Dimension Reduction

Jiayu Zhou

[1]Department of Computer Science and Engineering
Michigan State University
East Lansing, MI USA

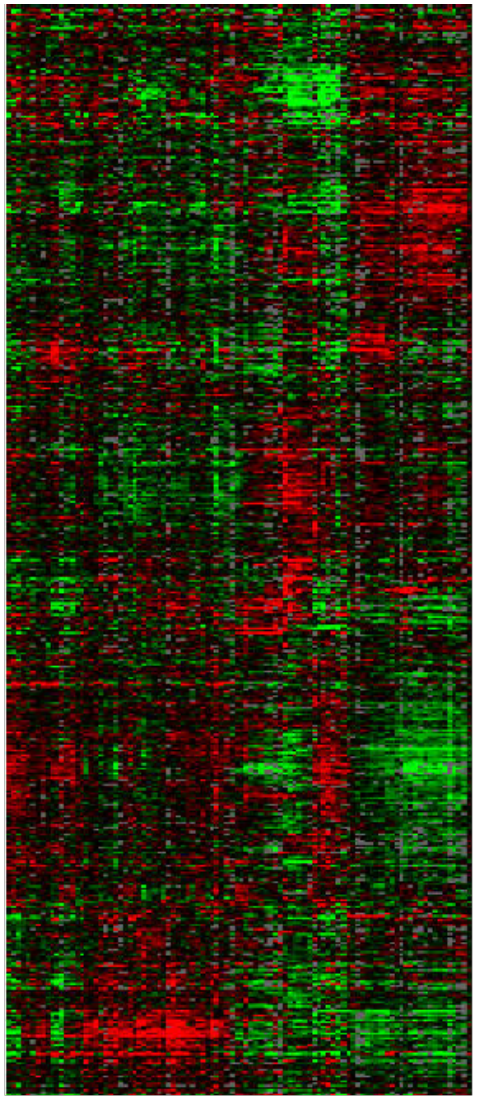
April 3, 2016

Some slides from "Principal Component Analysis." by Frank Wood.

Table of contents

Feature Reduction

High-Dimensional Data



Gene expression

Face images

Challenges with High-Dimensional Data

- Most machine learning and data mining techniques may not be effective for high-dimensional data
  - **Curse of Dimensionality**
  - Query accuracy and efficiency degrade rapidly as the dimension increases.

Challenges with High-Dimensional Data

- Most machine learning and data mining techniques may not be effective for high-dimensional data
  - **Curse of Dimensionality**
  - Query accuracy and efficiency degrade 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.

Feature Reduction

- **Feature reduction** refers to the mapping of the original high-dimensional data onto a lower-dimensional space.
- Criterion for feature reduction can be different based on different problem settings.
  - Supervised setting: maximize the class discrimination

Feature Reduction

- **Feature reduction** refers to the mapping of the original high-dimensional data onto a lower-dimensional space.
- Criterion for feature reduction can be different based on different problem settings.
  - Supervised setting: maximize the class discrimination
  - Unsupervised setting: minimize the information loss
- Given a set of data points of p variables $\{x_1, x_2, \ldots, x_n\}$, compute the linear transformation (projection)

$$A \in \mathbb{R}^{p \times d} : x \in \mathbb{R}^p \rightarrow y = A^T x \in \mathbb{R}^d \quad \text{(typically } d \ll p\text{)}$$

Other benefits of feature reduction

- **Visualization**: projection of high-dimensional data onto 2D or 3D.
- **Data compression**: efficient storage and retrieval.
- **Noise removal**: positive effect on query accuracy.

