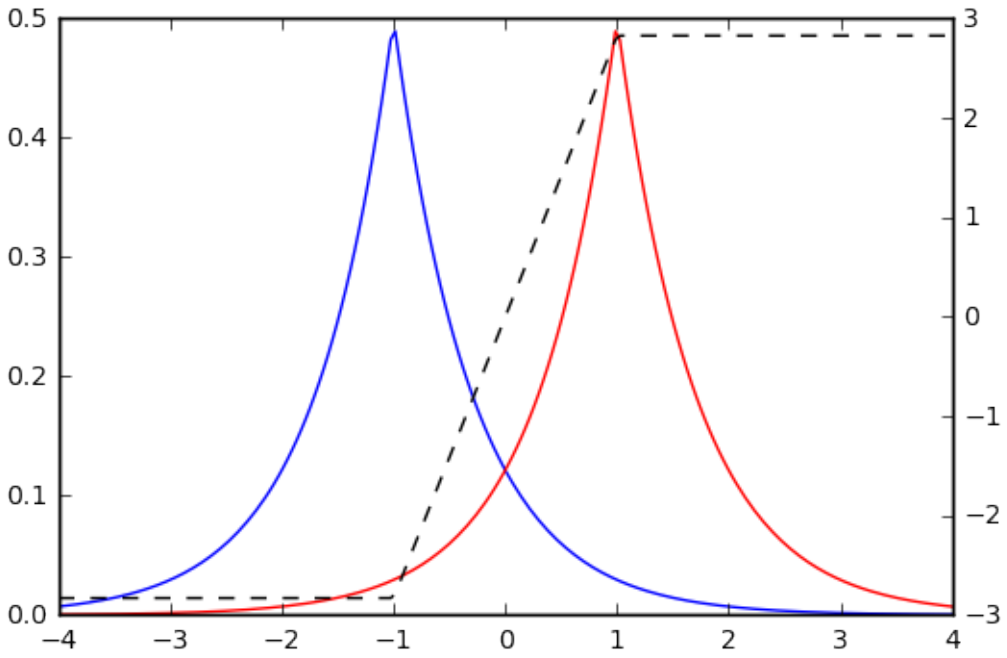
1Challenge

May 21, 2017

In [1]:
```
import numpy as np
from scipy.stats import laplace
import matplotlib.pyplot as plt
import math

fig, ax1 = plt.subplots()

mu0=-1
variance0=1
sigma0=math.sqrt(variance0/2)


x = np.linspace(-4, 4, 200)
ax1.plot(x,laplace.pdf((x-mu0)/sigma0),'b-')

mu1 = 1
variance1 = 1
sigma1 = math.sqrt(variance1/2)

x = np.linspace(-4, 4, 200)
ax1.plot(x,laplace.pdf((x-mu1)/sigma1),color='r')

ax2 = ax1.twinx()
ax2.plot(x,np.log(laplace.pdf((x-mu1)/sigma1)) - np.log(laplace.pdf((x-mu0)
```

