最优化算法：简介

陈士祥

中国科学技术大学

致谢：本教案部分参考北京大学文再文教授《最优化与建模》

课程考核

(1) 课后习题(20%)，共5次，需要在布置作业后3周内提交

(2) project (15%)。要求：实现一个算法。

(3) presentation (15%)。要求：给定论文列表（也可自选），挑选你感兴趣的一篇，以小组（3人以内）形式做presentation。

(4) 期末考试(50%)

我的联系方式：shxchen@ustc.edu.cn。答疑时间：每周三下午4点-5点，办公室管理学院1307，请通过邮件预约。

助教：王健铭(ustc2017wjm@mail.ustc.edu.cn)、朱峻添(zjt1229@mail.ustc.edu.cn)

班级群：412201796

提纲

机器学习：监督学习简介

- 机器学习，人工智能
  - 计算机视觉，自然语言处理(ChatGPT), 视频生成(Sora)
  - AlphaGo, AlphaGo Zero
- 科学发现经常围绕建立函数关系：$f : X \to Y$
  - 蛋白质结构预测(Alphafold2): X: 蛋白质序列; Y: 三维结构
  - 深度势能: X: 分子结构; Y: 原子势能
- 挑战: f 非常高维、高度非线性，只有它的很少量已知的知识或者计算非常昂贵
- 机遇: 我们有很多的数据:

$$\{(x_i, y_i) \mid x_i \in X, y_i \in Y, 1 \le i \le N\}.$$

- 机器学习是构造 $\hat{f} \approx f$ 的强大工具

监督学习中典型问题形式

机器学习构造$\hat{f}$的典型方式：$\hat{f} = h(x, \theta)$

$$\min_{\theta \in \mathcal{W}} \quad \frac{1}{N} \sum_{i=1}^{N} \|x_i^\top \theta - y_i\|_2^2 + \mu\varphi(\theta) \quad \text{线性回归}$$

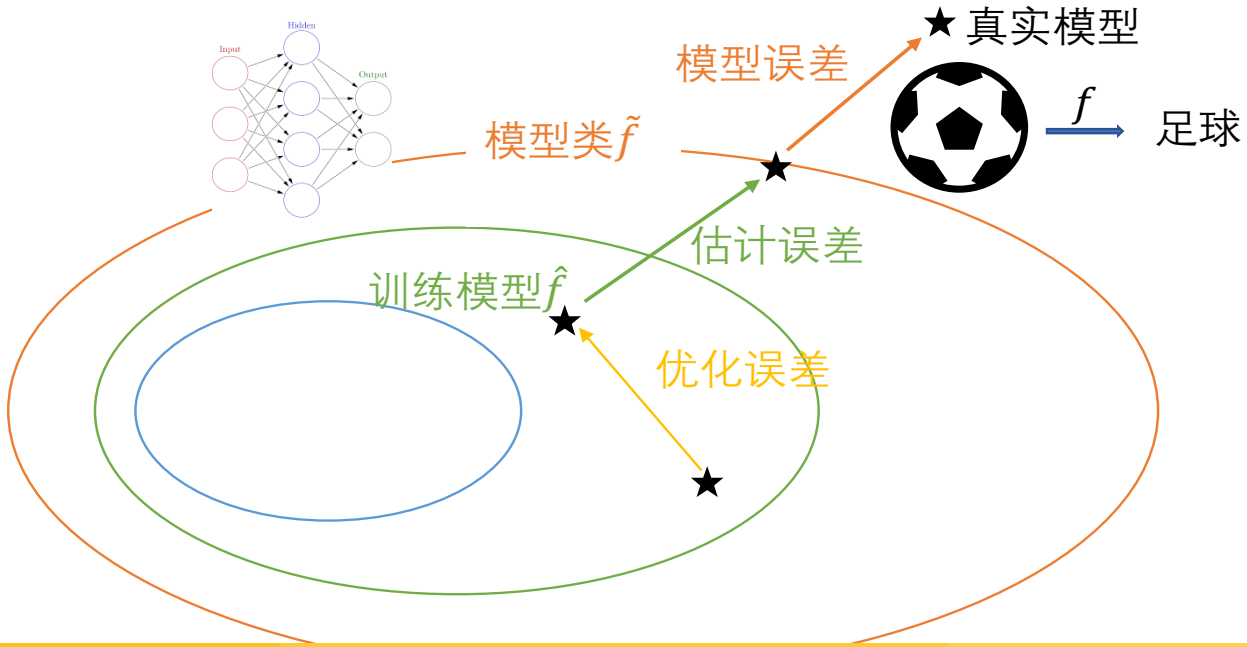
$$\min_{\theta \in \mathcal{W}} \quad \frac{1}{N} \sum_{i=1}^{N} \log(1 + \exp(-y_i x_i^\top \theta)) + \mu\varphi(\theta) \quad \text{逻辑回归}$$

$$\min_{\theta \in \mathcal{W}} \quad \frac{1}{N} \sum_{i=1}^{N} \ell(h(x_i, \theta), y_i) + \mu\varphi(\theta) \quad \text{一般形式-有限求和形式}$$

- (x_i, y_i)是给定的数据对，y_i是数据x_i对应的标签
- $\ell_i(\cdot)$: 度量模型拟合数据点i的程度(避免拟合不足)
- $\varphi(\theta)$: 避免过拟合的正则项: 岭回归$\|\theta\|_2^2$或者稀疏正则$\|\theta\|_1$等等
- $h(x, \theta)$: 线性函数、核函数或者由深度神经网络构造的模型

机器学习:逼近误差、估计误差和优化误差

- 模型误差（Approximation/Modeling error）：造成原因：使用机器学习模型逼近真实模型。
- 估计误差（Estimation Error）：造成的原因：训练数据是有限的。
- 优化误差(Optimization error):寻找最佳模型参数过程中的误差。造成的原因：问题的非凸性质，计算资源，优化算法的选择。



模型的非凸性质

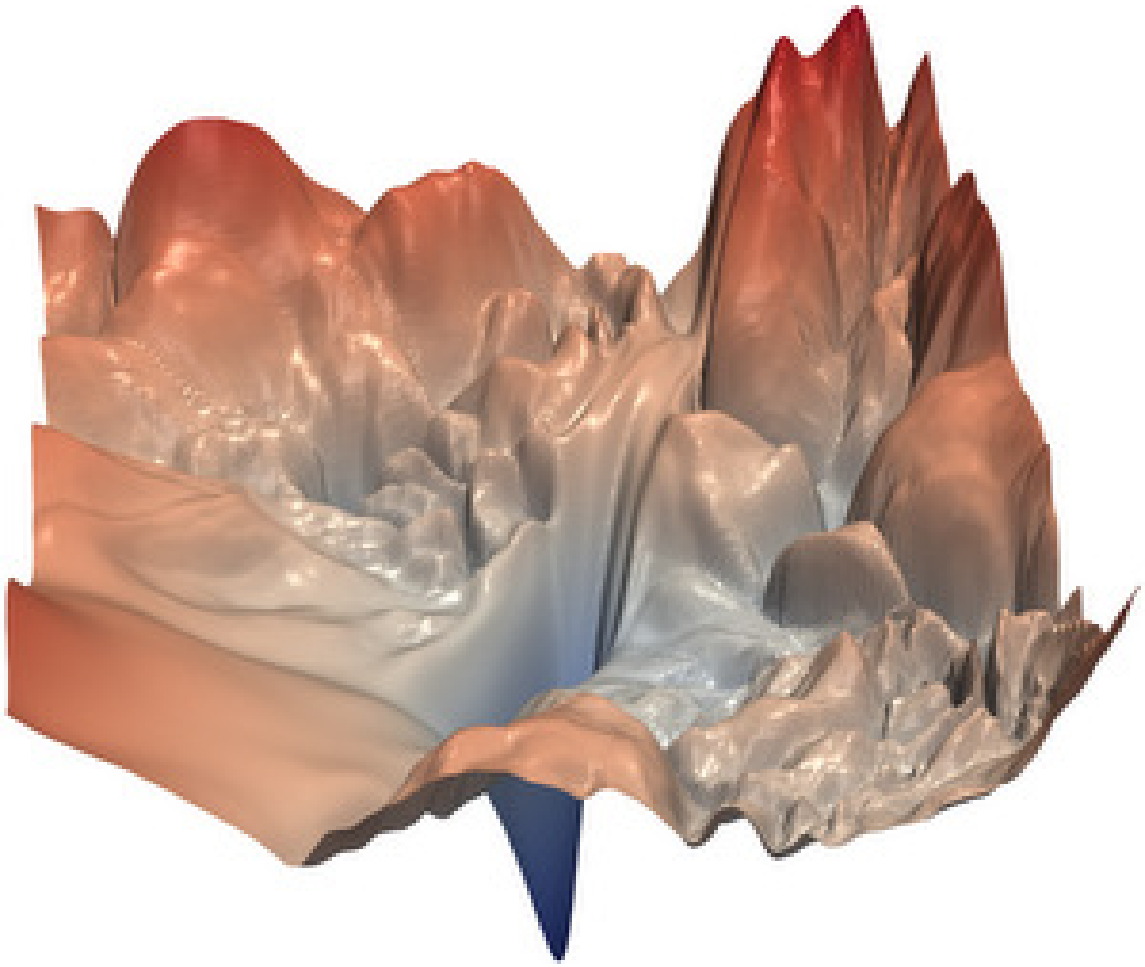
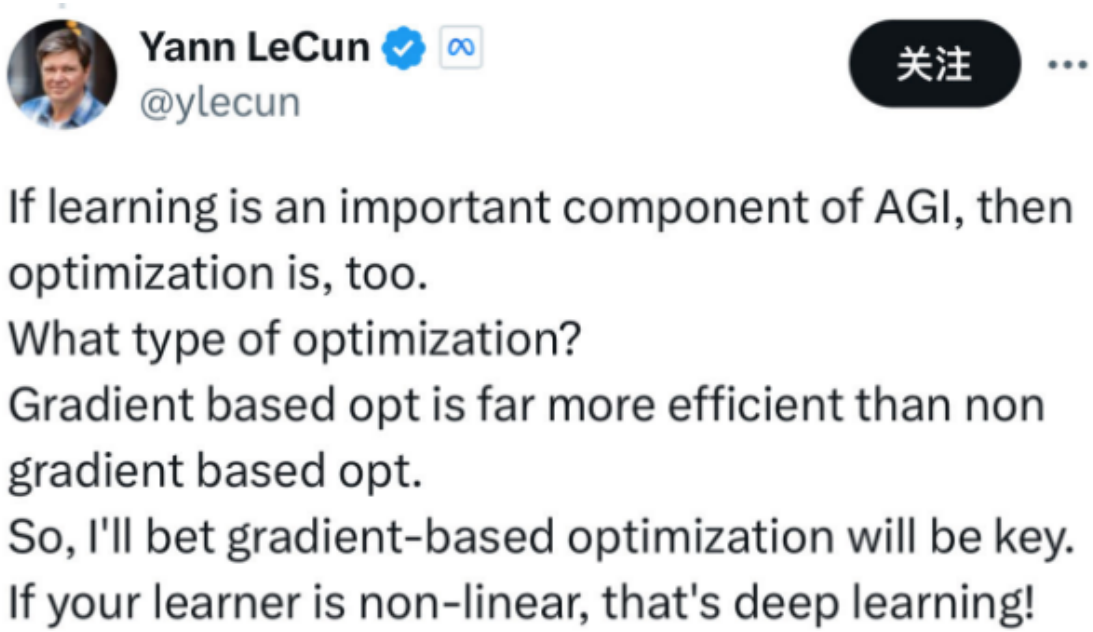


Figure: 神经网络ResNet56的损失函数局部图示。图片来源：https://github.com/tomgoldstein/loss-landscape

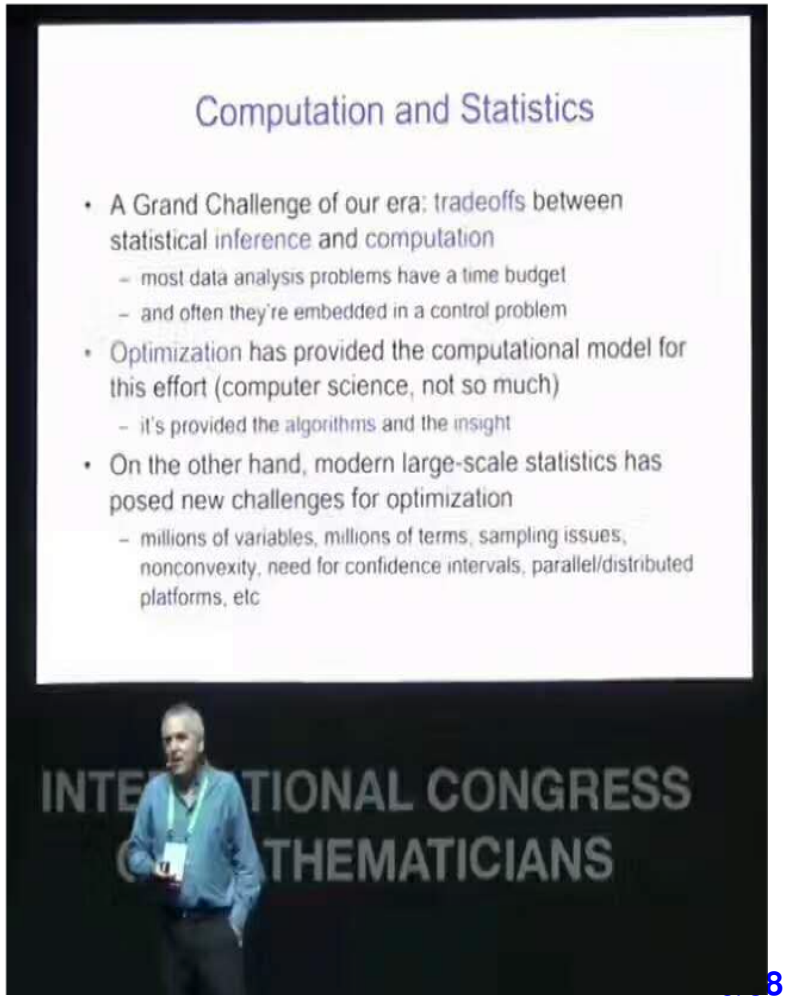
图灵奖得主Yann Le Cun(杨立昆)这样评价优化和梯度算法。



机器学习与优化

美国三院院士Michael Jordan教授国际数学家大会一小时报告：Dynamical, symplectic and stochastic perspectives on optimization – Michael Jordan – ICM2018

- 我们时代的一个很大挑战是统计推理和计算的平衡。大部分数据分析有时间限制，他们经常被嵌入到某个控制问题里。
- 最优化为这个努力提供了计算模型，给出了算法和深刻的理解。
- 现代大规模统计给优化带来了新的挑战：百万量级变量/函数项，抽样问题，非凸，置信区间，并行/分布式平台等等



提纲

最优化分类与应用

最优化问题一般可以描述为

$$\begin{aligned} \min \quad & f(x), \quad \leftarrow \text{目标函数} \\ \text{s.t.} \quad & x \in \mathcal{X} \quad \leftarrow \text{约束} \end{aligned}$$

- 按照目标函数和约束函数的形式或性质来分：
  线性规划/非线性规划、凸优化/非凸优化、非光滑优化、半定规划、锥规划、整数规划、无导数优化、几何优化、稀疏优化、低秩矩阵优化、张量优化、鲁棒优化、全局优化、组合优化、网络规划、随机优化、动态规划、带微分方程约束优化、微分流形约束优化、分布式优化等

- 具体应用涵盖：
  运筹学、供应链管理、物流管理、资产管理、统计学习、最优运输、信号处理、图像处理、机器学习、强化学习金融工程、电力系统等领域

全局和局部最优解

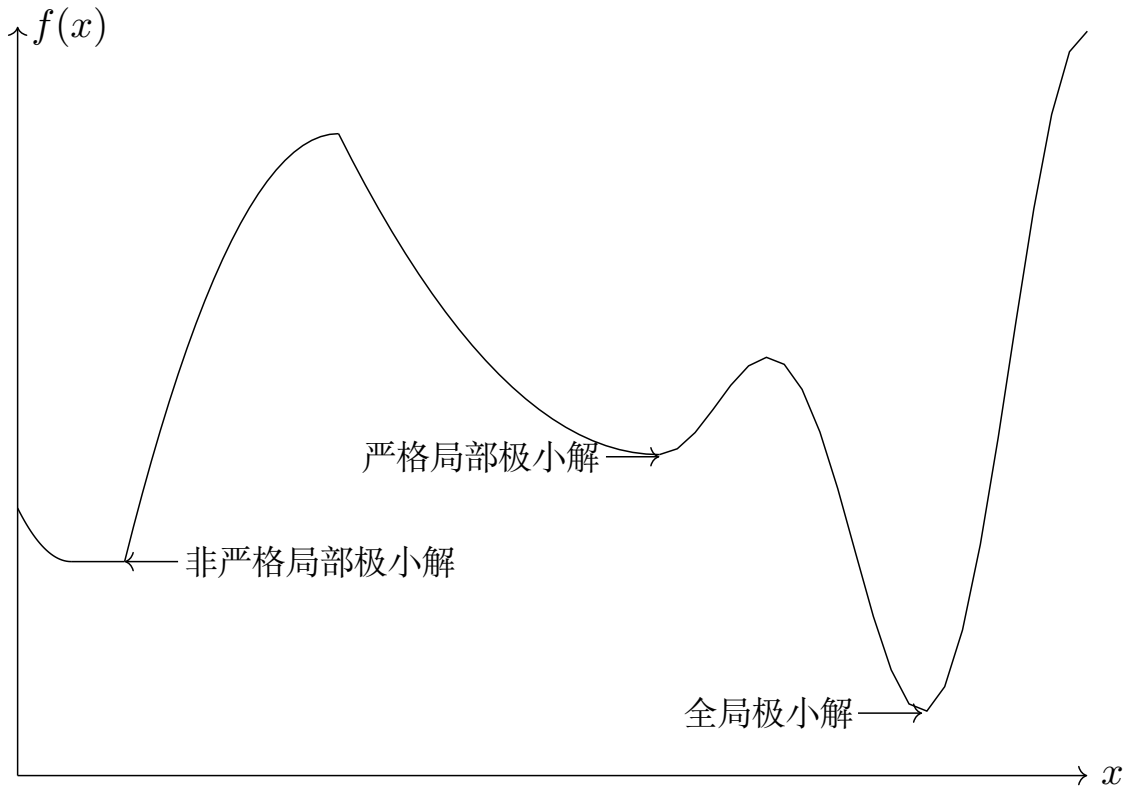


Figure: 函数的全局极小、严格局部极小和非严格局部极小解

在问题(11)的求解中，我们想要得到的是其全局最优解，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.

提纲

投资组合优化

数学建模很容易给出应用问题不同的模型，可以对应性质很不相同的问题，其求解难度和需要的算法也将差别很大．在投资组合优化中，人们希望通过寻求最优的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项资产上的投资比例，向量$x \in \mathbb{R}^n$表示整体的投资分配．
- 约束条件可能为总资金数、每项资产的最大（最小）投资额、最低收益等．
- 目标函数通常是某种风险度量．
- 如果是极小化收益的方差，则该问题是典型的二次规划.
- 如果极小化风险价值(value at risk)函数，则该问题是混合整数规划
- 如果极小化条件风险价值（conditional value at risk)函数，则该问题是非光滑优化，也可以进一步化成线性规划．

投资组合优化

- r_i, 随机变量, 股票的回报率i
- x_i, 投资于股票的相对金额i
- 回报: $r = r_1x_1 + r_2x_2 + \ldots + r_nx_n$
- 期望回报: $R = E(r) = \sum E(r_i)x_i = \sum \mu_i x_i$
- 风险: $V = Var(r) = \sum_{i,j} \sigma_{ij}x_ix_j = x^\top \Sigma x$

$$\begin{aligned} \min \ & \frac{1}{2}x^\top \Sigma x, \\ \text{s.t.} \ & \sum \mu_i x_i \geq r_0 \\ & \sum x_i = 1, \\ & x_i \geq 0 \end{aligned} \qquad \begin{aligned} \min \ & \text{risk measure}, \\ \text{s.t.} \ & \sum \mu_i x_i \geq r_0 \\ & \sum x_i = 1, \\ & x_i \geq 0 \end{aligned}$$

提纲

线性回归模型

- 考虑线性模型$b = Ax + \varepsilon$
- 假设ε_i是i.i.d 高斯白噪声，即$\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$. 那么我们有

$$p(b_i \mid a_i; x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(b_i - a_i^{\mathrm{T}} x)^2}{2\sigma^2}\right),$$

则对数似然函数为

$$\ell(x) = \ln \prod_{i=1}^{m} p(b_i \mid a_i; x) = -\frac{m}{2}\ln(2\pi) - m\ln\sigma - \sum_{i=1}^{m} \frac{(b_i - a_i^{\mathrm{T}} x)^2}{2\sigma^2}.$$

- 最大似然估计是极大化对数似然函数得到最小二乘问题：

$$\min_{x \in \mathbb{R}^n} \quad \frac{1}{2}\|Ax - b\|_2^2.$$

注意，在构建最大似然估计时不需要知道ε_i的方差σ^2.

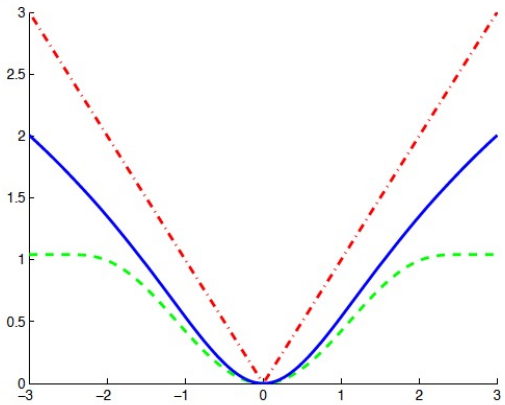
- 当假设误差是高斯白噪声时，最小二乘解就是线性回归模型的最大似然解.

线性回归模型

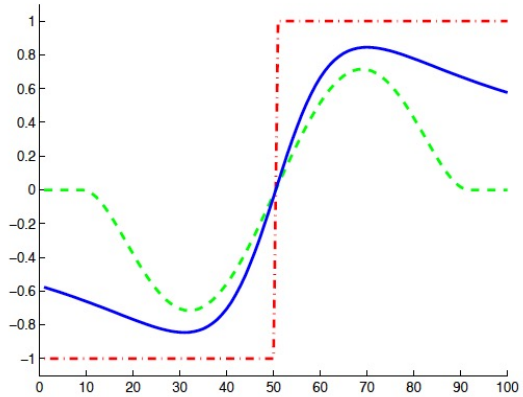
- 当ε_i不是高斯白噪声时，线性回归模型和最小二乘模型并不等价
- 在某些噪声下构造出的模型实际上为最小一乘问题：

$$\min_{x\in\mathbb{R}^n} \quad \|Ax-b\|_1.$$

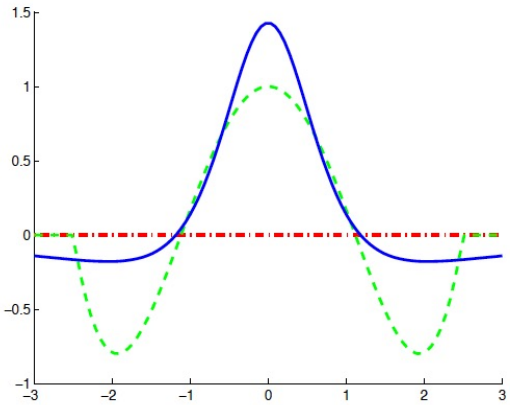
- 一般形式：$\min_{x\in\mathbb{R}^n} \quad \rho(Ax-b)$



(a) $\rho(r)$



(b) $\nabla\rho(r)$



(c) $\nabla^2\rho(r)$

Figure: Laplace (red dash-dot), Tukey (green dash), and Student's t(blue line).

Tikhonov 正则化

- 为了平衡模型的拟合性质和解的光滑性，Tikhonov 正则化或岭回归（ridge regression）添加ℓ_2范数平方为正则项.
- 假设ε_i是高斯白噪声，则带ℓ_2范数平方正则项的线性回归模型实际上是在求解如下问题：

$$\min_{x\in\mathbb{R}^n} \quad \frac{1}{2}\|Ax-b\|_2^2+\mu\|x\|_2^2.$$

  由于正则项的存在，该问题的目标函数是强凸函数，解的性质得到改善.
- 另一种常见的变形是给定参数$\sigma>0$，求解：

$$\min_{x\in\mathbb{R}^n} \quad \frac{1}{2}\|Ax-b\|_2^2, \quad \text{s.t.} \quad \|x\|_2\le\sigma. \tag{1}$$

- 上述两个问题的最优性条件类似，当参数μ和σ满足一定关系时，它们的解可以是相同的.

LASSO 问题及其变形

- 如果希望解x是稀疏的，可以添加ℓ_1范数得到LASSO 问题：

$$\min_{x\in\mathbb{R}^n} \quad \frac{1}{2}\|Ax-b\|_2^2+\mu\|x\|_1,$$

其中$\mu>0$为已知的实数，x是待估计的参数. LASSO 问题通过惩罚参数的ℓ_1 范数来控制解的稀疏性

- 也可以考虑问题

$$\min_{x\in\mathbb{R}^n} \quad \frac{1}{2}\|Ax-b\|_2^2, \quad \text{s.t.} \quad \|x\|_1\le\sigma.$$

- 考虑到噪声ε的存在，还可以给定$\nu>0$考虑模型：

$$\min_{x\in\mathbb{R}^n} \quad \|x\|_1, \quad \text{s.t.} \quad \|Ax-b\|_2\le\nu.$$

- 后两个优化模型本质思想是相似的，即“在控制误差的条件下使得x的ℓ_1范数尽量小”. 但它们所属的优化问题种类不一样

LASSO 问题及其变形

- 如果ε不是高斯白噪声，则需要根据具体类型选择损失函数，比如

$$\min_{x\in\mathbb{R}^n} \quad \|Ax-b\|_2+\mu\|x\|_1,$$
$$\min_{x\in\mathbb{R}^n} \quad \|Ax-b\|_1+\mu\|x\|_1.$$

  上述两个模型和LASSO问题的差别在于对损失函数选择的范数不同，它们的性能可能很不一样.

- 当然损失函数项还有很多变化形式，如同时考虑ℓ_2范数和ℓ_1范数的组合，或选择Student-t分布等

LASSO 问题及其变形

- 当特征x本身不稀疏但在某种变换下是稀疏的，则需调整正则项

$$\min_{x\in\mathbb{R}^n} \quad \frac{1}{2}\|Ax-b\|_2^2+\mu\|Fx\|_1.$$

如果要求x相邻点之间的变化是稀疏的，取F为

$$F=\begin{bmatrix} 1 & -1 & & & \\ & 1 & -1 & & \\ & & \ddots & \ddots & \\ & & & 1 & -1 \end{bmatrix},$$

- 实际上$\|Fx\|_1$还可以与$\|x\|_1$结合起来，这表示同时对Fx和x提出稀疏性的要求. 例如融合LASSO 模型（fused-LASSO）

$$\min_{x\in\mathbb{R}^n} \quad \frac{1}{2}\|Ax-b\|_2^2+\mu_1\|x\|_1+\mu_2\sum_{i=2}^{n}|x_i-x_{i-1}|,$$

其中$\mu_2\sum_{i=2}^{n}|x_i-x_{i-1}|$ 用来控制相邻系数之间的平稳度.

逻辑回归: logistic regression

- 对于二分类问题，预测变量只有两个取值，即$-1, 1$.
- 给定特征a，逻辑回归假设这个样本属于类别1的概率

$$p(1|a;x) = P(t = 1 \mid a;x) = \theta(a^{\mathrm{T}}x),$$

其中Sigmoid 函数

$$\theta(z) = \frac{1}{1+\exp(-z)},$$

那么属于类别-1的概率

$$p(-1|a;x) = 1 - p(1 \mid a;x) = \theta(-a^{\mathrm{T}}x).$$

因此对于$b \in \{-1, 1\}$，上述概率可以简洁地写为

$$p(b \mid a;x) = \theta(b \cdot a^{\mathrm{T}}x).$$

逻辑回归

- 假设数据对$\{a_i, b_i\}, i = 1, 2, \cdots, m$之间独立同分布，则在给定$a_1, a_2, \cdots, a_m$情况下，$b_1, b_2, \cdots, b_m$的联合概率密度是

$$p(b_1, b_2, \cdots, b_m \mid a_1, a_2, \cdots, a_m; x) = \prod_{i=1}^{m} p(b_i \mid a_i; x)$$
$$= \frac{1}{\prod_{i=1}^{m}(1 + \exp(-b_i \cdot a_i^{\mathrm{T}} x))}.$$

- 最大似然估计是求解如下最优化问题：

$$\min_{x \in \mathbb{R}^n} \quad \sum_{i=1}^{m} \ln(1 + \exp(-b_i \cdot a_i^{\mathrm{T}} x)).$$

- 加上正则项，如Tikhonov和ℓ_1范数正则化模型：

$$\min_{x \in \mathbb{R}^n} \quad \sum_{i=1}^{m} \ln(1 + \exp(-b_i \cdot a_i^{\mathrm{T}} x)) + \lambda \|x\|_2^2$$

$$\min \quad \sum_{i=1}^{m} \ln(1 + \exp(-b_i \cdot a_i^{\mathrm{T}} x)) + \lambda \|x\|_1.$$

提纲

低秩矩阵恢复

- 某视频网站提供了约48万用户对1万7千多部电影的上亿条评级数据，希望对用户的电影评级进行预测，从而改进用户电影推荐系统，为每个用户更有针对性地推荐影片.

- 显然每一个用户不可能看过所有的电影，每一部电影也不可能收集到全部用户的评级. 电影评级由用户打分1 星到5 星表示，记为取值1~5 的整数. 我们将电影评级放在一个矩阵M中，矩阵M的每一行表示不同用户，每一列表示不同电影. 由于用户只对看过的电影给出自己的评价，矩阵M中很多元素是未知的

	电影1	电影2	电影3	电影4	…	电影n
用户1	4	?	?	3	…	?
用户2	?	2	4	?	…	?
用户3	3	?	?	?	…	?
用户4	2	?	5	?	…	?
⋮	⋮	⋮	⋮	⋮		⋮
用户m	?	3	?	4	…	?

低秩矩阵恢复问题的性质

该问题在推荐系统、图像处理等方面有着广泛的应用。

- 由于用户对电影的偏好可进行分类，按年龄可分为：年轻人，中年人，老年人；且电影也能分为不同的题材：战争片，悬疑片，言情片等。故这类问题隐含的假设为补全后的矩阵应为低秩的。即矩阵的行与列会有“合作”的特性，故该问题具有别名“collaborative filtering”。
- 除此之外，由于低秩矩阵可分解为两个低秩矩阵的乘积，所以低秩限制下的矩阵补全问题是比较实用的，这样利于储存且有更好的诠释性。
- 有些用户的打分可能不为自身真实情况，对评分矩阵有影响，所以原矩阵是可能有噪声的。

低秩矩阵恢复

由上述分析可以引出该问题：

- 令Ω是矩阵M中所有已知评级元素的下标的集合，则该问题可以初步描述为构造一个矩阵X，使得在给定位置的元素等于已知评级元素，即满足$X_{ij} = M_{ij},\ (i,j) \in \Omega$.

- 低秩矩阵恢复（low rank matrix completion）

$$\begin{aligned} \min_{X \in \mathbb{R}^{m \times n}} \quad & \operatorname{rank}(X), \\ \text{s.t.} \quad & X_{ij} = M_{ij},\ (i,j) \in \Omega. \end{aligned} \tag{2}$$

  $\operatorname{rank}(X)$正好是矩阵X所有非零奇异值的个数

- 矩阵X的核范数（nuclear norm）为矩阵所有奇异值的和，即：$\|X\|_* = \sum_i \sigma_i(X)$:

$$\begin{aligned} \min_{X \in \mathbb{R}^{m \times n}} \quad & \|X\|_*, \\ \text{s.t.} \quad & X_{ij} = M_{ij},\ (i,j) \in \Omega. \end{aligned} \tag{3}$$

低秩矩阵恢复

- 可以证明问题(3) 是一个凸优化问题，称之位问题(2)的凸松弛(convex relaxation). 并且在一定条件下它与问题(2) 等价.
- 也可以将问题(3) 转换为一个半定规划问题，但是目前半定规划算法所能有效求解的问题规模限制了这种技术的实际应用.
- 考虑到观测可能出现误差，对于给定的参数$\mu > 0$，给出该问题的二次罚函数形式：

$$\min_{X \in \mathbb{R}^{m \times n}} \quad \mu \|X\|_* + \frac{1}{2} \sum_{(i,j) \in \Omega} (X_{ij} - M_{ij})^2. \tag{4}$$

提纲

子空间恢复

在经典的PCA中，数据被分解为几个主成分，这些主成分捕获了数据中的主要变异性。然而，当数据中存在离群点或异常值时，PCA的性能可能会大大下降，因为它试图捕获所有数据点的变异性，包括异常值。
鲁棒主成分分析(RPCA)解决了这个问题。它将数据矩阵分解为两部分：一个低秩矩阵和一个稀疏矩阵。低秩矩阵捕获数据的主要结构，而稀疏矩阵则包含异常值或离群点。通过这种方式，RPCA能够在保持数据主要结构的同时，有效地处理异常值。
其数学模型如下：

$$\begin{aligned} \min_{X,S} \quad & \|X\|_* + \mu\|S\|_1, \\ \text{s.t.} \quad & X + S = M, \end{aligned} \tag{5}$$

其中$\|\cdot\|_1$与$\|\cdot\|_*$分别表示矩阵ℓ_1范数与核范数.

例：矩阵分解

通过RPCA，将视频图片分为背景（低秩）和前景（稀疏）两个部分。

提纲

神经网络

深度神经网络(DNN) 是一种前馈结构，通过堆叠多个神经元层来构建，它通过已有的信息或者知识来对未知事物进行预测。每个神经元都会应用一个变换，然后再通过一个非线性激活函数进行激活。对于分类任务，最后一层通常后接一个softmax函数以产生类概率。然后，使用交叉熵损失来测量DNN的性能。

- 在神经网络中，已知的信息通常用数据集来表示. 数据集一般分为训练集和测试集：训练集是用来训练神经网络，从而使得神经网络能够掌握训练集上的信息;测试集是用来测试训练完的神经网络的预测准确性.
- 一个常见的任务是分类问题. 假设我们有一个猫和狗的图片集，将其划分成训练集和测试集（保证集合中猫和狗图片要有一定的比例）. 神经网络是想逼近一个从图片到$\{0,1\}$ 的函数，这里0 表示猫，1 表示狗.
- 因为神经网络本身的结构和大量的训练集信息，训练得到的函数与真实结果具有非常高的吻合性.

多层感知机模型介绍

多层感知机是深度神经网络的一种简单形式，主要区别在于深度神经网络通常具有更复杂的结构。
给定训练集$D = \{\{a_1, y_1\}, \{a_2, y_2\}, \cdots, \{a_m, y_m\}\}$.

- 假设数据$a_i \in \mathbb{R}^p$, $y_i \in \mathbb{R}^q$. $a_{i1} = 1$. 图4 给出了一种由p个输入单元和q个输出单元构成的$(L+2)$层感知机，其含有一个输入层，一个输出层，和L个隐藏层. 该感知机的第l个隐藏层共有$m^{(l)}$个神经元，为了方便用$l = 0$表示输入层，$l = L+1$ 表示输出层，并定义$m^{(0)} = p$和$m^{(L+1)} = q$.
  设$y^{(l)} \in \mathbb{R}^{m^{(l)}}$为第$l$层的所有神经元，令$y_1^{(l)} = 1, 0 \le l \le L$,
- 其余的元素则是通过上一层的神经元的值进行加权求和得到. 令参数$W = (W^{(1)}, W^{(2)}, \cdots, W^{(L+1)})$ 表示网络中所有层之间的权重，其中$W_{i,k}^{(l)}$是第$(l-1)$隐藏层的第k个单元连接到第l隐藏层的第i个单元对应的权重，$b^{(l)}$表示偏差，则在第l隐藏层中，第i个单元计算输出信息$y_i^{(l)}$为

$$y_i^{(l)} = \phi(z_i^{(l)}), \quad z_i^{(l)} = \sum_{k=1}^{m^{(l-1)}} W_{i,k}^{(l)} y_k^{(l-1)} + b^{(l)}. \tag{6}$$

这里函数$\phi(\cdot)$称为激活函数.

多层感知机

上述过程可用下图表示:

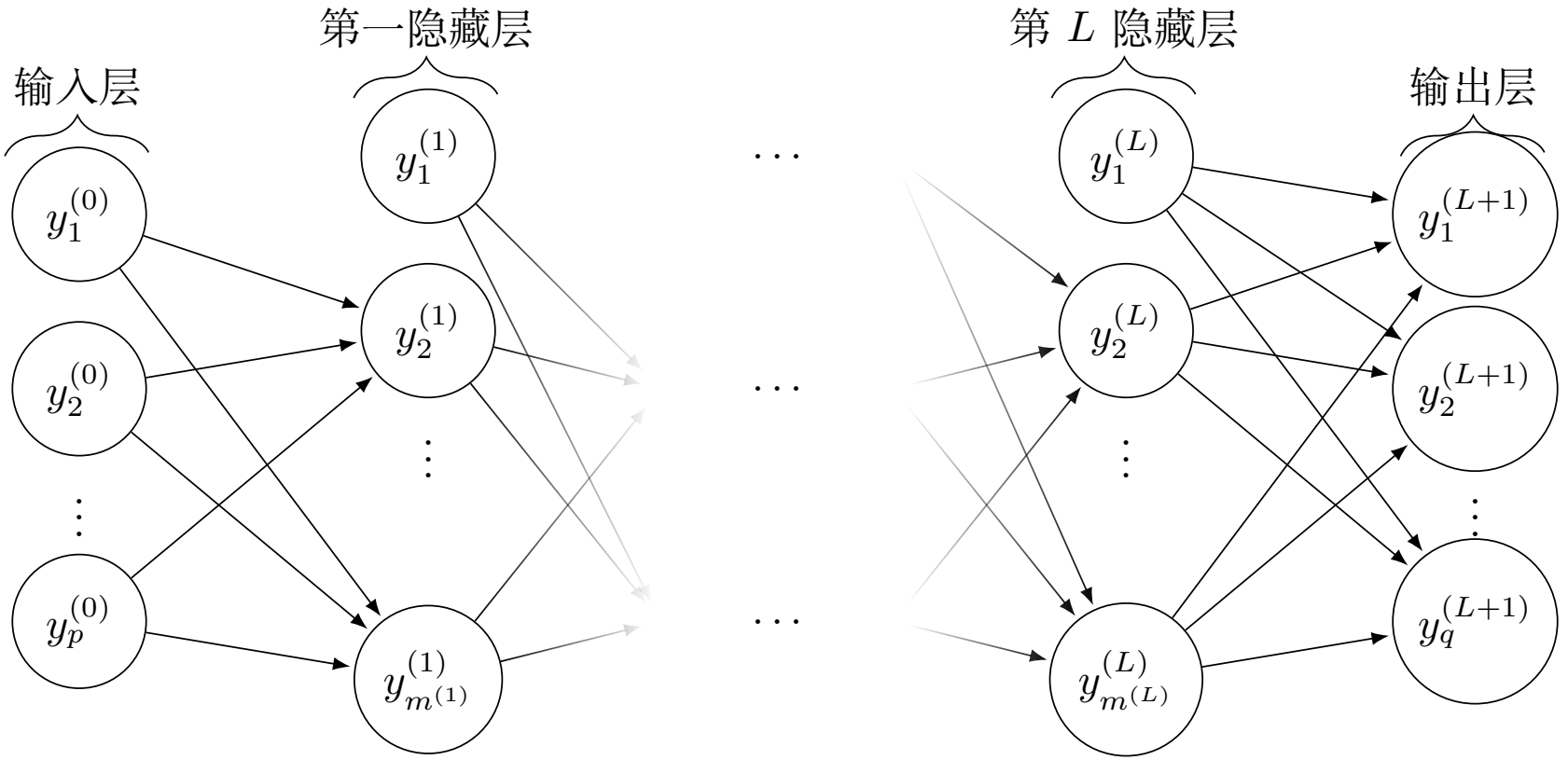


Figure: 带p个输入单元和q个输出单位的$(L+2)$层感知机的网络图，第l个隐藏层包含$m^{(l)}$个神经元.

神经网络的损失函数如下：

- 第l层，给定前一层的激活值$y^{(l-1)}$：

$$y^{(l)} = \phi(W^{(l)}, b^{(l)}; y^{(l-1)}) = \phi(W^{(l)} y^{(l-1)} + b^{(l)}) \tag{7}$$

- ReLU作为激活函数，得到第l层的激活值

$$y^{(l)} = \text{ReLU}(W^{(l)} y^{(l-1)} + b^{(l)}) \tag{8}$$

- 对于最后一层，我们使用softmax函数得到类概率：

$$\text{softmax}(y_i^{(L+1)}) = \frac{e^{y_i^{(L+1)}}}{\sum_{j=1}^{q} e^{y_j^{(L+1)}}} \tag{9}$$

其中，q是类别的数量。

- 测量网络预测与实际标签之间差异的损失函数是交叉熵损失：

$$\text{CE}(y, \hat{y}) = -\frac{1}{m} \sum_{i=1}^{m} \sum_{j=1}^{q} y_{i,j} \log(\text{softmax}(y_{i,j}^{(L)})) \tag{10}$$

神经网络模型具有以下两个特征：（1）参数量非常大；（2）训练样本数量非常大。表1 和表2 给出了一些经典的数据集和模型大小。模型规模和数据集大小制约了训练效率，大模型的训练有可能花费数天甚至数月时间。优化器的选择直接影响模型的训练效率。本课程中，我们将探讨常用的模型训练优化器。

Table: 大数据集数据量

数据集	数据集大小
Cifar10, Cifar100	60000 张图像
ImageNet	约1400万张图像
MS Coco	约33万张图像
GPT-3	45TB

Table: 大模型参数量

模型名称	参数量
VGG16	1.4亿
ResNet50	2500万
DenseNet-201	2000万
GPT-3	1750亿
GPT-4	1.8万亿

网络上爆料的GPT-4训练成本：一次的训练的成本为6300万美元，OpenAI训练GPT-4的FLOPS约为2.15e25，在大约25000个A100上训练了90到100天，利用率在32%到36%之间。

最优化算法：凸集

陈士祥

中国科学技术大学

致谢：本教案部分参考北京大学文再文教授《最优化与建模》

提纲

回顾：向量范数的定义

定义

令记号$\|\cdot\|:\mathbb{R}^n\to\mathbb{R}^+$是一种非负函数，如果它满足：

- 正定性：对于$\forall v\in\mathbb{R}^n$，有$\|v\|\geqslant 0$，且$\|v\|=0\Leftrightarrow v=0_{n\times 1}$；
- 齐次性：对于$\forall v\in\mathbb{R}^n$和$\alpha\in\mathbb{R}$，有$\|\alpha v\|=|\alpha|\,\|v\|$；
- 三角不等式：对于$\forall v,w\in\mathbb{R}^n$，均成立$\|v+w\|\leqslant\|v\|+\|w\|$.

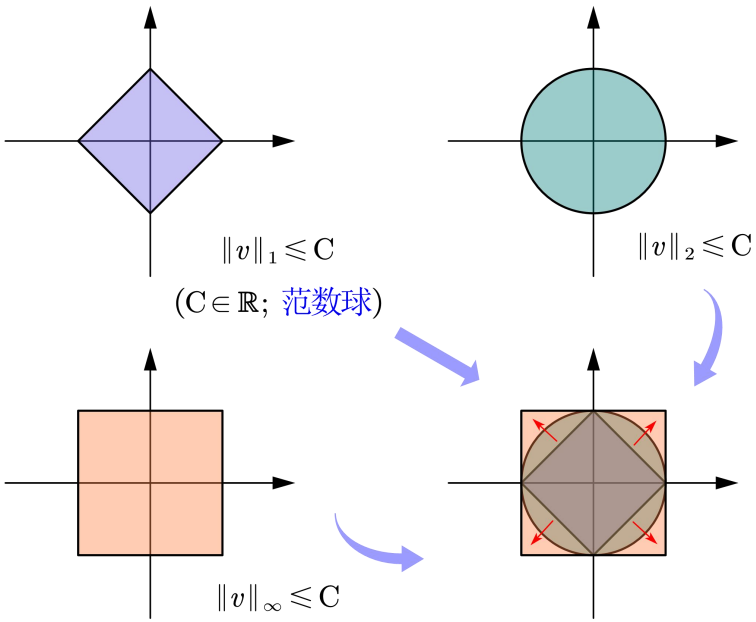
则称$\|\cdot\|$是定义在向量空间$\mathbb{R}^n$上的向量范数.

最常用的向量范数即我们熟知的ℓ_p范数(其中$p\geqslant 1$):

$$\|v\|_p=\left(\sum_{i=1}^n|v_i|^p\right)^{\frac{1}{p}};\quad \|v\|_\infty=\max_{1\leqslant j\leqslant n}|v_j|.$$

柯西不等式：设$a,b\in\mathbb{R}^n$，则$|a^{\mathrm{T}}b|\leqslant\|a\|_2\,\|b\|_2$，且等号成立的条件是$a$与$b$线性相关.

向量范数的定义



容易看出, $p = \infty$时, 有关"最大值"的定义要求向量的分量是有限的. 在一般化的空间中, 这一要求很可能不成立, 此时我们只需将"最大值"更换成"上确界"即可.

向量范数度量的是v与零点之间的距离. 在实际应用时, 我们通常使用$p = 1, 2, \infty$的情形, 即分别使用$\|v\|_1, \|v\|_2, \|v\|_\infty$度量$v$在不同意义下的距离, 这是因为它们具有鲜明的度量特征.

左图是它们各自的范数球实例, 请想一想不同范数所度量的距离分别具有怎样的特征? 这些特征分别适用于度量什么情形?

矩阵范数

矩阵范数可以由向量范数的定义推广得到。常见的矩阵范数有：

- $\|A\|_1 = \sum_{i,j} |A_{ij}|$
- $\|A\|_F = \sqrt{\sum_{i,j} A_{ij}^2} = \sqrt{\mathbf{Tr}\left(AA^{\mathrm{T}}\right)}$
- 算子范数是一类特殊的矩阵范数，它由向量范数诱导得到：

$$\|A\|_{(m,n)} = \max_{x\in\mathbb{R}^n, \|x\|_{(n)}=1} \|Ax\|_{(m)} .$$

  - $p=1$时，$\|A\|_{p=1} = \max\limits_{\|x\|_1=1} \|Ax\|_1 = \max\limits_{1\leqslant j\leqslant n} \sum_{i=1}^m |a_{ij}|$.
  - $p=2$时，$\|A\|_{p=2} = \max\limits_{\|x\|_2=1} \|Ax\|_2 = \sqrt{\lambda_{\max}(A^{\mathrm{T}}A)}$，又称为$A$的谱范数.
  - $p=\infty$时，$\|A\|_{p=\infty} = \max\limits_{\|x\|_\infty=1} \|Ax\|_\infty = \max\limits_{1\leqslant i\leqslant m} \sum_{j=1}^n |a_{ij}|$.

矩阵范数

- 核范数定义为

$$\|A\|_* = \sum_{i=1}^{r} \sigma_i,$$

其中$\sigma_i (i = 1, \cdots, r)$为$A$的所有非零奇异值, $r = \mathbf{rank}\,(A)$.

- 矩阵A, B的内积定义为

$$\langle A, B \rangle = \mathrm{Tr}\left(AB^{\mathrm{T}}\right) = \sum_{i=1}^{m}\sum_{j=1}^{n} a_{ij}b_{ij}.$$

- 柯西不等式: 设$A, B \in \mathbb{R}^{m\times n}$, 则

$$|\langle A, B \rangle| \leqslant \|A\|_F \|B\|_F,$$

等号成立当且仅当A和B线性相关.

提纲

凸集的几何定义

在$\mathbb{R}^n$空间中，经过不同的两点x_1, x_2可以确定一条直线，其方程为

$$y = \theta x_1 + (1-\theta)x_2, \theta \in \mathbb{R}.$$

特别，当$0 \leqslant \theta \leqslant 1$时，直线退化为以$x_1, x_2$为端点的线段.

定义

仿射集　如果过集合$\mathcal{C}$中的任意两点的直线都在$\mathcal{C}$内，则称$\mathcal{C}$为仿射集，即

$$x_1, x_2 \in \mathcal{C} \Rightarrow \theta x_1 + (1-\theta)x_2 \in \mathcal{C}, \forall \theta \in \mathbb{R}.$$

例　线性方程组$Ax = b$的解集$\mathcal{X}$是仿射集，因为$\forall x_1, x_2 \in \mathcal{X}(x_1 \neq x_2)$均满足$\theta Ax_1 + (1-\theta)Ax_2 = b$.
反之，任何仿射集均可表示为某一线性方程组的解集.

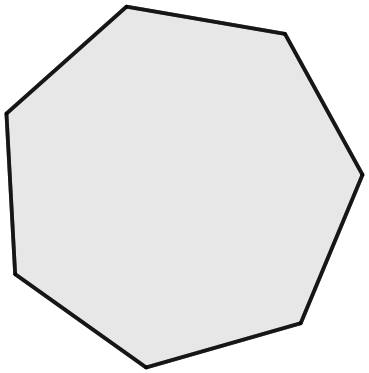
凸集的几何定义

定义

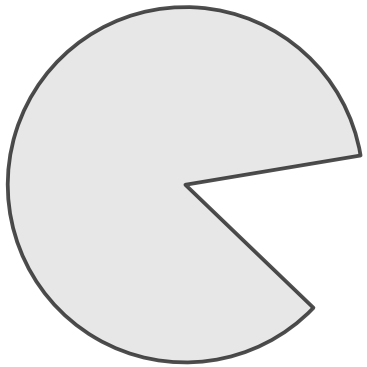
凸集　如果连接集合$\mathcal{C}$中的任意两点的线段都在$\mathcal{C}$内，则称$\mathcal{C}$为凸集，即

$$x_1, x_2 \in \mathcal{C} \Rightarrow \theta x_1 + (1-\theta)x_2 \in \mathcal{C}, \forall 0 \leqslant \theta \leqslant 1.$$

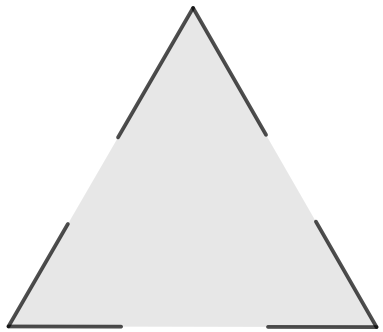
仿射集当然都是凸集.



(a)



(b)



(c)

例　在左图中我们列出了一些凸集、非凸集的例子. 其中(a)为凸集, (b)和(c)均为非凸集.

凸集的性质

定理

- 若$\mathcal{S}$是凸集, 则$k\mathcal{S} = \{ks | k \in \mathbb{R}, s \in \mathcal{S}\}$是凸集.
- 若$\mathcal{S}$和$\mathcal{T}$均是凸集, 则$\mathcal{S} + \mathcal{T} = \{s + t | s \in \mathcal{S}, t \in \mathcal{T}\}$是凸集.
- 若$\mathcal{S}$和$\mathcal{T}$均是凸集, 则$\mathcal{S} \cap \mathcal{T}$是凸集.
- 凸集的内部和闭包都是凸集.

上述定理前2点在凸集定义前提下是显然的. 我们简述第3点为何成立.

证明: 设$x, y \in \mathcal{S} \cap \mathcal{T}$且$\theta \in [0, 1]$. 由于$\mathcal{S}$和$\mathcal{T}$均为凸集, 则

$$\theta x + (1 - \theta)y \in \mathcal{S} \cap \mathcal{T},$$

这证明$\mathcal{S} \cap \mathcal{T}$是凸集.

实际上, 任意多凸集的交都是凸集. 该结论在证明复杂集合是凸集时非常有用, 因为我们可以考虑将其视为任意个凸集的交.

凸组合和凸包

从凸集中可以引出凸组合和凸包的概念.

定义

凸组合　形如

$$x = \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_k x_k,$$

$$\theta_1 + \cdots + \theta_k = 1, \theta_i \geqslant 0, i = 1, \cdots, k.$$

的点称为$x_1, \cdots, x_k$的凸组合.

定义

凸包　集合$\mathcal{S}$的所有点的凸组合构成的点集为$\mathcal{S}$的凸包, 记为$\mathbf{conv}\mathcal{S}$.

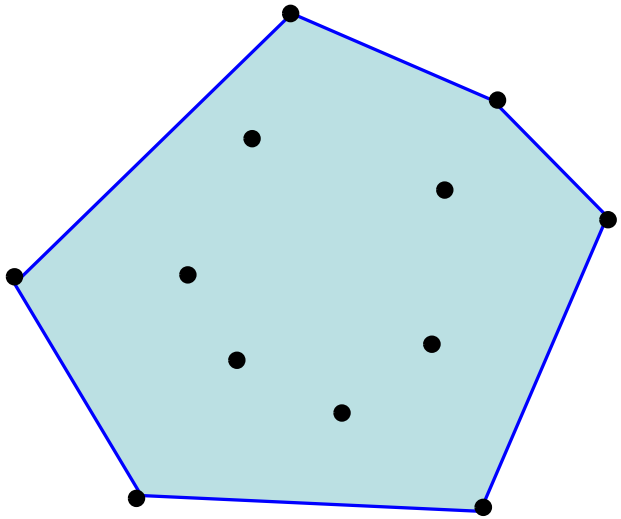
定理

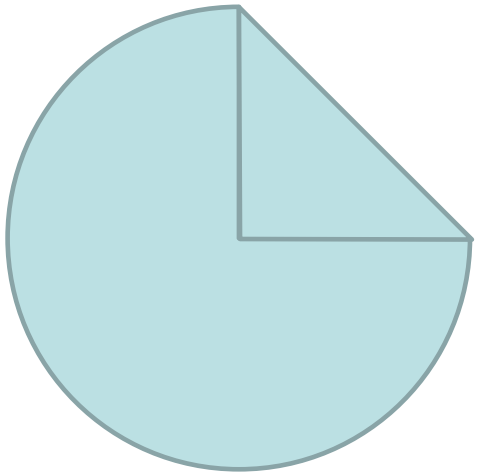
凸集和凸包的关系　若$\mathbf{conv}\mathcal{S} \subseteq \mathcal{S}$, 则$\mathcal{S}$是凸集; 反之亦然.

上述定理并不显然, 请尝试证明.　(提示: 用数学归纳法)

凸包的例子

例　在下图中我们列出了一些离散点集和连续点集的凸包. 其中, 左子图为离散点集的凸包, 右子图为扇形连续点集的凸包.





$\mathbf{conv}\mathcal{S}$是包含$\mathcal{S}$的最小凸集

定理

$\mathbf{conv}\mathcal{S}$是包含$\mathcal{S}$的最小凸集.

证明　由凸包的定义可知, $\mathcal{S} \in \mathbf{conv}\mathcal{S}$, 并且$\mathbf{conv}\mathcal{S}$是凸集.
若再设$\mathcal{X}$是另一凸集且满足$\mathcal{S} \subseteq \mathcal{X} \subseteq \mathbf{conv}\mathcal{S}$, 下面我们需要证明只可能是$\mathcal{X} = \mathbf{conv}\mathcal{S}$. 为证明此结论, 我们先证明一个重要的命题, 从而直接导出本定理的成立.

定理

对于任意向量集$\mathcal{S}$, $\mathbf{conv}\mathcal{S}$是包含$\mathcal{S}$的一切凸集的交集.

证明　令$\mathcal{X}$表示包含$\mathcal{S}$的所有凸集的交集. 我们之前证明, 凸集的交是凸集, 因此$\mathcal{X}$是凸集. 因为$\mathbf{conv}\mathcal{S}$是一个凸集且包含$\mathcal{S}$, 则$\mathcal{X} \subseteq \mathbf{conv}\mathcal{S}$.
另一方面, $\mathcal{S} \subseteq \mathcal{X}$, 因此$\mathbf{conv}\mathcal{S} \subseteq \mathbf{conv}\mathcal{X}$.
再由凸集和凸包的关系得到$\mathbf{conv}\mathcal{X} = \mathcal{X}$, 得到$\mathbf{conv}\mathcal{S} \subseteq \mathcal{X}$.
综上有$\mathcal{X} = \mathbf{conv}\mathcal{S}$.

仿射包

仿射集和凸集的定义很像, 除了θ的范围有所不同. 受此启发, 从凸组合和凸包的定义中可以自然引出仿射组合和仿射包的概念.

定义

仿射组合　形如

$$x = \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_k x_k,$$

$$\theta_1 + \cdots + \theta_k = 1, \theta_i \in \mathbb{R}, i = 1, \cdots, k.$$

的点称为$x_1, \cdots, x_k$的仿射组合.

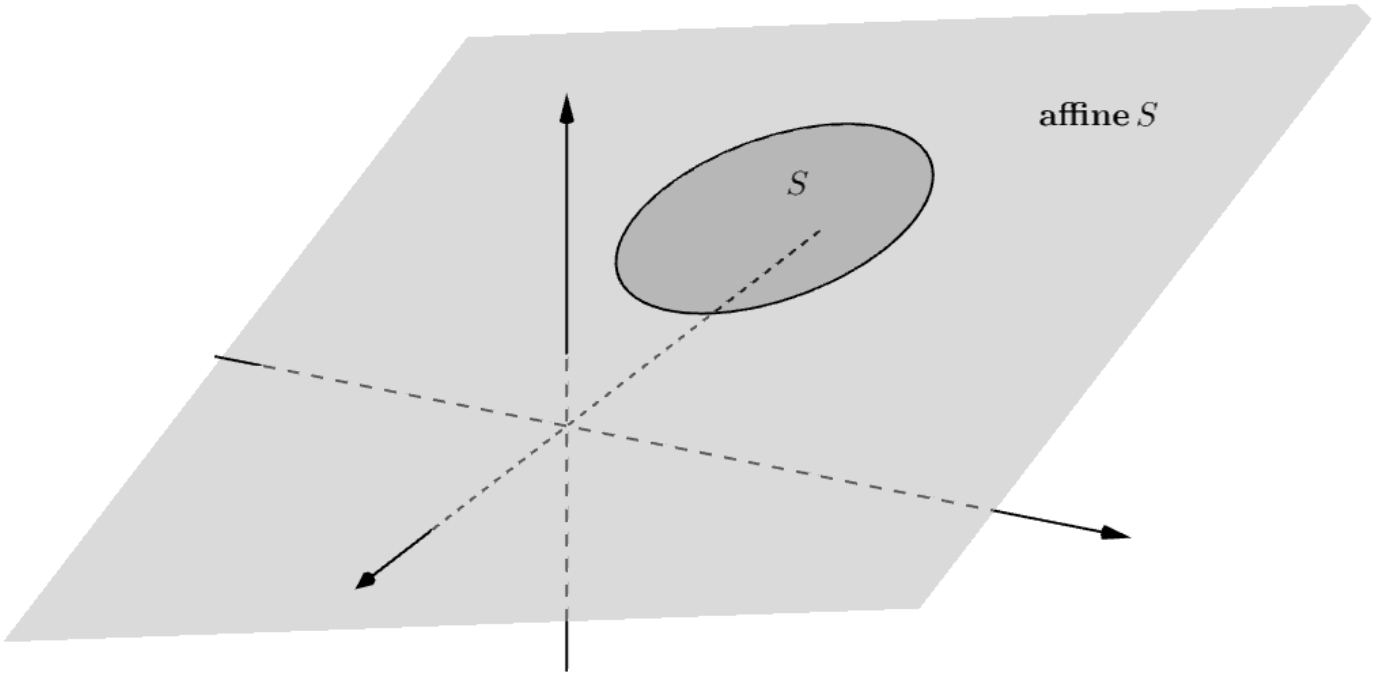
定义

仿射包　集合$\mathcal{S}$的所有点的仿射组合构成的点集为$\mathcal{S}$的仿射包, 记为**affine**$\mathcal{S}$.

affine$\mathcal{S}$是包含$\mathcal{S}$的最小仿射集.

$\mathbb{R}^3$中仿射包的例子

例　下图为$\mathbb{R}^3$中圆盘S的仿射包示意图, 可见仿射包直接将原集合拓展为了其所在的全平面.



锥组合和凸锥

定义

锥(Cone)　我们称集合$C \subset \mathbb{R}^n$是一个锥，如果对于任意$x \in C$，有$tx \in C, \forall t \geq 0$. 如果$C$是凸集，那么称其是凸锥。

相比于凸组合和仿射组合, 锥组合不要求系数的和为1, 因此一般而言锥组合都是开放的.

定义

锥组合　形如

$$x = \theta_1 x_1 + \cdots + \theta_k x_k, \theta_i > 0 (i = 1, \cdots, k).$$

的点称为$x_1, \cdots, x_k$的锥组合.

凸锥也有下面等价的定义。

定义

凸锥　若集合$\mathcal{S}$中任意点的锥组合都在$\mathcal{S}$中, 则称$\mathcal{S}$为凸锥.

凸锥的例

例　下图显示了$\mathbb{R}^2$中两点x_1, x_2的凸锥. 可见$\mathbb{R}^2$中若两点不与原点O共线, 则其形成的凸锥为一个半径无穷的圆的扇形部分.

提纲

超平面和半空间

定义

超平面　任取非零向量$a \in \mathbb{R}^n$, 形如

$$\{x | a^{\mathrm{T}} x = b\}$$

的集合称为超平面.

定义

半空间　任取非零向量$a \in \mathbb{R}^n$, 形如

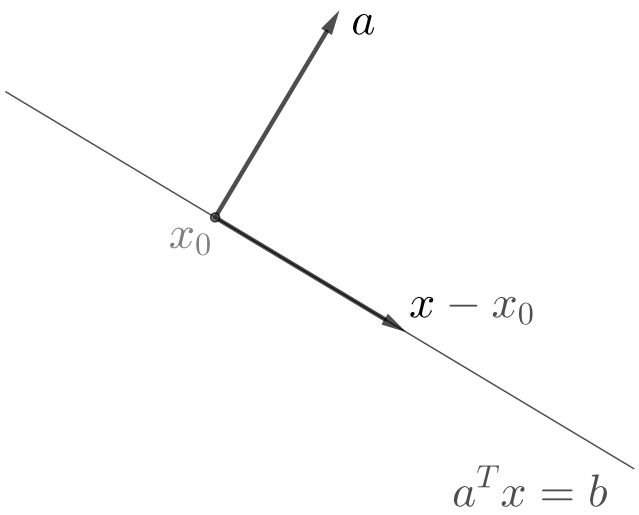
$$\{x | a^{\mathrm{T}} x \leqslant b\}$$

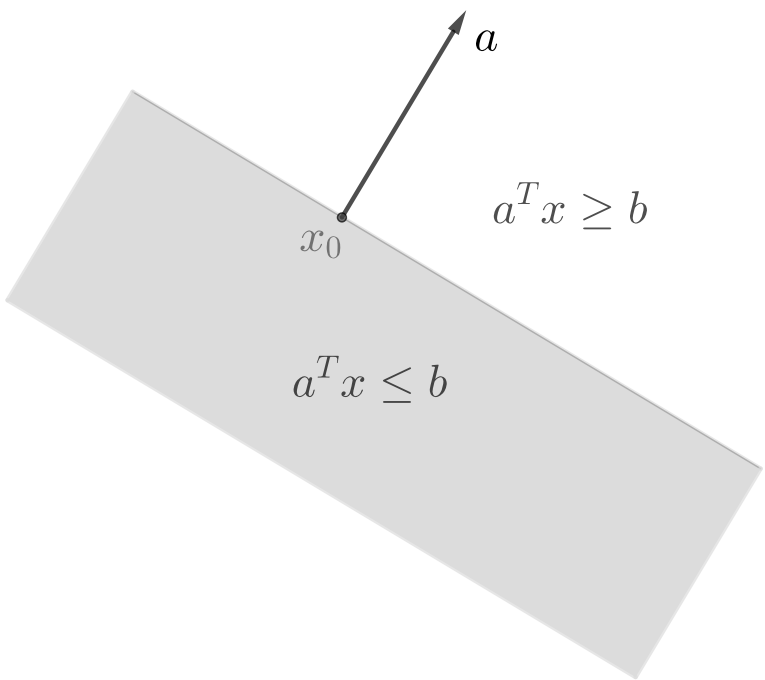
的集合称为半空间.

超平面是仿射集和凸集, 半空间是凸集但不是仿射集.

超平面和半空间的例

例　下图是$\mathbb{R}^2$中超平面和半空间的例子. 其中, 左子图为超平面, 其为一条直线; 右子图为半空间.





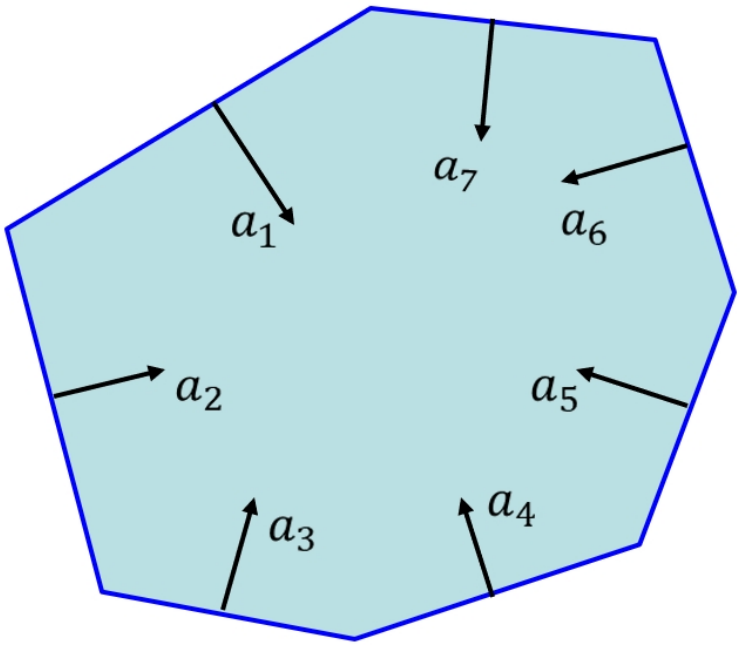
多面体

我们把满足线性等式和不等式组的点的集合称为多面体, 即

$$\{x|Ax \leqslant b, Cx = d\},$$

其中$A \in \mathbb{R}^{m\times n}$, $C \in \mathbb{R}^{p\times n}$, $x \leqslant y$表示向量x的每个分量都小于等于y的对应分量.

多面体是有限个半空间和超平面的交, 因此由凸集的性质可知, 其为凸集.



范数球、椭球

如下定义的球和椭球也是常见的凸集.

定义

球　设空间中到某一定点x_c(称为中心)的距离小于等于定值r(称为半径)的点的集合为(范数)球, 即

$$\mathrm{B}(x_c, r) = \{x \mid \|x - x_c\| \leqslant r\} = \{x_c + ru \mid \|u\| \leqslant 1\}.$$

一般而言, 我们使用$\|\cdot\|_2$度量距离, 即使用2-范数球.

定义

椭球　设形如

$$\left\{x \mid (x - x_c)^{\mathrm{T}} P^{-1} (x - x_c) \leqslant 1\right\} = \{x_c + Au \mid \|u\|_2 \leqslant 1\}$$

的集合为椭球, 其中x_c为椭球中心, P对称正定, 且A非奇异.

反例：球面$\{x \mid \|x\|_2 = 1\}$是非凸集。

范数锥

球和椭球的范围取决于x的范围，而锥的范围则同时取决于x和控制径t的范围.

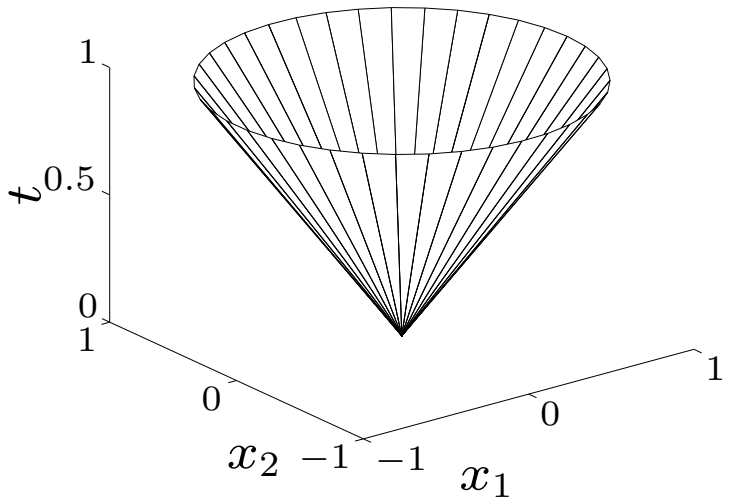
定义

范数锥　形如

$$\{(x,t) \in \mathbb{R}^{n+1} \mid \|x\| \leqslant t\}$$

的集合为范数锥.

锥是凸集. 同时，使用$\|\cdot\|_2$度量距离的锥为二次锥，也称冰淇淋锥。



特殊矩阵集合和(半)正定锥

我们介绍3类矩阵的集合.

定义

对称矩阵集合　记$\mathcal{S}^n$为$n \times n$对称矩阵的集合, 即

$$\mathcal{S}^n = \left\{X \in \mathbb{R}^{n\times n} | X^{\mathrm{T}} = X\right\}.$$

定义

半正定矩阵集合　记$\mathcal{S}_+^n$为$n \times n$半正定矩阵的集合, 即

$$\mathcal{S}_+^n = \{X \in \mathcal{S}^n | X \succeq 0\}.$$

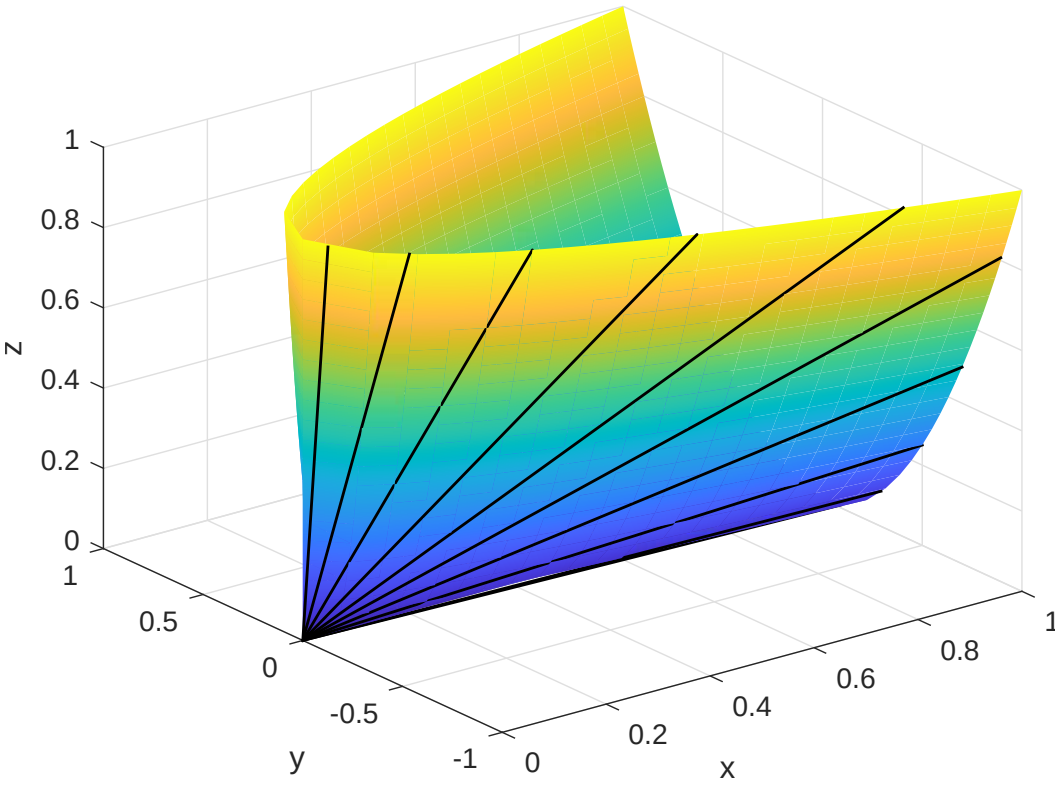
定义

正定矩阵集合　记$\mathcal{S}_{++}^n$为$n \times n$正定矩阵的集合, 即

$$\mathcal{S}_{++}^n = \{X \in \mathcal{S}^n | X \succ 0\}.$$

半正定锥的例子

我们一般称$\mathcal{S}_+^n$为半正定锥. 下图是二维半正定锥边界的几何形状.



实际上可以由半正定矩阵的性质得到:

对于矩阵$\begin{pmatrix} x & y \\ y & z \end{pmatrix}$, 其特征值应全部大于等于0, 由此可推出

$$x \geqslant 0, z \geqslant 0, xz \geqslant y^2.$$

由此可知, 二维半正定锥的实际范围是

$$\left\{(x, y, z) \,|\, x \geqslant 0, z \geqslant 0, xz \geqslant y^2\right\}.$$

提纲

仿射变换的保凸性

仿射变换(缩放、平移、投影等)也是保凸的.

定理

仿射变换的保凸性 设$f:\mathbb{R}^n \to \mathbb{R}^m$是仿射变换, 即$f(x)=Ax+b$, $A\in\mathbb{R}^{m\times n}, b\in\mathbb{R}^m$, 则

- 凸集在f下的像是凸集:

$$S\subseteq\mathbb{R}^n\text{是凸集}\Rightarrow f(S)=\{f(x)|x\in S\}\text{是凸集}.$$

- 凸集在f下的原像是凸集:

$$C\subseteq\mathbb{R}^m\text{是凸集}\Rightarrow f^{-1}(C)=\{x|f(x)\in C\}\text{是凸集}.$$

仿射变换的保凸性

例 椭球是凸集。

例 线性矩阵不等式的解集

$$\{x|x_1A_1+\cdots+x_mA_m\preceq B\}\,(A_i,i=1,\cdots,m,B\in\mathcal{S}^p)$$

是凸集. 这由仿射变换$f(x)=B-(x_1A_1+\cdots+x_mA_m)$在半正定锥的原像可以直接得到.

例 双曲锥

$$\left\{x|x^{\mathrm{T}}Px\leqslant\left(c^{\mathrm{T}}x\right)^2,c^{\mathrm{T}}x\geqslant 0,P\in\mathcal{S}_+^n\right\}$$

是凸集.

证明: 双曲锥可以转化为二阶锥

$$\left\{x|\,\|Ax\|_2\leqslant c^{\mathrm{T}}x,c^{\mathrm{T}}x\geqslant 0,A^{\mathrm{T}}A=P\right\},$$

这可以考虑$f(x)=(Ax,c^{\top}x)$在二次锥$\left\{(x,t)\,|x^{\top}x\leqslant t^2,t\geqslant 0\right\}$的原像得到, 因此二阶锥、二次锥均为凸集.

透视变换和分式线性变换的保凸性

- 透视变换 $P:\mathbb{R}^{n+1}\to\mathbb{R}$:

$$P(x,t)=x/t,\quad \mathrm{dom}P=\{(x,t)\mid t>0\}.$$

透视变换下凸集的像和原像是凸集。

- 分式线性变换 $f:\mathbb{R}^n\to\mathbb{R}^m$:

$$f(x)=\frac{Ax+b}{c^{\mathrm{T}}x+d},\quad \mathbf{dom}f=\{x\mid c^{\mathrm{T}}x+d>0\}$$

分式线性变换下凸集的像和原像是凸集。

提纲

适当锥

一个凸锥$K \subseteq \mathbb{R}^n$是适当锥(proper cone), 当其还满足

- K是闭集;
- K是实心的, 即$\mathbf{int}K \neq \emptyset$;
- K是尖的, 即内部不含有直线: 若$x \in K, -x \in K$, 则一定有$x = 0$.

例　非负卦限$K = \mathbb{R}^n_+ = \{x \in \mathbb{R}^n | x_i \geqslant 0, i = 1, \cdots, n\}$是适当锥.

例　半正定锥$K = \mathcal{S}^n_+$是适当锥.

例　$[0, 1]$上的有限非负多项式

$$K = \left\{x \in \mathbb{R}^n | x_1 + x_2 t + \cdots + x_n t^{n-1} \geqslant 0, \forall t \in [0, 1]\right\}$$

是适当锥.

广义不等式

广义不等式是一种偏序(不必要保证所有对象都具有可比较性)关系，可以使用适当锥诱导.

定义

广义不等式 对于适当锥K, 定义偏序广义不等式为

$$x \preceq_K y \iff y - x \in K,$$

严格偏序广义不等式为

$$x \prec_K y \iff y - x \in \mathbf{int}K.$$

例 坐标分量不等式$(K = \mathbb{R}^n_+)$

$$x \preceq_{\mathbb{R}^n_+} y \iff y_i \geqslant x_i.$$

例 矩阵不等式$(K = \mathcal{S}^n_+)$

$$X \preceq_{\mathcal{S}^n_+} Y \iff Y - X\text{半正定}.$$

广义不等式的性质

$\preceq_K$的诸多性质在$\mathbb{R}$中与$\leqslant$类似.

定理

广义不等式的性质　记$\preceq_K$是定义于适当锥K上的广义不等式, 则

- 自反性: $x \preceq_K x$;
- 反对称性: 若$x \preceq_K y$且$y \preceq_K x$, 则$x = y$;
- 传递性: 若$x \preceq_K y$且$y \preceq_K z$, 则$x \preceq_K z$;
- 可加性: 若$x \preceq_K y$且$u \preceq_K v$, 则$x + u \preceq_K y + v$;
- 非负缩放: 若$x \preceq_K y$且$\alpha \geqslant 0$, 则$\alpha x \preceq_K \alpha y$.

利用偏序关系和广义不等式的定义可以轻松证明上述性质.

对偶锥

设K是一个锥.

定义

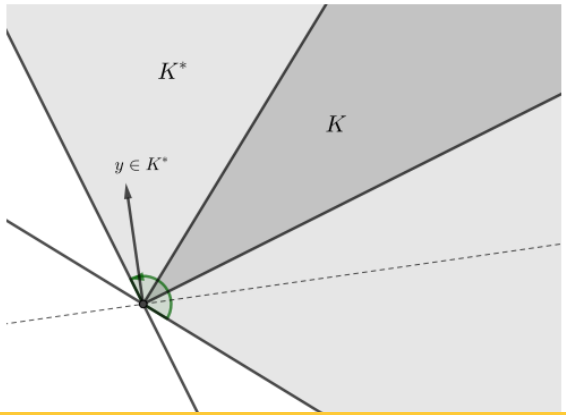
对偶锥 令锥K为全空间Ω的子集, 则K的对偶锥为

$$K^* = \{y \in \Omega | \langle x, y\rangle \geqslant 0, \forall x \in K\}.$$

对偶锥是相对于锥K定义的, 因此我们知道锥的同时也可以求出对偶锥.

例 我们在下图中给出了一个$\mathbb{R}^2$平面上的一个例子. 图中深色区域表示锥K, 根据对偶锥的定义, K^*中的向量和K中所有向量夹角均为锐角或直角. 因此, 对偶锥K^*为图中的浅色区域.

注意, 在这个例子中, K也为K^*的一部分.



对偶锥

我们将对偶锥为自身的锥称为自对偶锥

例　$K=\mathbb{R}^n_+$的对偶锥是它本身, 因此是自对偶锥.

例(请自证)　$K=\mathcal{S}^n_+$的对偶锥是它本身, 因此是自对偶锥.

例(请自证)　锥$K=\left\{(x,t)\,|\,\|x\|_p\leqslant t, t>0, p\geqslant 1\right\}$的对偶锥是

$$K^*=\left\{(x,t)\,|\,\|x\|_q\leqslant t, t>0, q\geqslant 1, (p,q)\text{共轭}\right\}.$$

例　上例中二次锥的对偶锥是它本身, 因此是自对偶锥.

对偶锥的性质

下面我们简单列举对偶锥满足的性质, 这是很重要的.

定理

对偶锥的性质 设K是一锥, K^*是其对偶锥, 则满足

- K^*是锥(哪怕K不是锥也成立);
- K^*始终是闭集, 且是凸集;
- 若$\mathbf{int}K \neq \varnothing$, 则$K^*$是尖的, 即内部不含有直线;
- 若K的闭包是尖的, 则$\mathring{K}^* \neq \varnothing$;
- 若K是适当锥, 则K^*是适当锥;
- (二次对偶)K^{**}是K的凸包的闭包. 特别, 若K是凸且闭的, 则$K^{**} = K$.

对偶锥诱导的广义不等式

既然适当锥的对偶锥仍是适当锥, 则可以用适当锥K的对偶锥K^*也可以诱导广义不等式. 我们在下文简称其为"对偶广义不等式".

定义

对偶广义不等式　适当锥的对偶锥K^*可定义广义不等式

$$x \preceq_{K^*} y \Longleftrightarrow y - x \in K^*,$$

其满足性质:

- $x \preceq_K y \Longleftrightarrow \lambda^{\mathrm{T}} x \leqslant \lambda^{\mathrm{T}} y, \forall \lambda \succeq_{K^*} 0;$
- $y \succeq_{K^*} 0 \Longleftrightarrow y^{\mathrm{T}} x \geqslant 0, \forall x \succeq_K 0.$

使用对偶广义不等式的好处是, 对偶锥始终是闭且凸的, 并可将一个偏序问题转换为满足一个偏序条件的全序问题.

提纲

分离超平面定理

超平面是空间中一类特殊的凸集(仿射集), 可以证明$\mathbb{R}^n$空间中的超平面恰好是$n-1$维的. 我们可以用超平面分离不相交的凸集.

定理

分离超平面定理　如果$\mathcal{C}$和$\mathcal{D}$是不相交的凸集, 则存在非零向量a和常数b, 使得

$$a^{\mathrm{T}}x \leqslant b, \forall x \in \mathcal{C},$$

且

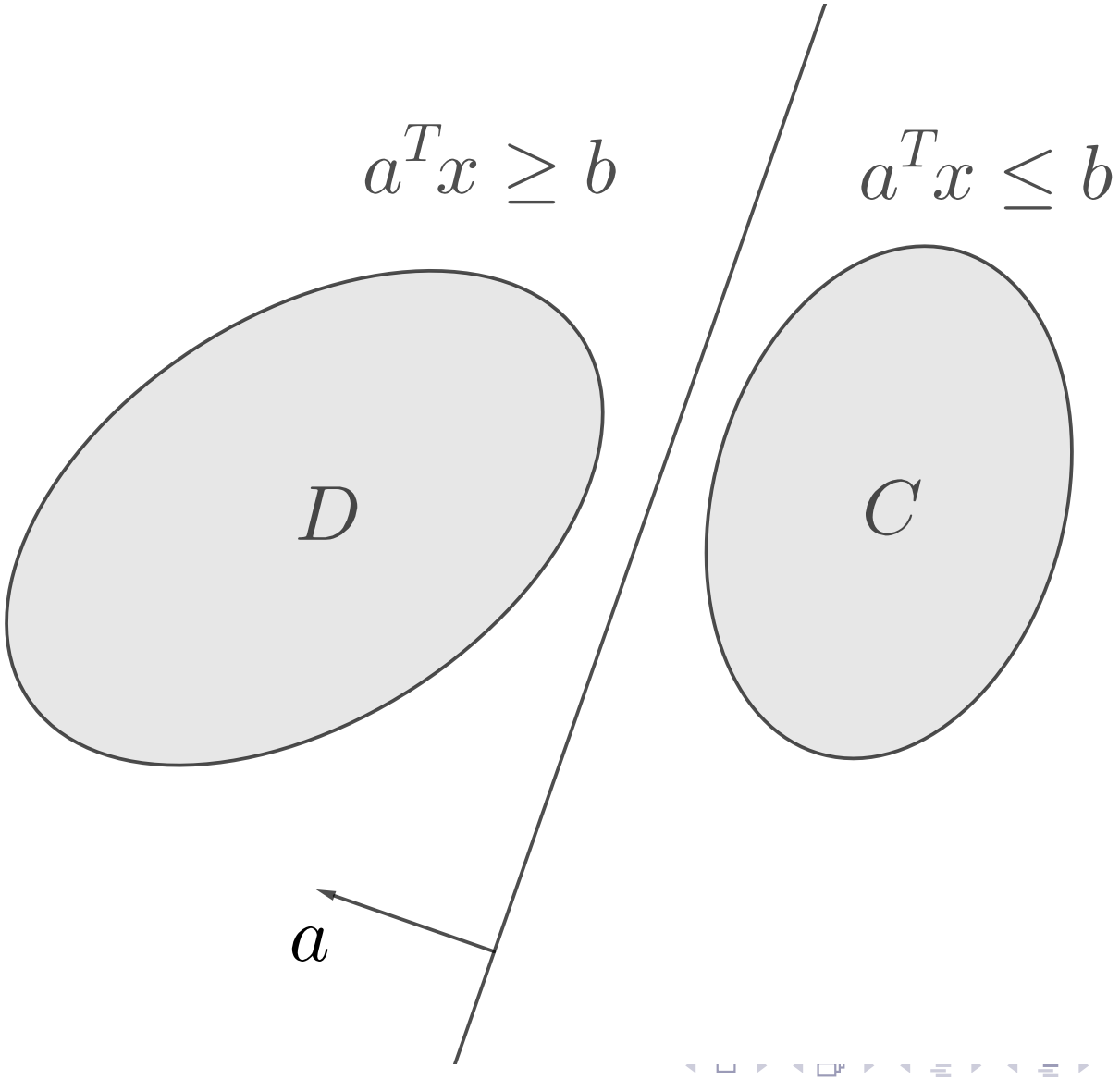
$$a^{\mathrm{T}}x \geqslant b, \forall x \in \mathcal{D},$$

即超平面$\{x|a^{\mathrm{T}}x = b\}$分离了$\mathcal{C}$和$\mathcal{D}$.

超平面分离定理表明, 如果要软划分$\mathbb{R}^n$中的2个凸集, 则只需要求得一个适当的超平面即可. 这在分类问题中属于很容易解决的问题. 实际上, 如果有任何一个集合不是凸集, 则定理一般不成立, 此时我们若要划分不同的集合, 则一般需要使用更加复杂的平面.

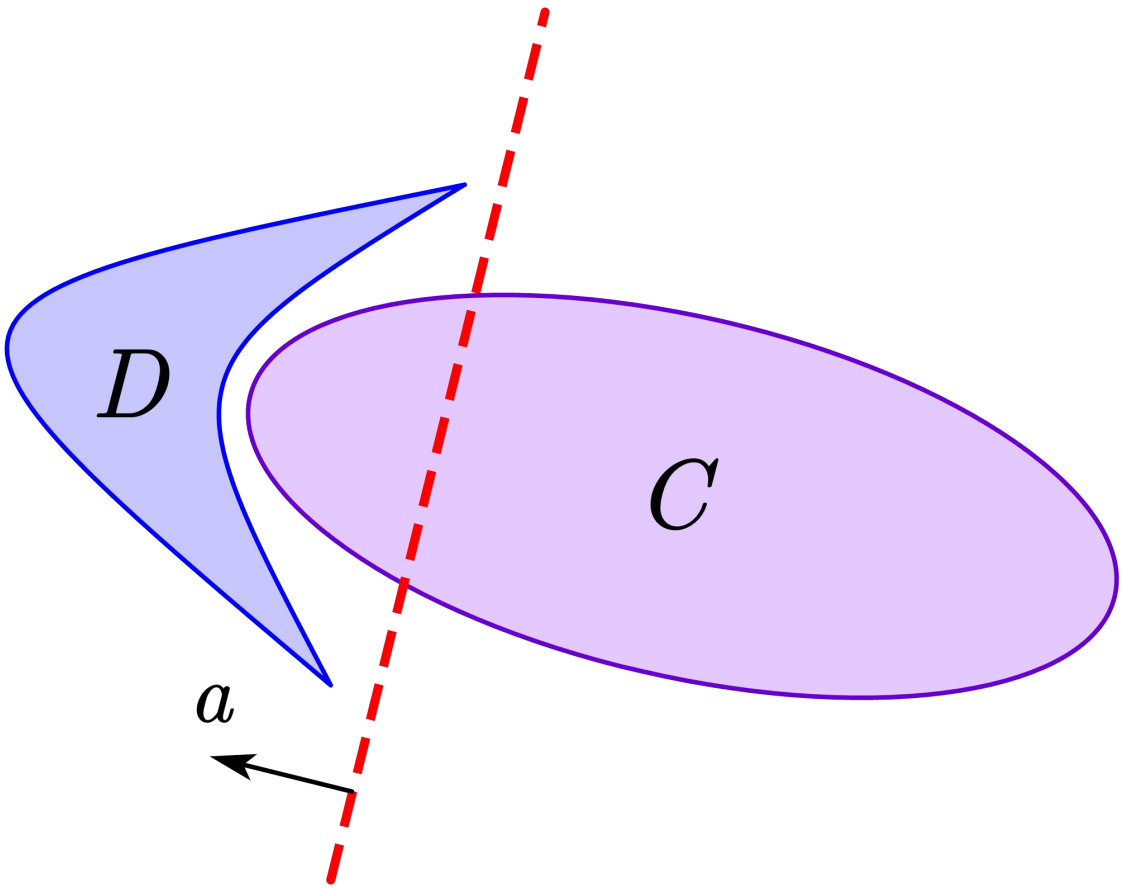
分离超平面的示意

例　下图是$\mathbb{R}^2$中的2个凸集, 我们使用超平面即可轻松划分.



分离超平面的示意

例　下图是$\mathbb{R}^2$中的两个集合, 其中一个不为凸集. 我们无法使用超平面对其划分, 而必须使用更加复杂的平面. 这就给划分问题带来了巨大的挑战.



分离超平面定理证明

这里仅考虑一个特殊情形。假设存在$c \in C$和$d \in D$使得:

$$\|c-d\|_2 = \mathbf{dist}(C,D) = \inf\{\|u-v\|_2 \mid u \in C, v \in D\} > 0.$$

定义$a = d-c$, $b = (\|d\|_2^2 - \|c\|_2^2)/2$和

$$f(x) = a^{\mathrm{T}}x - b = (d-c)^{\mathrm{T}}(x-(d+c)/2).$$

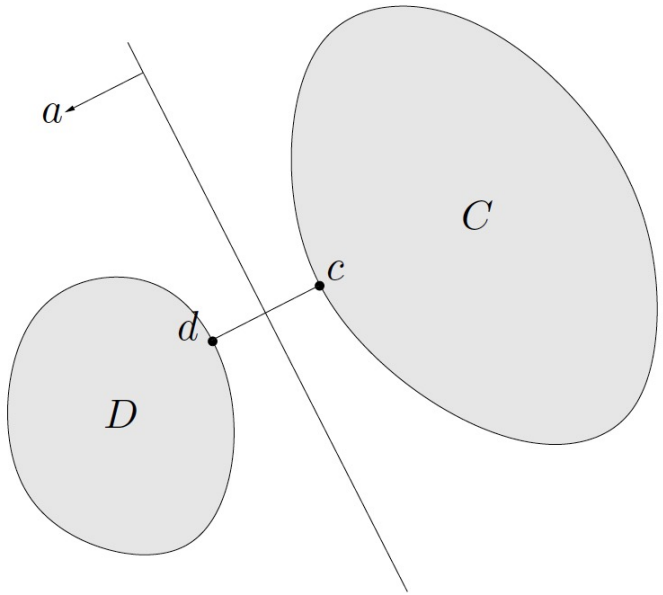
以下证:$f(x) \le 0, \forall x \in C$且$f(x) \ge 0, \forall x \in D$，即给出了分离超平面。

- 先证$f(x) \ge 0, \forall x \in D$。其它情形类似。
- 假设存在$u \in D$，使得

$$f(u) = (d-c)^{\mathrm{T}}(u-(d+c)/2) < 0.$$

可以将$f(u)$写成:

$$f(u) = (d-c)^{\mathrm{T}}(u-d) + \|d-c\|_2^2/2.$$



分离超平面定理证明

- 因此有：

$$(d-c)^{\mathrm{T}}(u-d)<0.$$

- 对于$t\in[0,1]$，构造d与u的凸组合$z(t)=d+t(u-d)$，因此$z(t)$也在集合D里. 由于

$$\frac{d}{dt}\|z(t)-c\|_2^2|_{t=0}=2(d-c)^{\mathrm{T}}(u-d)<0,$$

因此存在充分小的$t_1\in(0,1]$，使得

$$\|z(t_1)-c\|_2<\|d-c\|_2.$$

这意味着点$z(t_1)$到c的距离比d近，矛盾。

严格分离定理

我们在超平面分离时提到了软划分的概念, 其表明若集合仅是凸集, 则定理中等号可能成立, 即某一凸集与超平面相交(请尝试举一个简单例子). 很多时候进一步要求超平面与任何凸集都不交, 为此我们需要加强定理的条件.

定理

严格分离定理 如果$\mathcal{C}$和$\mathcal{D}$是不相交的凸集, 且$\mathcal{C}$是闭集, $\mathcal{D}$是紧集, 则存在非零向量a和常数b, 使得

$$a^{\mathrm{T}}x < b, \forall x \in \mathcal{C},$$

且

$$a^{\mathrm{T}}x > b, \forall x \in \mathcal{D},$$

即超平面$\{x|a^{\mathrm{T}}x = b\}$严格分离了$\mathcal{C}$和$\mathcal{D}$.

此定理的退化形式即$\mathcal{D}$退化为单点集$\{x_0\}$. 此时课本中的定理成立.

支撑超平面

上述严格分离定理的退化形式要求$x_0 \notin C$. 当点x_0恰好落在C的边界上时(此时不满足"不相交"的条件), 我们可以构造超平面.

定义

支撑超平面 给定集合C以及边界上的点x_0, 如果$a \neq 0$满足$a^{\mathrm{T}}x \leqslant a^{\mathrm{T}}x_0, \forall x \in C$, 那么称集合

$$\{x | a^{\mathrm{T}}x = a^{\mathrm{T}}x_0\}$$

为C在边界点x_0处的支撑超平面.

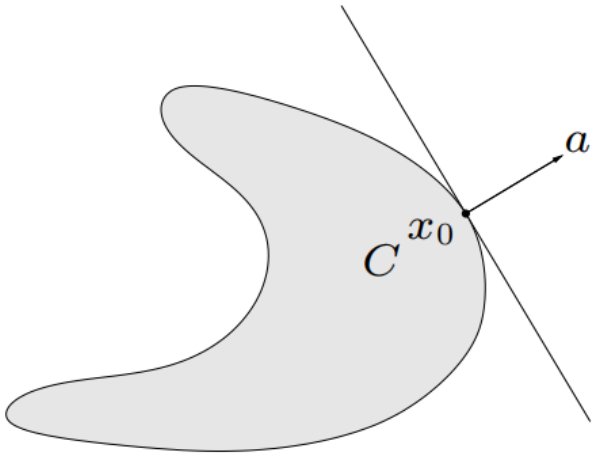
根据定义, 点x_0和集合C也被该超平面分开.
从集合上而言, 超平面$\{x | a^{\mathrm{T}}x = a^{\mathrm{T}}x_0\}$与集合$C$在点$x_0$处相切, 并且半空间$\{x | a^{\mathrm{T}}x \leqslant a^{\mathrm{T}}x_0\}$包含$C$.

支撑超平面定理

注意根据凸集成立的分离超平面定理, 凸集上任何的边界点都满足支撑超平面存在的条件, 则对于凸集成立如下的定理.

定理

支撑超平面定理 若C是凸集, 则C的任意边界点处都存在支撑超平面.



支撑超平面定理有非常强的几何直观: 给定一个平面后, 可把凸集边界上的任意一点当成支撑点, 将凸集放在该平面上.

这也是凸集的特殊性质, 一般的集合甚至无法保证存在平面上的支撑点.

最优化算法：凸函数

陈士祥

中国科学技术大学

致谢：本教案部分参考北京大学文再文教授《最优化与建模》

提纲

梯度

定义 (梯度)

给定函数$f:\mathbb{R}^n \to \mathbb{R}$，且$f$在点$x$的一个邻域内有意义，若存在向量$g \in \mathbb{R}^n$满足

$$\lim_{p\to 0} \frac{f(x+p)-f(x)-g^{\mathrm{T}}p}{\|p\|} = 0,$$

其中$\|\cdot\|$是任意的向量范数，就称f在点x处可微（或Fréchet 可微）. 此时g称为f在点x处的梯度，记作$\nabla f(x)$. 如果对区域D上的每一个点x都有$\nabla f(x)$存在，则称f在D上可微.

若f在点x处的梯度存在，在定义式中令$p=\varepsilon e_i$，e_i是第i个分量为1的单位向量，可知$\nabla f(x)$的第i个分量为$\dfrac{\partial f(x)}{\partial x_i}$. 因此，

$$\nabla f(x) = \left[\frac{\partial f(x)}{\partial x_1}, \frac{\partial f(x)}{\partial x_2}, \cdots, \frac{\partial f(x)}{\partial x_n}\right]^{\mathrm{T}}.$$

海瑟矩阵

定义 (海瑟矩阵)

如果函数$f(x):\mathbb{R}^n \to \mathbb{R}$在点$x$处的二阶偏导数$\frac{\partial^2 f(x)}{\partial x_i \partial x_j}$ $i,j=1,2,\cdots,n$都存在，则f 在点x 处的海瑟矩阵为：

$$\nabla^2 f(x) = \begin{bmatrix} \frac{\partial^2 f(x)}{\partial x_1^2} & \frac{\partial^2 f(x)}{\partial x_1 \partial x_2} & \frac{\partial^2 f(x)}{\partial x_1 \partial x_3} & \cdots & \frac{\partial^2 f(x)}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f(x)}{\partial x_2 \partial x_1} & \frac{\partial^2 f(x)}{\partial x_2^2} & \frac{\partial^2 f(x)}{\partial x_2 \partial x_3} & \cdots & \frac{\partial^2 f(x)}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \vdots & & \vdots \\ \frac{\partial^2 f(x)}{\partial x_n \partial x_1} & \frac{\partial^2 f(x)}{\partial x_n \partial x_2} & \frac{\partial^2 f(x)}{\partial x_n \partial x_3} & \cdots & \frac{\partial^2 f(x)}{\partial x_n^2} \end{bmatrix}$$

当$\nabla^2 f(x)$ 在区域D 上的每个点x 处都存在时，称f 在D 上二阶可微.
若$\nabla^2 f(x)$ 在D 上还连续，则称f 在D 上二阶连续可微，可以证明此时海瑟矩阵是一个对称矩阵.

矩阵变量函数的导数

多元函数梯度的定义可以推广到变量是矩阵的情形．对于以$m \times n$矩阵X为自变量的函数$f(X)$，若存在矩阵$G \in \mathbb{R}^{m\times n}$满足

$$\lim_{V\to 0} \frac{f(X+V) - f(X) - \langle G, V\rangle}{\|V\|} = 0,$$

其中$\|\cdot\|$是任意矩阵范数，就称矩阵变量函数f在X处Fréchet 可微，称G为f 在Fréchet 可微意义下的梯度．令$\frac{\partial f}{\partial x_{ij}}$ 表示f 关于x_{ij} 的偏导数．矩阵变量函数$f(X)$的梯度为

$$\nabla f(x) = \begin{bmatrix} \frac{\partial f}{\partial x_{11}} & \frac{\partial f}{\partial x_{12}} & \cdots & \frac{\partial f}{\partial x_{1n}} \\ \frac{\partial f}{\partial x_{21}} & \frac{\partial f}{\partial x_{22}} & \cdots & \frac{\partial f}{\partial x_{2n}} \\ \vdots & \vdots & & \vdots \\ \frac{\partial f}{\partial x_{m1}} & \frac{\partial f}{\partial x_{m2}} & \cdots & \frac{\partial f}{\partial x_{mn}} \end{bmatrix}.$$

矩阵变量函数的导数

在实际应用中，矩阵Fréchet 可微的定义和使用往往比较繁琐，为此我们需要介绍另一种定义——Gâteaux 可微.

定义 (Gâteaux 可微)

设$f(X)$为矩阵变量函数，如果对任意方向$V \in \mathbb{R}^{m\times n}$，存在矩阵$G \in \mathbb{R}^{m\times n}$满足

$$\lim_{t\to 0} \frac{f(X+tV) - f(X) - t\langle G, V\rangle}{t} = 0,$$

则称f 关于X 是Gâteaux 可微的. 满足上式的G称为f在X处在Gâteaux 可微意义下的梯度.

可以证明，当f是Fréchet 可微函数时，f也是Gâteaux 可微的，且这两种意义下的梯度相等.

矩阵变量函数的导数

- 线性函数：$f(X)=\mathrm{tr}(AX^{\mathrm{T}}B)$，其中$A\in\mathbb{R}^{p\times n},B\in\mathbb{R}^{m\times p},X\in\mathbb{R}^{m\times n}$
对任意方向$V\in\mathbb{R}^{m\times n}$以及$t\in\mathbb{R}$，有

$$\lim_{t\to 0}\frac{f(X+tV)-f(X)}{t}=\lim_{t\to 0}\frac{\mathrm{tr}(A(X+tV)^{\mathrm{T}}B)-\mathrm{tr}(AX^{\mathrm{T}}B)}{t}$$
$$=\mathrm{tr}(AV^{\mathrm{T}}B)=\langle BA,V\rangle .$$

因此，$\nabla f(X)=BA$.

- 二次函数：$f(X,Y)=\frac{1}{2}\|XY-A\|_F^2$，其中$(X,Y)\in\mathbb{R}^{m\times p}\times\mathbb{R}^{p\times n}$
对变量Y，取任意方向V以及充分小的$t\in\mathbb{R}$，有

$$f(X,Y+tV)-f(X,Y)=\frac{1}{2}\|X(Y+tV)-A\|_F^2-\frac{1}{2}\|XY-A\|_F^2$$
$$=\langle tXV,XY-A\rangle+\frac{1}{2}t^2\|XV\|_F^2$$
$$=t\left\langle V,X^{\mathrm{T}}(XY-A)\right\rangle+\mathcal{O}(t^2).$$

由定义可知$\frac{\partial f}{\partial Y}=X^{\mathrm{T}}(XY-A)$.
对变量X，同理可得$\frac{\partial f}{\partial X}=(XY-A)Y^{\mathrm{T}}$.

矩阵变量函数的导数

- *ln-det* 函数：$f(X)=\ln(\det(X))$, $X\in\mathcal{S}^n_{++}$，给定$X\succ 0$，对任意方向$V\in\mathcal{S}^n$ 以及$t\in\mathbb{R}$，我们有

$$\begin{aligned}f(X+tV)-f(X)&=\ln(\det(X+tV))-\ln(\det(X))\\&=\ln(\det(X^{1/2}(I+tX^{-1/2}VX^{-1/2})X^{1/2}))-\ln(\det(X))\\&=\ln(\det(I+tX^{-1/2}VX^{-1/2})).\end{aligned}$$

由于$X^{-1/2}VX^{-1/2}$是对称矩阵，所以它可以正交对角化，不妨设它的特征值为$\lambda_1,\lambda_2,\cdots,\lambda_n$，则

$$\ln(\det(I+tX^{-1/2}VX^{-1/2}))=\ln\prod_{i=1}^n(1+t\lambda_i)$$

$$=\sum_{i=1}^n\ln(1+t\lambda_i)=\sum_{i=1}^n t\lambda_i+\mathcal{O}(t^2)=t\,\mathrm{tr}(X^{-1/2}VX^{-1/2})+\mathcal{O}(t^2)$$

$$=t\left\langle(X^{-1})^{\mathrm{T}},V\right\rangle+\mathcal{O}(t^2).$$

因此，我们得到结论$\nabla f(X)=(X^{-1})^{\mathrm{T}}$.

广义实值函数与适当函数

定义 (广义实值函数)

令$\overline{\mathbb{R}} \xlongequal{\text{def}} \mathbb{R} \cup \{\pm\infty\}$为广义实数空间，则映射$f : \mathbb{R}^n \to \overline{\mathbb{R}}$称为广义实值函数.

和数学分析一样，我们规定

$$-\infty < a < +\infty, \quad \forall\, a \in \mathbb{R}$$

$$(+\infty) + (+\infty) = +\infty, \quad +\infty + a = +\infty, \ \forall\, a \in \mathbb{R}.$$

定义 (适当函数)

给定广义实值函数f和非空集合$\mathcal{X}$. 如果存在$x \in \mathcal{X}$使得$f(x) < +\infty$，并且对任意的$x \in \mathcal{X}$，都有$f(x) > -\infty$，那么称函数f关于集合$\mathcal{X}$是适当的.

概括来说，适当函数f的特点是“至少有一处取值不为正无穷”，以及“处处取值不为负无穷”.

下水平集与上方图

定义 (α-下水平集)

对于广义实值函数$f: \mathbb{R}^n \to \overline{\mathbb{R}}$,

$$C_\alpha = \{x \,|\, f(x) \leq \alpha\}$$

称为f 的α-下水平集.

定义 (上方图)

对于广义实值函数$f: \mathbb{R}^n \to \overline{\mathbb{R}}$,

$$\operatorname{epi} f = \{(x,t) \in \mathbb{R}^{n+1} \,|\, f(x) \leq t\}$$

称为f 的上方图.

闭函数

定义 (闭函数)

设$f: \mathbb{R}^n \to \overline{\mathbb{R}}$为广义实值函数，若$\operatorname{epi} f$为闭集，则称$f$为闭函数.

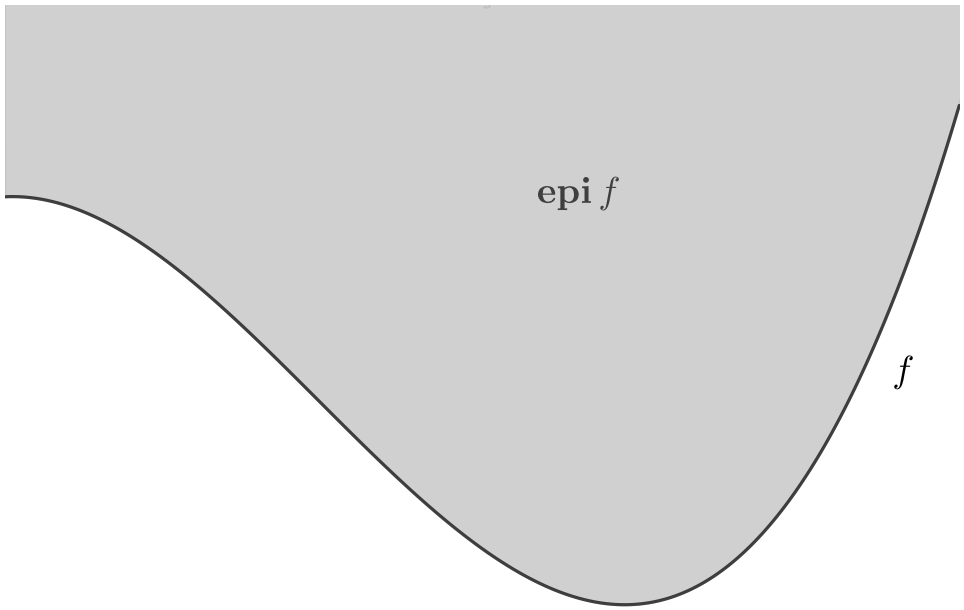


Figure: 函数f和其上方图$\operatorname{epi} f$

下半连续函数

定义 (下半连续函数)

设广义实值函数$f: \mathbb{R}^n \to \overline{\mathbb{R}}$，若对任意的$x \in \mathbb{R}^n$，有

$$\liminf_{y \to x} f(y) \geq f(x),$$

则$f(x)$为下半连续函数.

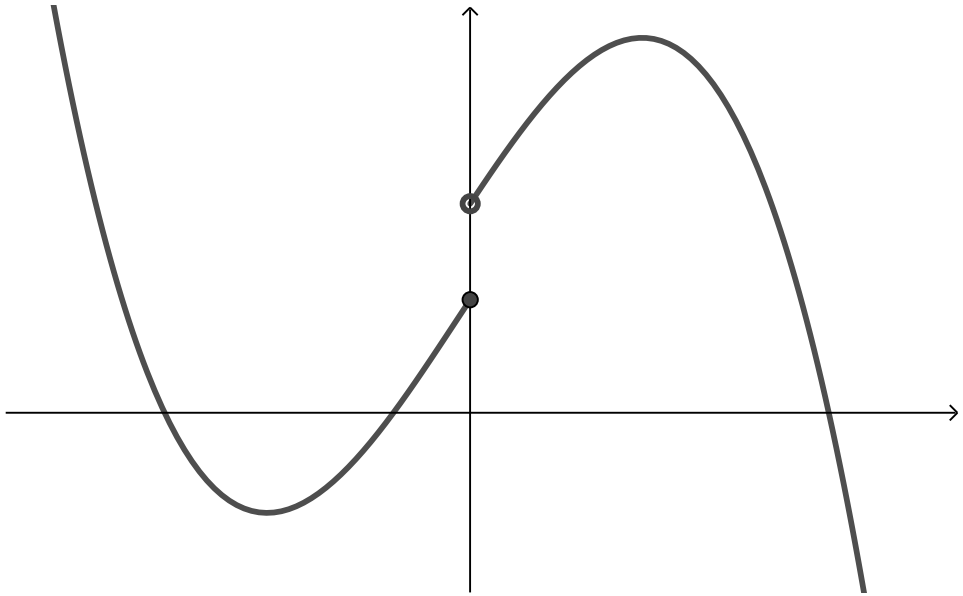


Figure: 下半连续函数$f(x)$

闭函数与下半连续函数

虽然表面上看这两种函数的定义方式截然不同，但闭函数和下半连续函数是等价的.

定理

设广义适当实值函数$f\colon \mathbb{R}^n \to \overline{\mathbb{R}}$，则以下命题等价：

1. $f(x)$的任意α-下水平集都是闭集；
2. $f(x)$是下半连续的；
3. $f(x)$是闭函数.

适当性在这里保证了下水平集定义有意义。

闭函数与下半连续函数

闭（下半连续）函数间的简单运算会保持原有性质：

- 加法：若f 与g 均为适当的闭（下半连续）函数，并且$\operatorname{dom} f \cap \operatorname{dom} g \neq \emptyset$，则$f+g$ 也是闭（下半连续）函数．其中适当函数的条件是为了避免出现未定式$(-\infty)+(+\infty)$ 的情况；
- 仿射映射的复合：若f 为闭（下半连续）函数，则$f(Ax+b)$ 也为闭（下半连续）函数；
- 取上确界：若每一个函数f_α 均为闭（下半连续）函数，则$\sup_\alpha f_\alpha(x)$ 也为闭（下半连续）函数．

提纲

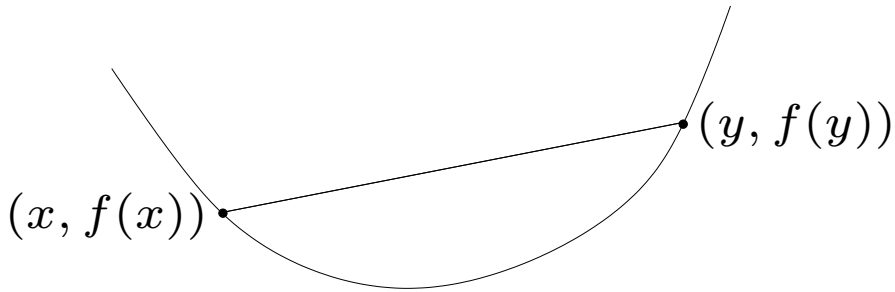
凸函数的定义

定义 (凸函数)

$f:\mathbb{R}^n \to \mathbb{R}$ 为适当函数，如果$\mathrm{dom}\, f$ 是凸集，且

$$f(\theta x + (1-\theta)y) \le \theta f(x) + (1-\theta)f(y)$$

对所有$x, y \in \mathrm{dom}\, f$, $0 \le \theta \le 1$ 都成立，则称f 是凸函数



- 若f 是凸函数，则称$-f$ 是凹函数
- 若对所有$x, y \in \mathrm{dom}\, f$, $x \ne y$, $0 < \theta < 1$，有

$$f(\theta x + (1-\theta)y) < \theta f(x) + (1-\theta)f(y)$$

则称f 是严格凸函数

一元凸函数的例子

凸函数:

- 仿射函数: 对任意$a, b \in \mathbb{R}$，$ax + b$ 是$\mathbb{R}$ 上的凸函数
- 指数函数: 对任意$a \in \mathbb{R}$，e^{ax} 是$\mathbb{R}$ 上的凸函数
- 幂函数: 对$\alpha \geq 1$ 或$\alpha \leq 0$，x^{α} 是$\mathbb{R}_{++}$上的凸函数
- 绝对值的幂: 对$p \geq 1$，$|x|^p$ 是$\mathbb{R}$上的凸函数
- 负熵: $x \log x$ 是$\mathbb{R}_{++}$上的凸函数

凹函数:

- 仿射函数: 对任意$a, b \in \mathbb{R}$，$ax + b$ 是$\mathbb{R}$ 上的凹函数
- 幂函数: 对$0 \leq \alpha \leq 1$，x^{α} 是$\mathbb{R}_{++}$上的凹函数
- 对数函数: $\log x$是$\mathbb{R}_{++}$上的凹函数

多元凸函数的例子

所有的仿射函数既是凸函数，又是凹函数。所有的范数都是凸函数。

欧氏空间$\mathbb{R}^n$中的例子

- 仿射函数: $f(x) = a^T x + b$
- 范数: $\|x\|_p = (\sum_{i=1}^n |x_i|^p)^{1/p}$ $(p \geq 1)$ ；特别地，$\|x\|_\infty = \max_k |x_k|$

矩阵空间$\mathbb{R}^{m\times n}$中的例子

- 仿射函数:

$$f(X) = \mathrm{tr}(A^T X) + b = \sum_{i=1}^m \sum_{j=1}^n A_{ij} X_{ij} + b$$

- 谱范数:

$$f(X) = \|X\|_2 = \sigma_{\max}(X) = (\lambda_{\max}(X^T X))^{1/2}$$

强凸函数

强凸函数比凸函数具有更好的性质。

- 定义1: 若存在常数$m > 0$，使得

$$g(x) = f(x) - \frac{m}{2}\|x\|^2$$

为凸函数，则称$f(x)$为强凸函数，其中m为强凸参数. 为了方便我们也称$f(x)$为m-强凸函数.

- 定义2: 若存在常数$m > 0$，使得对任意$x, y \in \mathrm{dom} f$以及$\theta \in (0,1)$，有

$$f(\theta x + (1-\theta y)) \leq \theta f(x) + (1-\theta) f(y) - \frac{m}{2}\theta(1-\theta)\|x-y\|^2,$$

则称$f(x)$为强凸函数，其中m为强凸参数.

- 设f为强凸函数且存在最小值，则f的最小值点唯一. （想一想为什么强凸函数不一定有最小值？）

凸函数判定定理

将函数限制在任意直线上，然后判断对应的一维函数是否是凸的.

定理

$f:\mathbb{R}^n\to\mathbb{R}$ 是凸函数，当且仅当对每个$x\in\operatorname{dom} f, v\in\mathbb{R}^n$，函数$g:\mathbb{R}\to\mathbb{R}$是关于$t$的凸函数

$$g(t)=f(x+tv),\quad \operatorname{dom} g=\{t|x+tv\in\operatorname{dom} f\}$$

凸函数判定定理

Proof.

必要性:设$f(x)$是凸函数，要证$g(t)=f(x+tv)$是凸函数．先说明domg是凸集．对任意的$t_1,t_2\in \mathrm{dom}g$以及$\theta\in(0,1)$，

$$x+t_1v\in \mathrm{dom}f, x+t_2v\in \mathrm{dom}f$$

由dom$f$是凸集可知$x+(\theta t_1+(1-\theta)t_2)v\in \mathrm{dom}f$,
这说明$\theta t_1+(1-\theta)t_2\in \mathrm{dom}g$，即domg是凸集．此外，我们有

$$\begin{aligned} g(\theta t_1+(1-\theta)t_2) &= f(x+(\theta t_1+(1-\theta)t_2)v) \\ &= f(\theta(x+t_1v)+(1-\theta)(x+t_2v)) \\ &\le \theta f(x+t_1v)+(1-\theta)f(x+t_2v) \\ &= \theta g(t_1)+(1-\theta)g(t_2). \end{aligned}$$

结合以上两点得到函数$g(t)$是凸函数．

凸函数判定定理

Proof.

充分性:先说明dom$f$是凸集，取$v = y - x$，以及$t_1 = 0, t_2 = 1$，由dom$g$是凸集可知$\theta \cdot 0 + (1-\theta)\cdot 1 \in \text{dom}g$，即$\theta x + (1-\theta)y \in \text{dom}f$，这说明dom$f$是凸集. 再根据$g(t) = f(x+tv)$的凸性，我们有

$$\begin{aligned} g(1-\theta) &= g(\theta t_1 + (1-\theta)t_2) \\ &\le \theta g(t_1) + (1-\theta)g(t_2) \\ &= \theta g(0) + (1-\theta)g(1) \\ &= \theta f(x) + (1-\theta)f(y). \end{aligned}$$

而等式左边有

$$g(1-\theta) = f(x + (1-\theta)(y-x)) = f(\theta x + (1-\theta)y),$$

这说明$f(x)$是凸函数.

上方图

定理

函数$f(x)$为凸函数当且仅当其上方图epi$f$是凸集.

Proof.

必要性：若f为凸函数，则对任意$(x_1,y_1),(x_2,y_2)\in \text{epi}f, t\in[0,1]$,

$$ty_1+(1-t)y_2\geq tf(x_1)+(1-t)f(x_2)\geq f(tx_1+(1-t)x_2),$$

故$(tx_1+(1-t)x_2, ty_1+(1-t)y_2)\in \text{epi}f, t\in[0,1]$.
充分性：若epi$f$是凸集，则对任意$x_1,x_2\in \text{dom}\, f, t\in[0,1]$,

$$(tx_1+(1-t)x_2, tf(x_1)+(1-t)f(x_2))\in \text{epi}f \Rightarrow$$
$$f(tx_1+(1-t)x_2)\leq tf(x_1)+(1-t)f(x_2).$$

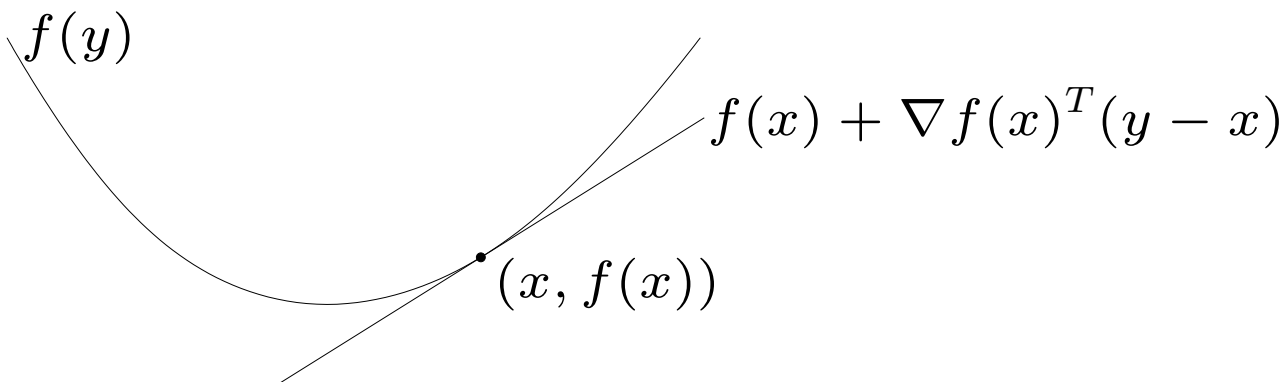
□

一阶条件

定理

一阶条件：对于定义在凸集上的可微函数f，f是凸函数当且仅当

$$f(y) \geq f(x) + \nabla f(x)^T (y - x) \quad \forall x, y \in \operatorname{dom} f$$



几何直观：f的一阶逼近始终在f的图像下方

一阶条件

Proof.

必要性：设f是凸函数，则对于任意的$x, y \in \mathrm{dom} f$以及$t \in (0,1)$，有

$$tf(y) + (1-t)f(x) \geq f(x + t(y-x)).$$

将上式移项，两边同时除以t，注意$t > 0$，则

$$f(y) - f(x) \geq \frac{f(x + t(y-x)) - f(x)}{t}.$$

令$t \to 0$，由极限保号性可得

$$f(y) - f(x) \geq \lim_{t \to 0} \frac{f(x + t(y-x)) - f(x)}{t} = \nabla f(x)^{\mathrm{T}}(y-x).$$

这里最后一个等式成立是由于方向导数的性质 .

一阶条件

Proof.

充分性：对任意的$x, y \in \mathrm{dom} f$以及任意的$t \in (0,1)$，定义$z = tx + (1-t)y$，应用两次一阶条件我们有

$$f(x) \geq f(z) + \nabla f(z)^{\mathrm{T}}(x - z),$$
$$f(y) \geq f(z) + \nabla f(z)^{\mathrm{T}}(y - z).$$

将上述第一个不等式两边同时乘t，第二个不等式两边同时乘$1-t$，相加得

$$tf(x) + (1-t)f(y) \geq f(z) + 0.$$

这正是凸函数的定义，因此充分性成立. □

梯度单调性

定理

设f为可微函数，则f为凸函数当且仅当$\mathrm{dom} f$为凸集且∇f为单调映射，

$$(\nabla f(x) - \nabla f(y))^{\mathrm{T}}(x - y) \geq 0, \quad \forall\, x, y \in \mathrm{dom} f.$$

Proof.

必要性：若f可微且为凸函数，根据一阶条件，我们有

$$f(y) \geq f(x) + \nabla f(x)^{\mathrm{T}}(y - x),$$
$$f(x) \geq f(y) + \nabla f(y)^{\mathrm{T}}(x - y).$$

将两式不等号左右两边相加即可得到结论.

Proof.

充分性：若∇f为单调映射，构造一元辅助函数

$$g(t) = f(x + t(y - x)), \quad g'(t) = \nabla f(x + t(y - x))^{\mathrm{T}}(y - x)$$

由∇f的单调性可知$t(g'(t) - g'(0)) \geq 0, \forall\, t \geq 0$. 因此

$$g'(t) \geq g'(0)$$

故

$$\begin{aligned} f(y) = g(1) &= g(0) + \int_0^1 g'(t)\mathrm{d}t \\ &\geq g(0) + g'(0) = f(x) + \nabla f(x)^{\mathrm{T}}(y - x). \end{aligned}$$

□

最后由一阶条件得证。

二阶条件

定理

二阶条件：设f 为定义在凸集上的二阶连续可微函数

- f 是凸函数当且仅当

$$\nabla^2 f(x) \succeq 0 \quad \forall x \in \operatorname{dom} f$$

- 如果$\nabla^2 f(x) \succ 0 \; \forall x \in \operatorname{dom} f$ ，则f 是严格凸函数

例：二次函数$f(x) = (1/2)x^T Px + q^T x + r$ (其中$P \in \mathbb{S}^n$)

$$\nabla f(x) = Px + q, \qquad \nabla^2 f(x) = P$$

f 是凸函数当且仅当$P \succeq 0$

二阶条件

Proof.

必要性：反设$f(x)$在点x处的海瑟矩阵$\nabla^2 f(x) \not\succeq 0$，即存在非零向量$v \in \mathbb{R}^n$使得$v^{\mathrm{T}} \nabla^2 f(x) v < 0$. 根据佩亚诺（Peano）余项的泰勒展开，

$$f(x+tv) = f(x) + t\nabla f(x)^{\mathrm{T}} v + \frac{t^2}{2} v^{\mathrm{T}} \nabla^2 f(x) v + o(t^2).$$

移项后等式两边同时除以t^2，

$$\frac{f(x+tv) - f(x) - t\nabla f(x)^{\mathrm{T}} v}{t^2} = \frac{1}{2} v^{\mathrm{T}} \nabla^2 f(x) v + o(1).$$

当t充分小时，

$$\frac{f(x+tv) - f(x) - t\nabla f(x)^{\mathrm{T}} v}{t^2} < 0,$$

这显然和一阶条件矛盾，因此必有$\nabla^2 f(x) \succeq 0$成立.

二阶条件

Proof.

充分性：设$f(x)$满足二阶条件$\nabla^2 f(x) \succeq 0$，对任意$x, y \in \mathrm{dom} f$，根据泰勒展开，

$$f(y) = f(x) + \nabla f(x)^{\mathrm{T}}(y-x) + \frac{1}{2}(y-x)^{\mathrm{T}}\nabla^2 f(x+t(y-x))(y-x),$$

其中$t \in (0,1)$是和x, y有关的常数. 由半正定性可知对任意$x, y \in \mathrm{dom} f$有

$$f(y) \geq f(x) + \nabla f(x)^{\mathrm{T}}(y-x).$$

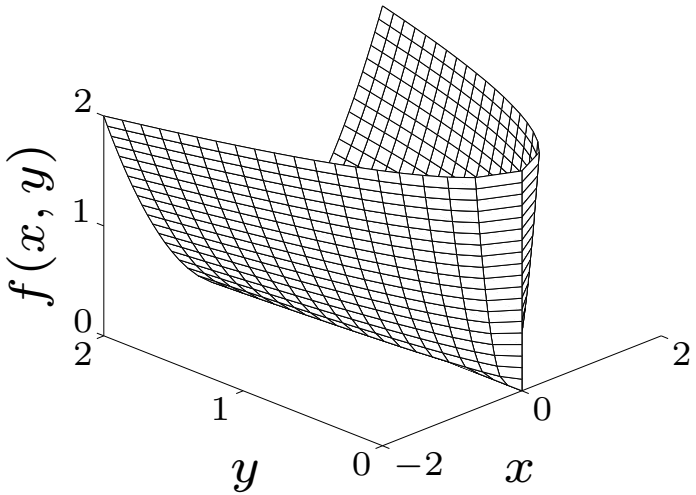
由凸函数判定的一阶条件知f为凸函数. 进一步，若$\nabla^2 f(x) \succ 0$，上式中不等号严格成立（$x \neq y$）. 利用一阶条件的充分性的证明过程可得$f(x)$为严格凸函数. □

二阶条件的应用

最小二乘函数：$f(x) = \|Ax - b\|_2^2$

$$\nabla f(x) = 2A^T(Ax - b), \quad \nabla^2 f(x) = 2A^T A$$

对任意A，f都是凸函数



quadratic-over-linear函数：$f(x,y) = x^2/y$

$$\nabla^2 f(x,y) = \frac{2}{y^3}\begin{bmatrix} y \\ -x \end{bmatrix}\begin{bmatrix} y \\ -x \end{bmatrix}^T \succeq 0$$

是区域$\{(x,y) \mid y > 0\}$上的凸函数

log-sum-exp函数: $f(x) = \log \sum_{k=1}^{n} \exp x_k$ 是凸函数

$$\nabla^2 f(x) = \frac{1}{\mathbf{1}^T z}\mathrm{diag}(z) - \frac{1}{(\mathbf{1}^T z)^2} zz^T \quad (z_k = \exp x_k)$$

要证明$\nabla^2 f(x) \succeq 0$，我们只需证明对任意$v, v^T \nabla^2 f(x) v \geq 0$，即

$$v^T \nabla^2 f(x) v = \frac{(\sum_k z_k v_k^2)(\sum_k z_k) - (\sum_k v_k z_k)^2}{(\sum_k z_k)^2} \geq 0$$

由柯西不等式，得$(\sum_k v_k z_k)^2 \leq (\sum_k z_k v_k^2)(\sum_k z_k)$，因此$f$是凸函数

几何平均: $f(x) = (\prod_{k=1}^{n} x_k)^{1/n} (x \in \mathbb{R}^n_{++})$ 是凹函数

$f(X) = -\log\det X$ 是凸函数，其中$\mathbf{dom}\, f = \mathbb{S}^n_{++}$. 任取$X \succ 0$ 以及方向$V \in \mathbb{S}^n$，将f 限制在直线$X + tV$（t 满足$X + tV \succ 0$）上，那么

$$\begin{aligned} g(t) = -\log\det(X + tV) &= -\log\det X - \log\det(I + tX^{-1/2}VX^{-1/2}) \\ &= -\log\det X - \sum_{i=1}^{n} \log(1 + t\lambda_i) \end{aligned}$$

其中λ_i 是$X^{-1/2}VX^{-1/2}$ 第i个特征值. 对每个$X \succ 0$以及方向V，g 关于t是凸的,这是因为$g'(t) = -\sum_{i=1}^{n} \frac{\lambda_i}{1+t\lambda_i}$, $g''(t) = \frac{\lambda_i^2}{(1+t\lambda_i)^2}$，因此$f$ 是凸的.

Jensen不等式

基础**Jensen**不等式: 设f 是凸函数，则对于$0 \leq \theta \leq 1$,

$$f(\theta x + (1-\theta)y) \leq \theta f(x) + (1-\theta)f(y)$$

概率**Jensen** 不等式: 设f 是凸函数，则对任意随机变量z

$$f(\mathbf{E}z) \leq \mathbf{E}f(z)$$

基础Jensen不等式可以视为概率Jensen 不等式在两点分布下的特殊情况

$$\text{prob}(z = x) = \theta, \quad \text{prob}(z = y) = 1 - \theta$$

提纲

保凸的运算

验证一个函数f是凸函数的方法：

1. 直接研究f的上方图$\operatorname{epi} f$
2. 用定义验证（通常将函数限制在一条直线上）
3. 利用一阶条件、二阶条件
4. 说明f可由简单的凸函数通过一些保凸的运算得到
   - 非负加权和
   - 与仿射函数的复合
   - 逐点取最大值
   - 与标量、向量函数的复合
   - 取下确界
   - 透视函数

非负加权和与仿射函数的复合

非负数乘: 若f 是凸函数，则αf 是凸函数，其中$\alpha \geq 0$.

求和: 若f_1, f_2 是凸函数，则$f_1 + f_2$ 是凸函数.

与仿射函数的复合: 若f 是凸函数，则$f(Ax+b)$ 是凸函数.

例子

- 线性不等式的对数障碍函数

$$f(x) = -\sum_{i=1}^{m} \log(b_i - a_i^T x), \quad \operatorname{dom} f = \{x | a_i^T x < b_i, i = 1, \ldots, m\}$$

- 仿射函数的（任意）范数：$f(x) = \|Ax+b\|$

逐点取最大值

若$f_1,\ldots,f_m$是凸函数，则$f(x)=\max\{f_1(x),\ldots,f_m(x)\}$是凸函数

例子

- 分段线性函数：$f(x)=\max_{i=1,\ldots,m}(a_i^Tx+b_i)$是凸函数
- $x\in\mathbb{R}^n$的前r个最大分量之和：

$$f(x)=x_{[1]}+x_{[2]}+\cdots+x_{[r]}$$

是凸函数($x_{[i]}$为x的从大到小排列的第i个分量)

事实上，$f(x)$可以写成如下多个线性函数取最大值的形式：

$$f(x)=\max\{x_{i_1}+x_{i_2}+\cdots+x_{i_r}|1\le i_1<i_2<\cdots<i_r\le n\}$$

逐点取上界

若对每个$y \in \mathcal{A}$，$f(x,y)$是关于x的凸函数，则

$$g(x) = \sup_{y \in \mathcal{A}} f(x,y)$$

是凸函数

例子

- 集合C的支撑函数：$S_C(x) = \sup_{y \in C} y^T x$ 是凸函数
- 集合C点到给定点x的最远距离：

$$f(x) = \sup_{y \in C} \|x - y\|$$

- 对称矩阵$X \in \mathbb{S}^n$的最大特征值

$$\lambda_{\max}(X) = \sup_{\|y\|_2 = 1} y^T X y$$

与标量函数的复合

给定函数$g:\mathbb{R}^n\to\mathbb{R}$和$h:\mathbb{R}\to\mathbb{R}$,

$$f(x)=h(g(x))$$

若 $\begin{array}{l} g\text{ 是凸函数},h\text{ 是凸函数},\tilde{h}\text{ 单调不减} \\ g\text{ 是凹函数},h\text{ 是凸函数},\tilde{h}\text{ 单调不增} \end{array}$，那么$f$是凸函数

- 对$n=1$，g,h均可微的情形，我们给出简证

$$f''(x)=h''(g(x))g'(x)^2+h'(g(x))g''(x)$$

- 注意：必须是$\tilde{h}$满足单调不减/不增的条件；如果仅是h满足单调不减/不增的条件，存在反例

推论

- 如果g是凸函数，则$\exp g(x)$ 是凸函数
- 如果g是正值凹函数，则$1/g(x)$ 是凸函数

与向量函数的复合

给定函数$g:\mathbb{R}^n \to \mathbb{R}^k$和$h:\mathbb{R}^k \to \mathbb{R}$:

$$f(x) = h(g(x)) = h(g_1(x), g_2(x), \ldots, g_k(x))$$

若 $\begin{matrix} g_i \text{ 是凸函数}, h \text{ 是凸函数}, \tilde{h} \text{ 关于每个分量单调不减} \\ g_i \text{ 是凹函数}, h \text{ 是凸函数}, \tilde{h} \text{ 关于每个分量单调不增} \end{matrix}$ ，那么f是凸函数

对$n=1$，g, h均可微的情形，我们给出简证

$$f''(x) = g'(x)^T \nabla^2 h(g(x)) g'(x) + \nabla h(g(x))^T g''(x)$$

推论

- 如果g_i是正值凹函数，则$\sum_{i=1}^m \log g_i(x)$是凹函数
- 如果g_i是凸函数，则$\log \sum_{i=1}^m \exp g_i(x)$是凸函数

取下确界

若$f(x,y)$关于(x,y)整体是凸函数，C是凸集，则

$$g(x) = \inf_{y \in C} f(x,y)$$

是凸函数.

例子

- 考虑函数$f(x,y) = x^{\mathrm{T}}Ax + 2x^{\mathrm{T}}By + y^{\mathrm{T}}Cy$，海瑟矩阵满足

$$\begin{bmatrix} A & B \\ B^{\mathrm{T}} & C \end{bmatrix} \succeq 0, \quad \text{并且} C \succ 0,$$

  则$f(x,y)$为凸函数.对y求最小值得

$$g(x) = \inf_{y} f(x,y) = x^{\mathrm{T}}(A - BC^{-1}B^{\mathrm{T}})x,$$

  因此g是凸函数.进一步地，A的Schur补$A - BC^{-1}B^{\mathrm{T}} \succeq 0$

- 点x到凸集S的距离$\mathrm{dist}(x,S) = \inf_{y \in S} \|x - y\|$是凸函数

透视函数

定义$f:\mathbb{R}^n\to\mathbb{R}$的透视函数$g:\mathbb{R}^n\times\mathbb{R}\to\mathbb{R}$,

$$g(x,t)=tf(x/t),\quad \operatorname{dom} g=\{(x,t)|x/t\in\operatorname{dom} f,t>0\}$$

若f是凸函数，则g是凸函数.

例子

- $f(x)=x^Tx$是凸函数，因此$g(x,t)=x^Tx/t$是区域$\{(x,t)\mid t>0\}$上的凸函数
- $f(x)=-\log x$是凸函数，因此相对熵函数$g(x,t)=t\log t-t\log x$是$\mathbb{R}^2_{++}$上的凸函数
- 若f是凸函数，那么

$$g(x)=(c^Tx+d)f\left((Ax+b)/(c^Tx+d)\right)$$

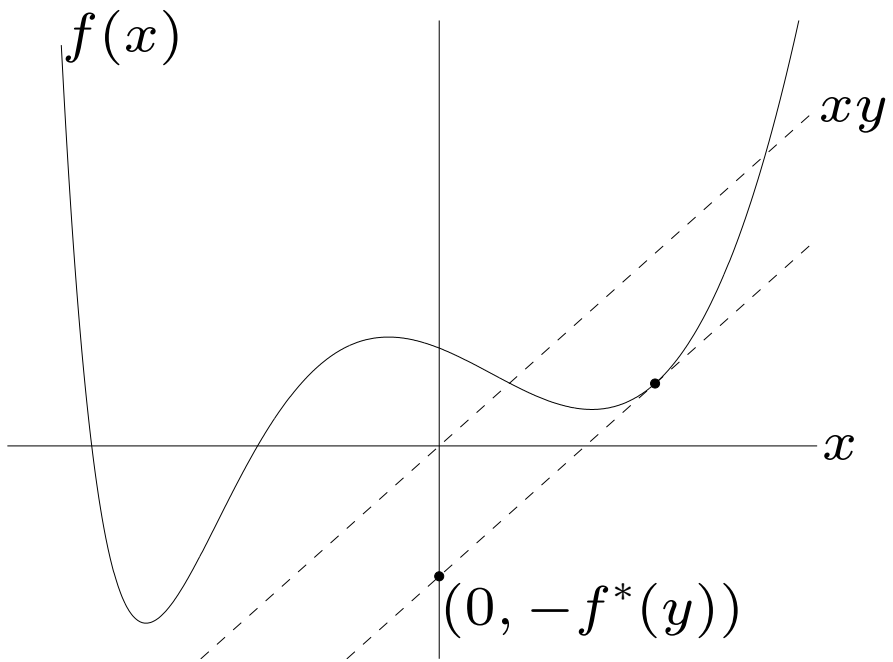
是区域$\{x|c^Tx+d>0,(Ax+b)/(c^Tx+d)\in\operatorname{dom} f\}$上的凸函数

共轭函数

适当函数f的共轭函数定义为

$$f^*(y) = \sup_{x \in \mathrm{dom}\, f} (y^T x - f(x))$$

- 几何意义：如下图所示，给定斜率y, 线性函数$y^\top x$与函数的最大差值。
- f^* 恒为凸函数，无论f 是否是凸函数



例子

- 负对数$f(x) = -\log x$

$$
\begin{aligned}
f^*(y) &= \sup_{x>0}(xy + \log x) \\
&= \begin{cases} -1 - \log(-y) & y < 0 \\ \infty & \text{其他} \end{cases}
\end{aligned}
$$

- 强凸二次函数$f(x) = (1/2)x^T Qx,\ Q \in \mathbb{S}^n_{++}$

$$
\begin{aligned}
f^*(y) &= \sup_x(y^T x - (1/2)x^T Qx) \\
&= \frac{1}{2} y^T Q^{-1} y
\end{aligned}
$$

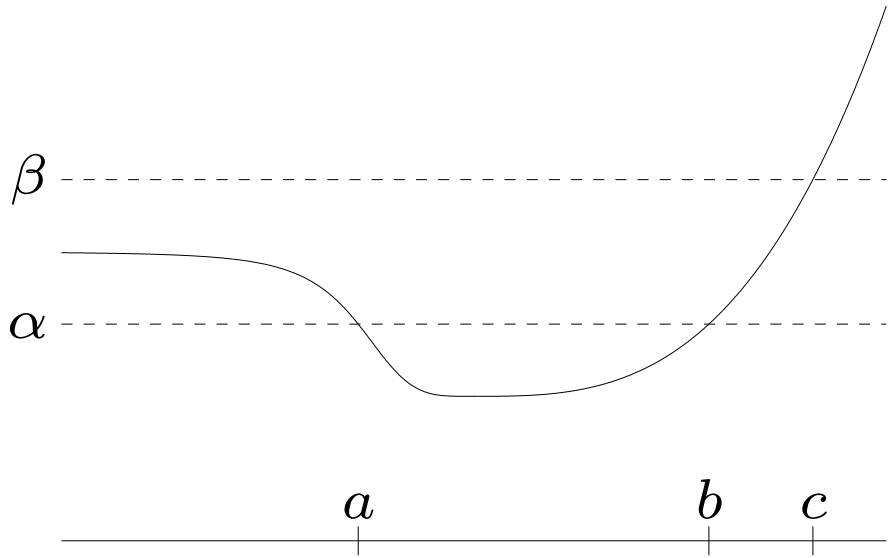
提纲

拟凸函数

$f:\mathbb{R}^n \to \mathbb{R}$ 称为拟凸的，如果$\mathrm{dom}\, f$ 是凸集，并且下水平集

$$S_\alpha = \{x \in \mathrm{dom}\, f | f(x) \leq \alpha\}$$

对任意α都是凸的



- 若f 是拟凸的，则称$-f$是拟凹的
- 若f既是拟凸的，又是拟凹的，则称f是拟线性的

拟凸、凹函数的例子

- $\sqrt{|x|}$ 是$\mathbb{R}$上的拟凸函数
- $\text{ceil}(x) = \inf\{z \in \mathbb{Z} | z \geq x\}$ 是拟线性的
- $\log x$ 是$\mathbb{R}_{++}$上的拟线性函数
- $f(x_1, x_2) = x_1 x_2$ 是$\mathbb{R}^2_{++}$上的拟凹函数
- 分式线性函数

$$f(x) = \frac{a^T x + b}{c^T x + d}, \quad \text{dom}\, f = \{x | c^T x + d > 0\}$$

是拟线性的

- 距离比值函数

$$f(x) = \frac{\|x - a\|_2}{\|x - b\|_2}, \quad \text{dom}\, f = \{x | \, \|x - a\|_2 \leq \|x - b\|_2\}$$

是拟凸的

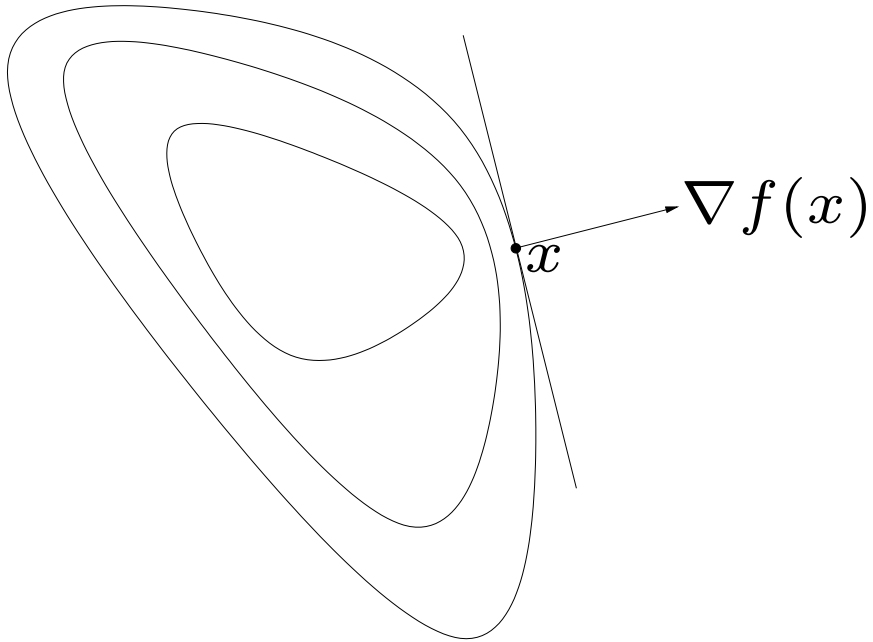
拟凸函数的性质

类**Jensen**不等式: 对拟凸函数f

$$0 \le \theta \le 1 \quad \Longrightarrow \quad f(\theta x + (1-\theta)y) \le \max\{f(x), f(y)\}$$

一阶条件: 定义在凸集上的可微函数f 是拟凸的，当且仅当

$$f(y) \le f(x) \quad \Longrightarrow \quad \nabla f(x)^T (y - x) \le 0$$



注：拟凸函数的和不一定是拟凸函数

对数凸函数

如果正值函数f满足$\log f$ 是凸函数，则f称为对数凸函数，即

$$f(\theta x + (1-\theta)y) \le f(x)^{\theta} f(y)^{1-\theta} \quad \text{for } 0 \le \theta \le 1.$$

如果$\log f$ 是凹函数，则f 称为对数凹函数，

- 幂函数：当$a \le 0$时，x^a 是$\mathbb{R}_{++}$ 上的对数凸函数；当$a \ge 0$，x^a 是$\mathbb{R}_{++}$ 上的对数凹函数
- 许多常见的概率密度函数是对数凹函数，例如正态分布

$$f(x) = \frac{1}{\sqrt{(2\pi)^n \det \Sigma}} e^{-\frac{1}{2}(x-\bar{x})^T \Sigma^{-1}(x-\bar{x})}$$

- 高斯分布的累计分布函数Φ 是对数凹函数

$$\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-u^2/2} du$$

对数凸、凹函数的性质

- 定义在凸集上的二阶可微函数f是对数凹的，当且仅当

$$f(x)\nabla^2 f(x) \preceq \nabla f(x)\nabla f(x)^T$$

对任意$x \in \operatorname{dom} f$成立

- 对数凹函数的乘积仍为对数凹函数
- 对数凹函数的和不一定为对数凹函数
- 若$f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ 是对数凹函数，那么

$$g(x) = \int f(x,y)dy$$

是对数凹函数

广义不等式意义下的凸函数

$f:\mathbb{R}^n \to \mathbb{R}^m$ 称为K-凸函数: 如果$\mathrm{dom}\, f$ 是凸集，并且

$$f(\theta x + (1-\theta)y) \preceq_K \theta f(x) + (1-\theta)f(y)$$

对任意$x, y \in \mathrm{dom}\, f$, $0 \le \theta \le 1$成立

例子 $f:\mathbb{S}^m \to \mathbb{S}^m, f(X) = X^2$ 是$\mathbb{S}^m_+$-凸函数

证明：对固定的$z \in \mathbb{R}^m$, $z^T X^2 z = \|Xz\|_2^2$ 关于X是凸函数，即

$$z^T(\theta X + (1-\theta)Y)^2 z \le \theta z^T X^2 z + (1-\theta) z^T Y^2 z$$

对任意$X, Y \in \mathbb{S}^m$, $0 \le \theta \le 1$成立.

