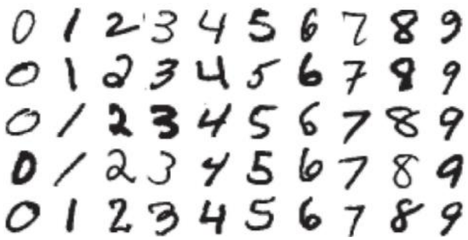
COMS 4771
Dimensionality Reduction

Nakul Verma

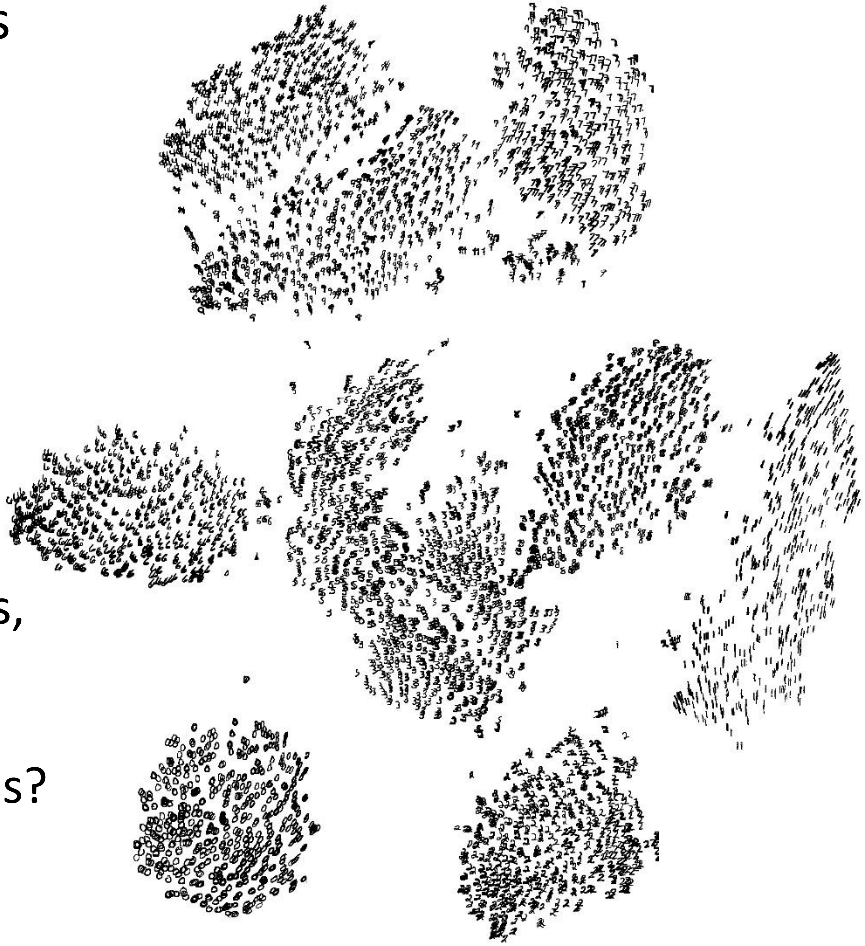
Example: Handwritten digits

Handwritten digit data, but with no labels



What can we do?

- Suppose know that there are 10 groupings, can we find the groups?
- What if we don’t know there are 10 groups?
- How can we discover/explore other structure in such data?



A 2D visualization of digits dataset

Dimensionality Reduction

Data: $\vec{x}_1, \vec{x}_2, \ldots \vec{x}_n \in \mathbf{R}^d$

Goal: find a 'useful' transformation $\phi : \mathbf{R}^d \to \mathbf{R}^k$ that helps in the downstream prediction task.

Some previously seen useful transformations:

- z-scoring $(x_1, \ldots, x_d) \mapsto \left(\frac{x_1 - \mu_1}{\sigma_1}, \ldots, \frac{x_d - \mu_d}{\sigma_d}\right)$

  Keeps same dimensionality but with better scaling

- Kernel transformations.

  Higher dimensionality, making data linearly separable

What are other desirable feature transformations?

How about lower dimensionality while keeping the relevant information?

Principal Components Analysis (PCA)

Data: $\vec{x}_1, \vec{x}_2, \ldots \vec{x}_n \in \mathbf{R}^d$

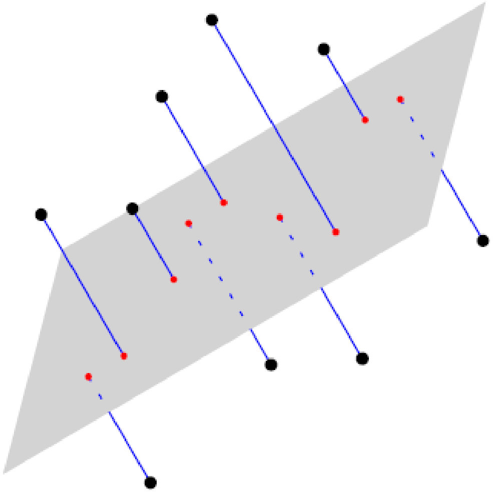
Goal: find the best **linear** transformation $\phi : \mathbf{R}^d \rightarrow \mathbf{R}^k$ that best maintains reconstruction accuracy.

Equivalently, minimize aggregate residual error

Define: $\Pi^k : \mathbf{R}^d \rightarrow \mathbf{R}^d$ *k-dimensional orthogonal linear projector*

$$\underset{\Pi^k}{\text{minimize}} \quad \frac{1}{n}\sum_{i=1}^{n}\left\|\vec{x}_i - \Pi^k(\vec{x}_i)\right\|^2$$

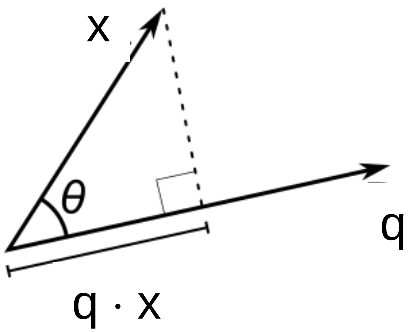
How do we optimize this?



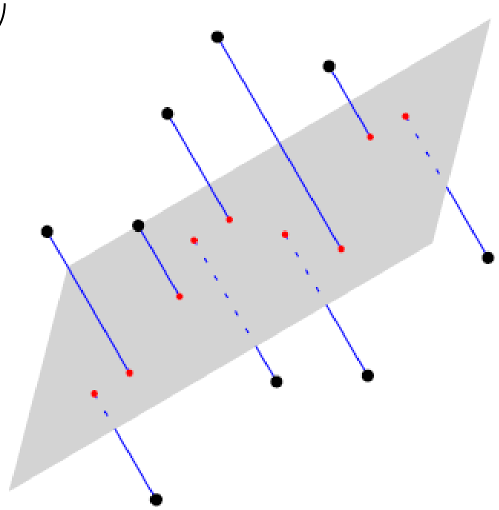
Dimensionality Reduction via Projections

A *k* dimensional subspace can be represented by $\vec{q}_1, \ldots, \vec{q}_k \in \mathbf{R}^d$ orthonormal vectors.

The projection of any $\vec{x} \in \mathbf{R}^d$ in the $\text{span}(\vec{q}_1, \ldots, \vec{q}_k)$ is given by



$$\sum_{i=1}^{k} (\vec{q}_i \cdot \vec{x})\, \vec{q}_i = \underbrace{\left(\sum_{i=1}^{k} \vec{q}_i \vec{q}_i^{\mathsf{T}} \right)}_{\Pi^k} \vec{x}$$



To represent it in $\mathbf{R}^k$ (using basis $\vec{q}_1, \ldots, \vec{q}_k$) the coefficients simply are: $(\vec{q}_1 \cdot \vec{x}), \ldots, (\vec{q}_k \cdot \vec{x})$

PCA: k = 1 case

If projection dimension k = 1, then looking for a q such that

$$\textit{minimize}_{\|q\|=1} \quad \frac{1}{n}\sum_{i=1}^{n} \left\| \vec{x}_i - (\vec{q}\,\vec{q}^{\top})\vec{x}_i \right\|^2$$

$$\frac{1}{n}\sum_{i=1}^{n} \left\| \vec{x}_i - (\vec{q}\,\vec{q}^{\top})\vec{x}_i \right\|^2 = \Big(\frac{1}{n}\sum_{i=1}^{n} \vec{x}_i^{\top}\vec{x}_i\Big) - \vec{q}^{\top}\Big(\frac{1}{n}\sum_{i=1}^{n} \vec{x}_i\vec{x}_i^{\top}\Big)\vec{q}$$

$$\propto -\vec{q}^{\top}\Big(\frac{1}{n}XX^{\top}\Big)\vec{q}$$

$$\left(X := \begin{bmatrix} | & & | \\ x_1 & \cdots & x_n \\ | & & | \end{bmatrix} \right)$$

Equivalent formulation:

$$\textit{maximize}_{\|q\|=1} \quad \vec{q}^{\top}\Big(\frac{1}{n}XX^{\top}\Big)\vec{q}$$

How to solve?

