南昌大学
NANCHANG UNIVERSITY

大数据优化：理论、算法及其应用

——来源于《最优化计算方法》（高教社）

★★★ 主讲人：彭振华 ★★★

联系方式：zhenhuapeng@ncu.edu.cn
zhenhuapeng@whu.edu.cn
15870605317（微信同号）

研究兴趣：1. 非凸非光滑优化算法与理论
2. 智能决策
3. 智能计算与机器学习

数学与计算机学院

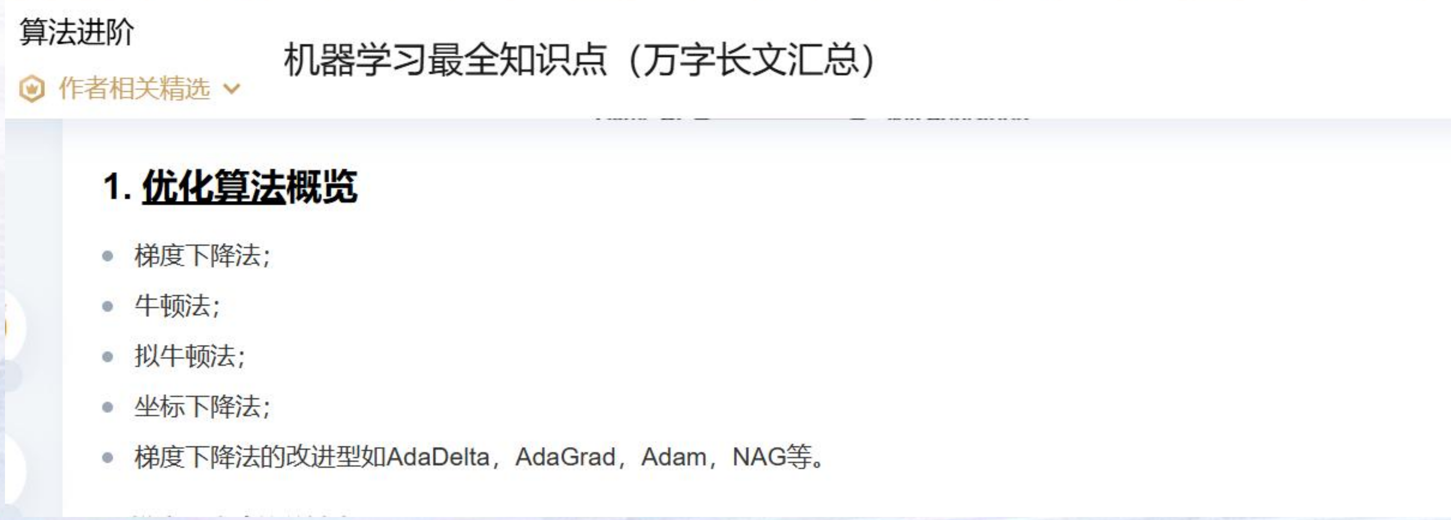
2026年

课件资源：https://zhenhuapeng.github.io//coursematerials/

目录

为什么要学习大数据优化?

- 应用广泛，是建模与求解核心工具，有助于提升决策效率，是连接抽象理论与实际应用桥梁，如：资源调配、材料轻量化设计、药物筛选
- 机器学习算法（含深度神经网络算法）依赖优化方法最小化损失函数，从而确定模型参数，如：梯度下降、牛顿法等是训练过程核心工具


算法进阶

作者相关精选

机器学习最全知识点（万字长文汇总）

1. 优化算法概览

- 梯度下降法；
- 牛顿法；
- 拟牛顿法；
- 坐标下降法；
- 梯度下降法的改进型如AdaDelta，AdaGrad，Adam，NAG等。

为什么要学习大数据优化?

➢ 就业岗位(运筹优化算法岗、机器学习（深度学习）算法岗) 需要

运筹优化算法工程

上海-青浦区 3-5年

岗位职责:

1. 负责运筹优化或机器学习方

线性规划等。

2. 负责将系统优化和机器学习

低效率提高，协助提高核心竞

3. 负责从大数据分析和建模，

4. 具备良好的数学建模能力，

5. 利用运筹优化和机器学习领

岗位要求:

1、学历与专业：本科及以上

2、知识与能力:

1）熟悉机器学习或者优化、

2）较强的逻辑分析能力和数

3）较好的编程能力，熟练掌

模求解工具。

3、工作经验：有物流、快递、

BDI-运筹优化岗(物流

深圳 3-5年 硕士

查看更多优选职位

职位描述

发表算法相关优秀论文 Java

岗位职责

1、负责运筹优化相关的模型工具开

2、深入参与营运模式创新相关决策

景;

3、具备良好的数学建模能力和算法

任职要求

1、具有3年及以上工作经验，具备

2、熟练掌握运筹优化的建模与算法

3、具备业务和数据敏感性，积极了

和求解算法的技术路线选择有一定

4、熟练掌握至少一门主流编程语言

熟悉Hive、SQL等数据查询分析语言

运筹优化算法/开发工程师 25-40

北京 经验不限 本科

感兴趣 立即沟通

职位描述

求解器 AMPL 高性能计算 运筹优化

【岗位职责】

1. 参与或负责数学优化求解器的整体架构设计、功能模块设计

改进和优化，构筑有竞争力的数学优化求解器迭代演进方案。

2. 参与或负责洞察业界进展和行业动态，不断发掘求解器的商

3. 参与或负责华为云企业智能的决策优化算法服务的设计、实

遇到的各种问题。

4. 参与或负责华为工业、供应链、交通、能源和证券金融等领

【岗位要求】

1. 熟悉数学优化理论（如线性规划、整数规划，非线性约束规

和优化，有较强的系统实现能力。

2. 具有解决大规模线性规划（LP）、混合整数规划（MLP）或

（比如CPLEX、Gurobi）解决过物流、航空、港口、能源、工

3. 熟悉常用的高级建模语言（比如AMPL、GAMS），熟悉Lin

多种开发语言。

4. 有求解器开发经验者优先; 有高性能计算、分布式开发经验者

5. 能快速掌握业务需求，有较强的学习能力和快速解决问题的

学习和善于思考。

岗位信息

对象: 26年及以后毕业本硕博校招/实习

工作地点：深圳/北京/武汉/上海/西安/杭州等

黄女士 刚刚活跃

华为技术有限公司 · 招聘主管

高级数据工程师（运筹优... 22-35K

五险一金 补充医疗

武汉 5-10年 硕士

查看更多优选职位

职位描述

微信扫码分享 举报

R 自然语言处理经验 机器学习经验 计算机相关专业 运筹优化 Python

岗位职责:

1. 算法模型：基于手机供应链管理或者用户行为分析等方向，对业务场景及问题，构建/完善算法与模型，寻找数据应用突破点，促进业务数字化建设和AI技术应用以助力业务发展。

2. 数据处理和分析：利用数据挖掘、机器学习和相关算法处理大量数据，进行模式识别和预测，并能将分析结论形成分析报告和建议，为部门运营、质量改进，提升用户体验等需求提供决策支持。

3. 技术研究和应用：跟进供应链运营、优化、库存管理、机器学习、数据挖掘、数据可视化等领域的技术发展，并将这些技术应用于供应链管理中。

任职资格:

1. 有扎实的数学知识和逻辑思维能力，运筹学，统计学、应用数学、计算机软件等相关专业，硕士以上学历。

2. 至少精通两种编程语言，SQL+Python/R语言，有工程实现能力。有算法设计，数据挖掘的经验优先考虑;

3. 至少7年以上的数据分析/算法开发等相关的工作经验，深度学习或运筹学或自然语言处理等领域有深入研究。供应链相关行业经验者优先。

4. 有数据项目管理经验以及PMP认证者优先。

5. 能力全面，自我驱动，结果导向，沟通能力佳（英语熟练），有强烈的责任心。

林女士 今日活跃

联想集团 · Recruiter

为什么要学习大数据优化?

- 国家重点研发计划” 数学与应用研究” 等重点专项2024 年度项目
  - I. 3.3 大模型辅助运筹优化决策建模与计算
  - II. 6.3 移动网络性能建模的数学理论与优化方法
- 2025 年度国家自科委数学物理科学部重点项目资助领域
  - I. 模型或数据驱动的优化理论与方法(A04)
  - II. 大规模问题的优化建模与高效算法(A04)
  - III. 经济与金融中的关键数学问题(A04-A06)
  - IV. 人工智能和数据科学的数学理论与方法(A04-A06)
- 2025 年度国家自科委管理科学部重点项目资助领域
  - I. 供应链管理中非凸问题的优化理论与算法（G0102）
  - II. 人工智能驱动的供应链创新设计研究（G0109）
  - III. 模型、数据双驱动的大规模城市道路交通流协同优化（G0116）
  - IV. 区块链和供应链深度融合下的药品安全治理模式（G0406）

*课程思政：关注国家急需解决的问题，激发他们学习相关知识的热情，为国家的未来发展培养更多有理想、有本领、有担当的新时代青年.

最优化发展历程

20世纪50年代中期

华罗庚、钱学森、许国志等教授将运筹学由西方引入我国

1959年

运筹学部门在中国科学院数学研究所成立

1960年

力学所小组与数学所的小组合并成为数学研究所的一个研究室

1980年

中国数学会运筹学会成立

1991年

国家一级学会中国运筹学会成立

中国运筹学会下设18个分会和在筹2个分会

https://www.orsc.org.cn/division/list?page=1

＊课程思政：融入先辈卓越贡献和成功事迹，增强对中华传统文化认同感，激发青年学子的报国情怀与时代担当，同时体会科学研究需要坚持不懈的努力和坚韧不拔的精神。

什么是最优化?

- 在给定的区域中寻找最小化或最大化某一函数的最优解

$$\left(\max_{x}\right)\min_{x} \; f(x) \quad s.t. \;\; x \in \chi.$$

决策变量

目标函数

约束条件

可行域

注意： $x=(x_1,x_2,\ldots,x_n)^T$是一个n维向量

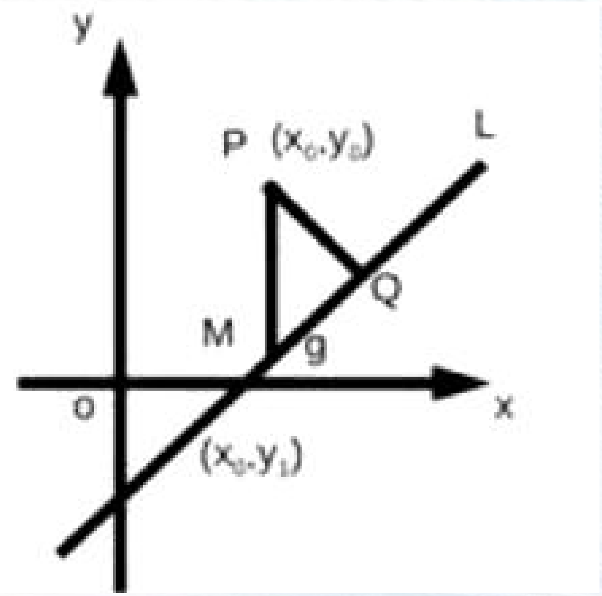
研究流程： 定性或定量分析 → 构建优化模型 → 设计求解算法 → 分析模型算法性质 → 编程实现 → 结果检验 → 定性或定量分析

中学数学中的优化案例

点到直线的距离

$$\min_{x} \sqrt{\left(x-x_0\right)^2+\left(ax+b-y_0\right)^2}$$

$$x=\frac{x_0+ay_0-ab}{1+a^2}, d_{\min}=\frac{\left|ax_0+b-y_0\right|}{\sqrt{1+a^2}}.$$



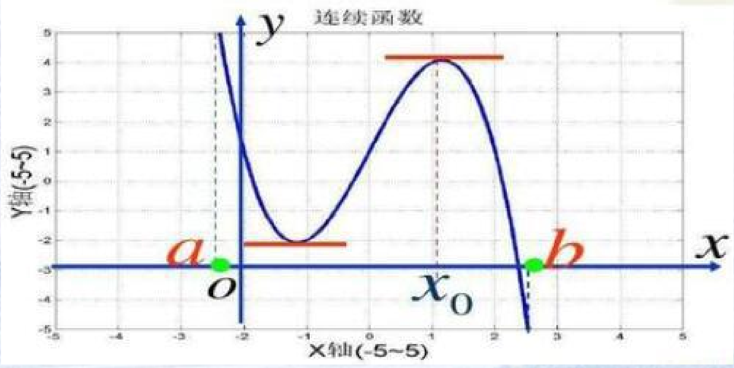
固定周长面积最大的矩形是正方形

$$\max_{\ell,d} \ \ell\times d \quad s.t.\ \left(\ell,d\right)\in\left\{\left(\ell,d\right):2\left(\ell+d\right)=22\right\}.$$

$$\ell=d=5.5.$$

费马引理: 利用导数为0 确定潜在的最优点

$$\left(\max_{x}\right)\min_{x} \ f\left(x\right) \longrightarrow f'\left(x\right)=0$$



从正态分布到最小二乘问题

- **正态分布的概率密度函数：**

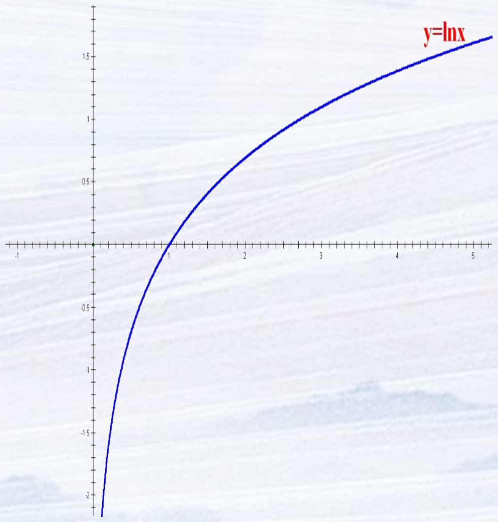
$$P(x)=\frac{1}{\sqrt{2\pi\sigma}}\exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

- **联合概率密度函数(似然函数)（独立同分布数据$\{x_i\}$）：**

$$\prod_i P(x_i)=\frac{1}{\left(\sqrt{2\pi\sigma}\right)^N}\exp\left(-\frac{\sum_i (x_i-\mu)^2}{2\sigma^2}\right)$$

- **对数似然函数**

$$\ln\left(\prod_i P(x_i)\right)=-N\ln\left(\sqrt{2\pi\sigma}\right)-\frac{\sum_i (x_i-\mu)^2}{2\sigma^2}$$



从正态分布到最小二乘问题

- 设$\{(x^i,y^i)\}$是观测数据，其中x^i是特征向量，y^i是对应的观测值。假设它们之间存在如下函数关系：

$$y^i = f(x^i) + \epsilon^i,$$

其中ϵ^i是对应的数据误差。

- 假设ϵ^i是高斯白噪声，即服从均值为0的正态分布。

$$\max_f - N\ln\left(\sqrt{2\pi}\sigma\right) - \frac{\sum_i \left(f\left(x^i\right) - y^i\right)^2}{2\sigma^2}$$

$$\Downarrow$$

$$\min_f \sum_i \left(f\left(x^i\right) - y^i\right)^2$$

从正态分布到最小二乘问题

- **多元线性回归**

$$\min_{a,b} \sum_i \left(a^T x^i + b - y^i\right)^2$$

- **一元多项式回归**

$$\min_{a} \sum_i \left(a_0 + a_1 x^i + a_2 \left(x^i\right)^2 \cdots - y^i\right)^2$$

- 要判断一个模型好不好，仅仅看显著性是不够的。还需用拟合优度，指回归直线对观测值的拟合程度

$$R^2 = \frac{\sum_{i=1}^{N}\left(\hat{y}^i - \bar{y}\right)^2}{\sum_{i=1}^{N}\left(y^i - \bar{y}\right)^2}$$

- R的平方越接近1，回归方程对于样本数据点的拟合优度越高；反之，R的平方越接近0，回归方程对于样本数据点的拟合优度越低。

教材与参考书籍

➢ 教材

1. 刘浩洋, 户将, 李勇锋，文再文，最优化计算方法，高等教育出版社，2021.

➢ 参考书籍

1. 刘浩洋, 户将, 李勇锋，文再文，最优化：建模、算法与理论, 高等教育出版社，2022.
2. 文再文, 袁亚湘, 最优化方法与理论(“101计划”核心教材), 高等教育出版社, 2024.
3. 袁亚湘，孙文瑜, 最优化理论与方法, 科学出版社, 2011.
4. Mykel J. Kochenderfer, Tim A. Wheeler, 优化理论与实用算法, 机械工业出版社, 2022.
5. Amir Beck, First-Order Methods in Optimization, SIAM, 2017.
6. Jorge Nocedal, Stephen J. Wright. Numerical Optimization, Springer, 2006.

南昌大学
NANCHANG UNIVERSITY

大数据优化简介

第 一 部分

模型与基本概念

优化建模技术

应用实例

求解器与大模型

最优化模型与分类

- **通用形式：**

$$\min_{x} \ f(x) \quad s.t. \ x \in \chi.$$

- **案例：**

$$(I)\min_{x} \sqrt{(x-x_0)^2+(ax+b-y_0)^2}$$

$$(II)\min_{\ell,d} \ \ell \times d \quad s.t. \ 2(\ell+d)=22.$$

$$(III)\left(\max_{x}\right)\min_{x} \ f(x)$$

$$(IV)\min_{f}\sum_{i}\left(f\left(x^i\right)-y^i\right)^2$$

$$(V)\min_{a,b}\sum_{i}\left(a^T x^i+b-y^i\right)^2$$

$$(VI)\min_{a}\sum_{i}\left(a_0+a_1x^i+a_2\left(x^i\right)^2\cdots-y^i\right)^2$$

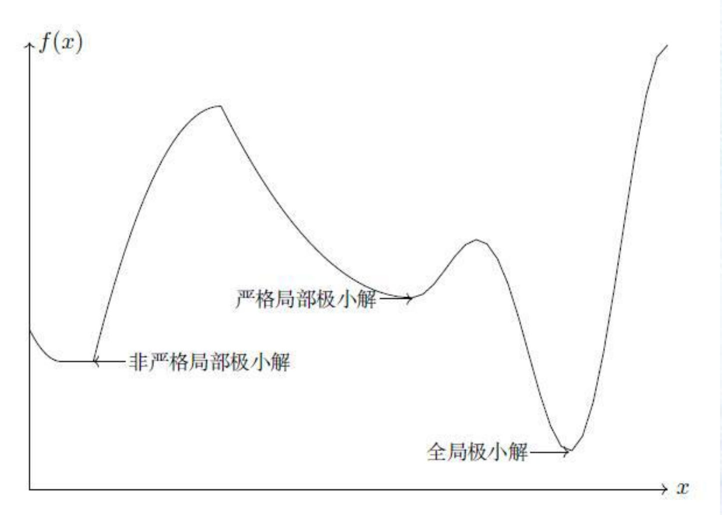
最优解的概念

$$\min_{x} \; f(x) \quad s.t. \; x \in \chi.$$

- **可行解：** 可行域内所有的点

- **局部最优解：** 设 $z \in X$, 若对所有 $x \in N(z, \delta) \cap X$, 有

  $f(z) \le f(x)$, 则称 z 是优化问题的局部最优解；



- **全局最优解：** 设 $z \in X$, 若对所有 $x \in X$, 有 $f(z) \le f(x)$,

  则称 z 是优化问题的(全局)最优解；

问题：试借助问题II阐释最优解的概念?

- **注记：若把"$\le$"换成"$<$"，则称严格(局部)极小解**

＊课程思政：只有坚持党的领导，增强“四个意识”、坚定“四个自信”、做到“两个维护”，牢记“国之大者”(可行域X 内)，才有可能能实现自己的理想(最优解$\bar{x}$)

最优化模型与分类

$$\min_{x} \ f(x) \quad s.t. \ \ x \in \chi.$$

- **有无约束**

  I. 无约束优化： $\chi = \mathbb{R}^n \longrightarrow \min_{x} \ f(x)$

  II. 约束优化： $\chi \neq \mathbb{R}^n$

注意：等式和不等式是表达数量关系的两个最基本、最重要的数学工具

约束域可以表述为

$$\chi = \left\{ x \in \mathbb{R}^n : g_i(x) \le 0, i = 1,2,\ldots,p, h_j(x) = 0, j = 1,2,\ldots,q \right\}.$$

约束优化可重述为

$$\begin{aligned} &\min_{x} \ f(x) \\ &s.t. \ \ g_i(x) \le 0, i = 1,2,\ldots,p, \\ &\qquad h_j(x) = 0, j = 1,2,\ldots,q. \end{aligned}$$

问题：六个案例中，哪些是无约束优化，哪些是约束优化？

最优化模型与分类

$$\min_{x} \; f(x) \quad s.t. \; x \in \chi.$$

- 决策变量类型
  - I. 连续优化：可行集合是连续的
  - II. 离散优化：可行集合是离散的，如整数规划
- 函数类型
  - I. 线性规划：所有目标函数和约束函数均是线性的
  - II. 非线性规划：目标函数和约束函数中至少有一个是非线性的

  或光滑/非光滑优化：所有目标函数和约束函数的可导性
- 凸与非凸优化
  - I. 凸优化：目标函数是凸函数+可行域为凸集
  - II. 非凸优化：目标函数不是凸函数或可行域不为凸集

问题：六个案例中，是否都为非线性规划？

两类特殊的最优化问题

无约束/约束　非线性　光滑　凸优化

➢ **线性规划**

$$\min_x c^T x = c_1x_1 + c_2x_2 + \cdots + c_nx_n$$

$$s.t.\quad Ax = b\left(a_i^T x = b_i\right), x \geq 0.$$

➢ **二次规划：目标函数是二次函数而约束函数是线性函数**

$$\min_x \frac{1}{2}x^T Qx + c^T x = \frac{1}{2}\sum_{i=1}^{n}\sum_{i=1}^{n} q_{ij}x_ix_j + \sum_{i=1}^{n} c_ix_i$$

$$s.t.\quad Ax = b\left(a_i^T x = b_i\right), x \geq 0.$$

➢ **练习：**

$$\min_x x_1^2 + x_2^2 + x_1x_2 + x_1 + x_2 \quad s.t.\quad x_1 + x_2 = 1, x \geq 0,$$

$$\min_x 0.5x_1^2 + x_2^2 - x_1x_2 - 2x_1 - 6x_2 \quad s.t.\quad x_1 + x_2 \leq 2, -x_1 + 2x_2 \leq 2.$$

问题：将案例(II,V,VI)转化为二次规划标准形式?

优化建模技术

- 三要素：决策变量、目标函数、约束条件
- 决策变量：对应关键可控因素
- 目标函数：

I. 极大似然估计

II. 代价、损失、收益函数、能量泛函

III. 正则化（$||\cdot||_0$（稀疏解）、$||\cdot||_1$（Lasso）、$||\cdot||^2$（TV正则））

IV. 松弛问题

(1) $f_L(x) \leq f(x) \leq f_U(x)$

(2) $f_L(x)$ 或 $f_U(x)$ 具有简单结构.

- 三要素：决策变量、目标函数、约束条件
- 约束条件：

I. 问题本身的物理性质

II. 等价转换

$$\min_{x} f\left(Ax-b\right) \longrightarrow \min_{x} f\left(y\right) \;\; s.t. \;\; y=Ax-b.$$

III. 松弛问题

(1) $f_L(x) \le f(x) \le f_U(x)$ $f_L(x)$ 或 $f_U(x)$ 具有简单结构

(2) $x \in \{0,1\} \rightarrow x \in [0,1]$ (3)不等式约束代替等式约束

(4) $\|x\|=1 \rightarrow X \in S^n,\ Tr(X)=1,\ X \succcurlyeq 0\ (X=xx^T)$

实际案例——回归分析

➢ **多元线性回归**

$$\min_{a,b}\sum_i\left(a^T x^i+b-y^i\right)^2 \quad \theta=\begin{pmatrix} b\\ a\end{pmatrix}, X=\begin{pmatrix} 1 & \left(x^1\right)^T\\ 1 & \left(x^2\right)^T\\ \vdots & \vdots\\ 1 & \left(x^N\right)^T\end{pmatrix}, y=\begin{pmatrix} y^1\\ y^2\\ \vdots\\ y^N\end{pmatrix}$$

$$\|z\|=\sqrt{\sum_{i=1}^N z_i^2}, \|z\|_1=\sum_{i=1}^N |z_i|$$

$$\Rightarrow \min_\theta \|X\theta-y\|^2$$

➢ **一元二次回归**

$$\min_a\sum_i\left(a_0+a_1x^i+a_2\left(x^i\right)^2-y^i\right)^2 \quad \theta=\begin{pmatrix} a_0\\ a_1\\ a_2\end{pmatrix}, X=\begin{pmatrix} 1 & x^1 & \left(x^1\right)^2\\ 1 & x^2 & \left(x^2\right)^2\\ \vdots & \vdots & \vdots\\ 1 & x^N & \left(x^N\right)^2\end{pmatrix}$$

$$(Lasso)\min_\theta \|X\theta-y\|^2+\mu\|\theta\|_1$$

$$(\text{岭回归})\min_\theta \|X\theta-y\|^2+\mu\|\theta\|^2$$

$$(\mathrm{TV})\min_\theta \|X\theta-y\|^2+\mu\|D\theta\|^2$$

实际案例——回归分析

- **考虑数据集$\{((0, 1), 2), ((1, 1), 3)\}$，线性回归训练模型为**

$$\min_{a,b} \left(a_2 + b - 2\right)^2 + \left(a_1 + a_2 + b - 3\right)^2 \longrightarrow \min_{\theta} \left\| \begin{pmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} b \\ a_1 \\ a_2 \end{pmatrix} - \begin{pmatrix} 2 \\ 3 \end{pmatrix} \right\|^2$$

$$\min_{\theta} \left\| \begin{pmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} b \\ a_1 \\ a_2 \end{pmatrix} - \begin{pmatrix} 2 \\ 3 \end{pmatrix} \right\|^2 + \mu\left(|b| + |a_1| + |a_2|\right)$$

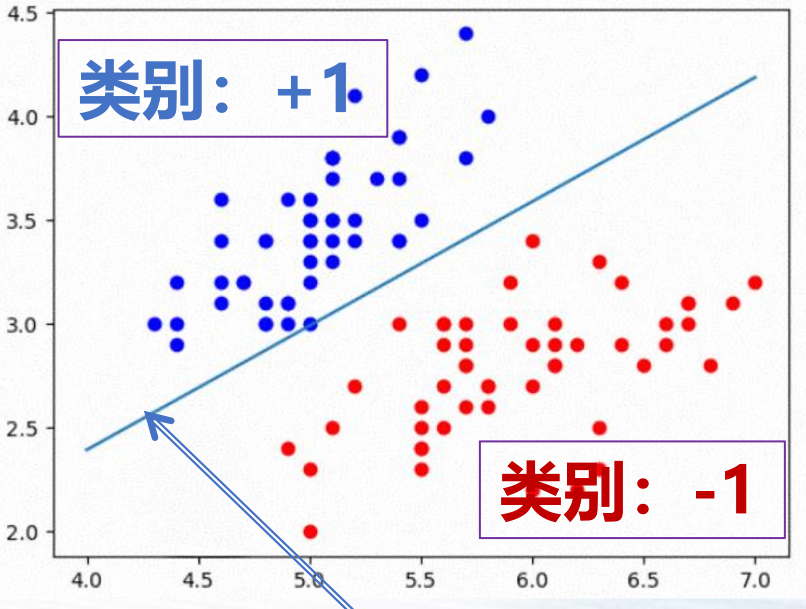
$$\min_{\theta} \left\| \begin{pmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} b \\ a_1 \\ a_2 \end{pmatrix} - \begin{pmatrix} 2 \\ 3 \end{pmatrix} \right\|^2 + \mu\left(b^2 + a_1^2 + a_2^2\right)$$

问题：考虑数据集$\{(1,1),(2,4),(3,9)\}$，给出一元二次回归方程？

实际案例——SVM/SVR

➢ 数据分类

如何找到一超平面将这两类数据划分为不同的类别？



超平面: $w^T x + b = 0$

➢ 简单逻辑运算

▫ 逻辑与

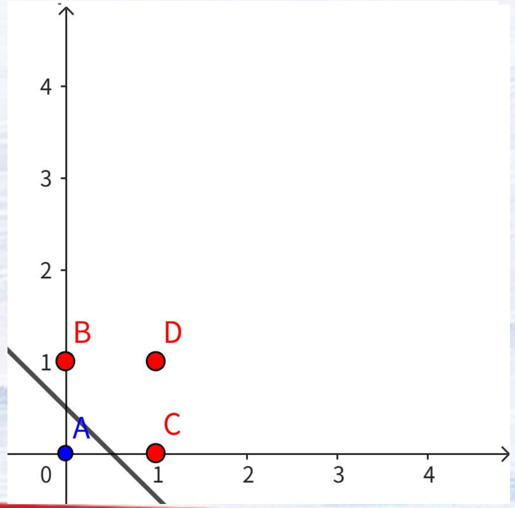
$x^1 = 0, x^2 = 0, z = 0 \quad x^1 = 1, x^2 = 0, z = 0$

$x^1 = 0, x^2 = 1, z = 0 \quad x^1 = 1, x^2 = 1, z = 1$

▫ 逻辑或

$x^1 = 0, x^2 = 0, z = 0 \quad x^1 = 1, x^2 = 0, z = 1$

$x^1 = 0, x^2 = 1, z = 1 \quad x^1 = 1, x^2 = 1, z = 1$



实际案例——SVM/SVR

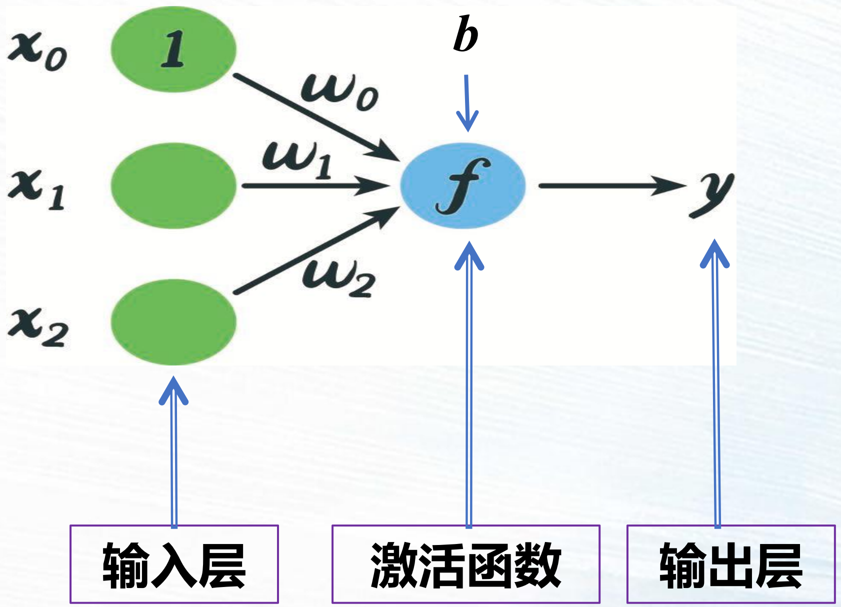
- **类型：** 二分类的线性分类模型
- **输入：** 实例的特征向量，**输出：** 实例的类别（+1，-1）
  - 数据集：$T=\{(x_1,y_1),(x_2,y_2),\ldots,(x_N,y_N)\}$，$x_i \in R^n$，$y_i \in \{-1,+1\}$
  - 简单逻辑运算：$\{(x^1, x^2)\}$，$y_i \in \{-1,+1\}$，$z_i = sgn(y_i+1)$
- **目的：** 求出训练数据进行线性划分的分离超平面 $w^T x + b = 0$
- 1964年由 Vapnik，Chervonenkis，Cortes 等提出

实际案例——SVM/SVR

- **模型**：$y=sign(w^Tx+b)$

权重　偏置

$$sign(a)=\begin{cases}+1, & a\geq 0,\\ -1, & a<0.\end{cases}$$



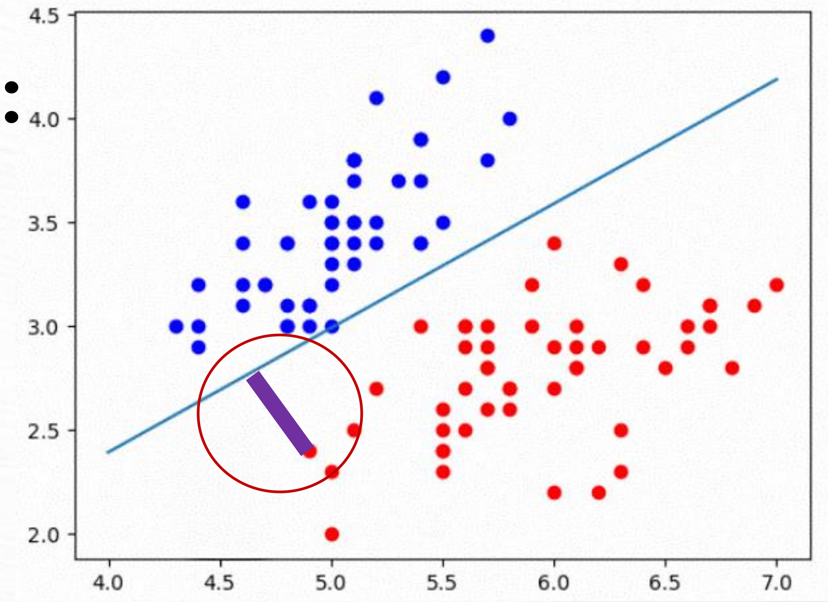
- **它是神经网络的基础。**

利用已知训练数据集$\{(x_i,y_i)\}$，如何求得SVM模型即模型参数w（权重）和b（偏置），对新输入实例给出其对应输出类别

实际案例——SVM/SVR

➢ 对于正确分类的数据(x_i, y_i)，点到超平面距离：

$$d_i = \frac{\left|w^T x_i + b\right|}{\|w\|}$$



注意：绝对值函数在0处是不可导的

➢ 正确分类点到超平面距离可重写为

$$d_i = \frac{\left|w^T x_i + b\right|}{\|w\|} = \frac{y_i\left(w^T x_i + b\right)}{\|w\|}$$

正确分类点解读：

1. $w^T x_i + b > 0,\ y_i = +1 > 0$

2. $w^T x_i + b < 0,\ y_i = -1 < 0$

实际案例——SVM/SVR

$$\max_{w.b}\min_{i}\frac{y_i\left(w^T x_i+b\right)}{\|w\|} \longrightarrow \max_{w.b,\gamma}\gamma \quad s.t.\ \frac{y_i\left(w^T x_i+b\right)}{\|w\|}\geq\gamma.$$

- **取$\|w\|_2 = 1/\gamma$，则硬间隔SVM模型为**

$$\min_{w.b}\frac{1}{2}\|w\|^2 \quad s.t.\ y_i\left(w^T x_i+b\right)\geq 1.$$

- **例：考虑数据集：正实例点$x_1=(3,3)^T$, $x_2=(4,3)^T$, 负实例点$x_3=(1,1)^T$, 试用硬间隔SVM算法求SVM模型$f(x)=sign(w^Tx + b)$.**

$$\min_{w.b}\frac{1}{2}\left(w_1^2+w_2^2\right)$$

$$s.t.\ \ 3w_1+3w_2+b\geq 1, 4w_1+3w_2+b\geq 1, -w_1-w_2-b\geq 1.$$

问题：试用硬间隔SVM算法实现逻辑与、逻辑或？

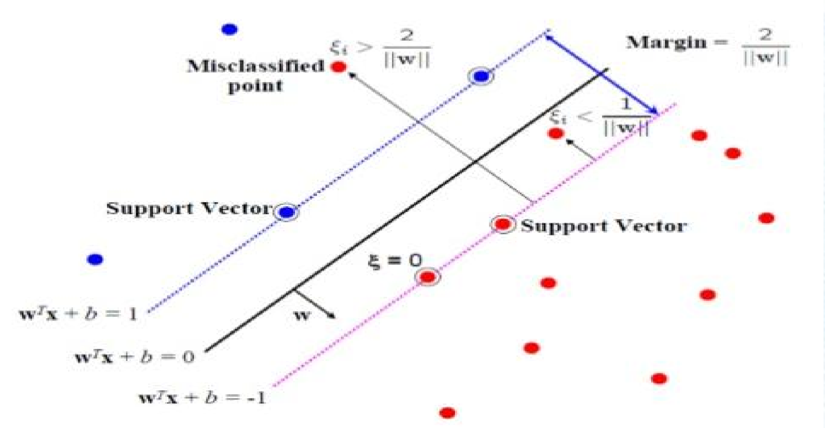
实际案例——SVM/SVR

➢ **软间隔SVM模型为**

$$\min_{w,b,\xi} \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N}\xi_i$$

$$s.t.\ \ y_i\left(w^T x_i + b\right) \ge 1-\xi_i, \xi_i \ge 0, i = 1,2,\ldots,N.$$

可用于药物筛选中ADMET（吸收、分布、代谢、排泄和毒性）预测



➢ **例：考虑数据集：正实例点$x_1=(3,3)^T$, $x_2=(4,3)^T$, 负实例点$x_3=(1,1)^T$, 试用软间隔SVM算法求SVM模型$f(x)=sign(w^Tx + b)$.**

$$\min_{w,b,\xi} \frac{1}{2}\left(w_1^2 + w_2^2\right) + C\left(\xi_1 + \xi_2 + \xi_3\right)$$

$$s.t.\ \ 3w_1 + 3w_2 + b \ge 1-\xi_1, 4w_1 + 3w_2 + b \ge 1-\xi_2, -w_1 - w_2 - b \ge 1-\xi_3, \xi_1, \xi_2, \xi_3 \ge 0.$$

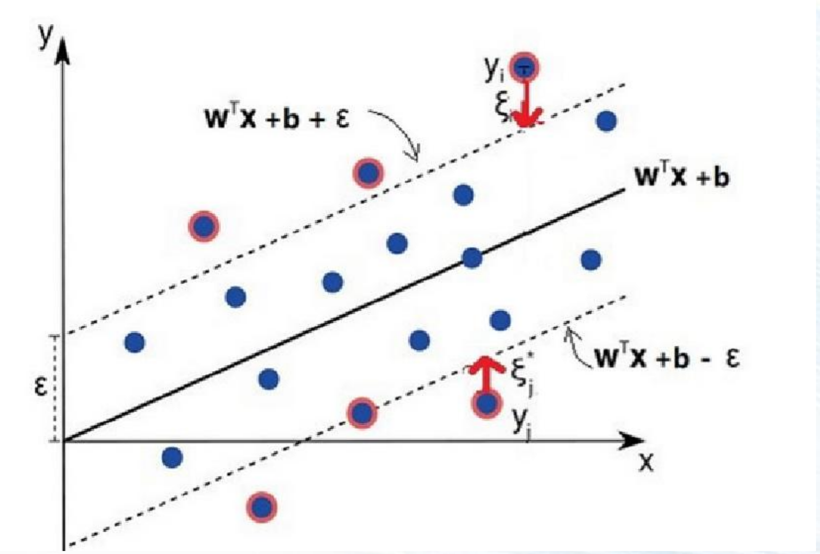
问题：试用软间隔SVM算法实现逻辑与、逻辑或?

*课程思政：软间隔SVM 允许数据结果存在一些偏差，我们做人和做事中也要养成包容的性格

实际案例——SVM/SVR

➢ **SVR模型为**

$$\min_{w,b} \frac{1}{2}\|w\|^2 \quad s.t. \left|w^T x_i + b - y_i\right| \le \varepsilon.$$



➢ **考虑数据集*{((0, 1), 2), ((1, 1), 3)}*, 线性回归训练模型为**

$$\begin{aligned} &\min_{w,b} \frac{1}{2}\left(w_1^2 + w_2^2\right) \\ s.t.\ &\left|w_2 + b - 2\right| \le \varepsilon, \\ &\left|w_1 + w_2 + b - 3\right| \le \varepsilon. \end{aligned}$$

实际案例——Markowitz均值-方差投资组合

- **均值方差(MV) 模型**中, 希望通过寻求最优投资组合以降低风险、提高收益.
- 这时决策变量x_i 表示在第i 项资产上的投资额, 向量$x \in R^n$ 表示整体的投资分配;
- 约束条件可能为总资金数、每项资产的投资额限制、最低收益等.

$$\min_{x} \sum_{i=1}^{n} x_i^2 Var(r_i) + \sum_{i \neq j} x_i x_j Cov(r_i, r_j)$$

$$s.t. \sum_{i=1}^{n} x_i E(r_i) \geq \bar{\alpha}, \sum_{i=1}^{n} x_i = 1, x \geq 0.$$

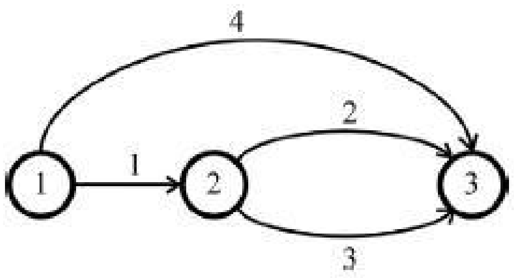
二次规划

实际案例——交通网络分析

OD: $x_1(1 \to 2), x_2(1 \to 3), x_3(2 \to 3)$;

Link: 1, 2, 3, 4;

Path: $y_1(\{1\}, 1 \to 2), y_2(\{1, 2\}, 1 \to 3), y_3(\{1, 3\}, 1 \to 3),$
$y_4(\{4\}, 1 \to 3), y_5(\{2\}, 2 \to 3), y_6(\{3\}, 2 \to 3).$



- **link/path 关联矩阵:**

$$\Delta = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

- **O-D/path 关联矩阵:**

$$\Gamma = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$

实际案例——交通网络分析

- y_j : j- 路径上的流量;
- x_i: i-OD 上的流量;
- z_k: k-link 上的流量.

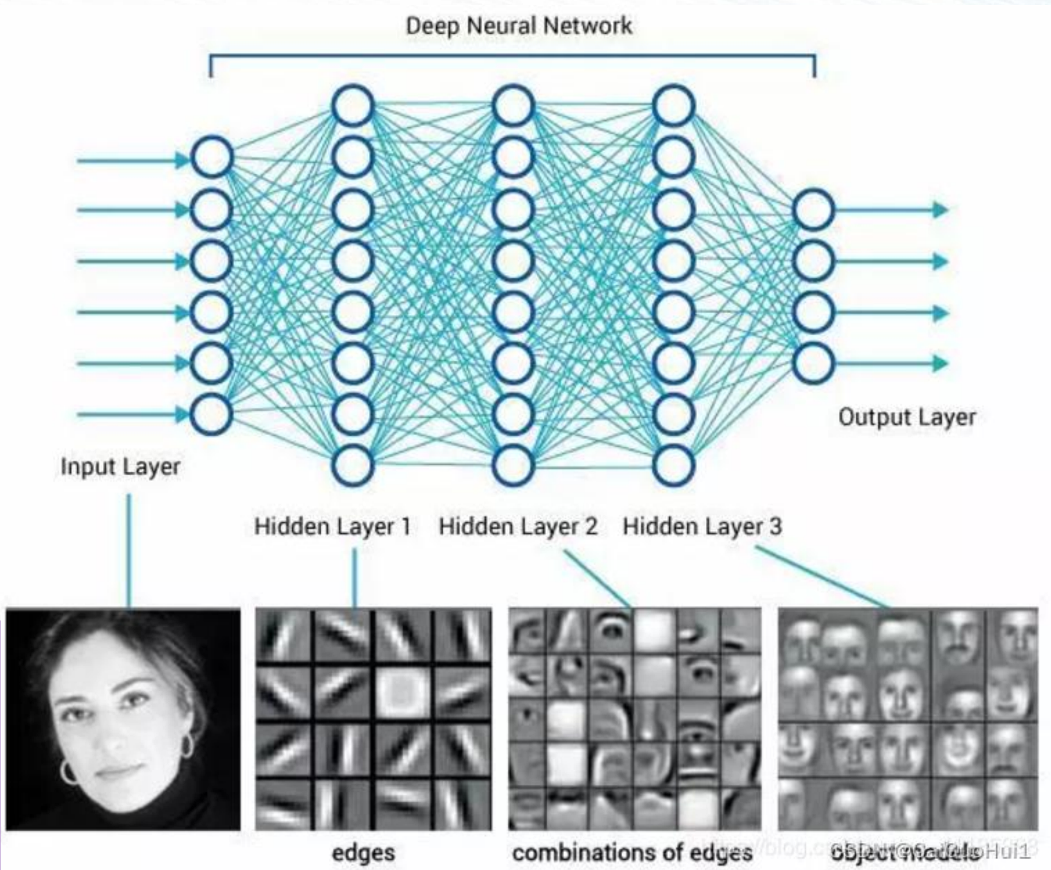
$$x = \Gamma y,\ z = \Delta y.$$

- 考虑目标: $\ell(z,w_0)$, z 和真实流量w_0 的损失, 于是模型为

$$\begin{aligned}
\min_{x,y,z}\ & \ell\left(z, w_0\right) \\
s.t.\ \ & x = \Gamma y, z = \Delta y, \\
& y \geq 0, z \geq 0.
\end{aligned}$$

实际案例——深度神经网络

- 输入：特征 x^{input}，标签 y^{output}
- 权重W_i，偏置b_i
- 激活函数σ_i
- 正则项$R(W)$



- **深度学习算法:**

可用于药物筛选中
ADMET（吸收、分布、代谢、排泄和毒性）预测

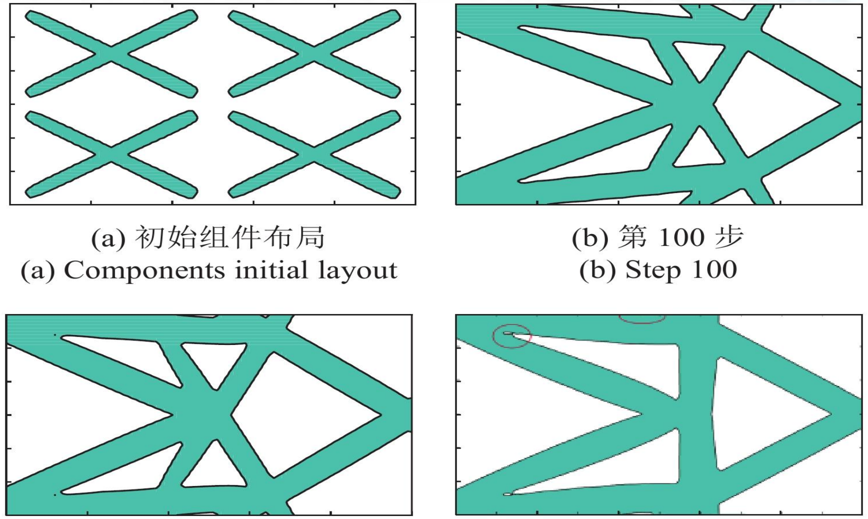
$$\min_{W,b}\frac{1}{N}\sum_{i=1}^{N}\left(\sigma_L\left(W_L\left(\cdots\sigma_1\left(W_1x^{input}+b_1\right)+\cdots\right)+b_L\right)-y^{output}\right)^2+\lambda R\left(W\right)$$

实际案例——基于SIMP方法的拓扑优化

- **采用插值函数后拓扑优化：**

$$\min_{x_e} U^T KU = \sum_{e=1}^{N}\left(E_{\min} + x_e^p\left(E_0 - E_{\min}\right)\right)u_e^T k_0 u_e$$

$$s.t.\ \frac{V\left(x_e\right)}{V_0} \le f, KU = F, 0 \le x_{\min} \le x_e \le x_{\max}.$$



(a) 初始组件布局
(a) Components initial layout

(b) 第 100 步
(b) Step 100

(c) 第 400 步
(c) Step 400

(d) 优化结果
(d) Optimization result

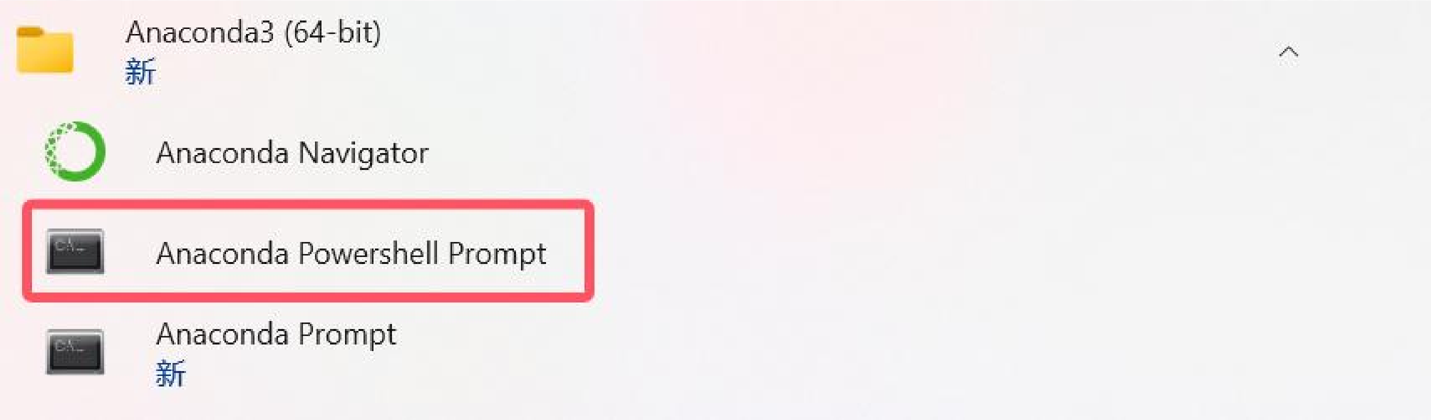
- U：位移向量；K：整体刚度矩阵；F：外荷载向量
- x_e：单元设计变量
- $V(x_e)$：迭代过程中实体单元体积
- V_0：初始状态下结构设计域总体积
- f：材料用量的体积分数限值

求解器与大模型

- 由于实际问题往往无法显式求解，因此常用求解器或设计算法进行求解。
- 常用求解器
- CVX
- 面向凸规划，线性规划、二次规划、二阶锥规划、半定规划和一些复杂的凸优化问题, 如含$\ell 1$ 范数的凸规划.
- Gurobi
- 大规模线性规划、二次规划、混合整数规划、多目标优化、并行计算和分布式计算.
- CPLEX
- 大规模线性规划、二次规划、混合整数规划、二次约束规划和二阶锥规划.
- COPT
- 大规模混合整数规划、线性规划、半定规划、二阶锥规划、凸二次规划、凸二次约束规划

求解器与大模型

- **STEP1：安装Anaconda3-2023.03-1-Windows-x86_64**



- **STEP2：pip install cvxpy**
- **STEP3：pip install "cvxpy[CVXOPT,GLOP,GLPK,MOSEK]"**
- **STEP4：下载mosek进行安装，并生成mosek.lic到C:\Users\zhenh\Mosek**

https://www.mosek.com/products/academic-licenses/

求解器与大模型

- **STEP5：** 下载Gurobi 12.0.3进行按照到C:\gurobi, 并激活到C:\gurobi
- **STEP6：** 以管理员权限运行Anaconda Prompt
- **STEP7：** conda install -c gurobi gurobi　过程需要输入y
- **STEP8：** Win+R，输入gurobi
- **STEP9：** 检验环境

```
from cvxpy import *

from gurobipy import *
```

求解器与大模型——最小二乘

➢ 准备数据

```
import numpy as np
m = 20                    n = 15
np.random.seed(1)
A = np.random.randn(m, n)
b = np.random.randn(m)
```

➢ 加载CVX

```
import cvxpy as cp
```

设置决策变量

```
x = cp.Variable(n)
x = cp.Variable(n, integer=True)
```

➢ 录入优化问题

```
cost = cp.sum_squares(A @ x - b)          cp.norm2(A @ x - b)**2
prob = cp.Problem(cp.Minimize(cost))
```

➢ 求解

```
prob.solve()
print("\nThe optimal value is", prob.value)
print("\nThe optimal x is",x.value)
print("The norm of the residual is ", cp.norm(A @ x - b, p=2).value)
```

求解器与大模型——MV模型

➢ 准备数据

```
import numpy as np
mu = np.array([[1.6],[0.6],[0.5],[1.1],[0.9],[2.3],[1.7],[0.7],[0.9],[0.3]])
np.random.seed(1)                    n = 10
Sigma = np.random.randn(n, n)
Sigma = Sigma.T.dot(Sigma)
```

➢ 加载CVX

```
import cvxpy as cp
```

设置决策变量

```
w = cp.Variable(n)
```

➢ 录入优化问题

```
ret = mu.T @ w                risk = cp.quad_form(w, Sigma)                gamma = cp.Parameter(nonneg=True)
prob = cp.Problem(cp.Maximize(ret - gamma * risk), [cp.sum(w) == 1, w >= 0])
```

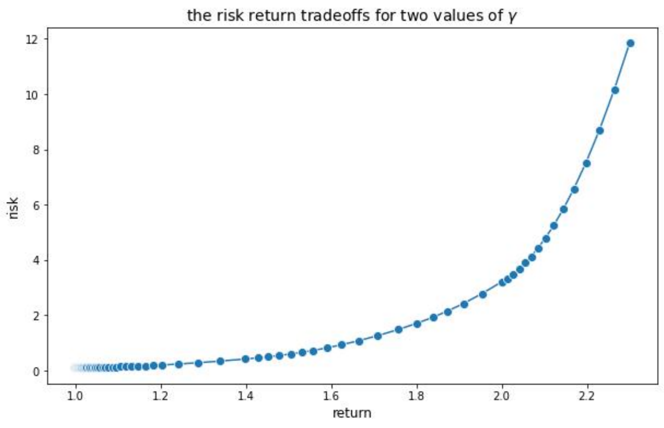
➢ 求解

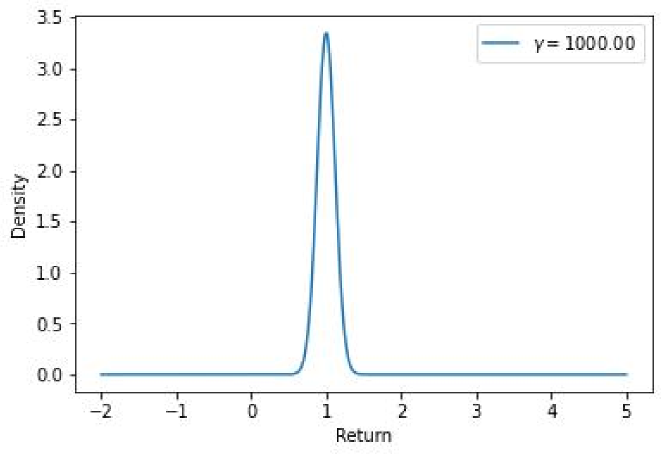
```
_SAMPLES = 100                gamma_vals = np.logspace(-2, 3, num=_SAMPLES)        ret_data = []        risk_data = []
for _i in range(_SAMPLES):
    gamma.value = gamma_vals[_i]                    prob.solve()
    ret_data.append(ret.value)                      risk_data.append(risk.value)
```

求解器与大模型——MV模型

➢ 画图

```
import seaborn as sns
import matplotlib.pyplot as plt
ret_data = np.array(ret_data).reshape(100)
risk_data = np.array(risk_data).reshape(100)
g = sns.lineplot(x=ret_data, y=risk_data, marker='o', markersize=8)
plt.title('the risk return tradeoffs for two values of $\gamma$', fontsize=14)
plt.xlabel('return', fontsize=12)                plt.ylabel('risk', fontsize=12)
plt.show()
import scipy.stats as spstats
x = np.linspace(-2, 5, 1000)
plt.plot(x,spstats.norm.pdf(x, ret.value, risk.value),label="$\\gamma = %.2f$" % gamma.value)
plt.xlabel("Return")                plt.ylabel("Density")
plt.legend(loc="upper right")
plt.show()
```





求解器与大模型——SDP

➢ 准备数据

```
import numpy as np
n = 3                    p = 3
np.random.seed(1)        C = np.random.randn(n, n)
A = []                   b = []
for i in range(p):
    A.append(np.random.randn(n, n))
    b.append(np.random.randn())
```

➢ 加载CVX

```
import cvxpy as cp
```

设置决策变量

```
X = cp.Variable((n,n), symmetric=True)
```

➢ 录入优化问题

```
constraints = [X >> 0]
constraints += [cp.trace(A[i] @ X) == b[i] for i in range(p)]
prob = cp.Problem(cp.Minimize(cp.trace(C @ X)),constraints)
```

➢ 求解

```
prob.solve()
print("\nThe optimal value is", prob.value)
print("\nThe optimal x is",x.value)
```

求解器与大模型——图像恢复

➢ 准备数据

```
import matplotlib.pyplot as plt
import numpy as np
u_orig = plt.imread("C:/Users/zhenh/Desktop/loki512.png")
u_corr = plt.imread("C:/Users/zhenh/Desktop/loki512_corrupted.png")
rows, cols, kols = u_orig.shape
known = np.zeros((rows, cols))
for i in range(rows):
    for j in range(cols):
        for k in range(kols):
            if u_orig[i, j, 1] == u_corr[i, j, 1]:
                known[i, j] = 1
```

➢ 加载CVX

```
import cvxpy as cp
```

设置决策变量

```
U = cp.Variable(shape=(rows, cols))
```

➢ 录入优化问题

```
obj = cp.Minimize(cp.tv(U))
constraints = [cp.multiply(known, U) == cp.multiply(known, u_corr[:,:,1])]
prob = cp.Problem(obj, constraints)
```

➢ 求解

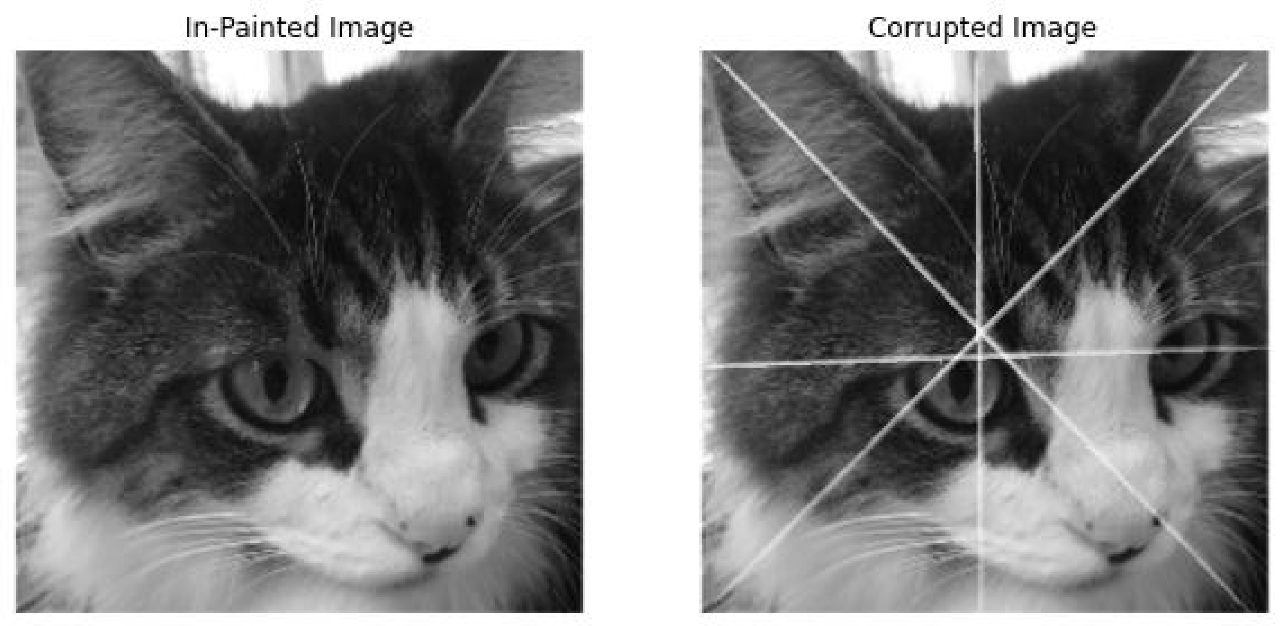
```
prob.solve(verbose=True, solver=cp.SCS)
print("optimal objective value: {}".format(obj.value))
```

求解器与大模型——图像恢复

➢ 画图

```
fig, ax = plt.subplots(1, 2, figsize=(10, 5))
ax[0].imshow(U.value, cmap='gray')
ax[0].set_title("In-Painted Image")
ax[0].axis('off')

ax[1].imshow(u_corr, cmap='gray')
ax[1].set_title("Corrupted Image")
ax[1].axis('off')
```



求解器与大模型——Gurobi

准备模块库

```
import gurobipy as gp

from gurobipy import GRB
```

创建模型

```
model = gp.Model("my_model")
```

添加变量

```
x = model.addVars(range(2),vtype=GRB.CONTINUOUS, name="x")

y = model.addVars(range(2),vtype=GRB.BINARY, name="y")

z = model.addVars(range(2),lb = 0,ub=2.3,vtype=GRB.INTEGER, name="z")
```

求解器与大模型——Gurobi

□ 目标函数

```
model.setObjective(2 * (x[0] + x[1]) + 3 * (y[0] + y[1]) + 4 * (z[0] + z[1]),
GRB.MAXIMIZE)
```

□ 约束函数

```
model.addConstr(x[0] + y[0] <= 10, "c1")

model.addConstr(x[1] + y[1] == 10, "c2")
```

□ 求解模型

```
NLP.Params.NonConvex=2
```

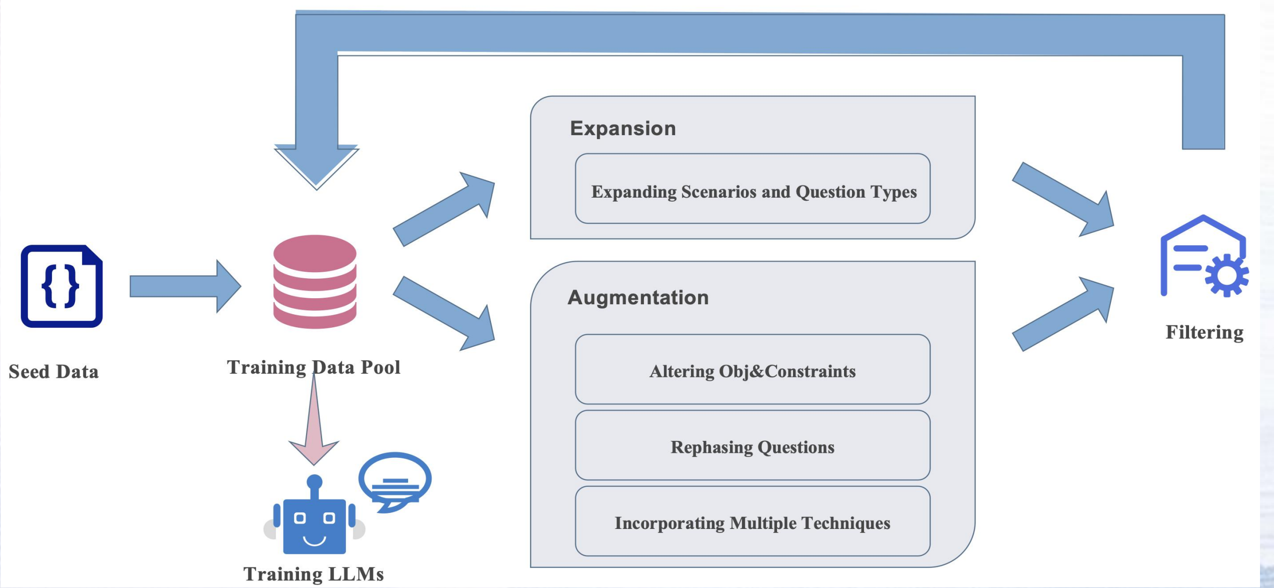
```
model.optimize()
```

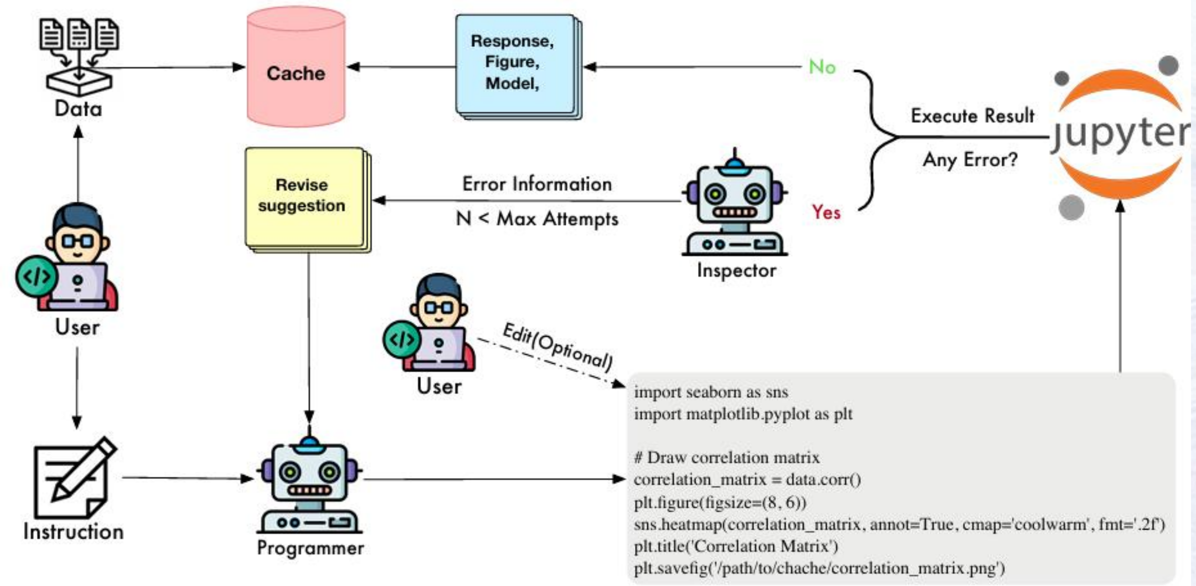
□ 获取结果

```
for i in x.keys():
    print('x[%d] = %f' % (i,x[i].x))
```

求解器与大模型——领域大模型

- 通用大语言模型：Deepseek、ChatGPT
- 运筹优化建模领域专用大模型：ORLM
- 数据科学领域专用大模型：LAMBDA





https://github.com/Cardinal-Operations/ORLM
https://wisemodel.cn/codes/CardinalOperations/ORLM/intro

https://github.com/AMA-CMFAI/LAMBDA
https://www.polyu.edu.hk/ama/cmfai/lambda.html

算法基础——终止准则

- 常采用**迭代算法**求解实际问题．为使算法能在**有限步内终止**，一般会通过一些**收敛准则**来保证迭代停在问题的一定精度逼近解上．
- **无约束优化**问题常用的收敛准则

$$\frac{\left|f\left(x_{k+1}\right)-f\left(x_k\right)\right|}{\max\left\{\left|f\left(x_k\right)\right|,1\right\}}\le\varepsilon,\left\|\nabla f\left(x_k\right)\right\|\le\varepsilon,\frac{\left|x_{k+1}-x_k\right|}{\max\left\{\left\|x_k\right\|,1\right\}}\le\varepsilon$$

- **约束优化**问题，还需要考虑**约束违反度**，常用的收敛准则

$$g_i\left(x_k\right)\le\varepsilon,\left|h_j\left(x_k\right)\right|\le\varepsilon$$

- 只能反映迭代点列接近收敛，但**不能代表收敛到优化问题的最优解**．

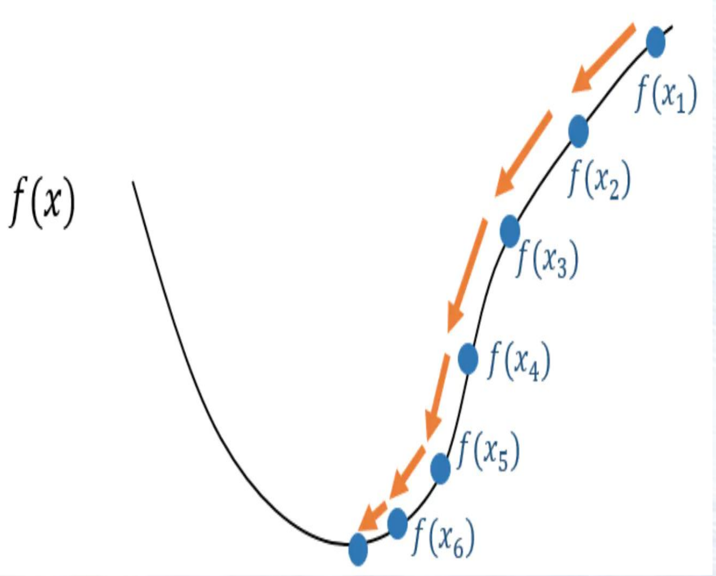
＊课程思政：算法终止准则对数据偏差具有一定的包容性，我们做人和做事中也要养成包容的性格.

算法基础——收敛性

- 在算法设计中，还需考虑算法产生的点列是否收敛到优化问题的解．给定**初始点x_0**，记算法迭代产生的点列为$\{x_k\}$．如果$\{x_k\}$ 在某种范数$\|\cdot\|$的意义下满足

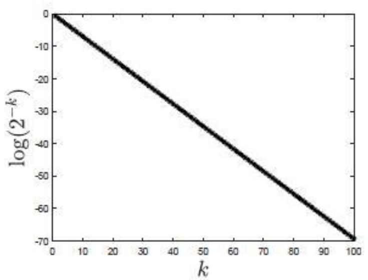
$$\lim_{k\to\infty}\left\|x_k - x^*\right\| = 0$$

且收敛的点x^* 为一个**局部**（**全局**）极小解，则称该点列收敛到局部（全局）极小解，相应的算法称为是依点列收敛到局部（全局）极小解的．

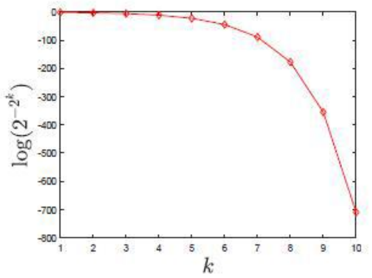


算法基础——收敛速度

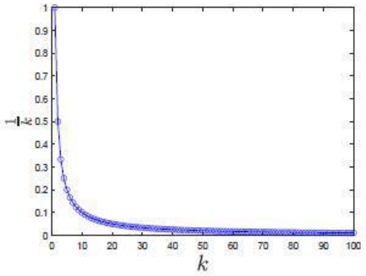
- 设$\{x_k\}$为算法产生的迭代点列且收敛于x^*
- $a\epsilon(0,1)$, $b=1$: 算法（点列）Q- 线性收敛；$a>0$, $b=2$: 算法（点列）Q-二次收敛. 对充分大的k有 $\left\|x_{k+1}-x^*\right\| / \left\|x_k-x^*\right\|^b \le a$
- $a=0$: 算法（点列）Q- 超线性收敛；$a=1$: 算法（点列）Q-次线性收敛.对充分大的k有 $\lim\limits_{k\to\infty}\left\|x_{k+1}-x^*\right\| / \left\|x_k-x^*\right\| = a$
- 点列{2^{−k}} 是Q- 线性收敛的
- 点列{2^{−2^{k}}} 是Q- 二次收敛的（也是Q- 超线性收敛的）
- 点列{ 1/k} 是Q- 次线性收敛的.



(a) Q-线性收敛



(b) Q-二次收敛



(c) Q-次线性收敛

算法基础——收敛速度

- 算法（点列）R- 线性收敛: 设$\{x_k\}$为算法产生的迭代点且收敛于x^*，若存在Q- 线性收敛于0的非负序列t_k并且对任意的k成立 $\left\|x_k - x^*\right\| \le t_k$
- 类似地，可定义R-超线性收敛和R-二次收敛等收敛速度.
- 算法复杂度：设x^* 为全局极小点，某一算法产生的迭代序列$\{x_k\}$满足

$$\left\|f\left(x_k\right)-f\left(x^*\right)\right\| \le c/\sqrt{k}$$

- 如果计算算法满足精度$f(x_k) - f(x_*) \le \varepsilon$ 所需迭代次数，只需令c/√k≤ ε 则得到k ≥ c^2/ε^2，因此该优化算法对应复杂度为N(ε) = $O(1/\varepsilon^2)$.
- **练习：考虑如下点列，给出其Q 收敛速度. (1)** $x_k = 1/k!\ , k = 1, 2, \ldots;$
- **(2)** $y_k = (1+1/2^k)(\cos k, \sin k)$; **(3)** $f(y_k) = \|y_k\|^2$.

南昌大学
NANCHANG UNIVERSITY

课件资源：https://zhenhuapeng.github.io//coursematerials/

感谢观看

大数据优化：理论、算法及其应用

——来源于《最优化计算方法》（高教社）

★★★ 主讲人：彭振华 ★★★

联系方式：zhenhuapeng@ncu.edu.cn
zhenhuapeng@whu.edu.cn
15870605317（微信同号）

研究兴趣：1. 非凸非光滑优化算法与理论
2. 智能决策
3. 智能计算与机器学习

数学与计算机学院