

Sebastian Raschka  
last modified: 03/30/2014

## Problem Category

- Statistical Pattern Recognition
  - Supervised Learning
  - Parametric Learning
  - Bayes Decision Theory
  - Univariate data
  - 2-class problem
  - equal variances
  - equal priors
  - Gaussian model (2 parameters)
  - No Risk function
- 

## Given Information:

**model: continuous univariate normal (Gaussian) density**

$$p(x|\omega_j) \sim N(\mu|\sigma^2)$$

$$p(x|\omega_j) \sim \frac{1}{\sqrt{2\pi\sigma^2}} \exp \left[ -\frac{1}{2} \left( \frac{x-\mu}{\sigma} \right)^2 \right]$$

**parameters:**

$$P(\omega_1) = P(\omega_2) = 0.5$$

$$\sigma_1^2 = \sigma_2^2 = 1$$

$$\mu_1 = 4, \quad \mu_2 = 10$$

## Bayes' Rule:

$$P(\omega_j|x) = \frac{p(x|\omega_j) * P(\omega_j)}{p(x)}$$

## Bayes' Decision Rule:

Decide  $\omega_1$  if  $P(\omega_1|x) > P(\omega_2|x)$  else decide  $\omega_2$ .

$$\Rightarrow \frac{p(x|\omega_1) * P(\omega_1)}{p(x)} > \frac{p(x|\omega_2) * P(\omega_2)}{p(x)}$$

We can drop  $p(x)$  since it is just a scale factor.

$$\Rightarrow P(x|\omega_1) * P(\omega_1) > p(x|\omega_2) * P(\omega_2)$$

$$\Rightarrow \frac{p(x|\omega_1)}{p(x|\omega_2)} > \frac{P(\omega_2)}{P(\omega_1)}$$

$$\Rightarrow \frac{p(x|\omega_1)}{p(x|\omega_2)} > \frac{0.5}{0.5}$$

$$\Rightarrow \frac{p(x|\omega_1)}{p(x|\omega_2)} > 1$$

$$\Rightarrow \frac{1}{\sqrt{2\pi\sigma_1^2}} \exp \left[ -\frac{1}{2} \left( \frac{x-\mu_1}{\sigma_1} \right)^2 \right] > \frac{1}{\sqrt{2\pi\sigma_2^2}} \exp \left[ -\frac{1}{2} \left( \frac{x-\mu_2}{\sigma_2} \right)^2 \right]$$

Since we have equal variances, we can drop the first term completely.

$$\Rightarrow \exp \left[ -\frac{1}{2} \left( \frac{x-\mu_1}{\sigma_1} \right)^2 \right] > \exp \left[ -\frac{1}{2} \left( \frac{x-\mu_2}{\sigma_2} \right)^2 \right] \quad \left| \ln, \quad \mu_1 = 4, \quad \mu_2 = 10, \quad \sigma = 1 \right.$$

$$\Rightarrow -\frac{1}{2}(x-4)^2 > -\frac{1}{2}(x-10)^2 \quad | \times (-2)$$

$$\Rightarrow (x-4)^2 < (x-10)^2$$

$$\Rightarrow x^2 - 8x + 16 < x^2 - 20x + 100$$

$$\Rightarrow 12x < 84$$

$$\Rightarrow x < 7$$

## Plotting the Class Conditional Densities and Decision Boundary

In [2]: `%pylab inline`

```
import numpy as np
from matplotlib import pyplot as plt

def pdf(x, mu, sigma):
    """
    Calculates the normal distribution's probability density
    function (PDF).

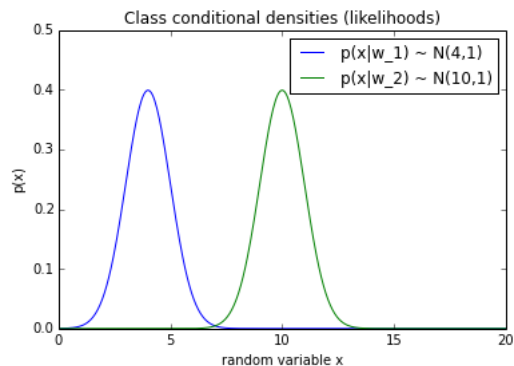
    """
    term1 = 1.0 / ( math.sqrt(2*np.pi) * sigma )
    term2 = np.exp( -0.5 * ( (x-mu)/sigma )**2 )
    return term1 * term2

# generating some sample data
x = np.arange(0, 100, 0.05)

# probability density functions
pdf1 = pdf(x, mu=4, sigma=1)
pdf2 = pdf(x, mu=10, sigma=1)

# Class conditional densities (likelihoods)
plt.plot(x, pdf1)
plt.plot(x, pdf2)
plt.title('Class conditional densities (likelihoods)')
plt.ylabel('p(x)')
plt.xlabel('random variable x')
plt.legend(['p(x|w_1) ~ N(4,1)', 'p(x|w_2) ~ N(10,1)'], loc='upper right')
plt.ylim([0,0.5])
plt.xlim([0,20])
plt.show()
```

Populating the interactive namespace from numpy and matplotlib



In [4]:

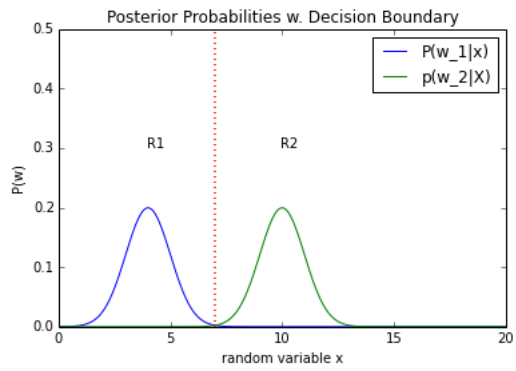
```
def posterior(likelihood, prior):
    """
    Calculates the posterior probability (after Bayes Rule) without
    the scale factor p(x) (=evidence).

    """
    return likelihood * prior

# probability density functions
posterior1 = posterior(pdf(x, mu=4, sigma=1), 0.5)
posterior2 = posterior(pdf(x, mu=10, sigma=1), 0.5)

# Class conditional densities (likelihoods)
plt.plot(x, posterior1)
plt.plot(x, posterior2)
plt.title('Posterior Probabilities w. Decision Boundary')
plt.ylabel('P(w)')
plt.xlabel('random variable x')
```

```
plt.legend(['P(w_1|x)', 'p(w_2|X)'], loc='upper right')
plt.ylim([0,0.5])
plt.xlim([0,20])
plt.axvline(7, color='r', alpha=0.8, linestyle=':', linewidth=2)
plt.annotate('R1', xy=(4, 0.3), xytext=(4, 0.3))
plt.annotate('R2', xy=(10, 0.3), xytext=(10, 0.3))
plt.show()
```



In []: