

## Fonctions discriminantes linéaires

1

### • Objectif

- déterminer **directement** les fonctions discriminantes
- **linéaires**:  $g(\mathbf{x}) = w_0 + \sum_{i=1}^d w_i x_i = \mathbf{w}'\mathbf{x} + w_0$
- linéaires **généralisées**:  $g(\mathbf{x}) = \sum_{i=1}^{\hat{d}} a_i y_i(\mathbf{x}) = \mathbf{a}'\mathbf{y}$
- en minimisant le **risque empirique**

## Fonctions discriminantes linéaires

2

### • Justifications

- parfois **optimal**
- **facile à calculer**
- candidates pour des **classifieurs initiales**
- **aborder** quelques **principes** importants

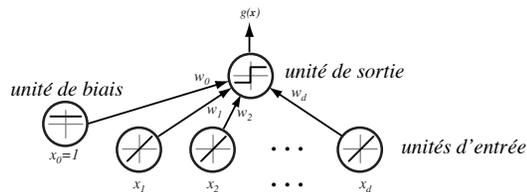
## Fonctions discriminantes linéaires

3

### • Géométrie – deux classes

- fonction de décision:

$$f(\mathbf{x}) = \begin{cases} C_1 & \text{si } g(\mathbf{x}) > 0, \\ C_2 & \text{si } g(\mathbf{x}) < 0 \end{cases} = \begin{cases} C_1 & \text{si } \mathbf{w}'\mathbf{x} > -w_0, \\ C_2 & \text{si } \mathbf{w}'\mathbf{x} < -w_0 \end{cases}$$



## Fonctions discriminantes linéaires

4

### • Géométrie – deux classes

- **frontière** de décision  $H$  est un **hyperplan**:  $g(\mathbf{x}) = 0$
- $\mathbf{x}_1, \mathbf{x}_2 \in H$ :  $\mathbf{w}'(\mathbf{x}_1 - \mathbf{x}_2) = 0$
- **régions** de décision:  $R_1$ : coté **positif**,  $R_2$ : coté **négatif**
- $r =$  **distance algébrique** de  $\mathbf{x}$  et  $H$ :

$$\mathbf{x} = \mathbf{x}_p + r \frac{\mathbf{w}}{\|\mathbf{w}\|}$$

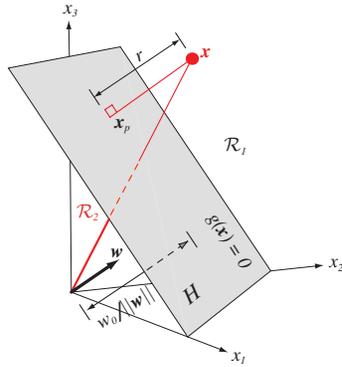
$$g(\mathbf{x}) = \mathbf{w}'\mathbf{x} + w_0 = r\|\mathbf{w}\|$$

$$r = \frac{g(\mathbf{x})}{\|\mathbf{w}\|}$$

## Fonctions discriminantes linéaires

5

### • Géométrie – deux classes

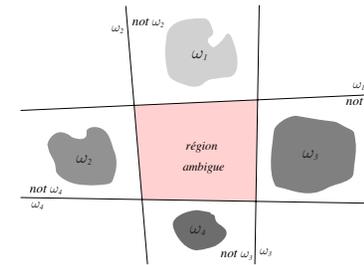


## Fonctions discriminantes linéaires

6

### • Géométrie – multiclass

#### • $C_i / \text{non}C_i$

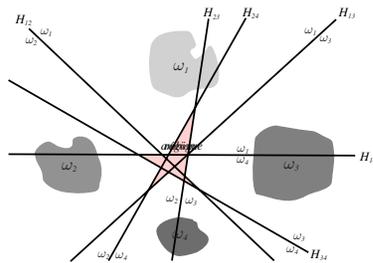


## Fonctions discriminantes linéaires

7

### • Géométrie – multiclass

#### • $N(N-1)/2$ fonctions discriminantes



## Fonctions discriminantes linéaires

8

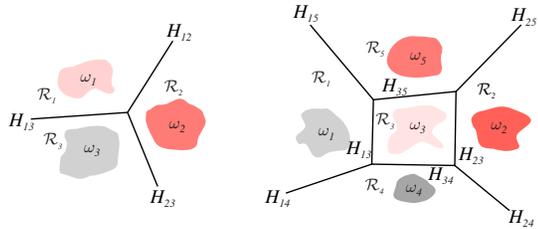
### • Fonctions discriminantes linéaires

- **machine linéaire**:  $g_j(\mathbf{x}) = \mathbf{w}_j^T \mathbf{x} + w_{j0}$ ,  $j = 1, \dots, N$
- **frontières** de décision  $H_{i,j}; g_i(\mathbf{x}) = g_j(\mathbf{x})$
- $(\mathbf{w}_i - \mathbf{w}_j)$  est **orthogonal** à  $H_{i,j}$
- $r(\mathbf{x}, H_{i,j}) = \frac{g_i(\mathbf{x}) - g_j(\mathbf{x})}{\|\mathbf{w}_i - \mathbf{w}_j\|}$

## Fonctions discriminantes linéaires

9

- Fonctions discriminantes linéaires



## Fonctions discriminantes linéaires

10

- Fonctions discriminantes linéaires généralisées:

$$g(\mathbf{x}) = \sum_{i=1}^{\hat{d}} a_i y_i(\mathbf{x}) = \mathbf{a}^t \mathbf{y}$$

- exemple: fonction discriminante quadratique:

$$g(\mathbf{x}) = w_0 + \sum_{i=1}^d w_i x_i + \sum_{i=1}^d \sum_{j=1}^d w_{ij} x_i x_j$$

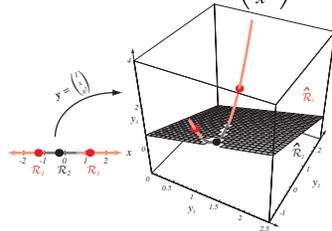
- frontière de décision: hyperquadrique

## Fonctions discriminantes linéaires

11

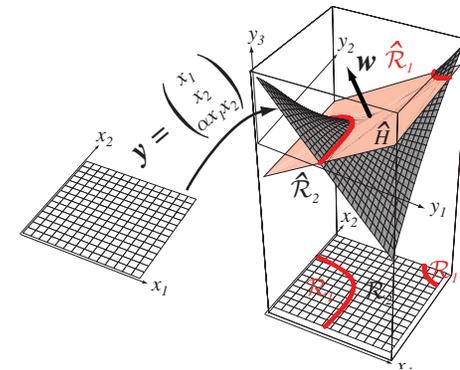
- Fonctions discriminantes linéaires généralisées

- exemple:  $g(x) = a_1 + a_2 x + a_3 x^2$ ,  $\mathbf{y} = \begin{pmatrix} 1 \\ x \\ x^2 \end{pmatrix}$



12

- exemple:  $\mathbf{y} = \begin{pmatrix} x_1 \\ x_2 \\ \alpha x_1 x_2 \end{pmatrix}$

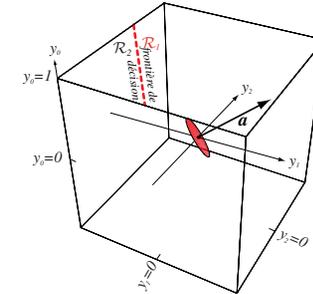


• Vecteur augmenté

•  $g(\mathbf{x}) = w_0 + \sum_{i=1}^d w_i x_i = \sum_{i=0}^d w_i x_i \quad (x_0 = 1)$

•  $g(\mathbf{x}) = \sum_{i=1}^{\hat{d}} a_i y_i, \hat{d} = d + 1, \mathbf{y} = \begin{pmatrix} 1 \\ x_1 \\ \vdots \\ x_d \end{pmatrix}, \mathbf{a} = \begin{pmatrix} w_0 \\ w_1 \\ \vdots \\ w_d \end{pmatrix} = \begin{pmatrix} w_0 \\ \mathbf{w} \end{pmatrix}$

• Vecteur augmenté



• Séparabilité linéaire

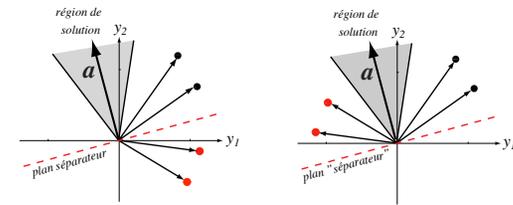
•  $D_n = ((\mathbf{y}_1, z_1), \dots, (\mathbf{y}_n, z_n)), z_i = \begin{cases} 1 & \text{si } \mathbf{y}_i \text{ est classifié } C_1 \\ -1 & \text{si } \mathbf{y}_i \text{ est classifié } C_2 \end{cases}$

•  $g(\mathbf{x}) = \mathbf{a}'\mathbf{y}$  sépare  $D_n$  sans erreur:

$\mathbf{a}'\mathbf{y}_i z_i > 0, \quad i = 1, \dots, n$

•  $\mathbf{a}$ : vecteur séparateur, vecteur de solution

• Séparabilité linéaire



- **Marge** de séparation:

$$m_i = g(\mathbf{x}_i)z_i = \mathbf{a}^t \mathbf{y}_i z_i$$

- Séparation avec une **marge**  $b$ :

$$m_i = \mathbf{a}^t \mathbf{y}_i z_i > b, \quad i = 1, \dots, n$$

- Procédures de **descente de gradient**

- fonction de **critère**:  $J(\mathbf{a})$  – minimisée si  $\mathbf{a}$  est une solution

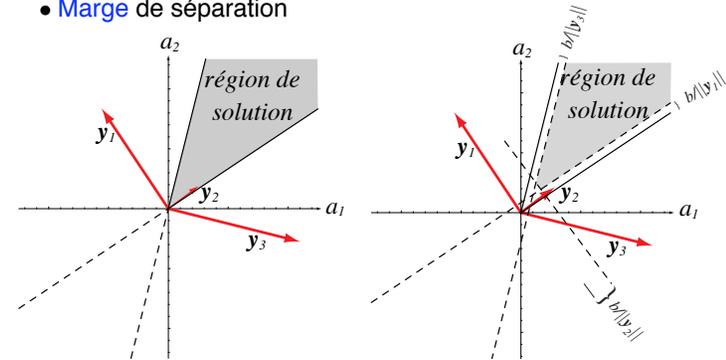
- $\mathbf{a}(k+1) = \mathbf{a}(k) - \eta(k) \nabla J(\mathbf{a}(k))$

- $\eta(k)$ : **taux d'apprentissage**

DESCENTE DE GRADIENT SIMPLE( $\Theta, \eta(\cdot), \mathbf{a}_0$ )

- 1  $\mathbf{a} \leftarrow \mathbf{a}_0$
- 2  $k \leftarrow 0$
- 3 **faire**
- 4      $k \leftarrow k + 1$
- 5      $\mathbf{a} \leftarrow \mathbf{a} - \eta(k) \nabla J(\mathbf{a})$
- 6     **jusqu'à**  $|\eta(k) \nabla J(\mathbf{a})| < \Theta$
- 7     **retourner**  $\mathbf{a}$

- **Marge** de séparation



- Descente de **Newton**

- $J(\mathbf{a}) \simeq J(\mathbf{a}(k)) + \nabla J^t(\mathbf{a} - \mathbf{a}(k)) + \frac{1}{2}(\mathbf{a} - \mathbf{a}(k))^t \mathbf{H}(\mathbf{a} - \mathbf{a}(k))$

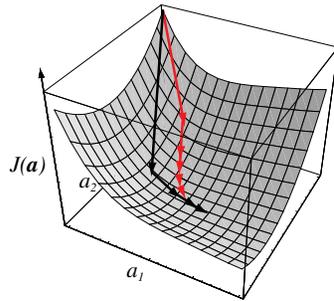
- matrice **hessienne**:  $\mathbf{H}_{ij} = \frac{\delta^2 J}{\delta a_i \delta a_j}$

- $\mathbf{a}(k+1) = \mathbf{a}(k) - \mathbf{H}^{-1} \nabla J$

DESCENTE DE NEWTON( $\Theta, \mathbf{a}_0$ )

- 1  $\mathbf{a} \leftarrow \mathbf{a}_0$
- 2 **faire**
- 3      $\mathbf{a} \leftarrow \mathbf{a} - \mathbf{H}^{-1} \nabla J(\mathbf{a})$
- 4     **jusqu'à**  $|\mathbf{H}^{-1} \nabla J(\mathbf{a})| < \Theta$
- 5     **retourner**  $\mathbf{a}$

- Descente de **Newton**



- Le **perceptron**

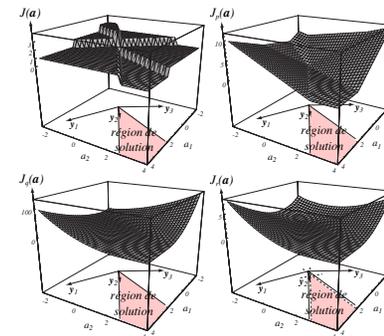
- $J_p(\mathbf{a}) = \sum_{i=1}^n I_{\{\mathbf{a}'\mathbf{y}_i z_i \leq 0\}} (-\mathbf{a}'\mathbf{y}_i z_i)$
- $\nabla J_p = \sum_{i=1}^n I_{\{\mathbf{a}'\mathbf{y}_i z_i \leq 0\}} (-\mathbf{y}_i z_i)$
- $\mathbf{a}(k+1) = \mathbf{a}(k) + \eta(k) \sum_{i=1}^n I_{\{\mathbf{a}'\mathbf{y}_i z_i \leq 0\}} \mathbf{y}_i z_i$

- Le **perceptron**

```

PERCEPTRONBATCH( $\Theta, \eta(\cdot), \mathbf{a}_0$ )
1   $\mathbf{a} \leftarrow \mathbf{a}_0$ 
2   $k \leftarrow 0$ 
3  faire
4     $k \leftarrow k + 1$ 
5     $\mathbf{a} \leftarrow \mathbf{a} + \eta(k) \sum_{i=1}^n I_{\{\mathbf{a}'\mathbf{y}_i z_i \leq 0\}} \mathbf{y}_i z_i$ 
6  jusqu'à  $|\eta(k) \sum_{i=1}^n I_{\{\mathbf{a}'\mathbf{y}_i z_i \leq 0\}} \mathbf{y}_i z_i| < \Theta$ 
7  retourner  $\mathbf{a}$ 
    
```

- Fonctions de **critère**



- Le perceptron en-ligne

```

PERCEPTRONENLIGNE(a0)
1  a ← a0
2  k ← 0
3  faire
4    k ← (k + 1) mod n
5    si a'ykzk ≤ 0 alors      ▷ yk mal classifié
6      a ← a + ykzk
7  jusqu'à ∑i=1n I{a'yizi ≤ 0} = 0      ▷ pas d'erreur
8  retourner a

```

- Théorème

- Si l'ensemble d'entraînement est linéairement séparable, l'algorithme PERCEPTRONENLIGNE se termine à un vecteur de solution après un nombre fini de corrections.

- Le perceptron en-ligne, avec marge, d'incrément variable

```

PERCEPTRONENLIGNEMARGEVARIABLE(η(·), a0, b)
1  a ← a0
2  k ← 0
3  faire
4    k ← k + 1
5    k' ← k mod n
6    si a'yk'zk' ≤ b alors
7      a ← a + η(k)yk'zk'
8  jusqu'à ∑i=1n I{a'yizi ≤ b} = 0      ▷ pas d'erreur
    par rapport à la marge b
9  retourner a

```

- Conditions de convergence

- $\eta(k) \geq 0$
- $\lim_{m \rightarrow \infty} \sum_{k=1}^m \eta(k) = \infty$
- $\lim_{m \rightarrow \infty} \frac{\sum_{k=1}^m \eta^2(k)}{(\sum_{k=1}^m \eta(k))^2} = 0$

- Le perceptron batch d'incrément variable

$$\mathbf{y}^{(k)} = \sum_{i=1}^n I_{\{a^{(k)}y_i z_i \leq 0\}} y_i z_i$$

```

PERCEPTRONBATCHVARIABLE(η(·), a0)
1  a ← a0
2  k ← 0
3  faire
4    k ← k + 1
5    a ← a + η(k) ∑i=1n I{a'yizi ≤ 0} yizi
6  jusqu'à ∑i=1n I{a'yizi ≤ 0} = 0
7  retourner a

```

• Procédures de **relaxation**

- $J_q(\mathbf{a}) = \sum_{i=1}^n I_{\{\mathbf{a}'\mathbf{y}_i z_i \leq 0\}} (\mathbf{a}'\mathbf{y}_i z_i)^2$
- $J_r(\mathbf{a}) = \frac{1}{2} \sum_{i=1}^n I_{\{\mathbf{a}'\mathbf{y}_i z_i \leq b\}} \frac{(\mathbf{a}'\mathbf{y}_i z_i - b)^2}{\|\mathbf{y}_i z_i\|^2}$
- $\nabla J_r = \sum_{i=1}^n I_{\{\mathbf{a}'\mathbf{y}_i z_i \leq b\}} \frac{\mathbf{a}'\mathbf{y}_i z_i - b}{\|\mathbf{y}_i z_i\|^2} \mathbf{y}_i z_i$
- $\mathbf{a}(k+1) = \mathbf{a}(k) + \eta(k) \sum_{i=1}^n I_{\{\mathbf{a}'\mathbf{y}_i z_i \leq b\}} \frac{b - \mathbf{a}'\mathbf{y}_i z_i}{\|\mathbf{y}_i z_i\|^2} \mathbf{y}_i z_i$

• Relaxation **en-ligne**

```

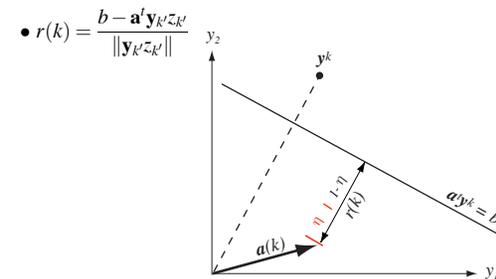
RELAXATIONENLIGNEMARGE( $\eta(\cdot), \mathbf{a}_0, b$ )
1   $\mathbf{a} \leftarrow \mathbf{a}_0$ 
2   $k \leftarrow 0$ 
3  faire
4       $k \leftarrow k + 1$ 
5       $k' \leftarrow k \bmod n$ 
6      si  $\mathbf{a}'\mathbf{y}_{k'} z_{k'} \leq b$  alors
7           $\mathbf{a} \leftarrow \mathbf{a} + \eta(k) \frac{b - \mathbf{a}'\mathbf{y}_{k'} z_{k'}}{\|\mathbf{y}_{k'} z_{k'}\|^2} \mathbf{y}_{k'} z_{k'}$ 
8  jusqu'à  $\sum_{i=1}^n I_{\{\mathbf{a}'\mathbf{y}_i z_i \leq b\}} = 0$ 
9  retourner  $\mathbf{a}$ 
    
```

• Procédures de **relaxation**

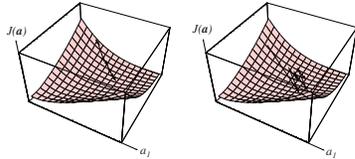
```

RELAXATIONBATCHMARGE( $\eta(\cdot), \mathbf{a}_0, b$ )
1   $\mathbf{a} \leftarrow \mathbf{a}_0$ 
2   $k \leftarrow 0$ 
3  faire
4       $k \leftarrow k + 1$ 
5       $\mathbf{a} \leftarrow \mathbf{a} + \eta(k) \sum_{i=1}^n I_{\{\mathbf{a}'\mathbf{y}_i z_i \leq b\}} \frac{b - \mathbf{a}'\mathbf{y}_i z_i}{\|\mathbf{y}_i z_i\|^2} \mathbf{y}_i z_i$ 
6  jusqu'à  $\sum_{i=1}^n I_{\{\mathbf{a}'\mathbf{y}_i z_i \leq b\}} = 0$ 
7  retourner  $\mathbf{a}$ 
    
```

• Relaxation **en-ligne**



- Relaxation **en-ligne**
  - $\eta > 1$ : **sur**-relaxation
  - $\eta < 1$ : **sous**-relaxation
  - condition de **convergence**:  $0 < \eta < 2$



- Incrément fixe
  - boucle **infinie**
  - engendre un processus d'**état fini**
  - **moyenner** les vecteurs de poids
- Incrément variable
  - **converge** si  $\eta(k) \rightarrow 0$

- Comportement dans le cas **non-séparable**
  - procédures de **correction d'erreur**
  - fonctionnent **bien** si
    - la décision de Bayes est **à peu près linéaire**
    - l'erreur de Bayes est **petite**
  - si  $2\hat{d} > n$ , la probabilité de non-séparabilité est petite

- L'approche d'erreur carrée (régression)
  - soit  $\mathbf{b} = (z_1, \dots, z_n)'$
  - Idéalement on voudrait trouver  $\mathbf{a}$  tel que  $\mathbf{Y}\mathbf{a} = \mathbf{b}$
  - Mais on commet des erreurs  $\mathbf{e} = \mathbf{Y}\mathbf{a} - \mathbf{b}$
  - $J_s(\mathbf{a}) = \|\mathbf{Y}\mathbf{a} - \mathbf{b}\|^2 = \sum_{i=1}^n (\mathbf{a}'\mathbf{y}_i - b_i)^2$
  - $\nabla J_s(\mathbf{a}) = \sum_{i=1}^n 2(\mathbf{a}'\mathbf{y}_i - b_i)\mathbf{y}_i = 2\mathbf{Y}'(\mathbf{Y}\mathbf{a} - \mathbf{b})$
  - $\mathbf{Y}'\mathbf{Y}\mathbf{a} = \mathbf{Y}'\mathbf{b}$
  - $\mathbf{a} = (\mathbf{Y}'\mathbf{Y})^{-1}\mathbf{Y}'\mathbf{b} = \mathbf{Y}^+\mathbf{b}$

- $\mathbf{Y}^\dagger = (\mathbf{Y}'\mathbf{Y})^{-1}\mathbf{Y}'$ : pseudoinverse de  $\mathbf{Y}$

37

## Fonctions discriminantes linéaires

38

- Procédure de **Widrow-Hoff (LMS)**

- **batch**:  $\mathbf{a}(k+1) = \mathbf{a}(k) + \eta(k)\mathbf{Y}'(\mathbf{b} - \mathbf{Y}\mathbf{a}(k))$
- **en ligne**:  $\mathbf{a}(k+1) = \mathbf{a}(k) + \eta(k)\mathbf{y}_{k'}(b_{k'} - \mathbf{a}'\mathbf{y}_{k'})$

| LMS( $\Theta, \eta(\cdot), \mathbf{a}_0$ ) |  |
|--|--|
| 1  | $\mathbf{a} \leftarrow \mathbf{a}_0$   |
| 2  | $k \leftarrow 0$   |
| 3  | <b>faire</b>   |
| 4  | $k \leftarrow k + 1$   |
| 5  | $k' \leftarrow k \bmod n$  |
| 6  | $\mathbf{a} \leftarrow \mathbf{a} + \eta(k)\mathbf{y}_{k'}(b_{k'} - \mathbf{a}'\mathbf{y}_{k'})$ |
| 7  | <b>jusqu'à</b> $ \eta(k)\mathbf{y}_{k'}(b_{k'} - \mathbf{a}'\mathbf{y}_{k'})  < \Theta$          |
| 8  | <b>retourner</b> $\mathbf{a}$  |

## Fonctions discriminantes linéaires

39

- Procédure de **Widrow-Hoff (LMS)**

- se comporte **bien** dans le cas **non-séparable**
- **ne converge pas nécessairement** à un hyperplan séparateur dans les cas séparables

## Fonctions discriminantes linéaires

40

- La machine de **support vector (SVM)**

- objectif: trouver un hyperplan séparateur avec une **grande marge**  
 $z_i g(\mathbf{y}_i) = z_i \mathbf{a}' \mathbf{y}_i$

- maximiser  $b$ :  $\frac{z_i g(\mathbf{y}_i)}{\|\mathbf{a}\|} \geq b \quad i = 1, \dots, n$

