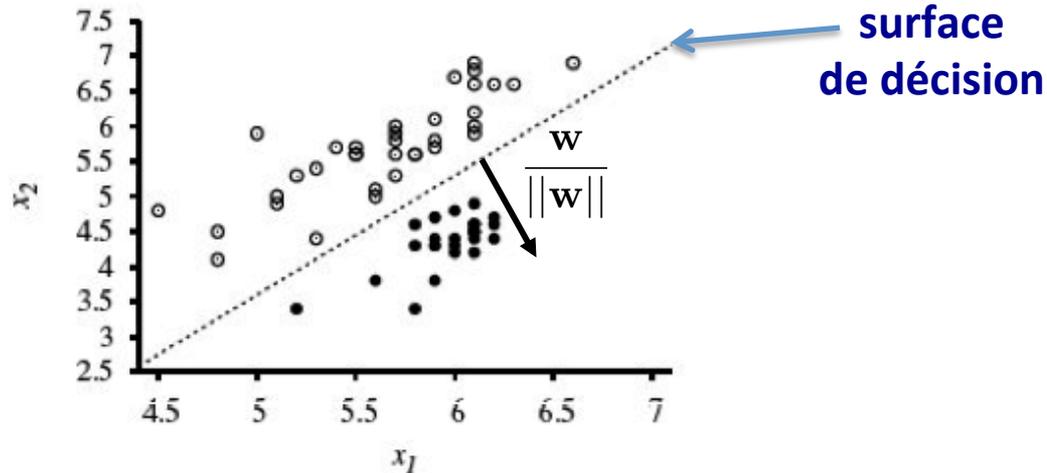


# Surface de séparation

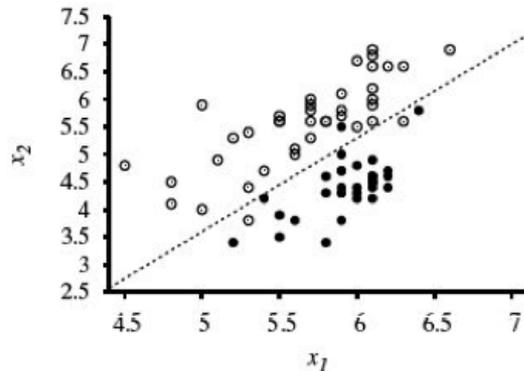
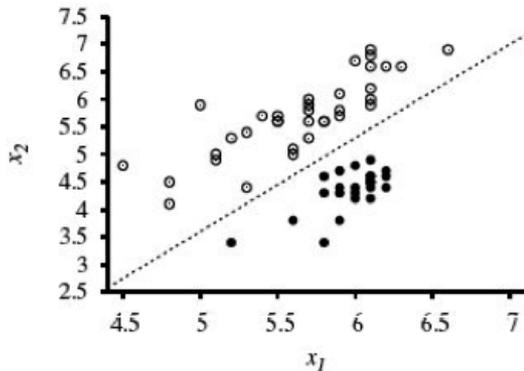
- Le perceptron cherche donc un **séparateur linéaire** entre les deux classes



- La **surface de décision** d'un classifieur est la surface qui sépare les deux régions classifiées dans les deux classes différentes

# Convergence et séparabilité

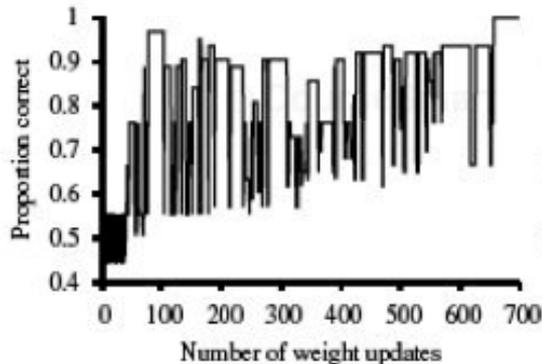
- Si les exemples sont **linéairement séparables** (gauche), le perceptron est garanti de converger à **une solution avec une erreur nulle** sur l'ensemble d'entraînement, pour tout  $\alpha$



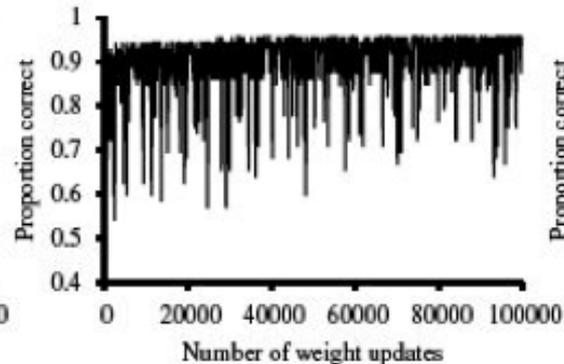
- Sinon, pour garantir la convergence à une **solution ayant la plus petite erreur possible en entraînement**, on doit décroître le taux d'apprentissage, par ex. selon  $\alpha_k = \frac{\alpha}{1 + \beta k}$

# Courbe d'apprentissage

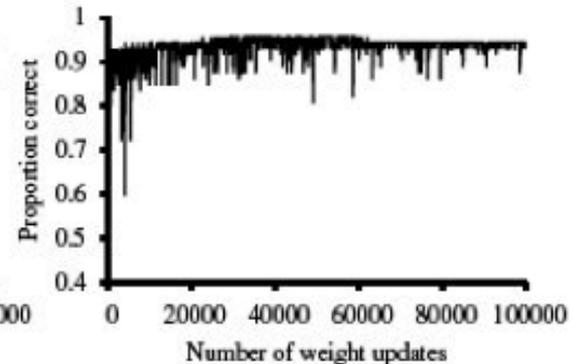
- La **courbe d'apprentissage** est la courbe du taux d'erreur (ou de succès) en fonction du nombre de mises à jour des paramètres
  - ◆ utile pour visualiser la progression de l'apprentissage



linéairement  
séparable



pas linéairement  
séparable



taux d'app. décroissant

# Apprentissage vue comme la minimisation d'une perte

- Le problème de l'apprentissage peut être formulé comme un problème d'optimisation
  - ◆ pour chaque exemple d'entraînement, on souhaite minimiser une certaine distance  $Loss(y_t, h_{\mathbf{w}}(\mathbf{x}_t))$  entre la cible  $y_t$  et la prédiction  $h_{\mathbf{w}}(\mathbf{x}_t)$
  - ◆ on appelle cette distance une **perte**
- On peut dériver l'algorithme du perceptron de cette façon ...