Approche SVM pour la combinaison des scores d'une méthode de reconnaissance de repliement

LORIA – 22/06/04

Philippe Veber

pveber@free.fr

DEA Génomique et Informatique, Université de Rennes1

Détection d'homologie

Méthodes basées sur la **séquence**

- classiques
  - BLAST, FASTA
  - PSI-BLAST

Détection d'homologie

Méthodes basées sur la **séquence**

- classiques
  - BLAST, FASTA
  - PSI-BLAST
- plus sophistiquées ...
  - SAM-T98 (HMM)
  - SVM+noyaux dédiés (mismatch kernel)

Détection d'homologie

Méthodes basées sur la **séquence**

- classiques
  - BLAST, FASTA
  - PSI-BLAST
- plus sophistiquées ...
  - SAM-T98 (HMM)
  - SVM+noyaux dédiés (mismatch kernel)

→ insuffisantes pour les homologies distantes

→ grande variabilité de la séquence entre deux homologues

Reconnaissance de repliement

Observations :

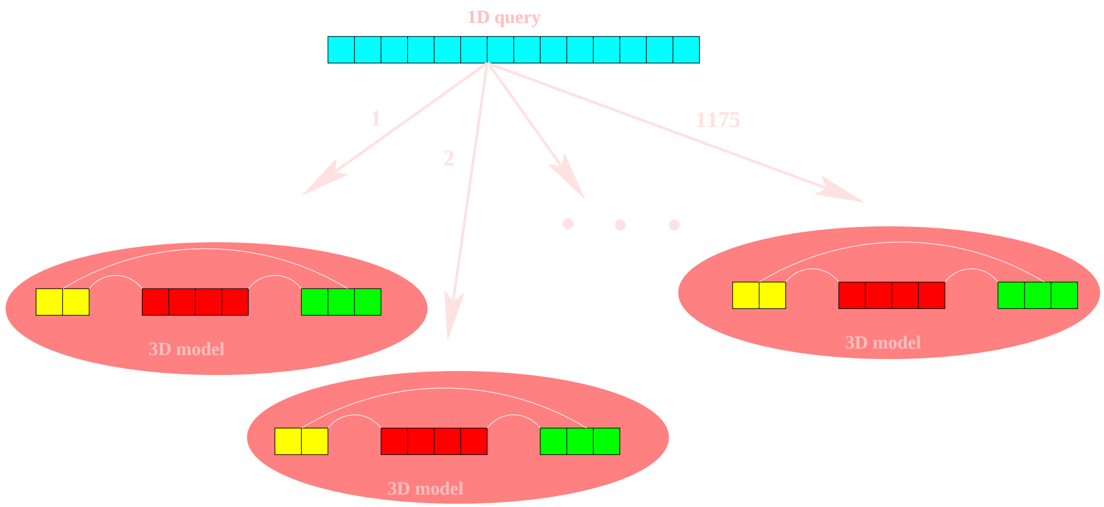
- la **structure 3D** évolue beaucoup moins vite que la séquence.
- le nombre de repliements présents dans la nature est relativement faible

Reconnaissance de repliement

Observations :

- la **structure 3D** évolue beaucoup moins vite que la séquence.
- le nombre de repliements présents dans la nature est relativement faible

Démarche :



Méthode FROST : 4 composants

- Bibliothèque de *coeurs*
  - coeur = modèle d'une famille de protéines adoptant un même repliement
  - environ 1000 coeurs

Méthode FROST : 4 composants

- Bibliothèque de *coeurs*
  - coeur = modèle d'une famille de protéines adoptant un même repliement
  - environ 1000 coeurs
- fonction de score
  - mesure l'affinité de la séquence pour un repliement

Méthode FROST : 4 composants

- Bibliothèque de *coeurs*
    - coeur = modèle d'une famille de protéines adoptant un même repliement
    - environ 1000 coeurs
- fonction de score
    - mesure l'affinité de la séquence pour un repliement
- algorithme d'alignement séquence/structure

Méthode FROST : 4 composants

- Bibliothèque de *coeurs*
  - coeur = modèle d'une famille de protéines adoptant un même repliement
  - environ 1000 coeurs
- fonction de score
  - mesure l'affinité de la séquence pour un repliement
- algorithme d'alignement séquence/structure
- analyse statistique du score d'alignement
  - rend les scores obtenus pour des cœurs différents comparables

La fonction de score . . .

- est cruciale pour les performances

La fonction de score . . .

- est cruciale pour les performances

MAIS

- est difficile à concevoir
- doit synthétiser des considérations très diverses

La fonction de score . . .

- est cruciale pour les performances

MAIS

- est difficile à concevoir
- doit synthétiser des considérations très diverses

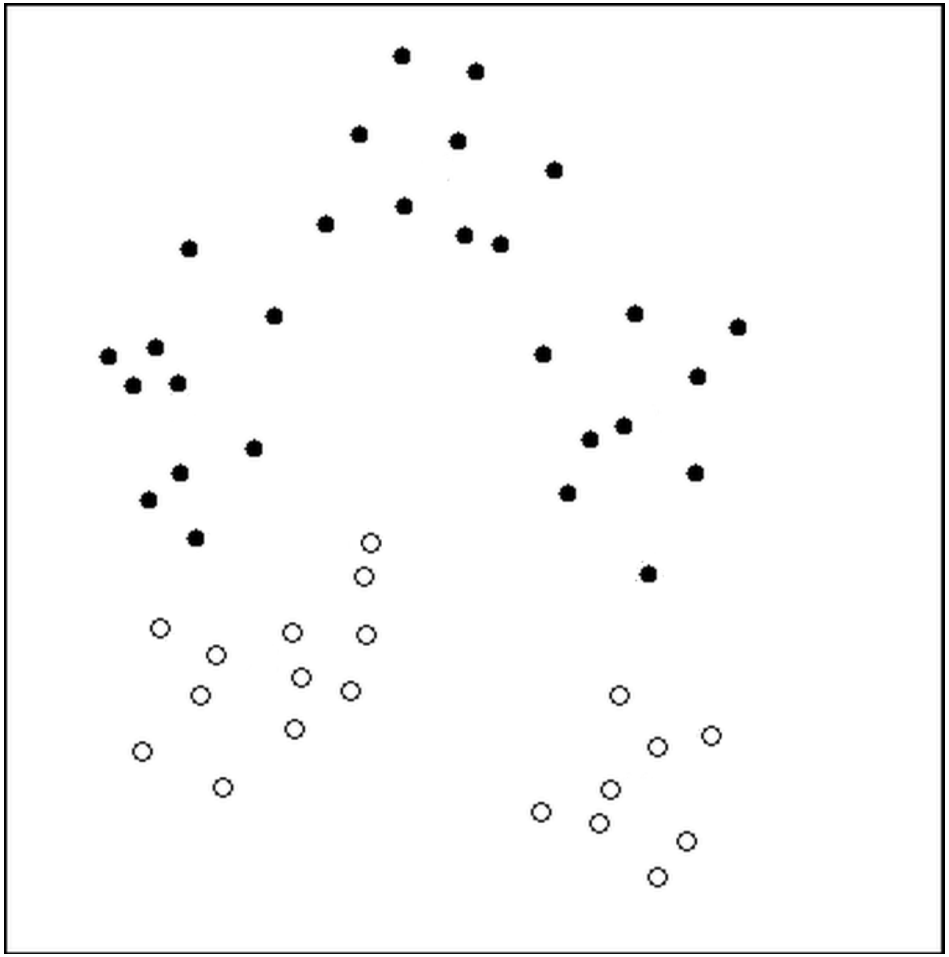
Stratégie adoptée dans FROST :

- développer M fonctions de scores indépendantes
  - démarche modulaire
  - facilite le développement de nouvellles fonctions de score
- combinaison des scores ?

Apprendre à classifier

$\mathcal{X}$: ensemble des observations
$\mathcal{Y} = \{-1; 1\}$: étiquettes

apprentissage = choix d'une fonction $f : \mathcal{X} \rightarrow \mathcal{Y}$ dans un ensemble $\mathcal{F}$ à partir d' exemples

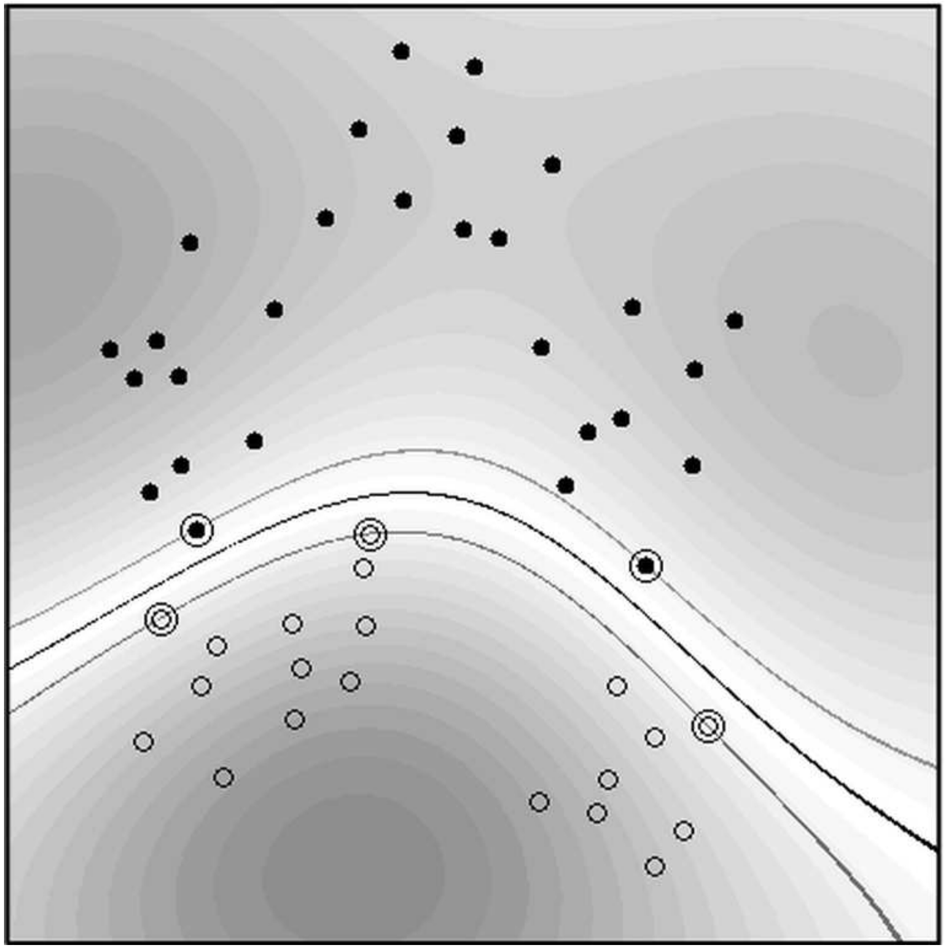


Apprendre à classifier

$\mathcal{X}$: ensemble des observations
$\mathcal{Y} = \{-1; 1\}$: étiquettes

apprentissage = choix d'une fonction $f : \mathcal{X} \rightarrow \mathcal{Y}$ dans un ensemble $\mathcal{F}$ à partir d' exemples

heuristique pour une bonne généralisation ?



Cas particulier

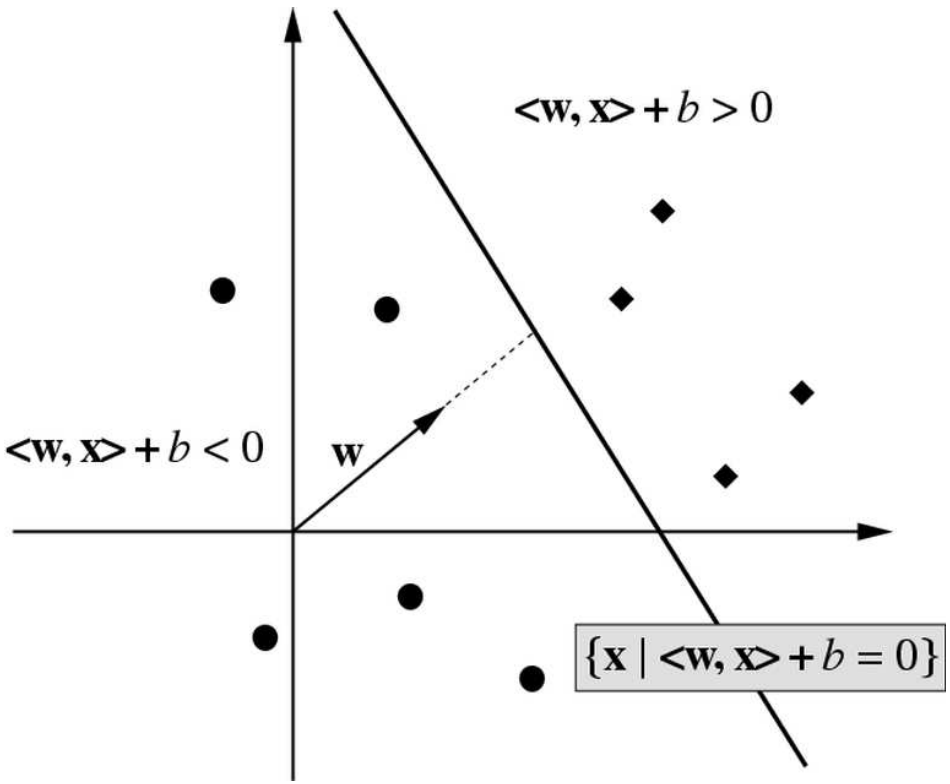
Prenons :

$\mathcal{X} = \mathbb{R}^n$

$\mathcal{F} =$ ens. des fonctions linéaires

exemples $\{(x_1, y_1), \ldots, (x_l, y_l)\}$

Quel hyperplan choisir ?



Formulation

Cas particulier de classes linéairement séparables :

Trouver $(w, b) \in \mathbb{R}^n \times \mathbb{R}$

Tel que $\forall i \in \{1, \ldots, l\} \quad y_i(\langle w, x_i \rangle + b) \geq 0$

Problème mal posé :

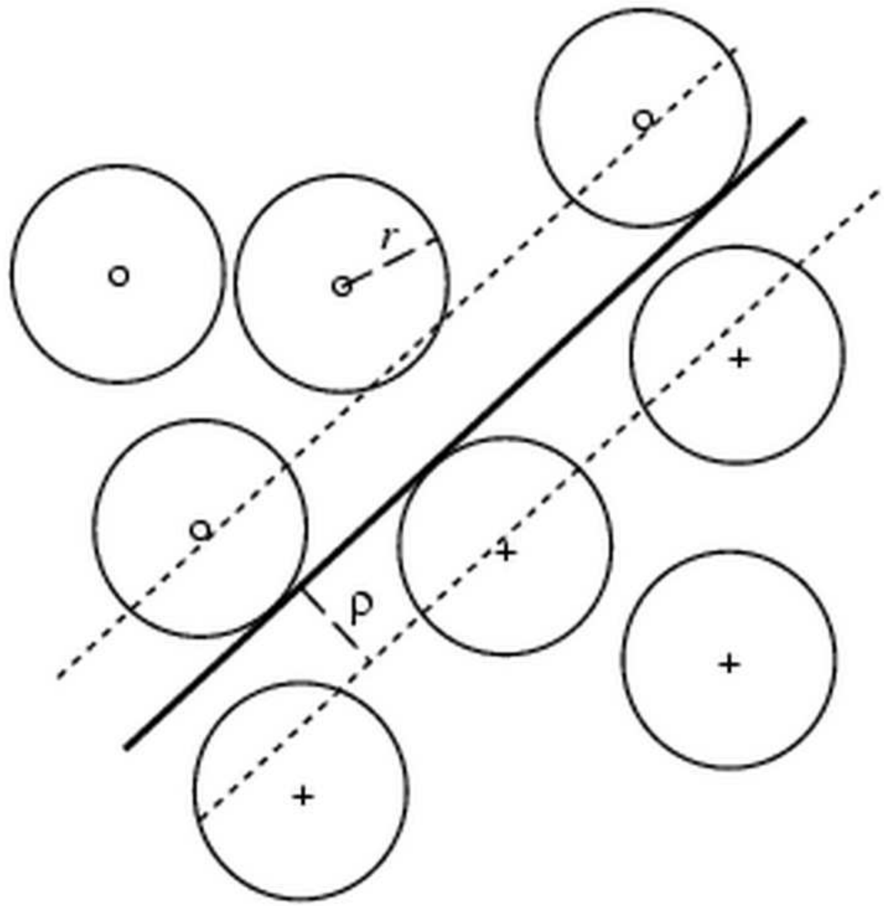
(w, b) solution et $\lambda > 0 \Rightarrow (\frac{w}{\lambda}, \frac{b}{\lambda})$ solution

Trouver $(w, b) \in \mathbb{R}^n \times \mathbb{R}$

Tel que $\forall i \in \{1, \ldots, l\} \quad y_i(\langle w, x_i \rangle + b) \geq 1$

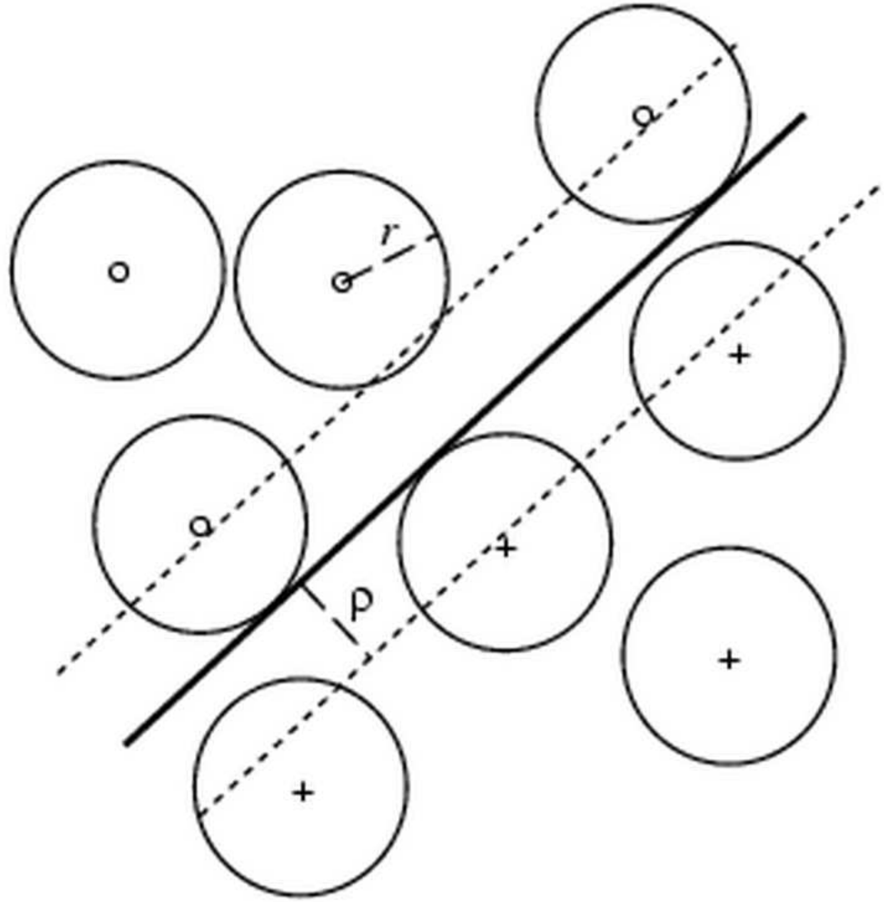
Heuristique : $\boxed{\text{Maximiser la marge}}$

- marge = distance du plan au point le plus proche



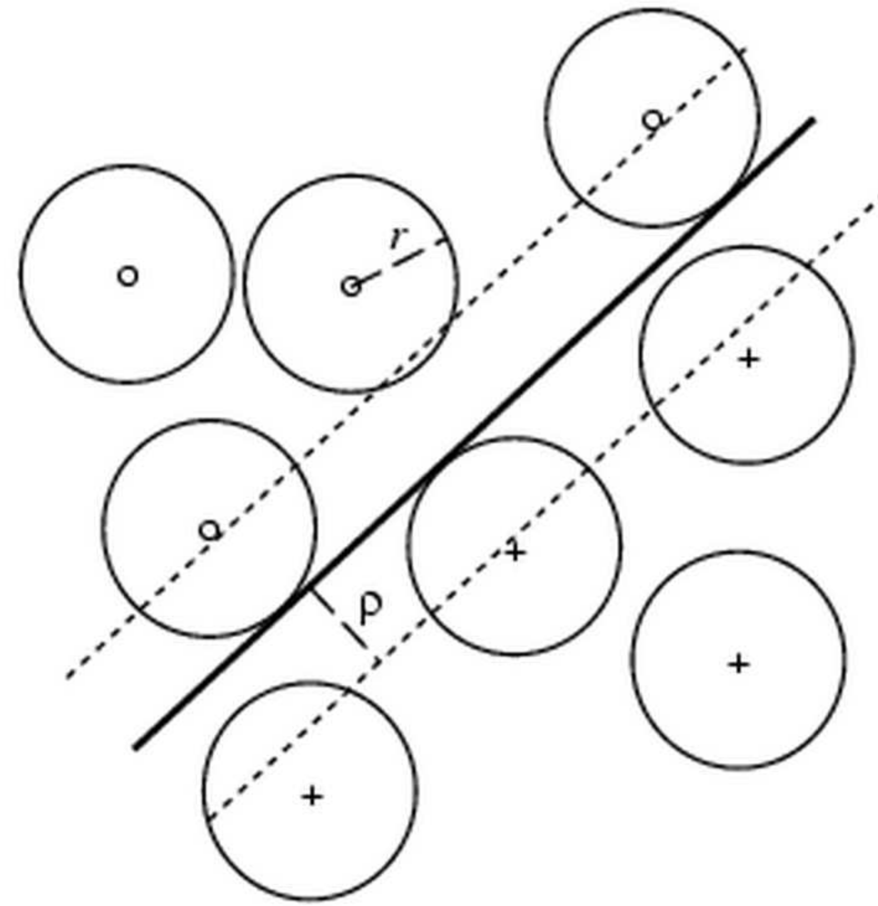
Heuristique : $\boxed{\text{Maximiser la marge}}$

- marge = distance du plan au point le plus proche
- argument théorique = converge vers la meilleure $f \in \mathcal{F}$



Heuristique : $\boxed{\text{Maximiser la marge}}$

- marge = distance du plan au point le plus proche
- argument théorique = converge vers la meilleure $f \in \mathcal{F}$
- argument pratique = séparation robuste, moins sensible au bruit des données réelles



Formulation primale

Marge inversement proportionnelle à $||w||^2$, d'où ...

Trouver $(w, b) \in \mathbb{R}^n \times \mathbb{R}$

Avec $||w||^2$ minimal

Tel que $\forall i \in \{1, \ldots, l\} \quad y_i(\langle w, x_i \rangle + b) \geq 1$

$\longrightarrow$ Problème d'optimisation convexe

Séparation non linéaire

Discrimination linéaire souvent insuffisante :

- dû au bruit ? $\longrightarrow$ classifieur à “marge douce”

Séparation non linéaire

Discrimination linéaire souvent insuffisante :

- dû au bruit ? $\longrightarrow$ classifieur à “marge douce”
  - contraintes réalisées à ξ_i près

$$\forall i \in \{1, \ldots, l\} \quad y_i(\langle w, x_i \rangle + b) + \xi_i \geq 1 \text{ avec } \xi_i \geq 0$$

Séparation non linéaire

Discrimination linéaire souvent insuffisante :

- dû au bruit ? $\longrightarrow$ classifieur à “marge douce”
  - contraintes réalisées à ξ_i près

$$\forall i \in \{1, \ldots, l\} \quad y_i(\langle w, x_i \rangle + b) + \xi_i \geq 1 \text{ avec } \xi_i \geq 0$$

  - pénalité

$$\text{Minimiser } ||w||^2 + C \sum \xi_i$$

Séparation non linéaire

Discrimination linéaire souvent insuffisante :

- dû au bruit ? $\longrightarrow$ classifieur à “marge douce”
  - contraintes réalisées à ξ_i près

$$\forall i \in \{1, \ldots, l\} \quad y_i(\langle w, x_i \rangle + b) + \xi_i \geq 1 \text{ avec } \xi_i \geq 0$$

  - pénalité

$$\text{Minimiser } ||w||^2 + C \sum \xi_i$$

  - toujours convexe

Séparation non linéaire

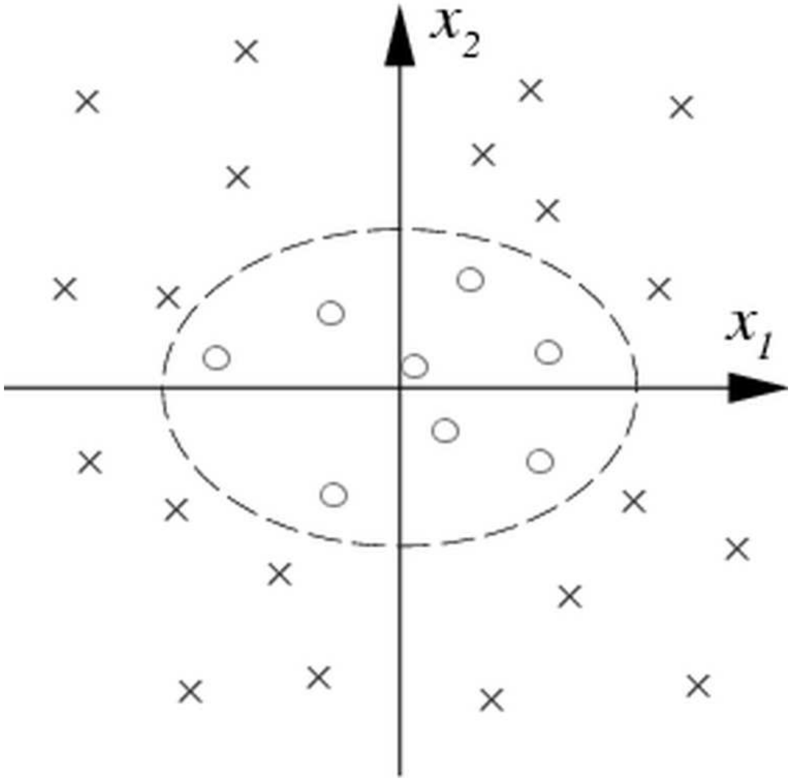
Discrimination linéaire souvent insuffisante :

- problème de représentation ?

Séparation non linéaire

Discrimination linéaire souvent insuffisante :

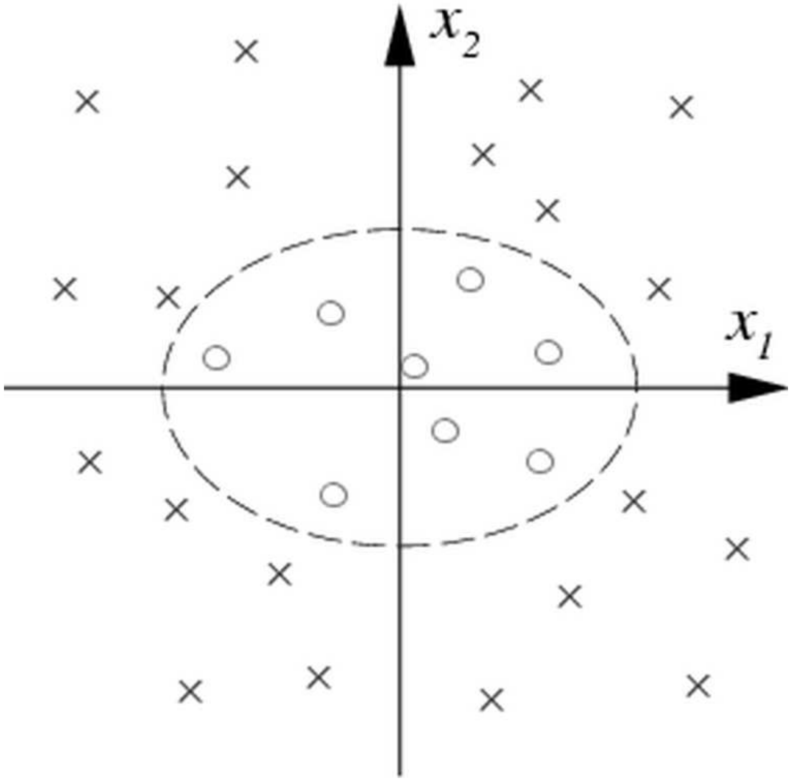
- problème de représentation ?



Séparation non linéaire

Discrimination linéaire souvent insuffisante :

- problème de représentation ?



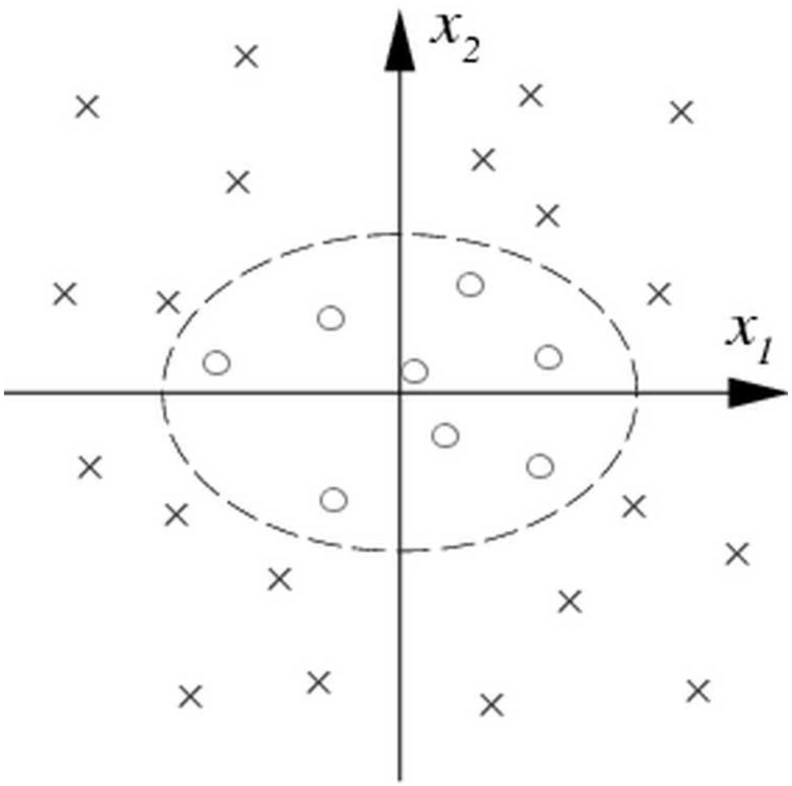
Changement de représentation :

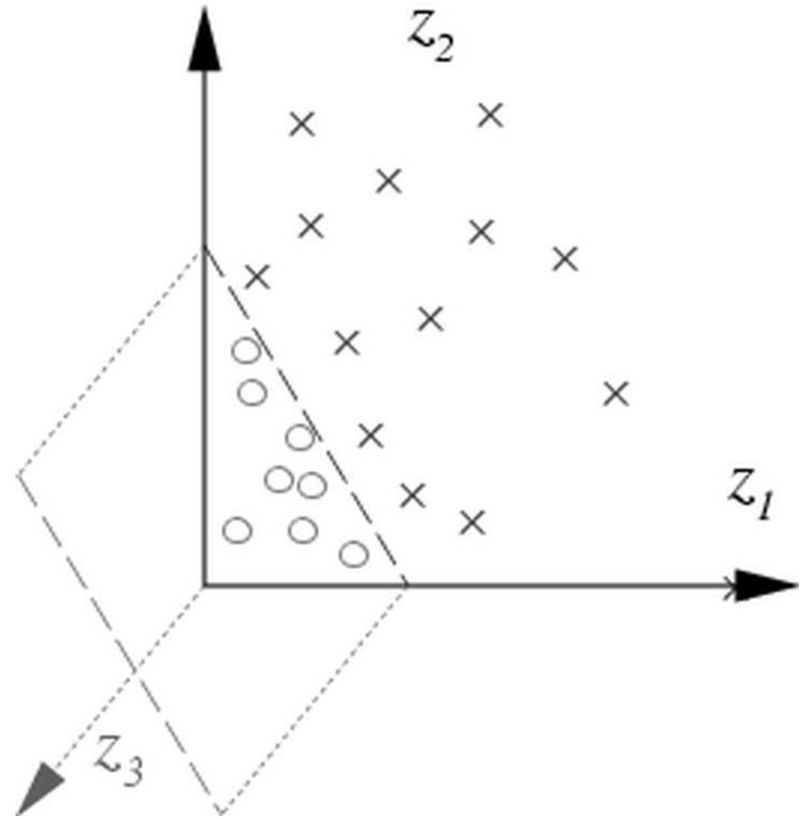
$$\Phi : \begin{cases} \mathbb{R}^2 \longrightarrow \mathbb{R}^3 \\ (x_1, x_2) \longmapsto (x_1^2, \sqrt{2}x_1x_2, x_2^2) \end{cases}$$

Séparation non linéaire

Discrimination linéaire souvent insuffisante :

- problème de représentation ?





Kernel trick

Problème :

$$x_1, \ldots, x_n \longrightarrow \Phi(x_1), \ldots, \Phi(x_n) \longrightarrow (w, b) \text{ dans } \Phi(\mathcal{X})$$

$=$ processus coûteux en temps de calcul

$\boxed{\text{MAIS}}$

le calcul de l'hyperplan requiert seulement la matrice de Graam, *i.e.* les produits scalaires $\langle x_i \mid x_j \rangle$
On calcule implicitement l'hyperplan dans l'espace de représentation, en donnant à l'algorithme la matrice $k(x_i, x_j)$, où le noyau k est défini par :

$$k(x_i, x_j) = \langle \Phi(x_i) \mid \Phi(x_j) \rangle$$

Résumé – Une SVM c'est ...

- un algorithme qui calcule l'hyperplan séparateur maximisant la marge

Résumé – Une SVM c'est ...

- un algorithme qui calcule l'hyperplan séparateur maximisant la marge
- les données sont envoyées dans un espace de représentation par une application non linéaire

Résumé – Une SVM c'est ...