因此$(\theta X + (1-\theta)Y)^2 \preceq \theta X^2 + (1-\theta)Y^2$

典型优化问题简介

陈士祥

中国科学技术大学

致谢：本教案部分参考北京大学文再文教授《最优化与建模》

典型优化问题简介

先回顾一下最优化问题的一般形式:

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & c_i(x) \le 0,\ i = 1, \cdots, m \\ & c_i(x) = 0,\ i = m+1, \cdots, m+l. \end{aligned}$$

- 按照目标和约束函数的简易程度分，可以分为线性规划和非线性规划．线性规划是指所有的目标函数和约束函数都是线性的，非线性规划是指目标函数和约束函数中至少有一个是非线性的
- 另外也可按照问题最优解的性质，分为凸优化问题与非凸优化问题．凸优化问题的任何稳定点都是全局极小点，非凸优化问题的稳定点则可能是局部极小点，全局极小点甚至是鞍点．

提纲

凸优化问题

标准形式的凸优化问题

$$\begin{array}{ll} \min & f_0(x) \\ \text{s.t.} & f_i(x) \le 0, \quad i = 1, \ldots, m \\ & a_i^T x = b_i, \quad i = 1, \ldots, p \end{array}$$

- $f_0, f_1, \ldots, f_m$ 凸函数; 线性等式约束
- 拟凸问题：如果f_0 是拟凸(且$f_1, \ldots, f_m$ 为凸函数)

经常写成：

$$\begin{array}{ll} \min & f_0(x) \\ \text{s.t.} & f_i(x) \le 0, \quad i = 1, \ldots, m \\ & Ax = b \end{array}$$

重要性质：凸问题的可行集为凸集.

例子

值得注意的是，优化问题性质和其定义的形式有关，例如

$$\begin{aligned} \min \quad & f_0(x) = x_1^2 + x_2^2 \\ \text{s.t.} \quad & f_1(x) = x_1/(1+x_2^2) \le 0 \\ & h_1(x) = (x_1 + x_2)^2 = 0 \end{aligned}$$

- f_0 凸函数; 可行集$\{(x_1, x_2) | x_1 = -x_2 \le 0\}$ 为凸集
- 根据我们定义不是凸问题:f_1 非凸, h_1 不是线性函数
- 等价于(但不完全相等) 凸优化问题 :

$$\begin{aligned} \min \quad & x_1^2 + x_2^2 \\ \text{s.t.} \quad & x_1 \le 0 \\ & x_1 + x_2 = 0 \end{aligned}$$

局部和全局极小

定理

凸优化问题的任意局部极小点都是全局最优

证明：假设x是局部极小，y全局最优且$f_0(y) < f_0(x)$.
x 局部最优意味着存在$R > 0$使得

$$z\text{可行}, \quad \|z - x\|_2 \le R \quad \Longrightarrow \quad f_0(z) \ge f_0(x).$$

考虑$z = \theta y + (1-\theta)x$ 且$\theta = R/(2\|y - x\|_2)$

- $\|y - x\|_2 > R$, 因此$0 < \theta < 1/2$
- z 是两个可行点的凸组合，因此也可行。
- $\|z - x\|_2 = R/2$，并且

$$f_0(z) \le \theta f_0(x) + (1-\theta) f_0(y) < f_0(x),$$

这与x是局部极小的假设矛盾.

提纲

线性规划基本形式

线性规划问题的一般形式如下：

$$\begin{aligned} \min_{x\in\mathbb{R}^n} \quad & c^{\mathrm{T}}x, \\ \text{s.t.} \quad & Ax = b, \\ & Gx \leq e, \end{aligned} \tag{1}$$

其中$c \in \mathbb{R}^n, A \in \mathbb{R}^{m\times n}, b \in \mathbb{R}^m, G \in \mathbb{R}^{p\times n}$ 和$e \in \mathbb{R}^p$ 是给定的矩阵和向量，$x \in \mathbb{R}^n$ 是决策变量. 在实际中，由于其他形式都可进行转化，故考虑问题(1) 的两种特殊形式：标准形式

$$\begin{aligned} \min_{x\in\mathbb{R}^n} \quad & c^{\mathrm{T}}x, \\ \text{s.t.} \quad & Ax = b, \\ & x \geq 0, \end{aligned} \tag{2}$$

以及不等式形式

$$\begin{aligned} \max_{y\in\mathbb{R}^n} \quad & b^{\mathrm{T}}y, \\ \text{s.t.} \quad & A^{\mathrm{T}}y \leq c. \end{aligned} \tag{3}$$

线性规划的发展

- 线性规划（Linear Programming, LP）是数学优化的一个重要分支，旨在最大化或最小化线性目标函数，同时受到线性不等式（称为约束条件）的限制。它是由俄国数学家Leonid Kantorovich和美国数学家George Dantzig在20世纪40年代独立发展的。
- 早期发展：1939年，Kantorovich发表了关于线性规划理论的论文。他提出了一种用于资源分配的方法，后来这种方法被称为线性规划。
- 单纯形法：1947年，George Dantzig发明了单纯形法，这是解决线性规划问题的第一个有效的算法，标志着线性规划作为一门学科的诞生。
- 后续发展：随后几十年中，线性规划领域经历了迅速发展，包括理论的深化、算法的改进，以及与其他数学领域的交叉融合，如对偶理论和内点法等。

应用场景：

线性规划广泛应用于各个领域，包括但不限于：

- 经济学：资源分配、成本最小化、利润最大化等。
- 军事：在第二次世界大战中用于军事物资的分配和后勤计划。
- 生产与制造：产品混合选择、生产计划、物流等。
- 服务行业：人力资源规划、运输路线设计、网络设计等。
- 金融：投资组合优化、风险管理等。

线性规划的应用：运输问题

- 令x_{ij}为从港口P_i运输到市场M_j的商品数量，总的运输代价为

$$\sum_{i=1}^{I}\sum_{j=1}^{J} x_{ij}b_{ij}. \tag{4}$$

- 港口P_i总输出量为$\sum_{j=1}^{J} x_{ij}$，因为港口P_i存有的商量总量为s_i，所以

$$\sum_{j=1}^{J} x_{ij} = s_i, \quad i = 1, 2, \cdots, I. \tag{5}$$

- 市场M_j总输入量为$\sum_{i=1}^{I} x_{ij}$，因为市场M_j的需求量为r_j，所以

$$\sum_{i=1}^{I} x_{ij} = r_j, \quad j = 1, 2, \cdots, J. \tag{6}$$

- 因为运输量是非负的，所以

$$x_{ij} \geq 0, \quad i = 1, 2, \cdots, I, \quad j = 1, 2, \cdots, J. \tag{7}$$

线性规划的应用：运输问题

因此，想要在约束(5)—(7) 成立的情况下极小化(4)式．针对决策变量的$I \times J$ 矩阵(x_{ij})，可以得到如下线性规划问题：

$$
\begin{aligned}
\min_{x} \quad & \sum_{i=1}^{I}\sum_{j=1}^{J} x_{ij}b_{ij}, \\
\text{s.t.} \quad & \sum_{j=1}^{J} x_{ij} = s_i, \quad i = 1, 2, \cdots, I, \\
& \sum_{i=1}^{I} x_{ij} = r_j, \quad j = 1, 2, \cdots, J, \\
& x_{ij} \geq 0, \quad i = 1, 2, \cdots, I, \quad j = 1, 2, \cdots, J.
\end{aligned}
$$

基追踪问题

基追踪问题是压缩感知中的一个基本问题，可以写为

$$\begin{aligned} \min_{x\in\mathbb{R}^n} \quad & \|x\|_1, \\ \text{s.t.} \quad & Ax = b. \end{aligned} \tag{8}$$

对每个$|x_i|$ 引入一个新的变量z_i，可以将问题(8) 转化为

$$\begin{aligned} \min_{z\in\mathbb{R}^n} \quad & \sum_{i=1}^{n} z_i, \\ \text{s.t.} \quad & Ax = b, \\ & -z_i \le x_i \le z_i, \quad i = 1, 2, \cdots, n, \end{aligned} \tag{9}$$

这是一个线性规划问题.

基追踪问题

- 除此之外，也可以引入x_i的正部和负部,其中$x_i^+ = \max\{x_i, 0\}$, $x_i^- = \max\{-x_i, 0\}$..
- 利用$x_i = x_i^+ - x_i^-, |x_i| = x_i^+ + x_i^-$，则问题(8)转化为的另外一种等价的线性规划形式可以写成

$$\begin{aligned} \min_{x^+, x^- \in \mathbb{R}^n} \quad & \sum_{i=1}^{n} (x_i^+ + x_i^-), \\ \text{s.t.} \quad & Ax^+ - Ax^- = b, \\ & x^+, x^- \geq 0. \end{aligned}$$

可以看出这也是一个线性规划问题，且与原问题等价.

数据拟合

在数据拟合中，除了常用的最小二乘模型外，还有最小ℓ_1范数模型

$$\min_{x\in\mathbb{R}^n} \quad \|Ax-b\|_1, \tag{10}$$

和最小ℓ_∞范数模型

$$\min_{x\in\mathbb{R}^n} \quad \|Ax-b\|_\infty. \tag{11}$$

这两个问题都可以转化成线性规划的形式.

- 对于问题(10)，通过引入变量$y=Ax-b$，可以得到如下等价问题：

$$\begin{aligned} \min_{x,y\in\mathbb{R}^n} \quad & \|y\|_1, \\ \text{s.t.} \quad & y=Ax-b. \end{aligned}$$

- 利用基追踪问题中类似的技巧，可以将上述绝对值优化问题转化成线性规划问题.

数据拟合

- 对于问题(11)，令$t = \|Ax - b\|_\infty$，则得到等价问题

$$\begin{aligned} \min_{x\in\mathbb{R}^n,\, t\in\mathbb{R}} \quad & t, \\ \text{s.t.} \quad & \|Ax - b\|_\infty \le t. \end{aligned}$$

- 利用ℓ_∞范数的定义，可以进一步写为

$$\begin{aligned} \min_{x\in\mathbb{R}^n,\, t\in\mathbb{R}} \quad & t, \\ \text{s.t.} \quad & -t\mathbf{1} \le Ax - b \le t\mathbf{1}, \end{aligned}$$

这是一个线性规划问题.

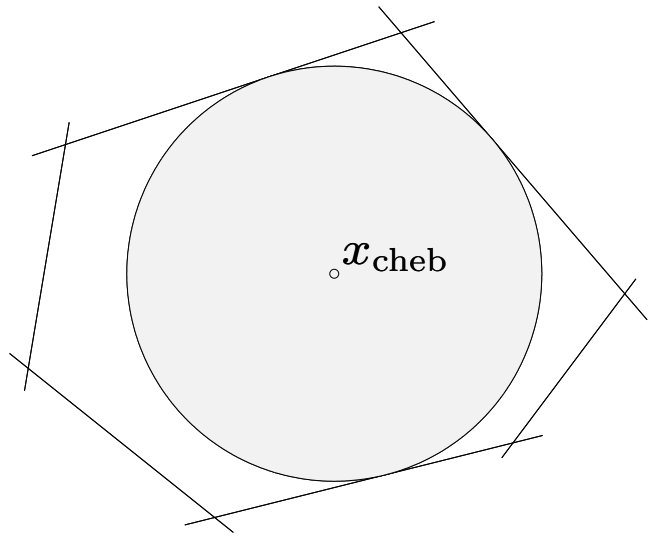
多面体的切比雪夫中心

多面体的

$$\mathcal{P} = \{x | a_i^T x \leq b_i, i = 1, \ldots, m\}$$

的切比雪夫中心即为其最大半径内接球的球心

$$\mathcal{B} = \{x_c + u | \ \|u\|_2 \leq r\}$$



- $a_i^T x \leq b_i$ 对$\forall\, x \in \mathcal{B}$ 当且仅当

$$\sup\{a_i^T(x_c + u) | \ \|u\|_2 \leq r\} = a_i^T x_c + r\|a_i\|_2 \leq b_i$$

- 因此, x_c, r 可以用LP方式求解

$$\begin{array}{ll} \max\limits_{x_c, r} & r \\ \text{s.t.} & a_i^T x_c + r\|a_i\|_2 \leq b_i, \quad i = 1, \ldots, m \end{array}$$

分式线性问题

$$\begin{aligned} \min \quad & f_0(x) \\ \text{s.t.} \quad & Gx \leq h, Ax = b \end{aligned} \tag{12}$$

分式线性函数：

$$f_0 = \frac{c^T x + d}{e^T x + f}, \quad \text{dom} f_0(x) = \{x | e^T x + f > 0\} \tag{13}$$

则该问题等价于一个线性问题：

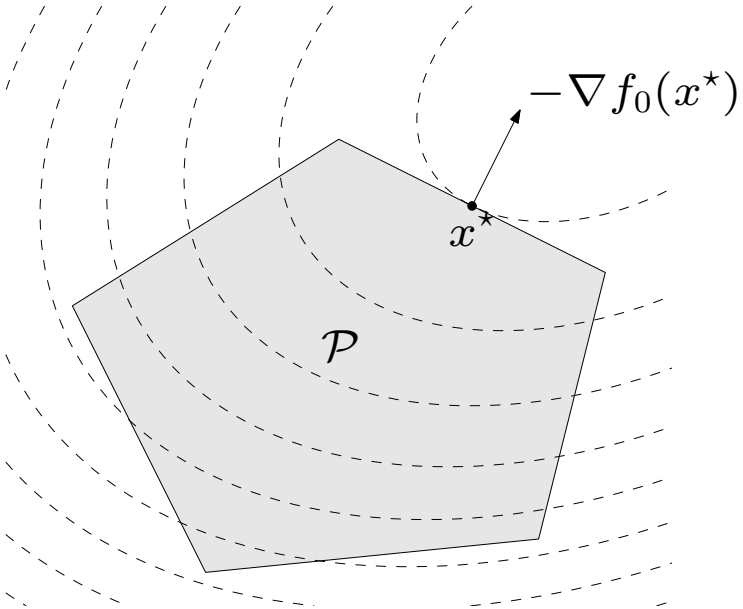
$$\begin{aligned} \min \quad & c^T y + dz \\ \text{s.t.} \quad & Gy \leq hz \\ & Ay = bz \\ & e^T y + fz = 1 \\ & z \geq 0 \end{aligned} \tag{14}$$

提纲

二次规划问题

$$\begin{aligned} \min \quad & \frac{1}{2}x^T Px + q^T x + r \\ \text{s.t.} \quad & Gx \leq h \\ & Ax = b \end{aligned} \tag{15}$$

- $P \in \mathcal{S}_+^n$,故目标函数是二次的
- 在一个多面体内最小化一个二次凸问题



二次规划的应用

- 最小二乘问题：

$$\min \|Ax - b\|_2^2 \tag{16}$$

  - 该问题的解析解为$x^* = A^\dagger b$ (其中$A^\dagger$为广义逆)
- 随机线性规划

$$\begin{aligned} \min \quad & \bar{c}^T x + \gamma x^T \Sigma x = \mathbf{E} c^T x + \gamma \mathrm{var}(c^T x) \\ \text{s.t.} \quad & Gx \leq h, \quad Ax = b \end{aligned}$$

- c 是随机向量并且均值为$\bar{c}$，方差为Σ
- $c^T x$ 均值为$\bar{c}^T x$,方差为$x^T \Sigma x$
- $\gamma > 0$ 为风险参数,控制预期成本与风险.

带有二次约束的二次规划问题(QCQP)

考虑带有二次约束的二次规划问题：

$$\begin{array}{ll} \min & (1/2)x^T P_0 x + q_0^T x + r_0 \\ \text{s.t.} & (1/2)x^T P_i x + q_i^T x + r_i \leq 0, \quad i = 1, \ldots, m \\ & Ax = b \end{array}$$

- $P_i \in \mathbb{S}_+^n$;即目标函数与约束均为二次凸的
- 如果$P_1, \ldots, P_m \in \mathbb{S}_{++}^n$, 则可行域为$m$个椭球与一个仿射集的交集.

广义不等式约束

- 凸问题的广义不等式约束

$$\begin{array}{ll} \min & f_0(x) \\ \text{s.t.} & f_i(x) \preceq_{K_i} 0, \quad i = 1, \ldots, m \\ & Ax = b \end{array}$$

  - $f_0 : \mathbb{R}^n \to \mathbb{R}$ 为凸函数; $f_i : \mathbb{R}^n \to \mathbb{R}^{k_i}$ 关于适当锥K_i为K_i-凸的.
  - 与标准凸问题有相同的性质

- 锥形式问题(具有仿射目标函数与约束的特殊情况)

$$\begin{array}{ll} \min & c^T x \\ \text{s.t.} & Fx + g \preceq_K 0 \\ & Ax = b \end{array}$$

将线性规划问题延伸到了非多面体锥上($K = \mathbb{R}^m_+$)

二次锥

- 二次锥(SOC: second-order cone):

$$Q = \left\{ x \in \mathbb{R}^{n+1} \mid \|\bar{x}\|_2 \leq x_1, \quad x = \begin{bmatrix} x_1 \\ \bar{x} \end{bmatrix} \right\}$$

- 旋转二次锥

$$Q = \left\{ x \in \mathbb{R}^{n+2} \mid \|\bar{x}\|_2^2 \leq x_1 x_2, x_1, x_2 \geq 0, \quad x = \begin{bmatrix} x_1 \\ x_2 \\ \bar{x} \end{bmatrix} \right\}$$

  - 旋转二次锥$\|\bar{x}\|^2 \leq x_1 x_2$, 其中$x_1, x_2 \geq 0$, 等价于

$$\left\| \begin{pmatrix} x_1 - x_2 \\ 2\bar{x} \end{pmatrix} \right\| \leq x_1 + x_2$$

二次锥规划(SOCP)

$$\begin{aligned} \min \quad & f^T x \\ \text{s.t.} \quad & \|A_i x + b_i\|_2 \le c_i^T x + d_i, \quad i = 1, \ldots, m \\ & Fx = g \end{aligned}$$

$(A_i \in \mathbb{R}^{n_i \times n}, F \in \mathbb{R}^{p \times n})$

- 优化问题中的不等式:

$$(c_i^T x + d_i, A_i x + b_i) \in \text{二次锥}$$

被称为二次锥约束。

- 对于$n_i = 0$ 或$A_i = 0$, 问题退化为LP; 若$c_i = 0$, 问题退化为QCQP
- 问题形式比LP与QCQP更常规
- 注意：两边平方后得到$\|A_i x + b_i\|_2^2 \le (c_i^T x + d_i)^2$，可能是非凸，故不等价! 需要加上$c_i^\top x + d_i \ge 0$. 例：$|x| \le y$，等价于平方后$x^2 - y^2 \le 0, y \ge 0$.

线性约束转化为二次锥约束

$$a_i^\top x \le b_i, i = 1, \ldots, m.$$

等价于

$$\|C_i x + d_i\| \le b_i - a_i^\top x, i = 1, \ldots, m,$$

这里$C_i = 0, d_i = 0.$

二次规划(QP)与SOCP

$$\begin{array}{ll} \min & q(x) = x^\top Q x + a^\top x + \beta \qquad \text{假设} \quad Q \succ 0, Q = Q^\top \\ \text{s.t.} & Ax = b \\ & x \geq 0 \end{array}$$

- $q(x) = \|\bar{u}\|^2 + \beta - \frac{1}{4} a^\top Q^{-1} a$, 其中 $\bar{u} = Q^{1/2} x + \frac{1}{2} Q^{-1/2} a$.
- 等价的SOCP问题形式：

$$\begin{array}{ll} \min & u_0 \\ \text{s.t.} & \bar{u} = Q^{1/2} x + \dfrac{1}{2} Q^{-1/2} a \\ & Ax = b \\ & x \geq 0, \quad (u_0, \bar{u}) \succeq_{\mathcal{Q}} 0 \end{array}$$

二次约束

$$q(x) = x^\top B^\top B x + a^\top x + \beta \le 0$$

等价于二次锥约束

$$(u_0, \bar{u}) \succeq_{\mathcal{Q}} 0,$$

其中

$$\bar{u} = \begin{pmatrix} Bx \\ \frac{a^\top x + \beta + 1}{2} \end{pmatrix} \quad \text{并且} \quad u_0 = \frac{1 - a^\top x - \beta}{2}$$

例：最小范数问题

令$\bar{v}_i = A_i x + b_i \in \mathbb{R}^{n_i}$.

- $\min_x \quad \sum_i \|\bar{v}_i\|$ 等价于

$$\begin{aligned} \min \quad & \sum_i v_{i0} \\ \text{s.t.} \quad & \bar{v}_i = A_i x + b_i \\ & (v_{i0}, \bar{v}_i) \succeq_{\mathcal{Q}} 0 \end{aligned}$$

- $\min_x \quad \max_{1 \le i \le r} \|\bar{v}_i\|$ 等价于

$$\begin{aligned} \min \quad & t \\ \text{s.t.} \quad & \bar{v}_i = A_i x + b_i \\ & (t, \bar{v}_i) \succeq_{\mathcal{Q}} 0 \end{aligned}$$

旋转二次锥例子

- 极小化仿射函数的调和平均值

$$\min \quad \sum_i 1/(a_i^\top x + \beta_i), \ \text{s.t.} \quad a_i^\top x + \beta_i > 0$$

该问题等价于

$$\begin{aligned} \min \quad & \sum_i u_i \\ \text{s.t.} \quad & \bar{v}_i = a_i^\top x + \beta_i \\ & 1 \le u_i v_i \\ & u_i \ge 0 \end{aligned}$$

提纲

半定规划

半定规划问题的一般形式如下：

$$\begin{aligned} \min \quad & c^{\mathrm{T}}x, \\ \text{s.t.} \quad & x_1A_1 + x_2A_2 + \cdots + x_nA_n + B \preceq 0, \\ & Gx = h, \end{aligned} \tag{17}$$

其中$c \in \mathbb{R}^n, A_i \in \mathcal{S}^m, i = 1, 2, \cdots, m, B \in \mathcal{S}^m, G \in \mathbb{R}^{p\times n}, h \in \mathbb{R}^p$ 为已知的向量和矩阵，$x = (x_1, x_2, \cdots, x_n) \in \mathbb{R}^n$是自变量.

- 半定规划（SDP）是线性规划在矩阵空间中的一种推广．它的目标函数和等式约束均为关于矩阵的线性函数，而它与线性规划不同的地方是其自变量取值于半正定矩阵空间.
- 若$A_1, \ldots A_n, B$都是对角矩阵，那么约束即为线性规划约束。
- 作为一种特殊的矩阵优化问题，半定规划在某些结构上和线性规划非常相似，很多研究线性规划的方法都可以作为研究半定规划的基础．由于半定规划地位的特殊性，我们将在本节中单独讨论半定规划的形式和应用.
- 由于许多非凸问题松弛为半定规划后，得到的近似问题比线性规划更紧，所以半定规划更普遍。

半定规划

类似于线性规划问题，我们考虑半定规划的标准形式

$$\begin{aligned} \min \quad & \langle C, X \rangle, \\ \text{s.t.} \quad & \langle A_1, X \rangle = b_1, \\ & \cdots \\ & \langle A_m, X \rangle = b_m, \\ & X \succeq 0, \end{aligned} \tag{18}$$

和对偶形式：

$$\begin{aligned} \min \quad & -b^{\mathrm{T}} y, \\ \text{s.t.} \quad & y_1 A_1 + y_2 A_2 + \cdots + y_n A_n \preceq C. \end{aligned} \tag{19}$$

形如(17) 式的优化问题都可以转化成(18) 式或者(19) 式的形式.

LP,SOCP是SDP的特殊形式

LP改写为SDP

$$\begin{array}{llll} \text{LP:} & \min & c^T x & \\ & \text{s.t.} & Ax \le b & \end{array} \qquad \begin{array}{llll} \text{SDP:} & \min & c^T x & \\ & \text{s.t.} & \operatorname{diag}(Ax-b) \preceq 0 & \end{array}$$

- 注意到

$$\operatorname{diag}(Ax-b) = \begin{pmatrix} a_1^\top x - b_1 & & & \\ & a_2^\top x - b_2 & & \\ & & \ddots & \\ & & & a_n^\top - b_n \end{pmatrix} = \sum_{i=1}^n x_i A_i - C,$$

其中，$A_i = \begin{pmatrix} a_{1i} & & & \\ & a_{2i} & & \\ & & \ddots & \\ & & & a_{ni} \end{pmatrix}$, $C = \begin{pmatrix} b_1 & & & \\ & b_2 & & \\ & & \ddots & \\ & & & b_n \end{pmatrix}$. 故LP的一般形式可以转换为SDP的对偶形式。

- 与线性规划约束多面体不同的是，半正定锥可能有无穷个极点，如圆盘$\{(x,y) \mid x^2+y^2 \le 1\} = \{(x,y) \mid \begin{pmatrix} 1+x & y \\ y & 1-x \end{pmatrix} \succeq 0\}$。所以求解LP的单纯形法无法拓展到求解SDP问题上，而是一般用内点法求解。内点法可以在多项式时间内找到任意精度的解。

SOCP 改写为SDP

SOCP: $\min \quad f^T x$

$$\text{s.t.} \quad \|A_i x + b_i\|_2 \le c^T x + d_i, \quad i = 1, \ldots, m$$

SDP: $\min \quad f^T x$

$$\text{s.t.} \quad \begin{bmatrix} (c_i^T x + d_i) I & A_i x + b_i \\ (A_i x + b_i)^T & c_i^T x + d_i \end{bmatrix} \succeq 0, \quad i = 1, \ldots, m$$

这里使用了**Schur**引理：对于分块矩阵$X = \begin{bmatrix} A & B \\ B^\top & C \end{bmatrix}$, 如果$A \succ 0$, 那么$X \succeq 0$ 等价于A的**Schur**补$X/A = C - B^\top A^{-1} B \succeq 0$.
考虑$c^\top x + d_i = 0$ 和$c^\top x + d_i > 0$情况即可。

非凸QCQP问题的半定规划松弛

- 考虑二次约束二次规划问题

$$\begin{aligned} \min_{x\in\mathbb{R}^n} \quad & x^{\mathrm{T}}A_0x + 2b_0^{\mathrm{T}}x + c_0, \\ \text{s.t.} \quad & x^{\mathrm{T}}A_ix + 2b_i^{\mathrm{T}}x + c_i \le 0, \quad i = 1, 2, \cdots, m, \end{aligned} \tag{20}$$

其中A_i为$n \times n$对称矩阵. 当部分A_i为对称不定矩阵时，问题(20)是NP难的非凸优化问题.

- 写出问题(20)的半定规划松弛问题. 对任意$x \in \mathbb{R}^n$以及$A \in \mathcal{S}^n$，有恒等式

$$x^{\mathrm{T}}Ax = \mathrm{Tr}(x^{\mathrm{T}}Ax) = \mathrm{Tr}(Axx^{\mathrm{T}}) = \left\langle A, xx^{\mathrm{T}} \right\rangle,$$

因此问题(20)中所有的二次项均可用下面的方式进行等价刻画：

$$x^{\mathrm{T}}A_ix + 2b_i^{\mathrm{T}}x + c_i = \left\langle A_i, xx^{\mathrm{T}} \right\rangle + 2b_i^{\mathrm{T}}x + c_i.$$

非凸QCQP问题的半定规划松弛

由上述分析，原始问题等价于

$$\begin{aligned} \min_{x\in\mathbb{R}^n} \quad & \langle A_0, X\rangle + 2b_0^{\mathrm{T}}x + c_0 \\ \text{s.t.} \quad & \langle A_i, X\rangle + 2b_i^{\mathrm{T}}x + c_i \le 0, \quad i = 1, 2, \cdots, m, \\ & X = xx^{\mathrm{T}}. \end{aligned} \tag{21}$$

进一步地，

$$\begin{aligned} x^{\mathrm{T}}A_i x + 2b_i^{\mathrm{T}}x + c_i &= \left\langle \begin{pmatrix} A_i & b_i \\ b_i^{\mathrm{T}} & c_i \end{pmatrix}, \begin{pmatrix} X & x \\ x^{\mathrm{T}} & 1 \end{pmatrix} \right\rangle, \\ &\overset{\text{def}}{=\!=} \left\langle \overline{A_i}, \overline{X} \right\rangle, \quad i = 0, 1, \cdots, m. \end{aligned}$$

二次约束二次规划问题的半定规划松弛

接下来将等价问题(21) 松弛为半定规划问题.

- 在问题(21) 中，唯一的非线性部分是约束$X = xx^{\mathrm{T}}$，我们将其松弛成半正定约束$X \succeq xx^{\mathrm{T}}$. 可以证明，$\overline{X} \succeq 0$ 与$X \succeq xx^{\mathrm{T}}$ 是等价的.
- 因此这个问题的半定规划松弛可以写成

$$\begin{aligned} \min \quad & \langle \overline{A_0}, \overline{X} \rangle \\ \text{s.t.} \quad & \langle \overline{A_i}, \overline{X} \rangle \leq 0, \quad i = 1, 2, \cdots, m, \\ & \overline{X} \succeq 0, \\ & \overline{X}_{n+1,n+1} = 1. \end{aligned}$$

其中“松弛”来源于我们将$X = xx^{\mathrm{T}}$ 替换成了$X \succeq xx^{\mathrm{T}}$.

最大割问题的半定规划松弛

- 令G为一个无向图，其节点集合为$V=\{1,2,\cdots,n\}$和边的集合为E. 令$w_{ij}=w_{ji}$为边$(i,j)\in E$上的权重，并假设$w_{ij}\geq 0,\ (i,j)\in E$. 最大割问题是找到节点集合$V$的一个子集$S$使得$S$与它的补集$\overline{S}$之间相连边的权重之和最大化.
- 可以将最大割问题写成如下整数规划的形式：令$x_j=1,\ j\in S$和$x_j=-1,\ j\in\overline{S}$，则

$$\begin{aligned}\max\quad & \frac{1}{2}\sum_{i<j}(1-x_ix_j)w_{ij}\\ \text{s.t.}\quad & x_j\in\{-1,1\},\ j=1,2,\cdots,n.\end{aligned}\tag{22}$$

- 在问题(22)中，只有当x_i与x_j不同时，目标函数中w_{ij}的系数非零. 最大割问题是一个离散优化问题，很难在多项式时间内找到它的最优解.

二次约束二次规划问题的半定规划松弛

接下来介绍如何将问题(22) 松弛成一个半定规划问题.

- 令$W = (w_{ij}) \in \mathcal{S}^n$，并定义$C = -\frac{1}{4}(\mathrm{Diag}(W\mathbf{1}) - W)$为图$G$的拉普拉斯矩阵的$-\frac{1}{4}$倍，则问题(22) 可以等价地写为

$$\begin{aligned} \min \quad & x^{\mathrm{T}} C x, \\ \text{s.t.} \quad & x_i^2 = 1,\ i = 1, 2, \cdots, n. \end{aligned}$$

  由于目标函数是关于x的二次函数，可将其等价替换为$\langle C, xx^{\mathrm{T}} \rangle$.

- 接下来令$X = xx^{\mathrm{T}}$，注意到约束$x_i^2 = 1$，这意味着矩阵X对角线元素$X_{ii} = 1$. 因此利用矩阵形式我们将最大割问题转化为

$$\begin{aligned} \min \quad & \langle C, X \rangle, \\ \text{s.t.} \quad & X_{ii} = 1,\ i = 1, 2, \cdots, n, \\ & X \succeq 0,\ \mathrm{rank}(X) = 1. \end{aligned} \tag{23}$$

- 问题(23) 和(22) 是等价的，这是因为$X = xx^{\mathrm{T}}$ 可以用约束$X \succeq 0$和$\mathrm{rank}(X) = 1$等价刻画.

极小化最大特征值

- 问题的形式可表示为：$\lambda_{\max}(A_0 + \sum_i x_i A_i)$:

$$\min \quad \lambda_{\max}(A_0 + \sum_i x_i A_i)$$

SDP形式：

$$\begin{array}{ll} \min & z \\ \text{s.t.} & zI - \sum_i x_i A_i \succeq A_0 \end{array}$$

对偶问题形式：

$$\begin{array}{ll} \max & \langle A_0, Y \rangle \\ \text{s.t.} & \langle A_i, Y \rangle = 0 \\ & \langle I, Y \rangle = 1 \\ & Y \succeq 0 \end{array}$$

- 等价形式来源于：

$$\lambda_{\max}(A) \leq t \Longleftrightarrow A \preceq tI$$

极小化二范数问题

- 令$A_i \in \mathbb{R}^{m\times n}$. 极小化$A(x) = A_0 + \sum_i x_i A_i$的二范数:

$$\min_x \quad \|A(x)\|_2$$

该问题的SDP形式:

$$\begin{aligned} \min_{x,t} \quad & t \\ \text{s.t.} \quad & \begin{pmatrix} tI & A(x) \\ A(x)^\top & tI \end{pmatrix} \succeq 0 \end{aligned}$$

- 约束形式来源于:

$$\begin{aligned} \|A(x)\|_2 \le t \quad &\Longleftrightarrow \quad A(x)^\top A(x) \preceq t^2 I, \quad t \ge 0 \\ &\Longleftrightarrow \quad \begin{pmatrix} tI & A(x) \\ A(x)^\top & tI \end{pmatrix} \succeq 0 \end{aligned}$$

提纲

数学规划求解器：国内外现状

- 1939年，苏联数学家和经济学家Leonid Kantorovich发明线性规划
- 1979年芝加哥大学Charge 发布Lingo
- 1983年英国爱丁堡大学Ashford创建XPRESS
- 1987年美国莱斯大学Bixby创建CPLEX公司
- 2000年COIN-OR成立，开放源代码CLP和CBC
- 2005年德国ZIB发布了开源整数规划工具SCIP
- 2008年Cplex创始人Bixby离开创办GUROBI
- 2017年，上海交大创建一个开源数学规划工具LEAVES
- 2018年中科院推出CMIP

- GUROBI
- CPLEX
- XPRESS

美国与英国三大求解器巨头，累计三十年研发历史和95%以上市场。线性、整数、非线性模块功能齐全

- SAS: 最大的商业统计软件（北卡）
- CVX: 最著名的求解器建模平台（斯坦福）
- IPOPT: 著名的非线性规划开源求解全局（卡耐基梅隆）
- Coin-OR: 最好的开源线性规划（多组织维护）
- Baron: 最好的非线性规划（多组织维护）
- NEOS: 最大的优化求解免费平台
- SCIP: 最好的整数规划
- CBC: 美国
- SOPLEX: 德国
- MOSEK: 丹麦
- GLPK: 俄罗斯

典型优化算法软件

前面介绍了各种各样的优化问题. 对于每一类优化问题, 我们都有相应的求解算法以及一些流行的算法软件包.

- SDPT3: 这个开源软件包的基本代码是用MATLAB 来写的, 但是关键的子程序是用FORTRAN 和C 语言通过MEX 文件来完成的. 它可以求解锥规划问题, 其中锥可以是半定矩阵锥、二次锥和非负象限中的一个或者多个的乘积. 这个软件主要实现的算法是一种原始–对偶内点法.
- MOSEK: 这个商业软件包可以求解线性规划、二次锥规划、半定规划、二次规划等凸优化问题, 以及混合整数线性规划和混合整数二次规划等. 它的重点是求解大规模稀疏问题, 尤其在求解线性规划、二次锥规划和半定规划的内点法设计上做得非常有效. 除了内点法之外, MOSEK 还实现了线性规划问题的单纯形算法, 特殊网络结构问题的网络单纯形算法以及针对混合整数规划问题的算法. 它提供C, C#, Java, Python, MATLAB 和R 等接口.

典型优化算法软件

- CPLEX: 这个商业软件可以求解整数规划问题，非常大规模的线性规划问题（使用单纯形方法或者内点法），凸和非凸二次规划问题，二次锥规划问题．它提供C++, C#, Java, Microsoft Excel和MATLAB 接口，并且提供一个独立的交互式优化器可执行文件，用于调试和其他目的．
- Gurobi: 这个商业软件可以求解线性规划（单纯形法和并行的内点法），二次规划（采用单纯形法和内点法），二次约束规划，混合整数线性规划，混合整数二次规划，混合整数二次约束规划．它提供C, C++, Java, .NET, Python, MATLAB 和R 等接口．

典型优化算法软件

- IPOPT: 这个开源软件可以求解大规模非线性规划问题，主要实现了原始– 对偶内点法，并使用滤波（filter）方法代替线搜索．IPOPT主要使用C++ 语言编写，并提供C, C++, FORTRAN, Java, MATLAB和R 等接口．
- Knitro: 用来求解大规模非线性优化问题的商业软件．这个软件提供了四种不同的优化方法，两种内点型方法和两种积极集（active set）方法，可以用来求解一般非凸非线性规划问题，非线性方程组，线性规划，二次（线性）约束二次规划问题，线性（非线性）最小二乘问题，混合整数规划问题以及无导数优化问题，等等．Knitro 支持的编程语言有C, FORTRAN, C++, C#, Java, MATLAB, R, Python 等，以及模型语言AMPL，AIMMS，GAMS和MPL 等．因其具有大量的用户友善的选项以及自动调试器，全局优化的并行多重启动策略，导数逼近和检查以及内部预分解器，在实际中被广泛采用．

优化模型语言

- 模型语言的发展开始于19世纪70年代后期，其主要动因是计算机的出现．在优化模型语言中，优化模型可以写成和数学表达式很类似的方式，以此给用户带来更便捷的服务．
- 模型的表达式形式是与求解器无关的，不同的求解器需要用优化模型语言将给定的模型和数据转为其求解的标准形式，然后再对其进行求解．这类工具有三个优点：一是将容易出错的转化步骤交给计算机完成，降低错误率；二是在模型和算法之间建立了一个清晰的界限；三是对于困难的问题，可以尝试不用的求解器，得到更好的结果．

CVX

- CVX是以MATLAB 为基础的优化模型语言，用来求解凸优化问题. 它允许将优化问题的目标函数以及约束用MATLAB 语法写出.
- CVX 采用了一种快速构造和识别凸性的准则，服从这个准则的凸问题都可以很快地被识别出来. 之后CVX 根据用户的选择调用已有软件包来求解变形后的凸优化问题，这些软件包括免费软件SDPT3 和SeDuMi 以及商业软件Gurobi 和MOSEK 等. 除了一些典型问题外，CVX 还可以识别一些更复杂的凸优化问题，例如带ℓ_1范数的优化问题. 目前CVX 还有Julia 语言版本和Python 语言版本CVXPY.
- 除CVX 外，还有很多发展成熟的优化模型语言可供我们使用，如AMPL，YALMIP 等.

CVX

考虑如下优化问题：

$$\begin{aligned} \min \quad & \|Ax - b\|_2, \\ \text{s.t.} \quad & Cx = d, \\ & \|x\|_\infty \le e, \end{aligned} \tag{24}$$

它可以写成:

```
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
```

CVX

- 代码中的前三行是关于A, b, C, d, e的构造. 在调用CVX 求解的时候，对应的代码需要以`cvx_begin` 开始，并且以`cvx_end` 结尾. 在这两行语句之间，我们需要定义
- 要求解的优化问题. 在上面的例子中，`variable x(n)` 表示决策变量x 为n 维空间中的向量. 目标函数$\|Ax - b\|_2$ 则用`norm( A * x - b, 2 )` 来表示，`minimize` 表示求解目标函数的极小值. 最后以`subject to` 开始描述问题的约束，`C * x == d` 和`norm( x, Inf ) <= e` 分别表示约束$Cx = d$ 和$\|x\|_\infty \leq e$.
- 执行上述代码，CVX 会选取默认的凸优化问题算法来返回上面问题的解.

AMPL

- AMPL（a mathematical programming language）是用来描述高复杂度的大规模优化问题的模型语言.
- 几十个求解器支持AMPL，包含开源软件和商业软件，例如CBC, CPLEX, FortMP, Gurobi, MINOS, IPOPT, SNOPT, Knitro 和LGO 等，因此可以支持一大类问题的求解. 它的一个优点是其语法与数学表达式非常类似，因此可对优化问题进行非常简洁并且可读的定义.
- 对于优化模型语言，常用的还有YALMIP . 对于半定规划问题，最新的一些软件包有SDPNAL，SDPNAL+ 和SSNSDP . 对于流形优化，软件包有OptM ，Manopt 和ARNT.

最优性理论

陈士祥

中国科学技术大学

致谢：本教案部分参考北京大学文再文教授《最优化与建模》

提纲

最优化问题解的存在性

考虑优化问题

$$\begin{aligned} \min_{x\in\mathbb{R}^n} \quad & f(x), \\ \text{s.t.} \quad & x \in \mathcal{X}, \end{aligned}$$

其中$\mathcal{X} \subseteq \mathbb{R}^n$ 为可行域.

- 首先要考虑的是最优解的存在性, 然后考虑如何求出其最优解.
- 在数学分析课程中, 我们学习过Weierstrass 定理, 即定义在紧集上的连续函数一定存在最大(最小) 值点.
- 而在许多实际问题中, 定义域可能不是紧的, 目标函数也不一定连续, 因此需要将此定理推广来保证最优化问题解的存在性.

最优化问题解的存在性: 推广的Weierstrass 定理

定理 (推广的Weierstrass定理)

若函数$f:\mathcal{X}\to(-\infty,+\infty]$ 适当且闭, 且以下条件中任意一个成立:

1. $\mathrm{dom} f \overset{\text{def}}{=\!=} \{x\in\mathcal{X}: f(x)<+\infty\}$ 是有界的;
2. 存在一个常数$\bar{\gamma}$ 使得下水平集

$$C_{\bar{\gamma}} \overset{\text{def}}{=\!=} \{x\in\mathcal{X}: f(x)\le\bar{\gamma}\}$$

是非空且有界的;

3. f 是强制的, 即对于任一满足$\|x^k\|\to+\infty$ 的点列$\{x^k\}\subset\mathcal{X}$, 都有

$$\lim_{k\to\infty} f(x^k)=+\infty,$$

则函数f 的最小值点集$\{x\in\mathcal{X}\mid f(x)\le f(y),\ \forall y\in\mathcal{X}\}$ 非空且紧.

条件(2) 下的证明

假设条件(2) 成立, 且$t \overset{\text{def}}{=\!=} \inf_{x\in\mathcal{X}} f(x) = -\infty$.

- 由下确界的定义, 存在点列$\{x^k\}_{k=1}^{\infty} \subset C_{\bar{\gamma}}$, 使得$\lim\limits_{k\to\infty} f(x^k) = -\infty$.
- 由$C_{\bar{\gamma}}$ 有界知点列$\{x^k\}$ 存在聚点x^*.
- 由f 是闭函数知$\mathbf{epi} f$ 为闭集, 因此$(x^*, t) \in \mathbf{epi} f$. 根据上方图的定义知$f(x^*) \le t = -\infty$, 这与$f$ 适当矛盾, 故$t \overset{\text{def}}{=\!=} \inf_{x\in\mathcal{X}} f(x)$ 有限.
- 由f 为闭函数知$C_{\bar{\gamma}}$ 为闭集, 由假设知$C_{\bar{\gamma}}$ 有界, 故为紧集.
- 由$f(x^*) = t$ 知最小值点集非空, 且为紧集$C_{\bar{\gamma}}$ 的子集, 而紧集的子集也是紧集, 故最小值点集为非空紧集.

条件(1)(3)下的证明

假设条件(1)成立, 则$\mathrm{dom} f$ 是有界的.

- 由f 适当知存在$x_0 \in \mathcal{X}$ 使得$f(x_0) < +\infty$.
- 记$\bar{\gamma} = f(x_0)$, 此时f 的$\bar{\gamma}$ 下水平集$C_{\bar{\gamma}}$ 非空有界, 条件(2)成立.

假设条件(3)成立, 我们证明条件(2)成立.

- 用反证法, 假设存在一个无界的下水平集$C_{\bar{\gamma}}$, 那么可以取点列$\{x^k\} \subset C_{\bar{\gamma}}$ 使得$\lim\limits_{k\to\infty} \|x^k\| = +\infty$, $\lim\limits_{k\to\infty} f(x^k) = +\infty$, 这与$f(x^k) \leq \bar{\gamma}$ 矛盾, 故此时f 的任意下水平集都有界, 条件(2)成立.

推广的Weierstrass定理:注记

- 推广的Weierstrass定理的三个条件在本质上都是保证$f(x)$的最小值不能在无穷远处取到.
- 因此我们可以仅在一个有界的下水平集中考虑$f(x)$的最小值.
- 定理仅要求$f(x)$为适当且闭的函数, 并不需要$f(x)$的连续性, 因此比数学分析中的Weierstrass定理应用范围更广.
- 当定义域不是有界闭集时, 对于强制函数$f(x) = x^2$, $x \in \mathcal{X} = \mathbb{R}$, 其全局最优解一定存在.
- 对于适当且闭的函数$f(x) = e^{-x}$, $x \in \mathcal{X} = \mathbb{R}$, 它不满足三个条件中任意一个, 因此我们不能断言其全局极小值点存在.事实上, 其全局极小值点不存在.

解的存在唯一性

- 推广的Weierstrass定理给出了最优解的存在性条件, 但其对应的解可能不止一个.
- 当最优解是唯一存在时, 我们可以通过比较不同算法收敛至该解的速度来判断算法好坏.
- 但是如果问题有多个最优值点, 不同的算法收敛到的最优值点可能不同, 那么这些算法收敛速度的比较就失去了参考价值.
- 因此, 最优化问题解的唯一性在理论分析和算法比较中扮演着重要角色.

定义 (拟强凸函数)

给定凸集$\mathcal{X}$ 和函数$f\colon \mathcal{X} \to (-\infty, +\infty]$.若任取$x \neq y$ 和$\lambda \in (0,1)$, 有

$$f(\lambda x + (1-\lambda)y) < \max\{f(x), f(y)\},$$

那么我们称函数f 是强拟凸的.

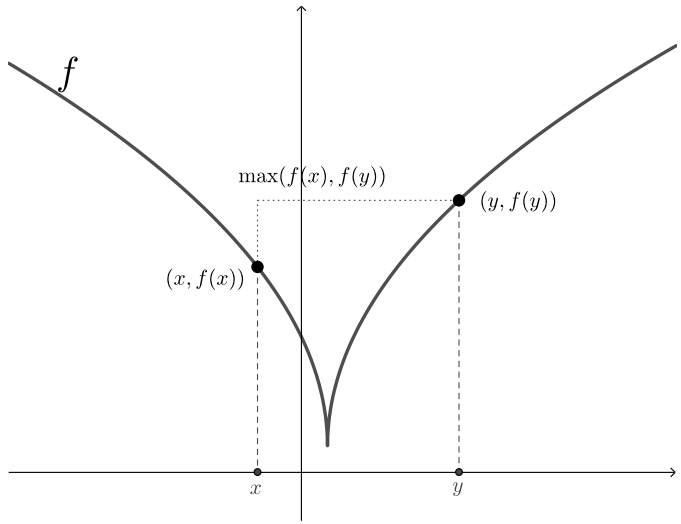


Figure: 一个强拟凸函数

拟强凸函数

- 强拟凸函数不一定是凸函数, 但其任意一个下水平集都是凸集, 并可以包含一部分性质较好的非凸函数.
- 任意强凸函数均为强拟凸的, 但凸函数并不一定是强拟凸的.
- 任何定义在闭有界凸集上的强凸函数(如$f(x) = x^2$), 其最优解都是唯一存在的.
- 对于一般的凸函数, 其最优解可能不唯一, 比如$f(x) = \max\{x, 0\}$, 任意$x \leq 0$ 都是$f(x)$的最优解.

唯一性定理

定理 (唯一性定理)

设$\mathcal{X}$ 是$\mathbb{R}^n$ 的一个非空, 紧且凸的子集, 如果$f:\mathcal{X}\to(-\infty,+\infty]$ 是适当, 闭且强拟凸函数, 那么存在唯一的x^* 满足

$$f(x^*)<f(x),\quad \forall x\in\mathcal{X}\backslash\{x^*\}.$$

证明:由Weierstrass定理知f 至少存在一个全局极小点x^*.若x^*,y^* 皆为全局极小点, 则有:

$$f(\lambda x^*+(1-\lambda)y^*)<\max\{f(x^*),f(y^*)\}=f(x^*),\quad \forall\lambda\in(0,1).$$

这与x^* 的全局极小性矛盾.

提纲

无约束可微问题的最优性理论:引言

无约束可微优化问题通常表示为如下形式：

$$\min_{x\in\mathbb{R}^n} \quad f(x),$$

其中f 是连续可微函数.

- 给定一个点$\bar{x}$, 我们想要知道这个点是否是函数f 的一个局部极小解或者全局极小解.
- 如果从定义出发, 需要对其邻域内的所有点进行判断, 这不可行.
- 因此, 需要一个更简单的方式来验证一个点是否为极小值点.我们称其为最优性条件, 它主要包含一阶最优性条件和二阶最优性条件.

一阶必要条件:下降方向

定义 (下降方向)

对于可微函数f 和点$x \in \mathbb{R}^n$, 如果存在向量d 满足

$$\nabla f(x)^{\mathrm{T}} d < 0,$$

那么称d 为f 在点x 处的一个下降方向.

- 一阶最优性条件是利用梯度(一阶)信息来判断给定点的最优性.
- 由下降方向的定义, 容易验证: 如果f 在点x 处存在一个下降方向d, 那么对于任意的$T > 0$, 存在$t \in (0, T]$, 使得

  $$f(x + td) < f(x).$$

  因此, 在局部最优点处不能有下降方向.

定理 (一阶必要条件)

假设f 在全空间$\mathbb{R}^n$ 可微.如果x^* 是一个局部极小点, 那么

$$\nabla f(x^*) = 0.$$

证明:任取$v \in \mathbb{R}^n$, 考虑f 在点$x = x^*$ 处的泰勒展开

$$f(x^* + tv) = f(x^*) + tv^{\mathrm{T}}\nabla f(x^*) + o(t),$$

整理得

$$\frac{f(x^* + tv) - f(x^*)}{t} = v^{\mathrm{T}}\nabla f(x^*) + o(1).$$

根据x^* 的最优性, 在上式中分别对t 取点0 处的左, 右极限可知

$$\lim_{t\to 0^+} \frac{f(x^* + tv) - f(x^*)}{t} = v^{\mathrm{T}}\nabla f(x^*) \geq 0,$$

$$\lim_{t\to 0^-} \frac{f(x^* + tv) - f(x^*)}{t} = v^{\mathrm{T}}\nabla f(x^*) \leq 0,$$

即对任意的v 有$v^{\mathrm{T}}\nabla f(x^*) = 0$, 由$v$ 的任意性知$\nabla f(x^*) = 0$.

二阶最优性条件

- 在没有额外假设时, 如果一阶必要条件满足, 我们仍然不能确定当前点是否是一个局部极小点.
- 假设f 在点x 的一个开邻域内是二阶连续可微的.类似于一阶必要条件的推导, 可以借助当前点处的二阶泰勒展开来逼近该函数在该点附近的取值情况, 从而来判断最优性.
- 在点x 附近我们考虑泰勒展开

$$f(x+d) = f(x) + \nabla f(x)^{\mathrm{T}} d + \frac{1}{2} d^{\mathrm{T}} \nabla^2 f(x) d + o(\|d\|^2).$$

- 当一阶必要条件满足时, $\nabla f(x) = 0$, 那么上面的展开式简化为

$$f(x+d) = f(x) + \frac{1}{2} d^{\mathrm{T}} \nabla^2 f(x) d + o(\|d\|^2).$$

由此, 我们可以导出二阶最优性条件.

二阶最优性条件

定理 (二阶最优性条件)

必要条件:若x^* 是f 的一个局部极小点, 则$\nabla f(x^*) = 0, \nabla^2 f(x^*) \succeq 0$.

充分条件:若$\nabla f(x^*) = 0, \nabla^2 f(x^*) \succ 0$, 则$x^*$ 是f 的一个局部极小点.

- 必要性:若$\nabla^2 f(x^*)$ 有负的特征值$\lambda_- < 0$, 设$\nabla^2 f(x^*)d = \lambda_- d$, 则

$$\frac{f(x^*+d)-f(x^*)}{\|d\|^2} = \frac{1}{2}\frac{d^{\mathrm{T}}}{\|d\|}\nabla^2 f(x^*)\frac{d}{\|d\|} + o(1) = \frac{1}{2}\lambda_- + o(1).$$

  当$\|d\|$ 充分小时,$f(x^*+d) < f(x^*)$, 这和点x^* 的最优性矛盾.

- 充分性:由$\nabla f(x^*) = 0$ 时的二阶展开,

$$\frac{f(x^*+d)-f(x^*)}{\|d\|^2} = \frac{d^{\mathrm{T}}\nabla^2 f(x^*)d + o(\|d\|^2)}{\|d\|^2} \geq \frac{1}{2}\lambda_{\min} + o(1).$$

  当$\|d\|$ 充分小时有$f(x^*+d) \geq f(x^*)$, 即二阶充分条件成立.

二阶最优性条件:注记

- 若点$\bar{x}$满足一阶最优性条件(即$\nabla f(\bar{x}) = 0$), 则称$\bar{x}$是一个一阶稳定点。若该点处的海瑟矩阵$\nabla^2 f(\bar{x})$不是半正定的, 那么$\bar{x}$不是一个局部极小点.
- 进一步地, 如果海瑟矩阵$\nabla^2 f(\bar{x})$既有正特征值又有负特征值, 我们称稳定点$\bar{x}$为一个鞍点.
- 鞍点: 事实上, 记d_1, d_2为其正负特征值对应的特征向量, 那么对于任意充分小的$t > 0$, 我们都有$f(\bar{x} + td_1) > f(\bar{x})$且$f(\bar{x} + td_2) < f(\bar{x})$.

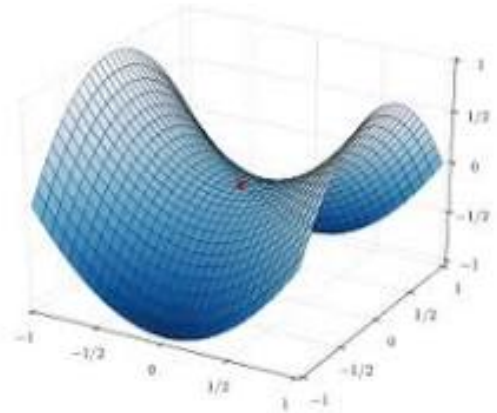


Figure: 鞍点图示

- 注意, 二阶最优性条件给出的仍然是关于局部最优性的判断.对于给定点的全局最优性判断, 我们还需要借助实际问题的性质, 比如目标函数是凸的、非线性最小二乘问题中目标函数值为0 等.
- 注意到，凸问题的最优点x^*满足的一阶充分必要条件是$\nabla f(x^*) = 0$. 这是因为由一阶判定条件$f(y) \geq f(x^*) + \nabla f(x^*)^\top (y - x^*)$.
- 例1：线性最小二乘:

$$\min_{x \in \mathbb{R}^n} \quad f(x) \overset{\text{def}}{=\!=} \frac{1}{2}\|b - Ax\|_2^2,$$

其中$A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$.由f 可微且凸知

$$x^* \text{ 为一个全局最优解} \Leftrightarrow \nabla f(x^*) = A^{\mathrm{T}}(Ax^* - b) = 0.$$

- 例2：$f(x) = x^3$ 的一阶稳定点$x^* = 0$不是局部最优。
- 例3：$f(x) = x_1^3 + x_2^2$, 我们有$\nabla f(x) = (3x_1^2, 2x_2)^\top$, $\nabla^2 f(x) = \begin{pmatrix} 6x_1 & 0 \\ 0 & 2 \end{pmatrix}$. $x = (0,0)^\top$满足二阶必要性条件，但并非局部极小。这是因为$f(-\epsilon, 0) = -3\epsilon^3 < 0$.

提纲

对偶理论: 一般的约束优化问题

$$\begin{aligned} \min_{x\in\mathbb{R}^n} \quad & f(x), \\ \text{s.t.} \quad & c_i(x) \le 0,\ i \in \mathcal{I}, \\ & c_i(x) = 0,\ i \in \mathcal{E}, \end{aligned}$$

其中c_i为定义在$\mathbb{R}^n$或其子集上的实值函数, $\mathcal{I}$和$\mathcal{E}$分别表示不等式约束(inequality)和等式约束(equality)对应的下标集合且各下标互不相同.

- 这个问题的可行域定义为

$$\mathcal{X} = \{x \in \mathbb{R}^n \mid c_i(x) \le 0,\ i \in \mathcal{I} \text{ 且 } c_i(x) = 0,\ i \in \mathcal{E}\}.$$

- 通过将$\mathcal{X}$的示性函数加到目标函数中可以得到无约束优化问题.但是转化后问题的目标函数是不连续的、不可微的以及不是有限的, 这导致我们难以分析其理论性质以及设计有效的算法.

拉格朗日函数

一般的约束优化问题:

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & c_i(x) \le 0,\ i \in \mathcal{I},\ |\mathcal{I}| = m \\ & c_i(x) = 0,\ i \in \mathcal{E},\ |\mathcal{E}| = p \end{aligned}$$

变量$x \in \mathbb{R}^n$, 最优值为p^*, 原问题的定义域为

$$\mathcal{X} = \{x \in \mathbb{R}^n \mid c_i(x) \le 0,\ i \in \mathcal{I} \text{ 且 } c_i(x) = 0,\ i \in \mathcal{E}\}$$

拉格朗日函数$L : \mathbb{R}^n \times \mathbb{R}^m_+ \times \mathbb{R}^p \to \mathbb{R}$

$$L(x, \lambda, \nu) = f(x) + \sum_{i \in \mathcal{I}} \lambda_i c_i(x) + \sum_{i \in \mathcal{E}} \nu_i c_i(x)$$

- $\lambda_i \ge 0$ 为第i 个不等式约束对应的拉格朗日乘子
- $\nu_i \in \mathbb{R}$ 为第i 个等式约束对应的拉格朗日乘子

拉格朗日对偶函数

拉格朗日对偶函数$g:\mathbb{R}^m_+ \times \mathbb{R}^p \to [-\infty,+\infty)$

$$\begin{aligned} g(\lambda,\nu) &= \inf_{x\in\mathbb{R}^n} L(x,\lambda,\nu) \\ &= \inf_{x\in\mathbb{R}^n} \left(f(x) + \sum_{i\in\mathcal{I}} \lambda_i c_i(x) + \sum_{i\in\mathcal{E}} \nu_i c_i(x) \right) \end{aligned}$$

定理 (弱对偶原理)

若$\lambda \geq 0$, 则$g(\lambda,\nu) \leq p^*$.

证明:若$\tilde{x} \in \mathcal{X}$, 则

$$g(\lambda,\nu) = \inf_x L(x,\lambda,\nu) \leq L(\tilde{x},\lambda,\nu) \leq f(\tilde{x}),$$

对右边$f(\tilde{x})$ 取下界得

$$g(\lambda,\nu) \leq \inf_{\tilde{x}\in\mathcal{X}} f(\tilde{x}) = p^*.$$

拉格朗日对偶问题

拉格朗日对偶问题(Lagrangian dual problem):

$$\max_{\lambda\geq 0,\nu} \; g(\lambda,\nu) = \max_{\lambda\geq 0,\nu} \inf_{x\in\mathbb{R}^n} \; L(x,\lambda,\nu)$$

- 称λ 和ν 为对偶变量, 设最优值为q^*
- q^* 为p^* 的最优下界, 称$p^* - q^*$ 为对偶间隙(duality gap)
- 拉格朗日对偶问题是一个凸优化问题(这是因为对偶问题是一个最大化凹函数问题)
- $\mathrm{dom} g = \{(\lambda,\nu) \mid \lambda \geq 0, g(\lambda,\nu) > -\infty\}$, 称其元素为对偶可行解

例: 标准形式线性规划及其对偶

$$\begin{array}{lll} \min & c^T x & \\ \text{s.t.} & Ax = b & \\ & x \geq 0 & \end{array} \qquad\qquad \begin{array}{lll} \max & -b^T\nu & \\ \text{s.t.} & A^T\nu + c \geq 0 & \end{array}$$

实例:线性方程组具有最小模的解

$$\begin{aligned} \min \quad & x^T x \\ \text{s.t.} \quad & Ax = b \end{aligned}$$

对偶函数

- 拉格朗日函数为$L(x,\nu) = x^T x + \nu^T(Ax - b)$
- 求L 关于x 的最小值, 由一阶条件:

$$\nabla_x L(x,\nu) = 2x + A^T\nu = 0 \quad \Longrightarrow \quad x = -(1/2)A^T\nu$$

- 将上式代入L 得到对偶函数g:

$$g(\nu) = L((-1/2)A^T\nu, \nu) = -\frac{1}{4}\nu^T AA^T\nu - b^T\nu$$

它是关于ν 的凹函数

弱对偶性: $p^* \geq -(1/4)\nu^T AA^T\nu - b^T\nu, \ \forall\nu$

线性规划问题的对偶

$$\begin{aligned} \min_{x} \quad & c^{\mathrm{T}}x \\ \text{s.t.} \quad & Ax = b \\ & x \geq 0 \end{aligned}$$

- 拉格朗日函数:

$$L(x, s, \nu) = c^{\mathrm{T}}x + \nu^{\mathrm{T}}(Ax - b) - s^{\mathrm{T}}x = -b^{\mathrm{T}}\nu + (A^{\mathrm{T}}\nu - s + c)^{\mathrm{T}}x$$

- 对偶函数:

$$g(s, \nu) = \inf_{x} L(x, s, \nu) = \begin{cases} -b^{\mathrm{T}}\nu, & A^{\mathrm{T}}\nu - s + c = 0 \\ -\infty, & \text{其他} \end{cases}$$

- 对偶问题:

$$\begin{aligned} \max_{s,\nu} \quad & -b^{\mathrm{T}}\nu, \\ \text{s.t.} \quad & A^{\mathrm{T}}\nu - s + c = 0, \\ & s \geq 0. \end{aligned} \quad \overset{y=-\nu}{\Longleftrightarrow} \quad \begin{aligned} \max_{s,y} \quad & b^{\mathrm{T}}y, \\ \text{s.t.} \quad & A^{\mathrm{T}}y + s = c, \\ & s \geq 0. \end{aligned}$$

线性规划问题的对偶

- 值得注意的是，也可保留约束$x \geq 0$定义拉格朗日函数：

$$L(x,y) = c^{\mathrm{T}}x - y^{\mathrm{T}}(Ax-b) = b^{\mathrm{T}}y + (c - A^{\mathrm{T}}y)^{\mathrm{T}}x, \quad x \geq 0.$$

- 对偶问题需要将$x \geq 0$ 添加到约束里：

$$\max_y \left\{ \inf_x \ b^{\mathrm{T}}y + (c - A^{\mathrm{T}}y)^{\mathrm{T}}x, \quad \text{s.t.} \ \ x \geq 0 \right\} \Rightarrow \begin{array}{ll} \max\limits_y & b^{\mathrm{T}}y, \\ \text{s.t.} & A^{\mathrm{T}}y \leq c. \end{array}$$

此对偶问题可以通过将上页最后一个问题中的变量s 消去得到.

线性规划问题对偶的对偶

- 视$\max\ b^{\mathrm{T}}y$为$\min\ -b^{\mathrm{T}}y$, 对偶问题的拉格朗日函数为

$$L(y,x) = -b^{\mathrm{T}}y + x^{\mathrm{T}}(A^{\mathrm{T}}y - c) = -c^{\mathrm{T}}x + (Ax-b)^{\mathrm{T}}y.$$

- 因此得到对偶函数

$$g(x) = \inf_y\ L(y,x) = \begin{cases} -c^{\mathrm{T}}x, & Ax = b, \\ -\infty, & \text{其他}. \end{cases}$$

- 相应的对偶问题是

$$\begin{aligned} \max_x \quad & -c^{\mathrm{T}}x, \\ \text{s.t.} \quad & Ax = b, \\ & x \geq 0. \end{aligned}$$

该问题与原始问题完全等价, 这表明线性规划问题与其对偶问题互为对偶.

线性规划问题的对偶理论

强对偶性: 若一个线性规划问题有最优解, 则其对偶问题有最优解, 且最优值相等.

对偶问题 \ 原问题	finite	unbounded	infeasible
finite	√	×	×
unbounded	×	×	√
infeasible	×	√	√

- 若$p^* = -\infty$, 原问题可行，但是无界，则$d^* \leq p^* = -\infty$, 因此对偶问题不适定
- 若$d^* = +\infty$, 则$+\infty = d^* \leq p^*$, 因此原问题不适定
- 若$p^* = +\infty$, 则原问题无可行解，此时对偶问题有可能无界或无可行解。

$$\begin{array}{lll} \min & x_1 + 2x_2 & \\ \text{s.t.} & x_1 + x_2 = 1 & \\ & 2x_1 + 2x_2 = 3 & \end{array} \qquad \begin{array}{lll} \max & p_1 + 3p_2 & \\ \text{s.t.} & p_1 + 2p_2 = 1 & \\ & p_1 + 2p_2 = 2 & \end{array}$$

实例:等式约束下的范数最小化

$$\begin{aligned} \min \quad & \|x\| \\ \text{s.t.} \quad & Ax = b \end{aligned}$$

对偶函数

$$g(\nu) = \inf_x(\|x\| - \nu^T Ax + b^T\nu) = \begin{cases} b^T\nu & \|A^T\nu\|_* \le 1 \\ -\infty & \text{其他} \end{cases}$$

其中$\|v\|_* = \sup_{\|u\|\le 1} u^T v$ 是$\|\cdot\|$ 的对偶范数

证明: 利用$\inf_x(\|x\| - y^T x)$ 在$\|y\|_* \le 1$ 时等于0 否则等于$-\infty$

- 若$\|y\|_* \le 1$, 则$x - y^T x \ge 0$ 对任意x 都成立, 当$x = 0$ 时取等
- 若$\|y\|_* > 1$, 取$x = tu$, 其中$\|u\| \le 1$, $u^T y = \|y\|_* > 1$:

$$\|x\| - y^T x = t(\|u\| - \|y\|_*) \to -\infty \text{ 当} t \to \infty$$

弱对偶性: $p^* \ge b^T\nu$ 若$\|A^T\nu\|_* \le 1$

拉格朗日对偶与共轭函数

$$\begin{aligned} \min \quad & f_0(x) \\ \text{s.t.} \quad & Ax \le b, \quad Cx = d \end{aligned}$$

对偶函数

$$\begin{aligned} g(\lambda,\nu) &= \inf_{x\in \mathrm{dom}\, f_0} \left(f_0(x) + (A^T\lambda + C^T\nu)^T x - b^T\lambda - d^T\nu\right) \\ &= -f_0^*(-A^T\lambda - C^T\nu) - b^T\lambda - d^T\nu \end{aligned}$$

- 回顾共轭函数的定义$f^*(y) = \sup_{x\in \mathrm{dom}\, f}(y^T x - f(x))$
- 在f_0 的共轭函数已知时可以简化对偶函数的推导

例：最大化熵

$$f_0(x) = \sum_{i=1}^n x_i \log x_i, \qquad f_0^*(y) = \sum_{i=1}^n e^{y_i - 1}$$

实例: 二路划分

$$\begin{aligned} \min \quad & x^T W x \\ \text{s.t.} \quad & x_i^2 = 1, \quad i = 1, \ldots, n \end{aligned}$$

- 非凸问题; 可行域由2^n 个离散点组成
- 含义: 将$\{1, \ldots, n\}$ 划分为两个集合; W_{ij} 是将i, j 划入同一集合的代价; $-W_{ij}$ 则是将i, j 划入不同集合的代价

对偶函数

$$g(\nu) = \inf_x (x^T W x + \sum_i \nu_i (x_i^2 - 1)) = \inf_x x^T (W + \mathrm{diag}(\nu)) x - \mathbf{1}^T \nu$$

$$= \begin{cases} -\mathbf{1}^T \nu & W + \mathrm{diag}(\nu) \succeq 0 \\ -\infty & \text{其他} \end{cases}$$

弱对偶性: $p^* \geq -\mathbf{1}^T \nu$ 若$W + \mathrm{diag}(\nu) \succeq 0$
例: 取$\nu = -\lambda_{\min}(W)\mathbf{1}$ 即得下界的估计, $p^* \geq n\lambda_{\min}(W)$

弱对偶性与强对偶性

弱对偶性: $d^* \leq p^*$

- 对任何问题都成立：不论是凸问题与非凸问题
- 可导出复杂问题的非平凡下界, 例如, SDP问题

$$\begin{aligned} \max \quad & -\mathbf{1}^T \nu \\ \text{s.t.} \quad & W + \operatorname{diag}(\nu) \succeq 0 \end{aligned}$$

给出了二路划分问题的一个下界:

$$\min\ x^T W x, \quad \text{s.t. } x_i^2 = 1, \quad i = 1, \ldots, n$$

强对偶性: $d^* = p^*$

- 对一般问题而言通常不成立
- (通常) 对凸问题成立
- 称保证凸问题强对偶性成立的条件为约束品性条件(constraint qualification condition)

问题形式与对偶性

- 一个问题不同等价形式的对偶可能差异巨大
- 当对偶问题难以推导或没有价值时, 可以尝试改写原问题的形式

常用的改写技巧

- 引入新变量与等式约束
- 将显式约束隐式化或将隐式约束显式化

例：引入新变量与等式约束

$$\min \quad f_0(Ax+b)$$

改写原问题及其对偶

$$\begin{array}{ll} \min & f_0(y) \\ \text{s.t.} & Ax+b-y=0 \end{array} \qquad \begin{array}{ll} \max & b^T\nu - f_0^*(\nu) \\ \text{s.t.} & A^T\nu = 0 \end{array}$$

对偶函数为

$$\begin{aligned} g(\nu) &= \inf_{x,y}(f_0(y) - \nu^T y + \nu^T Ax + b^T\nu) \\ &= \begin{cases} -f_0^*(\nu) + b^T\nu & A^T\nu = 0 \\ -\infty & \text{其他} \end{cases} \end{aligned}$$

范数逼近问题: $\min \|Ax-b\|$

$$\begin{aligned} \min \quad & \|y\| \\ \text{s.t.} \quad & y = Ax - b \end{aligned}$$

由 $\|\cdot\|$ 的共轭函数知其对偶函数为:

$$\begin{aligned} g(\nu) &= \inf_{x,y}(\|y\| + \nu^T y - \nu^T A x + b^T \nu) \\ &= \begin{cases} b^T\nu + \inf_y(\|y\| + \nu^T y) & A^T\nu = 0 \\ -\infty & \text{其他} \end{cases} \\ &= \begin{cases} b^T\nu & A^T\nu = 0, \quad \|\nu\|_* \le 1 \\ -\infty & \text{其他} \end{cases} \end{aligned}$$

范数逼近问题的对偶

$$\begin{aligned} \max \quad & b^T\nu \\ \text{s.t.} \quad & A^T\nu = 0, \quad \|\nu\|_* \le 1 \end{aligned}$$

ℓ_1 正则化问题的对偶

$$\min_{x\in\mathbb{R}^n} \quad \frac{1}{2}\|Ax-b\|^2+\mu\|x\|_1$$

令$r=Ax-b$, 问题等价于$\min_{x\in\mathbb{R}^n}\ \frac{1}{2}\|r\|^2+\mu\|x\|_1,\ \ \text{s.t. } r=Ax-b$

- 拉格朗日函数:

$$\begin{aligned} L(x,r,\lambda) &= \frac{1}{2}\|r\|^2+\mu\|x\|_1-\langle\lambda,Ax-b-r\rangle \\ &= \frac{1}{2}\|r\|^2+\lambda^{\mathrm{T}}r+\mu\|x\|_1-(A^{\mathrm{T}}\lambda)^{\mathrm{T}}x+b^{\mathrm{T}}\lambda \end{aligned}$$

- 对偶函数:

$$g(\lambda)=\inf_{x,r}\ L(x,r,\lambda)=\begin{cases} b^{\mathrm{T}}\lambda-\frac{1}{2}\|\lambda\|^2, & \|A^{\mathrm{T}}\lambda\|_\infty\le\mu \\ -\infty, & \text{其他} \end{cases}$$

- 对偶问题:

$$\max \quad b^{\mathrm{T}}\lambda-\frac{1}{2}\|\lambda\|^2, \quad \text{s.t.} \quad \|A^{\mathrm{T}}\lambda\|_\infty\le\mu$$

例：显示变为隐式约束

带边界约束的线性规划: 原问题与对偶问题

$$\begin{array}{ll} \min & c^T x \\ \text{s.t.} & Ax = b \\ & -\mathbf{1} \le x \le \mathbf{1} \end{array} \qquad \begin{array}{ll} \max & -b^T\nu - \mathbf{1}^T\lambda_1 - \mathbf{1}^T\lambda_2 \\ \text{s.t.} & c + A^T\nu + \lambda_1 - \lambda_2 = 0 \\ & \lambda_1 \ge 0, \quad \lambda_2 \ge 0 \end{array}$$

通过隐式化边界约束改写原问题

$$\begin{array}{ll} \min & f_0(x) = \begin{cases} c^T x & -\mathbf{1} \le x \le \mathbf{1} \\ -\infty & \text{其他} \end{cases} \\ \text{s.t.} & Ax = b \end{array}$$

对偶函数为

$$\begin{aligned} g(\nu) &= \inf_{-\mathbf{1} \le x \le \mathbf{1}} (c^T x + \nu^T (Ax - b)) \\ &= -b^T\nu - \|A^T\nu + c\|_1 \end{aligned}$$

对偶问题: $\max \ -b^T\nu - \|A^T\nu + c\|_1$

带广义不等式约束优化问题

- 问题

$$\begin{aligned} \min_{x\in\mathbb{R}^n} \quad & f(x), \\ \text{s.t.} \quad & c_i(x) \leq 0,\ i \in \mathcal{I}, \\ & c_i(x) = 0,\ i \in \mathcal{E}, \end{aligned}$$

  中的不等式约束$c_i(x),\ i \in \mathcal{I}$都是实值函数的形式.

- 在许多实际应用中, 我们还会遇到大量带广义不等式约束的优化问题, 例如自变量x可能取值于半正定矩阵空间中. 例如我们刚刚遇到的二路划分的对偶问题。

- 对于这类约束我们不易将其化为$c_i(x) \leq 0$的形式, 此时又该如何构造拉格朗日对偶函数呢?

适当锥与广义不等式

回顾适当锥与偏序关系：

定义 (适当锥)

称满足如下条件的锥K为适当锥*(proper cone)*：

1. K是凸锥；
2. K是闭集；
3. K是实心的*(solid)*, 即$\mathbf{int}\, K \neq \emptyset$；
4. K是尖的*(pointed)*, 即对任意非零向量x, 若$x \in K$, 则$-x \notin K$, 也即K中无法容纳直线.

- 适当锥K可以诱导出广义不等式, 它定义了全空间上的偏序关系：

$$x \preceq_K y \iff y - x \in K.$$

- 类似地, 可以定义严格广义不等式：

$$x \prec_K y \iff y - x \in \mathbf{int}\, K.$$

- 当$K = \mathcal{S}_+^n$时, $X \preceq_K Y$表示$Y - X \succeq 0$, 即$Y - X$是半正定矩阵

对偶锥与拉格朗日乘子

- 对广义不等式, 该如何构造广义不等式约束所对应的乘子?

定义 (对偶锥)

令K 为全空间Ω 的子集, 称集合

$$K^* = \{y \in \Omega \mid \langle x, y\rangle \geq 0,\ \forall x \in K\}$$

为其对偶锥.

- 如果$K = \mathbb{R}^n_+, \Omega = \mathbb{R}^n$ 并且定义$\langle x, y\rangle = x^{\mathrm{T}}y$, 那么易知$K^* = \mathbb{R}^n_+$. 故一般情况下，我们有

$$-c_i(x) \in K, \quad \lambda_i \in K^*.$$

- 我们可以将此推广到一般偏序情况. 直观来说, 对偶锥K^* 中向量和原锥K 中向量的内积恒非负, 这一性质可以被用来构造拉格朗日对偶函数.

广义不等式约束

- 凸问题的广义不等式约束

$$\begin{array}{ll} \min & f_0(x) \\ \text{s.t.} & f_i(x) \preceq_{K_i} 0, \quad i = 1, \ldots, m \\ & Ax = b \end{array}$$

  - $f_0 : \mathbb{R}^n \to \mathbb{R}$ 为凸函数; $f_i : \mathbb{R}^n \to \mathbb{R}^{k_i}$ 关于适当锥K_i为K_i-凸的.
  - 与标准凸问题有相同的性质

- 锥形式问题(具有仿射目标函数与约束的特殊情况)

$$\begin{array}{ll} \min & c^T x \\ \text{s.t.} & Fx + g \preceq_K 0 \\ & Ax = b \end{array}$$

将线性规划问题延伸到了非多面体锥上($K = \mathbb{R}^m_+$)

广义不等式约束优化问题拉格朗日函数的构造

- 广义不等式约束优化问题:

$$\begin{aligned} \min_{x\in\mathbb{R}^n} \quad & f(x) \\ \text{s.t.} \quad & c_i(x) \preceq_{K_i} 0,\ i\in\mathcal{I} \\ & c_i(x) = 0,\ i\in\mathcal{E} \end{aligned}$$

其中$c_i:\mathbb{R}^n\to\mathbb{R}^{k_i},\ k_i\in\mathbb{N}_+, i\in\mathcal{I}$为向量值函数,$f$与$c_i, i\in\mathcal{E}$为实值函数,$K_i, i\in\mathcal{I}$为适当锥.

- 拉格朗日函数L:

$$L(x,\lambda,\nu) = f(x) + \sum_{i\in\mathcal{I}} \langle c_i(x),\lambda_i\rangle + \sum_{i\in\mathcal{E}} \nu_i c_i(x), \quad \lambda_i\in K_i^*, \nu_i\in\mathbb{R}.$$

- 容易验证$L(x,\lambda,\nu)\le f(x),\ \forall\, x\in\mathcal{X},\ \lambda_i\in K_i^*, \nu_i\in\mathbb{R}.$

- 对偶函数$g(\lambda,\nu) = \inf_{x\in\mathbb{R}^n}\ L(x,\lambda,\nu)$, 对偶问题为

$$\max_{\lambda_i\in K_i^*,\ \nu_i\in\mathbb{R}} \quad g(\lambda,\nu).$$

例：半定规划问题的对偶问题

半正定规划问题具有如下标准形式，是线性规划问题的推广：

$$\begin{aligned} \min_{X\in\mathcal{S}^n} \quad & \langle C, X\rangle \\ \text{s.t.} \quad & \langle A_i, X\rangle = b_i,\ i = 1, 2, \cdots, m \\ & X \succeq 0 \end{aligned}$$

其中$A_i \in \mathcal{S}^n,\ i = 1, 2, \cdots, m,\ C \in \mathcal{S}^n, b \in \mathbb{R}^m$

- 拉格朗日函数:

$$L(X, y, S) = \langle C, X\rangle - \sum_{i=1}^{m} y_i(\langle A_i, X\rangle - b_i) - \langle S, X\rangle, \quad S \succeq 0$$

- 对偶函数: $g(y, S) = \inf_X\ L(X, y, S) = \begin{cases} b^{\mathrm{T}}y, & \sum_{i=1}^m y_i A_i - C + S = 0 \\ -\infty, & \text{其他} \end{cases}$
- 对偶问题:

$$\min_{y\in\mathbb{R}^m} \quad -b^{\mathrm{T}}y, \quad \text{s.t.} \quad \sum_{i=1}^{m} y_i A_i - C + S = 0,\ S \succeq 0$$

半定规划对偶问题的对偶问题

$$\min_{y\in\mathbb{R}^m} \quad -b^{\mathrm{T}}y, \qquad \text{s.t.} \qquad \sum_{i=1}^{m} y_i A_i \preceq C$$

- 拉格朗日函数:

$$\begin{aligned} L(y,X) &= -b^{\mathrm{T}}y + \langle X, \sum_{i=1}^{m} y_i A_i - C\rangle \\ &= \sum_{i=1}^{m} y_i(-b_i + \langle A_i, X\rangle) - \langle C, X\rangle, \quad X \succeq 0 \end{aligned}$$

- 对偶函数: $g(X) = \inf_y \ L(y,X) = \begin{cases} -\langle C, X\rangle, & \langle A_i, X\rangle = b_i,\ i = 1,2,\cdots,m \\ -\infty, & \text{其他} \end{cases}$
- 对偶问题:

$$\min_{X\in\mathcal{S}^n} \quad \langle C, X\rangle, \qquad \text{s.t.} \qquad \langle A_i, X\rangle = b_i,\ i = 1,2,\cdots,m,\ X \succeq 0$$

最大割问题

- 令G为一个无向图，其节点集合为$V=\{1,2,\cdots,n\}$和边的集合为E. 令$w_{ij}=w_{ji}$为边$(i,j)\in E$上的权重，并假设$w_{ij}\geq 0,\ (i,j)\in E$. 最大割问题是找到节点集合$V$的一个子集$S$使得$S$与它的补集$\overline{S}$之间相连边的权重之和最大化.
- 可以将最大割问题写成如下整数规划的形式：令$x_j=1,\ j\in S$和$x_j=-1,\ j\in\overline{S}$，则

$$\begin{aligned}\max\quad & \frac{1}{2}\sum_{i<j}(1-x_ix_j)w_{ij}\\ \text{s.t.}\quad & x_j\in\{-1,1\},\ j=1,2,\cdots,n.\end{aligned}\tag{1}$$

- 在问题(1)中，只有当x_i与x_j不同时，目标函数中w_{ij}的系数非零. 最大割问题是一个离散优化问题，很难在多项式时间内找到它的最优解.

最大割问题的对偶

$$\begin{aligned} \max_{x\in\mathbb{R}^n} \quad & x^{\mathrm{T}}Cx \\ \text{s.t.} \quad & x_i^2 = 1,\ i = 1, 2, \cdots, n \end{aligned}$$

- 拉格朗日函数:

$$L(x, y) = -x^{\mathrm{T}}Cx + \sum_{i=1}^{n} y_i(x_i^2 - 1) = x^{\mathrm{T}}(\mathrm{Diag}(y) - C)x - \mathbf{1}^{\mathrm{T}}y$$

- 对偶函数:

$$g(y) = \inf_x L(x, y) = \begin{cases} -\mathbf{1}^{\mathrm{T}}y, & \mathrm{Diag}(y) - C \succeq 0 \\ -\infty, & \text{其他} \end{cases}$$

- 对偶问题:

$$\begin{aligned} \min_{y\in\mathbb{R}^n} \quad & \mathbf{1}^{\mathrm{T}}y \\ \text{s.t.} \quad & \mathrm{Diag}(y) - C \succeq 0 \end{aligned}$$

最大割对偶问题的对偶问题

- 拉格朗日函数:

$$L(y,X)=\mathbf{1}^{\mathrm{T}}y-\langle \mathrm{Diag}(y)-C,X\rangle=\sum_{i=1}^{n}(1-X_{ii})y_i+\langle C,X\rangle$$

- 对偶函数:

$$g(X)=\inf_y L(y,X)=\begin{cases}\langle C,X\rangle, & X_{ii}=1,\ i=1,2,\cdots,n\\ -\infty, & \text{其他}\end{cases}$$

- 对偶问题:

$$\begin{aligned}\max\quad & \langle C,X\rangle\\ \text{s.t.}\quad & X_{ii}=1,\ i=1,2,\cdots,n\\ & X\succeq 0\end{aligned}$$

这是最大割问题SDP松弛的另一个理解。

- 可以证明，最大割问题的SDP松弛问题的解，至少是原问题最优解的0.87856倍的近似解。参考文献：Goemans, Michel X., and David P. Williamson. "Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming." Journal of the ACM (JACM) 42.6 (1995): 1115-1145.

SOCP/SDP 问题对偶

$$\textbf{(P)} \quad \begin{aligned} \min \quad & c^\top x \\ \text{s.t.} \quad & Ax = b, x_{\mathcal{Q}} \succeq 0 \end{aligned} \qquad \textbf{(D)} \quad \begin{aligned} \max \quad & b^\top y \\ \text{s.t.} \quad & A^\top y + s = c, s_{\mathcal{Q}} \succeq 0 \end{aligned}$$

$$\textbf{(P)} \quad \begin{aligned} \min \quad & \langle C, X \rangle \\ \text{s.t.} \quad & \langle A_1, X \rangle = b_1 \\ & \cdots \\ & \langle A_m, X \rangle = b_m \\ & X \succeq 0 \end{aligned} \qquad \textbf{(D)} \quad \begin{aligned} \max \quad & b^\top y \\ \text{s.t.} \quad & \sum_i y_i A_i + S = C \\ & S \succeq 0 \end{aligned}$$

强对偶性：

- If $p^* > -\infty$, (P) is **strictly** feasible, then (D) is feasible and $p^* = d^*$
- If $d^* < +\infty$, (D) is **strictly** feasible, then (P) is feasible and $p^* = d^*$
- If (P) and (D) has **strictly** feasible solutions, then both have optimal solutions.

这里的问题(P)是严格可行的，表示存在$X \succ 0$.

SOCP强对偶定理失效

对于$x \in \mathbb{R}^3$,

$$\begin{array}{ll} \inf & (1,-1,0)x \\ \text{s.t.} & (0,0,1)x = 1 \\ & x_{\mathcal{Q}} \succeq 0 \end{array} \qquad \begin{array}{ll} \sup\limits_{y\in\mathbb{R}, z\in\mathbb{R}^3} & y \\ \text{s.t.} & (0,0,1)^\top y + z = (1,-1,0)^\top \\ & z_{\mathcal{Q}} \succeq 0 \end{array}$$

- primal: $\min\ x_0 - x_1$, s.t. $x_0 \geq \sqrt{x_1^2+1}$; It holds $x_0 - x_1 > 0$ and $x_0 - x_1 \to 0$ if $x_0 = \sqrt{x_1^2+1} \to \infty$. Hence, $p^* = 0$, no finite solution
- dual: $\sup\ y$ s.t. $1 \geq \sqrt{1+y^2}$. Hence, $y = 0$

$p^* = d^*$ but primal is not attainable.

SDP强对偶定理失效

Consider

$$\begin{aligned} \min \quad & \left\langle \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}, X \right\rangle \\ \text{s.t.} \quad & \left\langle \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, X \right\rangle = 0 \\ & \left\langle \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 2 \end{pmatrix}, X \right\rangle = 2 \\ & X \succeq 0 \end{aligned} \qquad \begin{aligned} \max \quad & 2y_2 \\ \text{s.t.} \quad & \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} y_1 + \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 2 \end{pmatrix} y_2 \preceq \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \end{aligned}$$

- primal: $x^* = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$, $p^* = 1$
- dual: $y^* = (0, 0)$. Hence, $d^* = 0$

Both problems have finite optimal values, but $p^* \neq d^*$

提纲

带约束凸优化问题

- 稀疏优化问题:

$$\min_{x\in\mathbb{R}^n} \quad \|x\|_1, s.t. \quad Ax = b.$$

- 低秩矩阵恢复问题:

$$\min_{X\in\mathbb{R}^{m\times n}} \|X\|_*, \text{ s.t. } X_{ij} = M_{ij}, (i,j) \in \Omega.$$

- 矩阵分离问题:

$$\min_{X,S\in\mathbb{R}^{m\times n}} \|X\|_* + \mu\|S\|_1, \text{ s.t.} \quad X + S = M.$$

- 回归分析中的问题:

$$\min_{x\in\mathbb{R}^n} \frac{1}{2}\|Ax - b\|_2^2, \quad \text{s.t.} \quad \|x\|_1 \leqslant \sigma.$$

带约束凸优化问题

- 前述问题都可以写为

$$\begin{aligned} \min_{x\in\mathcal{D}} \quad & f(x), \\ \text{s.t.} \quad & c_i(x) \leqslant 0, \quad i = 1, 2, \cdots, m, \\ & Ax = b, \end{aligned}$$

其中$f(x)$ 为适当的凸函数, $\forall i, c_i(x)$ 是凸函数且$\mathrm{dom}c_i = \mathbb{R}^n$. $A \in \mathbb{R}^{p\times n}, b \in \mathbb{R}^p$ 是已知的.

- 集合$\mathcal{D}$ 表示自变量x 的自然定义域, 即

$$\mathcal{D} = \mathrm{dom}f = \{x \mid f(x) < +\infty\}.$$

  - 若$\mathrm{dom}c_i \neq \mathbb{R}^n$, 则

$$\mathcal{D} = \mathrm{dom}f \cap (\cap_{i=1}^m \mathrm{dom}c_i).$$

- 自变量x 还受约束的限制, 定义可行域

$$\mathcal{X} = \{x \in \mathcal{D} \ : \ c_i(x) \leq 0, \ i = 1, 2, \cdots, m; \ Ax = b\}.$$

- 由于凸优化问题的可行域是凸集, 因此等式约束一般只能是线性约束.

Slater约束品性与强对偶原理

- 凸优化问题有很多好的性质.一个自然的问题是:我们能否像研究无约束问题那样找到该问题最优解的一阶充要条件?如果这样的条件存在, 它在什么样的约束品性下成立?
- 在通常情况下, 优化问题的对偶间隙大于0, 即强对偶原理不满足.
- 但对很多凸优化问题, 在特定约束品性下可以证明强对偶原理.
- 直观的一种约束品性是存在满足所有约束条件的严格可行解.

Slater约束品性与强对偶原理:相对内点

首先给出集合$\mathcal{D}$ 的相对内点集$\mathrm{relint}\mathcal{D}$ 的定义.给定集合$\mathcal{D}$, 记其仿射包为

$$\textbf{affine}\,\mathcal{D} = \{x \mid x = \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_k x_k,\ x_1, x_2, \cdots, x_k \in \mathcal{D},\ \sum_{i=1}^{k} \theta_i = 1\}.$$

定义 (相对内点)

集合$\mathcal{D}$ 的相对内点集定义为

$$\mathrm{relint}\mathcal{D} = \{x \in \mathcal{D} \mid \exists\, r > 0,\ \text{使得}\ B(x, r) \cap \textbf{affine}\,\mathcal{D} \subseteq \mathcal{D}\}.$$

相对内点是内点的推广, 若$\mathcal{D}$ 本身的"维数"较低, 则$\mathcal{D}$ 不可能有内点, 但如果在它的仿射包$\textbf{affine}\,\mathcal{D}$ 中考虑, 则$\mathcal{D}$ 可能有相对内点.

例 :

Slater约束品性

定义 (Slater约束品性)

若对凸优化问题

$$\min_{x\in\mathcal{D}} \quad f(x), \text{s.t. } c_i(x) \leqslant 0,\ i=1,2,\cdots,m, \quad Ax=b,$$

存在$x \in \text{relint}\mathcal{D}$ 满足

$$c_i(x) < 0, \quad i=1,2,\cdots,m, \quad Ax=b,$$

则称对此问题Slater 约束品性满足.该约束品性也称为 Slater 条件.

- Slater约束品性实际上是要求自然定义域$\mathcal{D}$ 的相对内点中存在使得不等式约束严格成立的点, **affine** $\mathcal{D}=\mathbb{R}^n$ 时相对内点就是内点.
- 不等式约束是仿射函数时, Slater条件可以放宽.设前k 个不等式约束是仿射的, 此时Slater约束品性变为:存在$x \in \text{relint}\mathcal{D}$, 满足

$$c_i(x) \leq 0,\ i=1,2,\cdots,k; \quad c_i(x) < 0,\ i=k+1,k+2\cdots,m; \quad Ax=b,$$

即对线性不等式约束无需要求其存在严格可行点.

定理

若凸优化问题满足*Slater*条件, 则强对偶原理成立.

该定理表明，Slater条件成立时，当$d^* > -\infty$ 时, 对偶问题的最优解可以取到, 即存在对偶可行解(λ^*, ν^*), 满足$g(\lambda^*, \nu^*) = d^* = p^*$.

证明：

- 考虑简单情况，假设集合$\mathcal{D}$ 内部非空(即$\operatorname{relint}\mathcal{D} = \mathbf{int}\,\mathcal{D}$), A 行满秩(否则可以去掉多余的线性等式约束)以及原始问题最优函数值p^* 有限（$p^* = -\infty$时，由弱对偶性知对偶问题无可行解,此时$d^* = \infty$）.
- 定义集合

$$\begin{aligned}\mathbb{A} = \{(u, v, t) \mid \exists x \in \mathcal{D},\ c_i(x) \le u_i,\ i = 1, 2, \cdots, m,\\ Ax - b = v, f(x) \le t\}.\\ \mathbb{B} = \{(0, 0, s) \in \mathbb{R}^m \times \mathbb{R}^p \times \mathbb{R} \mid s < p^*\}.\end{aligned}$$

- 可以证明集合$\mathbb{A}$ 和$\mathbb{B}$ 是不相交的.
- 假设存在$(u, v, t) \in \mathbb{A} \cap \mathbb{B}$.根据$(u, v, t) \in \mathbb{B}$, 有$u = 0, v = 0$ 和$t < p^*$.
- 由$(u, v, t) \in \mathbb{A}$, 可知$f(x) \le t < p^*$, 这与p^* 是原始问题最优值矛盾.

Slater约束品性与强对偶原理

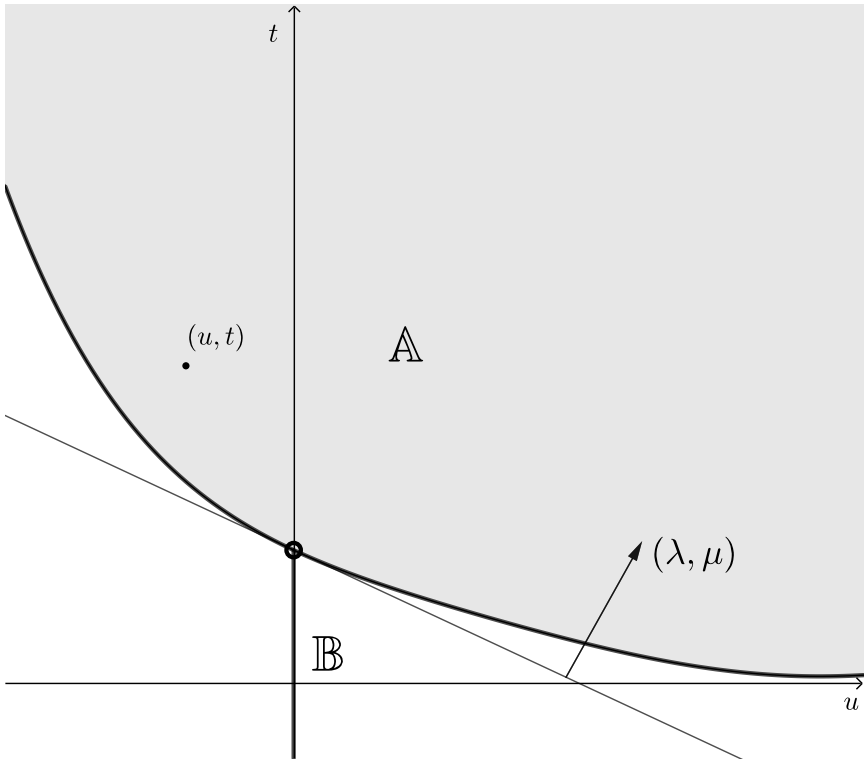


Figure: 集合$\mathbb{A}$ 和$\mathbb{B}$ 在u - t 方向投影的示意图
($\mathbb{A}$ 一般为有内点的凸集, $\mathbb{B}$ 是一条射线且不含点$(0,0,p^*)$)

因为$\mathbb{A}$ 和$\mathbb{B}$ 均为凸集, 由超平面分离定理, 存在$(\lambda,\nu,\mu)\neq 0$ 和α, 使得

$$\lambda^{\mathrm{T}}u+\nu^{\mathrm{T}}v+\mu t\geq\alpha,\quad \forall\,(u,v,t)\in\mathbb{A},$$
$$\lambda^{\mathrm{T}}u+\nu^{\mathrm{T}}v+\mu t\leq\alpha,\quad \forall\,(u,v,t)\in\mathbb{B}.$$

Slater约束品性与强对偶原理

- 我们断言$\lambda \geq 0$和$\mu \geq 0$(否则可以取u_i和t为任意大的正实数以及$\nu = 0$, 这会导致$\lambda^{\mathrm{T}} u + \mu t$在集合$\mathbb{A}$上无下界).
- 同时, 由于$\mu t \leq \alpha$对于所有$t < p^*$成立, 可得$\mu p^* \leq \alpha$.
- 对任意$x \in \mathcal{D}$, 取$(u, v, t) = (c_i(x), Ax - b, f(x)) \in \mathbb{A}$, 可知

$$\sum_{i=1}^{m} \lambda_i c_i(x) + \nu^{\mathrm{T}}(Ax - b) + \mu f(x) \geq \alpha \geq \mu p^*.$$

- 假设$\mu > 0$, 则

$$L(x, \frac{\lambda}{\mu}, \frac{\nu}{\mu}) \geq p^*.$$

进一步地, 我们有$g(\frac{\lambda}{\mu}, \frac{\nu}{\mu}) \geq p^*$, 根据弱对偶性$g(\frac{\lambda}{\mu}, \frac{\nu}{\mu}) \leq p^*$自然成立.因此, 必有$g(\frac{\lambda}{\mu}, \frac{\nu}{\mu}) = p^*$成立.说明在此情况下强对偶性满足, 且对偶最优解可以达到.

