

Machine Learning in Python

Reconocimiento de Patrones
IIE - FING - UdelaR

Agosto de 2018

Bibliotecas Científicas en Python

- ▶ **Python:** lenguaje de programación
- ▶ **IPython:** consola interactiva (<http://ipython.org>)
- ▶ **Numpy:** arreglos numéricos y rutinas para manipularlos (<http://www.numpy.org>)
- ▶ **Matplotlib:** gráficas, visualización 2D (<http://matplotlib.org>)
- ▶ **Scipy:** métodos de optimización, regresión, interpolación (<http://www.scipy.org>)
- ▶ **scikit-learn:** machine learning (<http://scikit-learn.org>)

Instalación

- ▶ Anaconda (<https://docs.continuum.io/anaconda/>) es una distribución de python que incluye los paquetes necesarios
- ▶ Disponible para Linux, Windows, Mac
- ▶ IDE incluido: Spyder.

The screenshot displays the Spyder Python IDE interface. The main editor window shows a Python script named 'interpolation.py' with the following code:

```
4 from the SciPy Cookbook
5 """
6
7 from numpy import arange, cos, linspace, pi, sin, random
8 from scipy.interpolate import splprep, splev
9
10 # make ascending spiral in 3-space
11 t=linspace(0,1.75*2*pi,100)
12
13 x = sin(t)
14 y = cos(t)
15 z = t
16
17 # NN add noise
18 x+= random.normal(scale=0.1, size=x.shape)
19 y+= random.normal(scale=0.1, size=y.shape)
20 z+= random.normal(scale=0.1, size=z.shape)
21
22 # NN spline parameters
23 s=3.0 # smoothness parameter
24 k=2 # spline order
25 nest=-1 # estimate of number of knots needed (-1 = maximal)
26
27 # NN find the knot points
28 tckp,u = splprep([x,y,z],s=s,k=k,nest=-1)
29
30 # NN evaluate spline, including interpolated points
31 xnew,ynew,znew = splev(linspace(0,1,400),tckp)
32
33 import pylab
```

The right-hand side of the IDE is divided into several panels. The 'Object Inspector' panel shows the definition of the `mean` function from `numpy.core.fromnumeric`. The 'Python console' panel shows the execution of the script, resulting in two plots. The first plot shows the 'data' points (blue dots) and a 'fit' (red line) for the x-y plane. The second plot shows the 'data' points (blue dots) and a 'fit' (red line) for the x-z plane.

IPython

- ▶ consola interactiva que es parte de la distribución Anaconda y además está integrada a Spyder
- ▶ algunas características útiles:
 - ▶ histórico de comandos
 - ▶ "funciones mágicas"
 - ▶ quickref
 - ▶ run
 - ▶ cd
 - ▶ whos
 - ▶ reset
 - ▶ timeit
 - ▶ ...
 - ▶ sugerencias con tab

Python

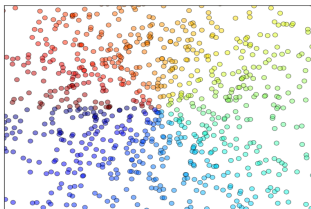
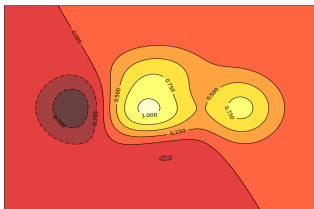
- ▶ Lenguaje de programación
 - ▶ interpretado
 - ▶ orientado a objetos
 - ▶ código de lectura sencilla
 - ▶ buena interaz con otros lenguajes (C,C++)
- ▶ Estructuras básicas
 - ▶ tipos numéricos (Integer, Float, Complex, Boolean)
 - ▶ contenedores (Listas, Strings, Sets, Diccionarios, Tuplas)
- ▶ Control de flujo
 - ▶ if/elif/else
 - ▶ for/range
 - ▶ while/break/continue

Numpy

- ▶ Arreglos numpy
 - ▶ contenedores multidimensionales
 - ▶ diseñados para cálculos científicos
 - ▶ uso eficiente de memoria que permite cálculos mas rápidos
- ▶ Típicamente se usa para almacenar
 - ▶ señales de audio, imágenes
 - ▶ **características y etiquetas**
- ▶ Operaciones con arreglos
 - ▶ crearlos
 - ▶ generarlos, cargarlos desde archivo, guardarlos
 - ▶ operaciones aritméticas
 - ▶ indexado, redimensionamiento, etc

Matplotlib

- ▶ visualización 2D: `from matplotlib import pyplot as plt`
- ▶ pyplot prevé una interfaz que permite acceder a los métodos implementados en la biblioteca Matplotlib
- ▶ inspirada en Matlab
 - ▶ `figure()`, `subplot()`



Scipy

- ▶ Paquete que a su vez contiene varios sub paquetes especializados en tareas específicas de la programación científica
 - ▶ interpolación (`scipy.interpolate`)
 - ▶ integración (`scipy.integrate`)
 - ▶ transformada de Fourier (`scipy.fftpack`)
 - ▶ optimización (`scipy.optimize`)
 - ▶ estadística (`scipy.stats`)
 - ▶ procesamiento de señales (`scipy.signal`)
 - ▶ procesamiento de imágenes (`scipy.ndimage`)
 - ▶ álgebra lineal (**`linalg`**)
- ▶ Pensada para trabajar eficientemente con numpy arrays

Scikit-learn

scikit-learn algorithm cheat-sheet

