UNSUPERVISED LEARNING 2011

DIMENSIONALITY REDUCTION: PCA, MDS

Rita Osadchy

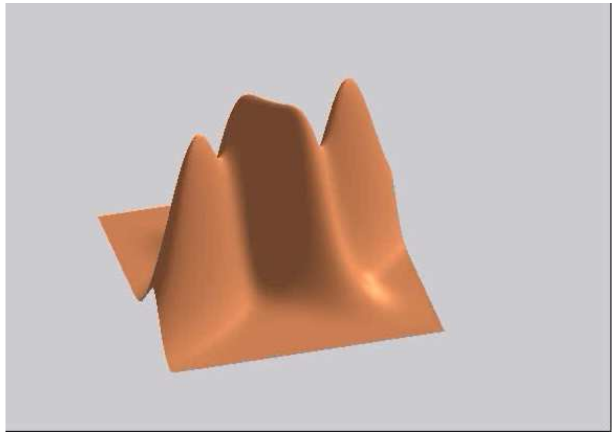
slides are due to L.Saul , A. Ng, and A. Ghodsi

Topics

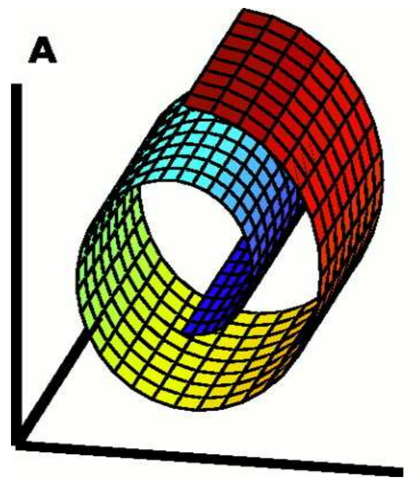
- PCA
- MDS
- IsoMap
- LLE
- EigenMaps

Types of Structure in High Dimension

- Clumps
  - Clustering
  - Density Estimation



- Low Dimensional Manifolds
  - Linear
  - NonLinear



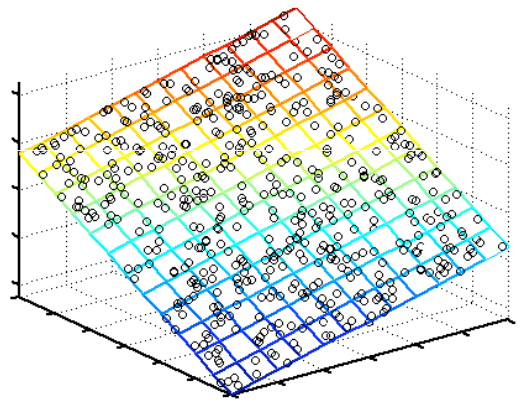
Dimensionality Reduction

- Data representation

  Inputs are real-valued vectors in a high dimensional space.

- Linear structure

  Does the data live in a low dimensional subspace?



- Nonlinear structure

  Does the data live on a low dimensional submanifold?

Dimensionality Reduction

- Question

  How can we detect low dimensional structure in high dimensional data?

- Applications
  - Digital image and speech processing
  - Analysis of neuronal populations
  - Gene expression microarray data
  - Visualization of large networks

Notations

- Inputs (**high dimensional**)

  $\boldsymbol{x_1, x_2, \ldots, x_n}$ points in R^D

- Outputs **(low dimensional)**

  $\boldsymbol{y_1, y_2, \ldots, y_n}$ points in R^d ($d \ll D$)

- Goals

  Nearby points remain nearby.

