Master 2 Recherche
Apprentissage Statistique, Optimisation et Applications

Michèle Sebag – Balazs Kégl – Anne Auger

TAO: Theme Apprentissage & Optimisation

http://tao.lri.fr/tiki-index.php

24 novembre 2010

Apprendre et Optimiser, quel rapport ?

Apprentissage

- Input: des points des exemples
- Output: une fonction

Optimisation

- Input: une fonction fonction objectif, ou fitness
- Output: un (ou plusieurs) points: les optima de la fonction

Apprendre et Optimiser, 2

Applications: d'abord apprendre, puis optimiser

1. Rassembler des exemples de pannes
2. Généraliser: Apprendre dans quel contexte se produisent les pannes
3. Régler le système pour minimiser le taux de panne

Algorithme: apprendre $\equiv$ optimiser

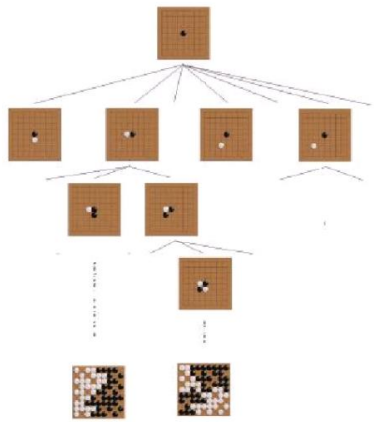
- Apprentissage: trouver "la meilleure" fonction.
- Optimisation:
  - Si la fonction est simple, algorithmes directs
  - Sinon, trouver une fonction "assez proche" de l'objectif pour guider la recherche.

Plan du module

- Introduction — ce cours
- Optimisation — Anne Auger
  - Optimisation continue; stratégies d"evolution (ES)
  - Algorithme CMA-ES
  - Bornes ($\equiv$ complexité)
- Apprentissage — Balazs Kégl
  - Machines à vecteurs support
  - Apprentissage d'ensembles et Boosting
  - Modèles génératifs
- Applications
  - Jouer au Go
  - Apprentissage Actif, Sélection d'attributs, Navigation robotique,...

Pour quoi faire

- Prédiction
  pannes, maladies, achats, préférences,...
- Compréhension, Modélisation
  facteurs de risque, analyse de survie
  e-Science



- Interaction
  Jeux ; “Super-Google” ;
  Brain Computer Interface
- Optimisation–Conception
  décision et conception optimale: des jeux aux politiques d’énergie

Quelques bonnes adresses

- Où sont les cours :
  http://tao.lri.fr/tiki-index.php?page=Courses
  http://www.limsi.fr/Individu/allauzen/wiki/index.php/TSI09
- Les cours (transparents) d'Andrew Moore
  http://www.autonlab.org/tutorials/index.html
- Les cours (videos) de PASCAL
  http://videolectures.net/pascal/
- Les tutoriels de NIPS Neuro Information Processing Systems
  http://nips.cc/Conferences/2006/Media/
- Des questions intéressantes
  http://hunch.net/

Plan de ce cours

- Où allons-nous, d'où venons-nous: apprentissage supervisé
  - Définitions
  - Objectif
  - Validation: théorie; méthodologie
- Représentation du problème
  - Sélection d'attributs
  - Changements de représentation linéaires
  - Changements de représentation non linéaires
  - Propositionalisation
  - Une étude de cas

Apprentissage supervisé

Contexte

$$\text{Monde} \longrightarrow \text{instance } \mathbf{x}_i \longrightarrow \begin{array}{c}\text{Oracle}\\ \downarrow \\ y_i\end{array}$$

Input: Base d'apprentissage $\mathcal{E} = \{(\mathbf{x}_i, y_i), i = 1 \ldots n, \mathbf{x}_i \in \mathcal{X}, y_i \in \mathcal{Y}$

Output: Hypothèse $h : \mathcal{X} \mapsto \mathcal{Y}$

Critère: Qualité de h

Vocabulaire

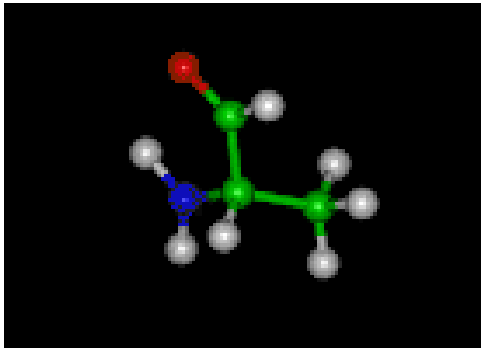
Exemple

- ligne : exemple/ cas/individus/transactions
- colonne : attribut/ feature/variables/ items
- apprentissage supervisé : attribut classe

age	employme	education	edun	marital	...	job	relation	race	gender	hour	country	wealth
					...							
39	State_gov	Bachelors	13	Never_mar	...	Adm_cleric	Not_in_fam	White	Male	40	United_Sta	poor
51	Self_emp_	Bachelors	13	Married	...	Exec_man	Husband	White	Male	13	United_Sta	poor
39	Private	HS_grad	9	Divorced	...	Handlers_c	Not_in_fam	White	Male	40	United_Sta	poor
54	Private	11th	7	Married	...	Handlers_c	Husband	Black	Male	40	United_Sta	poor
28	Private	Bachelors	13	Married	...	Prof_speci	Wife	Black	Female	40	Cuba	poor
38	Private	Masters	14	Married	...	Exec_man	Wife	White	Female	40	United_Sta	poor
50	Private	9th	5	Married_sp	...	Other_ser	Not_in_fam	Black	Female	16	Jamaica	poor
52	Self_emp_	HS_grad	9	Married	...	Exec_man	Husband	White	Male	45	United_Sta	rich
31	Private	Masters	14	Never_mar	...	Prof_speci	Not_in_fam	White	Female	50	United_Sta	rich
42	Private	Bachelors	13	Married	...	Exec_man	Husband	White	Male	40	United_Sta	rich
37	Private	Some_coll	10	Married	...	Exec_man	Husband	Black	Male	80	United_Sta	rich
30	State_gov	Bachelors	13	Married	...	Prof_speci	Husband	Asian	Male	40	India	rich
24	Private	Bachelors	13	Never_mar	...	Adm_cleric	Own_child	White	Female	30	United_Sta	poor
33	Private	Assoc_acd	12	Never_mar	...	Sales	Not_in_fam	Black	Male	50	United_Sta	poor
41	Private	Assoc_voc	11	Married	...	Craft_repai	Husband	Asian	Male	40	*MissingV	rich
34	Private	7th_8th	4	Married	...	Transport_	Husband	Amer_Indi	Male	45	Mexico	poor
26	Self_emp_	HS_grad	9	Never_mar	...	Farming_fi	Own_child	White	Male	35	United_Sta	poor
33	Private	HS_grad	9	Never_mar	...	Machine_c	Unmarried	White	Male	40	United_Sta	poor
38	Private	11th	7	Married	...	Sales	Husband	White	Male	50	United_Sta	poor
44	Self_emp_	Masters	14	Divorced	...	Exec_man	Unmarried	White	Female	45	United_Sta	rich
41	Private	Doctorate	16	Married	...	Prof_speci	Husband	White	Male	60	United_Sta	rich
:	:	:	:	:	:	:	:	:	:	:	:	:

Espace des instances $\mathcal{X}$

- Propositionnel : $\mathcal{X} \equiv \mathbb{R}^d$
- Relationnel : ex. chimie.



molécule alanine

Apprentissage supervisé, 2

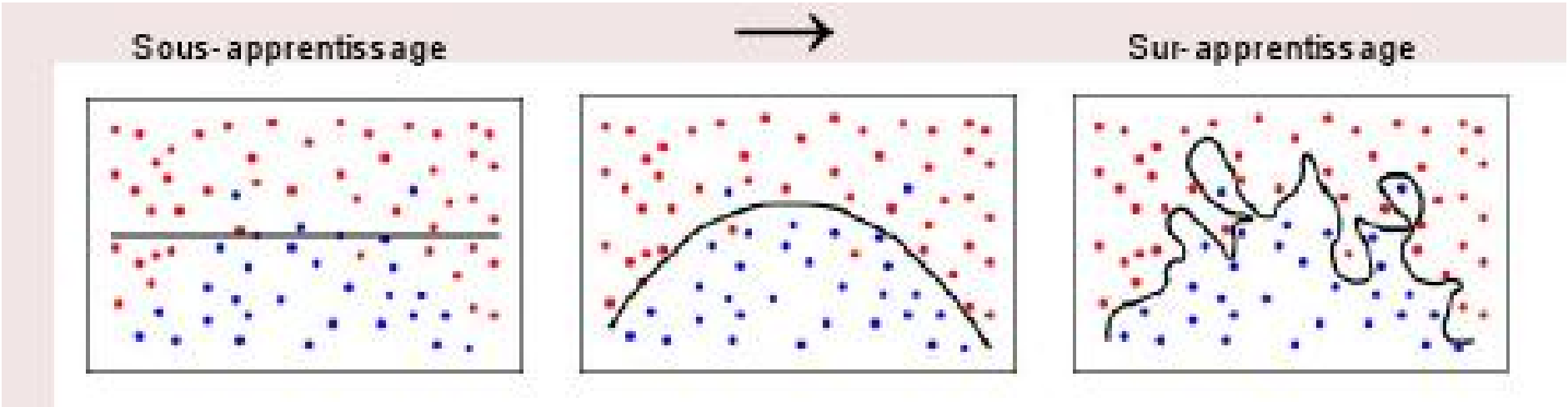
Définitions

- $\mathcal{E} = \{(\mathbf{x}_i, y_i), \mathbf{x}_i \in \mathcal{X}, y_i \in \mathcal{Y}, i = 1 \dots n\}$
  - Classification : $\mathcal{Y}$ fini (ex, nom de maladie)
  - Régression : $\mathcal{Y} \subseteq \mathbb{R}$ (ex, durée de survie)
- Espace des hypothèses $\mathcal{H} : \mathcal{X} \to \mathcal{Y}$

Tâches

- Choisir $\mathcal{H}$ — sélection de modèle
- Evaluer $h \in \mathcal{H}$ — $score(h)$
- Choisir h^* dans $\mathcal{H}$ — $argmax\ score(h)$

Choisir l'espace d'hypothèses



Remarque: le but n'est pas de ne faire aucune erreur sur l'ensemble d'apprentissage...

Quel est l'objectif ? Qualité de h

Etre bon sur les futurs exemples

Condition nécessaire:

qu'ils ressemblent aux exemples d'entrainement

"même distribution"

Prendre en compte le coût des erreurs $\ell(h(x), y) \geq 0$

toutes les erreurs ne sont pas aussi graves...

Validation: 1. Théorie
Apprentissage Statistique

Minimiser l'espérance du coût de l'erreur

$$\text{Minimize } E[\ell(h(x), y)]$$

Validation: 1. Théorie
Apprentissage Statistique

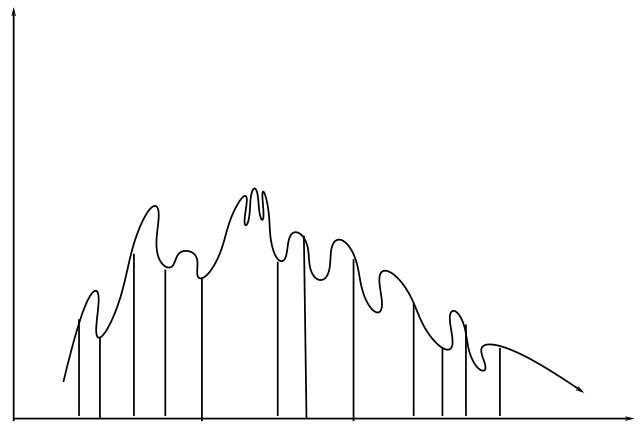
Minimiser l'espérance du coût de l'erreur

$$\text{Minimize } E[\ell(h(x), y)]$$

Principe

Si h "se comporte bien" sur $\mathcal{E}$, et h est "assez régulier", h se comporte bien en espérance.

$$E[F] \leq \frac{\sum_{i=1}^{N} F(x_i)}{n} + c(F, n)$$



Classification, Problème posé

INPUT $\sim P(x, y)$

$$\mathcal{E} = \{(x_i, y_i), x_i \in \mathcal{X}, y_i \in \{0, 1\}, i = 1 \dots n\}$$

ESPACE des HYPOTHESES

$$\mathcal{H} \quad h : \mathcal{X} \mapsto \{0, 1\}$$

FONCTION de PERTE

$$\ell : \mathcal{Y} \times \mathcal{Y} \mapsto \mathbb{R}$$

OUTPUT

$$h^* = arg\ max\{score(h), h \in \mathcal{H}\}$$

Classification, critères

Erreur en généralisation

$$Err(h) = E[\ell(y, h(x))] = \int \ell(y, h(x)) dP(x, y)$$

Erreur empirique

$$Err_e(h) = \frac{1}{n} \sum_{i=1}^{n} \ell(y_i, h(x_i))$$

Borne

risque structurel

$$Err(h) < Err_e(h) + \mathcal{F}(n, d(\mathcal{H}))$$

$d(\mathcal{H})$ = dimension de VC de $\mathcal{H}$, voir après

Dimension de Vapnik Cervonenkis

Principe

Soit $\mathcal{H}$ un ensemble d'hypothèses: $\mathcal{X} \mapsto \{0, 1\}$

Soit $x_1, \ldots, x_n$ un ensemble de points de $\mathcal{X}$.

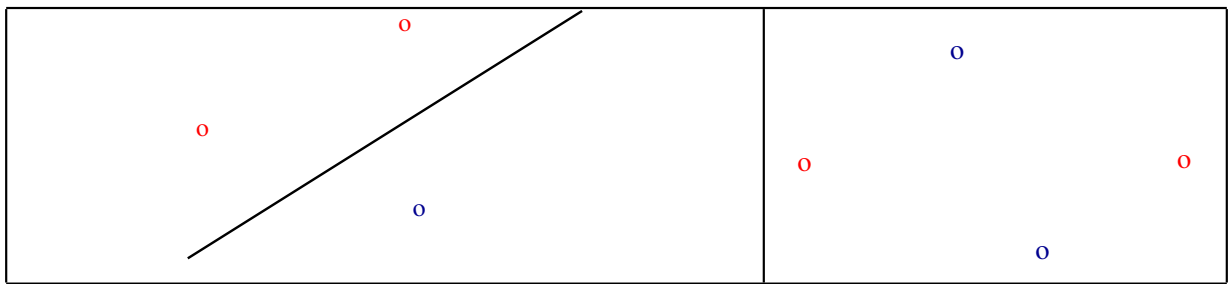
Si, $\forall (y_i)_{i=1}^n \in \{0,1\}^n, \exists h \in \mathcal{H} / h(x_i) = y_i$,

$\mathcal{H}$ pulvérise $\{x_1, \ldots, x_n\}$

Exemple: $\mathcal{X} = \mathbb{R}^p$

$d(\text{hyperplans de } \mathbb{R}^p) = p + 1$

Rq: si $\mathcal{H}$ pulvérise $\mathcal{E}$, $\mathcal{E}$ ne nous apprend rien...



3 pts pulvérisés par une droite	4 points, non

Définition

$$d(\mathcal{H}) = max\{n / \exists (x_1 \ldots, x_n\} \text{ pulvérisé par } \mathcal{H}\}$$

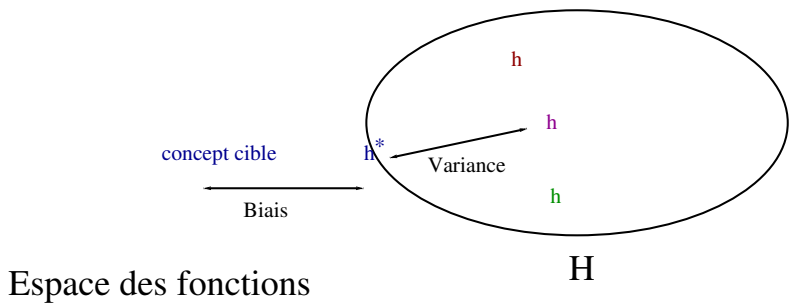
Classification, termes d'erreur

Biais

Biais ($\mathcal{H}$): erreur de la meilleure hypothèse h^* de $\mathcal{H}$

Variance

Variance de h_n en fonction de $\mathcal{E}$



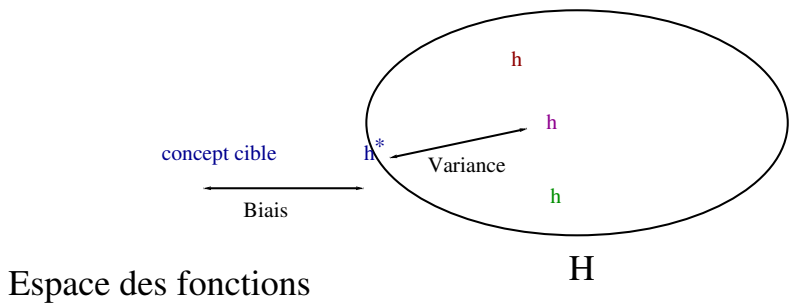
Classification, termes d'erreur

Biais

Biais ($\mathcal{H}$): erreur de la meilleure hypothèse h^* de $\mathcal{H}$

Variance

Variance de h_n en fonction de $\mathcal{E}$



Erreur d'optimisation

négligeable à la petite échelle
prend le dessus à la grande échelle (Google)

Validation: 2. Méthodologie

Validation croisée

1. Découper $\mathcal{E}$ en K sous-ensembles $\mathcal{E}_i$
2. $\mathcal{E}^i = \mathcal{E} \setminus \mathcal{E}_i$
3. Apprendre h_i à partir de $\mathcal{E}^i$
4. $\text{Err}(h_i) =$ pourcentage d'erreurs sur $\mathcal{E}_i$
5. Qualité (performance): moyenne des $\text{Err}(h_i)$.

Attention

Distribution des classes dans $\mathcal{E}_i \sim$ distribution des classes ds $\mathcal{E}$
(découpage stratifié)

Plan de ce cours

- Où allons-nous, d'où venons-nous: apprentissage supervisé
  - Définitions
  - Objectif
  - Validation: théorie; méthodologie
- Représentation du problème
  - Sélection d'attributs
  - Changements de représentation linéaires
  - Changements de représentation non linéaires
  - Propositionalisation
  - Une étude de cas

Au début sont les données...

	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

Motivations : Trouver et élaguer des descripteurs

Avant l'apprentissage : décrire les données.

- Une description trop pauvre $\Rightarrow$ on ne peut rien faire
- Une description trop riche $\Rightarrow$ on doit élaguer 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.

Feature Selection, Position du problème

Contexte

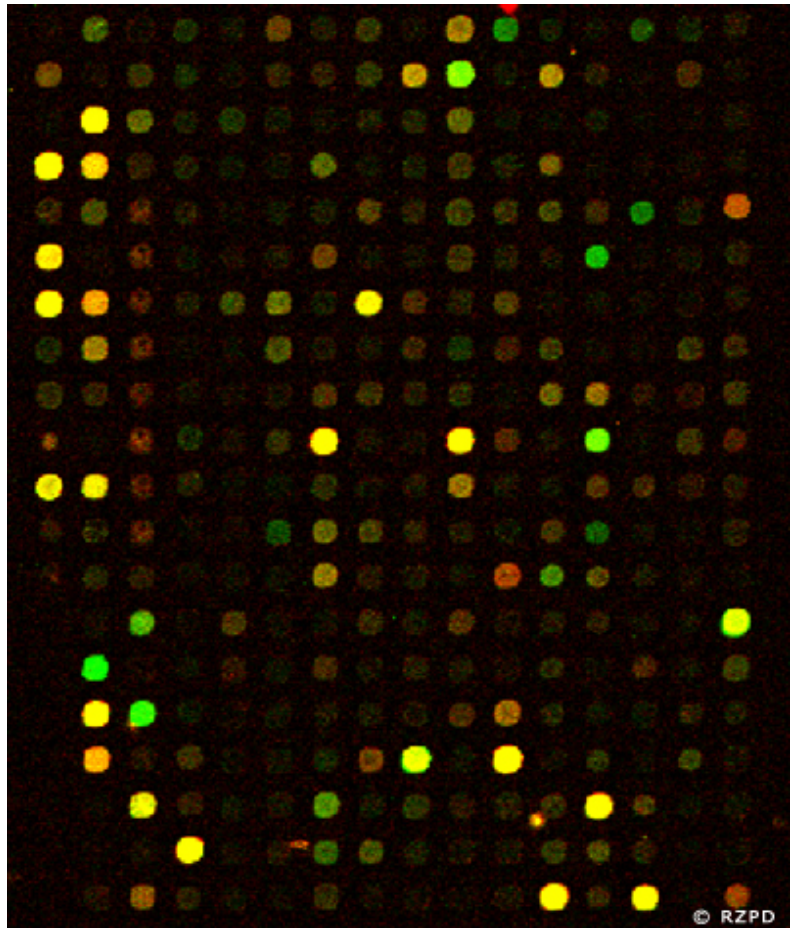
- Trop d'attributs % nombre exemples
  - En enlever — Feature Selection
  - En construire d'autres — Feature Construction
  - En construire moins — Dimensionality Reduction
- Cas logique du 1er ordre : — Propositionalisation

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

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...

Selection de features: approche filtre

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

Backtrack possible

- Etat courant $\mathcal{A}$
- Ajouter a_i à $\mathcal{A}$
- Peut être ajouter a_i rend $a_j \in \mathcal{A}$ inutile ?
- Essayer d'enlever les features de $\mathcal{A}$

Backtrack = moins glouton; meilleures solutions ; beaucoup plus cher.

Selection de features: approche wrapping

Méthode multivariée

Mesurer la qualité d'un ensemble d'attributs :

estimer $\mathcal{F}(a_{i1}, ...a_{ik})$

Contre

Beaucoup plus cher : une estimation = un pb d'apprentissage.

Pour

Optima meilleurs

Selection de features: approche embarquée (embedded)

Principe – online

On rajoute à l'apprentissage un critère qui favorise les hypothèses à peu d'attributs.

Par exemple : trouver w, $h(x) =< w, x >$, qui minimise

$$\sum_i (h(x_i) - y_i)^2 + ||w||$$

Premier terme : coller aux données

Deuxième terme : favoriser w avec beaucoup de coordonnées nulles

Principe – offline

On a trouvé

$$h(x) =< w, x >= \sum_{j=1}^{d} w_j x_j$$

Si $|w_j|$ petit, l'attribut j n'est pas important... Les enlever et recommencer.

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^*

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

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

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

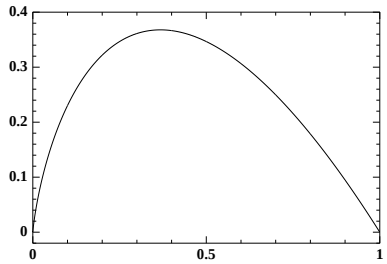
Gain d'information

arbres de décision

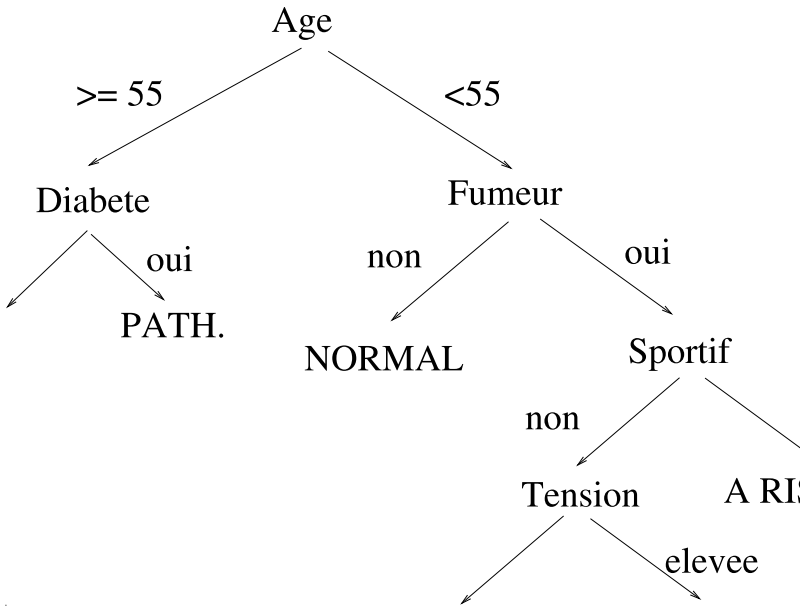
$$p([a = v]) = Pr(y = 1|a(x_i) = v)$$

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

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



Gain d'information, suite



Limitations

- Critère myope — cas du XOR
- Favorise les attributs avec de nombreuses valeurs
- Limité pour les attributs numériques

(voir arbres de décision)

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 $\sum_{c_i,\neg c_i}\sum_{a_k,\neg a_k} P(c,a)Log\frac{p(a,c)}{P(a)P(c)}$
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 $\rightarrow h(x) = sign(\sum w_i.a_i(x) + b)$
- 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

Représentation pour l'apprentissage

- Sélection d'attributs
- Changements de représentation linéaires
- Changements de représentation non linéaires

Partie 2. Changements de représentation lineaires

- Réduction de dimensionalité
- Analyse en composantes principales
- Projections aléatoires
- Analyse sémantique latente

Dimensionality Reduction — Intuition

Degrees of freedom

- Image: 4096 pixels; but not independent
- Robotics: (# camera pixels + # infra-red) $\times$ time; but not independent

Goal

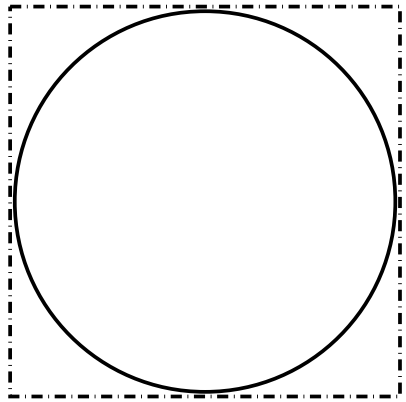
Find the (low-dimensional) structure of the data:

- Images
- Robotics
- Genes

Dimensionality Reduction

In high dimension

- Everybody lives in the corners of the space
  Volume of Sphere $V_n = \frac{2\pi r^2}{n} V_{n-2}$
- All points are far from each other



Approaches

- Linear dimensionality reduction
  - Principal Component Analysis
  - Random Projection
- Non-linear dimensionality reduction

Criteria

- Complexity/Size
- Prior knowledge

e.g., relevant distance

Linear Dimensionality Reduction

Training set *unsupervised*

$$\mathcal{E} = \{(\mathbf{x}_k), \mathbf{x}_k \in \mathbb{R}^D, k = 1 \dots N\}$$

Projection from $\mathbb{R}^D$ onto $\mathbb{R}^d$

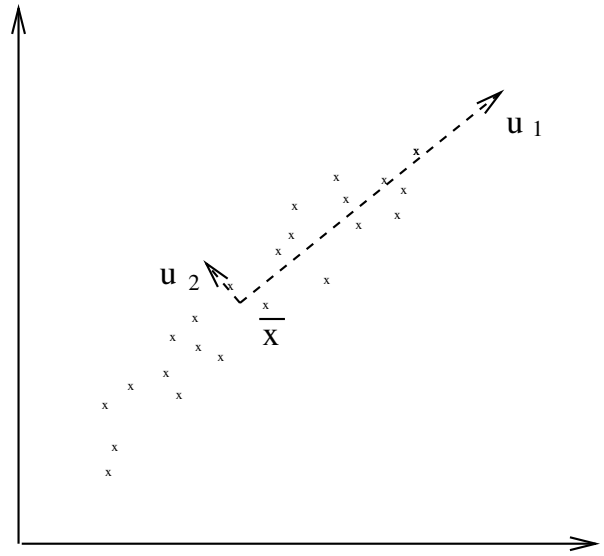
$$\begin{aligned} \mathbf{x} \in \mathbb{R}^D \rightarrow \quad & h(\mathbf{x}) \in \mathbb{R}^d, \ d << D \\ & h(\mathbf{x}) = A\mathbf{x} \\ s.t. \text{ minimize} \quad & \sum_{k=1}^{N} ||\mathbf{x}_k - h(\mathbf{x}_k)||^2 \end{aligned}$$

Principal Component Analysis

Covariance matrix S

Mean $\mu_i = \frac{1}{N}\sum_{k=1}^{N} X_i(\mathbf{x}_k)$

$$S_{ij} = \frac{1}{N}\sum_{k=1}^{N}(X_i(\mathbf{x}_k) - \mu_i)(X_j(\mathbf{x}_k) - \mu_j)$$



symmetric $\Rightarrow$ can be diagonalized

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

Thm: Optimal projection in dimension d

projection on the first d eigenvectors of S

Let u_i the eigenvector associated to eigenvalue λ_i $\lambda_i > \lambda_{i+1}$

$$h : \mathbb{R}^D \mapsto \mathbb{R}^d, h(\mathbf{x}) = < \mathbf{x}, u_1 > u_1 + \ldots + < \mathbf{x}, u_d > u_d$$

Sketch of the proof

1. Maximize the variance of $h(\mathbf{x}) = A\mathbf{x}$

$\sum_k ||\mathbf{x}_k - h(\mathbf{x}_k)||^2 = \sum_k ||\mathbf{x}_k||^2 - \sum_k ||h(\mathbf{x}_k)||^2$

$$\text{Minimize } \sum_k ||\mathbf{x}_k - h(\mathbf{x}_k)||^2 \Rightarrow \text{ Maximize } \sum_k ||h(\mathbf{x}_k)||^2$$

$$Var(h(\mathbf{x})) = \frac{1}{N}\left(\sum_k ||h(\mathbf{x}_k)||^2 - ||\sum_k h(\mathbf{x}_k)||^2\right)$$

As

$$||\sum_k h(\mathbf{x}_k)||^2 = ||A\sum_k \mathbf{x}_k||^2 = N^2||A\mu||^2$$

where $\mu = (\mu_1, \ldots . \mu_D)$.
Assuming that $\mathbf{x}_k$ are centered ($\mu_i = 0$) gives the result.

Sketch of the proof, 2

2. Projection on eigenvectors u_i of S

Assume $h(\mathbf{x}) = A\mathbf{x} = \sum_{i=1}^{d} < \mathbf{x}, v_i > v_i$ and show $v_i = u_i$.

$$Var(AX) = (AX)(AX)' = A(XX')A' = ASA' = A(U\Delta U')A'$$

Consider $d = 1,\ v_1 = \sum w_i u_i$ $\qquad$ $\sum w_i^2 = 1$

remind $\lambda_i > \lambda_{i+1}$

$$Var(AX) = \sum \lambda_i w_i^2$$

maximized for $w_1 = 1, w_2 = \ldots = w_N = 0$

that is, $v_1 = u_i$.

Principal Component Analysis, Practicalities

Data preparation

- Mean centering the dataset

$$\begin{aligned} \mu_i &= \frac{1}{N}\sum_{k=1}^{N} X_i(\mathbf{x}_k) \\ \sigma_i &= \sqrt{\frac{1}{N}\sum_{k=1}^{N} X_i(\mathbf{x}_k)^2 - \mu_i^2} \\ z_k &= \left(\frac{1}{\sigma_i}(X_i(\mathbf{x}_k) - \mu_i)\right)_{i=1}^{D} \end{aligned}$$

Matrix operations

- Computing the covariance matrix

$$S_{ij} = \frac{1}{N}\sum_{k=1}^{N} X_i(z_k)X_j(z_k)$$

- Diagonalizing $S = U'\Delta U$ might be not affordable... Complexity $\mathcal{O}(D^3)$

Random projection

Random matrix

$$A : \mathbb{R}^D \mapsto \mathbb{R}^d \qquad A[d, D] \qquad A_{i,j} \sim \mathcal{N}(0, 1)$$

define

$$h(\mathbf{x}) = \frac{1}{\sqrt{d}} A\mathbf{x}$$

Property: h preserves the norm in expectation

$$E[||h(\mathbf{x})||^2] = ||\mathbf{x}||^2$$

With high probability $1 - 2exp\{-(\varepsilon^2 - \varepsilon^3)\frac{d}{4}\}$

$$(1 - \varepsilon)||\mathbf{x}||^2 \leq ||h(\mathbf{x})||^2 \leq (1 + \varepsilon)||\mathbf{x}||^2$$

Random projection

Proof

$$
\begin{aligned}
h(\mathbf{x}) = & \quad \frac{1}{\sqrt{d}} A\mathbf{x} \\
E(||h(\mathbf{x})||^2) = & \quad \frac{1}{d} E\left[\sum_{i=1}^{d}\left(\sum_{j=1}^{D} A_{i,j} X_j(\mathbf{x})\right)^2\right] \\
= & \quad \frac{1}{d}\sum_{i=1}^{d} E\left[\left(\sum_{j=1}^{D} A_{i,j} X_j(\mathbf{x})\right)^2\right] \\
= & \quad \frac{1}{d}\sum_{i=1}^{d}\sum_{j=1}^{D} E[A_{i,j}^2] E[X_j(\mathbf{x})^2] \\
= & \quad \frac{1}{d}\sum_{i=1}^{d}\sum_{j=1}^{D} \frac{||\mathbf{x}||^2}{D} \\
= & \quad ||\mathbf{x}||^2
\end{aligned}
$$

Random projection, 2

Johnson Lindenstrauss Lemma

For $d > \frac{9 \ln N}{\varepsilon^2 - \varepsilon^3}$, with high probability

$$(1-\varepsilon)||\mathbf{x}_i - \mathbf{x}_j||^2 \leq ||h(\mathbf{x}_i) - h(\mathbf{x}_j)||^2 \leq (1+\varepsilon)||\mathbf{x}_i - \mathbf{x}_j||^2$$

More:

`http://www.cs.yale.edu/clique/resources/RandomProjectionMethod.pdf`

Analyse Sémantique Latente - LSA

1. Motivation
2. Algorithme
3. Discussion

Exemple

- c1: ***Human*** machine ***interface*** for ABC ***computer*** applications
- c2: A ***survey*** of ***user*** opinion of ***computer system response time***
- c3: The ***EPS user interface*** management ***system***
- c4: ***System*** and ***human system*** engineering testing of ***EPS***
- c5: Relation of ***user*** perceived ***response time*** to error measurement

- m1: The generation of random, binary, ordered ***trees***
- m2: The intersection ***graph*** of paths in ***trees***
- m3: ***Graph minors*** IV: Widths of ***trees*** and well-quasi-ordering
- m4: ***Graph minors***: A ***survey***

Exemple, suite

	c1	c2	c3	c4	c5	m1	m2	m3	m4
human	1	0	0	1	0	0	0	0	0
interface	1	0	1	0	0	0	0	0	0
computer	1	1	0	0	0	0	0	0	0
user	0	1	1	0	1	0	0	0	0
system	0	1	1	2	0	0	0	0	0
response	0	1	0	0	1	0	0	0	0
time	0	1	0	0	1	0	0	0	0
EPS	0	0	1	1	0	0	0	0	0
survey	0	1	0	0	0	0	0	0	1
trees	0	0	0	0	0	1	1	1	0
graph	0	0	0	0	0	0	1	1	1
minors	0	0	0	0	0	0	0	1	1

LSA, 2

Motivations

- Contexte : représentation sac de mots
- Malédiction de la dimensionalité $\mathbb{R}^D$
- Synonymie / Polysémie

Objectifs

- Réduire la dimension $\mathbb{R}^d$
- Avoir une "bonne topologie" une bonne distance

Remarque

- une similarité évidente : le cosinus
- pourquoi ce n'est pas bon ?

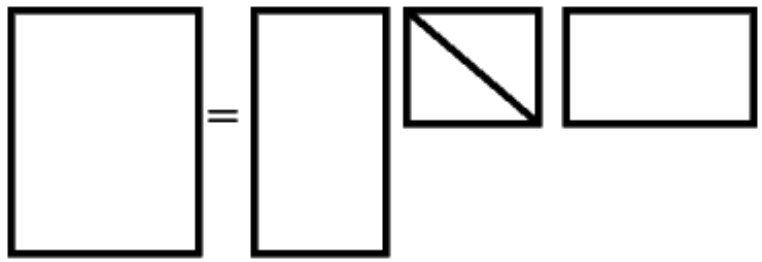
Plus d'info

`http://lsa.colorado.edu`

LSA, 3

Input

Matrice X = mots $\times$ documents



Principe

1. Changement de base des mots,documents aux concepts
2. Réduction de dimension

Différence Analyse en composantes principales

LSA ≡ Singular Value Decomposition

Input

X matrice mots × documents $m \times d$

$$X = U' \; S \; V$$

avec

- U : changement de base mots $m \times r$
- V : changement de base des documents $r \times d$
- S : matrice diagonale $r \times r$

Réduction de dimension

- S Ordonner par valeur propre décroissante
- $S' = S$ avec annulation de toutes les vp, sauf les (300) premières.

$$X' = U'S'V$$

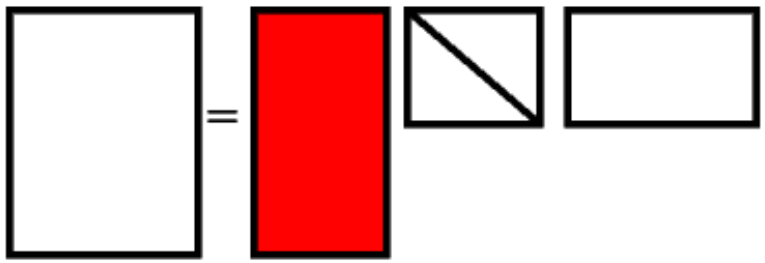
Intuition

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

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,

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

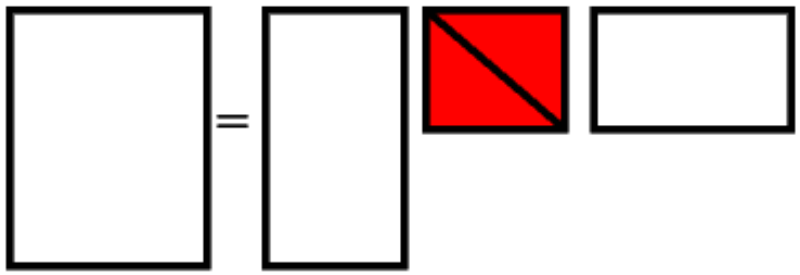
Algorithme



Singular value Decomposition of the words by contexts matrix

0.22	-0.11	0.29	-0.41	-0.11	-0.34	0.52	-0.06	-0.41
0.20	-0.07	0.14	-0.55	0.28	0.50	-0.07	-0.01	-0.11
0.24	0.04	-0.16	-0.59	-0.11	-0.25	-0.30	0.06	0.49
0.40	0.06	-0.34	0.10	0.33	0.38	0.00	0.00	0.01
0.64	-0.17	0.36	0.33	-0.16	-0.21	-0.17	0.03	0.27
0.27	0.11	-0.43	0.07	0.08	-0.17	0.28	-0.02	-0.05
0.27	0.11	-0.43	0.07	0.08	-0.17	0.28	-0.02	-0.05
0.30	-0.14	0.33	0.19	0.11	0.27	0.03	-0.02	-0.17
0.21	0.27	-0.18	-0.03	-0.54	0.08	-0.47	-0.04	-0.58
0.01	0.49	0.23	0.03	0.59	-0.39	-0.29	0.25	-0.23
0.04	0.62	0.22	0.00	-0.07	0.11	0.16	-0.68	0.23
0.03	0.45	0.14	-0.01	-0.30	0.28	0.34	0.68	0.18

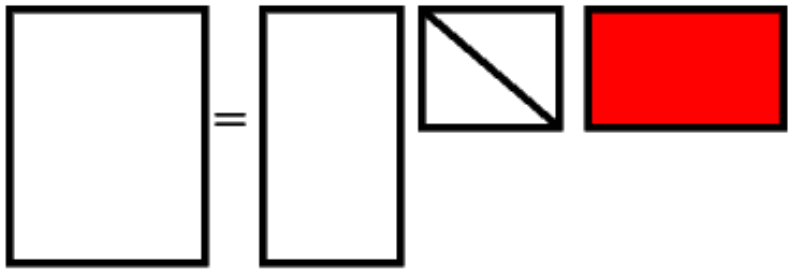
Algorithme, 2



Singular value Decomposition of the words by contexts matrix

3.34
2.54
2.35
1.64
1.50
1.31
0.85
0.56
0.36

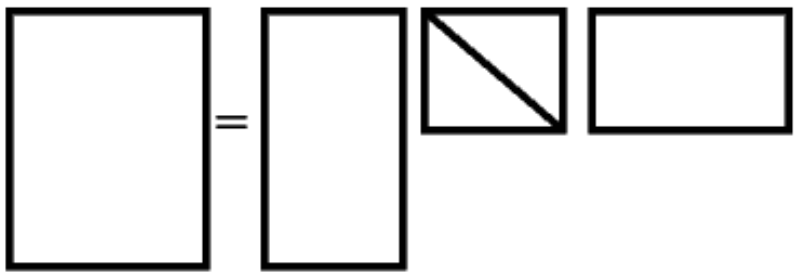
Algorithme. 3



Singular value Decomposition of the words by contexts matrix

0.20	0.61	0.46	0.54	0.28	0.00	0.01	0.02	0.08
-0.06	0.17	-0.13	-0.23	0.11	0.19	0.44	0.62	0.53
0.11	-0.50	0.21	0.57	-0.51	0.10	0.19	0.25	0.08
-0.95	-0.03	0.04	0.27	0.15	0.02	0.02	0.01	-0.03
0.05	-0.21	0.38	-0.21	0.33	0.39	0.35	0.15	-0.60
-0.08	-0.26	0.72	-0.37	0.03	-0.30	-0.21	0.00	0.36
0.18	-0.43	-0.24	0.26	0.67	-0.34	-0.15	0.25	0.04
-0.01	0.05	0.01	-0.02	-0.06	0.45	-0.76	0.45	-0.07
-0.06	0.24	0.02	-0.08	-0.26	-0.62	0.02	0.52	-0.45

Algorithme, 4

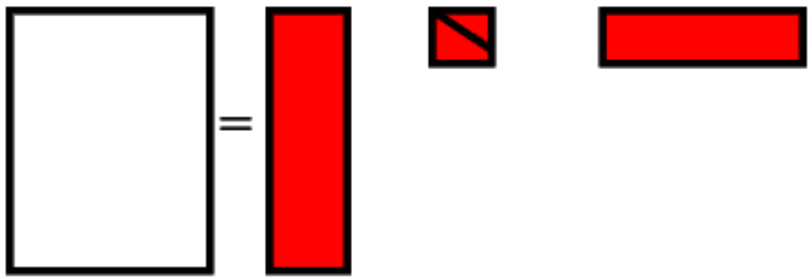


Singular value Decomposition of the words by contexts matrix

3.34

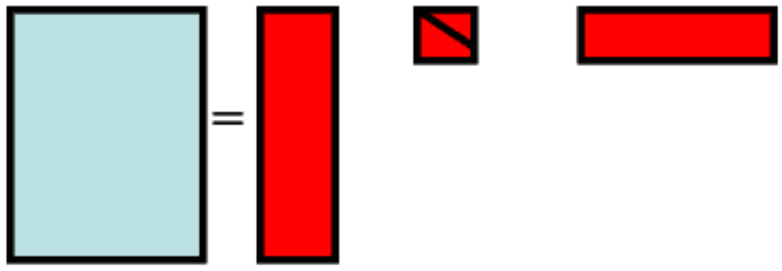
2.54

Algorithme, 5



Singular value
Decomposition of the
words by contexts matrix

Algorithme, 6



Singular value Decomposition of the words by contexts matrix

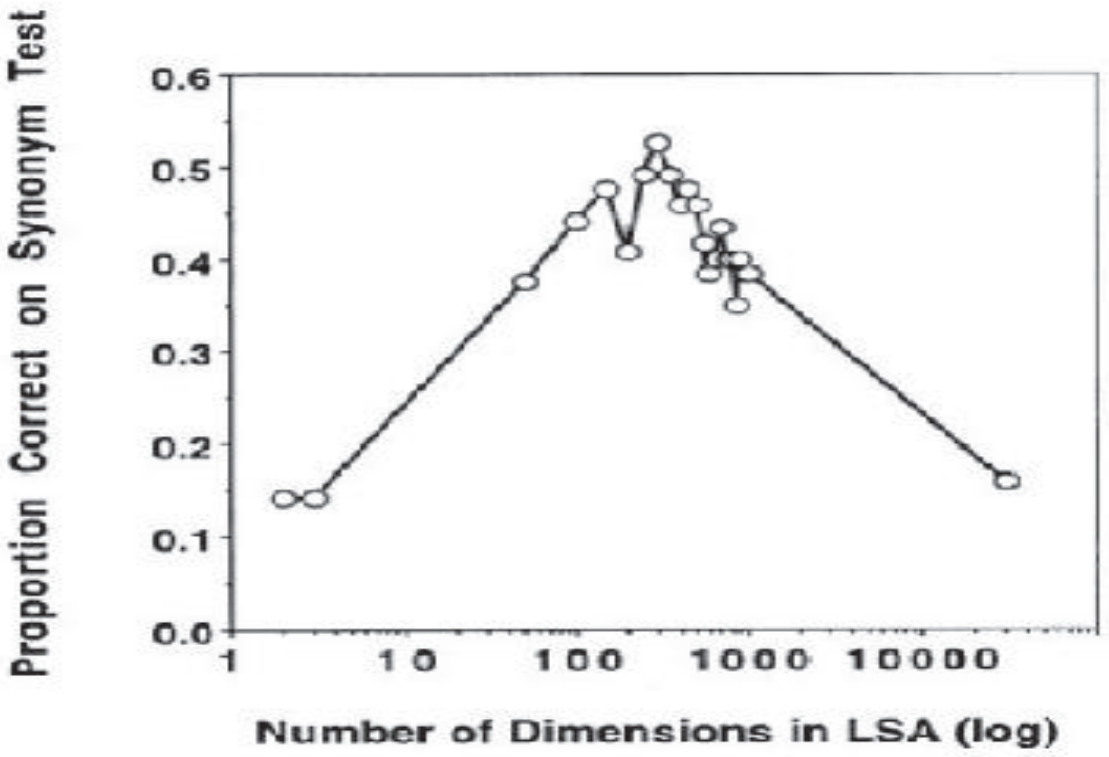
	c1	c2	c3	c4	c5	m1	m2	m3	m4
human	0.16	0.40	0.38	0.47	0.18	-0.05	-0.12	-0.16	-0.09
interface	0.14	0.37	0.33	0.40	0.16	-0.03	-0.07	-0.10	-0.04
computer	0.15	0.51	0.36	0.41	0.24	0.02	0.06	0.09	0.12
user	0.26	0.84	0.61	0.70	0.39	0.03	0.08	0.12	0.19
system	0.45	1.23	1.05	1.27	0.56	-0.07	-0.15	-0.21	-0.05
response	0.16	0.58	0.38	0.42	0.28	0.06	0.13	0.19	0.22
time	0.16	0.58	0.38	0.42	0.28	0.06	0.13	0.19	0.22
EPS	0.22	0.55	0.51	0.63	0.24	-0.07	-0.14	-0.20	-0.11
survey	0.10	0.53	0.23	0.21	0.27	0.14	0.31	0.44	0.42
trees	-0.06	0.23	-0.14	-0.27	0.14	0.24	0.55	0.77	0.66
graph	-0.06	0.34	-0.15	-0.30	0.20	0.31	0.69	0.98	0.85
minors	-0.04	0.25	-0.10	-0.21	0.15	0.22	0.50	0.71	0.62

	c1	c2	c3	c4	c5	m1	m2	m3	m4
human	1	0	0	1	0	0	0	0	0
interface	1	0	1	0	0	0	0	0	0
computer	1	1	0	0	0	0	0	0	0
user	0	1	1	0	1	0	0	0	0
system	0	1	1	2	0	0	0	0	0
response	0	1	0	0	1	0	0	0	0
time	0	1	0	0	1	0	0	0	0
EPS	0	0	1	1	0	0	0	0	0
survey	0	1	0	0	0	0	0	0	1
trees	0	0	0	0	0	1	1	1	0
graph	0	0	0	0	0	0	1	1	1
minors	0	0	0	0	0	0	0	1	1

Discussion

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

LSA – Analyse en composantes principales

Ressemblances

- Prendre une matrice
- La mettre sous forme diagonale
- Annuler toutes les valeurs propres sauf les plus grandes
- Projeter sur l'espace obtenu

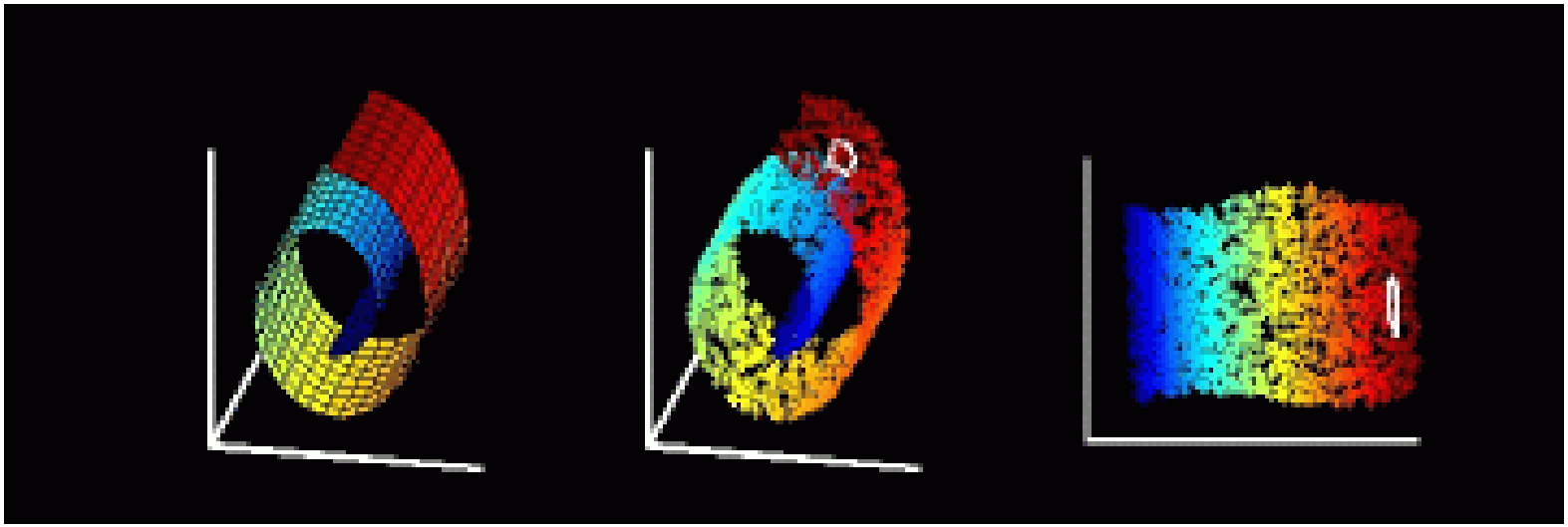
Différences

	ACP	LSA
Matrice	covariance attributs	mots $\times$ documents
d	2-3	100-300

Représentation pour l'apprentissage

- Sélection d'attributs
- Changements de représentation linéaires
- Changements de représentation non linéaires

Non-Linear Dimensionality Reduction



Conjecture

Examples live in a manifold of dimension $d << D$

Goal: consistent projection of the dataset onto $\mathbb{R}^d$

Consistency:

- Preserve the structure of the data
- e.g. preserve the distances between points

Multi-Dimensional Scaling

Position of the problem

- Given $\{\mathbf{x}_1, \ldots, \mathbf{x}_N,\ \mathbf{x}_i \in \mathbb{R}^D\}$
- Given $sim(\mathbf{x}_i, \mathbf{x}_j) \in \mathbb{R}^+$
- Find projection Φ onto $\mathbb{R}^d$

$$\begin{array}{rl} x \in \mathbb{R}^D \rightarrow & \Phi(x) \in \mathbb{R}^d \\ sim(\mathbf{x}_i, \mathbf{x}_j) \sim & sim(\Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j)) \end{array}$$

Optimisation

Define $X,\ X_{i,j} = sim(\mathbf{x}_i, \mathbf{x}_j)$; $X^\Phi,\ X^\Phi_{i,j} = sim(\Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j))$

Find Φ minimizing $||X - X'||$

Rq : Linear Φ = Principal Component Analysis

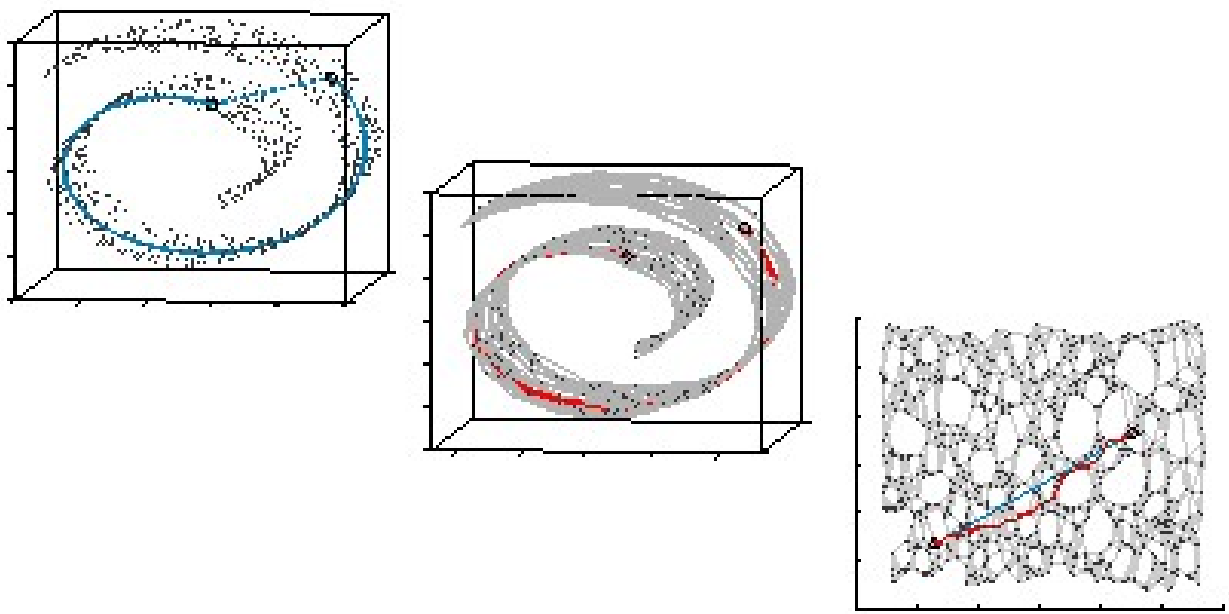
But linear MDS does not work: preserves all distances, while only *local* distances are meaningful

Non-linear projections

Approaches

- Reconstruct global structures from local ones and find global projection — Isomap
- Only consider local structures — LLE

Intuition: locally, points live in $\mathbb{R}^d$



Isomap

Tenenbaum, da Silva, Langford 2000

http://isomap.stanford.edu

Estimate $d(x_i, x_j)$

- Known if $\mathbf{x}_i$ and $\mathbf{x}_j$ are close
- Otherwise, compute the shortest path between $\mathbf{x}_i$ and $\mathbf{x}_j$
  geodesic distance (dynamic programming)

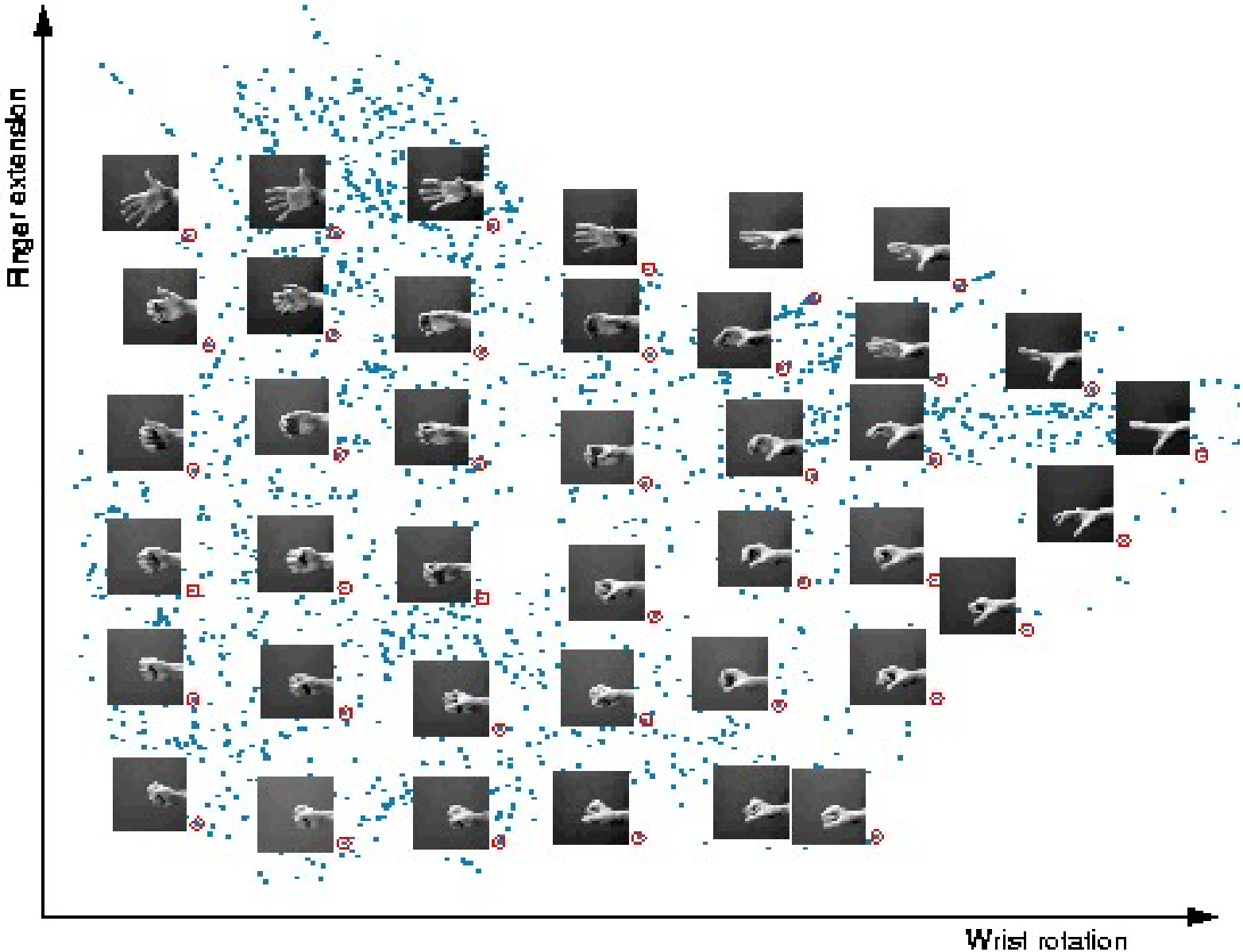
Requisite

If data points sampled in a convex subset of $\mathbb{R}^d$,
then geodesic distance $\sim$ Euclidean distance on $\mathbb{R}^d$.

General case

- Given $d(\mathbf{x}_i, \mathbf{x}_j)$, estimate $< \mathbf{x}_i, \mathbf{x}_j >$
- Project points in $\mathbb{R}^d$

Isomap, 2



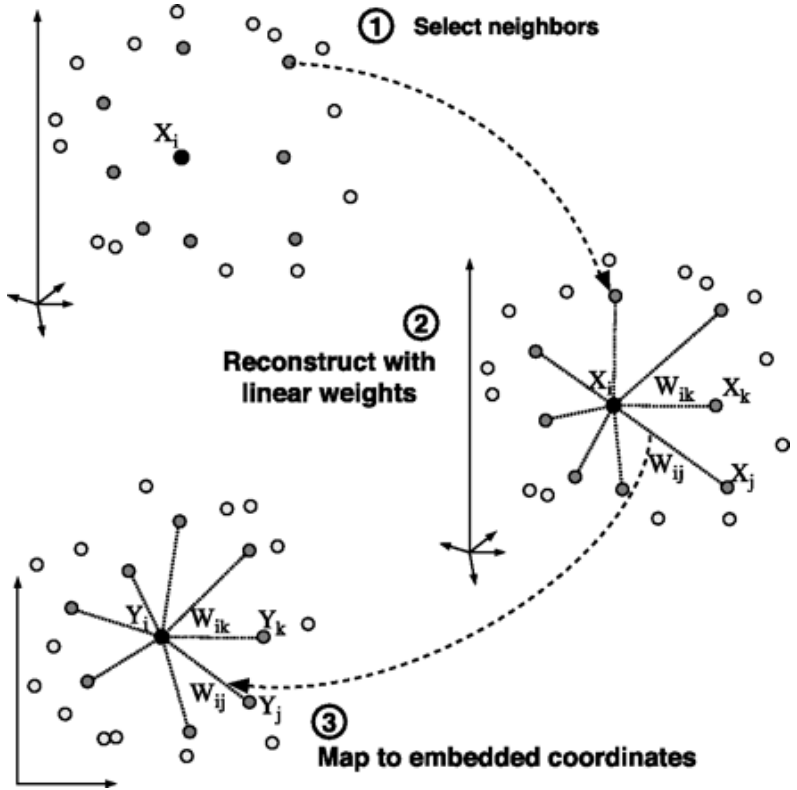
Locally Linear Embedding

Roweiss and Saul, 2000

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

Principle

- Find local description for each point: depending on its neighbors



Local Linear Embedding, 2

Find neighbors

For each $\mathbf{x}_i$, find its nearest neighbors $\mathcal{N}(i)$

Parameter: number of neighbors

Change of representation

Goal Characterize $\mathbf{x}_i$ wrt its neighbors:

$$\mathbf{x}_i = \sum_{j \in \mathcal{N}(i)} w_{i,j} \mathbf{x}_j \quad \text{with} \quad \sum_{j \in \mathcal{N}(i)} w_{ij} = 1$$

Property: invariance by translation, rotation, homothety

How Compute the local covariance matrix:

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

Find vector w_i s.t. $Cw_i = 1$

Local Linear Embedding, 3

Algorithm

Local description: Matrix W such that $\sum_j w_{i,j} = 1$

$$W = argmin\{\sum_{i=1}^{N} ||\mathbf{x}_i - \sum_j w_{i,j}\mathbf{x}_j||^2\}$$

Projection: Find $\{z_1, \ldots, z_n\}$ *in* $\mathbb{R}^d$ minimizing

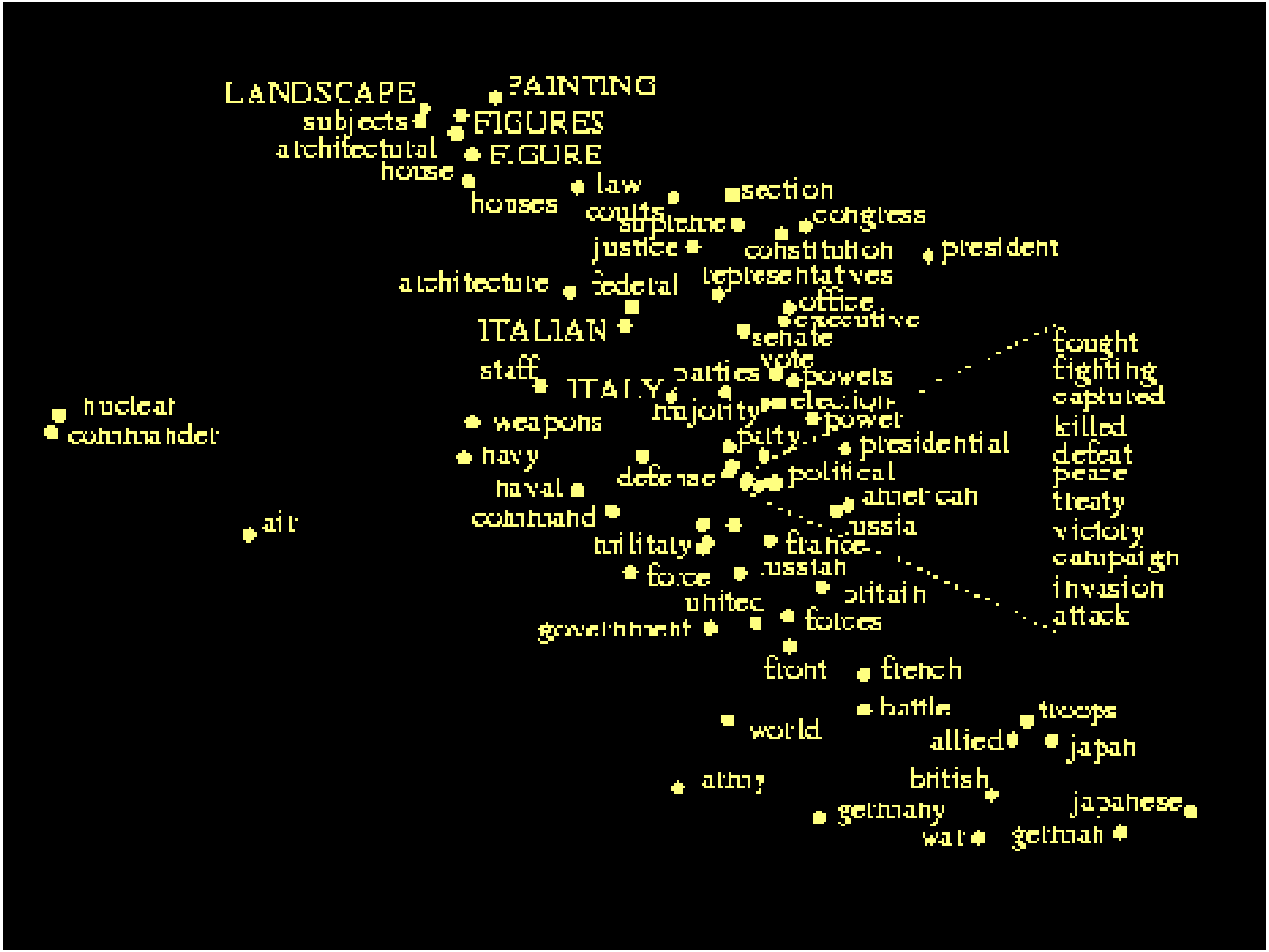
$$\sum_{i=1}^{N} ||z_i - \sum_j w_{i,j} z_j||^2$$

$$Minimize\ ((I - W)Z)'((I - W)Z) = Z'(I - W)'(I - W)Z$$

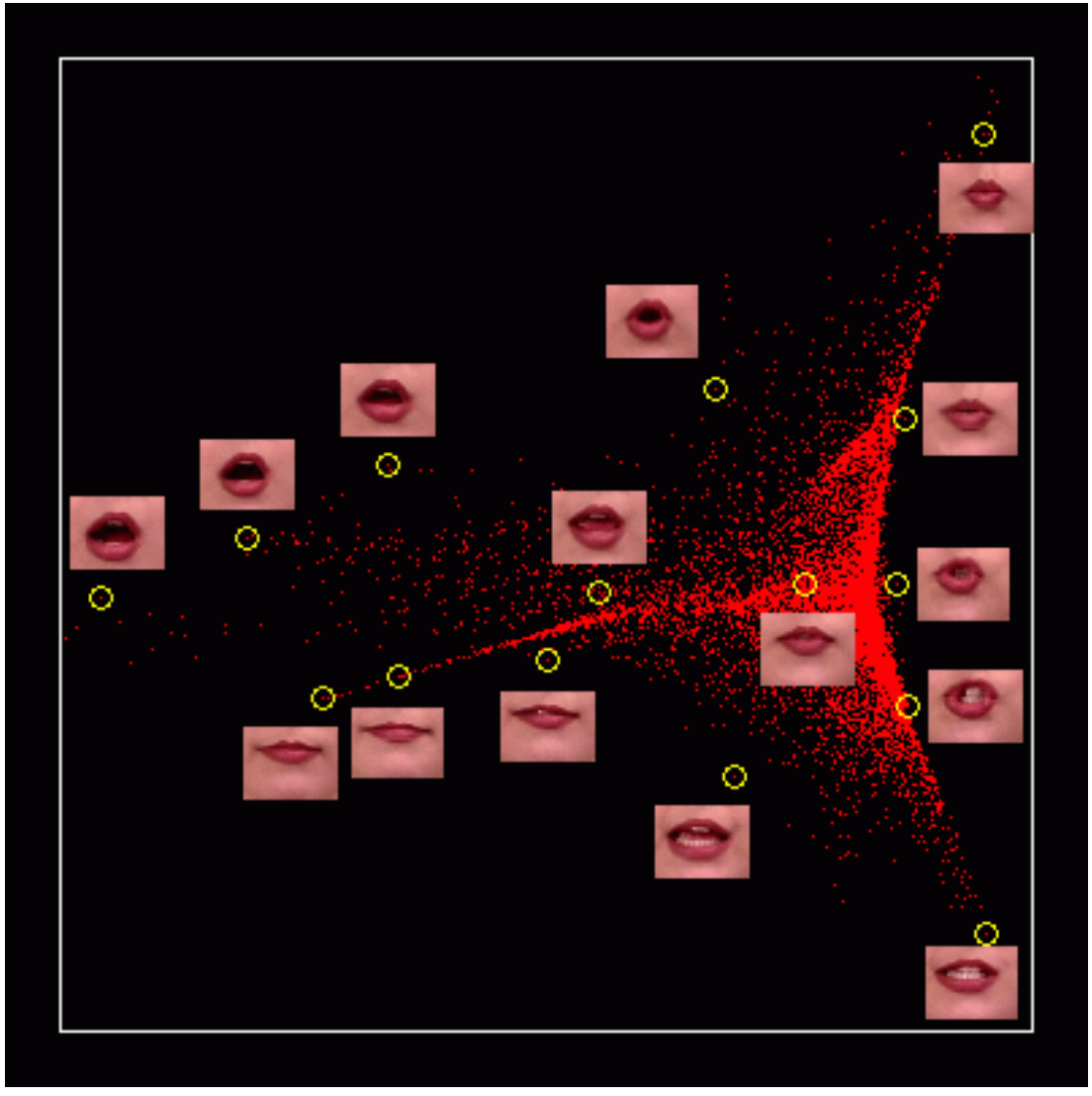
Solutions: vectors z_i are eigenvectors of $(I - W)'(I - W)$

- Keeping the d eigenvectors with lowest eigenvalues > 0

Example, Texts



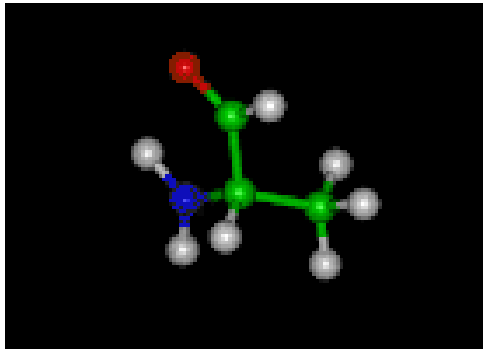
Example, Images

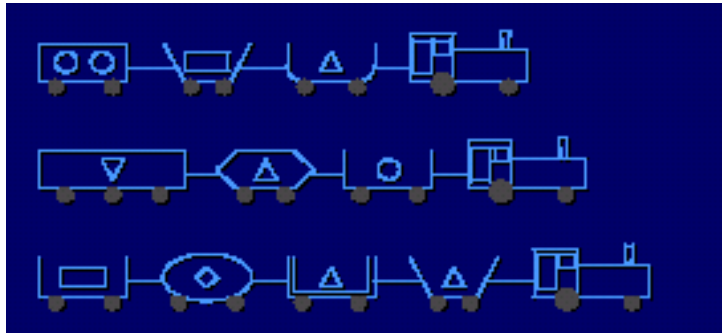


LLE

Propositionalization

Relational domains





Relational learning

PROS

Use domain knowledge

CONS

Covering test $\equiv$ subgraph matching

Inductive Logic Programming

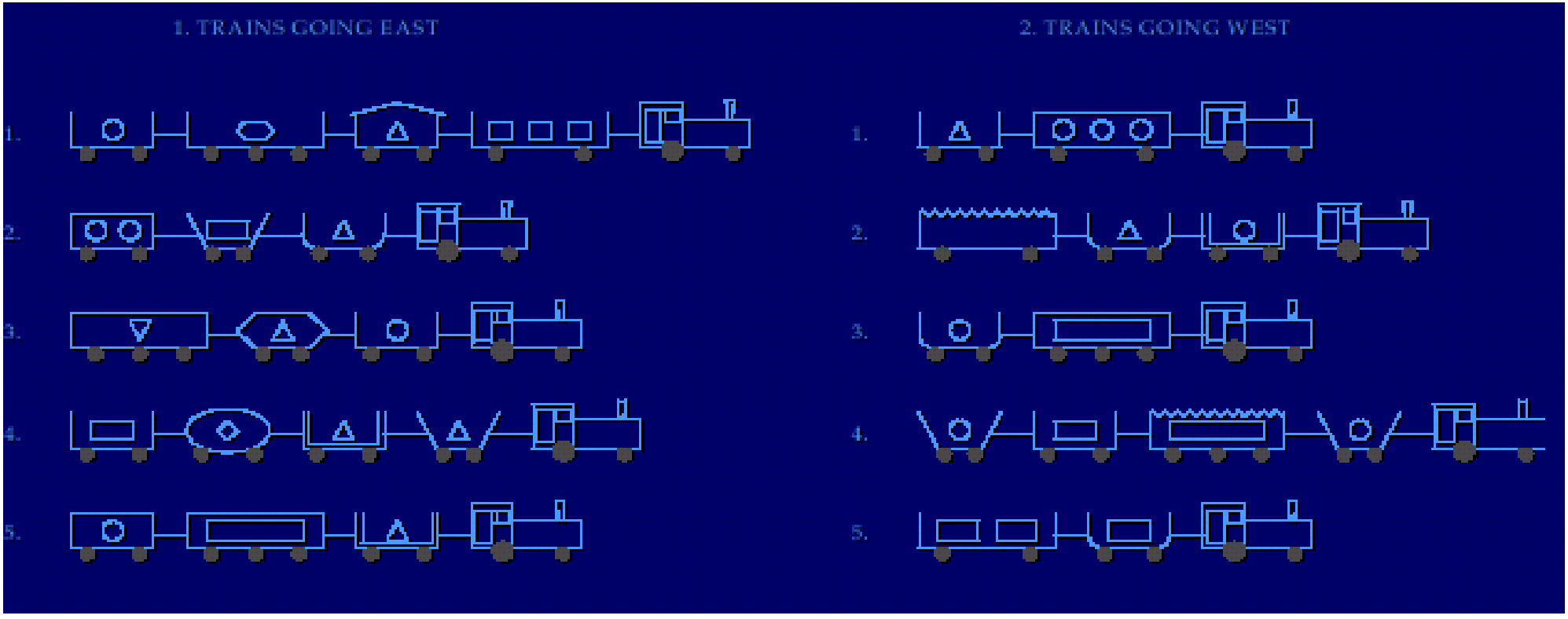
Data Mining

exponential complexity

Getting back to propositional representation: propositionalization

West - East trains

Michalski 1983



Propositionalization

Linus (ancestor)

Lavrac et al, 94

$$West(a) \leftarrow \quad Engine(a,b), first\_wagon(a,c), roof(c), load(c,square,3)...$$
$$West(a') \leftarrow \quad Engine(a',b'), first\_wagon(a',c'), load(c',circle,1)...$$

West	Engine(X)	First Wagon(X,Y)	Roof(Y)	$Load_1$ (Y)	$Load_2$ (Y)
a	b	c	yes	square	3
a'	b'	c'	no	circle	1

Each column: a role predicate, where the predicate is determinate linked to former predicates (left columns) with a single instantiation in every example

Propositionalization

Stochastic propositionalization

Kramer, 98

Construct random formulas $\equiv$ boolean features

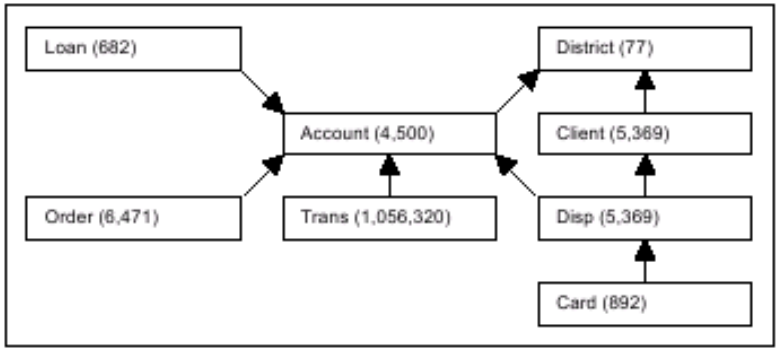
SINUS – RDS

`http://www.cs.bris.ac.uk/home/rawles/sinus`

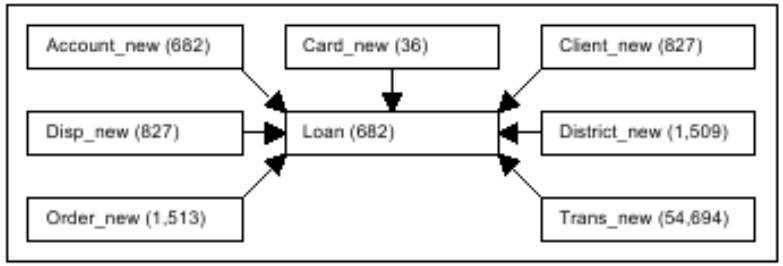
`http://labe.felk.cvut.cz/~zelezny/rsd`

- Use modes (user-declared) `modeb(2,hasCar(+train,-car))`
- Thresholds on number of variables, depth of predicates...
- Pre-processing (feature selection)

Propositionalization



DB Schema



Propositionalization

RELAGGS

Database aggregates

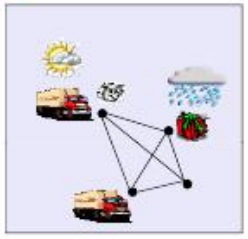
- average, min, max, of numerical attributes
- number of values of categorical attributes

Apprentissage par Renforcement Relationnel

Real Time Strategy Games

- Many objects of various types in complex interactions
- Good players can generalize across situations involving distinct object configurations

The Logistics Domain

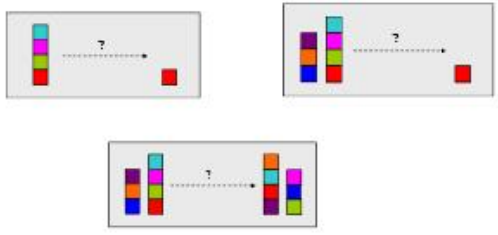


- Move many objects around with many other objects
- Identities and numbers of objects always changing

Robot Soccer

- Reasoning about relationship between objects (players and ball) key to good play

and of course ..... Blocksworld



- Would like a policy that is independent of number of objects/blocks

Propositionalisation

Contexte variable

- Nombre de robots, position des robots
- Nombre de camions, lieu des secours

Besoin: Abstraire et Generaliser

Attributs

- Nombre d'amis/d'ennemis
- Distance du plus proche robot ami
- Distance du plus proche ennemi