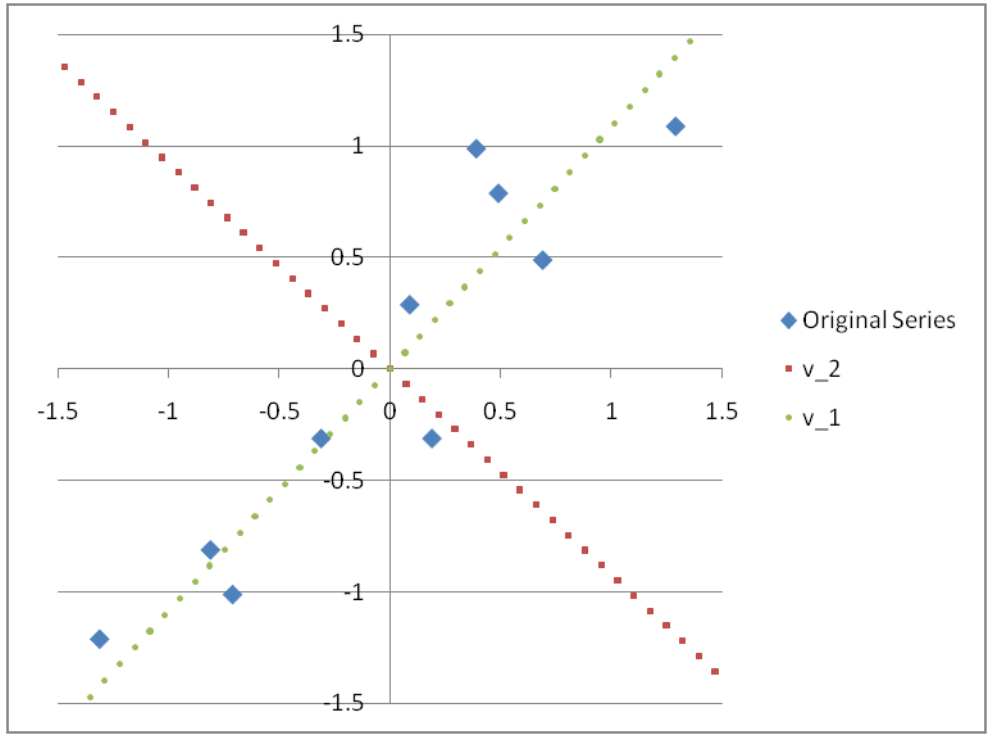
GTDM – 2019/20

- Eigenvalues/Eigenvectors and applications

Based on Linear Algebra and Its Applications, David C. Lay, Steven R. Lay, and Judi J. McDonald, PEARSON 5^th ed.

Principal Component Analysis (PCA)

x	y
2.5	2.4
0.5	0.7
2.2	2.9
1.9	2.2
3.1	3.0
2.3	2.7
2.0	1.6
1.0	1.1
1.5	1.6
1.1	0.9

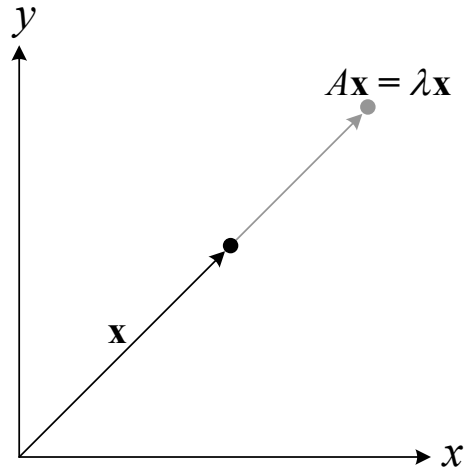


- PCA projects the data along the directions where the data varies most.
- These directions are determined by the eigenvectors of the covariance matrix corresponding to the largest eigenvalues.
- The magnitude of the eigenvalues corresponds to the variance of the data along the eigenvector directions.
- Find the projection that best preserves the variance.
- PCA preserves as much information as possible by minimizing the “reconstruction” error:

Eigenvalues and Eigenvectors

Almost all vectors change direction, when they are multiplied by A. Certain exceptional vectors **x** are in the same direction as A**x**. Those are the "**eigenvectors**". Multiply an eigenvector by A, and the vector A**x** is a number λ times the original **x**.

(The term eigenvalue is from the German word *Eigenwert*, meaning "proper value")



Geometric Interpretation

- Eigenvalue and Eigenvector:

A: an $n \times n$ matrix
λ: a scalar (could be **zero**)
$\mathbf{x}$: a **nonzero** vector in R^n

- Eigenvalue problem (one of the most important problems in the linear algebra):

If A is an $n \times n$ matrix, do there exist nonzero vectors $\mathbf{x}$ in R^n such that $A\mathbf{x}$ is a scalar multiple of $\mathbf{x}$?

Eigenvalue

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

Eigenvector

Introduction to Eigenvalues and Eigenvectors

The **eigenvalue** tells whether the special vector **x** is stretched or shrunk or reversed or left unchanged—when it is multiplied by A. The eigenvalue could be zero! Then A**x** = 0 **x** means that this eigenvector **x** is in the nullspace (the space of the vectors such that A**x**=**0**). If A is the identity matrix, every vector has A**x** = **x**. All vectors are eigenvectors of I.

All eigenvalues "lambda" are =1. This is unusual to say the least. Most 2 by 2 matrices have *two* eigenvector directions and *two* eigenvalues. We will show that det (A –λ I) = 0:

A**x** = λ **x** for **x**≠ **0** and a scalar λ, so (A- λI)**x** = **0** and we have a non trivial solution if and only if det (A –λ I) ≠ 0

Example Let $A = \begin{bmatrix} .8 & .3 \\ .2 & .7 \end{bmatrix}$, $\det \begin{bmatrix} .8-\lambda & .3 \\ .2 & .7-\lambda \end{bmatrix} = \lambda^2 - \frac{3}{2}\lambda + \frac{1}{2} = (\lambda - 1)\left(\lambda - \frac{1}{2}\right) = 0$ then λ=1, and λ=1/2.

For those numbers, the matrix (A - λI) becomes *singular* (zero determinant). The eigenvectors **V**1 and **V**2 are in the nullspaces of (A - I) and (A -1/2 I).

(A - I) **V**1 = **0** ⇒ $\mathbf{V_1} = t \begin{bmatrix} 1 \\ 2/3 \end{bmatrix}$ for any real value t

(A – ½ I)**V**2 =**0** ⇒ $\mathbf{V_2} = t \begin{bmatrix} 1 \\ -1 \end{bmatrix}$ for any real value t

Summary To solve the eigenvalue problem for an n by n matrix, follow these steps:

1. ***Compute the determinant of*** $A - \lambda I$. With λ subtracted along the diagonal, this determinant starts with λ^n or $-\lambda^n$. It is a polynomial in λ of degree n.
2. ***Find the roots of this polynomial,*** by solving $\det(A - \lambda I) = 0$. The n roots are the n eigenvalues of A. They make $A - \lambda I$ singular.
3. For each eigenvalue λ, ***solve*** $(A - \lambda I)x = 0$ ***to find an eigenvector*** x.

Theorem (The eigenspace of A corresponding to λ)
If A is an $n \times n$ matrix with an eigenvalue λ, then the set of all eigenvectors of λ together with the zero vector is a subspace of R^n. This subspace is called the eigenspace of λ .

Remark Are eigenvectors are unique? If **x** is an eigenvector, then β**x** is also an eigenvector and $\beta\lambda$ is an eigenvalue

$$\mathbf{A}(\beta\mathbf{x}) = \beta(\mathbf{A}\mathbf{x}) = \beta(\lambda\mathbf{x}) = \lambda(\beta\mathbf{x})$$

Remark The eigenvalues (roots of a polynomial) could be complex numbers!