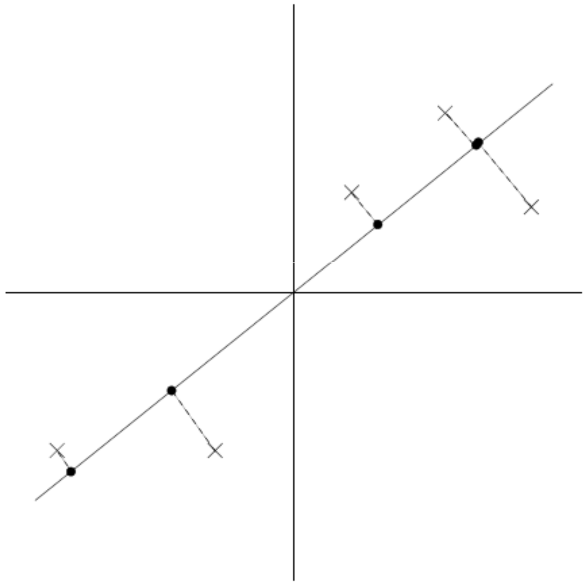
  Distant points remain distant.

Linear Methods

- PCA
- MDS

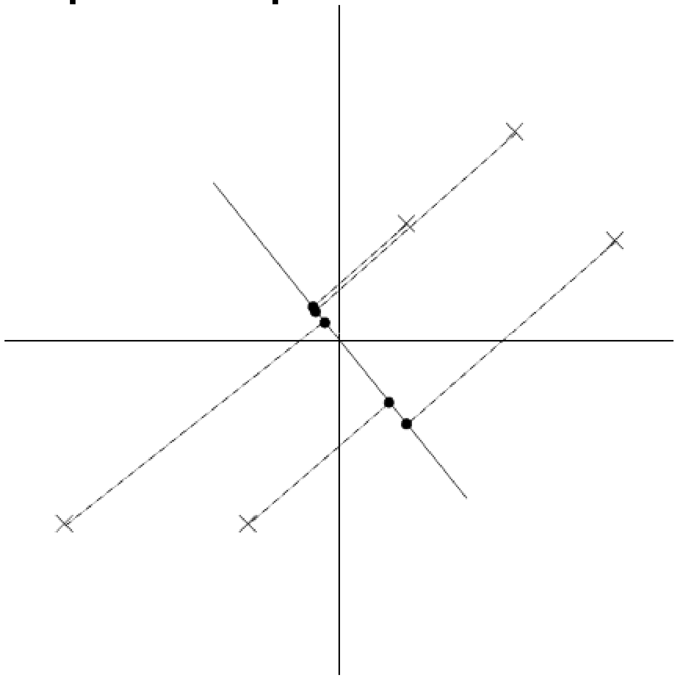
Principle Component Analysis

good representation



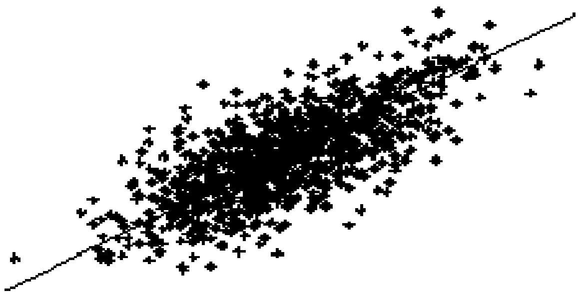
the projected data has a fairly large variance, and the points tend to be far from zero.

poor representation

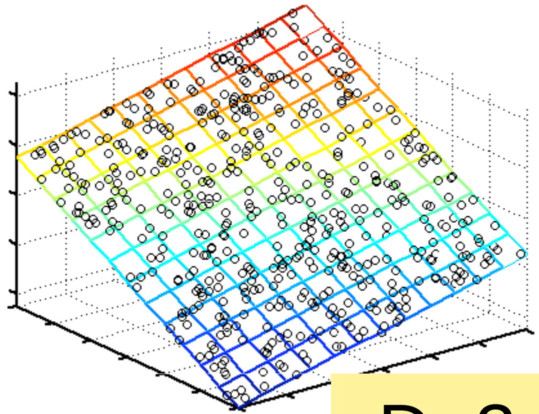


the projections have a significantly smaller variance, and are much closer to the origin.

