

Face Recognition and Ranking Data

1 Face Recognition

- problem and dimension
- machine learning
- covariances and eigenfaces

2 Ranking Data

- engineering a search engine
- the random surfer model
- beyond PageRank

MCS 472 Lecture 25
Industrial Math & Computation
Jan Verschelde, 11 March 2026

Face Recognition and Ranking Data

1 Face Recognition

- **problem and dimension**
- machine learning
- covariances and eigenfaces

2 Ranking Data

- engineering a search engine
- the random surfer model
- beyond PageRank

face recognition

- Face recognition is challenging because of
 - 1 the position of the head,
 - 2 lighting conditions, and
 - 3 moods and expressions.
- Many automated systems have been developed.
- A linear algebra approach is based on eigenfaces, a method proposed by Sirovich and Kirby, 1987.

dimension reduction

The input is a p -by- q grayscale image.

The resolution of an image is $m = p \times q$.

All images of the same resolution live in an m -dimensional space.

The subspace of *all facial images* has a low dimension,

independent of the resolution.

This result is described in the paper by

Neil Muller, Lourenço Magaia, B. M. Herbst:

[Singular Value Decomposition, Eigenfaces, and 3D Reconstructions.](#)

SIAM Review, Vol. 46, No. 3, pages 518–545, 2004.

the singular value decomposition

The singular value decomposition of a p -by- q matrix A is

$$A = U\Sigma V^T, \quad U \in \mathbb{R}^{p \times p}, \quad \Sigma \in \mathbb{R}^{p \times q}, \quad V \in \mathbb{R}^{q \times q},$$

where U and V are orthogonal: $U^{-1} = U^T$, $V^{-1} = V^T$,
and Σ is a diagonal matrix, with on its diagonal

$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{\min(p,q)},$$

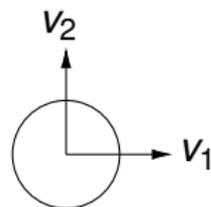
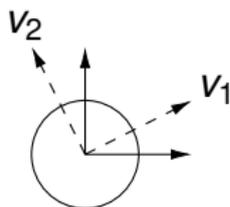
are the *singular values of the matrix A* .

If $\text{rank}(A) = r$, then $\sigma_i = 0$ for all $i > r$.

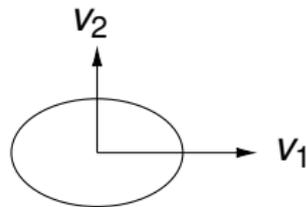
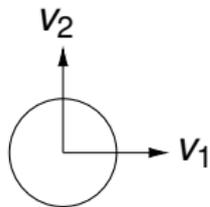
Ignoring the smallest singular values leads to a dimension reduction.

the geometric interpretation of $A = U\Sigma V^T$

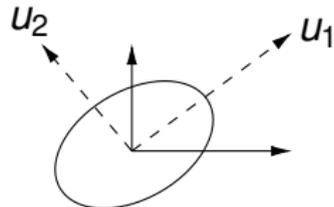
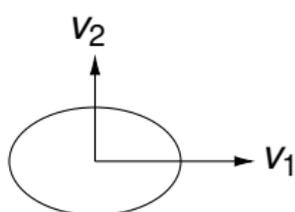
1 rotate a circle



2 stretch the circle



3 rotate the ellipse



relationship with eigenvalues and eigenvectors

By the singular value decomposition of A :

$$A = U\Sigma V^T \quad \Rightarrow \quad AV = U\Sigma = \Sigma U.$$

This implies

$$Av_j = \sigma u_j, \quad j = 1, 2, \dots, \min(p, q).$$

The largest singular value measures the magnitude of A :

$$\|A\|_2 = \max_{\|x\|_2=1} \|Ax\|_2 = \sigma_1.$$

Exercise 1: Verify that $A = U\Sigma V^T$ implies

$$A^TAV = V\Sigma^T\Sigma \quad \text{and} \quad AA^TU = U\Sigma\Sigma^T.$$

Face Recognition and Ranking Data

1 Face Recognition

- problem and dimension
- **machine learning**
- covariances and eigenfaces

2 Ranking Data

- engineering a search engine
- the random surfer model
- beyond PageRank

machine learning

In machine learning, the eigenfaces method is known as the Principal Component Analysis (PCA).

Alternatives to PCA are

- Histogram Oriented Gradients (HOG),
- Convolutional Neural Networks (CNN).

Óscar A. Alvarado-Morán and Oscar A. Esquivel-Flores:
Parallel face recognition algorithms using Julia + CUDAnative.jl.
Poster at JuliaCon 2020.

the procedure

The procedure by Neil Muller et al. is summarized below.

- The XM2VTS Face Database of the University of Surrey at <http://www.ee.surrey.ac.uk/Research/VSSP/xm2vtsdb> consists of many RGB color images against a blue background.
- The background is removed by color separation.
- Averaging RGB values, the images were converted to grayscale.
- A training set has 600 images, 3 of each, for 200 individuals.

If the training set is representative,
then all facial images would be in a 150-dimensional subspace.

Face Recognition and Ranking Data

1 Face Recognition

- problem and dimension
- machine learning
- **covariances and eigenfaces**

2 Ranking Data

- engineering a search engine
- the random surfer model
- beyond PageRank

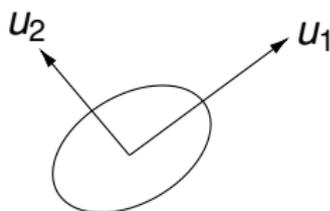
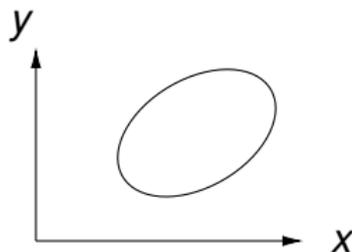
covariances

What is the typical range of measurements of a face?

Consider n measurements (x_i, y_i) for $i = 1, 2, \dots, n$, where

- 1 x_i is the width, measured ear-to-ear, and
- 2 y_i is the height, measured from chin to between the eyes.

Assume all n measurements lie within an ellipse:



The coordinate change to u_1 and u_2 is computed via a singular value computations on a shifted matrix.

singular values of the measurements

To find the typical range of width and height of a face:

- 1 Compute the mean (\bar{x}, \bar{y}) of all n observations.
- 2 Subtract \bar{x} from all x_i and subtract \bar{y} from all y_i .
- 3 Divide the n shifted observations by \sqrt{n} .
- 4 Apply the singular value decomposition on the matrix A , which has in its columns the shifted observations.

The first two columns of U give the directions of maximum variations and the two largest singular values are the magnitudes of the standard deviations in those two directions.

With the mean and standard deviation of the measurements, we can define a normal distribution.

The farther observations fall outside the normal ellipse, the less likely the measurements represent a face.

eigenfaces

Stacking the columns of the images into one vector, we obtain a matrix $F = [f_j \text{ is stacked image for } j = 1, 2, \dots, n]$.

- 1 Compute the mean $\mu = \frac{1}{n} \sum_{j=1}^n f_j$.
- 2 Shift the data $a_j = x_j - \mu, j = 1, 2, \dots, n$.
- 3 Define the matrix

$$A = \frac{1}{\sqrt{n}} \begin{bmatrix} x_1 & x_2 & \cdots & x_n \end{bmatrix}.$$

- 4 Compute the singular value decomposition $A = U\Sigma V^T$.

The columns of U are the eigenvectors of AA^T (the covariance matrix).

Those columns of U are *the eigenfaces of F* .

orthogonal projections

The mean of the training set is μ .

The matrix U is orthogonal. Use the first ν eigenfaces:

$$U_\nu = [u_1 \ u_2 \ \cdots \ u_\nu].$$

For any image f (as a stacked vector), compute its projection

$$y = U_\nu^T (f - \mu),$$

with is *the eigenface representation of f* .

Then *the eigenface reconstruction of f* is $\tilde{f} = U_\nu y + \mu$.

Comparing the reconstructed faces to the given ones in the training set leads to the finding of the best value for ν .

Steps in the Principal Component Analysis (L-22)

Three steps:

- 1 Standardize the data, via the formula

$$z = \frac{x - \text{mean}}{\text{standard deviation}},$$

applied to each number x in the input.

- 2 Setup the covariance matrix C . For each pair (z_i, z_j) ,

$\text{cov}(z_i, z_j)$ is the (i, j) -th element of the matrix.

The matrix C is symmetric and positive definite.

- 3 Compute the eigenvectors and eigenvalues of C .
Sort the eigenvectors in increasing magnitude of the eigenvalues.

Then the feature vector consist of the first p eigenvectors.

proposal for a project topic

Read the paper and consider the software:

- Florian Schroff, Dmitry Kalenichenko, and James Philbin:
FaceNet: A Unified Embedding for Face Recognition and Clustering. <https://arxiv.org/abs/1503.03832>
- OpenFace: free and open source face recognition with deep neural networks.
<https://cmusatyalab.github.io/openface/>

Do the following:

- 1 Read the paper and install the software.
- 2 Experiment with many pictures of the same two people. Train the neural network to recognize the faces on the first half of the pictures. Test the neural network on the second half.
- 3 What is the success rate? Can you improve the selection of the pictures? What if passport style pictures are used?

Face Recognition and Ranking Data

1 Face Recognition

- problem and dimension
- machine learning
- covariances and eigenfaces

2 Ranking Data

- engineering a search engine
- the random surfer model
- beyond PageRank

the search engine problem

Given a search query from a user,
among the immense set of web pages that contains the query words,
which pages are most important to the user?

Engineering challenge: answer tens of millions of queries each day.

Google is designed to be a scalable search engine.

- S. Brin and L. Page:
[The anatomy of large-scale hypertextual web search engine.](#)
Computer Networks and ISDN systems 30, pages 107-117, 1998.
- L. Page, S. Brin, R. Motwani, T. Winograd:
[The PageRank Citation Ranking: Bringing Order to the Web.](#)
Technical Report. Stanford InfoLab.

simplified PageRank

The crawlable web is a graph of pages (nodes) and links (edges).

- Every page has forward links (outedges) and backlinks (inedges).
- Highly linked pages are more important than pages with few links.

Definition (simplified PageRank)

For a web page u ,

- F_u is the set of pages u points to,
- B_u is the set of pages that point to u .

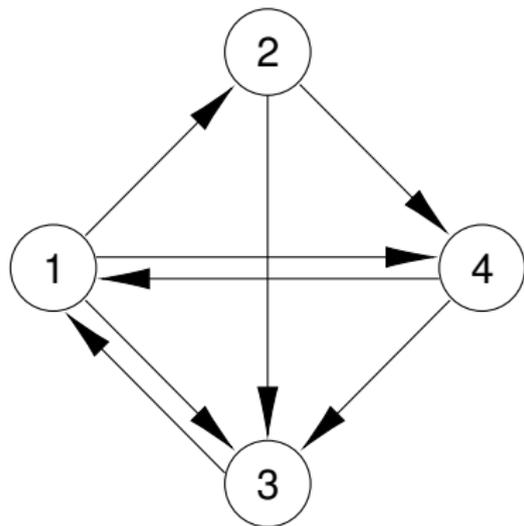
Let $N_u = |F_u|$ and c is a normalization factor.

A simple ranking is *a simplified PageRank* is

$$R(u) = c \sum_{v \in B_u} \frac{R(v)}{N_v}.$$

example: a 4-page web

Evaluating $R(u) = \sum_{v \in B_u} \frac{R(v)}{N_v}$ goes as follows:



- $B_1 = \{3, 4\}$ pages pointing to 1
- $F_3 = \{1\}$ pages 3 points to
- $F_4 = \{1, 3\}$ pages 4 points to
- $N_3 = |F_3| = 1$, $N_4 = |F_4| = 2$

$$\Rightarrow R(1) = R(3)/1 + R(4)/2$$

$$A = \begin{bmatrix} 0 & 0 & 1 & \frac{1}{2} \\ \frac{1}{3} & 0 & 0 & 0 \\ \frac{1}{3} & \frac{1}{2} & 0 & \frac{1}{2} \\ \frac{1}{3} & \frac{1}{2} & 0 & 0 \end{bmatrix}, R = AR$$

Example from *The \$25,000,000,000 Eigenvector: The Linear Algebra behind Google*, by Kurt Bryan and Tanya Leise, SIAM Review 48(3):569–581, 2006.

the dominant eigenvector

The rank of page u is $R(u) = c \sum_{v \in B_u} \frac{R(v)}{N_v}$.

- B_u is the number of pages that point to u .
- N_u is the number of pages the page u points to.
- $c < 1$ because many pages do not point to any other pages.

Let A be a square matrix indexed by web pages.

- $A_{u,v} = 1/N_u$, if there is an edge from u to v ,
- $A_{u,v} = 0$, otherwise.

Consider the vector $R = [R(u) \text{ for all pages } u]$, then:

$$R = cAR, \quad \text{or equivalently} \quad AR = (1/c)R,$$

so R is an eigenvector of A with eigenvalue $1/c$.

example continued

```
julia> A = [0 0 1 1/2; 1/3 0 0 0; 1/3 1/2 0 1/2; 1/3 1/2 0 0]  
4×4 Matrix{Float64}:
```

```
 0.0      0.0  1.0  0.5  
0.333333  0.0  0.0  0.0  
0.333333  0.5  0.0  0.5  
0.333333  0.5  0.0  0.0
```

```
julia> using LinearAlgebra
```

```
julia> L, V = eigen(A);
```

```
julia> L
```

```
4-element Vector{ComplexF64}:
```

```
-0.36062333350611053 - 0.4109755454950538im  
-0.36062333350611053 + 0.4109755454950538im  
-0.278753332987779 + 0.0im  
 1.0000000000000002 + 0.0im
```

```
julia> V[:,4]
```

```
4-element Vector{ComplexF64}:
```

```
 0.7210101217513317 + 0.0im  
 0.24033670725044398 + 0.0im  
 0.5407575913134985 + 0.0im  
 0.36050506087566603 + 0.0im
```

=> page 1 scores highest

the PageRank

Problem with the simplified PageRank: a rank sink caused

- by two pages pointing to each other but to no other page, and
- one third page pointing to one of those two pages.

Definition (the PageRank)

The *PageRank of page u* is

$$R'(u) = c \sum_{v \in B_u} \frac{R'(v)}{N_v} + c E(u),$$

where $E(u)$ is some vector that corresponds to the source of the rank, such that c is maximized and $\|R\|_1 = 1$.

The 1-norm is $\|R\|_1 = |R_1| + |R_2| + \dots + |R_n|$, n is the length of R .

In matrix notation: $R' = c(AR' + E)$.

Face Recognition and Ranking Data

1 Face Recognition

- problem and dimension
- machine learning
- covariances and eigenfaces

2 Ranking Data

- engineering a search engine
- **the random surfer model**
- beyond PageRank

the random surfer model

The PageRank can be derived from random walks in graphs.

If a surfer ever gets in a small loop of web pages, then the surfer will most likely jump out of the loop.

The likelihood of a surfer jumping to a random page is modeled by E in

$$R'(u) = c \sum_{v \in B_u} \frac{R'(v)}{N_v} + cE(u).$$

computing the PageRank

Initialize R_0 to some random vector.

do

$$R_{i+1} = AR_i$$

$$d = \|R_i\|_1 - \|R_{i+1}\|_1$$

$$R_{i+1} = R_i + dE$$

$$\delta = \|R_{i+1} - R_i\|_1$$

while $\delta > \epsilon$.

We recognize a variant of the power method to compute the eigenvector associated with the dominant eigenvalue.

Exercise 2: What about the growth of AR_i ?

Why does the algorithm above not divide R by its largest component?

a stochastic matrix

Let $P = [p_{i,j}]$ a square matrix,
with indices i and j running over all web pages.

$p_{i,j}$ is the probability of moving from page j to page i .

Because the $p_{i,j}$'s are probabilities,
the sum of all elements in the j -th column of P equals one.

We can look to the equation $\mathbf{x} = cP\mathbf{x}$,
which corresponds to the simplified PageRank definition.

teleportation

To model the likelihood that a surfer jumps out of a loop:

- With probability α , in the next move the surfer follows one outedge at page j .
- With probability $1 - \alpha$, the surfer *teleports* to any other page according to the distribution vector \mathbf{v} .

Then we arrive at the equation

$$\mathbf{x} = \alpha P\mathbf{x} + (1 - \alpha)\mathbf{v}.$$

The constant α is called the *teleportation parameter*.

the PageRank problem

Definition (the PageRank problem)

Let P be a stochastic matrix:

- 1 all entries of P are nonnegative, and
- 2 each column of P sums up to one.

Let \mathbf{v} be a column stochastic vector and α the teleportation constant. Then *the PageRank vector \mathbf{R}* satisfies

$$(I - \alpha P)\mathbf{x} = (1 - \alpha)\mathbf{v},$$

where I is the identity matrix of the same dimension as P .

This reduces the PageRank problem to solving a linear system.

relation with eigenvectors

Summing up the entries in the column stochastic vector \mathbf{v} :

$$\mathbf{e}^T = [1, 1, \dots, 1], \quad \mathbf{e}^T \mathbf{v} = 1.$$

Then we rewrite $(I - \alpha P)\mathbf{x} = (1 - \alpha)\mathbf{v}$ as

$$\begin{aligned}\mathbf{x} &= \alpha P\mathbf{x} + (1 - \alpha)\mathbf{v}, \\ &= (\alpha P + (1 - \alpha)\mathbf{v}\mathbf{e}^T)\mathbf{x}.\end{aligned}$$

In the rewrite we applied $\mathbf{e}^T \mathbf{x} = 1$,

as the PageRank vector is a column stochastic vector.

an exercise

Exercise 3: For the matrix A in the 4-page web example, consider the linear system

$$(I - \alpha A)\mathbf{x} = (1 - \alpha)\mathbf{v},$$

with $\alpha = 0.15$ and $v_i = 1/4$, $i = 1, 2, 3, 4$.

Do the following:

- 1 Solve the linear system.
Is the solution a column stochastic vector?
- 2 Compare the new rankings with the ones from before.
Does page 1 still score highest?

Face Recognition and Ranking Data

1 Face Recognition

- problem and dimension
- machine learning
- covariances and eigenfaces

2 Ranking Data

- engineering a search engine
- the random surfer model
- **beyond PageRank**

beyond PageRank

The mathematics of PageRank apply to any graph or network and the ideas occur in a wide range of applications:

- GeneRank and ProteinRank in bioinformatics
- IsoRank in ranking graph isomorphisms
- PageRank of the Linux kernel
- Roads and urban spaces
- PageRank of winner networks in sports
- BookRank in literature, recommender systems
- AuthorRank in the graph of coauthors
- Paper and citation networks

David F. Gleich: [PageRank Beyond the Web](#).
SIAM Review, vol. 57, no. 3, pages 321–363, 2015.