Lecture 5b: Eigenvalues, Eigenvectors and Principal Component Analysis



Dr. Nelson Díaz
Departamento de Ingeniería Informática
Universidad Técnica Federico Santa María, campus Valparaíso, Chile

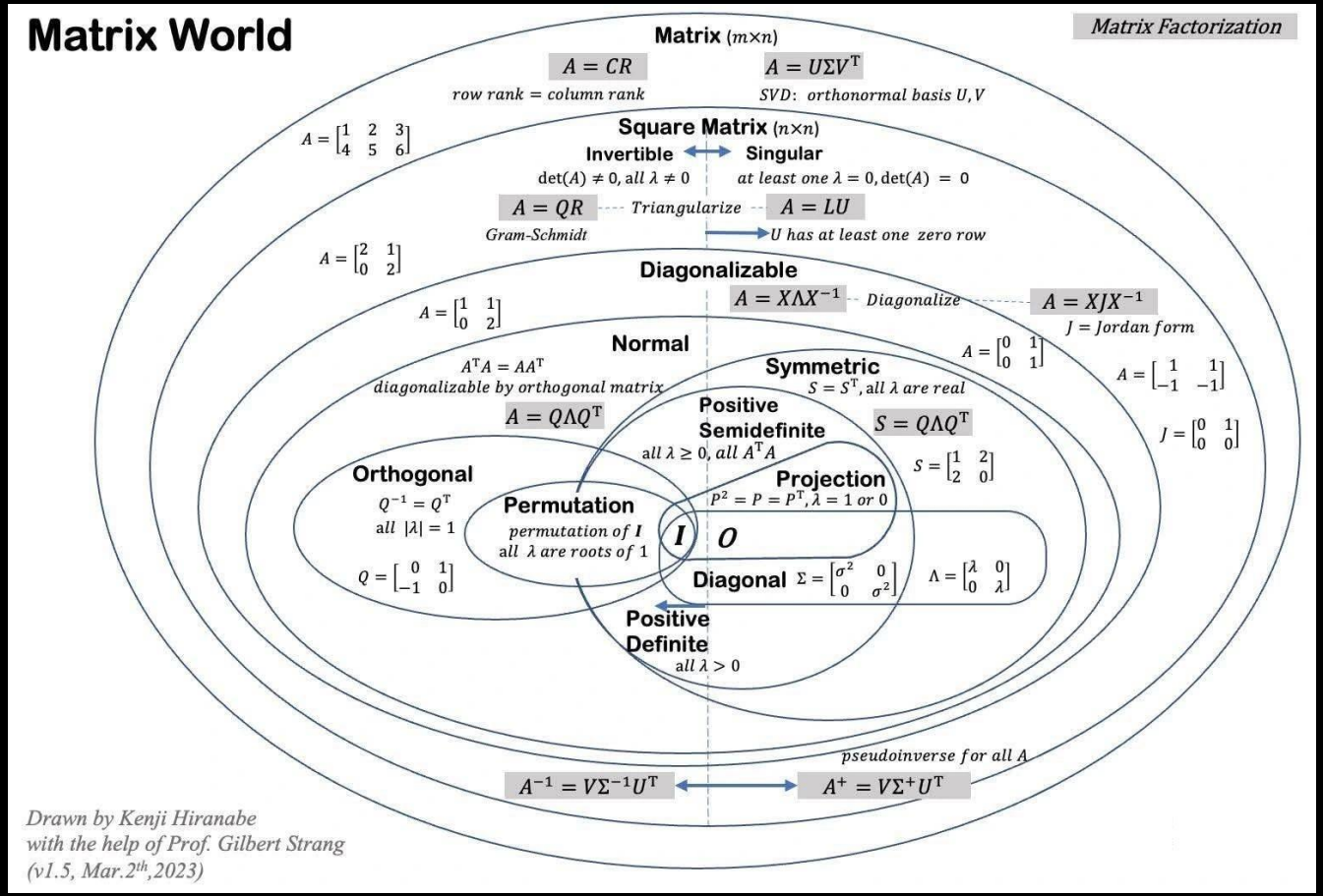
nelson.diazd@usm.cl

Academic website https://nelson-diaz.com/

UNIVERSIDAD TECNICA
FEDERICO SANTA MARIA

LabCI
COMPUTATIONAL IMAGING LABORATORY

Agenda


Matrix World
Matrix Factorization
Matrix (m×n)
A = CR
row rank = column rank
A = UΣV^T
SVD: orthonormal basis U, V
Square Matrix (n×n)
Invertible
det(A) ≠ 0, all λ ≠ 0
Singular
at least one λ = 0, det(A) = 0
A = QR
Gram-Schmidt
Triangularize
A = LU
U has at least one zero row
Diagonalizable
A = XΛX^-1
Diagonalize
A = XJX^-1
J = Jordan form
Normal
A^TA = AA^T
diagonalizable by orthogonal matrix
A = QΛQ^T
Symmetric
S = S^T, all λ are real
S = QΛQ^T
Positive Semidefinite
all λ ≥ 0, all A^TA
Orthogonal
Q^-1 = Q^T
all |λ| = 1
Permutation
permutation of I
all λ are roots of 1
Projection
P^2 = P = P^T, λ = 1 or 0
I
O
Diagonal
Positive Definite
all λ > 0
pseudoinverse for all A
A^-1 = VΣ^-1U^T
A^+ = VΣ^+U^T
Drawn by Kenji Hiranabe
with the help of Prof. Gilbert Strang