Principle Component Analysis



D=2, d=1



D=3, d=2

- Seek most accurate data representation in a lower dimensional space.
- The good direction/subspace to use for projection lies in the direction of largest variance.

Maximum Variance Subspace

- Assume inputs are centered: $\sum_i x_i = 0$
- Given a unit vector u and a point x, the length of the projection of x onto u is given by $x^T u$
- Maximize projected variance:

$$\mathrm{var}(y) = \frac{1}{n}\sum_i \left(x_i^T u\right)^2 = \frac{1}{n}\sum_i u^T x_i x_i^T u$$

$$= u^T \left(\frac{1}{n}\sum_i x_i x_i^T \right) u$$

1D Subspace

- Maximizing $u^T C u$ subject to $\|\mathrm{u}\| = 1$

  where $C = n^{-1} \sum_i x_i x_i^T$ is the empirical

  covariance matrix of the data,

  gives the principle eigenvector of C.

d-dimensional Subspace

- to project the data into a d-dimensional subspace (d <<D), we should choose $u_1,\ldots,u_d$ to be the top d eigenvectors of C.
- $u_1,\ldots,u_d$ now form a new, orthogonal basis for the data.
- The low dimensional representation of x is given by

$$y_i = \begin{bmatrix} u_1^T x_i \\ u_2^T x_i \\ \vdots \\ u_k^T x_i \end{bmatrix} \in \Re^d .$$

Interpreting PCA

- Eigenvectors:

  principal axes of maximum variance subspace.

- Eigenvalues:

  variance of projected inputs along principle axes.

- Estimated dimensionality:

  number of significant (nonnegative) eigenvalues.

PCA summary

Input: $z_i \in R^D,\ i=1,..,n$ Output: $y_i \in R^d,\ i=1,..,n$

1. Subtract sample mean from the data

$$x_i = z_i - \hat{\mu}, \quad \hat{\mu} = 1/n \sum_i z_i$$

2. Compute the covariance matrix

$$C = 1/n \sum_{i=1}^{n} x_i x_i^t$$

3. Compute eigenvectors ***e**$_1$*,***e**$_2$*,…,***e**$_d$* corresponding to the ***d*** largest eigenvalues of C (d<<D).

4. The desired y is

$$y = P^t x,\ \ P = [e_1,\ldots,e_d]$$

Equivalence

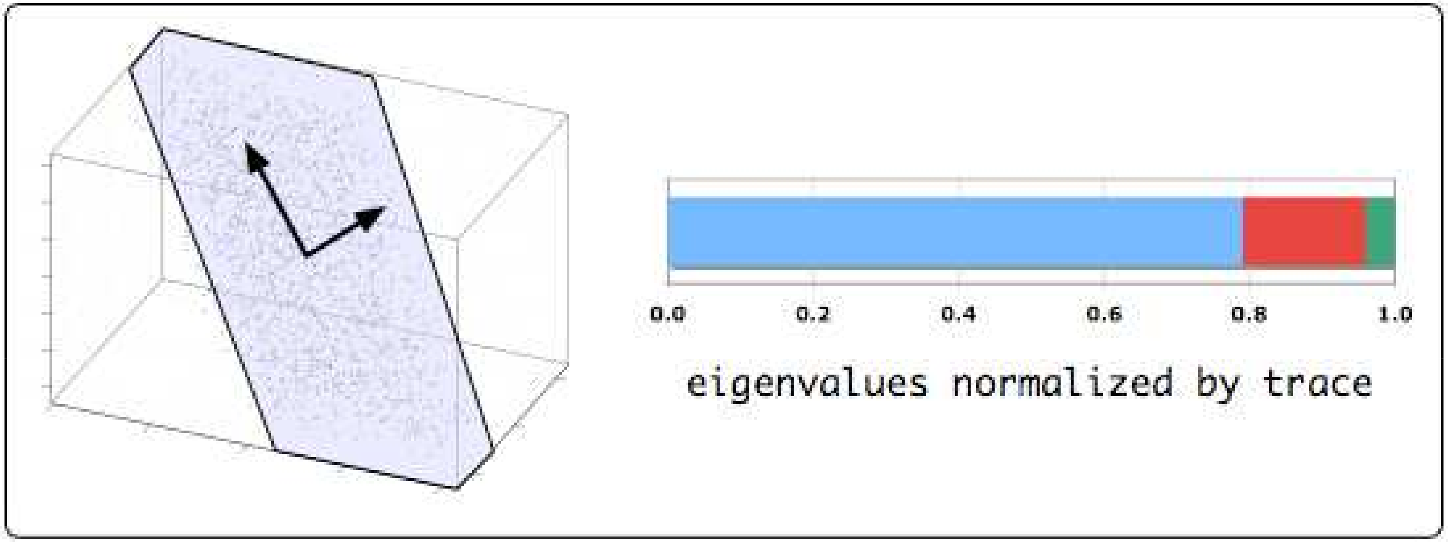
- PCA finds the directions that have the most variance.

