Applied Statistical Multivariate Analysis
Matrix Decomposition

Instructor: Yaser Zerehsaz

موسسه آموزش عالی آزاد توسعه
برگزار کننده دوره‌های تخصصی علم داده

Tihe.ac.ir

Eigendecomposition

Purpose:

- Analyzing the structure of a matrix

Why is it important?

- it is used to find the maximum (or minimum) of functions involving covariance or correlation matrices
- “eigen” in German means “specific”
- principal component analysis is obtained from the eigen-decomposition of a covariance matrix and gives the least square estimate of the original data matrix

Eigendecomposition-theory

The eigenvector and eigenvalue of matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$ must satisfy:

$$\mathbf{Au} = \lambda\mathbf{u}$$

where $\mathbf{u} \neq \mathbf{0} \in \mathbb{R}^{\boldsymbol{n}}$and λ is a scalar.

All eigenvalues are roots (real or complex) of equation

$$|\mathbf{A} - \lambda\mathbf{I}| = \mathbf{0}$$

- There are n eigenvalues (not necessarily distinct)
- If at least one of the eigenvalues are zero: $|\mathbf{A} - \lambda\mathbf{I}| = |\mathbf{A}| = \mathbf{0}$, $\mathbf{A}$ is singular.
- If $\mathbf{A}$ is positive definite ($\mathbf{u}^T\mathbf{Au} > 0$ for all $\mathbf{u} \neq \mathbf{0}$) all $\lambda > 0$
- If $\mathbf{A}$ is positive semi definite ($\mathbf{u}^T\mathbf{Au} \geq 0$ for all $\mathbf{u} \neq \mathbf{0}$) all $\lambda \geq 0$

Eigendecomposition-theory

- $|\mathbf{A}| = \prod_{i=1}^{n} \lambda_i$
- $tr(\mathbf{A}) = \sum_{i=1}^{n} \lambda_i$

Put all eigenvectors in matrix $\mathbf{U} \in \mathbb{R}^{n\times n}$ and define $\mathbf{\Lambda} \in \mathbb{R}^{n\times n}$ as

$$\mathbf{\Lambda} = \begin{pmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & \cdots & 0 & \lambda_\mathrm{n} \end{pmatrix}$$

We will have

$$\mathbf{AU} = \mathbf{U\Lambda}$$

$$\mathbf{A} = \mathbf{U\Lambda U}^{-1}$$

For a real-valued, symmetric matrix U is always invertible

Eigendecomposition-example

$$\mathbf{A} = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix} \qquad |\mathbf{A} - \lambda\mathbf{I}| = \mathbf{0}$$

$$\left|\begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix} - \begin{pmatrix} \lambda & 0 \\ 0 & \lambda \end{pmatrix}\right| = 0 \qquad \left|\begin{pmatrix} 2-\lambda & 0 \\ 0 & 1-\lambda \end{pmatrix}\right| = 0$$

$$(1-\lambda)(2-\lambda) = 0 \qquad \lambda_1 = 2, \lambda_2 = 1$$

$$\mathbf{Au} = \lambda\mathbf{u}$$

$$\mathbf{u}_1 = \begin{pmatrix} u_{11} \\ u_{21} \end{pmatrix}, \mathbf{u}_2 = \begin{pmatrix} u_{12} \\ u_{22} \end{pmatrix}$$

$$\begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} u_{11} \\ u_{21} \end{pmatrix} = \lambda_1 \begin{pmatrix} u_{11} \\ u_{21} \end{pmatrix}, \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} u_{12} \\ u_{22} \end{pmatrix} = \lambda_2 \begin{pmatrix} u_{12} \\ u_{22} \end{pmatrix}$$

$$u_{11} = u_{11} \in \mathbb{R}, u_{21} = 0 \rightarrow \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 2 \\ 0 \end{pmatrix}, \begin{pmatrix} -100 \\ 0 \end{pmatrix} \dots \begin{pmatrix} c \neq 0 \\ 0 \end{pmatrix}$$

$$u_{12} = 0, u_{22} = u_{22} \in \mathbb{R} \rightarrow \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ -3 \end{pmatrix}, \begin{pmatrix} 0 \\ 8.5 \end{pmatrix} \dots \begin{pmatrix} 0 \\ c \neq 0 \end{pmatrix}$$

Eigendecomposition-example

- There can be an infinite number of eigenvectors in association with each eigenvalue
- Normalize each eigenvector to get $||\mathbf{u}||=1$
- For real-valued, symmetric matrices the eigenvectors are orthonormal. In this case we will have

$$\mathbf{U}^{-1} = \mathbf{U}^{T}$$

$$\mathbf{A} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^{T}$$

g=np.linalg.eig(**A**)

(array([2., 1.]), array([[1., 0.],
[0., 1.]]))

$\boldsymbol{\Lambda}$ =np.diag(g[0])

array([[2., 0.],
[0., 1.]])

U.dot(delt).dot(np.matrix.transpose(**U**))

array([[2., 0.],
[0., 1.]])

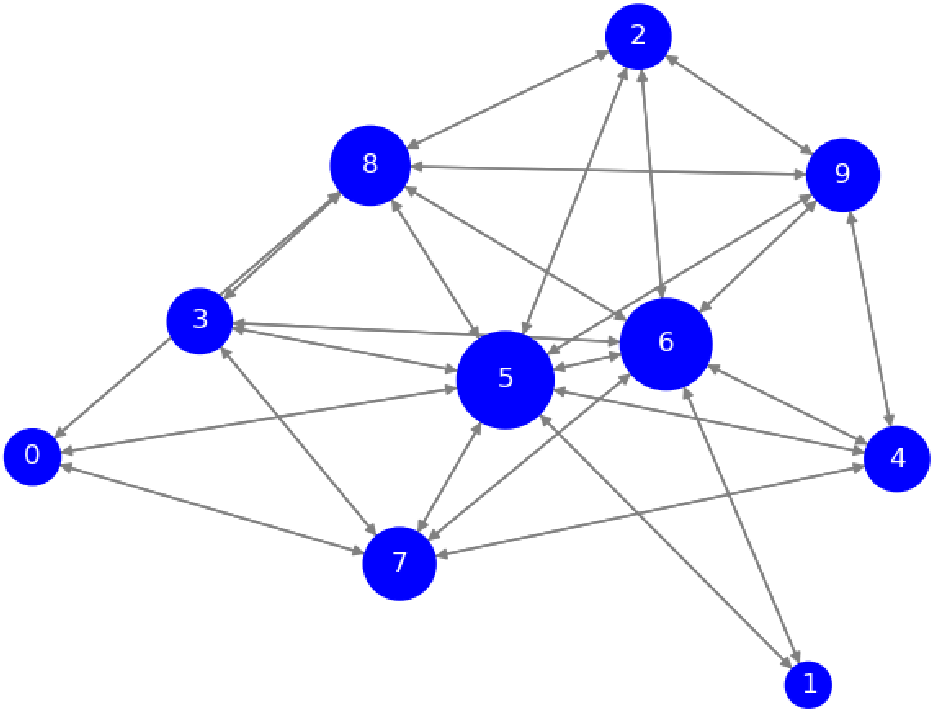
PageRank Algorithm

- This is the first algorithm proposed by Lary Page et al. (1998) to rank pages based on their importance
- Advanced versions of PR are used in google search engines
- The algorithm works based on link analysis
- According to google

"PageRank works by counting the number and quality of links to a page to determine a rough estimate of how important the website is. The underlying assumption is that more important websites are likely to receive more links from other websites"

PageRank Algorithm

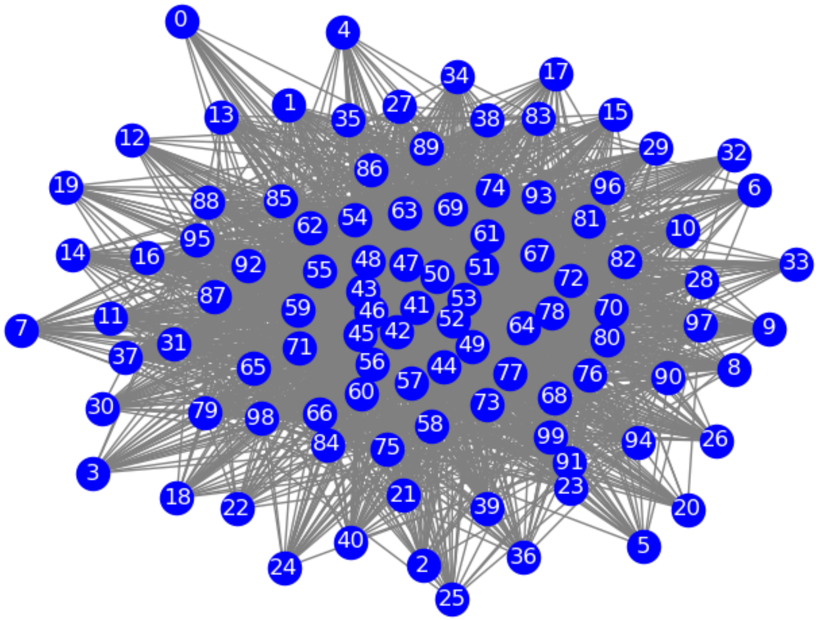
- The algorithm deals with a set of pages as nodes and the links among pages as edges
- The adjacency matrix of a small network is given as