Calculating the Eigenvectors/values

- Expand the det(**A** - λ**I**) = 0 for a 2 **x** 2 matrix

$$\det(A-\lambda I)=\det\left(\begin{bmatrix}a_{11} & a_{12}\\ a_{21} & a_{22}\end{bmatrix}-\lambda\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}\right)=0$$

$$\det\begin{bmatrix}a_{11}-\lambda & a_{12}\\ a_{21} & a_{22}-\lambda\end{bmatrix}=0\Rightarrow(a_{11}-\lambda)(a_{22}-\lambda)-a_{12}a_{21}=0$$

$$\lambda^2-\lambda(a_{11}+a_{22})+(a_{11}a_{22}-a_{12}a_{21})=0$$

- For a 2 **x**2 matrix, this is a simple quadratic equation with two solutions (maybe complex)

$$\lambda=(a_{11}+a_{22})\pm\sqrt{\frac{(a_{11}+a_{22})^2}{4(a_{11}a_{22}-a_{12}a_{21})}}$$

- This "characteristic equation" can be used to solve for **x**

Eigenvalue example

- Consider,

$$\mathbf{A}=\begin{bmatrix}1 & 2\\ 2 & 4\end{bmatrix}\Rightarrow\begin{cases}\lambda^2-(a_{11}+a_{22})\lambda+(a_{11}a_{22}-a_{12}a_{21})=0\\ \lambda^2-(1+4)\lambda+(1\cdot 4-2\cdot 2)=0\\ \lambda^2=(1+4)\lambda\Rightarrow\lambda=0,\lambda=5\end{cases}$$

- The corresponding eigenvectors can be computed as

$$\lambda=0\Rightarrow\left[\begin{bmatrix}1 & 2\\ 2 & 4\end{bmatrix}-\begin{bmatrix}0 & 0\\ 0 & 0\end{bmatrix}\right]\cdot\begin{bmatrix}x\\ y\end{bmatrix}=0\Rightarrow\begin{bmatrix}1 & 2\\ 2 & 4\end{bmatrix}\cdot\begin{bmatrix}x\\ y\end{bmatrix}=\begin{bmatrix}1x+2y\\ 2x+4y\end{bmatrix}=\begin{bmatrix}0\\ 0\end{bmatrix}$$

$$\lambda=5\Rightarrow\left[\begin{bmatrix}1 & 2\\ 2 & 4\end{bmatrix}-\begin{bmatrix}5 & 0\\ 0 & 5\end{bmatrix}\right]\cdot\begin{bmatrix}x\\ y\end{bmatrix}=0\Rightarrow\begin{bmatrix}-4 & 2\\ 2 & -1\end{bmatrix}\cdot\begin{bmatrix}x\\ y\end{bmatrix}=\begin{bmatrix}-4x+2y\\ 2x-1y\end{bmatrix}=\begin{bmatrix}0\\ 0\end{bmatrix}$$

  - For $\lambda = 0$, one possible solution is $\mathbf{x} = (2, -1)$
  - For $\lambda = 5$, one possible solution is $\mathbf{x} = (1, 2)$

Warning: we compute eigenvalues using the determinant only for very low dimensional case as an exercise, in the applications we must to consider efficient numerical methods and we (usually) have to calculate only a few eigenvalues / eigenvectors.

Theorem

If $\mathbf{v}_1, \ldots, \mathbf{v}_r$ are eigenvectors that correspond to distinct eigenvalues $\lambda_1, \ldots, \lambda_r$ of an $n \times n$ matrix A, then the set $\{\mathbf{v}_1, \ldots, \mathbf{v}_r\}$ is linearly independent.

Let A be a **square n x n** matrix with ***n* linearly independent eigenvectors** (a "non-defective" matrix)

Let P have the eigenvectors as columns:

$$P = [V_1 \; V_2 \; \ldots \; V_n]$$

Then, AP can be written

$$AP = A[\mathbf{V}_1 \; \mathbf{V}_2 \; \ldots \; \mathbf{V}_n] = [\lambda_1\mathbf{V}_1 \; \lambda_2\mathbf{V}_2 \; \ldots \; \lambda_n\mathbf{V}_n] = [\mathbf{V}_1 \; \mathbf{V}_2 \; \ldots \; \mathbf{V}_n] \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \ldots & 0 \\ 0 & 0 & \lambda_n \end{bmatrix} =$$

$$[\mathbf{V}_1 \; \mathbf{V}_2 \; \ldots \; \mathbf{V}_n] \, \Lambda$$

Thus $AP = P\Lambda$, or $\mathbf{P^{-1}AP} = \Lambda$, or $\mathbf{A = P\Lambda P^{-1}}$ with Λ diagonal matrix with eigenvalues.

A is called **diagonalizable**

Example. Let $A = \begin{bmatrix} -1 & -2 \\ 6 & 6 \end{bmatrix}$, then $|A - \lambda I| = \begin{vmatrix} -1-\lambda & -2 \\ 6 & 6-\lambda \end{vmatrix}$ = (-1-λ) (6-λ) +12 =0 ⇒

λ_1=3, λ_2=2.

Eigenvector $\mathbf{V}_1$ for λ_1=3, $\begin{bmatrix} -4 & -2 \\ 6 & 3 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}=\begin{bmatrix} 0 \\ 0 \end{bmatrix}$, so $\mathbf{V}_1 = x_2 \begin{bmatrix} -1/2 \\ 1 \end{bmatrix}$ with x_2 real parameter

Eigenvector $\mathbf{V}_2$ for λ_2=2, $\begin{bmatrix} -3 & -2 \\ 6 & 4 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}=\begin{bmatrix} 0 \\ 0 \end{bmatrix}$, so $\mathbf{V}_2 = x_2 \begin{bmatrix} -2/3 \\ 1 \end{bmatrix}$ with x_2 real parameter

The vectors $\mathbf{V}_1$ and $\mathbf{V}_2$ are linearly independent (for scalars c_1 and c_2 such that $c_1 \mathbf{V}_1 + c_2 \mathbf{V}_2 = \mathbf{0}$ implies c_1 =0 c_2=0). We fix an eigenvector for each eigenvalues, for example

$P = [\mathbf{V}_1 \quad \mathbf{V}_2] = \begin{bmatrix} -1 & -2 \\ 2 & 3 \end{bmatrix}$ and $P^{-1} = \begin{bmatrix} 3 & 2 \\ -2 & -1 \end{bmatrix}$ and

$P^{-1}A\,P = \begin{bmatrix} 3 & 2 \\ -2 & -1 \end{bmatrix}\begin{bmatrix} -1 & -2 \\ 6 & 6 \end{bmatrix}\begin{bmatrix} -1 & -2 \\ 2 & 3 \end{bmatrix}=\begin{bmatrix} 3 & 0 \\ 0 & 2 \end{bmatrix}$

Example (multiple eigenvalues) Eigenvalue problems and diagonalization

$$A = \begin{bmatrix} 1 & 3 & 0 \\ 3 & 1 & 0 \\ 0 & 0 & -2 \end{bmatrix}$$

Characteristic equation:

$$|A - \lambda I| = \begin{vmatrix} 1-\lambda & 3 & 0 \\ 3 & 1-\lambda & 0 \\ 0 & 0 & -2-\lambda \end{vmatrix} = (-2-\lambda)\left((1-\lambda)^2 - 9\right) = 0$$

The eigenvalues : $\lambda_1 = 4,\ \lambda_2 = -2,\ \lambda_3 = -2$

(1) $\lambda = 4 \Rightarrow$ the eigenvector $\mathbf{p}_1 = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}$ (**one possible eigenvector**)

(2) $\lambda = -2 \Rightarrow$ the eigenvector $\mathbf{p}_2 = \begin{bmatrix} 1 \\ -1 \\ 0 \end{bmatrix}$, $\mathbf{p}_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$

(two free variables in the corresponding Homogeneous linear system)

$$P = [\mathbf{p}_1 \quad \mathbf{p}_2 \quad \mathbf{p}_3] = \begin{bmatrix} 1 & 1 & 0 \\ 1 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \text{ and } P^{-1}AP = \begin{bmatrix} 4 & 0 & 0 \\ 0 & -2 & 0 \\ 0 & 0 & -2 \end{bmatrix}$$

($\mathbf{p}_1$ $\mathbf{p}_2$ $\mathbf{p}_3$ are linear independent)

NOTE If $P = [\mathbf{p}_2 \quad \mathbf{p}_1 \quad \mathbf{p}_3]$

$$= \begin{bmatrix} 1 & 1 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \Rightarrow P^{-1}AP = \begin{bmatrix} -2 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & -2 \end{bmatrix}$$

Compute the power of A: a simple case.

Let A a diagonalizable matrix: $A=P\Lambda P^{-1}$ for a suitable P. Then

$$A^k = (P\,\Lambda P^{-1})(P\,\Lambda\,P^{-1})v\ldots(P\Lambda\,P^{-1}) = P\,\Lambda^k\,P^{-1}$$

And the long range behavior is determined by the power of the eigenvaluesv(P and P^{-1} does not change)

Example Let initial population $\mathbf{x}_0 = \begin{bmatrix}8000\\2000\end{bmatrix}$ and a transition matrix $A = \begin{bmatrix}0.7 & 0.2\\0.3 & 0.8\end{bmatrix}$

$$\mathbf{x}_{k+1} = A\,\mathbf{x}_k \quad k=0,1,2,\ldots$$

Eigenvalues of A: $\lambda_1 = 1$ (easy from the entries of the matrix A… why?) $\lambda_2 = 1/2$ For the eigenvectors we choose:

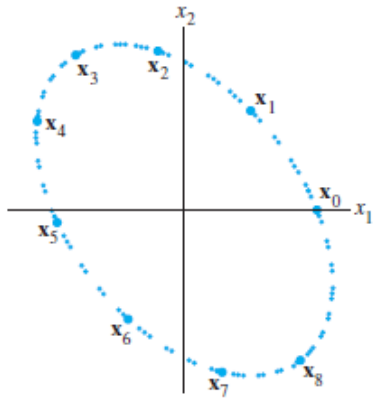
$\mathbf{V}_1 = \begin{bmatrix}2\\3\end{bmatrix}$ and $\mathbf{V}_2 = \begin{bmatrix}-1\\1\end{bmatrix}$ and $\mathbf{x}_k = A^k\,\mathbf{x}_0 = P\,\Lambda^k\,P^{-1}\,\mathbf{x}_0$ where $P=[\mathbf{V}_1\ \mathbf{V}_2\ ]$ then (after some computation)

