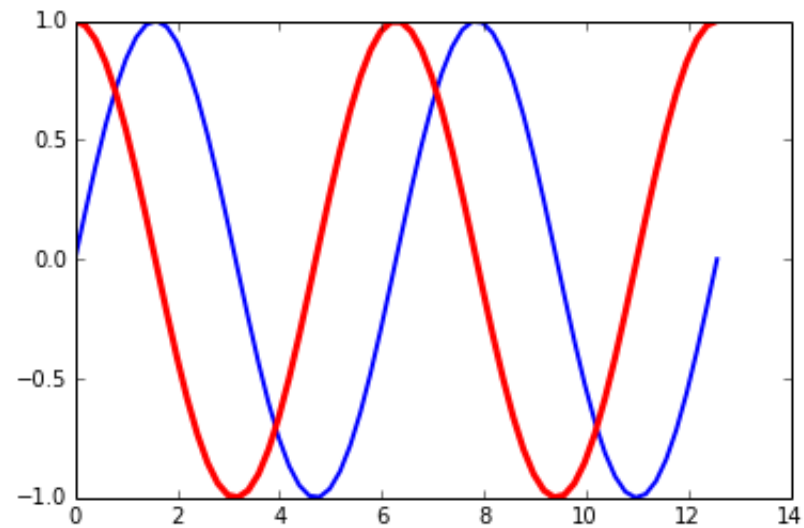


```
In [34]: %matplotlib inline
import numpy as np
import matplotlib.pyplot as plt

x = np.linspace(0, 4*np.pi, 64)
plt.plot(x, np.sin(x), color="blue", linewidth=2, linestyle="-")
plt.plot(x, np.cos(x), color="red", linewidth= 3, linestyle="-")
plt.show()
```



```
In [46]: %matplotlib qt
# this displays the figure in window outside
import matplotlib.pyplot as plt
```

```
x = np.linspace(0, 4*np.pi, 64)
plt.plot(x, np.sin(x), color="blue", linewidth=2, linestyle="-")
plt.plot(x, np.cos(x), color="red", linewidth= 3, linestyle="-")
plt.show()
```

```
In [50]: import matplotlib.pyplot as plt
import numpy as np
import mpl_toolkits.mplot3d.axes3d as axes3d

X, Y = np.meshgrid(np.arange(N), np.arange(N))
plt.scatter(X,Y)
plt.show()
```

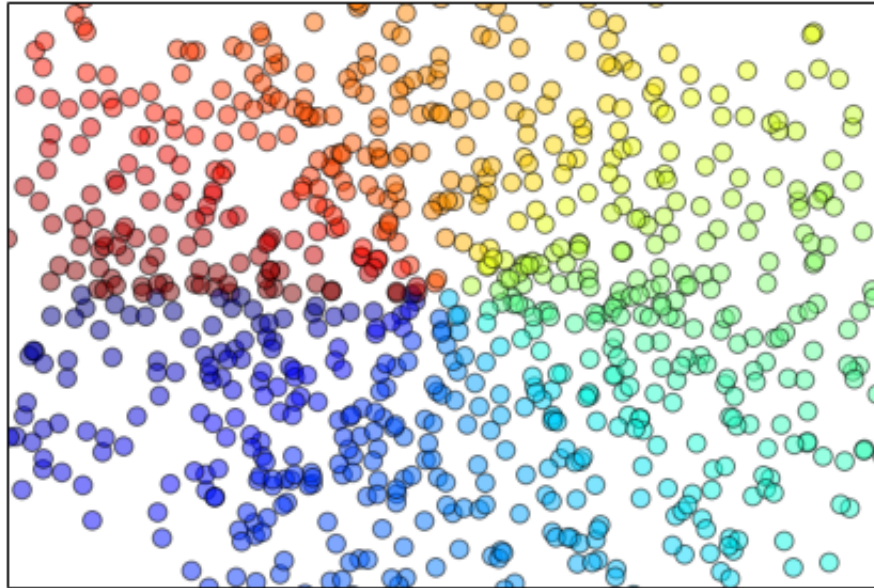
```
In [69]: import numpy as np

n = 1024
X = np.random.normal(0, 1, n)
Y = np.random.normal(0, 1, n)
T = np.arctan2(Y, X)

plt.axes([0.025, 0.025, 0.95, 0.95])
plt.scatter(X, Y, s=75, c=T, alpha=.5)

plt.xlim(-1.5, 1.5)
plt.xticks(())
plt.ylim(-1.5, 1.5)
plt.yticks(())
```

```
plt.show()
```

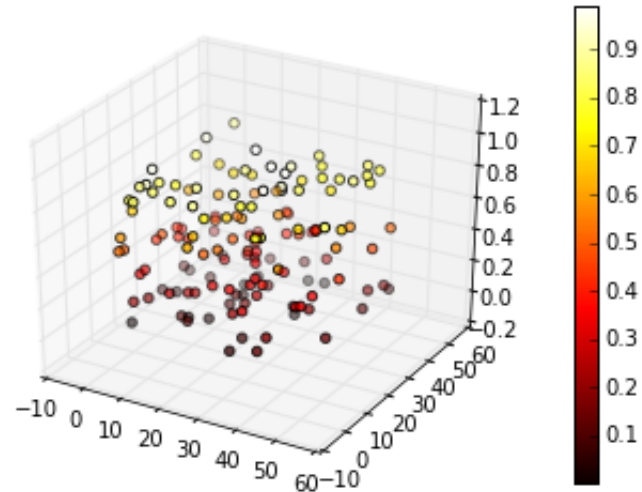


```
In [59]: import numpy as np
          from mpl_toolkits.mplot3d import Axes3D
          import matplotlib.pyplot as plt
          import random as ran

          fig = plt.figure()
          ax = fig.add_subplot(111, projection='3d')

          cmhot = plt.cm.get_cmap("hot")
          xs = [ran.random()*50 for n in range(0,150)]
          ys = [ran.random()*50 for n in range(0,150)]
          zs = [ran.random() for n in range(0,150)]
```

```
l = ax.scatter(xs, ys, zs, c=zs, cmap=cmhot)
fig.colorbar(l)
plt.show()
```



```
In [28]: import matplotlib.pyplot as plt
import numpy as np
import mpl_toolkits.mplot3d.axes3d as axes3d

fig, ax = plt.subplots(subplot_kw=dict(projection='3d'))
N = 128
X, Y = np.meshgrid(np.arange(N), np.arange(N))
heights = Y
ax.plot_surface(X, Y, heights, cmap=plt.get_cmap('jet'))
```