(v1.5, Mar.2th,2023)

Motivation

Applications Areas

1. Google's PageRank Algorithm: Determine the importance of web pages based on link structures.
2. Principal Component Analysis: In data science and machine learning for dimensional reduction.
3. Electrical Engineering: RLC circuit, Resistance (R), Inductor (I) and condenser (C)
4. Markov process



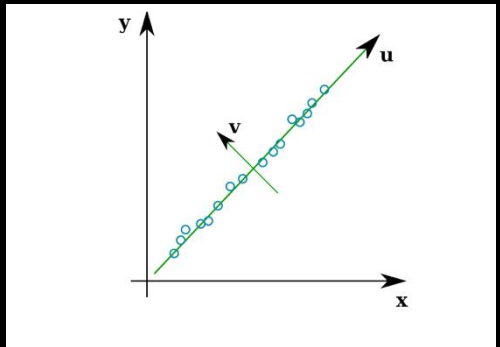
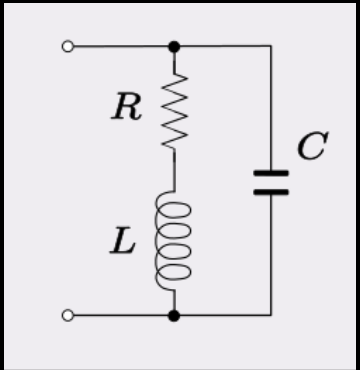


Figure 2: PCA for Dimension Reduction



https://www.ams.org/publicoutreach/feature-column/fcarc-pagerank

Example

El vector $\mathbf{x} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$ es un eigenvector de

$$A = \begin{bmatrix} 3 & 0 \\ 8 & -1 \end{bmatrix}$$

correspondiente al eigenvalor $\lambda = 3$, debido a que

$$A\mathbf{x} = \begin{bmatrix} 3 & 0 \\ 8 & -1 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \end{bmatrix} = \begin{bmatrix} 3 \\ 6 \end{bmatrix} = 3\mathbf{x}$$

Intuition

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

Goal

- It works like a function, i.e., x is the input and $\mathbf{A}x$ is the output
- The eigenvectors are vectors parallel to x
- How do we find x and λ?

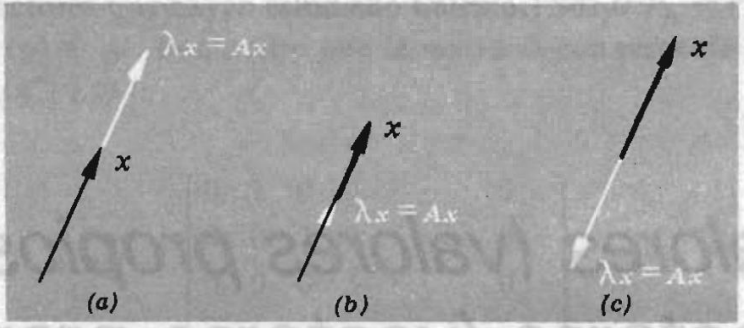


Figura 6.1 (*a*) Dilatación, $\lambda > 1$. (*b*) Contracción, $0 < \lambda < 1$. (*c*) Inversión de la dirección, $\lambda < 0$.

Compute eigenvalues and eigenvectors

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 and eigenvector x.

Example (part 1)

$\mathbf{A} = \begin{bmatrix} 4 & 2 \\ 3 & 3 \end{bmatrix}$ $\det(\mathbf{A} - \lambda\mathbf{I})$

$$\begin{bmatrix} 4 & 2 \\ 3 & 3 \end{bmatrix} - \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix} = \begin{bmatrix} 4-\lambda & 2 \\ 3 & 3-\lambda \end{bmatrix} = (4-\lambda)(3-\lambda) - 6$$

$$= 12 - 4\lambda - 3\lambda + \lambda^2 - 6$$

$$= \lambda^2 - 7\lambda + 6$$ Polynomial Characteristic

$$= (\lambda - 6)(\lambda - 1)$$

$\lambda = 6 \quad \lambda = 1$ Eigenvalues of the matrix

Example (part 2)

$\det(\mathbf{A} - \lambda\mathbf{I})\mathbf{x}$

$$\left[\begin{bmatrix}4 & 2\\ 3 & 3\end{bmatrix} - 1\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}\right]\begin{bmatrix}x_1\\ x_2\end{bmatrix} = 0$$

$$\begin{bmatrix}3 & 2\\ 3 & 2\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix} = 0$$

$$3x_1 + 2x_2 = 0$$
$$3x_1 + 2x_2 = 0$$

$$3x_1 = -2x_2$$
$$x_1 = 2$$
$$x_2 = -3$$

$$\begin{bmatrix}3 & 2\\ 3 & 2\end{bmatrix}\begin{bmatrix}2\\ -3\end{bmatrix} = 0$$ Eigenvector of the matrix

Example (part 3)

$\det(\mathbf{A} - \lambda\mathbf{I})\mathbf{x}$

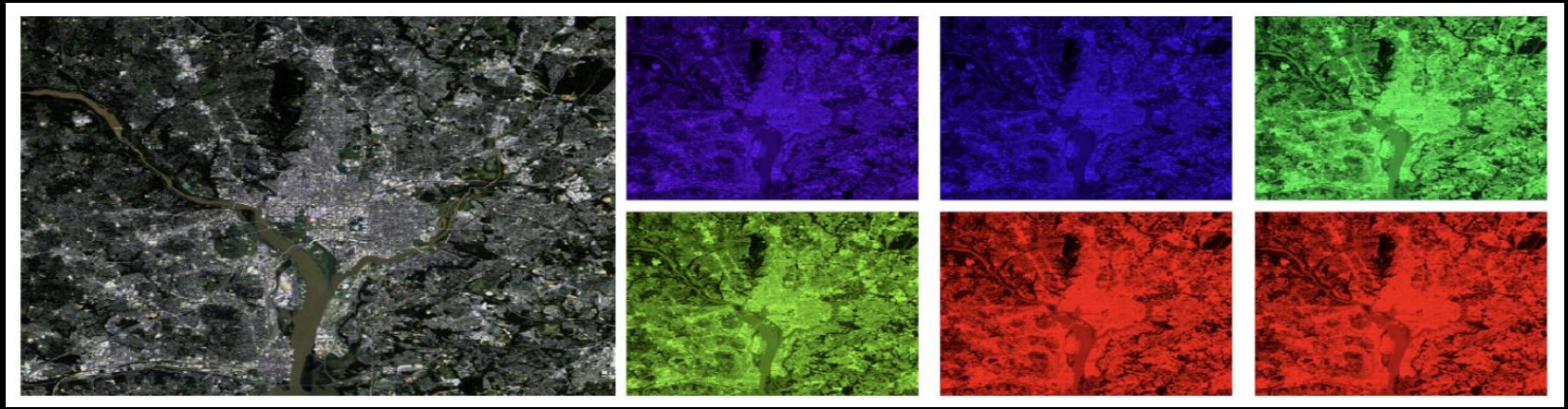
$$\left[\begin{bmatrix}4 & 2\\ 3 & 3\end{bmatrix} - 6\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}\right]\begin{bmatrix}x_1\\ x_2\end{bmatrix} = 0$$

