Fonctions discriminantes linéaires

Christine Sénac (slides manuscrits de R.André-Obrecht)
M1 Informatique

Fonctions discriminantes linéaires et non linéaires

$$\mathcal{Y} = \{y_1, \dots y_N\} \quad y_n \in \mathbb{R}^d$$

Observations

$$K = \{k_1, \dots, k_{|K|}\} \quad \text{classes}$$

$$g_i \quad \mathcal{Y} \to \mathbb{R}^+$$

1 fonction discriminante g_i par classe

$$y \in k_i \iff g_i(y) \geq g_j(y) \quad i \neq j$$

Problème à 2 classes

$$g \quad \mathcal{Y} \to \mathbb{R}^*$$

$$y \in k_1 \iff g(y) > 0$$

$$y \in k_2 \iff g(y) < 0$$

Posé par définition

Classification Binaire

- Données : quelques éléments (textes) qui appartiennent à deux classes différentes
  - classe 1 (+1) et classe 2 (-1)
  - classe positive (+1) et classe négative (-1)
- Tâche : entrainer un classifieur sur ces données (dites d'apprentissage) puis prédire la classe d'un nouvel élément (nouveau texte)
- Géométriquement : trouver une séparation entre les deux classes

Séparation Linéaire / Non Linéaire

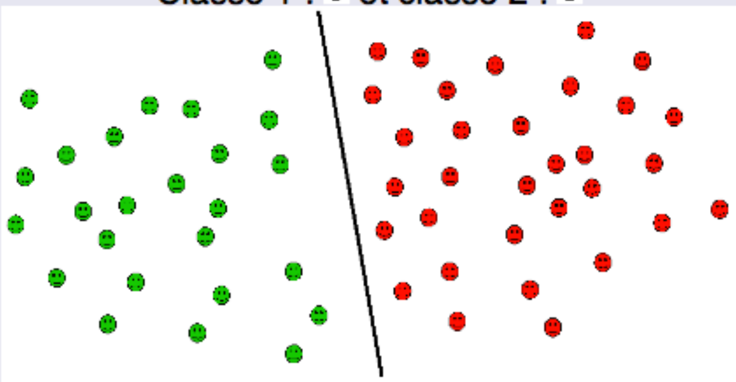
Données séparables linéairement

Si tous les points associés aux données peuvent être séparés correctement par une frontière linéaire (**hyperplan**)

Données séparables linéairement

Exemple : séparation linéaire

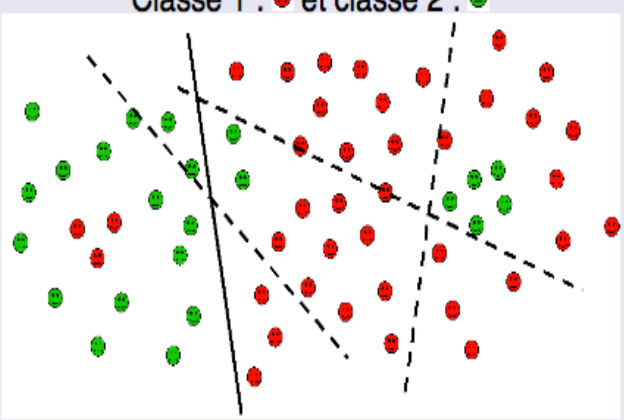
Classe 1 : ● et classe 2 : ●



Données non séparables linéairement

Exemple : pas de séparation linéaire

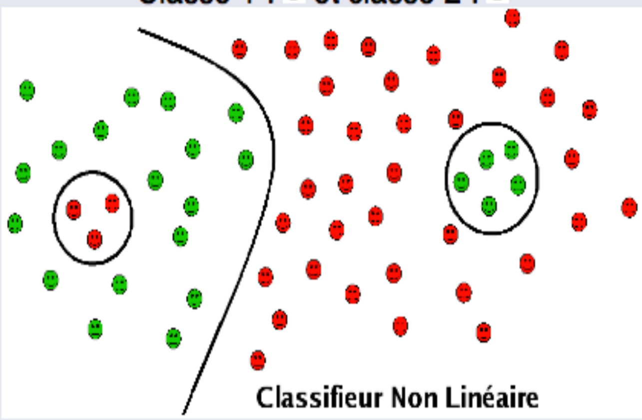
Classe 1 : ● et classe 2 : ●



Données non séparables linéairement

Exemple : séparation non linéaire

Classe 1 : ● et classe 2 : ●



Cas de 2 classes

Fonctions discriminantes linéaires

$$g(y) = w^t \bar{y}$$

avec $w = \begin{vmatrix} w_1 \\ \vdots \\ w_{d+1} \end{vmatrix}$ $\quad \bar{y} = \begin{vmatrix} y^1 \\ y^2 \\ \vdots \\ y^d \\ 1 \end{vmatrix}$

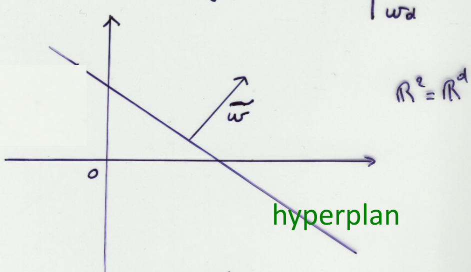
Frontière de décision :

$$\{ y \in \mathbb{R}^d \ / \ g(y) = 0 \}$$

hyperplan de $\mathbb{R}^d$

orthogonal à $\tilde{w} \begin{vmatrix} w_1 \\ \vdots \\ w_d \end{vmatrix}$

Exemple dans R^2



$$\sum_{i=1}^{d} w_i y^i + w_{d+1} = 0$$

Equation de l'hyperplan

Rappel :

$$y = y_p + r \frac{\tilde{w}}{\|\tilde{w}\|}$$

alors $$r = \frac{g(y)}{\|\tilde{w}\|}$$

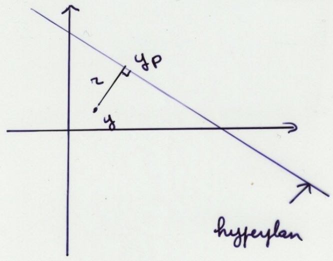
(distance algébrique à l'hyperplan)

dms

$$g(y) - g(y_p)$$

$$= \hat{w}^t . (y - y_p)$$

$$= \hat{w}^t . r \frac{\tilde{w}}{\|\tilde{w}\|}$$

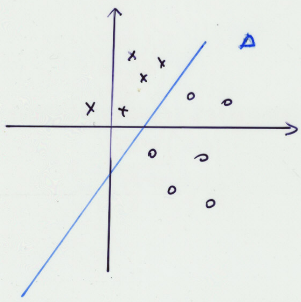


Exemple dans R^2

I Resolution du problème à 2 classes

→ neurone formel (1960 – 1980)

→ SVM (depuis 1990)



Recherche de l'hyperplan optimal
Existe –t-il?
Unicité?
⇒On cherche le vecteur
w orthogonal à l'hyperplan:
Par ap. séquentiel et sur les erreurs

Hypothèses = Ensemble d'échantillons
$\bar{y}$ appartenant à R_1 et R_2
Apprent. supervisé

w? | $w^t . \bar{y} > 0$ si $y \in R_1$
$w . \bar{y} < 0$ si $y \in R_2$

Sortie désirée
(connaissance =>
d'où ap. supervisé)

On pose $\delta_y = 1$ si $y \in R_1$
$\delta_y = -1$ si $y \in R_2$

w? . | $\delta_y w^t \bar{y} > 0$

$\delta_y \bar{y}$ = donnée normalisée

Cas: y est bien classé

I.1 Algorithme du Perceptron

Fonction coût (liée aux erreurs) à minimiser

$$J(w) = \sum_{\substack{y \in \mathcal{Y} \\ \text{mal classés}}} (-\delta_y w^t \bar{y})$$

Si y mal classé et $y \in \mathcal{R}_1$ $\Rightarrow$ $w^t . y < 0$, $\delta_y > 0$ $\Rightarrow -\delta_y w^t y > 0$

Si y mal classé et $y \in \mathcal{R}_2$ $\Rightarrow$ $w^t y > 0$, $\delta_y < 0$ $\Rightarrow -\delta_y w^t y > 0$

$J(w) > 0$ si erreur

$J(w) = 0$ sans erreur

Minimiser $J(w)$

Algorithme de Descente du gradient

$$\nabla J(w) = \sum_{\substack{y \in \mathcal{Y} \\ \text{mal classé}}} -\delta_y \bar{y}$$

$$w_{n+1} = w_n - \lambda_n \nabla J(w_n)$$

Règle du Perceptron (batch)

BATCH
L'hyperplan est mis à jour après traitement de TOUTES les données

$$w_{n+1} = w_n + \lambda_n \sum_{\substack{y \text{ mal classés} \\ (w_n)}} \delta_y \bar{y}$$

Convergence habituelle

$\sum \lambda_n = +\infty$

$\sum \lambda_n^2 < +\infty$

exple : $\lambda_n = \frac{c}{n}$

Théorème : Si les classes sont linéairement séparables, la règle de mise à jour converge en un nombre fini d'itérations ($\lambda_n = C$) ($\forall w_0$)

Minimum global ssi séparation linéaire

Algorithme de Rosenblatt (1958). (on line)

ON LINE
L'hyperplan est mis à jour après traitement de CHAQUE donnée

Itération sur les observations

$\mathcal{Y} = \{y_1, y_2 \cdots y_n, \quad y_N\}$.

Si y_n mal classé

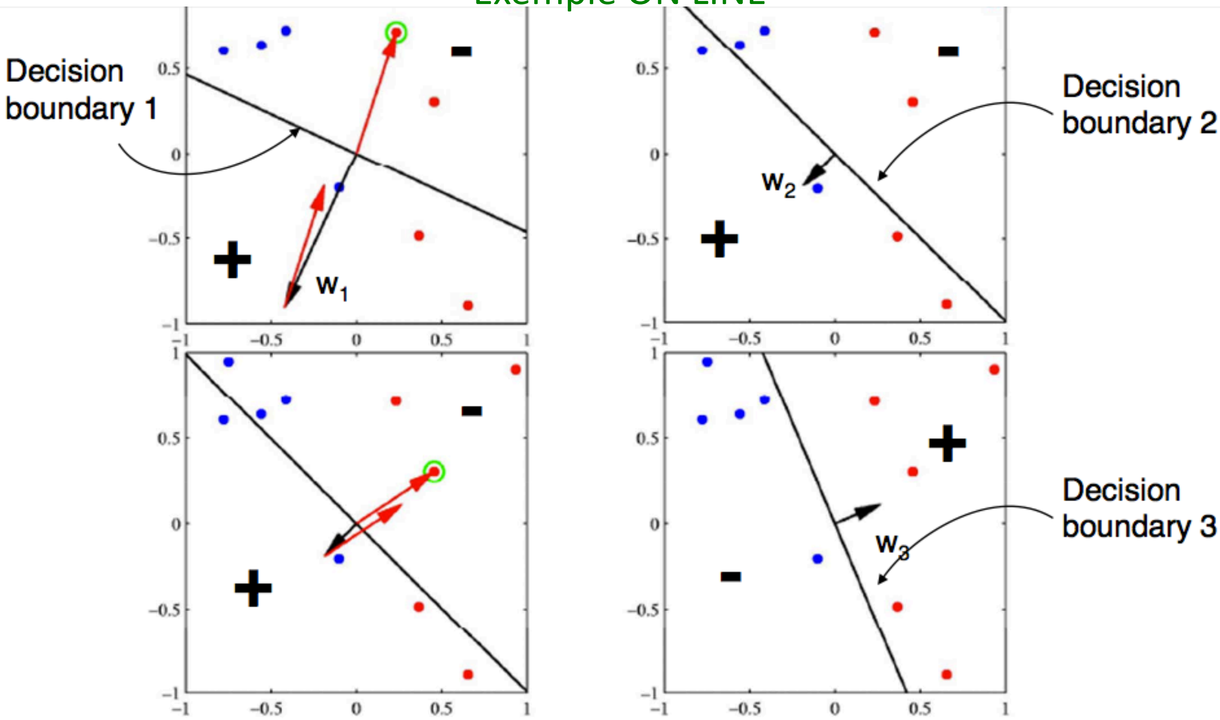
$w_{n+1} = w_n + \lambda_n \delta_{y_n} \bar{y}_n$

Sinon

$w_{n+1} = w_n$

(déplacement progressif de H).

Exemple ON LINE



Red points belong to the positive class,
blue points belong to the negative class

Batch Perceptron Algorithm

Given : training examples $(\mathbf{x}_i\ ,\ y_i)$, $i = 1,\dots,N$

Let $\mathbf{w} \leftarrow (0,0,0,\dots,0)$

do

$\quad delta \leftarrow (0,0,0,\dots,0)$

$\quad$ for $i = 1$ to N do

$\quad\quad u_i \leftarrow \mathbf{w} \cdot \mathbf{x}_i$

$\quad\quad$ if $y_i \cdot u_i \leq 0$

$\quad\quad\quad delta \leftarrow delta - y_i \cdot x_i$

$\quad delta \leftarrow delta / N$

$\quad \mathbf{w} \leftarrow \mathbf{w} - \eta\, delta$

until $|\,delta\,| < \varepsilon$

!!! Le pas η doit décroitre à chaque Itération de l'apprentissage

Simplest case: η = *1* and don't normalize – 'Fixed increment perceptron'

Algorithme du Perceptron

Perceptron : Analyse

- Progressif : s'adapte toujours aux nouvelles données
- Avantages
  - Simple et efficace
  - Garantie d'apprendre un problème linéairement séparable (convergence, optimum global)
- Limitations
  - Seulement séparations linéaires
  - Converge seulement pour données séparables
  - Pas très efficace dès qu'il y a trop de descripteurs

I 2 . Resolution avec marge

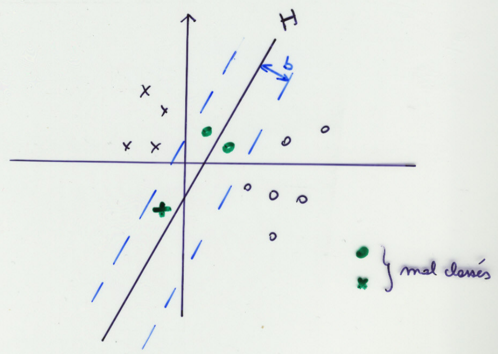
(procédures de relaxation)

$$J_r(w) = \frac{1}{2} \sum_{\substack{y \\ \text{mal classé}}} \left(\delta_y w^t \bar{y} - b \right)^2$$

($b > 0$)

y mal classé $\iff$ $\delta_y w^t \bar{y} < b$

Plus la marge est grande plus la classification est fiable



Algorithme du gradient

$$\nabla J = \sum_{y \text{ mal classé}} \delta_y \left(\delta_y w^t \bar{y} - b \right) \bar{y}$$

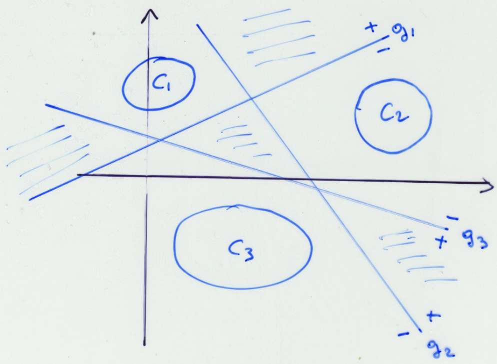
II Cas de plusieurs classes

m classes

1er cas 1 contre $(m-1)$

↳ m hyperplans g_i, $i = 1 \dots m$

$$y \in k_i \iff \begin{cases} g_i(y) > 0 \\ g_j(y) < 0 \quad \forall j \; j \neq i \end{cases}$$

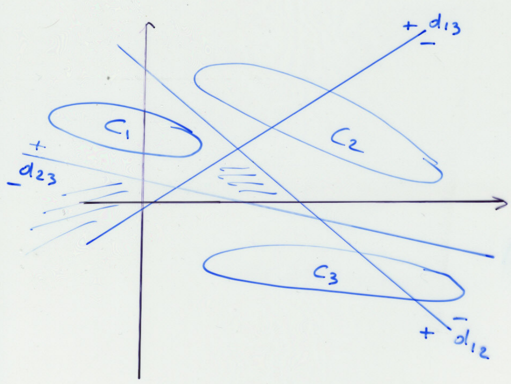


/// = indécision

2e cas 1 contre 1, 2 par 2 $\Big\}$ d_{ij}, $i = 1 \dots m$, $j = 1 \dots m$

$$\begin{matrix} y \in C_i \\ \text{et } y \notin C_j \end{matrix} \iff d_{ij}(y) > 0.$$

$$(\Rightarrow d_{ij} = - d_{ji}).$$



//// indécision

3^e cas $\quad$ m fonctions g_i

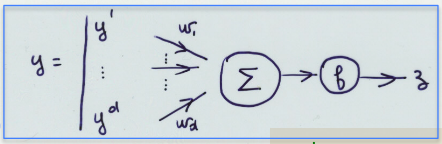
$$y \in C_i \iff g_i(y) > g_j(y)$$

- Pas de méthode derivée du "2 classes"
- Cas bayésien
- Pas de regions d'indécision.

III Reseaux de neurones Multicouches

III.1 Le Neurone formel

Cellule élémentaire (réseau à 2 couches)



z : 1 sortie booléenne

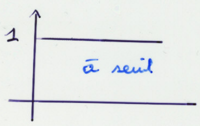
w^d : d poids

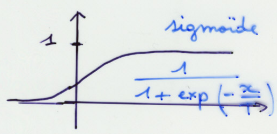
y^i : d entrées booléennes

f : fonction d'activation . Sert à introduire une non linéarité dans le fonctionnement du neurone

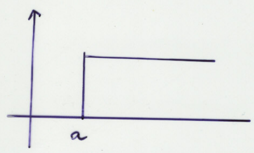
$$z = f\left(\sum_{i=1}^{d} w_i y^i\right)$$

$$z = f(\hat{w}^t . y)$$





$$\frac{1}{1+\exp\left(-\frac{x}{T}\right)}$$



ā seuil décalé

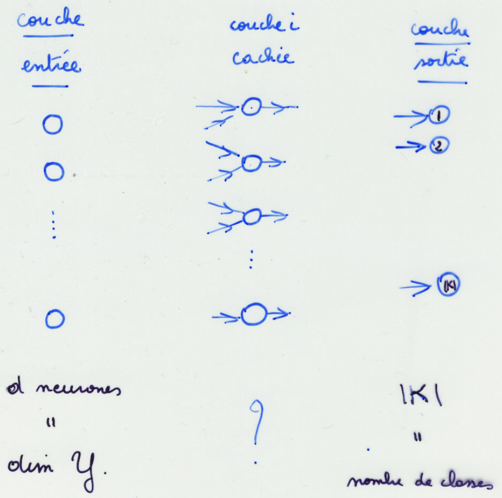
$$f(x) = 1 \quad \text{si } x > a$$
$$= 0 \quad \text{sinon}$$

$$w_{d+1} = -a$$

$$f(z) = 1 \iff w^t \bar{y} > 0$$

III 2 le reseau multicouche

= reseau de neurones formels organisés en "couches"

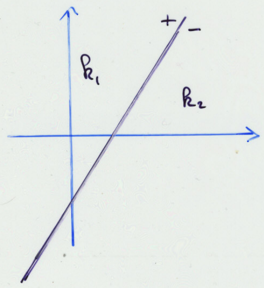


Topologie à définir

en sortie

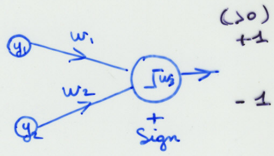
(k) → $\left|\begin{array}{ll} 1 & \text{si } y \in \text{Classe } k \\ 0 & \text{sinon} \end{array}\right.$

III 3 Exemple : vers le non linéaire

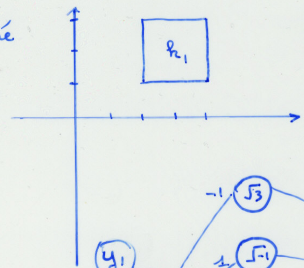


$w_1 y^1 + w_2 y^2 + w_3 \gtrless 0$

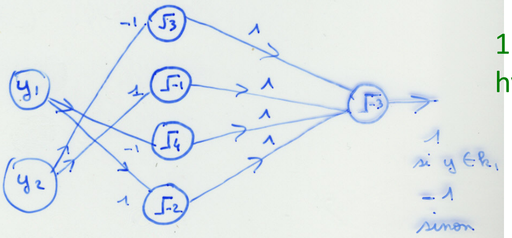
$\simeq$



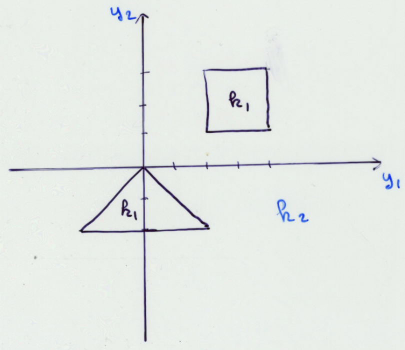
Carré



$$\begin{array}{lcl} y_2 < 3 & & -y_2 + 3 > 0 \\ y_2 > 1 & \Leftrightarrow & y_2 - 1 > 0 \\ y_1 < 4 & & -y_1 + 4 > 0 \\ y_1 > 2 & & y_1 - 2 > 0. \end{array}$$



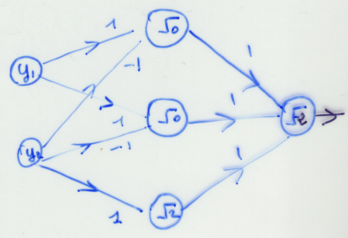
1 cellule élémentaire/
hyperplan



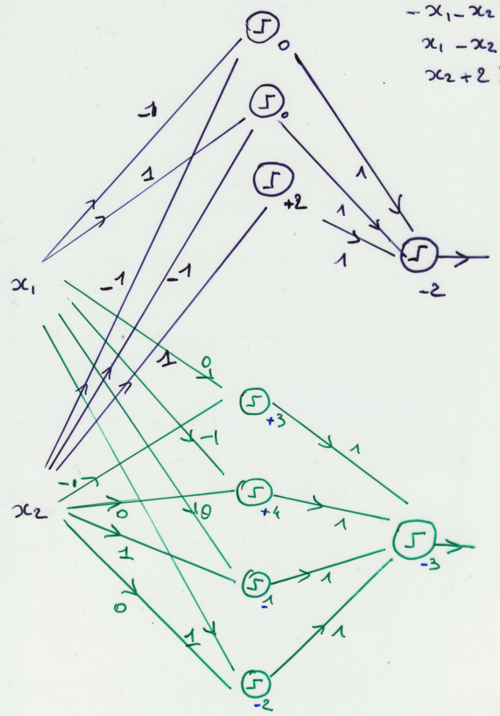
Triangle

$$\begin{cases} y_2 \leq y_1 \\ y_2 \leq -y_1 \\ y_2 \geq -2 \end{cases}$$

$$\begin{cases} -y_2 + y_1 > 0 \\ -y_1 - y_2 > 0 \\ y_2 + 2 > 0 \end{cases}$$



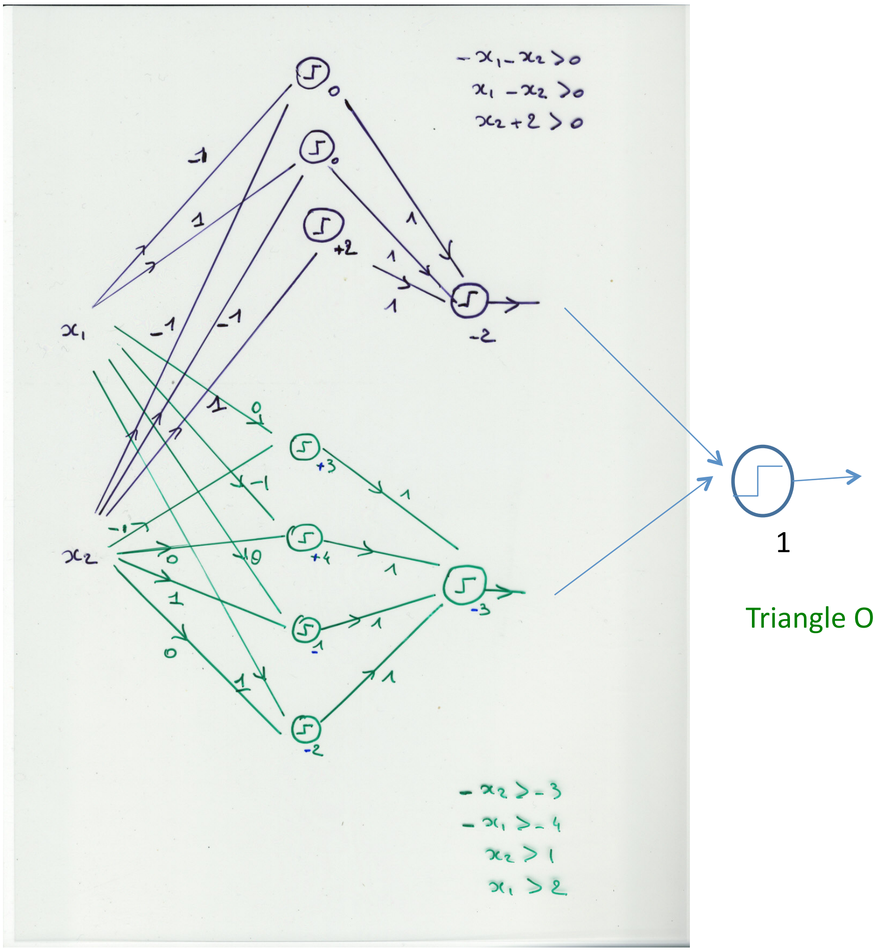
$$-x_1 - x_2 > 0$$
$$x_1 - x_2 > 0$$
$$x_2 + 2 > 0$$



triangle

carré

$$-x_2 > -3$$
$$-x_1 > -4$$
$$x_2 > 1$$
$$x_1 > 2$$



Triangle OU carré