Optimizing $q^\top M q$

For a symmetric PSD matrix M, how to optimize for *maximize*$_{\|q\|=1}$ $q^\top M q$

Recall, since M is symmetric PSD, by spectral decomposition theorem:

$$M = \sum_i \lambda_i v_i v_i^\top$$

$v_1,\ldots,v_d$ are an othonormal set of eigenvectors of M

$\lambda_1 \geq \ldots \geq \lambda_d \geq 0$ are the corresponding eigenvalues

Thus for any unit length *q*, $q^\top M q = \sum_i \lambda_i (q \cdot v_i)^2$ (where $\sum_i (q \cdot v_i)^2 = 1$)

Let, $\alpha_i := (q \cdot v_i)^2$ then the optimization becomes

$$\max_{\alpha_i} \sum_i \lambda_i \alpha_i \qquad \text{s.t.} \sum_i \alpha_i = 1$$

with the optimal solution as $\alpha_1=1, \alpha_2 = \alpha_3 = \ldots = \alpha_n = 0$, or equivalently $q = v_1$

Therefore

maximize$_{\|q\|=1}$ $\vec{q}^\top \left(\frac{1}{n} X X^\top\right) \vec{q}$ ***is maximized by the top eigenvector of matrix (1/n) $XX^\top$!***

PCA: k = 1 case

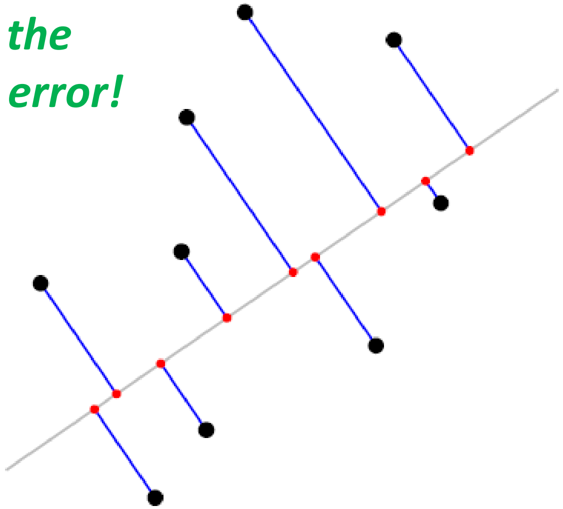
maximize$_{||q||=1}$ $\vec{q}^{\top}\left(\frac{1}{n}XX^{\top}\right)\vec{q}$

Covariance of data (if mean = 0)

For any q the quadratic form $\vec{q}^{\top}\left(\frac{1}{n}XX^{\top}\right)\vec{q}$ is the empirical variance of data in the direction q, ie, of data $\vec{q}^{\top}\vec{x}_1, \ldots, \vec{q}^{\top}\vec{x}_n$

why?

Therefore, the top eigenvector solution implies that the direction of maximum variance minimizes the residual error!



What about general k?

PCA: general *k* case

$$\arg\min_{\substack{Q\in\mathbf{R}^{d\times k}\\ Q^\top Q=I}} \frac{1}{n}\sum_{i=1}^{n}\left\|\vec{x}_i - QQ^\top\vec{x}_i\right\|^2 = \arg\min_{\substack{Q\in\mathbf{R}^{d\times k}\\ Q^\top Q=I}} \frac{1}{n}\left\|X - QQ^\top X\right\|_{\mathrm{F}}^2$$

