University of Dayton

Feature Extraction: Principal Component Analysis

CPS 563 – Data Visualization

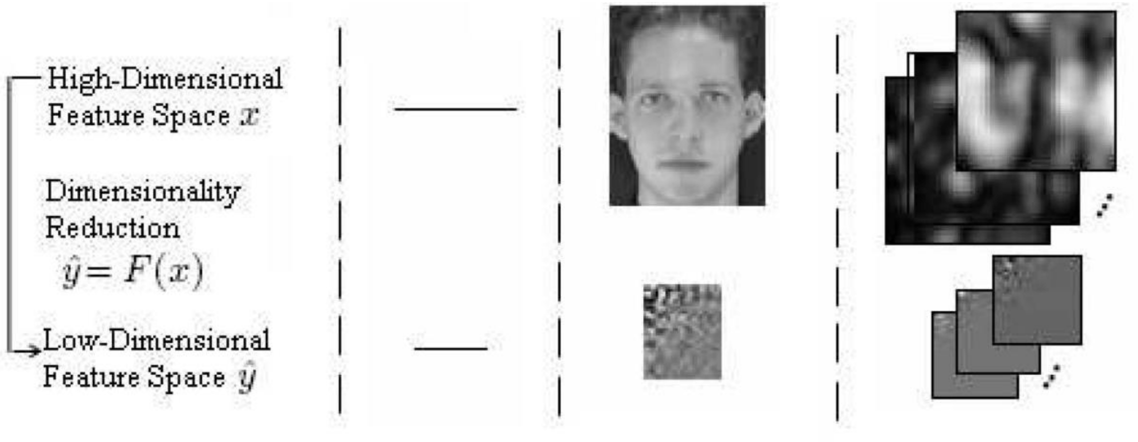
Dr. Tam Nguyen

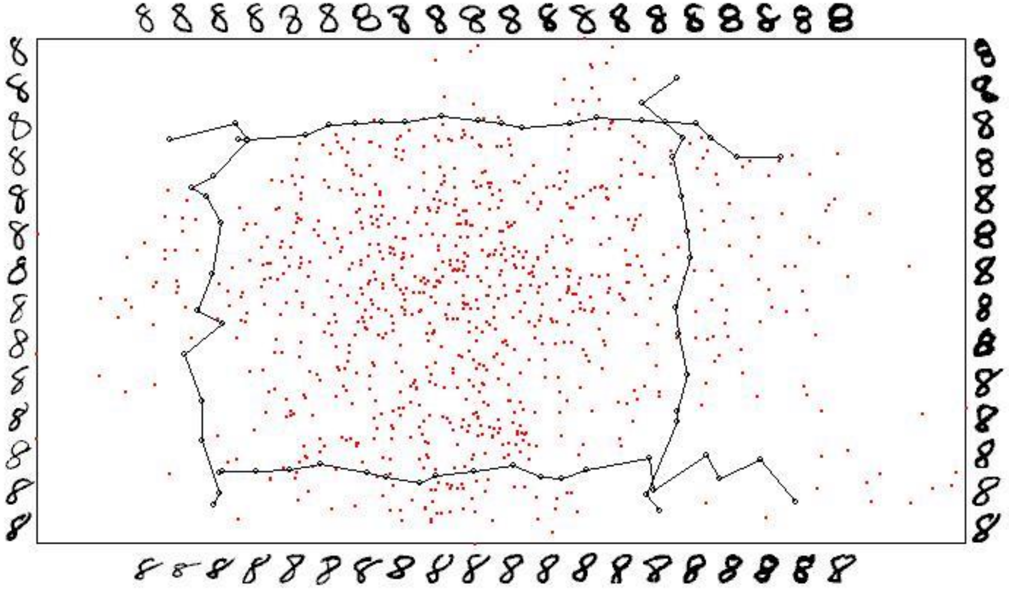
tamnguyen@udayton.edu

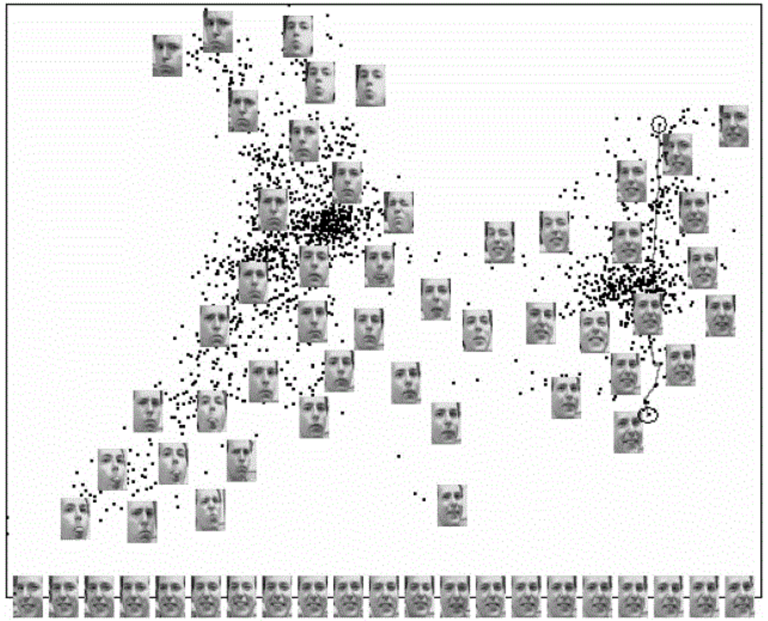
What is Feature Extraction?

- Feature extraction refers to the mapping of the original **HIGH-DIMENSIONAL** data into a **LOW-DIMENSIONAL** space
  - Criterion for feature reduction can be different based on different problem setting
    - Unsupervised setting: minimize the information loss (not use class information)
    - Supervised setting: maximize the class discrimination (use class information)
- Also called dimensionality reduction, feature reduction

What is Feature Extraction?

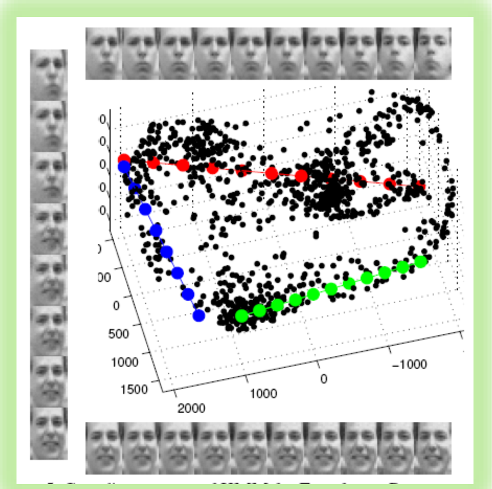






Why Feature Extraction?

