**Master Recherche Orsay 2006-2007**

# Fouille de Données et Apprentissage

**Michèle Sebag, Antoine Cornuéjols, Balazs Kegl**

TAO : Thème Apprentissage et Optimisation
Université Paris-Sud
http://tao.lri.fr/

# Problèmes de représentation

- Sélection d'attributs
- Changements de représentation linéaires
- Changements de représentation non linéaires
- Une étude de cas

# Au début sont les données...



- Diabete : $n$ individus et $m$ variables

$$X = \begin{pmatrix} age & sex & \dots & x_{m=10} \\ 59 & 2 & \dots & 87 \\ \vdots & \vdots & \dots & \vdots \\ 36 & 1 & \dots & 92 \end{pmatrix} ; \mathbf{y} = \begin{pmatrix} diab. \\ 151 \\ \vdots \\ 57 \end{pmatrix}$$

| | AGE | SEX | BMI | BP | ··· Serum Measurements ··· | | | | | | Response |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Patient | x1 | x2 | x3 | x4 | x5 | x6 | x7 | x8 | x9 | x10 | y |
| 1 | 59 | 2 | 32.1 | 101 | 157 | 93.2 | 38 | 4 | 4.9 | 87 | 151 |
| 2 | 48 | 1 | 21.6 | 87 | 183 | 103.2 | 70 | 3 | 3.9 | 69 | 75 |
| 3 | 72 | 2 | 30.5 | 93 | 156 | 93.6 | 41 | 4 | 4.7 | 85 | 141 |
| 4 | 24 | 1 | 25.3 | 84 | 198 | 131.4 | 40 | 5 | 4.9 | 89 | 206 |
| 5 | 50 | 1 | 23.0 | 101 | 192 | 125.4 | 52 | 4 | 4.3 | 80 | 135 |
| 6 | 23 | 1 | 22.6 | 89 | 139 | 64.8 | 61 | 2 | 4.2 | 68 | 97 |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| 441 | 36 | 1 | 30.0 | 95 | 201 | 125.2 | 42 | 5 | 5.1 | 85 | 220 |
| 442 | 36 | 1 | 19.6 | 71 | 250 | 133.2 | 97 | 3 | 4.6 | 92 | 57 |

**Table 1.** Diabetes study. 442 diabetes patients were measured on 10 baseline variables. . prediction model was desired for the response variable, a measure of disease progression one yea after baseline.

- *microarray* : $n = 38 \ll m = 7126$

# Motivations

## Avant l'apprentissage : décrire les données...

- Une description trop pauvre — on ne peut rien faire
- Une description trop riche — on doit filtrer les descripteurs

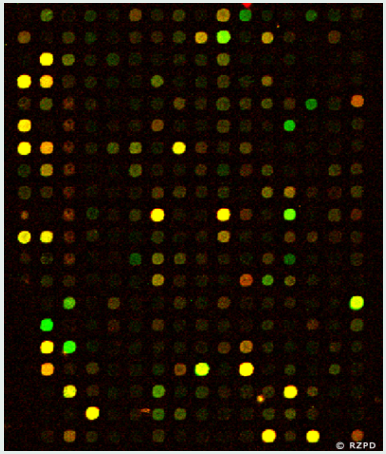
## Pourquoi ?

- L'apprentissage n'est pas un problème bien posé
- $\Longrightarrow$ Rajouter de l'information inutile (l'age du vélo de ma grand-mère) peut dégrader les hypothèses obtenues.

## Le but caché : sélectionner ou construire des descripteurs ?

- Feature Construction : construire les bons descripteurs
- A partir desquels il sera facile d'apprendre
- Les meilleurs descripteurs = les bonnes hypothèses...

# Quand l'apprentissage c'est la sélection d'attributs




Bio-informatique

- 30 000 gènes
- peu d'exemples (chers)
- but : trouver les gènes pertinents

# Il est facile de faire n'importe quoi

*Un exemple d'aventure fort désagréable...*

http://www-stat.stanford.edu/∼hastie/TALKS/barossa.pdf

# (Rappel) Définition de p-value

Contexte : observation — le rouge est sorti 14 fois sur 20

Question : est-ce le hasard ? — deux hypothèses

- $H_0$: le casino est honnête — Pr(rouge) = 1/2
- ... ou non

p-value : Proba ($\sim$ observation | $H_0$)

Nb de rouges sur N tirages $\sim \mathcal{B}(N, 1/2)$

Pr(# rouges >= 14) = .057

... On rejette l'hypothèse $H_0$ à 5% de niveau de confiance

# Position du problème

## Buts

- Sélection : trouver un sous-ensemble d'attributs
- Ordre/Ranking : ordonner les attributs

## Formulation

Soient les attributs $\mathcal{A} = \{a_1, ..a_d\}$. Soit la fonction :

$$\begin{aligned} \mathcal{F} : \mathcal{P}(\mathcal{A}) &\mapsto \mathbb{R} \\ A \subset \mathcal{A} &\mapsto Err(A) = \text{ erreur min. des hypothèses fondées sur } A \end{aligned}$$

**Trouver** $Argmin(\mathcal{F})$

## Difficultés

- Un problème d'optimisation combinatoire ($2^d$)
- D'une fonction $\mathcal{F}$ inconnue...

# Approches

Filter — méthode univariée

Définir $score(a_i)$; ajouter itérativement les attributs maximisant $score$
ou retirer itérativement les attributs minimisant $score$

$+$ simple et pas cher

$-$ optima très locaux

Rq : on peut bactracker : meilleurs optima, mais plus cher

Wrapping — méthode multivariée

Mesurer la qualité d'un ensemble d'attributs :
estimer $\mathcal{F}(a_{i1}, ...a_{ik})$

$-$ cher : une estimation = un pb d'apprentissage.

$+$ optima meilleurs

# Approches filtre, 1

## Notations

Base d'apprentissage : $\mathcal{E} = \{(x_i, y_i), i = 1..n, y_i \in \{-1, 1\}\}$

$a(x_i)$ = valeur attribut $a$ pour exemple $(x_i)$

## Corrélation

$$corr(a) = \frac{\sum_i a(x_i).y_i}{\sqrt{\sum_i (a(x_i))^2 \times \sum_i y_i^2}} \propto \sum_i a(x_i).y_i = \ < a, y >$$

## Limites

- Attributs corrélés entre eux
- Dépendance non linéaire

# Approches filtre, 2

## Corrélation et projection

Stoppiglia et al. 2003

Repeat

- $a^*$ = attribut le plus corrélé à la classe

$$a^* = argmax\{\sum_i a(x_i)y_i,\ a \in \mathcal{A}\}$$

- Projeter les autres attributs sur l'espace orthogonal à $a^*$

$$\forall b \in \mathcal{A} \qquad b \rightarrow\ b - \frac{<a^*,b>}{<a^*,a^*>}\, a^*$$

$$b(x_i) \rightarrow\ b(x_i) - \frac{\sum_j a^*(x_j)b(x_j)}{\sum_j a^*(x_j)^2} a^*(x_i)$$

# Corrélation et projection, suite

- Projeter $y$ sur l'espace orthogonal à $a^*$

$$y \rightarrow y - \frac{<a^*,y>}{<a^*,a^*>}a^*$$
$$y_i \rightarrow y_i - -\frac{\sum_j a^*(x_j)y_j}{\sum_j a^*(x_j)^2}a^*(x_i)$$

- Until Critère d'arrêt
  - Rajouter des attributs aléatoires ($r(x_i) = \pm 1$) *probe*
  - Quand le critère de corrélation sélectionne des attributs aléatoires, s'arrêter.

Limitations

quand il y a plus de 6-7 attributs pertinents, ne marche pas bien.

# Approches filtre, 3

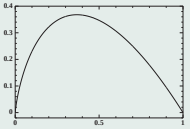
Gain d'information

arbres de décision

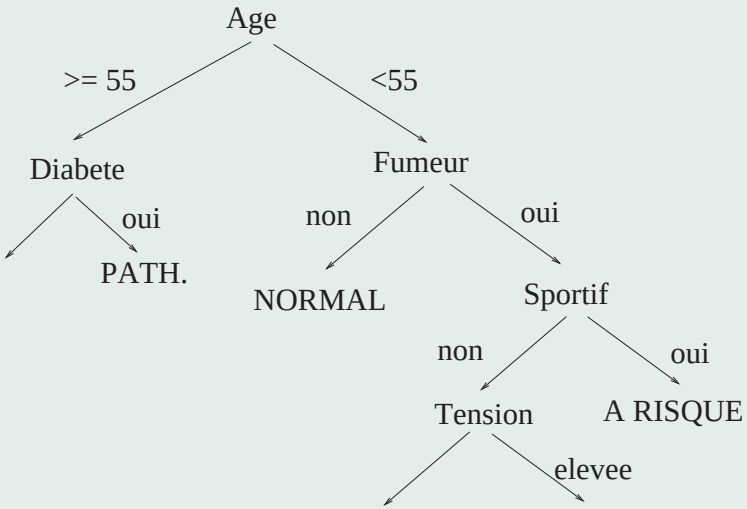
$$p([a = v]) = Pr(y = 1|a(x_i) = v)$$

$$QI([a = v]) = -p([a = v]) \log p([a = v])$$

$$QI(a) = \sum_v Pr(a(x_i) = v) QI([a = v])$$



# Gain d'information, suite




## Limitations

Les mêmes que celles des arbres de décision

Problème de XOR.