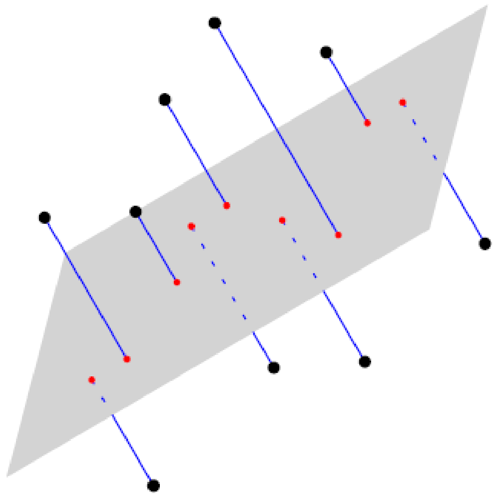
$$\left(Q := \begin{bmatrix} | & & | \\ q_1 & \cdots & q_k \\ | & & | \end{bmatrix} \right)$$

$$\propto \arg\max_{\substack{Q\in\mathbf{R}^{d\times k}\\ Q^\top Q=I}} \operatorname{tr}\left(Q^\top\left(\frac{1}{n}XX^\top\right)Q\right)$$

Solution: the top k eigenvectors of the matrix (1/n)$XX^\top$!

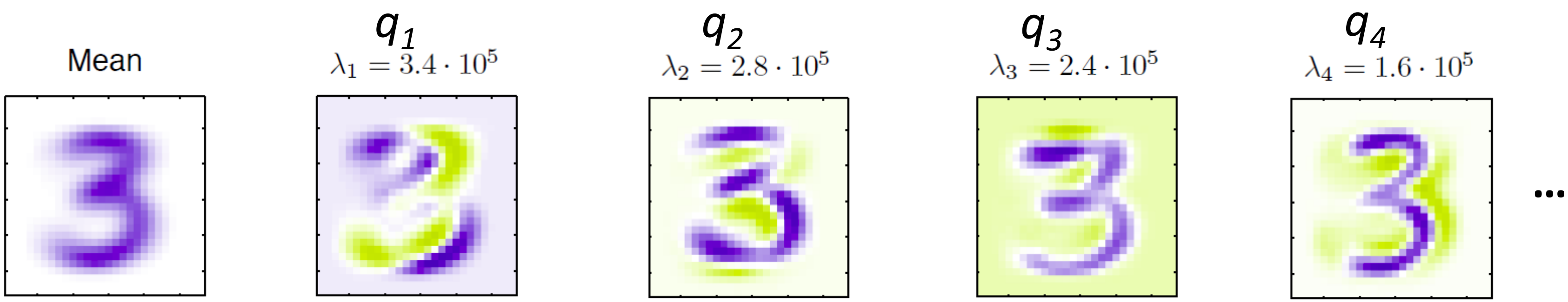
$$\operatorname{tr}\left(Q^\top\left(\frac{1}{n}XX^\top\right)Q\right) = \sum_{i=1}^{k}\text{empirical variance of } \vec{q}_i^{\,\top} x$$

k-dimensional subspace preserving maximum amount of variance

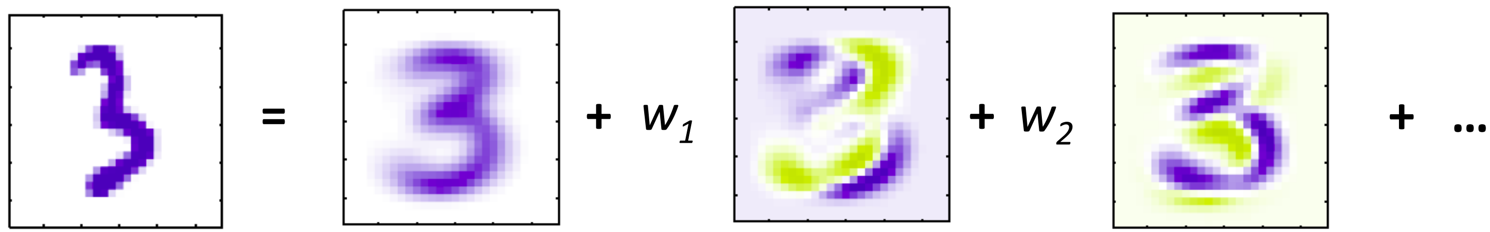


PCA: Example Handwritten Digits

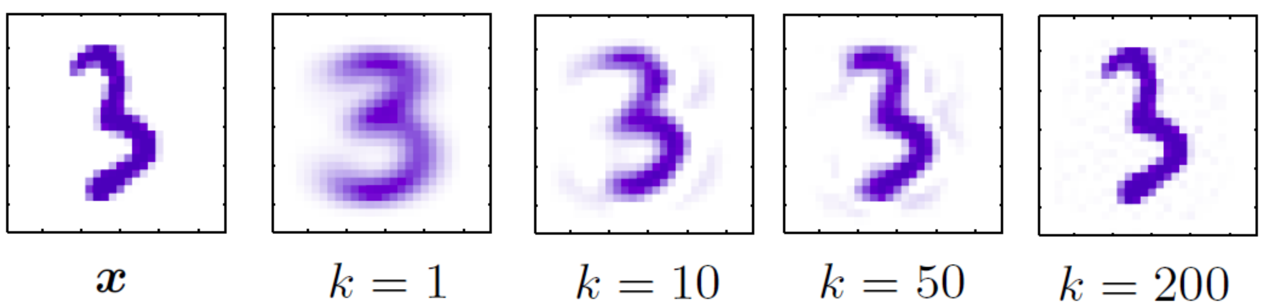
Images of handwritten 3s in $\mathbf{R}^{784}$



Any example:



Data Reconstruction:



We can compress the each datapoint to just k numbers!

Other Popular Dimension Reduction Methods

Multi-dimensional Scaling

Independent Component Analysis (ICA) (for blind source separation)

Non-negative matrix factorization (to create additive models)

Dictionary Learning

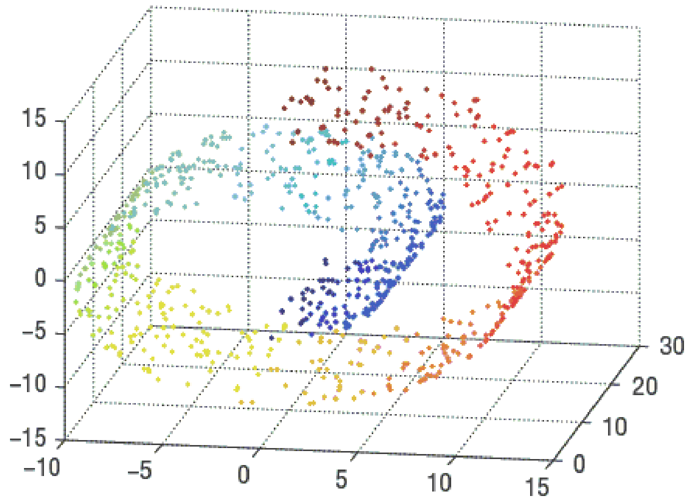
Random Projections

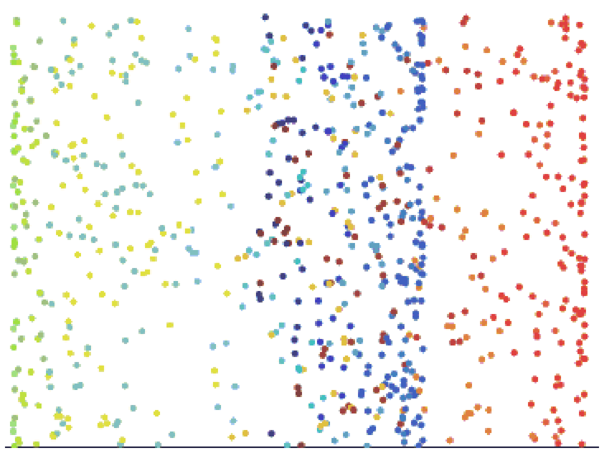
…

*All of them are **linear** methods*

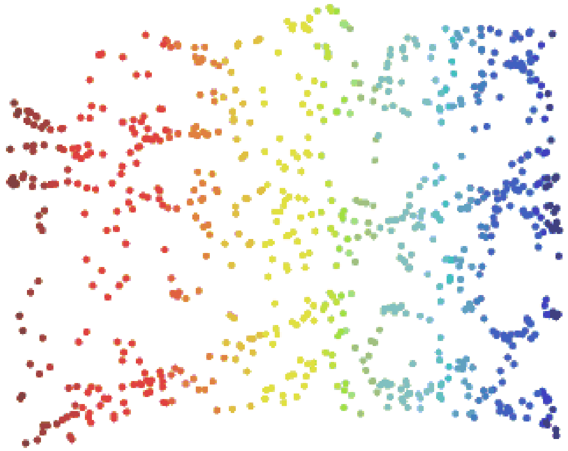
Non-Linear Dimensionality Reduction

Consider non-linear data





Linear embedding



non-linear embedding

Non-Linear Dimensionality Reduction

Basic optimization criterion:

