

Aprendiendo Python para Análisis Estadísticos

Juan C. Correa, Ph.D.

j.correa.n@gmail.com

<https://correajc.com/>



Objetivo de esta charla

Introducir los aspectos preliminares de Python para implementar un análisis de regresión simple y múltiple (sin conocer las otras funcionalidades de Python)



Agenda

- 1 Preliminares
 - 2 Trabajando con Jupyter Notebooks
 - 3 Trabajando con Spyder
 - 4 Recursos Bibliográficos
 - 5 Regresión Simple en Python
 - 6 Regresión Múltiple en Python
- References





El primer paso es descargar anaconda navigator

<https://www.anaconda.com/products/individual>

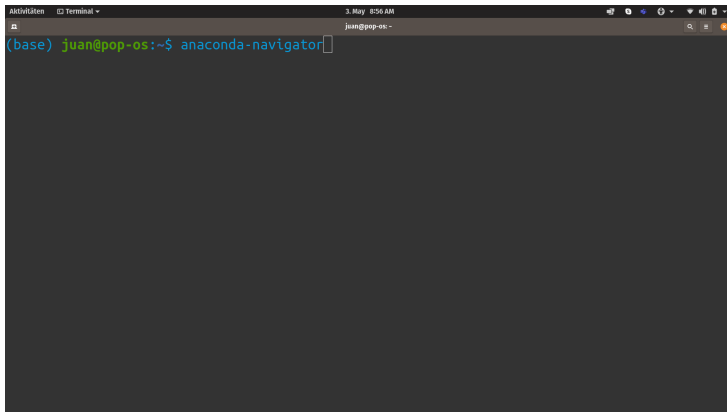


Anaconda gestiona múltiples versiones de Python sobre la máquina, además de ofrecer una gran colección de librerías comúnmente usadas en data science. Por eso es más práctico trabajar en Python a través de Anaconda en lugar de hacerlo con otras herramientas.



Preliminares

Para ingresar a Anaconda, se abre la terminal del sistema operativo y se escribe `anaconda-navigator` + enter

A screenshot of a terminal window. The window has a title bar with 'Aktivitäten' and 'Terminal'. The top status bar shows '3. May 8:56 AM' and 'juan@pop-os: ~'. The terminal content shows a prompt '(base) juan@pop-os:~\$' followed by the command 'anaconda-navigator' and a cursor. The background of the terminal is dark gray.

```
(base) juan@pop-os:~$ anaconda-navigator
```

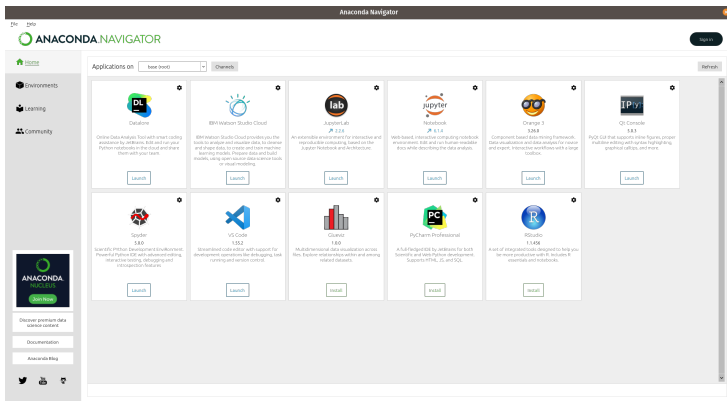


Los paquetes incluidos en Anaconda son:

- **NumPy** Computación Numérica sobre arrays n-dimensionales
- **SciPy** Computación Científica
- **Matplotlib** Visualización de datos en 2D
- **Pandas** Estructuras de datos y análisis de datos estructurados
- **Seaborn** Visualización de datos
- **Bokeh** Visualización Web Interactiva
- **Scikit-Learn** Machine learning y data mining
- **NLKT** Procesamiento de Lenguaje Natural
- **Jupyter Notebook** App Web para crear cuadernos reproducibles
- **R essentials**

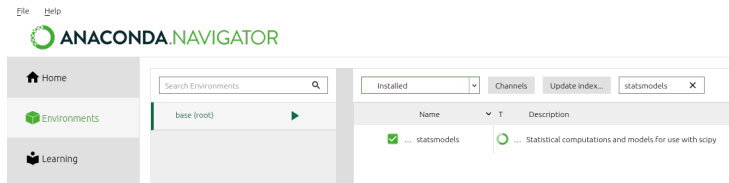


Preliminares



Interfaz gráfica de Anaconda

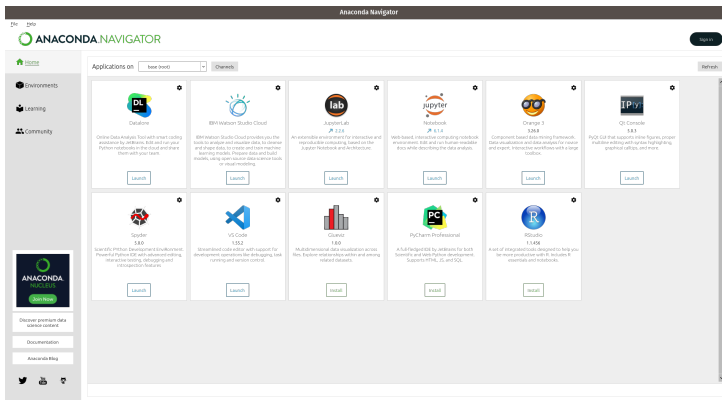




Para instalar nuevas librerías dentro de Anaconda, basta con hacer clic en “Environments” y luego seleccionar el menú **Not Installed** para escribir a mano derecho el nombre específico de la librería que se quiere instalar.



Trabajando con Jupyter Notebooks



Clic en **Launch** dentro del cuadro Jupyter Notebook



Trabajando con Jupyter Notebooks

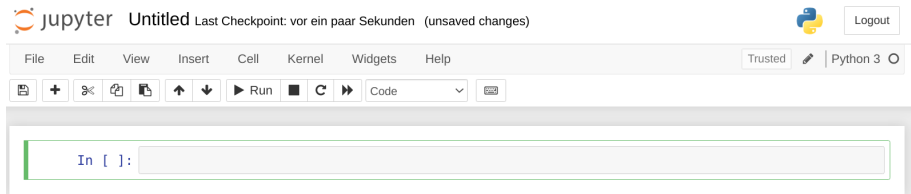


The screenshot shows the JupyterLab web interface. At the top left is the Jupyter logo. On the right are 'Quit' and 'Logout' buttons. Below the logo are tabs for 'Files', 'Running', and 'Clusters'. A message says 'Select items to perform actions on them.' Below this is a file browser showing the 'Documents' directory with files like 'beamer', 'Beamer', and 'Books'. On the right, there are 'Upload' and 'New' buttons. The 'New' button's dropdown menu is open, showing options: 'Notebook: Python 3' (highlighted with a blue box), 'Text File', 'Folder', and 'Terminal'.

Clic en **New** y luego en **Python 3**



Trabajando con Jupyter Notebooks



De manera predeterminada, los archivos creados en jupyter notebook tienen el nombre **Untitled**. Debemos cambiarle el nombre haciendo clic sobre Untitled.



Trabajando con Jupyter Notebooks

Rename Notebook

Enter a new notebook name:


RegMult

CancelRename

Acá vamos a poner el nombre a nuestro primer Jupyter Notebook como **RegMult**.



Trabajando con Jupyter Notebooks

jupyter RegMult Last Checkpoint: vor 21 Minuten (autosaved)  Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

In [1]: `import pandas as pd`

In [2]: `datos = pd.read_csv("/home/juan/cars.csv")`

In [3]: `datos.head()`

Out[3]:

	Car	Model	Volume	Weight	CO2
0	Toyoty	Aygo	1000	790	99
1	Mitsubishi	Space Star	1200	1160	95
2	Skoda	Citigo	1000	929	95
3	Fiat	500	900	865	90
4	Mini	Cooper	1500	1140	105

In [4]: `X = datos[['Weight', 'Volume']]`
`y = datos['CO2']`

In [5]: `import statsmodels.api as sm`

In [6]: `model = sm.OLS(y, X).fit()`

In [7]: `model.summary()`

Out[7]:

OLS Regression Results

Dep. Variable:	CO2	R-squared (uncentered):	0.977
Model:	OLS	Adj. R-squared (uncentered):	0.975
Method:	Least Squares	F-statistic:	714.9
Date:	Mon, 03 May 2021	Prob (F-statistic):	1.67e-28







Trabajando con Jupyter Notebooks

Jupyter RegMult Last Checkpoint: vor 3 Stunden (autosaved)

File Edit View Insert Cell Kernel Widgets Help

New Notebook
Open...
Make a Copy...
Save as...
Rename...
Save and Checkpoint **Ctrl-S**
Revert to Checkpoint
Print Preview...
Download as
Trusted Notebook
Close and Halt

Run     Code

Model	Volume	Weight	CO2
Aygo	1000	790	99
Space Star	1200	1160	95
Citigo	1000	929	95
500	900	865	90
Cooper	1500	1140	105

models.api as sm

results.sum
sm.graphics
plt.show()

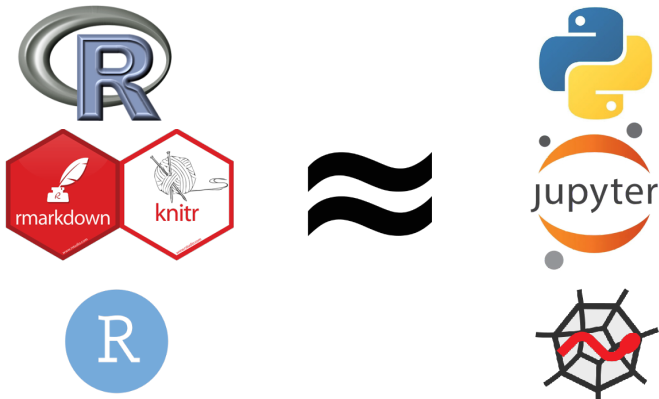
TypeError
<ipython-in
----> 1 fig
2 res

AsciiDoc (.asciidoc)
HTML (.html)
LaTeX (.tex)
Markdown (.md)
Notebook (.ipynb)
PDF via LaTeX (.pdf)
PDF via pypeteer (.html)
RevealJS Slides (.slides.html)

Python (.py)



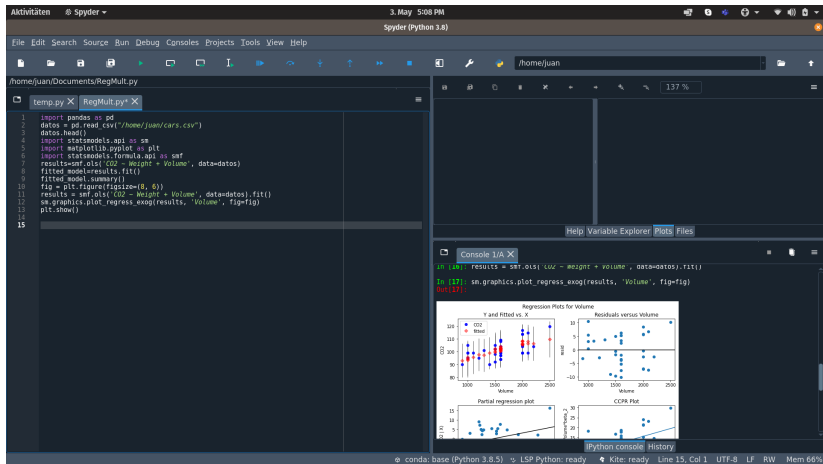
Trabajando con Spyder



La apariencia de Spyder es muy similar a la de RStudio



Trabajando con Spyder



Trabajar en Spyder es como elaborar un script en R (archivo .R)



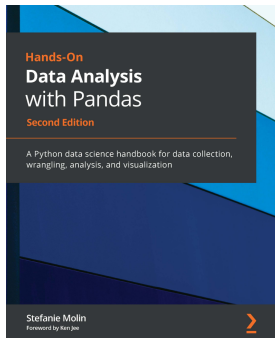
Trabajando con Spyder

Hay varias semejanzas entre Python y R, desde el punto de vista de las instrucciones (sintaxis) que debemos escribir para que el software realice lo que deseemos.

```
import pandas as pd  
library(readr)
```

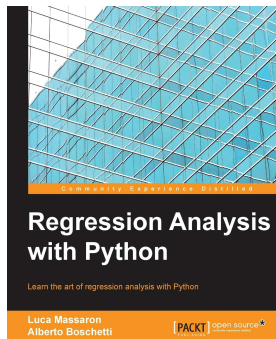
```
datos = pd.read_csv("/home/juan/cars.csv")  
readr::datos <- read_csv("/home/juan/cars.csv")
```





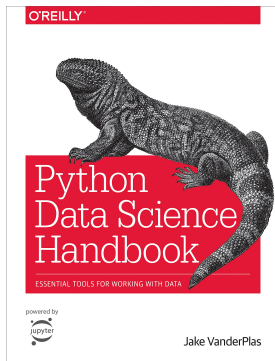
El libro de Molin (2021) es un buen recurso para aprender fundamentos de análisis de datos. Pero su aproximación dista mucho del tipo de estadística aplicada en psicología.





El libro de Massaron and Boschetti (2016) es un buen recurso para aprender a implementar análisis de regresiones. Esta aproximación es más orientada a machine learning, pero no hace una cobertura adecuada al problema del chequeo de supuestos en regresión.





El libro de VanderPlas (2017) aborda básicamente data manipulation, data visualization, y machine learning. Su covertedura deja por fuera un montón de estadística estándar en psicología u otras ciencias sociales (e.g., psicometría, modelos de ecuaciones estructurales, redes).





El paper de Seabold and Perktold (2010) es probablemente, a la fecha, el recurso más orientado a estadística que puede encontrarse en Python. Su documentación online está disponible en

<https://www.statsmodels.org/>



Regresión con Mínimos Cuadrados Ordinarios

```
import numpy as np
import statsmodels.api as sm
spector_data = sm.datasets.spector.load(as_pandas=False)
spector_data.exog = sm.add_constant(spector_data.exog, prepend=False)
mod = sm.OLS(spector_data.endog, spector_data.exog)
res = mod.fit()
print(res.summary())
```



Regresión con Cuadrados Ordinarios Ponderados

```
import numpy as np
import statsmodels.api as sm
spector_data = sm.datasets.spector.load(as_pandas=False)
spector_data.exog = sm.add_constant(spector_data.exog, prepend=False)
mod = sm.WLS(spector_data.endog, spector_data.exog)
res = mod.fit()
print(res.summary())
```