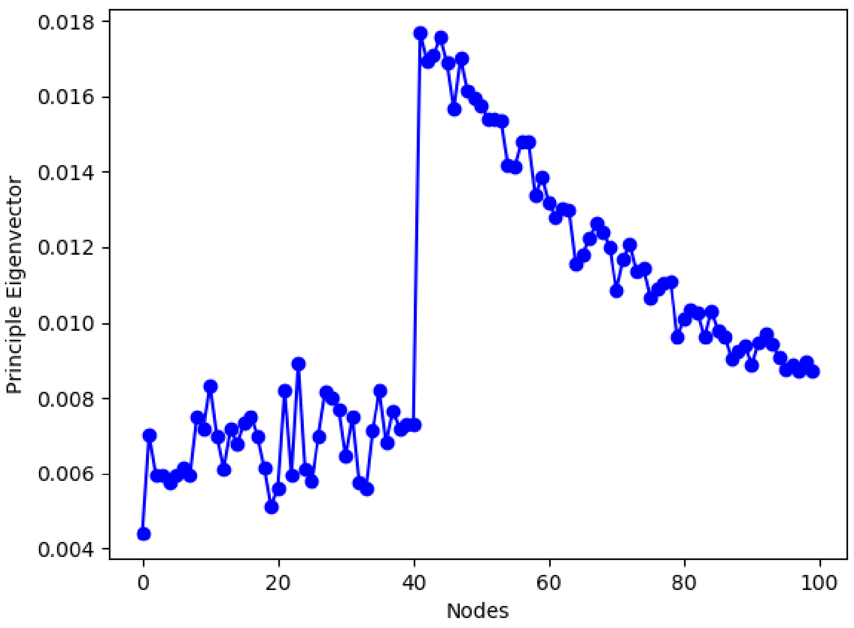


[[0., 0., 0., 0., 0., 1., 0., 1., 1., 0.],
[0., 0., 0., 0., 0., 1., 1., 0., 0., 0.],
[0., 0., 0., 0., 0., 1., 1., 0., 1., 1.],
[0., 0., 0., 0., 0., 1., 1., 1., 1., 0.],
[0., 0., 0., 0., 0., 1., 1., 1., 0., 1.],
[1., 1., 1., 1., 1., 0., 1., 1., 1., 1.],
[0., 1., 1., 1., 1., 1., 0., 1., 1., 1.],
[1., 0., 0., 1., 1., 1., 1., 0., 0., 0.],
[1., 0., 1., 1., 0., 1., 1., 0., 0., 1.],
[0., 0., 1., 0., 1., 1., 1., 0., 1., 0.]]

PageRank Algorithm

- It can be shown that the solution of the PageRank algorithm is the principal eigenvector of the adjacency matrix





Singular Value Decomposition (SVD)

For any real rectangular matrix $\mathbf{A} \in \mathbb{R}^{n \times p}$, we can write

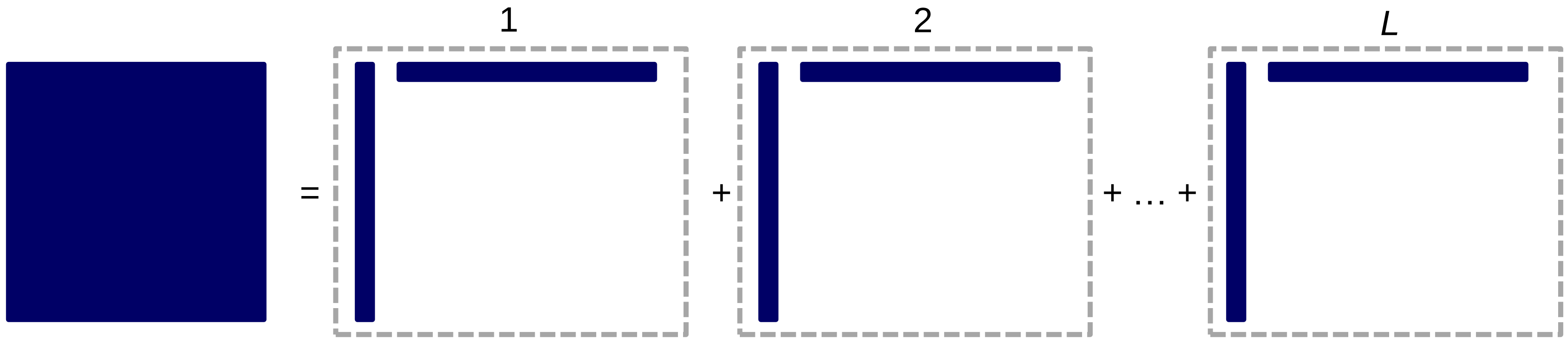
$$\mathbf{A} = \sum_{l=1}^{L} d_l \mathbf{u}_l \mathbf{v}_l^T = \mathbf{U}\mathbf{D}\mathbf{V}^T$$

where $\mathbf{U} \in \mathbb{R}^{n \times n}$ and $\mathbf{V} \in \mathbb{R}^{p \times p}$ are orthogonal square matrices and $\mathbf{D} \in \mathbb{R}^{n \times m}$

$$\mathbf{D} = \begin{pmatrix} diag(d_1, \dots, d_L) & 0 \\ 0 & 0 \end{pmatrix}$$