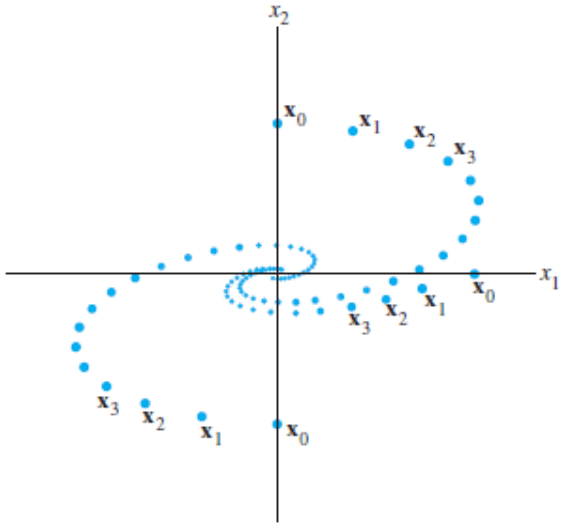
$\mathbf{x}_k = 2000\,\mathbf{V}_1 - 4000\left(\frac{1}{2}\right)^k\mathbf{V}_2$ when k>>1 $\mathbf{x}_k \sim 2000\,\mathbf{V}_1$

Remark (Complex eigenvalues). When a real 2x2 matrix A has complex eigenvalues, A is not diagonalizable, but the dynamical system $\mathbf{x}_{k+1} = A\mathbf{x}_k$ is easy to describe. If the modulus of the eigenvalues is 1, the iterates of a point $\mathbf{x}_0$ spiraled around the origin along an elliptical trajectory.

If A has two complex eigenvalues whose absolute value is greater than 1, then **0** is a repeller and iterates of $\mathbf{x}_0$ will spiral outward around the origin.

If the absolute values of the complex eigenvalues are less than 1, then the origin is an attractor and the iterates of $\mathbf{x}_0$ spiral inward toward the origin.





Example. The spotted owl population in the Willow Creek area of California was modeled by a dynamical system $\mathbf{x}_{k+1} = A\mathbf{x}_k$ with

$$A = \begin{bmatrix} 0 & 0 & .33 \\ .18 & 0 & 0 \\ 0 & .71 & .94 \end{bmatrix}$$

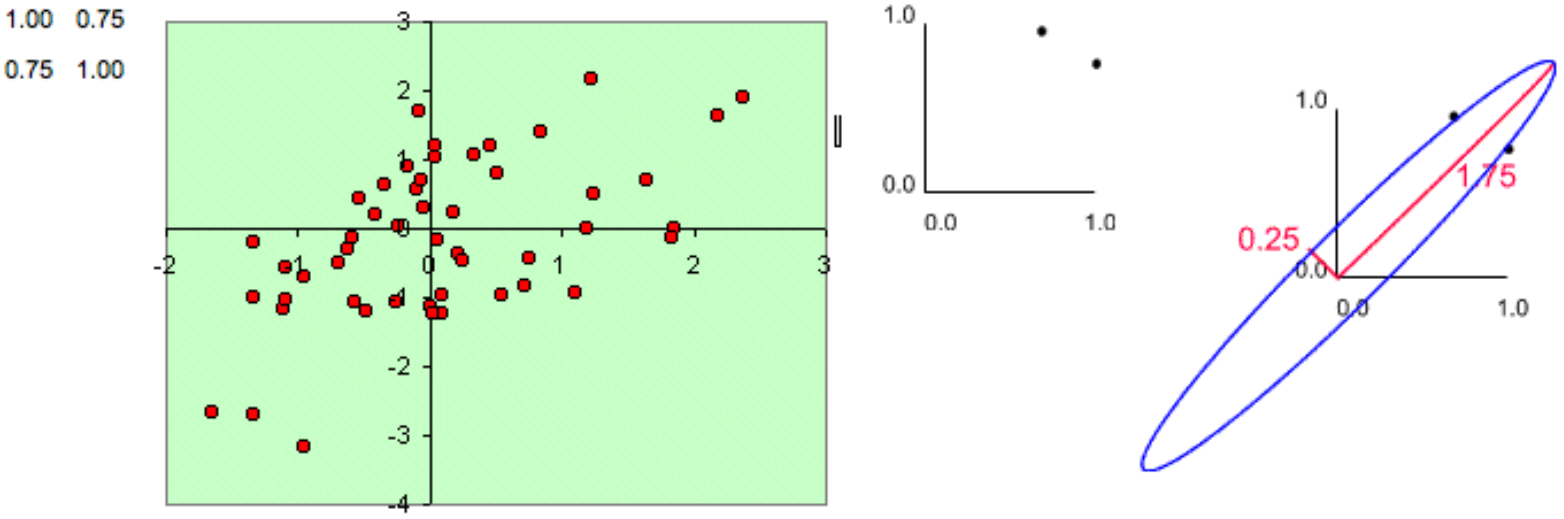
and $\lambda_1 = .98, \quad \lambda_2 = -.02 + .21i, \text{ and } \lambda_3 = -.02 - .21i.$ with modulus <1, then from

$$\mathbf{x}_k = c_1(\lambda_1)^k\mathbf{v}_1 + c_2(\lambda_2)^k\mathbf{v}_2 + c_3(\lambda_3)^k\mathbf{v}_3$$ the real sequence **x**k approaches the zero vector: this model predicts that the spotted owls will eventually all perish.

Principal component analysis

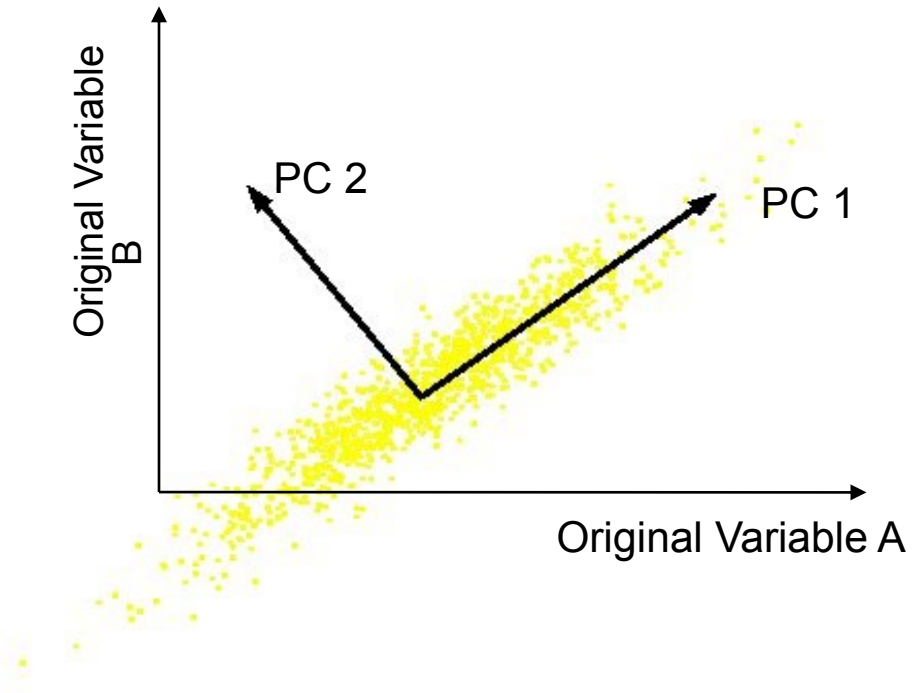
- Consider a covariance matrix, **A**, for some S (two variable, n subjects)

$$\mathbf{A} = \begin{bmatrix} 1 & .75 \\ .75 & 1 \end{bmatrix} \Rightarrow \lambda_1 = 1.75, \lambda_2 = 0.25$$



- Error ellipse with the major axis as the larger eigenvalue and the minor axis as the smaller eigenvalue

- First principal component is the direction of greatest variability (covariance) in the data
- Second is the next orthogonal (uncorrelated) direction of greatest variability
  - So first remove all the variability along the first component, and then find the next direction of greatest variability
- And so on …
- Thus each eigenvectors provides the directions of data variances in decreasing order of eigenvalues



Principal Component Analysis

Principal component analysis

- **It is a way of identifying the underlying patterns in data**
- **It can extract information in a large data set with many variables and approximate this data set with fewer factors**
- **In other words, it can reduce the number of variables to a more manageable set**

Steps of the principal component analysis

- **Step 1: Get some data**
- **Step 2: Subtract the mean**
- **Step 3: Calculate the covariance matrix**
- **Step 4: Calculate the eigenvectors and eigenvalues of the covariance matrix**
- **Step 5: Deriving the transformed data set**
- **Step 6: Getting the original data back**

Step 1:

x	y
2.5	2.4
0.5	0.7
2.2	2.9
1.9	2.2
3.1	3.0
2.3	2.7
2.0	1.6
1.0	1.1
1.5	1.6
1.1	0.9

Step 2:

x	y
0.69	0.49
-1.31	-1.21
0.39	0.99
0.09	0.29
1.29	1.09
0.49	0.79
0.19	-0.31
-0.81	-0.81
-0.31	-0.31
-0.71	-1.01

$$\equiv X^T = \begin{bmatrix} x & y \end{bmatrix}$$

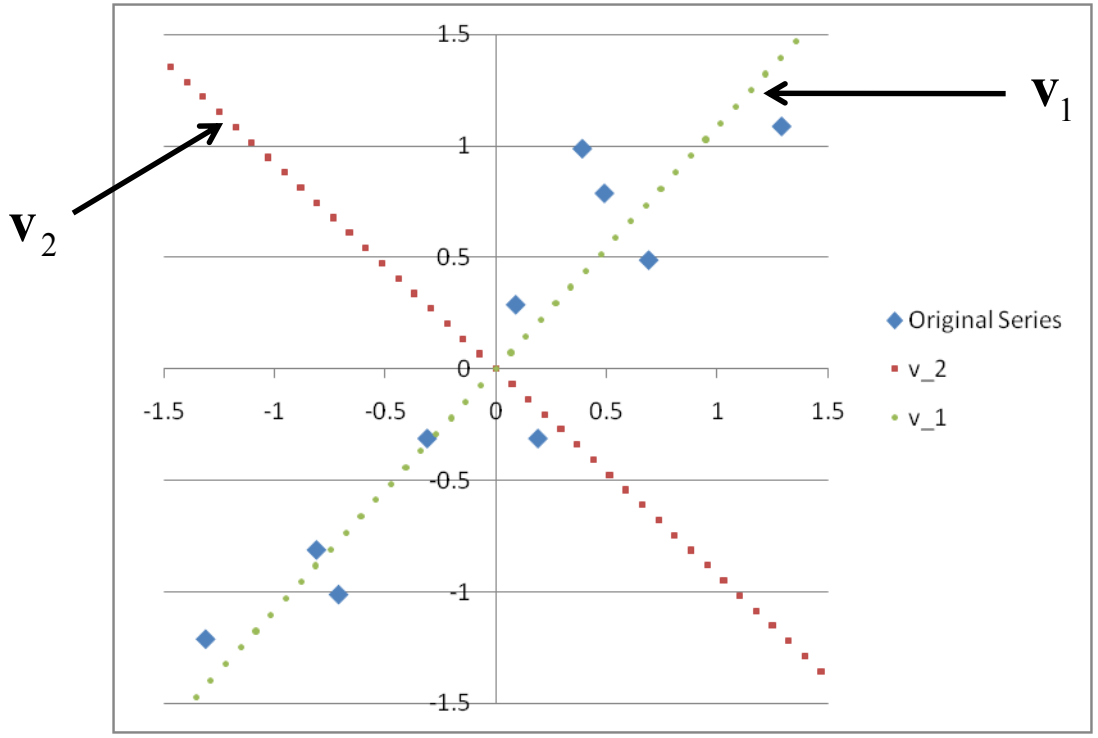
Step 3:

$$\text{var}(X^T) = E\left[XX^T\right] = E\left[\begin{bmatrix} x^T \\ y^T \end{bmatrix} \begin{bmatrix} x & y \end{bmatrix}\right] = E\begin{bmatrix} x^T x & x^T y \\ y^T x & y^T y \end{bmatrix}$$

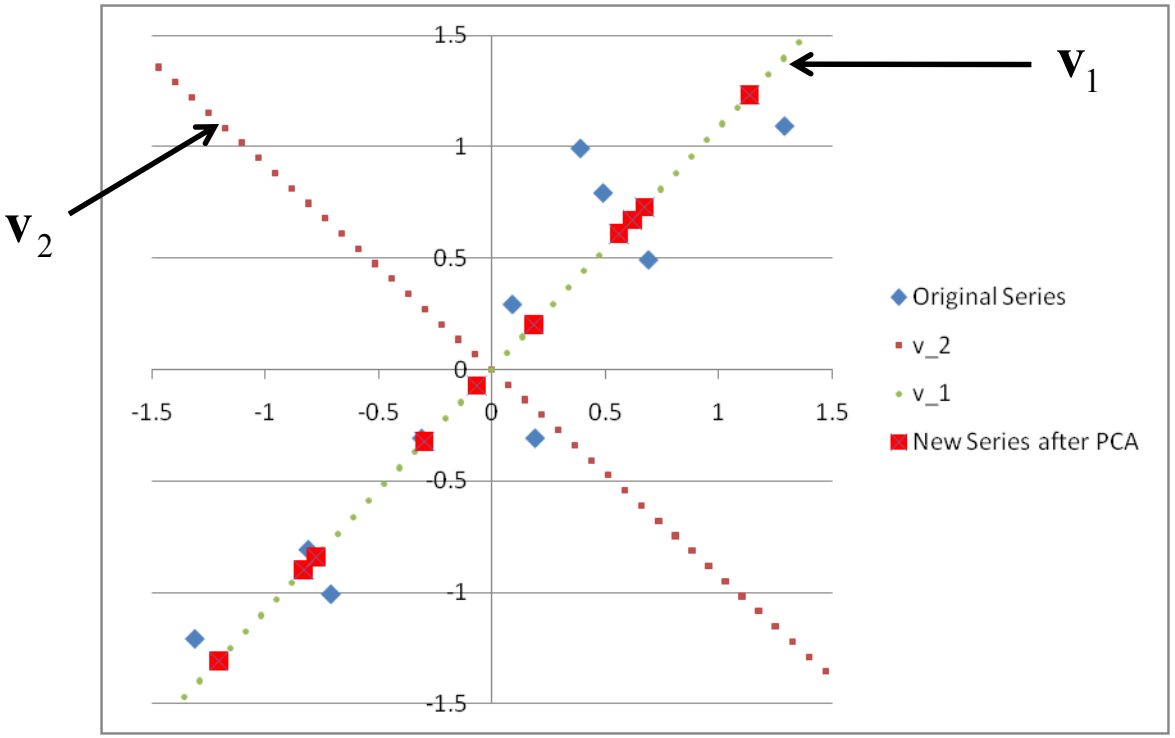
$$= \begin{bmatrix} \text{var}(x) & \text{cov}(x,y) \\ \text{cov}(x,y) & \text{var}(y) \end{bmatrix} = \begin{pmatrix} 0.616556 & 0.615444 \\ 0.615444 & 0.716556 \end{pmatrix} \equiv A$$

- **Step 4: Calculate the eigenvectors and eigenvalues of the covariance matrix *A***

$$\lambda_1 = 1.284028,\ \mathbf{v}_1 = \begin{pmatrix} -0.67787 \\ -0.73518 \end{pmatrix} \qquad \lambda_2 = 0.049083,\ \mathbf{v}_2 = \begin{pmatrix} -0.73518 \\ 0.67787 \end{pmatrix}$$



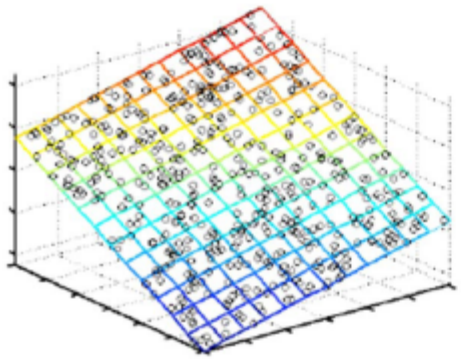
1. The two eigenvectors are orthogonal to each other-
2. $\mathbf{v}_1$ eigenvector (corresponding to the largest eigenvalue λ_1) is just like a best-fit regression line
3. $\mathbf{v}_2$ seems less important to explain the data since the projection of each node on the $\mathbf{v}_2$ axis is very close to zero
4. The interpretation of $\mathbf{v}_1$ is the new axis which retains as much as possible the variance information that was contained in the original two dimensions



- If only the principal component is considered in the Principal Component Analysis (PCA), it is equivalent to project all points onto the $\mathbf{v}_1$ vector
- It can be observed in the above figure that the projection onto $\mathbf{v}_1$ vector can retains as much as possible the "interpoint" distance information (variance) that was contained in the original series of (*x*, *y*)

The same for the n-dimensional case

- Consider data without class labels
- Try to find a more compact representation of the data