Slater约束品性与强对偶原理

- 考虑$\mu = 0$的情况, 可以从上面得到对于所有的$x \in \mathcal{D}$,

$$\sum_{i=1}^{m} \lambda_i c_i(x) + \nu^{\mathrm{T}}(Ax - b) \geq 0.$$

- 取满足Slater条件的点x_S, 有$\sum_{i=1}^{m} \lambda_i c_i(x_S) \geq 0$.
- 又$c_i(x_S) < 0$和$\lambda_i \geq 0$, 我们得到$\lambda = 0$, 上式化为

$$\nu^{\mathrm{T}}(Ax - b) \geq 0, \quad \forall\, x \in \mathcal{D}.$$

  根据$(\lambda, \nu, \mu) \neq 0$可知$\nu \neq 0$, 结合$A$行满秩可以得到$A^{\mathrm{T}}\nu \neq 0$.由于$x_S$是可行解, 我们有$\nu^{\mathrm{T}}(Ax_S - b) = 0$.

- 因为$x_S \in \mathbf{int}\,\mathcal{D}$, 则存在点$\tilde{x} = x_S + e \in \mathcal{D}$, 满足$\nu^{\top} Ae < 0$.这与$\nu^{\top} Ae = \nu^{\top} A(\tilde{x} - x_S) = \nu^{\mathrm{T}}(A\tilde{x} - b) \geq 0$矛盾.
- 综上所述, Slater条件能保证强对偶性.
- 在定理的证明中, Slater条件保证了$\mu \neq 0$.

一阶充要条件

- 对于一般的约束优化问题, 当问题满足特定约束品性时, 我们知道KKT条件是局部最优解处的必要条件.
- 而对于凸优化问题, 当Slater条件满足时, KKT条件则变为局部最优解的充要条件(根据凸性, 局部最优解也是全局最优解).

定理 (凸优化问题的一阶充要条件)

对于凸优化问题, 用a_i 表示矩阵A^{T} 的第i 列, $\partial f, \partial c_i$ 表示次梯度, 如果*Slater*条件成立, 那么x^*, λ^* 分别是原始, 对偶全局最优解当且仅当

$$
\begin{aligned}
\text{稳定性条件} \quad & 0 \in \partial f(x^*) + \sum_{i\in\mathcal{I}} \lambda_i^* \partial c_i(x^*) + \sum_{i\in\mathcal{E}} \lambda_i^* a_i, \\
\text{原始可行性条件} \quad & Ax^* = b, \ \forall i \in \mathcal{E}, \\
\text{原始可行性条件} \quad & c_i(x^*) \le 0, \ \forall i \in \mathcal{I}, \\
\text{对偶可行性条件} \quad & \lambda_i^* \ge 0, \ \forall i \in \mathcal{I}, \\
\text{互补松弛条件} \quad & \lambda_i^* c_i(x^*) = 0, \ \forall i \in \mathcal{I}.
\end{aligned}
$$

一阶充要条件:充分性

- 设存在$(\bar{x},\bar{\lambda})$满足KKT条件, 我们考虑凸优化问题的拉格朗日函数

$$L(x,\lambda)=f(x)+\sum_{i\in\mathcal{I}}\lambda_i c_i(x)+\sum_{i\in\mathcal{E}}\lambda_i(a_i^{\mathrm{T}}x-b_i).$$

- 当固定$\lambda=\bar{\lambda}$时, 注意到$\bar{\lambda}_i\geq 0, i\in\mathcal{I}$以及$\bar{\lambda}_i(a_i^{\mathrm{T}}x), i\in\mathcal{E}$是线性函数可知$L(x,\bar{\lambda})$是关于$x$的凸函数.

- 由凸函数全局最优点的一阶充要性可知, 此时$\bar{x}$就是$L(x,\bar{\lambda})$的全局极小点.根据拉格朗日对偶函数的定义,

$$L(\bar{x},\bar{\lambda})=\inf_{x\in\mathcal{D}}L(x,\bar{\lambda})=g(\bar{\lambda}).$$

- 根据原始可行性条件$A\bar{x}=b$以及互补松弛条件$\bar{\lambda}_i c_i(\bar{x})=0, i\in\mathcal{I}$,

$$L(\bar{x},\bar{\lambda})=f(\bar{x})+0+0=f(\bar{x}).$$

- 根据弱对偶原理,

$$L(\bar{x},\bar{\lambda})=f(\bar{x})\geq p^*\geq d^*\geq g(\bar{\lambda})=L(\bar{x},\bar{\lambda})\Rightarrow p^*=d^*,$$

故$\bar{x},\bar{\lambda}$分别是原始问题和对偶问题的最优解.

关于充分性的评述

- 定理的充分性说明, 若能直接求解出凸优化问题的KKT对, 则其就是对应问题的最优解.
- 在充分性部分的证明中, 我们没有使用Slater条件, 这是因为在证明的一开始假设了KKT点是存在的. Slater条件的意义在于当问题最优解存在时, 其相应KKT条件也会得到满足.
- 当Slater条件不满足时, 即使原始问题存在全局极小值点, 也可能不存在(x^*, λ^*) 满足KKT条件.
  - 例：
    $$\min x, \quad \text{s.t.} \quad x^2 \leq 0.$$
    该问题是凸问题，但是不满足KKT条件。
  - 事实上，只要KKT条件成立，那么其便为凸问题的充分必要条件。后面我们看到LICQ，MFCQ等条件也可也保证局部最优点满足KKT条件。

对偶理论与拉格朗日函数鞍点理论

为了简化问题，我们仅考虑有不等式约束的情况。利用拉格朗日函数，我们有如下原问题的表述：

$$\sup_{\lambda \geq 0} L(x, \lambda) = \begin{cases} f(x), & if\ c_i(x) \leq 0, i \in \mathcal{I} \\ +\infty, & else. \end{cases}$$

故，原问题极小值有

$$p^* = \inf_x \sup_{\lambda \geq 0} L(x, \lambda).$$

而

$$d^* = \sup_{\lambda \geq 0} \inf_x L(x, \lambda).$$

由弱对偶性，我们有

$$\sup_{\lambda \geq 0} \inf_x L(x, \lambda) \leq \inf_x \sup_{\lambda \geq 0} L(x, \lambda).$$

事实上，上述不等式有更一般的形式，不依赖于函数$L(x, \lambda)$的性质，被称之为$\max - \min$不等式：

$$\sup_{y \in Y} \inf_{x \in X} f(x, y) \leq \inf_{x \in X} \sup_{y \in Y} f(x, y),$$

对于任意$f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}, X \subset \mathbb{R}^m, Y \subset \mathbb{R}^n$成立。

强对偶定理成立时，我们有等式成立，从而可以交换$\sup_{\lambda\geq 0}, \inf_x L(x,\lambda)$的顺序。
我们称$(x^*,\lambda^*)\in\mathbb{R}^n\times\mathbb{R}_+^{|\mathcal{I}|}$是拉格朗日函数的鞍点，如果

$$L(x^*,\lambda)\leq L(x^*,\lambda^*)\leq L(x,\lambda^*),\quad \forall(x,\lambda)\in\mathbb{R}^n\times\mathbb{R}_+^{|\mathcal{I}|}$$

定理

如果$(x^*,\lambda^*)\in\mathbb{R}^n\times\mathbb{R}_+^{|\mathcal{I}|}$是拉格朗日函数的鞍点，那么$x^*$是原问题的解。反之，如果$x^*$是原问题的解，并且*Slater*条件在$x^*$处成立，那么存在$\lambda^*$是对偶问题的解，并且$(x^*,\lambda^*)\in\mathbb{R}^n\times\mathbb{R}_+^{|\mathcal{I}|}$是拉格朗日函数的鞍点.

光滑凸优化实例:仿射空间的投影问题

$$\begin{aligned} \min_{x\in\mathbb{R}^n} \quad & \frac{1}{2}\|x-y\|_2^2, \\ \text{s.t.} \quad & Ax=b, \end{aligned}$$

其中$A\in\mathbb{R}^{m\times n}$, $b\in\mathbb{R}^m$ 以及$y\in\mathbb{R}^n$ 为给定的矩阵和向量且A 满秩.

- 拉格朗日函数: $L(x,\lambda)=\frac{1}{2}\|x-y\|^2+\lambda^{\mathrm{T}}(Ax-b)$.
- Slater条件成立, x^* 为一个全局最优解当且仅当存在$\lambda^*\in\mathbb{R}^m$ 使得

$$\begin{cases} x^*-y+A^{\mathrm{T}}\lambda^*=0, \\ Ax^*=b. \end{cases}$$

- 由上述KKT 条件第一式, 等号左右两边同时左乘A 可得

$$Ax^*-Ay+AA^{\mathrm{T}}\lambda=0\Rightarrow\lambda^*=(AA^{\mathrm{T}})^{-1}(Ay-b).$$

- 将λ^* 代回KKT 条件第一式可知

$$x^*=y-A^{\mathrm{T}}(AA^{\mathrm{T}})^{-1}(Ay-b).$$

因此点y 到集合$\{x\mid Ax=b\}$ 的投影为$y-A^{\mathrm{T}}(AA^{\mathrm{T}})^{-1}(Ay-b)$.

example:water-filling 这个问题源自信息论，其中每个变量x_i代表分配给第i个通道的发射功率，而$\log(\alpha_i + x_i)$则给出了该通道的容量或通信速率。这个问题可以看作是在给各个通道分配总功率为一的情况下，最大化总通信速率的问题。

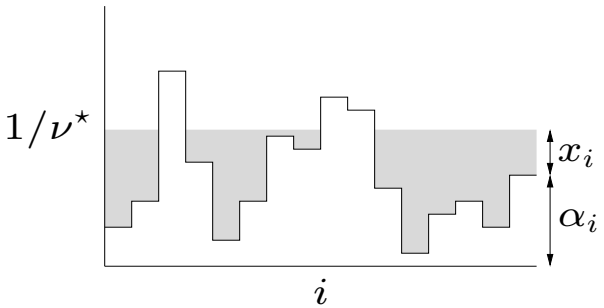
(assume $\alpha_i > 0$)

$$\begin{array}{ll} \min & -\sum_{i=1}^{n} \log(x_i + \alpha_i) \\ \text{s.t.} & x \geq 0, \quad \mathbf{1}^T x = 1 \end{array}$$

x is optimal iff $x \geq 0$, $\mathbf{1}^T x = 1$, and there exist $\lambda \in \mathbb{R}^n$, $\nu \in \mathbb{R}$ such that

$$\lambda \geq 0, \qquad \lambda_i x_i = 0, \qquad \frac{1}{x_i + \alpha_i} + \lambda_i = \nu$$

- if $\nu < 1/\alpha_i$: $\lambda_i = 0$ and $x_i = 1/\nu - \alpha_i$
- if $\nu \geq 1/\alpha_i$: $\lambda_i = \nu - 1/\alpha_i$ and $x_i = 0$
- determine ν from $\mathbf{1}^T x = \sum_{i=1}^{n} \max\{0, 1/\nu - \alpha_i\} = 1$



interpretation

- n patches; level of patch i is at height α_i
- flood area with unit amount of water
- resulting level is $1/\nu^*$

提纲

定义 (切锥)

给定可行域$\mathcal{X}$ 及$x \in \mathcal{X}$, 若存在序列$\{z_k\}_{k=1}^{\infty} \subset \mathcal{X}$, $\lim_{k\to\infty} z_k = x$ 以及正标量序列$\{t_k\}_{k=1}^{\infty}$, $t_k \to 0$ 满足

$$\lim_{k\to\infty} \frac{z_k - x}{t_k} = d$$

则称向量d 为$\mathcal{X}$ 在点x 处的一个切向量. 所有点x 处的切向量构成的集合称为切锥, 用$T_{\mathcal{X}}(x)$ 表示.

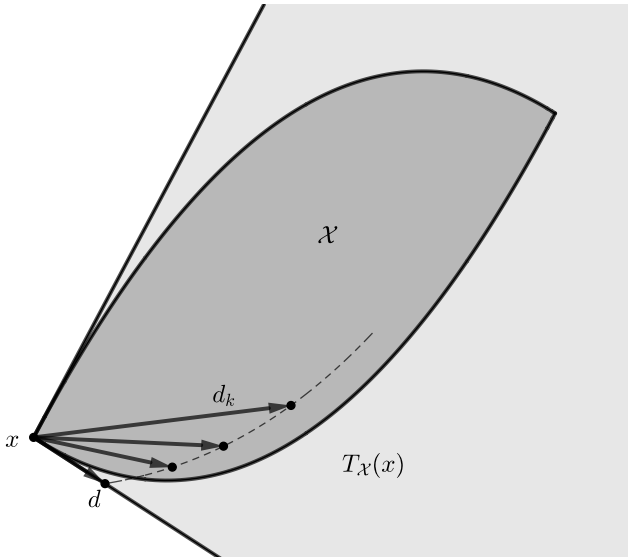


Figure: 不等式约束

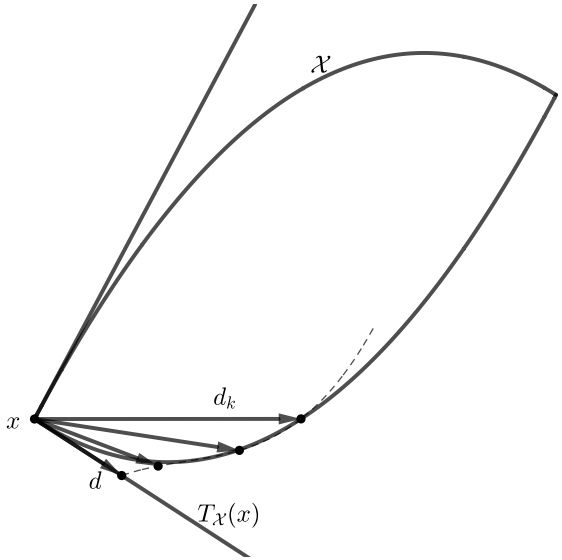


Figure: 等式约束

几何最优性条件

一般优化问题:

$$\begin{aligned} \min_{x\in\mathbb{R}^n} \quad & f(x) \\ \text{s.t.} \quad & c_i(x) \le 0,\ i \in \mathcal{I} \\ & c_i(x) = 0,\ i \in \mathcal{E} \end{aligned}$$

定理 (几何最优性条件)

假设可行点x^* 是上述问题的一个局部极小点.如果$f(x)$ 和$c_i(x),\ i \in \mathcal{I} \cup \mathcal{E}$ 在点x^* 处是可微的, 那么

$$d^{\mathrm{T}} \nabla f(x^*) \ge 0, \quad \forall d \in T_{\mathcal{X}}(x^*).$$

等价于

$$T_{\mathcal{X}}(x^*) \cap \{d \mid \nabla f(x^*)^{\mathrm{T}} d < 0\} = \emptyset.$$

证明

- 若$T_{\mathcal{X}}(x^*) \cap \{d \mid \nabla f(x^*)^{\mathrm{T}} d < 0\} \neq \emptyset$, 取$d \in T_{\mathcal{X}}(x^*)$ 且$f(x^*)^{\mathrm{T}} d < 0$.
- 存在$\{t_k\}_{k=1}^{\infty}$ 和$\{d_k\}_{k=1}^{\infty}$ 使得$x^* + t_k d_k \in \mathcal{X}$, 其中$t_k \to 0$ 且$d_k \to d$.
- 由于$\nabla f(x^*)^{\mathrm{T}} d < 0$, 对于充分大的$k$, 我们有

$$\begin{aligned} f(x^* + t_k d_k) &= f(x^*) + t_k \nabla f(x^*)^{\mathrm{T}} d_k + o(t_k) \\ &= f(x^*) + t_k \nabla f(x^*)^{\mathrm{T}} d + t_k \nabla f(x^*)^{\mathrm{T}} (d_k - d) + o(t_k) \\ &= f(x^*) + t_k \nabla f(x^*)^{\mathrm{T}} d + o(t_k) \\ &< f(x^*) \end{aligned}$$

这与x^* 的局部极小性矛盾.

线性化可行锥

定义 (可行方向)

设$x \in \mathcal{X}, d \in \mathbb{R}^n$是非零向量。若存在$\delta > 0$使得：$x + \lambda d \in \mathcal{X}, \forall \lambda \in (0, \delta)$. 则称$d$是可行域$\mathcal{X}$在$x$处的可行方向。

定义 (线性化可行锥)

对于可行点$x \in \mathcal{X}$, 定义该点的积极集$\mathcal{A}(x) = \mathcal{E} \cup \{i \in \mathcal{I} : c_i(x) = 0\}$, 点$x$处的线性化可行方向锥定义为

$$\mathcal{F}(x) = \left\{ d \;\middle|\; \begin{array}{l} d^{\mathrm{T}} \nabla c_i(x) = 0, \ \forall\, i \in \mathcal{E}, \\ d^{\mathrm{T}} \nabla c_i(x) \leq 0, \ \forall\, i \in \mathcal{A}(x) \cap \mathcal{I} \end{array} \right\}$$

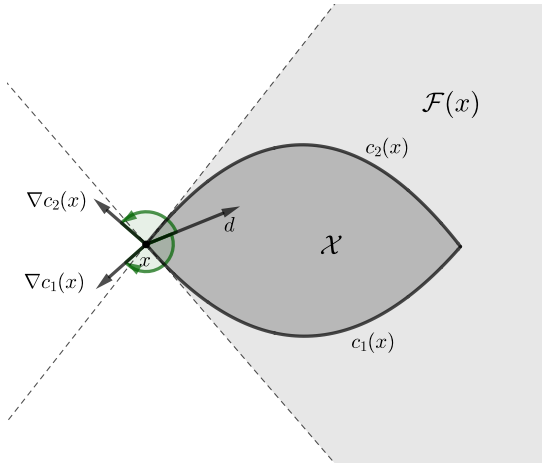


Figure: $\mathbb{R}^2$ 上的不等式约束集合和线性化可行方向锥

线性化可行锥包含切锥

定理 (线性化可行锥包含切锥)

设$c_i(x), i \in \mathcal{E} \cup \mathcal{I}$ 一阶连续可微, 则对任意可行点x 有$T_{\mathcal{X}}(x) \subseteq \mathcal{F}(x)$.

证明:不妨设积极集$\mathcal{A}(x) = \mathcal{E} \cup \mathcal{I}$, 设$d \in T_{\mathcal{X}}(x)$, 由定义

$$\lim_{k\to\infty} t_k = 0, \quad \lim_{k\to\infty} \frac{z_k - x}{t_k} = d \Leftrightarrow z_k = x + t_k d + e_k$$

其中残量e_k 满足$\|e_k\| = o(t_k)$.对$i \in \mathcal{E}$, 根据泰勒展开,

$$\begin{aligned} 0 &= \frac{1}{t_k} c_i(z_k) \\ &= \frac{1}{t_k}(c_i(x) + \nabla c_i(x)^{\mathrm{T}}(t_k d + e_k) + o(t_k)) \\ &= \nabla c_i(x)^{\mathrm{T}} d + \frac{\nabla c_i(x)^{\mathrm{T}} e_k}{t_k} + o(1). \end{aligned}$$

注意到$\frac{\|e_k\|}{t_k} \to 0$, 令$k \to \infty$ 即可得到

$$\nabla c_i(x)^{\mathrm{T}} d = 0, \quad i \in \mathcal{E}.$$

线性化可行锥包含切锥

同理, 对$i \in \mathcal{I}$, 根据泰勒展开,

$$\begin{aligned} 0 &\geq \frac{1}{t_k} c_i(z_k) \\ &= \frac{1}{t_k}(c_i(x) + \nabla c_i(x)^{\mathrm{T}}(t_k d + e_k) + o(t_k)) \\ &= \nabla c_i(x)^{\mathrm{T}} d + \frac{\nabla c_i(x)^{\mathrm{T}} e_k}{t_k} + o(1). \end{aligned}$$

注意到$c_i(x) = 0,\ i \in \mathcal{I}$, 因此我们有

$$\nabla c_i(x)^{\mathrm{T}} d \leq 0, \quad i \in \mathcal{I}$$

结合以上两点, 最终可得到$T_{\mathcal{X}}(x) \subseteq \mathcal{F}(x)$

反例:切锥未必包含线性化可行锥

$$\begin{aligned}\min_{x\in\mathbb{R}} \quad & f(x)=x\\ \text{s.t.} \quad & c(x)=-x+3\le 0\end{aligned}$$

- 则$T_{\mathcal{X}}(3)=\{d\mid d\ge 0\}, \mathcal{F}(3)=\{d:d\ge 0\}$, 于是$T_{\mathcal{X}}(3)=\mathcal{F}(3)$
- 将问题的约束变形为
$$c(x)=(-x+3)^3\le 0$$
因为可行域不变, 故点$x^*=3$ 处, 切锥$T_{\mathcal{X}}(x^*)=\{d:d\ge 0\}$ 不变.
- 由$c'(x^*)=-3(x^*-3)^2=0$ 知线性化可行锥$\mathcal{F}(x^*)=\{d\mid d\in\mathbb{R}\}$
- 此时, $\mathcal{F}(x^*)\supset T_{\mathcal{X}}(x^*)$ (严格包含).

约束品性的引入

- 线性化可行方向锥$\mathcal{F}(x)$ 受可行域$\mathcal{X}$ 代数表示方式的影响
- 切锥$T_{\mathcal{X}}(x)$ 仅由可行域$\mathcal{X}$ 决定
- 线性可行化方向锥容易计算, 但不能反映可行域$\mathcal{X}$ 的本质特征
- 切锥能反映可行域$\mathcal{X}$ 的本质特征, 但不容易计算
- 引入约束品性来沟通两者, 确保最优点x^* 处$T_{\mathcal{X}}(x^*) = \mathcal{F}(x^*)$, 从而可以用$\mathcal{F}(x)$ 取代$T_{\mathcal{X}}(x)$

线性无关约束品性

定义 (线性无关约束品性)

给定可行点x 及相应的积极集$\mathcal{A}(x)$.如果积极集对应的约束函数的梯度,
即$\nabla c_i(x),\ i \in \mathcal{A}(x)$, 是线性无关的, 则称线性无关约束品性(LICQ)在点x 处成立.

定理

给定任意可行点$x \in \mathcal{X}$, 若在该点LICQ 成立, 则有$T_{\mathcal{X}}(x) = \mathcal{F}(x)$.

- 证明:不失一般性, 我们假设积极集$\mathcal{A}(x) = \mathcal{E} \cup \mathcal{I}$.记矩阵

$$A(x) = [\nabla c_i(x)]^{\mathrm{T}}_{i \in \mathcal{I} \cup \mathcal{E}}.$$

- 假设下标集合$\mathcal{A}(x)$ 的元素个数为m, 那么矩阵$A(x) \in \mathbb{R}^{m \times n}$ 并且$\mathrm{rank}(A) = m$.
- 令矩阵$Z \in \mathbb{R}^{n \times (n-m)}$ 为$A(x)$ 的零空间的基矩阵, 则Z 满足

$$\mathrm{rank}(Z) = n - m, \quad A(x)Z = 0.$$

线性无关约束品性

- 令$d \in \mathcal{F}(x)$ 为任意线性化可行方向, 给定$\lim_{k\to\infty} t_k = 0$ 的正标量$\{t_k\}_{k=1}^{\infty}$, 定义映射$R : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^n$:

$$R(z,t) = \begin{bmatrix} c(z) - tA(x)d \\ Z^{\mathrm{T}}(z - x - td) \end{bmatrix},$$

  其中$c(z)$ 为向量值函数, 其第i 个分量为$c_i(z)$.

- 由$\mathcal{A}(x) = \mathcal{E} \cup \mathcal{I}$ 知

$$R(x,0) = \begin{bmatrix} c(x) \\ Z^{\mathrm{T}}(x - x) \end{bmatrix} = 0, \quad \frac{\partial R(x,0)}{\partial z} = \begin{bmatrix} A(x) \\ Z^{\mathrm{T}} \end{bmatrix}.$$

- 根据Z 的构造, 雅可比矩阵$\frac{\partial R(x,0)}{\partial z}$ 是非奇异的.因此, 由隐函数定理, 对任意充分小的t_k, 都存在唯一的z_k, 使得$R(z_k, t_k) = 0$.

- 由于$R(z_k, t_k) = 0$, 故$c_i(z_k) = t_k \nabla c_i(x)^{\mathrm{T}} d,\ i \in \mathcal{I} \cup \mathcal{E}$.根据线性化可行方向$d$ 的定义, $c_i(z_k) \geq 0,\ i \in \mathcal{I},\ c_i(z_k) = 0,\ i \in \mathcal{E}$, 即$z_k$ 为可行点.

- 由泰勒展开得

$$
\begin{aligned}
0 = R(z_k, t_k) &= \begin{bmatrix} c(z_k) - t_k A(x)d \\ Z^{\mathrm{T}}(z_k - x - t_k d) \end{bmatrix} \\
&= \begin{bmatrix} A(x)(z_k - x) + e_k - t_k A(x)d \\ Z^{\mathrm{T}}(z_k - x - t_k d) \end{bmatrix} \\
&= \begin{bmatrix} A(x) \\ Z^{\mathrm{T}} \end{bmatrix} (z_k - x - t_k d) + \begin{bmatrix} e_k \\ 0 \end{bmatrix}.
\end{aligned}
$$

其中残量e_k 满足$\|e_k\| = o(t_k)$.

- 两边同时作用$\begin{bmatrix} A(x)^{\mathrm{T}} & Z \end{bmatrix}^{-\mathrm{T}}$ 并除以t_k, 则有

$$
\frac{z_k - x}{t_k} = d + \begin{bmatrix} A(x) \\ Z^{\mathrm{T}} \end{bmatrix}^{-1} \begin{bmatrix} \frac{e_k}{t_k} \\ 0 \end{bmatrix} \Rightarrow \lim_{k \to \infty} \frac{z_k - x}{t_k} = d,
$$

即$d \in T_{\mathcal{X}}(x)$.故$\mathcal{F}(x) \subseteq T_{\mathcal{X}}(x)$.又$T_{\mathcal{X}}(x) \subseteq \mathcal{F}(x)$, 则两集合相同.

Mangasarian–Fromovitz约束品性(MFCQ)

定义 (Mangasarian–Fromovitz约束品性)

给定可行点x及相应的积极集$\mathcal{A}(x)$.如果存在一个向量$w \in \mathbb{R}^n$, 使得

$$\nabla c_i(x)^{\mathrm{T}} w < 0, \quad \forall i \in \mathcal{A}(x) \cap \mathcal{I},$$
$$\nabla c_i(x)^{\mathrm{T}} w = 0, \quad \forall i \in \mathcal{E},$$

并且等式约束对应的梯度集$\{\nabla c_i(x),\ i \in \mathcal{E}\}$是线性无关的, 则称点$x$处*MFCQ*成立.

- Mangasarian–Fromovitz约束品性是LICQ的的一个常用推广, 简称为MFCQ.
- LICQ可以推出MFCQ, 但是反过来不成立.在MFCQ 成立的情况下, 我们也可以证明$T_{\mathcal{X}}(x) = \mathcal{F}(x)$.

线性约束品性

- 另外一个用来保证$T_{\mathcal{X}}(x) = \mathcal{F}(x)$ 的约束品性是线性约束品性.

定义 (线性约束品性)

若所有的约束函数$c_i(x)$, $i \in \mathcal{I} \cup \mathcal{E}$ 都是线性的, 则称线性约束品性成立.

- 当线性约束品性成立时, 也有$T_{\mathcal{X}}(x) = \mathcal{F}(x)$.
- 因此对只含线性约束的优化问题, 例如线性规划、二次规划, 很自然地有$T_{\mathcal{X}}(x) = \mathcal{F}(x), \forall\, x$.我们无需再关注约束函数的梯度是否线性无关.
- 一般来说, 线性约束品性和LICQ之间没有互相包含的关系.

Karush-Kuhn-Tucker(KKT)条件:引入

- 回顾几何最优性条件:

$$x^* \text{局部极小} \Leftrightarrow T_{\mathcal{X}}(x^*) \cap \{d \mid \nabla f(x^*)^{\mathrm{T}} d < 0\} = \varnothing.$$

- $T_{\mathcal{X}}(x^*) = \mathcal{F}(x^*)$ 时(约束品性成立), 上述条件变为

$$\left\{ d \;\middle|\; \begin{array}{l} d^{\mathrm{T}} \nabla f(x^*) < 0, \\ d^{\mathrm{T}} \nabla c_i(x^*) = 0,\ i \in \mathcal{E}, \\ d^{\mathrm{T}} \nabla c_i(x^*) \le 0,\ i \in \mathcal{A}(x^*) \cap \mathcal{I} \end{array} \right\} = \varnothing.$$

- 上式依然难以验证, 但可使用Farkas引理进行化简.

定理 (Farkas引理)

设p和q为两个非负整数, 给定$\mathbb{R}^n$中$\{a_i\}_{i=1}^p, \{b_i\}_{i=1}^q$和$c$.则满足:

$$d^{\mathrm{T}}a_i = 0,\ i = 1, 2, \cdots, p,$$
$$d^{\mathrm{T}}b_i \geq 0,\ i = 1, 2, \cdots, q,$$
$$d^{\mathrm{T}}c < 0$$

的d不存在当且仅当存在的$\{\lambda_i\}_{i=1}^p$和$\mu_i \geq 0,\ i = 1, 2, \cdots, q$, 使得

$$c = \sum_{i=1}^{p} \lambda_i a_i + \sum_{i=1}^{q} \mu_i b_i.$$

证明:仅证必要性, 若这样的λ_i和μ_i不存在, 定义集合

$$S = \{z \mid z = \sum_{i=1}^{p} \lambda_i a_i + \sum_{i=1}^{q} \mu_i b_i,\ \lambda_i \in \mathbb{R}, \mu_i \geq 0\}.$$

Farkas引理

- S 是一个闭凸锥.因为$c \notin S$, 由凸集的严格分离超平面定理可知:存在$d \in \mathbb{R}^n, \alpha \in \mathbb{R}$ 使得

$$d^{\mathrm{T}}c < \alpha < d^{\mathrm{T}}z, \quad \forall z \in S.$$

- 因为$0 \in S$, 所以

$$d^{\mathrm{T}}c < \alpha < d^{\mathrm{T}}0 = 0.$$

- 任取b_i 与$t \geq 0$, 有$tb_i \in S$, 故$\alpha < td^{\mathrm{T}}b_i$. 由$t$ 的任意性知$d^{\mathrm{T}}b_i \geq 0$.
- 同理, 任取$t \in \mathbb{R}$ 与a_i, 有$ta_i \in S$, 故$td^{\mathrm{T}}a_i < \alpha$.由$t$ 的任意性知$d^{\mathrm{T}}a_i = 0$.
- 综上, 此时的d 为不等式系统的解.

从Farkas引理到KKT条件

- 由Farkas引理, 取$a_i = \nabla c_i(x^*), i \in \mathcal{E}, b_i = \nabla c_i(x^*), i \in \mathcal{A}(x^*) \cap \mathcal{I}$ 以及$c = -\nabla f(x^*)$, 则$T_{\mathcal{X}}(x^*) = \mathcal{F}(x^*)$ 时几何最优性条件等价于:

$$-\nabla f(x^*) = \sum_{i \in \mathcal{E}} \lambda_i^* \nabla c_i(x^*) + \sum_{i \in \mathcal{A}(x^*) \cap \mathcal{I}} \lambda_i^* \nabla c_i(x^*),$$

其中$\lambda_i^* \in \mathbb{R}, i \in \mathcal{E}, \lambda_i^* \geq 0, i \in \mathcal{A}(x^*) \cap \mathcal{I}.$

- 如果补充定义$\lambda_i^* = 0,\ i \in \mathcal{I} \backslash \mathcal{A}(x^*)$, 那么

$$-\nabla f(x^*) = \sum_{i \in \mathcal{I} \cup \mathcal{E}} \lambda_i^* \nabla c_i(x^*),$$

这恰好对应于拉格朗日函数关于x 的一阶最优性条件.

- 互补松弛条件:对于任意的$i \in \mathcal{I}$, 我们注意到

$$\lambda_i^* c_i(x^*) = 0.$$

这说明$i \in \mathcal{A}(x^*) \cap \mathcal{I}$ 时乘子$\lambda_i^* = 0$ 或$c_i(x^*) = 0$ 至少出现一种, 当两种情况恰好只有一种满足时, 我们也称严格互补松弛条件.

KKT条件:总结

假设x^* 是一般优化问题

$$\begin{aligned} \min_{x\in\mathbb{R}^n} \quad & f(x) \\ \text{s.t.} \quad & c_i(x) \le 0,\ i \in \mathcal{I} \\ & c_i(x) = 0,\ i \in \mathcal{E} \end{aligned}$$

的一个局部最优点.如果

$$T_{\mathcal{X}}(x^*) = \mathcal{F}(x^*)$$

成立, 那么存在拉格朗日乘子λ_i^* 使得如下条件成立:

$$\begin{aligned} \text{稳定性条件} \quad & \nabla_x L(x^*, \lambda^*) = \nabla f(x^*) + \sum_{i\in\mathcal{I}\cup\mathcal{E}} \lambda_i^* \nabla c_i(x^*) = 0, \\ \text{原始可行性条件} \quad & c_i(x^*) = 0,\ \forall i \in \mathcal{E}, \\ \text{原始可行性条件} \quad & c_i(x^*) \le 0,\ \forall i \in \mathcal{I}, \\ \text{对偶可行性条件} \quad & \lambda_i^* \ge 0,\ \forall i \in \mathcal{I}, \\ \text{互补松弛条件} \quad & \lambda_i^* c_i(x^*) = 0,\ \forall i \in \mathcal{I}. \end{aligned}$$

KKT条件:注记

- 称满足KKT条件的变量对(x^*, λ^*) 为KKT对.
- 称x^* 为KKT点.
- 如果局部最优点x^* 处$T_{\mathcal{X}}(x^*) \neq \mathcal{F}(x^*)$, 那么$x^*$ 不一定是KKT点.
- KKT条件只是必要的, KKT点不一定是局部最优点.

二阶最优性条件:引入

依旧考虑一般优化问题:

$$\min_{x\in\mathbb{R}^n} f(x), \quad \text{s.t. } c_i(x)\le 0,\ i\in\mathcal{I};\ c_i(x)=0,\ i\in\mathcal{E}.$$

若x^* 是满足KKT条件的点, 假设$T_{\mathcal{X}}(x^*)=\mathcal{F}(x^*)$, 则$\forall d\in\mathcal{F}(x^*)$,

$$d^{\mathrm{T}}\nabla f(x^*)=-\sum_{i\in\mathcal{E}}\underbrace{\lambda_i^* d^{\mathrm{T}}\nabla c_i(x^*)}_{=0}-\sum_{i\in\mathcal{A}(x^*)\cap\mathcal{I}}\underbrace{\lambda_i^* d^{\mathrm{T}}\nabla c_i(x^*)}_{\le 0}\ge 0,$$

此时一阶条件无法判断x^* 是否是最优值点.

- 若$d^{\mathrm{T}}\nabla f(x^*)=0$, 则需要利用二阶信息来进一步判断在其可行邻域内的目标函数值.
- 拉格朗日函数在这些方向上的曲率即可用来判断x^* 的最优性.
- 首先引入临界锥来精确刻画这些方向.

定义 (临界锥)

设(x^*, λ^*) 是满足*KKT*条件的*KKT*对, 定义临界锥为

$$\mathcal{C}(x^*, \lambda^*) = \{d \in \mathcal{F}(x^*) \mid \nabla c_i(x^*)^{\mathrm{T}} d = 0,\ \forall i \in \mathcal{A}(x^*) \cap \mathcal{I} \text{ 且 } \lambda_i^* > 0\},$$

其中$\mathcal{F}(x^*)$ 为点x^* 处的线性化可行方向锥.

- 临界锥是线性化可行方向锥$\mathcal{F}(x^*)$ 的子集.
- 沿着临界锥中的方向进行优化, 所有等式约束和$\lambda_i^* > 0$ 对应的不等式约束(此时这些不等式均取等)都会尽量保持不变.
- 当$d \in \mathcal{C}(x^*, \lambda^*)$ 时, $\forall i \in \mathcal{E} \cup \mathcal{I}$ 有$\lambda_i^* \nabla c_i(x^*)^{\mathrm{T}} d = 0$, 故

$$d^{\mathrm{T}} \nabla f(x^*) = \sum_{i \in \mathcal{E} \cup \mathcal{I}} \lambda_i^* d^{\mathrm{T}} \nabla c_i(x^*) = 0.$$

- 临界锥定义了依据一阶导数不能判断是否为下降或上升方向的线性化可行方向, 必须使用高阶导数信息加以判断.

二阶最优性条件

考虑一般优化问题:

$$\min_{x\in\mathbb{R}^n} f(x), \quad \text{s.t. } c_i(x)\le 0,\ i\in\mathcal{I};\ c_i(x)=0,\ i\in\mathcal{E}.$$

定理 (二阶最优性条件)

必要性:假设x^* 是问题的一个局部最优解, 并且$T_{\mathcal{X}}(x^*)=\mathcal{F}(x^*)$ 成立.令λ^* 为相应的拉格朗日乘子, 即(x^*,λ^*) 满足*KKT*条件, 那么

$$d^{\mathrm{T}}\nabla^2_{xx}L(x^*,\lambda^*)d\ge 0,\quad \forall d\in\mathcal{C}(x^*,\lambda^*).$$

充分性:假设在可行点x^* 处, 存在一个拉格朗日乘子λ^*, 使得(x^*,λ^*) 满足*KKT*条件.如果

$$d^{\mathrm{T}}\nabla^2_{xx}L(x^*,\lambda^*)d> 0,\quad \forall d\in\mathcal{C}(x^*,\lambda^*),\ d\ne 0,$$

那么x^* 为问题的一个严格局部极小解.

二阶最优性条件:无约束VS有约束

回顾无约束优化问题的二阶最优性条件:

- 问题: $\min_{x\in\mathbb{R}^n} f(x)$.
- 必要条件:若x^* 是f 的一个局部极小点, 则$\nabla f(x^*)=0, \nabla^2 f(x^*)\succeq 0$.
- 充分条件:若$\nabla f(x^*)=0, \nabla^2 f(x^*)\succ 0$, 则$x^*$ 是f 的一个局部极小点.

约束优化问题的二阶最优性条件也要求某种“正定性”, 但只需要考虑临界锥$\mathcal{C}(x^*,\lambda^*)$ 中的向量而无需考虑全空间的向量.

有些教材中将其称为“投影半正定性”.

约束优化的最优性理论:例子

$$\min \quad x_1^2 + x_2^2, \qquad \text{s.t.} \quad \frac{x_1^2}{4} + x_2^2 - 1 = 0,$$

其拉格朗日函数为

$$L(x, \lambda) = x_1^2 + x_2^2 + \lambda(\frac{x_1^2}{4} + x_2^2 - 1).$$

该问题可行域在任意一点$x = (x_1, x_2)^{\mathrm{T}}$ 处的线性化可行方向锥为

$$\mathcal{F}(x) = \{(d_1, d_2) \mid \frac{x_1}{4} d_1 + x_2 d_2 = 0\}.$$

因为只有一个等式约束且其对应函数的梯度非零, 故有LICQ成立, 于是$\mathcal{F}(x) = T_{\mathcal{X}}(x)$.若$(x, \lambda)$ 为KKT对, 由于无不等式约束, 故$\mathcal{C}(x, \lambda) = \mathcal{F}(x)$.
可以计算出其4 个KKT对

$$(x^{\mathrm{T}}, \lambda) = (2, 0, -4), \quad (-2, 0, -4), \quad (0, 1, -1) \quad 和 \quad (0, -1, -1).$$

约束优化的最优性理论:例子

考虑第一个KKT对$y = (2, 0, -4)^{\mathrm{T}}$, 计算可得

$$\nabla^2_{xx} L(y) = \begin{bmatrix} 0 & 0 \\ 0 & -6 \end{bmatrix}, \quad \mathcal{C}(y) = \{(d_1, d_2) \mid d_1 = 0\}.$$

取$d = (0, 1)$, 则

$$d^{\mathrm{T}} \nabla^2_{xx} L(y) d = -6 < 0,$$

因此y 不是局部最优点.类似地, 对第三个KKT对$z = (0, 1, -1)$,

$$\nabla^2_{xx} L(z) = \begin{bmatrix} \frac{3}{2} & 0 \\ 0 & 0 \end{bmatrix}, \quad \mathcal{C}(z) = \{(d_1, d_2) \mid d_2 = 0\}.$$

对于任意的$d = (d_1, 0)$ 且$d_1 \neq 0$,

$$d^{\mathrm{T}} \nabla^2_{xx} L(z) d = \frac{3}{2} d_1^2 > 0.$$

因此, z 为一个严格局部最优点.

$$\begin{aligned} \max \quad & \langle C, X\rangle \\ \text{s.t.} \quad & X_{ii} = 1,\ i = 1, 2, \cdots, n \\ & X \succeq 0. \end{aligned} \tag{2}$$

- 拉格朗日函数:

$$L(X, \mu, \Lambda) = \langle C, X\rangle + \sum_{i=1}^{n} \mu_i (X_{ii} - 1) - \mathrm{Tr}(X\Lambda), \quad \Lambda \in \mathcal{S}_+^n,\ \mu \in \mathbb{R}^n.$$

- Slater条件成立, 最优性条件:

$$\begin{cases} C + \mathrm{Diag}(\mu^*) - \Lambda^* = 0, \\ X_{ii}^* = 1,\ i = 1, \cdots, n, \\ X^* \succeq 0, \\ \Lambda^* \succeq 0, \\ \mathrm{Tr}(X^*\Lambda^*) = 0. \end{cases}$$

由于X^* 与Λ^* 的半正定性, $\mathrm{Tr}(X^*\Lambda^*) = 0$ 等价于$X^*\Lambda^* = 0$.

$$\begin{aligned} \max_{Y\in\mathbb{R}^{n\times p}} \quad & \mathrm{Tr}(CYY^{\mathrm{T}}), \\ \text{s.t.} \quad & \mathrm{diag}(YY^{\mathrm{T}}) = \mathbf{1}. \end{aligned} \tag{Burer-Monteiro}$$

- 利用$X = YY^{\mathrm{T}}$, $Y \in \mathbb{R}^{p\times n}$ 来减小规模, 此时$YY^{\mathrm{T}} \succeq 0$ 自然满足. 该分解方法可以推广到SDP的标准形式，问题性质研究见论文Burer, Samuel, and Renato DC Monteiro. "A nonlinear programming algorithm for solving semidefinite programs via low-rank factorization." Mathematical programming 95.2 (2003): 329-357.
- 可行点$Y = [y_1, y_2, \cdots, y_n]^{\mathrm{T}}$ 处约束为$c_i(Y) = \|y_i\|^2 - 1 = 0, \forall i.$

  $$\nabla c_i(Y) = 2[0, \cdots, 0, y_i, 0, \cdots 0]^{\mathrm{T}}.$$

  因为$y_i \neq 0$, 故$\{c_i(Y)\}_{i=1}^n$ 是线性无关的, 即LICQ 成立.
- 拉格朗日函数: $L(Y, \lambda) = \mathrm{Tr}(CYY^{\mathrm{T}}) + \sum_{i=1}^n \lambda_i c_i(Y)$.
- KKT条件:

  $$\begin{cases} 2CY - 2[\lambda_1 y_1, \lambda_2 y_2, \cdots, \lambda_n y_n]^{\mathrm{T}} = 0, \\ \mathrm{diag}(YY^{\mathrm{T}}) = \mathbf{1}. \end{cases}$$

- 令$\tilde{\Lambda} = \mathrm{Diag}(\lambda_1, \lambda_2, \cdots, \lambda_n)$, KKT条件第一式可以转换为

$$(C - \tilde{\Lambda})Y = 0.$$

- 因为该问题只有等式约束, 故临界锥就是切锥, 即

$$\mathcal{C}(Y, \lambda) = \{D \in \mathbb{R}^{n\times p} \mid \mathrm{diag}(Y^{\mathrm{T}}D) = 0\}.$$

- 拉格朗日函数的海瑟矩阵算子形式为

$$\nabla^2_{YY} L(X, \lambda)[D] = 2(C - \tilde{\Lambda})D.$$

- 必要性:假设Y 为一个局部最优解, 则存在λ_i, 使得

$$\begin{cases} \left\langle (C - \tilde{\Lambda})D, D \right\rangle \geq 0, \quad \forall \mathrm{diag}(Y^{\mathrm{T}}D) = 0, \\ \mathrm{diag}(YY^{\mathrm{T}}) = \mathbf{1}, \\ (C - \tilde{\Lambda})Y = 0. \end{cases}$$

- 充分性:假设在一点Y处, 存在$\lambda_i,\ i=1,2,\cdots,n$, 使得

$$\begin{cases} \left\langle (C-\tilde{\Lambda})D, D \right\rangle > 0, \quad \forall \mathrm{diag}(Y^{\mathrm{T}}D)=0, \\ \mathrm{diag}(YY^{\mathrm{T}})=\mathbf{1}, \\ (C-\tilde{\Lambda})Y=0, \end{cases}$$

  那么Y为问题的一个严格局部最优解.

- 利用关系式$(C-\tilde{\Lambda})Y=0$和约束$\mathrm{diag}(YY^{\mathrm{T}})=\mathbf{1}$可以显式求得$\lambda=\mathrm{diag}(CYY^{\mathrm{T}})$.

- 在这个例子中根据约束的特殊结构, 我们能显式给出乘子λ的表达式.这个性质在一般约束优化问题中是没有的.

- 满足一定条件下，例如$p \geq \frac{m(m+1)}{2}$，可以说明$C-\tilde{\Lambda}$是半正定的。因此若Y^*满足问题(Burer-Monteiro)的二阶必要条件, 那么$X=Y^*Y^{*\top}$是问题(2)的全局最优解。具体可以参考文献：Boumal, Nicolas, Vladislav Voroninski, and Afonso S. Bandeira. "Deterministic Guarantees for Burer-Monteiro Factorizations of Smooth Semidefinite Programs." Communications on Pure and Applied Mathematics 73.3 (2020): 581-608.

提纲

总结:无约束优化问题及其最优性条件

问题	一阶条件	二阶条件
可微问题	$\nabla f(x^*) = 0$ (必要)	$\nabla^2 f(x^*) \succeq 0$ (必要) $\nabla^2 f(x^*) \succ 0$ (充分)
凸问题	$0 \in \partial f(x^*)$ (充要)	—
复合优化问题	$-\nabla f(x^*) \in \partial h(x^*)$ (必要)	—
非凸非光滑	$0 \in \partial f(x^*)$ (必要)	—

总结:约束优化问题的最优性条件和相应约束品性

问题	一阶条件	二阶条件	约束品性
一般问题	KKT 条件(必要)	$d^{\mathrm{T}}\nabla^2_{xx}L(x^*,\lambda^*)d \geq 0, \quad \forall d \in \mathcal{C}(x^*,\lambda^*)$ (必要) $d^{\mathrm{T}}\nabla^2_{xx}L(x^*,\lambda^*)d > 0, \quad \forall d \in \mathcal{C}(x^*,\lambda^*),\ d \neq 0$, (充分)[1]	LICQ[2]
凸问题	KKT 条件(充要)	—	Slater

1. 一般约束优化问题的二阶充分条件不需要LICQ 作为前提.
2. 或其他可推出$T_{\mathcal{X}}(x^*) = \mathcal{F}(x^*)$ 的约束品性.

无约束问题的梯度法

陈士祥

中国科学技术大学

致谢：本教案部分参考北京大学文再文教授《最优化与建模》

提纲

无约束优化问题

本章，我们考虑如下问题:

$$\min_{x\in\mathbb{R}^n} \quad f(x).$$

这里，$f(x)$是连续可微的。

迭代算法

迭代算法的一般格式：

$$x_{k+1} = x_k + \alpha_k d_k,$$

- 其中，d_k是x_k处的下降方向，即

$$\nabla f(x_k)^\top d_k < 0.$$

  这也表明，方向导数$f'(x_k; d_k) = \lim\limits_{t \to 0+} \frac{f(x_k + t d_k) - f(x_k)}{t} = \nabla f(x_k)^\top d_k < 0.$

- $\alpha_k > 0$是步长，决定沿着下降方向走多远。

设计算法的目标是，使得迭代点x_k越来越接近最优点，

- $f(x_{k+1}) < f(x_k)$；
- $\|\nabla f(x_k)\| \to 0$, as $k \to \infty$；
- $\|x_{k+1} - x^*\| < \|x_k - x^*\|$, x^*为最优点；
- ...

梯度下降法

- 注意到$\phi(\alpha) = f(x^k + \alpha d^k)$有泰勒展开

$$\phi(\alpha) = f(x^k) + \alpha \nabla f(x^k)^{\mathrm{T}} d^k + \mathcal{O}(\alpha^2 \|d^k\|^2).$$

- 由柯西不等式, 当α足够小时取$d^k = -\nabla f(x^k)$会使函数下降最快.
- 因此梯度法就是选取$d^k = -\nabla f(x^k)$的算法, 它的迭代格式为

$$x^{k+1} = x^k - \alpha_k \nabla f(x^k).$$

步长α_k的选取可依赖于线搜索算法, 也可直接选取固定的α_k.

- 另一种理解方式，在局部使用二次函数逼近原函数，求解近似的函数，

$$\begin{aligned} x^{k+1} &= \arg\min_x f(x^k) + \nabla f(x^k)^\top (x - x^k) + \frac{1}{\alpha_k} \|x - x^k\|_2^2 \\ &= \arg\min_x \|x - (x^k - \alpha_k \nabla f(x^k))\|_2^2 \\ &= x^k - \alpha_k \nabla f(x^k) \end{aligned}$$

二次函数的梯度法

设二次函数$f(x,y)=x^2+10y^2$, 初始点(x^0,y^0) 取为$(10,1)$, 取固定步长$\alpha_k=0.085$. 我们使用梯度法$x^{k+1}=x^k-\alpha_k\nabla f(x^k)$ 进行15 次迭代.

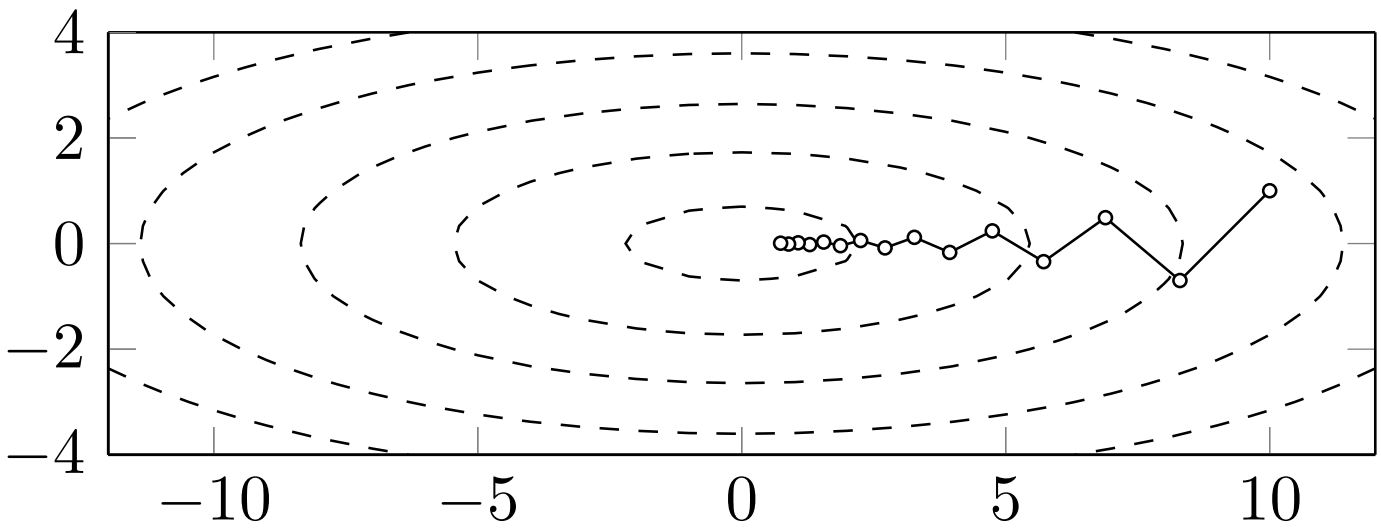


Figure: 梯度法的前15 次迭代

二次函数的收敛定理

为了理解梯度下降（GD）的收敛速率，让我们从二次目标函数开始：

$$\min_x f(x) := \frac{1}{2}(x - x^*)^T Q(x - x^*)$$

对于某个$n \times n$ 矩阵$Q \succ 0$，其中$\nabla f(x) = Q(x - x^*)$。

定理

收敛率：如果$\alpha_k \equiv \eta = \frac{2}{\lambda_1(Q)+\lambda_n(Q)}$，则

$$\|x_k - x^*\|_2^2 \leq \left(\frac{\lambda_1(Q) - \lambda_n(Q)}{\lambda_1(Q) + \lambda_n(Q)}\right)^t \|x_0 - x^*\|_2^2$$

其中$\lambda_1(Q)$（$\lambda_n(Q)$）分别是Q 的最大（最小）特征值。

- 正如我们将看到的，η 被选择以使$|1 - \eta\lambda_n(Q)| = |1 - \eta\lambda_1(Q)|$。
- 收敛率由Q 的条件数$\frac{\lambda_1(Q)}{\lambda_n(Q)}$ 决定，或等价地$\frac{\max_x \lambda_1(\nabla^2 f(x))}{\min_x \lambda_n(\nabla^2 f(x))}$。

Proof: 根据梯度下降（GD）更新规则：

$$x_{k+1} - x^* = x_k - x^* - \alpha_k \nabla f(x_k) = (I - \alpha_k Q)(x_k - x^*)$$

我们有

$$\|x_{k+1} - x^*\|_2^2 \leq \|I - \alpha_k Q\| \|x_k - x^*\|_2^2,$$

$\|I - \alpha_k Q\|$为矩阵的谱范数。收敛速率通过如下式子得到：

$$\|I - \alpha_k Q\| = \max\{|1 - \alpha_k \lambda_1(Q)|, |1 - \alpha_k \lambda_n(Q)|\} = 1 - \frac{2\lambda_n(Q)}{\lambda_1(Q) + \lambda_n(Q)} = \frac{\lambda_1(Q) - \lambda_n(Q)}{\lambda_1(Q) + \lambda_n(Q)}$$

这里第二个等号是因为我们选择了

$$\alpha_k = \frac{2}{\lambda_1(Q) + \lambda_n(Q)}$$

该选择也是最优选择。

精确线搜索

步长规则$\alpha_k \equiv \eta = \frac{2}{\lambda_1(Q)+\lambda_n(Q)}$ 依赖于Q 的谱范数，这需要通过实验得到。
另一个更实际的策略是精确线搜索规则：

$$\alpha_k = \underset{\eta \geq 0}{\operatorname{argmin}} f\left(x_k - \eta \nabla f(x_k)\right)$$

精确线索搜收敛速率

收敛率：如果$\alpha_k = \text{argmin}_{\eta \geq 0} f(x_k - \eta \nabla f(x_k))$，则

$$f(x_k) - f(x^*) \leq \left(\frac{\lambda_1(Q) - \lambda_n(Q)}{\lambda_1(Q) + \lambda_n(Q)} \right)^{2t} (f(x_0) - f(x^*))$$

- 该收敛速率用目标函数值表示
- 收敛率不快于常数步长规则

证明：为了简化符号，让$g_k = \nabla f(x_k) = Q(x_k - x^*)$。可以验证精确线搜索给出：

$$\alpha_k = \frac{g_k^T g_k}{g_k^T Q g_k}$$

所以

$$f(x_{k+1}) = \frac{1}{2}\|x_k - \alpha_k g_k - x^*\|_Q^2 = \frac{1}{2}\|x_k - x^*\|_Q^2 - \alpha_k \|g_k\|^2 + \frac{\alpha_k^2 g_k^T Q g_k}{2}$$

$$= \frac{1}{2}\|x_k - x^*\|_Q^2 - \frac{\|g_k\|^4}{2g_k^T Q g_k} = (1 - \frac{\|g_k\|^4}{2Qg_k g_k^T Q^{-1} g_k}) f(x_k)$$

最后一步等式使用了$f(x_k) = \frac{1}{2} g_k^T Q^{-1} g_k$.
使用Kantorovich's inequality:

$$\frac{\|y\|^4}{(y^T Q y)(y^T Q^{-1} y)} \geq \frac{4\lambda_1(Q)\lambda_n(Q)}{(\lambda_1(Q) + \lambda_n(Q))^2}$$

我们得出结论：

$$f(x_{k+1}) \leq \left(1 - \frac{4\lambda_1(Q)\lambda_n(Q)}{(\lambda_1(Q) + \lambda_n(Q))^2}\right) f(x_k)$$

结论得证，因为$f(x^*) = \min_x f(x) = 0$.

提纲

线搜索方法

- 线搜索: $x^{k+1} = x^k + \alpha_k d^k$
  1. 先确定下降方向: 负梯度、牛顿方向、拟牛顿方向等.
  2. 按某种准则搜索步长.

线搜索算法: 盲人下山

- 求解$f(x)$ 的最小值点如同盲人下山, 无法一眼望知谷底, 而是:
  1. 首先确定下一步该向哪一方向行走.
  2. 再确定沿着该方向行走多远后停下以便选取下一个下山方向.
- 线搜索类算法的数学表述:

$$x^{k+1} = x^k + \alpha_k d^k.$$

  我们称d^k 为迭代点x^k 处的搜索方向, α_k 为相应的步长.这里要求d^k 是一个下降方向, 即$(d^k)^{\mathrm{T}} \nabla f(x^k) < 0$.
- 线搜索类算法的关键是如何选取一个好的方向$d^k \in \mathbb{R}^n$ 以及合适的步长α_k.

α_k 的选取: 精确线搜索算法

- 选取d^k 的方法千差万别, 但选取α_k 的方法却非常相似.
- 首先构造一元辅助函数

$$\phi(\alpha) = f(x^k + \alpha d^k),$$

其中d^k 是给定的下降方向, $\alpha > 0$ 是该辅助函数的自变量.

- 线搜索的目标是选取合适的α_k 使得$\phi(\alpha_k)$ 尽可能减小. 这要求:
  1. α_k 应该使得f 充分下降
  2. 不应在寻找α_k 上花费过多的计算量
- 一个自然的想法是寻找α_k 使得

$$\alpha_k = \operatorname*{argmin}_{\alpha>0} \phi(\alpha),$$

即α_k 为最佳步长. 这种线搜索算法被称为精确线搜索算法

- 选取α_k 通常需要很大计算量, 在实际应用中较少使用

例子: 不合适的线搜索准则导致无法收敛

考虑一维无约束优化问题

$$\min_x \quad f(x) = x^2,$$

迭代初始点$x^0 = 1$. 由于问题是一维的, 下降方向只有$\{-1, +1\}$ 两种. 我们选取$d^k = -\text{sign}(x^k)$, 且只要求选取的步长满足迭代点处函数值单调下降, 即$f(x^k + \alpha_k d^k) < f(x^k)$. 考虑选取如下两种步长:

$$\alpha_{k,1} = \frac{1}{3^{k+1}}, \quad \alpha_{k,2} = 1 + \frac{2}{3^{k+1}},$$

通过简单计算可以得到

$$x_1^k = \frac{1}{2}\left(1 + \frac{1}{3^k}\right), \quad x_2^k = \frac{(-1)^k}{2}\left(1 + \frac{1}{3^k}\right).$$

显然, 序列$\{f(x_1^k)\}$ 和序列$\{f(x_2^k)\}$ 均单调下降, 但序列$\{x_1^k\}$ 收敛的点不是极小值点, 序列$\{x_2^k\}$ 则在原点左右振荡, 不存在极限

非精确线搜索: Armijo准则

定义 (Armijo 准则)

设d^k 是点x^k 处的下降方向, 若

$$f(x^k + \alpha d^k) \leq f(x^k) + c_1 \alpha \nabla f(x^k)^{\mathrm{T}} d^k,$$

则称步长α 满足***Armijo*** 准则, 其中$c_1 \in (0,1)$ 是一个常数.

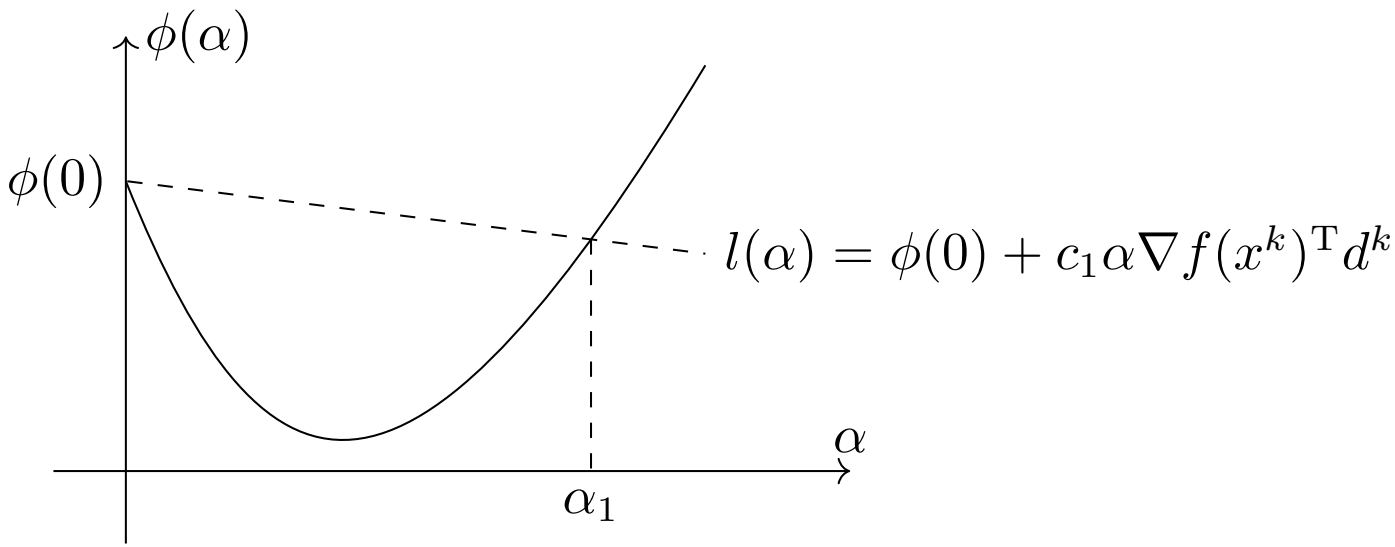


Figure: Armijo 准则

Armijo准则: 评注

- 引入Armijo 准则的目的是保证每一步迭代充分下降
- Armijo 准则有直观的几何含义, 它指的是点$(\alpha, \phi(\alpha))$ 必须在直线

$$l(\alpha) = \phi(0) + c_1 \alpha \nabla f(x^k)^{\mathrm{T}} d^k$$

  的下方, 上图中区间$[0, \alpha_1]$ 中的点均满足Armijo 准则
- 参数c_1 通常选为一个很小的正数, 例如$c_1 = 10^{-3}$, Armijo准则非常容易得到满足
- Armijo 准则需要配合其他准则以保证迭代的收敛性, 因为$\alpha = 0$ 显然满足Armijo准则, 此时迭代序列中的点固定不变

回溯(backtracking)法:以Armijo准则为例

- 给定初值$\hat{\alpha}$, 回溯法通过不断以指数方式缩小试探步长, 找到第一个满足Armijo 准则的点
- 回溯法选取

$$\alpha_k = \gamma^{j_0}\hat{\alpha},$$

其中

$$j_0 = \min\{j = 0, 1, \cdots \mid f(x^k + \gamma^j \hat{\alpha} d^k) \leq f(x^k) + c_1 \gamma^j \hat{\alpha} \nabla f(x^k)^{\mathrm{T}} d^k\},$$

参数$\gamma \in (0, 1)$ 为一个给定的实数

Algorithm 线搜索回溯法

1: 选择初始步长$\hat{\alpha}$, 参数$\gamma, c \in (0, 1)$. 初始化$\alpha \leftarrow \hat{\alpha}$.
2: **while** $f(x^k + \alpha d^k) > f(x^k) + c\alpha \nabla f(x^k)^{\mathrm{T}} d^k$ **do**
3: 令$\alpha \leftarrow \gamma\alpha$.
4: **end while**
5: 输出$\alpha_k = \alpha$.

回溯法:以Armijo准则为例

- 该算法被称为回溯法是因为α 的试验值是由大至小的, 它可以确保输出的α_k 能尽量地大
- 算法1不会无限进行下去, 因为d^k 是一个下降方向, 当α 充分小时, Armijo 准则总是成立的
- 实际应用中我们通常也会给α 设置一个下界, 防止步长过小

Wolfe准则

- Goldstein 准则: 设d^k 是点x^k 处的下降方向, 若

$$f(x^k + \alpha d^k) \leq f(x^k) + c\alpha \nabla f(x^k)^{\mathrm{T}} d^k, \tag{1a}$$

$$f(x^k + \alpha d^k) \geq f(x^k) + (1-c)\alpha \nabla f(x^k)^{\mathrm{T}} d^k, \tag{1b}$$

则称步长α 满足***Goldstein*** 准则, 其中$c \in \left(0, \frac{1}{2}\right)$.

- Wolfe 准则: 设d^k 是点x^k 处的下降方向, 若

$$f(x^k + \alpha d^k) \leq f(x^k) + c_1 \alpha \nabla f(x^k)^{\mathrm{T}} d^k, \tag{2a}$$

$$\nabla f(x^k + \alpha d^k)^{\mathrm{T}} d^k \geq c_2 \nabla f(x^k)^{\mathrm{T}} d^k, \tag{2b}$$

则称步长α 满足***Wolfe*** 准则, 其中$c_1, c_2 \in (0,1)$ 为给定的常数且$c_1 < c_2$.

Wolfe准则

- $\nabla f(x^k + \alpha d^k)^{\mathrm{T}} d^k$ 恰好就是$\phi(\alpha)$ 的导数, Wolfe 准则实际要求$\phi(\alpha)$ 在点α 处切线的斜率不能小于$\phi'(0)$ 的c_2 倍
- $\phi(\alpha)$ 的极小值点α^* 处有$\phi'(\alpha^*) = \nabla f(x^k + \alpha^* d^k)^{\mathrm{T}} d^k = 0$, 因此$\alpha^*$ 永远满足条件二. 而选择较小的c_1 可使得α^* 同时满足条件一, 即Wolfe 准则在绝大多数情况下会包含线搜索子问题的精确解

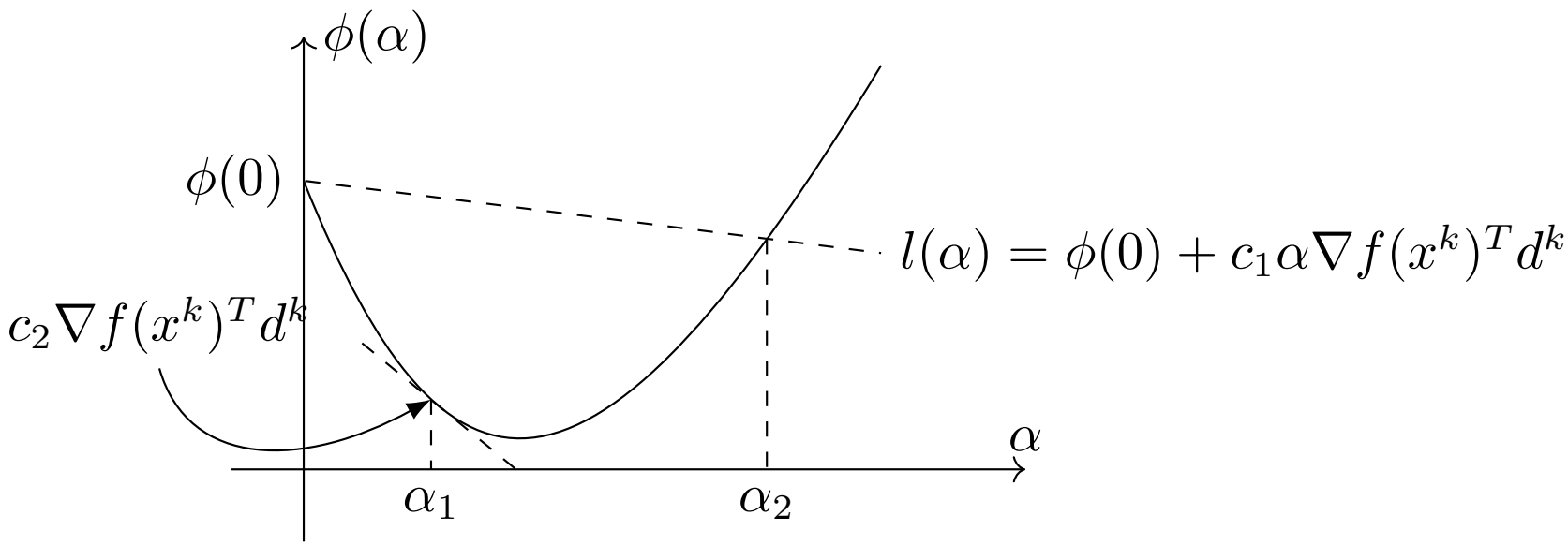


Figure: Wolfe 准则

提纲

梯度利普希茨连续

定义 (梯度利普希茨（Lipschitz)连续)

给定可微函数f，若存在$L>0$，对任意的$x,y\in \mathrm{dom}f$有

$$\|\nabla f(x)-\nabla f(y)\|\leq L\|x-y\|, \tag{3}$$

则称f是梯度利普希茨连续的，相应利普希茨常数为L. 有时也简记为梯度L-利普希茨连续或L-光滑.

引理 (二次上界)

设可微函数$f(x)$的定义域$\mathrm{dom}f=\mathbb{R}^n$，且为梯度$L$-利普希茨连续的，则函数$f(x)$有二次上界：

$$f(y)\leq f(x)+\nabla f(x)^{\mathrm{T}}(y-x)+\frac{L}{2}\|y-x\|^2,\quad \forall\, x,y\in \mathrm{dom}f. \tag{4}$$

注

$f(x)$二阶可微,那么f是-强凸,L-光滑的,如果

$$LI\succeq \nabla^2 f(x)\succeq mI,\quad \forall x.$$

证明

可以证明:

$$
\begin{aligned}
& f(y)-f(x)-\nabla f(x)^{\mathrm{T}}(y-x) \\
=& \int_0^1 (\nabla f(x+t(y-x))-\nabla f(x))^{\mathrm{T}}(y-x)\mathrm{d}t \\
\leq& \int_0^1 \|\nabla f(x+t(y-x))-\nabla f(x)\|\|y-x\|\mathrm{d}t \\
\leq& \int_0^1 L\|y-x\|^2 t\mathrm{d}t = \frac{L}{2}\|y-x\|^2,
\end{aligned}
$$

其中最后一行的不等式利用了梯度利普希茨连续的条件(3). 整理可得(4) 式成立.

梯度法在凸函数上的收敛性

考虑梯度法

$$x^{k+1} = x^k - \alpha_k \nabla f(x^k)$$

假设：

- 设函数$f(x)$ 为凸的梯度L -利普希茨连续函数
- 极小值$f^* = f(x^*) = \inf_x f(x)$ 存在且可达.
- 如果步长α_k 取为常数α 且满足$0 < \alpha < \frac{1}{L}$

结论：点列$\{x^k\}$的函数值收敛到最优值, 且在函数值的意义下收敛速度为$\mathcal{O}\left(\frac{1}{k}\right)$.

如果函数f 还是m -强凸函数, 则梯度法的收敛速度会进一步提升为Q-线性收敛.

证明

- 因为函数f 是利普希茨可微函数, 对任意的x, 根据二次上界引理,

$$f(x-\alpha\nabla f(x)) \leq f(x) - \alpha\left(1-\frac{L\alpha}{2}\right)\|\nabla f(x)\|^2.$$

- 记$\tilde{x} = x - \alpha\nabla f(x)$ 并限制$0 < \alpha < \frac{1}{L}$, 我们有

$$\begin{aligned} f(\tilde{x}) &\leq f(x) - \frac{\alpha}{2}\|\nabla f(x)\|^2 \\ &\leq f^* + \nabla f(x)^{\mathrm{T}}(x-x^*) - \frac{\alpha}{2}\|\nabla f(x)\|^2 \\ &= f^* + \frac{1}{2\alpha}\left(\|x-x^*\|^2 - \|x-x^*-\alpha\nabla f(x)\|^2\right) \\ &= f^* + \frac{1}{2\alpha}(\|x-x^*\|^2 - \|\tilde{x}-x^*\|^2), \end{aligned}$$

其中第一个不等式是因为$0 < \alpha < \frac{1}{L}$, 第二个不等式为f 的凸性.

证明

- 在上式中取$x = x^{i-1}, \tilde{x} = x^i$ 并将不等式对$i = 1, 2, \cdots, k$ 求和得到

$$\begin{aligned}\sum_{i=1}^{k}(f(x^i) - f^*) &\leq \frac{1}{2\alpha}\sum_{i=1}^{k}\left(\|x^{i-1} - x^*\|^2 - \|x^i - x^*\|^2\right)\\ &= \frac{1}{2\alpha}\left(\|x^0 - x^*\|^2 - \|x^k - x^*\|^2\right)\\ &\leq \frac{1}{2\alpha}\|x^0 - x^*\|^2.\end{aligned}$$

- 由于$f(x^i)$ 是非增的, 所以

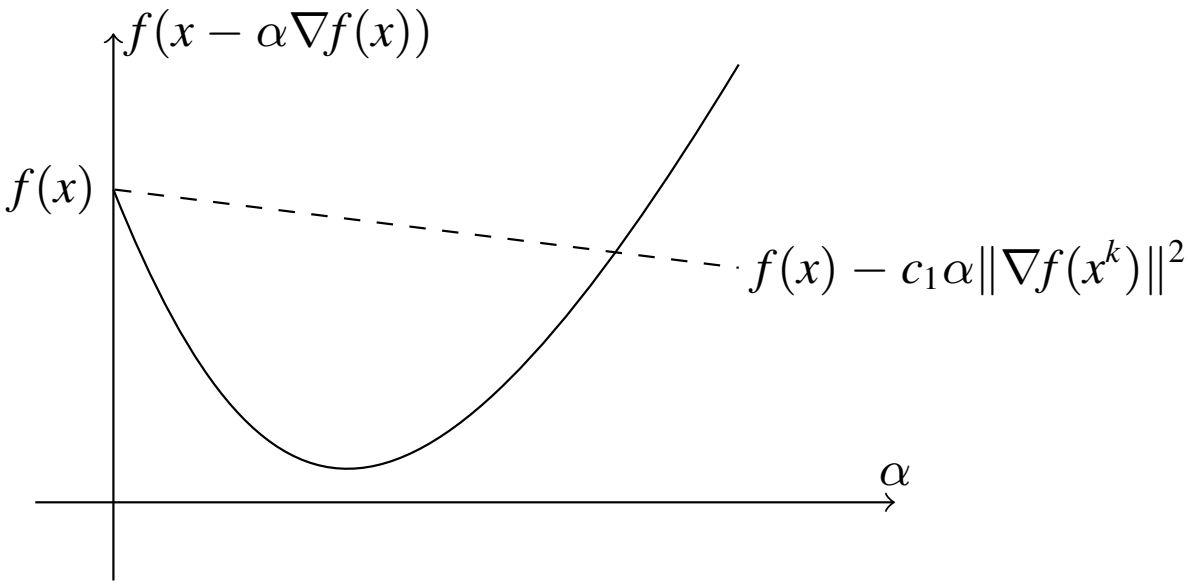
$$f(x^k) - f^* \leq \frac{1}{k}\sum_{i=1}^{k}(f(x^i) - f^*) \leq \frac{1}{2k\alpha}\|x^0 - x^*\|^2.$$

回溯法

- 给定初值$\hat{\alpha}$, 回溯法取步长$\alpha_k = \gamma^{j_0}\hat{\alpha}$, 其中$j_0$是第一个满足Armijo 准则的整数:

$$j_0 = \min\{j = 0, 1, \cdots \mid f(x^k - \gamma^j \hat{\alpha}\nabla f(x^k)) \leq f(x^k) - c_1\gamma^j\hat{\alpha}\|\nabla f(x^k)\|^2\},$$

参数$\gamma \in (0, 1)$ 为一个给定的实数



回溯法

- 由回溯法的机制，注意到Armijo 准则在$\alpha = \alpha_k/\gamma$不满足，即有:

$$f(x^k - \frac{\alpha_k}{\gamma}\nabla f(x^k)) > f(x^k) - c_1\frac{\alpha_k}{\gamma}\|\nabla f(x^k)\|^2.$$

- 另一方面由二次上界可得

$$f(x^k) - \frac{\alpha_k}{\gamma}\left(1 - \frac{L\alpha_k}{2\gamma}\right)\|\nabla f(x^k)\|^2 \geq f(x^k - \frac{\alpha_k}{\gamma}\nabla f(x^k))$$

- 整理上述两式可得:

$$-\frac{\alpha_k}{\gamma}\left(1 - \frac{L\alpha_k}{2\gamma}\right) > -c_1\frac{\alpha_k}{\gamma}.$$

取$c_1 = 1/2$得$\alpha_k \geq \frac{\gamma}{L}$. 综合初始步长得到$\alpha_k \geq \alpha_{\min} = \min\left(\hat{\alpha}, \frac{\gamma}{L}\right)$.

- 实际上，这告诉我们，回溯法可以有效估计Lipschitz常数L.

回溯法收敛性分析

- 由第28页:

$$\begin{aligned} f(x^i) &\leq f^* + \frac{1}{2\alpha_i}\left(\|x^{i-1} - x^*\|_2^2 - \|x^i - x^*\|_2^2\right) \\ &\leq f^* + \frac{1}{2\alpha_{\min}}\left(\|x^{i-1} - x^*\|_2^2 - \|x^i - x^*\|_2^2\right) \end{aligned}$$

- 将这些式子求和得到:

$$f(x^k) - f^* \leq \frac{1}{k}\sum_{i=1}^{k}(f(x^i) - f^*) \leq \frac{1}{2kt_{\min}}\|x^0 - x^*\|_2^2$$

结论: 类似固定步长情形有复杂度$O(1/k)$

引理

设函数$f(x)$是$\mathbb{R}^n$上的凸可微函数, 则以下结论等价:

1. f的梯度为L-利普希茨连续的;
2. 函数$g(x) \overset{\text{def}}{=\!=} \frac{L}{2}x^{\mathrm{T}}x - f(x)$是凸函数;
3. $\nabla f(x)$有余强制性, 即对任意的$x, y \in \mathbb{R}^n$, 有

$$(\nabla f(x) - \nabla f(y))^{\mathrm{T}}(x-y) \geq \frac{1}{L}\|\nabla f(x) - \nabla f(y)\|^2.$$

(1) $\Longrightarrow$ (2) 即证$g(x)$的单调性. 对任意$x, y \in \mathbb{R}^n$,

$$\begin{aligned}(\nabla g(x) - \nabla g(y))^{\mathrm{T}}(x-y) &= L\|x-y\|^2 - (\nabla f(x) - \nabla f(y))^{\mathrm{T}}(x-y)\\ &\geq L\|x-y\|^2 - \|x-y\|\|\nabla f(x) - \nabla f(y)\| \geq 0.\end{aligned}$$

因此$g(x)$为凸函数.

引理 (梯度L-利普希茨函数的性质)

设可微函数$f(x)$的定义域为$\mathbb{R}^n$且存在一个全局极小点x^*, 若$f(x)$为梯度L-利普希茨连续的, 则对任意的x有

$$\frac{1}{2L}\|\nabla f(x)\|^2 \leq f(x) - f(x^*).$$

(2) $\Longrightarrow$ (3)

- 构造辅助函数

$$f_x(z) = f(z) - \nabla f(x)^{\mathrm{T}} z,$$
$$f_y(z) = f(z) - \nabla f(y)^{\mathrm{T}} z,$$

  容易验证f_x和f_y均为凸函数.

- $g_x(z) = \frac{L}{2} z^{\mathrm{T}} z - f_x(z)$关于$z$是凸函数. 根据凸函数的性质, 我们有

$$g_x(z_2) \geq g_x(z_1) + \nabla g_x(z_1)^{\mathrm{T}}(z_2 - z_1), \quad \forall\, z_1, z_2 \in \mathbb{R}^n.$$

  整理可推出$f_x(z)$有二次上界, 且对应的系数也为L.

- 注意到$\nabla f_x(x)=0$, 这说明x 是$f_x(z)$ 的最小值点. 由上页引理,

$$\begin{aligned} f_x(y)-f_x(x) &= f(y)-f(x)-\nabla f(x)^{\mathrm{T}}(y-x) \\ &\geq \frac{1}{2L}\|\nabla f_x(y)\|^2 = \frac{1}{2L}\|\nabla f(y)-\nabla f(x)\|^2. \end{aligned}$$

- 同理, 对$f_y(z)$ 进行类似的分析可得

$$f(x)-f(y)-\nabla f(y)^{\mathrm{T}}(x-y) \geq \frac{1}{2L}\|\nabla f(y)-\nabla f(x)\|^2.$$

将以上两式不等号左右分别相加, 可得余强制性.

(3) $\Longrightarrow$ (1) 由余强制性和柯西不等式,

$$\begin{aligned} \frac{1}{L}\|\nabla f(x)-\nabla f(y)\|^2 &\leq (\nabla f(x)-\nabla f(y))^{\mathrm{T}}(x-y) \\ &\leq \|\nabla f(x)-\nabla f(y)\|\|x-y\|, \end{aligned}$$

整理后即可得到$f(x)$ 是梯度L -利普希茨连续的.

梯度法在强凸函数上的收敛性

定理 (梯度法在强凸函数上的收敛性)

设函数$f(x)$为m-强凸的梯度L-利普希茨连续函数,$f(x^*) = \inf_x f(x)$存在且可达. 如果步长α满足$0 < \alpha < \frac{2}{m+L}$, 那么由梯度下降法迭代得到的点列$\{x^k\}$收敛到$x^*$, 且为$Q$-线性收敛.

- 首先根据f强凸且∇f利普希茨连续, 可得

$$g(x) = f(x) - \frac{m}{2}x^{\mathrm{T}}x$$

  为凸函数且$\frac{L-m}{2}x^{\mathrm{T}}x - g(x)$为凸函数.

- 由引理知函数$g(x)$是梯度$(L-m)$-利普希茨连续的. 再次利用引理可得关于$g(x)$的余强制性

$$(\nabla g(x) - \nabla g(y))^{\mathrm{T}}(x-y) \geq \frac{1}{L-m}\|\nabla g(x) - \nabla g(y)\|^2.$$

梯度法在强凸函数上的收敛性

- 代入$g(x)$ 的表达式, 可得

$$(\nabla f(x)-\nabla f(y))^{\mathrm{T}}(x-y)\geq \frac{mL}{m+L}\|x-y\|^2+\frac{1}{m+L}\|\nabla f(x)-\nabla f(y)\|^2.$$

- 再估计固定步长下梯度法的收敛速度. 设步长$\alpha\in\left(0,\frac{2}{m+L}\right)$, 对$x^k,x^*$ 应用上式并注意到$\nabla f(x^*)=0$ 得

$$\begin{aligned}\|x^{k+1}-x^*\|_2^2&=\|x^k-\alpha\nabla f(x^k)-x^*\|^2\\&=\|x^k-x^*\|^2-2\alpha\nabla f(x^k)^{\mathrm{T}}(x^k-x^*)+\alpha^2\|\nabla f(x^k)\|^2\\&\leq\left(1-\alpha\frac{2mL}{m+L}\right)\|x^k-x^*\|^2+\alpha\left(\alpha-\frac{2}{m+L}\right)\|\nabla f(x^k)\|^2\\&\leq\left(1-\alpha\frac{2mL}{m+L}\right)\|x^k-x^*\|^2\end{aligned}$$

$$\Rightarrow\|x^k-x^*\|^2\leq c^k\|x^0-x^*\|^2,\quad c=1-\alpha\frac{2mL}{m+L}<1.$$

函数值收敛

强凸函数假设下

- 迭代点列$\{x^k\}$Q-线性收敛
- 如果取$t=\frac{2}{m+L}$，则有$c=\frac{(\gamma-1)^2}{(\gamma+1)}$且$\gamma=L/m$, 即

$$\|x^k-x^*\|^2\leq\frac{(\gamma-1)^{2k}}{(\gamma+1)}\|x^0-x^*\|^2 \tag{5}$$

如果$\mathbf{dom}\,f=\mathbf{R}^n$且$f$有极小点$x^*$, 则

$$\frac{1}{2L}\|\nabla f(x)\|_2^2\leq f(x)-f(x^*)\leq\frac{L}{2}\|x-x^*\|_2^2\quad\forall x$$

因此：

$$f(x^k)-f^*\leq\frac{L}{2}\|x^k-x^*\|_2^2\leq\frac{c^kL}{2}\|x^0-x^*\|_2^2$$

函数值的估计：达到$f(x^k)-f^*\leq\epsilon$的迭代步数是$O(\log(1/\epsilon))$

提纲

Barzilar-Borwein 方法

- Barzilar-Borwein (BB) 方法是一种特殊的梯度法, 经常比一般的梯度法有着更好的效果.
- BB 方法利用对角阵$B^k = \frac{1}{\alpha_k} I$近似海瑟矩阵$\nabla^2 f(x)$，希望尽可能满足类似拟牛顿条件:

$$B^k s^{k-1} \approx y^{k-1},$$

其中$s^{k-1} \overset{\text{def}}{=\!=} x^k - x^{k-1}$ 以及$y^{k-1} \overset{\text{def}}{=\!=} \nabla f(x^k) - \nabla f(x^{k-1})$.

- BB 方法选取的α_k 是如下两个最优问题之一的解:

$$\begin{aligned} \min_{\alpha} \quad & \|\alpha y^{k-1} - s^{k-1}\|^2, \\ \min_{\alpha} \quad & \|y^{k-1} - \alpha^{-1} s^{k-1}\|^2. \end{aligned}$$

- 因此得到梯度下降法的格式:

$$x^{k+1} = x^k - (B^k)^{-1} \nabla f(x^k) \Longleftrightarrow x^{k+1} = x^k - \alpha_k \nabla f(x^k)$$

Barzilar-Borwein 方法

- 容易验证两个子问题的解分别为

$$\alpha_{\mathrm{BB1}}^k \overset{\text{def}}{=\!=} \frac{(s^{k-1})^{\mathrm{T}} y^{k-1}}{(y^{k-1})^{\mathrm{T}} y^{k-1}} \quad \text{和} \quad \alpha_{\mathrm{BB2}}^k \overset{\text{def}}{=\!=} \frac{(s^{k-1})^{\mathrm{T}} s^{k-1}}{(s^{k-1})^{\mathrm{T}} y^{k-1}},$$

- 因此可以得到BB 方法的两种迭代格式：

$$x^{k+1} = x^k - \alpha_{\mathrm{BB1}}^k \nabla f(x^k) \quad \text{和} \quad x^{k+1} = x^k - \alpha_{\mathrm{BB2}}^k \nabla f(x^k).$$

- 计算两种BB 步长的任何一种仅仅需要函数相邻两步的梯度信息和迭代点信息，不需要任何线搜索算法即可选取算法步长.

- BB方法计算出的步长可能过大或过小，因此我们还需要将步长做上界和下界的截断，即选取$0 < \alpha_m < \alpha_M$ 使得

$$\alpha_m \leq \alpha_k \leq \alpha_M.$$

- BB 方法本身是非单调方法，有时也配合非单调收敛准则使用以获得更好的实际效果.

非单调线搜索的BB方法

Algorithm 非单调线搜索的BB方法

给定x^0, 选取初值$\alpha > 0$, 整数$M \geq 0$, $c_1, \beta, \varepsilon \in (0,1)$, $k = 0$.
while $\|\nabla f(x^k)\| > \varepsilon$ **do**
  while $f(x^k - \alpha \nabla f(x^k)) \geq \max_{0 \leq j \leq \min(k,M)} f(x^{k-j}) - c_1 \alpha \|\nabla f(x^k)\|^2$ **do**
    令$\alpha \leftarrow \beta \alpha$.
  end while
  令$x^{k+1} = x^k - \alpha \nabla f(x^k)$.
  根据BB步长公式之一计算α, 并做截断使得$\alpha \in [\alpha_m, \alpha_M]$.
  $k \leftarrow k + 1$.
end while

二次函数的BB 方法

- 设二次函数$f(x,y) = x^2 + 10y^2$, 并使用*BB* 方法进行迭代, 初始点为$(-10,-1)$.
- *BB* 方法的收敛速度较快, 在经历*15* 次迭代后已经接近最优值点. 从等高线也可观察到*BB* 方法是非单调方法.
- 实际上, 对于正定二次函数, BB方法有R-线性收敛速度.

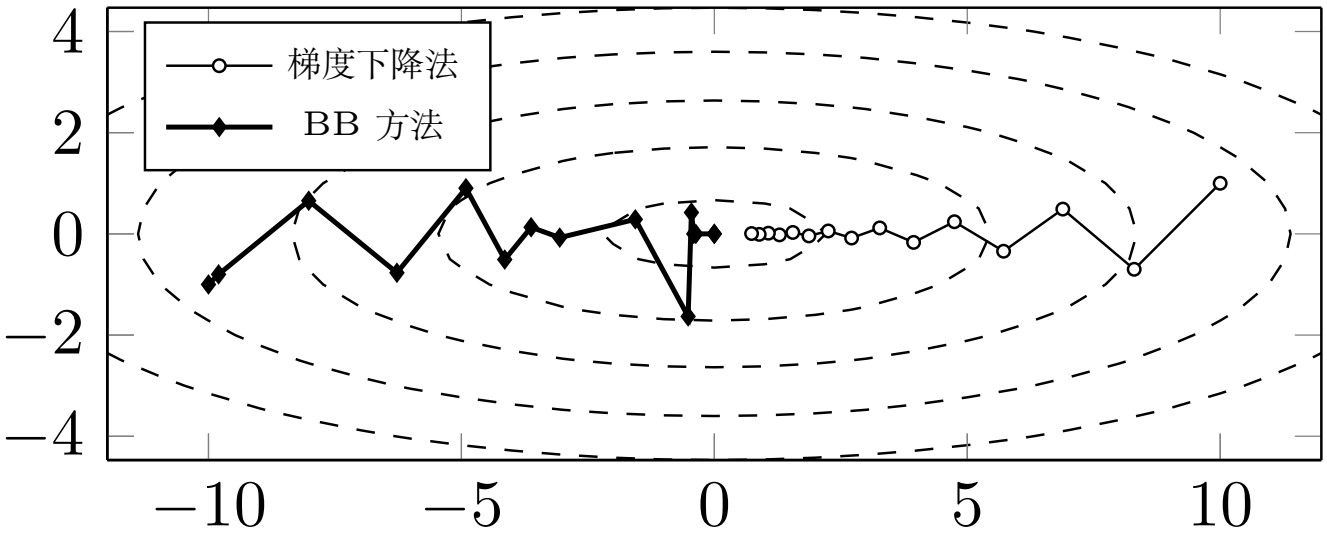


Figure: 梯度法与BB 方法的前15 次迭代

提纲

局部结构性条件

到目前为止，我们在强凸性和光滑性的条件下建立了梯度法的线性收敛。
强凸性要求往往可以放宽，例如有如下几种条件

- 局部强凸性
- 规则性条件
- Polyak-Łojasiewicz 条件

这些条件通常不需要假设问题是凸问题。
还有更多的条件，误差界、二次增长条件。这些条件之间的关系，参考文献：Necoara, Ion, Yu Nesterov, and Francois Glineur. "Linear convergence of first order methods for non-strongly convex optimization." Mathematical Programming 175 (2019): 69-107.