$$\begin{bmatrix}-2 & 2\\ 3 & -3\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix} = 0$$

$$-2x_1 + 2x_2 = 0$$
$$3x_1 - 3x_2 = 0$$
$$-2x_1 = -2x_2$$
$$3x_1 = 3x_2$$
$$x_1 = 1$$
$$x_2 = 1$$

$$\begin{bmatrix}-2 & 2\\ 3 & -3\end{bmatrix}\begin{bmatrix}1\\ 1\end{bmatrix} = 0$$ Eigenvector of the matrix

Application 1: Multispectral images



Introduction

Principal component analysis (PCA) is a technique for data dimensionality reduction by computing the covariance structure of a set of variables.

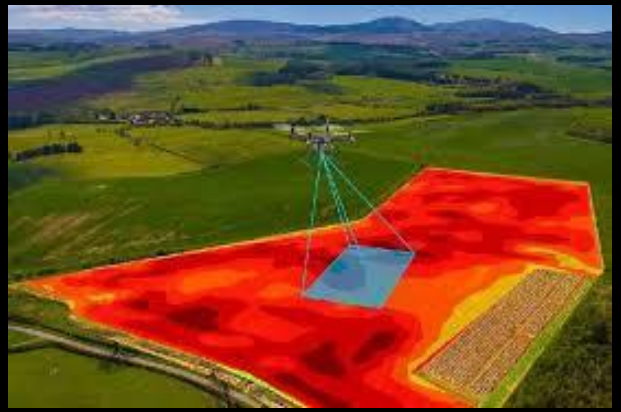
Goal

- To identify the principal direction in which the data varies

$$\hat{\mathbf{x}} = \mathbf{A}_k^T \mathbf{y} - \mathbf{m}_\mathbf{x}$$



- A set of faces



- Multispectral images

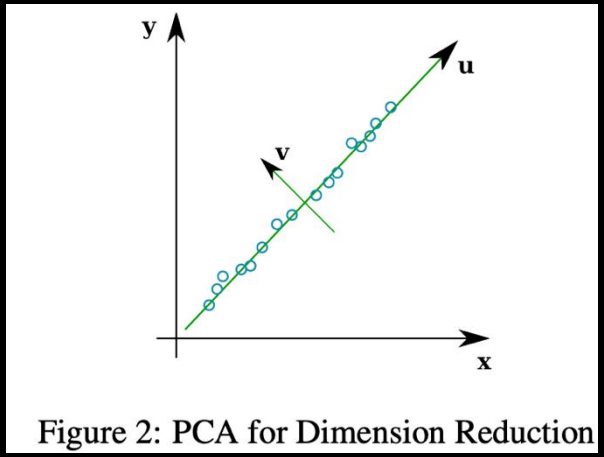


Figure 2: PCA for Dimension Reduction

- Plot that show dimensionality reduction

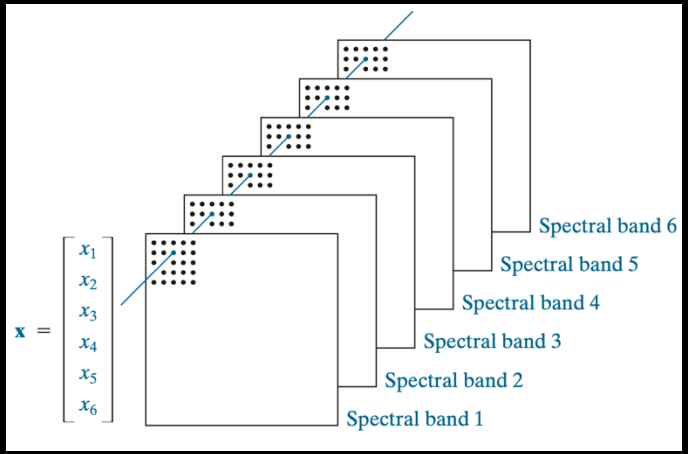
Mathematical Model

Given a n-dimensional vector, where $x \in \mathbb{R}^n$

$$\mathbf{X} = \begin{bmatrix} \mathrm{x}_1 \\ \mathrm{x}_2 \\ \vdots \\ \mathrm{x}_n \end{bmatrix} \qquad x_i \in \mathbb{R}^L$$

n=6

L is the number of pixels per image



Note that all vectors are column vectors (i.e, matrices of order n x 1.)

To compute the PCA, we must compute the mean vectors and covariance matrices.

The mean vector of the population is defined as

$$\mathbf{m}_\mathbf{x} = E\{\mathbf{X}\}$$

Note:

1. $E\{\mathbf{X}\}$ is the expected value of $\mathbf{X}$
2. The subscript denotes that $\mathbf{m}$ is associated with the population of $\mathbf{X}$ vectors.
3. The expected value of vectors and matrices is the expected value of each element