Applications of feature reduction

- Face recognition
- Handwritten digit recognition
- Text mining
- Image retrieval
- Microarray data analysis
- Protein classification

- Unsupervised
  - Latent Semantic Indexing (LSI): truncated SVD
  - Independent Component Analysis (ICA)
  - Principal Component Analysis (PCA)
  - Canonical Correlation Analysis (CCA)
- Supervised
  - Linear Discriminant Analysis (LDA)

Principle Component Analysis

Principal Component Analysis

- Principal component analysis (PCA) reduces the dimensionality of a data set by finding **a new set of variables**, smaller than the original set of variables. It retains most of the sample's variance.
- It is useful for the compression and classification of data.
- The new variables, called **principal components (PCs)**, are uncorrelated, and are ordered by the fraction of the total information each retains.

Data distribution (inputs in regression/classification)

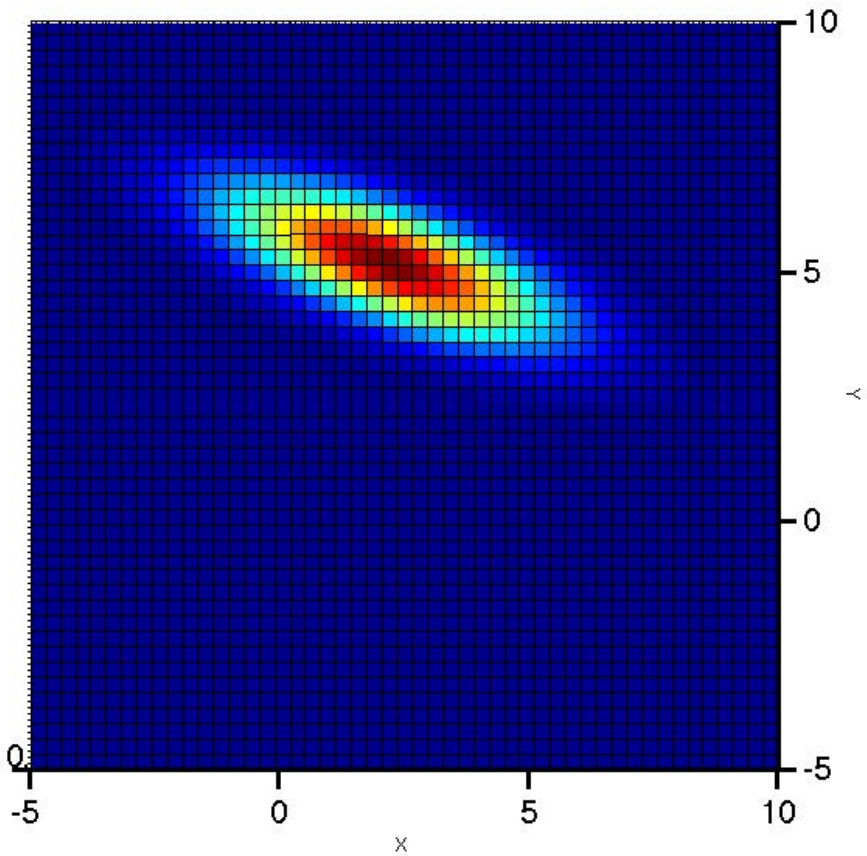


Figure: Gaussian PDF

Uncorrelated projections of principal variation Figure: Gaussian

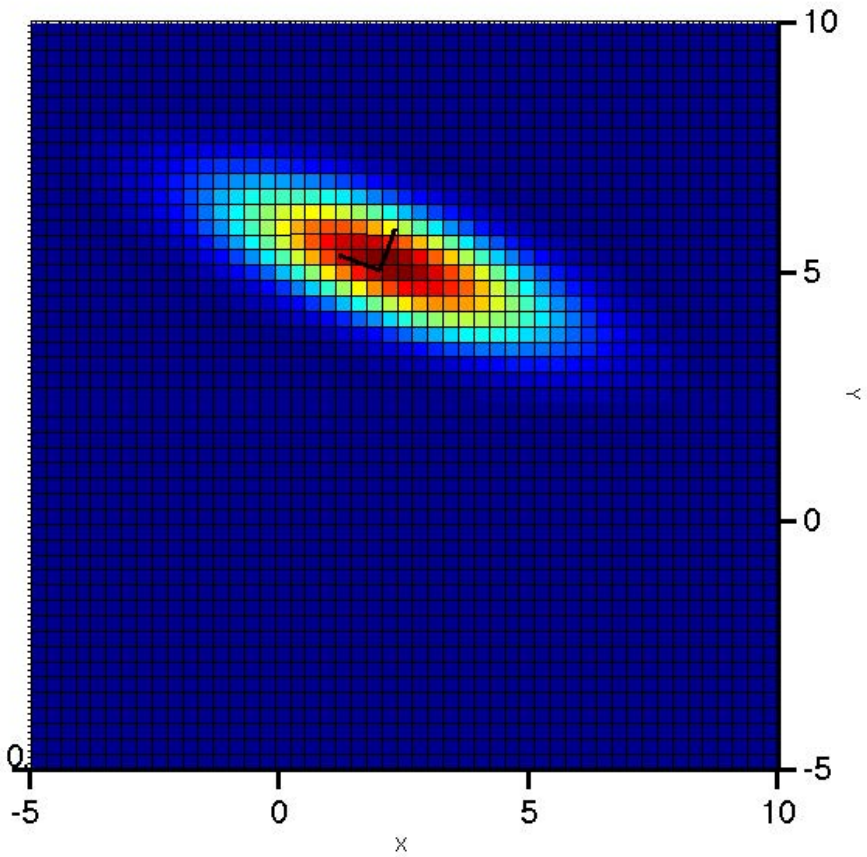


Figure: Gaussian PDF with PC eigenvectors

PCA Rotation

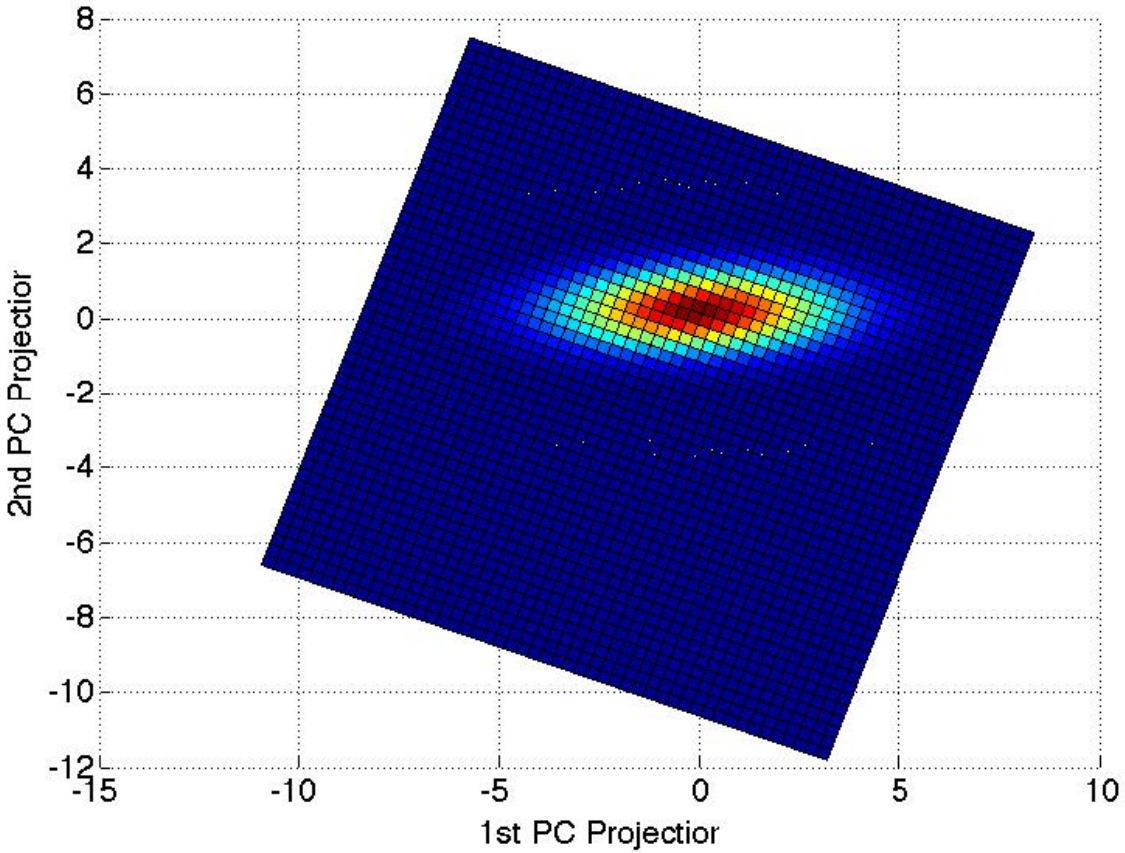


Figure: PCA Projected Gaussian PDF

PCA

Notation

- x is a vector of p random variables
- Weights/loading vector α_k is a vector of p constants (collectively $A = [\alpha_1, \ldots \alpha_d]$).
- $\alpha_k^T x = \sum_{j=1}^{p} \alpha_{kj} x_j$

Procedural description

- Find linear function of x, $\alpha_1^T x$ with maximum variance.
- Next find another linear function of x, $\alpha_2^T x$, uncorrelated with $\alpha_1^T x$ maximum variance.
- Iterate until we find α_d.

Goal

It is hoped, in general, that most of the variation in x will be accounted for by d PC's where $d \ll p$.

Derivation of PCA

Assumption and More Notation

- Σ is the known covariance matrix for the random variable x
- Foreshadowing: Σ will be replaced with S, the sample covariance matrix, when Σ is unknown.

Derivation of PCA

First Step

- Find $\alpha_k^T x$ that maximizes $\text{Var}(\alpha_k^T x) = \alpha_k^T S \alpha_k$

Derivation of PCA

First Step

- Find $\alpha_k^T x$ that maximizes $\text{Var}(\alpha_k^T x) = \alpha_k^T S \alpha_k$
- Without constraint we could pick a very big α_k

Derivation of PCA

First Step

- Find $\alpha_k^T x$ that maximizes $\text{Var}(\alpha_k^T x) = \alpha_k^T S \alpha_k$
- Without constraint we could pick a very big α_k
- Choose normalization constraint, namely $\alpha_k = 1$ (unit length vector).

Derivation of PCA

First Step

- Find $\alpha_k^T x$ that maximizes $\text{Var}(\alpha_k^T x) = \alpha_k^T S \alpha_k$
- Without constraint we could pick a very big α_k
- Choose normalization constraint, namely $\alpha_k = 1$ (unit length vector).

Constrained maximization - method of Lagrange multipliers

- To maximize $\alpha_k^T S \alpha_k$ subject to $\alpha_k^T \alpha_k = 1$ we use the technique of Lagrange multipliers. We maximize the function

$$\alpha_k^T S \alpha_k - \lambda(\alpha_k^T \alpha_k - 1)$$

w.r.t. to α_k by differentiating w.r.t. to α_k.

Derivation of PCA

First Step

- Find $\alpha_k^T x$ that maximizes $\text{Var}(\alpha_k^T x) = \alpha_k^T S \alpha_k$
- Without constraint we could pick a very big α_k
- Choose normalization constraint, namely $\alpha_k = 1$ (unit length vector).

Constrained maximization - method of Lagrange multipliers

- To maximize $\alpha_k^T S \alpha_k$ subject to $\alpha_k^T \alpha_k = 1$ we use the technique of Lagrange multipliers. We maximize the function

$$\alpha_k^T S \alpha_k - \lambda(\alpha_k^T \alpha_k - 1)$$

  w.r.t. to α_k by differentiating w.r.t. to α_k.
- Show that the solution relates to the eigen-decomposition of covariance matrix S.

Derivation of PCA

Constrained maximization - method of Lagrange multipliers

- This results in

$$\frac{d}{d\alpha_k}(\alpha_k^T S\alpha_k - \lambda_k(\alpha_k^T\alpha_k - 1)) = 0$$

$$S\alpha_k - \lambda_k\alpha_k = 0$$

$$S\alpha_k = \lambda_k\alpha_k$$

- This is an eigenvector equation where α_k is an eigenvector of S and λ_k is the associated eigenvalue.
- Which eigenvector should we choose?

Derivation of PCA

Constrained maximization - method of Lagrange multipliers

- If we recognize that the quantity to be maximized

$$\alpha_k^T S \alpha_k = \alpha_k^T \lambda_k \alpha_k = \lambda_k \alpha_k^T \alpha_k = \lambda_k$$

  then we should choose λ_k to be as big as possible. So, calling λ_1 the largest eigenvector of S and α_1 the corresponding eigenvector then the solution to

$$S\alpha_1 = \lambda_1 \alpha_1$$

  is the 1st principal component of x.
- In general α_k will be the kth PC of x and Var$(\alpha^T x) = \lambda_k$
- We will demonstrate this for $k = 2$, $k > 2$ is more involved but similar.

Derivation of PCA

Constrained maximization - method of Lagrange multipliers

- The second PC, $\alpha_2^T x$ maximizes $\alpha_2^T S \alpha_2$ subject to being uncorrelated with $\alpha_1^T x$

Derivation of PCA

Constrained maximization - method of Lagrange multipliers

- The second PC, $\alpha_2^T x$ maximizes $\alpha_2^T S \alpha_2$ subject to being uncorrelated with $\alpha_1^T x$
- The un-correlate constraint can be expressed using any of these equations

$$\begin{aligned}\text{Cov}(\alpha_1^T x, \alpha_2^T x) &= \alpha_1^T S \alpha_2 = \alpha_2^T S \alpha_1 = \alpha_2^T \lambda_1 \alpha_1 \\ &= \lambda_1 \alpha_2^T \alpha_1 = \lambda_1 \alpha_1^T \alpha_2 = 0\end{aligned}$$

Derivation of PCA

Constrained maximization - method of Lagrange multipliers

- The second PC, $\alpha_2^T x$ maximizes $\alpha_2^T S \alpha_2$ subject to being uncorrelated with $\alpha_1^T x$
- The un-correlate constraint can be expressed using any of these equations

$$\begin{aligned}\text{Cov}(\alpha_1^T x, \alpha_2^T x) &= \alpha_1^T S \alpha_2 = \alpha_2^T S \alpha_1 = \alpha_2^T \lambda_1 \alpha_1 \\ &= \lambda_1 \alpha_2^T \alpha_1 = \lambda_1 \alpha_1^T \alpha_2 = 0\end{aligned}$$

- Of these, if we choose the second last we can write an Langrangian to maximize α_2

$$\alpha_2^T S \alpha_2 - \lambda_2(\alpha_2^T \alpha_2 - 1) - \phi \alpha_2^T \alpha_1$$

Derivation of PCA

Constrained maximization - method of Lagrange multipliers

- Differentiation of this quantity w.r.t. α_2 (and setting the result equal to zero) yields

$$\frac{d}{d\alpha_2}(\alpha_2^T S\alpha_2 - \lambda_2(\alpha_2^T\alpha_2 - 1) - \phi\alpha_2^T\alpha_1) = 0$$

$$S\alpha_2 - \lambda_2\alpha_2 - \phi\alpha_1 = 0$$

Derivation of PCA

Constrained maximization - method of Lagrange multipliers

- Differentiation of this quantity w.r.t. α_2 (and setting the result equal to zero) yields

$$\frac{d}{d\alpha_2}(\alpha_2^T S\alpha_2 - \lambda_2(\alpha_2^T\alpha_2 - 1) - \phi\alpha_2^T\alpha_1) = 0$$

$$S\alpha_2 - \lambda_2\alpha_2 - \phi\alpha_1 = 0$$

- If we left multiply α_1 into this expression

$$\alpha_1^T S\alpha_2 - \lambda_2\alpha_1^T\alpha_2 - \phi\alpha_1^T\alpha_1 = 0$$

$$0 - 0 - \phi 1 = 0$$

then we can see that ϕ must be zero and that when this is true that we are left with:

$$S\alpha_2 - \lambda_2\alpha_2 = 0$$

Derivation of PCA

- Clearly

$$S\alpha_2 - \lambda_2\alpha_2 = 0$$

  is another eigenvalue equation and the same strategy of choosing α_2 to be the eigenvector associated with the second largest eigenvalue yields the second PC of x, namely $\alpha_2^T x$.
- This process can be repeated for $k = 1 \dots p$ yielding up to p different eigenvectors of S along with the corresponding eigenvalues $\lambda_1, \dots \lambda_p$.
- Furthermore, the variance of each of the PC's are given by

$$\text{Var}[\alpha_k^T x] = \lambda_k, \quad k = 1, 2, \dots, p$$

PCA using the sample covariance matrix

- If we recall that the sample covariance matrix (an unbiased estimator for the covariance matrix of x) is given by

$$S = \frac{1}{n-1} X^T X$$

where X is a $(n \times p)$ matrix with (i,j)th element $(x_{ij} - \bar{x}_j)$ (in other words, X is a zero mean design matrix).

PCA using the sample covariance matrix

- If we recall that the sample covariance matrix (an unbiased estimator for the covariance matrix of x) is given by

$$S = \frac{1}{n-1} X^T X$$

where X is a $(n \times p)$ matrix with (i,j)th element $(x_{ij} - \bar{x}_j)$ (in other words, X is a zero mean design matrix).

- In many places we derive PCA using the following biased covariance estimator:

$$\tilde{S} = \frac{1}{n} X^T X$$

Does it make PCA different?

PCA using the sample covariance matrix

- We construct the matrix A by combining the p eigenvectors of S (or eigenvectors of $X^T X$ - they're the same) then we can define a matrix of PC scores

$$Z = XA$$

- Of course, if we instead form Z by selecting the d eigenvectors corresponding to the d largest eigenvalues of S when forming A then we can achieve an "optimal" (in some senses) d-dimensional projection of x.

Optimality property of PCA

Main theoretical result:

The matrix A consisting of the first d eigenvectors of the covariance matrix S solves the following min problem:

$$\min_{A \in \mathbb{R}^{p \times d}} \|X - (XA)A^T\|_F^2 \quad \text{subject to:} \quad A^T A = I_d$$

PCA projection minimizes the reconstruction error among all linear projections of size d.

Optimality property of PCA

Main theoretical result:

The matrix A consisting of the first d eigenvectors of the covariance matrix S solves the following min problem:

$$\min_{A \in \mathbb{R}^{p \times d}} \|X - (XA)A^T\|_F^2 \quad \text{subject to:} \quad A^T A = I_d$$

PCA projection minimizes the reconstruction error among all linear projections of size d.

Show that the minimization problem is equivalent to the eigen problem/PCA.

Optimality property of PCA

Main theoretical result:

The matrix A consisting of the first d eigenvectors of the covariance matrix S solves the following min problem:

$$\min_{A \in \mathbb{R}^{p \times d}} \|X - (XA)A^T\|_F^2 \quad \text{subject to:} \quad A^T A = I_d$$

PCA projection minimizes the reconstruction error among all linear projections of size d.

Show that the minimization problem is equivalent to the eigen problem/PCA.

Computing the PCA

- Given the sample covariance matrix

$$S = \frac{1}{n-1} X^T X$$

  the most straightforward way of computing the PCA loading matrix is to utilize the eigen-decomposition/SVD of $S = A^T \tilde{\Sigma} A$ where A is a matrix consisting of the eigenvectors of S, $\tilde{\Sigma}$ here is a diagonal matrix whose diagonal elements are the eigenvalues corresponding to each eigenvector.
- Creating a reduced dimensionality projection of X is accomplished by selecting the d largest eigenvalues in Λ and retaining the d corresponding eigenvectors from A.

Computing the PCA via Singular Value Decomposition

It is natural to first form the covariance matrix of the centered data matrix X for computing the PCs. However, this is not a good idea. Why?

Computing the PCA via Singular Value Decomposition

It is natural to first form the covariance matrix of the centered data matrix X for computing the PCs. However, this is not a good idea. Why?

- The condition number of $\frac{1}{n-1}X^TX$ can be much larger than that of X.
- For a sparse X, $S = \frac{1}{n-1}X^TX$ may not be sparse any more.

Computing the PCA via Singular Value Decomposition

It is natural to first form the covariance matrix of the centered data matrix X for computing the PCs. However, this is not a good idea. Why?

- The condition number of $\frac{1}{n-1}X^TX$ can be much larger than that of X.
- For a sparse X, $S = \frac{1}{n-1}X^TX$ may not be sparse any more.

How to compute the PCA (effectively):

- Center the data by subtracting the mean of the rows.
- Compute the SVD of the centered data matrix X as $X = U\tilde{\Sigma}V^T$.
- The principal components are the columns of V ; the coordinates of the data in the basis defined by the principal components are $U\tilde{\Sigma}$.

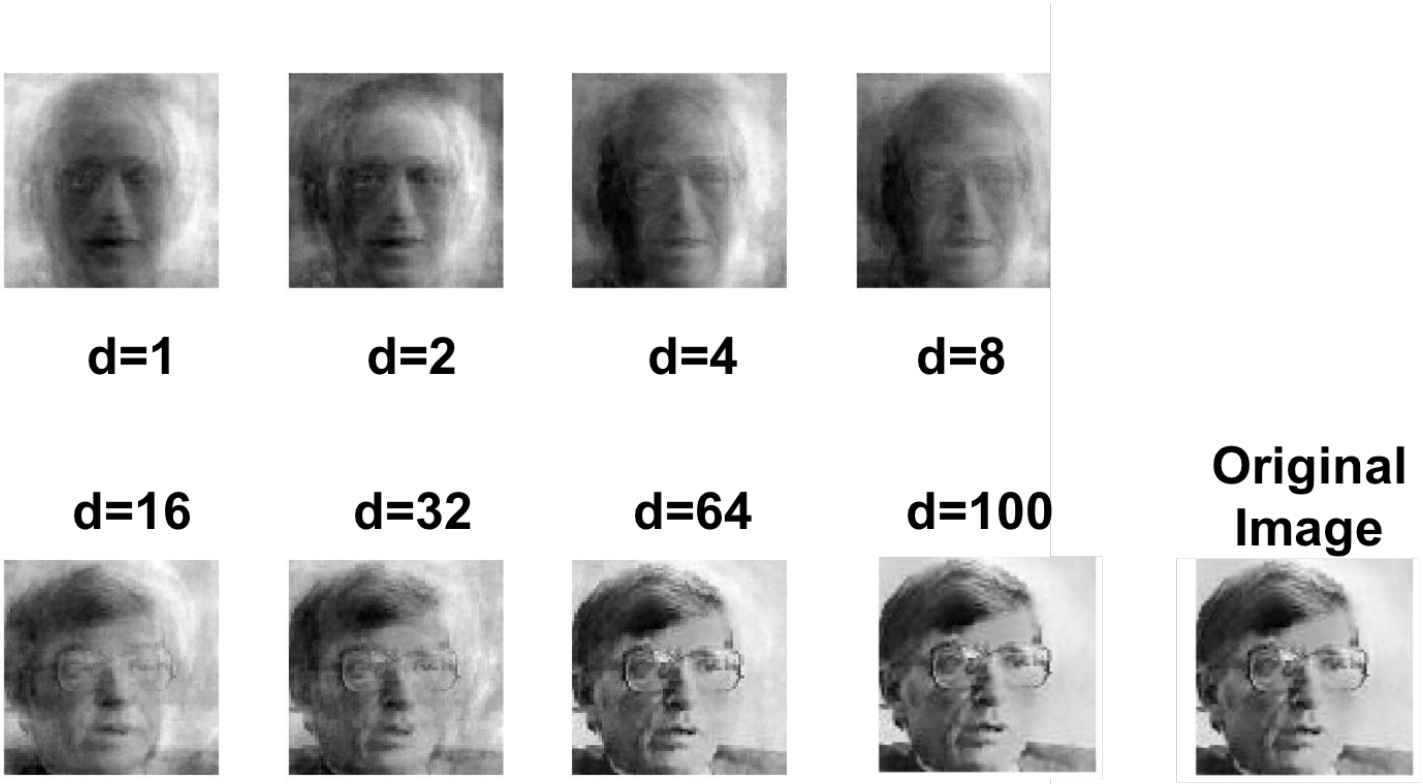
How to choose the number of PCs

- The variance in the direction of the k-th principal component is given by the corresponding singular value: σ_k^2.
- Singular values can be used to estimate how many principal components to keep.
- Rule of thumb: keep enough to explain 85% of the variation:

$$\frac{\sum_{j=1}^{k} \sigma_j^2}{\sum_{j=1}^{n} \sigma_j^2} \approx 0.85$$

- Typically, the squared singular values drop rapidly. Thus, the first few principal components are enough to capture most of the variation in the data. This leads to data compression.

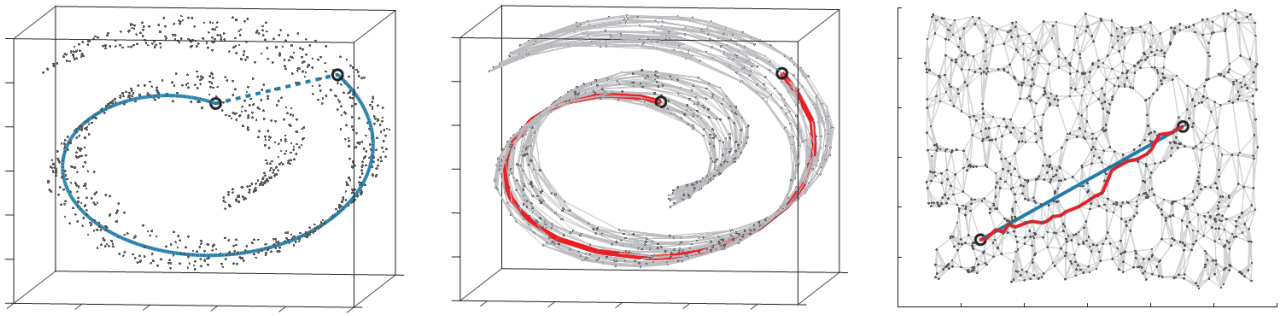
Application on Image Compression



Problems with PCA

PCA is not without its problems and limitations

- PCA assumes approximate normality of the input space distribution
  - However, PCA may still be able to produce a "good" low dimensional projection of the data even if the data isn't normally distributed
- PCA may "fail" if the data lies on a "complicated" manifold



- PCA assumes that the input data is real and continuous.
- Extensions to consider
  - Collins, et. al. "A generalization of principal components analysis to the exponential family." NIPS 2001.
  - Hyvärinen and Erkki "Independent component analysis: algorithms and applications." Neural networks 13.4 (2000): 411-430.
  - ISOMAP (Science 2000), local linear embedding (LLE, Science 2000), etc.