



Visualización avanzada

Entornos Python

Python 3: instalación de paquetes

Entornos virtuales

Jupyter notebooks

Librerías python para visualización

Una gramática de los gráficos

Herramientas de color

Herramientas comerciales



Entornos Python 3

Código y datos en este repo:

https://github.com/pedroconcejero/ucm_master_2023_PYTHON

Instalación de paquetes en entornos python y entornos virtuales:

<https://packaging.python.org/en/latest/guides/installing-using-pip-and-virtual-environments/>



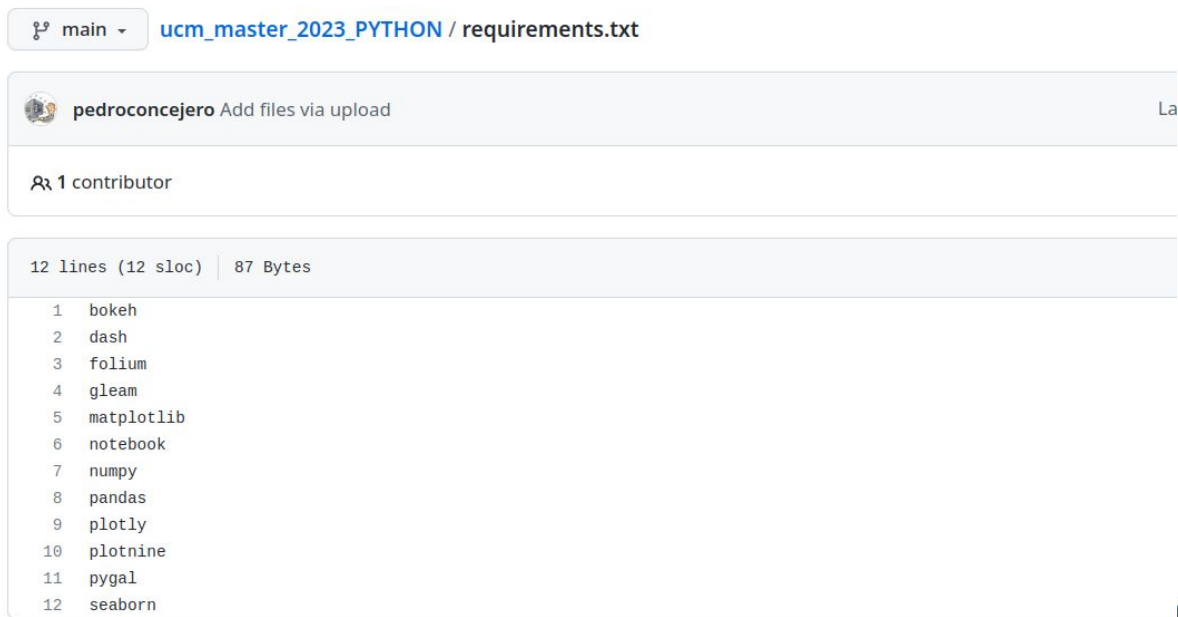
Entornos Python 3

Instalación de paquetes necesarios para el módulo desde requirements.txt

https://github.com/pedroconcejero/ucm_master_2023_PYTHON/blob/main/requirements.txt

(una vez descargado repo)

pip install -r requirements.txt



The screenshot shows the GitHub interface for the file `requirements.txt` in the repository `ucm_master_2023_PYTHON`. The file is located in the `main` branch. The interface includes a header with the repository name and file path, a section for contributors (showing 1 contributor), and a code viewer showing the contents of the file. The code lists 12 packages: bokeh, dash, folium, gleam, matplotlib, notebook, numpy, pandas, plotly, plotnine, pygal, and seaborn.

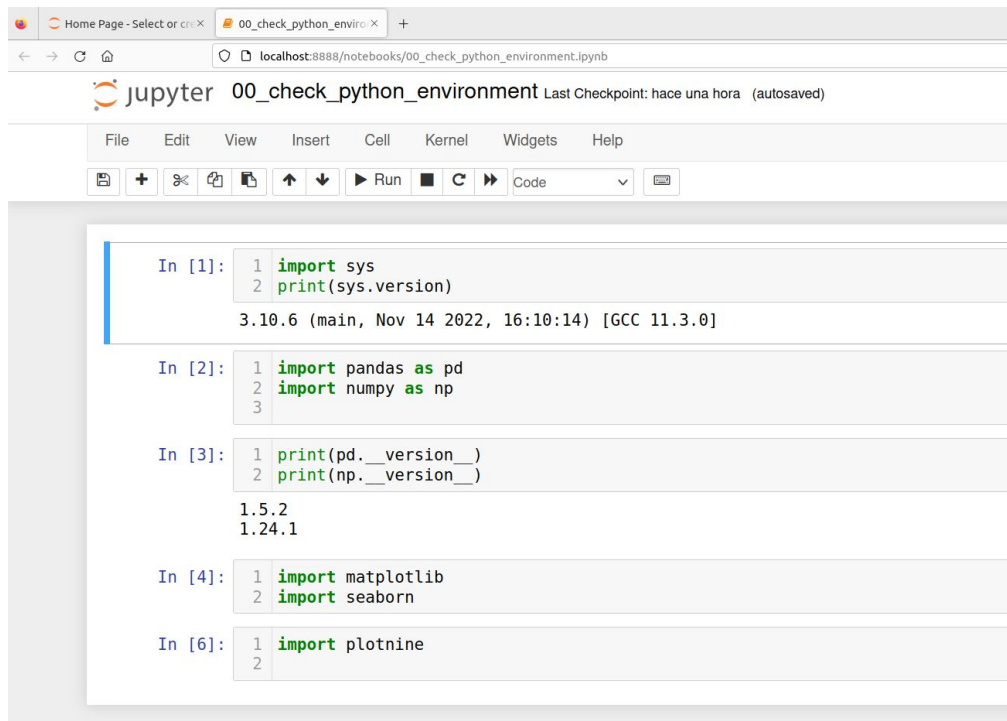
```
12 lines (12 sloc) | 87 Bytes
```

```
1 bokeh
2 dash
3 folium
4 gleam
5 matplotlib
6 notebook
7 numpy
8 pandas
9 plotly
10 plotnine
11 pygal
12 seaborn
```

Jupyter notebooks

Por ejemplo:

https://github.com/pedroconcejero/ucm_master_2023_PYTHON/blob/main/00_check_python_environment.ipynb



The screenshot displays a Jupyter Notebook interface in a web browser. The browser's address bar shows the URL `localhost:8888/notebooks/00_check_python_environment.ipynb`. The Jupyter logo and the notebook title `00_check_python_environment` are visible, along with a status message: `Last Checkpoint: hace una hora (autosaved)`. The interface includes a menu bar with `File`, `Edit`, `View`, `Insert`, `Cell`, `Kernel`, `Widgets`, and `Help`. Below the menu is a toolbar with icons for file operations, running the cell, and other functions. The notebook contains six code cells, each starting with `In [n]:`. The first cell imports `sys` and prints its version. The second cell imports `pandas` as `pd` and `numpy` as `np`. The third cell prints the versions of `pd` and `np`. The fourth cell imports `matplotlib` and `seaborn`. The fifth cell imports `plotnine`.

```
In [1]: 1 import sys
        2 print(sys.version)

3.10.6 (main, Nov 14 2022, 16:10:14) [GCC 11.3.0]

In [2]: 1 import pandas as pd
        2 import numpy as np
        3

In [3]: 1 print(pd.__version__)
        2 print(np.__version__)

1.5.2
1.24.1

In [4]: 1 import matplotlib
        2 import seaborn

In [6]: 1 import plotnine
        2
```



El entorno python para visualización de datos

<https://learnpython.com/blog/python-libraries-data-visualization/>

Podemos caracterizar las librerías de visualización de datos utilizando las siguientes dimensiones o criterios:

1. Interactividad: si la visualización generada permite interacción (o por el contrario es estrictamente estática)
2. Sintaxis: qué nivel de control permite la librería y si sigue un paradigma o teoría. Esto también tiene que ver con la *consistencia* de la sintaxis.
3. Fortalezas y documentación de la librería: para qué casos (ejemplos) es buena -está específicamente diseñada- la librería

La tabla siguiente pretende clasificar un conjunto bien establecido de librerías python para visualización:



El entorno python

<https://learnpython.com/blog/python-libraries-data-visualization/>

Library	Interactive Features	Syntax	Main Strength and Use Case
Matplotlib	Limited	Low-level	Highly customized plots
seaborn	Limited (via Matplotlib)	High-level	Fast, presentable reports
Bokeh	Yes	High- and low-level, influenced by grammar of graphics	Interactive visualization of big data sets
Altair	Yes	High level, declarative, follows grammar of graphics	Data exploration, and interactive reports
Plotly	Yes	High- and low-level	Commercial applications and dashboards



El entorno python

<https://www.geeksforgeeks.org/data-visualization-with-python/>

<https://mode.com/blog/python-data-visualization-libraries/>

(ver lista referencias adicionales al final)

<https://www.simplilearn.com/tutorials/python-tutorial/data-visualization-in-python>

<https://www.kaggle.com/code/benhamner/python-data-visualizations>

<https://learnpython.com/blog/python-libraries-data-visualization/>

<https://gilberttanner.com/blog/introduction-to-data-visualization-inpython/>

<https://www.projectpro.io/article/python-data-visualization-libraries/543>



ggplot2: la gramática de los gráficos

Para conocer sus orígenes:

[https://byrneslab.net/classes/biol607/readings/wickham_layered-grammar.p
df](https://byrneslab.net/classes/biol607/readings/wickham_layered-grammar.pdf)

Tiene como objetivo estandarizar muchos de los detalles que hacen un lío desarrollar un gráfico (como dibujar leyendas) así como proporcionar un **modelo potente de gráficos complejos basado en la superposición de capas.** Dedicaremos una lección completa a este paquete esencial.



Herramientas comerciales

Entorno python es opensource

- aunque no siempre quiere decir que sea gratuito

Hay (cómo no) herramientas comerciales –propietarias
–aunque no siempre requieran la compra

Centenares de “rankings” (y de “popups”)

<https://www.guru99.com/best-data-visualization-tools.html>

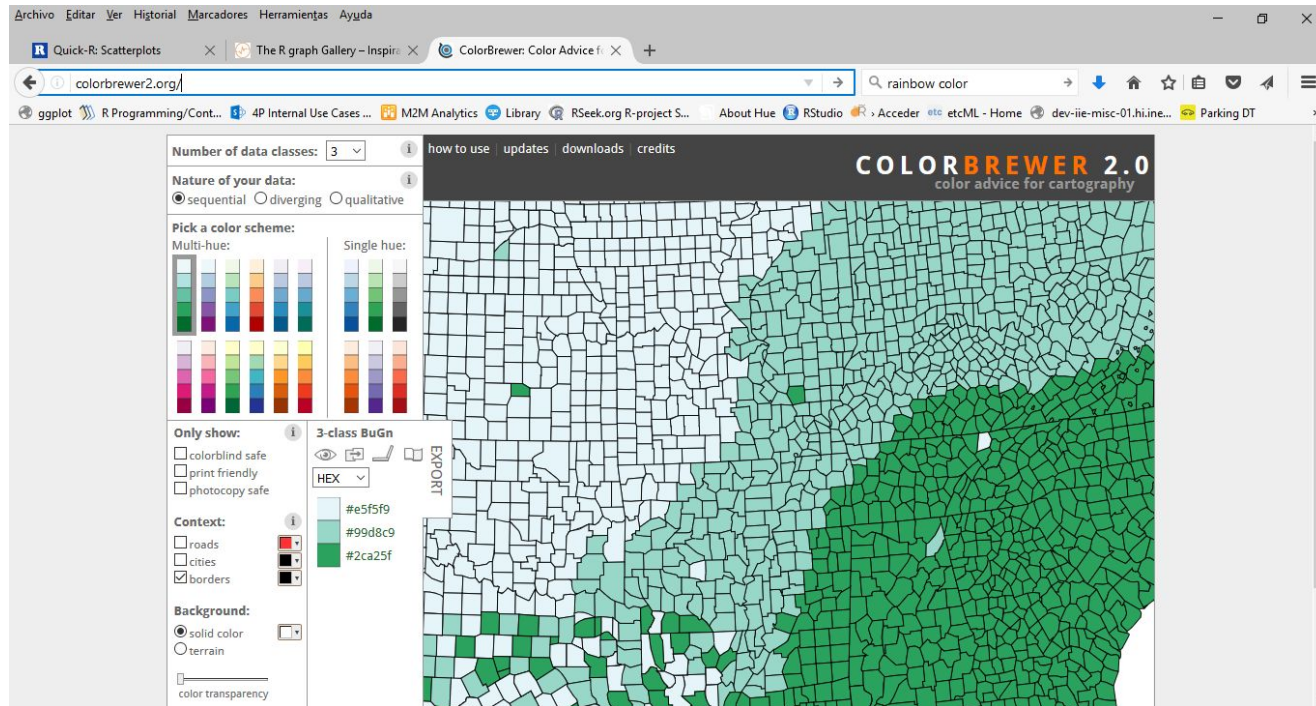
<https://financesonline.com/data-visualization/>

<https://www.predictiveanalyticstoday.com/top-data-visualization-software/>



Herramientas para el color

<http://colorbrewer2.org/>



¡Gracias!

Pedro Concejero

pedro.concejerocezoz@gmail.com