- By convention $d_1 \geq d_2 \geq \cdots \geq d_L$
- Rank of matrix $\mathbf{A}$ is L
- Diagonal elements of $\mathbf{D}$ are square roots of $\mathbf{A}^T\mathbf{A}$ eigenvalues
- Columns of $\mathbf{V}$ (right singular vectors) are the eigenvectors of $\mathbf{A}^T\mathbf{A}$
- Columns of $\mathbf{U}$ (left singular vectors) are the eigenvectors of $\mathbf{A}\mathbf{A}^T$

SVD- Problem definition



$$\mathbf{A} = \sum_{l=1}^{L} d_l \mathbf{u}_l \mathbf{v}_l^T + \mathbf{E}$$

minimize $\left\| \mathbf{A} - \widehat{\mathbf{A}} \right\|$

s.t.

$$\mathbf{U}^T\mathbf{U} = \mathbf{I}, \mathbf{V}^T\mathbf{V} = \mathbf{I}$$

with $\widehat{\mathbf{A}} = \mathbf{UDV}^T$.

The best rank k approximation to matrix $\mathbf{A}$ can be obtained by keeping the first k singular values and replacing the rest by 0.

```
np.linalg.svd(A)
                    (array([[ 1.,  0.],
                            [ 0.,  1.]]), array([ 2.,  1.]), array([[ 1.,  0.],
                                                                  [ 0.,  1.]]))
```

Hot Rolling

- is the plastic deformation of metals passed between rolls
- is a widely used forming process
- occurs above the recrystallization temperature of metals

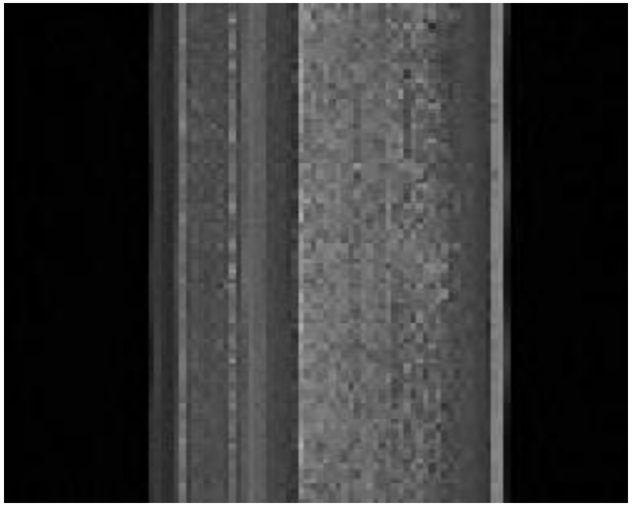
An Introduction to Image Processing

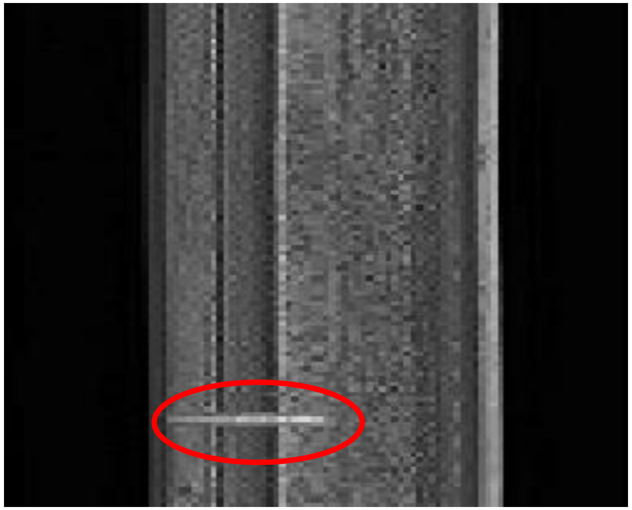
Billets

- Semi-finished products needing further operation
- Used to produce rods or bar stock

Defects

- Deviation in the appearance
- Caused by Mechanical stresses, thermal stresses, etc.
- Images are taken from billets using Hot-eye cameras





موسسه آموزش عالی آزاد توسعه

برگزار کننده دوره‌های تخصصی مهندس صنایع، مدیریت و کسب و کار

وب سایت: www.tihe.ac.ir

تلفن: 021-86741 داخلی ۱۲۰ – ۱۲۴ و ۱۲۵

کانال تلگرام: @tiheac

Tihe.ac.ir