$$\operatorname{var}(y) = \frac{1}{n}\sum_i \left\| P^T x_i \right\|^2$$

- Same result can be obtained by minimizing the squared reconstruction error.

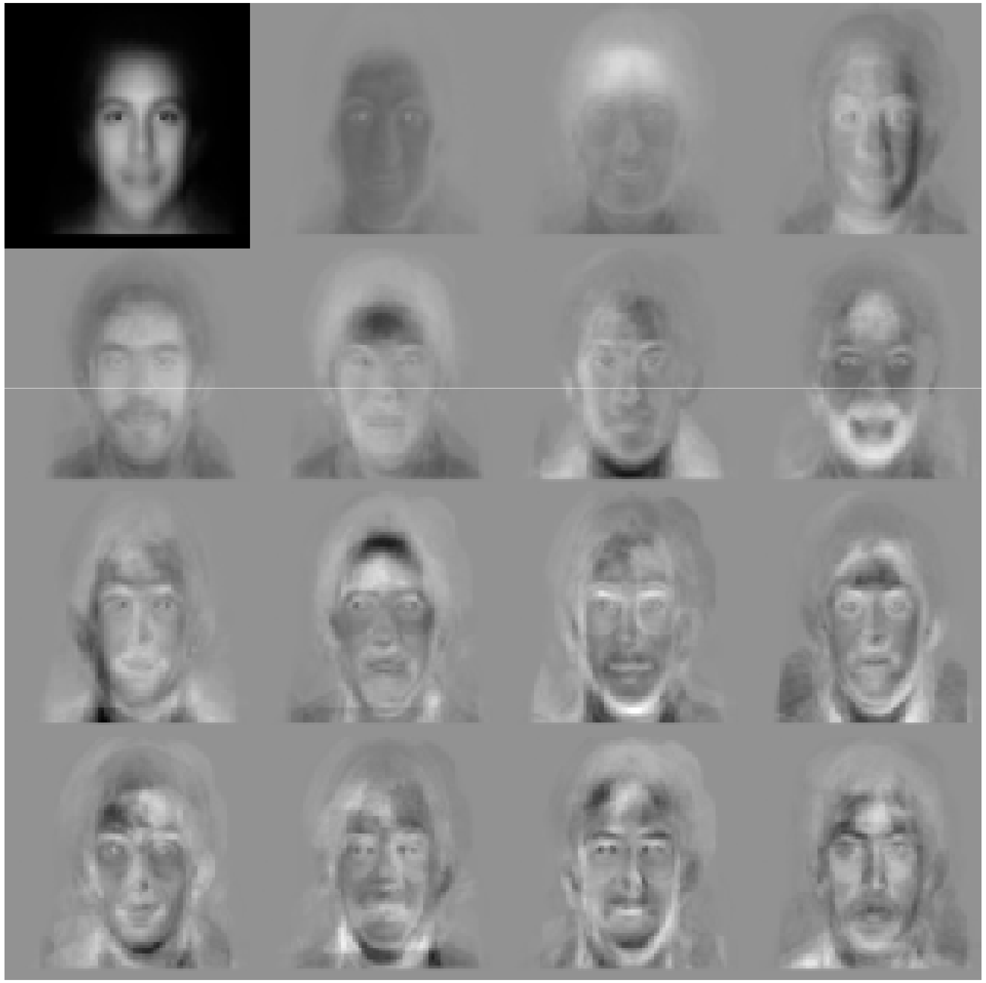
$$err(y) = \frac{1}{n}\sum_i \left\| x_i - PP^T x_i \right\|^2$$