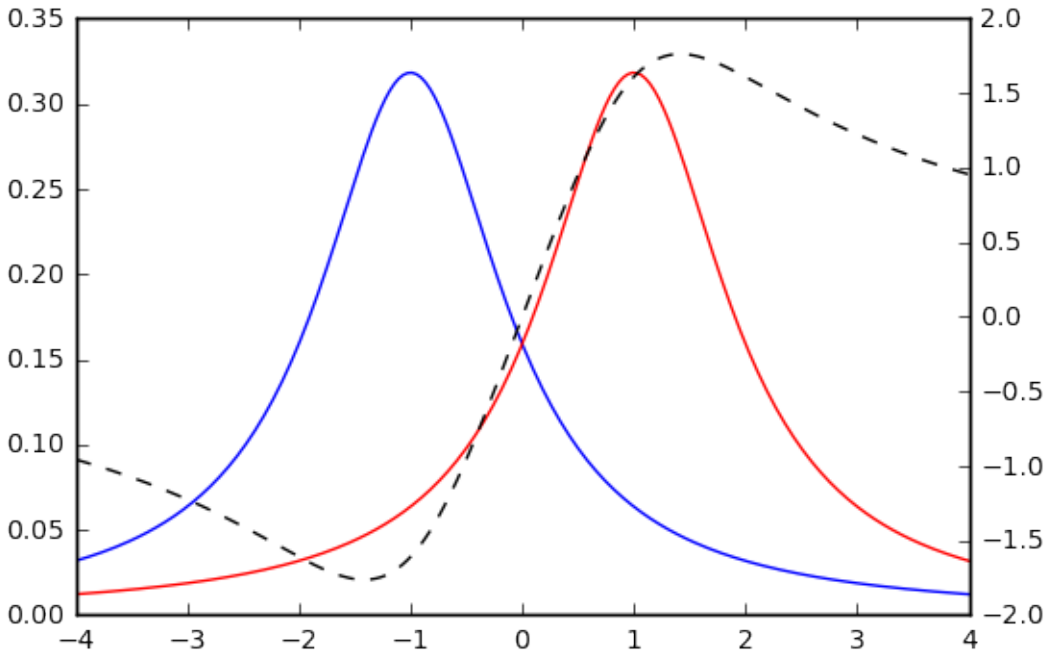
```
/opt/conda/lib/python3.5/site-packages/matplotlib/font_manager.py:273: UserWarning:
  warnings.warn('Matplotlib is building the font cache using fc-list. This may take
/opt/conda/lib/python3.5/site-packages/matplotlib/font_manager.py:273: UserWarning:
  warnings.warn('Matplotlib is building the font cache using fc-list. This may take
```

Out[1]:
```
[<matplotlib.lines.Line2D at 0x7efd51245400>]
```



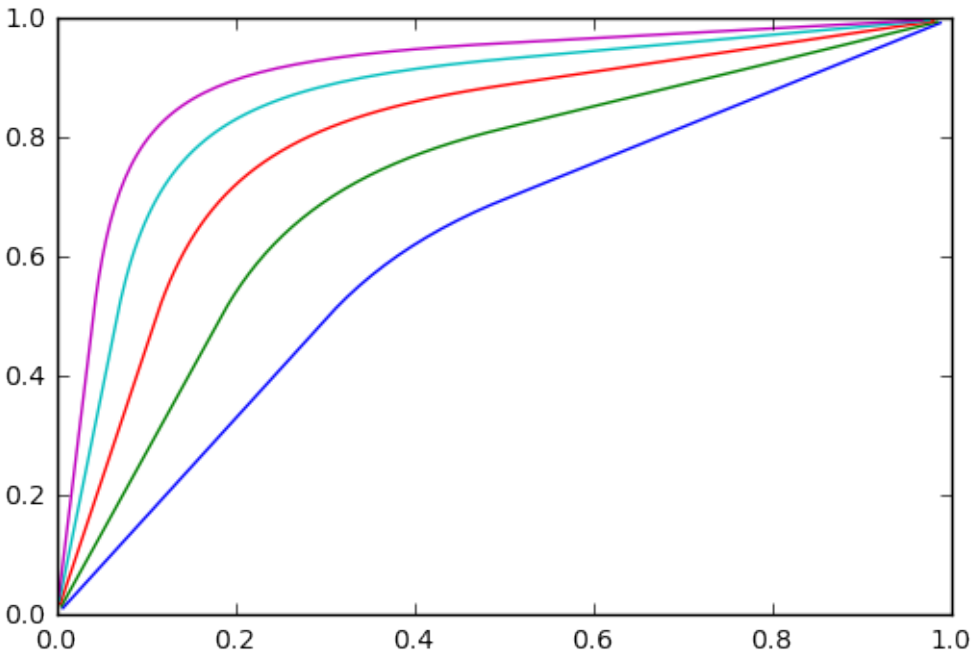
In [2]:
```
import numpy as np
from scipy.stats import cauchy
import matplotlib.pyplot as plt
import math

fig, ax1 = plt.subplots()

mu0=-1
variance0=1
sigma0=math.sqrt(variance0)

x = np.linspace(-4, 4, 200)
ax1.plot(x,cauchy.pdf((x-mu0)/sigma0),'b-')

mu1 = 1
variance1 = 1
sigma1 = math.sqrt(variance1)

x = np.linspace(-4, 4, 200)
ax1.plot(x,cauchy.pdf((x-mu1)/sigma1),color='r')

ax2 = ax1.twinx()
ax2.plot(x,np.log(cauchy.pdf(x-mu1)/sigma1) - np.log(cauchy.pdf(x-mu0)/sigm
```