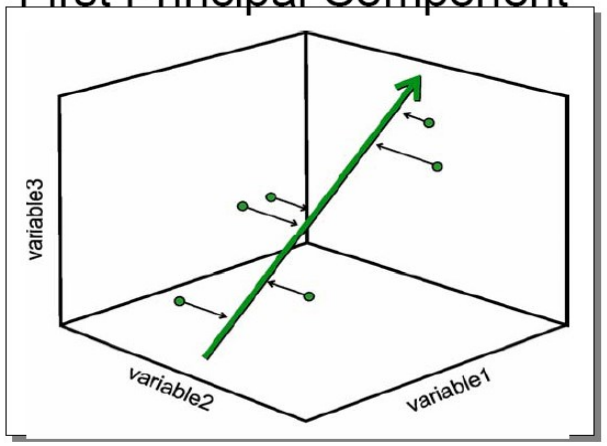
$3d \Rightarrow 2d$

- Assume that the high dimensional data actually resides in a inherent low-dimensional space
- Additional dimensions are just random noise
- Goal is to recover these inherent dimensions and discard noise dimensions

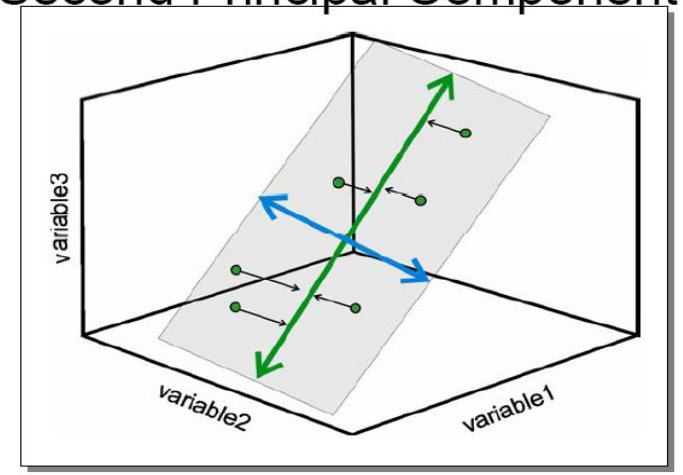
- takes a data matrix of *n* objects by ***p* variables, which may be correlated**, and **summarizes it by uncorrelated axes** (principal components or principal axes) that are linear combinations of the original *p* variables

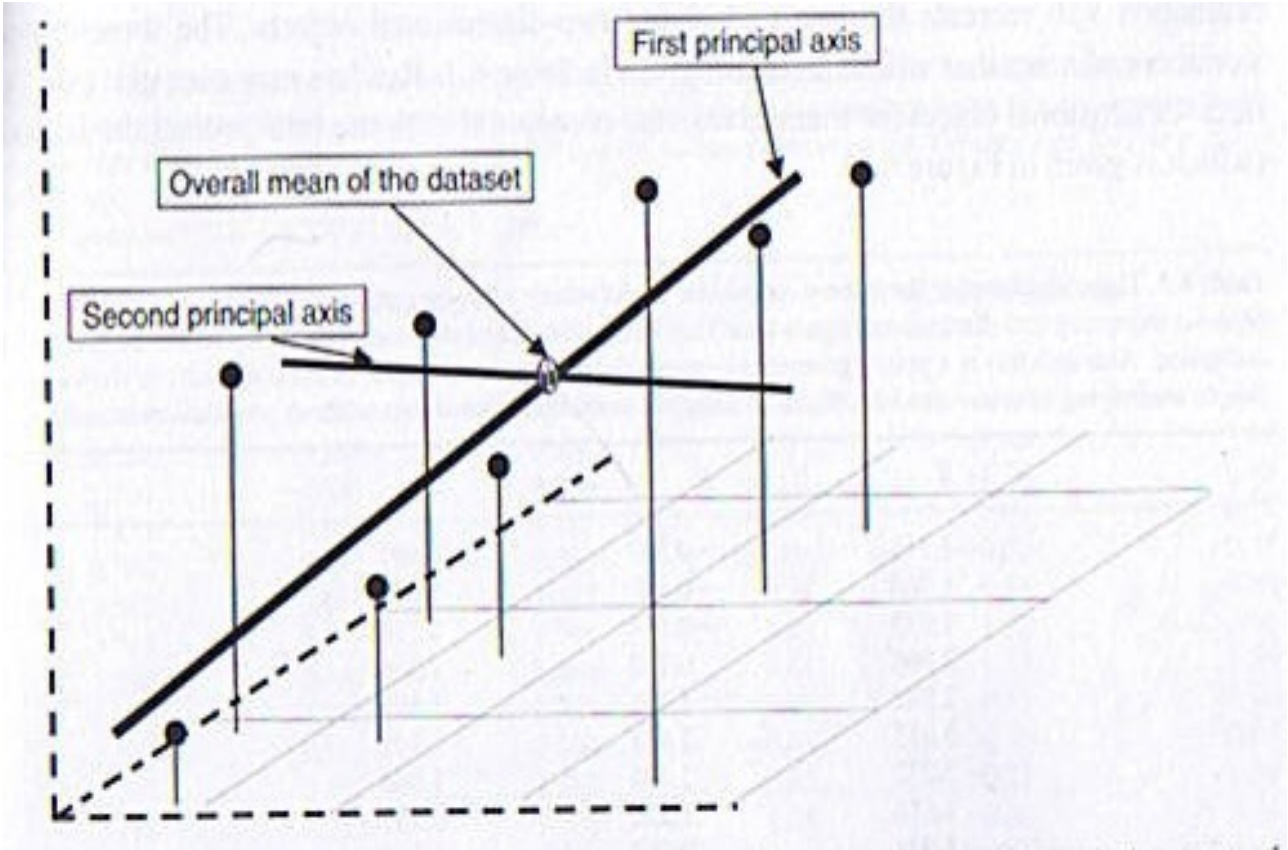
- the first *k* components display as much as possible of the variation among objects.

First Principal Component



Second Principal Component





Geometric Rationale of PCA

degree to which the variables are linearly correlated isrepresented by their covariances.

$$C_{ij} = \frac{1}{n-1}\sum_{m=1}^{n}\left(X_{im} - \overline{X}_i\right)\left(X_{jm} - \overline{X}_j\right)$$

Covariance of

Sum over all

Value of variable i in object m

Mean of variable i

Value of variable j in object m

Mean of variable j

The covariance matrix is a symmetric matrix $C=C^T$ (for the entries $C_{ij} = C_{ji}$)

The Spectral Theorem for Symmetric Matrices

An $n \times n$ symmetric matrix A has the following properties:

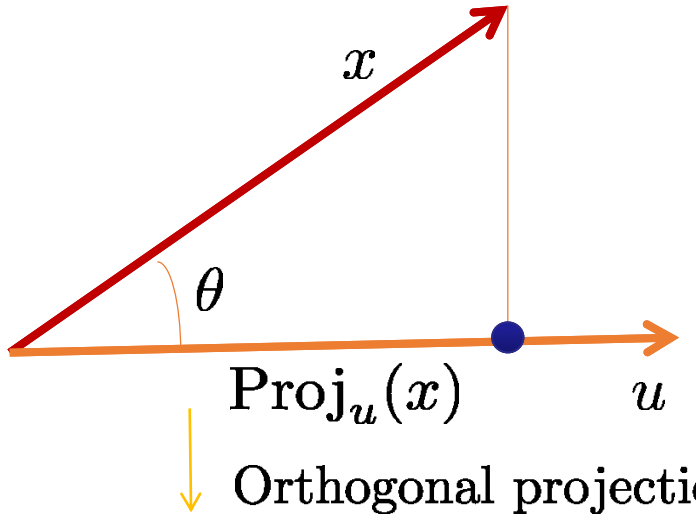
a. A has n real eigenvalues, counting multiplicities.

b. The dimension of the eigenspace for each eigenvalue λ equals the multiplicity of λ as a root of the characteristic equation.

c. The eigenspaces are mutually orthogonal, in the sense that eigenvectors corresponding to different eigenvalues are orthogonal.

d. A is orthogonally diagonalizable.

Projection on a Vector



Orthogonal projection of x on u

$$\text{Proj}_u(x) = \|x\| \cos\theta \frac{u}{\|u\|} = \|u\|\|x\| \cos\theta \frac{u}{\|u\|^2} = \frac{\langle u, x\rangle}{\langle u, u\rangle} u = \langle u, x\rangle u \text{ if } \|u\| = 1$$

$$= uu^T x \text{ if } \|u\| = 1$$

Projection on a Subspace

Let W be a subspace of $\mathbb{R}^n$. Let $U = [u_1, u_2, \ldots, u_k]$ be an $n \times k$ matrix, whose columns form an orthonormal basis of W. Then, the projection of a vector $x \in \mathbb{R}^n$ is given by

$$\text{Proj}_W(x) = UU^T x$$

Optimality Property of PCA

Orthogonal matrix

Main theoretical result:

The matrix G consisting of the first p eigenvectors of the covariance matrix S solves the following min problem:

$$\min_{G \in \Re^{d \times p}} \left\| X - G(G^T X) \right\|_F^2 \text{ subject to } \mathrm{G}^{\mathrm{T}} G = I_p$$

