

Actividad 1

Física Computacional 1
Corral Valdez Jesus Giovanni
Departamento de Física
Universidad de Sonora

1 Introducción

La primera actividad consta de leer un capítulo del libro "Python for Data Analysis", de Wes McKinney

SECOND EDITION

Python for Data Analysis

*Data Wrangling with Pandas, NumPy,
and IPython*

Wes McKinney

2 Desarrollo

Se leyó el capítulo 2

CHAPTER 2

Python Language Basics, IPython, and Jupyter Notebooks

2.1 The Python Interpreter

Python es un lenguaje interpretado que puede ejecutarse sencillamente desde su propio interpretador, pero quienes hacen análisis de datos prefieren hacer uso de intérpretes como IPython o Jupyter Notebook.

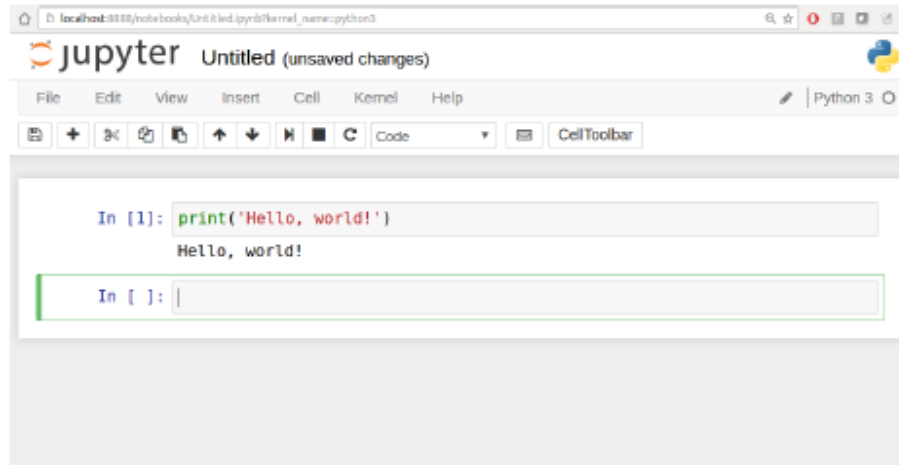
2.2 IPython Basics

Uno de los mayores componentes del proyecto Jupyter es el "notebook", un tipo de documento interactivo para código, texto, visualizaciones de data y otras cosas. Este interactúa con kernels. En muchas plataformas Jupyter se abre automáticamente en tu navegador web por default.



Puedes crear un nuevo notebook y empezar a escribir en las celdas vacías. Al momento de salvarlo se creará un archivo con la extensión `.ipynb`, el cual es un formato que contiene todo el contenido actual en el notebook. Estos pueden ser cargados y editados por otros usuarios de Jupyter.

Uno de las mejores implementaciones de los editores de Python es la "Tab



Completion", la cual al apretar la tecla Tab durante estas escribiendo una expresión se te darán recomendaciones y opciones para terminar ese comando.

```
In [3]: b = [1, 2, 3]

In [4]: b.<Tab>
b.append b.count b.insert b.reverse
b.clear b.extend b.pop b.sort
b.copy b.index b.remove
```

En Jupyter Notebook, es posible copiar y pegar código en cualquier celda de código y ejecutarlo. Otro dato útil es que conforme ha ido creciendo su popularidad y uso, se han creado cada vez mas "shortcuts" que nos facilitaran muchas acciones.

Una de las razones por la cual Jupyter es popular en analisis computacional es que se integra muy bien con la visualización y librerías de interfaz como matplotlib. Esta sección nos hablar de conceptos esenciales del lenguaje de

```
In [14]: %matplotlib inline
```

```
In [15]: import matplotlib.pyplot as plt  
plt.plot(np.random.randn(50).cumsum())
```

```
Out[15]: [<matplotlib.lines.Line2D at 0x7f828f0497f0>]
```

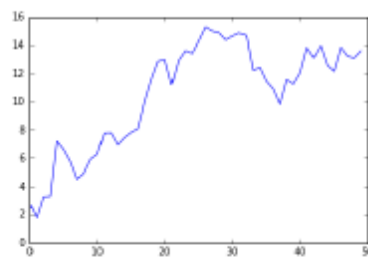


Fig 1.1

2.3 Python Language Basics

Python, el cual es distinguido por su énfasis en lectura, simplicidad y claridad. Python utiliza espacios en blanco para estructurar el código, y en cambio los "punto y coma" son usados para separar múltiples sentencias pero es preferible no hacerlo ya que generalmente hace menos entendible el código.

Cada cosa en Python es considerado un "object" y tiene asociado un propio "type"(como string o function). Es posible hacer comentarios en el código que no afecten la ejecución de la celda, todo lo que se escriba precedido de un "hashtag":

```
results = []  
for line in file_handle:  
    # keep the empty lines for now  
    # if len(line) == 0:  
    #     continue  
    results.append(line.replace('foo', 'bar'))
```

Puedes llamar funciones usando paréntesis, y casi cada objeto en Python tiene asociado funciones, conocidos como "métodos" los cuales tienen acceso a contenidos internos de estos objetos. Se llaman usando el siguiente código:

Cuando asignas una variable, se esta creando una referencia al objeto al lado

```
obj.some_method(x, y, z)
```

derecho del signo de igual. Por ejemplo si escribimos "a = b", estamos haciendo que a y b sean el mismo objeto.

Las operaciones aritmeticas son muy parecidas a las que uno pensaría, las cuales son las siguientes:

Operation	Description
a + b	Add a and b
a - b	Subtract b from a
a * b	Multiply a by b
a / b	Divide a by b
a // b	Floor-divide a by b, dropping any fractional remainder
a ** b	Raise a to the b power
a & b	True if both a and b are True; for integers, take the bitwise AND
a b	True if either a or b is True; for integers, take the bitwise OR
a ^ b	For booleans, True if a or b is True, but not both; for integers, take the bitwise EXCLUSIVE-OR

3 Conclusiones

Python es un lenguaje muy sencillo que la verdad no hace falta explicar casi cosas sobre el, tiene los clasicos "for, while, if", etc que tiene cualquier lenguaje, y los tipos numéricos y de variables basicos. Ademas sus interpretadores como Jupyter Notebook hacen aun mas accesible el análisis de datos y la programación con este lenguaje.