- Many pattern recognition techniques may not be effective for high-dimensional data
  - **Curse of Dimensionality**
  - Testing efficiency degrades rapidly as the dimension increases



- The **intrinsic** dimension may be small
  - For example, the number of genes responsible for a certain type of disease may be small
  - Another example, images of one person from left view to right view can be described by one dimension

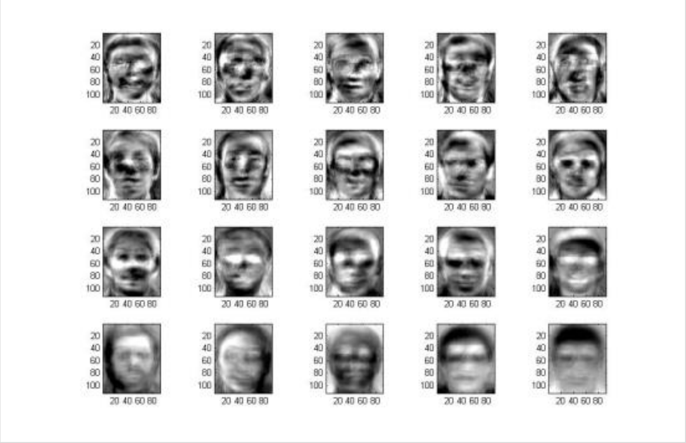
Why Feature Extraction?

- **Visualization**: projection of high-dimensional data onto 2D or 3D

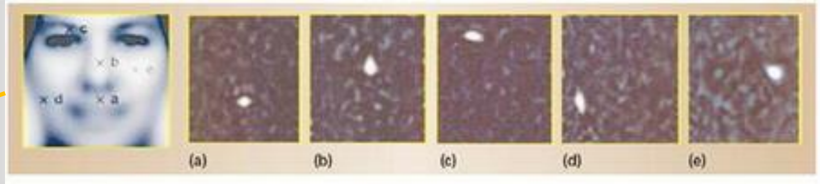
- **Data compression**: efficient storage and retrieval

- **Noise removal**: positive effect on testing accuracy

Feature Extraction vs. Feature Selection



- Feature Extraction
  - All original features are used
  - The transformed features are linear combinations of the original features.