Regresión Simple en Python

```
In [11]: print(res.summary())
```

WLS Regression Results

```
=====
Dep. Variable:                y      R-squared:                0.416
Model:                        WLS    Adj. R-squared:            0.353
Method:                       Least Squares    F-statistic:            6.646
Date:                         Mon, 03 May 2021    Prob (F-statistic):      0.00157
Time:                         19:15:39    Log-Likelihood:         -12.978
No. Observations:             32    AIC:                    33.96
Df Residuals:                 28    BIC:                    39.82
Df Model:                     3
Covariance Type:              nonrobust
=====
```

	coef	std err	t	P> t	[0.025	0.975]
x1	0.4639	0.162	2.864	0.008	0.132	0.796
x2	0.0105	0.019	0.539	0.594	-0.029	0.050
x3	0.3786	0.139	2.720	0.011	0.093	0.664
const	-1.4980	0.524	-2.859	0.008	-2.571	-0.425

```
=====
Omnibus:                      0.176    Durbin-Watson:          2.346
Prob(Omnibus):                0.916    Jarque-Bera (JB):       0.167
Skew:                         0.141    Prob(JB):               0.920
Kurtosis:                     2.786    Cond. No.                176.
=====
```



Regresión Múltiple en Python

```
import pandas as pd
datos = pd.read_csv("/home/juan/cars.csv")
import statsmodels.api as sm
import matplotlib.pyplot as plt
import statsmodels.formula.api as smf
results=smf.ols('CO2 ~ Weight + Volume', data=datos)
fitted_model=results.fit()
fitted_model.summary()
```



Regresión Múltiple en Python

OLS Regression Results

Dep. Variable:	C02	R-squared:	0.377
Model:	OLS	Adj. R-squared:	0.339
Method:	Least Squares	F-statistic:	9.966
Date:	Mon, 03 May 2021	Prob (F-statistic):	0.000411
Time:	19:19:18	Log-Likelihood:	-114.39
No. Observations:	36	AIC:	234.8
Df Residuals:	33	BIC:	239.5
Df Model:	2		
Covariance Type:	nonrobust		

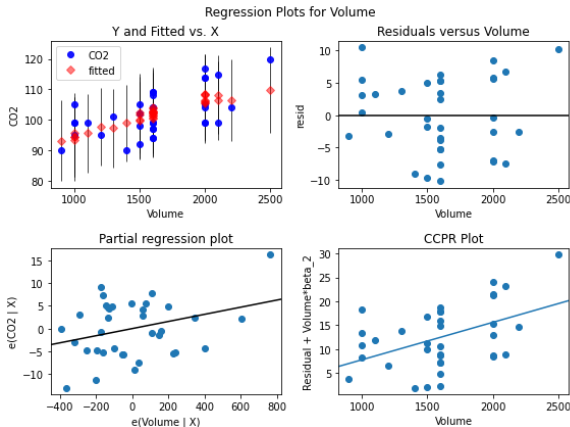
	coef	std err	t	P> t	[0.025	0.975]
Intercept	79.6947	5.564	14.322	0.000	68.374	91.016
Weight	0.0076	0.006	1.173	0.249	-0.006	0.021
Volume	0.0078	0.004	1.948	0.060	-0.000	0.016

Omnibus:	4.957	Durbin-Watson:	0.944
Prob(Omnibus):	0.084	Jarque-Bera (JB):	1.836
Skew:	-0.025	Prob(JB):	0.399
Kurtosis:	1.895	Cond. No.	1.16e+04



Regresión Múltiple en Python

```
fig = plt.figure(figsize=(8, 6))
results = smf.ols('CO2 ~ Weight + Volume', data=datos).fit()
sm.graphics.plot_regress_exog(results, 'Volume', fig=fig)
plt.show()
```



- Massaron, L., & Boschetti, A. (2016). *Regression Analysis with Python*. New York, USA: Pakt Publisher.
- Molin, S. (2021). *Data Analysis with Pandas* (2nd ed.). New York, USA: Pakt Publisher.
- Seabold, S., & Perktold, J. (2010). Statsmodels: Econometric and statistical modeling with python. In *Proceedings of the 9th python in science conference* (Vol. 57, p. 61).
- VanderPlas, J. (2017). *Python Data Science Handbook Essential Tools for Working with Data*. New York, USA: O'Reily.

