Matière couverte → apprentissage supervisé

$$\underbrace{\{x^{(i)}, t^{(i)}\}}_{}{}_{i=1}^{N}$$

$$x^{(i)} = \mathbb{R}^D$$

$$h_\beta(x) = \beta_0 + \beta_1 x_1^{(i)} + \ldots + \beta_D x_D^{(i)}$$

→ regression linéaire
  → descente de gradient
  → resolution equations normales
→ overfitting
  ↳ décomposition biais variance
→ Regularisation
  → Best subset selection
  → Ridge
  → LASSO

Classification $\quad t^{(i)} \in \{1, \dots, K\}$

↳ classification par moindre carré

→ 1 contre 1

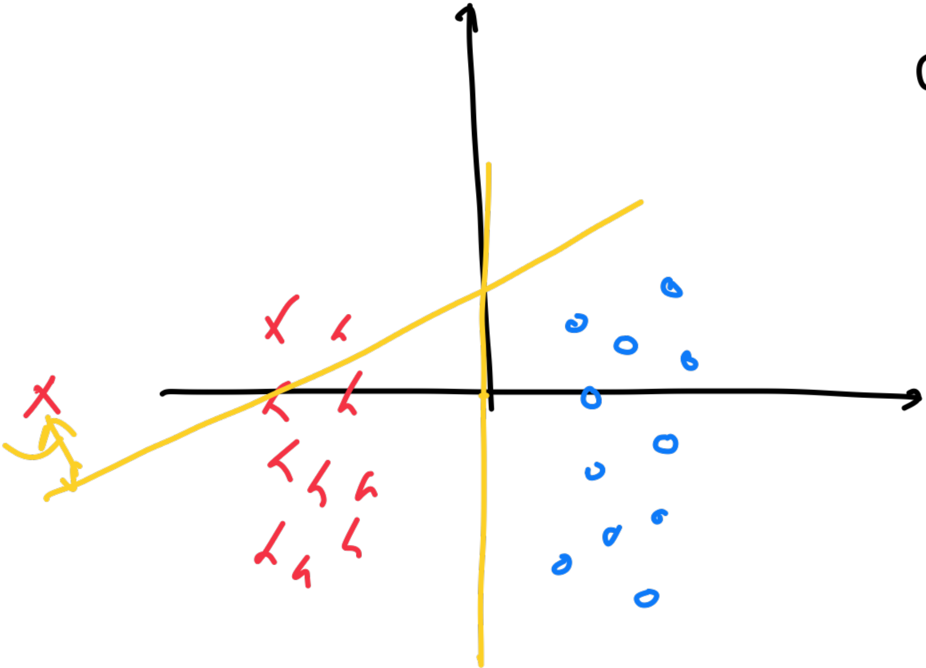
→ 1 contre tous

→ discriminant à K classes $\quad X \in \mathbb{R}^{N \times D+1}$

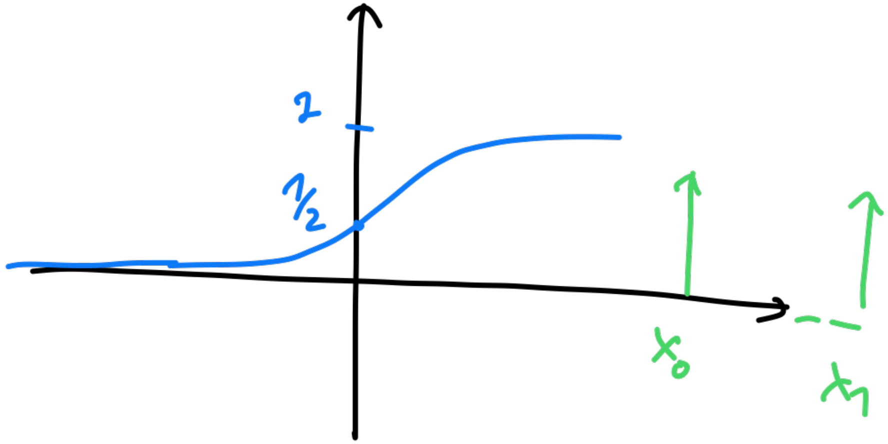
$$B = \begin{bmatrix} - & \beta_1^T & - \\ & \vdots & \\ - & \beta_K^T & - \end{bmatrix} \qquad T \approx X B^T$$

$$T = \begin{array}{c} \\ N \left\{ \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 1 & 0 & 0 & \cdots & 0 \\ & \vdots & & & \end{bmatrix} \right. \\ \underbrace{\hspace{3cm}}_{K} \end{array}$$

$$B = (X^T X)^{-1} X^T T$$



$$\sigma(x) = \frac{1}{1+e^{-x}}$$



→ amélioration : on va considérer l'ajout d'une fonction d'activation

$$\sigma(\beta_0 + \beta_1 x_1 + \ldots + \beta_D x_D) = \frac{1}{1+e^{-(\beta_0+\beta_1 x_1 + \ldots + \beta_D x_D)}}$$

Regression logistique

$$\rightarrow \; p(t=0 \mid x) = \sigma(\beta_0 + \beta_1 x_1 + \dots + \beta_D x_D) \; \leftarrow$$

$$p(t=1 \mid x) = 1 - \sigma(\beta_0 + \beta_1 x_1 + \dots + \beta_D x_D)$$

$\{x^{(i)}, t^{(i)}\}$

pour une paire $\{x^{(i)}, t^{(i)}\}$

$$p(t = t^{(i)} \mid x = x^{(i)}) = \underbrace{p(t=0 \mid x^{(i)})}^{1-t^{(i)}} \left(\underbrace{p(t=1 \mid x^{(i)})}\right)^{t^{(i)}}$$

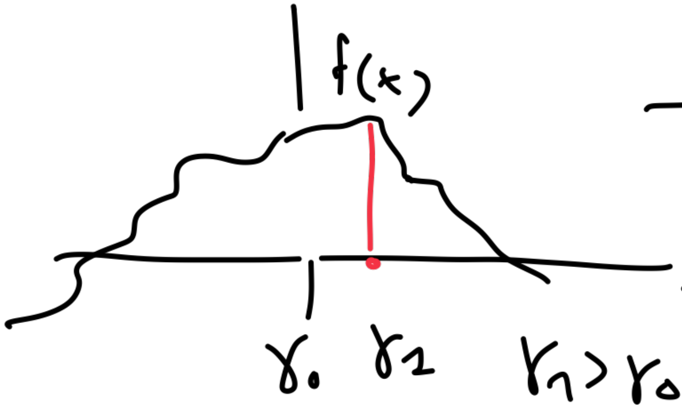
$$p(\{t^{(i)}\}_{i=1}^{N} \mid \{x^{(i)}\}) = \prod_{i=1}^{N} p(t = t^{(i)} \mid x = x^{(i)})$$

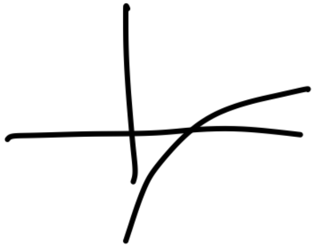
(si on suppose $\{x^{(i)}, t^{(i)}\}$ indépendance

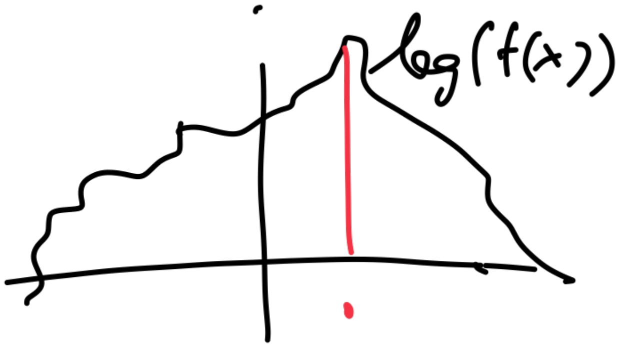
$$p(\{t^{(i)}\}_{i=1}^{N} \mid \{x^{(i)}\}) = \prod_{i=1}^{N} \sigma(\beta_0 + \beta_1 x_1^{(i)} + \dots + \beta_D x_D^{(i)})^{1-t^{(i)}}$$

