ENGG5781 Matrix Analysis and Computations
Lecture 0: Overview

Wing-Kin (Ken) Ma

2023–24 First Term

Department of Electronic Engineering
The Chinese University of Hong Kong

Course Information

General Information

- Instructor: Wing-Kin Ma
  - office: SHB 323
  - e-mail: wkma@ee.cuhk.edu.hk
- Lecture hours and venue:
  - Monday 1:30pm–3:15pm, Humanities Building 12
  - Monday 4:30pm–5:15pm, Fung King Hey Building Swire Hall 2
- Course website: http://www.ee.cuhk.edu.hk/~wkma/engg5781
  - course notes
- Blackboard: https://blackboard.cuhk.edu.hk/
  - assignments, assignment submissions, scores

Course Contents

- This is a foundation course on matrix analysis and computations, which are widely used in many different fields, e.g.,

  machine learning, artificial intelligence, computer vision, informal retrieval, systems and control, signal and image processing, communications, networks, optimization, data science, and many more...

- **Aim:** covers matrix analysis and computations at an advanced or research level.

- **Scope:**
  - basic matrix concepts, subspace, norms,
  - linear least squares, pseudo-inverse,
  - eigen and singular value decompositions, positive semidefinite matrices,
  - linear system of equations, LU decomposition, Cholesky decomposition,
  - QR decomposition,
  - advanced topics such as tensor decomposition, advanced matrix calculus, sparse recovery, non-negative matrix factorization

Learning Resources

- Notes by the instructor will be provided.
- Recommended readings:
  - Gene H. Golub and Charles F. van Loan, *Matrix Computations* (Fourth Edition), John Hopkins University Press, 2013.
  - Roger A. Horn and Charles R. Johnson, *Matrix Analysis* (Second Edition), Cambridge University Press, 2012.
  - Jan R. Magnus and Heinz Neudecker, *Matrix Differential Calculus with Applications in Statistics and Econometrics* (Third Edition), John Wiley and Sons, New York, 2007.
  - Giuseppe Calafiore and Laurent El Ghaoui, *Optimization Models,* Cambridge University Press, 2014.

Assessment and Academic Honesty

- Assessment:
  - Assignments: 60%
    - where to submit: online by Blackboard
    - no late submissions would be accepted, except for exceptional cases.
  - Final examination: 40%

- Students are required to read
  - Homework guideline, assessment scheme and appeal policy: Find it on Blackboard.
  - the University's guideline on academic honesty: http://www.cuhk.edu.hk/policy/academichonesty

  By taking this course, you are assumed to have read and understand the aspects described therein.

Additional Notice

- Course helpers whom you can consult:
  - Yuening Li, yuening@link.cuhk.edu.hk
  - Junbin Liu, liujunbin@link.cuhk.edu.hk
  - Yusheng Tian, ystian0617@link.cuhk.edu.hk

- Regularly check your CUHK Link e-mail! It's the only way we can reach you

A Glimpse of Topics

Linear Systems, Least Squares (LS), and More

- **Problem:** Let $\mathbf{A} \in \mathbb{R}^{m \times n}$, $\mathbf{y} \in \mathbb{R}^n$ be given. Find $\mathbf{x} \in \mathbb{R}^n$ such that $\mathbf{y} = \mathbf{A}\mathbf{x}$, or $\mathbf{A}\mathbf{x}$ best approximates $\mathbf{y}$.

- if $m = n$, then we will do $\mathbf{x} = \mathbf{A}^{-1}\mathbf{y}$ (you need to assume $\mathbf{A}$ is nonsingular)

- if $\mathbf{A}$ is tall, i.e., $m \geq n$, we may do LS

$$\min_{\mathbf{x} \in \mathbb{R}^n} \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2,$$

  where $\| \cdot \|_2$ is the Euclidean norm; i.e., $\|\mathbf{x}\|_2 = \sqrt{\sum_{i=1}^n |x_i|^2}$. The solution is

$$\mathbf{x}_{\text{LS}} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{y}$$

  in which $(\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T$ is like the inverse

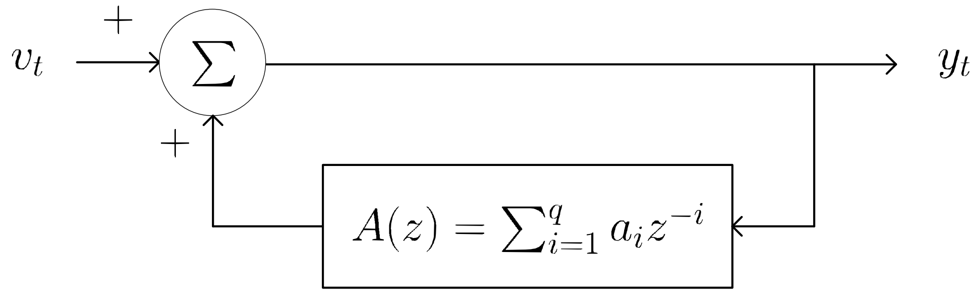
- widely used in science, engineering, and mathematics

Application Example: Linear Prediction (LP)

- let $\{y_t\}_{t \geq 0}$ be a time series.
- **Model** (autoregressive (AR) model):

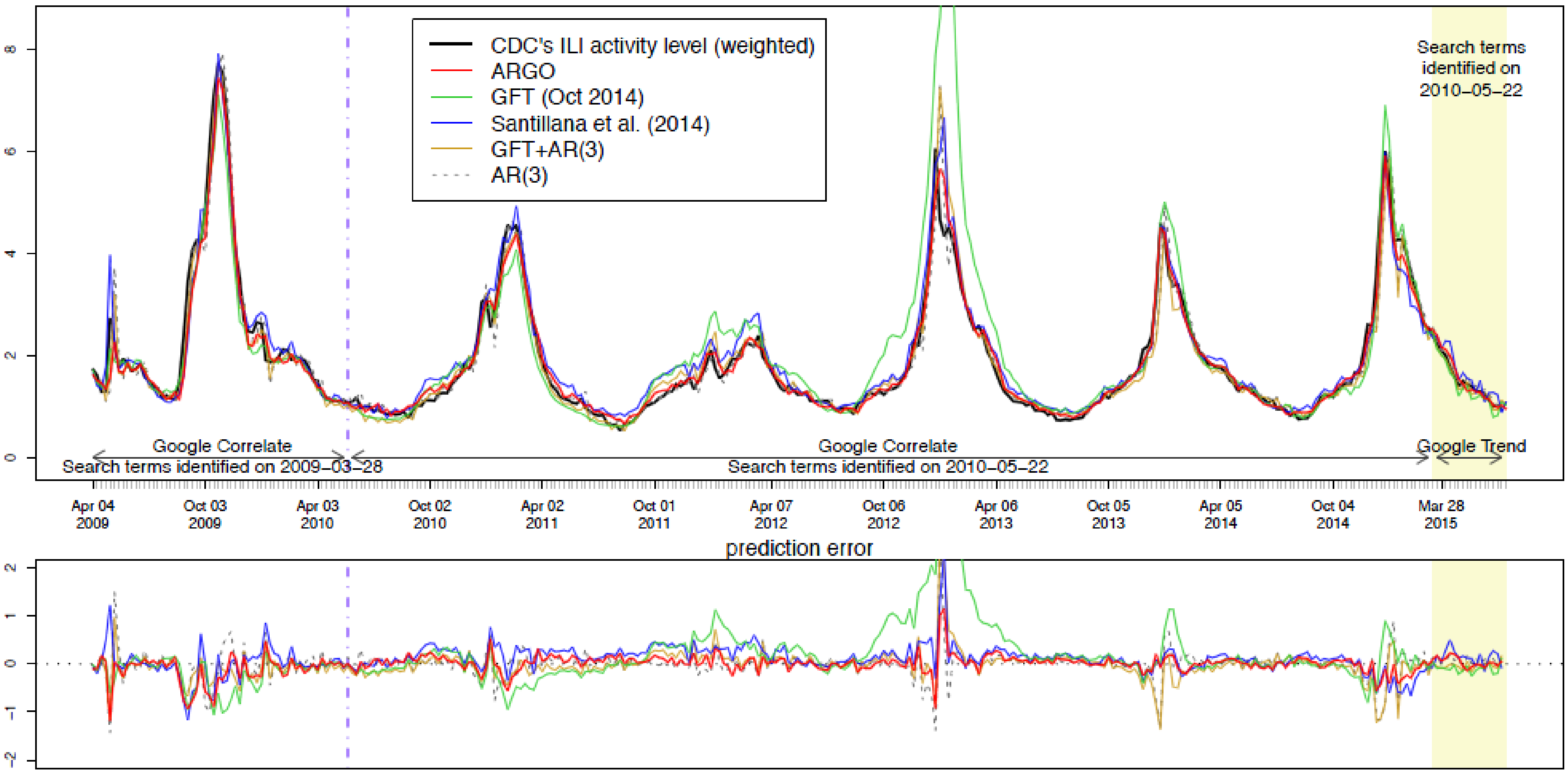
$$y_t = a_1 y_{t-1} + a_2 y_{t-2} + \cdots + a_q y_{t-q} + v_t, \quad t = 0, 1, 2, \ldots$$

for some coefficients $\{a_i\}_{i=1}^q$, where v_t is noise or modeling error.



- **Problem:** estimate $\{a_i\}_{i=1}^q$ from $\{y_t\}_{t \geq 0}$; can be formulated as LS
- **Applications:** time-series prediction, speech analysis and coding, spectral estimation...

A Real Application of LP: Real-Time Prediction of Flu Activity

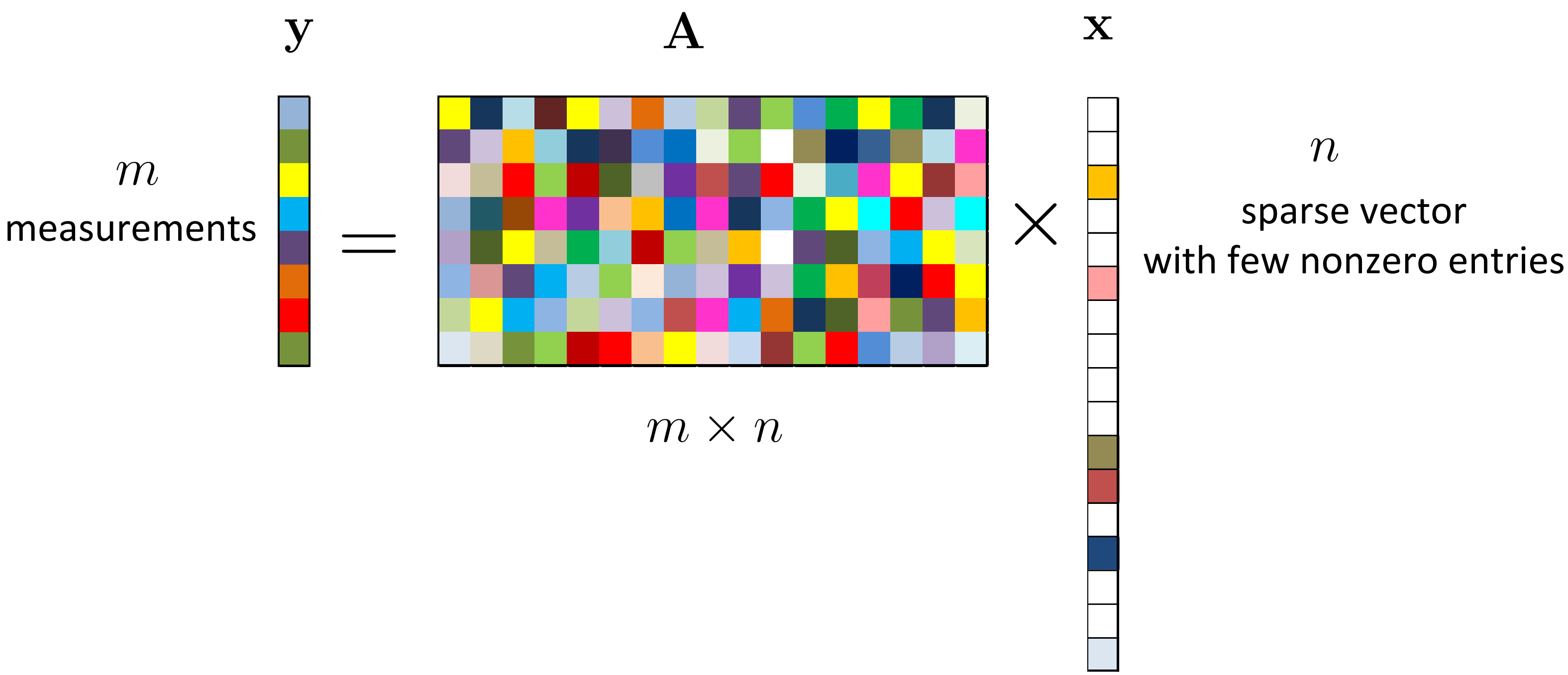


Tracking influenza outbreaks by ARGO — a model combining the AR model and Google search data.
Source: **[Yang-Santillana-Kou2015]**.

Advanced Topic: Sparse Recovery

- if $\mathbf{A}$ is fat, i.e., $m < n$, then $\mathbf{y} = \mathbf{A}\mathbf{x}$ will have infinitely many solutions for $\mathbf{x}$

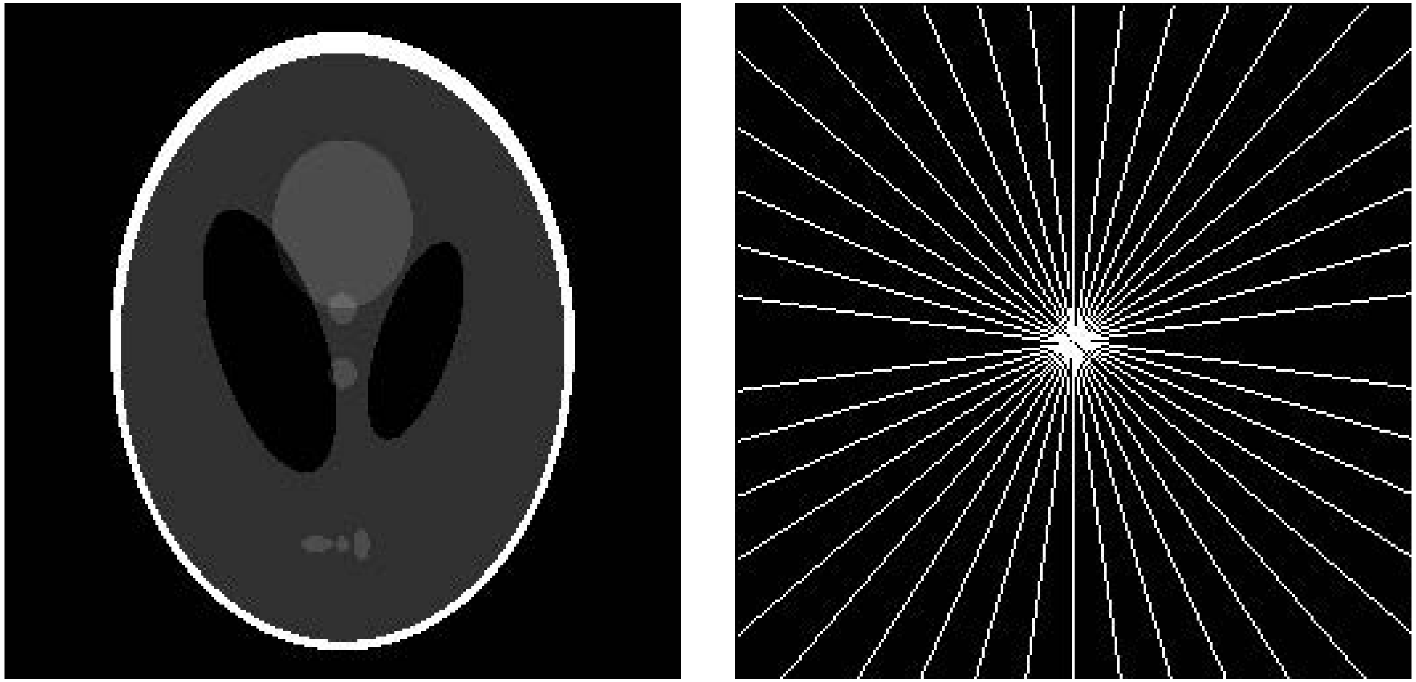
- **Problem:** find a sparsest $\mathbf{x} \in \mathbb{R}^n$ such that $\mathbf{y} = \mathbf{A}\mathbf{x}$



- by sparsest, we mean that $\mathbf{x}$ should have as many zero elements as possible

Application: Magnetic resonance imaging (MRI)

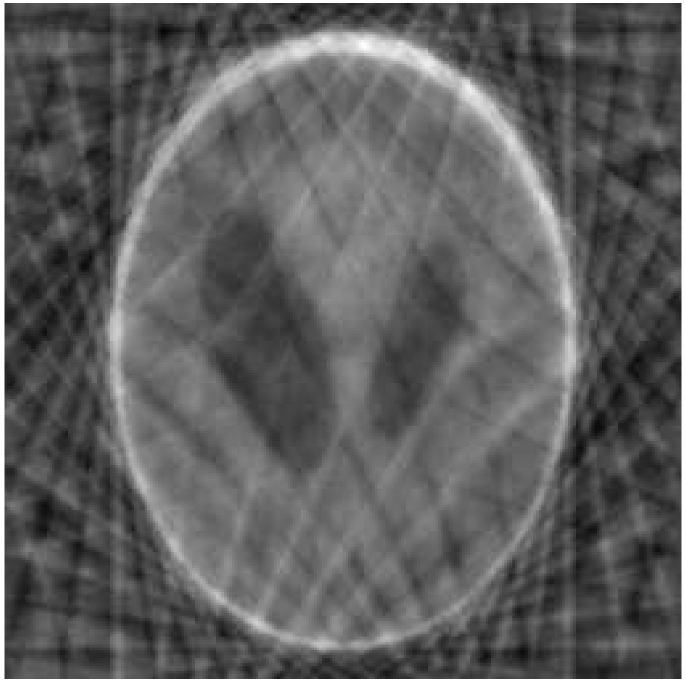
Problem: MRI image reconstruction.



(a) (b)

Fig. a shows the original test image. Fig. b shows the sampling region in the frequency domain. Fourier coefficients are sampled along 22 approximately radial lines. Source: **[Candès-Romberg-Tao2006]**

Application: Magnetic resonance imaging (MRI)



(c)

(d)

Fig. c is the recovery by filling the unobserved Fourier coefficients to zero. Fig. d is the recovery by a sparse recovery solution. Source: **[Candès-Romberg-Tao2006]**

Eigenvalue, Eigendecomposition, Singular Value Decomposition

- **Eigenvalue problem:** Let $\mathbf{A} \in \mathbb{R}^{n \times n}$ be given. Find a vector $\mathbf{v}$ such that

$$\mathbf{A}\mathbf{v} = \lambda \mathbf{v}, \qquad \text{for some } \lambda.$$

- **Eigendecomposition:** Let $\mathbf{A} \in \mathbb{R}^{n \times n}$ be given. Decompose

$$\mathbf{A} = \mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^{-1}, \qquad \text{for some square } \mathbf{V} \text{ and } \boldsymbol{\Lambda} = \mathrm{Diag}(\lambda_1, \ldots, \lambda_n)$$

- **Singular value decomposition (SVD):** Let $\mathbf{A} \in \mathbb{R}^{m \times n}$ be given. Decompose

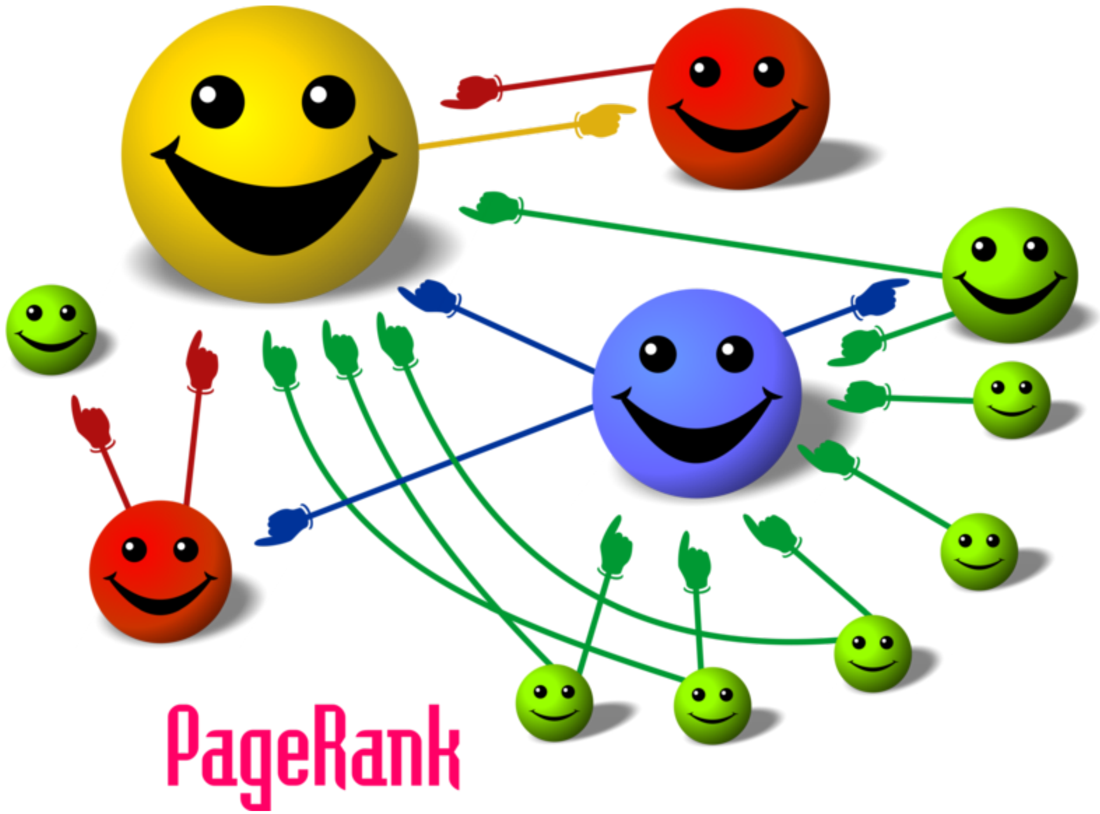
$$\mathbf{A} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T$$

  where $\mathbf{U} \in \mathbb{R}^{m \times m}, \mathbf{V} \in \mathbb{R}^{n \times n}$ are orthogonal; $\boldsymbol{\Sigma} \in \mathbb{R}^{m \times n}$ takes a diagonal form

- also widely used in science and engineering: PageRank, dimensionality reduction, PCA, extracting meaningful features from data, low-rank modeling, . . .

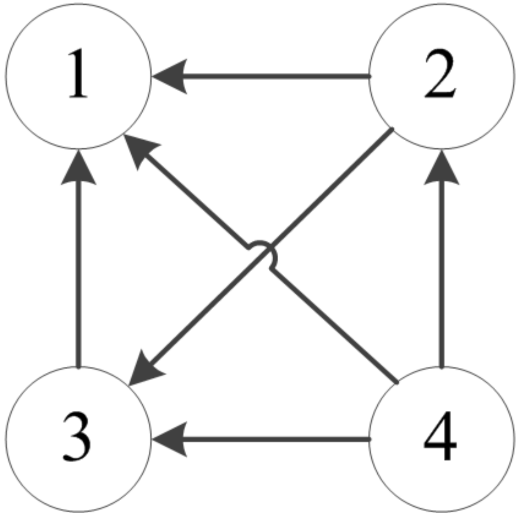
Application Example: PageRank

- PageRank is an algorithm used by Google to rank the pages of a search result.
- the idea is to use counts of links of various pages to determine pages' importance.



Source: Wiki.

One-Page Explanation of How PageRank Works



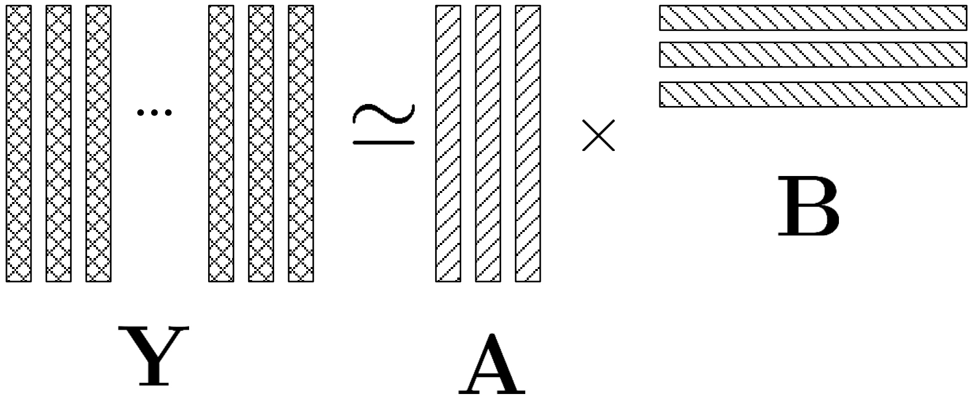
- Model:
$$\sum_{j \in \mathcal{L}_i} \frac{v_j}{c_j} = v_i, \quad i = 1, \ldots, n,$$
where c_j is the number of outgoing links from page j; $\mathcal{L}_i$ is the set of pages with a link to page i; v_i is the importance score of page i.

- as an example,
$$\overbrace{\begin{bmatrix} 0 & \frac{1}{2} & 1 & \frac{1}{3} \\ 0 & 0 & 0 & \frac{1}{3} \\ 0 & \frac{1}{2} & 0 & \frac{1}{3} \\ 0 & 0 & 0 & 0 \end{bmatrix}}^{\mathbf{A}} \overbrace{\begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix}}^{\mathbf{v}} = \overbrace{\begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix}}^{\mathbf{v}}.$$

- finding $\mathbf{v}$ is an eigenvalue problem—with n being of order of millions!
- further reading: **[Bryan-Tanya2006]**

Application Example: Low-Rank Matrix Approximation

- **Problem:** given $\mathbf{Y} \in \mathbb{R}^{m \times n}$ and an integer $r < \min\{m, n\}$, find an $(\mathbf{A}, \mathbf{B}) \in \mathbb{R}^{m \times r} \times \mathbb{R}^{r \times n}$ such that either $\mathbf{Y} = \mathbf{AB}$ or $\mathbf{Y} \approx \mathbf{AB}$.



- **Formulation:**

$$\min_{\mathbf{A} \in \mathbb{R}^{m \times r}, \mathbf{B} \in \mathbb{R}^{r \times n}} \|\mathbf{Y} - \mathbf{AB}\|_F^2,$$

where $\|\cdot\|_F$ is the Frobenius, or matrix Euclidean, norm.

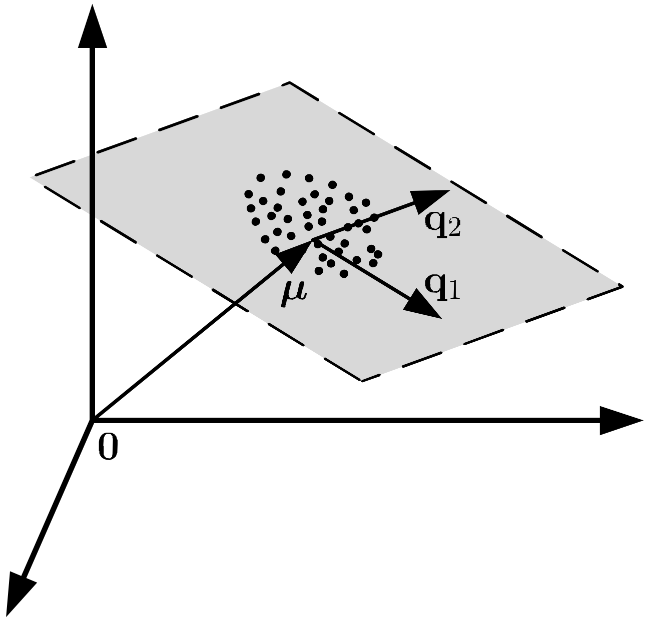
- can be solved by SVD

Application: Principal Component Analysis (PCA)

- **Aim:** given a set of data points $\{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_n\} \subset \mathbb{R}^n$ and an integer $k < \min\{m, n\}$, perform a low-dimensional representation

$$\mathbf{y}_i = \mathbf{Q}\mathbf{c}_i + \boldsymbol{\mu} + \mathbf{e}_i, \quad i = 1, \ldots, n,$$

where $\mathbf{Q} \in \mathbb{R}^{m \times k}$ is a basis; $\mathbf{c}_i$'s are coefficients; $\boldsymbol{\mu}$ is a base; $\mathbf{e}_i$'s are errors

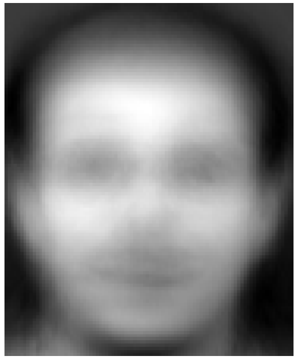


- the problem can be formulated as a low-rank matrix approximation problem

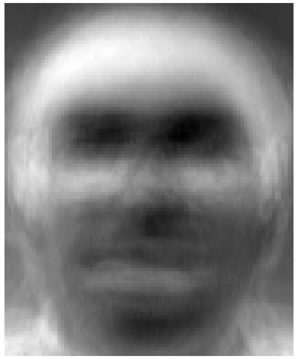
Toy Demo: Dimensionality Reduction of a Face Image Dataset

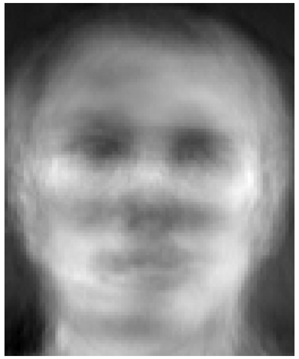
A face image dataset. Image size $= 112 \times 92$, number of face images $= 400$. Each $\mathbf{x}_i$ is the vectorization of one face image, leading to $m = 112 \times 92 = 10304$, $n = 400$.

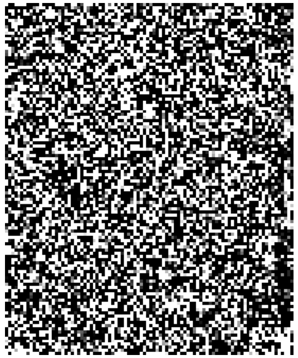
Toy Demo: Dimensionality Reduction of a Face Image Dataset

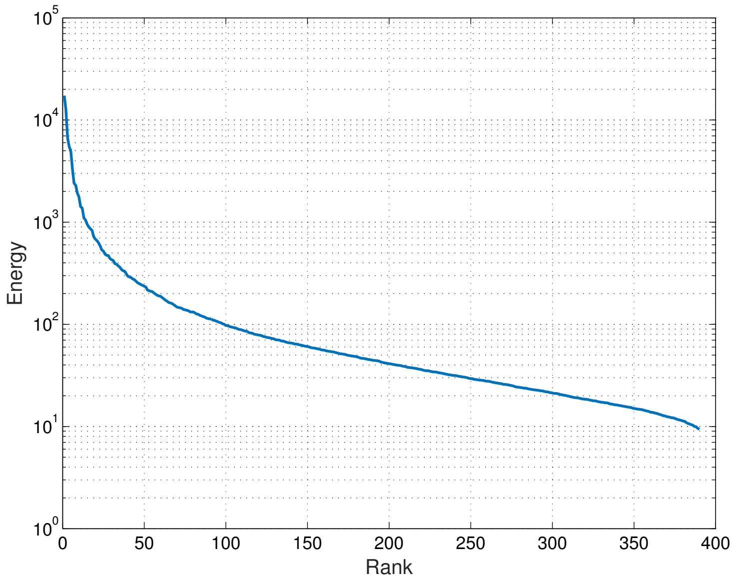

Mean face


1st principal left singular vector


2nd principal left singular vector


3rd principal left singular vector


400th left singular vector



Energy Concentration

Advanced Topic: Nonnegative Matrix Factorization (NMF)

- **Aim:** we want the factors to be non-negative
- **Formulation:**

$$\min_{\mathbf{A} \in \mathbb{R}^{m \times r}, \mathbf{B} \in \mathbb{R}^{r \times n}} \|\mathbf{Y} - \mathbf{A}\mathbf{B}\|_F^2 \quad \text{s.t. } \mathbf{A} \geq \mathbf{0}, \mathbf{B} \geq \mathbf{0},$$

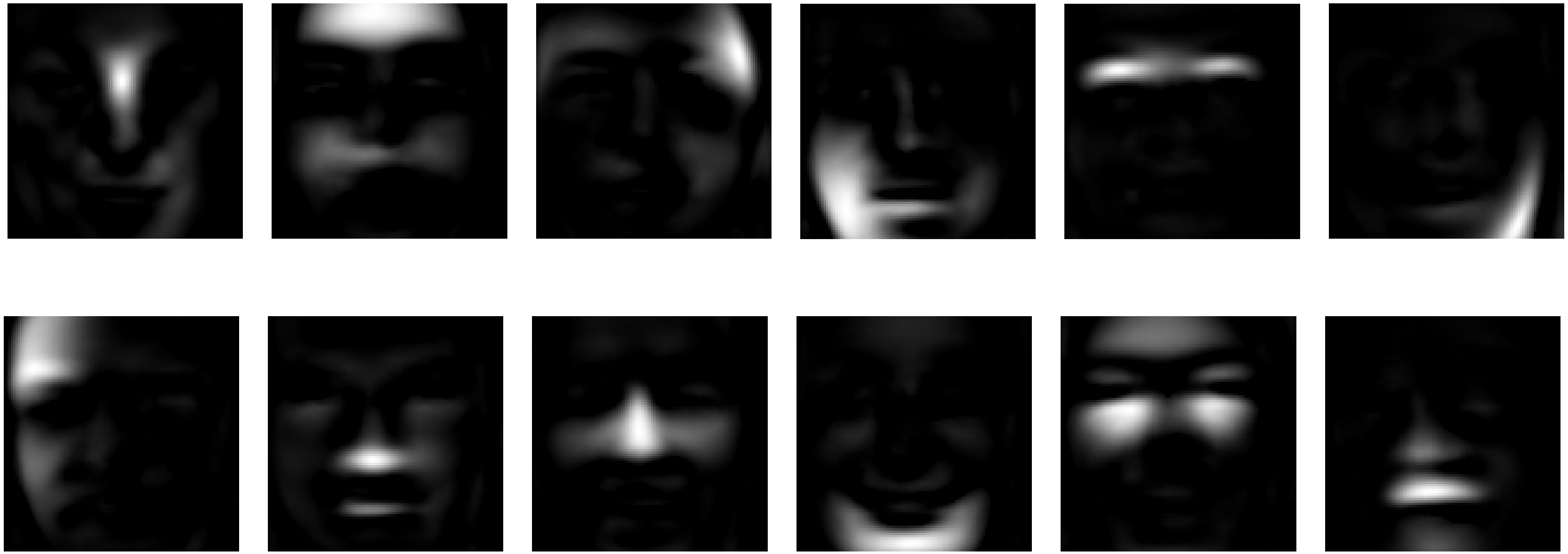
where $\mathbf{X} \geq \mathbf{0}$ means that $x_{ij} \geq 0$ for all i, j.

- arguably a topic in optimization
- found to be able to extract meaningful features (by empirical studies)
- numerous applications, e.g., in machine learning, signal processing, remote sensing

Toy Demonstration of NMF

A face image dataset. Image size $= 101 \times 101$, number of face images $= 13232$. Each $\mathbf{x}_i$ is the vectorization of one face image, leading to $m = 101 \times 101 = 10201$, $n = 13232$.

Toy Demonstration of NMF: NMF-Extracted Features



NMF settings: $r = 49$, Lee-Seung multiplicative update with 5000 iterations.

Linear System of Equations

- **Problem:** given $\mathbf{A} \in \mathbb{R}^{n \times n}$, $\mathbf{y} \in \mathbb{R}^n$, solve

$$\mathbf{A}\mathbf{x} = \mathbf{y}.$$

- **Question 1:** How to solve it?
  - don't tell me answers like `x=inv(A)*y` or `x= A\y` on MATLAB!
  - this is about matrix computations

- **Question 2:** How to solve it when n is very large?
  - it's too slow to do the generic trick `x= A\y` when n is very large
  - getting better understanding of matrix computations will enable you to exploit problem structures to build efficient solvers

Why Matrix Analysis and Computations is Important?

- as said, areas such as signal processing, image processing, machine learning, optimization, computer vision, control, communications, . . ., use matrix operations extensively

- it helps you build the foundations for understanding "hot" topics such as
  - sparse recovery;
  - matrix completion; non-negative matrix factorization; structured low-rank matrix approximation

A Few More Words to Say

- things I hope you will learn
  - how to read how people manipulate matrix operations, and how you can manipulate them (learn to use a tool);
  - what applications we can do, or to find new applications of our own (learn to apply a tool);
  - deep analysis skills (why is this tool valid, and how can I invent new tools?)

- through the course I also hope you will learn how to be a good thinker: greater researchers invent great things by questioning the unquestionable

- feedbacks are welcome!

References

[Yang-Santillana-Kou2015] S. Yang, M. Santillana, and S. C. Kou, "Accurate estimation of influenza epidemics using Google search data via ARGO," *Proceedings of the National Academy of Sciences*, vol. 112, no. 47, pp. 14473–14478, 2015.

[Candès-Romberg-Tao2006] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Information Theory*, vol. 52, no. 2, pp. 489–509, 2006.

[Bryan-Tanya2006] K. Bryan and L. Tanya, "The $25,000,000,000$ eigenvector: The linear algebra behind Google," *SIAM Review*, vol. 48, no. 3, pp. 569–581, 2006.

[Lee-Seung1999] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, pp. 788–791, 1999.