Find an embedding that:

- Keeps neighboring points close
- Keeps far-off points far

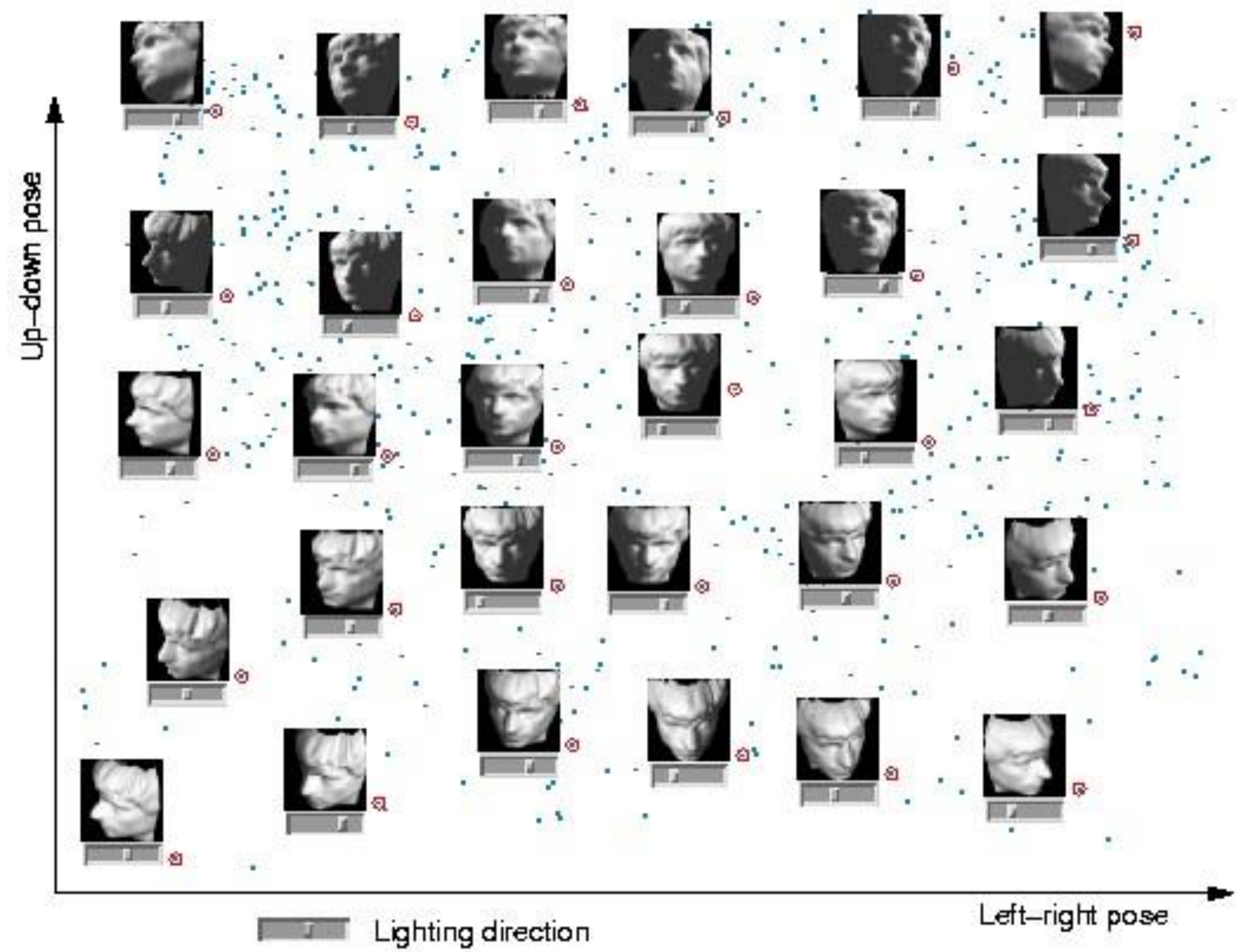
Example variation 1:

Distort neighboring distances by at most ($1 \pm \varepsilon$) factor, while maximizing non-neighbor distances.

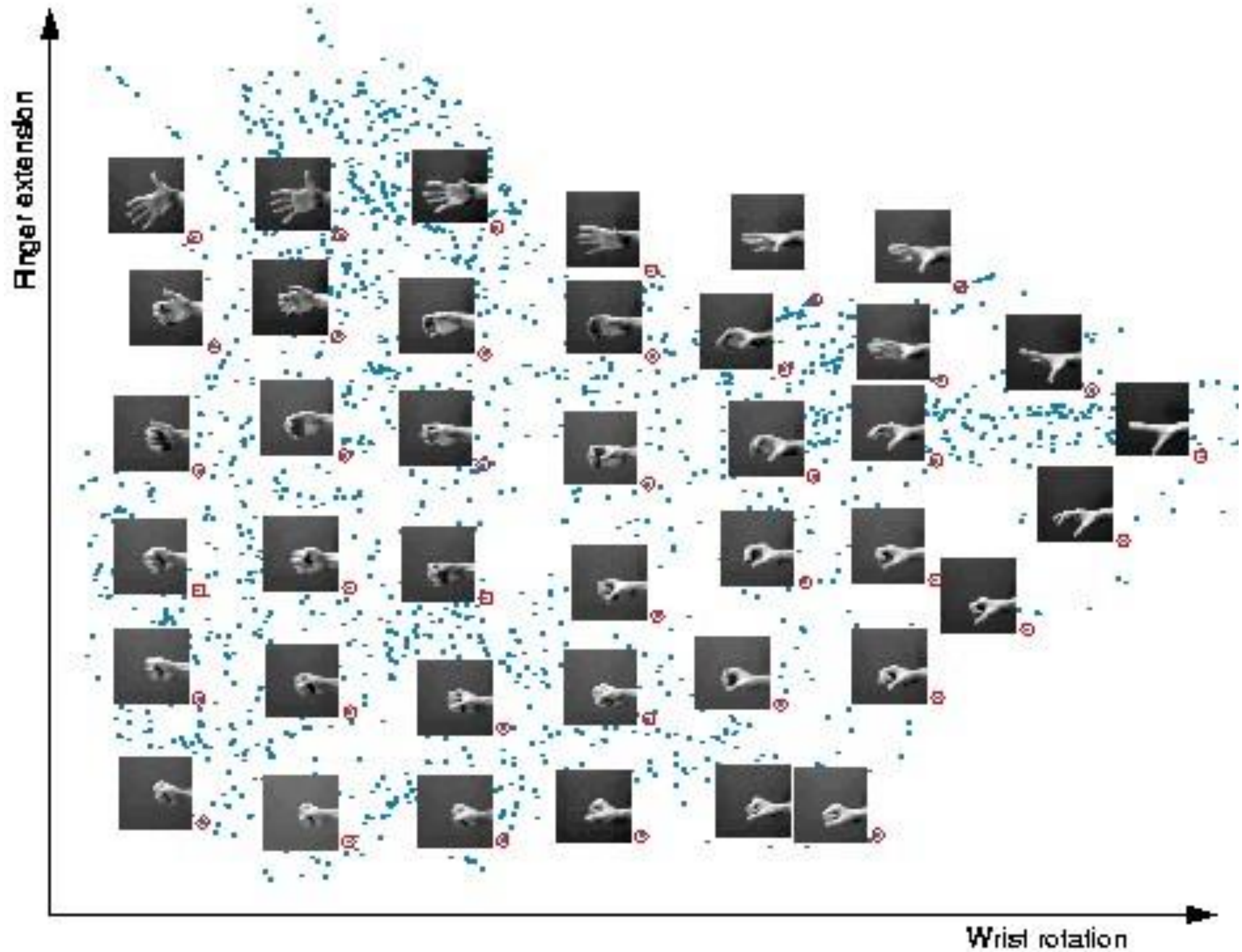
Example variation 2:

*Compute **geodesic** (local hop) distances, and find an embedding that best preserves geodesics.*

Non-linear embedding: Example



Non-linear embedding: Example



Popular Non-Linear Methods

Locally Linear Embedding (LLE)

Isometric Mapping (IsoMap)

Laplacian Eigenmaps (LE)

Local Tangent Space Alignment (LTSA)

Maximum Variance Unfolding (MVU)

…

What We Learned…

- Dimensionality Reduction
  - Linear vs non-linear Dimensionality Reduction
- Principal Component Analysis

Questions?