Dimension Reduction

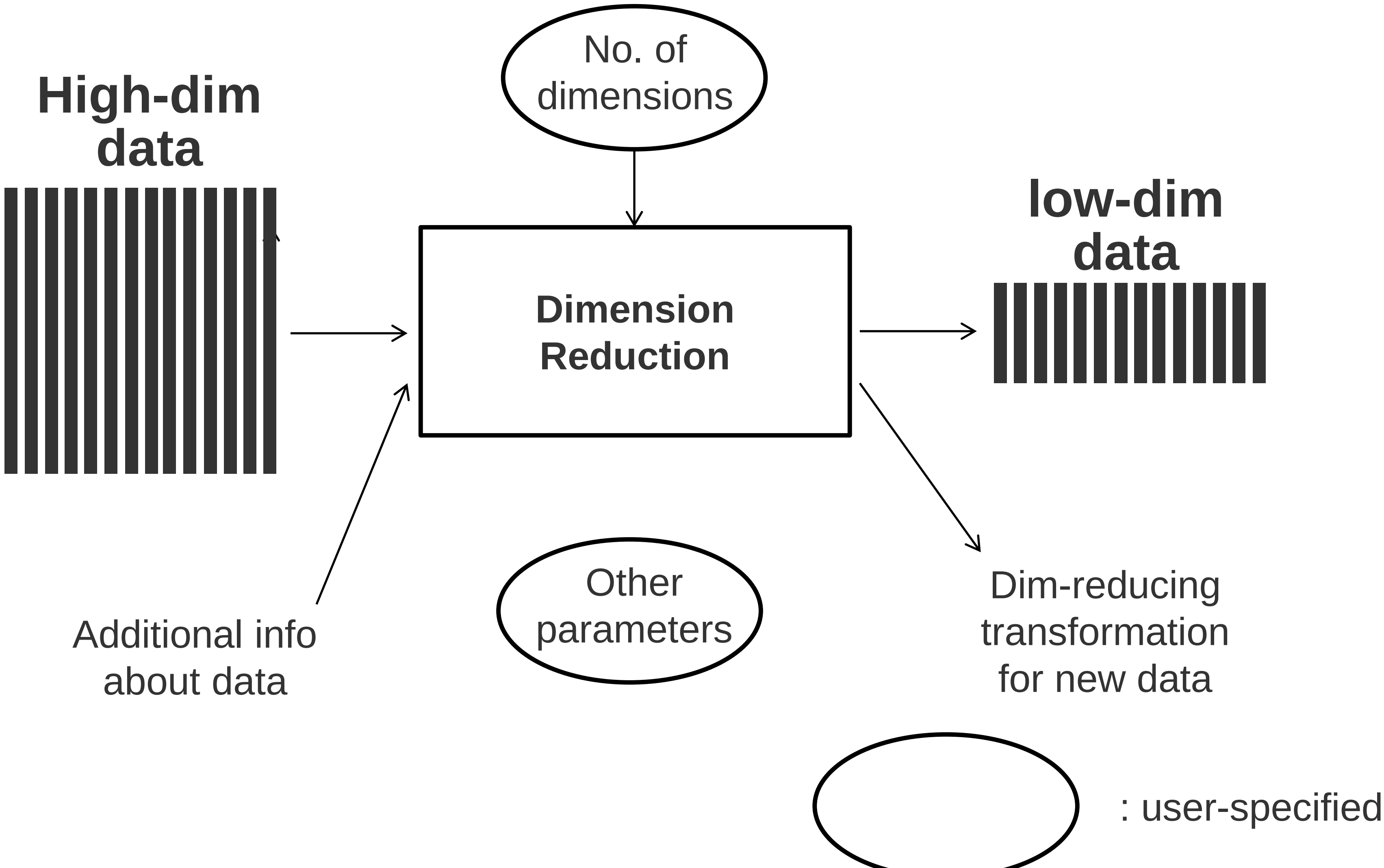
Jaegul Choo

Assistant Professor

Dept. of Computer Science and Engineering

Korea University

Overview of Dimension Reduction
Let's Reduce Data (along Dimension Axis)



Benefits of Dimension Reduction

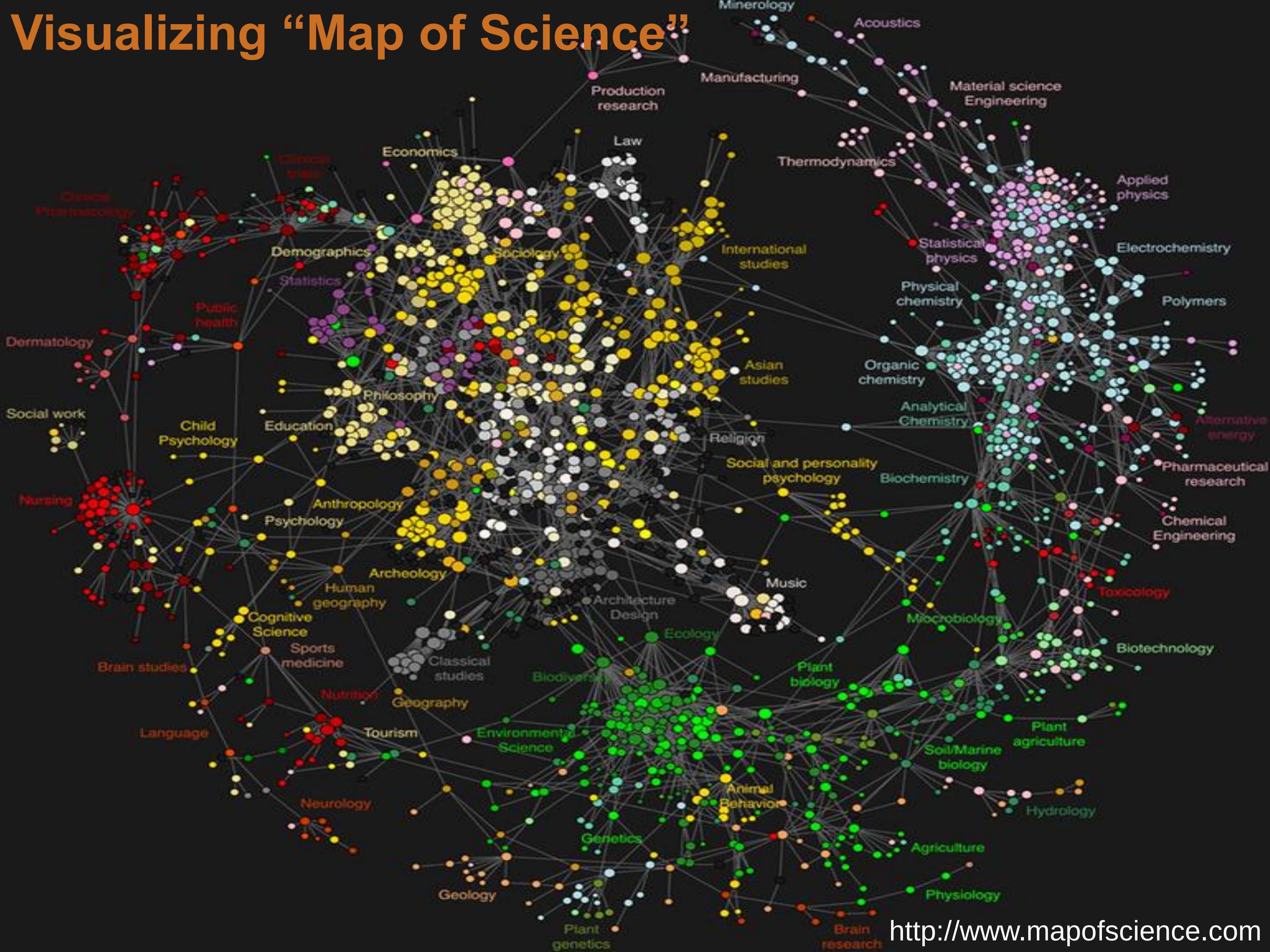
Obviously,

- Less storage
- Faster computation
  - Computing distances: 100,000-dim vs. 10-dim vectors

More importantly,

- Noise removal (improving data quality)
  - Works as **pre-processing** for **better performance**
  - e.g., microarray data analysis, information retrieval, face recognition, protein disorder prediction, network intrusion detection, document categorization, speech recognition
- **2D/3D representation**
  - **Interactive visual exploration**

Visualizing "Map of Science"



http://www.mapofscience.com

Two Main Techniques

1. Feature **selection**

- Selects a subset of the original variables as reduced dimensions
- e.g., the number of genes responsible for a particular disease may be small

2. Feature **extraction**

- Each reduced dimension combines multiple original dimensions
- Active research area

Feature = Variable = Dimension

Feature Selection

What are the optimal subset of ***m*** features to maximize a given criterion?

- Widely-used criteria
  - Information gain, correlation, …
- Typically combinatorial optimization problems
- Therefore, greedy methods are popular
  - Forward selection: **Empty** set → **Add** one variable at a time
  - Backward elimination: **Entire** set → **Remove** one variable at a time

Feature Extraction
(Our main topic from now on)

Aspects of Dimension Reduction
Linear vs. Nonlinear

Linear

- Represents each reduced dimension as a linear combination of original dimensions
  - e.g., Y1 = 3*X1 – 4*X2 + 0.3*X3 – 1.5*X4,
    Y2 = 2*X1 + 3.2*X2 – X3 + 2*X4
- Naturally capable of mapping new data to the same space

	D1	D2
X1	1	1
X2	1	0
X3	0	2
X4	1	1

→ **Dimension Reduction** →

	D1	D2
Y1	1.75	-0.27
Y2	-0.21	0.58

Principal Component Analysis

- Finds the axis showing the largest variation, and project all points into this axis
- Reduced dimensions are orthogonal
- Algorithm: Eigen-decomposition
- Pros: Fast
- Cons: Limited performances

Linear
Unsupervised
Global
Feature vectors

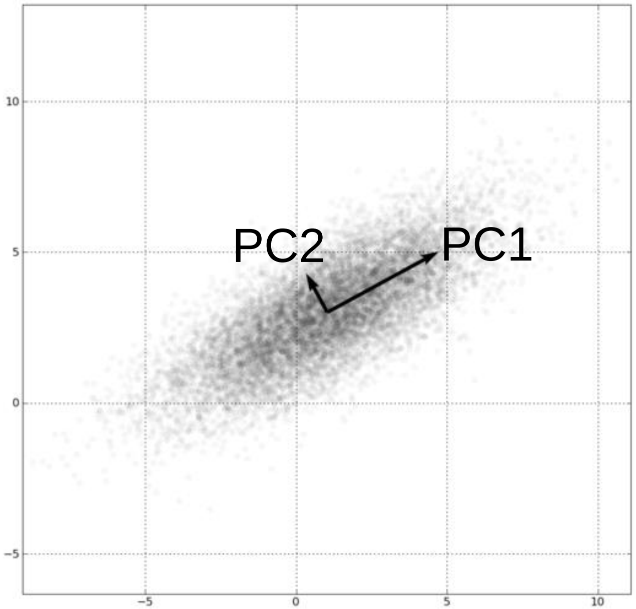


Image source: http://en.wikipedia.org/wiki/Principal_component_analysis

Multidimensional Scaling (MDS)

Main idea

- Tries to preserve given pairwise distances in low-dimensional space

Low-dim distance

ideal distance

$$\min_{x_1,\ldots,x_I} \sum_{i<j} \left(\boxed{\|x_i - x_j\|} - \boxed{\delta_{i,j}} \right)^2.$$

- Metric MDS
  - Preserves given distance values
- Nonmetric MDS
  - When you only know/care about ordering of distances
  - Preserves only **the orderings** of distance values

Nonlinear

Unsupervised

Global

Similarity input

- Algorithm: gradient-decent type

c.f. classical MDS is the same as PCA

Multidimensional Scaling
Sammon's mapping

Sammon's mapping

- Local version of MDS
- Down-weights errors in large distances

$$E = \frac{1}{\sum\limits_{i<j} d^*_{ij}} \sum_{i<j} \frac{(d^*_{ij} - d_{ij})^2}{d^*_{ij}}.$$

- Algorithm: gradient-decent type

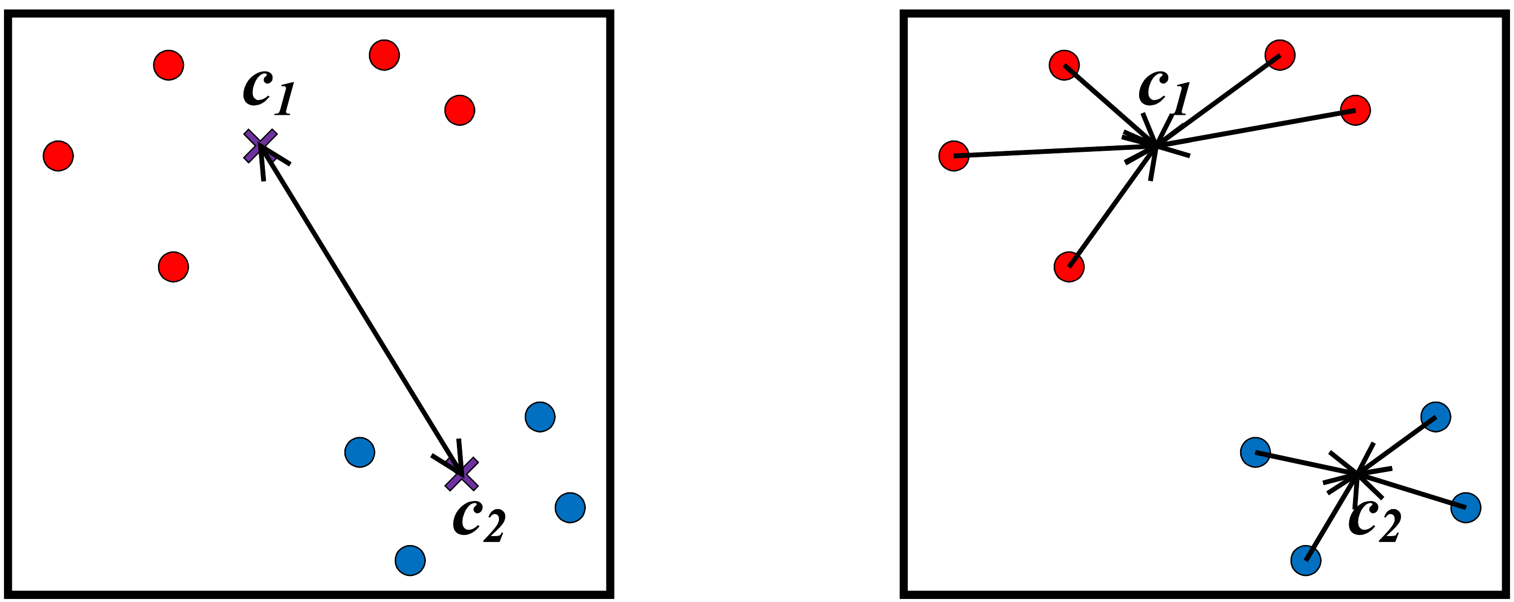
Nonlinear
Unsupervised
Local
Similarity input

Linear Discriminant Analysis

What if clustering information is available?

LDA tries to separate clusters by

- Putting different cluster as far as possible
- Putting each cluster as compact as possible

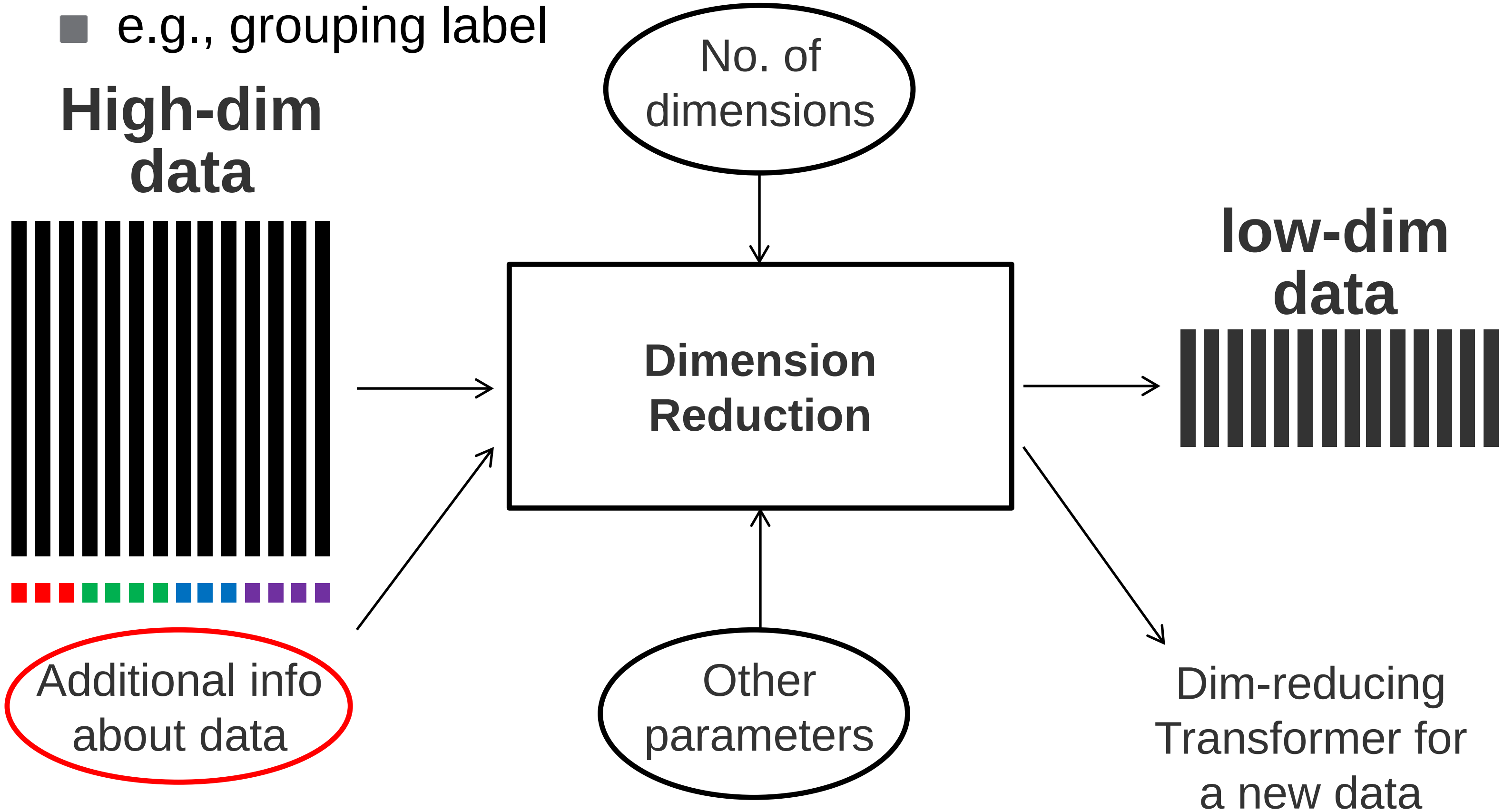


(a) (b)

Aspects of Dimension Reduction
Unsupervised vs. Supervised

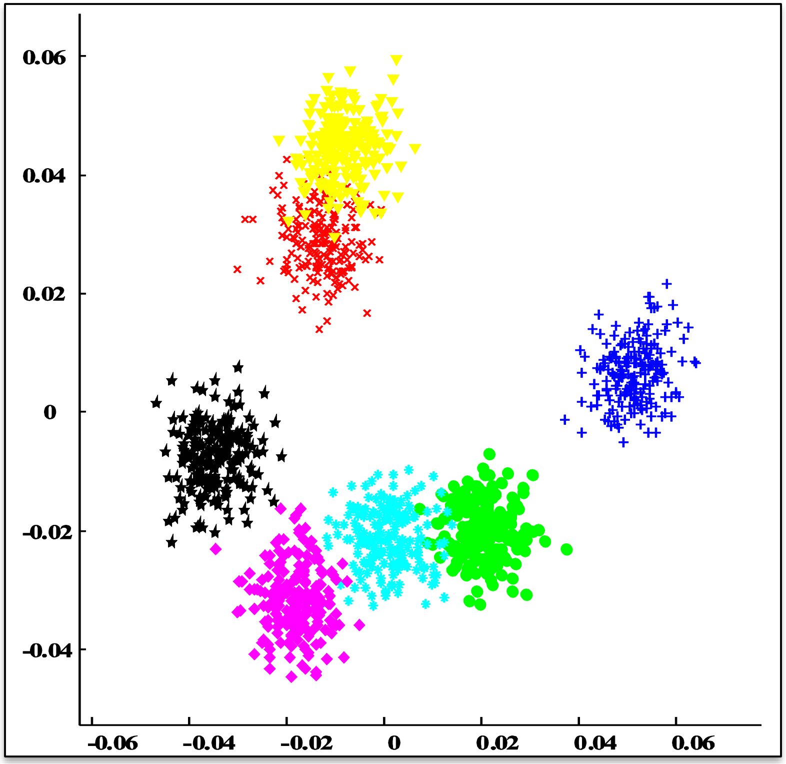
Supervised

- Uses the input data + additional info
  - e.g., grouping label



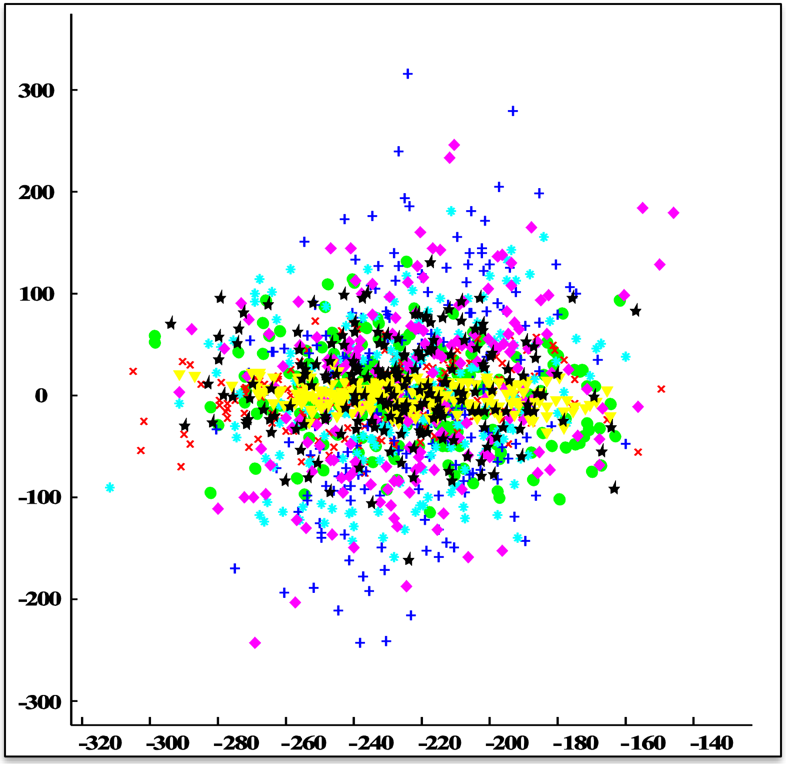
Linear Discriminant Analysis
vs. Principal Component Analysis

2D visualization of 7 Gaussian mixture of 1000 dimensions



Linear discriminant analysis

(Supervised)



Principal component analysis

(Unsupervised)