To appear at NIPS 2013

Similarity Component Analysis

Fei Sha
U. of Southern California

Joint work with Soravit Changpinyo and Kuan Liu

MLA 2013, Fudan U (Nov 2, 2013)

Similarity, distance, and metric

Profusely used conceptual tools

Nearest neighbor classifier: distances based classification

Kernel methods: inner products interpreted as similarity

Manifold learning: latent structures in Euclidean metric space

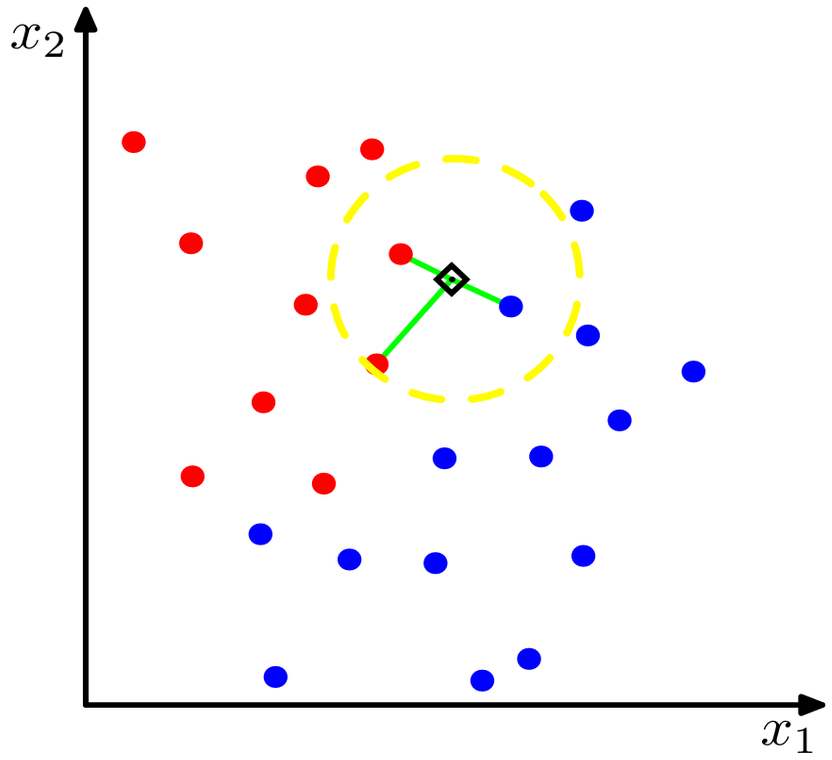
Many many more examples...

Motivation for Metric Learning

Nearest neighbor classifier

Cover and Hart (1967):

NNC classification error is at most twice of Bayes's optimal's



(a)

But how to measure the distance?

Related but different

Similarity

Non-negative and symmetric

Self-similarity is maximal

Distance (or dissimilarity)

Non-negative and symmetric

Self-dissimilarity is minimal

$$s : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}^+ \cup 0$$

$$s(\boldsymbol{x}, \boldsymbol{x}') \leq \min(s(\boldsymbol{x}, \boldsymbol{x}), s(\boldsymbol{x}', \boldsymbol{x}'))$$

$$d : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}^+ \cup 0$$

$$d(\boldsymbol{x}, \boldsymbol{x}) = 0, \ \forall \ \boldsymbol{x}$$

Linear relation

If self-similarity is the same, then *s* can be converted to *d*.

If *d* is bounded, then *d* can be converted to *s*.

Focus of many learning problems

Metric

A special distance function that satisfies the ***triangle*** inequality

$$d_m(\boldsymbol{x}, \boldsymbol{y}) \leq d_m(\boldsymbol{x}, \boldsymbol{z}) + d_m(\boldsymbol{y}, \boldsymbol{z})$$

Ex: the discrete metric from labels

$$d_m(\boldsymbol{x}, \boldsymbol{y}) = \mathbb{1}[\mathsf{label}(\boldsymbol{x}) \neq \mathsf{label}(\boldsymbol{y})]$$

Learning metric

Respect label-induced metric

$$d(\boldsymbol{x}_i, \boldsymbol{x}_j) \leq d(\boldsymbol{x}_i, \boldsymbol{x}_k) \quad \text{if} \quad \mathsf{label}(x_i) = \mathsf{label}(x_j) \neq \mathsf{label}(x_k)$$

Examples of existing work

Metric learning with side information: [Xing, Jordan and Russell 02]

Information-theoretical metric learning: [Davis, Kulis, Jain, Sra and Dhillon 07]

Large-margin metric learning: [Weinberger and Saul 06]

Many, many more...

Learning parameterized metric

Mahalanobis metric

Linear parameterization: amenable to convex optimization.

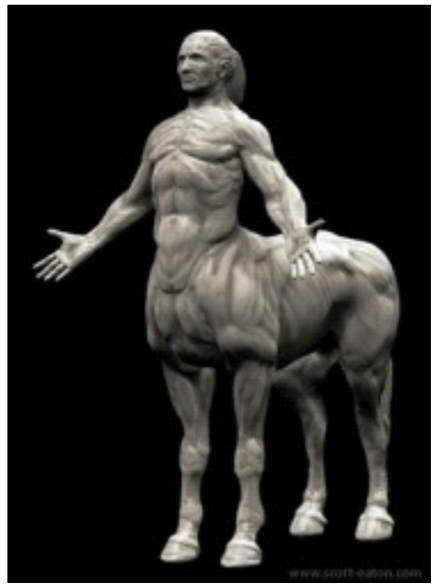
$$d(\boldsymbol{x}_i, \boldsymbol{x}_j) = (\boldsymbol{x}_i - \boldsymbol{x}_j)^{\mathrm{T}} \boldsymbol{M} (\boldsymbol{x}_i - \boldsymbol{x}_j)$$

New representation:

$$d(\boldsymbol{x}_i, \boldsymbol{x}_j) = \|\boldsymbol{z}_i - \boldsymbol{z}_j\|_2^2 \quad \text{with} \quad \boldsymbol{z}_i = \boldsymbol{M}^{1/2} \boldsymbol{x}$$

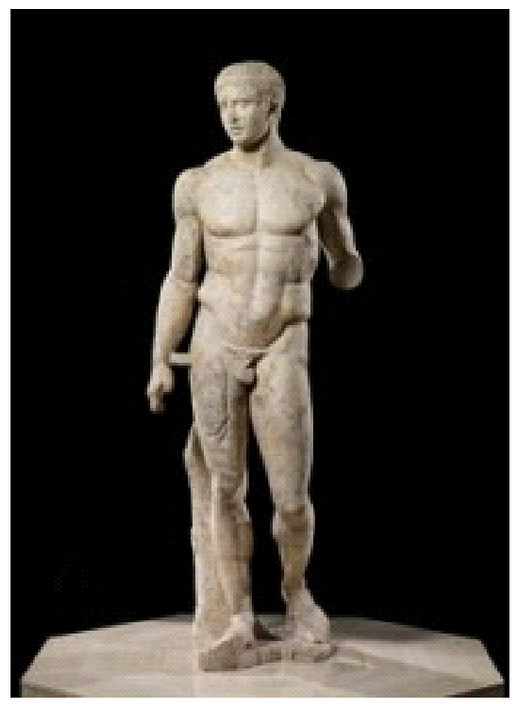
NB: *M* is constrained to be positive (semi)definite to be a (semi)metric

Metric learning is insufficient



Similar

Similar

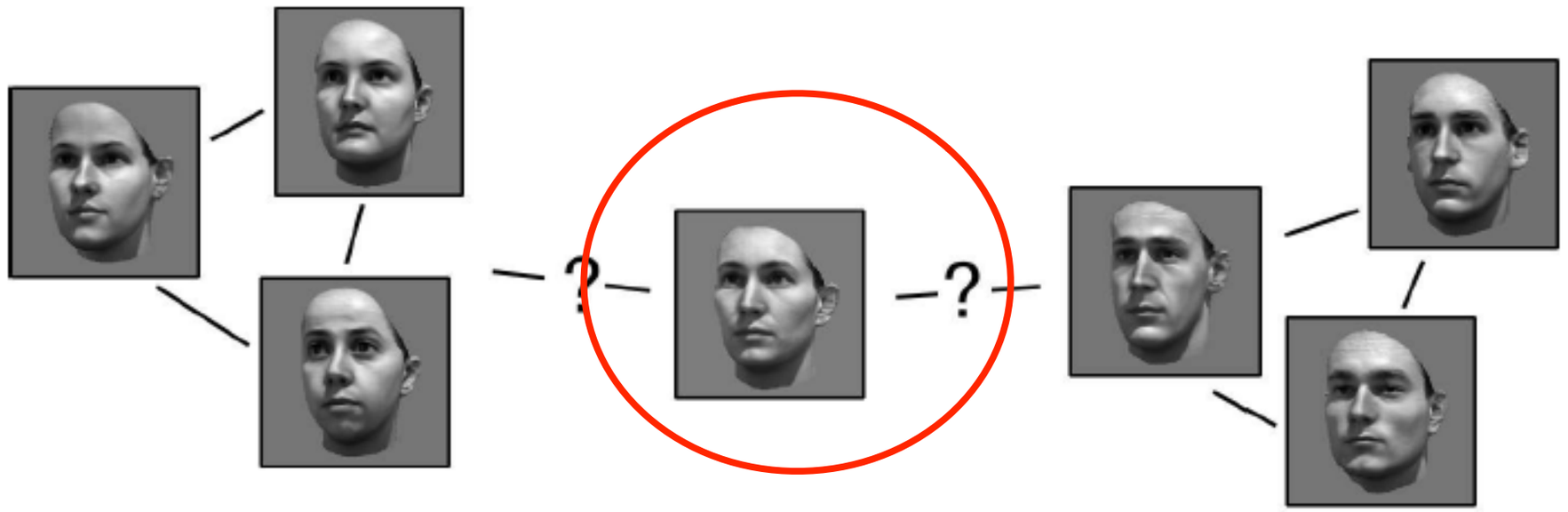


Dissimilar!

However, triangle inequality forces a small distance.

Non-metric similarity is common

Human perception of face



[Laub, Macke, Muller, and Wichmann. NIPS, 2006]

How to model such similarity?

Nonlinear embedding

Map similarity to a kernel space

Ex: [Shepard 87, Jakel, Scholkopf, and Wichmann 08]

Multiple and local metrics

Measuring similarity in local neighborhoods

Does not give rise to similarity for arbitrary pairs

Ex: [Weinberger and Saul 09, Noh, Zhang and Lee 10, Wang, Woznica and Kalousis 12]

Similarity Component Analysis (SCA)

Key aspects

Latent similarities

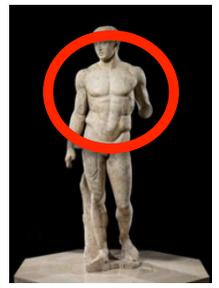
Model *non-metric and noisy similarity* values

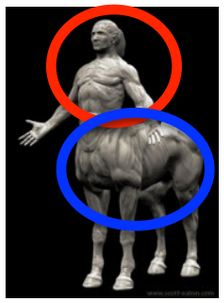
"localized" metrics focus on the relevant *subset* of features

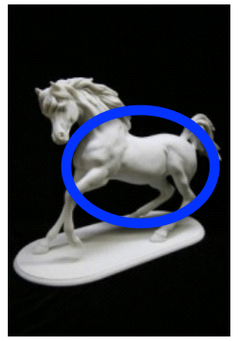
Multiplicative combination of latent components

leads to *tractable* inference

yields *sparse* solutions







A fun example

Young girl or old woman?

Another example: network structures

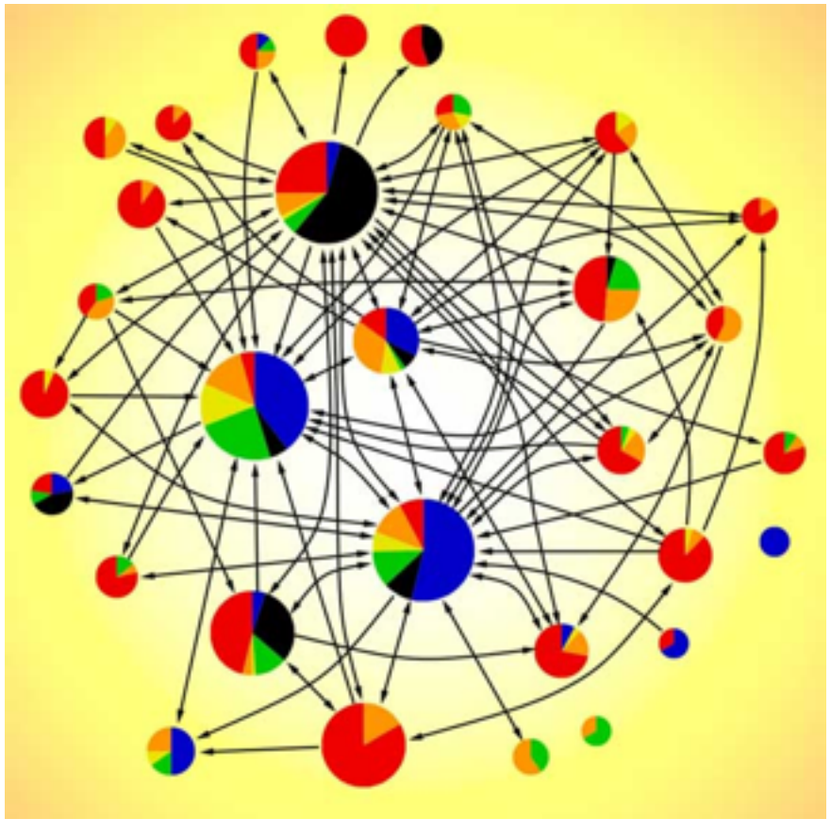
Homophily

Common assumption, explaining how links are formed

``same'' can be decomposed according to we happen to focus on

Same school, religion, zipcode, hobbies, political views?

Thus, complex (social) networks are multiplex:
[Fienberg, Meyer and Wasserman 85, Szell, Lambiotte and Thurner 10]



How to decompose into multiple networks of homogeneous node colors?

cf. [ACKS, SODA 2012]

Outline for the rest of this talk

Similarity component analysis

- Formal definition
- Inference and learning
- Extension and variants

Empirical results

- Synthetic data
- Multi-way classification
- Link prediction

Similarity component analysis (SCA)

Formal definition

Observed data

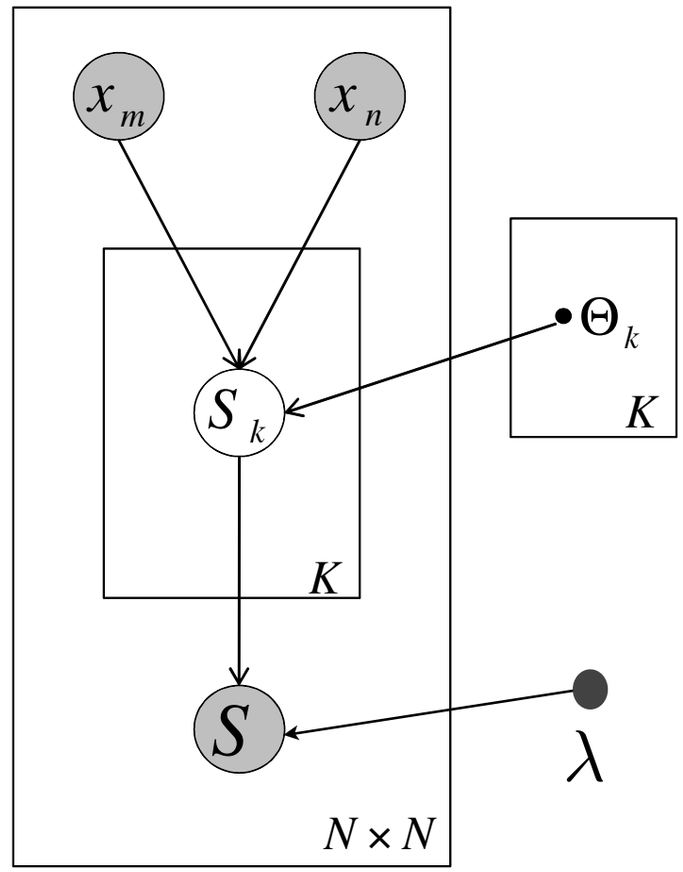
$$(\boldsymbol{x}_m, \boldsymbol{x}_n, s)$$

Latent components

$$s_k \sim p(s_k | \boldsymbol{x}_m, \boldsymbol{x}_n; \boldsymbol{\theta}_k), \quad \forall \quad k \in [\mathrm{K}]$$

Similarity

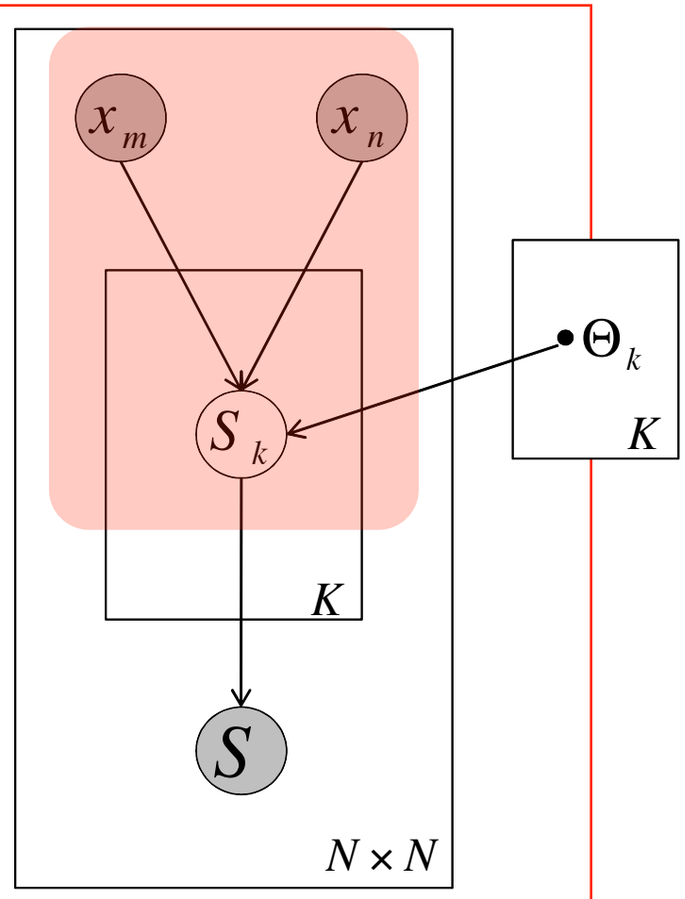
$$s \sim p(s | s_1, s_2, \cdots, s_{\mathrm{K}}; \boldsymbol{\lambda})$$



Latent components

Focus on a subset of features, e.g,

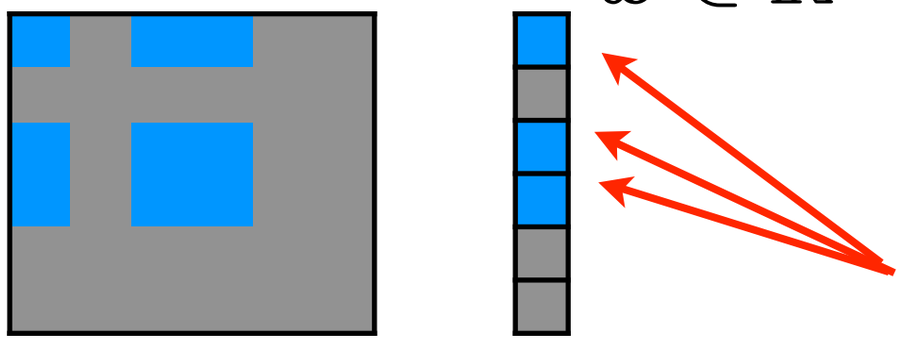
$$d_k = (\boldsymbol{x}_m - \boldsymbol{x}_n)^{\mathrm{T}} \boldsymbol{M}_k (\boldsymbol{x}_m - \boldsymbol{x}_n)$$



Sparse structure in the metric

$$\boldsymbol{M}_k \in \mathbb{R}^{\mathsf{D} \times \mathsf{D}}$$

$$\boldsymbol{x} \in \mathbb{R}^{\mathsf{D}}$$

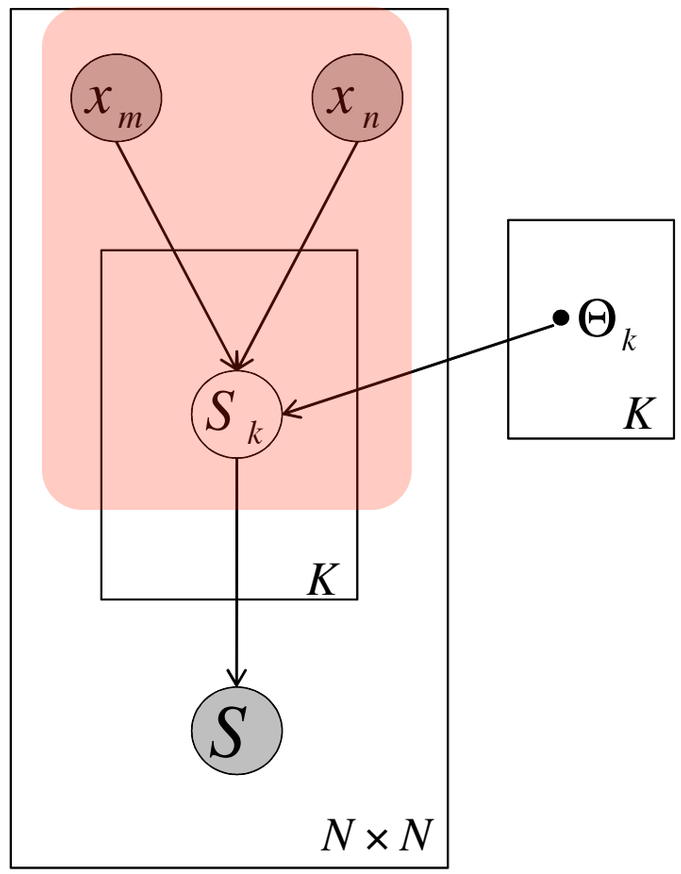


only these features are used

Localized similarity value

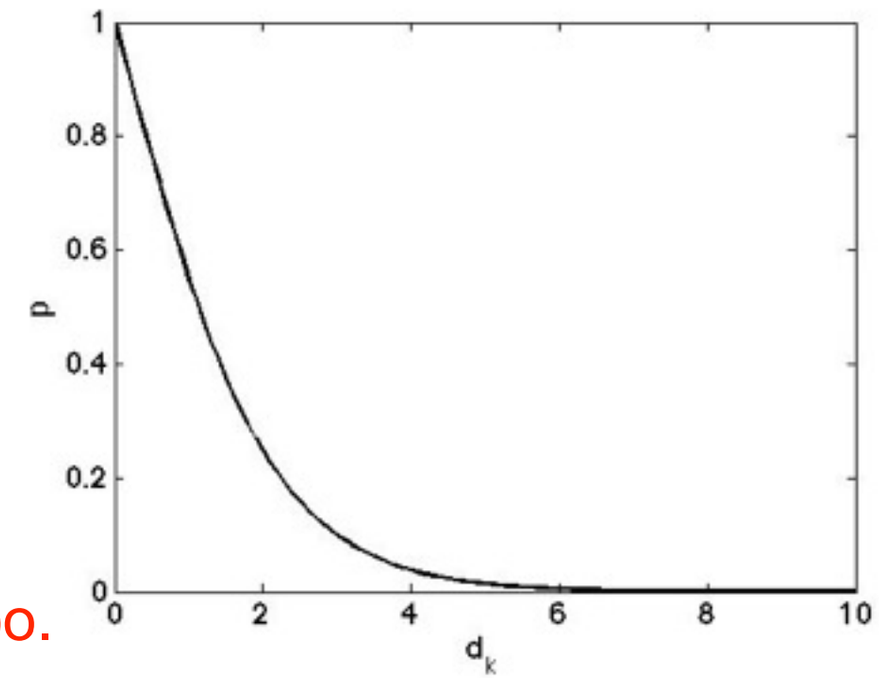
Bernoulli distributed random variable

$$p(s_k = 1 | \boldsymbol{x}_m, \boldsymbol{x}_n) = (1 + e^{-b_k}) \left[1 - \frac{1}{1 + e^{-(d_k - b_k)}} \right]$$



Intuition

Larger distances lead to higher likelihood of being dissimilar.



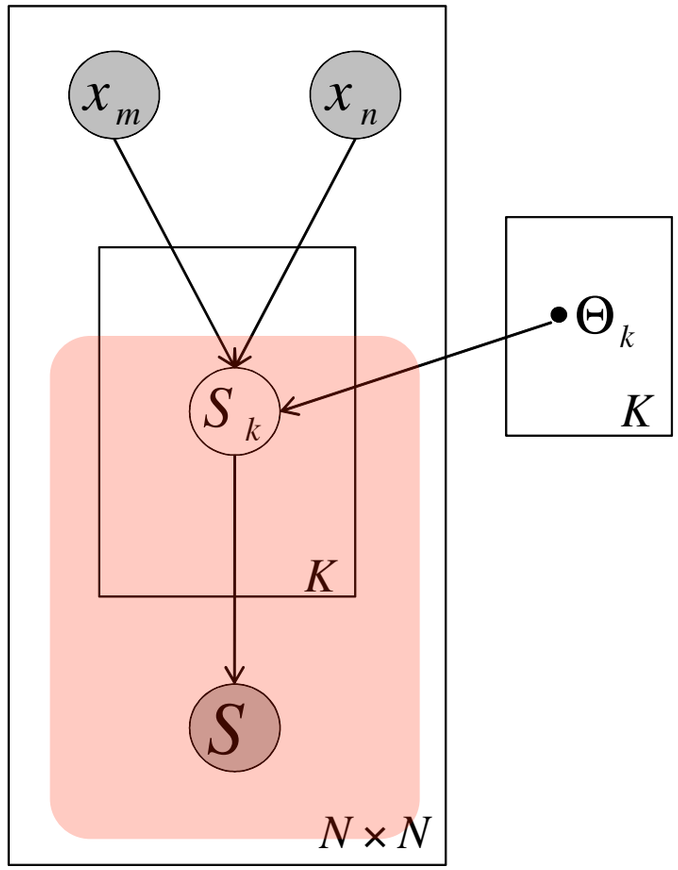
NB. Other forms of conditional probabilities can be used too.

Combining latent components

Multiplicatively combining with OR-gate

$$P(s=1|s_1, s_2, \cdots, s_{\mathsf{K}})$$

$$= 1 - \prod_{k=1}^{\mathsf{K}} \mathbb{I}[s_k = 0]$$



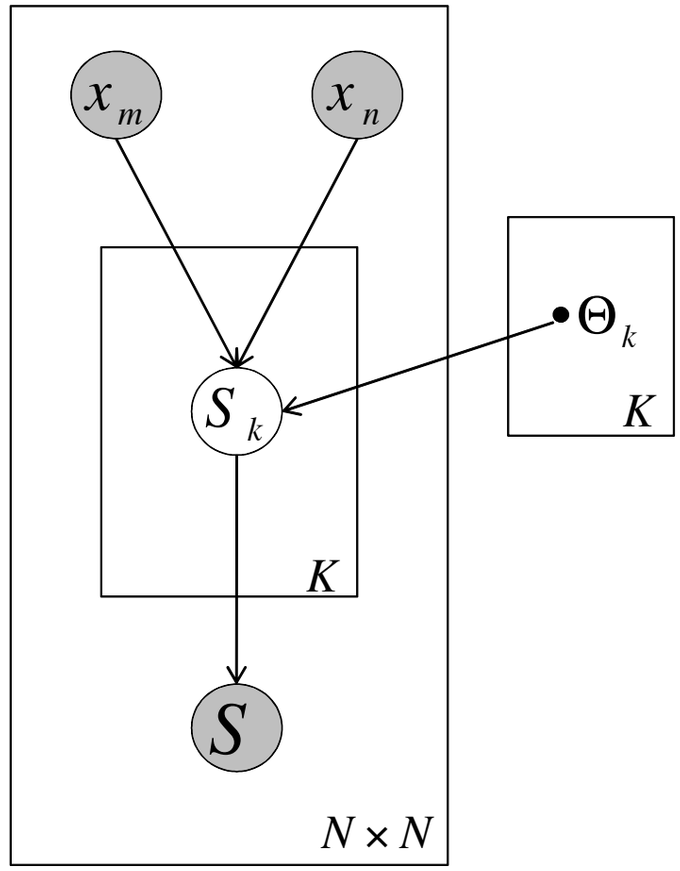
Intuition

Similar *only if at least one* latent component deems to be similar.

Probability of being similar

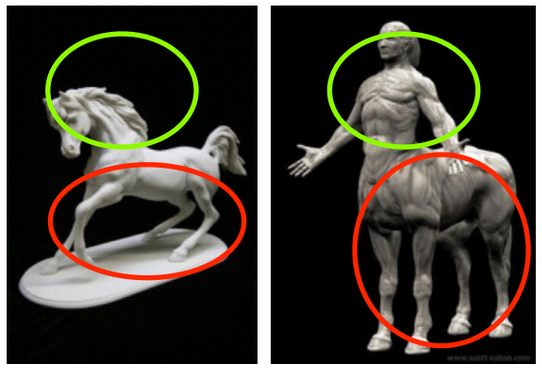
Marginalize out all latent components

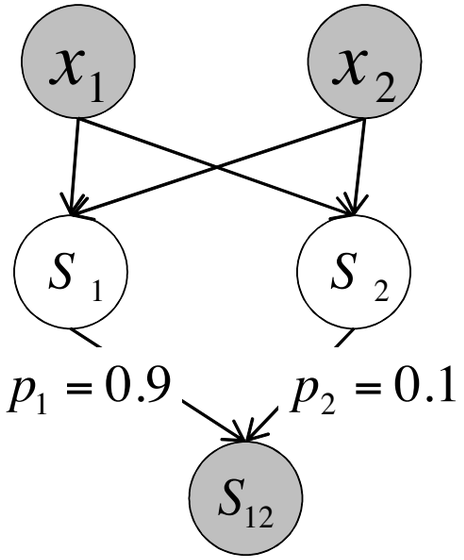
$$P(s=1|\boldsymbol{x}_m,\boldsymbol{x}_n)$$
$$=1-\prod_k[1-P(s_k=1|\boldsymbol{x}_m,\boldsymbol{x}_n)]$$



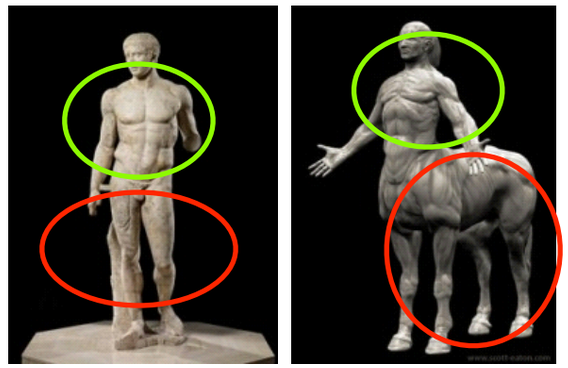
Metrics are combined highly nonlinearly!

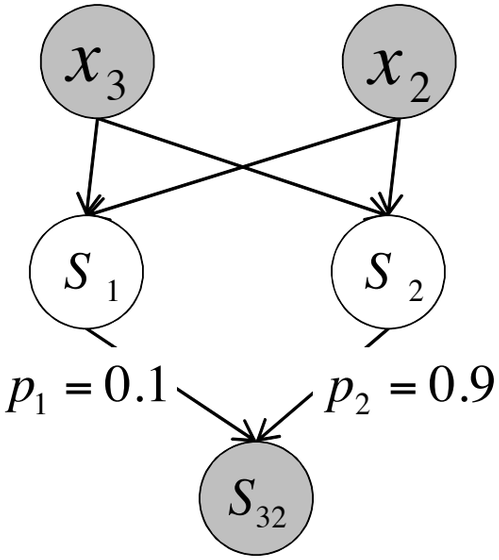
Example



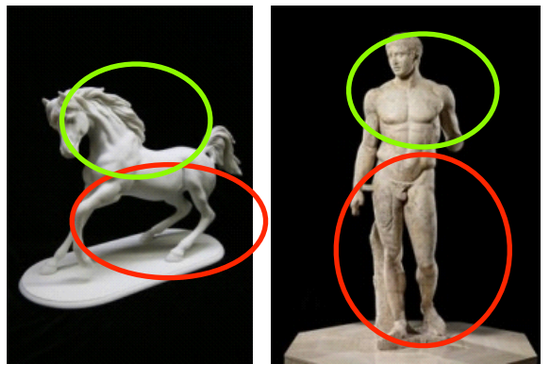


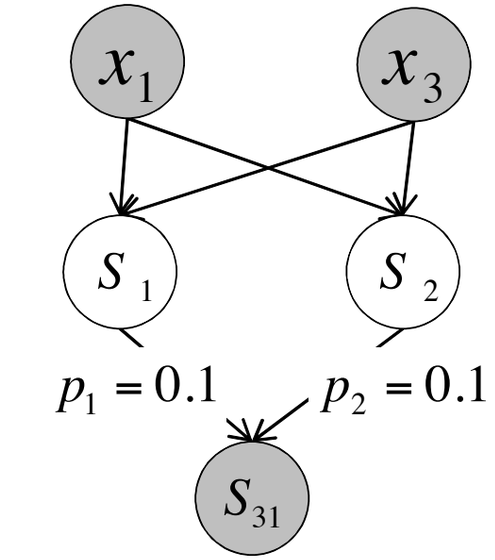
$$p(s=1) = 0.91$$





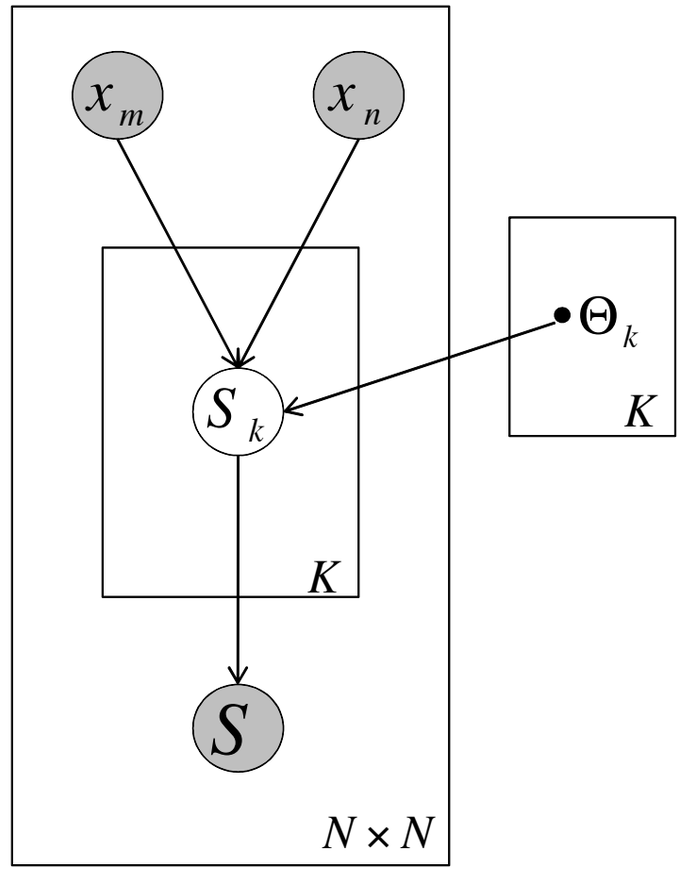
$$p(s=1) = 0.91$$





$$p(s=1) = 0.19$$

Inference over latent variables



Tractable posterior

$$q_k = P(s_k = 1 | \boldsymbol{x}_m, \boldsymbol{x}_n, s = 0) = 0$$

$$r_k = P(s_k = 1 | \boldsymbol{x}_m, \boldsymbol{x}_n, s = 1) = \frac{P(s_k = 1 | \boldsymbol{x}_m, \boldsymbol{x}_n)}{P(s = 1 | \boldsymbol{x}_m, \boldsymbol{x}_n)}$$

Learning

Maximum likelihood estimation

Need to use EM due to the latent variables

Tractable for both E and M steps

In M-step, learning each component independently

Each component is fit analogously as a softly labeled logistic regression

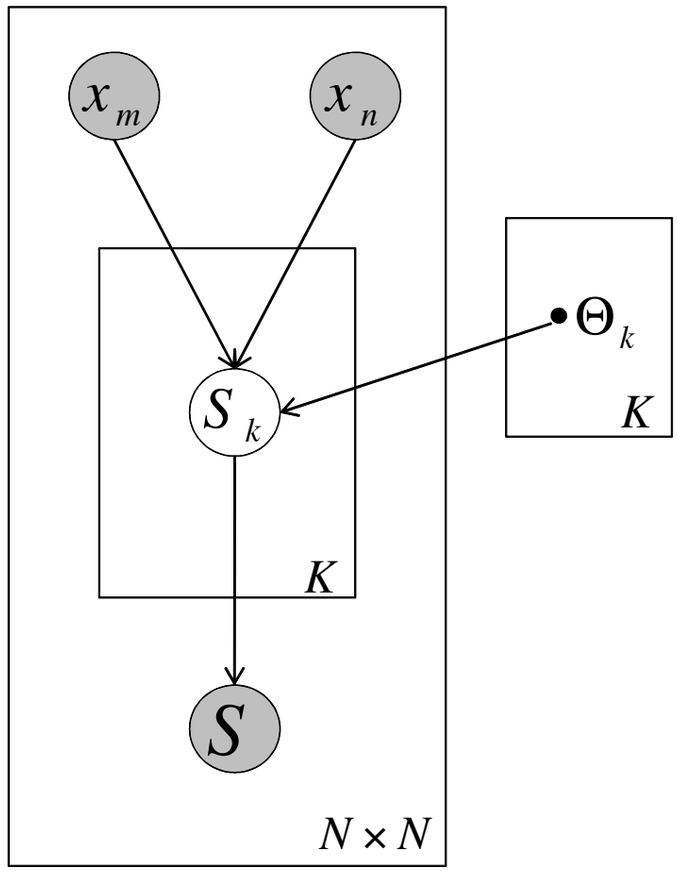
$$J_k = q_k^{1-s} r_k^s \log P(s_k = 1 | \boldsymbol{x}_m, \boldsymbol{x}_n) + (1 - q_k^{1-s} r_k^s) \log(1 - P(s_k = 1 | \boldsymbol{x}_m, \boldsymbol{x}_n))$$

convex in the metric M and
one dimensional search for bias b

Extensions and variants

Modeling latent component similarity

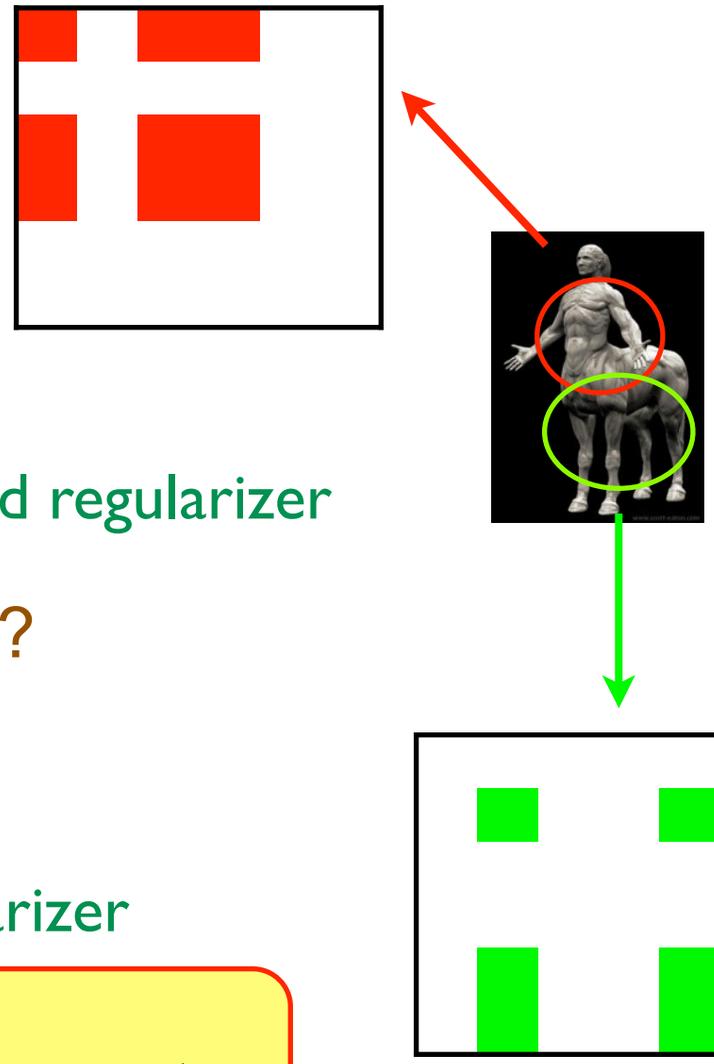
- Use diagonal metrics for high-dimensional data
- Relax positive semi-definitness to be more flexible



Modeling the process of combining

- Use noisy-OR to model each component's noise
- Build recursively and hierarchically more complex models

Structural sparsity



Desiderata

How to make the metrics sparse?

Relatively easy to implement, eg., with an L1-norm based regularizer

How to reduce overlapping between components?

Hard to implement, essentially nonconvex constraints

Approximated with the following disjoint convex regularizer

$$R(\{\boldsymbol{M}_k\}) = \sum_{k,k'} \mathsf{diag}(\boldsymbol{M}_k)^{\mathrm{T}} \mathsf{diag}(\boldsymbol{M}_{k'})$$

Good news: empirically sparse solutions even w/o these extensions

Outline for the rest of this talk

- ✓ **Similarity component analysis**
  - Formal definition
  - Inference and learning
  - Extension and variants
- **Empirical results**
  - Synthetic data
  - Multi-way classification
  - Link prediction

Empirical results on synthetic data

Setup

Sample training data from the model

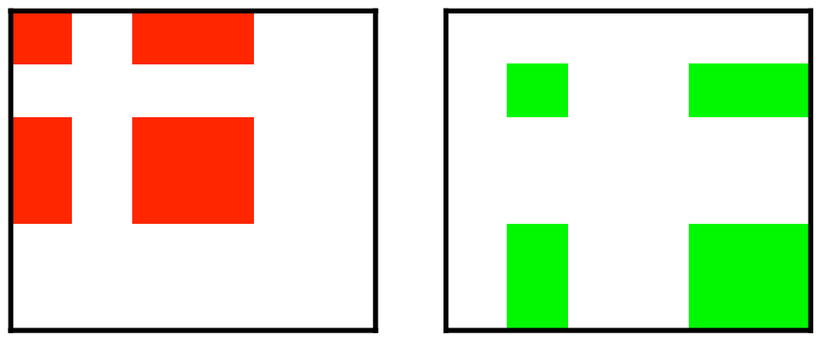
features: points in a 30-dimensional Euclidean space

number of components: 5

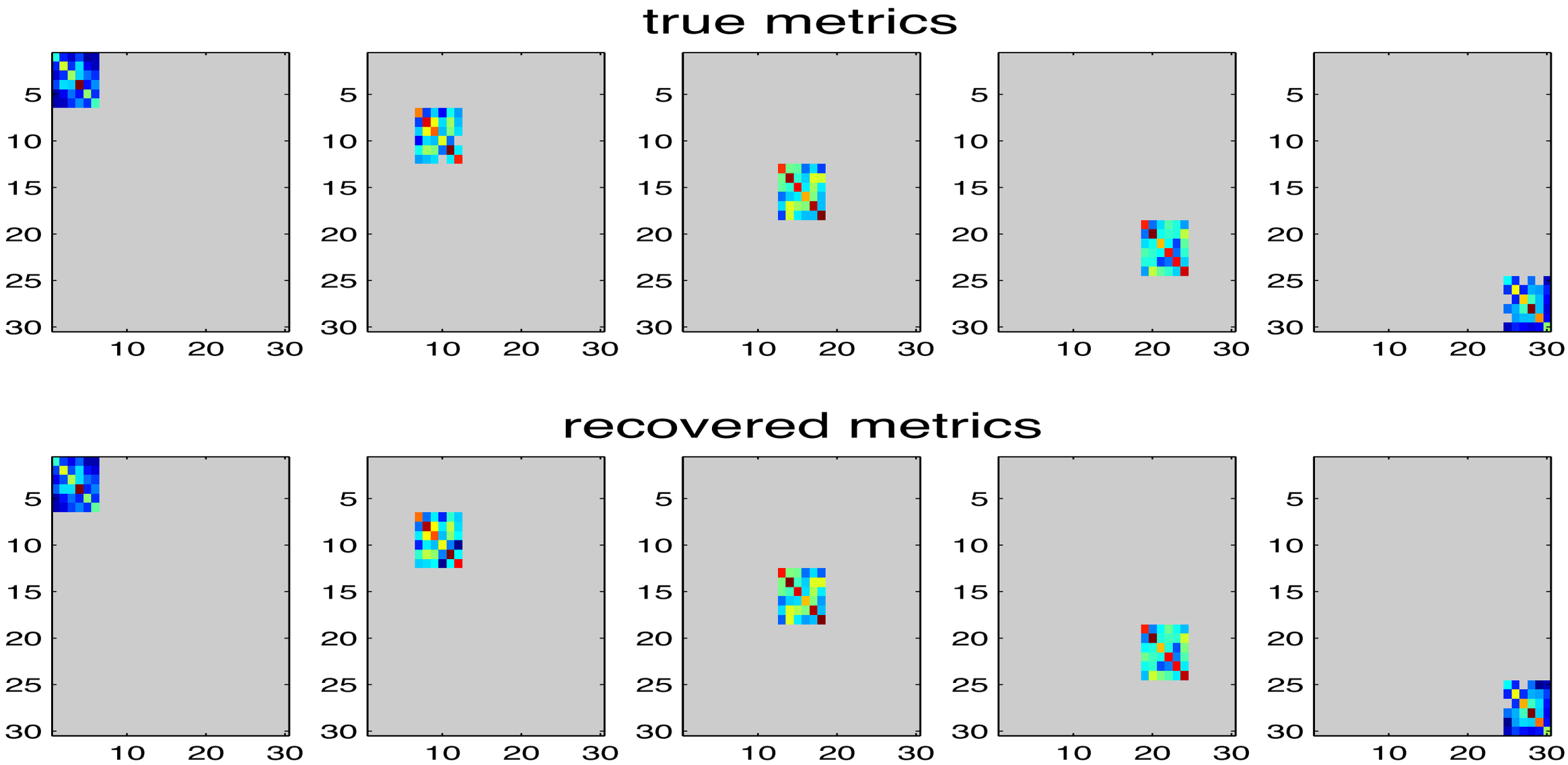
number of training pairs: 1/3 of all possible pairs

Set metrics deliberately to be sparse and non-overlapping

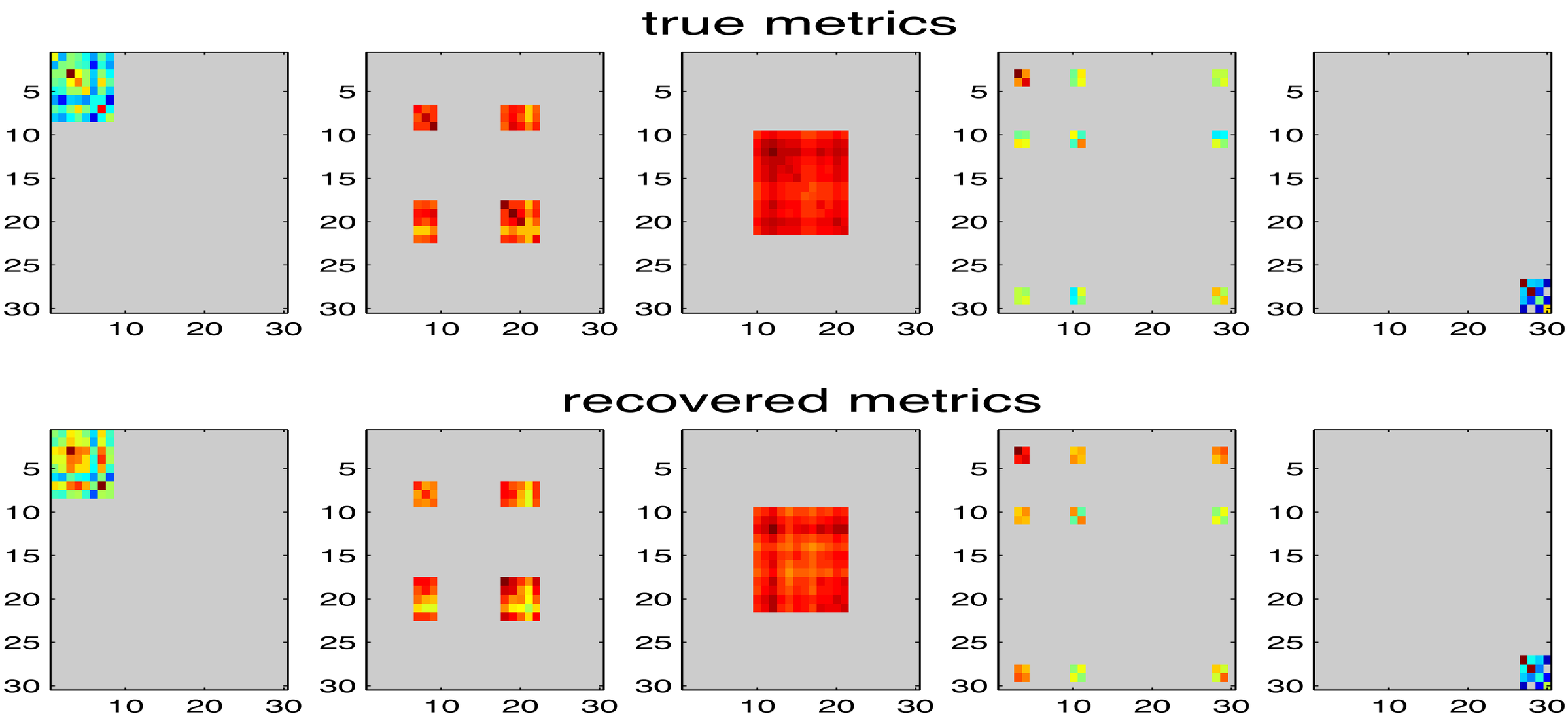
Can we recover the special structural patterns?



Recovering ground-truth metrics



Also work for overlapping metrics



What if we do not know K?

ground-truth

Table 1: Similarity prediction accuracies and standard errors (%)

BASELINES		SCA					
ITML	LMNN	K = 1	K = 3	K = 5	K = 7	K = 10	K = 20
72.7±0.0	71.3±0.2	72.8±0.0	82.1±0.1	**91.5±0.1**	**91.7±0.1**	**91.8±0.1**	90.2±0.4

Both overfit and underfit are possible, but

- Resilient to over-specifying the number of components
- Better more than less

Empirical results on classification

Setup

MNIST dataset: 4200 images for training

Data is given in the form of pairwise similarity

Formed by same/different labels for pairwise images.

Classification from similarity values

$$y = \arg\max_c s_c = \arg\max_c \sum_{\boldsymbol{x}' \in B_c(\boldsymbol{x})} P(s=1|\boldsymbol{x}, \boldsymbol{x}')$$

Contrast to metric learning methods

Table 1: Misclassification rates (%) on the MNIST recognition task

	BASELINES				SCA		
D	EUC.	ITML	LMNN	MM-LMNN	K = 1	K = 5	K = 10
25	21.6	15.1	20.6	20.2	17.7 ± 0.9	16.0 ± 1.5	**14.5 ± 0.6**
50	18.7	13.35	16.5	13.6	13.8 ± 0.3	12.0 ± 1.1	**11.4 ± 0.6**
100	18.1	11.85	13.4	**9.9**	12.1 ± 0.1	10.8 ± 0.6	11.1 ± 0.3

Outperform all single-metric methods

Close to multiple-metric method

MM-LMNN is highly specialized, requiring knowing class labels during training

Empirical results on network data

Setup

Network of NIPS papers across 13 years, split into 9 sections

Nodes: papers, annotated with features extracted from the documents

Edges: nodes are linked if they are from the same conference section

Highly noisy as the determination of the sections is not categorical.

Features

Bag of words: high-dimensional

Topic vectors (after fitting to a LDA): low-dimensional

Top Words: most frequently used words (selected by LDA): high-dimensional

Link prediction accuracy

Table 1: Link prediction accuracies and their standard errors (%)

Feature type	BASELINES			SCA-DIAG		SCA	
	SVM	ITML	LMNN	K = 1	K*	K = 1	K*
BoW	73.3±0.0	-	-	64.8 ± 0.1	**87.0 ± 1.2**	-	-
ToW	75.3±0.0	-	-	67.0 ± 0.0	**88.1 ± 1.4**	-	-
ToP	71.2±0.0	81.1±0.1	80.7±0.1	62.6 ± 0.0	81.0 ± 0.8	81.0 ± 0.0	**87.6 ± 1.0**

Outperform all competing methods significantly.

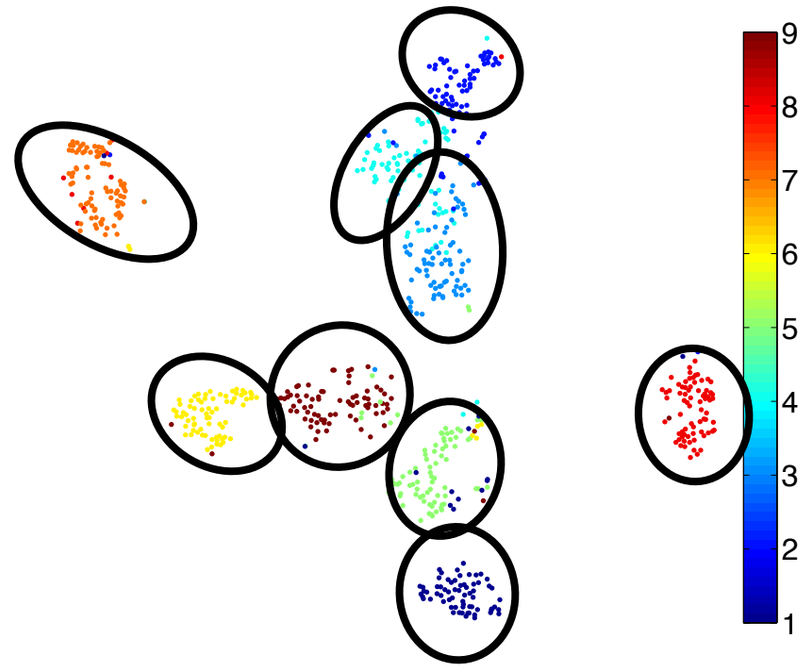
Visualization of link distributions

New representation from latent components

For each pair of documents, compute a similarity value for each component

Form a high-dimensional vector by stacking the latent similarity values

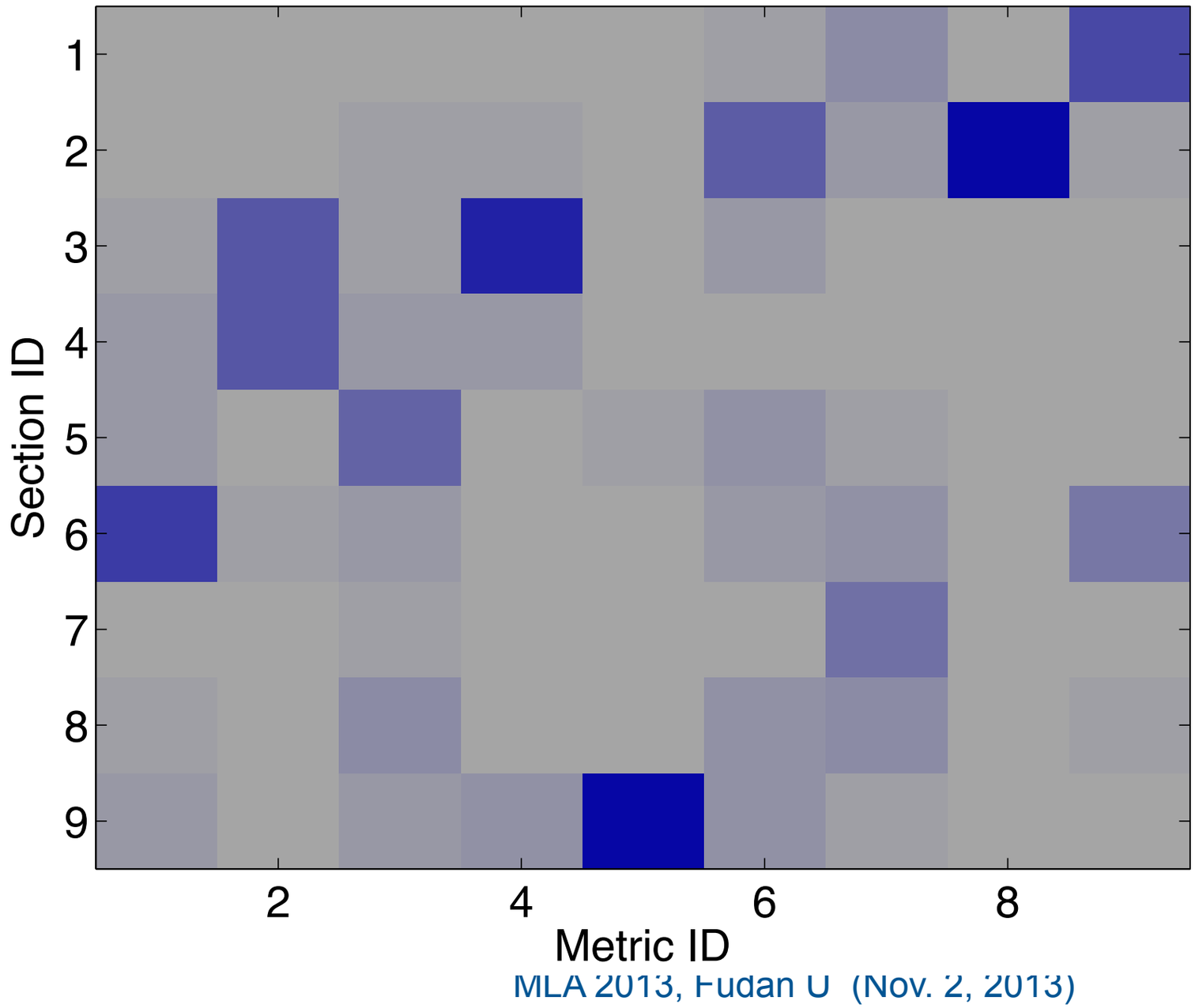
Project to low-dimensional space (optionally)



Latent similarities reveal clustering structures, corresponding to sections

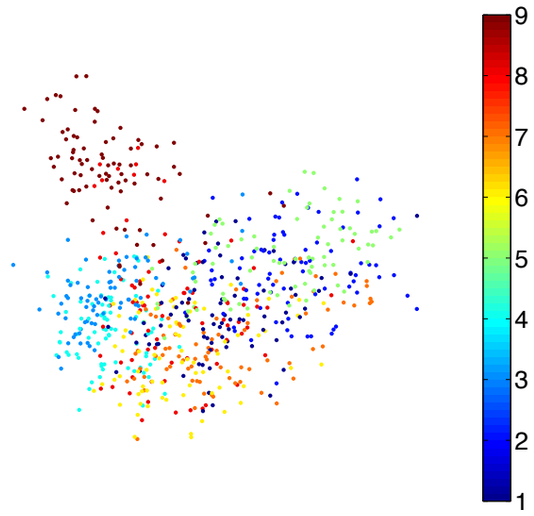
How "specific" are latent components?

Averaged "activation" of each component for different sections



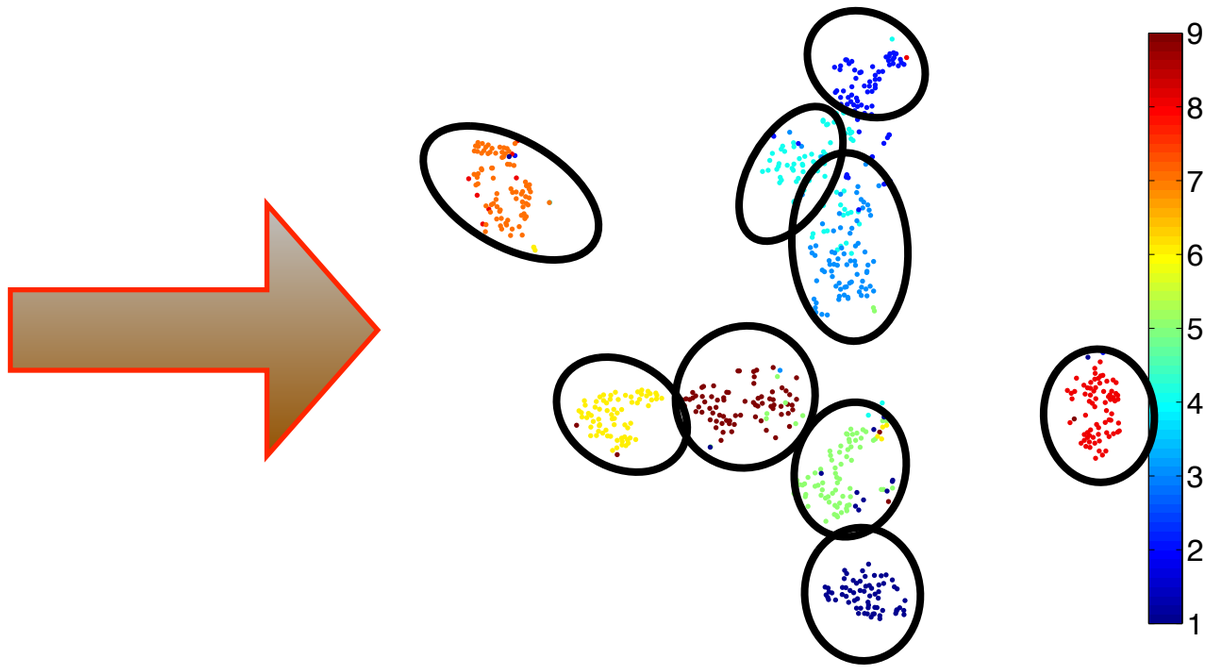
Original representation is inadequate

Embedding using original representation



Embedding proximity does not reveal similarity

Project into latent similarity space



Highly nonlinear transformation reveal clustering structures.

Summary

Similarity component analysis

- Probabilistic modeling of similarity
- Advantages
  - Infer multiple latent components of similarity
  - Provide insights to understand how (local) similarity is assessed
  - Combine into a global similarity nonlinearly (and probabilistically)
  - Maintain computational tractability
- Preliminary results are very encouraging