Computing the covariance matrix

The covariance matrix of the vector population is defined as

$$\mathbf{C_x} = \frac{1}{L}\{(\mathbf{X} - \boldsymbol{m_x})\,(\mathbf{X} - \boldsymbol{m_x})^T\}$$

Because $\mathbf{x}$ is n dimensional, $\mathbf{C_x}$ is an n x n matrix, where T indicates transpose.

Note:

1. Element c_{ii} of $\mathbf{C_x}$ is the variance of x_i the i^{th} component of the $\mathbf{x}$ vectors in the population
2. and element c_{ij} of $\mathbf{C_x}$ is the covariance between elements x_i and x_j of these vectors.
3. Because $\mathbf{C_x}$ is real and symmetric, finding a set of n orthonormal eigenvectors is always possible.

Hotelling transform

- Let $\mathbf{e}_i$ and λ_i, $i = 1,2, \ldots, n$, be the eigenvectors and the corresponding eigenvalues of $\mathbf{C_x}$
- By definition, the eigenvector and eigenvalues of an n x n matrix $\mathbf{C}$ satisfy the equation $\mathbf{C}\mathbf{e}_i = \lambda_i \mathbf{e}_i$
- Let $\mathbf{A}$ be a matrix whose rows are formed from the eigenvectors of $\mathbf{C_x}$ arranged in descending value of their eigenvalues, so that the first row of $\mathbf{A}$ is the eigenvector corresponding to the largest eigenvalue.
- Suppose that we use $\mathbf{A}$ as a transformation matrix to map the $\mathbf{x}$'s into vectors denoted by $\mathbf{y}$'s, as follows:

$$\mathbf{y} = \mathbf{A}(\mathbf{X} - \mathbf{m_x})$$

Note

Reconstruction property: Because the rows of $\mathbf{A}$ are orthonormal vectors, it follows that $\mathbf{A^{-1}} = \mathbf{A^T}$, and any vector $\mathbf{X}$ can be recovered from its corresponding $\mathbf{y}$

Reconstruction property (PCA)