Example of PCA



Eigenvectors and eigenvalues of covariance matrix for *n=1600* inputs in *d=3 dimensions.*

Example: faces

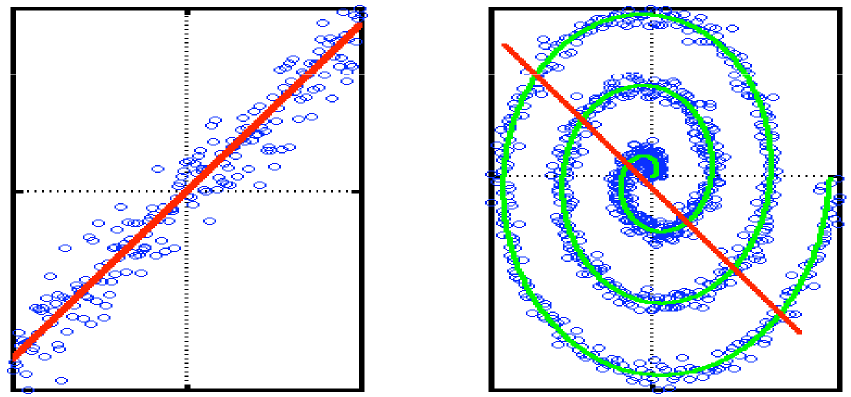


Eigenfaces from 7562 Images:
top left image
is linear
combination
of the rest.

Sirovich & Kirby (1987)
Turk & Pentland (1991)

Properties of PCA

- Strengths:
  - Eigenvector method
  - No tuning parameters
  - Non-iterative
  - No local optima



- Weaknesses:
  - Limited to second order statistics
  - Limited to linear projections

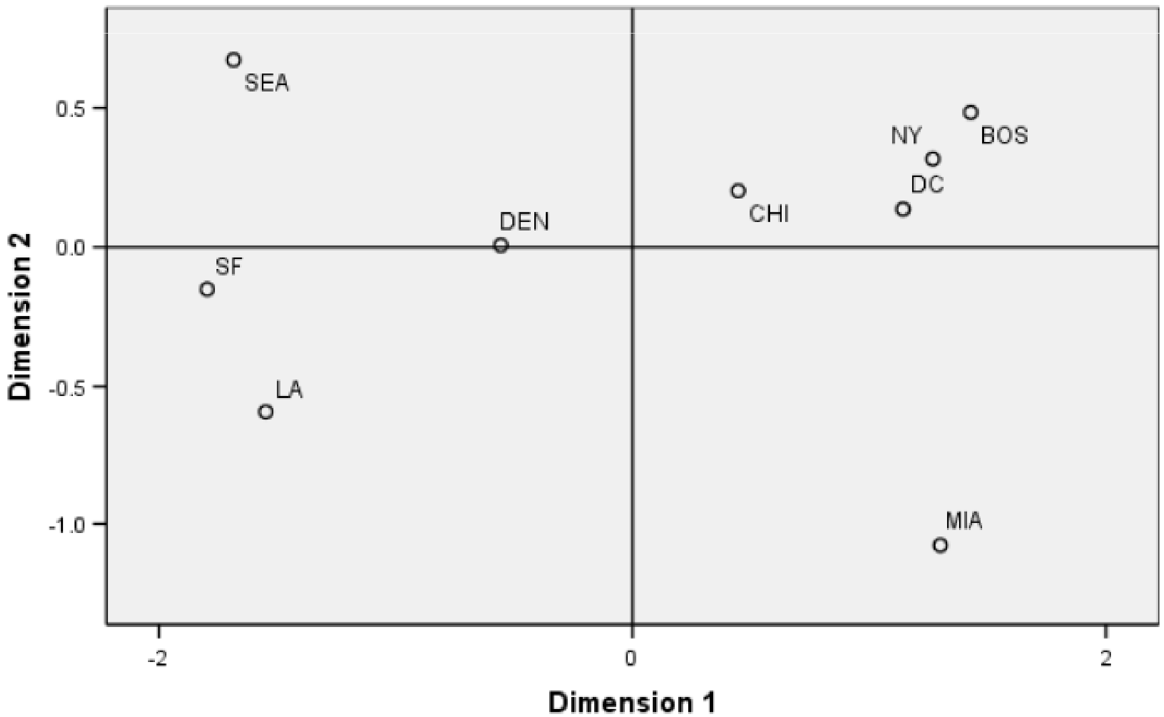
Multidimensional Scaling (MDS)