Out[2]:
```
[<matplotlib.lines.Line2D at 0x7efd50b20978>]
```



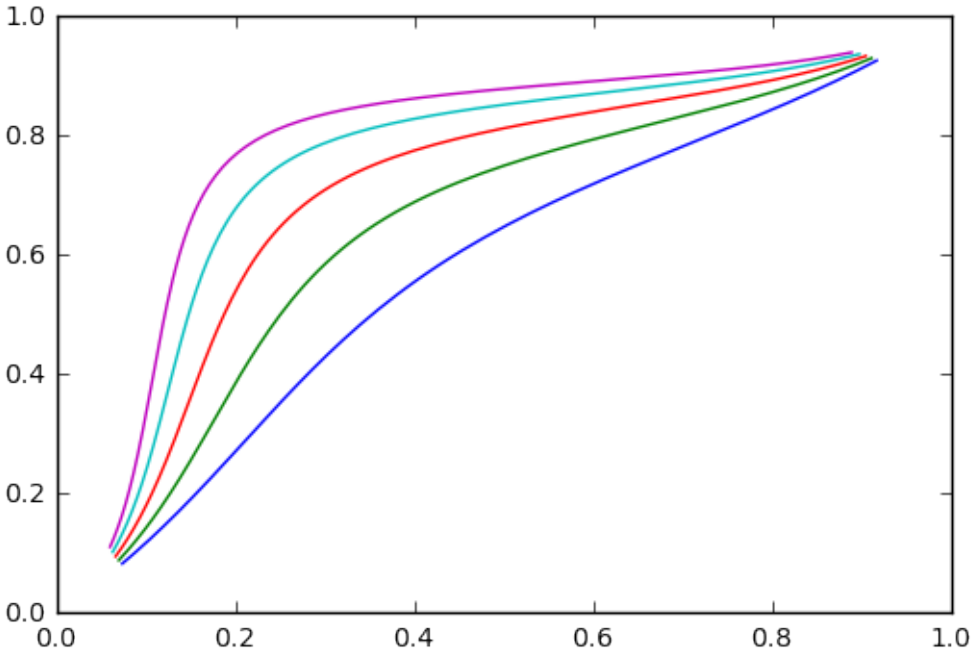
In [3]:
```
fig, ax1 = plt.subplots()

x = np.linspace(-4, 4, 200)
mu = np.linspace(0.25, 1.5, 6)

for index in range(5):
    mu0=-mu[index]
    mu1=mu[index]
    ax1.plot(laplace.cdf((x-mu1)/sigma1),laplace.cdf((x-mu0)/sigma0))
plt.show
```

Out[3]: <function matplotlib.pyplot.show>



In [4]:
```
fig, ax1 = plt.subplots()

x = np.linspace(-4, 4, 200)
mu = np.linspace(0.25, 1.5, 6)

for index in range(5):
    mu0=-mu[index]
    mu1=mu[index]
    ax1.plot(cauchy.cdf((x-mu1)/sigma1),cauchy.cdf((x-mu0)/sigma0))
plt.show
```

Out[4]: <function matplotlib.pyplot.show>



In [ ]: