课程概述

王尧

西安交通大学智能决策与机器学习中心
(Email: yao.s.wang@gmail.com)

2022. 4

个人简介

- 教育经历
  - 电子科技大学信息与计算科学学士
  - 西安交通大学应用数学硕士、博士(导师：徐宗本院士)
  - 西安交通大学管理科学与工程博士后(合作导师：廖貅武教授)
- 工作经历
  - 美国佐治亚理工学院工业与系统工程系联合培养博士
  - 西安交通大学统计系讲师、副教授
  - 西安交通大学信息系统与智能商务系副教授、博士生导师(2018.11至今)
- 研究兴趣
  - 机器学习与运筹优化的融合交叉(如大规模混合整数规划的求解)
  - 动态环境下的序列决策(如动态推荐与选品优化)
  - 高维数据分析(如图像与视频分析)

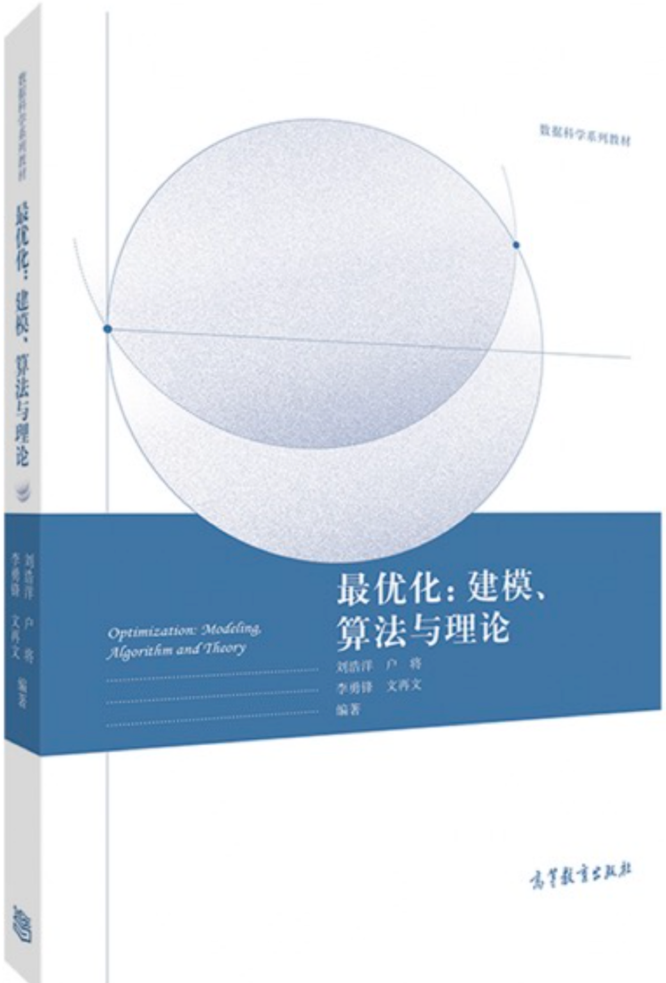
课程要求

- 课程主要目标
  - 共32学时
  - 掌握最优化的一些基本方法与理论
  - 会用程序语言(R,Python或matlab)编程求解一些典型的最优化问题
- 学习形式
  - 组成学习小组，每组2-4人
  - 除了教师授课以外，课后需阅读文献以及完成作业与实验课题
- 考核方式
  - 程序实验报告：2-3份(小组形式)，占比20%
  - 课程作业：平均每周一次+不定期随堂，占比20%
  - 闭卷考试：期末进行，占比60%

课程主要内容

- 基本概念介绍
  - 什么是最优化问题?
  - 凸集与凸函数
  - 一些数学基础
- 一阶优化方法
  - 梯度下降方法
  - 次梯度与次梯度方法
  - 近似点梯度方法
- 对偶与最优性理论
  - 对偶理论
  - KKT条件
- 二阶优化方法
  - 牛顿法
  - 拟牛顿法

主要参考书籍



数据科学系列教材
最优化：建模、算法与理论
Optimization: Modeling, Algorithm and Theory
刘浩洋 户 将
李勇锋 文再文
编著
高等教育出版社

数据科学与大数据技术专业
［理科］系列教材
最优化计算方法
刘浩洋 户 将
李勇锋 文再文
编著
高等教育出版社

主要参考书籍

教材下载/购买信息

- 下载“最优化：建模、算法与理论”第一版作者版本
- 下载“最优化：建模、算法与理论”第二版草稿，主要增加了流形约束优化算法和半光滑牛顿算法等新内容
- 下载“最优化计算方法”第一版作者版本
- “最优化：建模、算法与理论”第一版勘误表
- “最优化：建模、算法与理论” 第一版出版社版本购买链接如下。出版社版本和作者版本在排版上有较大的区别。
  - 高教书城（已经可以购买）：www.hepmall.com/index.php/product-34415.html
  - 微店的地址（已经可以手机端购买）：weidian.com/item.html?itemID=4276223598
  - 京东自营链接（已经可以购买）：item.jd.com/13064530.html
  - 当当自营链接（已经可以购买）：product.dangdang.com/29185022.html
- “最优化计算方法” 第一版出版社版本购买链接如下。出版社版本和作者版本在排版上有较大的区别。
  - 高教书城（已经可以购买）：www.hepmall.com/index.php/product-35610.html
  - 微店的地址（已经可以手机端购买）：weidian.com/item.html?itemID=4388758095
  - 京东自营链接（已经可以购买）：item.jd.com/13318228.html
  - 当当自营链接（已经可以购买）：product.dangdang.com/29261079.html
- 读者调查问卷：感谢您对本书的兴趣！如果本书对您有帮助，请您帮忙反馈一些用户信息供作者参考。
- 限于作者的知识水平，书中难免有不妥和错误之处，恳请读者不吝批评和指正。
  勘误收集: 如果您发现任何不妥和错误之处，麻烦您通过此链接反馈。
  2021年6月底左右出版社将给提供反馈的老师和同学挂号信方式邮寄 “最优化计算方法”样书。

https://bicmr.pku.edu.cn/~wenzw/optbook.html

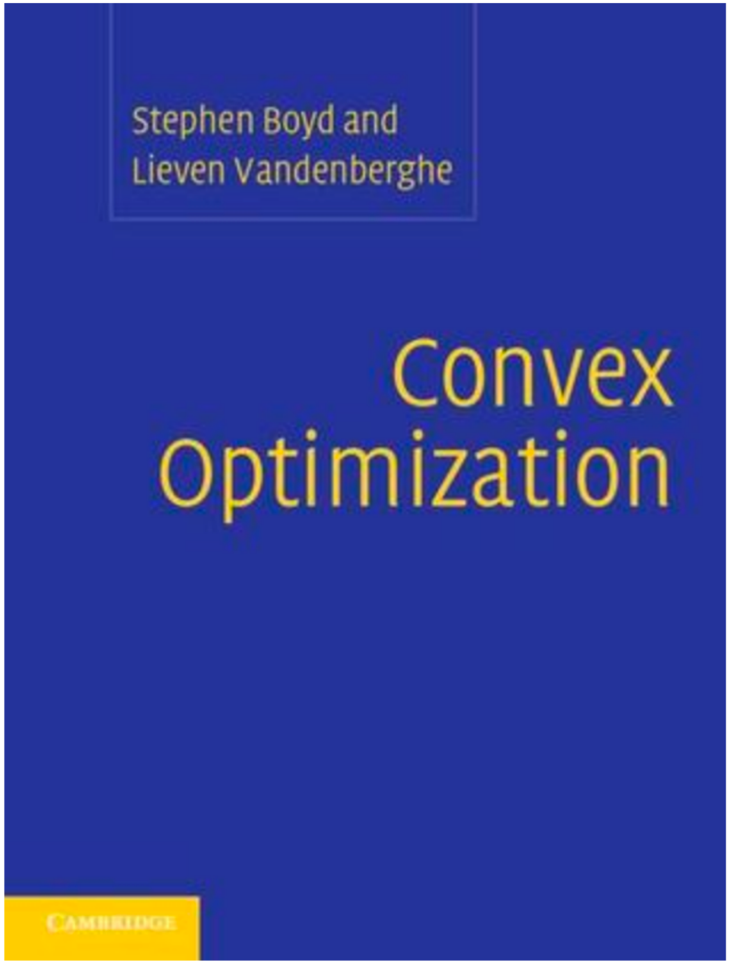
演示程序

- 梯度类算法
  - LASSO 问题的梯度下降法
  - 实例：利用梯度法解 LASSO 问题
  - LASSO 连续化策略（此算法应用在梯度下降以及若干其它算法中）
  - BB 步长梯度下降法
  - 实例：Tikhonov 正则化模型用于图片去噪
- 次梯度算法
  - LASSO 问题的次梯度解法
  - 实例：次梯度法解 LASSO 问题
  - LASSO 问题的连续化次梯度法
  - 实例：连续化次梯度法解 LASSO 问题
- 牛顿类算法
  - 牛顿-共轭梯度法
  - 实例：牛顿-共轭梯度法解逻辑回归问题
- 拟牛顿类算法
  - L-BFGS 算法
  - 实例：L-BFGS算法解基追踪问题
  - 实例：L-BFGS算法解逻辑回归问题

https://bicmr.pku.edu.cn/~wenzw/optbook/pages/contents/contents.html

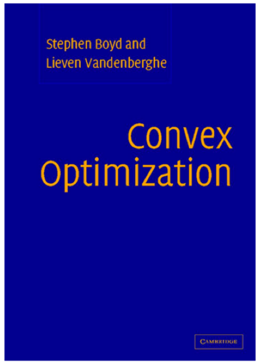
其他参考书籍


计算方法丛书
最优化理论与方法
袁亚湘 孙文瑜 著
科学出版社
www.sciencep.com


Stephen Boyd and
Lieven Vandenberghe
Convex Optimization
CAMBRIDGE

其他参考书籍

Convex Optimization – Boyd and Vandenberghe



Convex Optimization
Stephen Boyd and Lieven Vandenberghe

Cambridge University Press

A MOOC on convex optimization, CVX101, was run from 1/21/14 to 3/14/14. If you register for it, you can access all the course materials.

More material can be found at the web sites for EE364A (Stanford) or EE236B (UCLA), and our own web pages. Source code for almost all examples a
CVXOPT (in the book examples directory), and in CVXPY. Source code for examples in Chapters 9, 10, and 11 can be found here. Instructors can obtai
the course you are teaching.

If you find an error not listed in our errata list, please do let us know about it.

Stephen Boyd & Lieven Vandenberghe

Download

Copyright in this book is held by Cambridge University Press, who have kindly agreed to allow us to keep the book available on the web.

- Book
- Lecture slides
- Additional exercises; data files

https://web.stanford.edu/~boyd/cvxbook/

必备工具书

Introduction

What is this? These pages are a collection of facts (identities, approximations, inequalities, relations, ...) about matrices and matters relating to them. It is collected in this form for the convenience of anyone who wants a quick desktop reference .

Disclaimer: The identities, approximations and relations presented here were obviously not invented but collected, borrowed and copied from a large amount of sources. These sources include similar but shorter notes found on the internet and appendices in books - see the references for a full list.

Errors: Very likely there are errors, typos, and mistakes for which we apologize and would be grateful to receive corrections at cookbook@2302.dk.

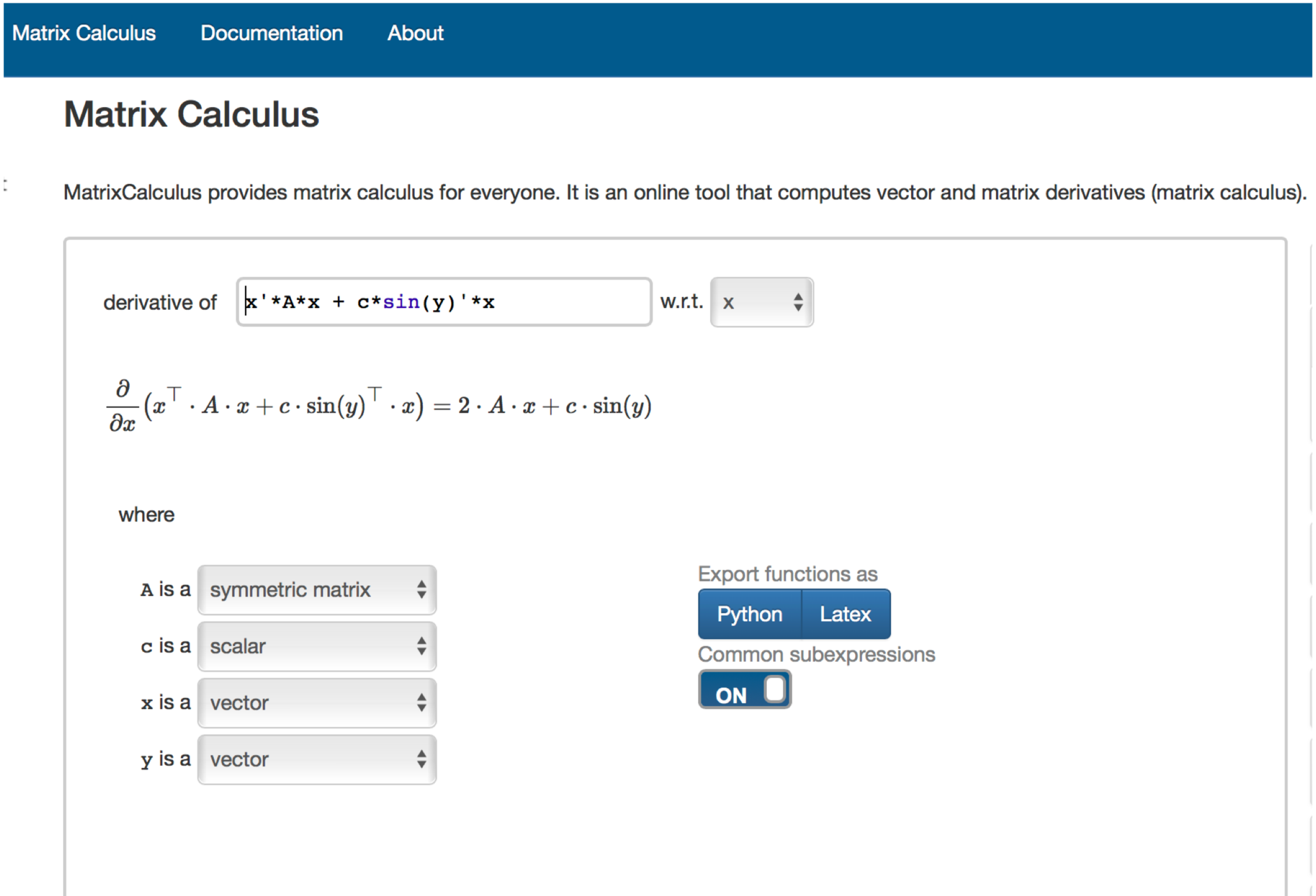
Its ongoing: The project of keeping a large repository of relations involving matrices is naturally ongoing and the version will be apparent from the date in the header.

Suggestions: Your suggestion for additional content or elaboration of some topics is most welcome acookbook@2302.dk.

Keywords: Matrix algebra, matrix relations, matrix identities, derivative of determinant, derivative of inverse matrix, differentiate a matrix.

https://www.math.uwaterloo.ca/~hwolkowi/matrixcookbook.pdf

必备查询网站



http://www.matrixcalculus.org

CVX

CVX RESEARCH

CVX TFOCS About us News CVX Forum

Home Download Documentation Examples Support Licensing Citing

CVX: Matlab Software for Disciplined Convex Programming

Version 2.2, January 2020, Build 1148

New: Professor Stephen Boyd recently recorded a video introduction to CVX for Stanford's convex optimization courses. Click here to watch it.

CVX 3.0 beta: We've added some interesting new features for users and system administrators. Give it a try!

CVX is a Matlab-based modeling system for convex optimization. CVX turns Matlab into a modeling language, allowing constraints and objectives to be specified using standard Matlab expression syntax. For example, consider the following convex optimization model:

$$\begin{array}{ll} \text{minimize} & \|Ax - b\|_2 \\ \text{subject to} & Cx = d \\ & \|x\|_\infty \leq e \end{array}$$

The following code segment generates and solves a random instance of this model:

```
m = 20; n = 10; p = 4;
A = randn(m,n); b = randn(m,1);
C = randn(p,n); d = randn(p,1); e = rand;
cvx_begin
    variable x(n)
    minimize( norm( A * x - b, 2 ) )
    subject to
        C * x == d
        norm( x, Inf ) <= e
cvx_end
```

http://cvxr.com/cvx/

CVXPY

CVXPY

Star 2,752

Navigation

- Install
- Tutorial
- Examples
- API Documentation
- FAQ
- Citing CVXPY
- Contributing
- Related Projects
- Changes to CVXPY
- CVXPY Short Course
- License

Quick search

Go

Welcome to CVXPY 1.1

Convex optimization, for everyone.

For the best support, join the CVXPY mailing list and post your questions on Stack Overflow.

CVXPY is a Python-embedded modeling language for convex optimization problems. It allows you to express your problem in a natural way that follows the math, rather than in the restrictive standard form required by solvers.

For example, the following code solves a least-squares problem with box constraints:

```
import cvxpy as cp
import numpy as np

# Problem data.
m = 30
n = 20
np.random.seed(1)
A = np.random.randn(m, n)
b = np.random.randn(m)

# Construct the problem.
x = cp.Variable(n)
objective = cp.Minimize(cp.sum_squares(A*x - b))
constraints = [0 <= x, x <= 1]
prob = cp.Problem(objective, constraints)

# The optimal objective value is returned by `prob.solve()`.
result = prob.solve()
# The optimal value for x is stored in `x.value`.
print(x.value)
# The optimal Lagrange multiplier for a constraint is stored in
# `constraint.dual_value`.
print(constraints[0].dual_value)
```

https://www.cvxpy.org/

优化为什么重要?

Optimization problems underlie nearly everything we do in Machine Learning and Statistics. In other courses, you learn how to:

translate  into $P : \min_{x \in D} f(x)$

Conceptual idea *Optimization problem*

This course: how to solve P, and why this is a good skill to have

优化问题的一般形式

(mathematical) optimization problem

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq b_i, \quad i = 1, \ldots, m \end{array}$$

- $x = (x_1, \ldots, x_n)$: optimization variables
- $f_0 : \mathbf{R}^n \to \mathbf{R}$: objective function
- $f_i : \mathbf{R}^n \to \mathbf{R}$, $i = 1, \ldots, m$: constraint functions

optimal solution $x^\star$ has smallest value of f_0 among all vectors that satisfy the constraints

优化问题的求解

general optimization problem

- very difficult to solve
- methods involve some compromise, $e.g.$, very long computation time, or not always finding the solution

exceptions: certain problem classes can be solved efficiently and reliably

- least-squares problems
- linear programming problems
- convex optimization problems

最小二乘问题

$$\text{minimize} \quad \|Ax - b\|_2^2$$

solving least-squares problems

- analytical solution: $x^\star = (A^T A)^{-1} A^T b$
- reliable and efficient algorithms and software
- computation time proportional to $n^2 k$ ($A \in \mathbf{R}^{k \times n}$); less if structured
- a mature technology

using least-squares

- least-squares problems are easy to recognize
- a few standard techniques increase flexibility ($e.g.$, including weights, adding regularization terms)

机器学习中的最小二乘问题

Ridge regression/Tikhonov regularization

$$\min_{x} \quad \frac{1}{2}\|Ax-b\|_2^2+\mu\|x\|_2^2$$

sparse regularization

$$\min_{x} \quad \frac{1}{2}\|Ax-b\|_2^2+\mu\|x\|_1$$

Lasso/Basis pursuit

$$\min_{x} \quad \|x\|_1, \ \text{s.t. } \|Ax-b\|_2 \le \epsilon$$

or

$$\min_{x} \quad \|Ax-b\|_2, \ \text{s.t. } \|x\|_1 \le \sigma$$

线性规划问题

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & a_i^T x \leq b_i, \quad i = 1, \ldots, m \end{array}$$

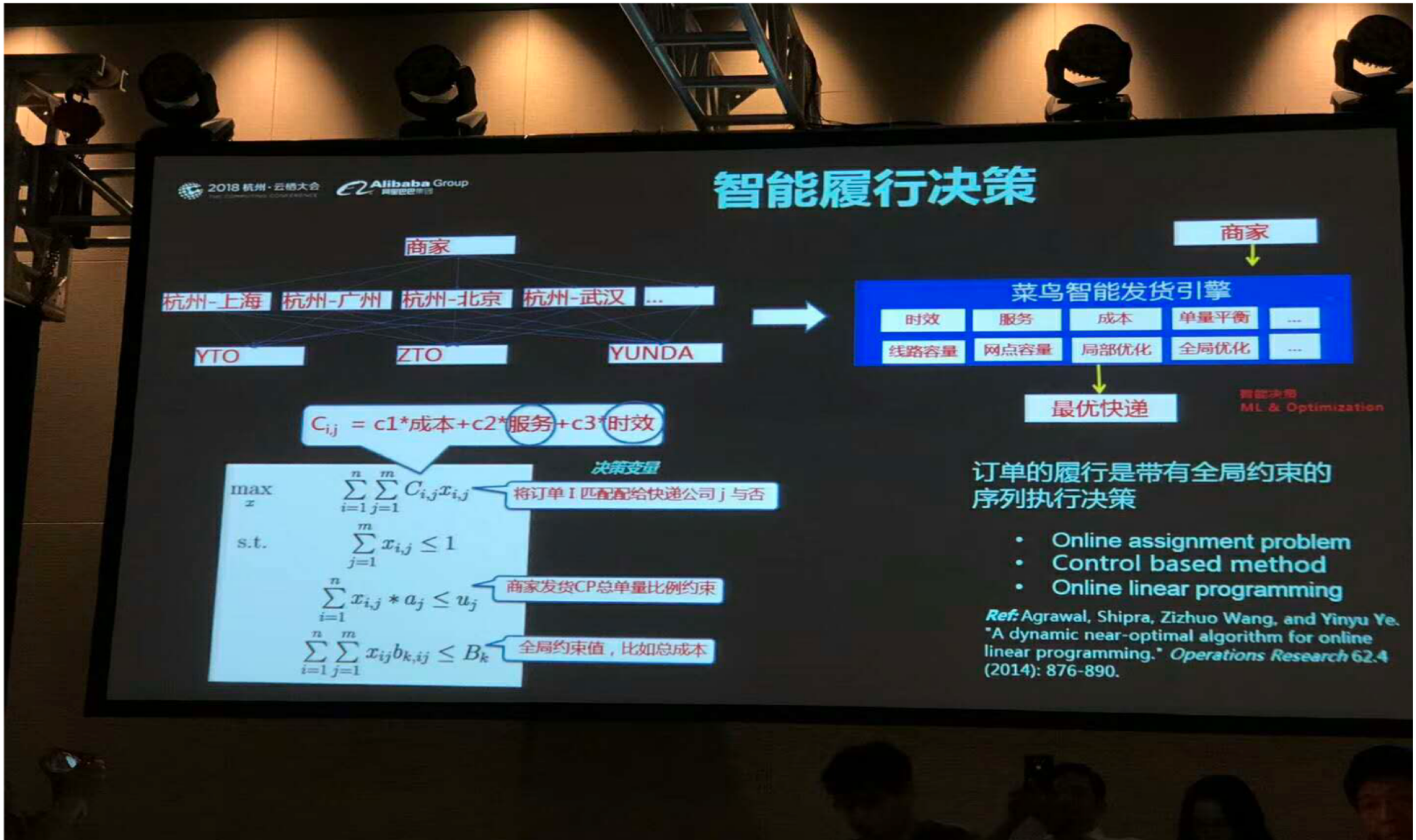
solving linear programs

- no analytical formula for solution
- reliable and efficient algorithms and software
- computation time proportional to $n^2 m$ if $m \geq n$; less with structure
- a mature technology

using linear programming

- not as easy to recognize as least-squares problems
- a few standard tricks used to convert problems into linear programs ($e.g.$, problems involving ℓ_1- or ℓ_∞-norms, piecewise-linear functions)

一个实例：菜鸟网络



2018 杭州·云栖大会
Alibaba Group
智能履行决策
商家
杭州-上海
杭州-广州
杭州-北京
杭州-武汉
...
YTO
ZTO
YUNDA
菜鸟智能发货引擎
时效
服务
成本
单量平衡
...
线路容量
网点容量
局部优化
全局优化
...
最优快递
ML & Optimization
$C_{i,j}$ = c1*成本+c2*服务+c3*时效
决策变量
$\max_x \sum_{i=1}^{n}\sum_{j=1}^{m} C_{i,j}x_{i,j}$
将订单 I 匹配配给快递公司 j 与否
s.t. $\sum_{j=1}^{m} x_{i,j} \le 1$
$\sum_{i=1}^{n} x_{i,j} * a_j \le u_j$
商家发货CP总单量比例约束
$\sum_{i=1}^{n}\sum_{j=1}^{m} x_{ij}b_{k,ij} \le B_k$
全局约束值，比如总成本
订单的履行是带有全局约束的
序列执行决策
Online assignment problem
Control based method
Online linear programming
Ref: Agrawal, Shipra, Zizhuo Wang, and Yinyu Ye. "A dynamic near-optimal algorithm for online linear programming." Operations Research 62.4 (2014): 876-890.

机器学习的核心是优化

Many problems in ML can be written as

$$\min_{x \in \mathcal{W}} \quad \sum_{i=1}^{N} \frac{1}{2} \|a_i^\top x - b_i\|_2^2 + \mu \|w\|_1 \qquad \text{linear regression}$$

$$\min_{x \in \mathcal{W}} \quad \frac{1}{N} \sum_{i=1}^{N} \log(1 + \exp(-b_i a_i^\top x)) + \mu \|x\|_1 \qquad \text{logistic regression}$$

机器学习的核心是优化

Many problems in ML can be written as

$$\min_{x\in\mathcal{W}} \quad \sum_{i=1}^{N} \frac{1}{2}\|a_i^\top x - b_i\|_2^2 + \mu\|w\|_1 \quad \text{linear regression}$$

$$\min_{x\in\mathcal{W}} \quad \frac{1}{N}\sum_{i=1}^{N} \log(1+\exp(-b_i a_i^\top x)) + \mu\|x\|_1 \quad \text{logistic regression}$$

$$\min_{w\in\mathcal{W}} \quad \sum_{i=1}^{N} \ell(h(x,a_i), b_i) + \mu r(x) \quad \text{general formulation}$$

The pairs (a_i, b_i) are given data, b_i is the label of the data point a_i

$\ell(\cdot)$: measures how model fit for data points (avoids under-fitting)

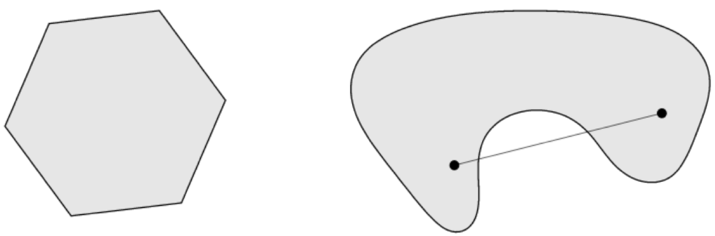
$r(x)$: regularization term (avoids over-fitting)

$h(x, a)$: linear function or models constructed from deep neural networks

凸集与凸函数

Convex set: $C \subseteq \mathbb{R}^n$ such that

$$x, y \in C \implies tx + (1-t)y \in C \text{ for all } 0 \le t \le 1$$



Convex function: $f : \mathbb{R}^n \to \mathbb{R}$ such that $\mathrm{dom}(f) \subseteq \mathbb{R}^n$ convex, and

$$f(tx + (1-t)y) \le tf(x) + (1-t)f(y) \text{ for all } 0 \le t \le 1$$

and all $x, y \in \mathrm{dom}(f)$

凸优化问题

$$
\begin{array}{ll}
\text{minimize} & f_0(x) \\
\text{subject to} & f_i(x) \leq b_i, \quad i = 1, \ldots, m
\end{array}
$$

- objective and constraint functions are convex:

$$f_i(\alpha x + \beta y) \leq \alpha f_i(x) + \beta f_i(y)$$

  if $\alpha + \beta = 1$, $\alpha \geq 0$, $\beta \geq 0$

- includes least-squares problems and linear programs as special cases

凸优化问题

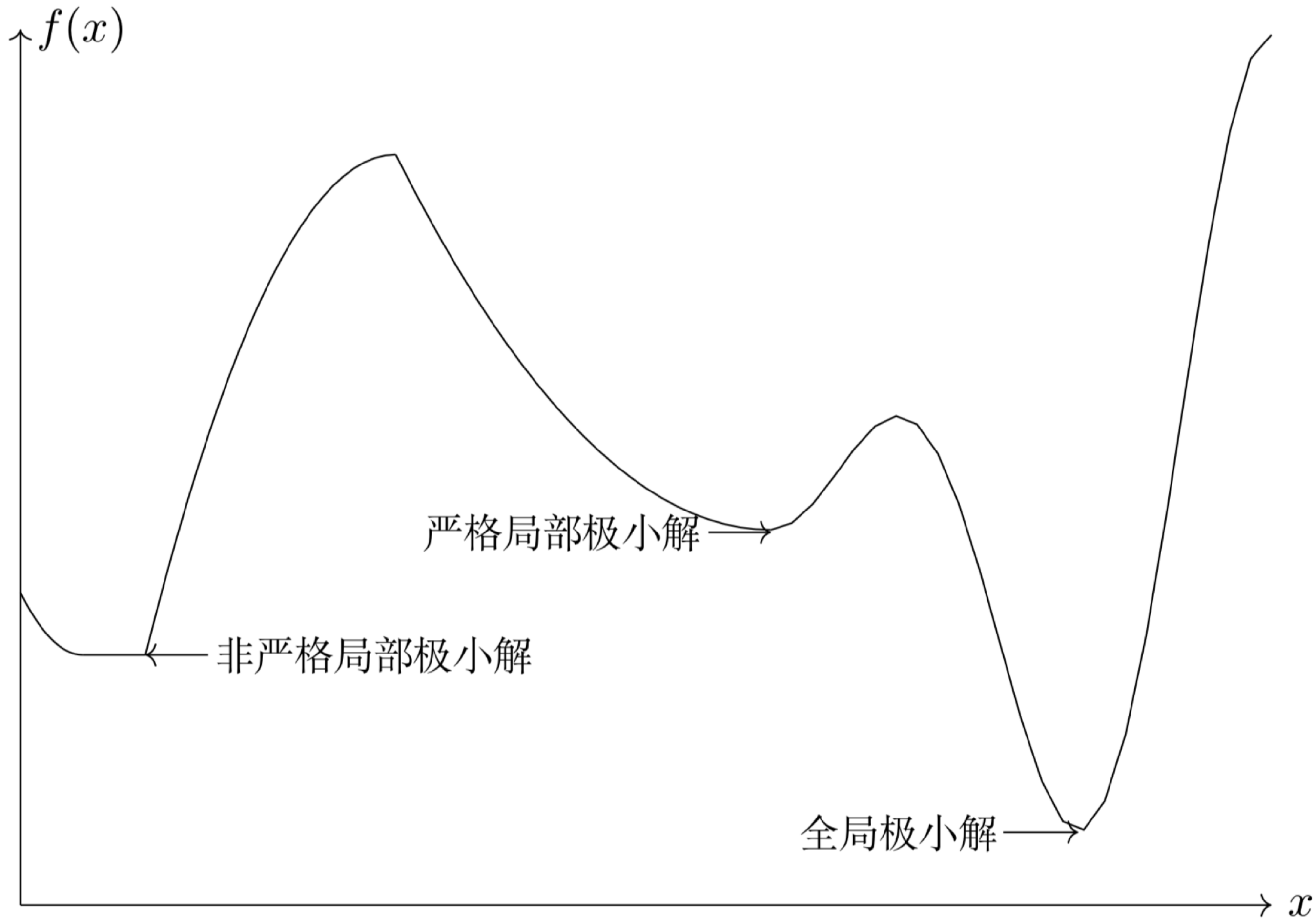
solving convex optimization problems

- no analytical solution
- reliable and efficient algorithms
- computation time (roughly) proportional to $\max\{n^3, n^2m, F\}$, where F is cost of evaluating f_i's and their first and second derivatives
- almost a technology

using convex optimization

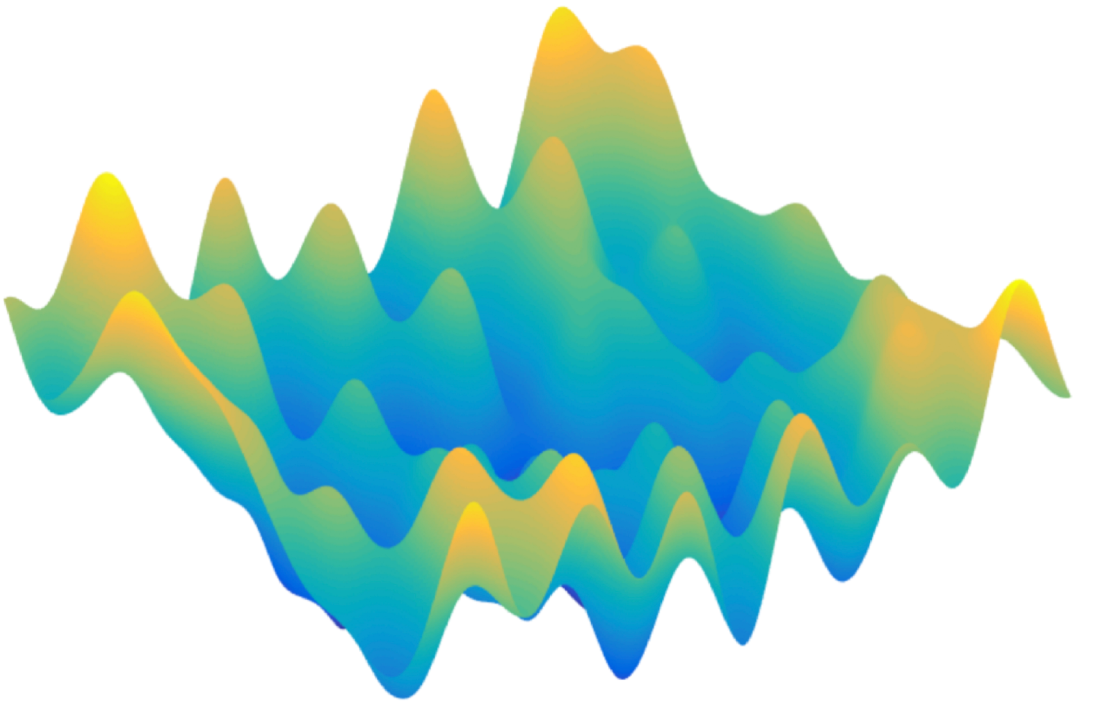
- often difficult to recognize
- many tricks for transforming problems into convex form
- surprisingly many problems can be solved via convex optimization

全局和局部最优解



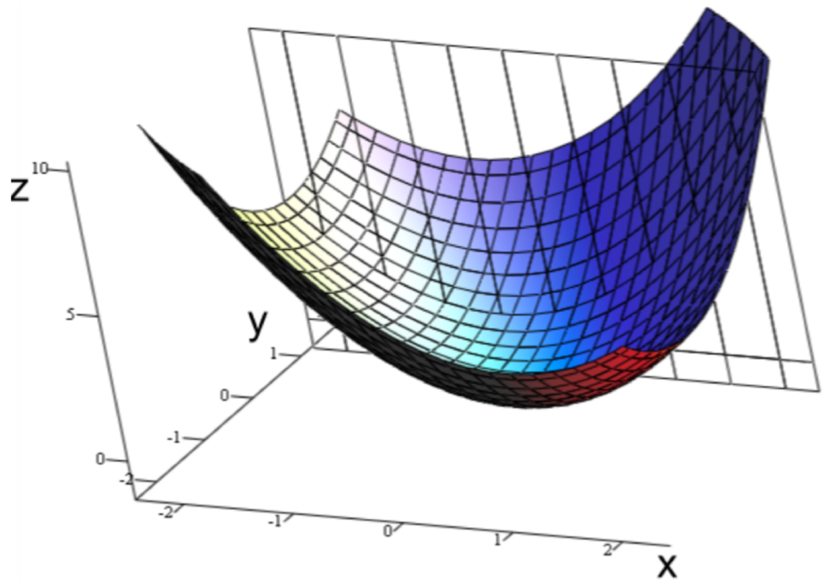
我们想要得到全局最优解，但是由于实际问题的复杂性，往往只能够得到其局部最优解。

全局和局部最优解



There may be bumps everywhere and exponentially many local optima

全局和局部最优解



For convex optimization problems, local minima are global minima

Formally, if x is feasible—$x \in D$, and satisfies all constraints—and minimizes f in a local neighborhood,

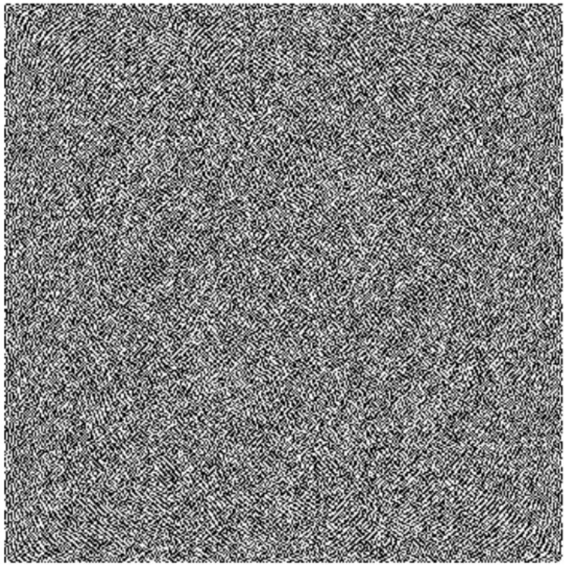
$$f(x) \leq f(y) \;\; \text{for all feasible} \;\; y, \;\; \|x - y\|_2 \leq \rho,$$

then

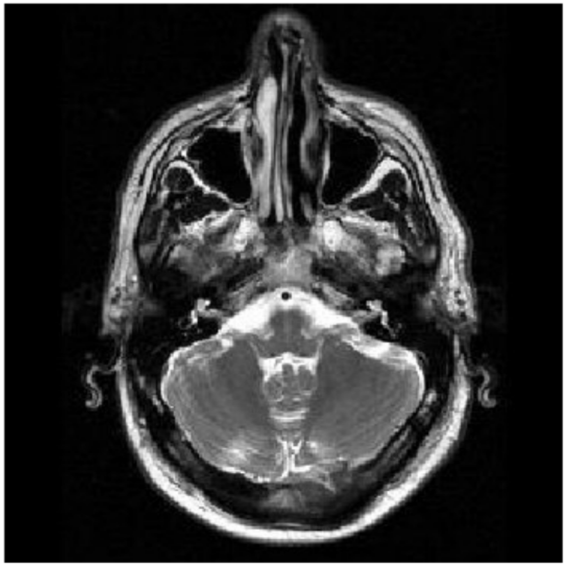
$$f(x) \leq f(y) \;\; \text{for all feasible} \;\; y$$

核磁共振成像

(a) MRI Scan

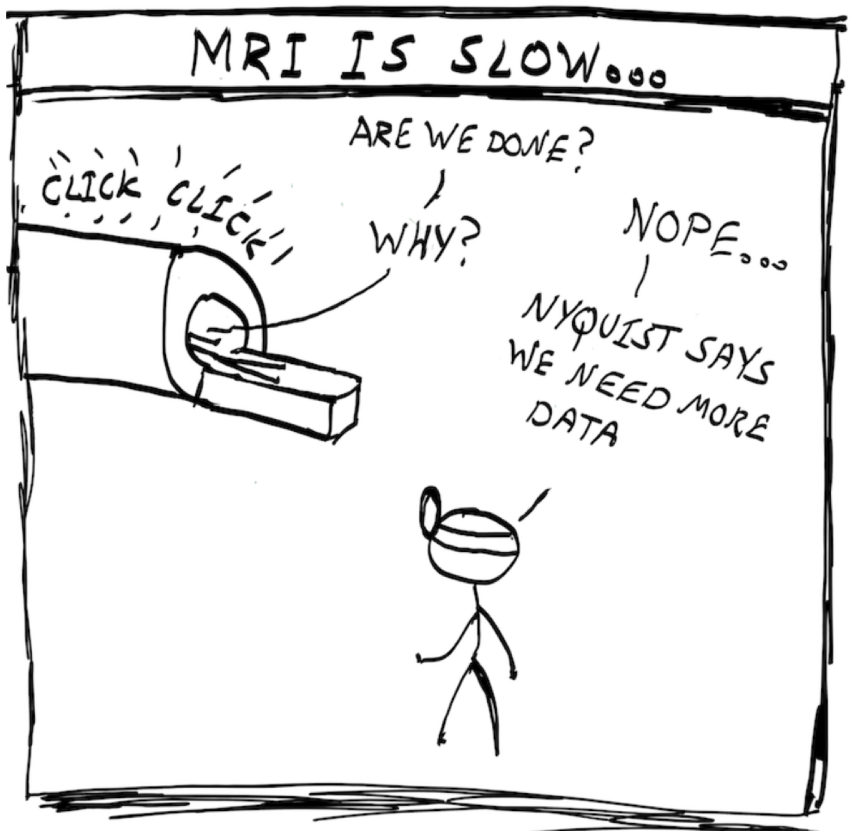


(b) Fourier Coefficients



(c) Image

核磁共振成像

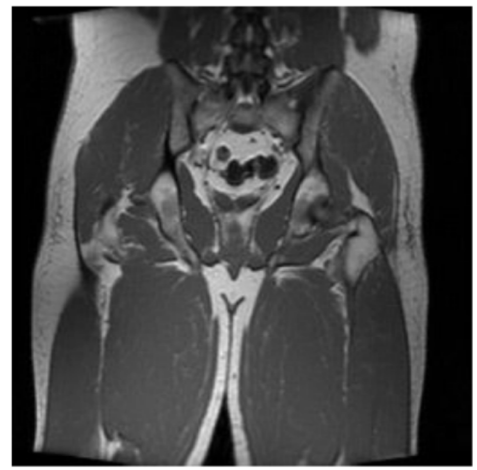


Is it possible to cut the scan time into half?

加速核磁共振成像

The higher the SNR (signal-noise ratio) is, the better the image quality is.

(a) full sampling



(b) 39% sampling, SNR=32.2

solve convex program: $\min_{u} \|\Phi u\|_1 + \frac{\mu}{2}\|Ru - b\|^2$

Netflix Prize

	movie I	movie II	movie III	movie IV	$\cdots$
User A	1	?	5	4	$\cdots$
User B	?	2	3	?	$\cdots$
User C	4	1	2	?	$\cdots$
User D	?	5	1	3	$\cdots$
User E	1	2	?	?	$\cdots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$

- Training Set: 480,000 customer ratings on 18,000 movies.
- Around 98.7% missing ratings!
- $1,000,000 prize!

Netflix与大数据

纸牌屋探秘：Netflix的大数据炼金术

作者：王萌 2013年02月13日 动态, 大数据, 热点 19,817 阅读

Low-rank modeling

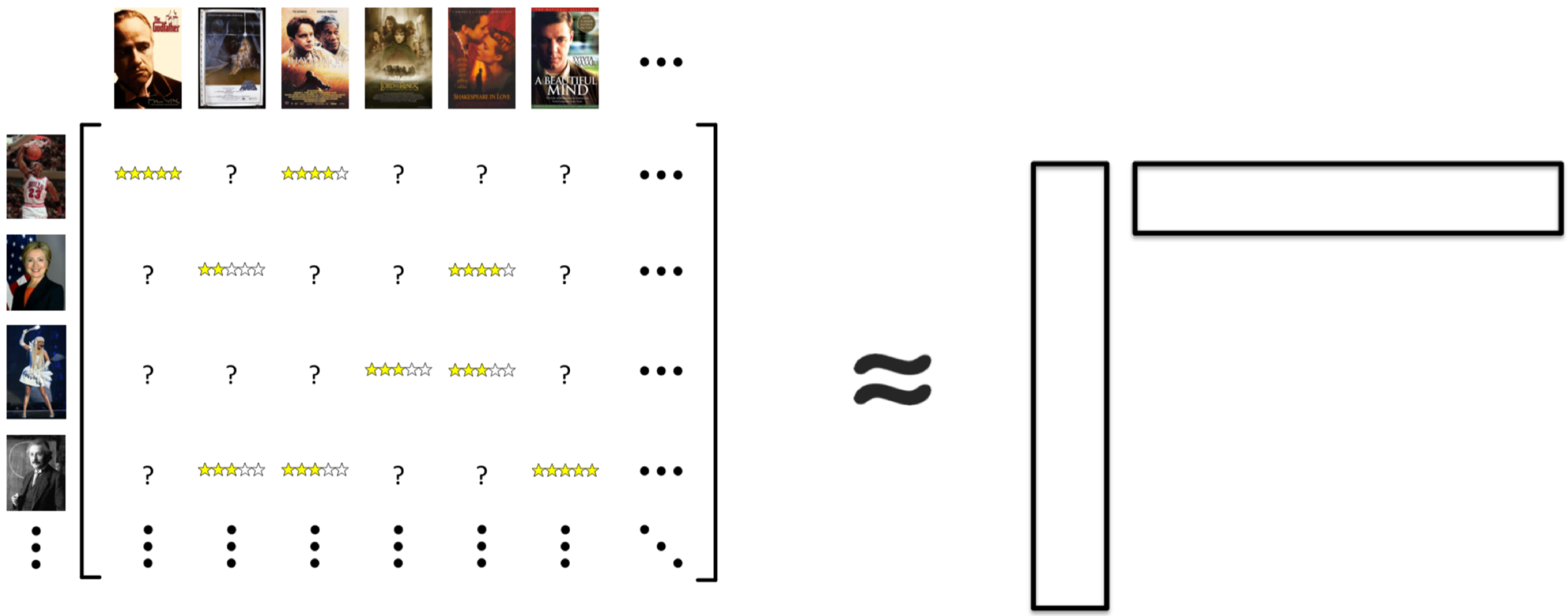


figure credit: E. Candes

A few factors explain most of the data $\longrightarrow$ low-rank approximation

How to exploit (approx.) low-rank structure in prediction?

Matrix Rank Minimization

Given $X \in \mathbb{R}^{m\times n}$, $\mathcal{A} : \mathbb{R}^{m\times n} \to \mathbb{R}^p$, $b \in \mathbb{R}^p$, we consider

- matrix completion problem:

$$\min \ \operatorname{rank}(X), \ \text{s.t. } X_{ij} = M_{ij}, (i,j) \in \Omega$$

- the matrix rank minimization problem:

$$\min \ \operatorname{rank}(X), \ \text{s.t. } \mathcal{A}(X) = b$$

Matrix Rank Minimization

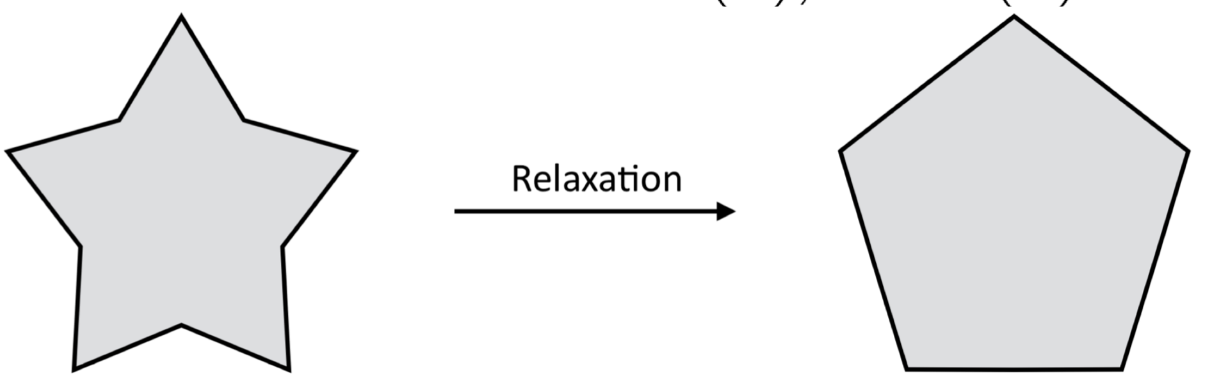
Given $X \in \mathbb{R}^{m\times n}$, $\mathcal{A} : \mathbb{R}^{m\times n} \rightarrow \mathbb{R}^p$, $b \in \mathbb{R}^p$, we consider

- matrix completion problem:

$$\min \ \operatorname{rank}(X), \ \text{s.t. } X_{ij} = M_{ij}, (i,j) \in \Omega$$

- the matrix rank minimization problem:

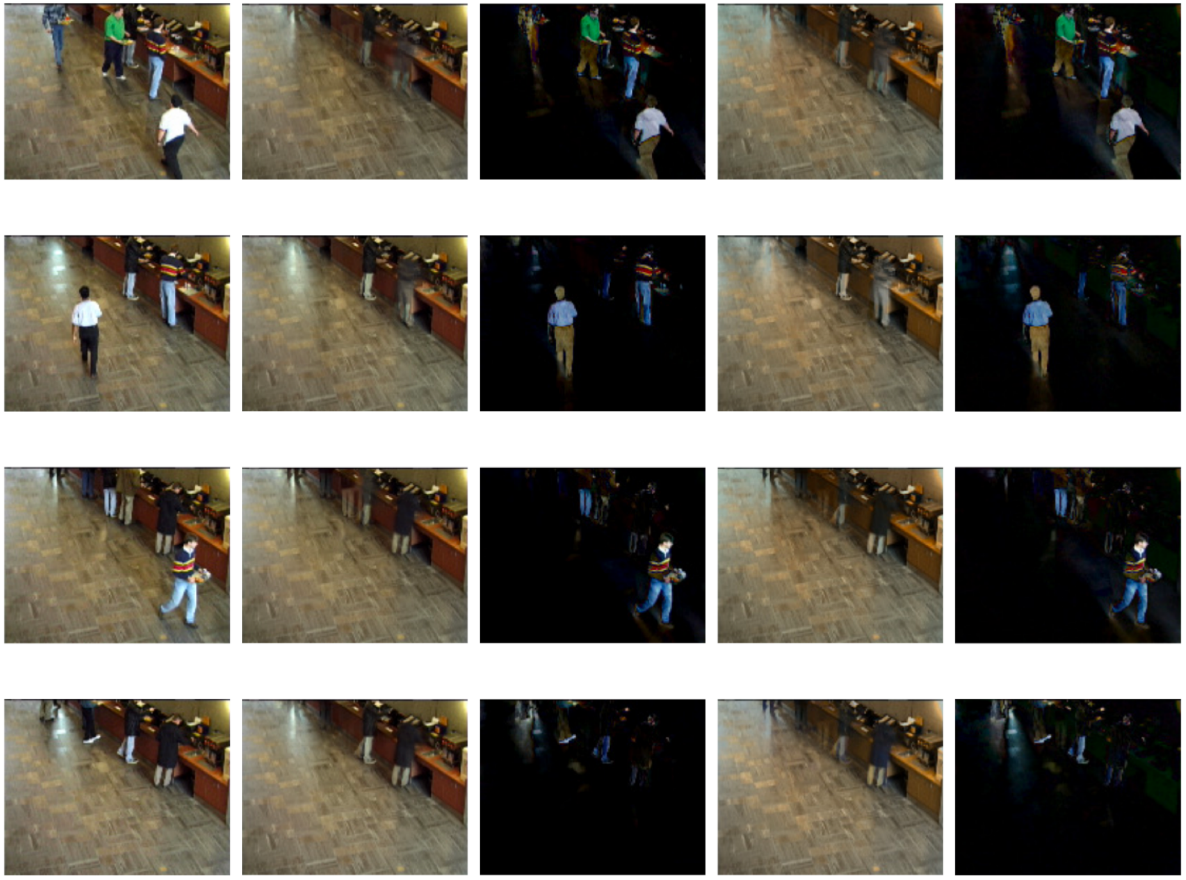
$$\min \ \operatorname{rank}(X), \ \text{s.t. } \mathcal{A}(X) = b$$



convex nuclear norm minimization: $\min \ \|X\|_* \ \text{s.t. } \mathcal{A}(X) = b$

监控视频分离

Partition the video into moving and static parts



低秩与稀疏矩阵分解

Given a matrix M, we want to find a low rank matrix W and a sparse matrix E, so that $W + E = M$.

Convex approximation:

$$\min_{W,E} \|W\|_* + \mu\|E\|_1, \text{ s.t. } W + E = M$$

Robust PCA

Thank you for your attentions !