- MDS attempts to preserve pairwise distances.

- Attempts to construct a configuration of n points in Euclidian space by using the information about the distances between the n patterns.

Example : Distances between US Cities

	BOS	CHI	DC	DEN	LA	MIA	NY	SEA	SF
BOS	0	963	429	1,949	2,979	1,504	206	2,976	3,095
CHI	963	0	671	996	2,054	1,329	802	2,013	2,142
DC	429	671	0	1,616	2,631	1,075	233	2,684	2,799
DEN	1,949	996	1,616	0	1,059	2,037	1,771	1,307	1,235
LA	2,979	2,054	2,631	1,059	0	2,687	2,786	1,131	379
MIA	1,504	1,329	1,075	2,037	2,687	0	1,308	3,273	3,053
NY	206	802	233	1,771	2,786	1,308	0	2,815	2,934
SEA	2,976	2,013	2,684	1,307	1,131	3,273	2,815	0	808
SF	3,095	2,142	2,799	1,235	379	3,053	2,934	808	0



Multidimensional Scaling (MDS)

- A $n \times n$ matrix $\mathcal{D}$ is called a distance or affinity matrix if it is symmetric, $\mathtt{d}_{ii} = 0$, and $\mathtt{d}_{ij} > 0, \quad i \neq j$.

- Given a distance matrix $\mathcal{D}^{(X)}$, MDS attempts to find n data points $y_1, \ldots, y_n$ in d dimensions, such that if $d_{ij}^{(Y)}$ denotes the Euclidean distance between y_i and y_j, then $\mathcal{D}^{Y}$ is similar to $\mathcal{D}^{(X)}$.

Metric MDS

- Metric MDS minimizes

$$\min_{Y} \sum_{i=1}^{n} \sum_{j=1}^{n} (d_{ij}^{(X)} - d_{ij}^{(Y)})^2$$

where

$$d_{ij}^{(X)} = \left\| x_i - x_j \right\| \quad \text{and} \qquad d_{ij}^{(Y)} = \left\| y_i - y_j \right\|.$$

Metric MDS

- The distance matrix $D^{(X)}$ can be converted to a Gram matrix K by

$$K = -\frac{1}{2} H (D^{(X)})^2 H$$

where $H = I - \frac{1}{n} e e^T$ and e is the vector of ones.

Metric MDS

- K is *p.s.d*, thus it can be written as $K = X^T X$
- $\min_Y \sum_{i=1}^{n} \sum_{j=1}^{n} (d_{ij}^{(X)} - d_{ij}^{(Y)})^2$ is equivalent to

$$\min_Y \sum_{i=1}^{n} \sum_{j=1}^{n} (x_i^T x_j - y_i^T y_j)^2$$

- The norm can be converted to a trace:

$$\min_Y Tr\left(X^T X - Y^T Y\right)^2$$

Metric MDS

- Using Singular Value Decomposition we can decompose:

$$X^T X = V \Lambda V^T$$

$$Y^T Y = Q \hat{\Lambda} Q^T$$

- Since $Y^T Y$ is *p.s.d.*, $\hat{\Lambda}$ has no negative values, thus

$$Y = \hat{\Lambda}^{1/2} Q^T$$

Metric MDS

- Returning to the minimization, we can write

$$\min_{Q,\hat{\Lambda}} Tr\left(V\Lambda V^T - Q\hat{\Lambda}Q^T\right)^2$$

$$= \min_{Q,\hat{\Lambda}} Tr\left(\Lambda - \boxed{V^T Q}\hat{\Lambda}Q^T V\right)^2$$

$$\underbrace{}_{=\,G}$$

$$= \min_{G,\hat{\Lambda}} Tr\left(\Lambda - G\hat{\Lambda}G^T\right)^2$$

$$= \min_{G,\hat{\Lambda}} Tr\left(\Lambda^2 + G\hat{\Lambda}G^T G\hat{\Lambda}G^T - 2\Lambda G\hat{\Lambda}G^T\right)$$

Metric MDS

- For a fixed $\hat{\Lambda}$ we can minimize for G, obtaining

$$G = I$$

$$\min_{\hat{\Lambda}} Tr\left(\Lambda^2 + \hat{\Lambda}^2 - 2\Lambda\hat{\Lambda}G\right)$$

$$= \min_{\hat{\Lambda}} Tr\left(\Lambda - \hat{\Lambda}\right)^2$$

Metric MDS

- To make the two matrices Λ and $\hat{\Lambda}$ similar, we can make $\hat{\Lambda}$ to be the top d diagonal elements of Λ.
- Also $G = V^T Q$ and $G = I$ imply that $V = Q$.
- Therefore,

$$Y = \hat{\Lambda}^{1/2} Q^T \quad \Longrightarrow \quad Y = \hat{\Lambda}^{1/2} V^T$$

where V comprises the eigenvectors of $X^T X$ corresponding to the top d eigenvalues and $\hat{\Lambda}$ comprises the top d eigenvalues of $X^T X$.

Interpreting MDS

- Eigenvectors:

  Ordered, scaled, and truncated to yield low dimensional embedding.

- Eigenvalues:

  Measure how each dimension contributes to dot products.

- Estimated dimensionality:

  Number of significant (nonnegative) eigenvalues.

Relation to PCA

	PCA	MDS
Spectral Decomposition	Covariance matrix (D x D)	Gram matrix (n x n)
Eigenvalues	Matrices share nonzero eigenvalues up to constant factor	
Results	Same	
Computation	$O((n+d)D^2)$	$O((D+d)n^2)$

Non-Metric MDS

- Transform pairwise distances: $\delta_{ij} \rightarrow g(\delta_{ij})$
  - Transformation: nonlinear, but monotonic.
  - Preserves rank order of distances.

- Find vectors y_i such that $\left\| y_i - y_j \right\| \approx g(\delta_{ij})$

$$Cost = \min_{y} \sum_{ij} \left(g(\delta_{ij}) - \left\| y_i - y_j \right\| \right)^2$$

Non-Metric MDS

- Possible objective function:

$$Cost = \sum_{i\,j} \left(\frac{\| \mathbf{x}_i - \mathbf{x}_j \| - \| \mathbf{y}_i - \mathbf{y}_j \|}{\| \mathbf{x}_i - \mathbf{x}_j \|} \right)^2$$

Properties of non-metric MDS

- Strengths
  - Relaxes distance constraints.
  - Yields nonlinear embeddings.
- Weaknesses
  - Highly nonlinear, iterative optimization with local minima.
  - Unclear how to choose distance transformation.