Because the rows of $\mathbf{A}$ are orthonormal vectors, it follows that $\mathbf{A}^{-1} = \mathbf{A}^{T}$, and any vector $\mathbf{x}$ can be recovered from its corresponding $\mathbf{y}$

$$\mathbf{x} = \mathbf{A}^{T}\mathbf{y} - \mathbf{m}_{\mathbf{x}}$$

But, suppose that, instead of using all the eigenvectors of $\mathbf{C}_{\mathbf{x}}$, we form a matrix $\mathbf{A}_k$ from k largest eigenvectors corresponding to the k largest eigenvalues, yielding a transformation matrix of order $k \times n$. The $\mathbf{y}$ vectors would then be exact

$$\hat{\mathbf{x}} = \mathbf{A}_{k}^{T}\mathbf{y} - \mathbf{m}_{\mathbf{x}}$$

Then the mean square error between $\mathbf{x}$ and $\hat{\mathbf{x}}$ is given by the expression

$$e_{ms} = \sum_{j=1}^{n} \lambda_j - \sum_{j=1}^{k} \lambda_j = \sum_{j=k+1}^{n} \lambda_j$$

PCA matlab code (1)

```
1   clear all;
2   close all;
3   clc;
4   N = 564;
5   L = 6;
6   X = zeros(N,N,L);
7   x = zeros(L,N*N);
8   m = zeros(L,1);
9   X(:,:,1)= imread("WashingtonDC_a.tif");
10  X(:,:,2)= imread("WashingtonDC_b.tif");
11  X(:,:,3)= imread("WashingtonDC_c.tif");
12  X(:,:,4)= imread("WashingtonDC_d.tif");
13  X(:,:,5)= imread("WashingtonDC_e.tif");
14  X(:,:,6)= imread("WashingtonDC_f.tif");
15
16  for l=1:L
17      temp = X(:,:,l);
18      x(l,:) = temp(:);
19      m(l) = mean(temp(:));
20  end
```

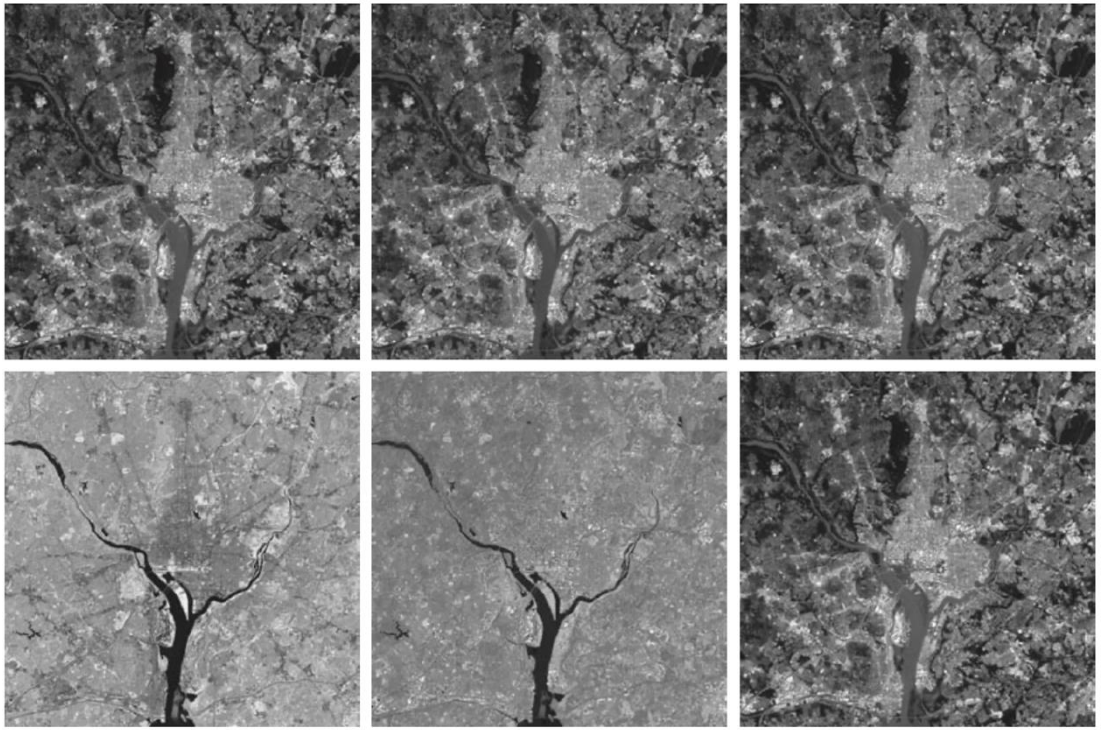
PCA matlab code (2)

```
22  C = 1/(N*N)*((x - m)*(x - m)');
23  [V,D] = eig(C); % Compute eigenvalues and eigenvectors
24  [E,I]=sort(diag(D),'descend');
25  A = V(:,I)';
26  % Principal components
27  y = A*(x-m);
28  %% Verification
29  A*C*A';
30
31  figure(1)
32  colormap('gray')
33  title("Principal Components")
34  for l=1:L
35      Y(:,:,l) = reshape(y(l,:),[N,N]);
36      subplot(2,3,l),imagesc(Y(:,:,l))
37  end
38
39
40  % Projecting the data
41  k = 2;
42  xhat = A(1:k,:)'*y(1:k,:) + m;
43
44  figure(2)
45  title("Projection using k components")
46  colormap('gray')
47  for l=1:L
48      Xhat(:,:,l) = reshape(xhat(l,:),[N,N]);
49      subplot(2,3,l),imagesc(Xhat(:,:,l))
50  end
```

