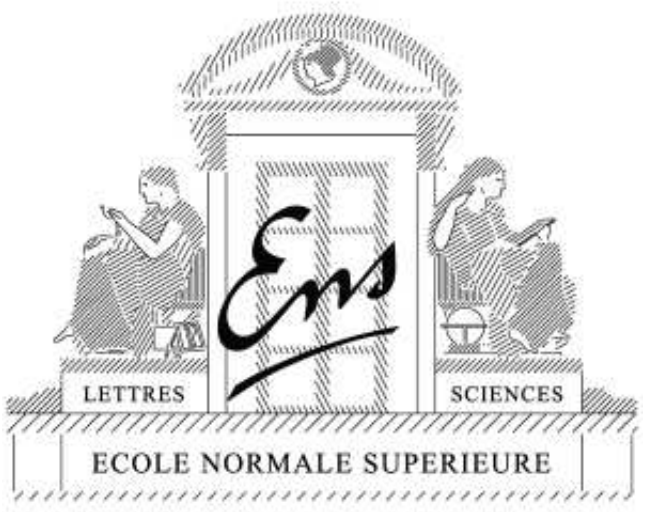
Méthodes à noyaux en apprentissage statistique

Francis Bach

INRIA - Ecole Normale Supérieure



Colloque RASMA - Janvier 2008

Méthodes à noyaux en apprentissage statistique

- Motivations:
  - Algorithmes généraux et modulaires pour l'analyse de données multivariées
  - Peu d'hypothèse sur le type de données
  - Garanties théoriques

- Principe de séparation entre
  1. Représentation des données à l'aide de noyaux (fonction de comparaison de deux données)
  2. Algorithmes utilisant uniquement des évaluations de noyaux

Plan du cours

1. Noyaux et espaces de Hilbert à noyaux reproduisants (RKHS)
   - Noyaux définis positifs, Noyaux de Mercer, RKHS
2. Méthodes à noyaux générales
   - Astuce du noyau et théorème du représentant
   - Kernel ridge regression, Kernel PCA / CCA
3. Méthodes à noyaux et optimisation convexe
   - Rappels d'optimisation convexe
   - Support vector machines
4. Design/apprentissage du noyau
   - Données structurées - applications
   - Normes ℓ_1 et parcimonie

Principe

- Représenter les données d'entrée $x_1, \ldots, x_n \in \mathcal{X}$ par une matrice carrée définie par $K_{ij} = k(x_i, x_j)$
- $\mathcal{X}$ "input space" arbitraire
- $K \in \mathbb{R}^{n \times n} =$ matrice de noyau
- Algorithmes utilisent toujours K!

Noyaux définis positifs

- Fonction $k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$
- Symmétrique: $\forall x, y \in \mathcal{X},\ k(x, y) = k(y, x)$
- Condition de positivité : $\forall x_1, \ldots, x_n \in \mathcal{X}$, la matrice de noyau K est définie positive, i.e.,

$$\forall \alpha \in \mathbb{R}^n,\ \alpha^\top K \alpha = \sum_{i,j=1}^{n} \alpha_i \alpha_j k(x_i, x_j) \geqslant 0$$

Examples de noyaux définis positifs

- Linéaire: $\mathcal{X} = \mathbb{R}^d$, $k(x, y) = x^\top y$
- Générique: supposons donné un "feature map" $\Phi : \mathcal{X} \mapsto \mathcal{F}$,

$$k(x, y) = \langle \Phi(x), \Phi(y) \rangle_{\mathcal{F}}$$

  - Example: $\mathcal{F} = \mathbb{R}^d$, mais peut être plus général

- Polynomial en 1 dimension ($\mathcal{X} = \mathbb{R}$):
  - $\Phi(x) = (1, 2^{1/2}x, x^2)^\top \Rightarrow k(x, y) = (1 + xy)^2$
  - $\Phi(x) \in \mathbb{R}^{d+1}$ avec $\Phi(x)_k = \binom{d}{k-1}^{1/2} x^{k-1} \Rightarrow k(x, y) = (1 + xy)^d$
  - Polynomial en dimension $p > 1$?

Noyaux définis positifs = produits scalaires

- Théorème (Aronszajn, 1950): k est un noyau d.p. ssi il existe un espace de Hilbert $\mathcal{F}$ et un "feature map" $\Phi : \mathcal{X} \mapsto \mathcal{F}$ tels que

$$k(x, y) = \langle \Phi(x), \Phi(y) \rangle_{\mathcal{F}}$$

- Remarques:
  - $\mathcal{F}$ peut avoir une dimension infinite
  - Déterminer Φ à partir de k pas évident a priori
  - Recettes plus ou moins explicites (RKHS, Mercer)

Opérations sur noyaux définis positifs

- structure de cone
  - k noyau d.p., $\alpha > 0 \Rightarrow \alpha k$ noyau d.p.
  - k_1, k_2 noyaux d.p. $\Rightarrow k_1 + k_2$ noyau d.p.

- k_1, k_2 noyaux d.p. $\Rightarrow k_1 k_2$ noyau d.p.

Noyau polynomial en dimension $p > 1$

- Définition: $k(x, y) = (1 + x^\top y)^d$
- Première expansion: $k(x, y) = \sum_{k=1}^{d+1} \binom{d}{k-1} (x^\top y)^{k+1}$
- Deuxième expansion:

$$(x^\top y)^k = \left(\sum_{i=1}^{d} x_i y_i \right)^k = \sum_{i_1 + \cdots + i_d = k} \frac{k!}{i_1! \cdots i_d!} (x_1 y_1)^{i_1} \cdots (x_d y_d)^{i_d}$$

- $\Phi(x)$ contient tous les monomes (avec des poids) $x_1^{i_1} \cdots x_d^{i_d}$ avec $i_1 + \cdots + i_d \leqslant k$
- Dimension de $\mathcal{F}$: $\binom{p+d}{d}$ (grand!)

Noyaux invariants par translation sur $\mathcal{X} = \mathbb{R}^p$

- Noyau de la forme $k(x, y) = q(x - y)$, $q \in L^2(\mathbb{R}^p)$ continue
- Proposition: k est d.p. ssi la transformée de Fourier $Q(\omega)$ de q est positive ou nulle pour tout $\omega \in \mathbb{R}^p$
  - Preuve ...
- Example classique: noyau Gaussien $k(x, y) = e^{-\alpha\|x-y\|^2}$
- Quel est (si il existe) le "feature space" et le "feature map" ?

Résumé provisoire

- Noyau d.p. $\Leftrightarrow$ K matrice symmétrique semi-définie positive
- Noyaux explicitement de la forme $k(x, y) = \langle \Phi(x), \Phi(y) \rangle_{\mathcal{F}}$
- Noyaux implicitement de cette forme (e.g., noyau Gaussien)
- Deux théories permettent de "construire" Φ:
  - Espaces de Hilbert à noyaux reproduisants (RKHS)
  - Noyaux de Mercer

Définition d'un RKHS

- Soit $\mathcal{X}$ un ensemble quelconque et $\mathcal{F}$ un sous-espace de des fonctions de $\mathcal{X}$ dans $\mathbb{R}$, qui est muni d'un produit scalaire Hilbertien.

- $\mathcal{F}$ est un RKHS avec noyau reproduisant $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ ssi:
  - $\mathcal{F}$ contient toutes les fonctions de la forme

$$k(x, \cdot) : y \mapsto k(x, y)$$

  - $\forall x \in \mathcal{X}$ and $f \in \mathcal{F}$,

$$f(x) = \langle f, k(\cdot, x) \rangle_{\mathcal{F}}$$

(i.e., $k(\cdot, x)$ correspond au "Dirac" en x)

Propriétés des RKHS (Aronszajn, 1950)

- Unicité: si il existe un noyau reproduisant, il est unique
- Existence: un noyau reproduisant existe ssi $\forall x \in X$, la forme linéaire $f \mapsto f(x)$ est continue
- Si k est un noyau reproduisant, alors k est un noyau défini positif
- Si k est un noyau défini positif, alors k est un noyau reproduisant (pour un certain RKHS $\mathcal{F}$)
- Preuves...

Construction du RKHS pour un noyau d.p.

- Construction de $\mathcal{F}_0$ l'ensemble de combinaisons linéaires finies de fonctions $k(\cdot, y)$, $y \in \mathcal{X}$

- Produit scalaire sur $\mathcal{F}_0$ défini par

$$\langle \sum_i \alpha_i k(\cdot, x_i), \sum_j \beta_j k(\cdot, y_j) \rangle_{\mathcal{F}_0} = \sum_i \sum_j \alpha_i \beta_j k(x_i, y_j)$$

  - Indépendant de la décomposition (preuve...)
  - Produit scalaire (preuve...)
  - Complétion de l'espace pré-Hilbertien $\mathcal{F}_0$ par les limites de suites de Cauchy pour obtenir l'espace de Hilbert $\mathcal{F}$

- Interprétation de la norme $\| f \|^2_{\mathcal{F}}$?

Interprétation de la norme $\|f\|_{\mathcal{F}}^2$

- La norme contrôle les variations de f:
- $|f(x) - f(y)| = |\langle f, k(\cdot, x) - k(\cdot, y)\rangle_{\mathcal{F}}| \leqslant \|f\|_{\mathcal{F}} \|k(\cdot, x) - k(\cdot, y)\|_{\mathcal{F}}$
- La norme du RKHS controle la constante de Lipschitz de f pour la métrique $d_k(x, y) = \|k(\cdot, x) - k(\cdot, y)\|_{\mathcal{F}}$
- Cas des noyaux invariants par translation sur $\mathbb{R}^p$:
  - $k(x, y) = q(x - y)$
  - On obtient $\|f\|_{\mathcal{F}}^2 = \int_{\mathbb{R}^p} \frac{|F(\omega)|^2}{Q(\omega)} d\omega$
  - NB: transformée de Fourier $F(\omega) = \int f(x) e^{-i\omega x} dx$
  - $d = 1$, Gaussian kernel, Exponential kernel (Sobolev space)

Noyaux de Mercer

- Construction semi-explicite d'un "feature space" égale a l'ensemble des suites de réels
- Hypothèses: $\mathcal{X}$ espace métrique compact muni d'une mesure ν, noyau k noyau d.p. continu.
- Théorème de Mercer: il existe une base Hilbertienne de $L^2(\nu)$ de fonctions continues $(\psi_i$ et une suite décroissante (λ_i) tendant vers zero, telles que

$$\forall x, y \in \mathcal{X},\ k(x,y) = \sum_{i=1}^{\infty} \lambda_i \psi_i(x) \psi_i(y)$$

- Corollaire: Feature map $\Phi : \mathcal{X} \to \ell^2$, $x \mapsto (\sqrt{\lambda_i}\psi_i(x))_{i \in \mathbb{N}}$

Noyaux de Mercer - Schéma de preuve

- Opérateur linéaire L_k défini par

$$L_k f(x) = \int_{\mathcal{X}} k(x,t) f(t) d\nu(t)$$

- Cet opérateur est continu, compact, auto-adjoint et positif

- Théorème spectral (résultat classique d'analyse fonctionnelle implique l'existence d'une base Hilbertienne ψ_i et de la suite λ_i de vecteurs propres et valeurs propres: $L_k \psi_i = \lambda_i \psi_i$.

- Construction "semi-explicite"

"Example où tout peut être calculé"

- "input space": $\mathcal{X} = [0,1]$ avec contraintes de périodicité, muni de la mesure de Lebesgue
- Base de L^2: $c_0(x) = 1$, $c_\nu(x) = \sqrt{2}\cos 2\pi\nu x$, $s_\nu(x) = \sqrt{2}\sin 2\pi\nu x$, $\nu > 0$.
- Norme:

$$\begin{aligned} \|f\|_{\mathcal{F}}^2 &= \left(\int_0^1 f(x)dx\right)^2 + \int f'(x)^2 dx \\ &= \langle c_0, f\rangle_{L^2}^2 + \sum_{\nu>0}(\langle c_\nu, f\rangle_{L^2}^2 + \langle s_\nu, f\rangle_{L^2}^2)(2\pi\nu)^2 \end{aligned}$$

- Noyau

$$\begin{aligned} k(x,y) &= 1 + \sum_{\nu>0}(2\pi\nu)^{-2}(c_\nu(x)c_\nu(y) + s_\nu(x)s_\nu(y)) \\ &= 1 + \frac{1}{2}\left[(x-y-\lfloor x-y\rfloor)^2 - (x-y-\lfloor x-y\rfloor) + 1/6\right] \end{aligned}$$

Résumé - Noyaux

- Noyaux définis positifs = produits scalaire de "features"

$$k(x, y) = \langle \Phi(x), \Phi(y) \rangle$$

- Noyaux de Mercer: "feature map" obtenu à partir de l'opérateur de convolution

- RKHS: construction explicite sans hypothèses

- interprétation de la norme du RKHS en terme de régularité des fonctions

- Noyaux classiques: linéaires, polynomiaux, Gaussiens

Plan du cours

1. Noyaux et espaces de Hilbert à noyaux reproduisants (RKHS)
   - Noyaux définis positifs, Noyaux de Mercer, RKHS
2. Méthodes à noyaux générales
   - Astuce du noyau et théorème du représentant
   - Kernel ridge regression, Kernel PCA / CCA
3. Méthodes à noyaux et optimisation convexe
   - Rappels d'optimization convexe
   - Support vector machines
4. Design/apprentissage du noyau
   - Données structurées - applications
   - Normes ℓ_1 et parcimonie

Méthodes à noyaux
Principes et premiers algorithmes

- Astuce du noyau - exemples simples
- Théorème du représentant
- Apprentissage non supervisé: Kernel ridge regression
- Apprentissage supervisé: Kernel PCA / CCA

Astuce du noyau

- noyau d.p. correspond à des "features" potentiellement nombreux et souvent implicites

- Principe: tout algorithme sur des vecteurs de dimension finie n'utilisant que des produits scalaires peut être utilisé en remplacant le produit scalaire par n'importe quel noyau défini positif

- Nombreuses applications

Exemple de méthodes à noyaux - I

- Distances dans le "feature space"

$$d_k(x,y)^2 = \|\Phi(x) - \Phi(y)\|_{\mathcal{F}}^2 = k(x,x) + k(y,y) - 2k(x,y)$$

Exemple de méthodes à noyaux - II
Algorithme simple de discrimination

- Données $x_1, \ldots, x_n \in \mathcal{X}$, classes $y_1, \ldots, y_n \in \{-1, 1\}$
- Compare les distances aux moyennes de chaque classe
- Equivalent à classifier x en utilisant le signe de

$$\frac{1}{\#\{i, y_i = 1\}} \sum_{i, y_i = 1} k(x, x_i) - \frac{1}{\#\{i, y_i = -1\}} \sum_{i, y_i = -1} k(x, x_i)$$

- Preuve...
- Interprétation géométrique des fenêtres de Parzen

Exemple de méthodes à noyaux - III
Centrage des données

- n points $x_1, \ldots, x_n \in \mathcal{X}$
- Matrice de noyau $K \in \mathbb{R}^n$, $K_{ij} = k(x_i, x_j) = \langle \Phi(x_i), \Phi(x_j) \rangle$
- Matrice de noyau des données centrées $\tilde{K}_{ij} = \langle \Phi(x_i) - \mu, \Phi(x_j) - \mu \rangle$ où $\mu = \frac{1}{n} \sum_{i=1}^{n} \Phi(x_i)$
- Formule $\tilde{K} = \Pi_n K \Pi_n$ avec $\Pi_n = I_n - \frac{E}{n}$, et E matrice constante égale à 1.
- preuve...
- NB: μ n'est pas de la forme $\Phi(z)$, $z \in \mathcal{X}$ (cf. problème de la pré-image)

Théorème du représentant

- Soit $\mathcal{X}$ un ensemble, un noyau d.p. k et son RKHS associé $\mathcal{F}$, et $x_1, \ldots, x_n$ n points dans $\mathcal{X}$.
- Soit $J : \mathbb{R}^{n+1} \to \mathbb{R}$ strictement croissante par rapport à la dernière variable
- Toute solution du problème d'optimisation suivant

$$\min_{f \in \mathcal{F}} J(f(x_1), \ldots, f(x_n), \|f\|_{\mathcal{F}})$$

  s'écrit de la forme $f = \sum_{i=1}^{n} \alpha_i k(\cdot, x_i)$.
- Cadre classique: $\min_{f \in \mathcal{F}} \sum_{i=1}^{n} \ell_i(f(x_i)) + \lambda \|f\|_{\mathcal{F}}^2$
- Preuve...

Kernel ridge regression (spline smoothing)

- Données $x_1, \ldots, x_n \in \mathcal{X}$, noyau d.p. k, $y_1, \ldots, y_n \in \mathbb{R}$
- Moindres carrés

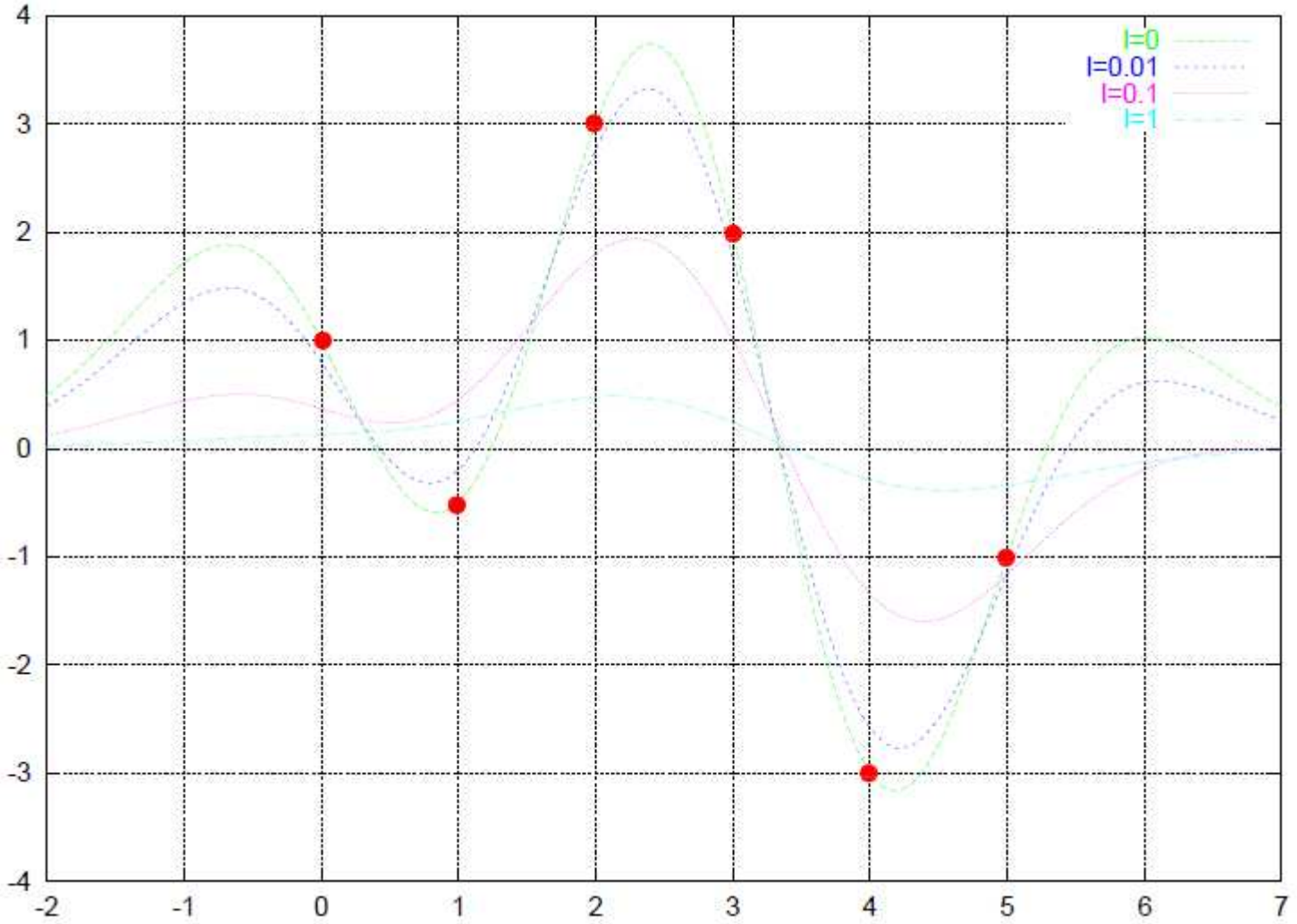
$$\min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} (y_i - f(x_i))^2 + \lambda \|f\|_{\mathcal{F}}^2$$

- Vue 1: théorème du représentant $\Rightarrow f = \sum_{i=1}^{n} \alpha_i k(\cdot, x_i)$
  - équivalent à

$$\min_{\alpha \in \mathbb{R}^n} \frac{1}{n} \sum_{i=1}^{n} (y_i - (K\alpha)_i)^2 + \lambda \alpha^\top K \alpha$$

  - Solution égale à $\alpha = (K + n\lambda I)^{-1} y + \varepsilon$ avec $K\varepsilon = 0$
  - Solution f unique!

Kernel ridge regression
Exemple (from Vert, 2007)



Kernel ridge regression
Remarques

- Liens avec le lissage par splines
- Autre vue: $\mathcal{F} \in \mathbb{R}^d$, $\Phi \in \mathbb{R}^{n \times d}$

$$\min_{w \in \mathbb{R}^d} \frac{1}{n} \|y - \Phi w\|^2 + \lambda \|w\|^2$$

- Solution égale à $w = (\Phi^\top \Phi + n\lambda I)^{-1} \Phi^\top y$
- Noter que $w = \Phi^\top (\Phi \Phi^\top + n\lambda I)^{-1} y$
- Φw égal à $K\alpha$

Kernel PCA

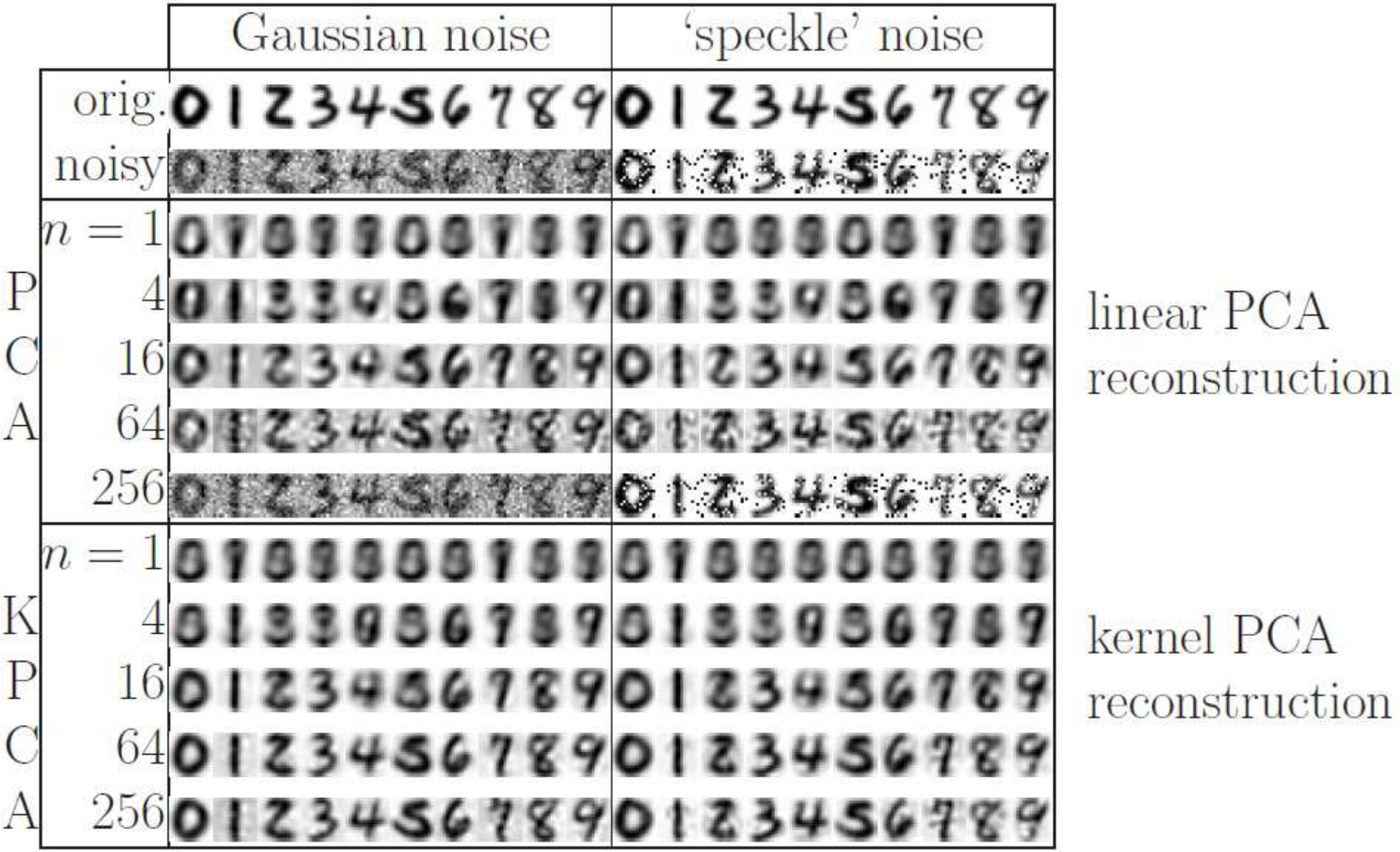
- Analyse en composante principale linéaire
  - données $x_1, \ldots, x_n \in \mathbb{R}^p$,

$$\max_{w \in \mathbb{R}^p} \frac{w^\top \hat{\Sigma} w}{w^\top w} = \max_{w \in \mathbb{R}^p} \frac{\mathrm{var}(w^\top X)}{w^\top w}$$

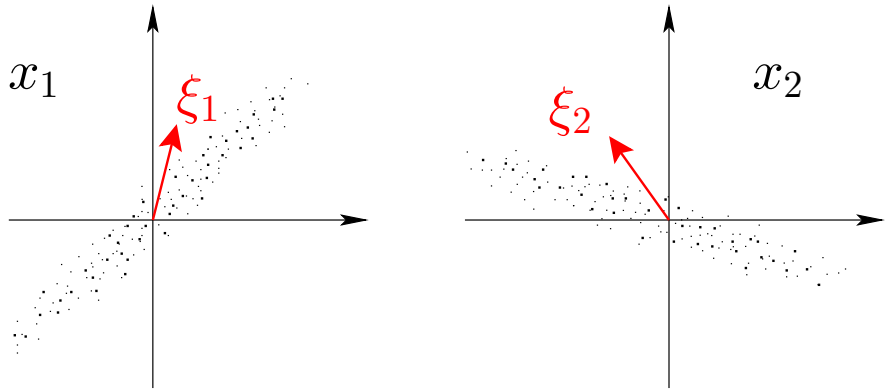
  - w est le plus grand vecteur propre de $\hat{\Sigma}$
  - Débruitage, représentation des données

- Kernel PCA: données $x_1, \ldots, x_n \in \mathcal{X}$, noyau d.p. k
  - Vue 1: $\max_{w \in \mathcal{F}} \frac{\mathrm{var}(\langle \Phi(X), w \rangle)}{w^\top w}$ Vue 2: $\max_{f \in \mathcal{F}} \frac{\mathrm{var}(f(X))}{\|f\|_{\mathcal{F}}^2}$
  - Solution $f, w = \sum_{i=1}^n \alpha_i k(\cdot, x_i)$ et α plus grand vector propre de $\tilde{K} = \Pi_n K \Pi_n$
  - Interprétation en termes d'opérateurs de covariance

Denoising with kernel PCA (From Schölkopf, 2005)



Canonical correlation analysis



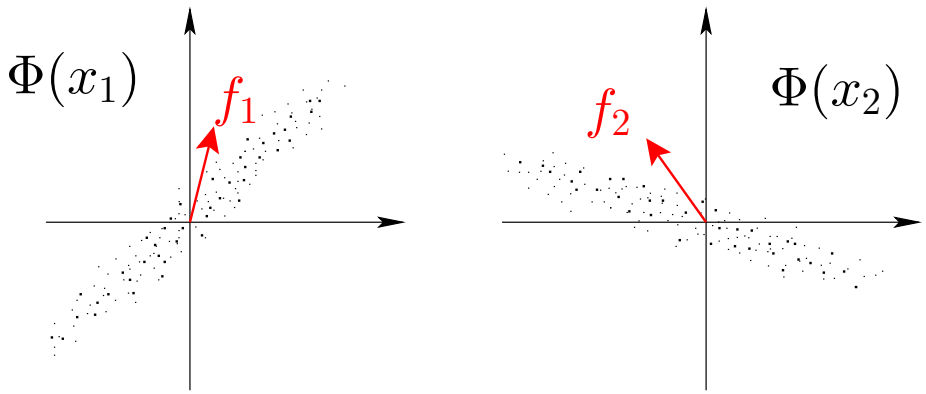
- Given two multivariate random variables x_1 and x_2, finds the pair of directions ξ_1, ξ_2 with maximum correlation:

$$\rho(x_1, x_2) = \max_{\xi_1, \xi_2} \operatorname{corr}(\xi_1^T x_1, \xi_2^T x_2) = \max_{\xi_1, \xi_2} \frac{\xi_1^T C_{12} \xi_2}{\left(\xi_1^T C_{11} \xi_1\right)^{1/2} \left(\xi_2^T C_{22} \xi_2\right)^{1/2}}$$

- Generalized eigenvalue problem:

$$\begin{pmatrix} 0 & C_{12} \\ C_{21} & 0 \end{pmatrix} \begin{pmatrix} \xi_1 \\ \xi_2 \end{pmatrix} = \rho \begin{pmatrix} C_{11} & 0 \\ 0 & C_{22} \end{pmatrix} \begin{pmatrix} \xi_1 \\ \xi_2 \end{pmatrix}.$$

Canonical correlation analysis in feature space



- Given two random variables x_1 and x_2 and two RKHS $\mathcal{F}_1$ and $\mathcal{F}_2$, finds the pair of functions f_1, f_2 with maximum regularized correlation:

$$\max_{f_1, f_2 \in \mathcal{F}} \frac{\mathrm{cov}(f_1(X_1), f_2(X_2))}{(\mathrm{var}(f_1(X_1)) + \lambda_n \|f_1\|^2_{\mathcal{F}_1})^{1/2} (\mathrm{var}(f_2(X_2)) + \lambda_n \|f_2\|^2_{\mathcal{F}_2})^{1/2}}$$

- Criteria for independence (NB: independence $\neq$ uncorrelation)

Kernel Canonical Correlation Analysis

- Analogous derivation as Kernel PCA
- K_1, K_2 Gram matrices of $\{x_1^i\}$ and $\{x_2^i\}$

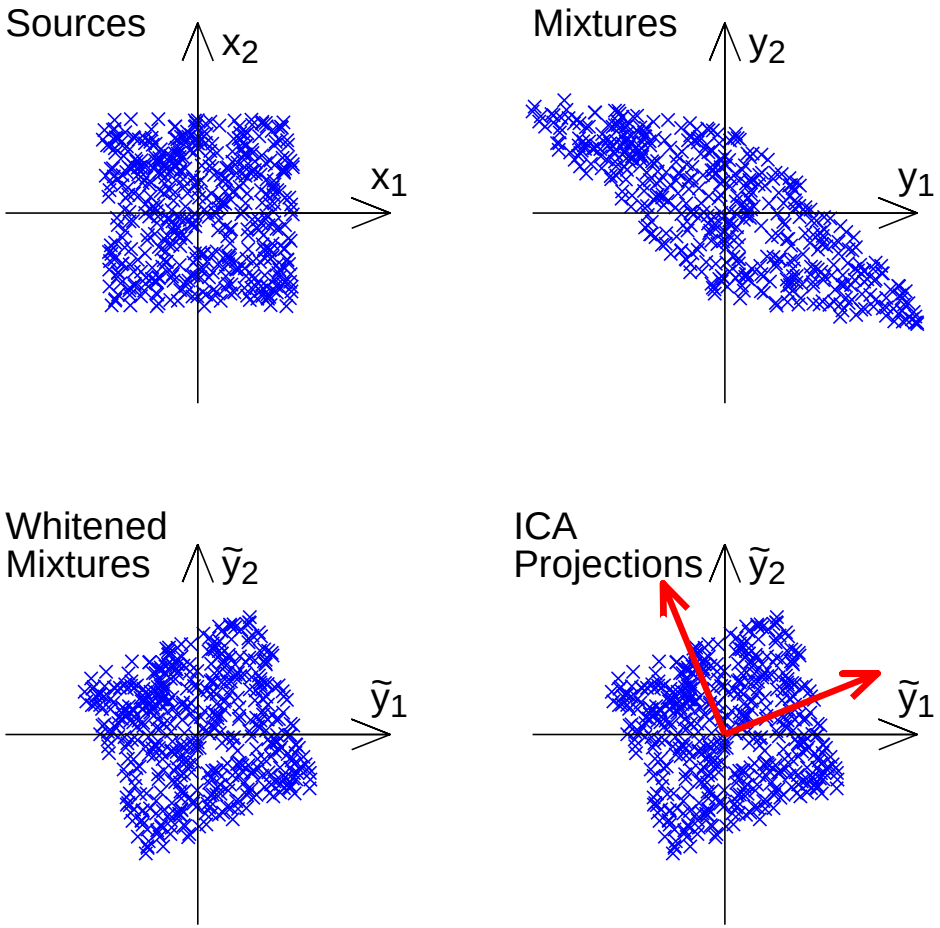
$$\max_{\alpha_1,\ \alpha_2 \in \Re^N} \frac{\alpha_1^T K_1 K_2 \alpha_2}{(\alpha_1^T (K_1^2 + \lambda K_1)\alpha_1)^{1/2} (\alpha_2^T (K_2^2 + \lambda K_2)\alpha_2)^{1/2}}$$

- Maximal generalized eigenvalue of

$$\begin{pmatrix} 0 & K_1 K_2 \\ K_2 K_1 & 0 \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \end{pmatrix} = \rho \begin{pmatrix} K_1^2 + \lambda K_1 & 0 \\ 0 & K_2^2 + \lambda K_2 \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \end{pmatrix}$$

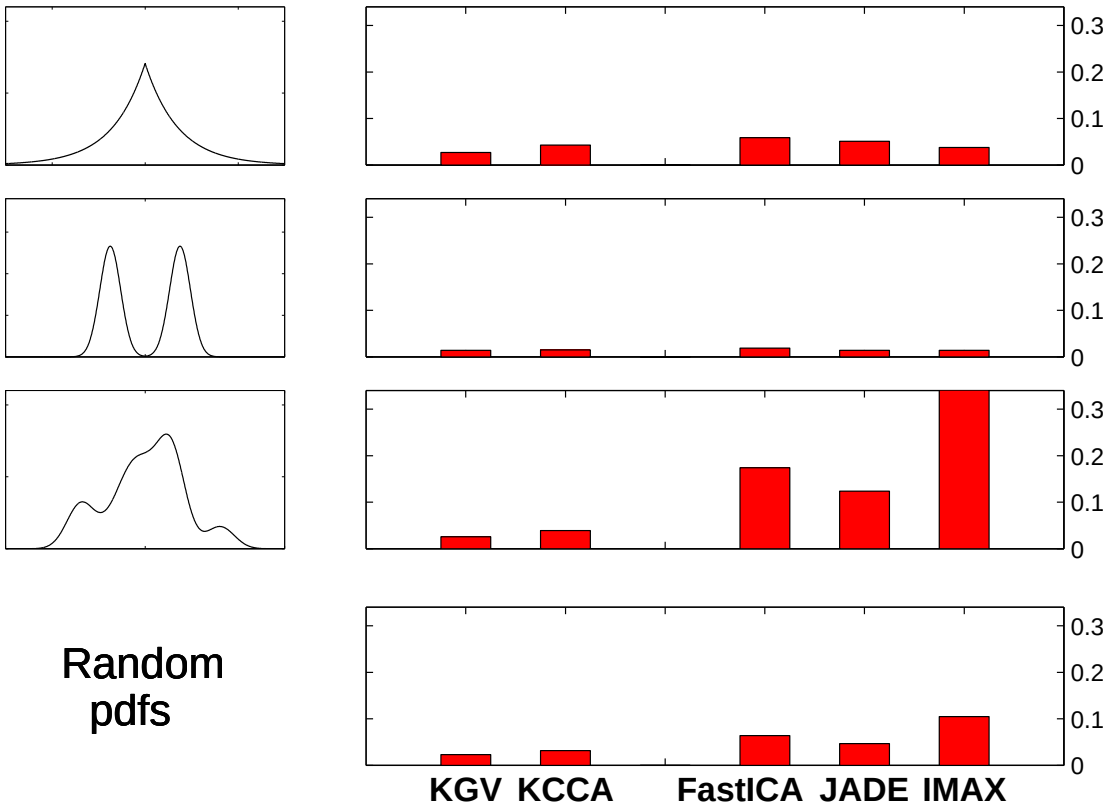
Kernel CCA
Application to ICA (Bach & Jordan, 2002)

- Independent component analysis: linearly transform data such to get independent variables



Empirical results - Kernel ICA

- Comparison with other algorithms: FastICA (Hyvarinen,1999), Jade (Cardoso, 1998), Extended Infomax (Lee, 1999)
- Amari error : standard ICA distance from true sources



Plan du cours

1. Noyaux et espaces de Hilbert à noyaux reproduisants (RKHS)
   - Noyaux définis positifs, Noyaux de Mercer, RKHS
2. Méthodes à noyaux générales
   - Astuce du noyau et théorème du représentant
   - Kernel ridge regression, Kernel PCA / CCA
3. Méthodes à noyaux et optimisation convexe
   - Rappels d'optimisation convexe
   - Support vector machines
4. Design/apprentissage du noyau
   - Données structurées - applications
   - Normes ℓ_1 et parcimonie

Rappels d'optimisation

- Livre très utile (et gratuit!):
  S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge Univ. Press, 2003.

Rappels d'optimisation

- Problème général, $x \in \mathcal{X}$

$$\begin{aligned} \text{minimiser} \quad & f(x) \\ \text{soumis a} \quad & h_i(x) = 0, \ \forall i = 1, \ldots, m \\ & g_j(x) \leqslant 0, \ \forall j = 1, \ldots, p \end{aligned}$$

- Pas d'hypothèses sur f, g_j, h_i (pour le moment!)
- $f^* \in [-\infty, \infty)$ le minimum global
- Lagrangien: $\lambda \in \mathbb{R}^m$, $\mu \in \mathbb{R}^p_+$

$$\mathcal{L}(x, \lambda, \mu) = f(x) + \sum_i \lambda_i h_i(x) + \sum_j \mu_j g_j(x)$$

Rappels d'optimisation

- Lagrangien: $\lambda \in \mathbb{R}^m$, $\mu \in \mathbb{R}^p_+$

$$\mathcal{L}(x, \lambda, \mu) = f(x) + \sum_i \lambda_i h_i(x) + \sum_j \mu_j g_j(x)$$

- Fonction duale

$$\begin{aligned} q(\lambda, \mu) &= \inf_{x \in \mathcal{X}} \mathcal{L}(x, \lambda, \mu) \\ &= \inf_{x \in \mathcal{X}} \left\{ f(x) + \sum_i \lambda_i h_i(x) + \sum_j \mu_j g_j(x) \right\} \end{aligned}$$

- Problème dual (toujours concave):

$$\min_{\lambda \in \mathbb{R}^m,\ \mu \in \mathbb{R}^p_+} q(\lambda, \mu)$$

Dualité

- d^* maximum global du problème dual
- Dualité faible (toujours vraie) $d^* \leqslant f^*$
- Dualité forte $d^* = f^*$ si:
  - h_j affines, g_i convexes, f convexe
  - Condition de Slater (point primal strictement faisable)
  - Conditions nécessaires et suffisantes d'optimalité:
    - x^* minimizes $\mathcal{L}(x, \lambda^*, \mu^*)$
    - "complementary slackness": $\forall i, j, \lambda_j^* h_j(x^*) = 0,\ \mu_i^* h_i(x^*) = 0$
- preuve...

Algorithmes d'apprentissage "linéaires" et régularisation

- Données: $x_i \in \mathcal{X}$, $y_i \in \mathcal{Y}$, $i = 1, \ldots, n$
- Minimiser par rapport à $f \in \mathcal{F}$:

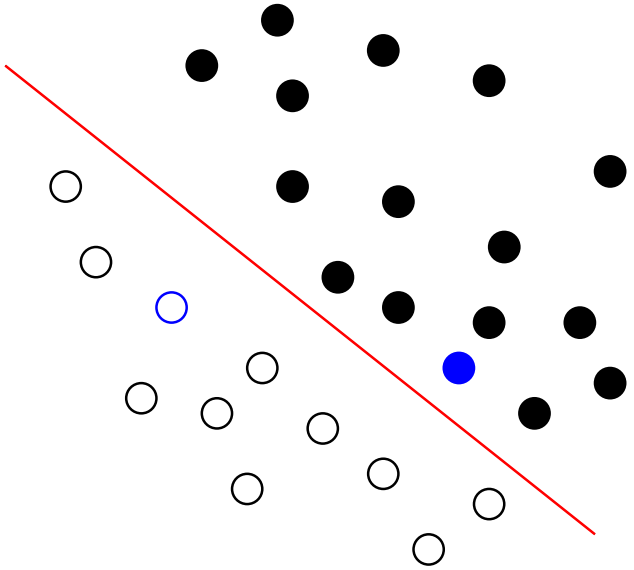
$$\sum_{i=1}^{n} \ell(y_i, f(x_i)) \qquad + \qquad \frac{\lambda}{2}\|f\|_{\mathcal{F}}^2$$

Erreur sur les données $\quad + \quad$ Régularisation

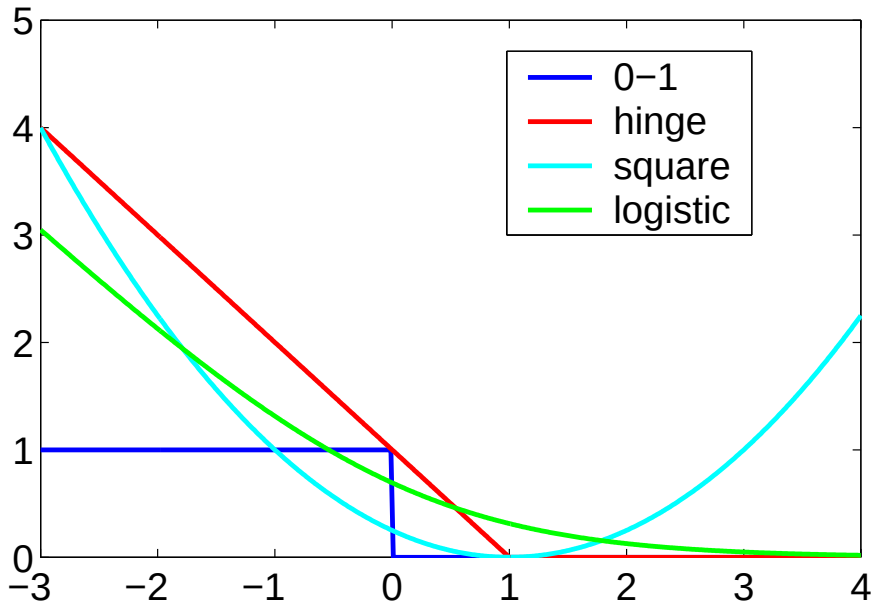
- **Régression linéaire**: $y \in \mathbb{R}$, prédiction $\hat{y} = f(x)$, coût quadratique $\ell(y, f) = \frac{1}{2}(y - \hat{y})^2 = \frac{1}{2}(y - f)^2$

Coûts pour la classification linéaire

- **Classification linéaire**: $y \in \{-1, 1\}$
  prédiction $\hat{y} = \text{signe}(f(x))$



- coût de la forme $\ell(y, f) = \ell(yf)$
- "Vrai" coût: $\ell(yf) = 1_{yf<0}$
- Coûts convexes classiques:



Support vector machine (SVM)

- Données: $x_i \in \mathbb{R}^p$, $y_i \in \{-1, 1\}$, $i = 1, \ldots, n$

- Problème primal:

$$\begin{aligned} \text{minimiser} \quad & \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n} \xi_i \\ \text{soumis a} \quad & \xi_i \geqslant 0,\ \xi_i \geqslant 1 - y_i(w^\top x_i + b),\ \forall i \end{aligned}$$

- Problème dual:

$$\begin{aligned} \text{maximiser} \quad & \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j K_{ij} \\ \text{soumis a} \quad & 0 \leqslant \alpha_i \leqslant C,\ \forall i \\ & \sum_{i=1}^{n} \alpha_i y_i = 0 \end{aligned}$$

Support vector machine (SVM)

- Problème dual:

$$\begin{aligned} \text{maximiser} \quad & \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j K_{ij} \\ \text{soumis a} \quad & 0 \leqslant \alpha_i \leqslant C, \ \forall i \\ & \sum_{i=1}^{n} \alpha_i y_i = 0 \end{aligned}$$

- A l'optimum, $w = \sum_{i=1}^{n} \alpha_i x_i$
- Conditions d'optimalité - vecteur supports
- Interprétation géométrique
- Kernelization

Algorithmes pour la SVM

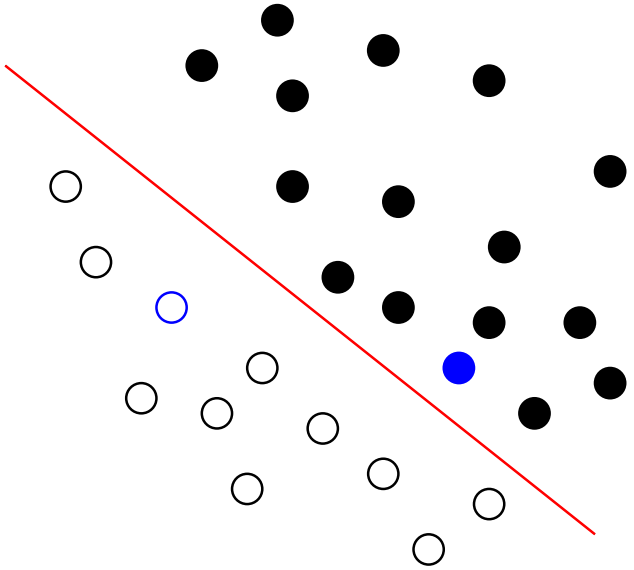
- Programmation quadratique $O(n^{3,5})$ en général pour précision maximale

- Précision requise $> 10^{-16}$, i.e., $\approx n^{-1/2}$
  - Algorithmes du premier ordre effiaces
  - complexité pratique de l'ordre de $O(n^2)$ (SMO)

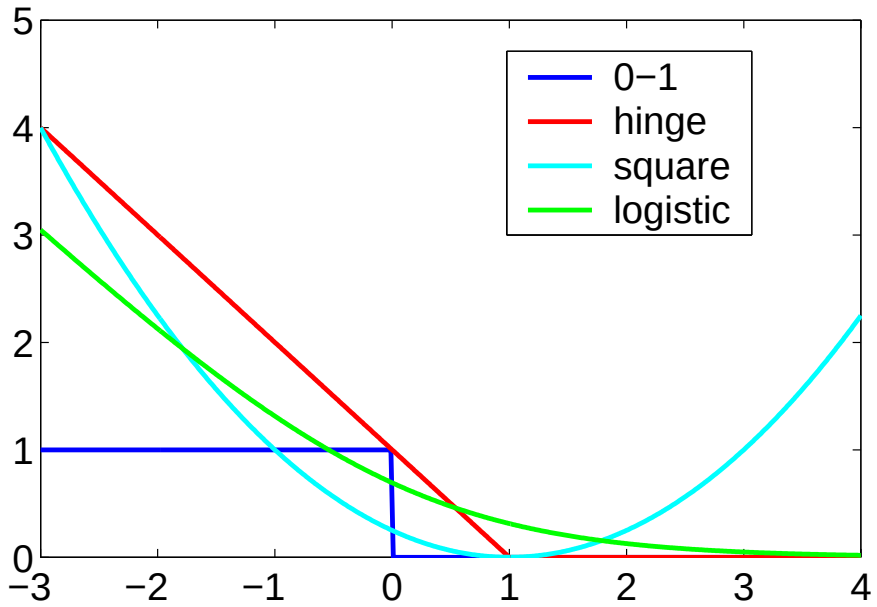
- Algorithmes de chemin (Hastie et al., 2004)

Coûts pour la classification linéaire

- **Classification linéaire**: $y \in \{-1, 1\}$
  prédiction $\hat{y} = \text{signe}(f(x))$



- coût de la forme $\ell(y, f) = \ell(yf)$
- "Vrai" coût: $\ell(yf) = 1_{yf<0}$
- Coûts convexes classiques:



Régression logistique

- Données: $x_i \in \mathbb{R}^p$, $y_i \in \{-1, 1\}$, $i = 1, \ldots, n$
- Problème primal:

$$\text{minimiser} \qquad \frac{\lambda}{2}\|w\|^2 + \sum_{i=1}^{n} \log(1 + \exp(-y_i(w^\top x_i + b))$$

- Coût différentiable
  - $O(n^3)$ si kernelisé
  - $O(nd^2 + d^3)$ si l'input space a d dimensions
- Comparaison régression logistique / SVM

SVM multi-classes

- Plusieurs stratégies

  1. pertes dédiées
  2. Utilisation de SVM binaires
     - “one-vs-one”
     - “one-vs-rest”

Estimation de support

- Problème primal de la SVM:

$$\begin{aligned} \text{minimiser} \quad & \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i \\ \text{soumis a} \quad & \xi_i \geqslant 0,\ \xi_i \geqslant 1 - y_i(w^\top x_i + b),\ \forall i \end{aligned}$$

- Et si toutes les étiquettes sont égales à 1?

$$\begin{aligned} \text{minimiser} \quad & \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i \\ \text{soumis a} \quad & \xi_i \geqslant 0,\ w^\top x_i + b \geqslant 1 - \xi_i,\ \forall i \end{aligned}$$

Méthodes à noyaux - Résumé

- Classification / régression
- Kernel PCA / CCA
- Autres
  - Clustering
  - Ranking
  - etc...

Théorie de l'apprentissage pour les méthodes à noyaux

- Classification avec perte $\sum_{i=1}^{n} \phi(y_i f(x_i))$
- $\hat{f}_n$ estimateur à partir de n points sur la boule $\{f, \|f\|_{\mathcal{F}} \leqslant B\}$
- ϕ-perte $= L_\phi(f) = \mathbb{E}\phi(Y f(X))$
- Résultat 1

$$EL_\phi(\hat{f}_n) - L_\phi^* \leqslant \frac{8 L_\phi B}{\sqrt{n}} + \left[\inf_{\|f\|_{\mathcal{F}} \leqslant B} L_\phi(f) - L_\phi^* \right]$$

- Résultat 2 (Liens avec la "vraie" perte), $\forall f$:

$$L(f) - L^* \leqslant \psi(L_\phi(f) - L_\phi^*)$$

Choix du noyau - données vectoriels

- Noyau linéaire : choix de C
- Noyau polynomial : choix de C et de l'ordre
- Noyau Gaussien : choix de C et largeur de bande
  - grande largeur de bande $=$ noyau linéaire
  - faible largeur de bande $\approx$ plus proche voisin
- Validation croisée ou optimization des bornes?
- Données non vectorielles - autres noyaux?

Plan du cours

1. Noyaux et espaces de Hilbert à noyaux reproduisants (RKHS)
   - Noyaux définis positifs, Noyaux de Mercer, RKHS
2. Méthodes à noyaux générales
   - Astuce du noyau et théorème du représentant
   - Kernel ridge regression, Kernel PCA / CCA
3. Méthodes à noyaux et optimisation convexe
   - Rappels d'optimisation convexe
   - Support vector machines
4. Design/apprentissage du noyau
   - Données structurées - applications
   - Normes ℓ_1 et parcimonie

Données structurées

- L'input space $\mathcal{X}$ est arbitraire!

- Domaines d'applications avec données structurées
  - Traitement du texte
  - Bioinformatique
  - Analyse d'image

- Principes de construction de noyau
  - $\Phi(x)$ explicite, $k(x, y)$ calculé comme $\langle \Phi(x), \Phi(y) \rangle$
  - $\Phi(x)$ explicite très grand, $k(x, y)$ simple à calculer
  - $\Phi(x)$ implicite très grand, $k(x, y)$ simple à calculer

Noyaux pour documents

- Document représenté par le compte de mots
- $\Phi(x)_{\text{mot}} =$ nombre d'occurence du mot dans le document x
- Très utilisé en texte

Noyaux pour séquences

- Séquences $=$ suite finite (de longueur arbitraire) déléments d'un alphabet Σ
- Feature space indexé par toutes les séquences possibles s
- $\Phi(x)_s =$ nombre d'occurence de s dans x
- noyau $k(x, y) = \sum_s \langle \Phi(x)_s, \Phi(y)_s \rangle$
- calculable en temps polynomial
- Variantes

Noyaux pour images (Harchaoui & Bach, 2007)

- La plupart des applications des méthodes à noyaux:
  - Construction d'une large base de descripteurs (e.g., ondelettes)
  - Utiliser une SVM avec beaucoup de points étiquetés
- Développer des noyaux spécifiques
  - Utiliser la structure naturelle des images
  - Information *a priori* pour réduire le nombre de données étiquetées

Noyaux pour images

- Représentations et noyaux classiques
  - Vecteurs de pixels + noyaux entre vecteurs
  - "Sacs" de pixels ou de pixels filtrés + noyaux entre histogrammes

    $\Rightarrow$ Géométrie globale naturelle peu utilisée

- Utilisation de la géométrie?
  - Extraction de points saillants (e.g., descripteurs SIFT, Lowe, 2004)
  - Segmentation

Segmentation

- But: Extraire des objets d'intérêt
- Beaucoup de méthodes disponibles, ...
  ... mais, trouvent rarement l'objet d'intérêt en entier
- Graphes de segmentation
  - Permet de travailler sur des sur-segmentations "plus sûres"
  - D'une grande trame carrée (millions de pixels) à un petit graphe (dizaines ou centaine de noeuds)

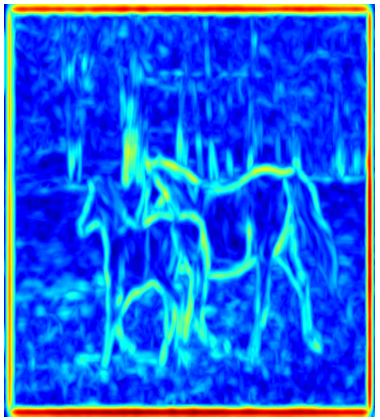
Graphe de segmentation

- Méthode de segmentation
  - Gradient LAB avec filtres de contours orientés (Malik et al, 2001)
  - Ligne de partage des eaux avec post-traitement (Meyer, 2001)
  - Très rapide

Ligne de partage des eaux

image

gradient



watershed

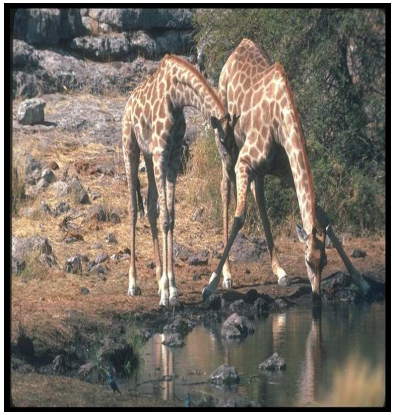
287 segments

64 segments

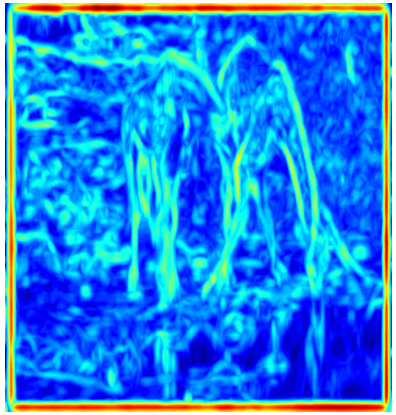
10 segments

Ligne de partage des eaux

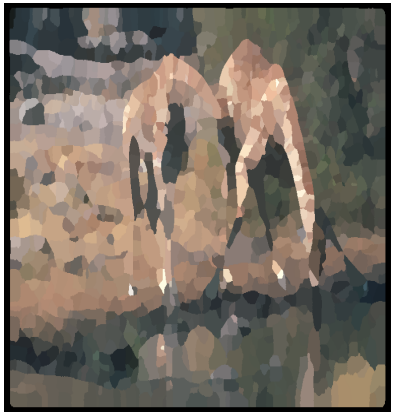
image



gradient



watershed



287 segments

64 segments

10 segments

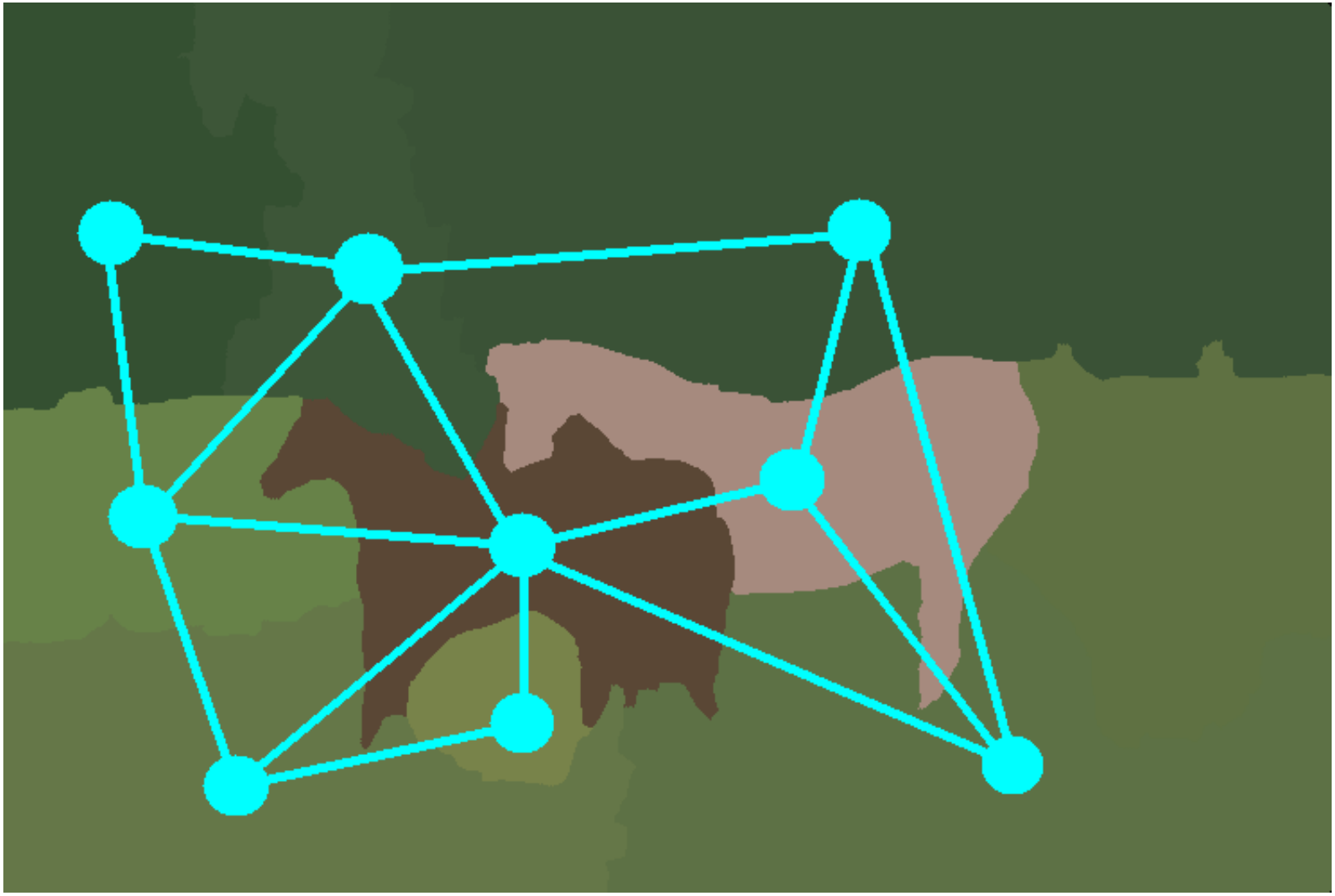
Graphe de segmentation

- Méthode de segmentation
  - Gradient LAB avec filtres de contours orientés (Malik et al, 2001)
  - Ligne de partage des eaux avec post-traitement (Meyer, 2001)
  - Très rapide

- Graphe étiqueté non orienté
  - Sommets: régions connexes
  - Arêtes: entre régions voisines
  - Etiquettes: ensemble des pixels de la région

- Difficultés
  - Etiquettes de très grande dimension
  - Graphe planaire non orienté
  - Nécessite des comparisons inexactes entre graphes

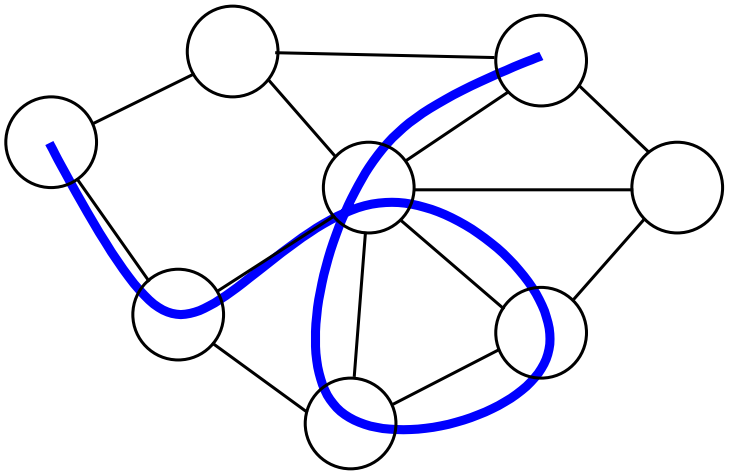
Graphes de segmentation

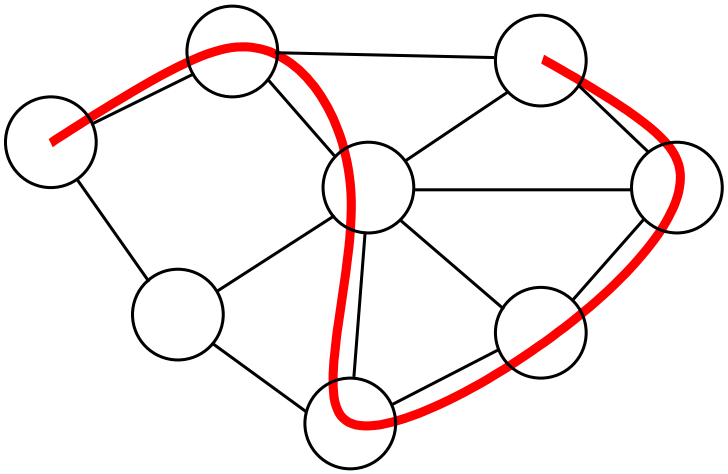


Chemins et marches

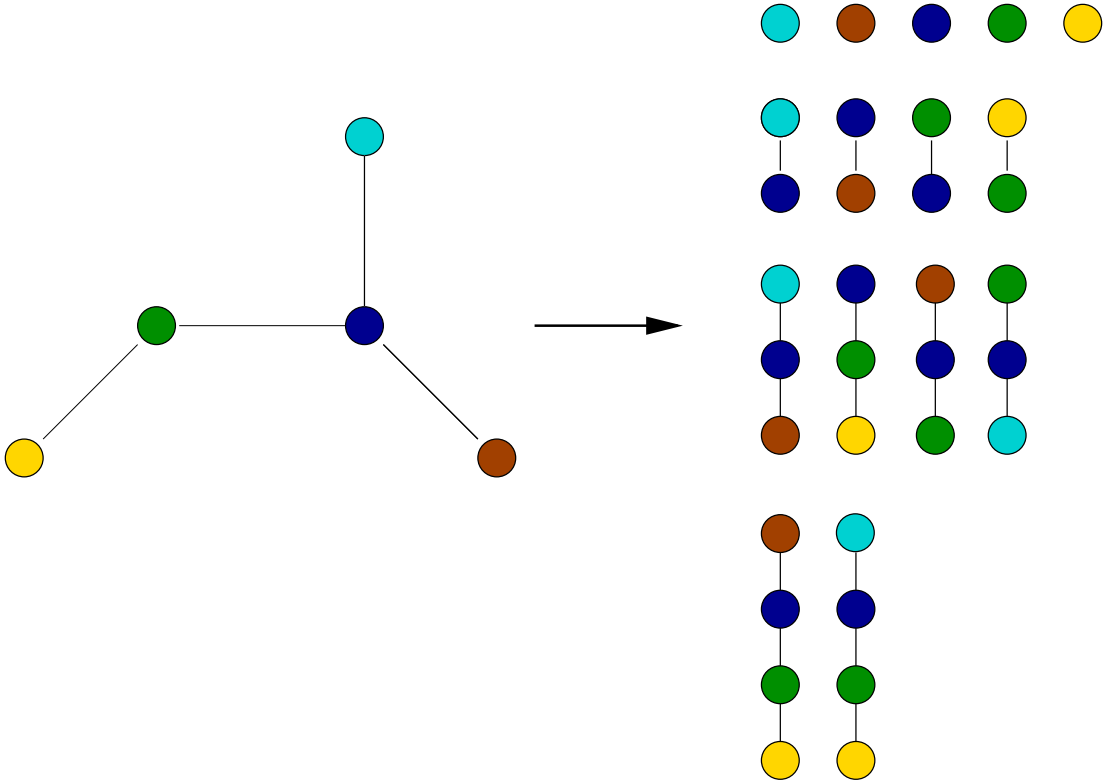
- Etant donné un graphe G,
  - Un chemin est une suite de sommets voisins distincts
  - Une marche est une suite de sommets voisins

- Notions apparemment similaires

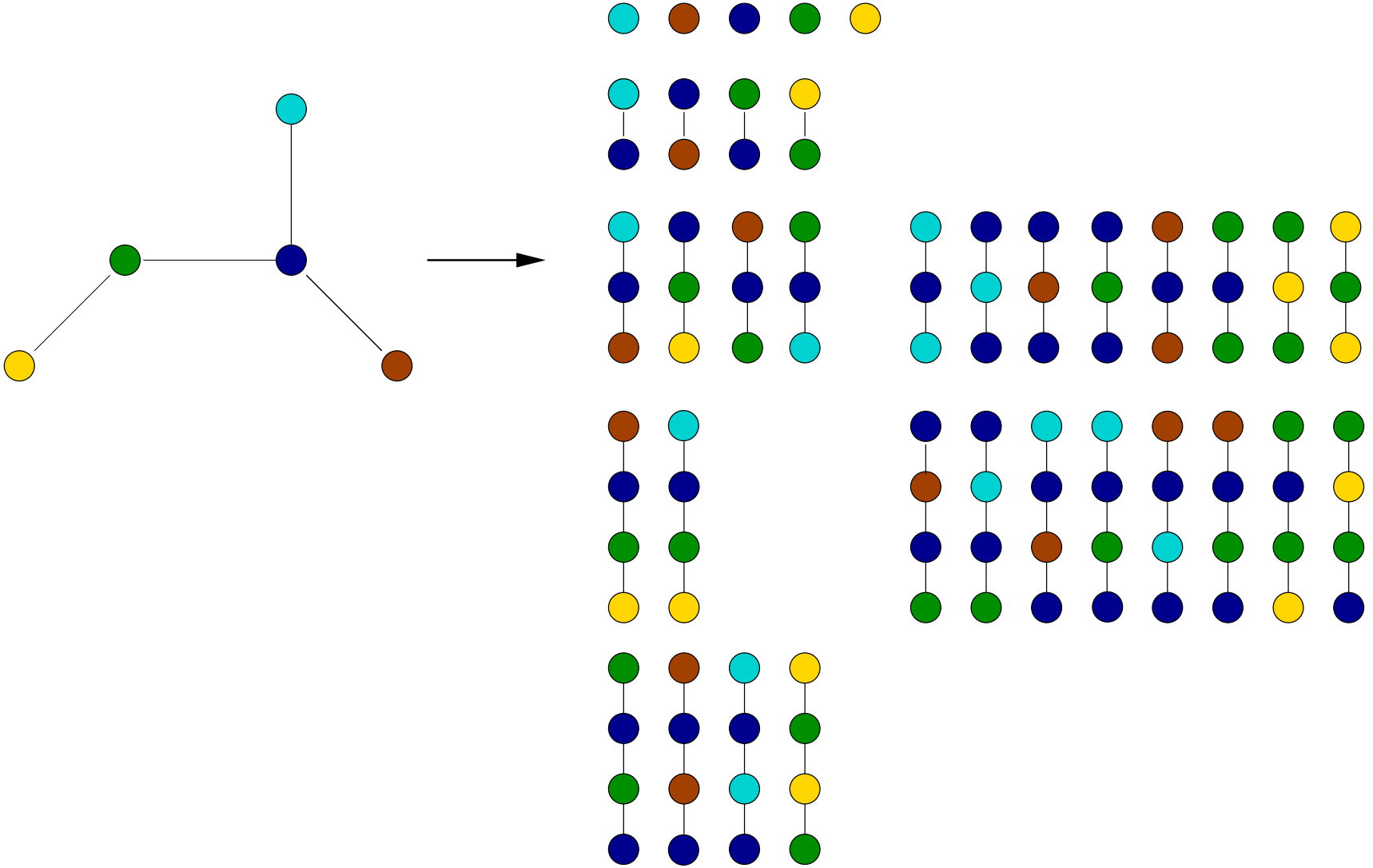




Chemins



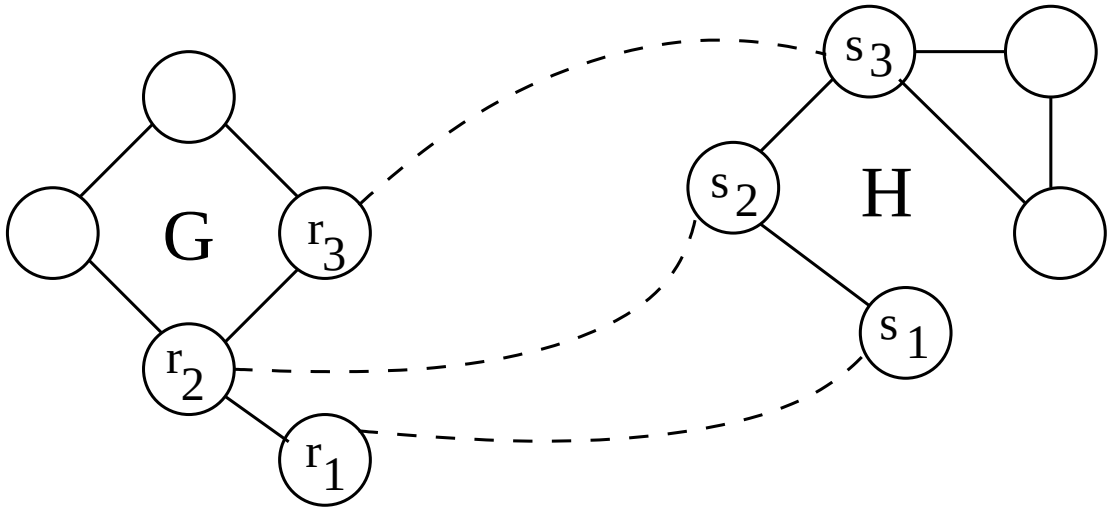
Marches



Noyaux de marches

- $\mathcal{W}^p_{\mathbf{G}}$ (resp. $\mathcal{W}^p_{\mathbf{H}}$) = marches de longueur p dans $\mathbf{G}$ (resp. $\mathbf{H}$)
- *Noyaux de bases* sur les étiquettes $k(\ell, \ell')$
- **Proposition/définition**: noyaux de marches d'ordre p:

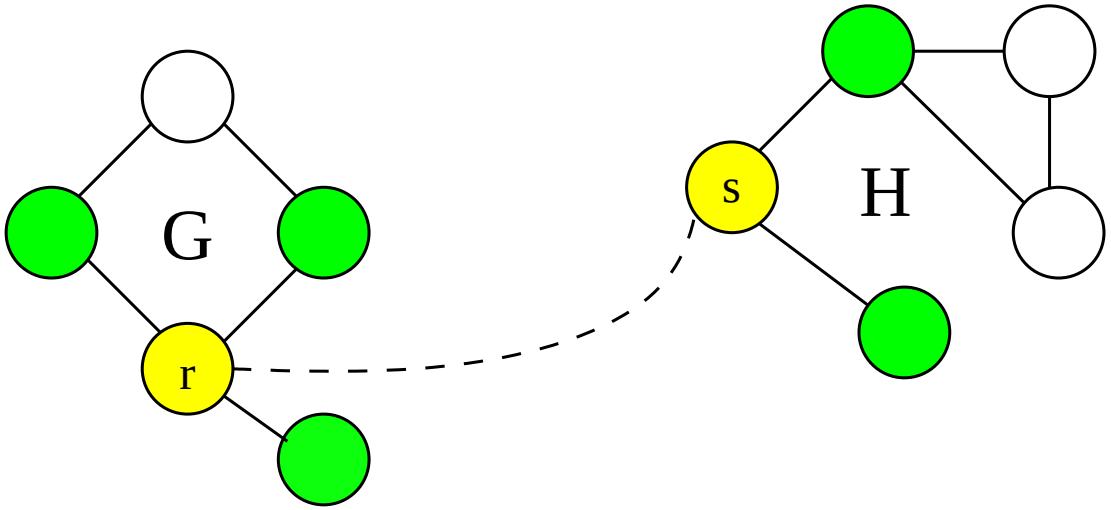
$$k^p_{\mathcal{W}}(\mathbf{G}, \mathbf{H}) = \sum_{\substack{(r_1, \ldots, r_p) \in \mathcal{W}^p_{\mathbf{G}} \\ (s_1, \ldots, s_p) \in \mathcal{W}^p_{\mathbf{H}}}} \prod_{i=1}^{p} k(\ell_{\mathbf{H}}(r_i), \ell_{\mathbf{G}}(s_i)).$$



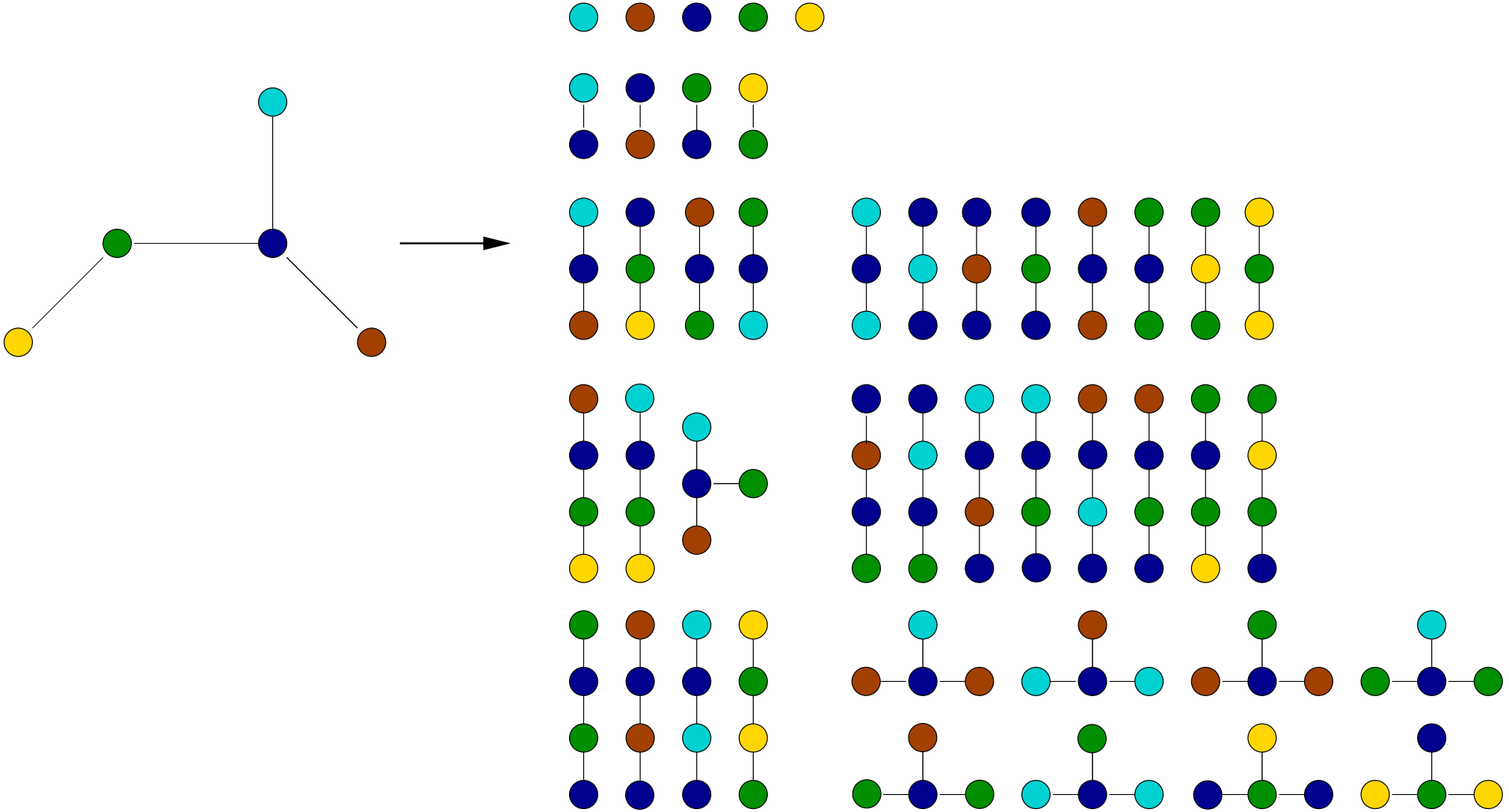
Programmation dynamique pour le noyau de marches

- Programmation dynamique en $O(pd_{\mathbf{G}}d_{\mathbf{H}}n_{\mathbf{G}}n_{\mathbf{H}})$
- $k^p_{\mathcal{W}}(\mathbf{G},\mathbf{H},r,s) =$ somme restreinte aux marches démarrant de r et s
- **Proposition**: Récurrence entre ordre $p-1$ et p

$$k^p_{\mathcal{W}}(\mathbf{G},\mathbf{H},r,s) = k(\ell_{\mathbf{H}}(r),\ell_{\mathbf{G}}(s)) \sum_{\substack{r' \in \mathcal{N}_{\mathbf{G}}(r) \\ s' \in \mathcal{N}_{\mathbf{H}}(s)}} k^{p-1}_{\mathcal{W}}(\mathbf{G},\mathbf{H},r',s').$$

