matplotlib

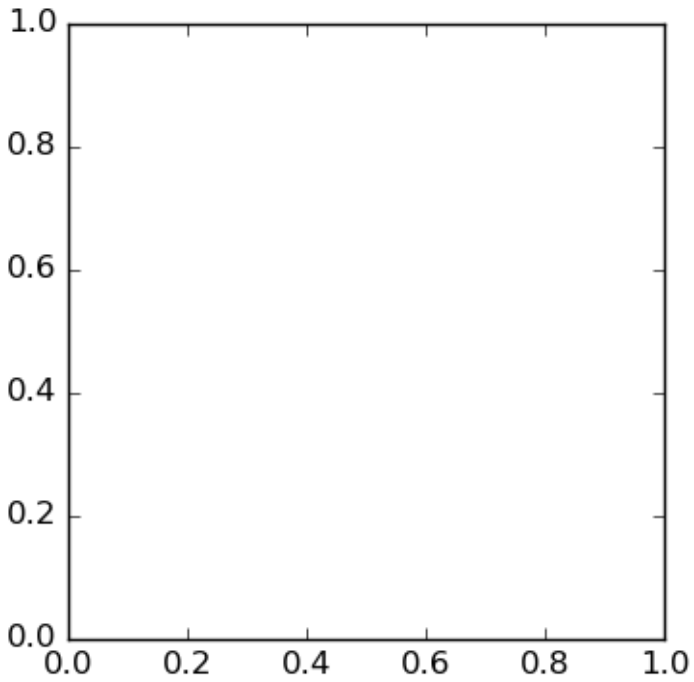
Ben Bolker

10:27 18 March 2015

- matplotlib cheat sheet
- matplotlib gallery

Basic setup

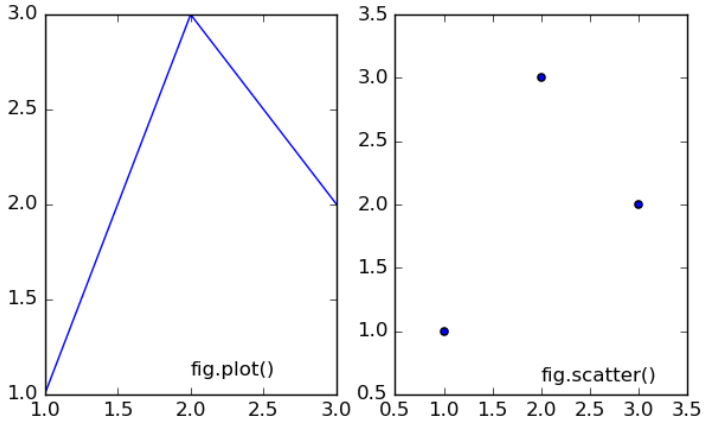
- create a figure: `fig=plt.figure()`
- can include figure size `figsize=(w,h)`, background/edge color, resolution (dpi=dots per inch)
- add a *subplot* (or "axes"): `ax = fig.add_subplot(1,1,1)` (rows,columns,which plot)
- now can show or save the figure: `fig.show()` or `fig.savefig("filename")`
- your operating system probably knows what to do if you click on the saved figure (or you can stick it in a Word document, etc.)



Basic plots

Basic things we can put on the plot: lines, scatter plots

Putting more than one thing on a plot



- You can do more than one `plot()` or `scatter()` on the same set of axes

Distinguish lines: * `color` * `marker` (`+`, `o`, `x`, ...) * `linewidth` * `linestyle` (`-`, `--`, `-.`, `None`, ...)