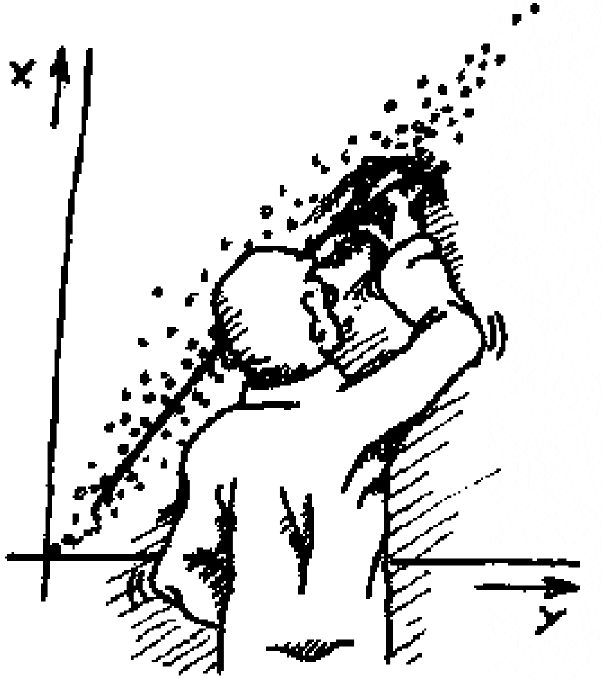


- Feature Selection
  - Only a subset of the original features are used.

Feature Extraction Algorithms: Principal Component Analysis

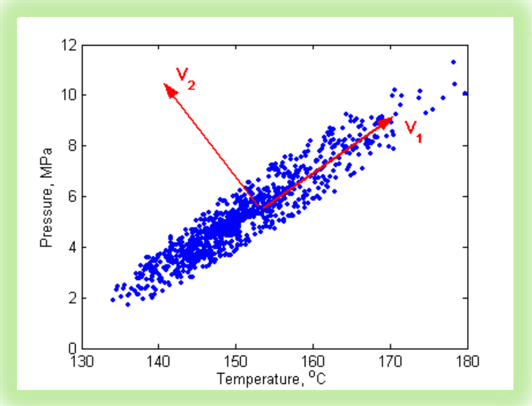
Karl Pearson

- probably the most widely-used and well-known of the "standard" multivariate methods
- invented by Pearson (1901) and Hotelling (1933)
- first applied in ecology by Goodall (1954) under the name "factor analysis" ("principal factor analysis" is a synonym of PCA).



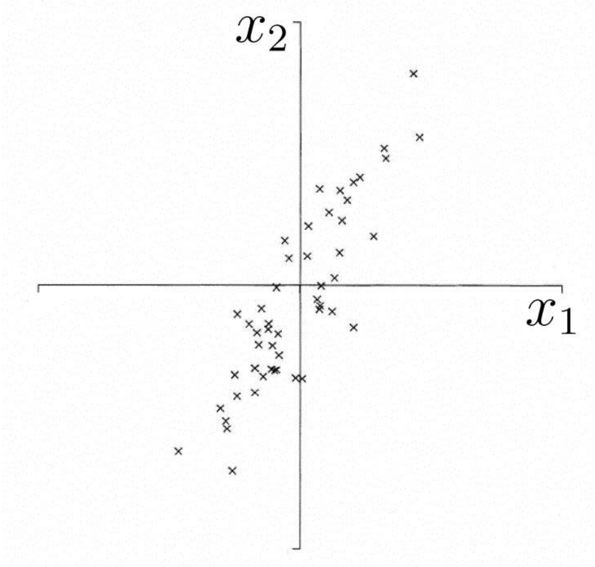
What is Principal Component Analysis?

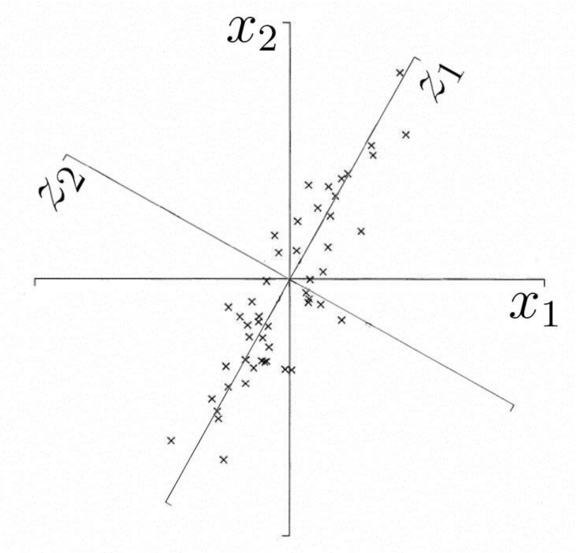
- Principal component analysis (PCA)
  - Reduce the dimensionality of a data set by finding a new set of variables, smaller than the original set of variables
  - Capture big (principal) variability in the data and ignore small variability



- Variation in samples
  - The new variables, called principal components (PCs), are ordered by variations corresponding to different PCs.

Geometric Picture of Principal Components

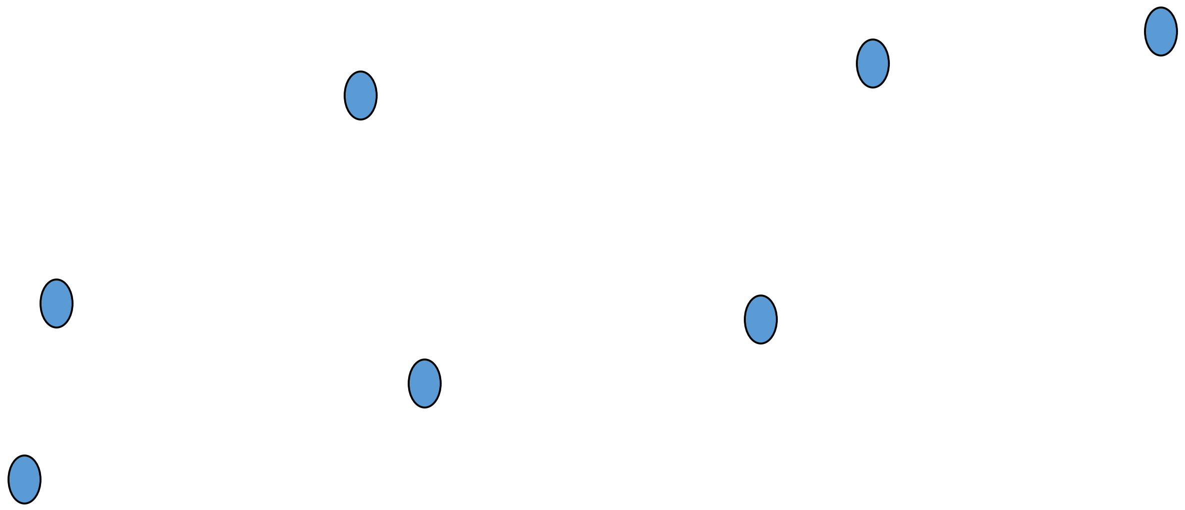




- The 1st PC $\boldsymbol{Z}_1$ is a minimum distance fit to a line in X space
- The 2nd PC $\boldsymbol{Z}_2$ is a minimum distance fit to a line in the plane orthogonal to the 1st PC

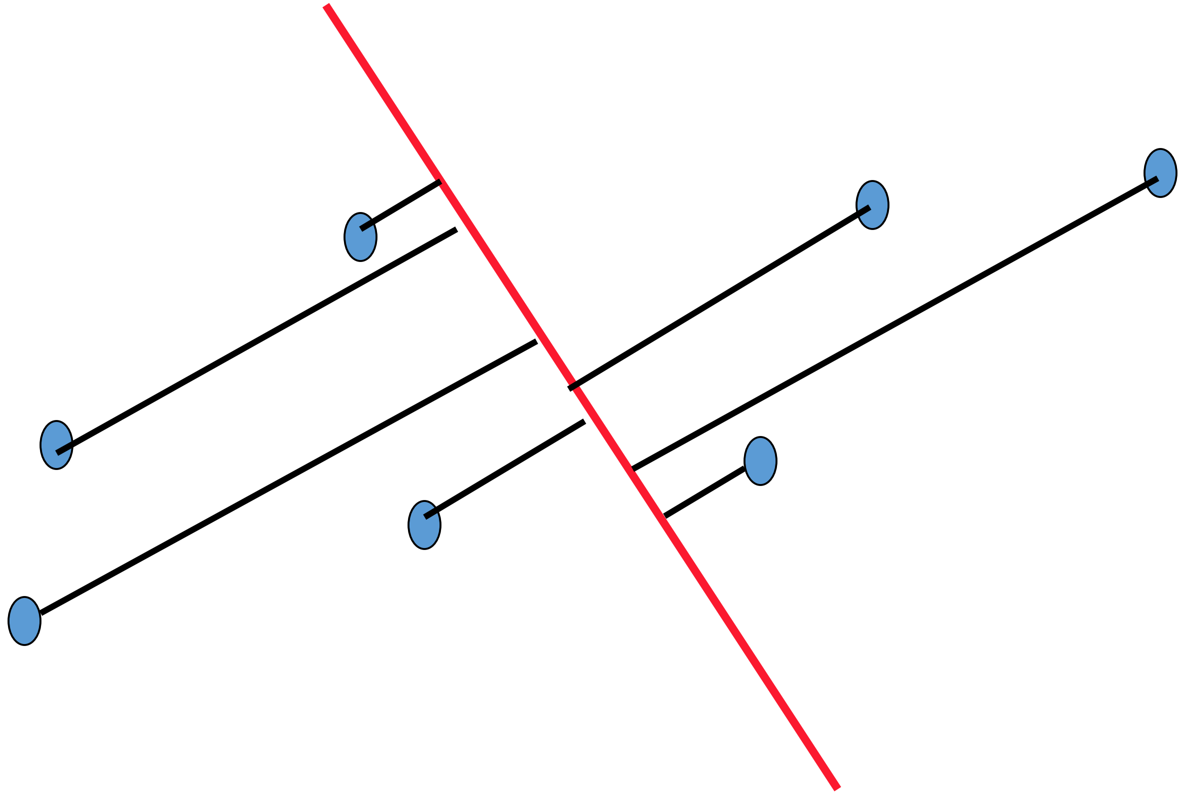
PCs are a series of linear least squares fits to a sample set, each orthogonal to all the previous ones.

Geometric Picture of Principal Components



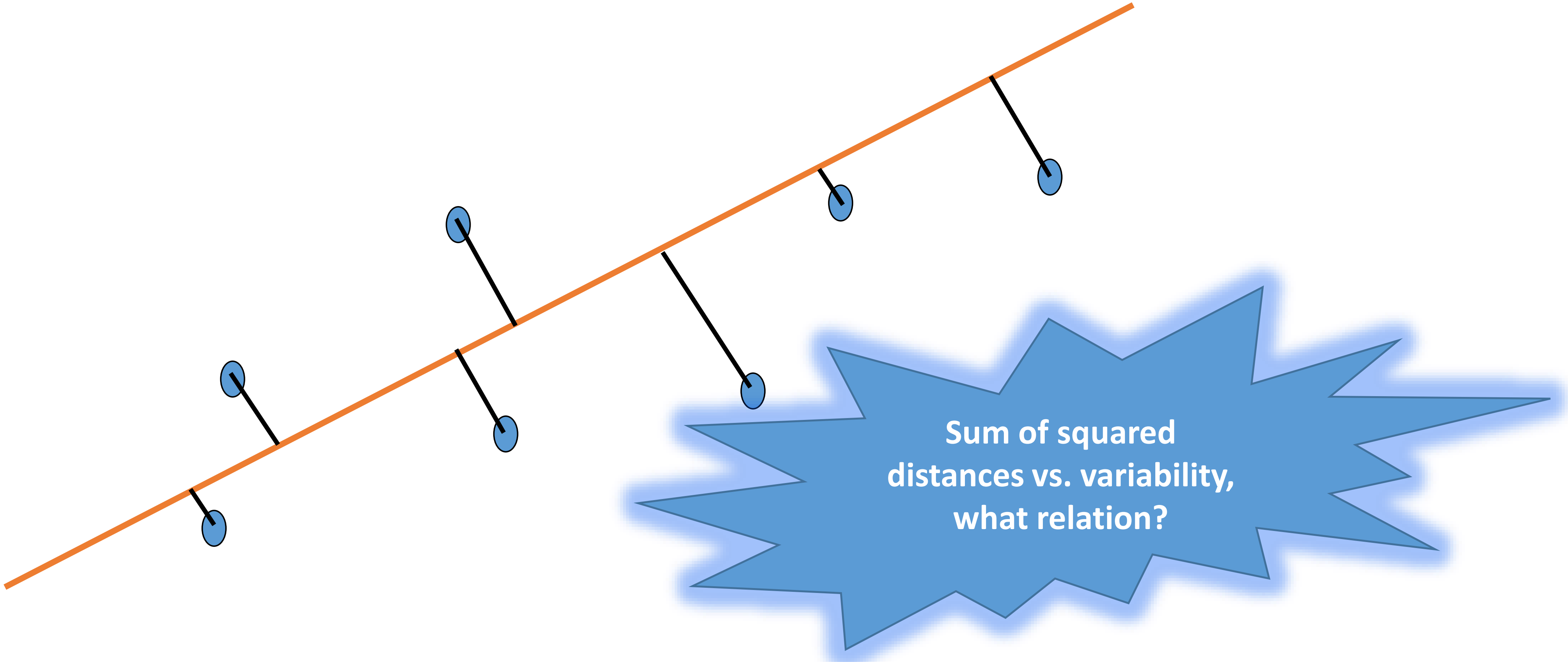
Sample Points

Geometric Picture of Principal Components



linear least squares fit: Large

Geometric Picture of Principal Components



Sum of squared distances vs. variability, what relation?

linear least squares fit: Small

Algebraic Definition of PCs

Given a sample set of *n* observations on a vector of *d* variables

$$\left\{x_1, x_2, \cdots, x_n\right\} \subset \Re^d$$

define the first principal component by the linear projection a_1

$$z_1 = a_1^T x$$

where the vector $a_1 = (a_{11}, a_{21}, \cdots, a_{d1})^T$

is chosen such that $\mathrm{var}[z_1]$ is maximum.

Algebraic Definition of PCs

To find a_1 first note that

$$\operatorname{var}[z_1] = E((z_1 - \overline{z}_1)^2) = \frac{1}{n}\sum_{i=1}^{n}\left(a_1^T x_i - a_1^T \overline{x}\right)^2$$

$$= \frac{1}{n}\sum_{i=1}^{n} a_1^T \left(x_i - \overline{x}\right)\left(x_i - \overline{x}\right)^T a_1 = a_1^T S a_1$$

where $S = \frac{1}{n}\sum_{i=1}^{n}\left(x_i - \overline{x}\right)\left(x_i - \overline{x}\right)^T$

is the covariance matrix,

$\overline{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$ is the mean.

Algebraic Derivation of PCs

To find a_1 that maximizes $\operatorname{var}[z_1]$ subject to $a_1^T a_1 = 1$

Why?

Let λ be a Lagrange multiplier

$$L = a_1^T S a_1 - \lambda(a_1^T a_1 - 1)$$

$$\Rightarrow \frac{\partial}{\partial a_1} L = S a_1 - \lambda a_1 = 0$$

$$\Rightarrow (S - \lambda I_d) a_1 = 0$$

therefore a_1 is an eigenvector of S

Why?

corresponding to the largest eigenvalue $\lambda = \lambda_1$.

Eigenvalues and eigenvectors

- Given a square matrix **A**, if it occurs

$$A\mathbf{v} = \lambda\mathbf{v}$$

, then *v* is an **eigenvector** of the linear transformation **A** and the scale factor λ is the **eigenvalue** corresponding to that eigenvector. The equation above is the eigenvalue equation for the matrix A.

Algebraic Derivation of PCs

Similarly, a_2 is also an eigenvector of S

whose eigenvalue $\lambda = \lambda_2$ is the second largest.

In general

$$\operatorname{var}[z_k] = a_k^T S a_k = \lambda_k$$

- The *k*th largest eigenvalue of S is the variance of the *k*th PC.
- The *k*th PC z_k retains the *k*th greatest variation in the samples

Algebraic Derivation of PCs

- Main steps for computing PCs
  - Calculate the covariance matrix S.
  - Compute its eigenvectors: $\{a_i\}_{i=1}^{d}$
  - The first *p* eigenvectors $\{a_i\}_{i=1}^{p}$ form the *p* PCs.
  - The transformation matrix *G* consists of the *p* PCs:

How to do in Matlab?

$$G \leftarrow [a_1, a_2, \cdots, a_p]$$

$$y = G^T x$$

Practical Computation of PCA

- In practice, we compute the PCs via singular value decomposition (SVD) on the centered data matrix.

- Form the centered data matrix:

$$X_{d,n} = [(x_1 - \bar{x}) \ldots (x_n - \bar{x})]$$

- Compute its SVD:

$$X = U_{d,d} D_{d,n} (V_{n,n})^T$$

How to do in MATLAB?
What are the advantages?