Projection on the Subspace generates by the Columns of G

$$\left\| X - \overline{X} \right\|_F^2$$

reconstruction error

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

Approximation error minimization

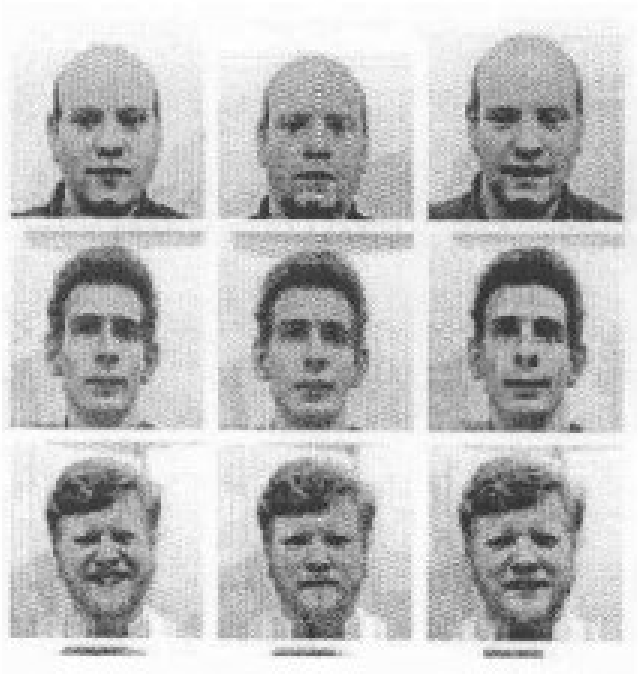
- A rank-k linear approximation model
- Fit the model with minimal reconstruction error
- Optimal condition
- Objective
- can be expressed as SVD of X, $X = U\Sigma V^T$

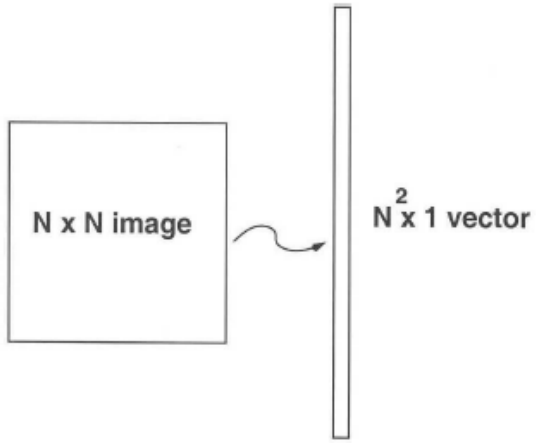
Application to Faces (from slides of CS479/679 Pattern Recognition Dr. George Bebis)

- Computation of low-dimensional basis (i.e.,eigenfaces):

Step 1: obtain face images $I_1, I_2, ..., I_M$ (training faces)

(**very important:** the face images must be *centered* and of the same *size*)





Step 2: represent every image I_i as a vector Γ_i

- Computation of the eigenfaces – cont.

Step 3: compute the average face vector Ψ:

$$\Psi = \frac{1}{M}\sum_{i=1}^{M}\Gamma_i$$

Step 4: subtract the mean face:

$$\Phi_i = \Gamma_i - \Psi$$

Step 5: compute the covariance matrix C:

$$C = \frac{1}{M}\sum_{n=1}^{M}\Phi_n\Phi_n^T = \frac{1}{M}AA^T \quad (N^2 \text{x} N^2 \text{ matrix})$$

where $A = [\Phi_1\ \Phi_2 \cdots \Phi_M] \quad (N^2 \text{x} M \text{ matrix})$

- Computation of the eigenfaces – cont.

Step 6: compute the eigenvectors u_i of AA^T ⇨ $AA^T u_i = \lambda_i u_i$

The matrix AA^T is very large --> not practical !!

Step 6.1: consider the matrix $A^T A$ (M x M matrix)

Step 6.2: compute the eigenvectors v_i of $A^T A$

$$A^T A v_i = \mu_i v_i$$

What is the relationship between us_i and v_i?

$A^T A v_i = \mu_i v_i$ => $AA^T A v_i = \mu_i A v_i$ =>

$CAv_i = \mu_i Av_i$ or $Cu_i = \mu_i u_i$ where $u_i = Av_i$

$$u_i = Av_i \quad and \quad \lambda_i = \mu_i$$

- Computation of the eigenfaces – cont.

Note 1: AA^T can have up to N^2 eigenvalues and eigenvectors.

Note 2: $A^T A$ can have up to M eigenvalues and eigenvectors.

Note 3: The M eigenvalues of $A^T A$ (along with their corresponding eigenvectors) correspond to the M *largest* eigenvalues of AA^T (along with their corresponding eigenvectors).

Step 6.3: compute the M best eigenvectors of AA^T: $u_i = Av_i$ (i.e., using A^TA)

(**important:** normalize u_i such that $\|u_i\| = 1$)

Step 7: keep only K eigenvectors (corresponding to the K largest eigenvalues)

each face Φ_i can be represented as follows:

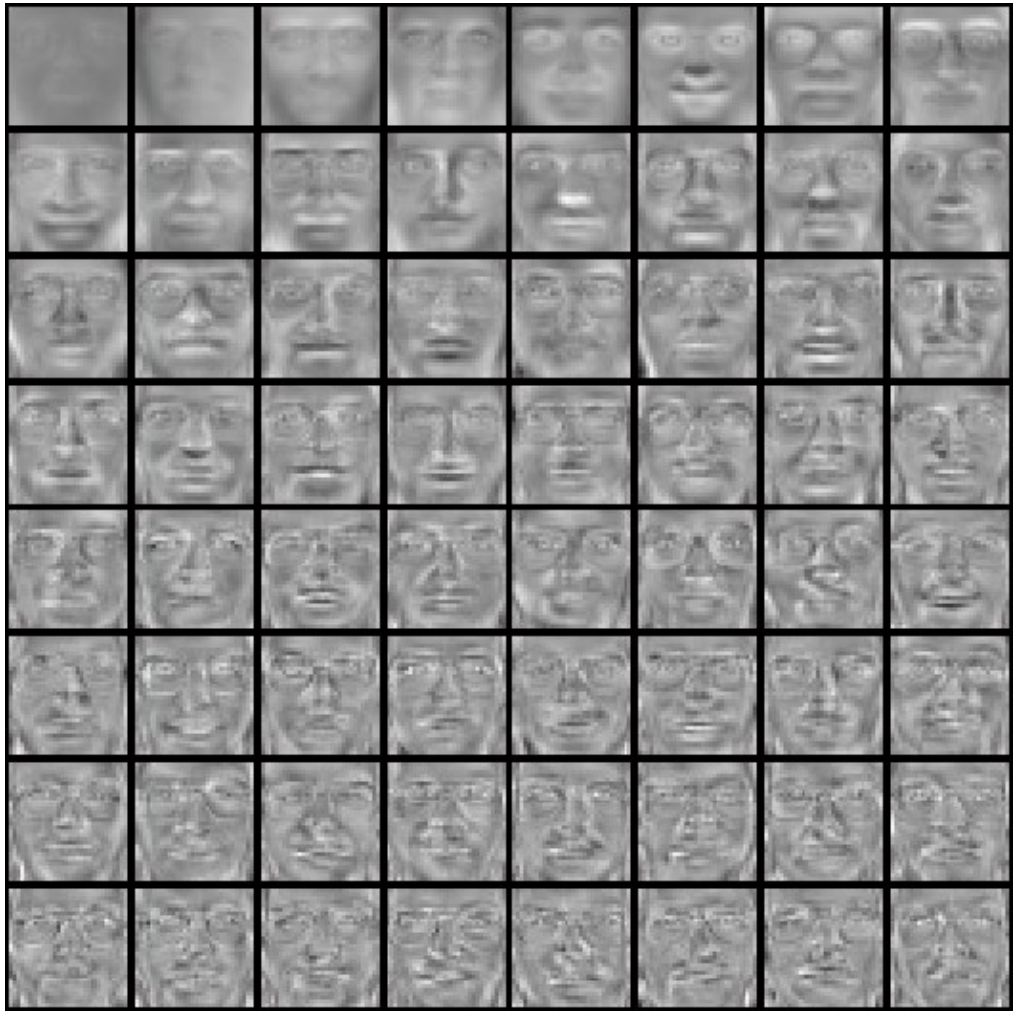
$$\hat{\Phi}_i - mean = \sum_{j=1}^{K} w_j u_j, \quad (w_j = u_j^T \Phi_i) \Rightarrow \Omega = \begin{bmatrix} w_1 \\ w_2 \\ \dots \\ w_K \end{bmatrix}$$

Example

Example (cont'd)

Mean: μ

Top eigenvectors: $u_1,\ldots u_k$



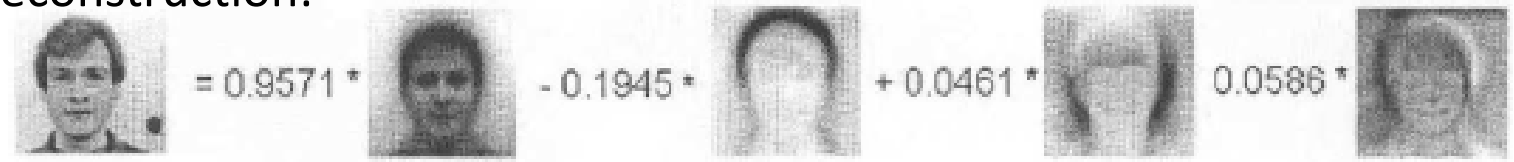
- Representing faces onto this basis

- Each face (minus the mean) Φ_i in the training set can be represented as a linear combination of the best K eigenvectors:

$$\hat{\Phi}_i - mean = \sum_{j=1}^{K} w_j u_j, \quad (w_j = u_j^T \Phi_i) \quad (where \| u_j \| = 1)$$

(we call the u_j's *eigenfaces*)

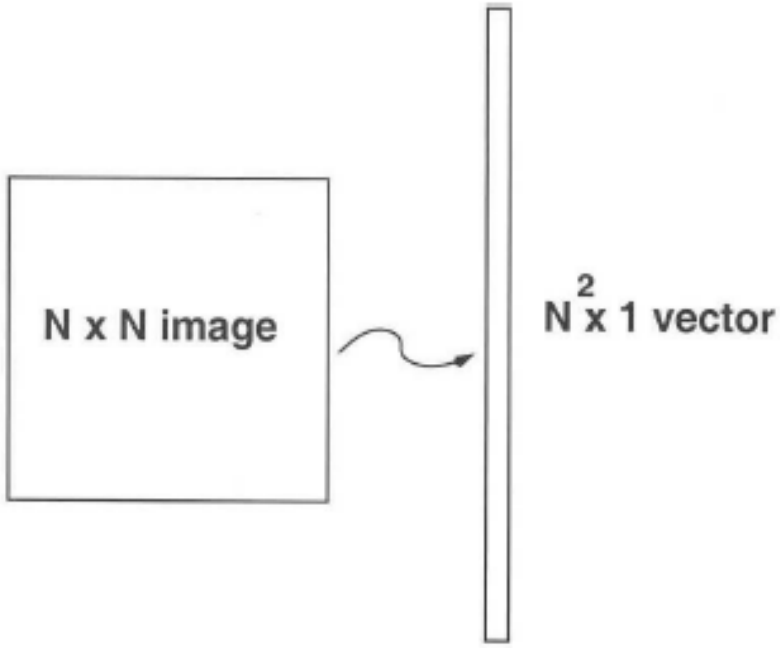
Face reconstruction:



Case Study: Eigenfaces for Face Detection/Recognition

- M. Turk, A. Pentland, "Eigenfaces for Recognition", *Journal of Cognitive Neuroscience*, vol. 3, no. 1, pp. 71-86, 1991.

- Face Recognition
  - The simplest approach is to think of it as a template matching problem.
  - Problems arise when performing recognition in a high-dimensional space.
  - Use *dimensionality reduction*!



Eigenfaces

- Face Recognition Using Eigenfaces

- Given an unknown face image Γ (centered and of the same size like the training faces) follow these steps:

Step 1: normalize Γ: $\Phi = \Gamma - \Psi$

Step 2: project on the eigenspace

$$\hat{\Phi} = \sum_{i=1}^{K} w_i u_i \quad (w_i = u_i^T \Phi) \qquad (where \| u_i \| = 1)$$

Step 3: represent Φ as: $\Omega = \begin{bmatrix} w_1 \\ w_2 \\ \dots \\ w_K \end{bmatrix}$

Step 4: find $e_r = \min_l \|\Omega - \Omega^l\|$ where $\| \Omega - \Omega^l \| = \sum_{i=1}^{K} (w_i - w_i^l)^2$

Step 5: if $e_r < T_r$, then Γ is recognized as face l from the training set.

The distance e_r is called *distance in face space (difs)*

Eigenfaces

- Face Detection Using Eigenfaces

- Given an unknown image Γ

Step 1: compute $\Phi = \Gamma - \Psi$

Step 2: compute $\hat{\Phi} = \sum_{i=1}^{K} w_i u_i \quad (w_i = u_i^T \Phi) \qquad (where \| u_i \| = 1)$

Step 3: compute $e_d = \|\Phi - \hat{\Phi}\|$

Step 4: if $e_d < T_d$, then Γ is a face.

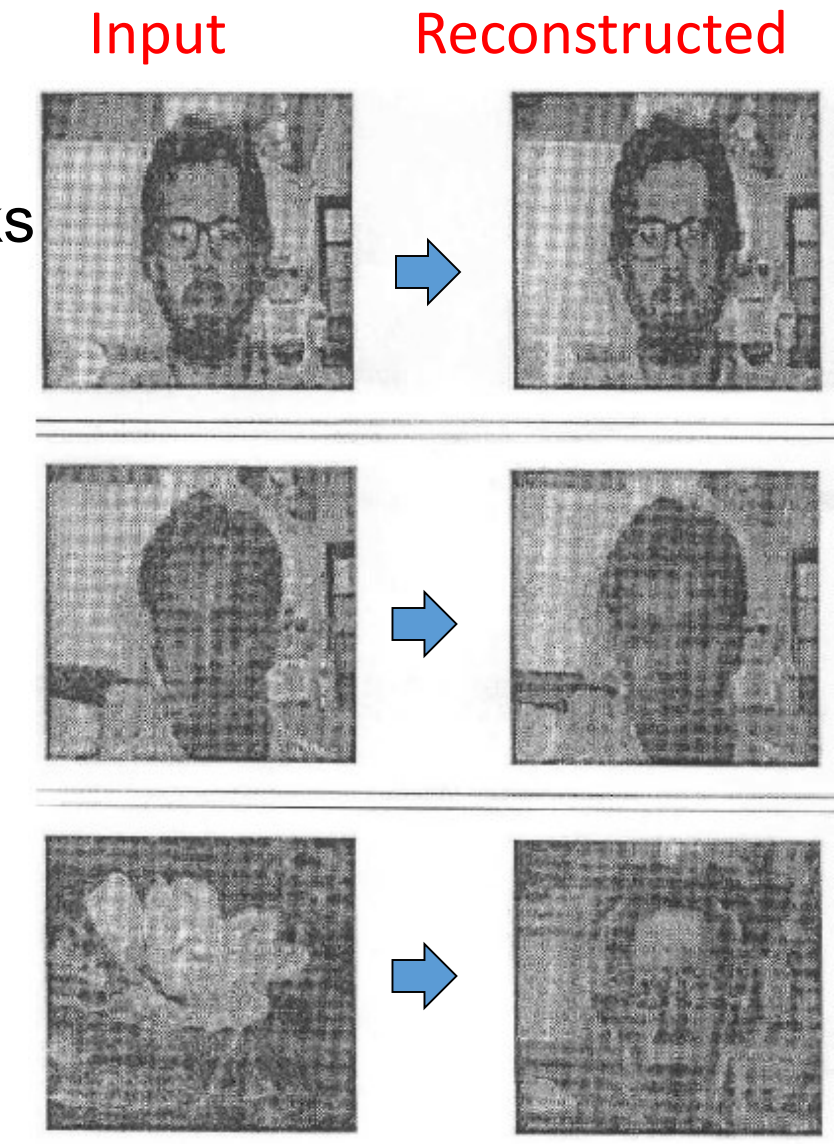
- The distance e_d is called distance from face space (dffs)

Eigenfaces

Reconstructed image looks like a face.

Reconstructed image looks like a face.

Reconstructed image looks like a face again!



Reconstruction using partial information

- Robust to partial face occlusion.

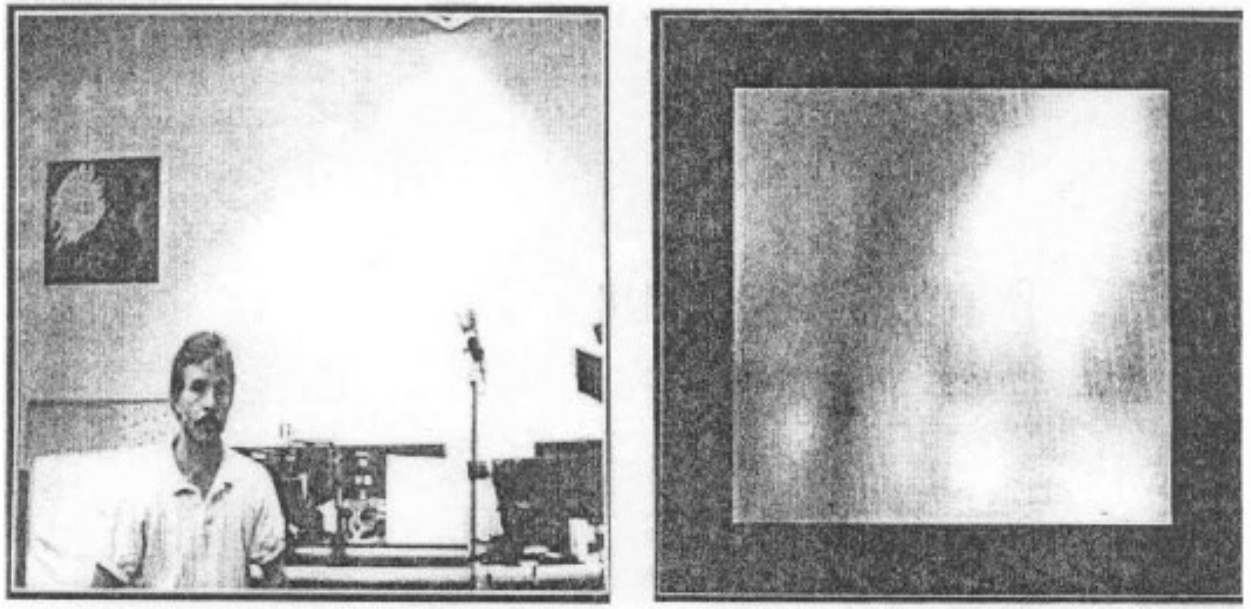
Input Reconstructed

Eigenfaces

- Face detection, tracking, and recognition

Visualize dffs:

$$e_d = \|\Phi - \hat{\Phi}\|$$



Limitations

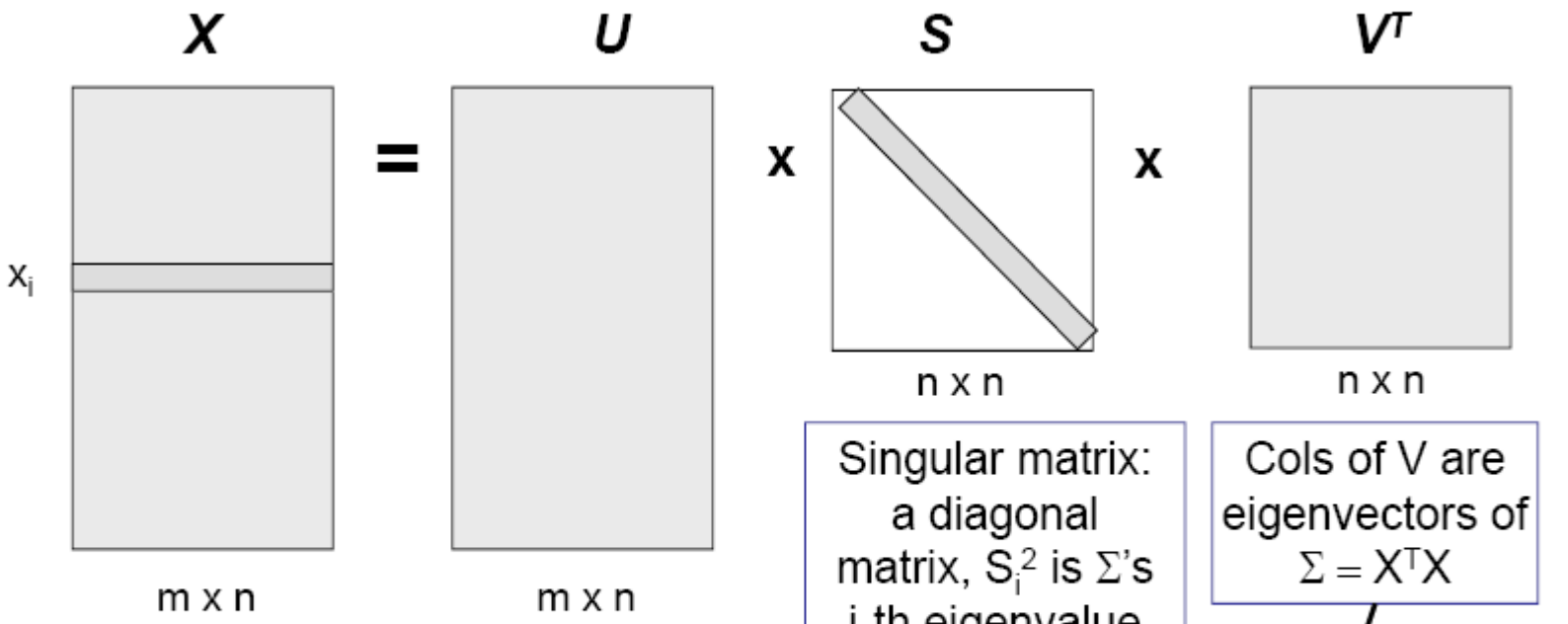
- Background changes cause problems
  - De-emphasize the outside of the face (e.g., by multiplying the input image by a 2D Gaussian window centered on the face).
- Light changes degrade performance
  - Light normalization might help but this is a challenging issue.
- Performance decreases quickly with changes to face size
  - Scale input image to multiple sizes.
  - Multi-scale eigenspaces.
- Performance decreases with changes to face orientation (but not as fast as with scale changes)
  - Out-of-plane rotations are more difficult to handle.
  - Multi-orientation eigenspaces.

Limitations (cont'd)

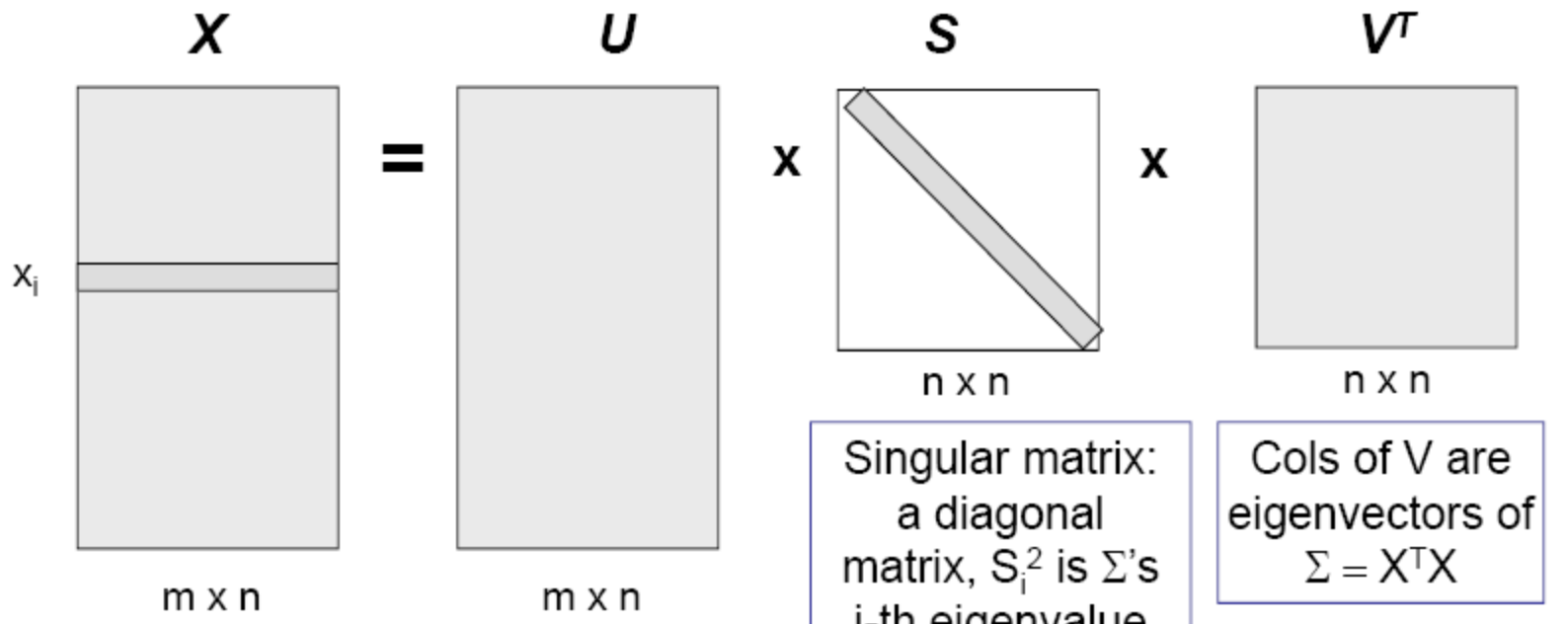
- Not robust to misalignment.

For non square matrix: Singular value decomposition (SVD)

$$X = U \cdot S \cdot V^T$$



$$X = U \cdot S \cdot V^T$$



X: our m x n data matrix, one row per data point

Each row of US gives coordinates of a data point in the projected space

Singular matrix: a diagonal matrix, S_i^2 is Σ's i-th eigenvalue

Cols of V are eigenvectors of $\Sigma = X^T X$

If X is centered, then cols of V are the principal components

SVD for PCA

- Create mean-centered data matrix $\mathbf{X}$.
- Solve SVD: $\mathbf{X} = \mathbf{U} \cdot \mathbf{S} \cdot \mathbf{V}^{\mathbf{T}}$.
- Columns of $\mathbf{V}$ are the eigenvectors of Σ sorted from largest to smallest eigenvalues.
- Select the first *k* columns as our *k* principal components.

Example in Data Mining: