Let's learn the normal distribution! Note: You should check out the binomial distribution first.

We'll start with the definition of the PDF, we'll see how to create the distribution in python using scipy and numpy, and discuss some properties of the normal distribution.

The distribution is defined by the probability density function equation:

$$f(x, \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-1}{2z^2}}$$

Where:

$$z = \frac{(X - \mu)}{\sigma}$$

where: μ=mean , σ=standard deviation , π=3.14... , e=2.718... The total area bounded by curve of the probability density function equation and the X axis is 1; thus the area under the curve between two ordinates X=a and X=b, where a<b, represents the probability that X lies between a and b. This probability can be expressed as:

$$Pr(a < X < b)$$

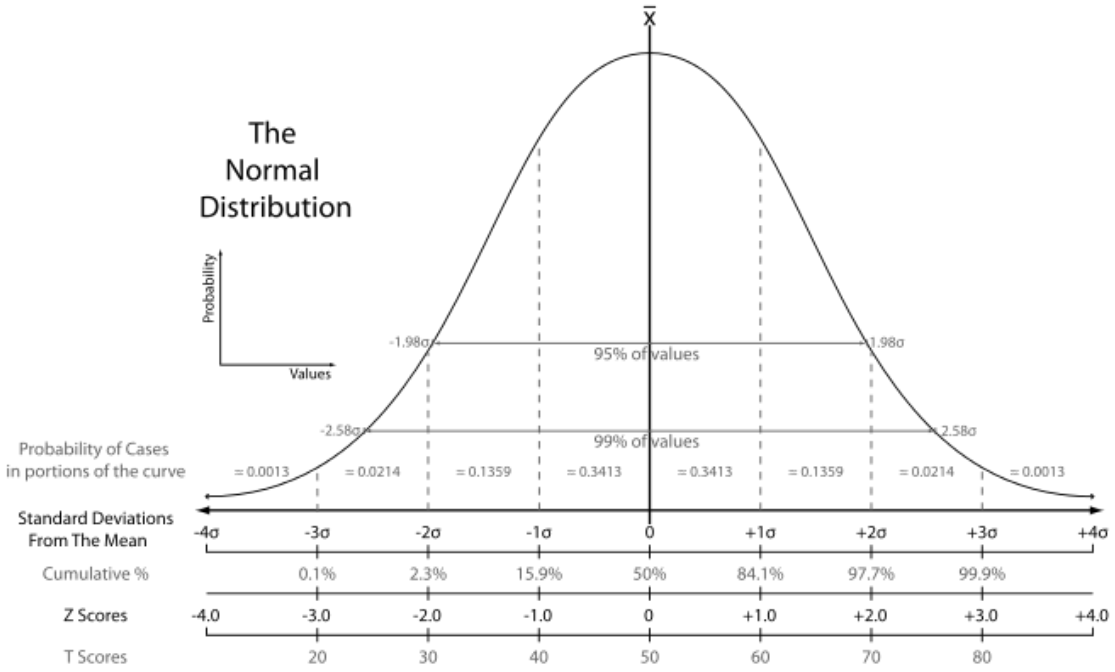
Let's look at the curve. The normal distribution has several characteristics:

```
1.) It has a lower tail (on the left) and an upper tail (on the right)
2.) The curve is symmetric (for the theoretical distribution)
3.) The peak occurs at the mean.
4.) The standard deviation gives the curve a different shape:
    -Narrow and tall for a smaller standard deviation.
    -Shallower and fatter for a larger standard deviation.
5.) The area under the curve is equal to 1 (the total probaility space)
6.) The mean=median=mode.
```

For the normal distribution, we can see what percentage of values lie between +/- a standard deviation. 68% of the values lie within 1 TSD, 95% between 2 STDs, and 99.7% between 3 STDs. The number of standard deviations is also called the z-score, which we saw above in the PDF.

In [7]:
```
from IPython.display import Image
Image(url='http://upload.wikimedia.org/wikipedia/commons/thumb/2/25/The_Normal_Di
```

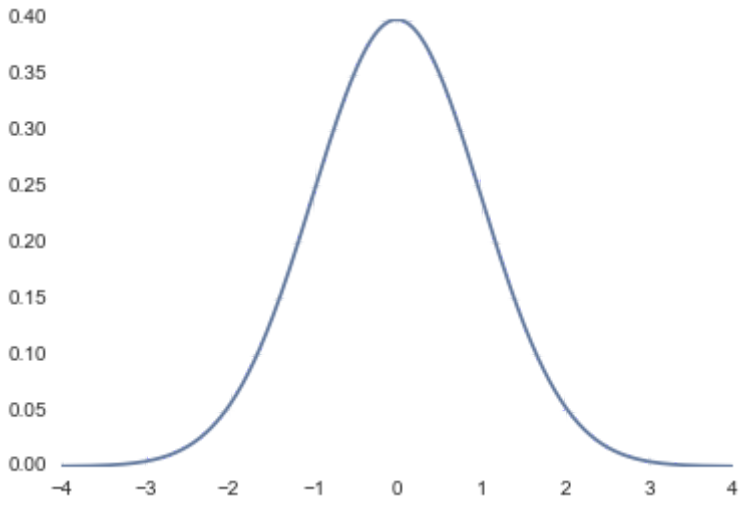
Out[7]:



Now let's learn how to use scipy to create a normal distribution

In [46]:
```
#Import
import matplotlib as mpl
import matplotlib.pyplot as plt
%matplotlib inline

#Import the stats library
from scipy import stats

# Set the mean
mean = 0

#Set the standard deviation
std = 1


# Create a range
X = np.arange(-4,4,0.01)

#Create the normal distribution for the range
Y = stats.norm.pdf(X,mean,std)

#
plt.plot(X,Y)
```

Out[46]:
```
<matplotlib.collections.PolyCollection at 0x239a6f28>
```

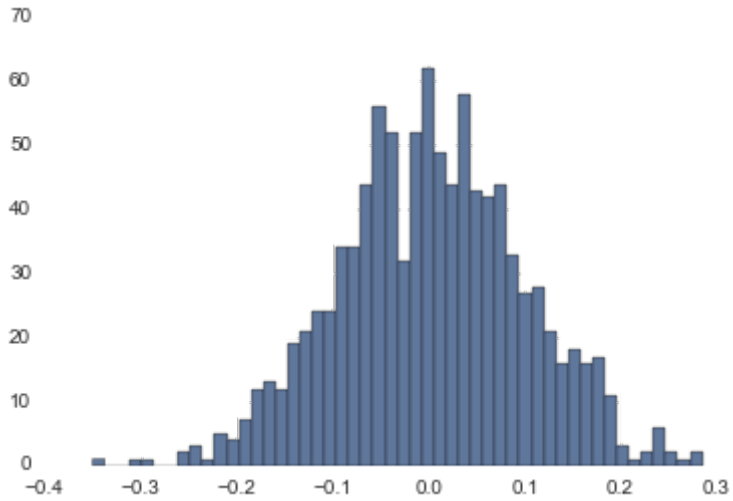


Now let's learn how to use numpy to create the normal distribution.

In [26]:
```
import numpy as np

#Set the mean and the standard deviaiton
mu,sigma = 0,0.1

# Now grab 1000 random numbers from the normal distribution
norm_set = np.random.normal(mu,sigma,1000)
```

In [27]:
```
#Now let's plot it using seaborn

import seaborn as sns

plt.hist(norm_set,bins=50)
```

Out[27]:
```
(array([  1.,   0.,   0.,   1.,   1.,   0.,   0.,   2.,   3.,   1.,   5.,
          4.,   7.,  12.,  13.,  12.,  19.,  21.,  24.,  24.,  34.,  34.,
         44.,  56.,  52.,  32.,  52.,  62.,  49.,  44.,  58.,  43.,  42.,
         44.,  33.,  27.,  28.,  21.,  16.,  18.,  16.,  17.,  11.,   3.,
          1.,   2.,   6.,   2.,   1.,   2.]),
 array([-0.34999642, -0.33728494, -0.32457346, -0.31186198, -0.2991505 ,
        -0.28643903, -0.27372755, -0.26101607, -0.24830459, -0.23559312,
        -0.22288164, -0.21017016, -0.19745868, -0.1847472 , -0.17203573,
        -0.15932425, -0.14661277, -0.13390129, -0.12118982, -0.10847834,
        -0.09576686, -0.08305538, -0.0703439 , -0.05763243, -0.04492095,
        -0.03220947, -0.01949799, -0.00678652,  0.00592496,  0.01863644,
         0.03134792,  0.0440594 ,  0.05677087,  0.06948235,  0.08219383,
         0.09490531,  0.10761678,  0.12032826,  0.13303974,  0.14575122,
         0.1584627 ,  0.17117417,  0.18388565,  0.19659713,  0.20930861,
         0.22202008,  0.23473156,  0.24744304,  0.26015452,  0.272866  ,
         0.28557747]),
 <a list of 50 Patch objects>)
```



There's a ton on information to go over for the normal distribution, this notebook should just serve as a very mild introduction, for more info check out the following sources:

1.) http://en.wikipedia.org/wiki/Normal_distribution (http://en.wikipedia.org/wiki/Normal_distribution)

2.) http://mathworld.wolfram.com/NormalDistribution.html (http://mathworld.wolfram.com/NormalDistribution.html)

3.) http://stattrek.com/probability-distributions/normal.aspx (http://stattrek.com/probability-distributions/normal.aspx)

Thanks!

In [ ]: