

Apprentissage interactif et multi-classes pour la détection de concepts sémantiques dans des données multimédia

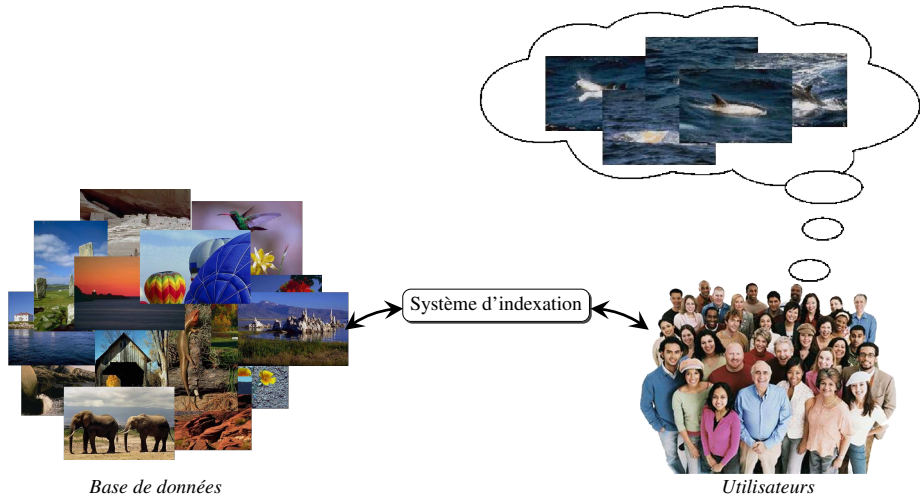
Dirigée par Philippe-Henri Gosselin et Frédéric Precioso

Alexis Lechervy

6 décembre 2012

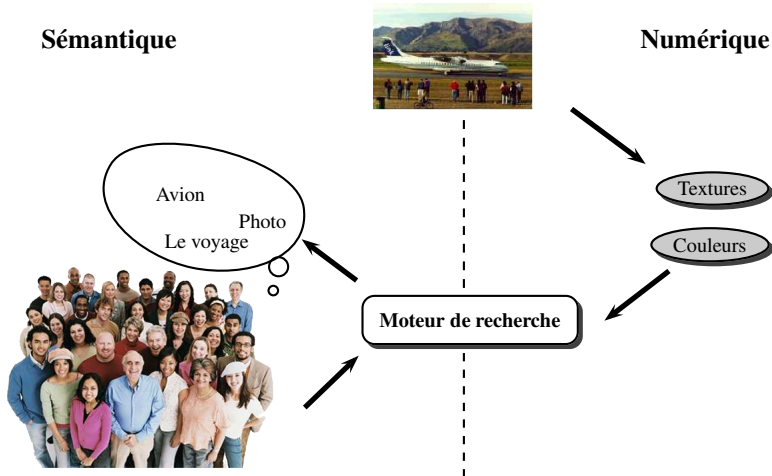


La recherche de données multimédia



Comment aider les utilisateurs à traiter des bases de données toujours plus vastes ?

Le fossé numérique/sémantique



Défi : Combler le fossé sémantique/numérique entre les concepts sémantiques de l'utilisateur et la représentation numérique des images ?



Résumé

- 1 Une méthode de recherche interactive par Boosting
- 2 Boosting pour la construction de fonction noyau
- 3 Conclusion générale et Perspectives



Plan

- 1 Une méthode de recherche interactive par Boosting
 - ▷ Présentation de la recherche interactive
 - ▷ Une introduction au Boosting
 - ▷ Boosting actif pour la recherche interactive d'objets
 - ▷ Expériences
- 2 Boosting pour la construction de fonction noyau
- 3 Conclusion générale et Perspectives



Plan

1 Une méthode de recherche interactive par Boosting



Présentation de la recherche interactive

▷ Une introduction au Boosting



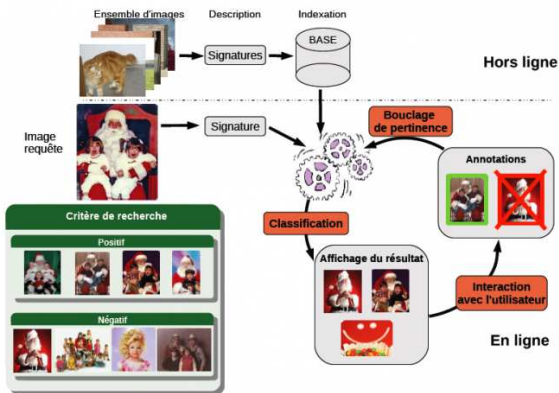
Boosting actif pour la recherche interactive d'objets

▷ Expériences

2 Boosting pour la construction de fonction noyau

3 Conclusion générale et Perspectives

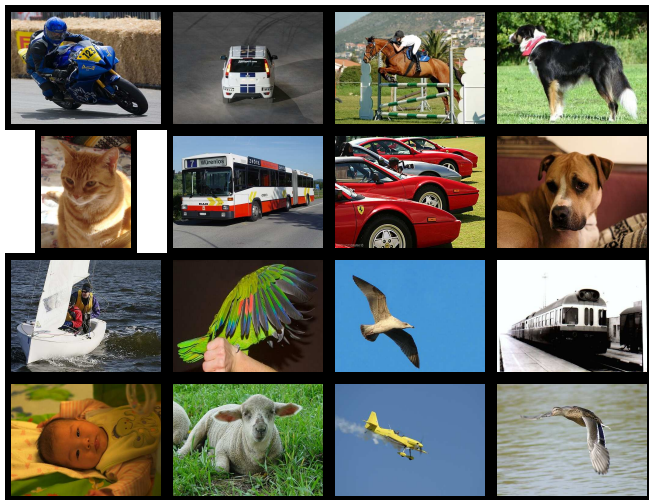
La recherche interactive



Objectif : Réduire le fossé sémantique/numérique.

Solution : Établir un dialogue entre l'utilisateur et le système.

Exemple de recherche interactive



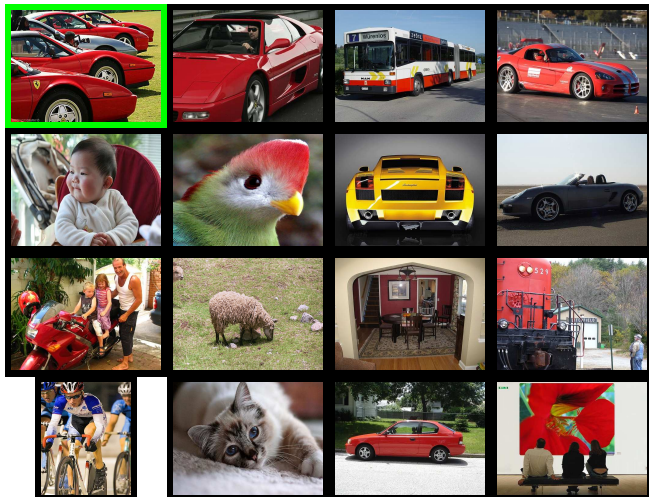
Base d'images à l'état initial

Exemple de recherche interactive



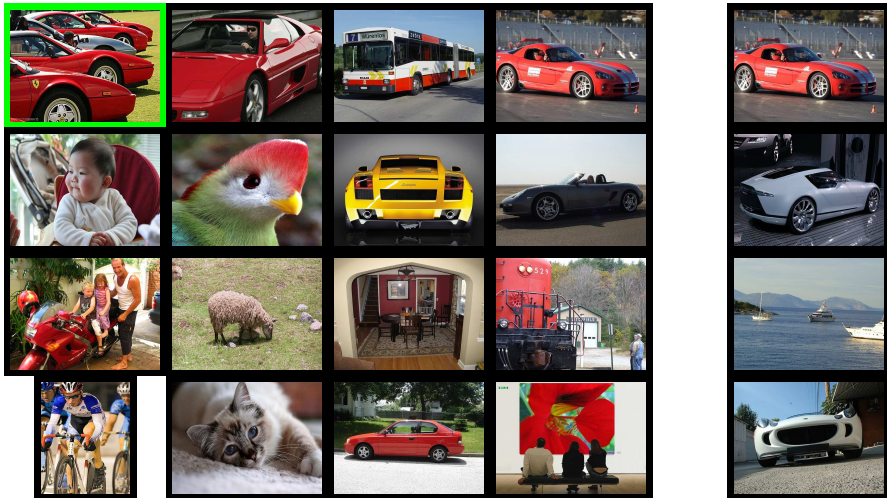
Premières annotations (carré vert= positif, carré rouge= négatif)

Exemple de recherche interactive



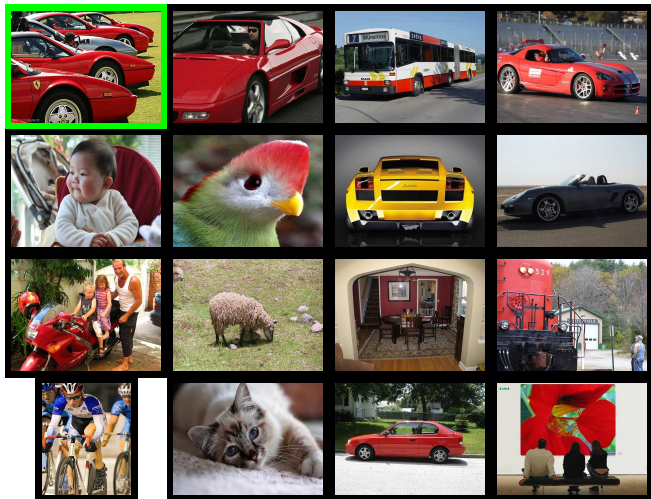
Premier classement

Exemple de recherche interactive



Sélection par stratégie active de nouveaux exemples

Exemple de recherche interactive



Deuxième passe d'annotation

Exemple de recherche interactive



Classement suivant



Plan

1 Une méthode de recherche interactive par Boosting



Présentation de la recherche interactive

▷ Une introduction au Boosting



Boosting actif pour la recherche interactive d'objets

▷ Expériences

2 Boosting pour la construction de fonction noyau

3 Conclusion générale et Perspectives

Boosting

Observations :

- Il est “simple” de construire une règle meilleure qu’un processus aléatoire
 - Si un objet est dans le ciel ALORS c’est un avion
- Il est “difficile” de construire une règle ne commettant pas d’erreur

⇒ Peut-on construire un classifieur fort en combinant des règles simples ?



Boosting

Boosting

- Utilise **un ensemble de règles faibles** pour construire une règle plus forte
- Pour chaque itération, on cherche à **améliorer** la classification des **exemples les moins bien classés**.
- Un algorithme de Boosting utilise généralement **beaucoup d'exemples d'apprentissage**.



Un algorithme de Boosting pour le tri : RankBoost

Buts

- Réalise une classification avec **tri** de l'ensemble des images par score de pertinence.
- Trouver les classifieurs qui maximisent la différence entre les scores des exemples positifs et des exemples négatifs.



Algorithme RankBoost [Freund 2003]

Algorithme

- Pour $t = 1 \dots T$ faire :

- 1 Calculer le score pondéré des classifieurs h ,

$$r_t(h) = \sum_{x_p, x_n} D_t(x_p, x_n)(h(x_p) - h(x_n))$$

- 2 Récupérer le classifieur faible maximisant ce score
- 3 Choisir $\alpha = \frac{1}{2} \ln \left(\frac{1+r}{1-r} \right)$
- 4 Mettre à jour la pondération des couples d'exemples

$$D_{t+1}(x_p, x_n) = \frac{D_t(x_p, x_n) \exp(\alpha_t(h_t(x_n) - h_t(x_p)))}{Z_t}$$

- Résultat :

$$H = \sum_{t=1}^T \alpha_t h_t(x)$$



Bilan sur la recherche interactive et le Boosting

La recherche interactive

- Protocole de recherche d'image permettant de mieux **prendre en compte le fossé sémantique/numérique**.
- Permet une **annotation plus efficace et pour moins d'exemples**.

Le Boosting

- Une méthode d'apprentissage par **combinaison de classifieurs faibles**.
- Utilise **un ensemble de règles** de classification.

⇒ Créer une méthode de **Boosting interactive**.



Plan

1 Une méthode de recherche interactive par Boosting



Présentation de la recherche interactive



Une introduction au Boosting



Boosting actif pour la recherche interactive d'objets



Expériences

2 Boosting pour la construction de fonction noyau

3 Conclusion générale et Perspectives

Un Algorithme de Boosting pour la recherche interactive

Notre algorithme

- Création des classifieurs faibles à partir des caractéristiques visuelles des images positives.
- Une première boucle réalisée par le processus interactif.
- Une deuxième boucle réalisée par un algorithme de Boosting.



Exemples non annotés

Un Algorithme de Boosting pour la recherche interactive

Notre algorithme

- Création des classifieurs faibles à partir des caractéristiques visuelles des images positives.
- Une première boucle réalisée par le processus interactif.
- Une deuxième boucle réalisée par un algorithme de Boosting.

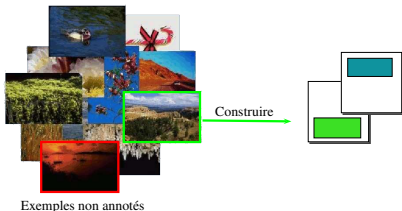


Exemples non annotés

Un Algorithme de Boosting pour la recherche interactive

Notre algorithme

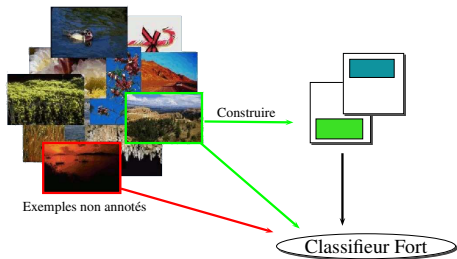
- Création des classifieurs faibles à partir des caractéristiques visuelles des images positives.
- Une première boucle réalisée par le processus interactif.
- Une deuxième boucle réalisée par un algorithme de Boosting.



Un Algorithme de Boosting pour la recherche interactive

Notre algorithme

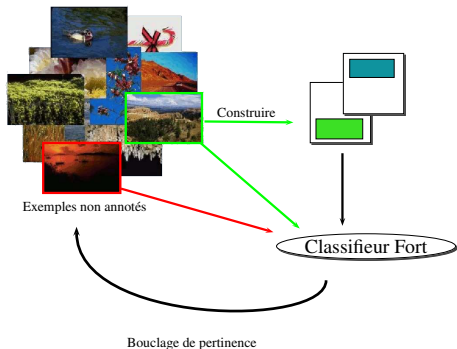
- Création des classifieurs faibles à partir des caractéristiques visuelles des images positives.
- Une première boucle réalisée par le processus interactif.
- Une deuxième boucle réalisée par un algorithme de Boosting.



Un Algorithme de Boosting pour la recherche interactive

Notre algorithme

- Création des classifieurs faibles à partir des caractéristiques visuelles des images positives.
- Une première boucle réalisée par le processus interactif.
- Une deuxième boucle réalisée par un algorithme de Boosting.



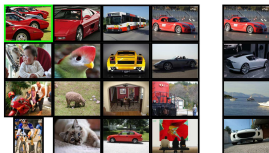
Stratégie d'apprentissage active

Stratégie d'apprentissage active

A chaque itération, la méthode active sélectionne les images x_i non annotées les plus "pertinentes" selon le critère suivant :

$$x_i^* = \arg \max_{x_i} \left(\max_{h \in \mathcal{H}_{x_i}} r_0(h) \right),$$

avec \mathcal{H}_{x_i} l'ensemble des classifieurs générés à partir de l'image x_i et $r_0(h)$ le critère de sélection des classifieurs faibles du rankboost calculés à la première boucle de cet algorithme.



Les classifieurs faibles proposés

Les classifieurs faibles

- Les classifieurs sont créés à partir des caractéristiques des images positives.
- Ils comparent un histogramme de référence à un histogramme dans l'image à tester à l'aide d'une distance d du χ^1 .

Classifieur de type 1

$$1 - d(\text{histo}_{\rho_m}(x_i), \text{histo}_{\rho_m}(x_j))$$



Exemple : la région ρ_{12356}

Les classifieurs faibles proposés

Les classifieurs faibles

- Les classifieurs sont créés à partir des caractéristiques des images positives.
- Ils comparent un histogramme de référence à un histogramme dans l'image à tester à l'aide d'une distance d du χ^1 .

Classifieur de type 2

$$1 - \min_{z_{m'} \in \text{zones}} d(\text{histo}_{z_m}(x_i), \text{histo}_{z_{m'}}(x_j))$$



Exemple : la zone z_3

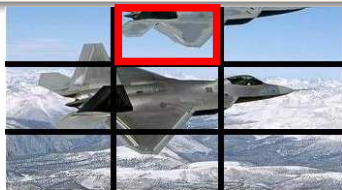
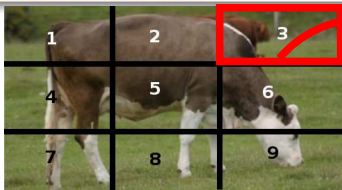
Les classifieurs faibles proposés

Les classifieurs faibles

- Les classifieurs sont créés à partir des caractéristiques des images positives.
- Ils comparent un histogramme de référence à un histogramme dans l'image à tester à l'aide d'une distance d du χ^1 .

Classifieur de type 2

$$1 - \min_{z_{m'} \in \text{zones}} d(\text{histo}_{z_m}(x_i), \text{histo}_{z_{m'}}(x_j))$$



Exemple : la zone z_3

Les classifieurs faibles proposés

Les classifieurs faibles

- Les classifieurs sont créés à partir des caractéristiques des images positives.
- Ils comparent un histogramme de référence à un histogramme dans l'image à tester à l'aide d'une distance d du χ^1 .

Classifieur de type 2

$$1 - \min_{z_{m'} \in \text{zones}} d(\text{histo}_{z_m}(x_i), \text{histo}_{z_{m'}}(x_j))$$



Exemple : la zone z_3

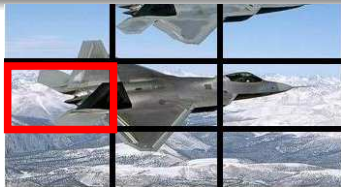
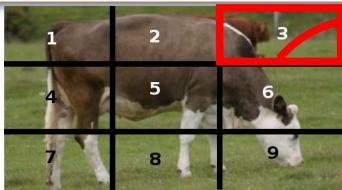
Les classifieurs faibles proposés

Les classifieurs faibles

- Les classifieurs sont créés à partir des caractéristiques des images positives.
- Ils comparent un histogramme de référence à un histogramme dans l'image à tester à l'aide d'une distance d du χ^1 .

Classifieur de type 2

$$1 - \min_{z_{m'} \in \text{zones}} d(\text{histo}_{z_m}(x_i), \text{histo}_{z_{m'}}(x_j))$$



Exemple : la zone z_3

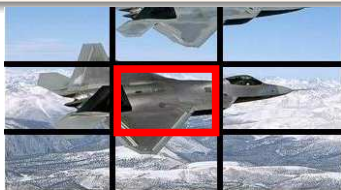
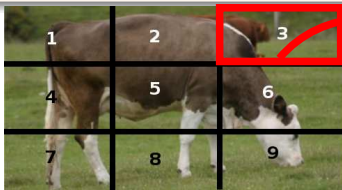
Les classifieurs faibles proposés

Les classifieurs faibles

- Les classifieurs sont créés à partir des caractéristiques des images positives.
- Ils comparent un histogramme de référence à un histogramme dans l'image à tester à l'aide d'une distance d du χ^1 .

Classifieur de type 2

$$1 - \min_{z_{m'} \in \text{zones}} d(\text{histo}_{z_m}(x_i), \text{histo}_{z_{m'}}(x_j))$$



Exemple : la zone z_3

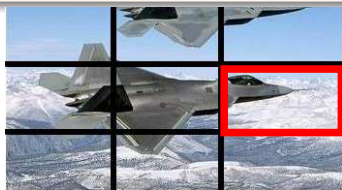
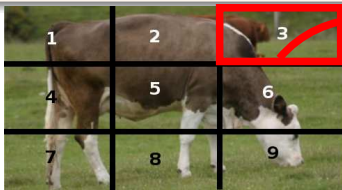
Les classifieurs faibles proposés

Les classifieurs faibles

- Les classifieurs sont créés à partir des caractéristiques des images positives.
- Ils comparent un histogramme de référence à un histogramme dans l'image à tester à l'aide d'une distance d du χ^1 .

Classifieur de type 2

$$1 - \min_{z_{m'} \in \text{zones}} d(\text{histo}_{z_m}(x_i), \text{histo}_{z_{m'}}(x_j))$$



Exemple : la zone z_3

Les classifieurs faibles proposés

Les classifieurs faibles

- Les classifieurs sont créés à partir des caractéristiques des images positives.
- Ils comparent un histogramme de référence à un histogramme dans l'image à tester à l'aide d'une distance d du χ^1 .

Classifieur de type 2

$$1 - \min_{z_{m'} \in \text{zones}} d(\text{histo}_{z_m}(x_i), \text{histo}_{z_{m'}}(x_j))$$



Exemple : la zone z_3

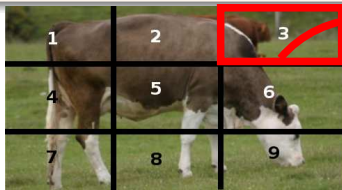
Les classifieurs faibles proposés

Les classifieurs faibles

- Les classifieurs sont créés à partir des caractéristiques des images positives.
- Ils comparent un histogramme de référence à un histogramme dans l'image à tester à l'aide d'une distance d du χ^1 .

Classifieur de type 2

$$1 - \min_{z_{m'} \in \text{zones}} d(\text{histo}_{z_m}(x_i), \text{histo}_{z_{m'}}(x_j))$$



Exemple : la zone z_3

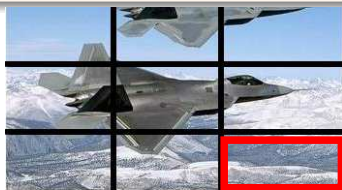
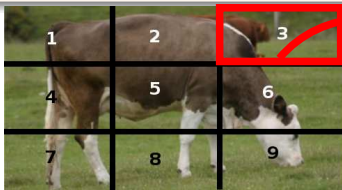
Les classifieurs faibles proposés

Les classifieurs faibles

- Les classifieurs sont créés à partir des caractéristiques des images positives.
- Ils comparent un histogramme de référence à un histogramme dans l'image à tester à l'aide d'une distance d du χ^1 .

Classifieur de type 2

$$1 - \min_{z_{m'} \in \text{zones}} d(\text{histo}_{z_m}(x_i), \text{histo}_{z_{m'}}(x_j))$$



Exemple : la zone z_3



Plan

1 Une méthode de recherche interactive par Boosting



Présentation de la recherche interactive



Une introduction au Boosting



Boosting actif pour la recherche interactive d'objets



Expériences

2 Boosting pour la construction de fonction noyau

3 Conclusion générale et Perspectives

Exemples d'image de la base VOC 2006



Bus



Chat



Cheval



Chien



Moto



Mouton



Personne



Vache



Vélo



Voiture

FIGURE: Les 10 catégories de Voc 2006 (5304 images,9507 annotations).

Expériences et résultats

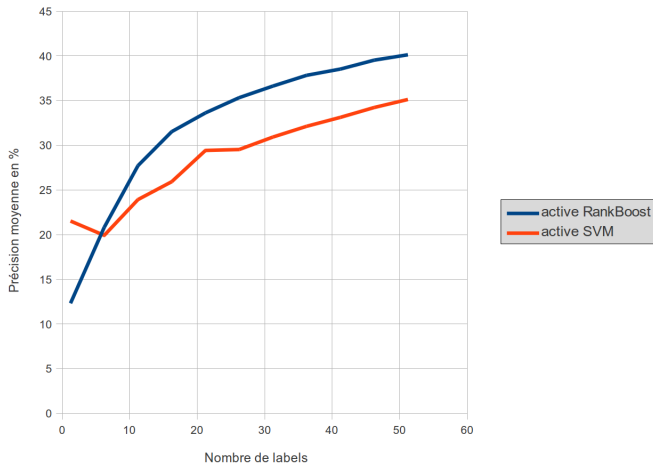


FIGURE: Précision moyenne en % sur VOC 2006. En bleu notre algorithme, en rouge un SVM avec méthode active.

Conclusions et perspectives

Conclusions

- Nous avons mis en place un **nouvel algorithme de Boosting** dans un contexte de **recherche interactif**.
- Les classifieurs faibles sont issus des images annotées positivement.
- Notre méthode nécessite **peu d'exemples** d'apprentissage.
- Les performances de classification sont meilleures que les approches concurrentes par SVM.

Perspectives

- Étudier d'autres classifieurs faibles construits à partir des images positives.
- Trouver une méthode active moins coûteuse en temps de calcul.



Plan

- 1 Une méthode de recherche interactive par Boosting
- 2 Boosting pour la construction de fonction noyau
 - ▷ La classification par hyperplan
 - ▷ Une nouvelle méthode de Boosting
 - ▷ Calcul de la pondération optimale
 - ▷ Calcul du noyau cible
 - ▷ Expériences
- 3 Conclusion générale et Perspectives



Plan

1 Une méthode de recherche interactive par Boosting

2 Boosting pour la construction de fonction noyau

▷ La classification par hyperplan



Une nouvelle méthode de Boosting



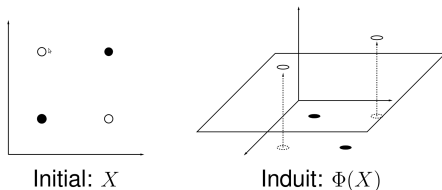
Calcul de la pondération optimale

▷ Calcul du noyau cible

▷ Expériences

3 Conclusion générale et Perspectives

La Classification par hyperplan



Objectif

Séparer les exemples d'apprentissage en deux ensembles à l'aide d'un hyperplan.

Problème

Certains espaces ne sont pas linéairement séparables.

Solution

Changer l'espace de représentation en utilisant une fonction noyau.

Les fonctions noyaux

Changement d'espace et mesure de similarité

- Un changement d'espace $\phi : \begin{cases} \mathcal{X} & \longrightarrow \mathcal{H} \\ x & \longmapsto \phi(x) \end{cases}$
- Une fonction noyau est une mesure de similarité dans un nouvel espace : $k : \begin{cases} \mathcal{X} \times \mathcal{X} & \longrightarrow \mathbb{R} \\ x_1, x_2 & \longmapsto k(x_1, x_2) = \langle \phi(x_1), \phi(x_2) \rangle \end{cases}$
- Astuce du noyau : Un algorithme utilisant un produit scalaire peut être "noyauté" en remplaçant tous ces produits scalaires par une fonction noyau.

Comment choisir/construire la fonction noyau ?

Utiliser une forme connue

- Noyau linéaire : $k(x_i, x_j) = \langle x_i, x_j \rangle$.
- Noyaux polynomiaux : $k(x_i, x_j) = (\langle x_i, x_j \rangle + c)^q$, avec $c \in \mathbb{R}$ et $q \in \mathbb{R}^+$.
- Noyaux triangulaires : $k(x_i, x_j) = 1 - \frac{\|x_i - x_j\|^2}{\sigma^2}$, $\sigma \in \mathbb{R}^+$.
- Noyaux Gaussiens : $k(x_i, x_j) = e^{-\frac{\|x_i - x_j\|^2}{2\sigma^2}}$, $\sigma \in \mathbb{R}^+$.

Combiner un ensemble de fonctions noyaux pour en construire une nouvelle

- Les approches de type Multiple Kernel Learning (MKL).
- Les approches par Boosting.

Problématique des approches par MKL

Objectifs

- A partir d'un ensemble de fonctions noyaux simples $\{k_1, \dots, k_p\}$ construire une nouvelle fonction noyau k par combinaison linéaire :

$$k(x_1, x_2) = \sum_i \beta_i k_i(x_1, x_2), \forall i \quad \beta_i > 0. \quad (1)$$

- Calculer l'hyperplan permettant de séparer les données d'apprentissage :

$$f(x) = \sum_j \alpha_j y_j k(x, x_j), \forall i \quad \alpha_j > 0. \quad (2)$$

Remarque

- La combinaison linéaire de noyaux \iff concaténation de sous-espaces de descripteurs.



Multiple Kernel Learning (MKL) [Bach et al., 2004]

Une approche populaire de combinaison de noyaux

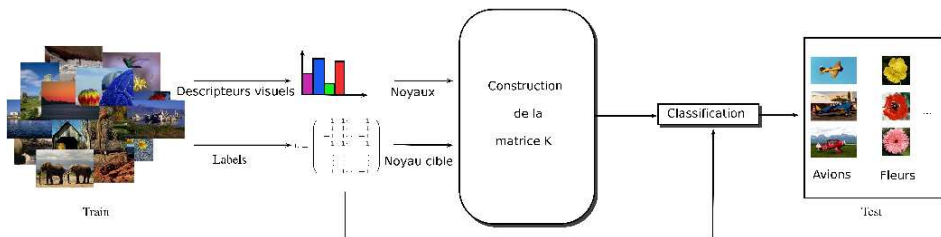
- **Optimisation jointe** de la meilleure combinaison de noyaux et de l'hyperplan d'un SVM.
- Plusieurs solutions ont été proposées pour résoudre ce problème d'optimisation.

Inconvénients

- Non adaptée à un nombre important de noyaux et d'images.
- L'ensemble des fonctions noyaux possibles est pré-établi.

Une autre approche en deux étapes

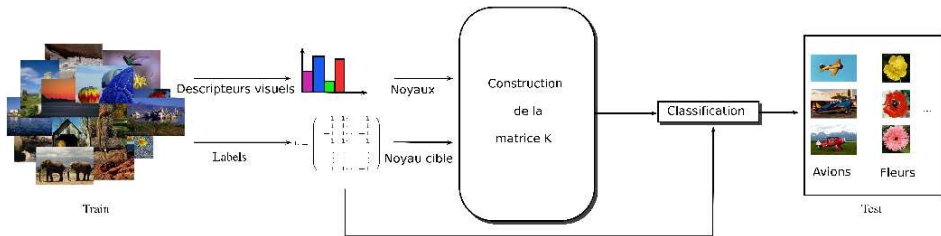
[Cortes et al., 2010]



Principe

- **Optimisation en deux temps.** On maximise l'alignement avec un noyau cible puis on calcule l'hyperplan optimal dans le nouvel espace.
- Réduit le sur-apprentissage de l'espace de représentation.

Une autre approche en deux étapes [Cortes et al., 2010]



Inconvénients

- Complexité sur-linéaire.
- L'ensemble des fonctions noyaux possibles est pré-établi.

Combinaison de noyaux et Boosting

L'approche de [Crammer et al. 2002]

- Les exemples d'apprentissage sont remplacés par des couples d'images $X_{ij} = (x_i, x_j)$.
- Les classifieurs faibles correspondent aux fonctions noyaux $k_t(X_{ij}) = k_t(x_i, x_j)$.
- Utilisation d'un algorithme de Boosting classique avec ces fonctions et exemples.

Avantages

- Les noyaux ne sont pas pré-établis.

Inconvénients

- Complexité quadratique.

Combinaison de noyaux et Boosting

L'approche de [Gehler et al. 2009]

- un SVM est appliqué à l'ensemble des fonctions noyaux disponibles
- on utilise ensuite un algorithme de Boosting sur les fonctions ainsi formées.
- les deux étapes de l'approche de [Cortes et al. 2010] sont inversées

Avantages

- Temps de calcul linéaire.
- Cette approche montre que le Boosting et les méthodes de MKL en deux temps sont des approches similaires où le calcul de l'hyperplan est réalisé dans un nouvel espace issu de la concaténation de plusieurs sous-espaces.

Combinaison de noyaux et Boosting

L'approche de [Gehler et al. 2009]

- un SVM est appliqué à l'ensemble des fonctions noyaux disponibles
- on utilise ensuite un algorithme de Boosting sur les fonctions ainsi formées.
- les deux étapes de l'approche de [Cortes et al. 2010] sont inversées

Inconvénients

- L'ensemble des fonctions noyaux possibles est pré-établi.

Bilan des méthodes de combinaisons de noyaux

Bilan

- Plusieurs approches de résolution sont proposées dans la littérature.
- La résolution des problèmes d'optimisation est réalisée soit conjointement, soit en deux temps.
- Le coût calculatoire est généralement important.
- Les fonctions noyaux sélectionnables sont généralement pré-établies.

⇒ Comment construire une nouvelle fonction noyau avec un coût calculatoire linéaire en utilisant des fonctions noyaux apprises à la volée ?

Plan

1 Une méthode de recherche interactive par Boosting

2 Boosting pour la construction de fonction noyau

▷ La classification par hyperplan



Une nouvelle méthode de Boosting



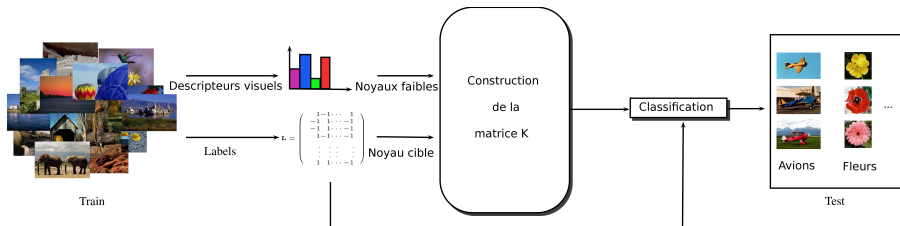
Calcul de la pondération optimale

▷ Calcul du noyau cible

▷ Expériences

3 Conclusion générale et Perspectives

Notre approche de Boosting



Notre approche interactive de construction de noyau fort

- 1 Un ensemble d'exemples $(x_1, y_1), \dots, (x_N, y_N)$ avec les labels $y_i \in \{-1, 1\}^c$
- 2 Pour $t = 1, \dots, T$ faire
 - **Construire** et **sélectionner** un noyau faible
 - **Choisir un poids** pour ce noyau
 - Ajouter ce noyau à la combinaison de noyau
- 3 Le classifieur fort final : une somme pondérée des noyaux sélectionnés

Objectifs de notre approche.

Objectifs de notre méthode

- Utiliser une approche itérative pour déterminer une combinaison de noyaux.
- Définir les fonctions noyaux au cours des itérations.
- Réaliser chaque itération en un temps linéaire en nombre d'images.

Objectifs d'une itération t

- Construire un **noyau fort itérativement**.

$$\mathbf{K}_t = \mathbf{K}_{t-1} + \beta_t \mathbf{k}_t. \quad (3)$$

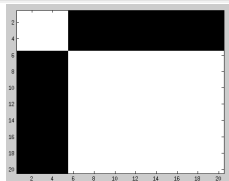
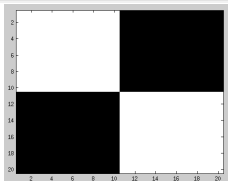
- Construction à l'aide de la résolution du problème d'optimisation $\{\beta_t, \mathbf{k}_t\} = \arg \max_{\beta > 0, k} \mathcal{A}_{\mathbf{H}}(\mathbf{K}_t + \beta \mathbf{k}, \mathbf{K}^*)$. (4)

Critère de sélection des fonctions noyaux

L'alignement centré entre les noyaux \mathbf{K} et \mathbf{K}^*

$$\mathcal{A}_H(\mathbf{K}, \mathbf{K}^*) = \frac{\langle \bar{\mathbf{K}}, \bar{\mathbf{K}}^* \rangle}{\|\bar{\mathbf{K}}\| \|\bar{\mathbf{K}}^*\|},$$

avec $\bar{\mathbf{K}} = \mathbf{H}\mathbf{K}\mathbf{H}$ et la matrice de centrage $\mathbf{H} = \mathbf{Id} - \frac{1}{n}\mathbf{1}\mathbf{1}^\top$,
 $\langle \cdot \rangle$ et $\|\cdot\|$ sont les produits scalaires et normes de Frobenius.



L'alignement centré entre ces deux matrices représentant une classification binaire est de 0.3333.

Le noyau cible K^*

Une première solution pour la construction d'un noyau cible

- La matrice LL^T ,
- Avec la matrice L des labels dans un cas multi-classes :

$$L = \left(\begin{array}{c|cccccc} & C_1 & C_2 & C_3 & \cdots & C_{nbClasses} \\ \hline x_1 & 1 & -1 & -1 & \cdots & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n & 1 & 1 & -1 & \cdots & -1 \end{array} \right)$$

Avantage de L

- La matrice L est bien issue des annotations de l'utilisateur.

Inconvénient de L

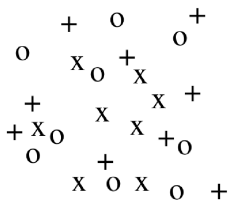
- La matrice L ne possède pas de bonnes propriétés pour la classification par hyperplan.

Le noyau cible K^*

La matrice cible idéale [Vapnik 1982]

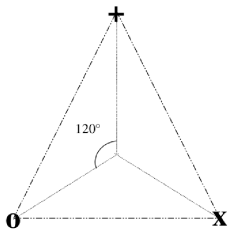
L'éclatement optimal d'un ensemble de points en c catégories avec une marge optimale est atteint lorsque **les points se répartissent autour des c sommets d'un simplexe unitaire de dimensions $c - 1$ centré sur l'origine.**

Espace initial d'entrée



$k_t(\Phi_t)$ →

Espace du descripteur cible



Le noyau cible K^*

Une première solution pour la construction d'un noyau cible

- La matrice LL^T ,
- Avec la matrice L des labels dans un cas multi-classes :

$$L = \left(\begin{array}{c|cccc} & c_1 & c_2 & c_3 & \cdots & c_{nbClasses} \\ \hline x_1 & 1 & -1 & -1 & \cdots & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n & 1 & 1 & -1 & \cdots & -1 \end{array} \right)$$

Propriétés attendues par la matrice cible

- les points sont les sommets d'un simplexe unitaire d'ordre $(c - 1)$
- le simplexe ainsi formé est centré à l'origine.

La matrice cible \mathbf{K}^*

Notre solution

- Centrage : \mathbf{HL} avec $\mathbf{H} = \text{Id} - \frac{1}{n}\mathbf{1}\mathbf{1}^\top$.
- Orthonormalisation de Gram Schmidt :
 $\{\mathbf{Q}, \mathbf{R}\} = \text{décomposition qr}(\mathbf{HL})$.
- Sélectionner les colonnes de \mathbf{Q} qui correspondent aux éléments diagonaux non nuls de la matrice \mathbf{R} (QR économique).
- Noyau Cible : $\mathbf{K}^* = \mathbf{Q}\mathbf{Q}^\top$.

Propriétés de la matrice \mathbf{Q}

- Matrice orthogonale $\mathbf{Q}^\top \mathbf{Q} = \text{Id}_c$.
- Matrice centrée $\mathbf{H}\mathbf{Q} = \mathbf{Q}$.
- $\|\mathbf{Q}\mathbf{Q}^\top\| = \|\mathbf{Q}^\top \mathbf{Q}\| = \sqrt{c}$.

Objectifs d'une itération t

Objectifs

- Construire un **noyau fort itérativement**.

$$\mathbf{K}_t = \mathbf{K}_{t-1} + \beta_t \mathbf{k}_t. \quad (5)$$

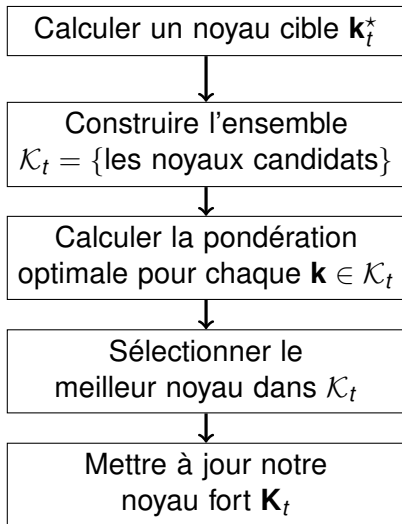
- Construction à l'aide de la résolution du problème d'optimisation suivant :

$$\{\beta_t, \mathbf{k}_t\} = \arg \max_{\beta > 0, k} \mathcal{A}_H(\mathbf{K}_t + \beta \mathbf{k}, \mathbf{K}^*). \quad (6)$$

En conséquence

⇒ A chaque itération, on doit trouver une fonction k_t et sa pondération β_t qui augmente l'alignement centré.

Notre approche de Boosting : une itération





Plan

1 Une méthode de recherche interactive par Boosting

2 Boosting pour la construction de fonction noyau

▷ La classification par hyperplan



Une nouvelle méthode de Boosting



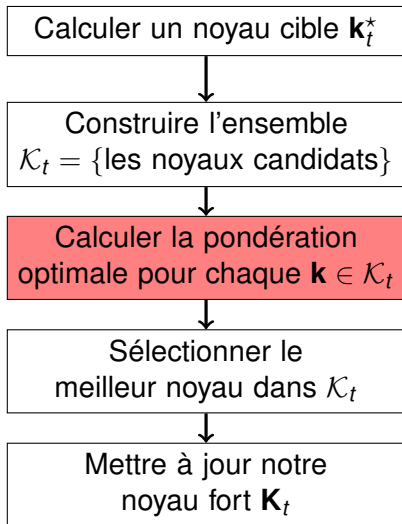
Calcul de la pondération optimale

▷ Calcul du noyau cible

▷ Expériences

3 Conclusion générale et Perspectives

Une itération de notre algorithme



Calcul de la pondération optimale β^*

Le problème d'optimisation

$$\beta^*(\mathbf{k}) = \arg \max_{\beta > 0} \mathcal{A}_H(\mathbf{K}_{t-1} + \beta \mathbf{k}, \mathbf{K}^*).$$

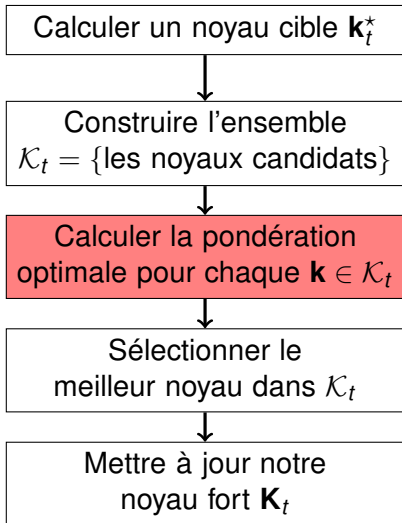
Une solution analytique

$$\beta^* = \frac{\|\overline{\mathbf{K}_{t-1}}\|}{\|\overline{\mathbf{k}}\|} \frac{\alpha - \mathcal{A}_H(\mathbf{K}, \mathbf{k})}{1 - \alpha \mathcal{A}_H(\mathbf{K}, \mathbf{k})} \text{ with } \alpha = \frac{\mathcal{A}_H(\mathbf{k}, \mathbf{K}^*)}{\mathcal{A}_H(\mathbf{K}_{t-1}, \mathbf{K}^*)}.$$

En outre, si $\beta^* \neq 0$ l'alignement est augmenté :

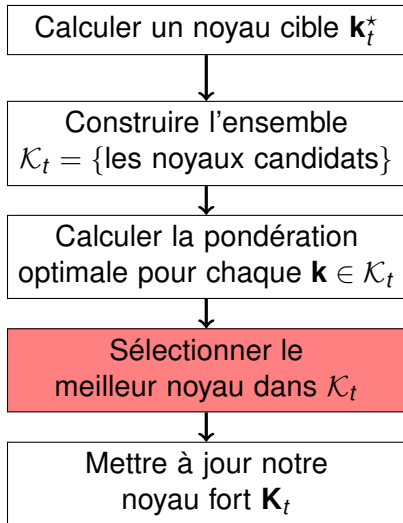
$$\mathcal{A}_H(\mathbf{K}_{t-1} + \beta^* \mathbf{k}, \mathbf{K}^*) > \mathcal{A}_H(\mathbf{K}_{t-1}, \mathbf{K}^*).$$

Une itération de notre algorithme



Une solution analytique

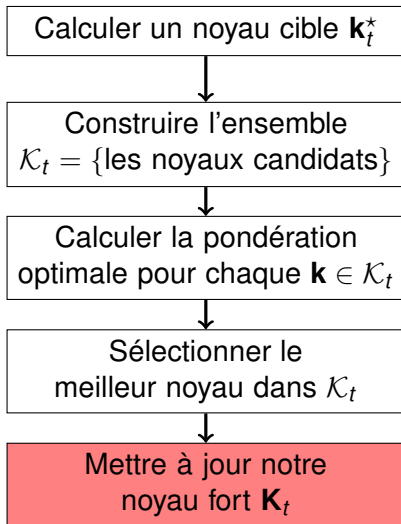
Une itération de notre algorithme



Maximiser l'alignement

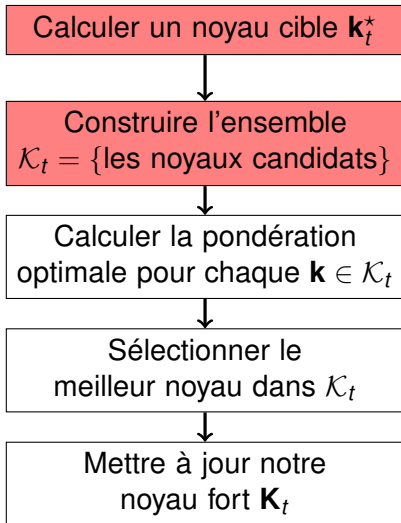
$$\{\beta_t, \mathbf{k}_t\} = \arg \max_{\beta > 0, \mathbf{k} \in \mathcal{K}_t} \mathcal{A}_H(\mathbf{K}_t + \beta \mathbf{k}, \mathbf{K}^*).$$

Une itération de notre algorithme



$$\mathbf{K}_{t+1} \leftarrow \mathbf{K}_t + \beta_t \mathbf{k}_t.$$

Une itération de notre algorithme





Plan

1 Une méthode de recherche interactive par Boosting

2 Boosting pour la construction de fonction noyau

▷ La classification par hyperplan



Une nouvelle méthode de Boosting



Calcul de la pondération optimale

▷ Calcul du noyau cible

▷ Expériences

3 Conclusion générale et Perspectives

Deux points importants

Notre méthode est itérative

Nous construisons un **noyau fort itérativement**. Pour chaque étape, nous **ajoutons un noyau de rang 1** à notre noyau fort.

$$\mathbf{K}_t = \mathbf{K}_{t-1} + \beta_t \mathbf{k}_t$$

Dualité noyau/descripteur sémantique

Ajouter un noyau faible de rang 1 est **équivalent** à **ajouter** une nouvelle dimension à une matrice de **descripteur sémantique** :

$$\mathbf{K}_t = \mathbf{K}_{t-1} + \beta_t \mathbf{k}_t \iff \mathbf{F}_t = (\mathbf{F}_{t-1} \beta_t^{\frac{1}{2}} \mathbf{f}_t).$$

Nous avons : $\mathbf{K}_t = \mathbf{F}_t \mathbf{F}_t^\top$.

Retour sur la formulation dans le primal

Le descripteur sémantique

- Formulation :

$$\mathbf{K}_t = \mathbf{F}_t \mathbf{F}_t^\top.$$

- Interprétation :

$$\mathbf{F}_t = \left(\begin{array}{c|ccccc} & f_1 & f_2 & f_3 & \cdots & f_t \\ \hline x_1 & 0.1 & -0.5 & -0.3 & \cdots & 0.9 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n & 0.2 & 0.4 & -0.1 & \cdots & -0.4 \end{array} \right)$$

L'image x_1 est décrite par le descripteur sémantique
 $[0.1 \quad -0.5 \quad -0.3 \quad \cdots \quad 0.9]$.

Retour sur la formulation dans le primal

L'espace cible

- Formulation :

$$\mathbf{K}^* = \mathbf{Q}\mathbf{Q}^\top.$$

- Interprétation :

$$\mathbf{Q} = \left(\begin{array}{c|cccc} & q_1 & q_2 & q_3 & \cdots & q_t \\ \hline x_1 & 1 & -1 & -1 & \cdots & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n & 1 & 1 & -1 & \cdots & -1 \end{array} \right)$$

L'image x_1 appartient aux classes $[1 \ -1 \ -1 \ \cdots \ 1]$.



La méthode que nous proposons

Le problème d'optimisation pour chaque étape du Boosting écrit avec la fonction noyau

$$\{\beta_t, k_t\} = \arg \max_{\beta > 0, k} \mathcal{A}_H(\mathbf{K}_t + \beta \mathbf{k}, \mathbf{K}^*).$$

La méthode que nous proposons

Le problème d'optimisation pour chaque étape du Boosting écrit avec la fonction noyau

$$\{\beta_t, k_t\} = \arg \max_{\beta > 0, k} \mathcal{A}_H(\mathbf{K}_t + \beta \mathbf{k}, \mathbf{K}^*).$$

Le problème d'optimisation pour chaque étape du Boosting écrit dans l'espace sémantique

$$\{\beta_t, f_t\} = \arg \max_{\beta > 0, f} \mathcal{A}_H(\mathbf{F}_t \mathbf{F}_t^\top + \beta \mathbf{f} \mathbf{f}^\top, \mathbf{K}^*).$$

La méthode que nous proposons

Le problème d'optimisation pour chaque étape du Boosting écrit avec la fonction noyau

$$\{\beta_t, k_t\} = \arg \max_{\beta > 0, k} \mathcal{A}_H(\mathbf{K}_t + \beta \mathbf{k}, \mathbf{K}^*).$$

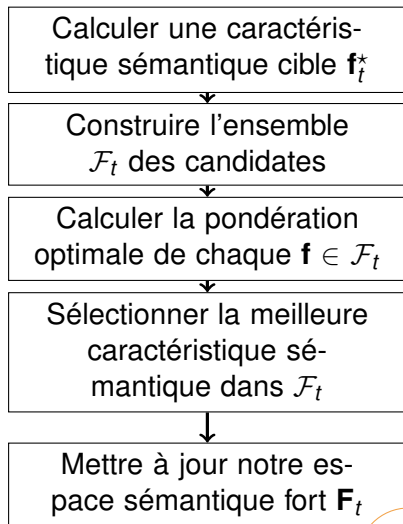
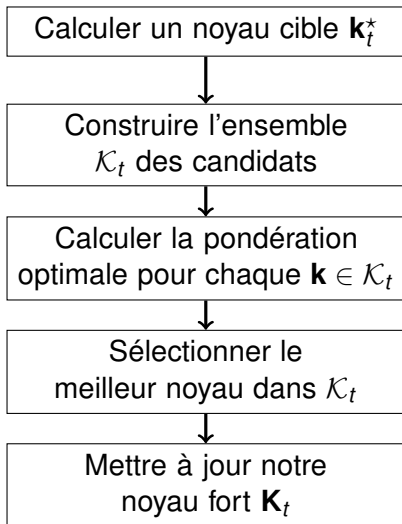
Le problème d'optimisation pour chaque étape du Boosting écrit dans l'espace sémantique

$$\{\beta_t, f_t\} = \arg \max_{\beta > 0, f} \mathcal{A}_H(\mathbf{F}_t \mathbf{F}_t^\top + \beta \mathbf{f} \mathbf{f}^\top, \mathbf{K}^*).$$

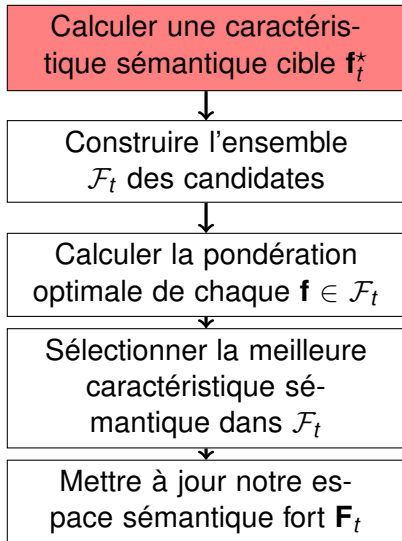
En conséquence

⇒ Trouver une caractéristique sémantique f et sa pondération β qui augmente l'alignement centré.

Notre méthode



Une itération de notre algorithme



Une approche avec barycentres

Les barycentres

Les barycentres des classes dans l'espace induit par la combinaison courante de noyau $\mathbf{K}_t = \mathbf{F}_t \mathbf{F}_t^\top$ correspond à :

$$\mathbf{G}_t = \mathbf{Q}^\top \mathbf{F}_t.$$

Explications et Idées

- Chaque ligne de la matrice \mathbf{G}_t correspond au barycentre d'une classe dans l'espace induit par \mathbf{F}_t .
- Pour que les centres de classes soient équidistants, il suffit que la matrice \mathbf{G}_t^* soit telle que $\mathbf{G}_t^* (\mathbf{G}_t^*)^\top$ est proportionnelle à la matrice identité Id_c .

Calcul de la caractéristique sémantique cible \mathbf{f}_t^*

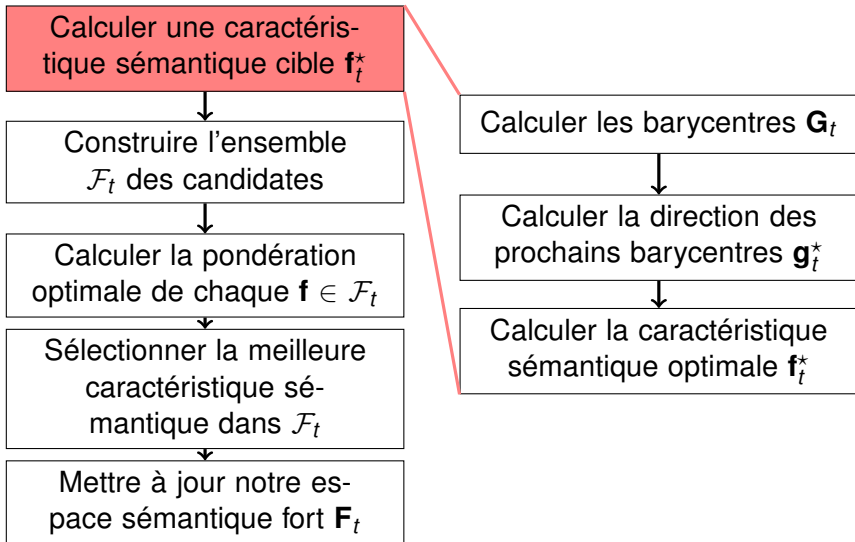
Calcul de la caractéristique sémantique cible \mathbf{f}^*

- On choisit la caractéristique sémantique \mathbf{f}_t^* telle que la position des barycentres des classes projetés dans l'espace engendré par la fonction se rapprochent le plus d'un barycentre cible \mathbf{g}_t^* .
- Solution du problème de régression : $\mathbf{f}_t^* = \arg \min_f \|\mathbf{Q}^\top \mathbf{f} - \mathbf{g}_t^*\|^2$,
- La solution retenue : $\mathbf{f}_t^* = \mathbf{Q}\mathbf{g}_t^*$.

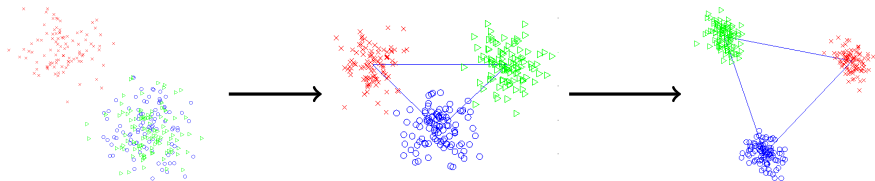
Choix du barycentre cible \mathbf{g}_t^*

- On fait converger les barycentres des classes vers une position optimale itérativement : $\mathbf{G}_t = [\mathbf{G}_{t-1}, \mathbf{g}_t]$.
- La direction idéale d'évolution des barycentres correspond à \mathbf{g}_t^* .

Une itération de notre algorithme



Évolution des barycentres



Positions optimales des barycentres

- La meilleure position des barycentres est suivant un simplexe.
- Nous voulons donc que $\mathbf{G}_t \mathbf{G}_t^T \rightarrow a \text{Id}_c$ à un facteur de normalisation a près.

Approche de résolution

- Annuler les plus grandes valeurs propres de $\mathbf{G}_t \mathbf{G}_t^T - a \text{Id}_c$ les unes après les autres.

Direction d'évolution des barycentres \mathbf{g}_t^*

Calcul de \mathbf{g}_t^*

Soit

$$\varphi(\mathbf{G}_t) = \frac{\|\bar{\mathbf{K}}^*\|}{\|\bar{\mathbf{K}}_t\|} \arg \min_{\lambda} \{\lambda \in \text{spec}(\mathbf{G}_t \mathbf{G}_t^\top)\},$$

La direction d'évolution des barycentres correspond au vecteur

$$\mathbf{g}_t^* = \sqrt{1 - \varphi(\mathbf{G}_{t-1})} \mathbf{v},$$

avec \mathbf{v} la valeur propre associée au vecteur propre $\varphi(\mathbf{G}_t)$.

Justification théorique des choix proposés

Théorème 1 : Lien entre $\varphi(\mathbf{G}_t)$ et l'alignement

L'alignement peut être minoré par

$$0 \leq \varphi(\mathbf{G}_t) \leq \mathcal{A}(\mathbf{K}_t, \mathbf{K}^*),$$

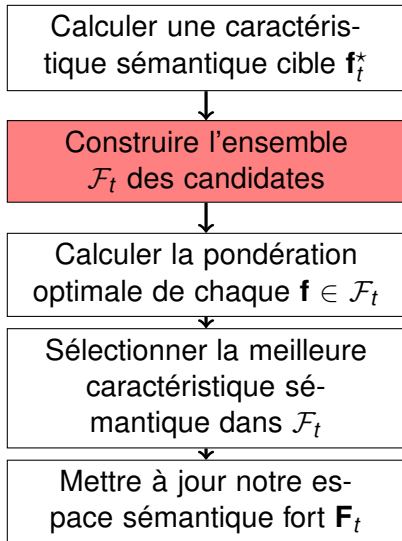
$$\text{où } \varphi(\mathbf{G}_t) = \frac{\|\bar{\mathbf{K}}^*\|}{\|\mathbf{K}_t\|} \arg \min_{\lambda} \{\lambda \in \text{spec}(\mathbf{G}_t \mathbf{G}_t^T)\}.$$

Théorème 2 : $\mathbf{f}^* = \mathbf{Qg}^*$ est un bon candidat

Pour $\mathbf{f}^* = \mathbf{Qg}^*$, si $\mathcal{A}_H(\mathbf{K}_{t-1}, \mathbf{K}^*) < 1$ alors on a

- $\beta_t^* > 0$
- $\mathcal{A}_H(\mathbf{K}_{t-1} + \beta_t^* \mathbf{f}_t^* \mathbf{f}_t^{*\top}, \mathbf{K}^*) > \mathcal{A}_H(\mathbf{K}_{t-1}, \mathbf{K}^*)$

Une itération de notre algorithme



Construction des fonctions f

Construction des f_s

On calcule les apprenants faibles sur l'ensemble d'apprentissage $\mathbf{X}_s \subset \mathcal{X}_s$ avec une méthode de régression par moindres carrés (Least Mean Square [LMS]) :

$$f_s = \arg \min_w \|\mathbf{X}_s \mathbf{w} - \mathbf{f}^*\|^2 \quad \text{s.t.} \quad \|\mathbf{w}\|^2 = 1$$



Plan

1 Une méthode de recherche interactive par Boosting

2 Boosting pour la construction de fonction noyau

▷ La classification par hyperplan



Une nouvelle méthode de Boosting



Calcul de la pondération optimale

▷ Calcul du noyau cible

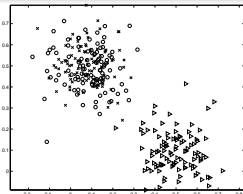
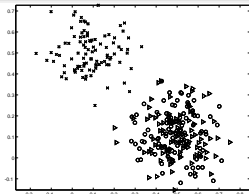
▷ Expériences

3 Conclusion générale et Perspectives

Expériences avec des données de synthèse

Données de synthèse

- **2 classes, 200 exemples** par classe (100 pour l'entraînement et 100 pour les test).
- **10 descripteurs de 2 dimensions**
- Pour chaque classe c et chaque descripteur f , **un centre** $C_{c, f}$ est tiré aléatoirement suivant une loi **uniforme** entre $[0, 1]^2$.
- **Chaque exemple** est décrit par **10 descripteurs** selon une loi $\mathcal{N}(C_{c, f}, 0.5)$.
- Pour la **visualisation**, nous utilisons une PCA sur F_t et nous sélectionnons uniquement les **deux premières dimensions**.



Expériences sur des données de synthèse

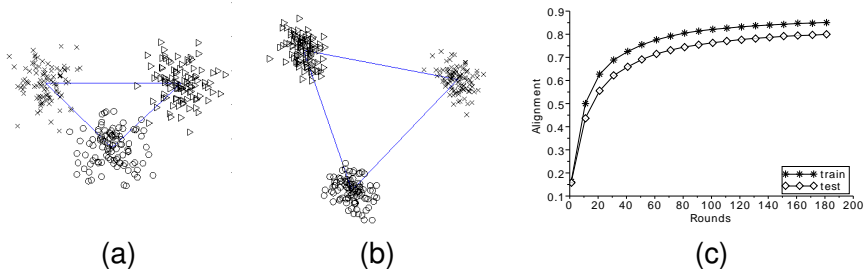


FIGURE: Convergence de la méthode vers un 2-simplex régulier pour 3 classes après 2 étapes(a) et 45 étapes(b). Évolution de l'alignement pour 10 classes(c)

k	1	2	3	4	5	6	7	8	9	10
Kawanabe	20	24	16	13	8	6	5	5	4	5
Notre méthode	14	18	11	9	8	7	7	6	5	5

FIGURE: % d'erreur pour un algorithme k-Plus-Proche-Voisin

KernBoost vs la concaténation des descripteurs

Descripteurs	1	2	3	4	5	6	7	8	9	10	Moyenne
Lab16	12.4	9.4	24.6	16.0	9.4	11.2	9.0	8.0	27.2	10.9	14.0
Lab32	10.7	8.1	45.7	27.1	34.2	15.9	10.0	9.1	27.0	25.6	22.8
Lab64	10.0	12.5	47.9	28.5	37.5	19.1	9.9	16.4	31.7	50.3	26.3
Lab128	18.7	24.7	46.6	28.8	34.1	20.0	16.0	16.5	33.2	50.0	28.7
Qw16	14.0	46.8	55.5	15.1	7.5	14.6	8.3	21.0	25.6	11.2	22.0
Qw32	38.5	52.2	60.2	22.2	7.7	15.9	8.9	36.0	36.9	25.6	30.4
Qw64	43.0	53.1	63.4	22.0	14.2	18.8	13.2	43.3	37.5	36.2	34.4
Qw128	47.9	57.4	65.6	25.8	14.8	20.3	21.6	45.5	33.2	48.6	37.6
Concaténation	52.1	58.2	72.2	37.4	38.5	27.1	26.7	44.4	39.5	56.1	45.3
Proposée	52.2	63.1	75.9	43.8	41.6	27.6	27.2	52.7	41.9	56.6	48.3

FIGURE: Précision moyenne (%) sur la base VOC2006, la catégorisation des images est réalisée en utilisant des SVMs linéaires et différents descripteurs.

Exemples des catégories de Flowers 102



Iris léopard



Eustoma de Russel



Bougainvillée



Bromelia



Camélia



Ancolie



Dahlia Orange



Gazania

FIGURE: Exemples de catégories de fleurs de la base Flowers 102

Oxford Flowers 17 et Oxford Flowers 102

Oxford Flowers 17

- 17 catégories, 1360 images (80 images de fleurs par catégorie)
- 3 découpages aléatoires prédéfinis
- chaque découpage est constitué de 3 ensembles :
 - d'entraînement (40 images par classe)
 - de validation/test (20 images par classe)
- 7 matrices de distances du χ^2

Oxford Flowers 102

- 102 catégories, 8189 images
- de 40 à 250 images de fleurs par catégorie
- entraînement/validation (10 images par classe)
- test (20 images par classe)
- 4 matrices de distances χ^2



Comparaison de notre approche sur Flowers 17

	16	32	64	128	256	512	1024
$T \leq 50$	78.24	-	-	-	-	-	-



Comparaison de notre approche sur Flowers 17

	16	32	64	128	256	512	1024
$T \leq 50$	78.24	–	–	–	–	–	–
$T \leq 100$	83.73	–	–	–	–	–	–

Comparaison de notre approche sur Flowers 17

	16	32	64	128	256	512	1024
$T \leq 50$	78.24	–	–	–	–	–	–
$T \leq 100$	83.73	–	–	–	–	–	–
$T \leq 200$	84.12	84.80	85.78	85.88	–	–	–

Comparaison de notre approche sur Flowers 17

	16	32	64	128	256	512	1024
$T \leq 50$	78.24	—	—	—	—	—	—
$T \leq 100$	83.73	—	—	—	—	—	—
$T \leq 200$	84.12	84.80	85.78	85.88	—	—	—
$T \leq 300$.	84.80	86.27	85.88	86.47	—	—

Comparaison de notre approche sur Flowers 17

	16	32	64	128	256	512	1024
$T \leq 50$	78.24	—	—	—	—	—	—
$T \leq 100$	83.73	—	—	—	—	—	—
$T \leq 200$	84.12	84.80	85.78	85.88	—	—	—
$T \leq 300$.	84.80	86.27	85.88	86.47	—	—
$T \leq 400$.	.	86.67	86.27	87.55	—	—

Comparaison de notre approche sur Flowers 17

	16	32	64	128	256	512	1024
$T \leq 50$	78.24	—	—	—	—	—	—
$T \leq 100$	83.73	—	—	—	—	—	—
$T \leq 200$	84.12	84.80	85.78	85.88	—	—	—
$T \leq 300$.	84.80	86.27	85.88	86.47	—	—
$T \leq 400$.	.	86.67	86.27	87.55	—	—
$T \leq 500$.	.	.	86.47	87.94	88.33	88.33

FIGURE: Taux de classification sur Flowers 17 pour différents nombres d'itérations max et différentes dimensions max des descripteurs visuels

Comparaison avec la littérature sur Flowers 17

Méthode	Résultats
MKL	87.2 \pm 2.7
NLP- β	87.9 \pm 1.8
NLP- ν MC	87.8 \pm 2.1
NLP-B	87.3 \pm 2.7
MKL-prod	85.5 \pm 1.2
MKL-avg (l_∞)	84.9 \pm 1.9
CF (l_∞) / AKM	86.7 \pm 2.7
CG-Boost	84.8 \pm 2.2
MKL (SILP or Simple)	85.2 \pm 1.5
LP- β	85.5 \pm 3.0
LP-B	85.4 \pm 2.4
MKL-FDA (l_p)	86.7 \pm 1.2
Méthode proposée	88.3 \pm 1.1

Comparaison avec la littérature sur Flowers 102

Méthode	Résultats
MKL (Nilsback & Zisserman, 2008)	72.8
MKL (Awais et al., 2011)	73.4
NLP- β	75.7
NLP- ν MC	73.4
NLP-B	73.6
MKL-prod	73.8
MKL-avg	73.4
Méthode proposées	77.8

FIGURE: Taux de classification sur Flowers 102. Mis à part pour notre solution, les résultats sont extraits de [Awais et al., BMVC 2011].

Conclusions

- Nous avons développé une **nouvelle méthode de Boosting** pour la **création de fonction noyau**.
- Notre approche ne nécessite pas le calcul explicite de la fonction noyau.
- La solution proposée est **linéaire** en fonction du nombre d'exemple d'apprentissage.
- La combinaison est réalisée à l'aide de fonctions noyaux qui ne sont pas pré-établies.



Plan

- 1 Une méthode de recherche interactive par Boosting
- 2 Boosting pour la construction de fonction noyau
- 3 Conclusion générale et Perspectives**

Conclusion générale

- Nous avons développé une **nouvelle méthode de Boosting interactive**.
- Nous avons développé une **nouvelle méthode de construction de noyau par Boosting**.
- Cette méthode se déroule en deux étapes : la construction d'**un nouvel espace de description** des données puis l'utilisation d'un classifieur SVM
- La solution proposée est **linéaire** en fonction du nombre d'exemples d'apprentissage.

Perspectives

A court terme

- Tester nos approches sur des bases d'images plus grandes.
- Étendre notre approche à la vidéo.
- Utiliser d'autres classifieurs faibles ou d'autres critères de comparaison des noyaux.

A moyen terme

- Fusionner les deux approches proposées dans cette thèse pour obtenir une méthode interactive de construction d'une fonction noyau.
- Descendre au niveau des descripteurs et apprendre également les dictionnaires visuels au cours des itérations de l'algorithme de Boosting.

Perspectives

A moyen terme (suite)

- Utiliser d'autres matrices cibles notamment des matrices issues d'un apprentissage collaboratif.

$$L = \left(\begin{array}{c|cccc} & s_1 & s_2 & s_3 & \cdots & s_{nbSession} \\ \hline x_1 & 1 & -1 & -1 & \cdots & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n & 0 & 1 & 0 & \cdots & -1 \end{array} \right)$$

- Réduire la complexité en passant à des solutions sous-linéaire en utilisant des approximations de l'alignement.

A long terme

- Apprendre à mieux décrire les images pour favoriser la classification et réduire le fossé entre sémantique et numérique.

Publications

Publiés

- Boosting actif pour la recherche interactive d'images, A.Lechervy, P-H. Gosselin et F.Precioso, RFIA 2010
- Active Boosting for interactive object retrieval, A.Lechervy, P-H. Gosselin and F.Precioso, ICPR 2010
- Linear kernel combination using boosting, A.Lechervy, P-H. Gosselin and F.Precioso, ESANN 2012
- Boosting kernel combination for multi-class image categorization, A.Lechervy, P-H. Gosselin and F.Precioso, ICIP 2012

Accepté

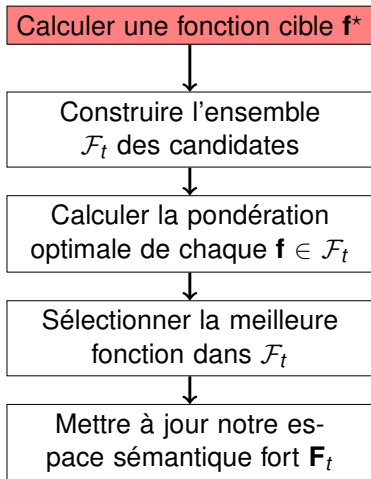
- Boosted Kernel for Image Categorization, A.Lechervy, P-H. Gosselin and F.Precioso, accepté à MTAP 2012



Merci pour votre attention.



Complexité linéaire : Calcul de la fonction cible : f^*



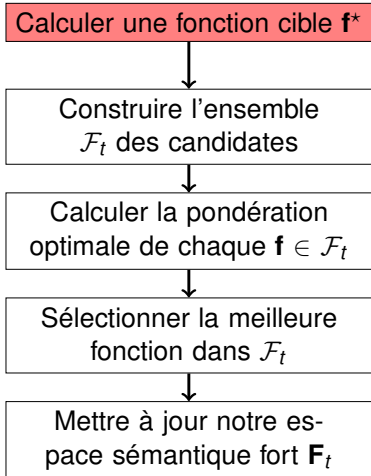
1. Calcul de la matrice de barycentres

- $\mathbf{G}_t \mathbf{G}_t^\top = \mathbf{G}_{t-1} \mathbf{G}_{t-1}^\top + \beta^* \mathbf{g}_t \mathbf{g}_t^\top$.
- $\mathcal{O}(c^2)$.

2. Calcul de la décomposition spectrale

- $\mathbf{G}_{t-1} \mathbf{G}_{t-1}^\top = \mathbf{V} \mathbf{\Lambda} \mathbf{V}^\top$.
- La décomposition spectrale complète de cette matrice a une complexité en $\mathcal{O}(c^3)$, mais en ne considérant que les premières valeurs de la décomposition, la complexité devient $\mathcal{O}(c^2)$.

Complexité linéaire : Calcul de la fonction cible : \mathbf{f}^*



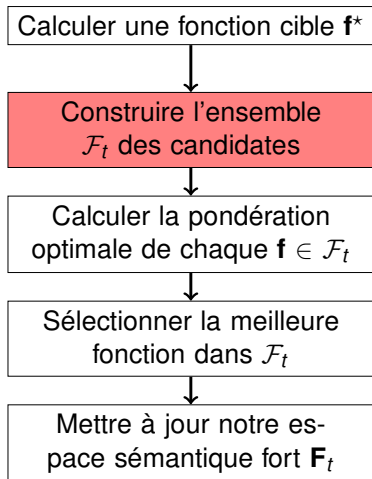
3. Calcul des nouveaux barycentres

- $\mathbf{g}^* = \sqrt{1 - \lambda \frac{\|\bar{\mathbf{K}}^*\|}{\|\bar{\mathbf{K}}_{t-1}\|}} \mathbf{v}$,
- $\|\bar{\mathbf{K}}^*\| = \sqrt{c} \rightarrow \mathcal{O}(1)$.
- $\|\bar{\mathbf{K}}_{t-1}\| \rightarrow \mathcal{O}(nt)$.

4. Calcul de la fonction faible cible

- $\mathbf{f}^* = \mathbf{Qg}^*$,
- $\mathcal{O}(nc)$.

Complexité linéaire : Construction de l'ensemble \mathcal{F}_t des candidates

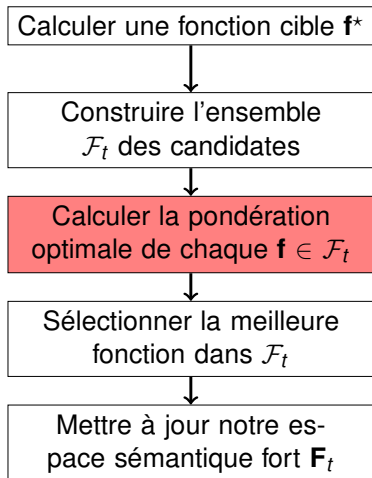


Construction des apprenants faibles

Les LMS sur l'ensemble d'apprentissage ont une complexité en $\mathcal{O}(nd)$, avec d la dimension des descripteurs visuels.

Complexité linéaire : Calcul de

$$\beta^*(\mathbf{f}) \quad \forall \mathbf{f} \in \mathcal{F}_t$$

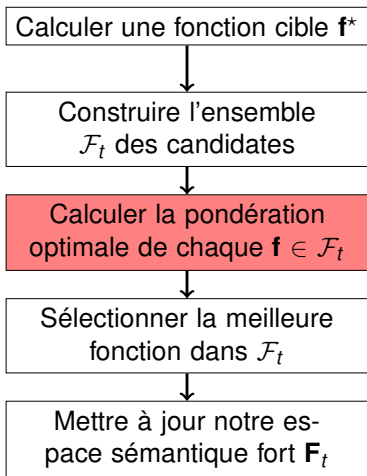


Compute $\beta^*(f)$

$$\beta^* = \frac{\|\bar{\mathbf{K}}_{t-1}\|^2 \langle \bar{\mathbf{k}}, \bar{\mathbf{K}}^* \rangle - \langle \bar{\mathbf{K}}_{t-1}, \bar{\mathbf{k}} \rangle \langle \bar{\mathbf{K}}_{t-1}, \bar{\mathbf{K}}^* \rangle}{\|\bar{\mathbf{k}}\|^2 \langle \bar{\mathbf{K}}_{t-1}, \bar{\mathbf{K}}^* \rangle - \langle \bar{\mathbf{K}}_{t-1}, \bar{\mathbf{k}} \rangle \langle \bar{\mathbf{k}}, \bar{\mathbf{K}}^* \rangle}$$

\implies nous avons besoin de
 seulement $\|\bar{\mathbf{K}}^*\|$, $\|\bar{\mathbf{f}\mathbf{f}^\top}\|$,
 $\langle \bar{\mathbf{f}\mathbf{f}^\top}, \bar{\mathbf{K}}_{t-1} \rangle$, $\langle \bar{\mathbf{f}\mathbf{f}^\top}, \bar{\mathbf{K}}^* \rangle$, $\langle \bar{\mathbf{K}}_{t-1}, \bar{\mathbf{K}}^* \rangle$
 et $\|\bar{\mathbf{K}}_{t-1}\|$.

Complexité linéaire : Calcul de $\|\overline{\mathbf{K}^*}\|$, $\|\overline{\mathbf{f}\mathbf{f}^\top}\|$, $\langle \overline{\mathbf{f}\mathbf{f}^\top}, \overline{\mathbf{K}_{t-1}} \rangle$, $\langle \overline{\mathbf{f}\mathbf{f}^\top}, \overline{\mathbf{K}^*} \rangle$, $\langle \overline{\mathbf{K}_{t-1}}, \overline{\mathbf{K}^*} \rangle$ et $\|\overline{\mathbf{K}_{t-1}}\|$



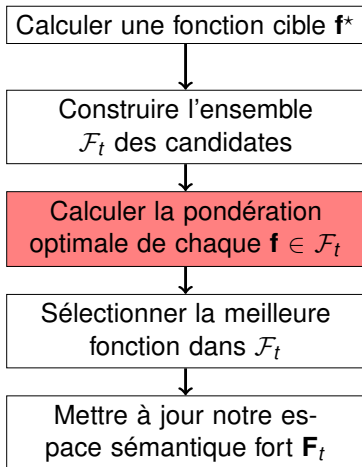
$\|\overline{\mathbf{K}^*}\|$ et $\|\overline{\mathbf{f}\mathbf{f}^\top}\|$

- $\|\overline{\mathbf{K}^*}\| = \sqrt{c} \rightarrow \mathcal{O}(1)$.
- $\|\overline{\mathbf{f}\mathbf{f}^\top}\| = \|\mathbf{f} - \text{mean}(\mathbf{f})\|^2 \rightarrow \mathcal{O}(n)$.

$\langle \overline{\mathbf{f}\mathbf{f}^\top}, \overline{\mathbf{K}_{t-1}} \rangle$ et $\langle \overline{\mathbf{f}\mathbf{f}^\top}, \overline{\mathbf{K}^*} \rangle$

- $\langle \overline{\mathbf{f}\mathbf{f}^\top}, \overline{\mathbf{K}^*} \rangle = \|\mathbf{f}^\top \mathbf{Q}\|^2 \rightarrow \mathcal{O}(nc)$.
- $\langle \overline{\mathbf{f}\mathbf{f}^\top}, \overline{\mathbf{K}_{t-1}} \rangle = \|(\mathbf{f} - \text{mean}(\mathbf{f}))^\top \mathbf{F}_{t-1}\|^2 \rightarrow \mathcal{O}(nt)$.

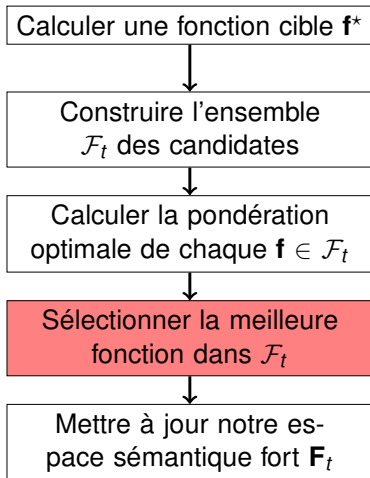
Complexité linéaire : Calcul de $\|\overline{\mathbf{K}}^*\|$, $\|\overline{\mathbf{f}\mathbf{f}^\top}\|$, $\langle \overline{\mathbf{f}\mathbf{f}^\top}, \overline{\mathbf{K}}_{t-1} \rangle$, $\langle \overline{\mathbf{f}\mathbf{f}^\top}, \overline{\mathbf{K}}^* \rangle$, $\langle \overline{\mathbf{K}}_{t-1}, \overline{\mathbf{K}}^* \rangle$ et $\|\overline{\mathbf{K}}_{t-1}\|$



$\langle \overline{\mathbf{K}}_{t-1}, \overline{\mathbf{K}}^* \rangle$ et $\|\overline{\mathbf{K}}_{t-1}\|$

- $\langle \overline{\mathbf{K}}_t, \overline{\mathbf{K}}^* \rangle = \langle \overline{\mathbf{K}}_{t-1}, \overline{\mathbf{K}}^* \rangle + \beta^* \langle \overline{\mathbf{f}\mathbf{f}^\top}, \overline{\mathbf{K}}^* \rangle \rightarrow \mathcal{O}(1)$.
- $\|\overline{\mathbf{K}}_t\|^2 = \|\overline{\mathbf{K}}_{t-1}\|^2 + \|\overline{\mathbf{f}\mathbf{f}^\top}\|^2 + 2\beta^* \langle \overline{\mathbf{K}}_{t-1}, \overline{\mathbf{f}\mathbf{f}^\top} \rangle \rightarrow \mathcal{O}(1)$.

Complexité linéaire : Sélection du candidat f qui maximise l'alignement

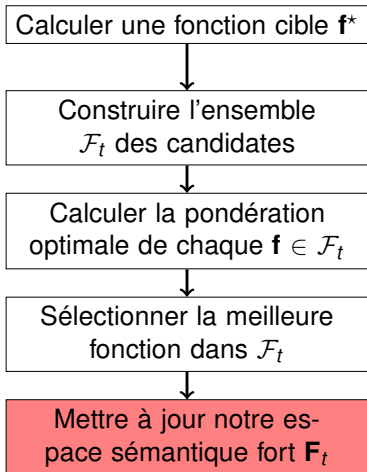


Calcul de l'alignement

$$\mathcal{A}_H(\mathbf{K}_{t-1} + \beta^* \mathbf{k}, \mathbf{K}^*) = \frac{\langle \bar{\mathbf{K}}_{t-1}, \bar{\mathbf{K}}^* \rangle + \beta^* \langle \bar{\mathbf{k}}, \bar{\mathbf{K}}^* \rangle}{\|\bar{\mathbf{K}}^*\| \|\bar{\mathbf{K}}_t\|}.$$

Complexité linéaire : Mise à jour de

$$\mathbf{F}_t = (\mathbf{F}_{t-1} \sqrt{\beta_t} \mathbf{f}_t)$$



Conclusion

Chaque étape t a une complexité calculatoire de $\mathcal{O}(c^2 + nc + nt) = \mathcal{O}(nt)$, si $c < n$ et $c < t$.