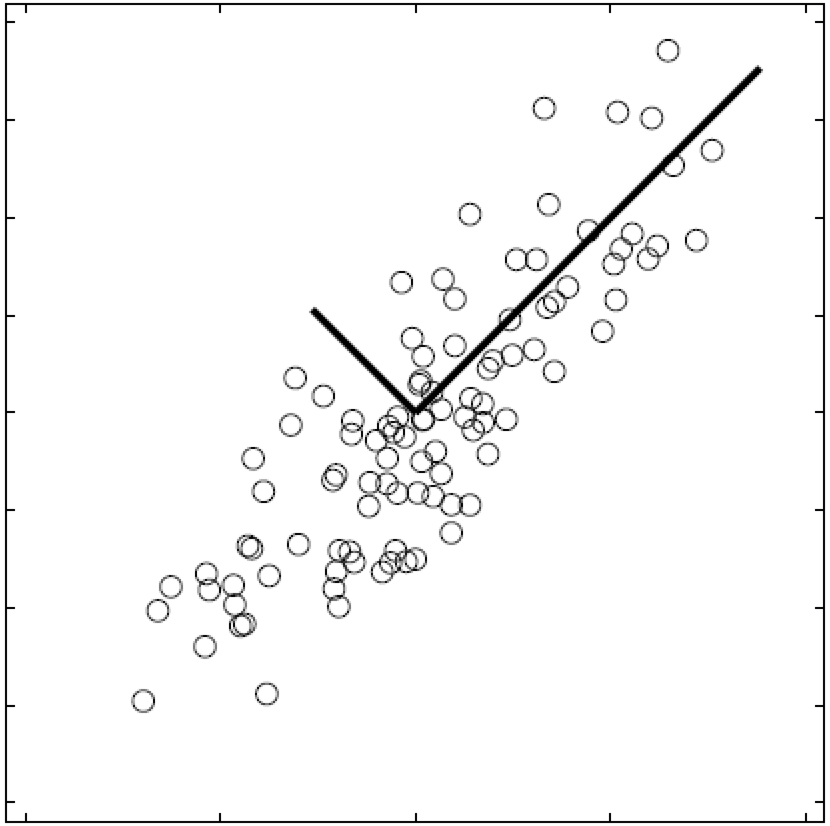
- *U* and *V* are orthogonal matrices, *D* is a diagonal matrix

How many principal components to keep?

- To choose *p* based on percentage of variation to retain, we can use the following criterion (smallest *p*):

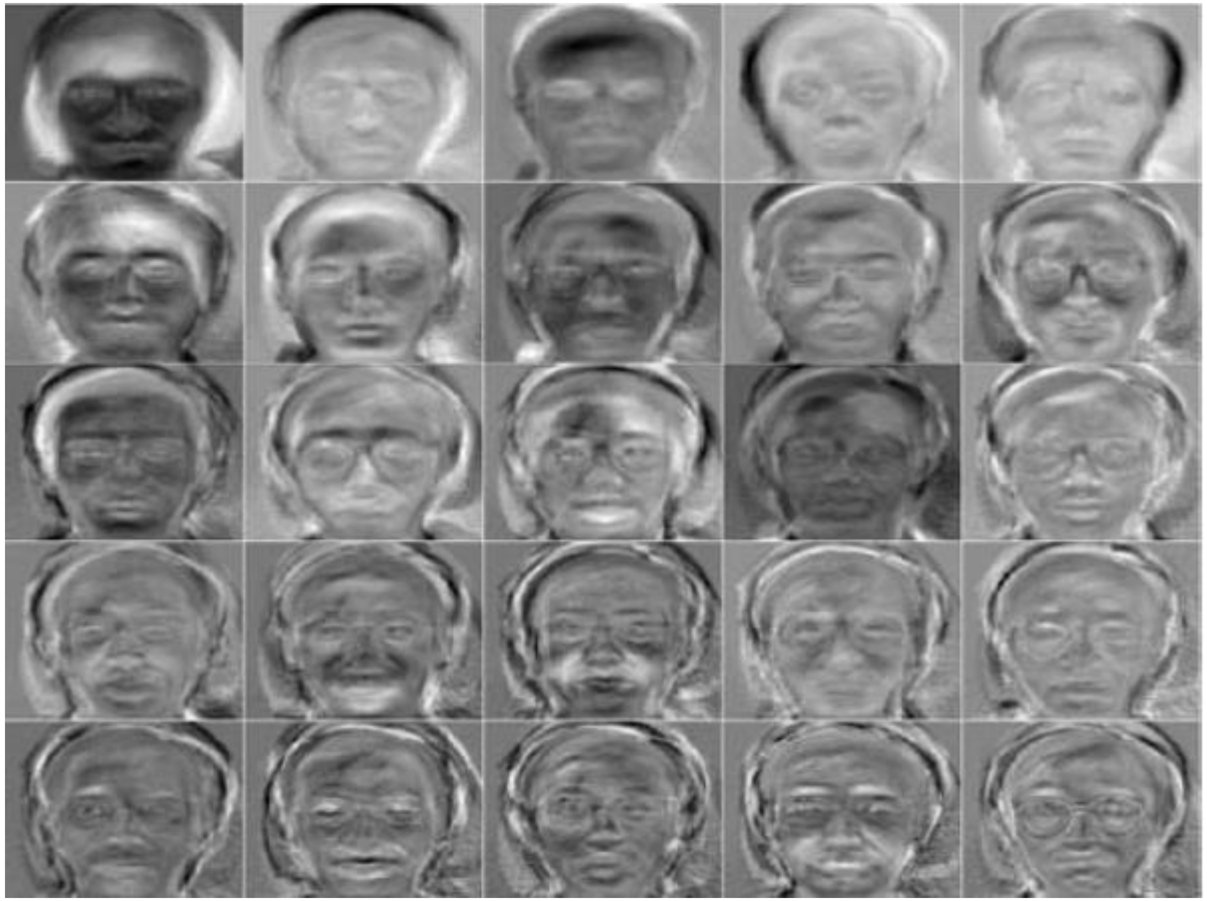
$$\frac{\sum_{i=1}^{p}\lambda_i}{\sum_{i=1}^{d}\lambda_i} \geq Threshold\ (e.g., 0.95)$$

Visualize PCs



Data points are represented in a rotated orthogonal coordinate system: the origin is the mean of the data points and the axes are provided by the eigenvectors.

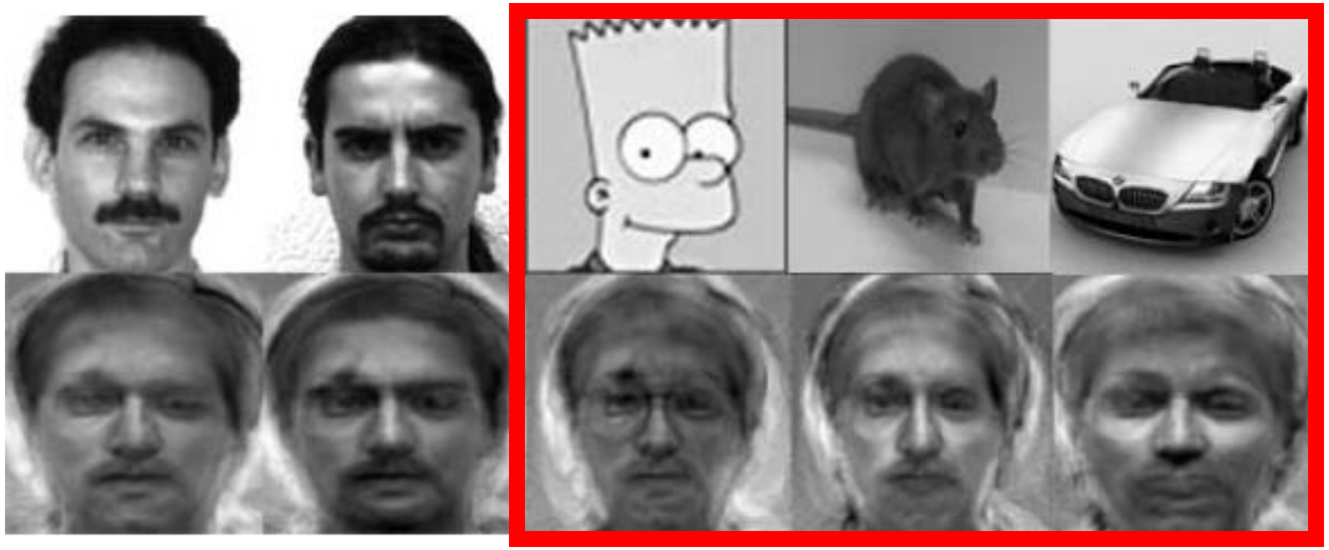
Visualize PCs



Face images

What shall happen for Other Objects

- For faces of person not in training set or non-faces (upper), what shall the reconstruction results (bottom) be?



PCA Conclusions

- PCA
  - finds orthonormal basis for data
  - Sorts dimensions in order of “importance”
  - Discard low significance dimensions

- Uses:
  - Get compact description
  - Ignore noise
  - Improve classification (hopefully)

Q&A