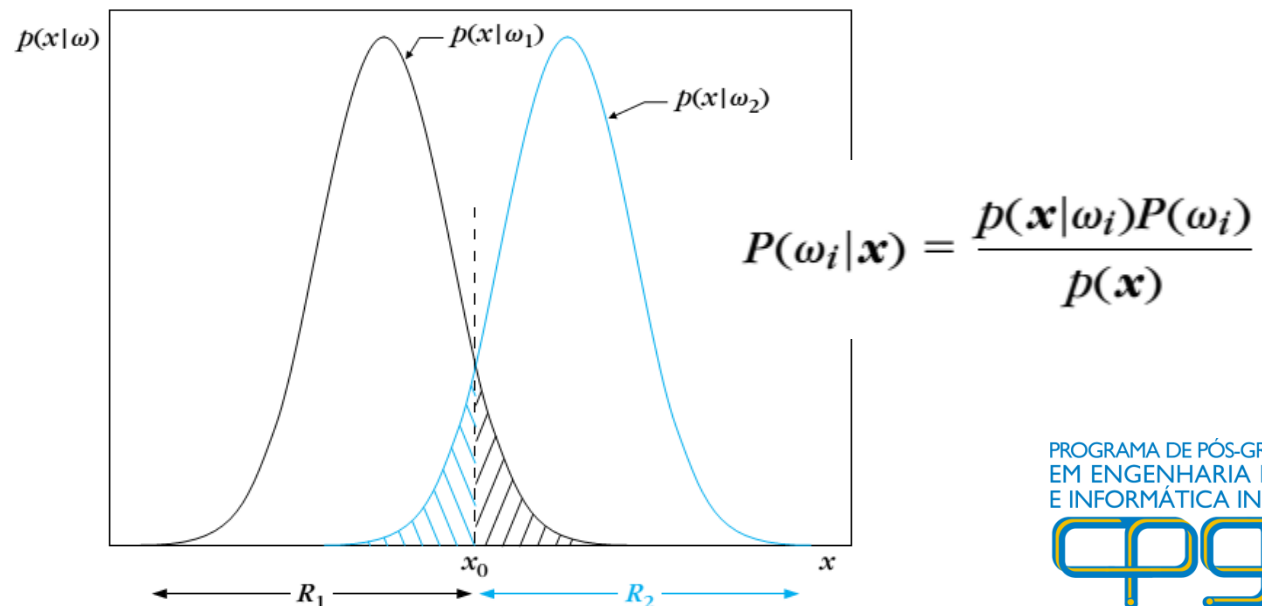


# Modelos Estatísticos e Classificadores Baseados na Teoria de Decisão Bayesiana

André E. Lazzaretti

UTFPR/CPGEI



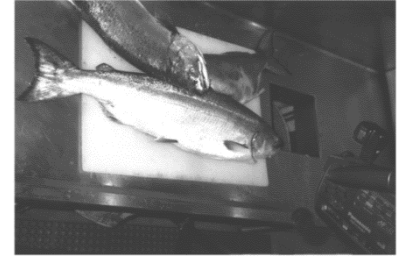
# Teoria de Decisão Bayesiana

**Problema:** Classificar salmão e badejo para separação automática em um esteira.

**Alternativa 1:** somente  $P(\omega_i)$  – probabilidade *a priori*

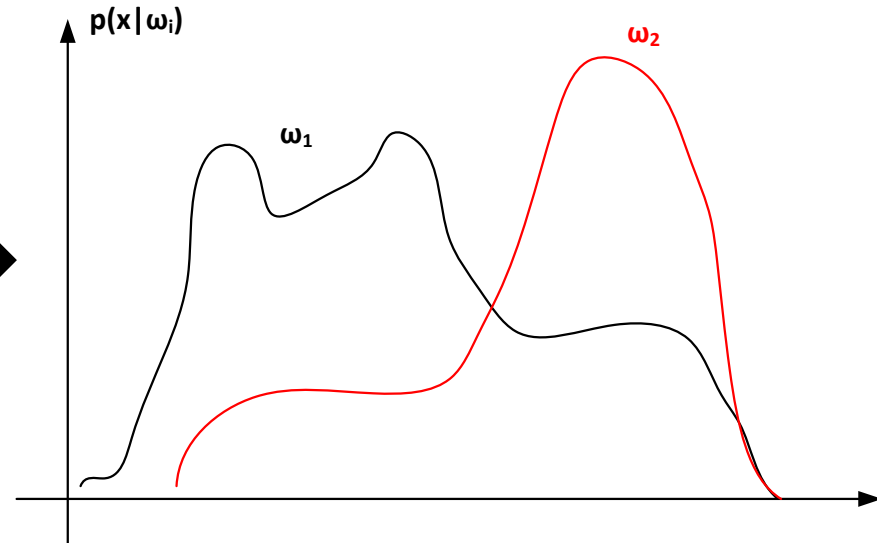
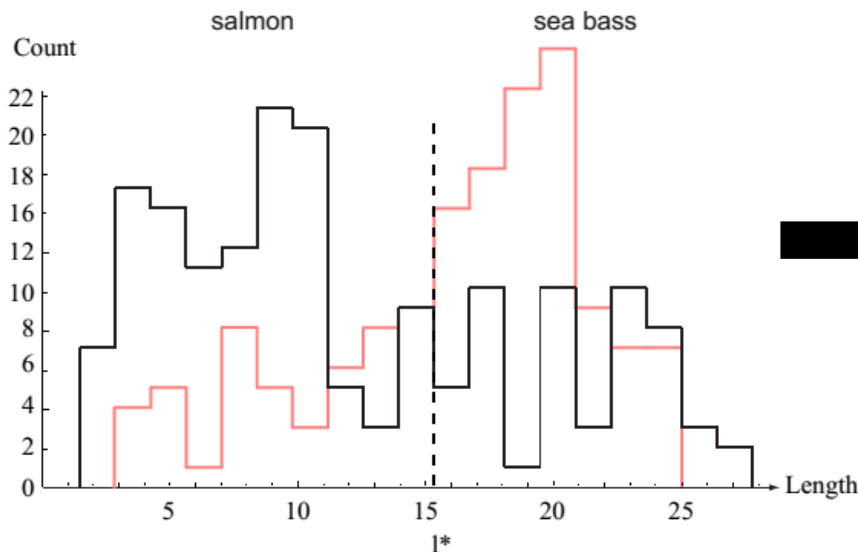
**Problema:** Para um peixe faz sentido, mas podemos sempre cometer o mesmo erro para muitas amostras.

**Alternativa 2:** regra de Bayes, combinando  $P(\omega_i)$  e  $p(\mathbf{x}|\omega_i)$



$$P(\omega_i|\mathbf{x}) = \frac{p(\mathbf{x}|\omega_i)P(\omega_i)}{p(\mathbf{x})}$$

Possível  $p(\mathbf{x}|\omega_i)$ :



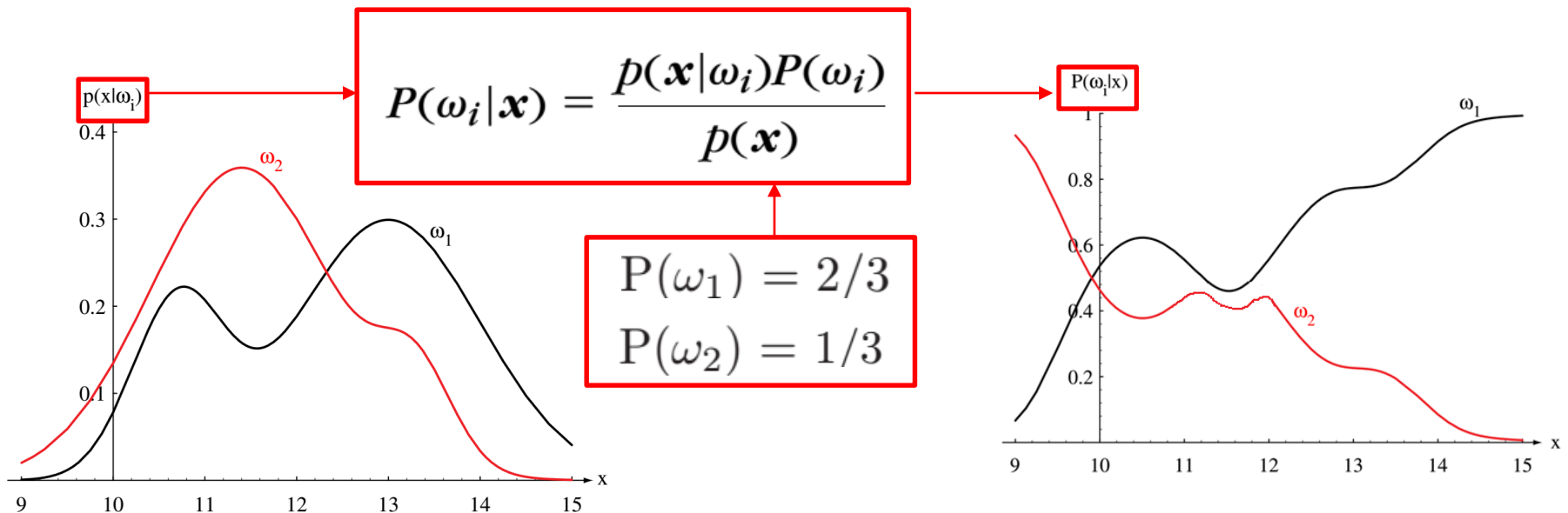
# Teoria de Decisão Bayesiana

**Problema:** Classificar salmão e badejo para separação automática em um esteira.

**Alternativa 1:** somente  $P(\omega_i)$

**Problema:** Para um peixe faz sentido, mas podemos sempre cometer o mesmo erro para muitas amostras.

**Alternativa 2:** regra de Bayes, combinando  $P(\omega_i)$  e  $p(\mathbf{x}|\omega_i)$



Superfície de decisão:  $P(\omega_i|\mathbf{x}) - P(\omega_j|\mathbf{x}) = 0$

# Abordagem desta Aula

$$P(\omega_i|\mathbf{x}) = \frac{p(\mathbf{x}|\omega_i)P(\omega_i)}{p(\mathbf{x})}$$

- 1)**  $P(\omega_i)$  e  $p(\mathbf{x}|\omega_i)$  são completamente conhecidas
  - Classificador Bayesiano e Classificadores baseados em distâncias
- 2)**  $p(\mathbf{x}|\omega_i)$  é conhecida, mas os parâmetros são desconhecidos (abordagem paramétrica) → definição dos parâmetros
  - MLE, MAP, Inferência Bayesiana, Combinação linear de PDFs
- 3)**  $p(\mathbf{x}|\omega_i)$  é estimada de forma não-paramétrica
  - k-NN e Janelas de Parzen
- 4)**  $p(\mathbf{x}|\omega_i)$  é estimada com modelos alternativos, levando em conta a independência/dependência das entradas
  - Naive Bayes e Redes Bayesianas

# Classificador Bayesiano para Classes Normalmente Distribuídas (Gaussianas)

- Considerando que  $p(\mathbf{x}|\omega_i)$  é uma distribuição Gaussiana multivariada, pode-se derivar o classificador Bayesiano para classes normalmente distribuídas:

$$p(\mathbf{x}|\omega_i) = \frac{1}{(2\pi)^{l/2} |\Sigma_i|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_i)^T \Sigma_i^{-1} (\mathbf{x} - \boldsymbol{\mu}_i)\right)$$

- Para tanto, será considerada seguinte função:

$$g_i(\mathbf{x}) = \ln(p(\mathbf{x}|\omega_i)P(\omega_i)) = \ln p(\mathbf{x}|\omega_i) + \ln P(\omega_i)$$

$$g_i(\mathbf{x}) = -\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_i)^T \Sigma_i^{-1} (\mathbf{x} - \boldsymbol{\mu}_i) + \ln P(\omega_i) + c_i$$

$$g_i(\mathbf{x}) = -\frac{1}{2}\mathbf{x}^T \Sigma_i^{-1} \mathbf{x} + \frac{1}{2}\mathbf{x}^T \Sigma_i^{-1} \boldsymbol{\mu}_i - \frac{1}{2}\boldsymbol{\mu}_i^T \Sigma_i^{-1} \boldsymbol{\mu}_i + \frac{1}{2}\boldsymbol{\mu}_i^T \Sigma_i^{-1} \mathbf{x} + \ln P(\omega_i) - (l/2) \ln 2\pi - (1/2) \ln |\Sigma_i|$$

 **Por classe!**

# Classificador Bayesiano para Classes Normalmente Distribuídas (Gaussianas)

- Na fronteira de decisão entre a classe  $i$  e  $j$ :

$$g_i(\mathbf{x}) - g_j(\mathbf{x}) = 0$$

- No caso em que  $\Sigma_i = \Sigma = \sigma^2 \mathbf{I}$  e  $P(\omega_j) = P(\omega_i)$ , o termo quadrático ( $\mathbf{x}^T \Sigma_i^{-1} \mathbf{x}$ ) é o mesmo para todas as classes, as constantes e  $P(\omega)$  também se anulam:

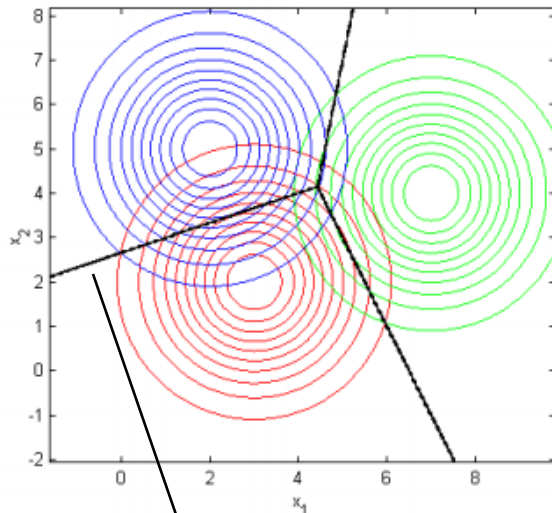
$$\begin{aligned} & \cancel{-\frac{1}{2} \mathbf{x}^T \Sigma^{-1} \mathbf{x}} + \cancel{\frac{1}{2} \mathbf{x}^T \Sigma^{-1} \mu_i} - \cancel{\frac{1}{2} \mu_i^T \Sigma^{-1} \mu_i} + \cancel{\frac{1}{2} \mu_i^T \Sigma^{-1} \mathbf{x}} = \\ & \cancel{-\frac{1}{2} \mathbf{x}^T \Sigma^{-1} \mathbf{x}} + \cancel{\frac{1}{2} \mathbf{x}^T \Sigma^{-1} \mu_j} - \cancel{\frac{1}{2} \mu_j^T \Sigma^{-1} \mu_j} + \cancel{\frac{1}{2} \mu_j^T \Sigma^{-1} \mathbf{x}} \\ & (\mu_i - \mu_j)^T \mathbf{x} - \frac{1}{2} (\mu_i^T \mu_i - \mu_j^T \mu_j) = 0 \end{aligned}$$

# Classificador Bayesiano para Classes Normalmente Distribuídas (Gaussianas)

- No caso que  $\Sigma_i = \Sigma = \sigma^2 \mathbf{I}$  e  $P(\omega_j) = P(\omega_i)$ :

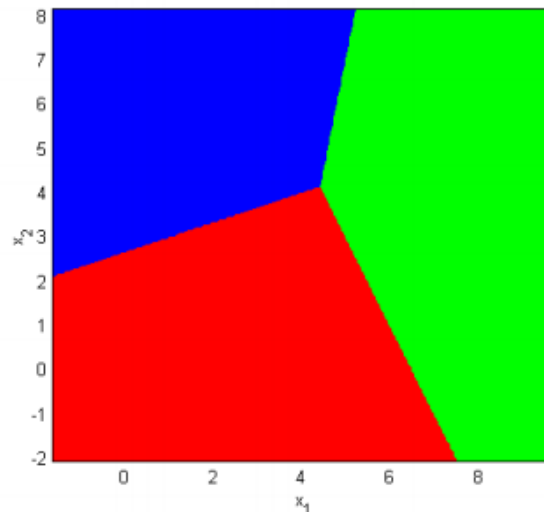
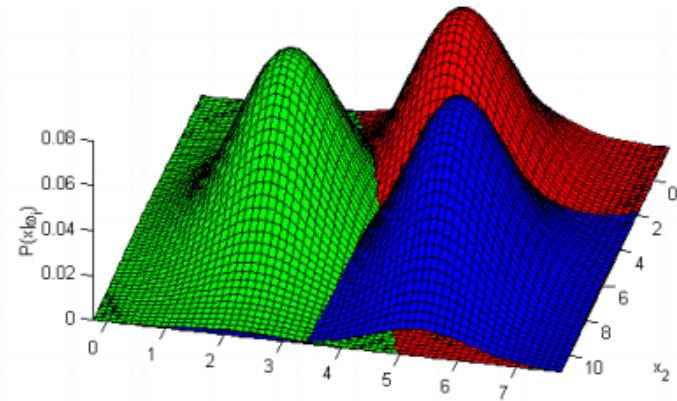
$$\mu_1 = [3 \ 2]^T \quad \mu_2 = [7 \ 4]^T \quad \mu_3 = [2 \ 5]^T$$

$$\Sigma_1 = \begin{bmatrix} 2 & \\ & 2 \end{bmatrix} \quad \Sigma_2 = \begin{bmatrix} 2 & \\ & 2 \end{bmatrix} \quad \Sigma_3 = \begin{bmatrix} 2 & \\ & 2 \end{bmatrix}$$



$$(\mu_1 - \mu_2)^T x - \frac{1}{2}(\mu_1^T \mu_1 - \mu_2^T \mu_2)$$

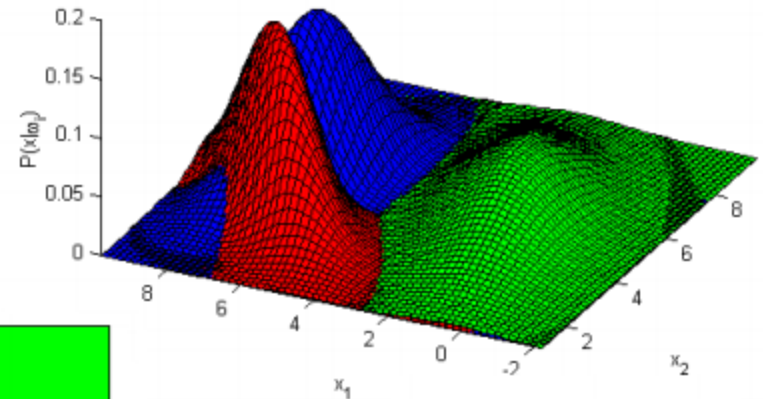
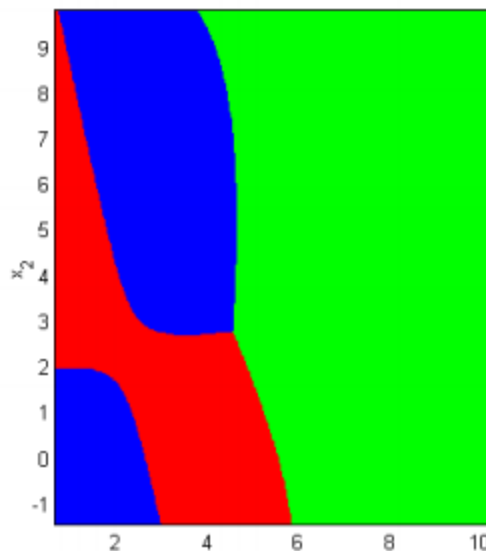
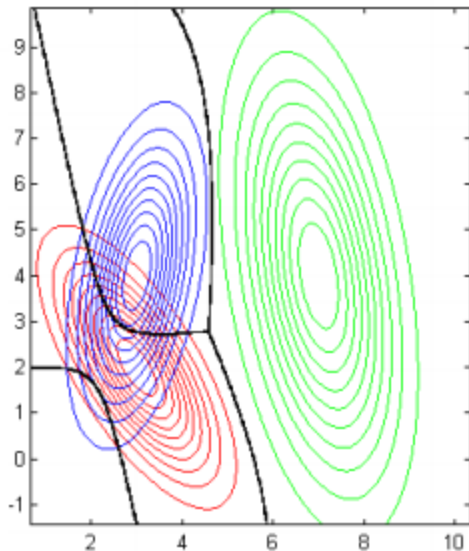
$$\mathbf{w}^T \mathbf{x} + w_0$$



# Classificador Bayesiano para Classes Normalmente Distribuídas (Gaussianas)

- No caso que  $\Sigma_i \neq \Sigma_j$  (não diagonal) e  $P(\omega_j) = P(\omega_i)$ :

$$\begin{aligned} \mu_1 &= [3 \ 2]^T & \mu_2 &= [5 \ 4]^T & \mu_3 &= [3 \ 4]^T \\ \Sigma_1 &= \begin{bmatrix} 1 & -1 \\ -1 & 2 \end{bmatrix} & \Sigma_2 &= \begin{bmatrix} 1 & -1 \\ -1 & 7 \end{bmatrix} & \Sigma_3 &= \begin{bmatrix} .5 & .5 \\ .5 & 3 \end{bmatrix} \end{aligned}$$





# Exemplo no Matlab

- Em um problema com duas classes (bidimensional), os vetores de característica em cada classe são normalmente distribuídos de acordo com os parâmetros apresentados abaixo. Assuma que  $P(\omega_1)=P(\omega_2)$  e projete um classificador Bayesiano que minimize o erro de classificação. Qual o erro percentual de classificação? Repita os experimentos para  $\mu_2 = [3.0, 3.0]^T$ .

$$p(\mathbf{x}|\omega_1) = \frac{1}{\left(\sqrt{2\pi\sigma_1^2}\right)^2} \exp\left(-\frac{1}{2\sigma_1^2}(\mathbf{x} - \boldsymbol{\mu}_1)^T(\mathbf{x} - \boldsymbol{\mu}_1)\right)$$

$$p(\mathbf{x}|\omega_2) = \frac{1}{\left(\sqrt{2\pi\sigma_2^2}\right)^2} \exp\left(-\frac{1}{2\sigma_2^2}(\mathbf{x} - \boldsymbol{\mu}_2)^T(\mathbf{x} - \boldsymbol{\mu}_2)\right)$$

$$\boldsymbol{\mu}_1 = [1, 1]^T, \quad \boldsymbol{\mu}_2 = [1.5, 1.5]^T, \quad \sigma_1^2 = \sigma_2^2 = 0.2$$

```
close all; clc; clear; rand('seed',0); randn('seed',0);
```

```
mu1 = [ 1, 1 ].';  
mu2 = [ 1.5, 1.5 ].';  
sigmasSquared = 0.2;  
d = size(mu1,1);
```

```
nFeats = 10000;
```

```
X1 = mvnrnd( mu1, sigmasSquared*eye(d), nFeats );  
X2 = mvnrnd( mu2, sigmasSquared*eye(d), nFeats );
```

```
h1 = plot( X1(:,1), X1(:,2), '.b' ); hold on;  
h2 = plot( X2(:,1), X2(:,2), '.r' ); hold on;  
legend( [h1,h2], {'classe 1', 'classe 2'} );
```

```
mean_diff = mu1 - mu2;
```

```
X = [ X1; X2 ];  
labels = [ ones(nFeats,1); 2*ones(nFeats,1) ];
```

```
rhs = 0.5 * ( dot(mu1,mu1) - dot(mu2,mu2) );  
lhs = mean_diff' * X';
```

```
class_decision = (lhs - rhs) > 0;  
choosen_class = zeros(2*nFeats,1);  
choosen_class(find(class_decision==1)) = 1;  
choosen_class(find(class_decision~=1)) = 2;
```

```
P_correct = sum(choosen_class == labels)/(2*nFeats);  
P_error = 1 - P_correct;
```

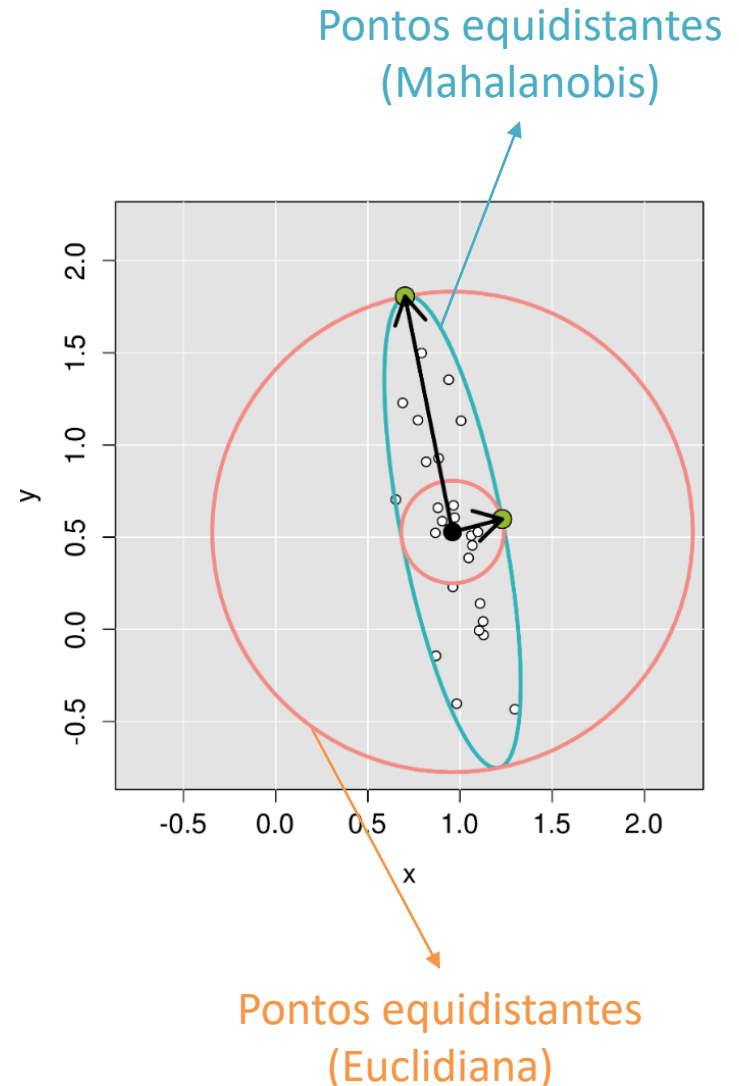
# Classificadores de Distância Mínima

- No caso que  $\Sigma_i = \Sigma = \sigma^2 \mathbf{I}$  e  $P(\omega_j) = P(\omega_i)$ , a classificação se torna (menor distância):

$$d_{\epsilon} = \|\mathbf{x} - \boldsymbol{\mu}_i\|$$

- No caso que  $\Sigma$  é não-diagonal e  $P(\omega_j) = P(\omega_i)$ , a classificação se torna (menor distância):

$$d_m = \left( (\mathbf{x} - \boldsymbol{\mu}_i)^T \Sigma^{-1} (\mathbf{x} - \boldsymbol{\mu}_i) \right)^{1/2}$$



# Classificadores de Distância Mínima

- Utilize o classificador Bayesiano para classificar  $[1,0; 2,2]^T$ , sendo que as classes são definidas por:

$$\Sigma = \begin{bmatrix} 1.1 & 0.3 \\ 0.3 & 1.9 \end{bmatrix} \quad \boldsymbol{\mu}_1 = [0, 0]^T, \boldsymbol{\mu}_2 = [3, 3]^T$$

$$\begin{aligned} d_m^2(\boldsymbol{\mu}_1, \mathbf{x}) &= (\mathbf{x} - \boldsymbol{\mu}_1)^T \Sigma^{-1} (\mathbf{x} - \boldsymbol{\mu}_1) \\ &= [1.0, 2.2] \begin{bmatrix} 0.95 & -0.15 \\ -0.15 & 0.55 \end{bmatrix} \begin{bmatrix} 1.0 \\ 2.2 \end{bmatrix} = 2.952 \end{aligned}$$

$$d_m^2(\boldsymbol{\mu}_2, \mathbf{x}) = [-2.0, -0.8] \begin{bmatrix} 0.95 & -0.15 \\ -0.15 & 0.55 \end{bmatrix} \begin{bmatrix} -2.0 \\ -0.8 \end{bmatrix} = 3.672$$

# Estimativa para PDFs Desconhecidas

- Até o momento, consideramos que as PDFs são conhecidas previamente;
- No entanto, esse não é o caso na maioria das situações práticas;
- Existem várias abordagens para esses casos:
  - Se conhece o tipo da pdf e se desconhece os parâmetros associados;
  - Pode-se conhecer algum parâmetro e se desconhecer o tipo da pdf → não será abordado aqui!

# Maximum Likelihood Parameter Estimation

- Estimativa paramétrica (Gaussiana): assume-se uma pdf para  $p(\mathbf{X}|\omega_i)$  ou  $p(\mathbf{X}|\omega_i;\boldsymbol{\theta})$  ou  $p(\mathbf{X}|\boldsymbol{\theta})$  e estima-se os parâmetros  $\boldsymbol{\theta}$  a partir dos dados;
- Ideia Geral (estimativa de parâmetros):

$$p(\boldsymbol{\theta}|X) = \frac{p(\boldsymbol{\theta})p(X|\boldsymbol{\theta})}{p(X)} \quad \rightarrow$$

$$posterior = \frac{likelihood \times prior}{evidence}$$

Assumindo que os dados de uma classe não afetam a estimativa de parâmetros das demais, i.e. removendo  $\omega_i$

# Maximum Likelihood Parameter Estimation

- Estimativa paramétrica (Gaussiana): assume-se uma pdf para  $p(\mathbf{X}|\boldsymbol{\theta}_i)$  e estima-se os parâmetros  $\boldsymbol{\theta}_i$  a partir dos dados;
- Ideia Geral (estimativa de parâmetros):

$$p(\boldsymbol{\theta}|X) = \frac{p(\boldsymbol{\theta})p(X|\boldsymbol{\theta})}{p(X)}$$

Foco Inicial do MLE

# Maximum Likelihood Parameter Estimation

- Formulação Geral (assume-se independência!):

$$p(\mathbf{X}|\boldsymbol{\theta}) = p(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N|\boldsymbol{\theta}) = \prod_{k=1}^N p(\mathbf{x}_k|\boldsymbol{\theta})$$

- Estimativa:

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{\max} \left[ \prod_{k=1}^N p(\mathbf{x}_k|\boldsymbol{\theta}) \right]$$

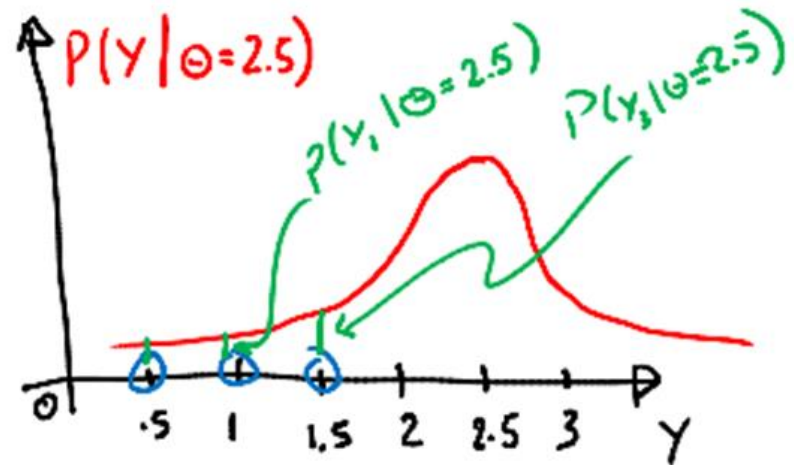
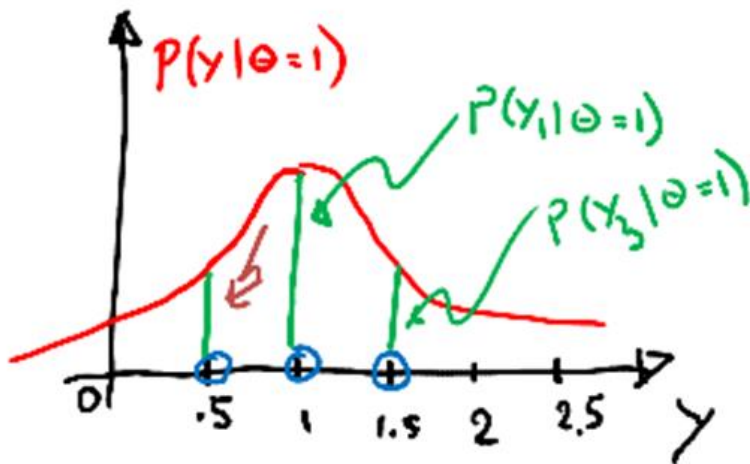


# Interpretação MLE

We have  $n=3$  data points  $y_1 = 1, y_2 = 0.5, y_3 = 1.5$ , which are independent and Gaussian with **unknown** mean  $\theta$  and variance **1**:

with likelihood  $P(y_1 y_2 y_3 | \theta) = P(y_1 | \theta) P(y_2 | \theta) P(y_3 | \theta)$ . Consider two guesses of  $\theta$ , 1 and 2.5. Which has higher likelihood?

Finding the  $\theta$  that maximizes the likelihood is equivalent to moving the Gaussian until the product of 3 green bars (likelihood) is maximized.



# Maximum Likelihood Parameter Estimation

- Normalmente se utiliza uma função logarítmica (preserva o máximo e evita produtos entre probabilidades):

$$L(\boldsymbol{\theta}) = \log \prod_{k=1}^N p(\mathbf{x}_k | \boldsymbol{\theta})$$

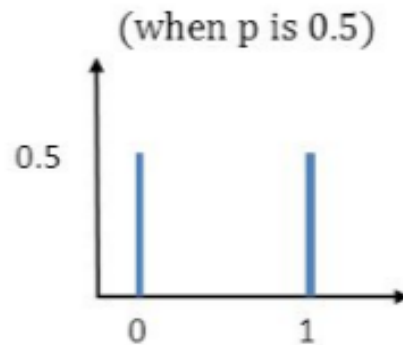
$$\frac{\partial L(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = \mathbf{0}$$

# Exemplo

- Considere  $N$  experimentos independentes seguindo distribuição de Bernoulli. Será solicitado a cada indivíduo da população ( $\chi$ ) se ele vai votar para os democratas ou republicanos na próxima eleição presidencial. Estime  $p$  usando ML.

$$\mathcal{X} = \left\{ x_i = \begin{cases} \text{Democratic} \\ \text{Republican} \end{cases}, \quad i = 1 \dots N \right\}$$

