University of Chicago Professional Education

MSCA 37016

Advanced Linear Algebra for Machine Learning

Session 4

Shaddy Abado Ph.D.

Session #4: Agenda

- Eigenvalues and Eigenvectors
- Diagonalization: $A = S\Lambda S^{-1}$
- Singular Value Decomposition (SVD)
- Principle Component Analysis (PCA)

BASIC CONCEPTS NEEDED FOR THIS SESSION

THE UNIVERSITY OF CHICAGO

Recall: Symmetric Matrix

A symmetric matrix is one for which $\mathbf{A}^{\mathrm{T}} = \mathbf{A}$.

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{12} & a_{22} & a_{23} \\ a_{13} & a_{23} & a_{33} \end{bmatrix}$$

$$\mathbf{A}^{\mathrm{T}} = \mathbf{A}$$

$$or$$

$$a_{ij} = a_{ji}$$

$$A = \begin{bmatrix} 1 & 3 & 4 \\ 3 & 2 & 5 \\ 4 & 5 & -1 \end{bmatrix} \qquad A^T = \begin{bmatrix} 1 & 3 & 4 \\ 3 & 2 & 5 \\ 4 & 5 & -1 \end{bmatrix}$$

Recall: Inner and Outer Products

A vector operating on a vector can yield a scalar or a matrix, depending on the order of operation.

$$\begin{bmatrix} Row1 \\ Row2 \end{bmatrix} \begin{bmatrix} Col1 & Col2 & Col3 \end{bmatrix}$$
$$= \begin{bmatrix} Row1\ X\ Col1 & Row1\ X\ Col2 & Row1\ X\ Col3 \\ Row2\ X\ Col1 & Row2\ X\ Col2 & Row2\ X\ Col3 \end{bmatrix}$$

$$v_{3X1} = \begin{bmatrix} 2 \\ 0 \\ 1 \end{bmatrix}$$

Inner Product

$$v^T{}_{1X3} \cdot v_{3X1} = \begin{bmatrix} 2 & 0 & 1 \end{bmatrix} \begin{bmatrix} 2 \\ 0 \\ 1 \end{bmatrix} = 2*2+0*0+1*1 = 5$$

Scalar

Outer Product

$$v_{3X1} \cdot v^T{}_{1X3} = \begin{bmatrix} 2 \\ 0 \\ 1 \end{bmatrix} \begin{bmatrix} 2 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 4 & 0 & 2 \\ 0 & 0 & 0 \\ 2 & 0 & 1 \end{bmatrix}$$

Matrix

Recall: Inverse Matrices

A is singular, i.e., not invertible

If $\boldsymbol{A}$ is invertible, then $\boldsymbol{A}x = \vec{0}$ can only have the zero solution $x = \boldsymbol{A}^{-1} \cdot \vec{0} = \vec{0}$ **(Trivial solution)**

$$\boldsymbol{det(A) \neq 0}$$

Suppose there is a non-zero vector x such that $\boldsymbol{A}x = \vec{0}$. Then A cannot have an inverse (i.e., $\boldsymbol{A}$ is singular).

$$\boldsymbol{det(A) = 0}$$

Recall: Orthogonal Matrices

- Matrix $\boldsymbol{Q}$ is orthogonal if all its columns are orthogonal to each other.
- If the columns are also normalized $(norm = 1)$ then the matrix is called orthogonal.

Recall (Orthogonal Matrix):

- $\boldsymbol{Q^T Q = I}$
- $\boldsymbol{Q}$ is not required to be square
- When $\boldsymbol{Q}$ is square, $\boldsymbol{Q^T Q = Q Q^T = I}$ means that $\boldsymbol{Q^T = Q^{-1}}$

EIGENVALUES AND EIGENVECTORS

THE UNIVERSITY OF CHICAGO

Introduction to Eigenvalues and Eigenvectors

A Few Notes Before Starting

A matrix $\boldsymbol{A}$ *acts on* vectors $\boldsymbol{x}$ like a function does, with input $\boldsymbol{x}$ and output $\boldsymbol{A}\boldsymbol{x}$

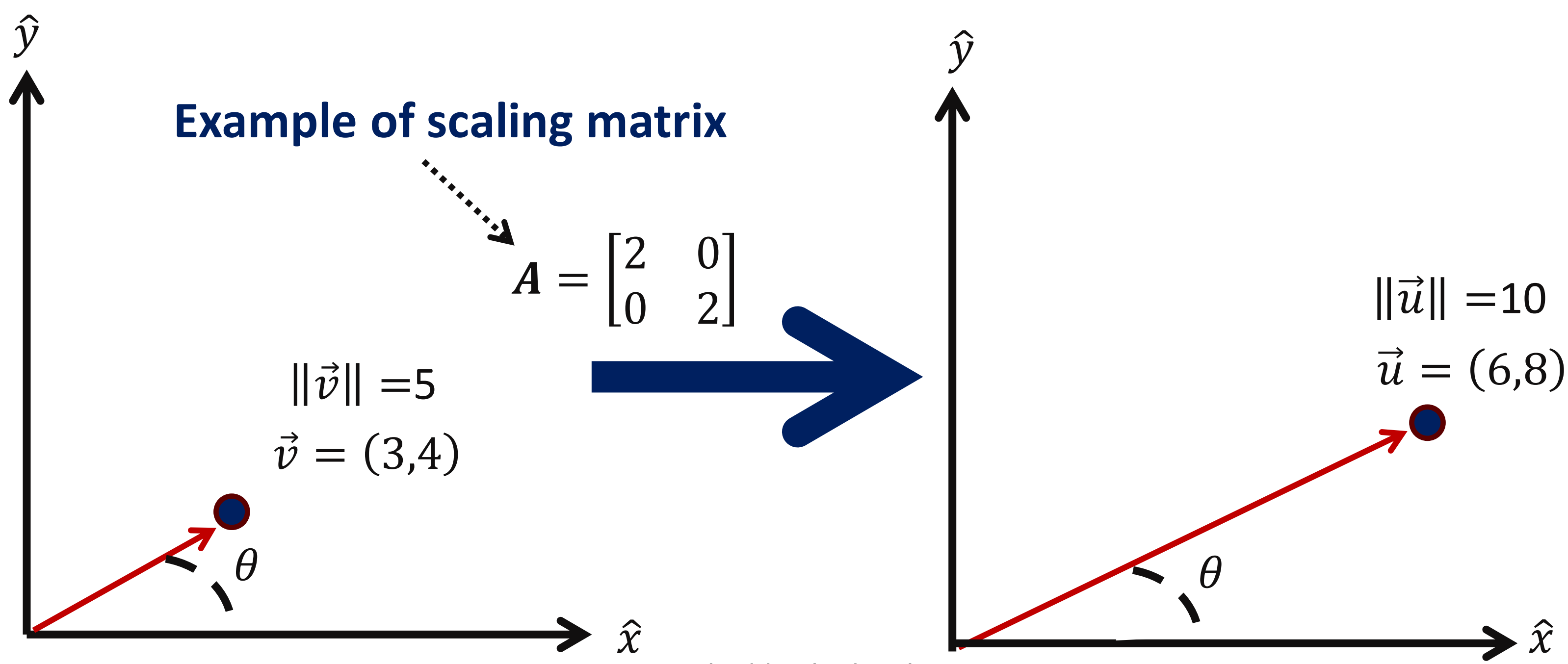
$$x = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \longrightarrow \boxed{\boldsymbol{A} = \begin{bmatrix} 1 & 1 \\ -2 & 1 \end{bmatrix}} \longrightarrow Ax = \begin{bmatrix} 2 \\ -1 \end{bmatrix}$$

A Few Notes Before Starting – Scaling Matrix

Scaling Matrix –

Leaves the direction of the vector unchanged, but changes its length

$$x = \begin{bmatrix} 3 \\ 4 \end{bmatrix} \rightarrow A = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \rightarrow Ax = \begin{bmatrix} 6 \\ 8 \end{bmatrix}$$



A Few Notes Before Starting – Rotation Matrix

Rotation Matrix –

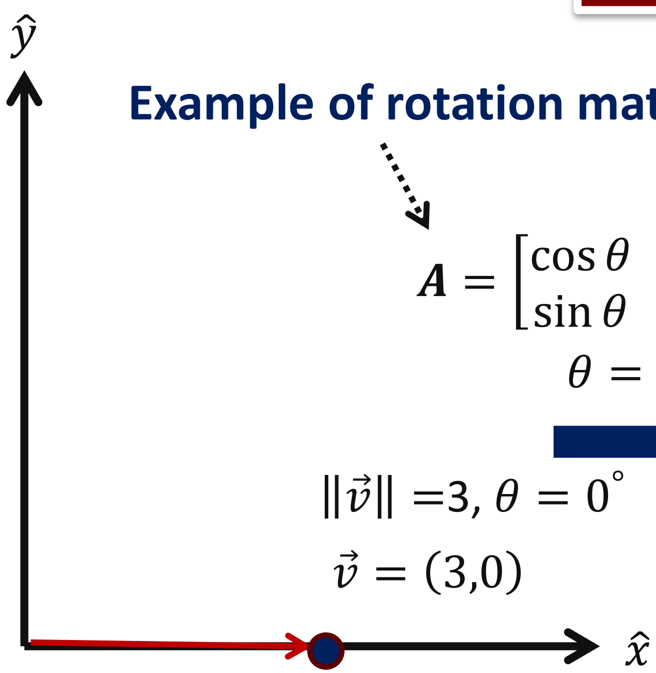
Changes the direction of vector, but leaves its norm unchanged

$$x = \begin{bmatrix} 3 \\ 0 \end{bmatrix} \rightarrow A = \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} \rightarrow Ax = \begin{bmatrix} \frac{3}{\sqrt{2}} \\ \frac{3}{\sqrt{2}} \end{bmatrix}$$

Example of rotation matrix

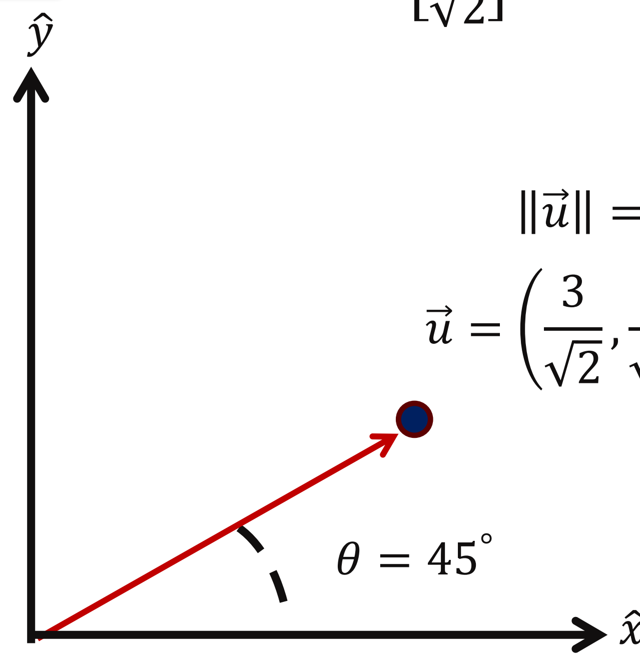
$$\boldsymbol{A} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$$

$$\theta = 45^{\circ}$$



$\|\vec{v}\| = 3$, $\theta = 0^{\circ}$

$\vec{v} = (3,0)$



$\|\vec{u}\| = 3$

$\vec{u} = \left(\frac{3}{\sqrt{2}}, \frac{3}{\sqrt{2}}\right)$

Motivation: Session #4

- Balance, equilibrium and steady state
- Solved by elimination

(LSE, System of Equations, etc.)

$$\boldsymbol{A}_{nxn}x = b$$

$$\boldsymbol{A}_{mxn}x = b$$

Linear System of Equations

Least Squares

- Change/ transient state
- **Not** solved by elimination

(PDE, ODE, Dynamic Systems, etc.)

$$\boldsymbol{A}_{nxn}x = \lambda x$$

Eigenvalues and Eigenvectors

Introduction

Eigenvalues and Eigenvectors

For at least some matrices, some vectors are special, i.e., multiplication by A just takes them to scalar multiples of themselves

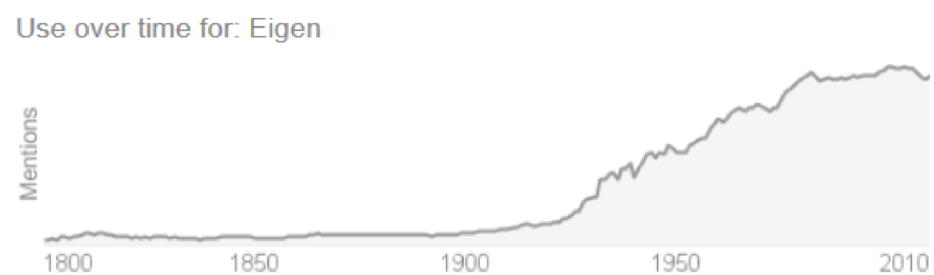
- Eigenvalues and eigenvectors are special numbers and vectors associated with square matrices.

Eigen

proper; characteristic.

Origin : German

Source: dictionary.com



Definitions:

Eigenvalues and Eigenvectors

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

<u>Eigenvectors</u> (i.e., x's) are the vectors that are unchanged, except in magnitude, when multiplied by matrix A.

<u>Eigenvalues</u> (i.e., λ's) are scaling factors.

- **<u>Eigenvectors</u>** are vectors for which Ax is parallel to x
- **<u>Eigenvalues</u>** tell us whether the eigenvector is
  - stretched (e.g., $\lambda = 2$),
  - shrunk (e.g., $\lambda = 1/2$),
  - reflected (e.g., $\lambda = -1$) or
  - left unchanged ($\lambda = 1$)

Eigenvalues and Eigenvectors – Characteristic Equation

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

x and λ are unknown

$$\boldsymbol{A}x - \lambda x = \boldsymbol{0} \Longrightarrow (\boldsymbol{A} - \lambda I)x = \boldsymbol{0}$$

Non-trivial solution ($x \neq 0$) can be found only if $\det(\boldsymbol{A} - \lambda I) = 0$ (Singular)

Recall session 2

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

The polynomial equation that arises in the eigenvalue problem is the ***characteristic equation*** of the matrix

Once we've found an eigenvalue λ, we can use elimination to find the nullspace of $A - \lambda I$. The vectors in that nullspace are eigenvectors of $\boldsymbol{A}$ with eigenvalue λ.

Eigenvalues and Eigenvectors – Solution Steps

Step 1: Compute the determinant of $\boldsymbol{A} - \lambda I$

(Subtract λ along the diagonal of $\boldsymbol{A}$)

Step 2: Find the roots of the characteristic equation

Step 3: For each eigenvalue, solve

$(\boldsymbol{A} - \lambda \boldsymbol{I})x = 0$ to find the eigenvectors

Note: Will not be discussed:

- Complex eigenvalues
- Solving differential equations

Example –
Eigenvalues and Eigenvectors

$$\boldsymbol{A}x = \lambda\boldsymbol{x}$$
$$|\boldsymbol{A} - \lambda I| = \boldsymbol{0}$$

Find the Eigenvalues and Eigenvectors of

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

$$|\boldsymbol{A} - \lambda I| = \boldsymbol{0}$$
$$\begin{vmatrix} 3-\lambda & 1 \\ 1 & 3-\lambda \end{vmatrix} = 0$$
$$(3-\lambda)^2 - 1 = 0$$
$$\lambda^2 - 6\lambda + 8 = 0$$

Step 1: Compute the determinant of $\boldsymbol{A} - \lambda I$ (Subtract λ along the diagonal of $\boldsymbol{A}$)

Step 2: Find the roots of the characteristic equation

$$\lambda = 2 \qquad \lambda = 4$$

Unique Eigenvalues

Python – Roots of a Polynomial

numpy.roots

numpy.roots(*p*) [source]

Return the roots of a polynomial with coefficients given in p.

The values in the rank-1 array *p* are coefficients of a polynomial. If the length of *p* is n+1 then the polynomial is described by:

```
p[0] * x**n + p[1] * x**(n-1) + ... + p[n-1]*x + p[n]
```

Parameters:	**p** : *array_like* Rank-1 array of polynomial coefficients.
Returns:	**out** : *ndarray* An array containing the roots of the polynomial.
Raises:	**ValueError** When *p* cannot be converted to a rank-1 array.

See also:

$$\lambda^2 - 6\lambda + 8 = 0$$

$$\lambda = 2, 4$$

```
In [5]: import numpy as np
   ...: coeff = [1, -6, 8]
   ...: print(np.roots(coeff))
   ...:
[ 4.  2.]
```


https://docs.scipy.org/doc/numpy/reference/generated/numpy.roots.html

Example – Eigenvalues and Eigenvectors

$$\boldsymbol{A}x = \lambda\boldsymbol{x}$$
$$|\boldsymbol{A} - \lambda I| = \boldsymbol{0}$$

Recall:

Once we've found an eigenvalue λ, we can use elimination to find the nullspace of $\boldsymbol{A} - \lambda\boldsymbol{I}$. The vectors in that nullspace are eigenvectors of $\boldsymbol{A}$ with eigenvalue λ.

$$\lambda_1 = 2$$
$$\begin{bmatrix} 3-2 & 1 \\ 1 & 3-2 \end{bmatrix}\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$
$$\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

The nullspace is an entire line; the eigenvector (x_1) could be any vector on that line.

$$x_1 = t\begin{bmatrix} -1 \\ 1 \end{bmatrix}$$

$$\lambda_2 = 4$$
$$\begin{bmatrix} 3-4 & 1 \\ 1 & 3-4 \end{bmatrix}\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$
$$\begin{bmatrix} -1 & 1 \\ 1 & -1 \end{bmatrix}\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

The nullspace is an entire line; the eigenvector (x_2) could be any vector on that line.

$$x_2 = t\begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

Example – Eigenvalues and Eigenvectors

$$\boldsymbol{A}x = \lambda \boldsymbol{x}$$
$$|\boldsymbol{A} - \lambda I| = \boldsymbol{0}$$

Usually, the eigenvectors are normalized (magnitude of 1)

$$x_1 = \begin{bmatrix} -1 \\ 1 \end{bmatrix} \qquad x_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$$x_1 = \begin{bmatrix} -1/\sqrt{2} \\ 1/\sqrt{2} \end{bmatrix} \qquad x_2 = \begin{bmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \end{bmatrix}$$

Example – Eigenvalues and Eigenvectors

$$\boldsymbol{A}x = \lambda \boldsymbol{x}$$
$$|\boldsymbol{A} - \lambda I| = \boldsymbol{0}$$

Summary

$$x_1 = \begin{bmatrix} -1 \\ 1 \end{bmatrix}, \quad x_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

Eigenvectors

Eigenvalues

$$\lambda_1 = 2, \quad \lambda_2 = 4$$

Check

$$\boldsymbol{A} \cdot x_1 = \begin{bmatrix} 3 & 1 \\ 1 & 3 \end{bmatrix} \begin{bmatrix} -1 \\ 1 \end{bmatrix} = \begin{bmatrix} -2 \\ 2 \end{bmatrix} = 2 * \begin{bmatrix} -1 \\ 1 \end{bmatrix} = \lambda_1 x_1$$

$$\boldsymbol{A} \cdot x_2 = \begin{bmatrix} 3 & 1 \\ 1 & 3 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 4 \\ 4 \end{bmatrix} = 4 * \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \lambda_2 x_2$$

The vectors x_1 and x_2 are unchanged , except in magnitude

Python –
Eigenvalues and Eigenvectors

numpy.linalg.eig

numpy.linalg.eig(*a*) [source]

Compute the eigenvalues and right eigenvectors of a square array.

Parameters: **a** : *(..., M, M) array*

Matrices for which the eigenvalues and right eigenvectors will be computed

Returns: **w** : *(..., M) array*

The eigenvalues, each repeated according to its multiplicity. The eigenvalues are not necessarily ordered. The resulting array will be of complex type, unless the imaginary part is zero in which case it will be cast to a real type. When *a* is real the resulting eigenvalues will be real (0 imaginary part) or occur in conjugate pairs

v : *(..., M, M) array*

The normalized (unit "length") eigenvectors, such that the column `v[:,i]` is the eigenvector corresponding to the eigenvalue `w[i]`.

Raises: **LinAlgError**

If the eigenvalue computation does not converge.

```
from numpy import linalg as la
import numpy as np

A = np.array([[3,1],
              [1,3]])
EigVal, EigVect = la.eig(A)
```

```
eigenvalues
[ 4.+0.j  2.+0.j]
eigenvalues (Real)
[ 4.  2.]


eigenvectors
[[ 0.70710678 -0.70710678]
 [ 0.70710678  0.70710678]]
```

numpy.linalg.eig gives

- ➢ Normalized eigenvectors (as matrix columns)
- ➢ Eigenvalues in decreasing order

https://docs.scipy.org/doc/numpy/reference/generated/numpy.roots.html

Eigenvalues and Eigenvectors – $2x2$ Matrix

In general, the eigenvalues of a $2x2$ matrix are the solutions to:

$$\lambda^2 - trace(A)\lambda + det(A) = 0$$

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

$$trace(A) = 3 + 3 = 6$$

$$det(A) = 9 - 1 = 8$$

$$\lambda^2 - 6\lambda + 8 = 0 \qquad \lambda = 2,4$$

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

$$trace(A) = 1 + 1 = 2$$

$$det(A) = 1 - 0 = 1$$

$$\lambda^2 - 2\lambda + 1 = 0 \qquad \lambda = 1,1$$

Eigenvalues and Eigenvectors - Properties

An nxn matrix will have n eigenvalues, and their sum will be the trace of the matrix.

Recall:

The trace of a square matrix A_{nxn} is the sum of its diagonal elements.

$$tr\ A = \sum_{i=1}^{N} a_{ii}$$

Example I

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

$\lambda_1 = 2$

$\lambda_2 = 4$

$$trace(A) = 3 + 3 = 6$$

$$\lambda_1 + \lambda_2 = 2 + 4 = 6$$

Eigenvalues and Eigenvectors - Properties

Just as the trace is the sum of the eigenvalues of a matrix, the product of the eigenvalues of <u>any matrix</u> equals its determinant.

Example I

$$\lambda_1 = 2$$
$$\lambda_2 = 4$$

$$\boldsymbol{A} = \begin{bmatrix} 3 & 1 \\ 1 & 3 \end{bmatrix}$$

$$det(\boldsymbol{A}) = 9 - 1 = 8$$

$$\lambda_1 * \lambda_2 = 2 * 4 = 8$$

Eigenvalues and Eigenvectors - Properties

For triangular matrices, the eigenvalues are exactly the entries on the diagonal

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

$$\lambda_1 = 1$$
$$\lambda_2 = 2$$

Eigenvalues and Eigenvectors - Properties

For <u>diagonal matrices,</u> the eigenvalues are exactly the entries on the diagonal.

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

$$\lambda_1 = 1$$
$$\lambda_2 = 2$$

Eigenvalues and Eigenvectors - Properties

The eigenvalues of A^2 and A^{-1} are λ^2 and ${}^{1}/_{\lambda}$, respectively, with the same eigenvectors.

Example I

$$A = \begin{bmatrix} 3 & 1 \\ 1 & 3 \end{bmatrix} \qquad \begin{matrix} \lambda_1 = 2 \\ \lambda_2 = 4 \end{matrix}$$

$$A^2 = \begin{bmatrix} 10 & 6 \\ 6 & 10 \end{bmatrix} \qquad \begin{matrix} \lambda_1 = 4 \\ \lambda_2 = 16 \end{matrix}$$

$$A^{-1} = \begin{bmatrix} 0.375 & -0.125 \\ -0.125 & 0.375 \end{bmatrix} \qquad \begin{matrix} \lambda_1 = 0.5 \\ \lambda_2 = 0.25 \end{matrix}$$

$$x_1 = \begin{bmatrix} 1/\sqrt{2} \\ -1/\sqrt{2} \end{bmatrix}$$

$$x_2 = \begin{bmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \end{bmatrix}$$

DIAGONALIZATION:
$A = S\Lambda S^{-1}$

$$\boldsymbol{A}x = \lambda \boldsymbol{x}$$
$$|\boldsymbol{A} - \lambda I| = \boldsymbol{0}$$

Definition: Diagonalization

If A_{nxn} has n linearly independent eigenvectors, we can put those vectors in the columns of a (square, invertible) **Eigenvector Matrix S**, and the eigenvectors in the diagonal of a square **Eigenvalue Matrix Λ**, then

$$\boldsymbol{A} = \boldsymbol{S\Lambda S^{-1}}$$

Notes

$$\boldsymbol{\Lambda} = \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \lambda_n \end{bmatrix} \qquad \boldsymbol{S} = [x_1 \quad \cdots \quad x_n]$$

- Matrix $\boldsymbol{S}$ has an inverse because its columns (the eigenvectors of $\boldsymbol{A}$) are assumed to be linearly independent. **Without n independent eigenvectors , we can't diagonalize A. (i.e.,: Not all matrices are diagonalizable.)**
- A and Λ have the same eigenvalues, but different eigenvectors.
- Diagonalization is helpful to solve system of differential equations.

Example: Diagonalization

$$\boldsymbol{A}x = \lambda\boldsymbol{x}$$
$$|\boldsymbol{A} - \lambda I| = \boldsymbol{0}$$
$$\boldsymbol{A} = \boldsymbol{S\Lambda S^{-1}}$$

$$A = \begin{bmatrix} -5 & 4 & 9 \\ -22 & 14 & 18 \\ 16 & -8 & -6 \end{bmatrix}$$

$$\lambda_1 = -6 \qquad x_1 = \begin{bmatrix} -1 \\ -2 \\ 1 \end{bmatrix}$$

$$\lambda_2 = 3 \qquad x_2 = \begin{bmatrix} 1 \\ 2 \\ 0 \end{bmatrix}$$

$$\lambda_3 = 6 \qquad x_3 = \begin{bmatrix} 2 \\ 1 \\ 2 \end{bmatrix}$$

Check: $Trace(A) \ = -5 + 14 - 6 \ = \ -6 + 3 + 6 = 3$

$$\boldsymbol{\Lambda} = \begin{bmatrix} -6 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 6 \end{bmatrix} \quad \boldsymbol{S} = \begin{bmatrix} -1 & 1 & 2 \\ -2 & 2 & 1 \\ 1 & 0 & 2 \end{bmatrix} \quad \boldsymbol{S}^{-1} = \begin{bmatrix} -4/3 & 2/3 & 1 \\ -5/3 & 4/3 & 1 \\ 2/3 & -1/3 & 0 \end{bmatrix}$$

$$Ax = \lambda x$$
$$|A - \lambda I| = 0$$
$$A = S\Lambda S^{-1}$$

Example: Diagonalization

$$A = \begin{bmatrix} -5 & 4 & 9 \\ -22 & 14 & 18 \\ 16 & -8 & -6 \end{bmatrix}$$

$$\begin{array}{cccccccc} & A & & S & & \Lambda & & S^{-1} \\ & \begin{bmatrix} -5 & 4 & 9 \\ -22 & 14 & 18 \\ 16 & -8 & -6 \end{bmatrix} & = & \begin{bmatrix} -1 & 1 & 2 \\ -2 & 2 & 1 \\ 1 & 0 & 2 \end{bmatrix} & & \begin{bmatrix} -6 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 6 \end{bmatrix} & & \begin{bmatrix} -4/3 & 2/3 & 1 \\ -5/3 & 4/3 & 1 \\ 2/3 & -1/3 & 0 \end{bmatrix} \end{array}$$

Check

Note:

Because the matrix is <u>not symmetric</u>, the eigenvectors are <u>not orthogonal</u> (More about this later)

What Matrices can be Diagonalized?

A nxn matrix $\boldsymbol{A}$ is diagonalizable *<u>if and only if</u>* $\boldsymbol{A}$ has n linearly independent eigenvectors.

In other words, $\boldsymbol{A}$ is diagonalizable <u>if and only if</u> there are enough eigenvectors to form a basis for R^n.

Note:

The eigenvector matrix (i.e., $\boldsymbol{S}$) is <u>not</u> unique.

Any matrix that has <u>no repeated</u> eigenvalues can be diagonalized (This is a *sufficient* condition.)

Diagonalization - k^{th} power of A

$$\boldsymbol{A}x = \lambda\boldsymbol{x}$$
$$|\boldsymbol{A} - \lambda I| = \boldsymbol{0}$$
$$\boldsymbol{A} = \boldsymbol{S}\boldsymbol{\Lambda}\boldsymbol{S}^{-1}$$

Definition - k^{th} power of A_{nxn}

$$\boldsymbol{A}^k = \boldsymbol{S}\boldsymbol{\Lambda}^k\boldsymbol{S}^{-1}$$
$$\boldsymbol{A}^k\boldsymbol{x} = (\boldsymbol{S}\boldsymbol{\Lambda}^k\boldsymbol{S}^{-1})\boldsymbol{x}$$

Notes

- If we can diagonalize a matrix, then we can find its powers easily.
- The eigenvectors of $\mathbf{A}^{\mathbf{k}}$ are the same as those of $\boldsymbol{A}$ (The eigenvalues of $\mathbf{A}^{\mathbf{k}}$ are λ_i^k.)

Example: k^{th} power of $\boldsymbol{A}$

$$\boldsymbol{A}x = \lambda \boldsymbol{x}$$
$$|\boldsymbol{A} - \lambda I| = \boldsymbol{0}$$
$$\boldsymbol{A} = \boldsymbol{S\Lambda S^{-1}}$$
$$\boldsymbol{A^k} = \boldsymbol{S\Lambda^k S^{-1}}$$

Diagonalize the matrix $A = \begin{bmatrix} 0.8 & 0.3 \\ 0.2 & 0.7 \end{bmatrix}$

$$\lambda_1 = 1 \qquad x_1 = \begin{bmatrix} 0.6 \\ 0.4 \end{bmatrix}$$

$$\lambda_2 = 0.5 \qquad x_2 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

Check: $Trace(A) = 0.8 + 0.5 \;\; = 1 + 0.5 = 1.5$

$$\underset{\boldsymbol{A}}{\begin{bmatrix} 0.8 & 0.3 \\ 0.2 & 0.7 \end{bmatrix}} = \underset{\boldsymbol{S}}{\begin{bmatrix} 0.6 & 1 \\ 0.4 & -1 \end{bmatrix}} \underset{\boldsymbol{\Lambda}}{\begin{bmatrix} 1 & 0 \\ 0 & 0.5 \end{bmatrix}} \underset{\boldsymbol{S^{-1}}}{\begin{bmatrix} 1 & 1 \\ 0.4 & -0.6 \end{bmatrix}}$$

$$\underset{\boldsymbol{A^k}}{\begin{bmatrix} 0.8 & 0.3 \\ 0.2 & 0.7 \end{bmatrix}^k} = \underset{\boldsymbol{S}}{\begin{bmatrix} 0.6 & 1 \\ 0.4 & -1 \end{bmatrix}} \underset{\boldsymbol{\Lambda^k}}{\begin{bmatrix} 1^k & 0 \\ 0 & 0.5^k \end{bmatrix}} \underset{\boldsymbol{S^{-1}}}{\begin{bmatrix} 1 & 1 \\ 0.4 & -0.6 \end{bmatrix}}$$

k^{th} power of $\boldsymbol{A}$ as $\mathrm{k} \rightarrow \infty$

$$\boldsymbol{A}x = \lambda\boldsymbol{x}$$
$$|\boldsymbol{A} - \lambda I| = \boldsymbol{0}$$
$$\boldsymbol{A} = \boldsymbol{S\Lambda S^{-1}}$$
$$\boldsymbol{A^k} = \boldsymbol{S\Lambda^k S^{-1}}$$

If A has n independent eigenvectors with eigenvalues λ_i, then

$A^k \rightarrow 0$ as $k \rightarrow \infty$ *if and only if* all $|\lambda_i| < 1$.

From previous example

$\lambda_{1,2} = 1, 0.5$

$$\begin{bmatrix} 0.8 & 0.3 \\ 0.2 & 0.7 \end{bmatrix}^k = \begin{bmatrix} 0.6 & 1 \\ 0.4 & -1 \end{bmatrix} \begin{bmatrix} 1^k & 0 \\ 0 & 0.5^k \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 0.4 & -0.6 \end{bmatrix}$$

$$\begin{bmatrix} 0.8 & 0.3 \\ 0.2 & 0.7 \end{bmatrix}^\infty = \begin{bmatrix} 0.6 & 1 \\ 0.4 & -1 \end{bmatrix} \begin{bmatrix} 1^\infty & 0 \\ 0 & 0.5^\infty \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 0.4 & -0.6 \end{bmatrix}$$

$$\begin{bmatrix} 0.8 & 0.3 \\ 0.2 & 0.7 \end{bmatrix}^\infty = \begin{bmatrix} 0.6 & 1 \\ 0.4 & -1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 0.4 & -0.6 \end{bmatrix}$$

$$\begin{bmatrix} 0.8 & 0.3 \\ 0.2 & 0.7 \end{bmatrix}^\infty = \begin{bmatrix} 0.6 & 0.6 \\ 0.4 & 0.4 \end{bmatrix} = \begin{bmatrix} x_1 & x_1 \end{bmatrix}$$

k^{th} power of $\boldsymbol{A}$ as $\mathrm{k} \rightarrow \infty$

$$\boldsymbol{Ax} = \lambda \boldsymbol{x}$$
$$|\boldsymbol{A} - \lambda \boldsymbol{I}| = \boldsymbol{0}$$
$$\boldsymbol{A} = \boldsymbol{S \Lambda S^{-1}}$$
$$\boldsymbol{A^k} = \boldsymbol{S \Lambda^k S^{-1}}$$

$$\begin{bmatrix} 0.8 & 0.3 \\ 0.2 & 0.7 \end{bmatrix}^{\infty} = \begin{bmatrix} 0.6 & 0.6 \\ 0.4 & 0.4 \end{bmatrix} = \begin{bmatrix} x_1 & x_1 \end{bmatrix}$$

$$\lambda_1 = 1 \qquad x_1 = \begin{bmatrix} 0.6 \\ 0.4 \end{bmatrix}$$

$$\lambda_2 = 0.5 \qquad x_2 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

Because $\lambda_1 = 1$ → x_1 is at steady state (Doesn't change)
Because $\lambda_2 < 1$ → x_2 is at decaying mode (Disappearing)

What about $\lambda > 1$?

This is a *Markov matrix*.

Other goals of Diagonalization

- Decouple **dynamic system** to understand long-term behavior, or evaluation, of a dynamic system.

- Real-life examples:
  - Population Growth
  - Economics
  - Page Ranking for web search engines

- Will the system reach
  - **Stability:** $|\lambda_{max}| = 1,$
  - **Converge:** $|\lambda_{max}| < 1,$
  - **Diverge:** $|\lambda_{max}| > 1$

More about in session #5

DIAGONALIZATION OF SYMMETRIC MATRICES

THE UNIVERSITY OF CHICAGO

Motivation

- If a matrix has some special property, its eigenvalues and eigenvectors are likely to have special properties as well.
- Symmetric matrices are great examples for matrices with special properties.
- Later we will see additional examples of special matrices (e.g., Markov matrix.)

Note:
Special doesn't mean rare

Diagonalization - Symmetric Matrices

For a symmetric matrix with real number entries,

1. The eigenvalues are real numbers and
2. It's possible to choose a complete set of eigenvectors that are orthogonal (or even orthonormal.)

We already saw that if $\boldsymbol{A}$ has n independent eigenvectors we can write

$$\boldsymbol{A} = \boldsymbol{S\Lambda S^{-1}}.$$

If $\boldsymbol{A}$ is symmetric we can write

$$\boldsymbol{A} = \boldsymbol{Q\Lambda Q^{-1}} = \boldsymbol{Q\Lambda Q^{T}}$$

where $\boldsymbol{Q}$ is an orthogonal matrix.

Diagonalization - Symmetric Matrices

Spectral Theorem

If $\boldsymbol{A}$ is symmetric, then there is an orthogonal matrix Q and a diagonal matrix for which

$$\boldsymbol{A} = \boldsymbol{Q}\boldsymbol{\Lambda}\boldsymbol{Q}^{\boldsymbol{T}}$$

What does this mean?

Eigenvectors of a real symmetric matrix (when they correspond to different λ's) are always perpendicular / orthogonal.

What if the eigenvalues are repeated?

This will *never* happen for symmetric matrices. There are always enough eigenvectors to diagonalize $\boldsymbol{A}$.

Example: Diagonalization of Symmetric Matrices

$$\boldsymbol{A} = \begin{bmatrix} 1 & 2 \\ 2 & 4 \end{bmatrix}$$

$$\boldsymbol{Ax} = \lambda \boldsymbol{x}$$
$$|\boldsymbol{A} - \lambda \boldsymbol{I}| = \boldsymbol{0}$$
$$\boldsymbol{A} = \boldsymbol{S\Lambda S^{-1}}$$
$$\boldsymbol{A^k} = \boldsymbol{S\Lambda^k S^{-1}}$$
$$\boldsymbol{A} = \boldsymbol{Q\Lambda Q^T}\ (\boldsymbol{if}\ \boldsymbol{A} = \boldsymbol{A^T})$$

$$\lambda_1 = 0 \qquad x_1 = \begin{bmatrix} \frac{2}{\sqrt{5}} \\ \frac{-1}{\sqrt{5}} \end{bmatrix}$$

$$\lambda_2 = 5 \qquad x_2 = \begin{bmatrix} \frac{1}{\sqrt{5}} \\ \frac{2}{\sqrt{5}} \end{bmatrix}$$

Check

$$x_1 \cdot x_2 = \frac{2}{\sqrt{5}} * \frac{1}{\sqrt{5}} - \frac{1}{\sqrt{5}} * \frac{2}{\sqrt{5}} = 0$$

$$\begin{array}{cccc} \boldsymbol{A} & \boldsymbol{Q} & \boldsymbol{\Lambda} & \boldsymbol{Q^T} \\ \begin{bmatrix} 1 & 2 \\ 2 & 4 \end{bmatrix} = & \begin{bmatrix} \frac{2}{\sqrt{5}} & \frac{1}{\sqrt{5}} \\ \frac{-1}{\sqrt{5}} & \frac{2}{\sqrt{5}} \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ 0 & 5 \end{bmatrix} & \begin{bmatrix} \frac{2}{\sqrt{5}} & \frac{-1}{\sqrt{5}} \\ \frac{1}{\sqrt{5}} & \frac{2}{\sqrt{5}} \end{bmatrix} \end{array}$$

BREAK

POSITIVE DEFINITE AND SEMIDEFINITE MATRICES

Motivation: Positive Definite Matrices

- Having symmetric matrices was not special enough. We want to define more special matrices where all $\lambda > 0$.
- The signs of the eigenvalues are important in many applications (e.g., solving systems of differential equations, Hessian matrix)
- How to find if $\lambda > 0$?
  - We want to avoid actually calculating the eigenvalues (**Too computationally expensive.**)

Again…
Special doesn't mean rare

Definition: Positive Definite Matrices

Definition

Positive Definite matrix is a symmetric matrix $\boldsymbol{A}$ for which all eigenvalues are positive (all $\lambda > 0$).

Notes:

- A bad way to tell if a matrix is positive definite is to check that all its eigenvalues are positive.
- For very large matrices ***A***, it's impractical to compute eigenvalues by solving $|\boldsymbol{A} - \lambda I| = \mathbf{0}$. However, it's not hard to compute the pivots, and the signs of the pivots of a symmetric matrix are the same as the signs of the eigenvalues.

Properties: Positive Definite Matrices

If a symmetric matrix $\boldsymbol{S}$ has one of these five properties, it has them all:

(a) All $\boldsymbol{n}$ **pivots** of $\boldsymbol{S}$ are positive.

(b) All $\boldsymbol{n}$ **eigenvalues** of $\boldsymbol{S}$ are positive.

(c) All $\boldsymbol{n}$ **upper left sub-determinants** (a.k.a. upper left determinants) are positive. (This is Sylvester's criterion.)

(d) The quadratic form $\boldsymbol{x\Lambda x^T}$ is positive for every x (except at $x = 0$).

(e) $\boldsymbol{S}$ equals $\boldsymbol{A^T A}$ for a matrix $\boldsymbol{A}$ where $\boldsymbol{A}$ has **independent columns.**

Example: Positive Definite Matrices

Given

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

Eigenvalues ➡ $\lambda_{1,2,3} = 1,1,4$

```
In [29]: LA.eig(np.eye(3)+np.ones((3,3)))[0].real
Out[29]: array([ 1.,  4.,  1.])
```

Pivots ➡ $2, 3/2, 4/3$

```
In [38]: LA.lu(A)[2]
Out[38]:
array([[ 2.        ,  1.        ,  1.        ],
       [ 0.        ,  1.5       ,  0.5       ],
       [ 0.        ,  0.        ,  1.33333333]])
```

Any one test is decisive

Definition: Positive Semidefinite Matrices

Definition

Positive semidefinite matrix is a symmetric matrix $\boldsymbol{A}$ for which all eigenvalues are all $\lambda \geq 0$.

Recall:

- Vectors with eigenvalue 0 make up the nullspace of A (i.e., Fills the nullspace.)
- If $\boldsymbol{A}$ is singular, then $\lambda = 0$ is an eigenvalue of $\boldsymbol{A}$.

Motivation:

Singular Value Decomposition

- Balance, equilibrium and steady state
- Solved by elimination

(LSE, System of Equations, etc.)

$$\boldsymbol{A}_{nxn}x = b$$

Linear System of equations

$$\boldsymbol{A}_{mxn}x = b$$

Least Squares

- Change/ transient state
- **Not** solved by elimination

(PDE, ODE, Dynamic Systems, etc.)

$$\boldsymbol{A}_{nxn}x = \lambda x$$

Eigenvalues

$$\boldsymbol{A}_{mxn}v = \sigma u$$

Singular Values

Motivation:

Singular Value Decomposition

Previously

- We found the eigenvalues of square matrix A_{nxn}.
- We also defined **Matrix Diagonalizations:**

$$A = S\Lambda S^{-1}$$

$$A = Q\Lambda Q^{-1} = Q\Lambda Q^{T} \text{ (For symmetric } A\text{)}$$

Next goal

- We will extend the Diagonalization $A = S\Lambda S^{-1}$ to any A_{mxn} matrix with rank r.

Challenges:

- Eigenvalues are not defined for rectangular matrices

Motivation:

Singular Value Decomposition

Singular Value Decomposition

- The Singular Value Decomposition (SVD) is used for non-square matrices.
- This is one of the most useful matrix factorization in applied linear algebra and is the most general form of diagonalization.

We will find the singular values of rectangular matrix A_{mxn} which are the eigenvalues of

1) Square,
2) Symmetric,
3) and Positively Definite

matrices A^TA and AA^T

Motivation:

Singular Value Decomposition

The price of SVD

Compared to $A = S\Lambda S^{-1}$ where we had only one eigenvector matrix, for a rectangular matrix we have two sets of orthogonal matrices.

Singular Values

Recall: Eigenvalues and Eigenvectors of Symmetric Matrices

For a symmetric matrix with real number entries,

1. The eigenvalues are real numbers and
2. It's possible to choose a complete set of eigenvectors that are orthogonal (or even orthonormal.)

Observation:

Let A be a mxn matrix. Then AA^T is symmetric and therefore can be orthogonally diagonalized.

Definition:
Singular Values

Definition

Let $\{v_1, \dots, v_n\}$ be orthonormal basis for R^n consisting of eigenvectors of $\boldsymbol{AA^T}$, and let $\lambda_1, \dots, \lambda_n$ be associated eigenvalues of $\boldsymbol{AA^T}$, Then:

The **singular values** of $\boldsymbol{A}$ are the square roots of the eigenvalues of $\boldsymbol{AA^T}$, denoted by $\sigma_1, \dots, \sigma_n$ (That is $\sigma_i = \sqrt{\lambda_i}$)

Assumption

Arrange by eigenvalue/singular value size

$\lambda_1 \geq \cdots \geq \lambda_n \geq 0$ Or $\sigma_1 \geq \cdots \geq \sigma_n \geq 0$

Example: Singular Values

$$A = \begin{bmatrix} 4 & 11 & 14 \\ 8 & 7 & -2 \end{bmatrix}$$

Symmetric

$$A^T A = \begin{bmatrix} 4 & 8 \\ 11 & 7 \\ 14 & -2 \end{bmatrix} \begin{bmatrix} 4 & 11 & 14 \\ 8 & 7 & -2 \end{bmatrix} = \begin{bmatrix} 80 & 100 & 40 \\ 100 & 170 & 140 \\ 40 & 140 & 200 \end{bmatrix}$$

Eigenvalues of $A^T A$

$$\lambda_1 = 360$$
$$\lambda_2 = 90$$
$$\lambda_3 = 0$$

Singular values of A

$$\sigma_1 = \sqrt{360}$$
$$\sigma_2 = \sqrt{90}$$
$$\sigma_3 = 0$$

Singular Value Decomposition (SVD)

THE UNIVERSITY OF CHICAGO

Motivation:

$$A_{mxn} v_{nx1} = \sigma u_{mx1}$$

Singular Value Decomposition

We can think of A as a linear transformation taking a vector v_i in its row space to a vector $u_i = Av_i$ in its column space.

$$u_{mx1} = A_{mxn} \cdot v_{nx1}$$

The vector u is a linear combination of the columns of A.

Motivation:

Singular Value Decomposition

The SVD arises from finding an orthogonal basis for the row space that gets transformed into an orthogonal basis for the column space:

$$\boldsymbol{Av_i = \sigma_i u_i}$$

u_i $-$ *Column Space*
v_i $-$ *Row Space*

Where σ_i is a scaling factor.

It's not hard to find an orthogonal basis for the row space (e.g., using Gram-Schmidt process). But in general, there's no reason to expect $\boldsymbol{A}$ to transform that basis to another orthogonal basis.

Definition:

Singular Value Decomposition

$$\boldsymbol{A} = \boldsymbol{U\Sigma V^T}$$

Orthonormal basis for the row space of $\boldsymbol{A}$

Orthonormal basis for the column space of $\boldsymbol{A}$

$$\boldsymbol{A}_{mxn} = \boldsymbol{U\Sigma V^T} = [\boldsymbol{u_1} \quad \cdots \quad \boldsymbol{u_r}] \begin{bmatrix} \sigma_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \sigma_r \end{bmatrix} \begin{bmatrix} \boldsymbol{v_1^T} \\ \vdots \\ \boldsymbol{v_r^T} \end{bmatrix}$$

$$\boxed{\boldsymbol{A}_{mxn} = \sum_{i=1}^{r} \sigma_i \boldsymbol{u_i v_i^T} = \sigma_1 \boldsymbol{u_1 v_1^T} + \cdots + \sigma_r \boldsymbol{u_r v_r^T}}$$

Each $\sigma_i \boldsymbol{u_i v_i^T}$ is a mxn rank-1 matrix

Any matrix is the sum of r matrices of rank 1

Definition:

Singular Value Decomposition

Let $\boldsymbol{A}$ be an mxn matrix with rank r. Then there exists an mxn matrix $\boldsymbol{\Sigma}$ for which the diagonal entries in $\boldsymbol{D}$ are the first r singular values of $\boldsymbol{A}$ $(\sigma_1 \geq \cdots \geq \sigma_n \geq 0)$, and there exist mxm orthogonal matrix $\boldsymbol{U}$ and nxn orthogonal matrix $\boldsymbol{V}$ such that

$$\boldsymbol{A_{mxn} = U_{mxm} \Sigma_{mxn} V_{nxn}^{T}}$$

Definition:

Singular Value Decomposition

Recall: A_{mxn} with rank r

$$A_{mxn} = U_{mxm}\Sigma_{mxn}V^T_{nxn}$$

Full rank decomposition

$$\Sigma = \begin{bmatrix} D_{rxr} & 0 \\ 0 & 0 \end{bmatrix}$$

m-r rows

n-r columns

$$A_{mxn} = U_{mxr}\Sigma_{rxr}V^T_{rxn}$$

Reduced rank decomposition (Truncated SVD)

$$AV = U\Sigma$$
$$A_{mxn} = U_{mxm}\Sigma_{mxn}V^T_{nxn}$$
$$A = \sum_{i=1}^{r} \sigma_i u_i v_i^T$$

$A^T A$ and AA^T

Recall: $A = \sum_{i=1}^{r} \sigma_i u_i v_i^T$

The SVD separates any matrix A into rank one pieces of

$$uv^T = \begin{bmatrix} \vdots \\ columns \\ \vdots \end{bmatrix} \begin{pmatrix} \cdots & rows & \cdots \end{pmatrix}$$

Where,

The columns are the eigenvectors of the symmetric matrix AA^T

and

The rows are the eigenvectors of the symmetric matrix $A^T A$

$A^T A$ and AA^T are not equal; however, they have the same positive eigenvalues.

U and V

$$AV = U\Sigma$$
$$A_{mxn} = U_{mxm}\Sigma_{mxn}V_{nxn}^T$$
$$A = \sum_{i=1}^{r} \sigma_i u_i v_i^T$$

Recall: A_{mxn} with rank r

Columns of matrix U are called <u>Left Singular Vectors</u> (i.e., Eigenvectors of AA^T)

Columns of matrix V are called <u>Right Singular Vectors</u> (i.e., Eigenvectors of A^TA)

The price of SVD, compared to $A = S\Lambda S^{-1}$, is that we have two sets of singular vectors (i.e., U and V)

U is in R^m **Column Space**

V is in R^n **Row Space**

$$AV = U\Sigma$$
$$A_{mxn} = U_{mxm}\Sigma_{mxn}V^T_{nxn}$$
$$A = \sum_{i=1}^{r} \sigma_i u_i v_i^T$$

U and V

Usually, $U \neq V$

U and V are not unique

Since V *and* U are orthonormal,

$$V^{-1} = V^T$$
$$U^{-1} = U^T$$

$$V^T V = I_{nxn}$$
$$U^T U = I_{mxm}$$

$$AV = U\Sigma$$
$$A_{mxn} = U_{mxm}\Sigma_{mxn}V^T_{nxn}$$
$$A = \sum_{i=1}^{r} \sigma_i u_i v_i^T$$

U and V

If A is a symmetric (i.e., square) positive definite, its eigenvectors are orthogonal, and we can write

$$A = Q\Lambda Q^T.$$

This is a special case of a SVD, with $U = V = Q$.

(The *same* basis for its row and column space!)

SVD - EXAMPLES

SVD – Solution Steps

$$\boldsymbol{A}_{mxn} v_{nx1} = \sigma u_{mx1}$$
$$\boldsymbol{A} v_i = \sigma_i u_i$$

Step 1: Find an orthogonal diagonalization of $\boldsymbol{A}^T\boldsymbol{A}$ (i.e., Find the eigenvalues and orthogonal eigenvectors of $\boldsymbol{A}^T\boldsymbol{A}$.)

Step 2: Set up $\boldsymbol{\Sigma}$ and $\boldsymbol{V}$ (i.e., arrange the eigenvalues of $\boldsymbol{A}^T\boldsymbol{A}$ in a decreasing order.)

Step 3: Construct $\boldsymbol{U}$ (i.e., Calculate $u_i = \frac{\boldsymbol{A}v_i}{\sigma_i}$)

$$A_{mxn} = U_{mxr}\Sigma_{rxr}V^T_{rxn}$$

Example: SVD of Square Matrix

$$A = \begin{bmatrix} 4 & 4 \\ -3 & 3 \end{bmatrix}$$

Note:
A has a rank = 2

Step1: Find an orthogonal diagonalization of $A^T A$

symmetric & positive definite

$$A^T A = \begin{bmatrix} 4 & -3 \\ 4 & 3 \end{bmatrix}\begin{bmatrix} 4 & 4 \\ -3 & 3 \end{bmatrix} = \begin{bmatrix} 25 & 7 \\ 7 & 25 \end{bmatrix}$$

$$\sigma_1^2 = \lambda_1 = 32$$

$$v_1 = \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix}$$

$$\sigma_2^2 = \lambda_2 = 18$$

$$v_2 = \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{-1}{\sqrt{2}} \end{bmatrix}$$

$A_{mxn} = U_{mxr}\Sigma_{rxr}V^T_{rxn}$

Example: SVD of Square Matrix

Step 2: Set up $\boldsymbol{\Sigma}$ and $\boldsymbol{V}$

$$\underset{\boldsymbol{A}}{\begin{bmatrix} 4 & 4 \\ -3 & 3 \end{bmatrix}} = \underset{\boldsymbol{U}}{\begin{bmatrix} & \end{bmatrix}} \underset{\boldsymbol{\Sigma}}{\begin{bmatrix} \sqrt{32} & 0 \\ 0 & \sqrt{18} \end{bmatrix}} \underset{\boldsymbol{V^T}}{\begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{bmatrix}}$$

Step 3: Construct $\boldsymbol{U}$ (i.e., Calculate $u_i = \frac{Av_i}{\sigma_i}$)

$$u_1 = \frac{Av_1}{\sigma_1} = \frac{1}{\sqrt{32}}\begin{bmatrix} 4 & 4 \\ -3 & 3 \end{bmatrix}\begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \qquad u_2 = \frac{Av_2}{\sigma_2} = \frac{1}{\sqrt{18}}\begin{bmatrix} 4 & 4 \\ -3 & 3 \end{bmatrix}\begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{-1}{\sqrt{2}} \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$$A_{mxn} = U_{mxr}\Sigma_{rxr}V^T_{rxn}$$

Example: SVD of Square Matrix

Alternatively: Find an orthogonal diagonalization of AA^T

$$AA^T = \begin{bmatrix} 4 & 4 \\ -3 & 3 \end{bmatrix}\begin{bmatrix} 4 & -3 \\ 4 & 3 \end{bmatrix} = \begin{bmatrix} 32 & 0 \\ 0 & 18 \end{bmatrix}$$

$$\sigma_1^2 = \lambda_1 = 32$$
$$u_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

$$\sigma_2^2 = \lambda_2 = 18$$
$$u_2 = \begin{bmatrix} 0 \\ -1 \end{bmatrix}$$

Note:
A has a rank = 2

$$\begin{array}{cccc} \boldsymbol{A} & \boldsymbol{U} & \boldsymbol{\Sigma} & \boldsymbol{V^T} \end{array}$$

$$\begin{bmatrix} 4 & 4 \\ -3 & 3 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}\begin{bmatrix} \sqrt{32} & 0 \\ 0 & \sqrt{18} \end{bmatrix}\begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{bmatrix}$$

Example: SVD of Square Matrix (Python)

scipy.linalg.svd

scipy.linalg.svd(*a*, *full_matrices=True*, *compute_uv=True*, *overwrite_a=False*, *check_finite=True*, *lapack_driver='gesdd'*) [source]

Singular Value Decomposition.

Factorizes the matrix a into two unitary matrices U and Vh, and a 1-D array s of singular values (real, non-negative) such that `a == U*S*Vh`, where S is a suitably shaped matrix of zeros with main diagonal s.

```
from scipy import linalg as la
import numpy as np

A = np.array([[4,4],
              [-3,3]])
U, S, Vt = la.svd(A)
```

```
In [2]: U
Out[2]:
array([[-1.,  0.],
       [ 0.,  1.]])

In [3]: Vt
Out[3]:
array([[-0.70710678, -0.70710678],
       [-0.70710678,  0.70710678]])

In [4]: S**2
Out[4]: array([ 32.,  18.])
```

$$\begin{bmatrix} 4 & 4 \\ -3 & 3 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} \sqrt{32} & 0 \\ 0 & \sqrt{18} \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{bmatrix}$$

haddy

Note: Set the 'full_matrices' = False to get the $A_{mxn} = U_{mxr}\Sigma_{rxr}V_{rxn}^T$ Shape

$$A_{mxn} = U_{mxr}\Sigma_{rxr}V^T_{rxn}$$

Example: SVD of Square Matrix

$$\underset{\boldsymbol{A}}{\begin{bmatrix} 4 & 4 \\ -3 & 3 \end{bmatrix}} = \underset{\boldsymbol{U}}{\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}} \underset{\boldsymbol{\Sigma}}{\begin{bmatrix} \sqrt{32} & 0 \\ 0 & \sqrt{18} \end{bmatrix}} \underset{\boldsymbol{V^T}}{\begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{bmatrix}}$$

$$\boxed{\boldsymbol{A} = \sigma_1 \boldsymbol{u_1 v_1^T} + \sigma_2 \boldsymbol{u_2 v_2^T}}$$

$$\sigma_1 \boldsymbol{u_1 v_1^T} = \sqrt{32}\begin{bmatrix} 1 \\ 0 \end{bmatrix}\begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} = \begin{bmatrix} 4 & 4 \\ 0 & 0 \end{bmatrix}$$

rank-1 matrix

$$\sigma_2 \boldsymbol{u_2 v_2^T} = \sqrt{18}\begin{bmatrix} 0 \\ -1 \end{bmatrix}\begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ -3 & 3 \end{bmatrix}$$

rank-1 matrix

$$\begin{bmatrix} 4 & 4 \\ -3 & 3 \end{bmatrix} = \begin{bmatrix} 4 & 4 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ -3 & 3 \end{bmatrix}$$

What is Unique about SVD?

SVD – Simple Pieces

Rank = 1

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

Full | Low Rank

$m = 6; n = 6 \quad m * n = 36; \qquad p = 1\,; \quad p(m+n) = 12$

Rank = 2

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

Full | Low Rank

SVD – Simple Pieces

$$AV = U\Sigma$$
$$A = U\Sigma V^T$$
$$A = \sum_{i=1}^{r} \sigma_i u_i v_i^T$$

Recall that

$$A = \sum_{i=1}^{r} \sigma_i u_i v_i^T = \sigma_1 u_1 v_1^T + \cdots + \sigma_r u_r v_r^T$$

Where each $\sigma_1 u_1 v_1^T$ is a rank-1 matrix

$$A = \begin{array}{c} \\ u_1 \end{array}\begin{array}{c} v_1^T \\ \\ \end{array} + \begin{array}{c} \\ u_2 \end{array}\begin{array}{c} v_2^T \\ \\ \end{array} + \begin{array}{c} \\ u_3 \end{array}\begin{array}{c} v_3^T \\ \\ \end{array} + \cdots$$

Each piece is a column vector times a row vector.

SVD – Information Ordering

Recall 1:

The **singular values** of A are the square roots of the eigenvalues of AA^T, denoted by $\sigma_1, \dots, \sigma_n$ (That is $\sigma_i = \sqrt{\lambda_i}$)

Where the eigenvalues / singular values can be arranged by size

$$\lambda_1 \geq \cdots \geq \lambda_n \geq 0 \text{ Or } \sigma_1 \geq \cdots \geq \sigma_n \geq 0$$

Recall 2:

The Singular Value Decomposition is based on the property that the absolute values of the eigenvalues of a symmetric matrix $A^T A$ measure the amount that $\boldsymbol{A}$ stretches or shrinks certain vectors (i.e., eigenvectors.)

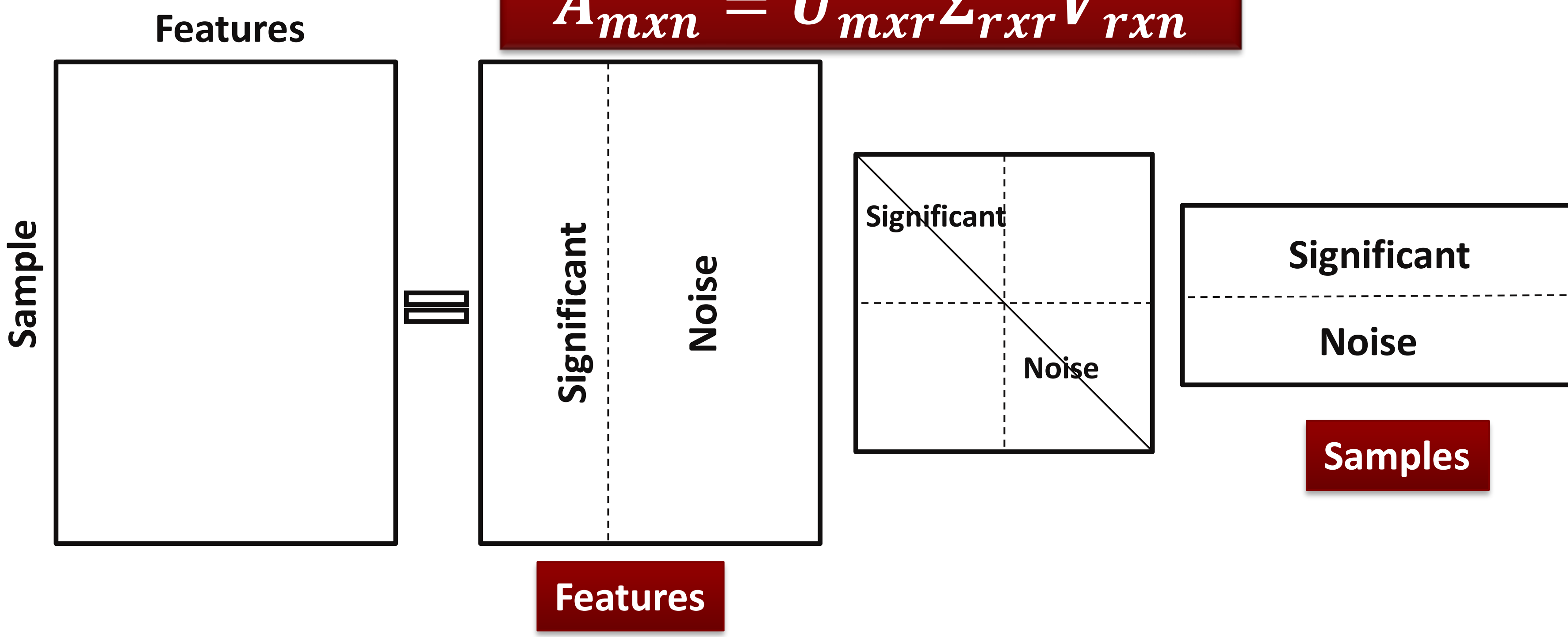
If λ_1 is the eigenvalues with the greatest magnitude, then a corresponding unit eigenvector v_1 identifies a direction in which the stretch effect of $\boldsymbol{A}$ is greatest.

SVD supplies information about the matrix such that its "orders" the information content in the matrix so the "dominating part" becomes visible.

SVD – Simple Pieces and Information Ordering

$$A_{mxn} = U_{mxm}\Sigma_{mxn}V_{nxn}^{T}$$

$$A_{mxn} = U_{mxr}\Sigma_{rxr}V_{rxn}^{T}$$



Can be used for noise reduction and low-rank approximation

$$AV = U\Sigma$$
$$A = U\Sigma V^T$$
$$A = \sum_{i=1}^{r} \sigma_i u_i v_i^T$$

SVD – Simple Pieces

Recall that

$$A = \sum_{i=1}^{r} \sigma_i u_i v_i^T = \sigma_1 u_1 v_1^T + \cdots + \sigma_r u_r v_r^T$$

Where each $\sigma_1 u_1 v_1^T$ is a rank-1 matrix

So A_{mxn} can be rebuilt from $m + n$ entries rather than $m * n$ for matrix A.

Assuming q is the number of components used to rebuild matrix A, then for big values of m and n, the sum $q(m + n + 1)$ is still much smaller than $m * n$, **so less storage is required,** and the matrix can be processed with extreme speed

$m = 100; n = 100 \quad m * n = 10K; \qquad p = 10\,; \quad p(m + n + 1) = 2K$

The SVD separate any matrix into simple pieces

L_2-norm of a Matrix

The L_2-norm of a matrix is given by

$$\|A\|_2 = \sigma_{max}(\boldsymbol{A})$$

$$\|A\|_2 = \textit{Largest singular value}$$

Example

$$A = \begin{bmatrix} 4 & 11 & 14 \\ 8 & 7 & -2 \end{bmatrix}$$

$$\sigma_1^2 = \lambda_1 = 360$$
$$\sigma_2^2 = \lambda_2 = 90$$
$$\sigma_3^2 = \lambda_3 = 0$$

```
from scipy import linalg as la
import numpy as np

A = np.array([[4,11,14],
              [8,7, -2]])

print('Singular Values of Matrix A: {}\n'.format( la.svdvals(A)) )
print('L2 Matrix of Matrix A: {}\n'.format( la.norm(A,2)) )
```

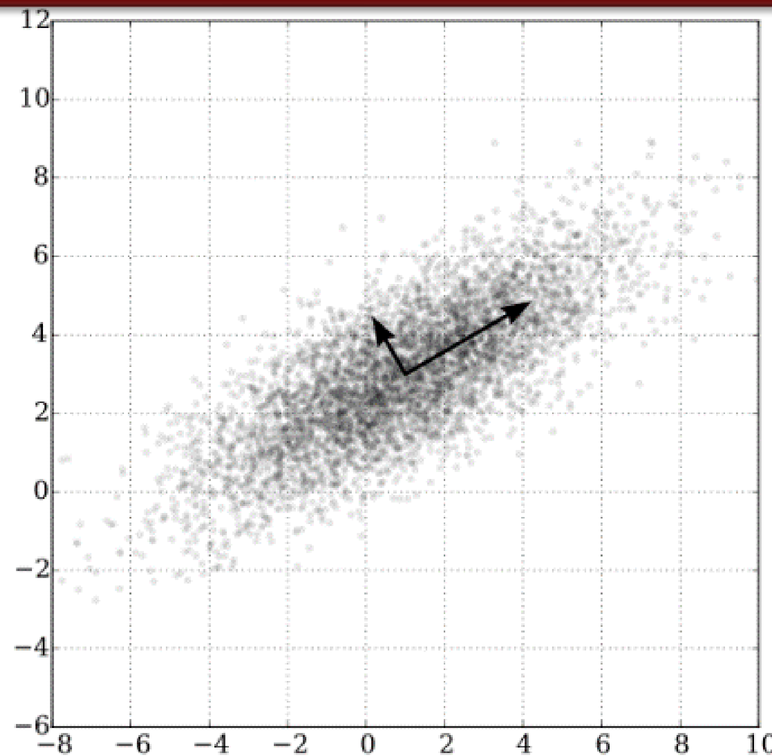
```
Singular Values of Matrix A: [18.97366596  9.48683298]

L2 Matrix of Matrix A: 18.973665961010276
```

Break

PRINCIPLE COMPONENT ANALYSIS (PCA)

Motivation –
Principle Component Analysis

Find the most dominant **linearly uncorrelated** direction of the scatter plot.



The SVD of the centered scatter data can show the dominant directions in the scatter plot.

PCA - Introduction

Assume A_{mxn} is a **Data Matrix** where each row is an attribute with mean zero

- PCA is equivalent to an SVD analysis of the data matrix, once the mean of each attribute has been removed.
- PCA is an example of a connection between statistics and the linear algebra of **positive definite matrices** and **SVD.**
- PCA can be applied to any data consisting of lists of measurements.
- PCA gives a way to understand a data plot in dimension $m = \; the\ number\ of\ measured\ variables$(features)

Definition: Data Matrix

$$A_{mxn} = A_{Attributes\ x\ Samples} = A_{Course\ x\ Student}$$

	Student 1	**Student 2**	**Student 3**	**Student 4**	**Student 5**	**Student 6**
Math	93	86	97	91	86	87
History	92	79	93	84	84	78

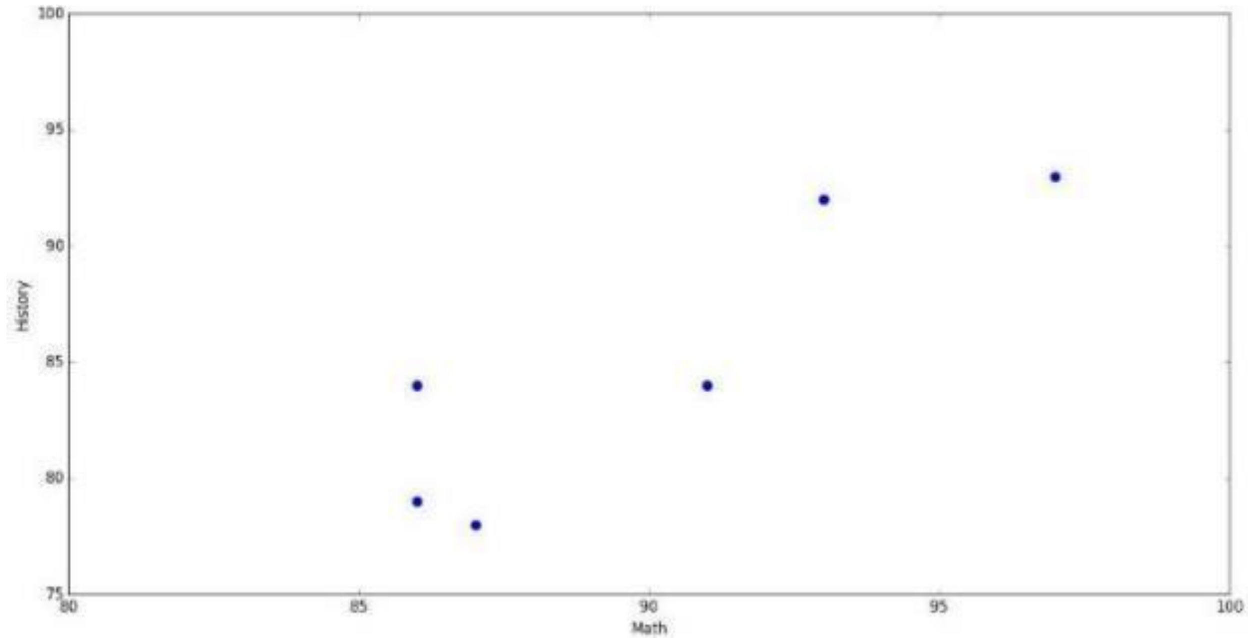
$$\boxed{m = 2; \quad n = 6}$$

$$\boldsymbol{A}_{2x6} = \begin{bmatrix} 93 & 86 & 97 & 91 & 86 & 87 \\ 92 & 79 & 93 & 84 & 84 & 78 \end{bmatrix}$$

```
A=np.array([[93,86,97,91,86,87],[92,79,93,84,84,78]])
```

Data Matrix

$$A_{2x6} = \begin{bmatrix} 93 & 86 & 97 & 91 & 86 & 87 \\ 92 & 79 & 93 & 84 & 84 & 78 \end{bmatrix}$$



Goal:

Find a new set of attributes that better capture the variability (scatteredness) of the data.

The new attributes need to be:

1) Uncorrelated
2) Arranged in order of decreasing variance

$$\boldsymbol{A}_{2x6} = \begin{bmatrix} 93 & 86 & 97 & 91 & 86 & 87 \\ 92 & 79 & 93 & 84 & 84 & 78 \end{bmatrix}$$

Definition: Sample Mean

Definition

The **Sample Mean** of variable i is defined as

$$\mu_i = \frac{X_1 + \cdots + X_n}{n}$$

```
A.mean(axis=1)
array([ 90.,  85.])
```

$$\mu_{Math} = \frac{93 + 86 + \cdots + 87}{6} = 90$$

Center the data

Subtract each mean μ_i from row i to center the data.

(The sample average along each row will become zero.)

```
A= A-A.mean(axis=1,keepdims=True)
```

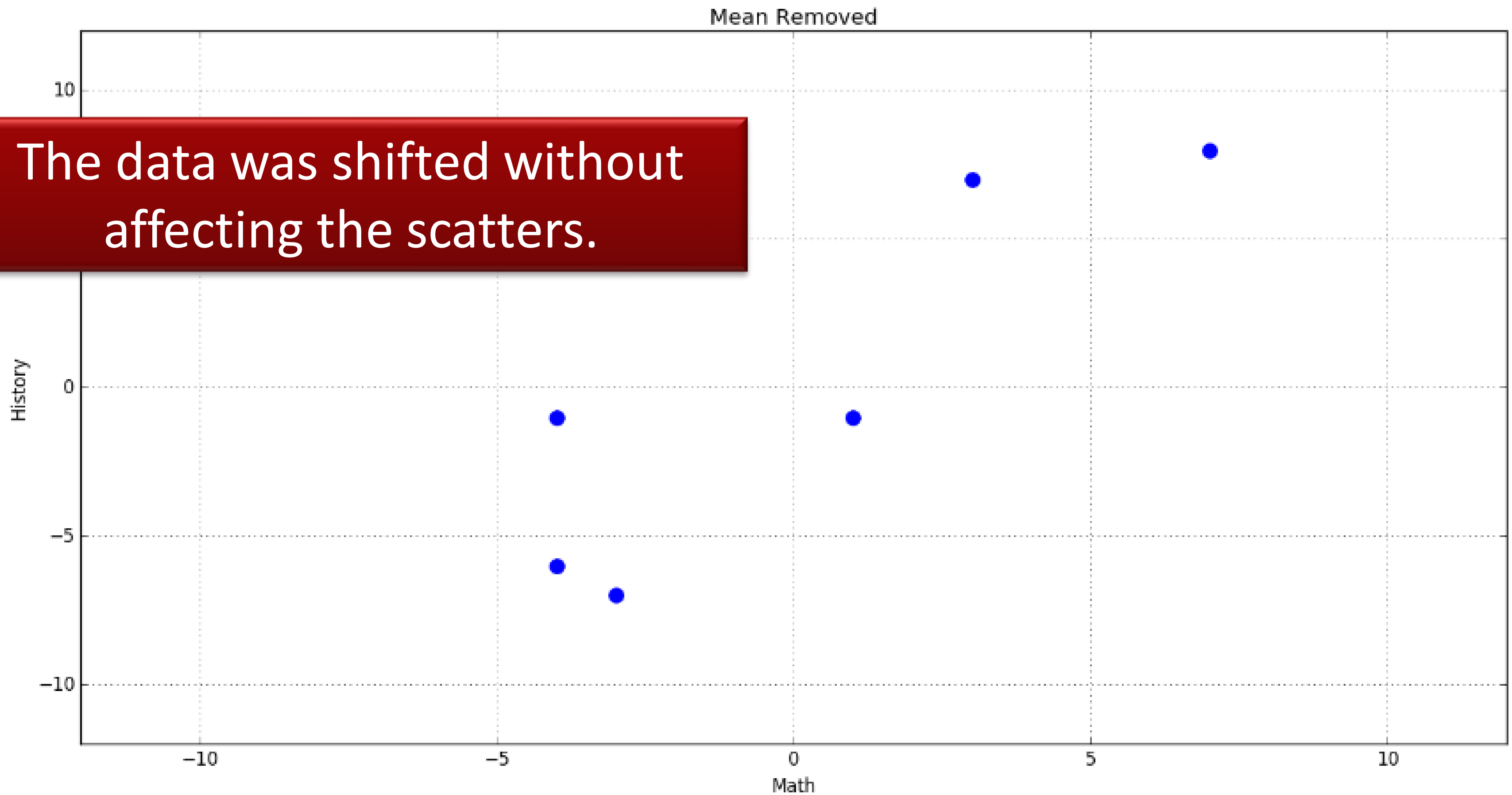
$$A_{2x6} = \begin{bmatrix} 3 & -4 & 7 & 1 & -4 & -3 \\ 7 & -6 & 8 & -1 & -1 & -7 \end{bmatrix}$$

Row - Course

Columns - Student

$$A_{2x6} = \begin{bmatrix} 93 & 86 & 97 & 91 & 86 & 87 \\ 92 & 79 & 93 & 84 & 84 & 78 \end{bmatrix}$$

Sample Mean



The data was shifted without affecting the scatters.

Row - Course
Columns - Student

$$A_{2x6} = \begin{bmatrix} 3 & -4 & 7 & 1 & -4 & -3 \\ 7 & -6 & -1 & 8 & -1 & -7 \end{bmatrix}$$

$$A_{2x6} = \begin{bmatrix} 3 & -4 & 7 & 1 & -4 & -3 \\ 7 & -6 & -1 & 8 & -1 & -7 \end{bmatrix} \qquad \boldsymbol{A}_{2x6} = \begin{bmatrix} 93 & 86 & 97 & 91 & 86 & 87 \\ 92 & 79 & 93 & 84 & 84 & 78 \end{bmatrix}$$

Definition: Sample Covariance Matrix

Definition

The **Sample Covariance Matrix** is a measure of how strong the attributes vary together. It is defined as

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

```
S = A.dot(A.T)/5
```

$$S = \frac{AA^T}{5} = \begin{bmatrix} 20 & 25 \\ 25 & 40 \end{bmatrix}$$

$$Correlation\ matrix = \begin{bmatrix} 1 & 0.88 \\ 0.88 & 1 \end{bmatrix}$$

Note:

1)The two variables (i.e., exam grades) are highly correlated. Above average math grades went with above average history grades.

2) $\boldsymbol{S}$ has positive trace and determinant; therefore, it is <u>semipositive definite</u>.

$$AV = U\Sigma$$
$$A = U\Sigma V^T$$
$$A = \sum_{i=1}^{r} \sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^T$$

Singular Values of $\boldsymbol{A}$

$$S = \frac{AA^T}{5} = \begin{bmatrix} 20 & 25 \\ 25 & 40 \end{bmatrix}$$

$$\sigma_1 = \sim\sqrt{57} \qquad \sigma_2 = \sim\sqrt{3}$$

$$\boldsymbol{A} = \sigma_1 \boldsymbol{u}_1 \boldsymbol{v}_1^T + \sigma_2 \boldsymbol{u}_2 \boldsymbol{v}_2^T$$
$$= \sqrt{57}\boldsymbol{u}_1 \boldsymbol{v}_1^T + \sqrt{3}\boldsymbol{u}_2 \boldsymbol{v}_2^T$$

Orthonormal basis for the column space of $\boldsymbol{A}$

$$\boldsymbol{U} = \begin{bmatrix} 0.56 & 0.82 \\ 0.82 & -0.56 \end{bmatrix}$$

1st principal component
2nd principal component

$$\boldsymbol{\Sigma} = \begin{bmatrix} \sqrt{57} & 0 \\ 0 & \sqrt{3} \end{bmatrix}$$

Definition: Total Variance

Definition

The **Total Variance** in the data is the sum of all eigenvalues of sample variances, which is also equal to the trace of the covariance matrix $\boldsymbol{S}$.

$$T = \sigma_1^2 + \cdots + \sigma_m^2 = Trace(\boldsymbol{S})$$

$$S = \frac{AA^T}{5} = \begin{bmatrix} 20 & 25 \\ 25 & 40 \end{bmatrix}$$

$$\boldsymbol{\Sigma} = \begin{bmatrix} \sqrt{57} & 0 \\ 0 & \sqrt{3} \end{bmatrix}$$

$$T = 57 + 3 = 20 + 40 = 60$$

PCA does not change the total variance of the data.

$$A_{2x6} = \begin{bmatrix} 3 & -4 & 7 & 1 & -4 & -3 \\ 7 & -6 & -1 & 8 & -1 & -7 \end{bmatrix} \qquad A_{2x6} = \begin{bmatrix} 93 & 86 & 97 & 91 & 86 & 87 \\ 92 & 79 & 93 & 84 & 84 & 78 \end{bmatrix}$$

What Does this Mean?

$$\boldsymbol{U} = \begin{bmatrix} 0.56 & 0.82 \\ 0.82 & -0.56 \end{bmatrix} = [u_1 \quad u_2] \qquad \boldsymbol{\Sigma} = \begin{bmatrix} \sqrt{57} & 0 \\ 0 & \sqrt{3} \end{bmatrix} \qquad S = \begin{bmatrix} 20 & 25 \\ 25 & 40 \end{bmatrix} \qquad T = 60$$

The 1st principal component, u_1, of $\boldsymbol{S}$ points in the most significant direction of the data. That direction accounts for (or explains) a fraction σ_1^2/T of the total variance.

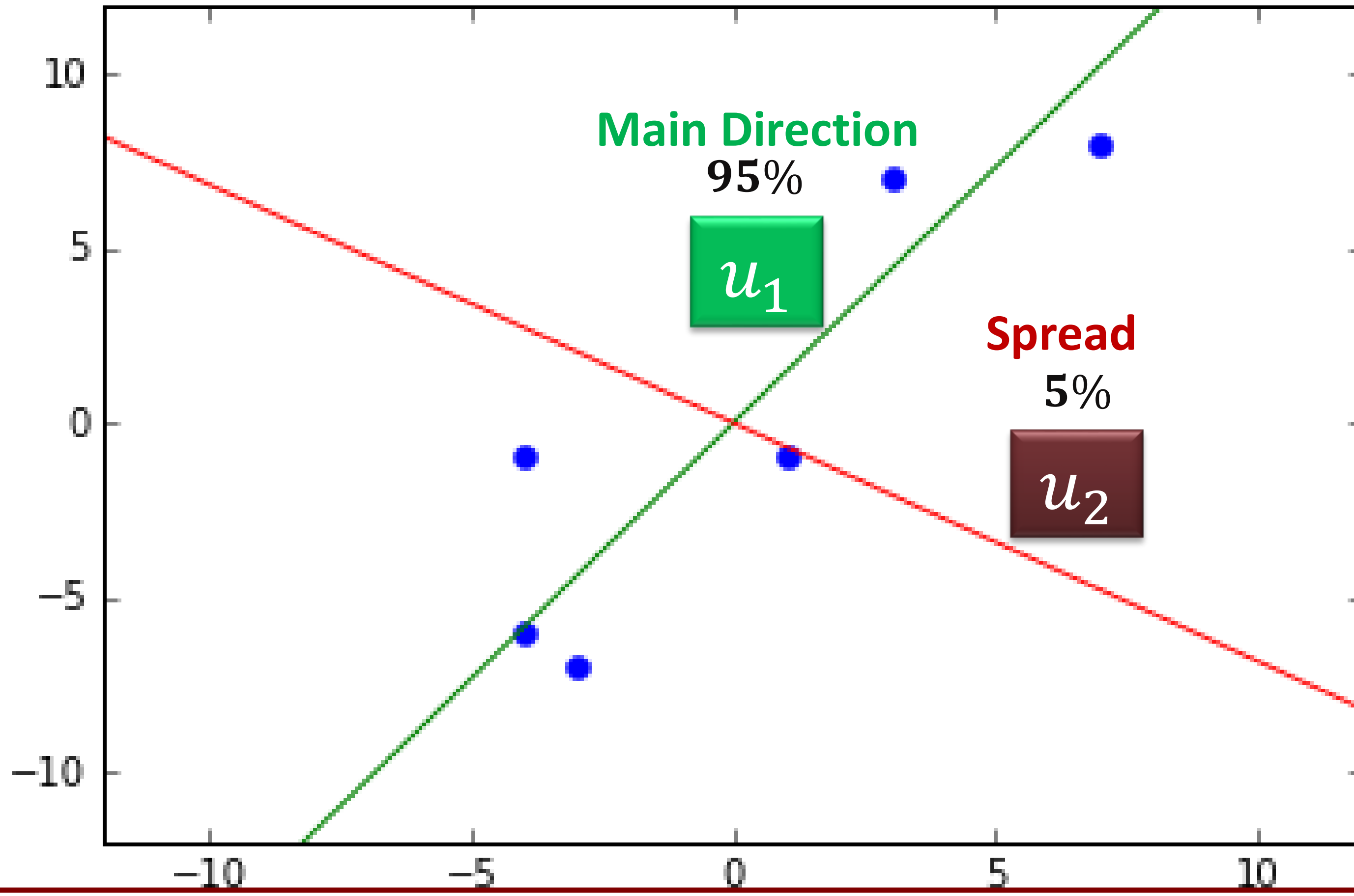
$$\frac{57}{60} = 95\%$$

The next principal component u_2 (orthogonal to u_1) accounts for a smaller fraction σ_2^2/T of the total variance.

$$\frac{3}{60} = 5\%$$

$$A_{2x6} = \begin{bmatrix} 3 & -4 & 7 & 1 & -4 & -3 \\ 7 & -6 & -1 & 8 & -1 & -7 \end{bmatrix} \qquad \boldsymbol{A}_{2x6} = \begin{bmatrix} 93 & 86 & 97 & 91 & 86 & 87 \\ 92 & 79 & 93 & 84 & 84 & 78 \end{bmatrix}$$

What Does this Mean?



$$\boldsymbol{U} = \begin{bmatrix} 0.56 & 0.82 \\ 0.82 & -0.56 \end{bmatrix} = \begin{bmatrix} u_1 & u_2 \end{bmatrix} \quad \boldsymbol{\Sigma} = \begin{bmatrix} \sqrt{57} & 0 \\ 0 & \sqrt{3} \end{bmatrix} \quad S = \begin{bmatrix} 20 & 25 \\ 25 & 40 \end{bmatrix} \quad T = 60$$

$$A_{2x6} = \begin{bmatrix} 3 & -4 & 7 & 1 & -4 & -3 \\ 7 & -6 & -1 & 8 & -1 & -7 \end{bmatrix} \qquad \boldsymbol{A}_{2x6} = \begin{bmatrix} 93 & 86 & 97 & 91 & 86 & 87 \\ 92 & 79 & 93 & 84 & 84 & 78 \end{bmatrix}$$

What Does this Mean?

$$A_{mxn} = A_{Attributes\ x\ Samples}$$
$$= A_{Course\ x\ Student}$$

- The columns of U are the left singular vectors which have the *Course coefficient vectors.*
- The rows of V^T are the right singular vectors *Student Coefficient vectors.*

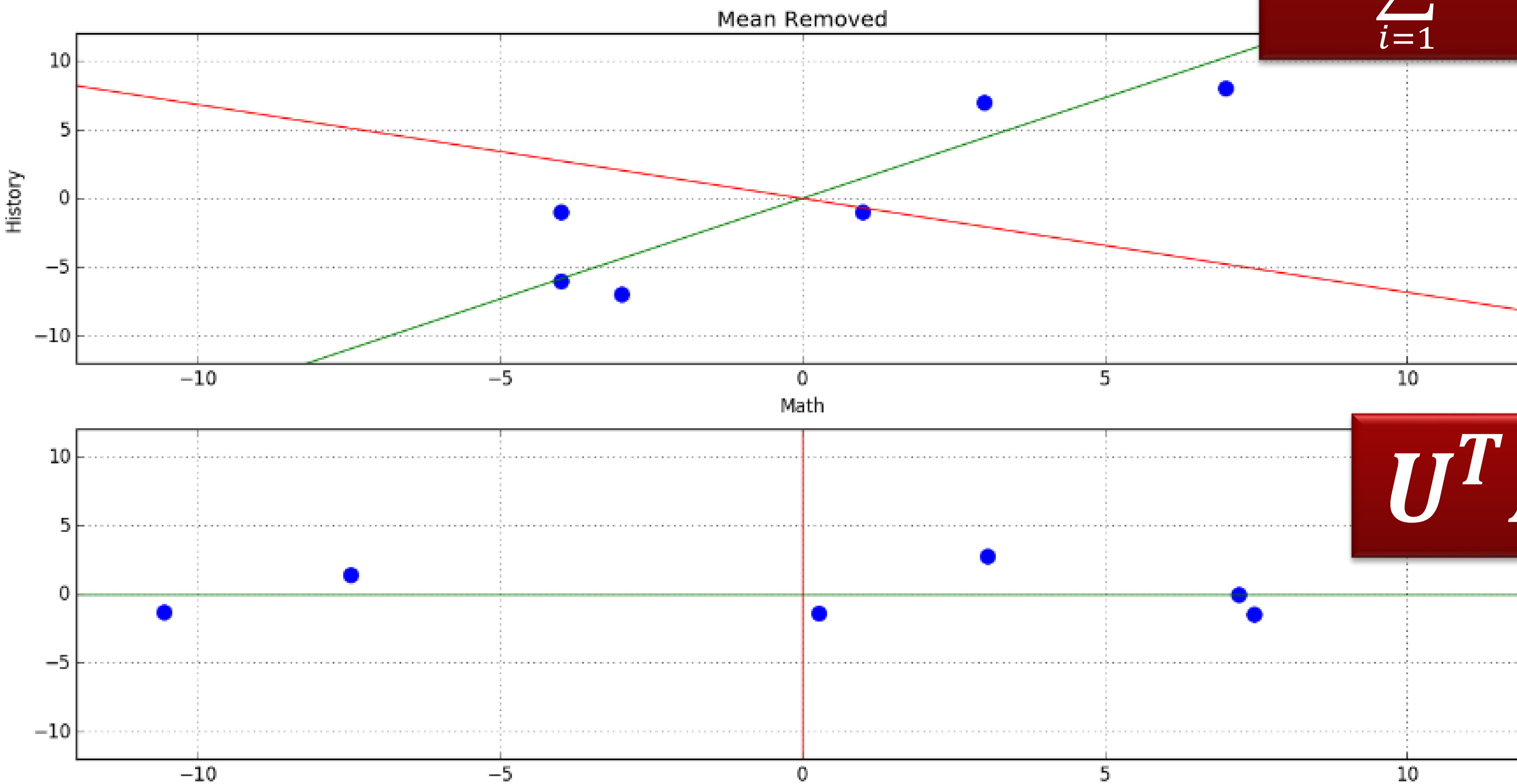
$$\boldsymbol{U} = \begin{bmatrix} 0.56 & 0.82 \\ 0.82 & -0.56 \end{bmatrix} \qquad \boldsymbol{\Sigma} = \begin{bmatrix} \sqrt{57} & 0 \\ 0 & \sqrt{3} \end{bmatrix}$$

$V^T=$

-0.443	0.427	-0.625	0.016	0.182	0.443
0.367	-0.013	-0.335	-0.354	0.702	-0.367

Projection

$$AV = U\Sigma$$
$$A = U\Sigma V^T$$
$$A = \sum_{i=1}^{r} \sigma_i u_i v_i^T$$



$$U^T A$$

Linear combinations of the original samples (exam score) using the entries in U (Principal Components) as weights

Linear Algebra and Statistics

The PCA is a main connection between statistics and the linear algebra of positive definite matrices and the SVD.

The aim is to capture and look for the fundamental variability in the data while not changing the total variance of the data. Once found, the data can be projected onto the direction which has maximum variance.

APPLICATIONS OF PCA

PCA - Applications

Principle Component Analysis (PCA) has several appealing characteristics and applications:

1. **Pattern Finding Technique -** PCA can be used to find the strongest patterns in the data.
2. **Dimensionality Reduction Technique -** Often most of the variability in a high-dimensional data can be captured by a small fraction of the total set of dimensions.
3. **Data preprocessing -** Assuming that the noise in the data is weaker than the patterns, dimensionality reduction can eliminate much of the noise.

Compression of Images - Introduction

It is possible to represent digital images using matrices. For example,

1. **Binary images:** An image can be represented by a matrix whose elements are the numbers 0 and 1 (binary images). These numbers specify the color of each pixel.
2. **Grayscale images** can also be represented by matrices. Each element of the matrix determines the intensity of the corresponding pixel. For convenience, most of the current digital files use integer numbers between 0 (to indicate black, the color of minimal intensity) and 255 (to indicate white, maximum intensity), giving a total of $256 = 2^8$ different levels of gray
3. **Color images**, can be represented by three matrices. Each matrix specifies the amount of red, green and blue that makes up the image. This color system is known as RGB. The elements of these matrices are integer numbers between 0 and 255, and they determine the intensity of the pixel with respect to the color of the matrix. Thus, in the RGB system, it is possible to represent $256^3 = 2^{24} = 16{,}777{,}216$ different colors.

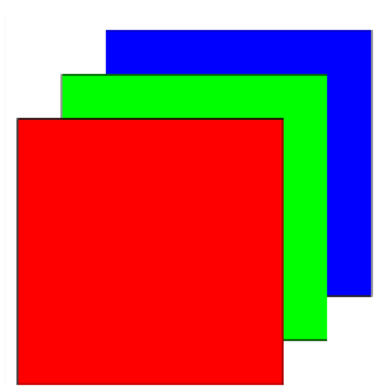
Source: http://blog.kleinproject.org/?p=588

Compression of Images - Introduction

Binary images

Grayscale images

Color images



Compression of Images - Motivation

- Suppose that a grayscale image of size $1000x1000$, must be transmitted from a satellite to a laboratory on Earth. In principle, the satellite would have to send 1 $million$ numbers (one for each pixel).
- Typically, only the first few s singular values of the SVD for the image matrix are significant (the others are "small".) It is enough, then, that the satellite sends, say, the 20 first columns of U and V, and the 20 first singular values (totaling only 20 $*$ $(1000 + 1000 + 1) = 40{,}020$ numbers that must be sent). Upon receiving these data, the laboratory on Earth calculates the reconstructed image that will give an approximation of the original image.

$$\boldsymbol{A} = \sum_{i=1}^{r} \sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^T$$

Source: http://blog.kleinproject.org/?p=588

Example – Compression of Gray-Scale Image

512-by-512 gray-scale image

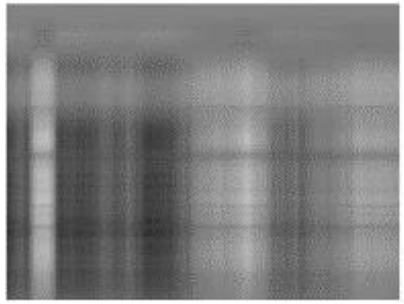
The image contains $512x512$ pixels, each carries intensity information. So, with no compression, a $512X512$ (i.e., 262,144 values) need to be stored and transmitted. It could have a rank up to 512, i.e., up to 512 singular values.

Suppose we use s singular values to approximate the matrix. Then we need to store and transmit $s(512 + 512 + 1) = 1025s$ values.

Source: https://www.projectrhea.org/rhea/index.php/PCA_Theory_Examples

Example – Compression of Gray-Scale Image

Original



(a) 1 principal component

(b) 5 principal component

(c) 9 principal component

(d) 13 principal component

(e) 17 principal component

(f) 21 principal component

(g) 25 principal component

(h) 29 principal component

As the number of principal components increases, the projected (reconstructed images from the original image) image becomes visually close to the original image.

Original: $512X512 = 262{,}144$ values
29 PCs : $29 * (512 + 512 + 1) = 29{,}725$ values

Source:
https://www.projectrhea.org/rhea/index.php/PCA_Theory_Examples

Eigenfaces - Face Recognition

Motivation - Identify a New Face

Given a database of face images (i.e., training set) and a test (new) face image as matrices of pixel values: Does the new image correspond to one of those in the database?

Source: http://www-users.cs.umn.edu/~saad/

Eigenfaces - Face Recognition

Problem:

Each image of a face is a point in a very high dimensional space.

Main Assumption:

Images of faces, being similar in overall configuration, will not be randomly distributed in a huge image space. They can be described by a relatively low dimensional subspace.

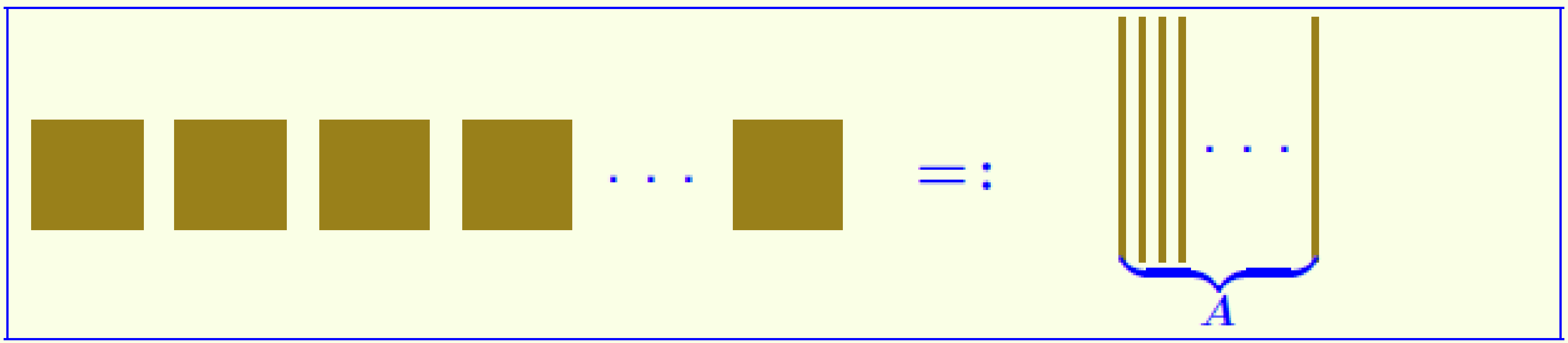
Note:

We are not compressing the images. We are identifying a new image.

Eigenfaces - Face Recognition

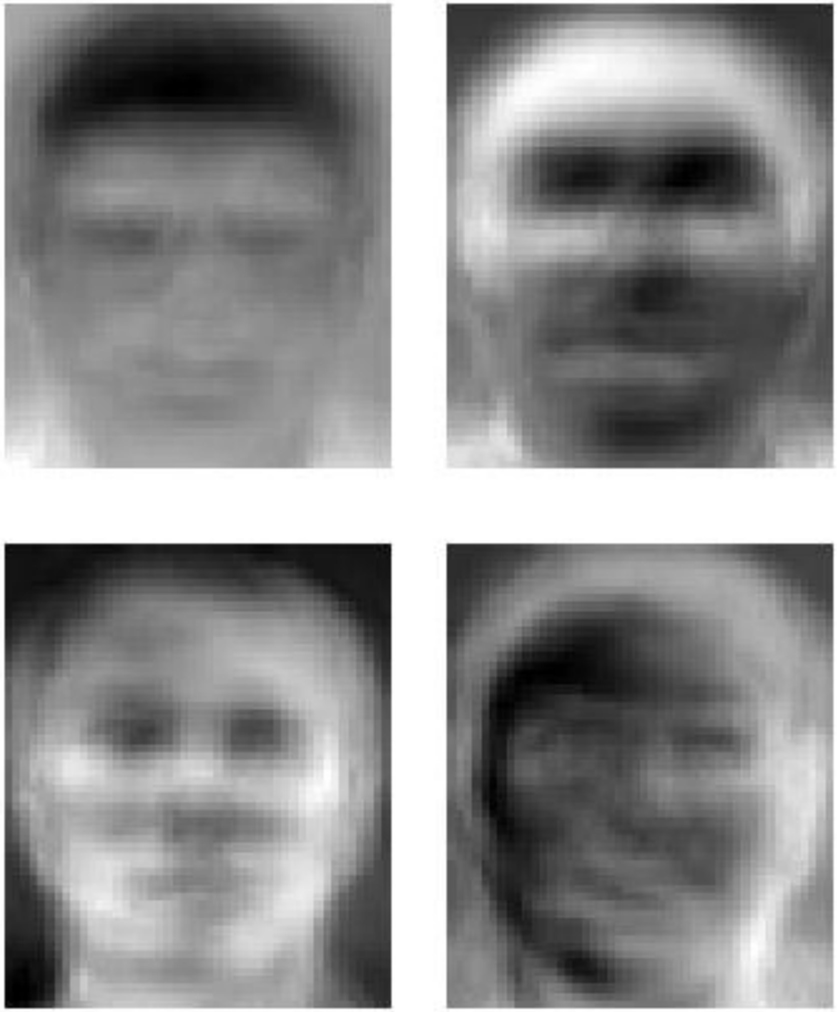
Process:

1. Consider each picture as a (1-D) column of all pixels.
2. Put together into a matrix A of size $\#\ pixels\ x\ \#\ images$.
3. Perform PCA of A and perform comparison with any test image in low-dimension space.



Source: http://www-users.cs.umn.edu/~saad/

Eigenfaces - Face Recognition



Eigenface pick out hairline, mouth and eye shape

Classification can be achieved by comparing how faces are represented by the basis set.

When given an unknown face, compute its distance to all of the existing points in a database of known faces.

This method is best when all the pictures are taken under similar conditions, and it does not perform well when several environmental factors are varied.

Latent Semantic Analysis (LSA)

- A technique in natural language processing, of analyzing relationships between a set of documents and the terms they contain by producing a set of concepts related to the documents and terms.

- LSA assumes that words that are close in meaning will occur in similar pieces of text (the distributional hypothesis). A matrix containing word counts per document (rows represent unique words and columns represent each document) is constructed from a large piece of text and a mathematical technique called **singular value decomposition (SVD)** is used to reduce the number of rows while preserving the similarity structure among columns.

- Documents are then compared by taking the **cosine of the angle between the two vectors** (or the **dot product between the normalizations of the two vectors**) formed by any two columns. Values close to 1 represent very similar documents while values close to 0 represent very dissimilar documents.

Source: Wikipedia

Text Mining

Text mining –
Methods to extract useful information from large and often unstructured collection of texts.

Word – Document Matrix

$$
\begin{array}{cc}
 & \text{Document} \\
A = & \begin{bmatrix} 2 & 0 & 8 & 6 & 0 \\ 1 & 6 & 0 & 1 & 7 \\ 5 & 0 & 7 & 4 & 0 \\ 7 & 0 & 8 & 5 & 0 \\ 0 & 10 & 0 & 0 & 7 \end{bmatrix} \text{Words}
\end{array}
$$

Words

Doctor
Car
Nurse
Hospital
Wheel

- SVD has the added benefit that in the process of dimensionality reduction, the representation of items that share substructure become more similar to each other, and items that were dissimilar to begin with may become more dissimilar as well.
- Documents about a particular topic become more similar even if the exact same words don't appear in all of them.

Source: http://davetang.org/file/Singular_Value_Decomposition_Tutorial.pdf

Text Mining

$$AA^T = \begin{bmatrix} 2 & 0 & 8 & 6 & 0 \\ 1 & 6 & 0 & 1 & 7 \\ 5 & 0 & 7 & 4 & 0 \\ 7 & 0 & 8 & 5 & 0 \\ 0 & 10 & 0 & 0 & 7 \end{bmatrix} \begin{bmatrix} 2 & 1 & 5 & 7 & 0 \\ 0 & 6 & 0 & 0 & 10 \\ 8 & 0 & 7 & 8 & 0 \\ 6 & 1 & 4 & 5 & 0 \\ 0 & 7 & 0 & 0 & 7 \end{bmatrix} = \begin{bmatrix} 104 & 8 & 90 & 108 & 0 \\ 8 & 87 & 9 & 12 & 109 \\ 90 & 9 & 90 & 111 & 0 \\ 108 & 12 & 111 & 138 & 0 \\ 0 & 109 & 0 & 0 & 149 \end{bmatrix}$$

$$A^T A = \begin{bmatrix} 79 & 6 & 107 & 68 & 7 \\ 6 & 136 & 0 & 6 & 112 \\ 107 & 0 & 177 & 116 & 0 \\ 68 & 6 & 116 & 78 & 7 \\ 7 & 112 & 0 & 7 & 98 \end{bmatrix}$$

Singular values

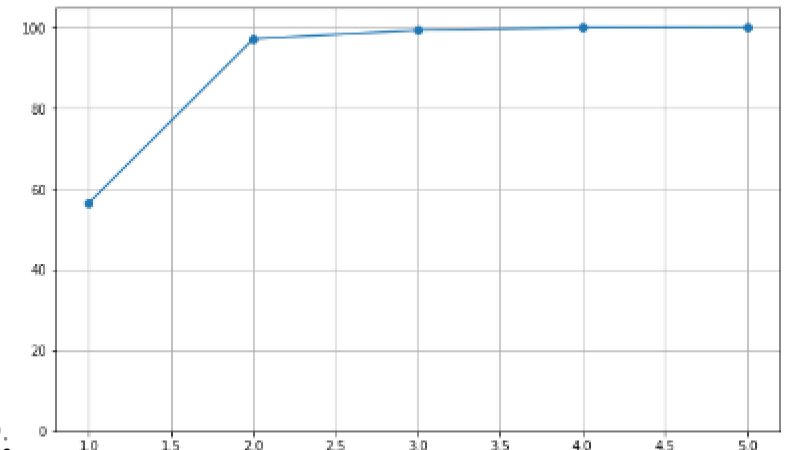
$\sigma_1 = \sim 321$ ~56%

$\sigma_2 = \sim 230$ ~40%

$\sigma_3 = \sim 12$ ~2%

$\sigma_4 = \sim 3.9$

$\sigma_5 = \sim 0.12$



Source: http://davetang.org/file/Singular_Value_Decomposition_Tutorial.pdf

Columns: principal components
Rows: words

Text Mining

$$U = \begin{array}{c} \text{PCs} \\ \begin{bmatrix} \boxed{\begin{matrix} -0.54 & 0.07 \\ -0.10 & -0.59 \\ -0.53 & 0.06 \\ -0.65 & 0.07 \\ -0.06 & -0.80 \end{matrix}} & \begin{matrix} \boxed{0.82} & -0.11 & 0.12 \\ -0.11 & -0.79 & -0.06 \\ -0.21 & 0.12 & -0.81 \\ -0.51 & 0.06 & 0.56 \\ 0.09 & 0.59 & 0.04 \end{matrix} \end{bmatrix} \end{array} \begin{array}{c} \text{Words} \end{array} \begin{array}{c} \text{Doctor} \\ \text{Car} \\ \text{Nurse} \\ \text{Hospital} \\ \text{Wheel} \end{array}$$

- Words are represented as row vectors containing linearly independent components.
- Some word cooccurrence patterns in these documents are indicated by the signs of the coefficients in U.

First PC: All negative, indicating the general cooccurrence of words and documents.

Second PC: Two groups visible in the second column vector of U car and wheel have negative coefficients, while doctor, nurse, and hospital are all positive, indicating a grouping in which wheel only cooccurs with car.

Third PC: Indicates a grouping in which car, nurse, and hospital occur only with each other.

Customer Data Matrix

Rows ➔ Customers
Columns ➔Days

Days

Customers

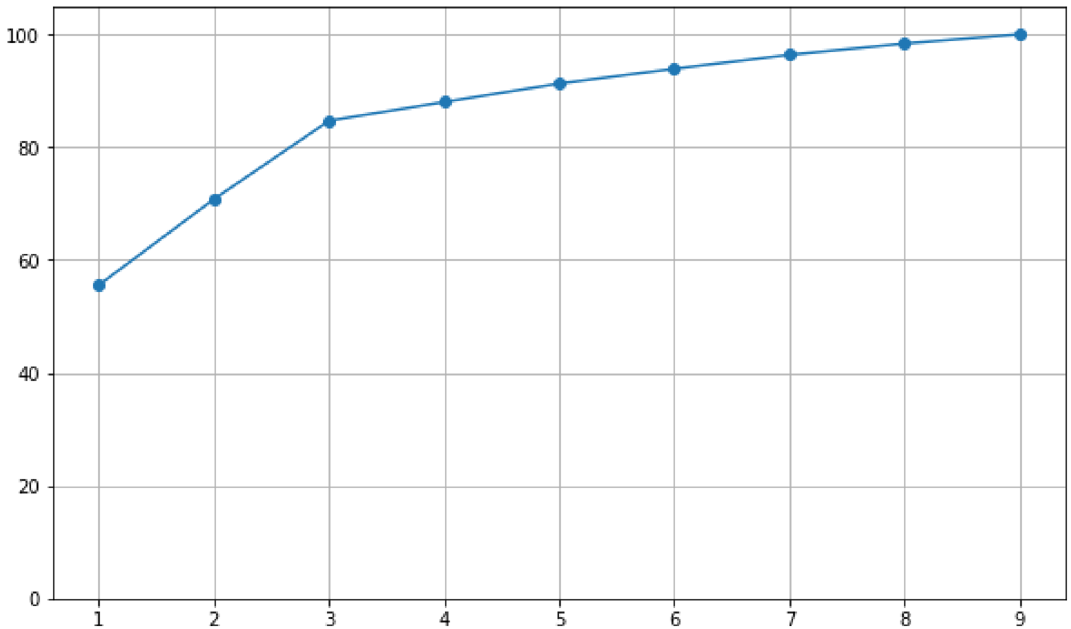
1	0	1	0	1	0	0	1	1	0	1	1	1	1
0	3	3	3	3	3	3	0	3	3	0	3	3	0
2	0	2	2	2	0	0	2	2	0	2	2	2	3
0	2	0	2	0	2	2	0	0	2	0	0	0	0
4	0	4	3	4	0	0	4	4	0	4	4	4	4
3	0	3	3	3	0	0	3	4	0	3	3	3	3
4	0	4	4	4	0	0	4	4	0	4	4	3	4
0	3	0	3	0	3	3	0	0	3	0	0	0	0
0	4	0	3	0	3	3	0	0	3	0	0	0	0
2	0	2	1	2	0	0	2	2	0	2	2	2	2
0	4	0	4	0	4	4	0	0	4	0	0	0	0
0	1	0	1	0	1	1	0	0	1	0	0	0	0
1	0	1	2	1	0	0	1	1	0	1	1	1	1
0	2	0	2	0	3	2	0	0	2	0	0	0	0
1	0	1	1	1	0	0	2	1	0	1	1	1	1

Source: Linear Algebra Methods for Data Mining Saara Hyv¨onen

Customer Data – Singular Values

The first three principal components are enough to capture most of the variation in the data.

16.8001
4.6731
4.2472
1.0000
0.9957
0.7907
0.7590
0.6026
0.5025
0.0000
0.0000
0.0000
0.0000
0.0000



Customer Data - PCs

pcc=

-0.2941	-0.0193	-0.3288	-0.2029
-0.3021	-0.0414	-0.3586	-0.2072
-0.2592	-0.2009	0.3244	-0.1377
-0.3012	-0.2379	0.2416	0.1667
-0.2833	-0.2276	0.2517	0.0071
0.2603	-0.3879	-0.0717	-0.2790
0.2437	-0.3683	-0.0229	-0.1669
-0.2921	-0.0554	-0.3398	-0.2089
-0.2921	-0.0554	-0.3398	-0.2089
-0.2833	-0.2276	0.2517	0.0071
-0.2833	-0.2276	0.2517	0.0071
-0.0146	-0.4309	-0.4138	0.7722
0.2437	-0.3683	-0.0229	-0.1669
0.2573	-0.3633	-0.0349	-0.2277

scc=

-0.6736	2.8514	0.1816	0.0140
-3.2867	1.1052	-0.3078	0.0041
-7.9357	-2.1006	-1.1928	0.3377
-5.8909	-0.8154	-0.1672	0.3723
-0.3940	2.4398	0.0966	0.9891
-2.9700	1.5775	0.4645	-0.5609
-8.1803	-1.8707	-0.4546	-0.5722
-0.3649	3.3016	0.9241	-0.5553
3.2162	2.6761	0.4037	0.1542
4.2067	0.7574	-0.1624	0.0859
5.1972	-1.1613	-0.7286	0.0175
6.1877	-3.0800	-1.2947	-0.0508
4.4640	0.3941	-0.1973	-0.1419
5.4575	-1.5492	-0.8002	-0.2615
0.9667	-4.5260	3.2352	0.1679

Columns: principal components
Rows: days

Columns: coordinates of customers in PC basis
Rows: customers

Source: Linear Algebra Methods for Data Mining Saara Hyv¨onen

Customer Data - PCs

pcc1=	
mon	-0.2941
tue	-0.3021
wed	-0.2592
thu	-0.3012
fri	-0.2833
sat	0.2603
sun	0.2437
mon	-0.2921
tue	-0.2921
wed	-0.2833
thu	-0.2833
fri	-0.0146
sat	0.2437
sun	0.2573

scc1=	
ABC Ltd	-0.6736
BCD Inc	-3.2867
CDECorp	-7.9357
DEF Ltd	-5.8909
EFG Inc	-0.3940
FGHCorp	-2.9700
GHI Ltd	-8.1803
HIJ Inc	-0.3649
Smith	3.2162
Jones	4.2067
Brown	5.1972
Black	6.1877
Blake	4.4640
Lake	5.4575
Mr. X	0.9667

pcc2=	
mon	-0.0193
tue	-0.0414
wed	-0.2009
thu	-0.2379
fri	-0.2276
sat	-0.3879
sun	-0.3683
mon	-0.0554
tue	-0.0554
wed	-0.2276
thu	-0.2276
fri	-0.4309
sat	-0.3683
sun	-0.3633

scc2=	
ABD Ltd	2.8514
BCD Inc	1.1052
CDECorp	-2.1006
DEF Ltd	-0.8154
EFG Inc	2.4398
FGHCorp	1.5775
GHI Ltd	-1.8707
HIJ Inc	3.3016
Smith	2.6761
Jones	0.7574
Brown	-1.1613
Black	-3.0800
Blake	0.3941
Lake	-1.5492
Mr. X	-4.5260

1st PC: weekdays vs. weekends
Result: weekday customers(companies) get separated from weekend customers (private citizens).
Big customers end up at extreme ends.

2nd PC: Weekends and weekdays have about equal total weight
Most weight on an exceptional Friday.
Result: Separates big customers from small ones. Mr. X gets separated from the other customers.

Posted Resource - PCA and Image Compression with Python

1. **PCA and image compression with numpy**

http://glowingpython.blogspot.com/2011/07/pca-and-image-compression-with-numpy.html

2. **Principal Component Analysis in an image with scikit-learn and scikit-image**

https://pakallis.wordpress.com/2013/06/20/principal-component-analysis-in-an-image-with-scikit-learn-and-scikit-image/

3. **Eigenfaces for recognition**

M. Turk; A. Pentland (1991). "**Eigenfaces for recognition**" (PDF). Journal of Cognitive Neuroscience. 3 (1): 71–86.

List of Important Dates

Section 1

- **Sessions:**
  - Session #5 – Tuesday 9/13/2022 6pm-9pm CST
- **Quizzes:**
  - Quiz #4 - Due Thursday 9/8/2022 Midnight CST
  - Quiz #5 - Due Thursday 9/15/2022 Midnight CST (Optional quiz)
- **Assignments:**
  - Assignment #4 - Due Saturday 9/10/2022 Midnight CST

List of Important Dates

Section 6

- **Sessions:**
  - Session #5 – Thursday 9/15/2022 6pm-9pm CST
- **Quizzes:**
  - Quiz #4 - Due Saturday 9/10/2022 Midnight CST
  - Quiz #5 - Due Saturday 9/17/2022 Midnight CST (Optional quiz)
- **Assignments:**
  - Assignment #4 - Due Saturday 9/10/2022 Midnight CST

EXTRA SLIDES

Recall: Quadratic and Cubic Equations

Quadratic Equation

$$\boxed{(x+b)^2 = x^2 + b^2 + 2bx}$$

Cubic Equation

$$\boxed{(x+b)^3 = x^3 + b^3 + 3bx^2 + 3x^2b}$$

Recall: Roots of Quadratic Equations

$$ax^2 + bx + c = 0$$

$$x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Example

$$x^2 - 2x - 2 = 0 \qquad a = 1; b = -2; c = -2$$

$$x_{1,2} = \frac{2 \pm \sqrt{(-2)^2 - 4 * 1 * (-2)}}{2 * 1}$$

$$x_{1,2} = \frac{2 \pm \sqrt{4 + 8}}{2} = \frac{2 \pm \sqrt{3 * 4}}{2} = \frac{2 \pm 2\sqrt{3}}{2} = 1 \pm \sqrt{3}$$

Recall: Roots of Cubic Equations

$$ax^3 + ab + cx + d = 0$$

How many <u>maximum unique</u> roots do you expect?

https://en.wikipedia.org/wiki/Cubic_function#General_solution_to_the_cubic_equation_with_real_coefficients

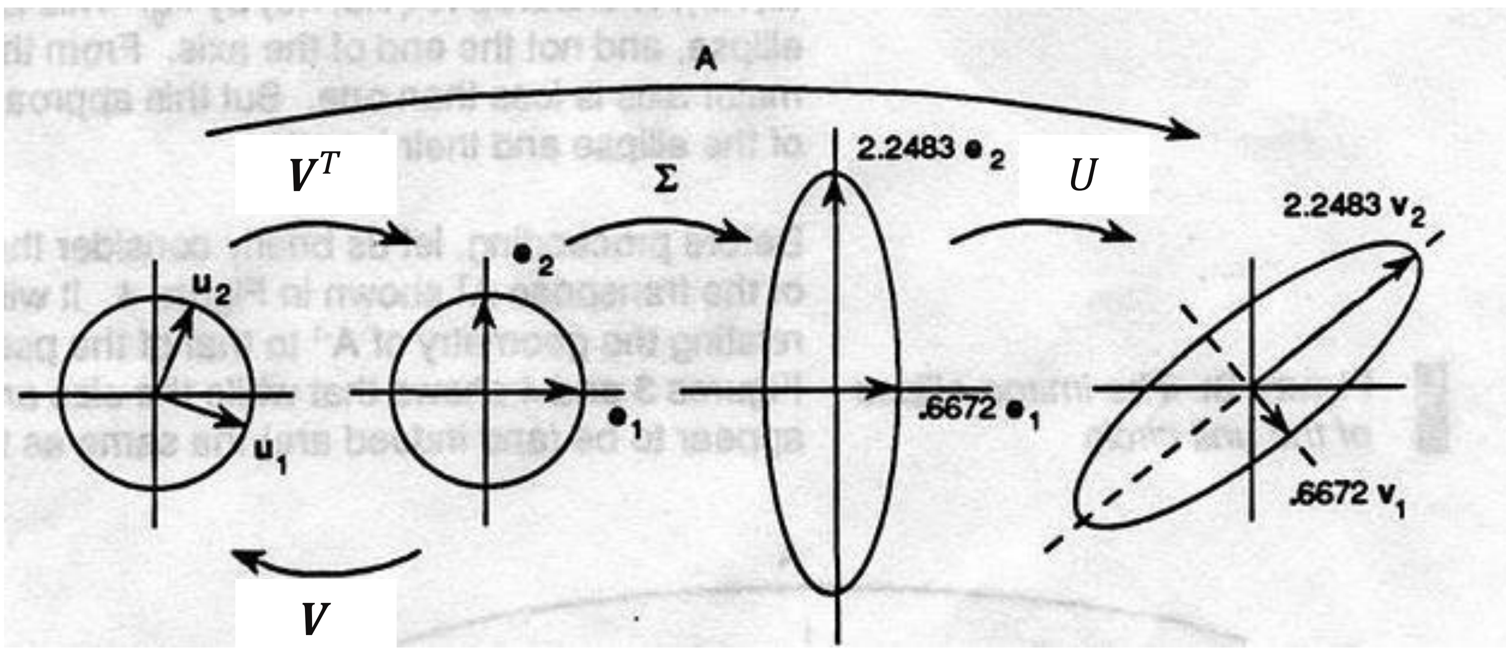
For this class we will use Python
numpy.roots()

The Geometry of SVD

SVD Separates a matrix into three steps

$$A_{mxn} = U_{mxm}\Sigma_{mxn}V_{nxn}^{T}$$
$$= (Orthonormal)(Diagonal)(Orthonormal)$$
$$= (Rotation)(Stretching)(Rotation)$$

Source: http://www.norsemathology.org/wiki/index.php?title=The_Singular_Value_Decomposition



$$U = \begin{bmatrix} 0.6257 & 0.78 \\ -0.78 & 0.6257 \end{bmatrix};\ \Sigma = \begin{bmatrix} 0.6672 & 0 \\ 0 & 2.2483 \end{bmatrix};\ V^T = \begin{bmatrix} 0.9378 & 0.3469 \\ -0.3469 & 0.9378 \end{bmatrix}$$

*

Diagonalization - Symmetric Matrices

$$A = Q \quad \Lambda \quad Q^T$$

counterclockwise rotation

Eigen-stretching

Clockwise rotation

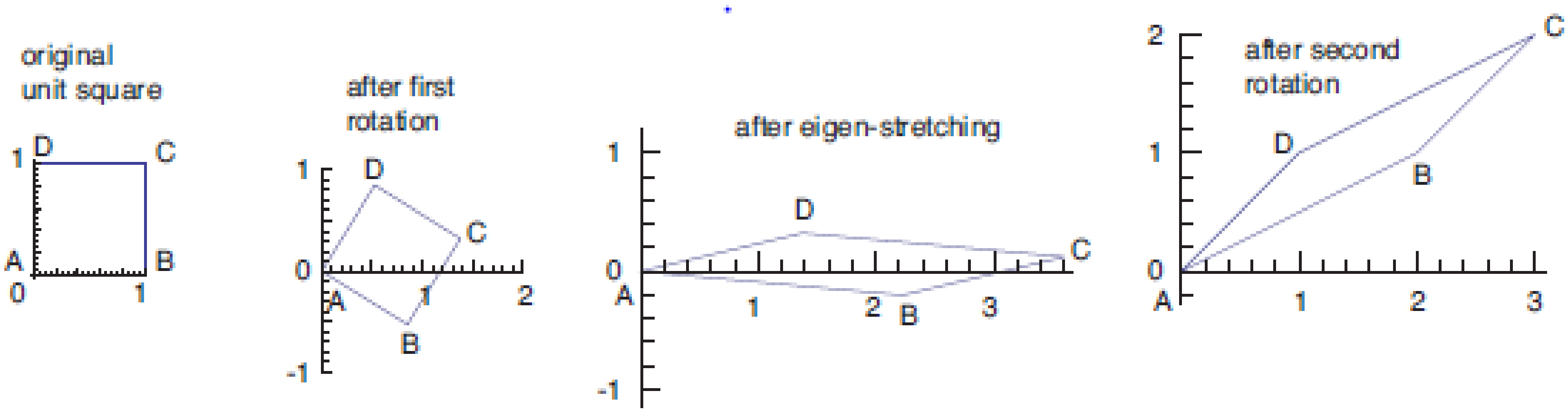


Image Source: Lecture Notes on Mathematica l methods By Mihir Sen and Joseph Powers