$$\times \left(1 - \sigma(\beta_0 + \beta_1 x_1 + \dots + \beta_D x_D)\right)^{t^{(i)}}$$

$$\beta = \underset{\beta}{\arg\max}\; p(\{t^{(i)}\}_{i=1}^{N} \mid \{x^{(i)}\})$$







$\log(\gamma_1)$

$\gamma_1 > \gamma_0 \quad > \log(\gamma_0)$

LLE

$$\beta^* = \arg\max \log\left(\prod_{i=1}^{N} \sigma(\beta_0 + \beta_1 x_1 + \dots + \beta_D x_D)^{1-t^{(i)}} \left(1 - \sigma(\beta_0 + \beta_1 x_1 + \dots + \beta_D x_D)\right)^{t^{(i)}} \right)$$

$$= \arg\max_{\beta} \sum_{i=1}^{N} (1-t^{(i)}) \log\left(\sigma(\beta_0 + \beta_1 x_1^{(i)} + \dots + \beta_D x_D^{(i)})\right) + t^{(i)} \log\left(1 - \sigma(\beta_0 + \beta_1 x_1^{(i)} + \dots + \beta_D x_D^{(i)})\right)$$

$$\arg\min_{\beta} - \sum_{i=1}^{N} (1-t^{(i)}) \log(\sigma) + t^{(i)} \log(1-\sigma)$$

BINARY CROSS ENTROPY

Modèles de classification probabilistes : Modèles discriminants

$$p(t|x)$$

ex : regression

Modèles génératifs

$$p(x|t)$$

$$p(t|x) = \frac{p(x|t)\,p(t)}{p(x)}$$

$$p(x) \rightarrow \sum_{\ell} p(x|t_\ell)\,p(t_\ell)$$

$$t(x) = \arg\max_k p(t=k|x^{(i)}) = \arg\max_k p(x|t=k)\,p(t=k)$$

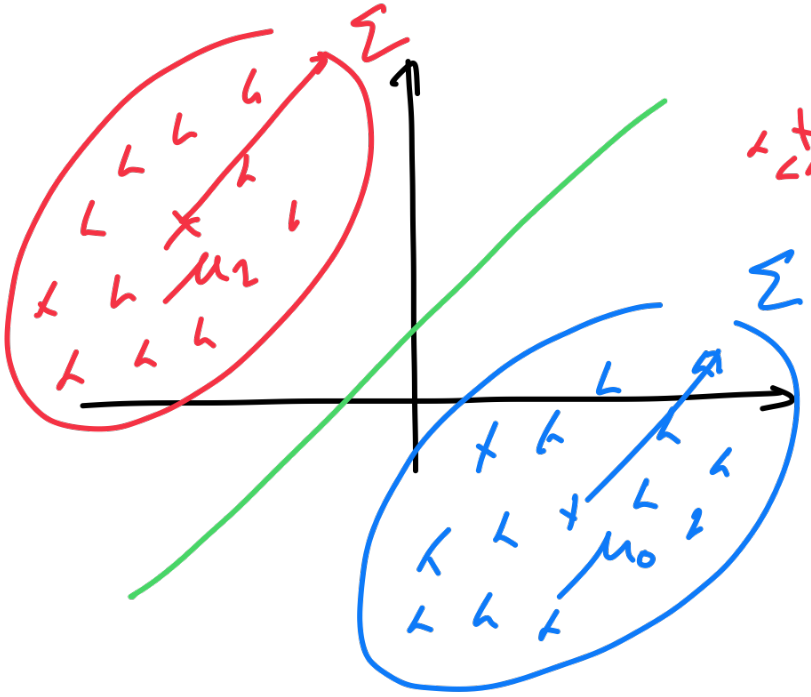
Un des modèles génératifs les plus populaires correspond à prendre $p(x|t)$ donné par une distribution normale multivariée.