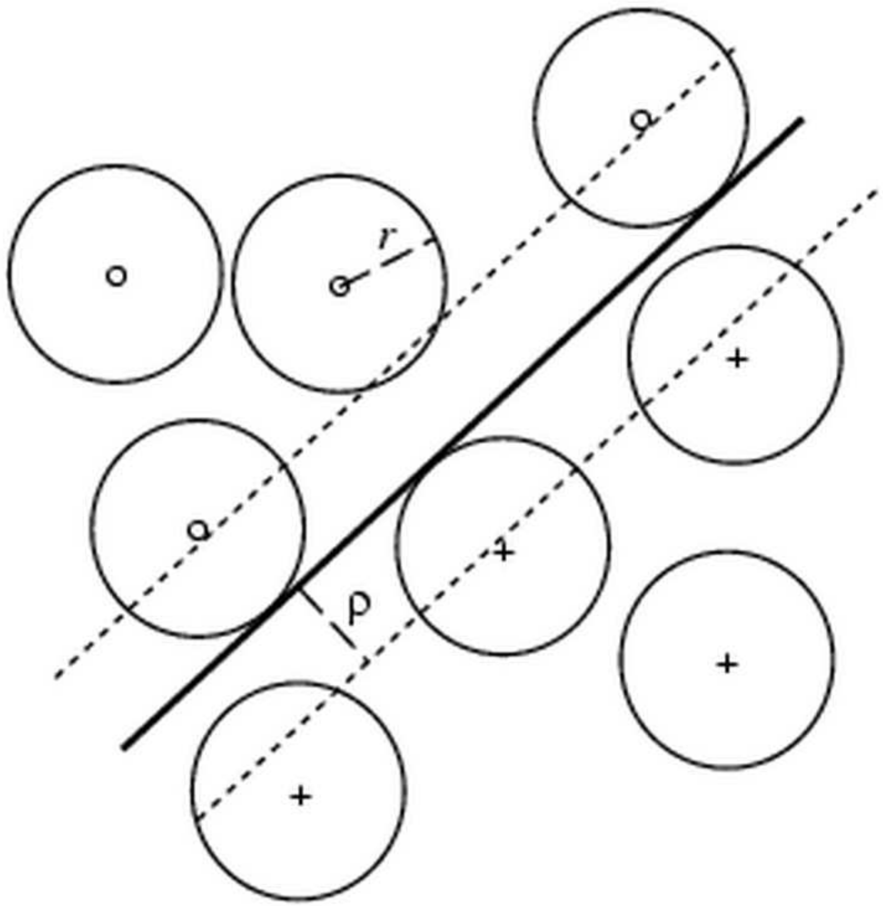
- un algorithme qui calcule l'hyperplan séparateur maximisant la marge
- les données sont envoyées dans un espace de représentation par une application non linéaire
- la séparation est effectuée dans l'espace de représentation

Pourquoi SV ?

- SV = Support Vectors

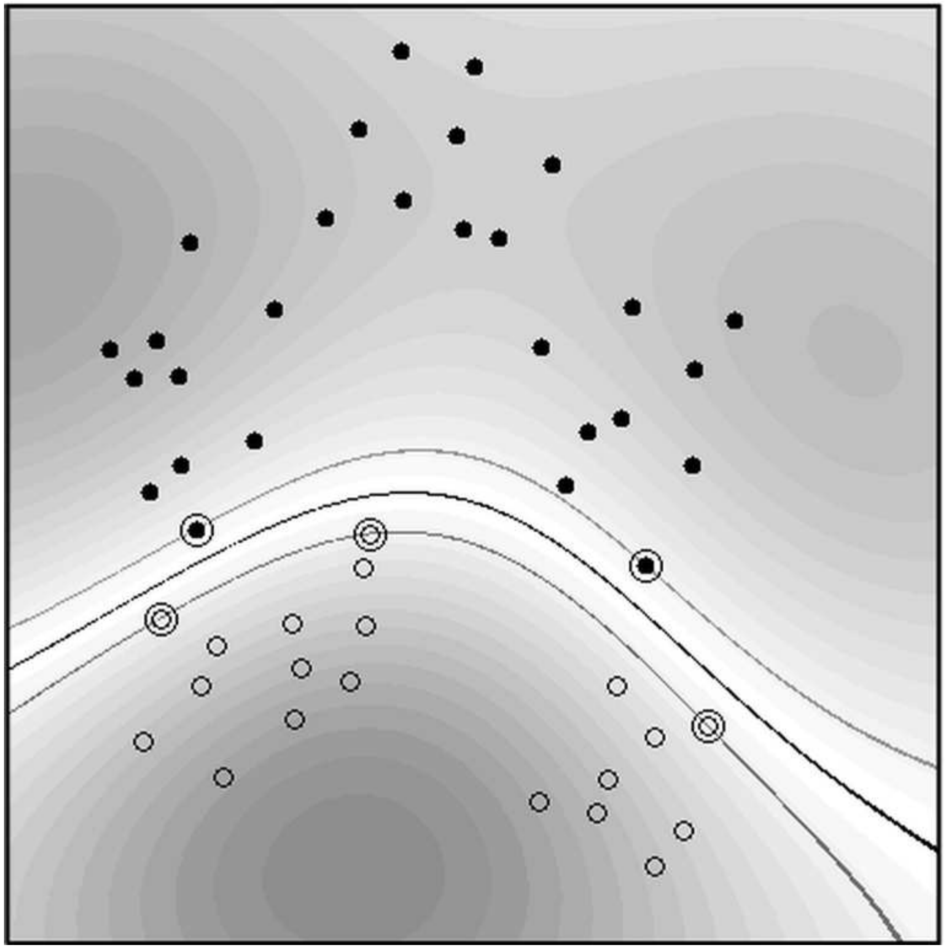
Pourquoi SV ?

- SV = Support Vectors
- l'hyperplan ne dépend que d'un sous-ensemble des exemples, appelés *vecteurs supports*



Pourquoi SV ?

- SV = Support Vectors
- l'hyperplan ne dépend que d'un sous-ensemble des exemples, appelés *vecteurs supports*
- intuitivement, les autres exemples sont considérés comme moins pertinents



Indice de confiance

- SVM $\longrightarrow$ sortie binaire

Indice de confiance

- SVM $\longrightarrow$ sortie binaire
- intuition :

  proximité de la surface de décision

  $\longrightarrow$ incertitude de la classification

Indice de confiance

- SVM $\longrightarrow$ sortie binaire
- intuition :

  proximité de la surface de décision

  $\longrightarrow$ incertitude de la classification
- on peut utiliser la distance (calibrée) à la surface de décision comme indice de confiance de la prédiction

Indice de confiance

- SVM $\longrightarrow$ sortie binaire
- intuition :

  proximité de la surface de décision

  $\longrightarrow$ incertitude de la classification
- on peut utiliser la distance (calibrée) à la surface de décision comme indice de confiance de la prédiction
- dans la suite, on jouera sur cet indice pour évaluer le ratio spécificité/sensibilité

Revenons à nos moutons

Actuellement, dans FROST, 2 fonctions de score :

- alignement de séquences
- alignement de séquences sachant que certains sites sont en contact

$\boxed{\text{Observation :}}$

ces deux fonctions sont complémentaires

Evaluation

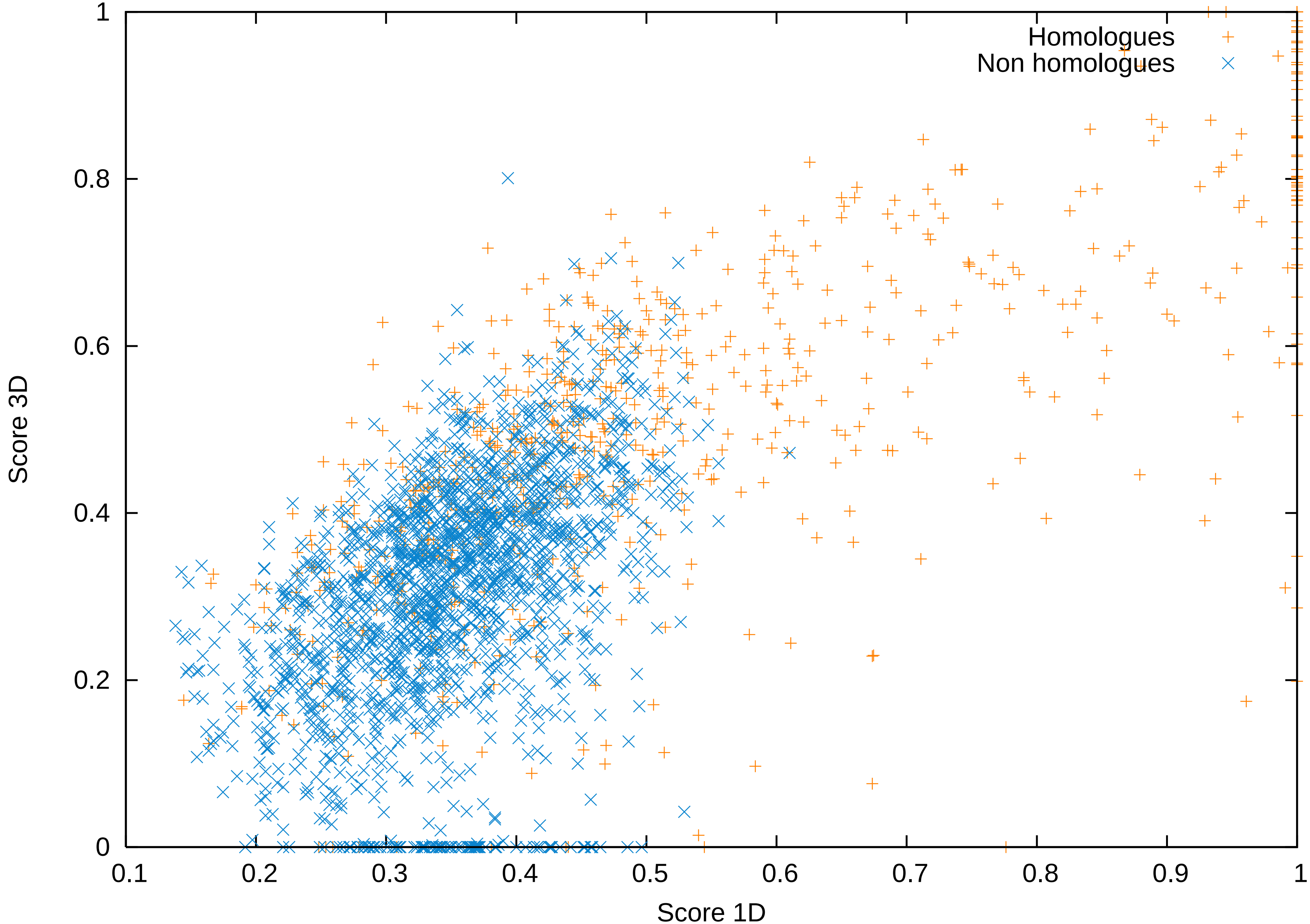
Benchmark :

- 257 protéines
- alignements “tous contre tous”
- environ 1600 couples de même repliement à détecter (non triviaux)

Mesure :

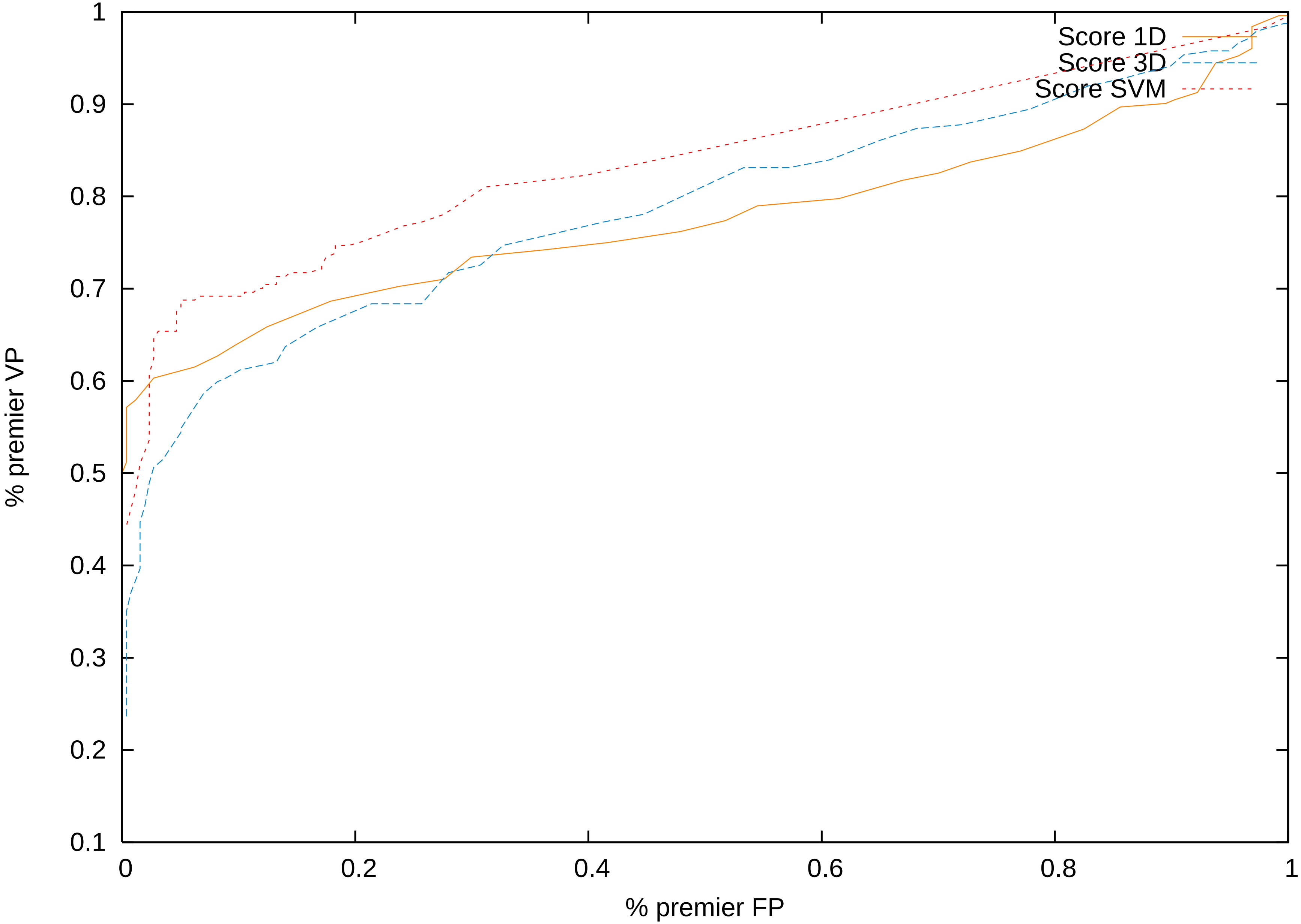
- sensibilité : capacité à détecter les paires homologues
- spécificité : capacité à discriminer efficacement les paires non-homologues
- courbe ROC (Receiver Operating Characteristic) : % VP contre % FP

Le problème



Homologues
Non homologues
Score 3D
Score 1D
0
0.2
0.4
0.6
0.8
1
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

Résultats



Score 1D
Score 3D
Score SVM
% premier VP
% premier FP
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0
0.2
0.4
0.6
0.8
1

Discussion

- amélioration légère . . .

Discussion

- amélioration légère . . .
- mais pas décisive vis à vis de la meilleure courbe

Discussion

- amélioration légère ...
- mais pas décisive vis à vis de la meilleure courbe
- en revanche ...

Discussion

- amélioration légère ...
- mais pas décisive vis à vis de la meilleure courbe
- en revanche ...
- on peut maintenant intégrer facilement beaucoup de nouveaux scores !

Application

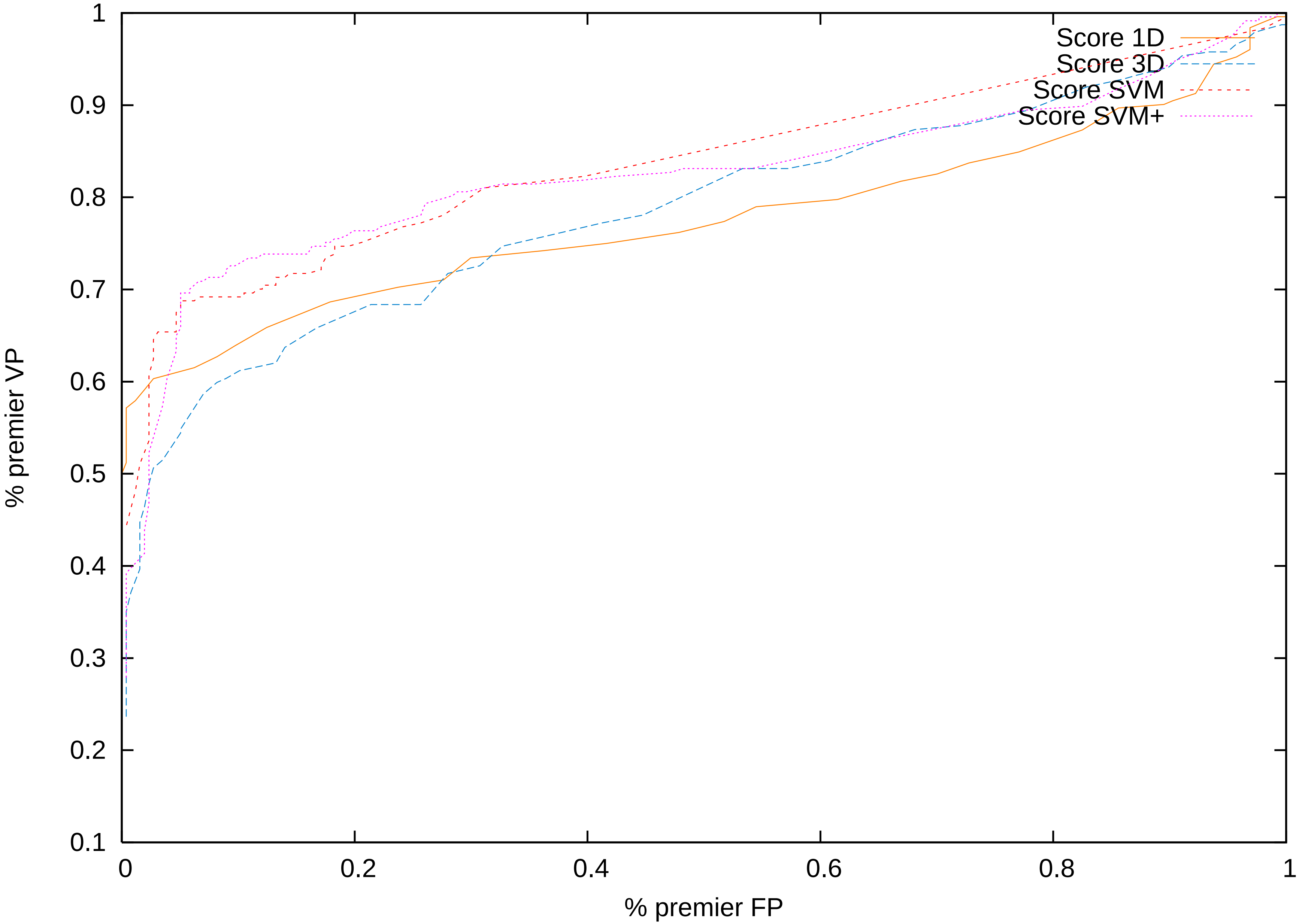
Utilisation de la prédiction de structure secondaire :

$$
\begin{array}{ccccc}
 & & \text{predictions} & & \\
\text{requête} & \longrightarrow & \ldots\text{CCHHHHCC}\ldots & \longrightarrow & \text{composition en H,E,C} \\
\text{cœur} & \longrightarrow & \ldots\text{CCHHHCCC}\ldots & \longrightarrow & \text{composition en H,E,C}
\end{array}
$$

Vecteur de score :

$$
\begin{pmatrix}
\text{score 1D} \\
\text{score 3D} \\
\text{\% H dans requête} \\
\text{\% E dans requête} \\
\text{\% C dans requête} \\
\text{\% H dans cœur} \\
\text{\% E dans cœur} \\
\text{\% C dans cœur}
\end{pmatrix}
$$

Résultats



Score 1D
Score 3D
Score SVM
Score SVM+
% premier VP
% premier FP
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0
0.2
0.4
0.6
0.8
1

Travail en cours

Prise en compte plus “sérieuse” de la prédiction de structure secondaire :

- score d’alignement des prédictions de structure secondaires
- score basé sur le SOV