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

30 novembre 2011

Apprendre et Optimiser, quel rapport ?

Apprentissage

- Input : des points des exemples

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

- Output : une fonction
  Classification : $\mathcal{Y} = \{0, 1\}$ Régression : $\mathcal{Y} = \mathbf{R}$

$$h : \mathcal{X} \mapsto \mathcal{Y} \qquad h(x_i) \sim y_i$$

Optimisation

- Input : une fonction fonction objectif, ou fitness

$$\mathcal{F} : \mathcal{X} \mapsto \mathbf{R}$$

- Output : un (ou plusieurs) points : les optima de la fonction

$$\text{Find } x^* = \arg\max \mathcal{F}$$

Apprendre et Optimiser

Apprendre c'est optimiser

- Définir une qualité / un score des hypothèses

$$\textit{score} : \mathcal{H} \mapsto \mathbf{R}$$

- Apprendre = trouver h^*, hypothèse optimale au sens du score
- Deux cas :
  1. Score est une fonction convexe. Tout va bien.
  2. Score n'est pas convexe. Optima locaux ; garanties ?

Optimiser c'est apprendre

- Cas des fonctions chères : apprentissage de modèle approché (surrogate model).

Motivations

Software as Knowledge Capitalization

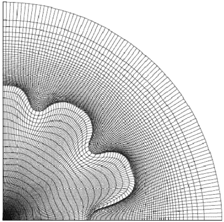
- In Numerical Engineering, part of know-how is encapsulated in softwares
- Computationally heavy
- Require significant expertise   not fool-proof

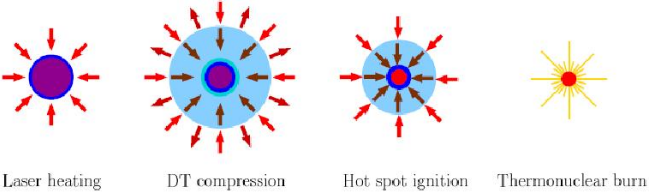
Goal : Simplified models

- h^* = software, input case $x \rightarrow$ response y
- Run $h^*(x_1) = y_1, \ldots h^*(x_n) = y_n$;
- Each run : hours or days of computation
- Learn h_n from $\{(x_1, y_1), \ldots (x_n, y_n)\}$

Inertial Confinement Fusion, ICF

- Large and complex codes (100,000 ℓ ++)
- Computationally heavy (several days on main frame)
- Require expertise





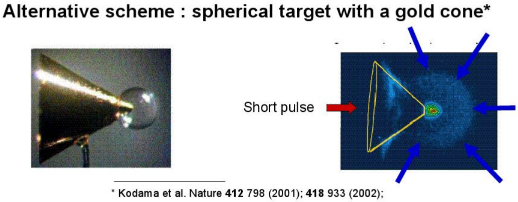
Inertial Confinement Fusion, ICF

Simplified Models in Numerical Engineering

Simplified models aim at providing

- An approximate response
- ...computed in a fraction of the computational time
- ...narrowing the range of design

Long-term goal : Optimal Design *More is Different*



Collaboration J.-M. Martinez, CEA

Apprendre et Optimiser, suite

Applications

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

Exemples

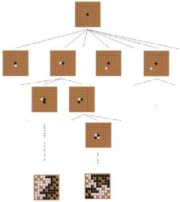
- production : modélisation, calibration
- robotique : modèle du monde, politique d'actions
- médicaments : modèle des effets, conception de molécules

Plan du module

- Introduction — ce cours
- Optimisation — Anne Auger
  - Optimisation continue ; stratégies d"evolution (ES)
  - Algorithme CMA-ES
  - Bornes ($\equiv$ complexité)
- Apprentissage — Balazs Kégl
  - Apprentissage d'ensembles et Boosting
  - Machines à vecteurs support
  - Apprentissage d'ordres
- Applications
  - Apprentissage par renforcement / Jouer au Go / Robotique
  - Validation

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 (video) d'Andrew Ng
  http ://ai.stanford.edu/ ang/courses.html
- Les cours (videos) de PASCAL
  http ://videolectures.net/pascal/
- Les tutoriels de NIPS Neuro Information Processing Systems
  http ://nips.cc/Conferences/
- Des questions intéressantes
  http ://hunch.net/

Contents

Supervised Machine Learning

Context

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

Input : Training set $\mathcal{E} = \{(\mathbf{x_i}, y_i), i = 1 \ldots n, x_i \in \mathcal{X}, y_i \in \mathcal{Y}\}$