# Quelques scores

*en fouille de textes, contexte supervisé*

Notations : $c_i$ une classe $a_k$ un mot (ou terme)

Critères

1. Fréquence conditionnelle $P(c_i|a_k)$
2. Information mutuelle $P(c_i, a_k) Log(\frac{P(c_i,a_k)}{P(c_i)P(a_k)})$
3. Gain d'information $\sum_{c_i,\neg c_i} \sum_{a_k,\neg a_k} P(c,a) Log \frac{p(t,c)}{P(t)P(c)}$
4. Chi-2 $\frac{(P(t,c)P(\neg t,\neg c)-P(t,\neg c)P(\neg t,c))^2}{P(t)P(\neg t)P(c)P(\neg c)}$
5. Pertinence $Log \frac{P(t,c)+d}{P(\neg t,\neg c)+d}$

# Approches wrapper

## Principe générer/tester

Etant donné une liste de candidats $\mathcal{L} = \{A_1, .., A_p\}$

- Générer un candidat $A$
- Calculer $\mathcal{F}(A)$
  - apprendre $h_A$ à partir de $\mathcal{E}_{|A}$
  - tester $h_A$ sur un ensemble de test $= \hat{\mathcal{F}}(A)$
- Mettre à jour $\mathcal{L}$.

## Algorithmes

- hill-climbing / multiple restart
- algorithmes génétiques Vafaie-DeJong, IJCAI 95
- (*) programmation génétique & feature construction.

Krawiec, GPEH 01

# Approches a posteriori

Principe

- Construire des hypothèses
- En déduire les attributs importants
- Eliminer les autres
- Recommencer

Algorithme : SVM Recursive Feature Elimination Guyon et al. 03

- SVM linéaire $\longrightarrow h(x) = sign(\sum w_i.a_i(x) + b)$
- Si $|w_i|$ est petit, $a_i$ n'est pas important
- Eliminer les $k$ attributs ayant un poids min.
- Recommencer.

# Limites

Hypothèses linéaires

- Un poids par attribut.

Quantité des exemples

- Les poids des attributs sont liés.
- La dimension du système est liée au nombre d'exemples.

Or le pb de FS se pose souvent quand il n'y a pas assez d'exemples

# Problèmes de représentation

- Sélection d'attributs
- Changements de représentation linéaires
- Changements de représentation non linéaires
- Une étude de cas

# Données : Matrices

(Rappels d'algèbre)



- Diabete : $n$ individus et $m$ variables

$$X = \begin{pmatrix} age & sex & \dots & x_{m=10} \\ 59 & 2 & \dots & 87 \\ \vdots & \vdots & \dots & \vdots \\ 36 & 1 & \dots & 92 \end{pmatrix} ; \mathbf{y} = \begin{pmatrix} diab. \\ 151 \\ \vdots \\ 57 \end{pmatrix}$$

| | AGE | SEX | BMI | BP | ··· Serum Measurements ··· | | | | | | Response |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Patient | x1 | x2 | x3 | x4 | x5 | x6 | x7 | x8 | x9 | x10 | y |
| 1 | 59 | 2 | 32.1 | 101 | 157 | 93.2 | 38 | 4 | 4.9 | 87 | 151 |
| 2 | 48 | 1 | 21.6 | 87 | 183 | 103.2 | 70 | 3 | 3.9 | 69 | 75 |
| 3 | 72 | 2 | 30.5 | 93 | 156 | 93.6 | 41 | 4 | 4.7 | 85 | 141 |
| 4 | 24 | 1 | 25.3 | 84 | 198 | 131.4 | 40 | 5 | 4.9 | 89 | 206 |
| 5 | 50 | 1 | 23.0 | 101 | 192 | 125.4 | 52 | 4 | 4.3 | 80 | 135 |
| 6 | 23 | 1 | 22.6 | 89 | 139 | 64.8 | 61 | 2 | 4.2 | 68 | 97 |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| 441 | 36 | 1 | 30.0 | 95 | 201 | 125.2 | 42 | 5 | 5.1 | 85 | 220 |
| 442 | 36 | 1 | 19.6 | 71 | 250 | 133.2 | 97 | 3 | 4.6 | 92 | 57 |

**Table 1.** Diabetes study. 442 diabetes patients were measured on 10 baseline variables. . prediction model was desired for the response variable, a measure of disease progression one yea after baseline.

- *microarray* : $n = 38 \ll m = 7126$

# Projeter

Base d'exemples

$$\mathcal{L} = \{(x_i, y_i), x_i \in \mathbb{R}^D, i = 1 \ldots N\}$$

Changement de représentation linéaire

$$x \in \mathbb{R}^D \rightarrow \begin{array}{c} x' \in \mathbb{R}^d,\ d << D \\ x' = Ax \end{array}$$

$\equiv$ Changement de base : $\{e_1, ..e_D\} \rightarrow \{u_1, \ldots u_d\}$

$$\begin{array}{rl} x = & a_1 e_1 + \ldots + a_D e_D \\ \rightarrow x' = & b_1 u_1 + \ldots + b_d u_d \end{array}$$

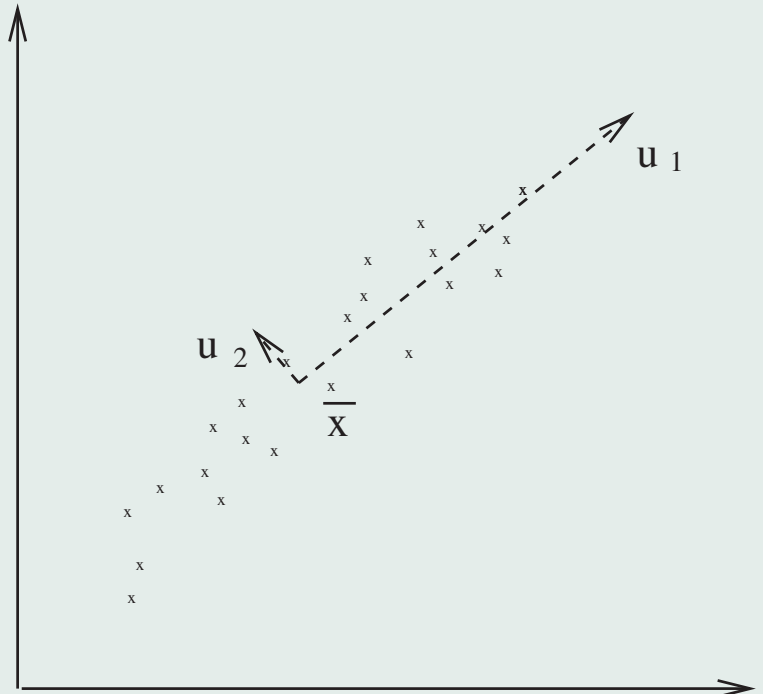
Critère

Minimiser la perte d'information $\sum_{i=1}^{N} ||x_i - x'_i||^2$

# Analyse en Composantes Principales

**Théorème**

La meilleure projection en dimension $d$ : sur les $d$ premiers vecteurs propres de la matrice de covariance des vecteurs $x_1, \ldots, x_N$.



# Analyse en Composantes Principales, Méthode, 1

- Centrer les vecteurs $x_i$

$$\begin{aligned} \bar{x} &= \frac{1}{N}\sum_{i=1}^{N} x_i \\ z_i &= x_i - \bar{x} \end{aligned}$$

- Calculer la matrice de covariance

$$C = (C_{ij})_{N\times N} \quad C_{ij} = \ < z_i, z_j >$$

  Cette matrice est symétrique.

- La diagonaliser

$$C = U\Delta U', \ \Delta = Diag(\lambda_1, \ldots, \lambda_D)$$

# Analyse en Composantes Principales, suite

(sans perte de généralité, $\lambda_i > \lambda_{i+1}$)

- A chaque valeur propre $\lambda_i$ est associé un vecteur propre $u_i$
- Ces vecteurs propres sont orthogonaux : définissent une base
- On écrit $z$ dans cette base :

$$z = x - \bar{x} = \sum_{i=1}^{D} b_k u_k$$

- On néglige les vecteurs propres $u_k$ pour $k > d$

$$z' = x' - \bar{x} = \sum_{i=1}^{d} b_k u_k$$

# Analyse en composantes principales, suite

Comment choisir $d$

$$\frac{\sum_{i=1}^{d} \lambda_i}{\sum_{i=1}^{D} \lambda_i} > .95$$

*"capturer 95% de l'inertie du nuage"*

On montre

$$Erreur < \frac{1}{2} \sum_{i=d+1}^{D} \lambda_i$$

Remarque

- La matrice de covariance des $x'$ est $\Delta = Diag(\lambda_1, \ldots, \lambda_d)$
- Les nouveaux attributs, combinaisons linéaires des anciens, sont difficiles à interpréter
- Il faut diagonaliser la matrice (complexité $\mathcal{O}(N^3)$)

# Préliminaires - bonnes pratiques

- L'approche dépend des unités de mesure choisies
- Toujours normaliser les données AVANT
- Une normalisation particuliere : centrer normer

$$(a_1, ..., a_N) \rightarrow (\frac{a_1 - \mu}{\sigma}, ..., \frac{a_N - \mu}{\sigma})$$

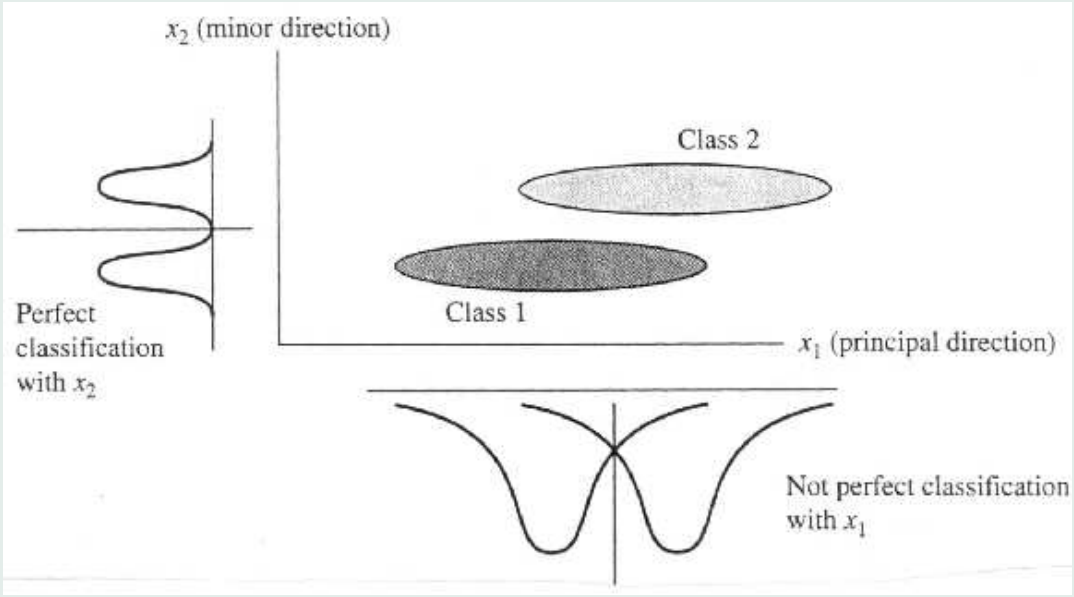
où $\mu$ est la moyenne, $\frac{1}{N}\sum_{i=1}^{N} a_i$
$\sigma$ est la variance, $\frac{1}{N}\sum_{i=1}^{N} a_i^2 - \mu^2$

- Cas des attributs nominaux : exploser en attributs binaires

couleur $\rightarrow$ { couleur_bleue, couleur_rouge, .. }

## • PCA and classification

- PCA is **not** always an optimal dimensionality-reduction procedure for classification purposes:




## • Other problems

# Analyse Sémantique Latente

Contexte : Fouille de textes

- $N$ : milliers de documents
- $D$ : dizaines de milliers de mots 100 000 mots en anglais

Matrice $M \in \mathbb{R}^{N \times D}$

- Problèmes : polysémie, synonymie
- Besoin de trouver les similarités entre mots, entre documents
- Analyse de co-occurrences inefficace
- Principe: si $m_1$ et $m_2$ sont trouvés dans le même contexte, ils sont peut-être similaires

# Décomposition en valeurs singulières

## Singular Value Decomposition

$$M = U \Delta V'$$

- U : changement de base pour les documents (→ “thème”)
- V : changement de base pour les mots (→ “unité linguistique”)

## Principe

$$\begin{array}{rcl} \Delta & \rightarrow & \Delta' \quad \textit{(annuler les val. propres les plus petites)} \\ M' & = & U \Delta' V' \end{array}$$

# Décomposition en valeurs singulières, 2

Fait apparaitre des similarités entre mots indétectables par

- cosinus

$$sim(m_i, m_j) = \frac{\sum_{doc} m_i(doc) \times m_j(doc)}{\sqrt{\sum_{doc} m_i(doc)^2} \times \sqrt{\sum_{doc} m_j(doc)^2}}$$

- mesure de Spearman

$$R(m_i, m_j) = \frac{\sum_{doc}(m_i(doc) - \bar{m}_i) \times (m_j(doc) - \bar{m}_j)}{\sqrt{(\sum_{doc}(m_i(doc) - \bar{m}_i)^2) \times (\sum_{doc}(m_j(doc) - \bar{m}_j)^2)}}$$

# Intuition

$$X = \begin{pmatrix} & m_1 & m_2 & m_3 & m_4 \\ d_1 & 0 & 1 & 1 & 1 \\ d_2 & 1 & 1 & 1 & 0 \end{pmatrix}$$

$m_1$ et $m_4$ ne sont pas “physiquement” ensemble dans les mêmes documents ; mais ils sont avec les mêmes mots ; “donc” ils sont un peu “voisins”...
Après SVD + Réduction,

$$X = \begin{pmatrix} & m_1 & m_2 & m_3 & m_4 \\ d_1 & \epsilon & 1 & 1 & 1 \\ d_2 & 1 & 1 & 1 & \epsilon \end{pmatrix}$$

# Discussion

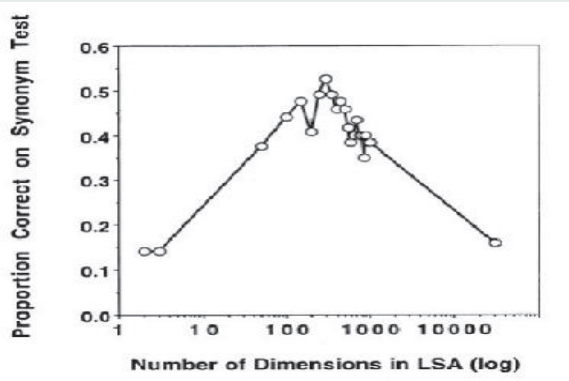
| Différences SVD | vs | ACP |
| --- | --- | --- |
| • Matrice initiale | | matrice de covariance |
| • qq centaines de valeurs propres | | très peu |
| • usage similarité | | usage visualisation, interprétation |

q

# Discussion, 2

## Une application

- Test de synonymie — TOEFL




## Déterminer le nb de dimensions/vp

- Expérimentalement...

## Quelques remarques

- et la négation ? — aucune importance (!)
- battu par: nb de hits sur le Web — P. Turney

# Quelques applications

- Educational Text Selection
  *Permet de sélectionner automatiquement des textes permettant d'accroître les connaissances de l'utilisateur.*
- Essay Scoring
  *Permet de noter la qualité d'une rédaction d'étudiant*
- Summary Scoring & Revision
  *Apprendre à l'utilisateur à faire un résumé*
- Cross Language Retrieval
  *permet de soumettre un texte dans une langue et d'obtenir un texte équivalent dans une autre langue*

Plus d'info `http://lsa.colorado.edu`

# Probabilistic LSA

T. Hoffman, UAI 99

Principe : supposons

- des "groupes" de documents $\equiv$ variables cachées $z$
- "peu" de var. cachées (comparé aux paires mots $\times$ documents)

$$P(documents|mots) = P(documents|z) \times P(z|mots)$$

Alors : Contraindre la décomposition

$$X_p = U_p S_p V_p^t$$

- $U_p : p(documents|z)$
- $S_p : p(z)$
- $V_p = p(mots|z)$

Comment ? Expectation Maximization

# Expectation Maximization

## Principe de l'apprentissage génératif

- Input : éléments $g_1, \ldots g_N$
- Output : modèles $\mathcal{G}_1, ..\mathcal{G}_n$

## Algorithmes itératifs

A chaque itération

Pour tout $g_i$ EXPECTATION

Trouver $\mathcal{G}_j$ tq

$$p(g_i|\mathcal{G}_j) = max\{p(g_i|\mathcal{G}_k),\ k = 1..n\}$$

Pour tout $\mathcal{G}_j$ MAXIMISATION

Soit $E_j = \{g_i$ affecté à $\mathcal{G}_j\}$

Mettre à jour $\mathcal{G}_j$ pour maximiser

$$\sum_{g \in E_j} p(g|\mathcal{G}_j)$$

# Support Vector Machines et Catégorisation de textes

## Représentation

- Sac de mots (avec stemming/radicaliseur)
- Feature selection (gain information) (?)

## Discussion sur ce qu'on cherche

- Distribution de fréquences des mots — Zipf law
- Les mots rares contiennent *aussi* des informations...
- Notion de concepts denses

## SVM adaptés

- espaces de grande dimension, peu de dimensions non pertinentes...
- documents "sparses", concepts séparables...

## Plus de détails

http://citeseer.ist.psu.edu/joachims97text.html

# Problèmes de représentation

- Sélection d'attributs
- Changements de représentation linéaires
- Changements de représentation non linéaires
- Une étude de cas

# Intuition

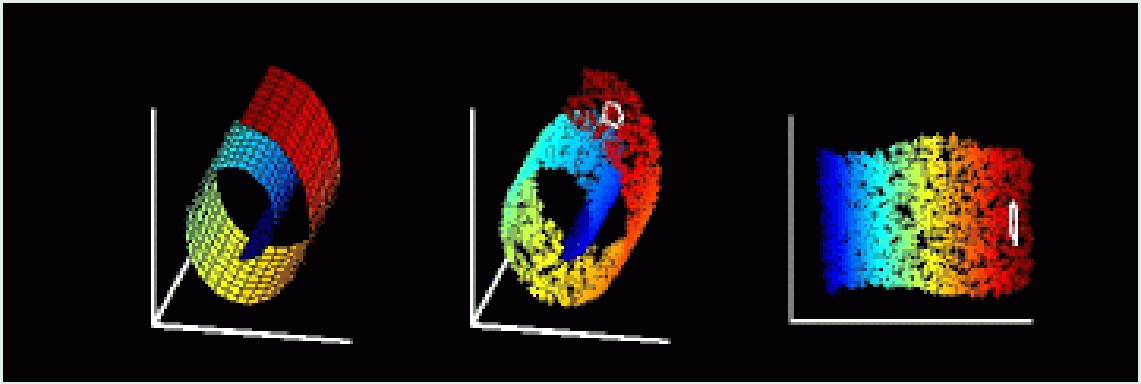
## Degrés de liberté

- Images : 4096 pixels; mais pas 4096 degrés de liberté...
- Robotique : (camera + IR) x temps ; mais info pas indépendantes...

## Objectif : accéder à la structure (peu de dimensions) des données

- Images, Robotique, Textes, Genes...

# The Swiss Roll



# Grandes lignes

- Conjecture : les exemples sont dans une variété

  *espace de dimension $d$ inclus dans un espace de dimension $D$*

- Aucune projection linéaire ne permet de découvrir que le Swiss Roll est dans $\mathbb{R}^2$
- Le but du jeu est
  - de déterminer que les données sont dans $\mathbb{R}^d$
  - de les projeter de manière “cohérente” sur $\mathbb{R}^d$

## Cohérence ?

- Préserver les relations locales $\equiv$ la structure
- Par exemple : préserver les distances

# Mise à l'échelle Multi-Dimensionnelle
# Multidimensional Scaling (MDS)

## Problème posé

- $N$ points de $\mathbb{R}^D$, et une matrice $X$ de similarités
- Les projeter dans $\mathbb{R}^d$ en "préservant" les similarités

$$\begin{aligned} x \in \mathbb{R}^D &\rightarrow \Phi(x) \in \mathbb{R}^d \\ sim(x, x') &\rightarrow sim(\Phi(x), \Phi(x')) \\ X(\in \mathbb{R}^{N \times N}) &\rightarrow X'(\in \mathbb{R}^{N \times N}) \end{aligned}$$

## Optimisation

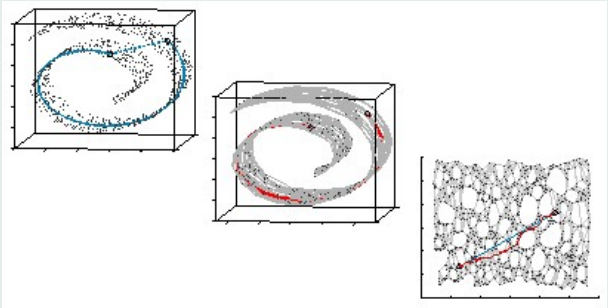
Trouver $\Phi$ minimisant $||X - X'||$

Rq : Si $\Phi$ est linéaire : Analyse en Composantes Principales...

# MDS, 2

Linéaire MDS : ne marche pas

- Préserve toutes les distances
- Mais seules les distances locales ont un sens...



# Projections non linéaires

APPROCHES

- Estimer les structures globales à partir des locales
  et chercher un plongement global — Isomap
- Se préoccuper uniquement des structures locales
  et laisser tomber les autres — LLE

Principe : localement c'est plat.

Comment identifier cette localité ?

- plus proches voisins
- boule de rayon $\varepsilon$

# Isomap

Tenenbaum, da Silva, Langford 2000

`http://isomap.stanford.edu`

Estimation des distances $d(x_i, x_j)$

- Connue si $x_i$ et $x_j$ sont proches
- Sinon, calculer le plus court chemin (programmation dynamique)

distance géodésique

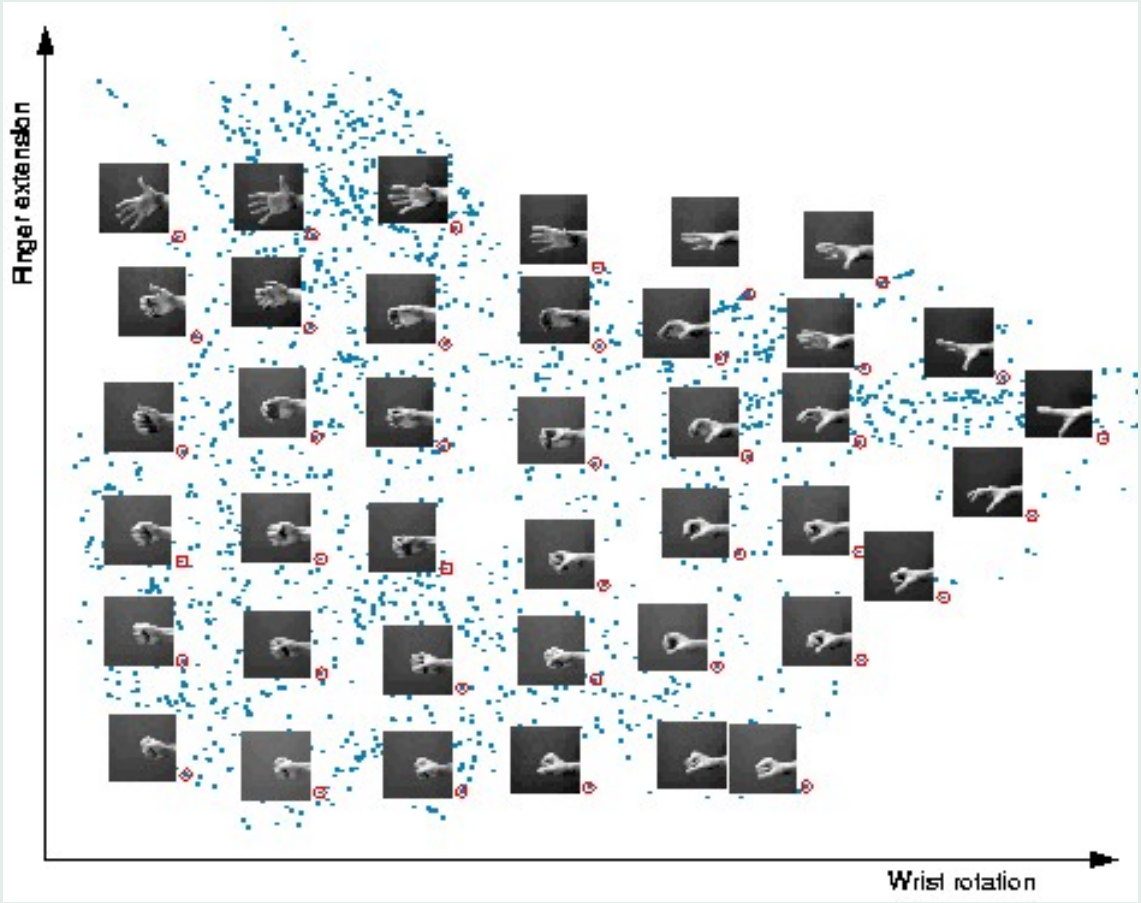
Condition necessaire

- Données viennent d'un ensemble convexe de $\mathbb{R}^d$
- Alors, la distance géodésique $\sim$ distance Euclidienne de $\mathbb{R}^d$

Se ramener au cas précédent : ACP

- On a la matrice $d(x_i, x_j)$, on estime $< x_i, x_j >$
- On se projette ds un espace de dimension $d$

# Isomap, 2

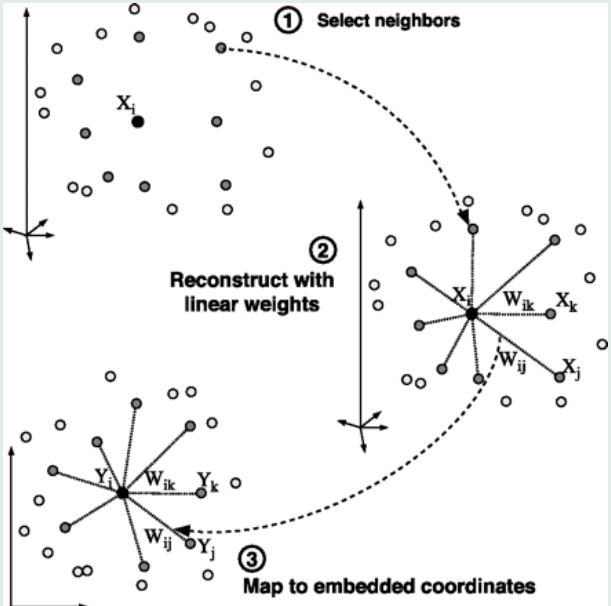


# Locally Linear Embedding

Roweiss and Saul, 2000

http://www.cs.toronto.edu/~roweis/lle/

- Cherche représentation locale en chaque point
- Exprimer chaque point en fonction des points voisins



# LLE, Algorithme

## Trouver les voisins

- Pour chaque point $x_i$, trouver ses plus proches voisins $\mathcal{N}(i)$

## Trouver les poids

- Exprimer le point dans la base donnée par ses voisins

$$x_i = \sum_{j \in \mathcal{N}(i)} w_{i,j} x_j$$

  On impose $\sum_{j \in \mathcal{N}(i)} w_{ij} = 1$

- Conséquence : invariance par translation, rotation, homothétie.
- Concrètement

$$C = (C_{j,k}), C_{j,k} = < x_j - x_i, x_k - x_i >$$

  Trouver $w_i$ tq $Cw_i = 1$ (attention ici $w_i$ et $1$ sont des vecteurs)

# LLE Algorithme, 2

## Reconstruction

- Matrice $W$ :

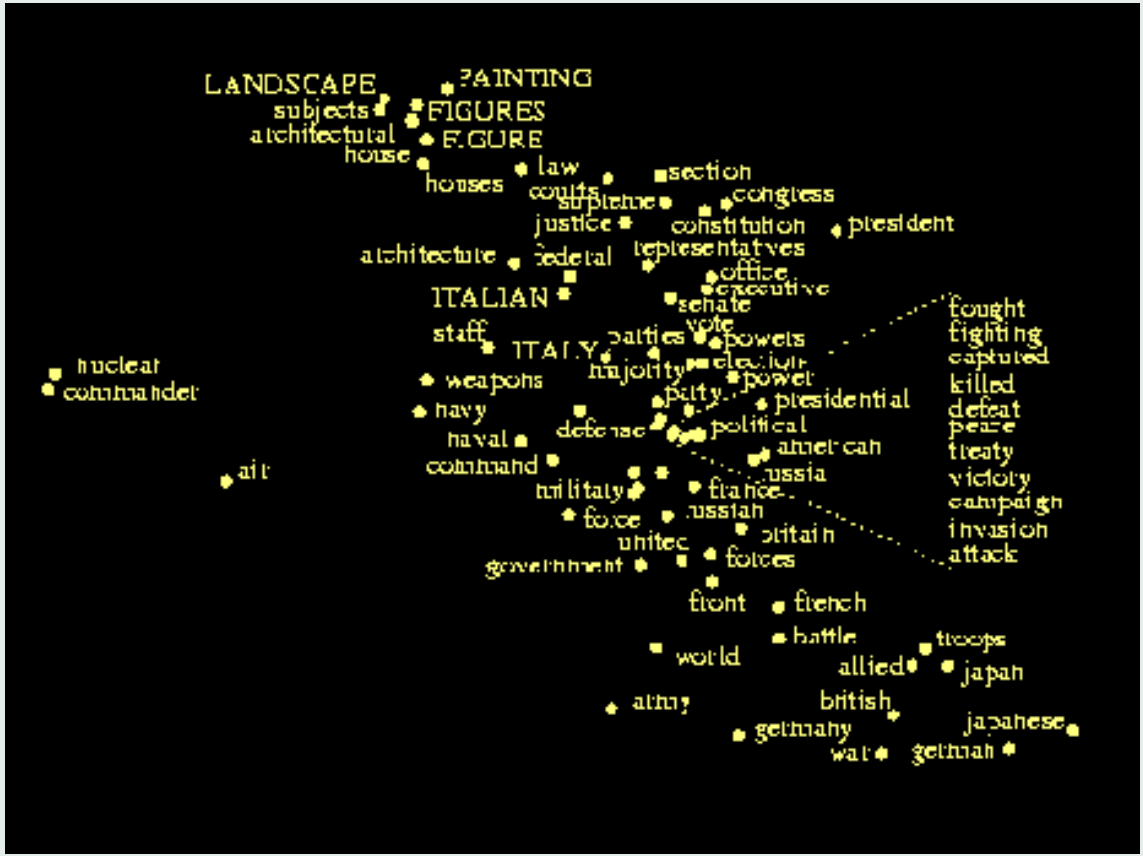
$$W_{i,j} = \begin{cases} 0 & \text{si } j \text{ non voisin de } i \\ \frac{w_{i,j}}{\sum_{k \in \mathcal{N}(i)} w_{i,k}} & \text{sinon} \end{cases}$$

- Matrice $M = (I - W)'(I - W)$

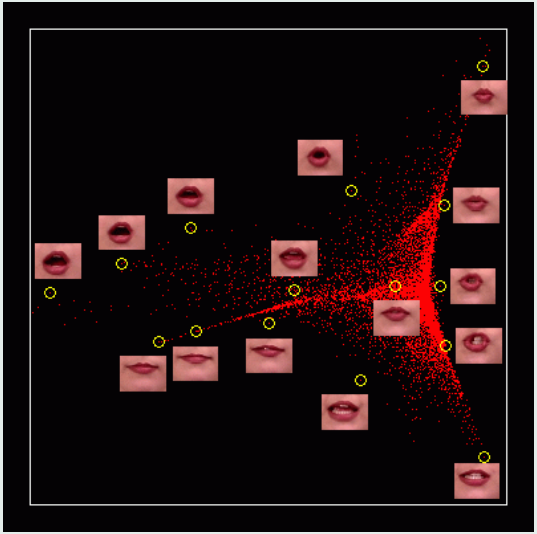
- Construire les vecteurs propres de $M$

- $Y'$ : prendre les $d$ vecteurs propres
  associés aux plus petites valeurs propres (sauf 0)

# Exemple, Textes



# Exemple, Images



LLE

# LLE : Discussion

## Global / Local

- Voisinages locaux : pour trouver les $W_{i,j}$
- Résolution globale : pour trouver les $Y$

## 1 Paramètre !

Le nombre de voisins considérés / Le rayon de la boule voisinage

## Force / Faiblesse

$+$ Optimum global

$+$ Les exemples initiaux peuvent être connus seulement par leur similarités.

$-$ Généralisation ?

# Problèmes de représentation

- Sélection d'attributs
- Changements de représentation linéaires
- Changements de représentation non linéaires
- Une étude de cas

# Une étude de cas

## Plan

- Un critère d'apprentissage combinatoire - optimisé par évolution artificielle

- Une application réelle : Maladies Cardio-Vasculaires
  aperçus inattendus sur le risque du tabac et de l'alcool...

- Un espace d'hypothèses plus intéressant
  non-linéaire mais permettant de scorer les attributs

- Utiliser la variabilité des solutions d'un algorithme stochastique (AGs) $\Longrightarrow$ Méthode d'ensemble.

# Critère ROC
# Receiver Operating Characteristics

Principe — traitement du signal, médecine

Soit $h(x)$ mesurant le risque du patient $x$. $h : X \mapsto \mathbb{R}$

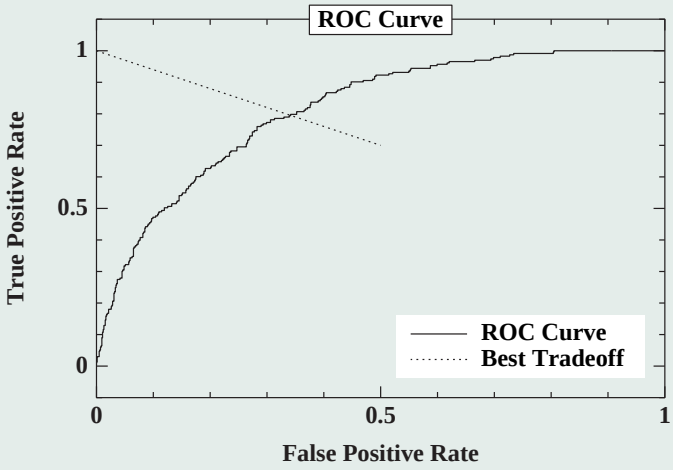
$$t \in \mathbb{R} \mapsto h_t(x) = \begin{cases} malade & si\ h(x) > t \\ OK & sinon \end{cases}$$

Pour $h_t$, définir:

- TP(t) : true positive rate, $Pr(h_t(x) = malade | x\ malade)$
- FP(t) : false positive rate, $Pr(h_t(x) = malade | x\ pas\ malade)$.

Tracer la courbe $(TP(t), FP(t), t \in \mathbb{R})$.

# ROC Curve


ROC Curve
True Positive Rate
False Positive Rate
0
0.5
1
0
0.5
1
ROC Curve
Best Tradeoff

# ROC Curve, 2

ROC depicts the trade-off False Positive / True Positive.

Standard: misclassification cost (Domingos, KDD 99)

$$\mathcal{F} = \#\text{ false positive} + c \times \#\text{ false negative}$$

In a multi-objective perspective, ROC = Pareto front.

Best solution: intersection of Pareto front with $\Delta(-c, -1)$

ROC: Extensively Used by Physicians

# ROC Curve, 3

Used to compare learners — Bradley 97

multi-objective-like
insensitive to imbalanced distributions
shows sensitivity to error cost.

Used as learning criterion: Area under the ROC curve

Given Dataset = $\{(\mathbf{x}_i, y_i), \mathbf{x}_i \in \mathbb{R}^d, y_i \in \{-1, 1\}\}$

Genotype: hypothesis $h \mapsto$ Phenotype: ordered examples

$$+\,+\,+\,-\,+\,+\,-\,+\,+\,+\,+\,-\,-\,-\,+\,-\,-\,-\,+\,-\,-\,-\,-\,-\,-\,-\,-\,-$$

$\mathcal{F}(h)$ = sum of ranks of positive examples. — AUC : to be minimized

# Area Under the ROC Curve

Previous

| | |
|---|---|
| EP-based NN optimization | Fogel+, 1998 |
| GA-based linear optimization | Mozer+, 2001 |
| greedy Decision Tree optimization | Ferri-Flach, 2002 |

ROGER: ROC-based Genetic Evolutionary LearneR

$(\mu + \lambda)$-ES (Evolution Strategy)

Parameters

| | | |
|---|---|---|
| population size | # parents $\mu$ | 10 |
| | # offspring $\lambda$ | 50 |
| max nb evaluations | | 10,000 |
| crossover | uniform | rate .6 |
| mutation | self-adaptive | rate 1 |

# Experiments

## Reference results: Support Vector Machines (SVMTorch)

Search space: linear classifiers : $\mathbb{R}^d$

## Datasets from Irvine repository

| | #att | #weight | #Train | #Test |
|---|---|---|---|---|
| Br. Canc. | 9 | 42 | 189 | 97 |
| Crx | 15 | 47 | 70 | 620 |
| German | 25 | 25 | 100 | 900 |
| Promoters | 59 | 229 | 70 | 36 |
| Satimage | 36 | 36 | 139 | 1237 |
| Vehicle | 18 | 18 | 125 | 291 |
| Votes | 16 | 32 | 287 | 148 |
| Waveform | 22 | 22 | 211 | 3321 |

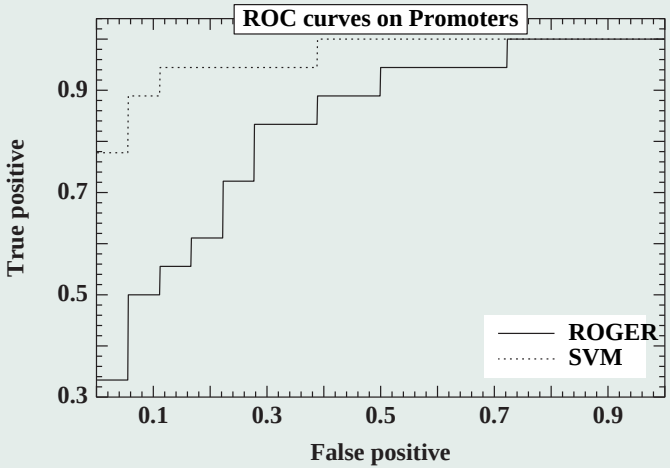
| ROGER | | SVMTorch | |
|---|---|---|---|
| AUC | time | AUC | time |
| $.674 \pm .05$ | 7” | $.672 \pm .05$ | 1” |
| $.816 \pm .06$ | 7” | $.839 \pm .04$ | 886” |
| $.712 \pm .03$ | 6” | $.690 \pm .02$ | 96” |
| $.863 \pm .07$ | 2” | $.974 \pm .02$ | $< 1$” |
| $.918 \pm .01$ | 4” | $.876 \pm .02$ | 14” |
| $.994 \pm .005$ | 1” | $.993 \pm .007$ | $< 1$” |
| $.993 \pm .004$ | 7” | $.989 \pm .005$ | $> 1{,}000$ |
| $.971 \pm .004$ | 4” | $.963 \pm .008$ | 2” |

## Experimental setting

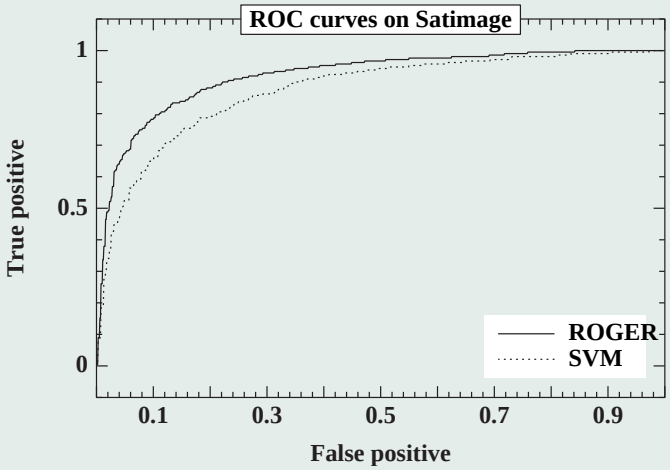
10 train/test splits

For each split, 1 SVMTorch run, 21 ROGER runs (take median)

# ROC Curve, Promoters



# ROC Curve, Satimage



# Partial conclusions - ML aspects

PROS

- Competitive wrt state of art, SVM.
- Affordable cost, fitness computation $nlog(n)$
- Learning stability wrt imbalanced distribution, error cost

CONS

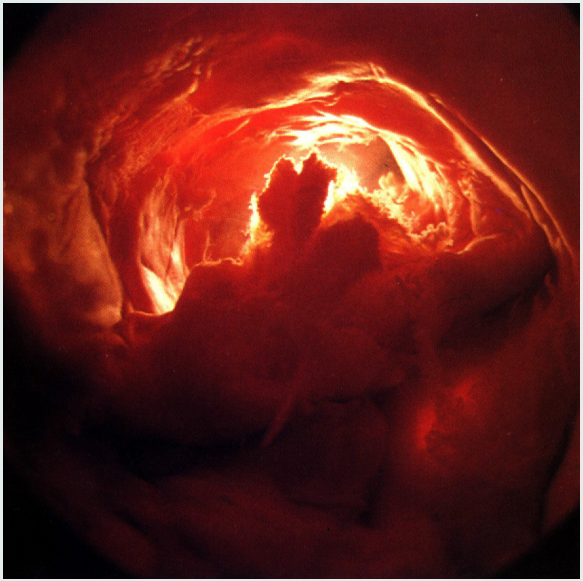
- Does not scale up well with # attributes

# Une autre approche : ROGER

Plan

- Un critère d'apprentissage combinatoire et son optimisation
- Une application réelle : Maladies Cardio-Vasculaires
  aperçus inattendus sur le risque du tabac et de l'alcool...
- Un espace d'hypothèses plus intéressant
  non-linéaire mais permettant de scorer les attributs
- Utiliser la variabilité des solutions d'un algorithme stochastique (AGs) $\Longrightarrow$ Méthode d'ensemble.

# A Medical Data Mining Application



# Understanding Cardio-vascular Diseases

PKDD 2002-2003 Challenge

- Study Atherosclerosis Risk Factors First death cause in Western countries

Data

- ENTRY database (medical cliché, 1419 men, 219 attributes, 1976)
- CONTROL database (longitudinal study of a sample, 1976-1996)

First goal

- Given the medical cliché at $t_0$, predict health state at $t_0 + 20$.

# Some limitations of the data

Initial description :

very detailed — diseases 1st..4th brother, 1st..4th sister

...not usable... — 4th sister INF MYOCARD....

What cannot be learned : sufficient conditions for diseases

(1) If father or mother diabetic
(2) And high stress
(3) And does not laugh once a day
Then disease

... (Condition 3 likely missing in hospital db)

→ find at best necessary conditions

# Changing the problem

Initial goal: classification — predefined classes

$$\text{Patient} \mapsto \{ \text{normal, at risk, pathological} \}$$

Alternative: ranking — Mr $X$ is more at risk than Ms $Y$

$$(\text{Patient} \times \text{Patient}) \mapsto \{true, false\}$$

concept is smoother (frontier between normal and pathological)
more flexible (medical / economical concerns)

Proposed: "underconstrained regression" — Risk(Mr $X$) is 3.7

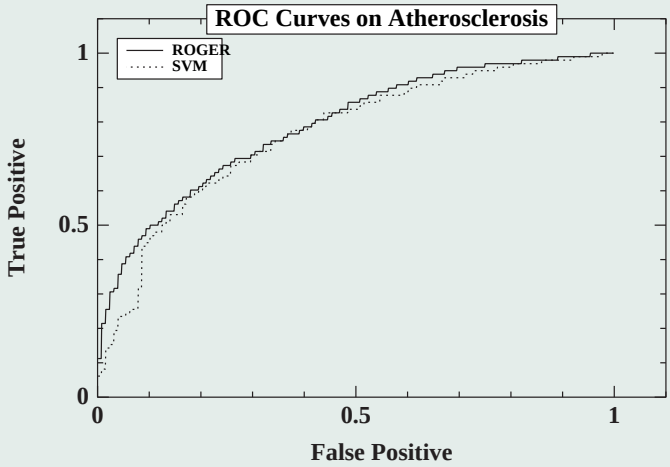
$$\text{Patient} \mapsto \mathbf{R}$$

# Atherosclerosis

Experimental setting: 2/3 training, 1/3 test $\times$ 10

On each training set, 21 independent runs

Display the median ROC curve



# Influence Analysis - The tobacco factor

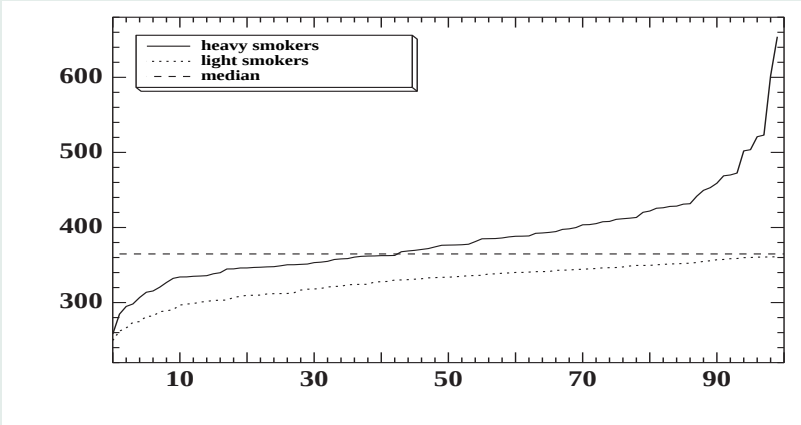
## Procedure

A = { 100 non smoking individuals }

B = { 100 most smoking individuals }

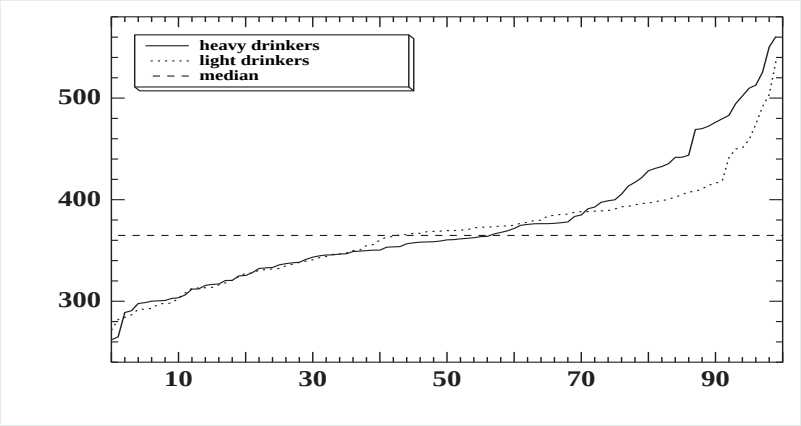
Sort A and B by increasing value of the risk

Plot (i, risk(i))

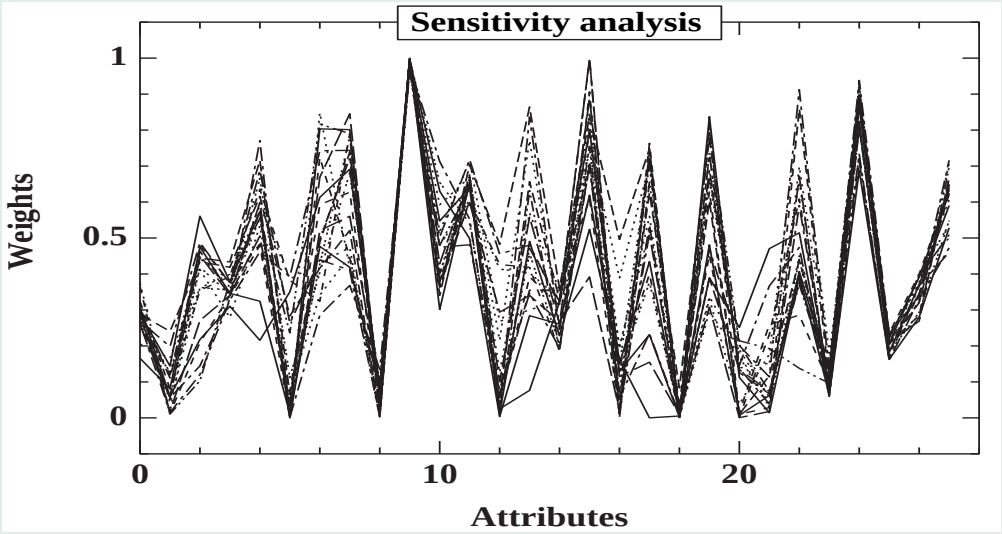


# Influence Analysis - The alcool factor

A = { 100 light drinkers }
B = { 100 heavy drinkers }
Sort A and B by increasing value of the risk
Plot (i, risk(i))



# Sensitivity Analysis - For free




21 runs, 21 solutions, 21 curves: $(i, weight(attribute_i))$

# Conclusions - Perspectives

ICDM 2003, AE 2003

Present

- Good predictive performances
- Affordable complexity
- UNDERSTANDABLE RESULTS

Using Vision to Think, Card et al. 2001

Next

- Extend to kernel spaces
- Use for constructive induction

# ROGER, suite

Plan

- Un critère d'apprentissage combinatoire - optimisé par évolution artificielle

- Une application réelle : Maladies Cardio-Vasculaires
  aperçus inattendus sur le risque du tabac et de l'alcool...

- Un espace d'hypothèses plus intéressant
  non-linéaire mais permettant de scorer les attributs

- Utiliser la variabilité des solutions d'un algorithme stochastique (AGs) $\Longrightarrow$ Méthode d'ensemble.

# Un espace d'hypothèses plus intéressant

Espace linéaire

$$h(x) = \sum_i w_i a_i(x) \qquad h \equiv w \in \mathbb{R}^d$$

Espace non linéaire pauvre

$$h(x) = \sum_i w_i |a_i(x) - c_i| \qquad h \equiv (w, c) \in \mathbb{R}^{2d}$$

Intérêt

hypothèses non linéaires
espace de recherche linéaire $\mathbb{R}^{2d}$
$score(atribut\ a_i) = w_i$.

# ROGER, suite

Plan

- Un critère d'apprentissage combinatoire - optimisé par évolution artificielle
- Une application réelle : Maladies Cardio-Vasculaires
  aperçus inattendus sur le risque du tabac et de l'alcool...
- Un espace d'hypothèses plus intéressant
  non-linéaire mais permettant de scorer les attributs
- Utiliser la variabilité des solutions d'un algorithme stochastique (AGs) $\Longrightarrow$ Méthode d'ensemble.

# Evolution artificielle et méthodes d'ensemble

## Méthodes d'ensemble, rappel

$\mathcal{H}$: espace d'hypothèses

- Erreur = biais + variance
- Biais : le mieux que l'on puisse faire sur l'espace d'hypothèses : $Err(h^*) = Argmin\{Err(h),\ h \in \mathcal{H}\}$
- Variance : on n'apprend pas $h^*$, hélas.
  On apprend $\hat{h}_n$, dépendant des $n$ exemples d'apprentissage.

# Méthodes d'ensemble, 2

Principe : réduire la variance

- Apprendre $h_1, ..h_T$, décorrélées
- Tel que la probabilité d'erreur soit "raisonnable" weak learning

$$Pr(h_i(x) = y) = \frac{1}{2} + \eta$$

- Le vote, ou la combinaison linéaire des $h_t$ fait mieux que le meilleur $h_t$.

Idée de preuve : Inégalité de Hoeffding

- Soit $V_i$ des variables booléennes indépendantes de probabilité $p$.
- Soit $Y_T$ la somme des variables $V_1, ..V_T$

$$Pr(|Y_T - T \times p| > \epsilon) < exp^{-2\epsilon T^2}$$

# Evolution artificielle et méthodes d'ensemble, 2

## Algorithme stochastique

- Chaque run $\longrightarrow$ une hypothèse indépendante.
- Chaque hypothèse $\longrightarrow$ un ordre sur les attributs.

## Ordre faible

- Soit $\{a_1, ..a_N\}$ l'ordre parfait
- $h_t$ : induit un ordre $<_t$ sur les attributs
- Supposons un ordre faible :

$$P(a_i <_t a_j | i < j) > \frac{1}{2} + \eta$$

## Agrégation

- On définit $<_*$ comme :

$$(a_i <_* a_j) \Longleftrightarrow |\{t / a_i <_t j\}| > \frac{T}{2}$$

# Evolution artificielle et méthodes d'ensemble, 3

L'ordre agrégé est bien un ordre

$$Pr(i <_* k | i <_* j \text{ et } j <_* k) \rightarrow 1 \text{ quand } T \rightarrow \infty$$

.. et tend vers l'ordre parfait

- Soit $O_*(i) = |\{j / i <_* j\}|$ alors

$$Pr(|O_*(i) - i| > \tau) \rightarrow 0$$

# Validation

## Difficulté

• Validation d'un ens d'attributs ==
qualité de la meilleure hypothèse fondée sur ces attributs

$\Rightarrow$ Pas moyen de tester une méthode de sélection en soi.

## Approche

• Pbs artificiels

• On connait la solution ; est-ce qu'on la retrouve ?

• Permet étude de "Lésions" : bruit, passage à l'échelle % nb exemples, nb attributs...

# Problèmes artificiels

## Paramètres d'ordre

- Nb attributs $d = 100, 200, 500$
- Nombre d'exemples $n = d/2, d, 2d$
- Nombre d'attributs pertinents $r = d/20, d/10, d/5$
- Type de concept à apprendre : Linéaire ou Non.
- Bruit de classe $e = 0, 5, 10\%$
- Bruit d'attribut $\sigma = 0, 0.05, 0.1$

# Construire un pb artificiel $(d, n, r, l, e, \sigma)$

Se donner les attributs pertinents : $\{1, 2, ...r\}$ parmi $\{1, ..d\}$

Pour chaque exemple $x_j$

- Pour $i = 1..d$, tirer $a_i(x_j)$ uniformément ds $[0, 1]$

Construction de $y_j$

- Cas linéaire :

$$y_j = (\sum_{i=1}^{r} a_i(x_j) > \frac{r}{2})$$

- Cas non-linéaire :

$$y_j = (\sum_{i=1}^{r} |a_i(x_j) - .5| < \frac{r}{12})$$

# Pbs artificiels, suite

Perturbation

- $y_j = -y_j$ avec probabilité $e$
- $a_i(x_j) += \mathcal{N}(0, \sigma)$

Méthodologie expérimentale

Pour chaque $(d, n, r, l, e, \sigma)$, construire 20 problèmes

  Pour chaque problème, apprendre 20 hypothèses — 20 runs

  Agréger les poids des 20 hypothèses

  Comparer l'ordre obtenu à l'ordre désiré

Moyenner l'erreur sur les 20 problèmes

# Algorithme de référence

Stoppiglia et al., JMLR 2003

## Score d'un attribut

- Cosinus : score(a) = $\sum_i a(x_i).y_i$

## Projection itérative de Gauss

- Trouver le meilleur attribut $a$
- Projeter les données et le concept sur l'espace ortogonal à $a$

# Mesure de performance

## Qualité pour une sélection itérative

$p_b$ probabilité du top d'être pertinent

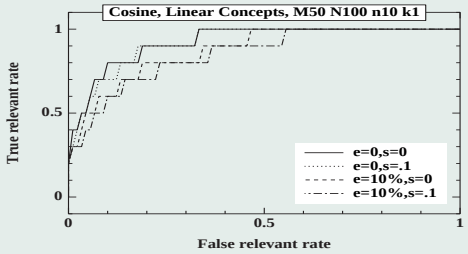
## Qualité pour une élimination itérative

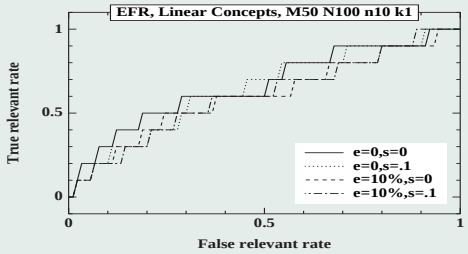
$p_w$ pire rang d'un attribut pertinent

## Compromis

Taux de vrais pertinents lem vs taux de faux pertinents : AUC.

# Comparaison sur des concepts linéaires



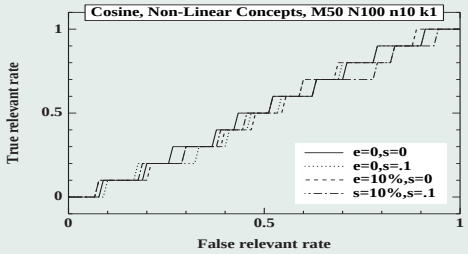



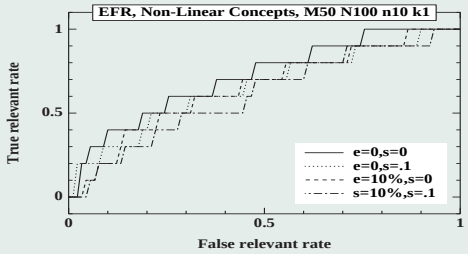

Stoppiglia ROGER

Stoppiglia >> ROGER >> Random

Ici : $d = 100$, $n = d/2$, $r = d/10$

# Comparaison sur des concepts non linéaires



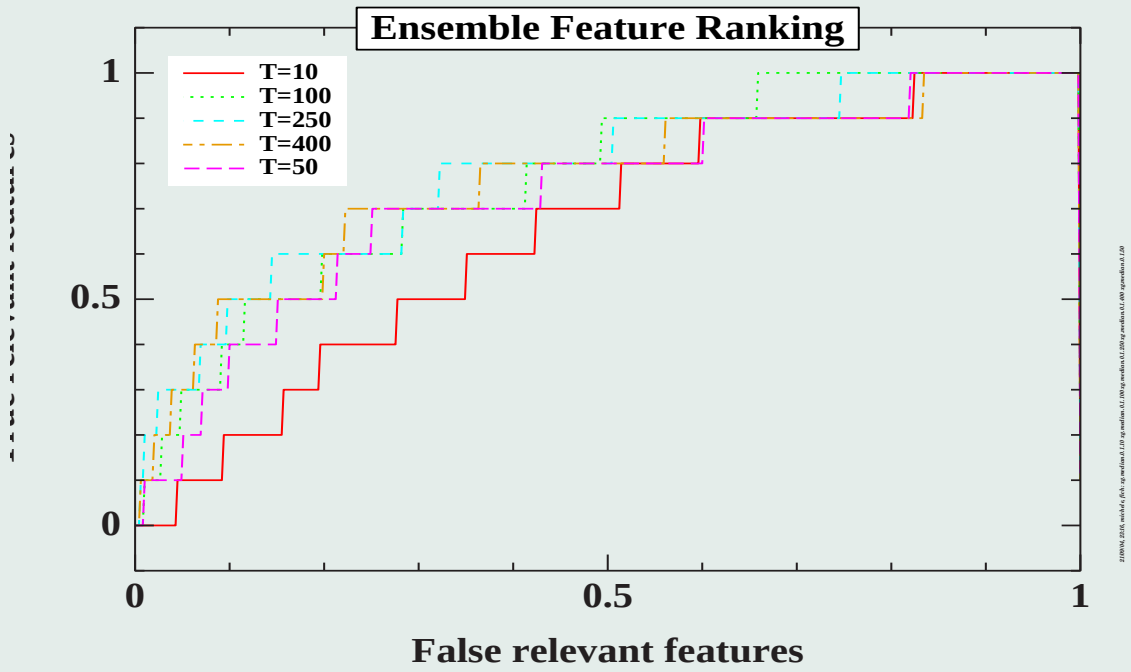




Stoppiglia ROGER

ROGER >> Stoppiglia = Random

Ici : $d = 100$, $n = d/2$, $r = d/10$

# Ensemble Feature Ranking




Quand on augmente $T$ : de 10 à 400.

# Conclusion

## Contributions

- weak ranking $\Rightarrow$ strong ranking
- EC enables ensemble methods “for free”
- a principled framework for evaluating feature ranking/selection

## Limits

- Only conjunctive concepts.
- And ?

## Next

- Multi-modal evolution / several hypotheses in a population.