**Bernoulli (pmf):**



$$f(k; p) = \begin{cases} p & \text{if } k = 1, \\ 1 - p & \text{if } k = 0. \end{cases}$$
$$f(k; p) = p^k (1 - p)^{1-k} \quad \text{for } k \in \{0, 1\}.$$

$$\text{Solução: } \mathcal{L}(\theta) = \log \prod_{i=1}^N p(x_i|\theta) = \sum_{i=1}^N \log(p(x_i|\theta))$$

$$= \sum_{i=1}^{n_d} \log[p(x_i = Demo|\theta)] + \sum_{i=1}^{N-n_d} \log[p(x_i = Repub|\theta)]$$

$$pmf(k; \theta) = \theta^k + (1 - \theta)^{1-k} \text{ para } k \in \{Demo, Repub\}$$

$$\mathcal{L}(\theta) = \sum_{i=1}^{n_d} \log \theta + \sum_{i=1}^{N-n_d} \log(1 - \theta) = n_d \log \theta + (N - n_d) \log(1 - \theta)$$

$$\frac{\partial \mathcal{L}}{\partial \theta} = \frac{n_d}{\theta} - \frac{N - n_d}{1 - \theta} = 0$$

$$\hat{\theta}_{ML} = \frac{n_d}{N}$$

# Caso Gaussiano

- Caso Gaussiano 1-D:

$$m_{ML} = \frac{1}{N} \sum_{i=1}^N x_i$$

$$S_{ML} = \frac{1}{N} \sum_{i=1}^N (x_i - m_{ML})(x_i - m_{ML})^T$$

- Exercício da Lista!

```
close all; clc; clear; rand('seed',0); randn('seed',0);
```

```
mu1 = [ 1, 1 ].';  
mu2 = [ 1.5, 1.5 ].';  
sigmasSquared = 0.2;  
d = size(mu1,1);
```

```
nFeats = 10000;
```

```
X1 = mvnrnd( mu1, sigmasSquared*eye(d), nFeats );  
X2 = mvnrnd( mu2, sigmasSquared*eye(d), nFeats );
```

```
mu1_ML = (1/nFeats)*sum(X1)';  
mu2_ML = (1/nFeats)*sum(X2)';
```

```
h1 = plot( X1(:,1), X1(:,2), '.b' ); hold on;  
h2 = plot( X2(:,1), X2(:,2), '.r' ); hold on;  
legend( [h1,h2], {'classe 1', 'classe 2'} );
```

```
mean_diff = mu1_ML - mu2_ML;
```

```
X = [ X1; X2 ];  
labels = [ ones(nFeats,1); 2*ones(nFeats,1) ];
```

```
rhs = 0.5 * ( dot(mu1_ML,mu1_ML) - dot(mu2_ML,mu2_ML) );  
lhs = mean_diff' * X';
```

```
class_decision = (lhs - rhs) > 0;  
choosen_class = zeros(2*nFeats,1);  
choosen_class(find(class_decision==1)) = 1;  
choosen_class(find(class_decision~=1)) = 2;
```

```
P_correct = sum(choosen_class == labels)/(2*nFeats);  
P_error = 1 - P_correct;
```

# Maximum a Posteriori Estimation (MAP)

- No MLE,  $\theta_i$  representava o vetor de parâmetros desconhecidos.
- No caso do MAP, considera-se que esse vetor tem uma característica aleatória, ou seja, tem uma determinada pdf *a priori* e queremos estabelecer sua nova distribuição, à luz dos dados:

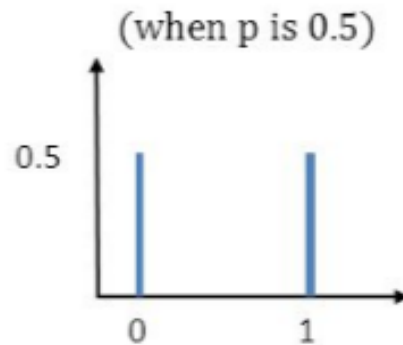
$$p(\theta|X) = \frac{\overset{\text{MAP}}{p(\theta)} \overset{\text{MLE}}{p(X|\theta)}}{p(X)}$$

# Exemplo

- Considere  $N$  experimentos independentes seguindo distribuição de Bernoulli. Será solicitado a cada indivíduo da população ( $\chi$ ) se ele vai votar para os democratas ou republicanos na próxima eleição presidencial. Estime  $p$  usando MAP.

$$\mathcal{X} = \left\{ x_i = \begin{cases} \textit{Democratic} \\ \textit{Republican} \end{cases}, \quad i = 1 \dots N \right\}$$

**Bernoulli (pmf):**

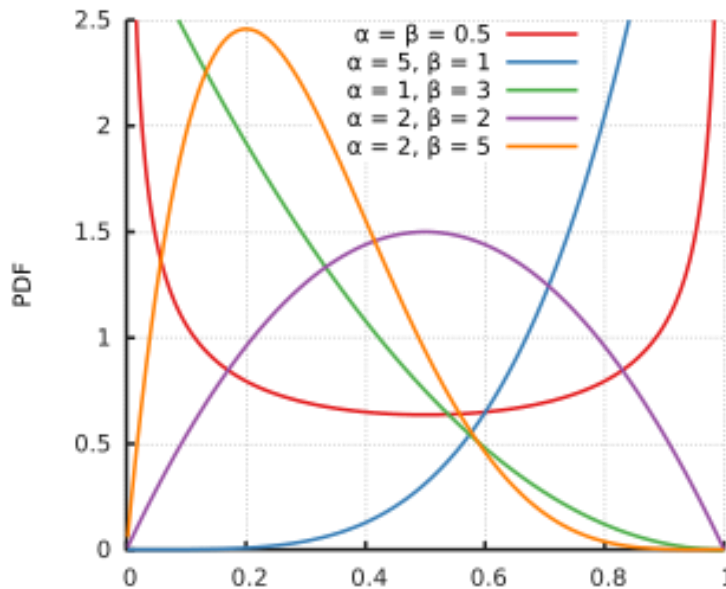


$$f(k; p) = \begin{cases} p & \text{if } k = 1, \\ 1 - p & \text{if } k = 0. \end{cases}$$
$$f(k; p) = p^k (1 - p)^{1-k} \quad \text{for } k \in \{0, 1\}.$$



# Exemplo cont.

- Beta (pdf) a priori:



$$\frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} x^{\alpha-1}(1-x)^{\beta-1}$$

$$\Gamma(z) = \int_0^{\infty} x^{z-1} e^{-x} dx.$$

- Estimativa é dada por:

$$\hat{\boldsymbol{\theta}}_{MAP} : \frac{\partial}{\partial \boldsymbol{\theta}} p(\boldsymbol{\theta}|X) = 0 \quad \text{or} \quad \frac{\partial}{\partial \boldsymbol{\theta}} (p(\boldsymbol{\theta})p(X|\boldsymbol{\theta})) = 0$$

$$\begin{aligned}\mathcal{L}(\theta) &= \log \prod_{i=1}^N p(x_i|\theta)p(\theta) = \sum_{i=1}^N \log(p(x_i|\theta)) p(\theta) \\ &= \sum_{i=1}^{n_d} \log[p(x_i = Demo|\theta)] + \sum_{i=1}^{N-n_d} \log[p(x_i = Repub|\theta)] + \log(p(\theta))\end{aligned}$$

$$pmf(k; \theta) = \theta^k + (1 - \theta)^{1-k} \text{ para } k \in \{Demo, Repub\}$$

$$p(\theta) = \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} \theta^{\alpha-1} (1 - \theta)^{\beta-1}$$

$$\mathcal{L}(\theta) = \sum_{i=1}^{n_d} \log \theta + \sum_{i=1}^{N-n_d} \log(1 - \theta) + \log(p(\theta))$$

$$= n_d \log \theta + (N - n_d) \log(1 - \theta) + (\alpha - 1) \log \theta + (\beta - 1) \log(1 - \theta)$$

$$\frac{\partial \mathcal{L}}{\partial \theta} = \frac{n_d}{\theta} - \frac{N - n_d}{1 - \theta} + \frac{\alpha - 1}{\theta} - \frac{\beta - 1}{1 - \theta} = 0$$

$$\hat{\theta}_{MAP} = \frac{n_d + \alpha - 1}{N + \alpha + \beta - 2}$$

# MAP e MLE

- Uma vez que os parâmetros estejam determinados ( $\theta_{MAP}$  e  $\theta_{ML}$ ), pode-se realizar a classificação de um novo exemplo  $\tilde{\mathbf{x}}$  de acordo com a probabilidade a posteriori de cada classe e a teoria de decisão Bayesiana:

$$P(\omega_i|\tilde{\mathbf{x}}) = p(\tilde{\mathbf{x}}|\omega_i; \theta_{ML})P(\omega_i)$$

$$P(\omega_i|\tilde{\mathbf{x}}) = p(\tilde{\mathbf{x}}|\omega_i; \theta_{MAP})P(\omega_i)$$

If  $P(\omega_1|\tilde{\mathbf{x}}) > P(\omega_2|\tilde{\mathbf{x}})$ ,  $\tilde{\mathbf{x}}$  is classified to  $\omega_1$

If  $P(\omega_1|\tilde{\mathbf{x}}) < P(\omega_2|\tilde{\mathbf{x}})$ ,  $\tilde{\mathbf{x}}$  is classified to  $\omega_2$

# Inferência Bayesiana

- Tanto ML quanto MAP retornam apenas um único valor (maximização de uma função) para o conjunto de parâmetros  $\Theta$ ;
- Já na inferência Bayesiana, calcula-se a distribuição de probabilidade completa  $p(\Theta|X)$ ;
- Denominador não pode ser desprezado agora:

$$p(\boldsymbol{\theta}|X) = \frac{p(X|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(X)} = \frac{p(X|\boldsymbol{\theta})p(\boldsymbol{\theta})}{\int p(X|\boldsymbol{\theta})p(\boldsymbol{\theta}) d\boldsymbol{\theta}}$$

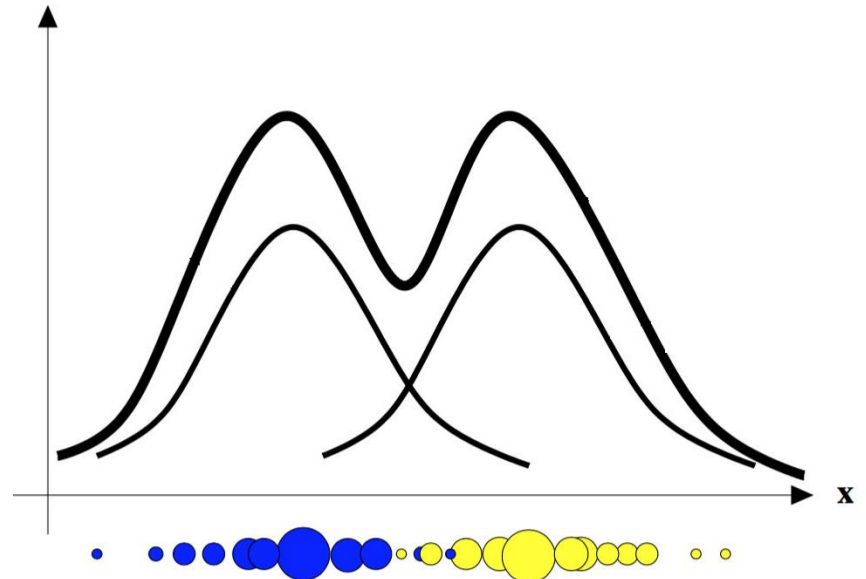
- Sugestão de leitura: *ML, MAP, and Bayesian — The Holy Trinity of Parameter Estimation and Data Prediction*. Autor: Avinash Kak .

# Combinação Linear de PDFs

- Uma forma alternativa ao apresentado até aqui é utilizar uma combinação linear de  $K$  PDFs para representar  $p(\mathbf{x})$ :

$$p(\mathbf{x}) = \sum_{k=1}^K \pi_k \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k).$$

$$\sum_{k=1}^K \pi_k = 1$$



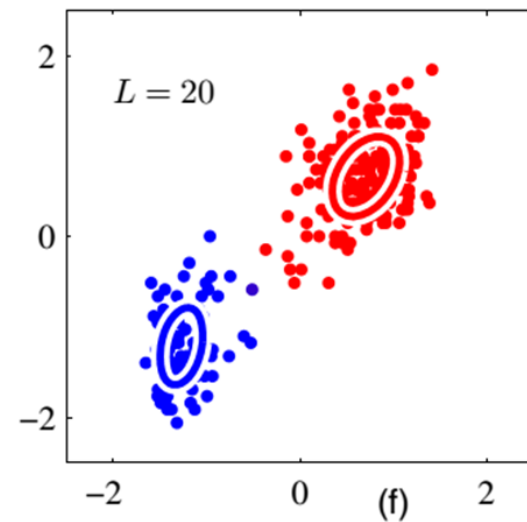
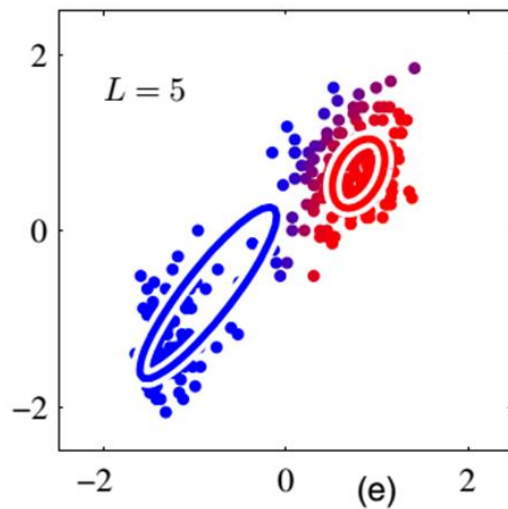
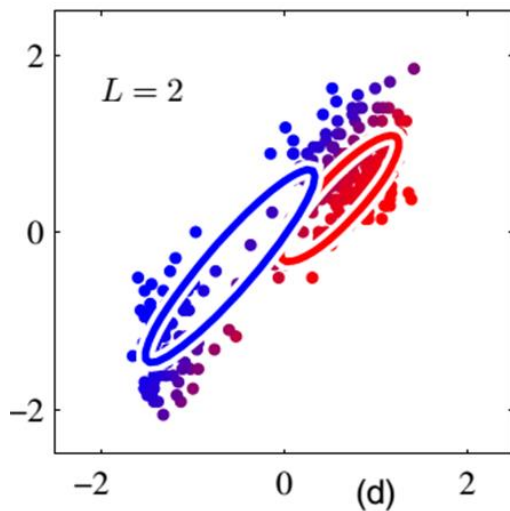
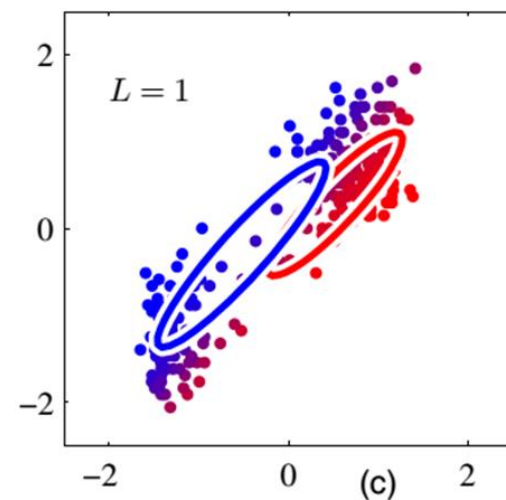
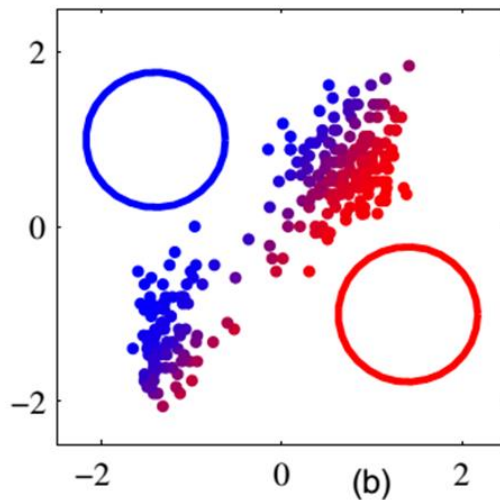
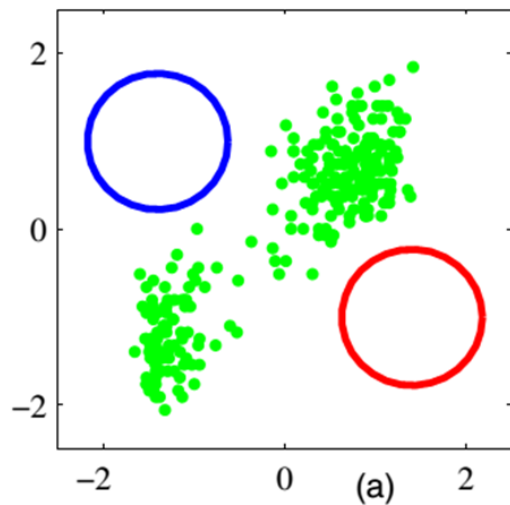
# Combinação Linear de PDFs

- No caso de uma combinação linear de Gaussianas (Mistura de Gaussianas), uma ideia seria aplicar o método MLE (por exemplo):

$$\ln p(\mathbf{X}|\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \sum_{n=1}^N \ln \left\{ \sum_{k=1}^K \pi_k \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right\}$$

- O problema é que a formulação acima resulta em um problema de otimização não-linear, por vezes, muito difícil de resolver.
- Existem alternativas...

# Maximização da Expectativa (EM)




# Mistura de Gaussianas

## Maximização da Expectativa

### EM (Expectation Maximization Algorithm)

1. Initialize the means  $\mu_k$ , covariances  $\Sigma_k$  and mixing coefficients  $\pi_k$ , and evaluate the initial value of the log likelihood.
2. **E step.** Evaluate the responsibilities using the current parameter values

**Importante!** 

$$\gamma(z_{nk}) = \frac{\pi_k \mathcal{N}(\mathbf{x}_n | \mu_k, \Sigma_k)}{\sum_{j=1}^K \pi_j \mathcal{N}(\mathbf{x}_n | \mu_j, \Sigma_j)}.$$

3. **M step.** Re-estimate the parameters using the current responsibilities

$$\begin{aligned}\mu_k^{\text{new}} &= \frac{1}{N_k} \sum_{n=1}^N \gamma(z_{nk}) \mathbf{x}_n \\ \Sigma_k^{\text{new}} &= \frac{1}{N_k} \sum_{n=1}^N \gamma(z_{nk}) (\mathbf{x}_n - \mu_k^{\text{new}}) (\mathbf{x}_n - \mu_k^{\text{new}})^T \\ \pi_k^{\text{new}} &= \frac{N_k}{N}\end{aligned}$$

where

$$N_k = \sum_{n=1}^N \gamma(z_{nk}).$$

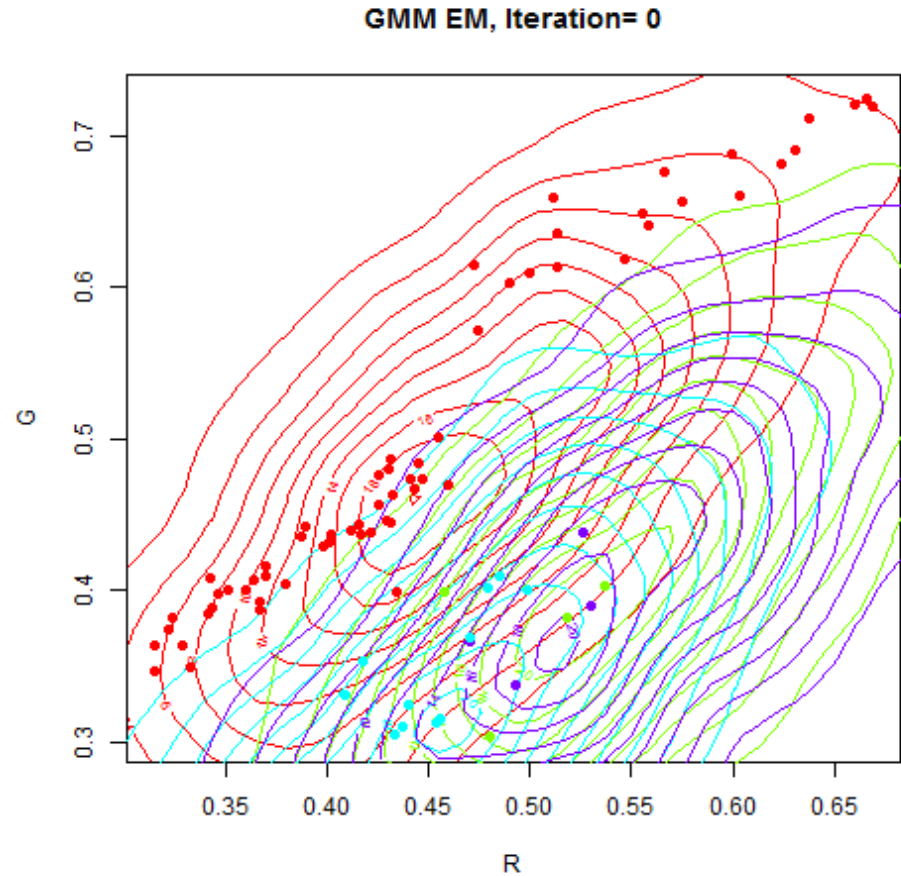
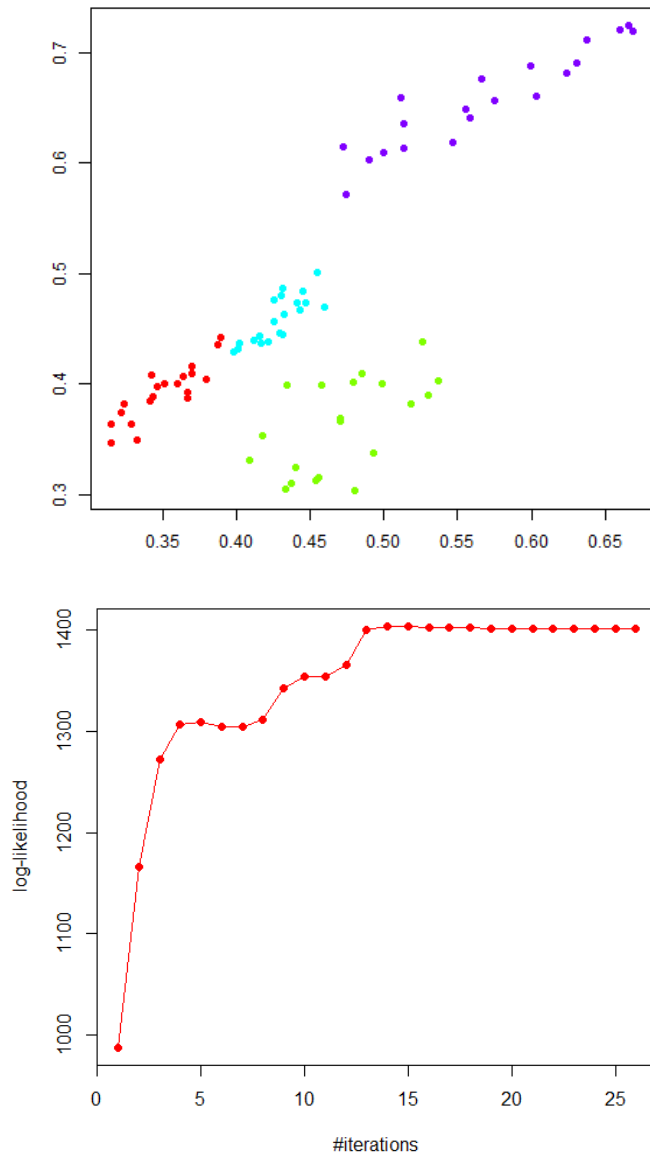
4. Evaluate the log likelihood  **Sempre crescente**

$$\ln p(\mathbf{X} | \mu, \Sigma, \pi) = \sum_{n=1}^N \ln \left\{ \sum_{k=1}^K \pi_k \mathcal{N}(\mathbf{x}_n | \mu_k, \Sigma_k) \right\}$$

and check for convergence of either the parameters or the log likelihood. If the convergence criterion is not satisfied return to step 2.

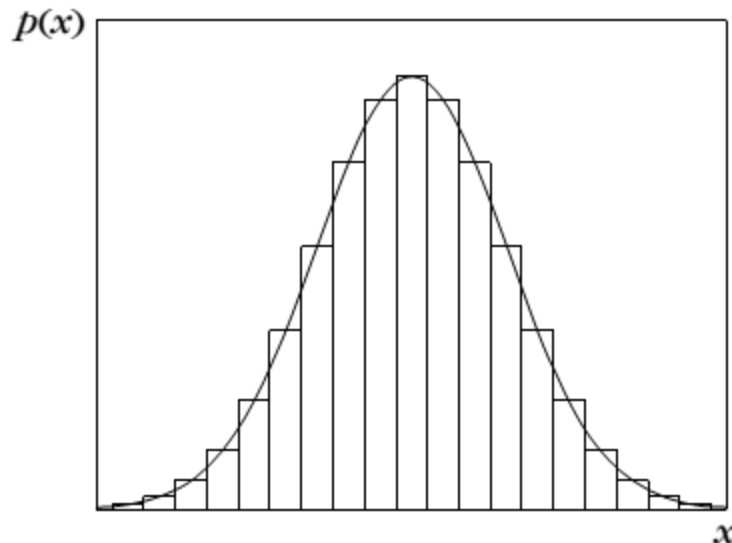


# Maximização da Expectativa (EM)



# Estimando PDFs não-paramétricas

- Até então assumimos uma pdf e estimamos os parâmetros;
- Agora passaremos para uma abordagem não-paramétrica: não estamos assumindo uma pdf prévia  $\rightarrow$  histogramas;



$$\hat{p}(x) \equiv \hat{p}(\hat{x}) \approx \frac{1}{b} \frac{k_N}{N}$$

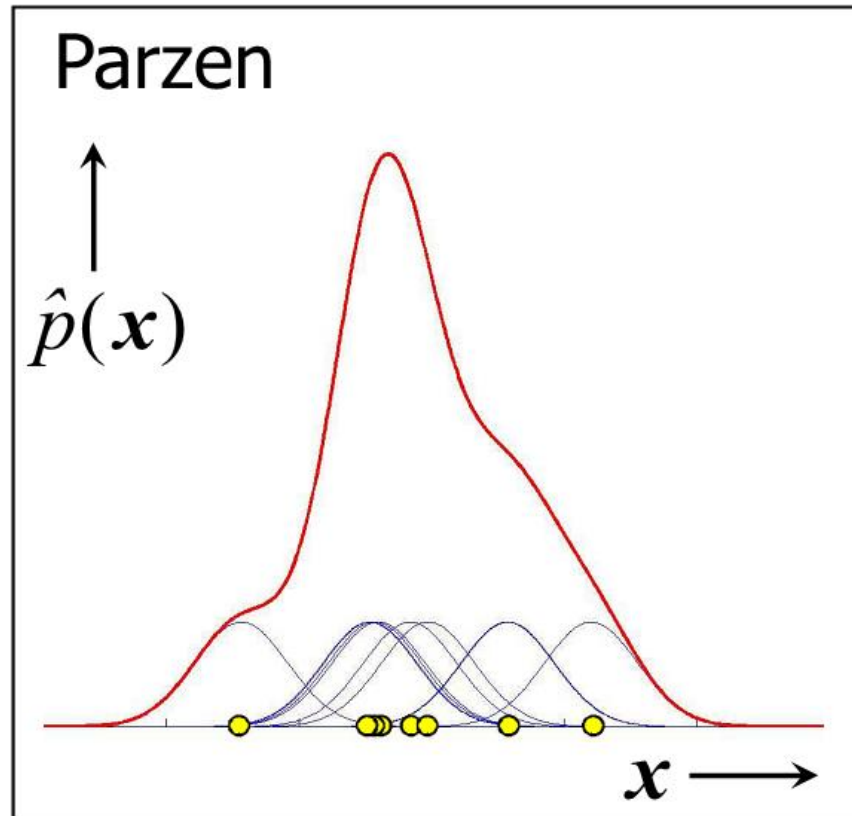
$$b_N \rightarrow 0$$

$$\hat{p}(x) \text{ converges } k_N \rightarrow \infty$$

$$\frac{k_N}{N} \rightarrow 0$$

# Janelas de Parzen

- Suavizando (Gaussianas):



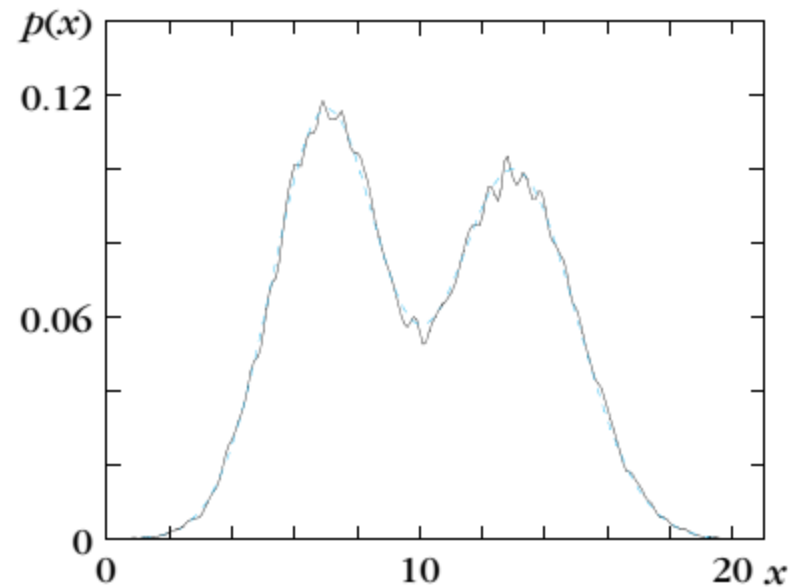
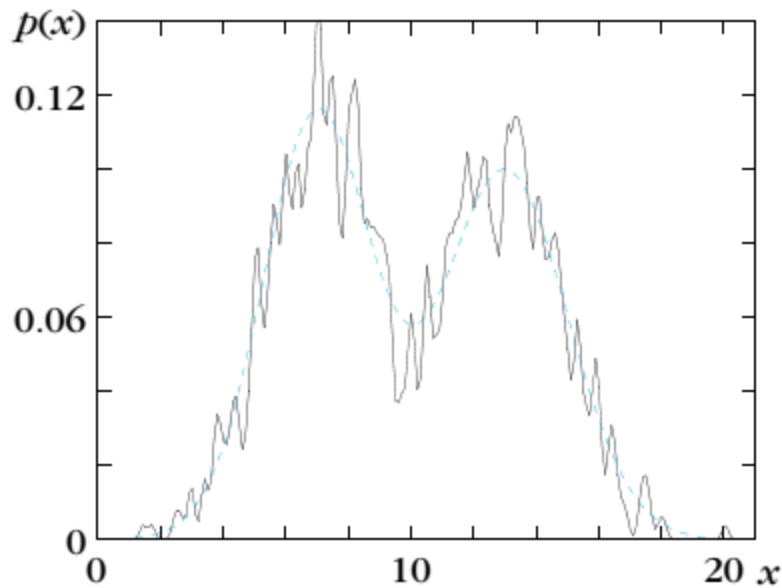
Suavizando  
(Gaussianas):

$$\hat{p}(\mathbf{x}) = \frac{1}{N} \sum_{i=1}^N \frac{1}{(2\pi)^{\frac{l}{2}} h^l} \exp\left(-\frac{(\mathbf{x} - \mathbf{x}_i)^T (\mathbf{x} - \mathbf{x}_i)}{2h^2}\right)$$

# Janelas de Parzen

- Exemplo:

$h = 0.8$  and 20,000 samples



$h = 0.8$  and 1,000 training samples

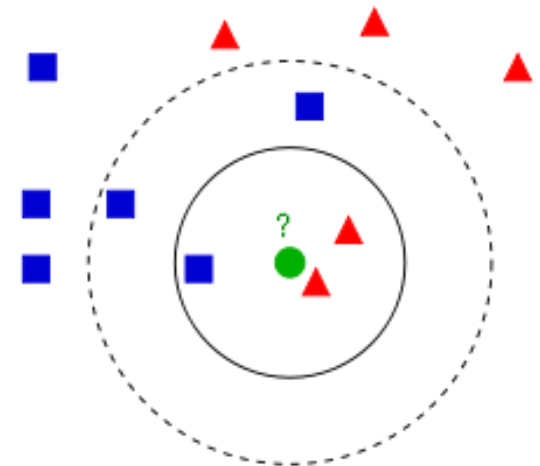
# Estimativa K-Vizinhos Mais Próximos para PDF

- Janela de Parzen: o volume (h) era fixo e o número de pontos no interior do volume variava;
- K-vizinhos-mais-próximos: o volume é variável e o número de pontos é fixo. Portanto:

$$\hat{p}(\mathbf{x}) = \frac{k}{NV(\mathbf{x})}$$

$$V(\mathbf{x}) = \frac{\pi^{\frac{d}{2}}}{\Gamma\left(\frac{d}{2} + 1\right)} R^d$$

- Não muito usual!
- Regra dos k vizinhos:
  - Identifique os k-vizinhos-mais-próximos do padrão a ser classificado;
  - Identifique a classe majoritária dentre os vizinhos selecionados e atribua a classe correspondente.




# Classificador Naive Bayes

- Com o objetivo de ter uma boa estimativa das pdfs, o número de exemplos de treinamento deve ser grande o suficiente;
- Se  $N$  for o suficiente para um caso unidimensional, no caso  $l$ -dimensional seria necessário  $N^l$
- Assume-se que as características são estatisticamente independentes:

$$p(\mathbf{x}|\omega_i) = \prod_{j=1}^l p(x_j|\omega_i), \quad i = 1, 2, \dots, M$$

MLE para cada distribuição 1-D  
(para cada  $l$ )



$$\omega_m = \arg \max_{\omega_i} \prod_{j=1}^l p(x_j|\omega_i), \quad i = 1, 2, \dots, M$$

# Referências

- **Capt. 2 - Livro Theodoridis (Pattern Recognition Fourth Edition e Pattern Recognition Matlab);**
- Capt. 1 - Livro Bishop – somente para Misturas de Gaussianas;
- Avinash Kak (Purdue University). ML, MAP, and Bayesian — The Holy Trinity of Parameter Estimation and Data Prediction.
- Aulas Professor Nando de Freitas (UBC/Oxford – ML 2013) – Aulas 1, 2, 3, 6 e 7.