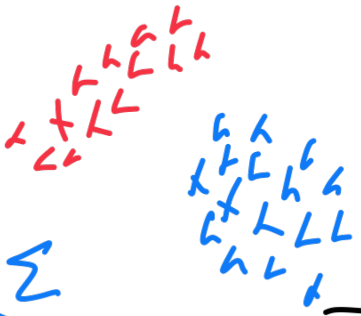
$$p(\bar{x} \mid t=0) = \frac{1}{(2\pi)^{D/2} |\Sigma_0|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu_0)^T \Sigma_0^{-1} (x-\mu_0)\right)$$

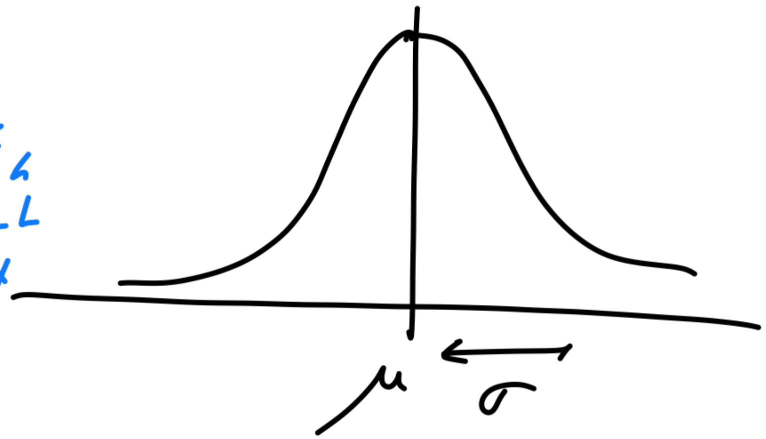
$$p(\bar{x} \mid t=1) = \frac{1}{(2\pi)^{D/2} |\Sigma_1|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu_1) \Sigma_1^{-1} (x-\mu_1)\right)$$

GDA → Analyse Gaussienne discriminante

Si $\Sigma_0 = \Sigma_1 = \Sigma$ → LDA Analyse linéaire discriminante







$$p(x; \mu\, \sigma)$$

$$= \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2\sigma^2}(x-\mu)^2\right)$$

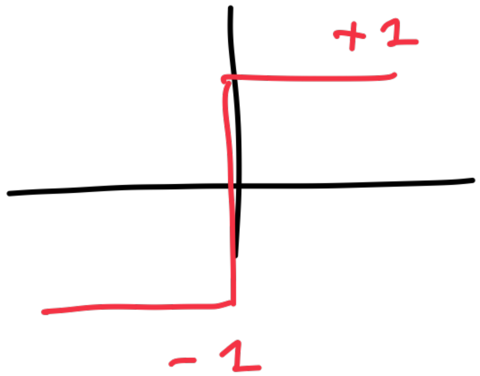
Les paramètres du modèle peuvent être calculés explicitement

$$p(t=0) = \frac{N_0}{N} \qquad p(t=1) = \frac{N_1}{N}$$

$$\mu_0 = \frac{1}{N_0} \sum_{i \in C_0} x^{(i)} \qquad \mu_1 = \frac{1}{N_1} \sum_{i \in C_1} x^{(i)}$$

$$\Sigma = \frac{1}{N}\left(\sum_{i\in C_0} (x^{(i)}-\mu_0)(x^{(i)}-\mu_0)^T + \sum_{i\in C_1} (x^{(i)}-\mu_1)(x^{(i)}-\mu_1)^T \right)$$

$$\sigma(x) = \begin{cases} +1 & \text{if } x \geq 0 \\ -1 & \text{if } x < 0 \end{cases}$$



→ fonction d'activ. de gradient nul presque partout

on va considérer une nouvelle fonction de coût

$$\ell = - \sum_{i \in \text{misclassified}} \underbrace{t^{(i)}\left(\beta_0 + \beta_1 x_1^{(i)} + \dots + \beta_D x_D^{(i)}\right)}$$

$$\beta = \arg\min \ell(\beta)$$

PERCEPTRON

→ Tant qu'il existe des points incorrectement classés

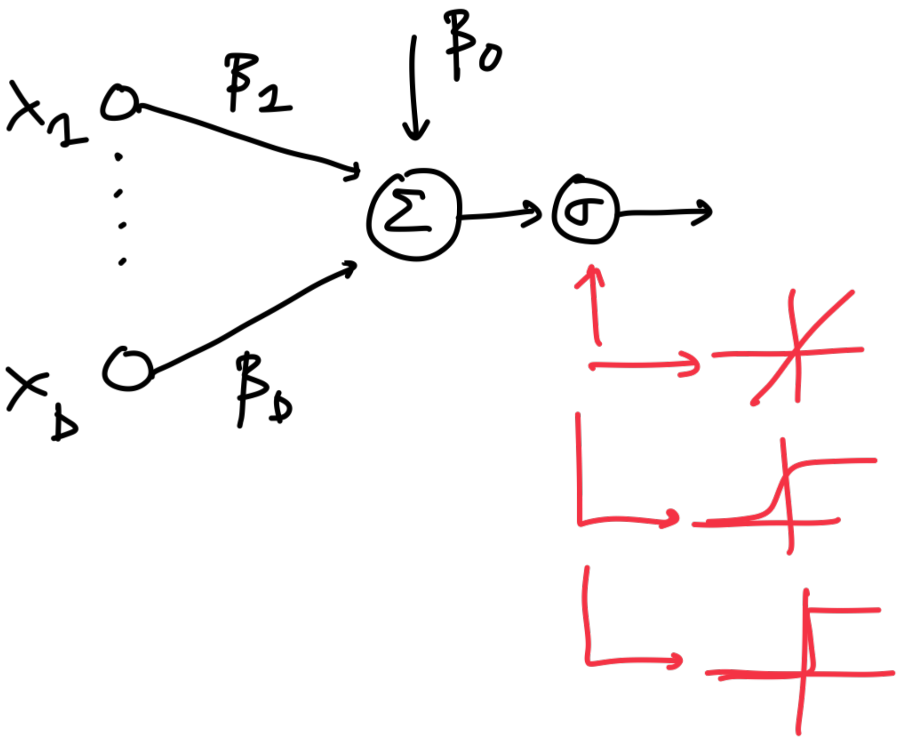
→ Sélectionner un de ces points

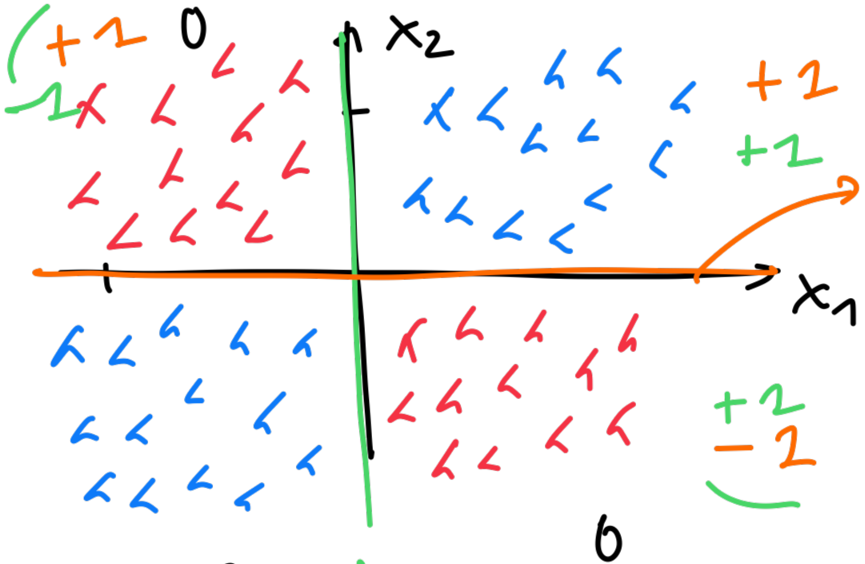
→ Mettre à jour β

$$\beta \leftarrow \beta + \eta \cdot t^{(i)} \tilde{x}^{(i)}$$

→ Si les données sont linéairement séparables, l'algorithme converge en 1 nombre fini d'itérations.

Tous les modèles couverts jusqu'ici peuvent être représentés par le diagramme suivant





-1 2

-1

$$y_2(x) = \sigma(x_2) = \begin{cases} 1 & x_2 \geq 0 \\ -1 & x_2 < 0 \end{cases}$$

$$y_1(x) = \sigma(x_1) = \begin{cases} 1 & x_1 \geq 0 \\ -1 & x_1 < 0 \end{cases}$$

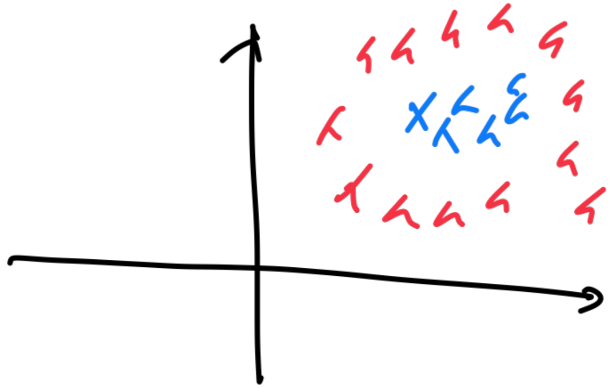
$$y(x) = |\underbrace{y_2}(x) + \underbrace{y_2}(x)| = \sigma^{(2)}\left(1.\,\sigma^{(1)}(x_1) + \sigma^{(2)}(x_2)\right)$$

$$= f(x)$$

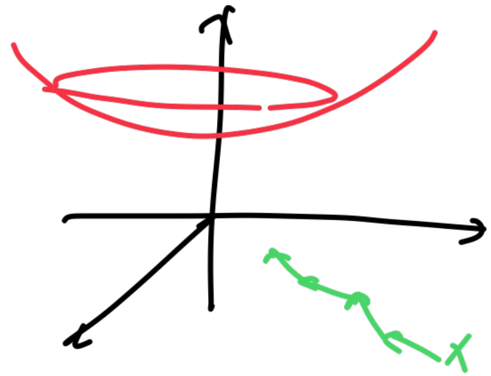
$w_1 = 2$ $\qquad$ $w_2 = 1$

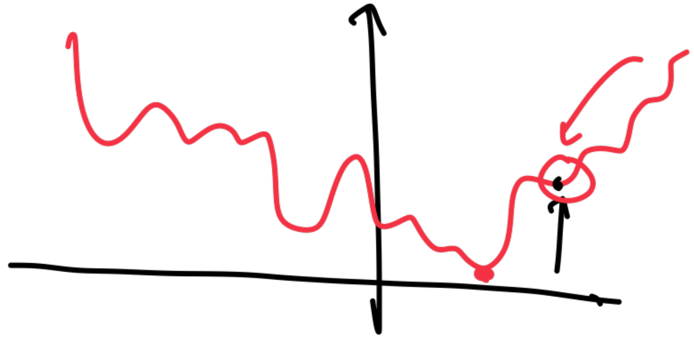
$$p(t=0 \mid x) = y_w(x)$$

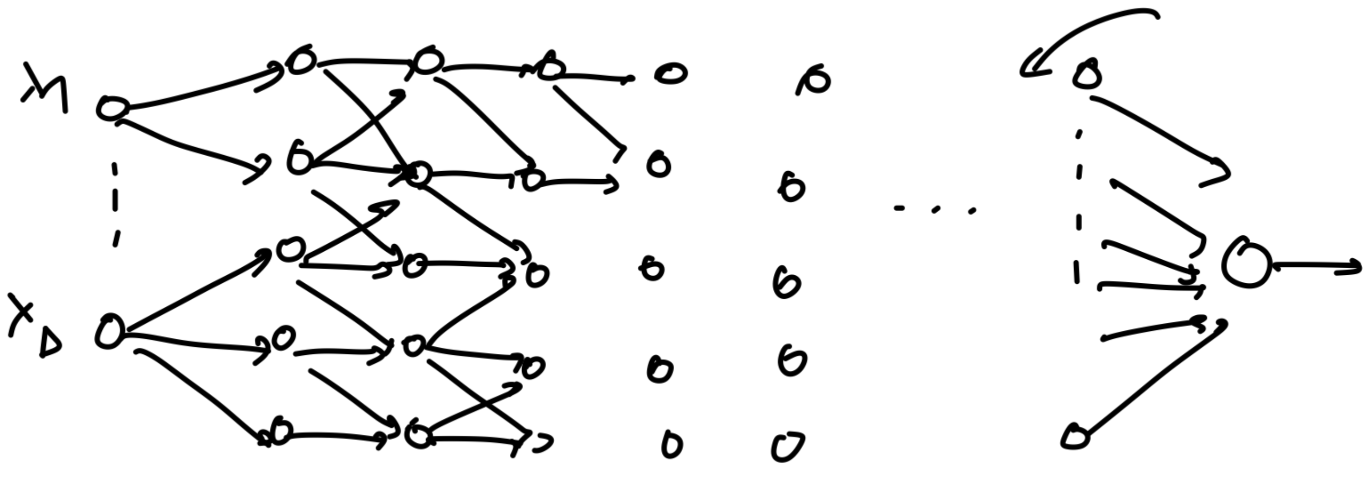
$$p(t=1 \mid x) = 1 - y_w(x)$$



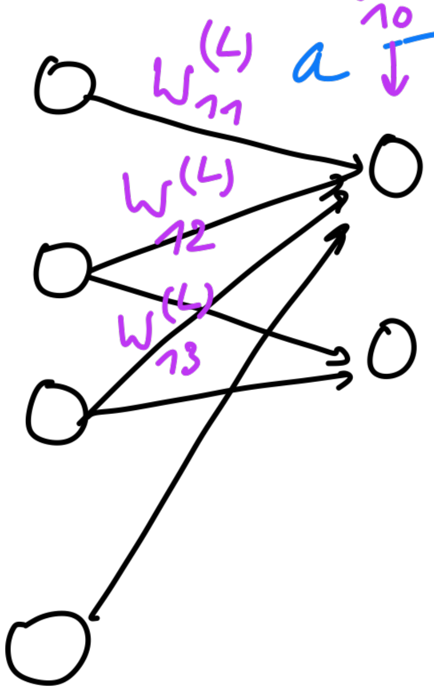
$$l(w) = - \sum_{i=1}^{N} (1 - t^{(i)}) \log(y_w) + t^{(i)} \log(1 - y_w)$$







Dans le cadre des réseaux de neurones le calcul du gradient se base sur l'idée de <u>backpropagation</u>



$w_{10}^{(L)}$

a → pre activation

z → post activation

$$a_1^{(L)} = \sum_{i=1}^{N_{L-1}} w_{1i}^{(L)} z_i^{(L-1)} + w_{10}^{(L)}$$

$$z_1^{(L)} = \sigma\left(a_1^{(L)}\right)$$

$$= \sigma\left(\sum_{i=1}^{N_{L-1}} w_{1i}^{(L)} z_i^{(L-1)} + w_{10}^{(L)}\right)$$