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

Criterion : few mistakes (details later)

Definitions

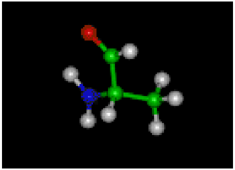
Example

- row : example/ case
- column : feature/variables/attribute
- attribute : class/label

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_serv	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_acc	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	*MissingVa	rich
34	Private	7th_8th	4	Married	...	Transport_	Husband	Amer_India	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
:	:	:	:	:	:	:	:	:	:	:	:	:

Instance space $\mathcal{X}$

- Propositionnal : $\mathcal{X} \equiv \mathbf{R}^d$
- Relational : ex. chemistry.



molecule : alanine

Contents

Difficulty factors

Quality of examples / of representation

+ Relevant features   Feature extraction
– Not enough data
– Noise ; missing data
– Structured data : spatio-temporal, relational, textual, videos ..

Distribution of examples

+ Independent, identically distributed examples
– Other : robotics ; data stream ; heterogeneous data

Prior knowledge

+ Constraints on sought solution
+ Criteria ; loss function

Difficulty factors, 2

Learning criterion

$+$ Convex function : a single optimum

$\searrow$ Complexity : n, $n \log n$, n^2 Scalability

$-$ Combinatorial optimization

What is your agenda ?

- Prediction performance
- Causality
- INTELLIGIBILITY
- Simplicity
- Stability
- Interactivity, visualisation

Difficulty factors, 3

Crossing the chasm

- There exists no *killer algorithm*
- Few general recommendations about algorithm selection

Performance criteria

- Consistency

When number n of examples goes to ∞
and the target concept h^* is in $\mathcal{H}$
Algorithm finds $\hat{h}_n$, with

$$lim_{n \to \infty} h_n = h^*$$

- Convergence speed

$$||h^* - h_n|| = \mathcal{O}(1/n), \mathcal{O}(1/\sqrt{n}), \mathcal{O}(1/\ln n)$$

Contents

Context

Related approaches | **criteria**

- Data Mining, KDD

  scalability

- Statistics and data analysis

  Model selection and fitting ; hypothesis testing

- Machine Learning

  Prior knowledge ; representations ; distributions

- Optimisation

  well-posed / ill-posed problems

- Computer Human Interface

  No ultimate solution : a dialog

- High performance computing

  Distributed data ; privacy

Methodology

Phases

1. Collect data — expert, DB
2. Clean data — stat, expert
3. Select data — stat, expert
4. Data Mining / Machine Learning
   - Description — *what is in data ?*
   - Prediction — *Decide for one example*
   - Agregate — *Take a global decision*
5. Visualisation — chm
6. Evaluation — stat, chm
7. Collect new data — expert, stat

An interative process

depending on expectations, data, prior knowledge, current results

Contents

Au début sont les données...

Patient	AGE	SEX	BMI	BP	··· Serum Measurements ···						Response
	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

Pourquoi la représentation est-elle si importante ?

Deux extrêmes

- Une description trop pauvre $\Rightarrow$ on ne peut rien faire
- Une description trop riche $\Rightarrow$ danger

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

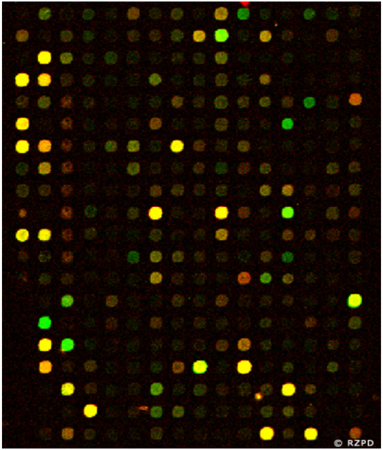
- Trop d'attributs % nombre exemples
  - En enlever — Feature Selection
  - En construire d'autres — Feature Construction
  - En construire moins — Dimensionality Reduction
- Cas logique du 1er ordre : — Propositionalisation

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

Feature Selection

Approches

- Filtre : Définir un score par feature puis sélection gloutonne
- Wrapper : évaluer un sous-ensemble de features
- Intégré : critère d'apprentissage

Détails
http ://tao.lri.fr/tiki-index.php ?page=Courses
Cours_Representation.pdf

Contents

Linear change of representation

- Dimensionality Reduction
- Principal Component Analysis (*)
- Random Projections (*)
- Latent Semantic Analysis

(*) http ://tao.lri.fr/tiki-index.php ?page=Courses
Cours_Representation.pdf

Dimensionality Reduction – Intuition

Degrees of freedom

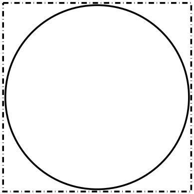
- Image : 4096 pixels ; but not independent
- Robotics : (# camera pixels + # infra-red) $\times$ time ; but not independent

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

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 $\mathbf{R}^D$
- Synonymie / Polysémie

Objectifs

- Réduire la dimension $\mathbf{R}^d$
- Avoir une “bonne topologie” une bonne distance

Remarque

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

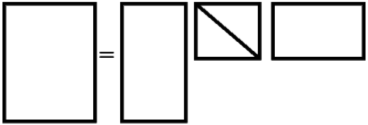
Plus d’info

`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 $\equiv$ Singular Value Decomposition

Input

X matrice mots $\times$ 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.

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

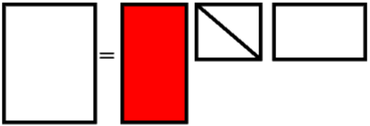
Intuition

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

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,

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

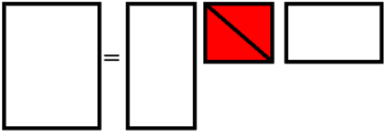
Algorithme



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

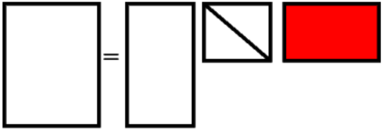
Algorithme, 2



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

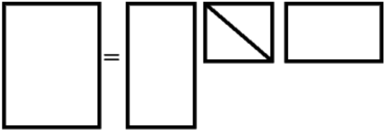
Algorithme. 3



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

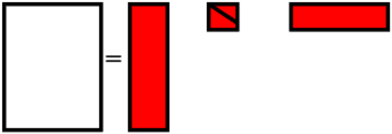
Algorithme, 4



Singular value
Decomposition of the
words by contexts matrix

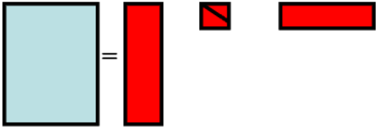
3.34	
	2.54

Algorithme, 5



Singular value
Decomposition of the
words by contexts matrix

Algorithme, 6



Singular value Decomposition of the words by contexts matrix

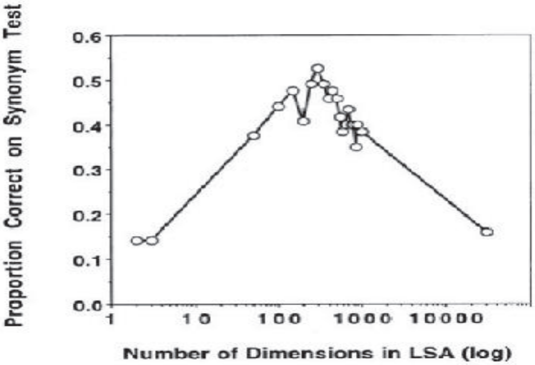
	c1	c2	c3	c4	c5	m1	m2	m3	m4
human	0.16	0.40	0.38	0.47	0.18	-0.05	-0.12	-0.16	-0.09
interface	0.14	0.37	0.33	0.40	0.16	-0.03	-0.07	-0.10	-0.04
computer	0.15	0.51	0.36	0.41	0.24	0.02	0.06	0.09	0.12
user	0.26	0.84	0.61	0.70	0.39	0.03	0.08	0.12	0.19
system	0.45	1.23	1.05	1.27	0.56	-0.07	-0.15	-0.21	-0.05
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

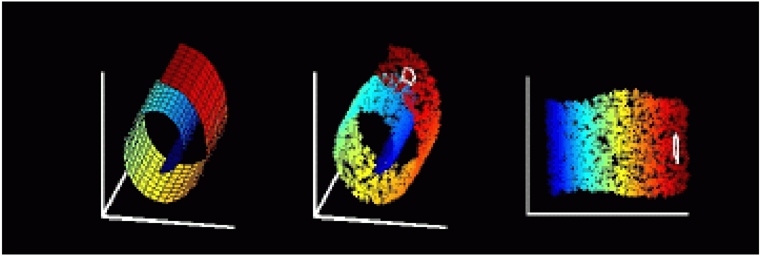
Différences

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

Contents

Contents

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 $\mathbb{R}^d$

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

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

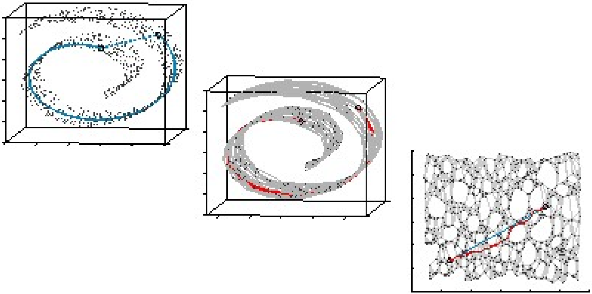
But linear MDS does not work : preserves all distances, while only *local* distances are meaningful

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 $\mathbf{R}^d$



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)

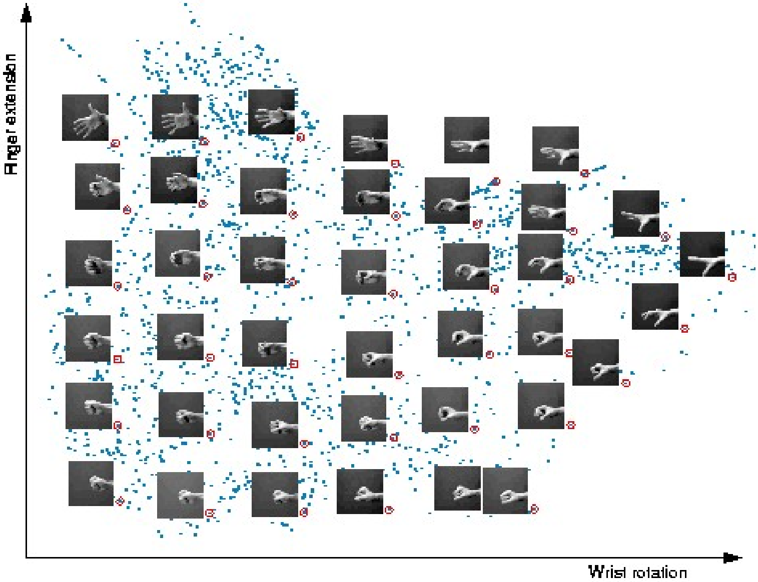
Requisite

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

General case

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

Isomap, 2



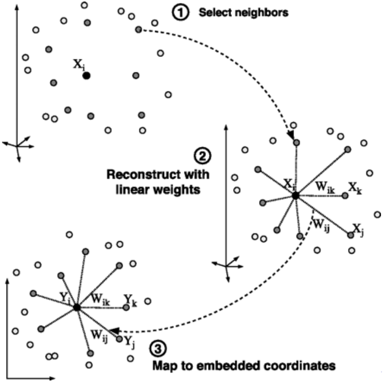
Locally Linear Embedding

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 :

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

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* $\mathbf{R}^d$ minimizing

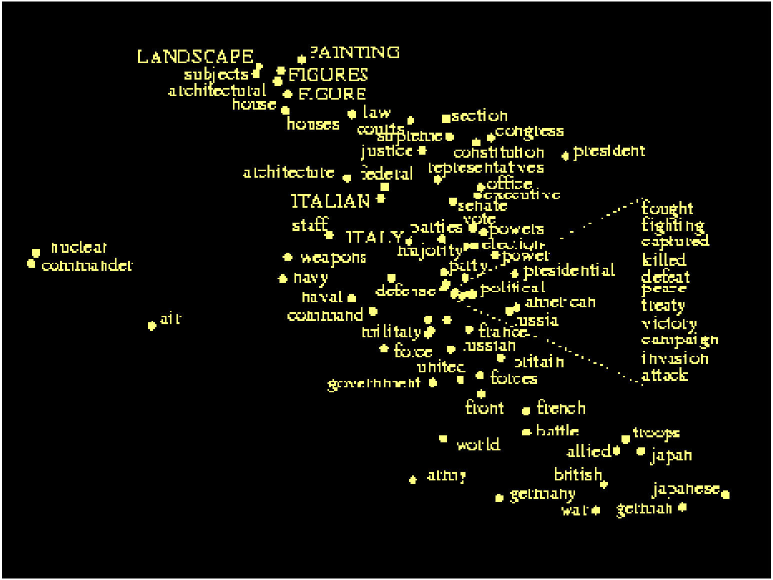
$$\sum_{i=1}^{N} ||z_i - \sum_j w_{i,j} z_j||^2$$

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

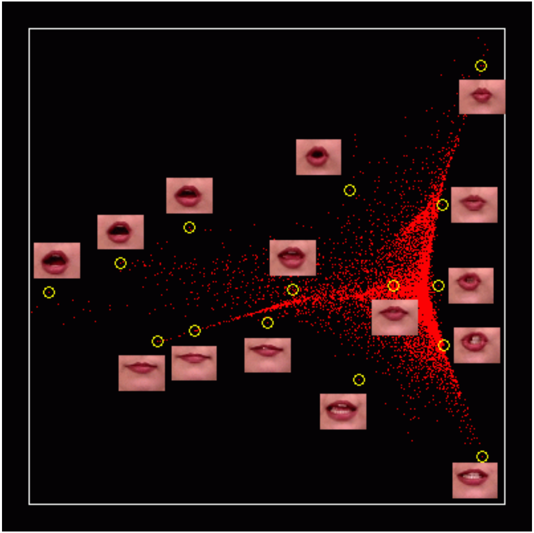
Solutions : vectors z_i are eigenvectors of $(I-W)'(I-W)$

- Keeping the d eigenvectors with lowest eigenvalues > 0

Example, Texts



Example, Images

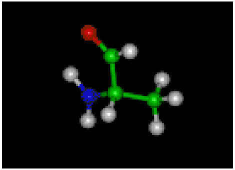


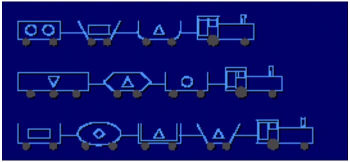
LLE

Contents

Propositionalization

Relational domains





Relational learning

PROS — Inductive Logic Programming

Use domain knowledge

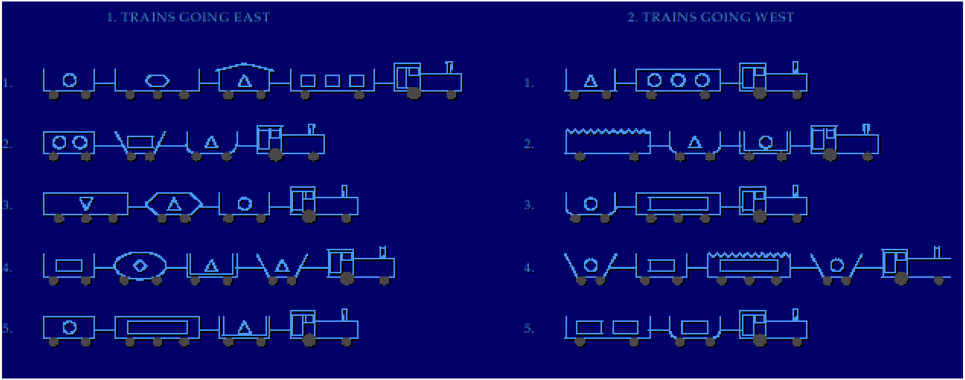
CONS — Data Mining

Covering test $\equiv$ subgraph matching — 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

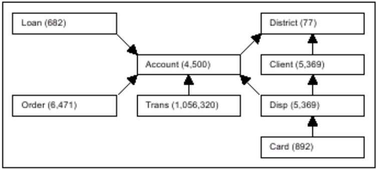
SINUS – RDS

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

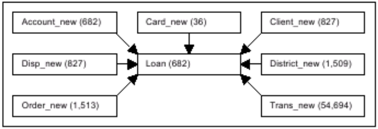
`http://labe.felk.cvut.cz/~zelezny/rsd`

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

Propositionalization



DB Schema



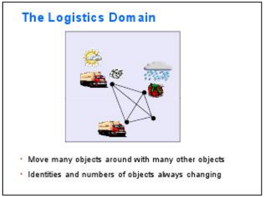
Propositionalization

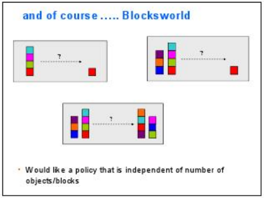
RELAGGS

Database aggregates

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

Apprentissage par Renforcement Relationnel





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

Contents

En guise de conclusion

- Toute attention prêtée à la représentation est rendue au centuple
- La suite : les noyaux SVM
- + sélection
- Travail de fond : prendre en compte la connaissance du domaine.