

Lorenz attractor

November 7, 2017

```
In [18]: import numpy as np
import matplotlib.pyplot as plt
from scipy.integrate import odeint
from mpl_toolkits.mplot3d import Axes3D

In [70]: n = 1000
dt = 0.01
seq = np.empty( ((n+1),3) )
seq[0,0], seq[0,1], seq[0,2] = (1,1,1)

In [71]: def lorenz(x, y, z, a=10, b=28, c=8/3):
    dx = a*(y-x)
    dy = x*(b-z)-y
    dz = x*y-c*z
    return (dx, dy, dz)

In [72]: for i in range(n):
    dx, dy, dz = lorenz(seq[i,0], seq[i,1], seq[i,2])
    seq[i+1,0] = seq[i,0] + dt*dx #  $x + dt \cdot dx$ 
    seq[i+1,1] = seq[i,1] + dt*dy #  $y + dt \cdot dy$ 
    seq[i+1,2] = seq[i,2] + dt*dz #  $z + dt \cdot dz$ 

In [73]: plt.plot(seq[:,1], seq[:,2])
plt.show()
```

