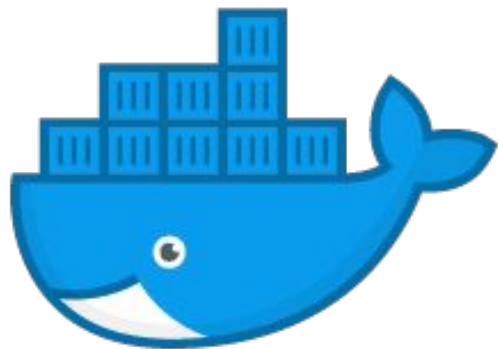
CONDA

<https://conda.io/projects/conda/en/latest/user-guide/install/index.html>

<https://jupyter.org/install>

---





docker

---

---

# Usando o Docker

Usar o Jupyter com o Docker é provavelmente o melhor caminho.

Instruções de como instalar o Docker Compose em qualquer Sistema Operacional:

<https://docs.docker.com/compose/install/>

---

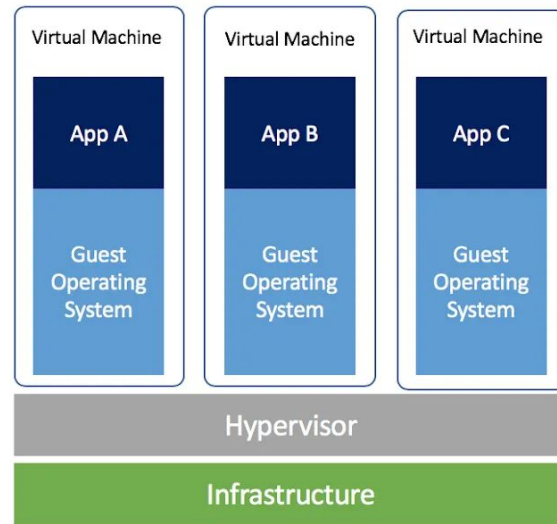
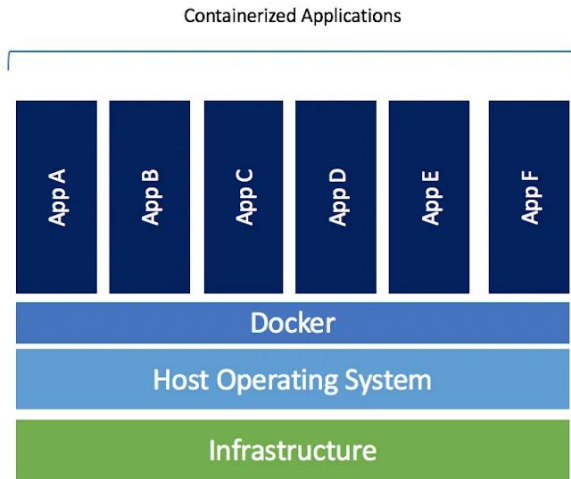
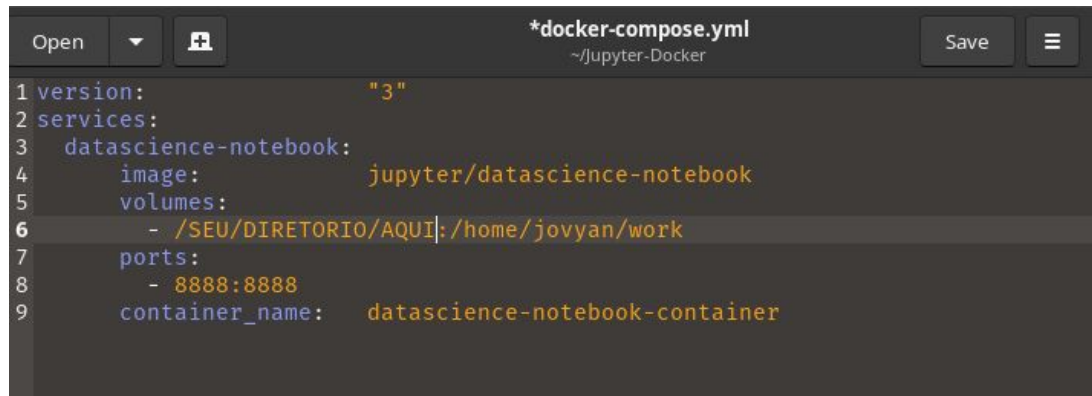


Imagem: Virtualização pelo Docker

---



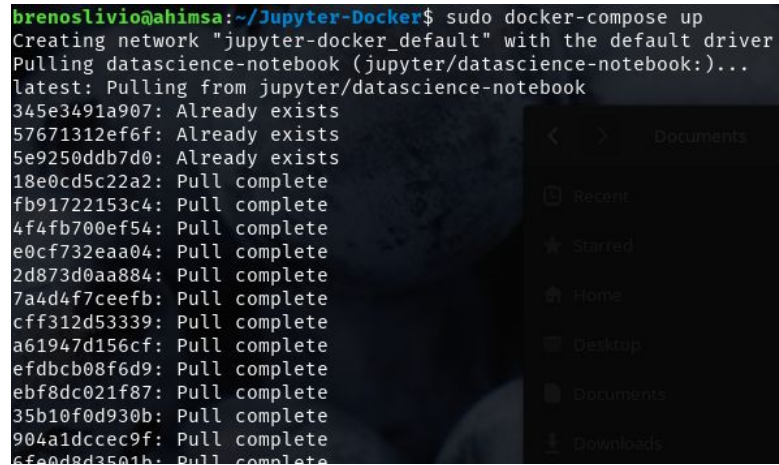
The image shows a code editor window with a dark theme. The title bar at the top reads '\*docker-compose.yml' and '~/.jupyter-Docker'. On the left, there are buttons for 'Open', a dropdown arrow, and a file icon. On the right, there are buttons for 'Save' and a hamburger menu icon. The code is as follows:

```
1 version: "3"
2 services:
3   datascience-notebook:
4     image: jupyter/datascience-notebook
5     volumes:
6     - /SEU/DIRETORIO/AQUI:/home/jovyan/work
7     ports:
8     - 8888:8888
9     container_name: datascience-notebook-container
```

Imagem: Arquivo docker-compose.yml

---

# Abrindo o Jupyter no Docker



A terminal window showing the execution of the command `sudo docker-compose up`. The output indicates the creation of a network and the pulling of the `jupyter/datascience-notebook` image. A file manager window is overlaid on the right side of the terminal, showing a sidebar with options like Documents, Recent, Starred, Home, Desktop, and Downloads.

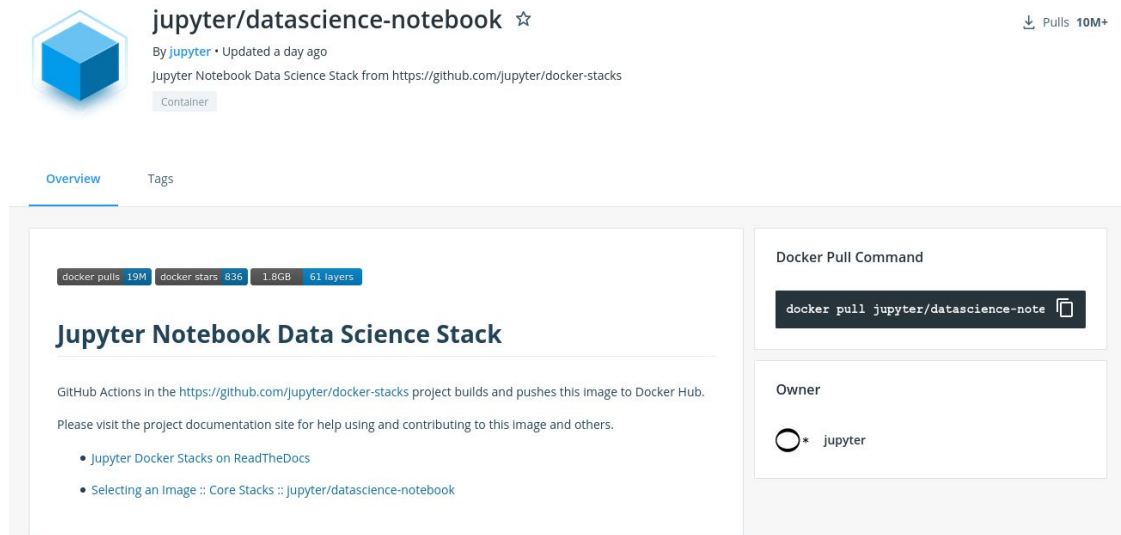
```
brenoslivio@ahimsa:~/Jupyter-Docker$ sudo docker-compose up
Creating network "jupyter-docker_default" with the default driver
Pulling datascience-notebook (jupyter/datascience-notebook)...
latest: Pulling from jupyter/datascience-notebook
345e3491a907: Already exists
57671312ef6f: Already exists
5e9250ddb7d0: Already exists
18e0cd5c22a2: Pull complete
fb91722153c4: Pull complete
4f4fb700ef54: Pull complete
e0cf732eaa04: Pull complete
2d873d0aa884: Pull complete
7a4d4f7ceefb: Pull complete
cff312d53339: Pull complete
a61947d156cf: Pull complete
efdbcb08f6d9: Pull complete
ebf8dc021f87: Pull complete
35b10f0d930b: Pull complete
904a1dccec9f: Pull complete
6fe0d8d3501b: Pull complete
```

Imagem: Comando docker-compose up

---

---

# Imagem do Jupyter



The screenshot shows the Docker Hub interface for the `jupyter/datascience-notebook` image. At the top, there's a blue cube icon representing the image. To its right, the text `jupyter/datascience-notebook` is displayed with a star icon. Below this, it says "By `jupyter` • Updated a day ago" and "Jupyter Notebook Data Science Stack from <https://github.com/jupyter/docker-stacks>". A "Container" button is visible. On the far right, it shows "Pulls 10M+" with a download icon. Below the header, there are tabs for "Overview" (selected) and "Tags". The main content area is divided into two columns. The left column contains a summary of the image: "docker pulls 19M", "docker stars 836", "1.8GB", and "61 layers". Below this is the title "Jupyter Notebook Data Science Stack". A paragraph mentions "GitHub Actions in the <https://github.com/jupyter/docker-stacks> project builds and pushes this image to Docker Hub." and "Please visit the project documentation site for help using and contributing to this image and others." There are two bullet points: "Jupyter Docker Stacks on ReadTheDocs" and "Selecting an Image :: Core Stacks :: jupyter/datascience-notebook". The right column contains a "Docker Pull Command" section with a code block showing `docker pull jupyter/datascience-note` and a copy icon. Below that is the "Owner" section, which shows a circular profile icon and the name `jupyter`.

jupyter/datascience-notebook ☆

By `jupyter` • Updated a day ago

Jupyter Notebook Data Science Stack from <https://github.com/jupyter/docker-stacks>

Container

Pulls 10M+

Overview Tags

docker pulls 19M docker stars 836 1.8GB 61 layers

## Jupyter Notebook Data Science Stack

GitHub Actions in the <https://github.com/jupyter/docker-stacks> project builds and pushes this image to Docker Hub.


Please visit the project documentation site for help using and contributing to this image and others.

- Jupyter Docker Stacks on ReadTheDocs
- Selecting an Image :: Core Stacks :: jupyter/datascience-notebook

Docker Pull Command

```
docker pull jupyter/datascience-note
```

Owner

 `jupyter`

<https://hub.docker.com/r/jupyter/datascience-notebook/>

---

---

---

colab

---

---

# nbextensions

O pacote `jupyter_contrib_nbextensions` contém uma coleção de extensões não-oficiais com contribuições da comunidade que adicionam funcionalidade ao caderno Jupyter. Estas extensões são em sua maioria escritas em Javascript e serão carregadas localmente em seu navegador.

Mais em:

<https://jupyter-contrib-nbextensions.readthedocs.io/en/latest/>

---



## Configurable nbextensions

☐ disable configuration for nbextensions without explicit compatibility (they may break your notebook environment, but can be useful to show for nbextension development)

filter: by description, section, or tags

- |  |  |  |   |
|--|--|--|---|
| <input type="checkbox"/> (some) LaTeX environments for Jupyter | <input type="checkbox"/> 2to3 Converter                          | <input type="checkbox"/> AddBefore                                 | <input type="checkbox"/> Autopep8                               |
| <input type="checkbox"/> AutoSaveTime                          | <input type="checkbox"/> Autoscroll                              | <input type="checkbox"/> Cell Filter                               | <input type="checkbox"/> Code Font Size                         |
| <input type="checkbox"/> Code prettify                         | <input type="checkbox"/> Codefolding                             | <input type="checkbox"/> Codefolding in Editor                     | <input type="checkbox"/> CodeMirror mode extensions             |
| <input checked="" type="checkbox"/> Collapsible Headings       | <input type="checkbox"/> Comment/Uncomment Hotkey                | <input checked="" type="checkbox"/> contrib_nbextensions_help_item | <input type="checkbox"/> datestamper                            |
| <input type="checkbox"/> Equation Auto Numbering               | <input type="checkbox"/> ExecuteTime                             | <input type="checkbox"/> Execution Dependencies                    | <input type="checkbox"/> Exercise                               |
| <input type="checkbox"/> Exercise2                             | <input type="checkbox"/> Export Embedded HTML                    | <input type="checkbox"/> Freeze                                    | <input type="checkbox"/> Gist-it                                |
| <input type="checkbox"/> Help panel                            | <input type="checkbox"/> Hide Header                             | <input type="checkbox"/> Hide input                                | <input type="checkbox"/> Hide input all                         |
| <input type="checkbox"/> Highlight selected word               | <input type="checkbox"/> highlighter                             | <input type="checkbox"/> Hinterland                                | <input type="checkbox"/> Initialization cells                   |
| <input type="checkbox"/> isort formatter                       | <input checked="" type="checkbox"/> jupyter-js-widgets/extension | <input checked="" type="checkbox"/> jupyter-matplotlib/extension   | <input type="checkbox"/> Keyboard shortcut editor               |
| <input type="checkbox"/> Launch QTConsole                      | <input type="checkbox"/> Limit Output                            | <input type="checkbox"/> Live Markdown Preview                     | <input type="checkbox"/> Load TeX macros                        |
| <input type="checkbox"/> Move selected cells                   | <input type="checkbox"/> Navigation-Hotkeys                      | <input checked="" type="checkbox"/> Nbextensions dashboard tab     | <input checked="" type="checkbox"/> Nbextensions edit menu item |
| <input type="checkbox"/> nbTranslate                           | <input type="checkbox"/> Notify                                  | <input type="checkbox"/> Printview                                 | <input type="checkbox"/> Python Markdown                        |
| <input type="checkbox"/> Rubberband                            | <input type="checkbox"/> Ruler                                   | <input type="checkbox"/> Ruler in Editor                           | <input type="checkbox"/> Runtools                               |
| <input type="checkbox"/> Scratchpad                            | <input type="checkbox"/> ScrollDown                              | <input type="checkbox"/> Select CodeMirror Keymap                  | <input type="checkbox"/> SKILL Syntax                           |
| <input type="checkbox"/> Skip-Traceback                        | <input type="checkbox"/> Snippets                                | <input type="checkbox"/> Snippets Menu                             | <input type="checkbox"/> spellchecker                           |
| <input type="checkbox"/> Split Cells Notebook                  | <input type="checkbox"/> Table of Contents (2)                   | <input type="checkbox"/> table_beautifier                          | <input type="checkbox"/> Toggle all line numbers                |
| <input type="checkbox"/> Tree Filter                           | <input type="checkbox"/> Variable Inspector                      | <input type="checkbox"/> zenmode                                   |   |

## zenmode

A little extension to give Zenmode functionality to the IPython notebook

Imagem: Aba para o nbextensions no Jupyter

---

# nbextensions

Ele não vem de padrão na imagem do Docker, podemos instalar com:

## **pip**

```
pip3 install jupyter_contrib_nbextensions
```

```
jupyter contrib nbextension install --user
```

## **Conda**

```
conda install -c conda-forge jupyter_contrib_nbextensions
```

---

---

---

## **nbconvert**

Como podemos observar o Jupyter tem uma habilidade muito boa quando se trata de interagir com diferente linguagens de marcação e afins. Dessa forma, é fácil imaginar que existem opções bem simples de convertemos nossos cadernos para formatos como PDF, LaTeX, HTML e assim por diante.

---

---

# **nbconvert**

O comando base para utilizar do nbconvert é:

```
jupyter nbconvert SEUNOTEBOOK.ipynb --to FORMATO
```

---

---

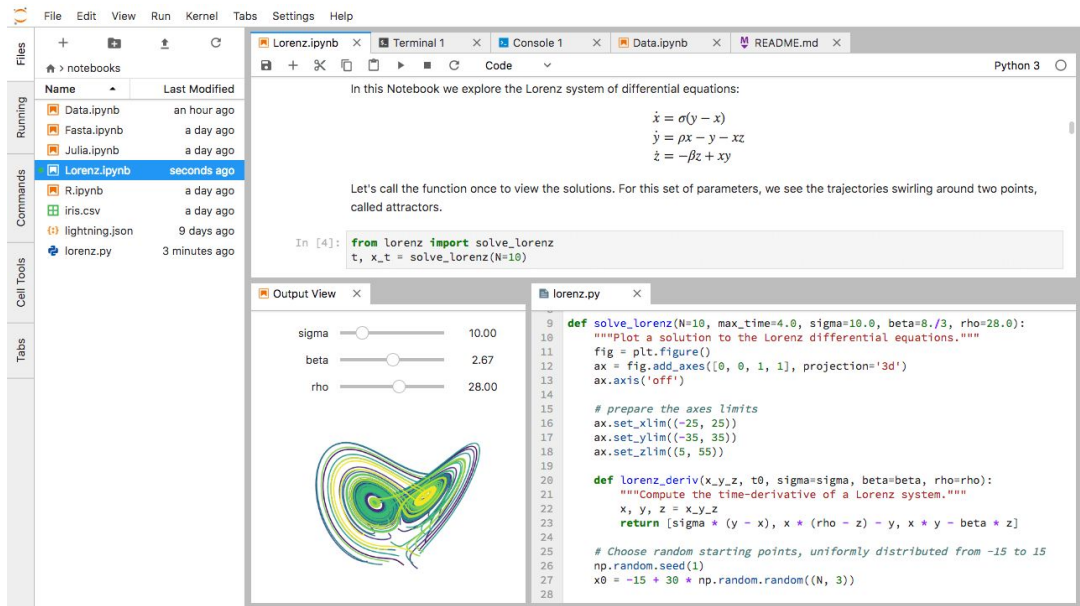
---

# JupyterLab

O JupyterLab é a próxima geração do Jupyter Notebook. Seu objetivo é resolver muitos problemas de usabilidade do Notebook, e expande muito seu escopo.

Além de fornecer uma interface melhorada aos notebooks existentes, o JupyterLab também traz dentro da mesma interface um navegador de arquivos, consoles, terminais, editores de texto, editores Markdown, editores CSV, editores JSON, mapas interativos, widgets e assim por diante.

---



Teste o JupyterLab no [Binder](#)

---

```
1 version: "3"
2 services:
3   datascience-notebook:
4     image: jupyter/datascience-notebook
5     volumes:
6       - /SEU/DIRETORIO/AQUI:/home/jovyan/work
7     environment:
8       GRANT_SUDO: "yes"
9       JUPYTER_ENABLE_LAB: "yes"
10    user: root
11    ports:
12      - 8888:8888
13    container_name: datascience-notebook-container
```

Imagem: Ativando o JupyterLab no Docker

---

---

# Resumo sobre usar o Jupyter

Há como usar o Jupyter em servidores que hospedam o serviço, como o Colab e o Binder.

Para instalar localmente, existem diversas opções. O mais recomendado para conseguir aproveitar das visualizações de dados que podem ser geradas é utilizar do Jupyter Notebook clássico ou uma IDE de sua escolha com um bom suporte para elas.

---



---

# Referências

[1] <https://www.nature.com/articles/d41586-018-07196-1>

[2] <https://jupyter.org/about>

[3] <https://www.datacamp.com/community/blog/ipython-jupyter>

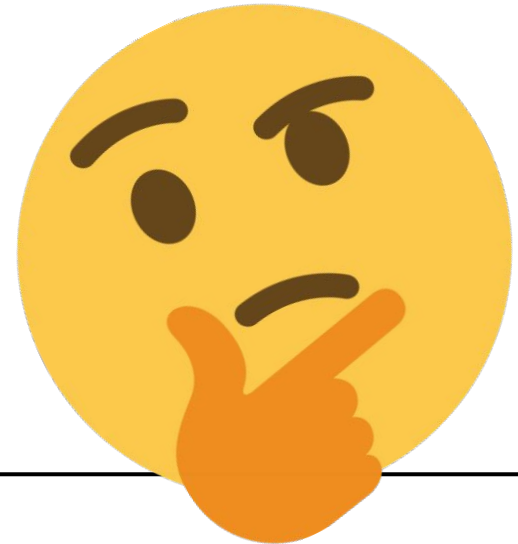
[4]

<https://blog.jetbrains.com/datalore/2020/12/17/we-downloaded-10-000-000-jupyter-notebooks-from-github-this-is-what-we-learned/>

---

---

# Dúvidas?





Obrigado pela presença!

Espero que tenham gostado :)