PCA example:1

6 Multispectral images

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$



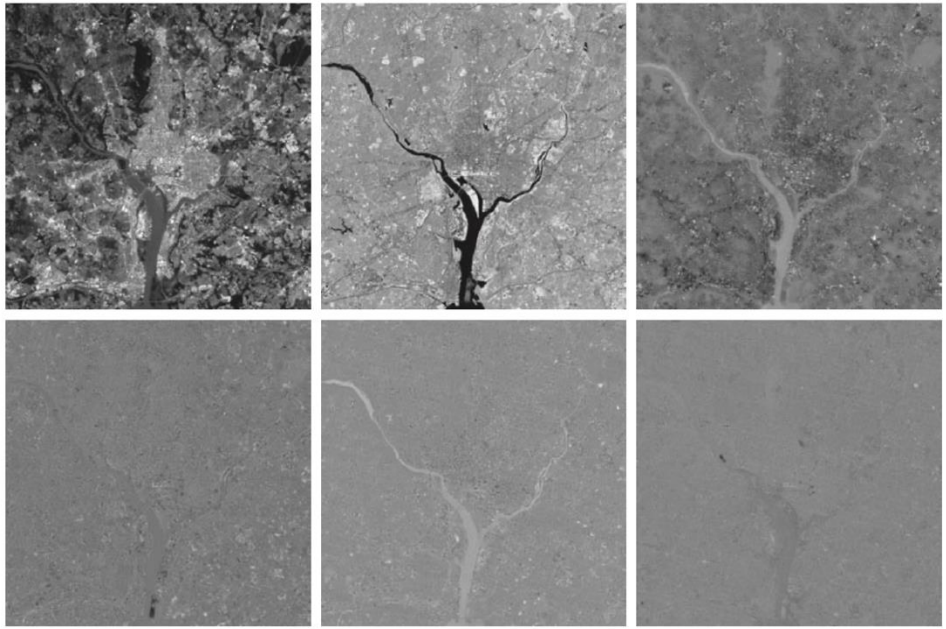
a b c
d e f

FIGURE 11.38 Multispectral images in the (a) visible blue, (b) visible green, (c) visible red, (d) near infrared, (e) middle infrared, and (f) thermal infrared bands. (Images courtesy of NASA.)

PCA example

Principal components:

$$\mathbf{y} = \mathbf{A}(\mathbf{x} - \mathbf{m}_\mathbf{x})$$



a b c
d e f

FIGURE 11.40 The six principal component images obtained from vectors computed using Eq. (11-49). Vectors are converted to images by applying Fig. 11.39 in reverse.

λ_1	λ_2	λ_3	λ_4	λ_5	λ_6
10344	2966	1401	203	94	31

Thanks for your attention!