Decorating plots

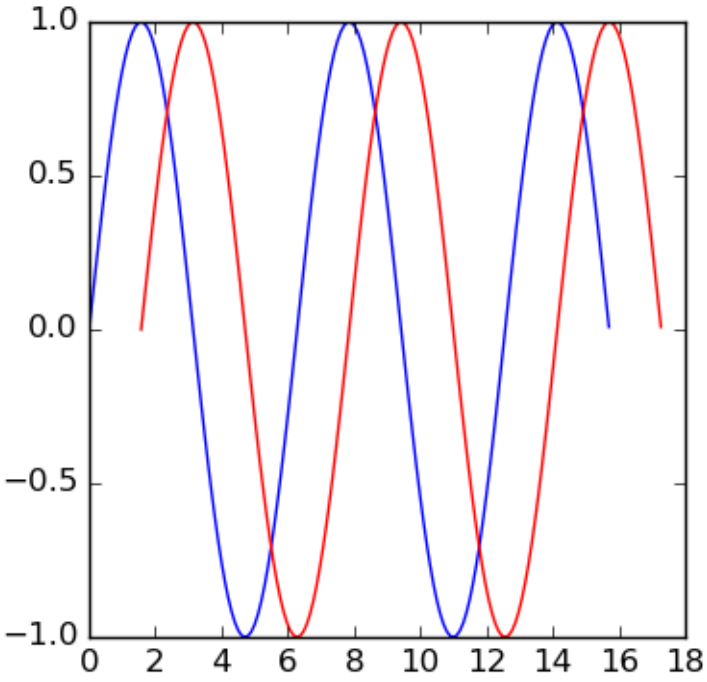
- titles (`ax.set_xlabel()`, `ax.set_ylabel()`)
- change limits
- title: `fig.suptitle()` (refers to figure, not individual axes)
- legend: need to label plotted stuff. e.g.

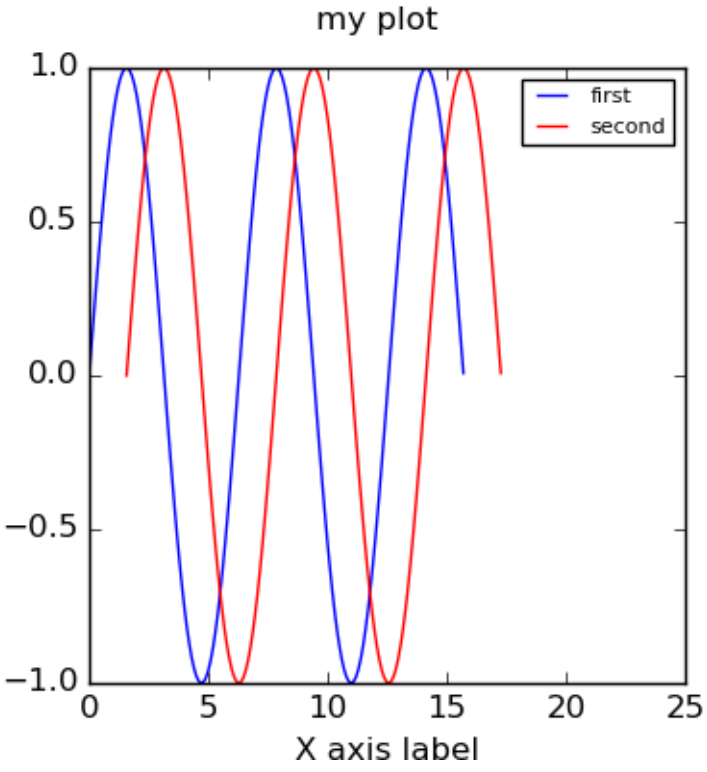
```
ax1.plot(x,y,label="first")
ax1.plot(x+np.pi/2,y,color="red",label="second")
ax1.set_xlim([0,25])
ax1.legend(fontsize=8)
fig.suptitle("my plot")
```

- Lorenz attractor example

```
import odesolve
import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D

def lorenz(time, state, params):
```


1.0
0.5
0.0
−0.5
−1.0
0
2
4
6
8
10
12
14
16
18


my plot
first
second
1.0
0.5
0.0
−0.5
−1.0
0
5
10
15
20
25
X axis label

```
    x, y, z = tuple(state)
    s, r, b = params
    x_dot = s*(y - x)
    y_dot = r*x - y - x*z
    z_dot = x*y - b*z
    return((x_dot, y_dot, z_dot))

tvec = np.arange(0,200,0.01)
lfit = odesolve.solveODE3(lorenz,(0.,1.,1.05),tvec,(10,28,2.667))
fig = plt.figure()
ax = fig.gca(projection='3d')

ax.plot(lfit[:,0], lfit[:,1], lfit[:,2])
ax.set_xlabel("X Axis")
ax.set_ylabel("Y Axis")
ax.set_zlabel("Z Axis")
ax.set_title("Lorenz Attractor")

fig.savefig("pix/lorenz.png")
```