

IFT3390/6390

Fondements de l'apprentissage machine

<http://www.iro.umontreal.ca/~vincentp/ift3390>

## L'astuce du noyau (kernel trick)

Professeur: Pascal Vincent

LISA  Laboratoire d'Informatique des Systèmes d'Apprentissage

## Rappel

- Il existe de nombreuses méthodes pour construire un classifieur linéaire. (Ex: l'algorithme du Perceptron, la régression logistique, la régression avec erreur quadratique.)
- Nous avons vu qu'il est possible d'obtenir des classifieurs non linéaires à partir de classifieurs linéaires, avec un simple prétraitement des données.
- Il suffit d'appliquer une transformation non-linéaire (*mapping*)  $\varphi$  aux points de données  $x$  qui les transforme dans un espace de plus haute dimension en  $\tilde{x} = \varphi(x)$

## Pour obtenir un classifieur non-linéaire

Il est possible de:

- Utiliser une fonction  $\varphi$  **explicite choisie à priori**, et de calculer explicitement les  $\tilde{x} = \varphi(x)$

Ex:  $\varphi : (x_{[1]}, x_{[2]}) \mapsto (1, x_{[1]}, x_{[2]}, x_{[1]}x_{[2]}, x_{[1]}^2, x_{[2]}^2, \sin x_{[1]}, \cos x_{[2]})$

- **Apprendre une transformation** non-linéaire  $\varphi$  ayant une forme particulière. On peut voir sous cet angle ce que fait la première couche cachée d'un réseau de neurones.
- Utiliser **l'astuce du noyau** (*kernel trick*)

## Problème avec un *mapping* explicite

- Si  $x$  est en haute dimension, un *mapping* polynômial même rapidement à calculer un  $\tilde{x}$  dans un espace gigantesque.
- Ex:  $x \in \mathbb{R}^d$  et *mapping* polynômial de degré  $k$  (tous les produits entre  $k$  éléments de  $x$ ), on doit calculer un  $\tilde{x}$  dans un espace de dimension de l'ordre de  $d^k$ . Ex:  $d=100, k=5$  donne 10 000 000 000

## L'astuce du noyau

- S'applique à tout algorithme qui peut s'exprimer sous forme d'une expression basée sur des produits scalaires entre des vecteurs observations d'entrée.
- L'astuce est de supposer qu'on peut calculer le produit scalaire  $\langle \varphi(x_i), \varphi(x_j) \rangle$  directement sans jamais avoir à calculer explicitement un  $\varphi(x)$
- On choisit un noyau  $K$  tel que

$$K(x_i, x_j) = \langle \varphi(x_i), \varphi(x_j) \rangle$$

## Illustration

$$\varphi : (x_{[1]}, x_{[2]}) \mapsto (x_{[1]}^2, \sqrt{2}x_{[1]}x_{[2]}, x_{[2]}^2)$$

$$\begin{aligned} K(x, w) &= \langle \varphi(x), \varphi(w) \rangle \\ &= \left\langle (x_{[1]}^2, \sqrt{2}x_{[1]}x_{[2]}, x_{[2]}^2), (w_{[1]}^2, \sqrt{2}w_{[1]}w_{[2]}, w_{[2]}^2) \right\rangle \\ &= x_{[1]}^2w_{[1]}^2 + \sqrt{2}x_{[1]}x_{[2]}\sqrt{2}w_{[1]}w_{[2]} + x_{[2]}^2w_{[2]}^2 \\ &= x_{[1]}^2w_{[1]}^2 + 2x_{[1]}x_{[2]}w_{[1]}w_{[2]} + x_{[2]}^2w_{[2]}^2 \\ &= (x_{[1]}w_{[1]} + x_{[2]}w_{[2]})^2 \\ &= (\langle x, w \rangle)^2 \end{aligned}$$

Il est possible de calculer le produit scalaire des vecteurs mappés sans jamais calculer le mapping explicitement !

## Terminologie

- On appellera l'espace de départ dans lequel se trouvent les  $x$ , l'espace de départ ou l'espace des  $x$  ou l'espace des entrées brutes (*raw input space*)
- L'espace dans lequel  $\varphi$  mappe les données sera appelé *espace- $\varphi$*  ou espace des traits (*feature space*)
- $K$  correspond à un produit scalaire dans l'espace- $\varphi$ .

## Ex d'utilisation de l'astuce du noyau avec un algorithme simple

On considère l'algorithme simple suivant

- Étant donné un ensemble d'entraînement contenant des exemples de 2 classes (classe +1 et classe -1)
- L'algorithme choisit au hasard un point de chaque classe:  $x^+$  et  $x^-$
- Pour un nouveau point de test  $x$ , la décision est basée sur lequel des deux points est le plus proche en distance Euclidienne.
- Ceci correspond à la fonction de décision:

$$f(x) = \text{sign}(d(x, x^-) - d(x, x^+))$$

## Comment exprimer la décision à l'aide de produits scalaires?

La fonction discriminante est:

$$\begin{aligned}g(x) &= d(x, x^-) - d(x, x^+) \\ &= \sqrt{\|x - x^-\|^2} - \sqrt{\|x - x^+\|^2} \\ &= \sqrt{\langle x - x^-, x - x^- \rangle} - \sqrt{\langle x - x^+, x - x^+ \rangle} \\ &= \sqrt{\langle x, x \rangle - 2\langle x, x^- \rangle + \langle x^-, x^- \rangle} \\ &\quad - \sqrt{\langle x, x \rangle - 2\langle x, x^+ \rangle + \langle x^+, x^+ \rangle} \\ &= \sqrt{K(x, x) - 2K(x, x^-) + K(x^-, x^-)} \\ &\quad - \sqrt{K(x, x) - 2K(x, x^+) + K(x^+, x^+)}\end{aligned}$$

On remplace les produits scalaires par un noyau K

## Noyaux fréquemment utilisés

- Le produit scalaire usuel:  $K(a, b) = \langle a, b \rangle$
- Noyau polynômial de degré k:  $K_k(a, b) = (1 + \langle a, b \rangle)^k$
- Noyau RBF (ou Gaussien):  $K_\sigma(a, b) = e^{-\frac{1}{2} \frac{\|a-b\|^2}{\sigma^2}}$

### Remarques:

- Il existe de nombreux autres noyaux utiles.
- Les noyaux peuvent comporter des (hyper)-paramètres. ex:  $\sigma$  ou  $k$
- On ne peut pas toujours exprimer explicitement la fonction  $\varphi$  correspondant à un noyau K donné. Ainsi le noyau RBF correspond à un *espace- $\varphi$*  de dimension infinie.

## Propriétés d'un noyau

- Pour qu'il existe un *espace- $\varphi$*  et une fonction  $\varphi$  correspondant à un noyau K, il suffit que K soit un "*noyau de Mercer*"
- Un "*noyau de Mercer*" K vérifie les propriétés suivantes:  $K(a, b)$  est continu, symétrique et défini positif.