Out[28]: <mpl\_toolkits.mplot3d.art3d.Poly3DCollection at 0x5058f90>

```
In [29]: import matplotlib.pyplot as plt
```

```

import numpy as np
import mpl_toolkits.mplot3d.axes3d as axes3d

fig, ax = plt.subplots(subplot_kw=dict(projection='3d'))
N = 128
X, Y = np.meshgrid(np.arange(N), np.arange(N))
heights = Y**2 + X
ax.plot_surface(X, Y, heights, cmap=plt.get_cmap('jet'))

```

Out[29]: <mpl\_toolkits.mplot3d.art3d.Poly3DCollection at 0x4faf990>

```

In [39]: import matplotlib.pyplot as plt
import numpy as np
import mpl_toolkits.mplot3d.axes3d as axes3d

fig, ax = plt.subplots(subplot_kw=dict(projection='3d'))
N = 128
X, Y = np.meshgrid(np.arange(N), np.arange(N))
heights = np.sin(2*np.pi*np.sqrt(X**2+Y**2)/N)
ax.plot_surface(X, Y, heights, cmap=plt.get_cmap('jet'))

```

Out[39]: <mpl\_toolkits.mplot3d.art3d.Poly3DCollection at 0x5488290>

In [35]:

```

In [45]: ##matplotlib inline
%matplotlib qt

from mpl_toolkits.mplot3d import axes3d

```

```

import matplotlib.pyplot as plt, numpy as np
plt.clf()
fig = plt.figure(1)
ax = fig.gca(projection='3d')
X, Y, Z = axes3d.get_test_data(0.05)
ax.plot_surface(X, Y, Z, rstride=8, cstride=8, alpha=0.3)
cset = ax.contourf(X, Y, Z, zdir='z', offset=-100,
                   levels=np.linspace(-100,100,1200), cmap=plt.cm.jet)
cset = ax.contourf(X, Y, Z, zdir='x', offset=-40, cmap=plt.cm.jet)
cset = ax.contourf(X, Y, Z, zdir='y', offset=40, cmap=plt.cm.jet)
ax.set_xlabel('X')
ax.set_xlim(-40, 40)
ax.set_ylabel('Y')
ax.set_ylim(-40, 40)
ax.set_zlabel('Z')
ax.set_zlim(-100, 100)

```

Out[45]: (-100, 100)

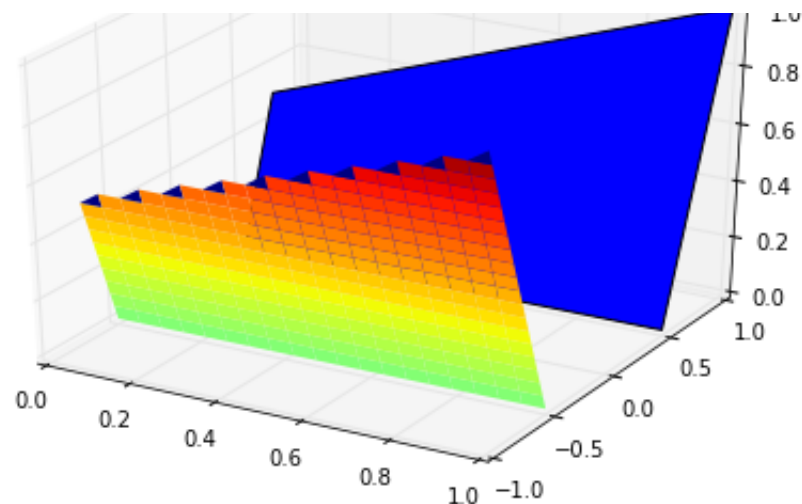
In [2]:

Out[2]: [<matplotlib.lines.Line2D at 0x5bc6c10>]

In [3]:

In [10]:





---

More info on IPython website (<http://ipython.org>). The code for this site

[Back to top](#)

(<https://github.com/ipython/nbviewer>) is licensed under BSD

(<https://github.com/ipython/nbviewer/blob/master/LICENSE.txt>). Some icons from Glyphicons Free

(<http://glyphicons.com>), built thanks to Twitter Bootstrap (<http://twitter.github.com/bootstrap/>)

This web site does not host notebooks, it only renders notebooks available on other websites. Thanks to all our contributors (<https://github.com/ipython/nbviewer/contributors>) and to Rackspace (<http://www.rackspace.com>) for hosting.

nbviewer version: e3a7f5b

(<https://github.com/ipython/nbviewer/commit/e3a7f5babbd25ba939ec88fd59162f59829bf7ca>) (Sun, 5 Jan 2014 11:24:48 -0800)