Painless embeddings of distributions: The function space view

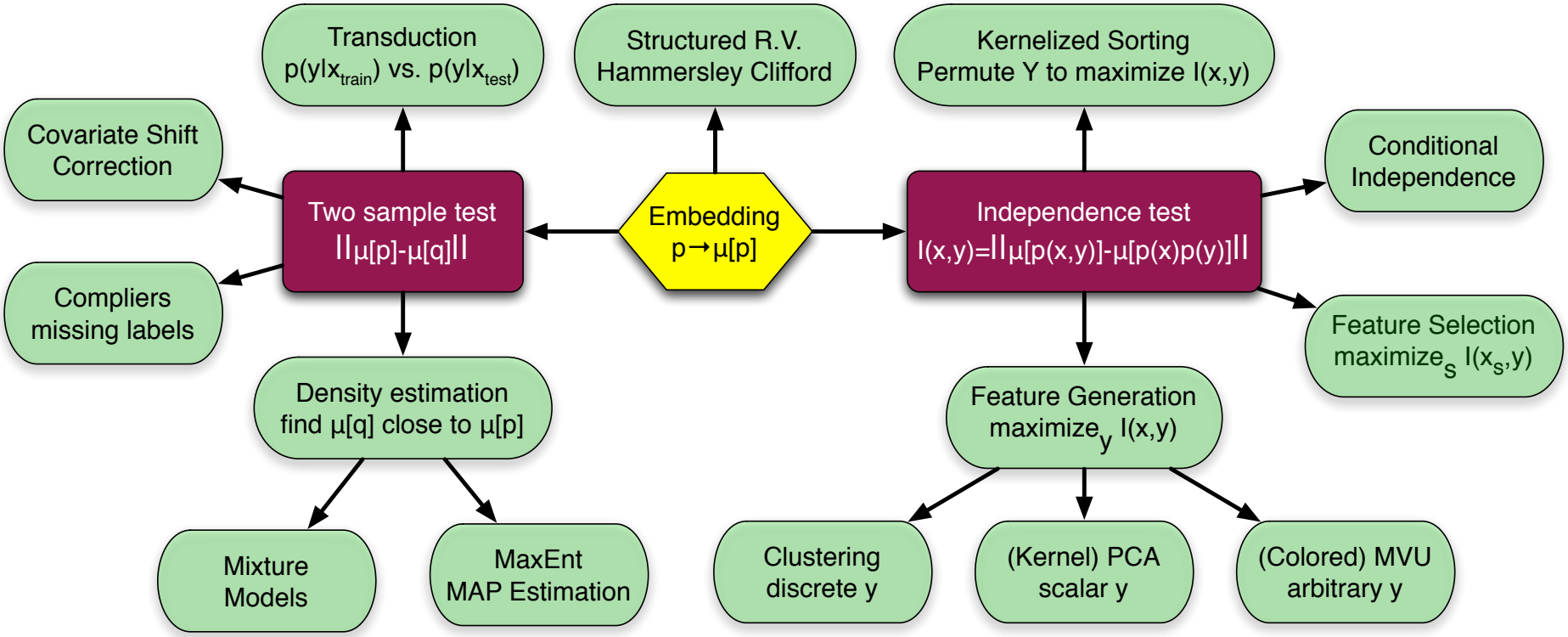
Part 2 — Applications of tr *HKHL*

Fukumizu, Gretton, Smola

RSISE, Australian National University
Statistical Machine Learning Program, NICTA
Canberra, ACT 0200 Australia
alex@smola.org

Helsinki, July 5, 2008

The Big Picture



Outline

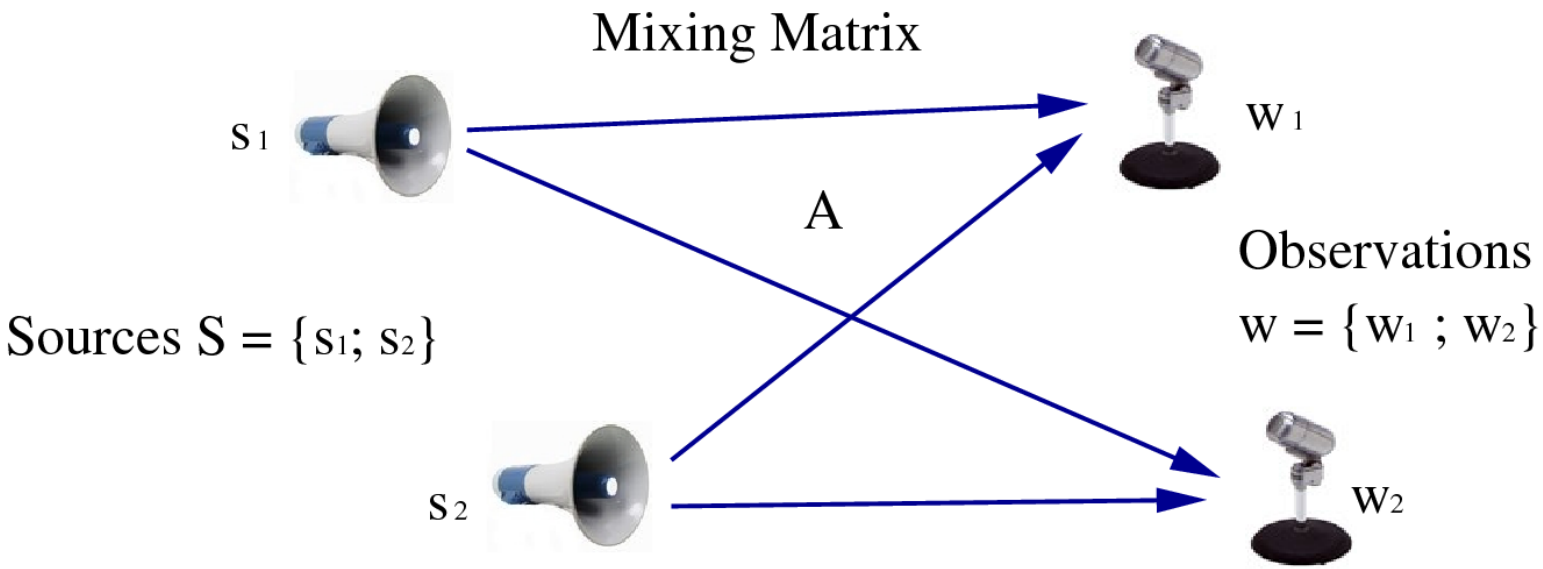
Outline

Blind Source Separation

Data

$w = Ms$, where all s_i are mutually independent.

The Cocktail Party Problem



Task

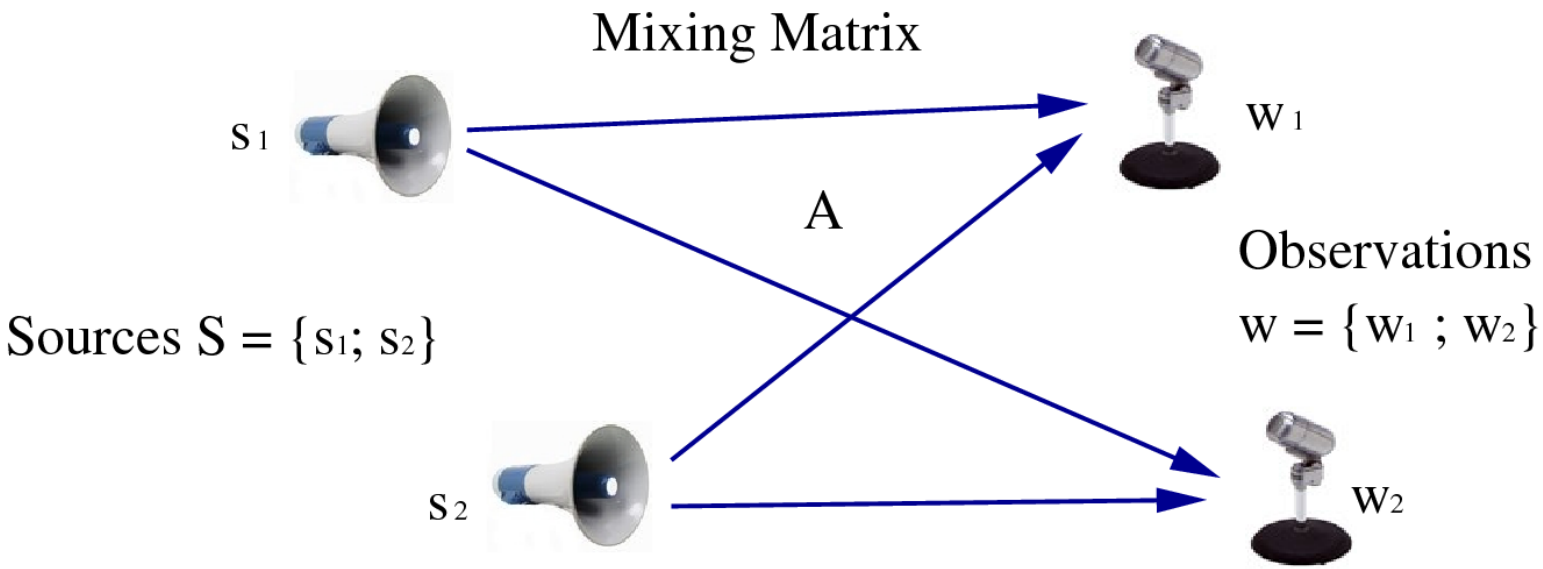
Recover the sources S and mixing matrix M given W.

Blind Source Separation

Data

$w = Ms$, where all s_i are mutually independent.

The Cocktail Party Problem



Task

Recover the sources S and mixing matrix M given W.

Basic Idea

Dependence Minimization

Find matrix M such that the coordinates of $w = Mx$ are as independent as possible.

Objective Function

$$\operatorname{tr} HKHL$$

Here K is the kernel matrix of the first coordinate of w and L is that of the second one.

Independent Component Analysis

Whitening

Rotate, center, and whiten data before separation.

Optimization

- We cannot recover scale of data anyway.
- Need to find orthogonal matrix U such that $Uw = s$.
- Optimization on the Stiefel manifold.
- Do this by a Newton method on manifolds.

Bag of Tricks

- Kernel matrix is huge $\Longrightarrow$ use reduced rank expansion

$$\operatorname{tr} H(AA^\top)H(BB^\top) = \left\|A^\top HB\right\|^2 \text{ instead of } \operatorname{tr} HKHL.$$

- Laplace kernel works best for audio (cheap to compute).

Independent Component Analysis

Whitening

Rotate, center, and whiten data before separation.

Optimization

- We cannot recover scale of data anyway.
- Need to find orthogonal matrix U such that $Uw = s$.
- Optimization on the Stiefel manifold.
- Do this by a Newton method on manifolds.

Bag of Tricks

- Kernel matrix is huge $\Longrightarrow$ use reduced rank expansion

$$\operatorname{tr} H(AA^\top)H(BB^\top) = \left\|A^\top HB\right\|^2 \text{ instead of } \operatorname{tr} HKHL.$$

- Laplace kernel works best for audio (cheap to compute).

Independent Component Analysis

Whitening

Rotate, center, and whiten data before separation.

Optimization

- We cannot recover scale of data anyway.
- Need to find orthogonal matrix U such that $Uw = s$.
- Optimization on the Stiefel manifold.
- Do this by a Newton method on manifolds.

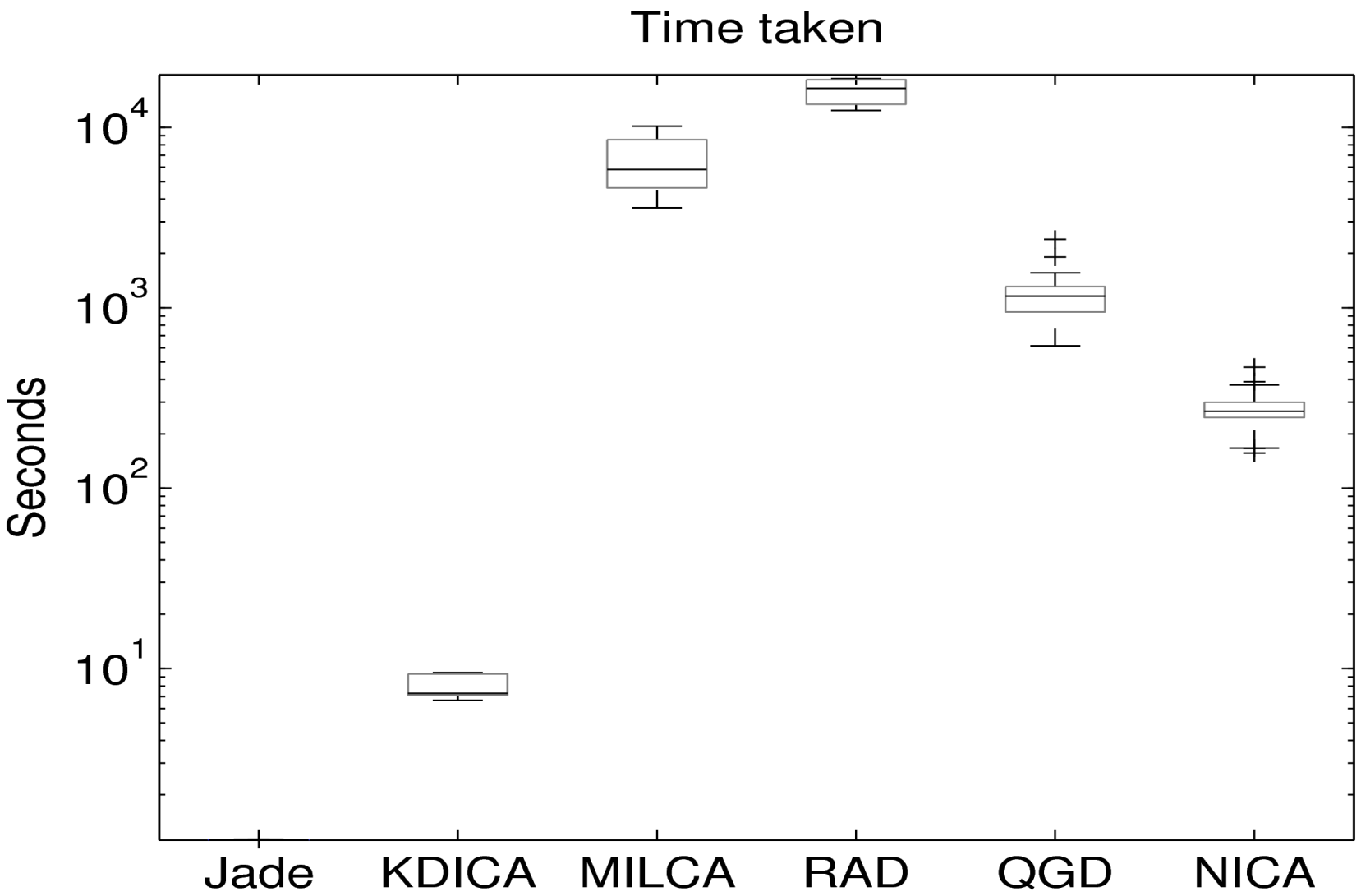
Bag of Tricks

- Kernel matrix is huge $\Longrightarrow$ use reduced rank expansion

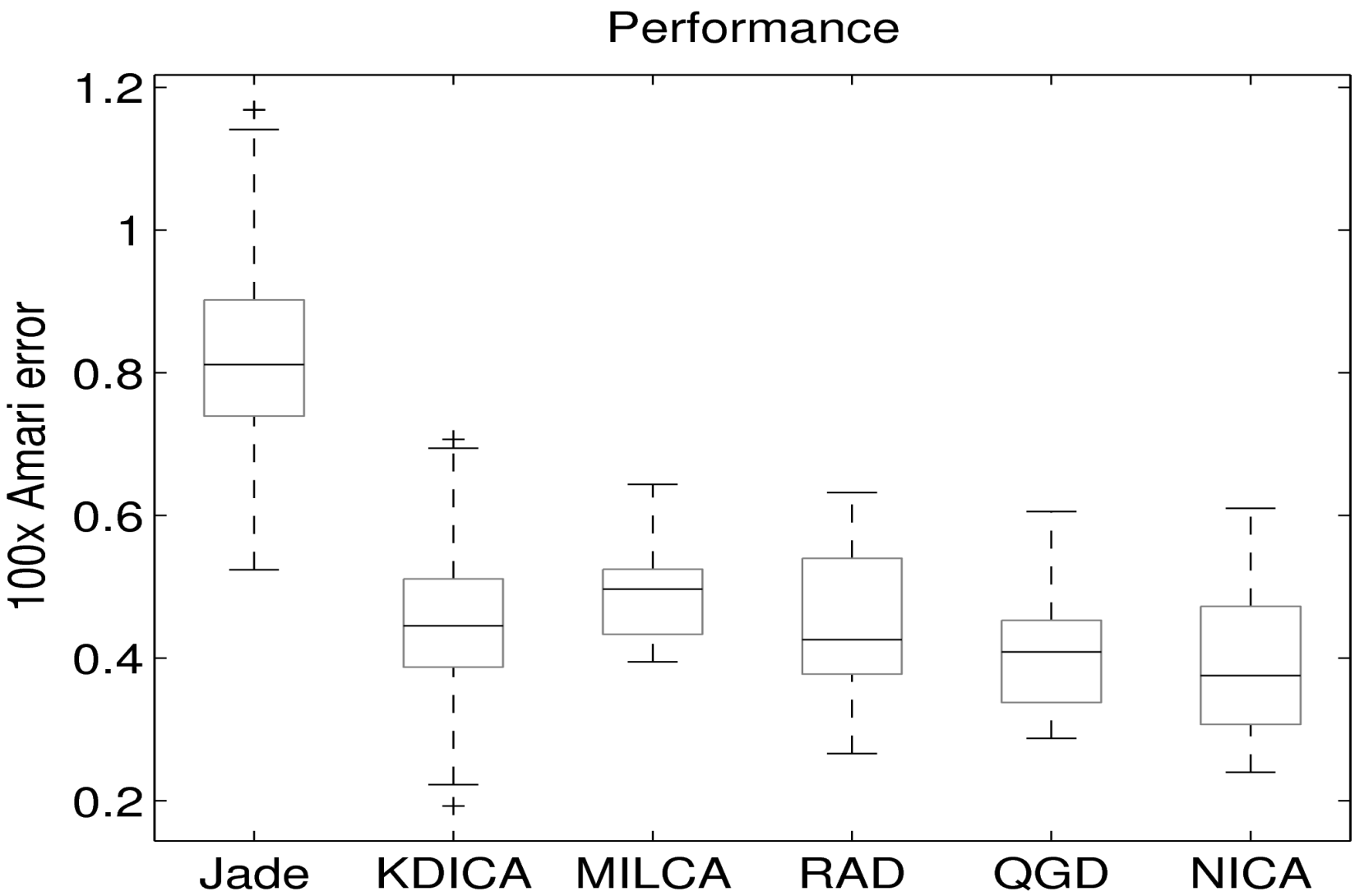
$$\operatorname{tr} H(AA^\top)H(BB^\top) = \left\|A^\top HB\right\|^2 \text{ instead of } \operatorname{tr} HKHL.$$

- Laplace kernel works best for audio (cheap to compute).

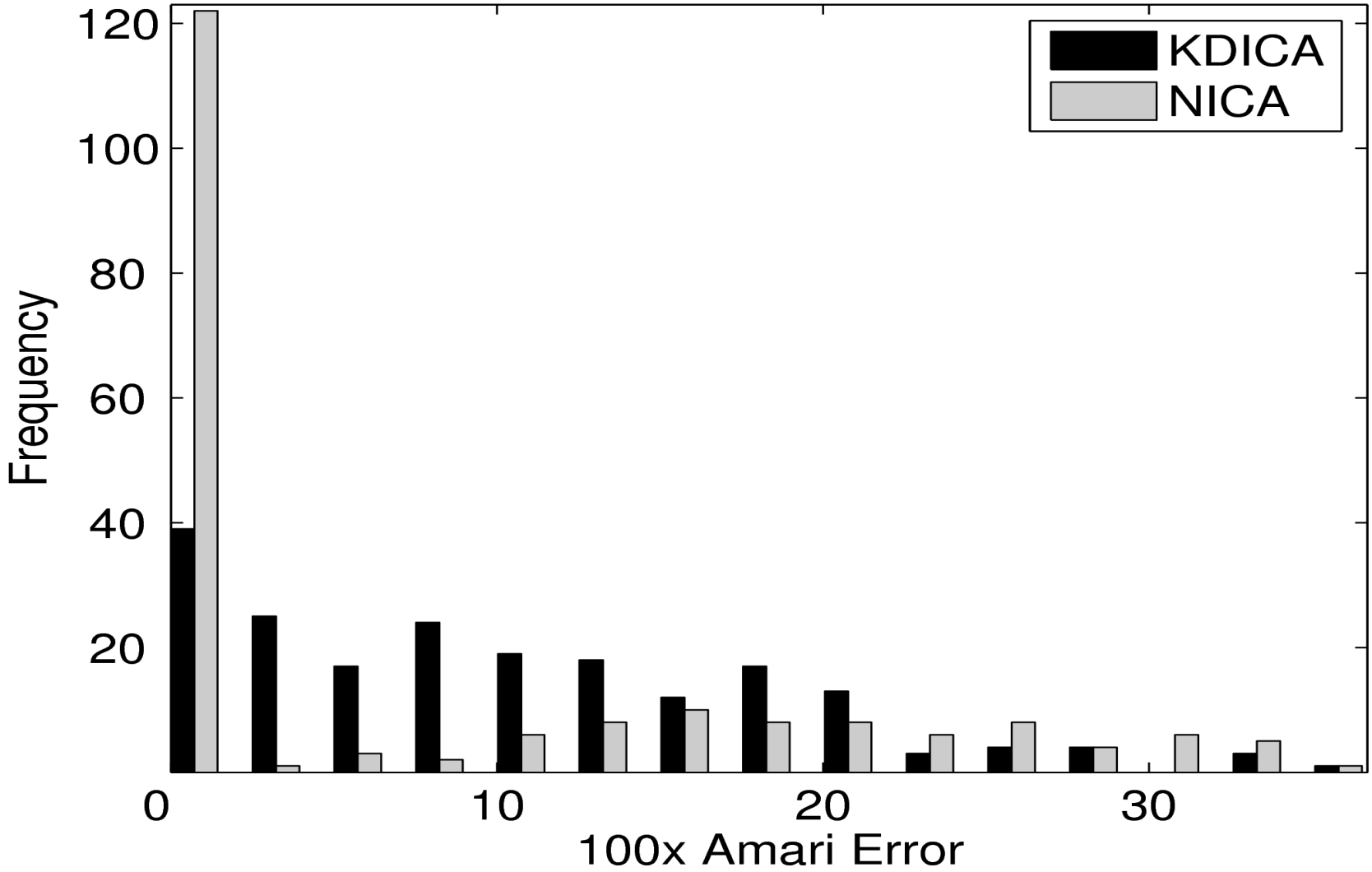
Speed


Time taken
Seconds
10^4
10^3
10^2
10^1
Jade
KDICA
MILCA
RAD
QGD
NICA

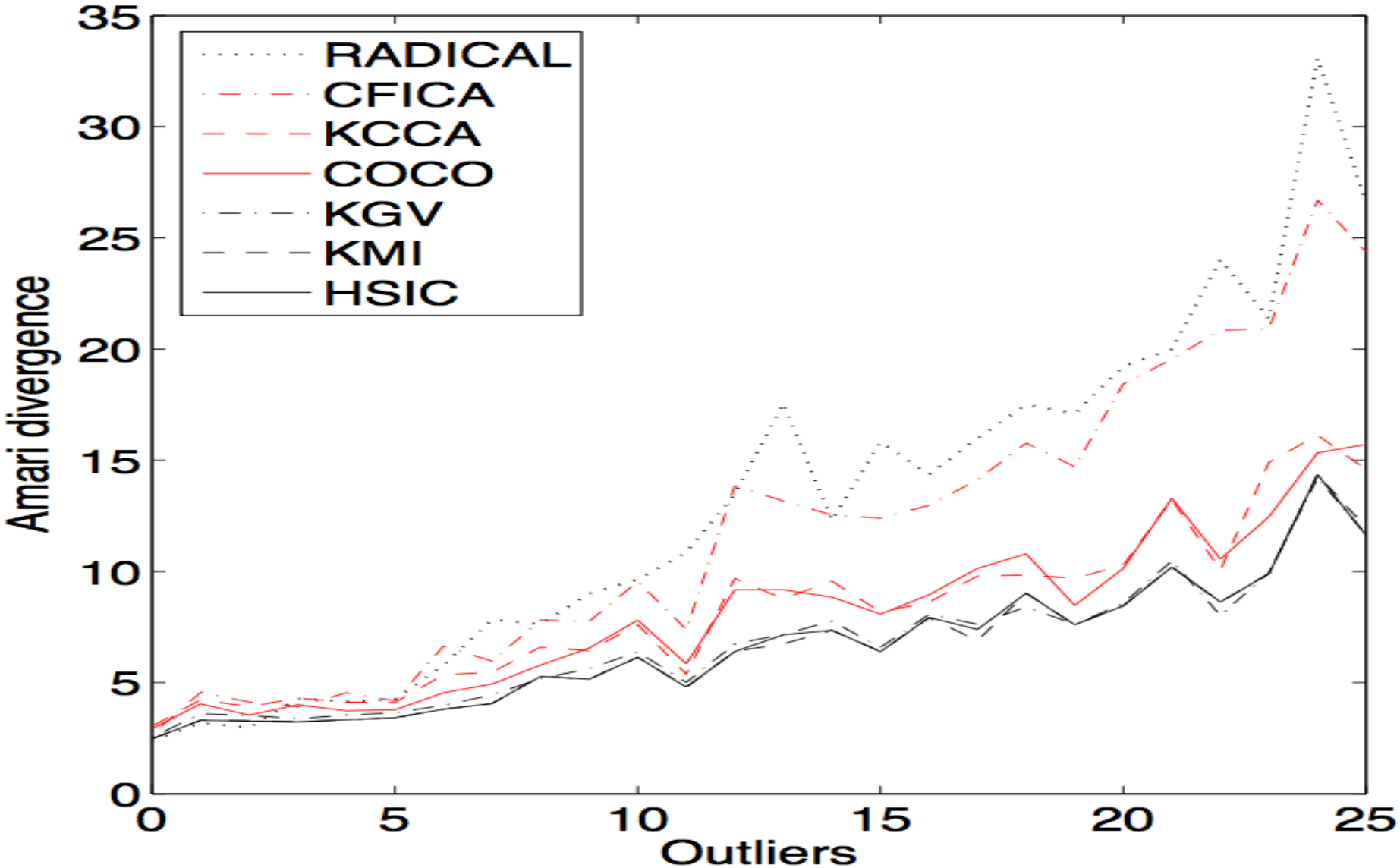
Precision


Performance
100x Amari error
1.2
1
0.8
0.6
0.4
0.2
Jade
KDICA
MILCA
RAD
QGD
NICA

Robustness with random initialization



Outlier Robustness



Extensions

Problem

For audio data or images the sequence / lattice of observations is *not* independent.

Solution

Treat the entire sequence as *one* random variable. Use the decomposition into cliques. That is, we have

$$\phi(x) = (\ldots, \phi_\tau(x_t, \ldots, x_{t+\tau}), \ldots)$$

Practical Consequence

- Use extended sequences $(x_t, \ldots, x_{t+\tau})$ and $(y_t, \ldots, y_{t+\tau})$ to construct kernel matrices K and L.
- Generalizes the TD-SEP ICA algorithms.
- Works well.

Outline

Feature Selection

The Problem

- Large number of features (e.g. genes in a microarray)
- Select a small subset of them

Basic Idea

- Given labels Y find features such that the distributions on X and Y are as dependent as possible.
- Use the Hilbert Schmidt Criterion for that.

Recursive Feature Elimination

Algorithm

- Start with full set of features
- Adjust kernel width to pick up maximum discrepancy
- Find feature which decreases dissimilarity the least
- Remove this feature
- Repeat

Applications

- Binary classification
- Multiclass
- Regression

Special Cases

Pearson Correlation

Use linear kernel and renormalize by variances

Mean Difference

Renormalize by sample size

***t*-statistic**

Renormalize by overall variance, coordinate wise

Signal to noise ratio

Renormalize by sum over variances, coordinate wise

Kernel Target Alignment

Very similar but broken centering and normalization

Recursive Feature Elimination

Perform penalized LMS regression and score by minimum value of penalized log-likelihood. Equivalent to kernel

$$K(K + \lambda \mathbf{1})^{-1}$$

Comparison to other feature selectors

Synthetic Data

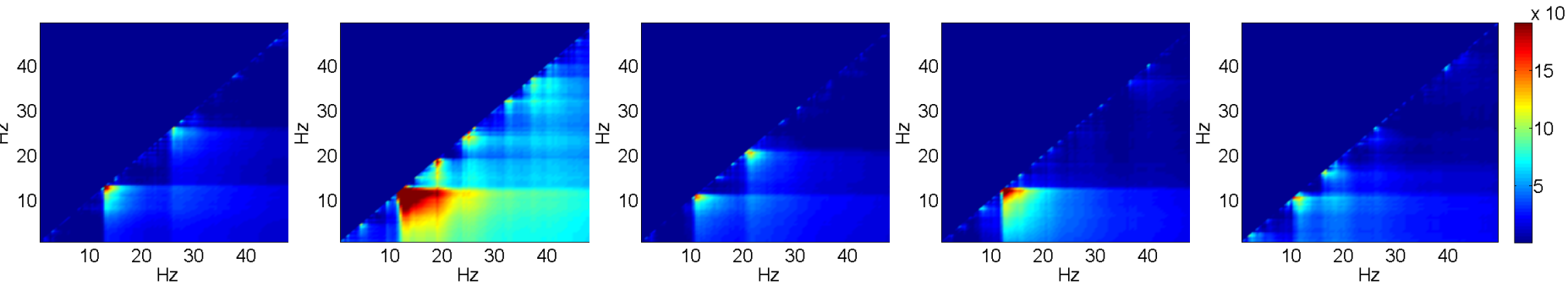
Classification error after feature selection

Method	Fisher	FSV	L_0	MI	R2W2	RFE	BAHSIC
WL-6	10 ± 4.5	2 ± 2.0	$\mathbf{0 \pm 0.0}$	6 ± 3.1	$\mathbf{0 \pm 0.0}$	$\mathbf{0 \pm 0.0}$	$\mathbf{0 \pm 0.0}$
WL-2	57 ± 3.7	58 ± 5.3	2 ± 1.3	18 ± 2.9	54 ± 6.5	2 ± 1.3	$\mathbf{1 \pm 1.0}$

Brain Computer Interface Data

Classification error on BCI data selecting frequency range.

Subject	aa	al	av	aw	ay
CSP (8-40Hz)	17.5 ± 2.5	3.1 ± 1.2	32.1 ± 2.5	7.3 ± 2.7	$\mathbf{6.0 \pm 1.6}$
CSSP	14.9 ± 2.9	2.4 ± 1.3	33.0 ± 2.7	$\mathbf{5.4 \pm 1.9}$	6.2 ± 1.5
CSSSP	$\mathbf{12.2 \pm 2.1}$	2.2 ± 0.9	31.8 ± 2.8	6.3 ± 1.8	12.7 ± 2.0
BAHSIC	13.7 ± 4.3	$\mathbf{1.9 \pm 1.3}$	$\mathbf{30.5 \pm 3.3}$	6.1 ± 3.8	9.0 ± 6.0



Microarray Feature Selection

Binary Classification (NCBI Omnibus)

Dataset	pc	snr	pam	t	m-t	lods	lin	RBF	dis	rfe
1	12.7|3	11.4|3	11.4|**4**	12.9|3	12.9|**4**	12.9|**4**	15.5|3	19.1|1	13.9|2	14.3|0
2	33.2|1	33.9|2	33.9|1	**29.5**|1	29.5|1	**27.8**|1	32.9|2	31.5|**3**	32.8|2	34.2|0
3	37.4|0	37.4|0	37.4|0	**34.6**|**6**	**34.6**|**6**	**34.6**|**6**	37.4|1	37.4|0	37.4|0	37.4|0
4	41.6|0	**38.8**|0	41.6|0	40.7|**1**	**40.7**|0	**37.8**|0	41.6|0	41.6|0	**39.7**|0	41.6|0
5	**27.8**|0	**26.7**|0	**27.8**|0	**26.7**|**2**	**26.7**|**2**	**26.7**|**2**	**27.8**|0	**27.8**|0	**27.6**|0	**27.8**|0
6	30.0|2	**25.0**|0	31.7|0	**25.0**|**5**	**25.0**|**5**	**25.0**|**5**	30.0|0	31.7|0	30.0|1	30.0|0
7	**2.0**|**6**	**2.0**|5	**2.0**|5	28.7|4	26.3|4	26.3|4	**2.0**|3	**2.0**|4	30.0|0	**2.0**|0
8	3.3|3	**0.0**|4	**0.0**|4	**0.0**|4	3.3|**6**	3.3|**6**	3.3|2	3.3|1	6.7|2	**0.0**|0
9	**10.0**|**6**	**10.0**|**6**	**8.7**|4	34.0|5	37.7|**6**	37.7|**6**	12.0|3	**10.0**|5	12.0|1	**10.0**|0
10	**16.0**|2	**18.0**|2	**14.0**|2	**14.0**|8	22.0|**9**	22.0|**9**	**16.0**|2	**16.0**|0	**18.0**|0	32.5|0
11	12.9|5	12.9|5	12.9|5	19.5|0	22.1|0	33.6|0	**11.2**|4	**9.5**|**6**	16.0|4	19.0|0
12	30.3|2	36.0|2	31.3|2	26.7|**3**	35.7|0	35.7|0	**18.7**|1	35.0|0	33.0|1	29.7|0
13	8.4|5	11.1|0	7.0|5	22.1|3	27.9|**6**	15.4|1	7.0|2	9.6|0	11.1|0	**4.3**|1
14	20.8|1	20.8|1	20.2|0	20.8|**3**	20.8|**3**	20.8|**3**	20.8|0	20.2|0	**19.7**|0	20.8|0
15	**0.0**|7	0.7|1	**0.0**|5	4.0|1	0.7|**8**	0.7|**8**	**0.0**|3	**0.0**|2	2.0|2	**0.0**|1
best	5|2	7|1	6|1	6|6	4|10	5|9	6|0	6|2	4|0	6|0
ℓ_2	16.9	20.9	17.3	43.5	50.5	50.3	**13.2**	22.9	35.4	26.3

Multiclass Classification (NCBI Omnibus)

Data	16	17	18	19	20	21	22	23	24	25	26	27	28	best	ℓ_2
lin	36.7|1	**0.0**|3	5.0|**3**	10.5|6	35.0|3	**37.5**|**6**	**18.6**|1	**40.3**|**3**	**28.1**|**3**	**26.6**|**6**	**5.6**|**6**	27.9|7	45.1|1	7|6	**32.4**
RBF	**33.3**|**3**	5.1|4	**1.7**|**3**	**7.2**|**9**	33.3|0	40.0|1	22.1|0	72.5|0	39.5|0	**24.7**|4	**5.6**|**6**	22.1|**10**	**21.5**|3	6|5	37.9
dis	**29.7**|2	28.8|**5**	6.7|0	**8.2**|**9**	**29.4**|**7**	**38.3**|4	43.4|**4**	66.1|0	40.8|0	38.9|4	7.6|1	**8.2**|8	31.6|3	5|4	51.0

Outline

Feature Extraction

Goal

Given a set X find a set $Y \subseteq \mathcal{Y}$ such that the dependence between X and Y is maximized.

$$Y^* = \operatorname*{argmax}_{Y \in \mathcal{Y}} \operatorname{tr} \bar{K} L(Y) \text{ subject to constraints on } Y$$

Here we define $\bar{K} = HKH$ as the centered kernel.

Information Extraction

Find set Y which retains maximum information about X.

Two Problems

- Efficient algorithm
- How to define $L(Y)$

Special Cases

(Kernel) Principal Component Analysis

$\mathcal{Y} = \mathbb{R}$ and require $\|y\| = 1$.

Discriminant Analysis

Projections of X onto Y with constraints on projection.

Clustering

$\mathcal{Y} = [1, \ldots, k]$. We can add structure if we want.

Segmentation and time-series PCA

Maximize dependency for random variables with Markovian structure. $\phi(x)$ decomposes along maximal cliques.

Independent Features

Maximize dependency while retaining independence from known disturbances.

More Details

Permutation Matrix

$$L = \Pi A \Pi^\top \text{ where } \Pi 1 = 1 \text{ and } \Pi_{ij} \in \{0, 1\}$$

Π_{ij} assigns observation i to cluster j for $\Pi_{ij} = 1$.

Cluster Kernel Matrix

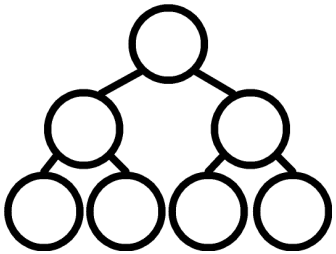
Design A such that a desired similarity measure is satisfied.

Examples

- n-means requires diagonal A.
  Values of the diagonal entries specify the cluster size
- Hierarchies, chains, rings with suitable label kernel matrix as imposed by the algorithm.
- By relaxing $L(Y)$ to subspaces we get spectral clustering

Structure

Hierarchies

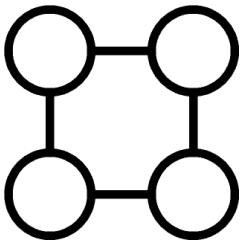


$$A = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \otimes \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$$

Chain

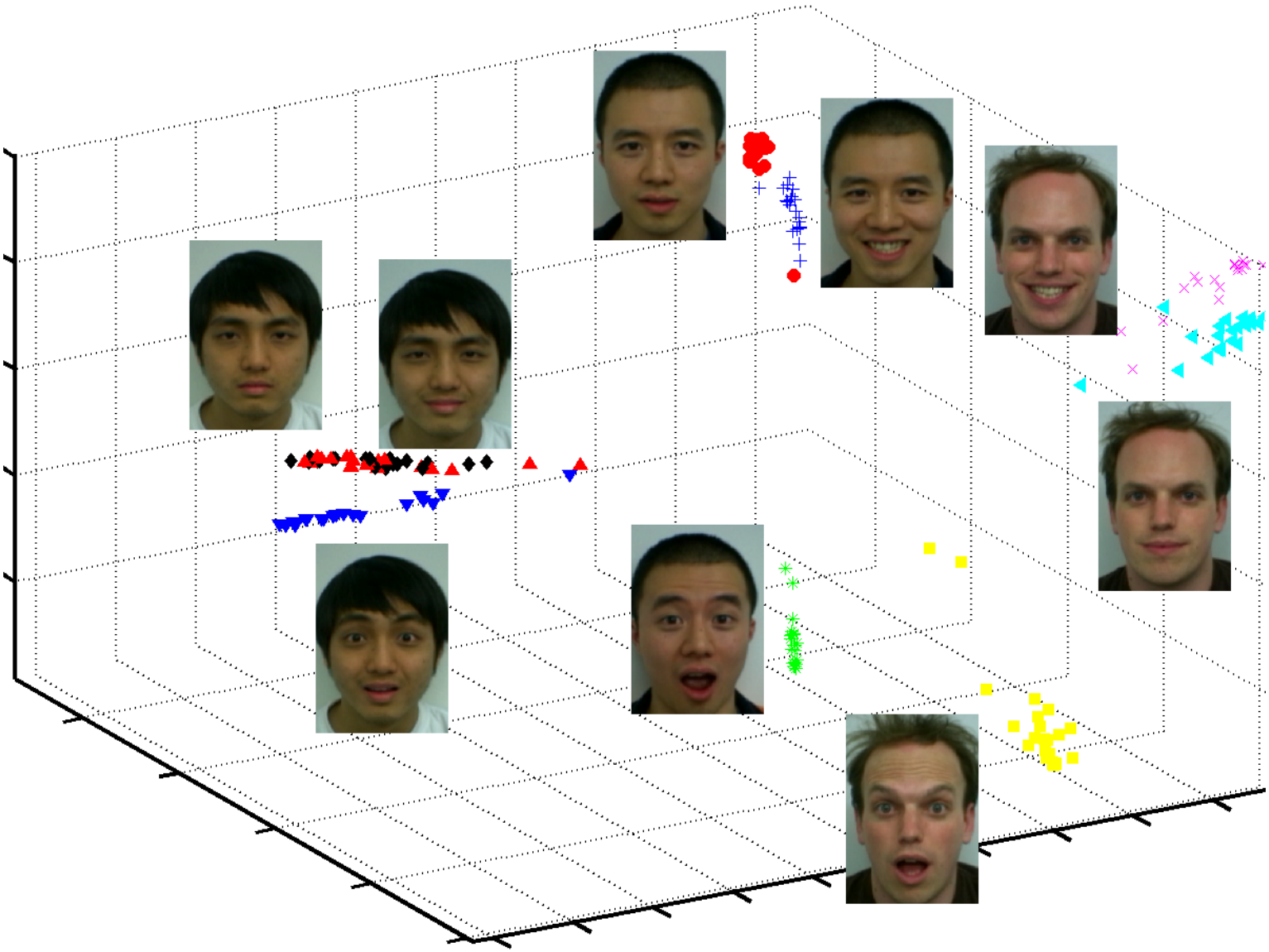
$$A = \begin{bmatrix} 2 & 1 & 0 \\ 1 & 2 & 1 \\ 0 & 1 & 2 \end{bmatrix}$$

Ring

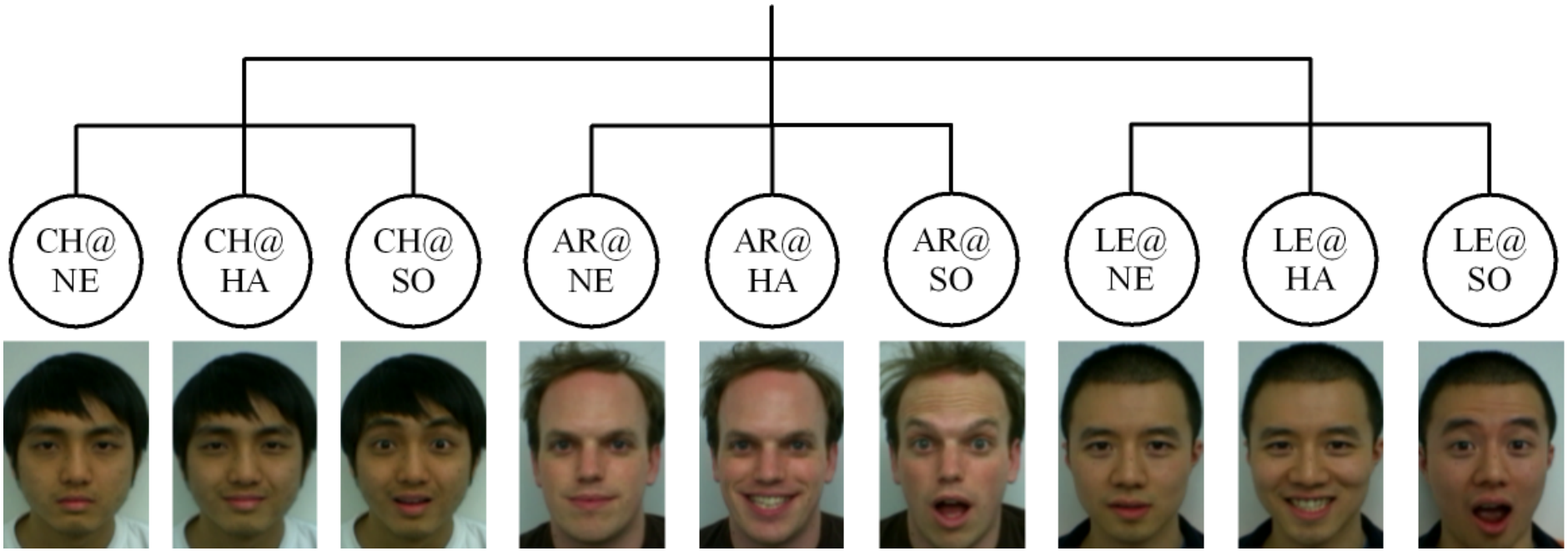


$$A = \begin{bmatrix} 2 & 1 & 0 & 1 \\ 1 & 2 & 1 & 0 \\ 0 & 1 & 2 & 1 \\ 1 & 0 & 1 & 2 \end{bmatrix}$$

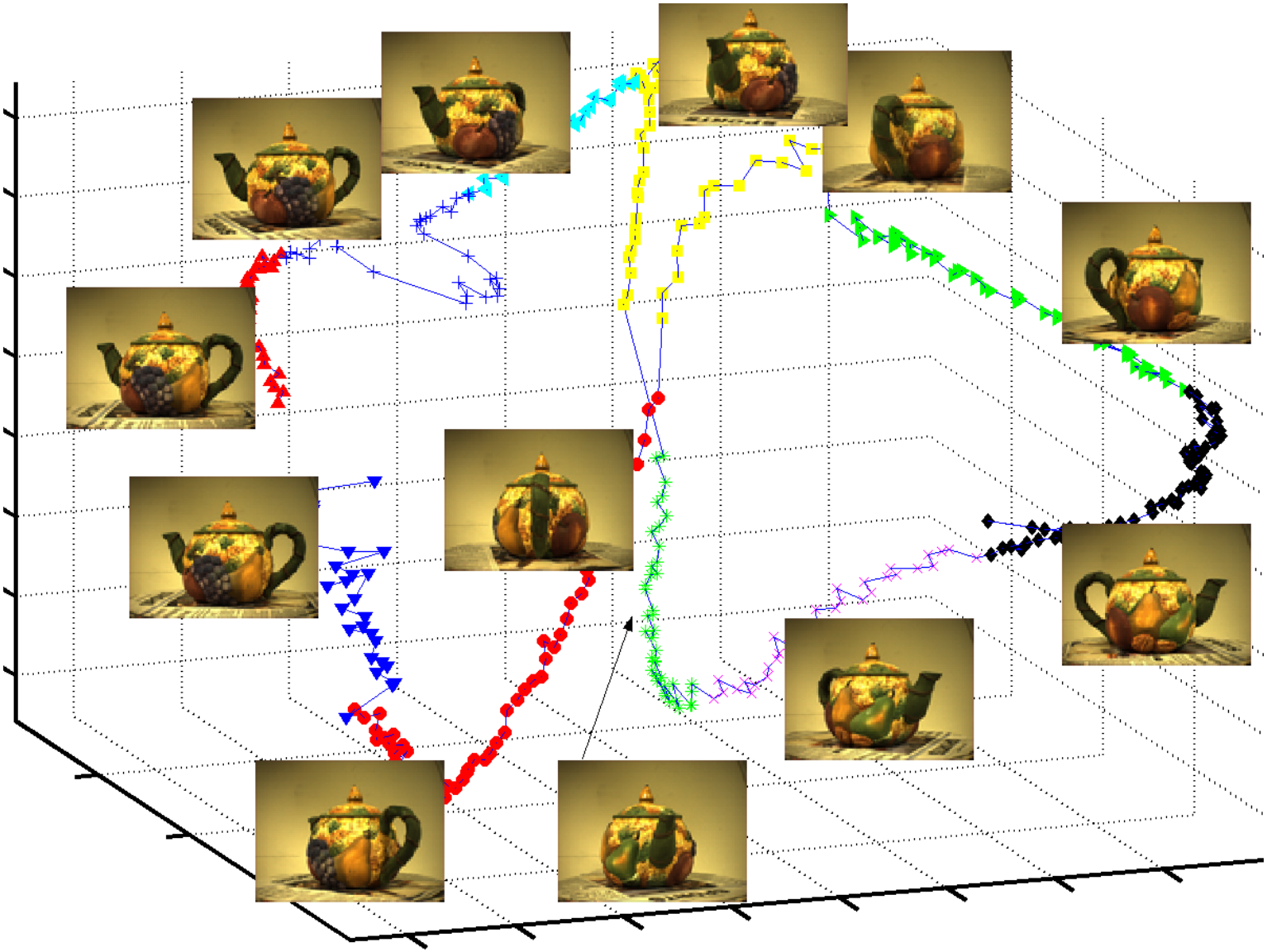
Hierarchical Image Data



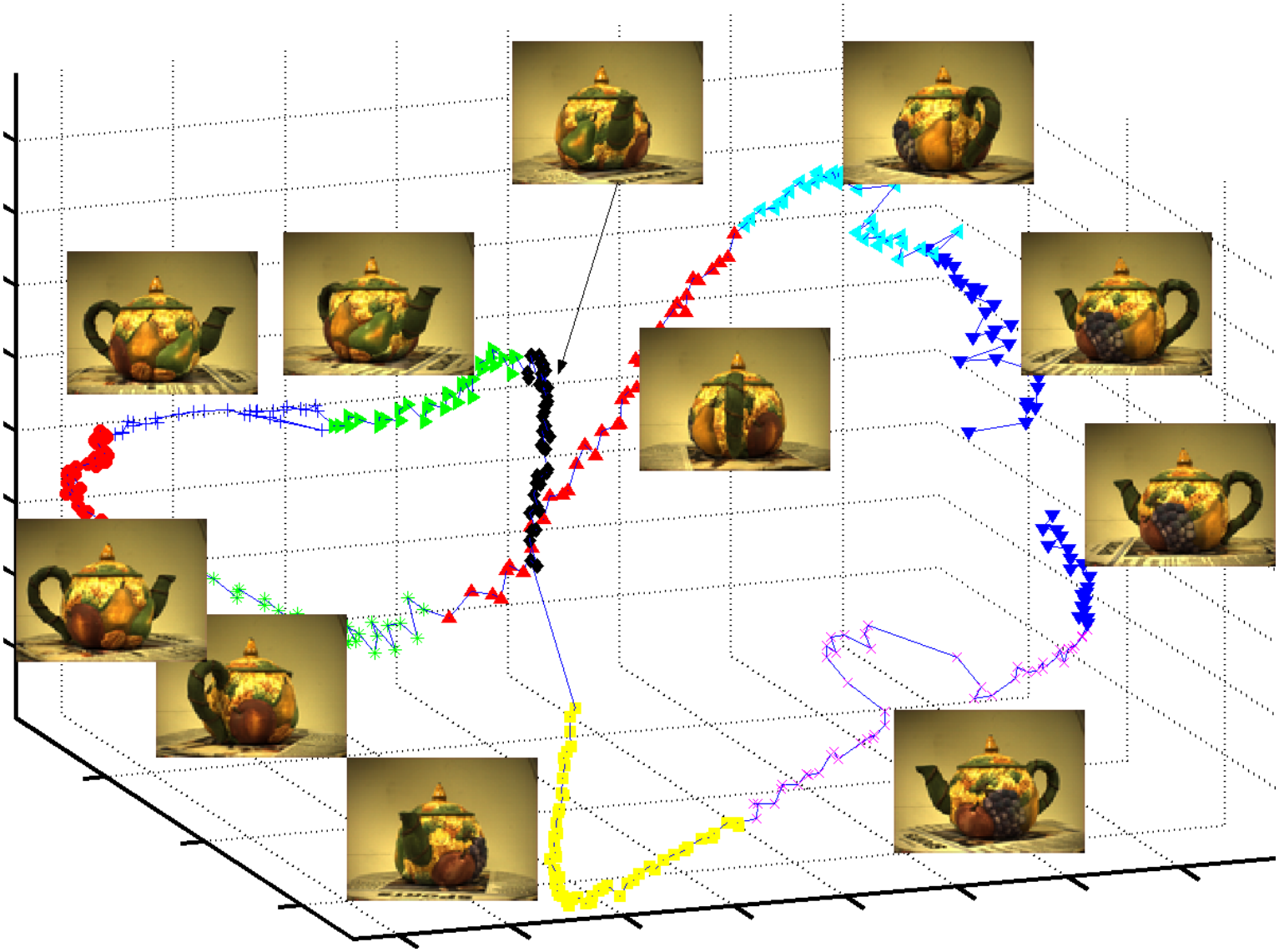
Hierarchical Image Data



Weinberger's Teapot (k-means)



Weinberger's Teapot (ring)



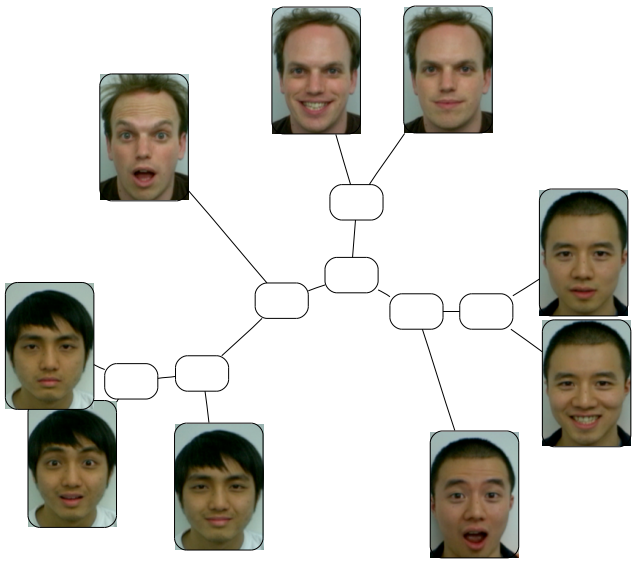
Numerical Taxonomy

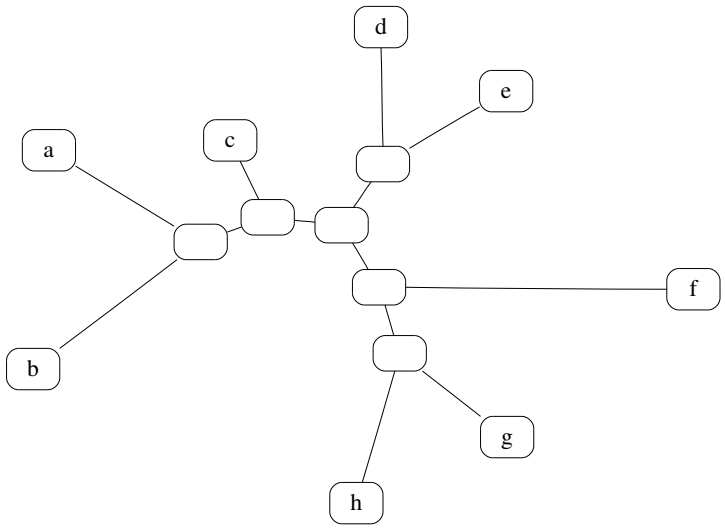
Learning the Kernel

Find a suitable kernel *in addition* to the cluster assignment.

Optimization

- Continuous relaxation of the assignment problem
- Find suitable kernel matrix subject to normalization
- Approximate this matrix with a tree metric (numerical taxonomy algorithms are plenty).
- Iterate . . .





Representative Clusters on NIPS Dataset

a	b	c	d	e	f	g	h
neurons	chip	memory	network	training	state	function	data
cells	circuit	dynamics	units	recognition	learning	error	model
model	analog	image	learning	network	policy	algorithm	models
cell	voltage	neural	hidden	speech	action	functions	distribution
visual	current	hopfield	networks	set	reinforcement	learning	gaussian
neuron	figure	control	input	word	optimal	theorem	likelihood
activity	vlsi	system	training	performance	control	class	parameters
synaptic	neuron	inverse	output	neural	function	linear	algorithm
response	output	energy	unit	networks	time	examples	mixture
firing	circuits	capacity	weights	trained	states	case	em
cortex	synapse	object	error	classification	actions	training	bayesian
stimulus	motion	field	weight	layer	agent	vector	posterior
spike	pulse	motor	neural	input	algorithm	bound	probability
cortical	neural	computational	layer	system	reward	generalization	density
frequency	input	network	recurrent	features	sutton	set	variables
orientation	digital	images	net	test	goal	approximation	prior
motion	gate	subjects	time	classifier	dynamic	bounds	log
direction	cmos	model	back	classifiers	step	loss	approach
spatial	silicon	associative	propagation	feature	programming	algorithms	matrix
excitatory	implementation	attractor	number	image	rl	dimension	estimation

Outline

Nonparametric Sorting

Basic Idea

Given two sets of observations $X = \{x_1, \ldots, x_m\}$ and $Y = \{y_1, \ldots, y_m\}$, find a permutation matrix π such that $(x_i, y_{\pi(i)})$ is maximally dependent.

Optimization Problem

$$\pi^* = \operatorname*{argmax}_{\pi} \operatorname{tr} \bar{K} \pi^\top L \pi$$

Sorting as a special case

For scalar x_i and y_i and a linear kernel on both sets, we can rewrite the optimization problem

$$\pi^* = \operatorname*{argmax}_{\pi} \left(\bar{x}^\top \pi \bar{y} \right)^2$$

This is maximized by sorting X and Y.

Optimization

Convexity

The objective function $\operatorname{tr} \bar{K}\pi^\top L\pi$ is convex in π.

DC Programming

Compute successive linear lower bounds and maximize

$$\pi \leftarrow \operatorname*{argmax}_{\pi'} \operatorname{tr}\left[\bar{K}\pi^\top L\right]\pi'$$

This will converge to a local maximum. Initialization via clustering and KPCA.

Aligning images to a grid

Pairing image parts (140 out of 320)

Layout to text

Outline

Basic Idea

Dependence Maximization

Maximize dependence between X and Y via Euclidean embedding, while preserving *local* distance information.

Optimization Problem

$$\underset{L}{\text{maximize}} \operatorname{tr} HKHL$$

$$\text{subject to } L_{ii} + L_{jj} - 2L_{ij} = D_{ij} \text{ for some subset } (i,j) \in N$$

Lots of variants possible to preserve metric locally. See Weinberger et al. for lots of details.

Special Case

If we pick $K = \mathbf{1}$ we obtain Maximum Variance Unfolding. Retain as much information as possible about each observation.

Examples

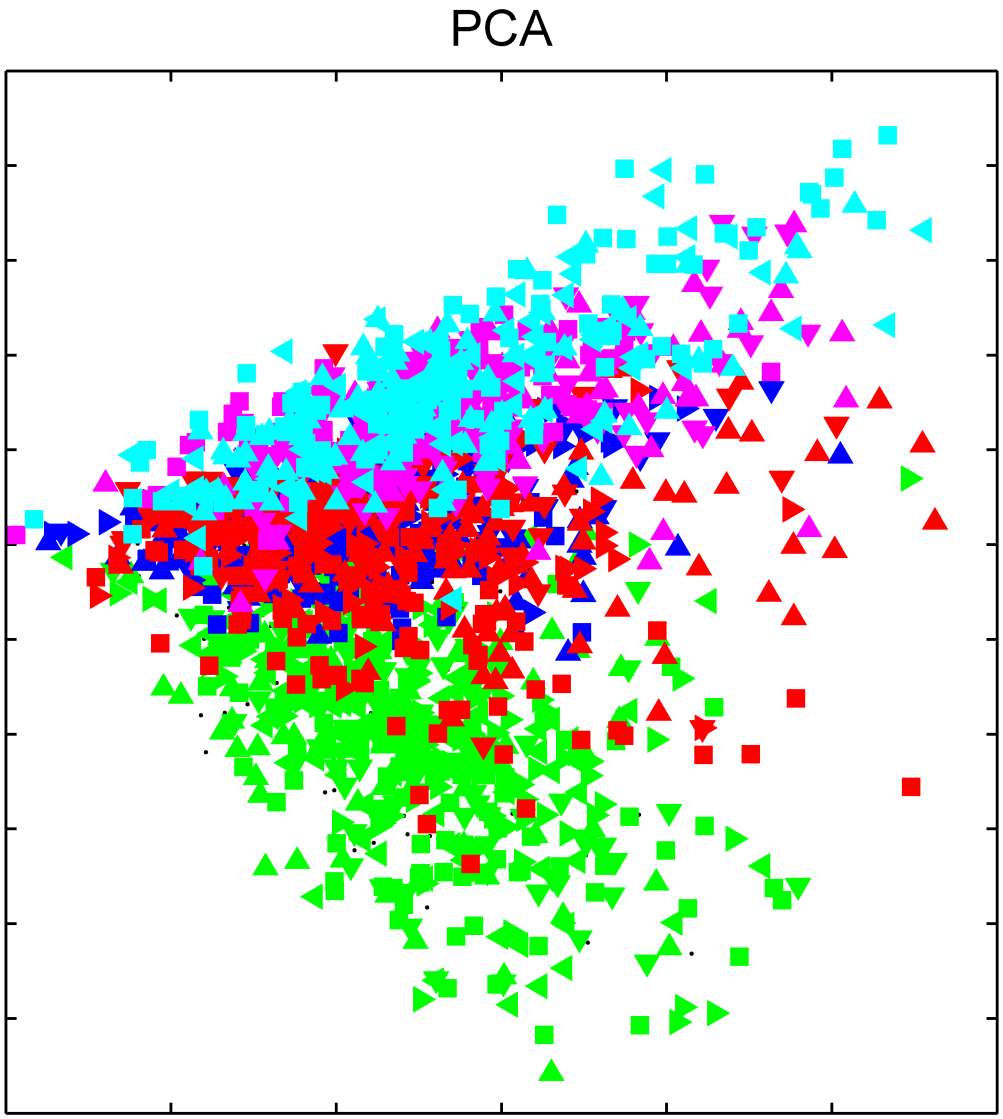
Newsgroups 20

Find low dimensional representation while locally preserving distance with respect to bag of words features. Use newsgroup labels.

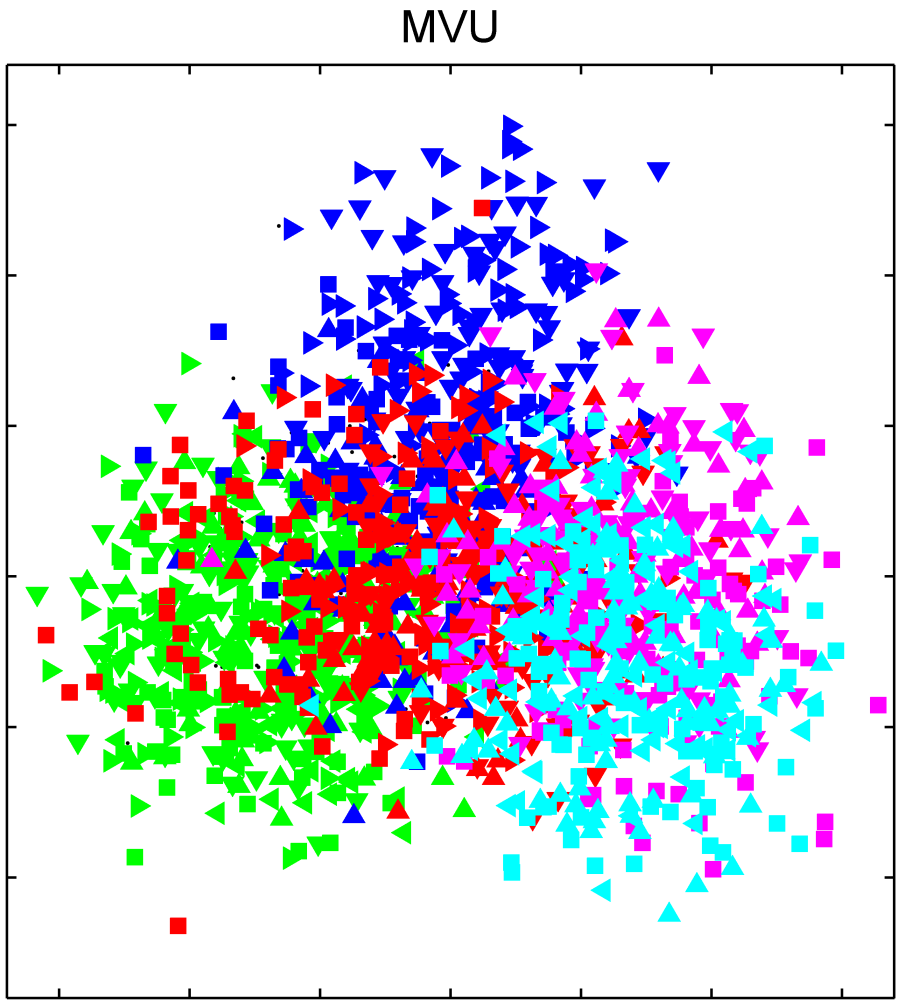
NIPS Papers

Find low dimensional representation while locally preserving distance with respect to bag of words features. Use kernel between Author names as side information.

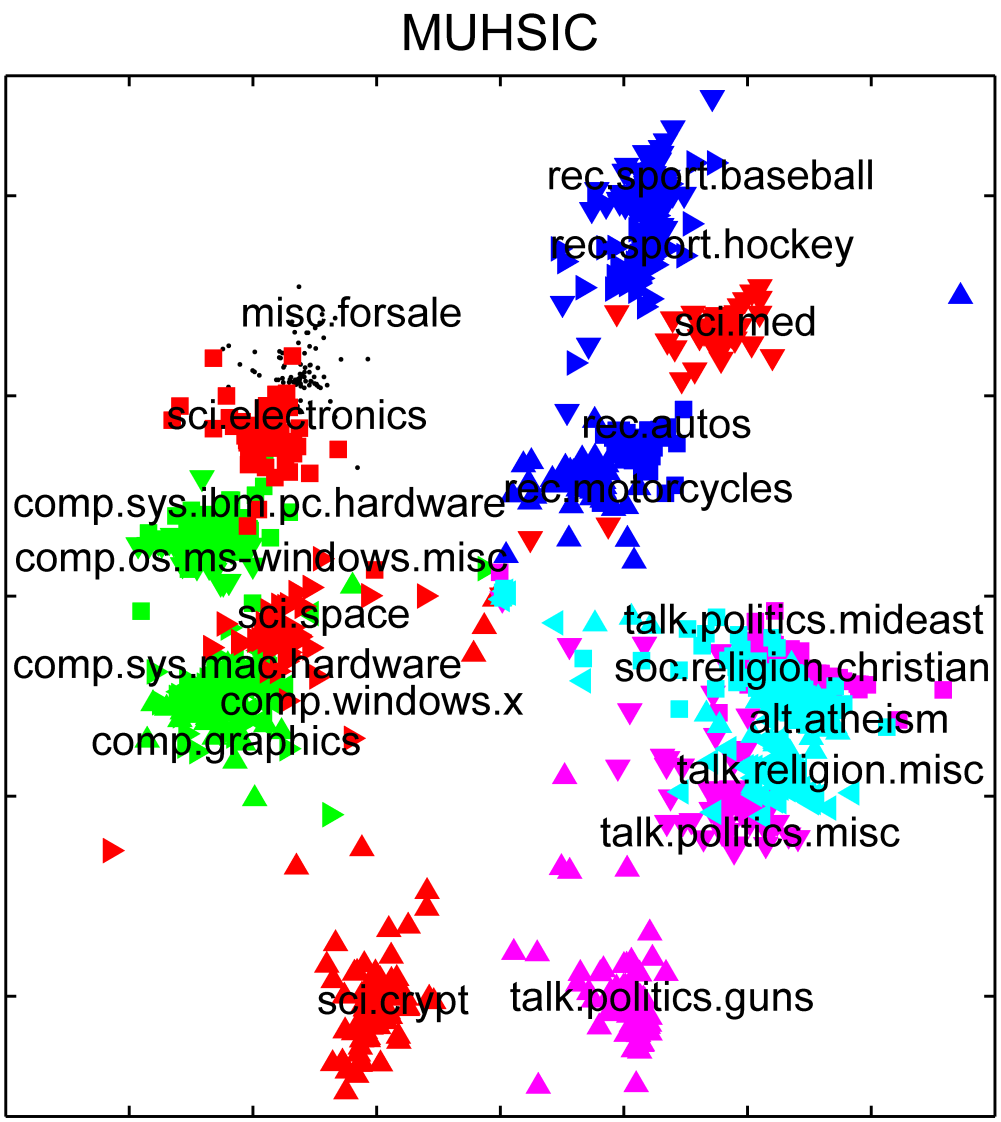
Newsgroups: PCA



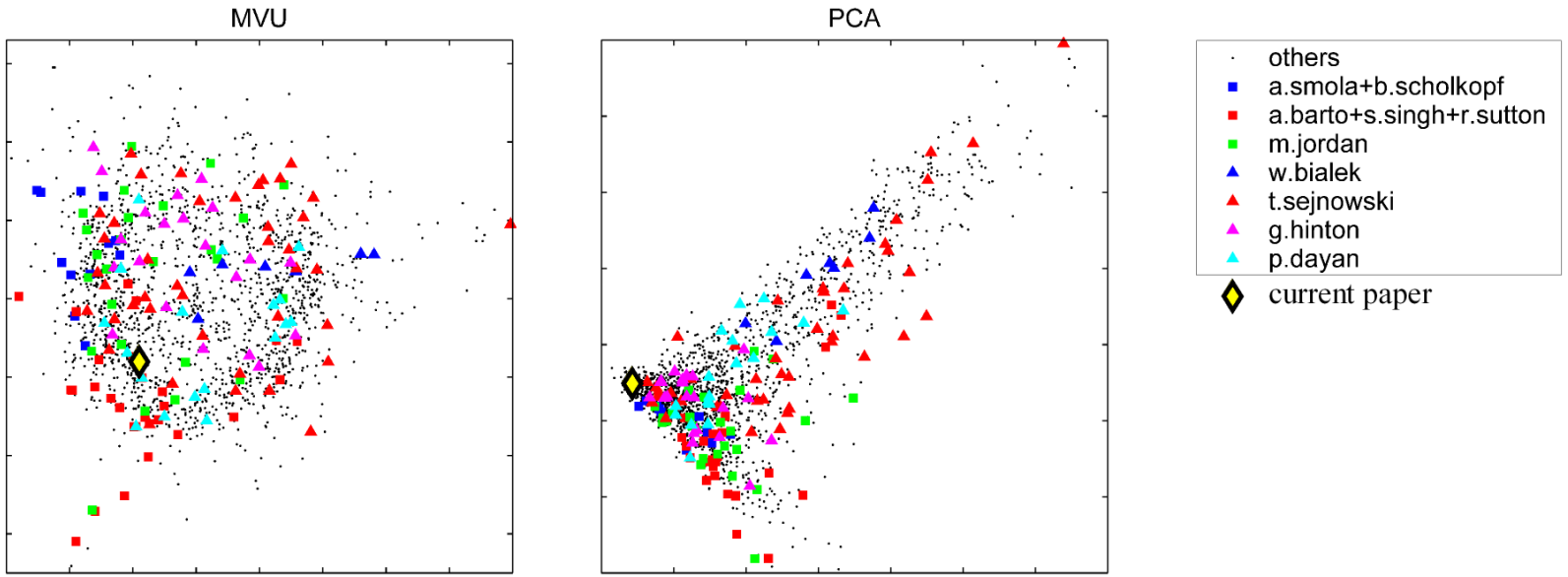
Newsgroups: MVU



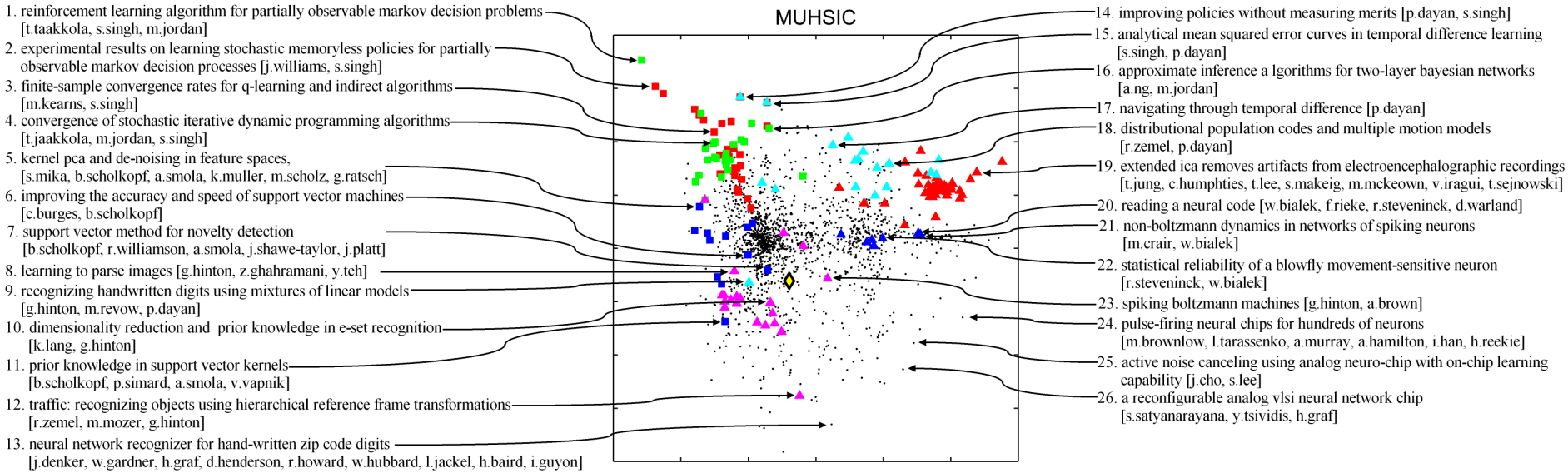
Newsgroups: MUHSIC



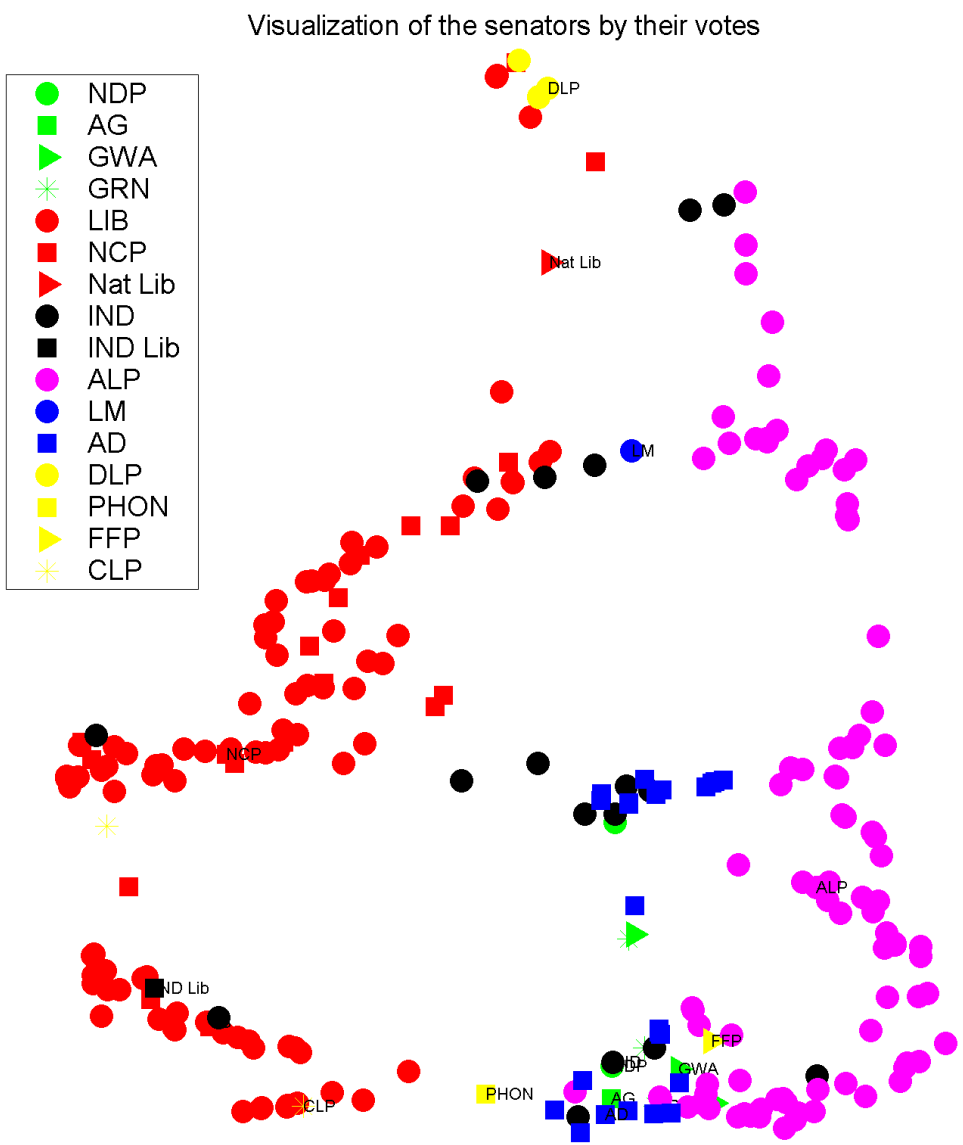
NIPS: PCA and MVU



NIPS: Maximum Variance Unfolding



Australian Senators



Summary

1. Independent Component Analysis
2. Feature Selection
3. Clustering and Feature Extraction
4. Nonparametric Sorting
5. Colored Maximum Variance Unfolding

Some References

ICASSP'03 Kernel Mutual Information

JMLR'05 Independence Measures via Kernels

ALT'05 Dependence Measure

ISMB'06 Two Sample Test for Microarrays

NIPS'06 Two Sample Test

NIPS'06 Covariate Shift Correction

ISMB'07 Feature Selection for Microarrays

ICML'07 Clustering connection

ICML'07 Feature Selection

NIPS'07 Colored Maximum Variance Unfolding

NIPS'07 Independence Test

JMLR'08 Feature Selection

ICML'08 Density Estimation

ICML'08 Estimation with summary label information