STA 141C - Big Data & High Performance Statistical Computing

Bo Y.-C. Ning

2022 Spring Quarter @ UC Davis

March 29, 2022

Source: [see this website]

Introduction

What is Big Data?

Perhaps for statisticians, big data means inference at large scale:

- We need new methods: for example, in multiple linear regression, when the number of predictors p is larger than the observed sample size n, the ordinary least square method fails as $p \gg n$. One has to invent some new techniques to deal with those types of data. For example, lasso is one of the example
- We need new computational algorithms. Even when $p < n$, the computational speed is significantly slower for solving a problem when $n = 20$ and $n = 20,000$. Thus, we need to 1) understand how the default package solves those types of problems; 2) which tool to use for solving different problems in terms of making the computational faster
- In this course, we focus on the second part. You can learn the first aspect from a machine learning/data mining course.

What is this course NOT about?

- This course is **not** about teaching you how to write R/python programming code; you are assumed to be capable of using one of those languages (or at least know how to learn programming by yourself)
- This course is **not** about teaching you how to fit packages

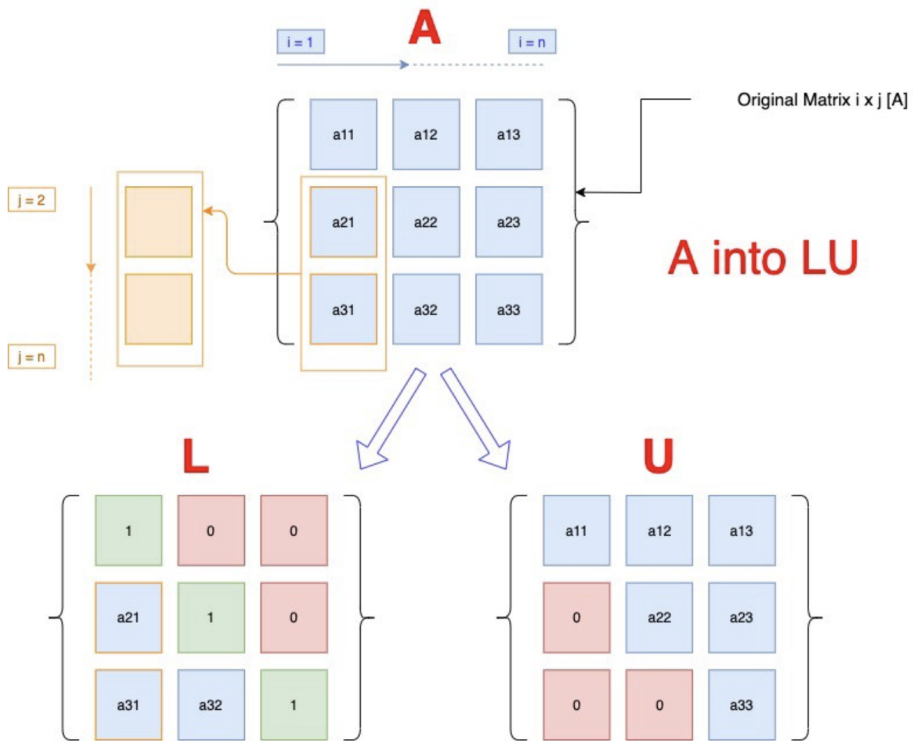
What is the course about then?

- One big component of the course is numeric linear algebra (we will spend $\sim$ 5 weeks on this topic!)
- Statistical methods often require to solve

$$A\beta = b$$

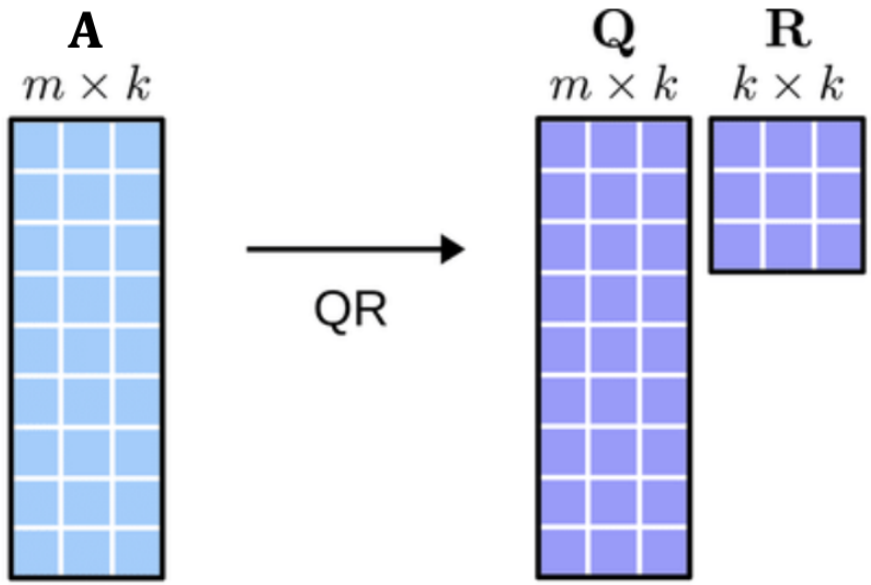
- We will learn different methods to solve β efficiently. Different ways to decompose A will lead to different computational speeds. While this difference may not distinguishable for "small data" but can be dramatic for "big data"
- We will look inside some "default" packages to understand which algorithm do those packages use
- In the homework, you will get a chance to compare the computational speed among different method
- You will also get a chance to implement an algorithm and beat the default package in terms of computational speed!

Example 1: LU decomposition

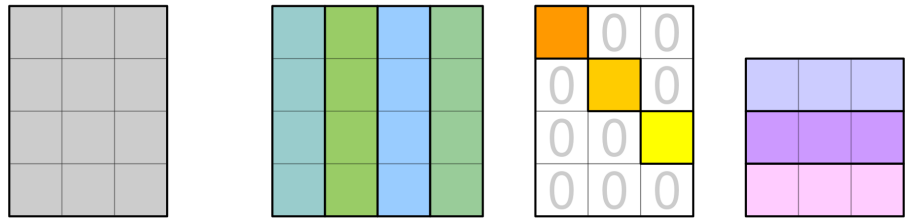


source: `https://towardsdatascience.com/understanding-matrix-factorization-for-recommender-systems-4d3c5e67f2c9`

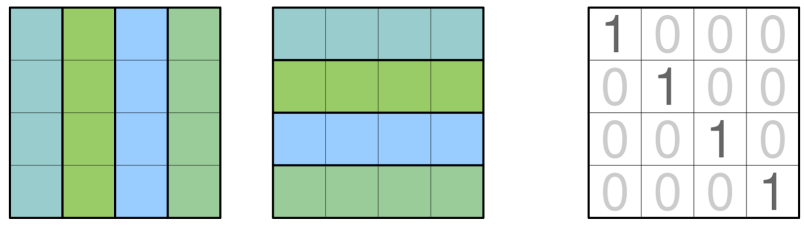
Example 2: QR factorization (decomposition)



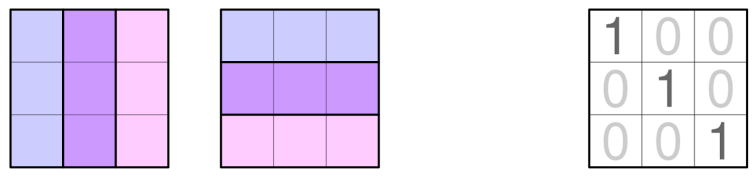
Example 3: SVD (Singular value decomposition)



$$\underset{m\times n}{\mathbf{M}} = \underset{m\times m}{\mathbf{U}}\ \underset{m\times n}{\mathbf{\Sigma}}\ \underset{n\times n}{\mathbf{V}^*}$$



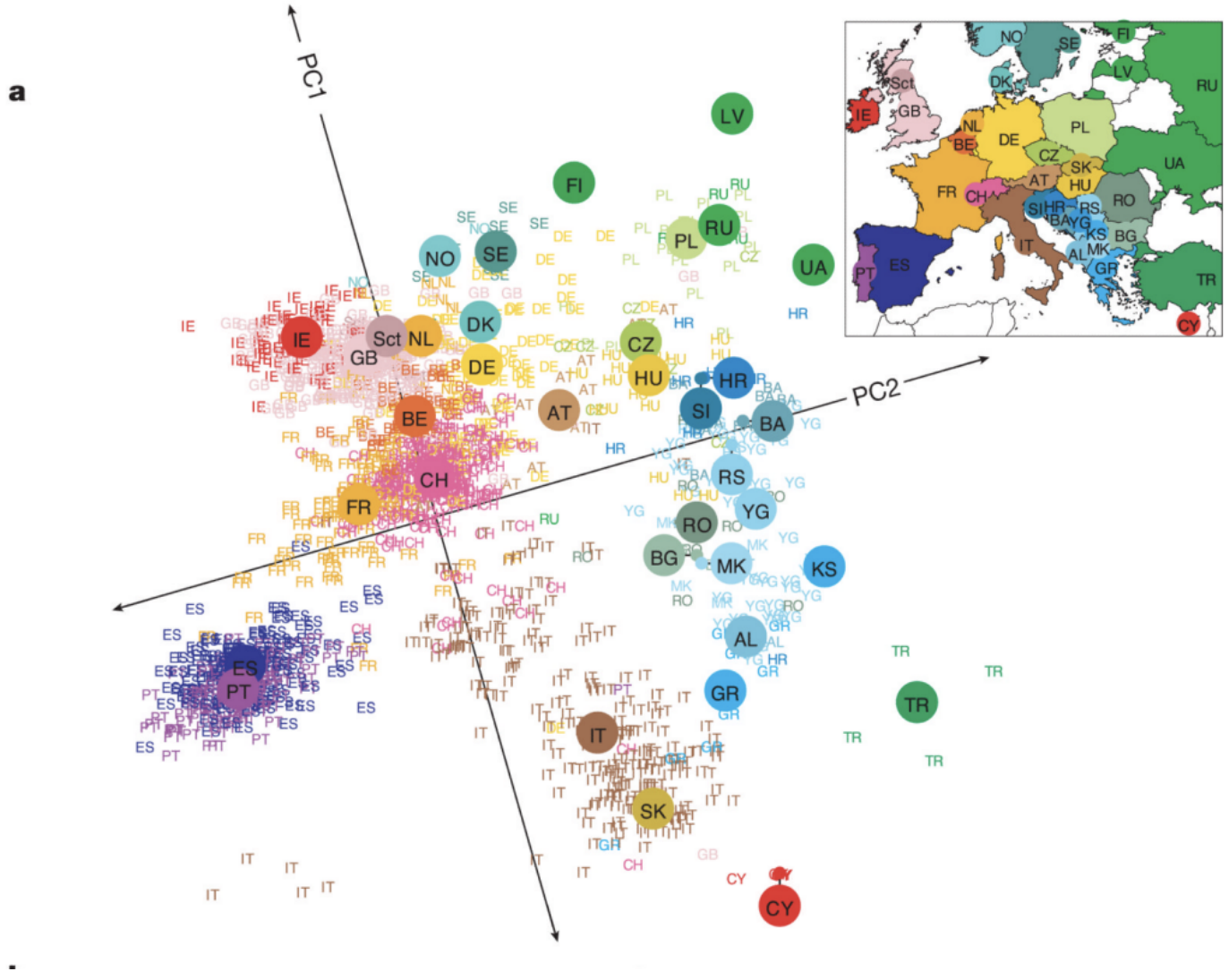
$$\mathbf{U}\ \mathbf{U}^* = \mathbf{I}_m$$



$$\mathbf{V}\ \mathbf{V}^* = \mathbf{I}_n$$

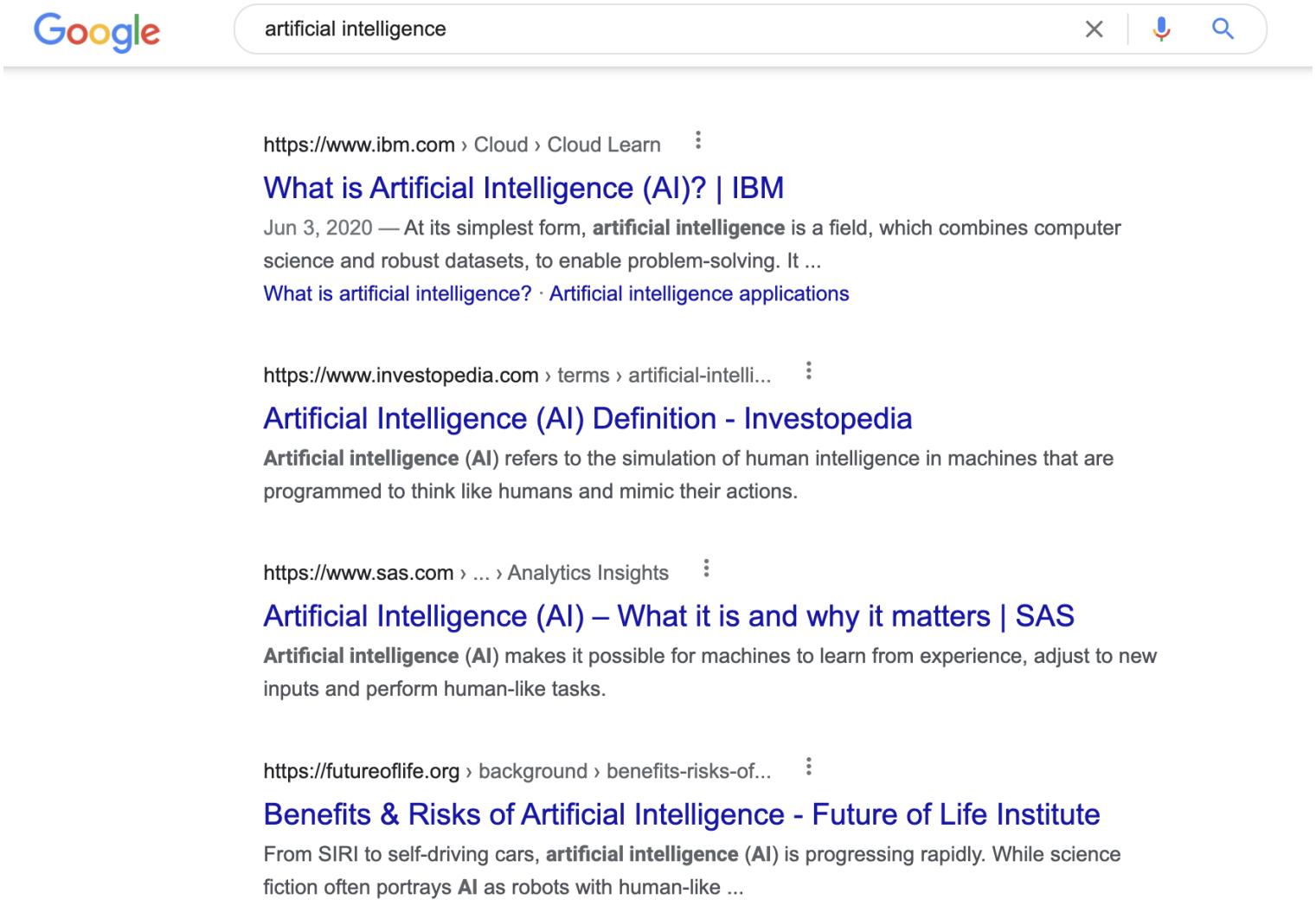
Source: wikipedia

Application: PCA (Principal component analysis)



Source: Genes mirror geography within Europe. *Nature*. 2008

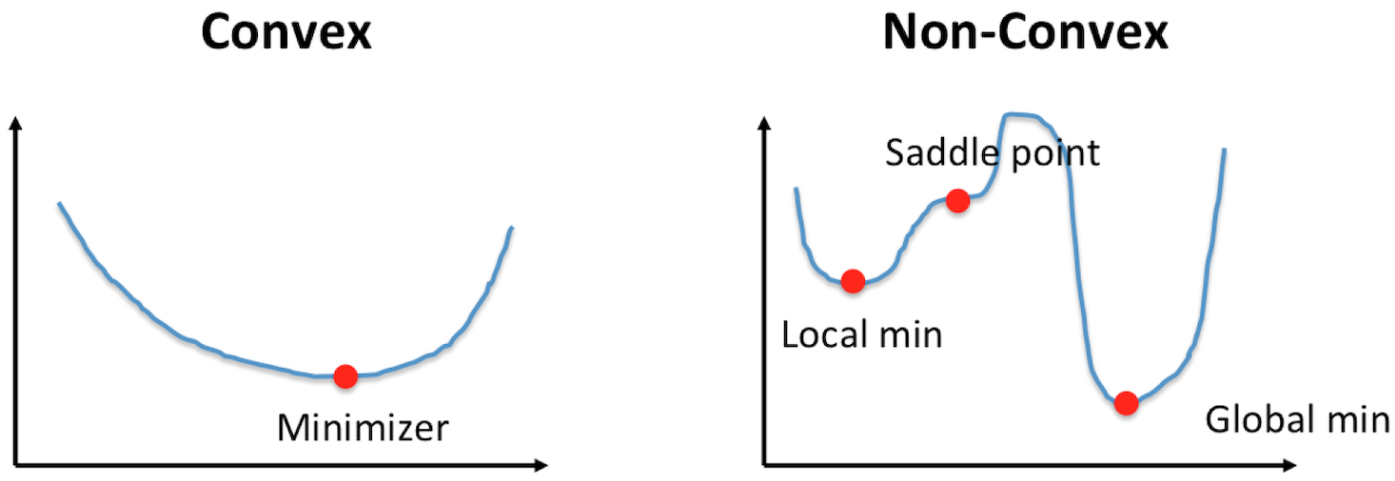
Page rank problem: how do google rank webpages?



- So, you need to know basic linear algebra (of course, I will review many of them, but it is better that you already knew them)
- You also need to know how to write code. For completing homework assignments, you are free to choose R or python.
- You will spend a significant amount of time on homework. I will give hints for each homework in class.
- Anything that is unclear to you, you should always ask or post questions on Piazza.

- The second component in this class is to learn several algorithms for solving statistical tasks for big data

Example 1: Optimization for nonlinear functions



source: `https://www.ibm.com/cloud/learn/gradient-descent`

Some topics I will cover:

- Newton's method
- Gradient descent/Stochastic gradient descent

Application: Neural Networks

When Neural Networks saw the first image of Black Hole.

Anuj shah (Exploring Neurons) Follow
Apr 19, 2019 · 6 min read

Black Hole — M87 — Event Horizon Telescope

source: `https://medium.com/analytics-vidhya/when-neural-networks-saw-the-first-image-of-black-hole-3205e28b6578`

- Sampling based methods in Bayesian analysis

Application: Gaussian Processes

- Transit method for finding exoplanets:
  `https://www.youtube.com/watch?v=xNeRqbw18Jk`
- Stellar activity: `https://www.physics.uu.se/research/astronomy-and-space-physics/research/stars/magnetism/`

- Sampling based methods in Bayesian analysis

Application: Gaussian Processes

- Transit method for finding exoplanets: `https://www.youtube.com/watch?v=xNeRqbw18Jk`
- Stellar activity: `https://www.physics.uu.se/research/astronomy-and-space-physics/research/stars/magnetism/`

IMPROVING EXOPLANET DETECTION POWER: MULTIVARIATE GAUSSIAN PROCESS MODELS FOR STELLAR ACTIVITY

By David E. Jones[1], David C. Stenning[2] Eric B. Ford[3], Robert L. Wolpert[4], Thomas J. Loredo[5], Christian Gilbertson[3], and Xavier Dumusque[6]

Texas A&M University[1], Simon Fraser University[2], Penn State University[3], Duke University[4], Cornell Center for Astrophysics and Planetary Science[5], and Observatoire Astronomique de l'Université de Genève[6]

If time permits, you will also learn

- EM algorithm
- Variational inference
- Bootstrap
- ....

Also, I will cover a lecture on parallel computing.

Algorithms and programming tools can be learned by yourself (you can even learn them on Coursera), but the reasoning for efficient algorithms is difficult for self-study. Numerical linear algebra is used everywhere in big data analysis. That is why we will focus on this topic a lot.

At the end of the course, you are expect to 1) know which algorithm is suitable for your problem; 2) without using the default package, be able to write at least one algorithm on your own.

Announcement

- No office hours this week
- Install Rstudio or the Jupyter Notebook
- If you have any issue, Wei can help you during her discussion sessions tomorrow
- Syllabus