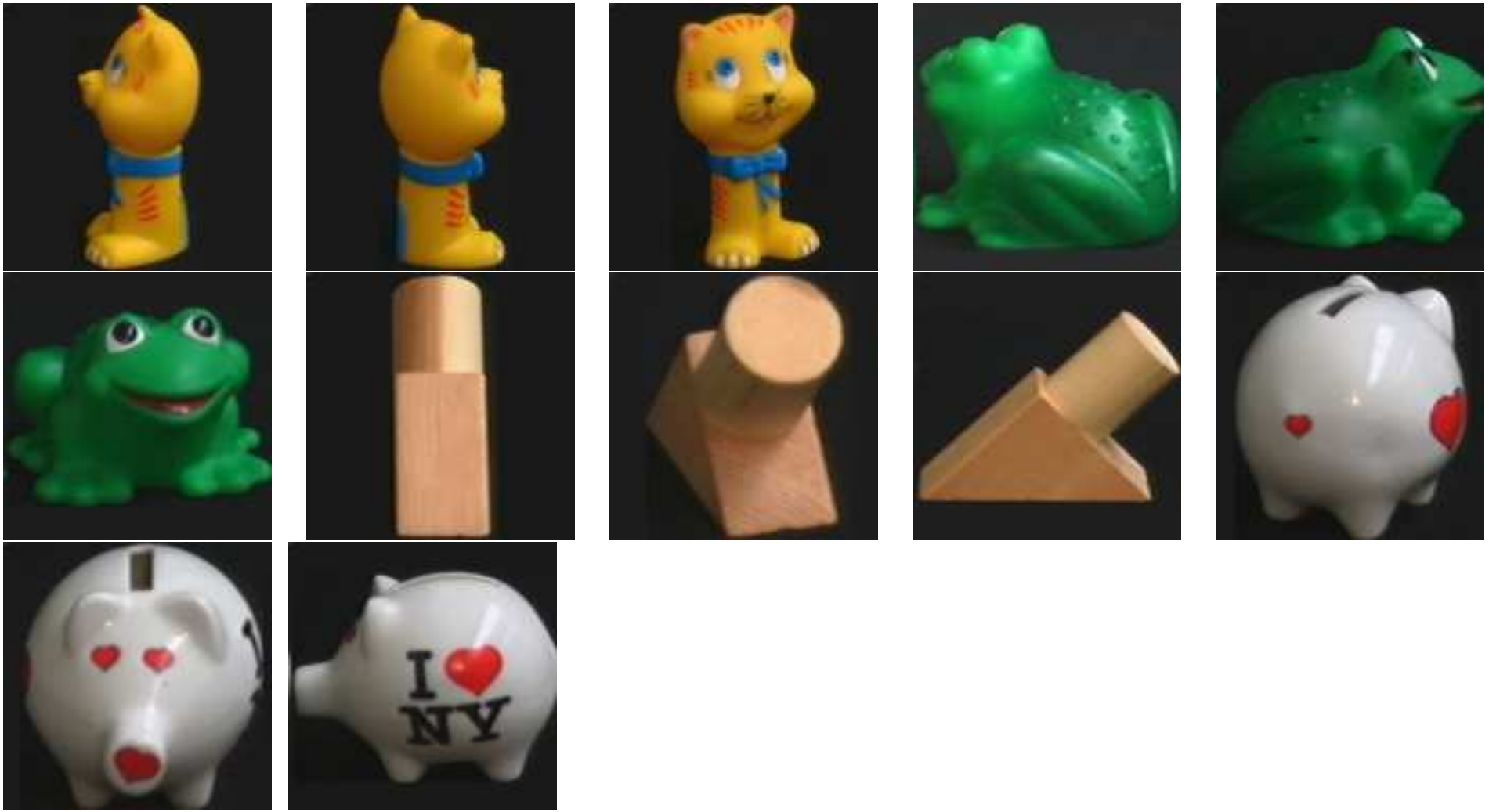


Extensions naturelles aux sous-arbres



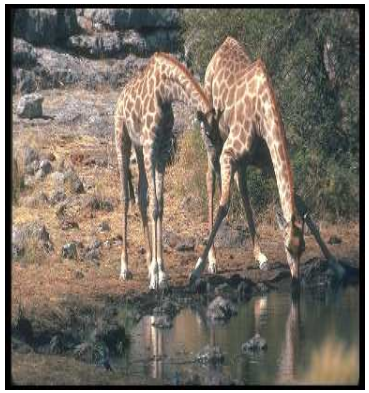
Expériences de classification

- `Coil100`: base de 7200 images de 100 *objets sur un fond uniforme*, avec 72 images par objet.



Expériences de classification

- `Corel14`: base de 1400 *images naturelles* avec 14 classes différentes






Comparaison de noyaux

- Noyaux :
  - noyaux entre histogrammes (**H**)
  - noyaux de marches (**W**)
  - noyaux de sous-arbres (**TW**)
  - noyaux de sous-arbres pondérés (**wTW**)
  - combinaison par algorithmes de noyaux multiples (**M**)

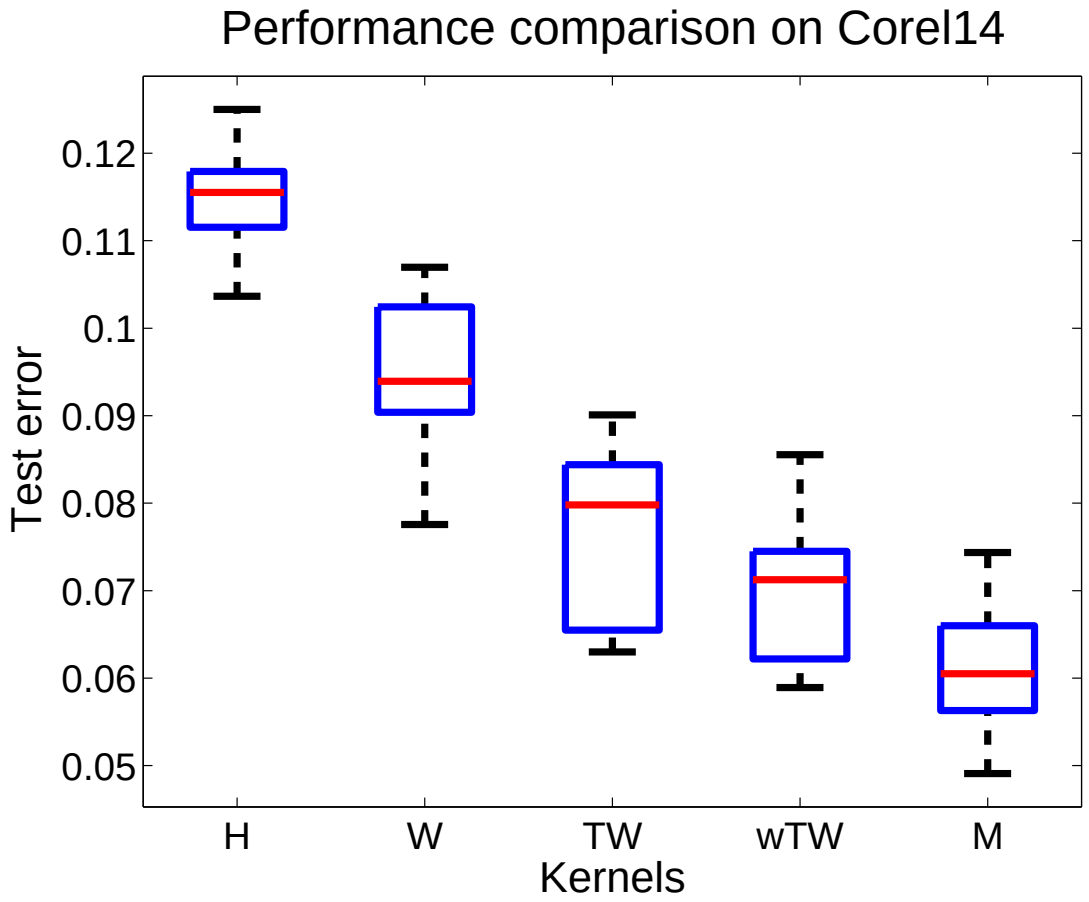
- Hyperparamètres sélectionnés par validation croisée

- Taux d'erreur moyens sur 10 réplications:

	H	W	TW	wTW	M
`Coil100`	1.2%	0.8%	0.0%	0.0%	0.0%
`Corel14`	10.36%	8.52%	7.24%	6.12%	5.38%

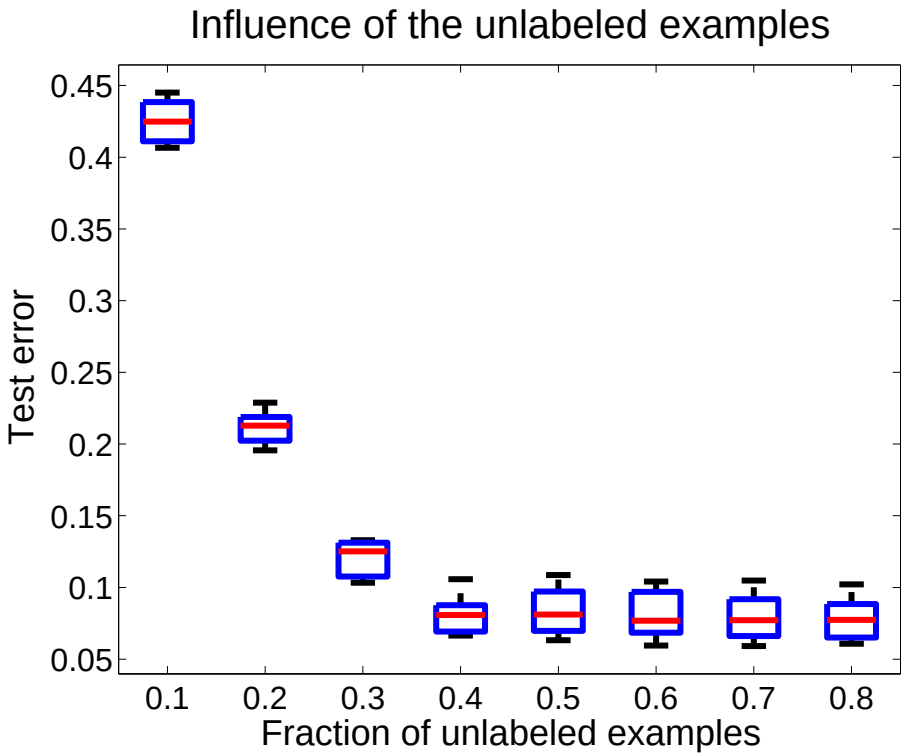
Performance sur Corel14

- noyaux entre histogrammes (**H**)
- noyaux de marches (**W**)
- noyaux de sous-arbres (**TW**)
- noyaux de sous-arbres pondérés (**wTW**)
- combinaison (**M**)



Apprentissage semi-supervisé

- Les méthodes à noyaux permettent la flexibilité
- Exemple: apprentissage semi-supervisé (Chapelle et Zien , 2004)
- 10% d'exemples étiquetés, 10% d'exemples de test, 10% to 80% d'exemples non étiquetés



Normes ℓ_1

- Cadre classique (linéaire): $\min_{w \in \mathbb{R}^d} \sum_{i=1}^{n} \ell(y_i, w^\top x_i) + \lambda \sum_{j=1}^{p} w_j^2$

- Cadre "parcimonieux": $\min_{w \in \mathbb{R}^d} \sum_{i=1}^{n} \ell(y_i, w^\top x_i) + \lambda \sum_{j=1}^{p} |w_j|$

- Propriétés:
  - Solution w parcimonieuse
  - Algorithme efficaces

- Etude des propriétés

Normes ℓ_1

- Coût quadratique (LASSO):

$$\min_{w \in \mathbb{R}^d} \|y - Xw\|^2 + \lambda \sum_{j=1}^{p} |w_j|$$

- Exemple simple: features indépendants
- Parcimonie?
- Etude détaillée:
  - Conditions d'optimalité (optimisation!)
  - Algorithmes (très) efficaces de chemin de régularisation
  - Consistence pour l'estimation du modèle?

Conditions d'optimalité

- w avec $J = \{j,\ w_j \neq 0\}$ est optimal ssi

$$X_J^\top X_J w_J - X_J^\top Y + \lambda \text{sign}(w_J) = 0$$

$$\|X_{J^c}^\top X_J w_J - X_J^\top Y\|_\infty \leqslant \lambda$$

- Preuve...

Algorithme de chemin

- Algorithme du LARS (Least angle regression)
- Si le modèle (signes) est connu, alors

$$w_J = (X_J^\top X_J)^{-1} X_J^\top Y - \lambda (X_J^\top X_J)^{-1} \mathrm{sign}(w_J)$$

- Affine en λ
- Tout le chemin pour le coût d'une inversion de matrice

Consistence d'estimation du modèle

- w supposé parcimonieux pour le modèle
- Théorème, 2007: Estimateur du modèle est consistent ssi

$$\|X_{J^c}^\top X_J (X_J^\top X_J)^{-1} \mathrm{sign}(w_j)\|_\infty \leqslant 1$$

- Peu de corrélation pour l'optimalité
- NB: extension à l'estimation du rang

Apprentissage avec noyaux multiples (Bach et al, 2004)

- Cadre limité à $K = \sum_{j=1}^{m} \eta_j K_j$, $\eta \geqslant 0$

- Interprétation en termes de normes ℓ_1 par blocs
  - m "feature maps" $\Phi_j : \mathcal{X} \mapsto \mathcal{F}_j$, $j = 1, \ldots, m$.
  - Minimisation par rapport à $w_1 \in \mathcal{F}_1, \ldots, w_m \in \mathcal{F}_m$
  - Prédicteur: $f(x) = {w_1}^\top \Phi_1(x) + \cdots + {w_m}^\top \Phi_m(x)$

$$x \begin{array}{c} \nearrow \\ \longrightarrow \\ \searrow \end{array} \begin{array}{cc} \Phi_1(x)^\top & w_1 \\ \vdots & \vdots \\ \Phi_j(x)^\top & w_j \\ \vdots & \vdots \\ \Phi_m(x)^\top & w_m \end{array} \begin{array}{c} \searrow \\ \longrightarrow \\ \nearrow \end{array} w_1^\top \Phi_1(x) + \cdots + w_m^\top \Phi_m(x)$$

  - Parcimonie par blocs $\Rightarrow$ régularisation par blocs: $\|w_1\| + \cdots + \|w_m\|$

Apprentissage du noyau
Noyaux multiples - dualité (Bach et al, 2004)

- Problème d'optimisation primal:

$$\sum_{i=1}^{n} \phi_i(w_1^\top \Phi_1(x_i) + \cdots + w_m^\top \Phi_m(x_i)) + \frac{\lambda}{2} \left(\|w_1\| + \cdots + \|w_m\|\right)^2$$

- **Proposition**: Problème dual (obtenu par cônes du second-ordre)

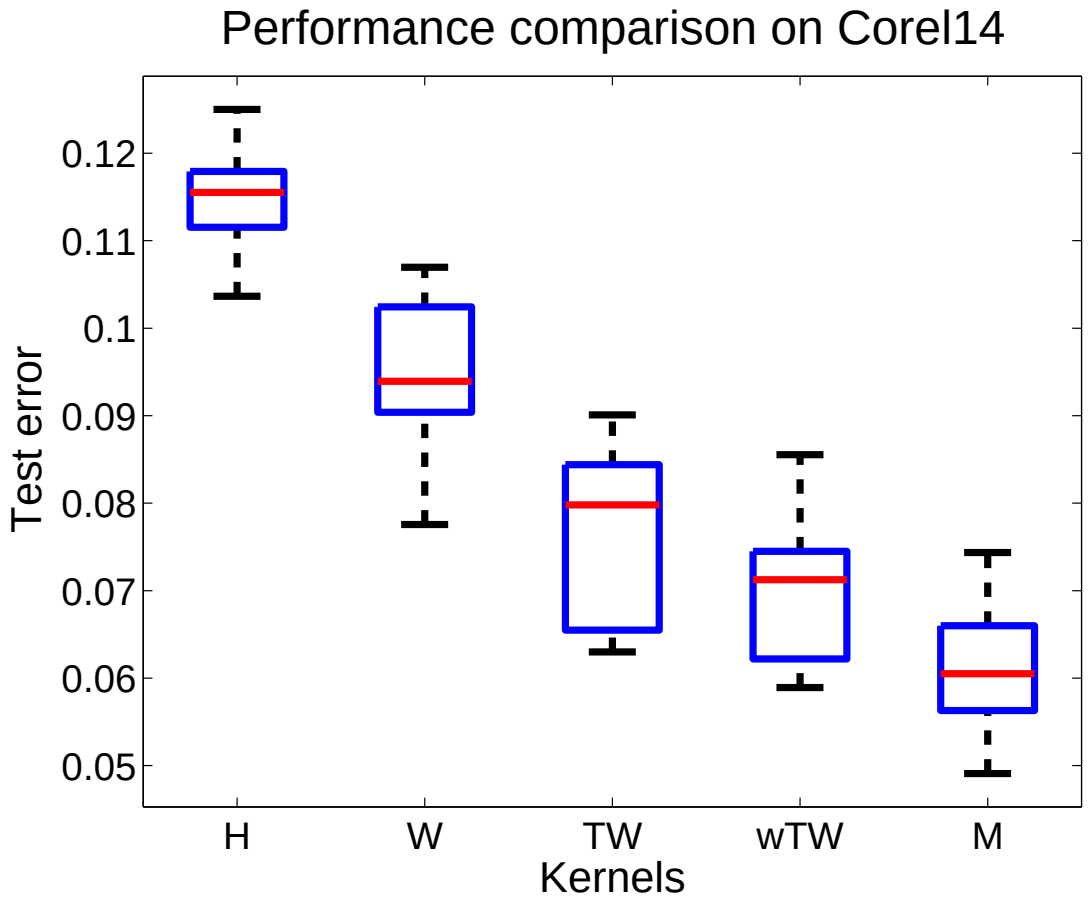
$$\max_{\alpha \in \mathbb{R}^n} -\sum_{i=1}^{n} \phi_i^*(-\lambda \alpha_i) - \frac{\lambda}{2} \min_{j \in \{1,\ldots,m\}} \alpha^\top K_j \alpha$$

Conditions de KKT: $w_j = \eta_j \sum_{i=1}^{n} \alpha_i \Phi_j(x_i)$
avec $\alpha \in \mathbb{R}^n$ and $\eta \geqslant 0$, $\sum_{j=1}^{m} \eta_j = 1$

– α est la solution duale pour le problème à noyaux simple et matrice de noyau $K(\eta) = \sum_{j=1}^{m} \eta_j K_j$

Image: Performance sur Corel14

- noyaux entre histogrammes (**H**)
- noyaux de marches (**W**)
- noyaux de sous-arbres (**TW**)
- noyaux de sous-arbres pondérés (**wTW**)
- combinaison (**M**)



Application à la bio-informatique (Lanckriet et. al, 2004)

- Prédire la fonction d'une protéine
- Sources de data hétérogènes
  - Séquence d'acide aminés
  - Interaction protéine-protéine
  - Interactions génétiques
  - Données d'expression
- Taux d'erreur passe de 70% à 90%

Plan du cours

1. Noyaux et espaces de Hilbert à noyaux reproduisants (RKHS)
   - Noyaux définis positifs, Noyaux de Mercer, RKHS
2. Méthodes à noyaux générales
   - Astuce du noyau et théorème du représentant
   - Kernel ridge regression, Kernel PCA / CCA
3. Méthodes à noyaux et optimisation convexe
   - Rappels d'optimisation convexe
   - Support vector machines
4. Design/apprentissage du noyau
   - Données structurées - applications
   - Normes ℓ_1 et parcimonie

Noyaux définis positifs

- Fonction $k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$
- Symmétrique: $\forall x, y \in \mathcal{X},\ k(x, y) = k(y, x)$
- Condition de positivité : $\forall x_1, \ldots, x_n \in \mathcal{X}$, la matrice de noyau K est définie positive, i.e.,

$$\forall \alpha \in \mathbb{R}^n,\ \alpha^\top K \alpha = \sum_{i,j=1}^{n} \alpha_i \alpha_j k(x_i, x_j) \geqslant 0$$

Noyaux définis positifs = produits scalaires

- Théorème (Aronszajn, 1950): k est un noyau d.p. ssi il existe un espace de Hilbert $\mathcal{F}$ et un "feature map" $\Phi : \mathcal{X} \mapsto \mathcal{F}$ tels que

$$k(x, y) = \langle \Phi(x), \Phi(y) \rangle_{\mathcal{F}}$$

- Remarques:
  - $\mathcal{F}$ peut avoir une dimension infinite
  - Φ souvent implicite!

Définition d'un RKHS

- Soit $\mathcal{X}$ un ensemble quelconque et $\mathcal{F}$ un sous-espace de des fonctions de $\mathcal{X}$ dans $\mathbb{R}$, qui est muni d'un produit scalaire Hilbertien.

- $\mathcal{F}$ est un RKHS avec noyau reproduisant $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ ssi:
  - $\mathcal{F}$ contient toutes les fonctions de la forme

$$k(x, \cdot) : y \mapsto k(x, y)$$

  - $\forall x \in \mathcal{X}$ and $f \in \mathcal{F}$,

$$f(x) = \langle f, k(\cdot, x) \rangle_{\mathcal{F}}$$

(i.e., $k(\cdot, x)$ correspond au "Dirac" en x)

Théorème du représentant

- Soit $\mathcal{X}$ un ensemble, un noyau d.p. k et son RKHS associé $\mathcal{F}$, et $x_1, \ldots, x_n$ n points dans $\mathcal{X}$.

- Soit $J : \mathbb{R}^{n+1} \to \mathbb{R}$ strictement croissante par rapport à la dernière variable

- Toute solution du problème d'optimisation suivant

$$\min_{f \in \mathcal{F}} J(f(x_1), \ldots, f(x_n), \|f\|_{\mathcal{F}})$$

  s'écrit de la forme $f = \sum_{i=1}^{n} \alpha_i k(\cdot, x_i)$.

- Cadre classique: $\min_{f \in \mathcal{F}} \sum_{i=1}^{n} \ell_i(f(x_i)) + \lambda \|f\|_{\mathcal{F}}^2$

Algorithmes d'apprentissage "linéaires" et régularisation

- Données: $x_i \in \mathcal{X}$, $y_i \in \mathcal{Y}$, $i = 1, \ldots, n$
- Minimiser par rapport à $f \in \mathcal{F}$:

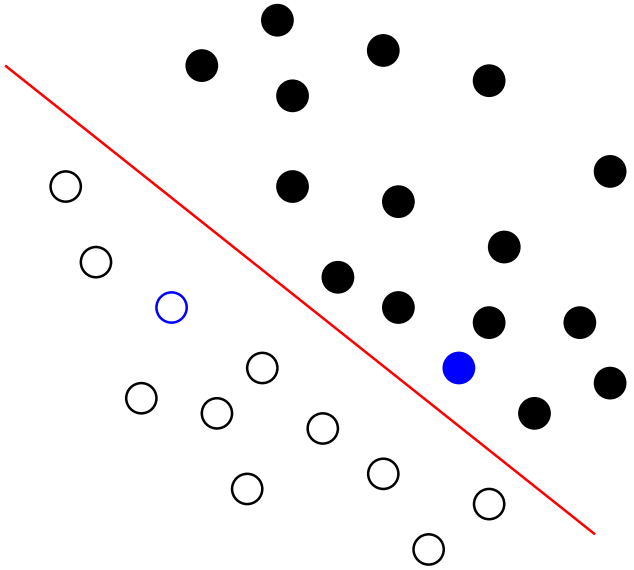
$$\sum_{i=1}^{n} \ell(y_i, f(x_i)) \qquad + \qquad \frac{\lambda}{2}\|f\|_{\mathcal{F}}^2$$

Erreur sur les données $\quad + \quad$ Régularisation

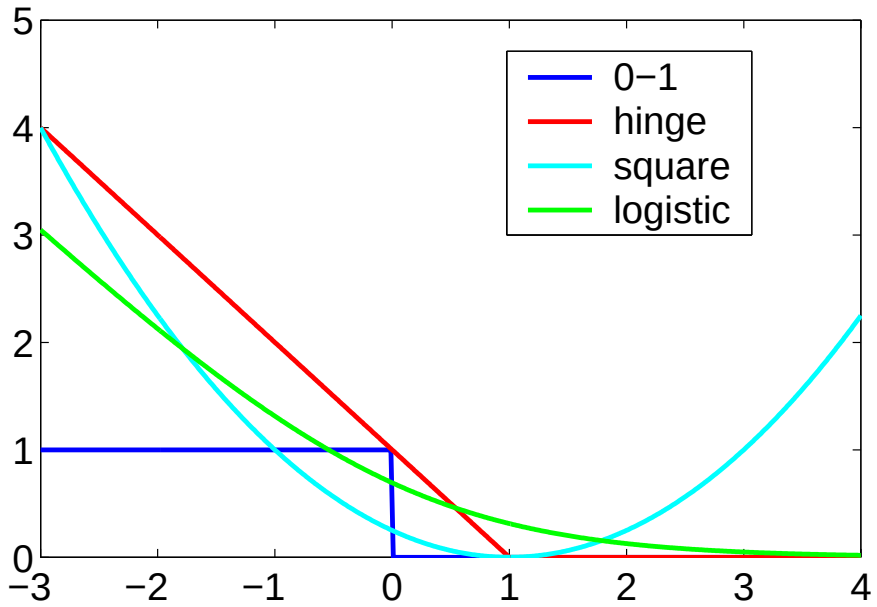
- **Régression** : coût quadratique
- **Classification**

Coûts pour la classification linéaire

- **Classification linéaire**: $y \in \{-1, 1\}$
  prédiction $\hat{y} = \text{signe}(f(x))$



- coût de la forme $\ell(y, f) = \ell(yf)$
- "Vrai" coût: $\ell(yf) = 1_{yf<0}$
- Coûts convexes classiques:



Support vector machine (SVM)

- Données: $x_i \in \mathbb{R}^p$, $y_i \in \{-1, 1\}$, $i = 1, \ldots, n$

- Problème primal:

$$\begin{aligned} \text{minimiser} \quad & \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n} \xi_i \\ \text{soumis a} \quad & \xi_i \geqslant 0,\ \xi_i \geqslant 1 - y_i(w^\top x_i + b),\ \forall i \end{aligned}$$

- Problème dual:

$$\begin{aligned} \text{maximiser} \quad & \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j K_{ij} \\ \text{soumis a} \quad & 0 \leqslant \alpha_i \leqslant C,\ \forall i \\ & \sum_{i=1}^{n} \alpha_i y_i = 0 \end{aligned}$$

Méthodes à noyaux - Résumé

- Classification / régression
- Kernel PCA / CCA
- Autres
  - Clustering
  - Ranking
  - etc...

Méthodes à noyaux - Points chauds

- Normes ℓ_1
- Apprentissage du noyau
- Données structurées en entrée
- Données structurées en sortie
- Collaborative filtering
- Algorithmes pour données massives
- Analyse fine de tests à base de noyaux

Références

- Livres
  - B. Schölkopf and A. J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, 2002.
  - J. Shawe-Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004.
  - G. Wahba. *Spline Models for Observational Data*. SIAM, 1990.
  - S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge Univ. Press, 2003.

- Cours et transparents en ligne
  - Cours de Jean-Philippe Vert (`cbio.ensmp.fr`)