例子：逻辑回归

假设我们获得了m个独立的二进制样本

$$y_i = \begin{cases} 1, & \text{以概率} \frac{1}{1+\exp(-a_i^T x)} \\ -1, & \text{以概率} \frac{1}{1+\exp(a_i^T x)} \end{cases}$$

其中$\{a_i\}$是已知的向量；$x \in \mathbb{R}^n$是未知参数。
最大似然估计（MLE）由以下给出（经过一些操作后）：

$$\min_{x\in\mathbb{R}^n} f(x) = \frac{1}{m}\sum_{i=1}^{m} \log(1+\exp(-y_i a_i^T x))$$

$$\nabla^2 f(x) = \frac{1}{m}\sum_{i=1}^{m} \frac{1}{(1+\exp(y_i a_i^T x))^2} a_i a_i^T \to 0 \text{ 如果} x \to \infty$$

这意味着f是0-强凸的。

局部强凸且光滑函数

定理 (局部强凸且光滑函数收敛速度)

设f 是局部μ-强凸且L-光滑的，满足

$$\mu I \preceq \nabla^2 f(x) \preceq LI, \quad \forall x \in B_0$$

其中$B_0 := \{x : \|x - x^*\|_2 \leq \|x_0 - x^*\|_2\}$ 且x^* 是最小值。那么收敛速度(5)继续成立。

在规则性条件下的收敛

另一种方法是通过以下规则性条件来替代强凸性和光滑性：

$$\langle \nabla f(x), x - x^* \rangle \geq \frac{\mu}{2}\|x - x^*\|_2^2 + \frac{1}{2L}\|\nabla f(x)\|_2^2, \quad \forall x \tag{6}$$

定理

假设f 满足上述条件(6)。如果$\alpha_k \equiv \eta = \frac{1}{L}$，那么

$$\|x_k - x^*\|_2^2 \leq \left(1 - \frac{\mu}{L}\right)^k \|x_0 - x^*\|_2^2$$

证明： 根据GD，我们有：

$$\|x_{k+1} - x^*\|_2^2 \leq \|x_k - x^*\|_2^2 - \frac{2}{L}\langle x_k - x^*, \nabla f(x_k)\rangle + \frac{1}{L^2}\|\nabla f(x_k)\|_2^2$$

进一步利用定理的条件：

$$\langle x_k - x^*, \nabla f(x_k)\rangle \geq \frac{\mu}{2}\|x_k - x^*\|_2^2 + \frac{1}{2L}\|\nabla f(x_k)\|_2^2$$

结合以上不等式，我们得到：

$$\|x_{k+1} - x^*\|_2^2 \leq \left(1 - \frac{\mu}{L}\right)\|x_k - x^*\|_2^2$$

Polyak-Łojasiewicz 条件下的收敛

另一个选择是Polyak-Łojasiewicz (PŁ) 条件：

$$\|\nabla f(x)\|_2^2 \geq 2\mu(f(x) - f(x^*)), \quad \forall x \tag{7}$$

该条件保证了一阶稳定点是全局最优点。

定理

假设f 满足(7)且是L-光滑的。如果$\alpha_k \equiv \eta = \frac{1}{L}$，则

$$f(x_k) - f(x^*) \leq \left(1 - \frac{\mu}{L}\right)^k (f(x_0) - f(x^*))$$

- 保证了向最优目标值的线性收敛
- 不意味着全局最小值的唯一性

证明：

$$\begin{aligned} f(x_{k+1}) - f(x^*) &\leq f(x_k) - f(x^*) - \frac{1}{2L}\|\nabla f(x_k)\|^2 \\ &\leq f(x_k) - f(x^*) - \frac{\mu}{L}(f(x_k) - f(x^*)) \\ &\leq \left(1 - \frac{\mu}{L}\right)(f(x_k) - f(x^*)) \end{aligned}$$

例子：过参数化线性回归

- m 个数据样本$\{a_i \in \mathbb{R}^n, y_i \in \mathbb{R}\}_{i=1}^m$
- 线性回归：找到一个最佳拟合数据的线性模型

$$\min_{x\in\mathbb{R}^n} f(x) = \frac{1}{2}\sum_{i=1}^{m}(a_i^T x - y_i)^2$$

过参数化：模型维度大于样本大小（即$n > m$）——这在深度学习中特别重要。
尽管这是一个凸问题，但如果$n > m$，则由于

$$\nabla^2 f(x) = \sum_{i=1}^{m} a_i a_i^T$$

是秩亏的，所以它不是强凸的。但在大多数“非退化”情况下，我们有$f(x^*) = 0$（为什么？），并且满足PŁ 条件，因此GD 线性收敛。

过参数化线性回归

假设$A = [a_1 \cdots a_m]^\top \in \mathbb{R}^{m\times n}$ 的秩为m，并且

$$\alpha_k \equiv \eta = \frac{1}{\lambda_{\max}(AA^T)}$$

那么GD 遵循

$$f(x_k) - f(x^*) \le \left(1 - \frac{\lambda_{\min}(AA^T)}{\lambda_{\max}(AA^T)}\right)^k (f(x_0) - f(x^*)), \quad \forall t$$

- 仅仅关于$\{a_i\}$ 非常温和的假设
- 没有关于$\{y_i\}$ 的假设
- 虽然这个过参数化问题存在许多全局最小值，但梯度下降有一个隐含的偏好——它倾向于收敛到离初始点x_0 最近的全局最小值。

过参数化线性回归收敛证明

为了证明线性收敛，关键是说明Polyak-Łojasiewicz (PŁ) 条件成立。我们需要证明以下不等式：

$$\|\nabla f(x)\|_2^2 \geq 2\lambda_{\min}(AA^T)f(x)$$

如果这成立，那么根据Polyak-Łojasiewicz (PŁ) 条件下的收敛定理和$f(x^*) = 0$ 的事实，立即可以得出结论。
为了证明上述不等式，考虑$y = [y_i]_{i=1}^m$ 并观察到

$$\nabla f(x) = A^T(Ax - y)$$

进一步有

$$\|\nabla f(x)\|_2^2 = (Ax-y)^T AA^T(Ax-y) \geq \lambda_{\min}(AA^T)\|Ax-y\|_2^2 = 2\lambda_{\min}(AA^T)f(x)$$

这满足PŁ 条件$\mu = \lambda_{\min}(AA^T)$。
这个证明基于A 的最小特征值和梯度的关系，利用了线性代数的基本性质来确立f 关于x 的梯度的下界。这表明，在特定的假设下，梯度下降算法能够以线性速率收敛到最优解。

梯度法的隐式正则

对于过参数化线性回归，若初始点$x_0 \in \text{range}(A^\top)$，那么，我们一定有$x_k$收敛到如下问题的解：

$$\min \|x\|^2, \quad \text{s.t.} \quad Ax = y. \tag{8}$$

- 因此，若想要线性回归的解范数较小，有时我们并不需要加上约束$\|x\| \le C$, 这种现象叫做梯度法的隐式正式(implicit regularization). 参考文献：Gunasekar, Suriya, et al. "Implicit regularization in matrix factorization." Advances in neural information processing systems 30 (2017).
- 更一般的现象存在于带有统计假设的非凸问题. 在没有显式正则化的情况下，梯度下降在各种统计模型下也隐式地执行了适当的正则化。实际上，梯度下降遵循的轨迹保持在一个具有良好几何性质的盆地内，由与采样机制不一致的点组成。参考文献：Ma, Cong, et al. "Implicit regularization in nonconvex statistical estimation: Gradient descent converges linearly for phase retrieval and matrix completion." International Conference on Machine Learning. PMLR, 2018.

证明：

- 梯度方向$\nabla f(x) = A^\top(Ax - y)$ 总是由m 个列向量a_i 张成的。
- 初始化$x_0 \in \text{range}(A^\top)$，我们因此将始终梯度法迭代保持在子空间$L = \{x = A^\top s | s \in \mathbb{R}^m\}$ 中。
- 现在考虑优化问题$\min_x \|x\|^2$，满足条件$Ax = y$。这个问题的KKT 最优性条件是$Ax = y$ 且存在ν 使得$x = A^\top \nu$。只要我们处于L 中，第二个条件就得到满足，如果我们收敛到零误差的全局最小值，则第一个条件也满足。
- 由于梯度下降保持在此子空间上，这就证明了如果梯度下降收敛到零误差解，它是最小二范数解。

提纲

非凸问题无处不在

很多机器学习最小化任务都是非凸的：

- 低秩矩阵补全
- 盲反卷积
- 字典学习
- 深度学习神经网络
- ...
- 可能到处都存在局部最小值
- 在一般情况下，没有算法可以有效地解决非凸问题

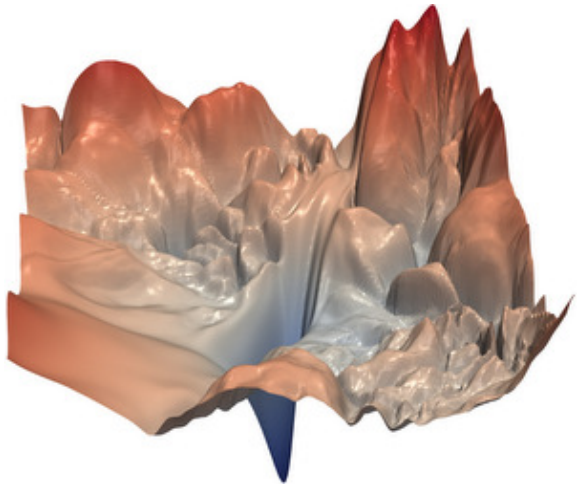


Figure: 神经网络ResNet56的损失函数局部图示。图片来源：https://github.com/tomgoldstein/loss-landscape

典型的收敛保证

我们不能期望通常有效地全局收敛到全局最小值，但我们可能有：

- 收敛到稳定点（即$\nabla f(x) = 0$）
- 收敛到局部最小值
- 在适当初始化时局部收敛到全局最小值

定理 (非凸光滑函数的收敛结果)

若f是$L-$光滑函数并且f有最小值f^*，则选取步长$\alpha_k = 1/L$, 我们有$\lim_{k\to\infty} \|\nabla f(x_k)\| = 0$并且并且对于任意正整数$T$，有

$$\min_{k=0,1,\ldots,T} \|\nabla f(x_k)\|^2 \leq \frac{2L(f(x_0) - f^*)}{T}.$$

- 梯度下降在$O(\frac{1}{\epsilon^2})$次迭代中找到一个ϵ-近似稳定点
- 这不意味着梯度下降收敛到稳定点；它只表明在梯度下降轨迹中存在近似稳定点

相关文献：Cartis, Coralia, Nick Gould, and Philippe Toint. "How much patience do you have? A worst-case perspective on smooth nonconvex optimization." Optima 88.1-10 (2012). & "How to make the gradients small," Y. Nesterov, Optima, 2012.

证明：

由李氏光滑性，$q_{x_k}(y) = f(x_k) + \nabla f(x_k)^T(y - x_k) + \frac{1}{2\alpha}\|y - x_k\|^2$为一个上界函数。梯度法迭代满足

$$x_{k+1} = \arg\min_y q_{x_k}(y) = x_k - 1/L\nabla f(x_k).$$

所以

$$f(x_{k+1}) \le q_{x_k}(x_{k+1}) = q_{x_k}(x_k) - \frac{1}{2L}\|\nabla f(x_k)\|^2 = f(x_k) - \frac{1}{2L}\|\nabla f(x_k)\|^2. \tag{9}$$

因此

$$\sum_{k=0}^{\infty} \frac{1}{2L}\|\nabla f(x_k)\|^2 < \infty.$$

故

$$\lim_{k\to\infty} \|\nabla f(x_k)\| = 0.$$

并且对于任意正整数T，有$\min_{k=0,1,\ldots,T} \|\nabla f(x_k)\|^2 \le \frac{2L(f(x_0)-f^*)}{T}$.

逃离鞍点

- 鞍点和局部最小值是两种具有消失梯度的点。鞍点看起来像“不稳定”的临界点；
- 我们可以避开鞍点吗?
- 梯度下降无法总是逃离鞍点。例如，如果x_0 恰好是一个鞍点，则梯度下降会陷入其中（因为$\nabla f(x_0) = 0$）。幸运的是，在温和的条件下，随机初始化的梯度下降几乎肯定会收敛到局部（有时甚至是全球）最小值！
  相关文献：”Gradient descent converges to minimizers,” J. Lee, M. Simchowitz,M. Jordan, B. Recht, COLT, 2016

示例

考虑一个简单的非凸二次最小化问题：

$$\min_x f(x) = \frac{1}{2} x^T A x$$

其中，$A = u_1 u_1^T - u_2 u_2^T$，$u_1^2 = u_2^2 = 1$ 且 $u_1 u_2 = 0$。
这个问题至少有一个鞍点：$x = 0$（为什么？）。

- 如果 $x_0 = 0$，那么梯度下降会卡在0（即，$x_k \equiv 0$）。
- 如果我们随机初始化梯度下降，我们可以希望避开鞍点吗？

Fact

如果$x_0 \sim \mathcal{N}(0, I)$，对于步长为$\alpha_k = \eta < 1$的梯度下降，以概率几乎为*1*，满足

$$\|x_k\|_2 \to \infty \text{ 当 } k \to \infty$$

- 有趣的是，梯度下降（几乎）从不会被困在鞍点0!

证明：
观察到

$$I - \eta A = I_\perp + (1-\eta)u_1 u_1^T + (1+\eta)u_2 u_2^T$$

其中$I_\perp := I - u_1 u_1^T - u_2 u_2^T$。可以容易地验证

$$(I - \eta A)^k = I_\perp + (1-\eta)^k u_1 u_1^T + (1+\eta)^k u_2 u_2^T$$

因此

$$x_k = (I - \eta A)^k x_0 = I_\perp x_0 + (1-\eta)^k (u_1^T x_0) u_1 + (1+\eta)^k (u_2^T x_0) u_2$$

显然，只要$\beta_0 = u_2^T x_0 \neq 0$，$|\beta_k| = (1+\eta)^k (u_2^T x_0) \to \infty$ 当$k \to \infty$，从而$\|x_k\|_2 \to \infty$ 几乎肯定发生。

次梯度方法

陈士祥

中国科学技术大学

致谢：本教案部分参考北京大学文再文教授《最优化与建模》

提纲

问题背景

许多优化问题，目标函数都是不可微的，例如前面我们见到的基追踪问题和矩阵补全问题，目标函数分别是最小化ℓ_1范数和矩阵变量的核范数。为了研究不可微时问题的最优条件，我们可以定义一般非光滑凸函数的次梯度。

次梯度

回顾可微凸函数f 的一阶等价条件:

$$f(y) \geq f(x) + \nabla f(x)^{\mathrm{T}}(y - x).$$

这表明，f 在点x 处的一阶近似是f 的一个全局下界。我们这里的想法是，将上述不等式拓展到一般不可微的情形。我们先考虑简单的函数$f(x) = |x|, x \in \mathbb{R}$. $f(x)$在$x = 0$处不可导，因为其左右导数分别为

$$\lim_{t \to 0^-} \frac{|t|}{t} = -1, \quad \lim_{t \to 0^+} \frac{|t|}{t} = 1.$$

可以验证,对于任意$g \in [-1, 1]$, 下面的不等式成立

$$|y| \geq 0 + g \cdot y,$$

此即

$$f(y) \geq f(0) + g \cdot (y - 0).$$

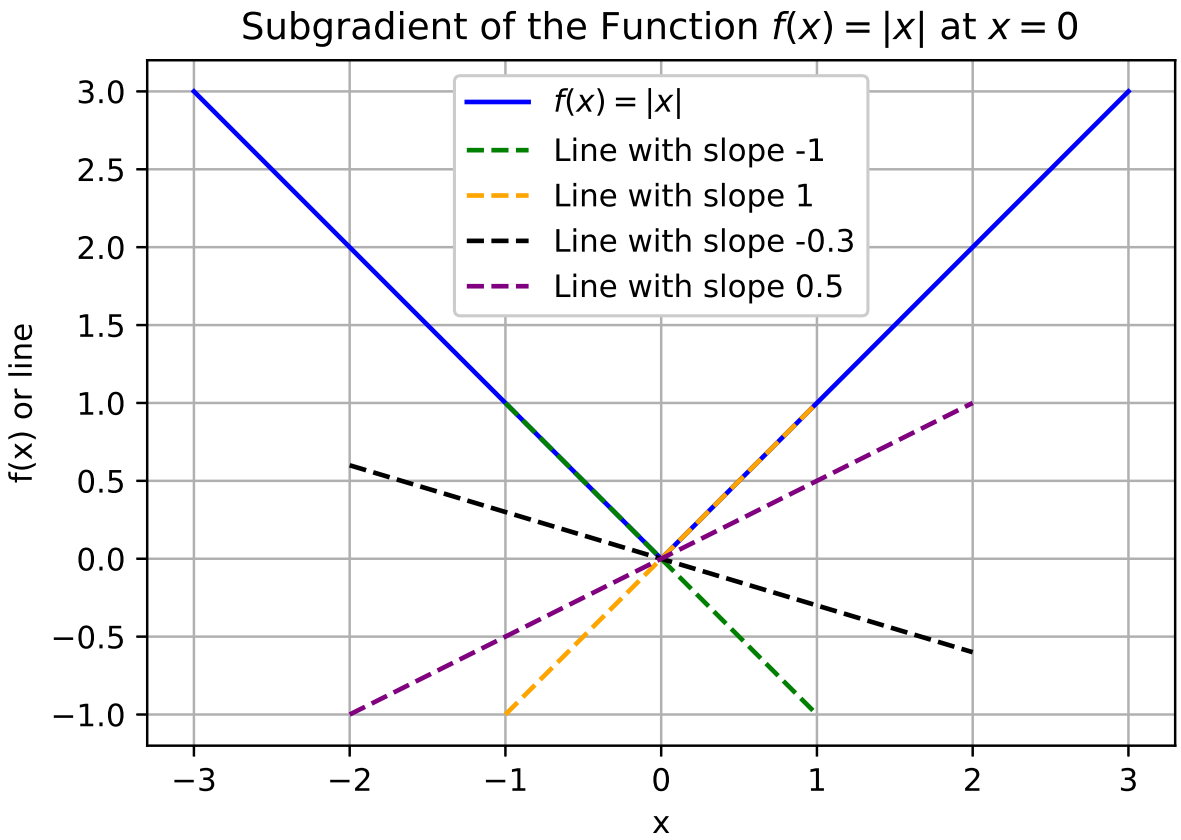


Figure: 函数$f(x) = |x|$次梯度示意图。任意斜率为$g \in [-1, 1]$过原点的直线，均为函数f的一个下界。

次梯度定义

Definition (次梯度和次微分)

设f为适当凸函数，x为定义域$\mathbf{dom}\, f$中的一点．若向量$g \in \mathbb{R}^n$满足

$$f(y) \geq f(x) + g^{\mathrm{T}}(y-x), \quad \forall y \in \mathbf{dom}\, f,$$

则称g为函数f在点x处的一个次梯度(subgradient)．进一步地，称集合

$$\partial f(x) = \{g \,|\, g \in \mathbb{R}^n, f(y) \geq f(x) + g^{\mathrm{T}}(y-x), \forall y \in \mathbf{dom}\, f\}$$

为f在点x处的次微分(subdifferential).

注

- 定义中的凸函数，值域可以为广义实数$\mathbb{R} \cup \{+\infty\}$空间。适当函数是指，存在$x$使得$f(x) < +\infty$.
- 由定义可知，次微分是一个集合，次梯度是某个次微分的元素。

次梯度存在性

当f可微时，我们有

$$f(x) + \nabla f(x)^T (y - x) \leq f(y) \leq z.$$

即

$$\begin{bmatrix} \nabla f(x) \\ -1 \end{bmatrix}^T \left(\begin{bmatrix} y \\ z \end{bmatrix} - \begin{bmatrix} x \\ f(x) \end{bmatrix} \right) \leq 0 \quad \forall\, (y,z) \in \mathbf{epi}\, f$$

这表明，$\nabla f(x)$ 可以诱导出上方图$\mathbf{epi}\, f$ 在点$(x, f(x))$ 处的支撑超平面，如下图所示。

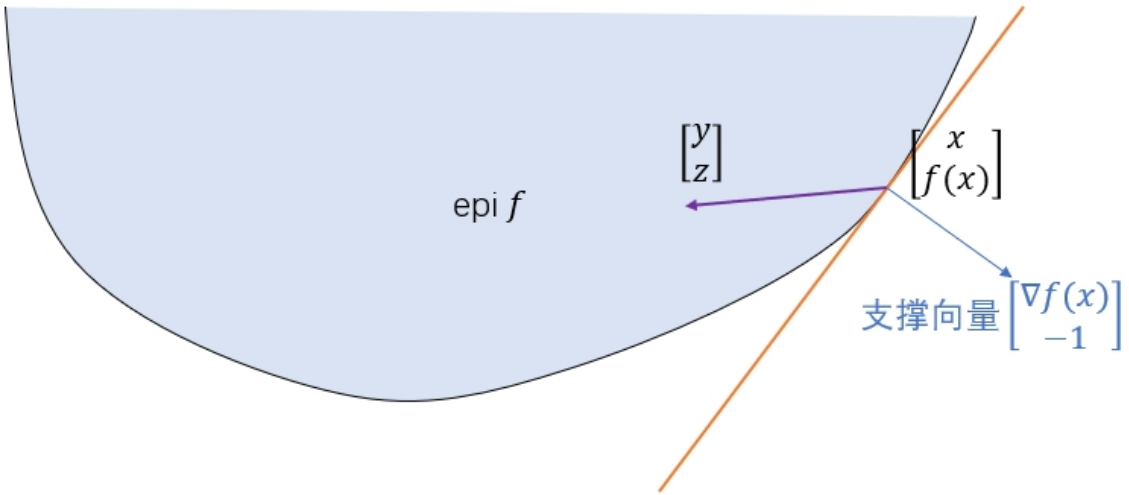


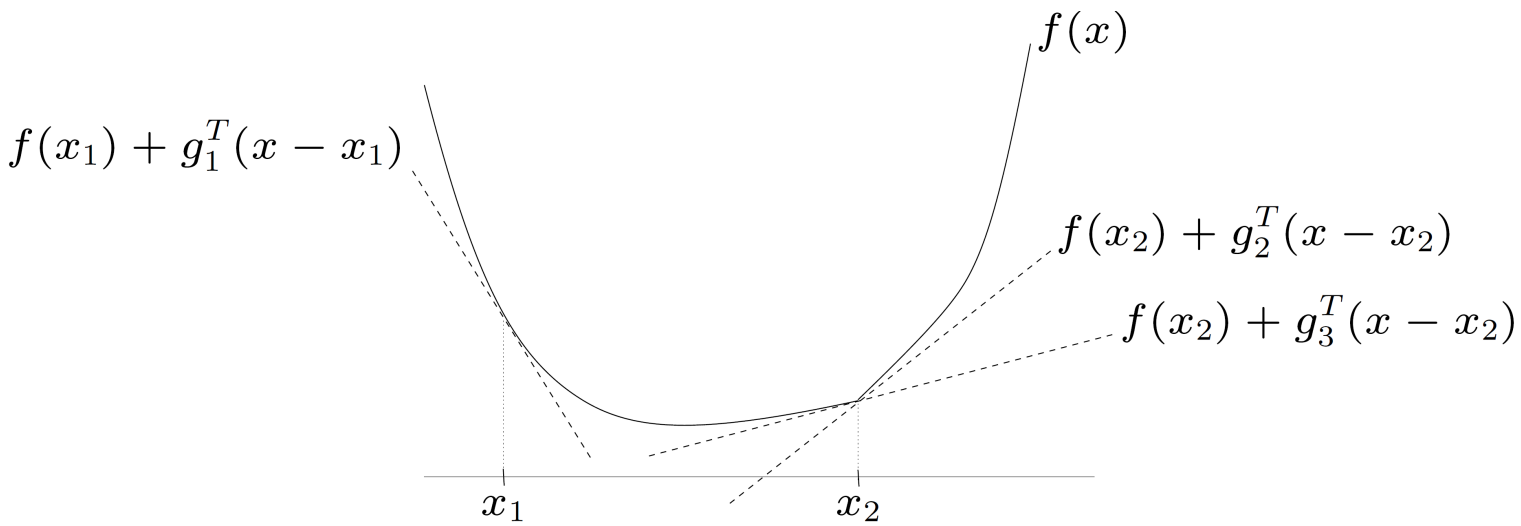
Figure: 对于凸函数$f(x)$,其上方图$\mathbf{epi}\, f$是一个凸集。$\begin{bmatrix} \nabla f(x) \\ -1 \end{bmatrix}$ 是$\mathbf{epi}\, f$的支撑向量。

次梯度

- $f(x)+g^{\mathrm{T}}(y-x)$ 是$f(y)$ 的一个全局下界
- g 可以诱导出上方图**epi** f 在点$(x,f(x))$ 处的一个支撑超平面

$$\begin{bmatrix} g \\ -1 \end{bmatrix}\left(\begin{bmatrix} y \\ t \end{bmatrix}-\begin{bmatrix} x \\ f(x) \end{bmatrix}\right)\leq 0 \quad \forall\,(y,t)\in \mathbf{epi}\, f$$

- 如果f 是可微凸函数, 那么$\nabla f(x)$ 是f 在点x 处的一个次梯度
- 例：g_2, g_3 是点x_2 处的次梯度；g_1 是点x_1 处的次梯度



次梯度存在性

设f为凸函数，$\mathbf{dom}\, f$为其定义域．如果$x \in \mathbf{int}\, \mathbf{dom}\, f$，则$\partial f(x)$是非空的，其中$\mathbf{int}\, \mathbf{dom}\, f$的含义是集合$\mathbf{dom}\, f$的所有内点.

证明：

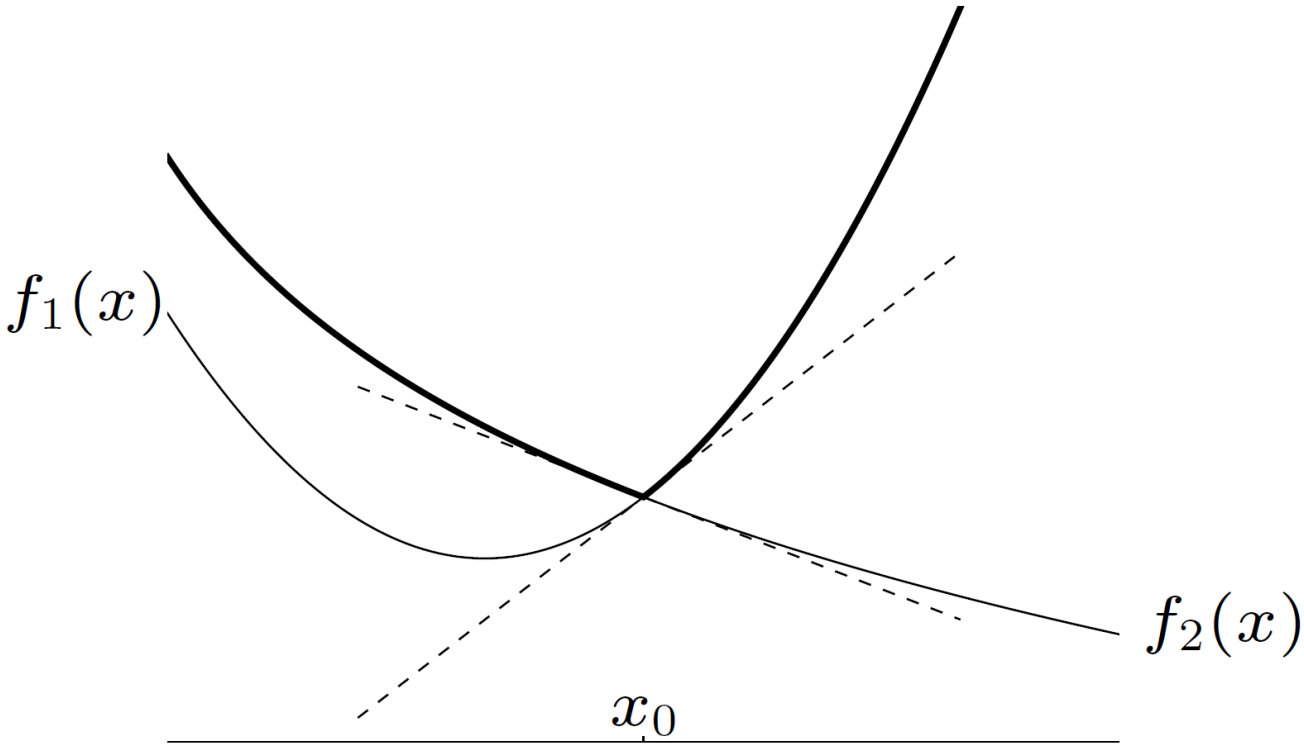
- $(x, f(x))$ 是$\mathbf{epi}\, f$ 边界上的点
- 因此存在$\mathbf{epi}\, f$ 在点$(x, f(x))$ 处的支撑超平面:

$$\exists\, (a,b) \neq 0, \quad \begin{bmatrix} a \\ b \end{bmatrix}^{\mathrm{T}} \left(\begin{bmatrix} y \\ t \end{bmatrix} - \begin{bmatrix} x \\ f(x) \end{bmatrix} \right) = a^{\top}(y-x) + b(t-f(x)) \leq 0, \forall\, (y,t) \in \mathbf{epi}\, f$$

- 令$t \to +\infty$, 可知$b \leq 0$
- 取$y = x + \epsilon a \in \mathbf{dom}\, f$, $\epsilon > 0$, 可知$b \neq 0$
- 因此$b < 0$ 并且$g = a/|b|$ 是f 在点x 处的次梯度

例

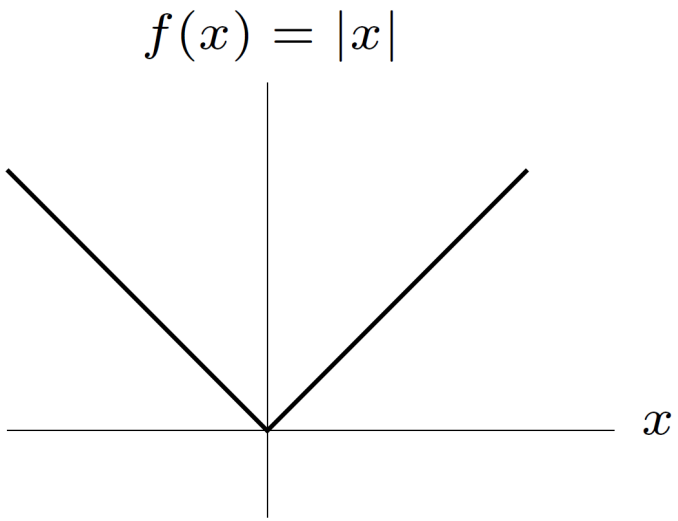
$$f(x) = \max\{f_1(x), f_2(x)\} \quad f_1, f_2 \text{ 是可微凸函数}$$

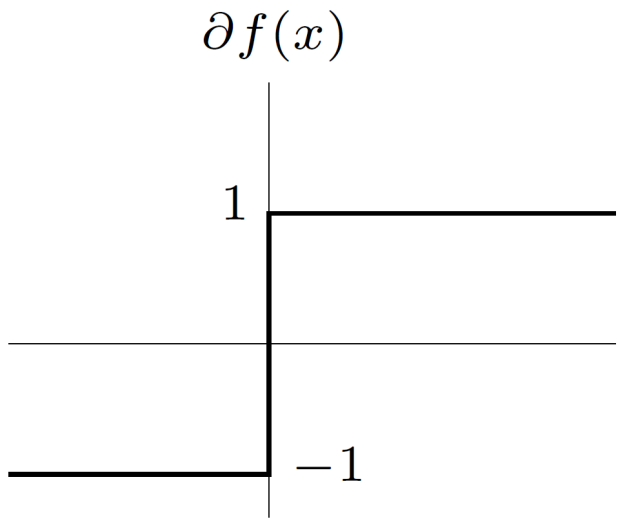


- 点x_0 处的次梯度可取范围$[\nabla f_1(x_0), \nabla f_2(x_0)]$
- 如果$f_1(\hat{x}) > f_2(\hat{x})$, f 在点$\hat{x}$ 处的次梯度等于$\nabla f_1(\hat{x})$
- 如果$f_1(\hat{x}) < f_2(\hat{x})$, f 在点$\hat{x}$ 处的次梯度等于$\nabla f_2(\hat{x})$

例

- 绝对值函数 $f(x) = |x|$





- 欧几里得范数 $f(x) = \|x\|_2$

$$\text{如果 } x \neq 0, \partial f(x) = \frac{1}{\|x\|_2} x, \quad \text{如果 } x = 0, \partial f(x) = \{g | \|g\|_2 \leq 1\}$$

例：非次可微函数

如下函数在点$x=0$处不是次可微的:

- $f:\mathbf{R}\to\mathbf{R},\mathbf{dom}\,f=\mathbf{R}_+$

$$x=0\text{ 时},\ f(x)=1,x>0\text{ 时},\ f(x)=0$$

- $f:\mathbf{R}\to\mathbf{R},\mathbf{dom}\,f=\mathbf{R}_+$

$$f(x)=-\sqrt{x},x\geq 0,\quad \text{否则}\ f(x)=+\infty.$$

$\mathbf{epi}\,f$ 在点$(0,f(0))$处的唯一支撑超平面是垂直的

提纲

次微分是闭凸集

对任何$x \in \mathbf{dom}\, f$，$\partial f(x)$ 是一个闭凸集（可能为空集）.

证明：

- 设$g_1, g_2 \in \partial f(x)$，并设$\lambda \in (0,1)$，由次梯度的定义

$$f(y) \geq f(x) + g_1^{\mathrm{T}}(y-x), \quad \forall y \in \mathrm{dom} f,$$
$$f(y) \geq f(x) + g_2^{\mathrm{T}}(y-x), \quad \forall y \in \mathrm{dom} f.$$

  由上面第一式的λ 倍加上第二式的$(1-\lambda)$ 倍，我们可以得到$\lambda g_1 + (1-\lambda) g_2 \in \partial f(x)$，从而$\partial f(x)$ 是凸集.

- 令$g_k \in \partial f(x)$为次梯度且$g_k \to g$，则

$$f(y) \geq f(x) + g_k^{\mathrm{T}}(y-x), \quad \forall y \in \mathrm{dom} f,$$

  在上述不等式中取极限，并注意到极限的保号性，最终我们有

$$f(y) \geq f(x) + g^{\mathrm{T}}(y-x), \quad \forall y \in \mathrm{dom} f.$$

  这说明$\partial f(x)$为闭集.

内点的次微分非空有界

如果$x \in \mathbf{int}\,\mathbf{dom}\, f$，则$\partial f(x)$ 非空有界集.

证明：

- 非空可由次梯度存在性直接得出
- 取充分小的$r > 0$, 使得

$$B = \{x \pm re_i | i = 1, \cdots, n\} \subset \mathbf{dom}\, f$$

- 对任意非零的$g \in \partial f(x)$, 存在$y \in B$ 满足

$$f(y) \geq f(x) + g^{\mathrm{T}}(y - x) = f(x) + r\|g\|_\infty$$

- 由此得到$\partial f(x)$ 有界:

$$\|g\|_\infty \leq \frac{\max_{y \in B} f(y) - f(x)}{r} < +\infty$$

可微函数的次微分

设凸函数$f(x)$在$x_0 \in \mathbf{int}\ \mathbf{dom}\ f$处可微，则$\partial f(x_0) = \{\nabla f(x_0)\}$.

证明：

- 根据可微凸函数的一阶条件可知梯度$\nabla f(x_0)$为次梯度.
- 下证$f(x)$在点x_0处不可能有其他次梯度. 设$g \in \partial f(x_0)$，根据次梯度的定义，对任意的非零$v \in \mathbb{R}^n$且$x_0 + tv \in \mathbf{dom}\ f, t > 0$有

$$f(x_0 + tv) \geq f(x_0) + tg^{\mathrm{T}}v.$$

  若$g \neq \nabla f(x_0)$，取$v = g - \nabla f(x_0) \neq 0$，上式变形为

$$\frac{f(x_0 + tv) - f(x_0) - t\nabla f(x_0)^{\mathrm{T}}v}{t\|v\|} \geq \frac{(g - \nabla f(x_0))^{\mathrm{T}}v}{\|v\|} = \|v\|.$$

- 不等式两边令$t \to 0$，根据Fréchet 可微的定义，左边趋于0，而右边是非零正数，可得到矛盾.

次梯度的单调性

设$f : \mathbb{R}^n \to \mathbb{R}$为凸函数，$x, y \in \mathbf{dom}\, f$，则$(u-v)^{\mathrm{T}}(x-y) \geq 0$, 其中$u \in \partial f(x)$，$v \in \partial f(y)$.

证明：

- 由次梯度的定义，

$$f(y) \geq f(x) + u^{\mathrm{T}}(y-x),$$
$$f(x) \geq f(y) + v^{\mathrm{T}}(x-y).$$

- 将以上两个不等式相加即得结论.

次梯度的连续性

设$f(x)$是闭凸函数且∂f在点$\bar{x}$附近存在且非空. 若序列$x^k \to \bar{x}$，$g^k \in \partial f(x^k)$为$f(x)$在点x^k处的次梯度，且$g^k \to \bar{g}$，则$\bar{g} \in \partial f(\bar{x})$.

证明：

- 对任意$y \in \mathrm{dom} f$，根据次梯度的定义，

$$f(y) \geq f(x^k) + \left\langle g^k, y - x^k \right\rangle.$$

- 对上述不等式两边取下极限，我们有

$$\begin{aligned} f(y) &\geq \liminf_{k\to\infty} [f(x^k) + \left\langle g^k, y - x^k \right\rangle] \\ &\geq f(\bar{x}) + \langle \bar{g}, y - \bar{x} \rangle, \end{aligned}$$

其中第二个不等式利用了$f(x)$的下半连续性以及$g^k \to \bar{g}$，由此可推出$\bar{g} \in \partial f(\bar{x})$.

提纲

方向导数

- 一般函数：设f为适当函数，给定点x_0以及方向$d \in \mathbb{R}^n$，方向导数（若存在）定义为

$$\lim_{t \downarrow 0} \phi(t) = \lim_{t \downarrow 0} \frac{f(x_0 + td) - f(x_0)}{t},$$

  其中$t \downarrow 0$表示t单调下降趋于0.

- 凸函数：易知$\phi(t)$ 在$(0, +\infty)$上是单调不减的，上式中的极限号$\lim$可以替换为下确界$\inf$. 上述此时极限总是存在（可以为无穷），进而凸函数总是可以定义方向导数.

- 方向导数的定义：对于凸函数f，给定点$x_0 \in \mathbf{dom}\, f$ 以及方向$d \in \mathbb{R}^n$，其方向导数定义为

$$\partial f(x_0; d) = \inf_{t>0} \frac{f(x_0 + td) - f(x_0)}{t}.$$

方向导数有限

设$f(x)$为凸函数，$x_0 \in \mathbf{int\ dom}\, f$，则对任意$d \in \mathbb{R}^n$，$\partial f(x_0; d)$有限.

证明：

- 首先$\partial f(x_0; d)$不为正无穷是显然的.
- 由于$x_0 \in \mathbf{int\ dom}\, f$，根据次梯度的存在性定理可知$f(x)$在点$x_0$处存在次梯度$g$.
- 根据方向导数的定义，我们有

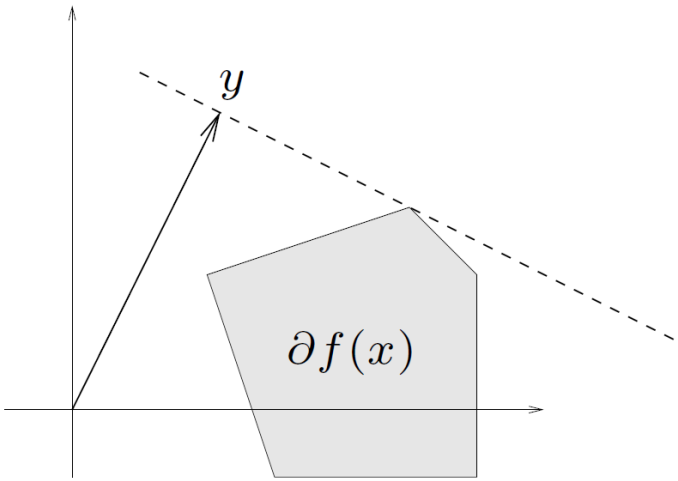
$$\begin{aligned}\partial f(x_0; d) &= \inf_{t>0} \frac{f(x_0 + td) - f(x_0)}{t} \\ &\geq \inf_{t>0} \frac{tg^{\mathrm{T}}d}{t} = g^{\mathrm{T}}d.\end{aligned}$$

  其中的不等式利用了次梯度的定义.
- 这说明$\partial f(x_0; d)$不为负无穷.

方向导数和次梯度

设$f:\mathbb{R}^n \to (-\infty,+\infty]$为凸函数，点$x_0 \in \mathbf{int\ dom}\, f$，$d$为$\mathbb{R}^n$中任一方向，则

$$\partial f(x_0;d) = \max_{g\in\partial f(x_0)} g^{\mathrm{T}} d.$$



$\partial f(x;y)$是$\partial f(x)$的支撑函数

- 对于可微函数, $\partial f(x_0;d) = \nabla f(x_0)^{\mathrm{T}} d$
- 这也说明$\partial f(x_0;d)$对所有的$x_0 \in \mathbf{int\ dom}\, f$, 以及所有的$d$都存在

证明：

- 记$q(v) = \partial f(x_0; v)$. 根据方向导数有限性的证明过程可得，

$$q(d) = \partial f(x_0; d) \geq g^{\mathrm{T}} d, \ \forall g \in \partial f(x_0).$$

这说明$\partial f(x_0; d)$是$g^{\mathrm{T}} d$的一个上界，接下来说明该上界为上确界.

- 构造函数

$$h(v, t) = t\left(f\left(x_0 + \frac{v}{t}\right) - f(x_0)\right),$$

可知$h(v, t)$为$\tilde{f}(v) = f(x_0 + v) - f(x_0)$的透视函数，并且

$$q(v) = \inf_{t'>0} \frac{f(x_0 + t'v) - f(x_0)}{t'} \overset{t=1/t'}{=} \inf_{t>0} h(v, t).$$

由于透视函数$h(v, t)$为凸函数，取下确界仍为凸函数，因此$q(v)$关于v是凸函数.

- 由方向导数的有限性直接可以得出$\mathbf{dom}\ q = \mathbb{R}^n$，因此$q(v)$在全空间任意一点次梯度存在.

- 对方向d，设$\hat{g} \in \partial q(d)$，则对任意$v \in \mathbb{R}^n$以及$\lambda \geq 0$，我们有

$$\lambda q(v) = q(\lambda v) \geq q(d) + \hat{g}^{\mathrm{T}}(\lambda v - d).$$

- 令$\lambda = 0$，我们有$q(d) \leq \hat{g}^{\mathrm{T}} d$；令$\lambda \to +\infty$，我们有

$$q(v) \geq \hat{g}^{\mathrm{T}} v,$$

- 由$\phi(t)$单调不减，$\phi(1) \geq \lim\limits_{t \downarrow 0} \phi(t)$,即

$$f(x_0 + v) \geq f(x_0) + q(v) \geq f(x) + \hat{g}^{\mathrm{T}} v.$$

这说明$\hat{g} \in \partial f(x)$且$\hat{g}^{\mathrm{T}} d \geq q(d)$. 即$q(d)$为$g^{\mathrm{T}} d$ 的上确界，且当$g = \hat{g}$时上确界达到.

更一般的推广

- 设f为适当凸函数，且在x_0处次微分不为空集，则对任意$d \in \mathbb{R}^n$有

$$\partial f(x_0; d) = \sup_{g \in \partial f(x_0)} g^{\mathrm{T}} d,$$

且当$\partial f(x_0; d)$不为无穷时，上确界可以取到.

提纲

次梯度的计算规则

弱次梯度计算：得到一个次梯度

- 足以满足大多数不可微凸函数优化算法
- 如果可以获得任意一点处$f(x)$的值，那么总可以计算一个次梯度

强次梯度计算：得到$\partial f(x)$，即所有次梯度

- 一些算法、最优性条件等，需要完整的次微分
- 计算可能相当复杂

下面我们假设$x \in \mathbf{int}\ \mathbf{dom}\ f$

基本规则

- 可微凸函数： 若凸函数f 在点x 处可微，则$\partial f(x) = \{\nabla f(x)\}$.
- 凸函数的非负线性组合：设凸函数f_1, f_2 满足$\mathbf{int}\,\mathbf{dom}\, f_1 \cap \mathbf{dom}\, f_2 \neq \emptyset$，而$x \in \mathbf{dom}\, f_1 \cap \mathbf{dom}\, f_2$. 若

$$f(x) = \alpha_1 f_1(x) + \alpha_2 f_2(x), \quad \alpha_1, \alpha_2 \geq 0,$$

则$f(x)$的次微分

$$\partial f(x) = \alpha_1 \partial f_1(x) + \alpha_2 \partial f_2(x).$$

- 线性变量替换： 设h 为适当凸函数，f 满足$f(x) = h(Ax + b)$. 若存在$x^\sharp \in \mathbb{R}^m$，使得$Ax^\sharp + b \in \mathbf{int}\,\mathbf{dom}\, h$，则

$$\partial f(x) = A^{\mathrm{T}} \partial h(Ax + b), \quad \forall\, x \in \mathbf{int}\,\mathbf{dom}\, f.$$

函数族的上确界

设$f_1, f_2, \cdots, f_m : \mathbb{R}^n \to (-\infty, +\infty]$ 均为凸函数，令

$$f(x) = \max\{f_1(x), f_2(x), \cdots, f_m(x)\}, \quad \forall x \in \mathbb{R}^n.$$

对$x_0 \in \bigcap_{i=1}^{m} \mathbf{int\ dom}\, f_i$，定义$I(x_0) = \{i \mid f_i(x_0) = f(x_0)\}$，则

$$\partial f(x_0) = \mathbf{conv} \bigcup_{i \in I(x_0)} \partial f_i(x_0).$$

- $I(x_0)$表示点x_0 处“有效”函数的指标
- $\partial f(x_0)$ 是点x_0 处“有效”函数的次微分并集的凸包
- 如果f_i 可微, $\partial f(x_0) = \mathbf{conv}\{\nabla f_i(x_0) \mid i \in I(x_0)\}$

证明：

- 若$f(x_0) = +\infty$，则$f_i(x_0) = +\infty, i \in I(x_0)$，于是等式两端均为$\emptyset$.
- 下设$f(x_0) < +\infty$. $\forall i \in I(x_0)$，容易验证$\partial f_i(x_0) \subseteq \partial f(x_0)$. 再由次微分是闭凸集可知

$$\mathbf{conv} \bigcup_{i \in I(x_0)} \partial f_i(x_0) \subseteq \partial f(x_0).$$

- 另一方面，设$g \in \partial f(x_0)$. 假设$g \notin \mathbf{conv} \bigcup_{i \in I(x_0)} \partial f_i(x_0)$，由严格分离定理（注意到$\mathbf{conv} \bigcup_{i \in I(x_0)} \partial f_i(x_0)$和$\{g\}$均为闭凸集）和方向导数与次梯度的关系，存在$a \in \mathbb{R}^n$和$b \in \mathbb{R}$，使得

$$a^{\mathrm{T}} g > b \geq \max_{i \in I(x_0)} \sup_{\xi \in \partial f_i(x_0)} a^{\mathrm{T}} \xi = \max_{i \in I(x_0)} \partial f_i(x_0; a).$$

- 因为

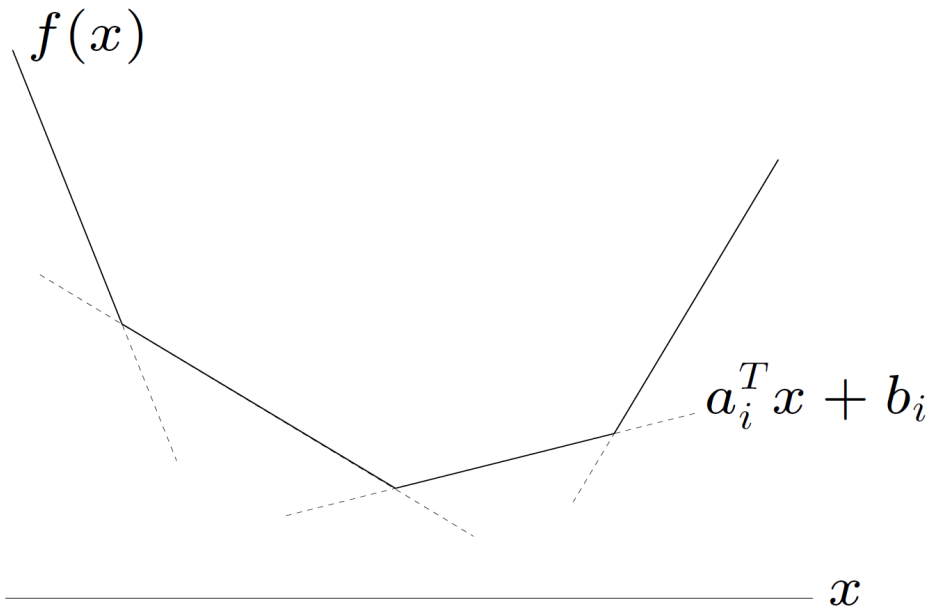
$$\begin{aligned}\partial f(x_0;a) &= \lim_{t\to 0^+}\frac{f(x_0+ta)-f(x_0)}{t}\\ &= \max_{i\in I(x_0)}\lim_{t\to 0^+}\frac{f_i(x_0+ta)-f_i(x_0)}{t}\\ &= \max_{i\in I(x_0)}\partial f_i(x_0;a).\end{aligned}$$

  故$a^{\mathrm{T}}g > \partial f(x_0;a)$.

- 但由于$g\in\partial f(x_0)$，我们有$f(x_0+ta)\geq f(x_0)+tg^{\mathrm{T}}a$，因而$\partial f(x_0;a)\geq a^{\mathrm{T}}g$，这就导致矛盾．故$g\in\mathbf{conv}\bigcup_{i\in I(x_0)}\partial f_i(x_0)$.

例：分段线性函数

$$f(x) = \max_{i=1,2,\cdots,m} \{a_i^{\mathrm{T}} x + b_i\}$$



- 点x处的次微分是一个多面体

$$\partial f(x) = \mathbf{conv}\{a_i \mid i \in I(x)\}$$

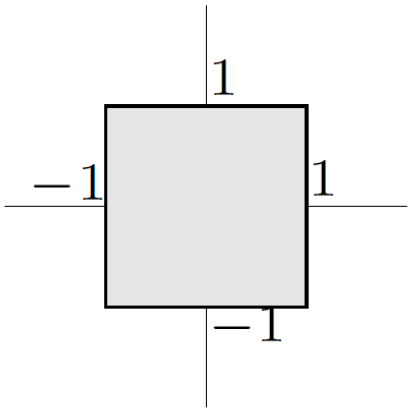
其中$I(x) = \{i \mid a_i^{\mathrm{T}} x + b_i = f(x)\}$

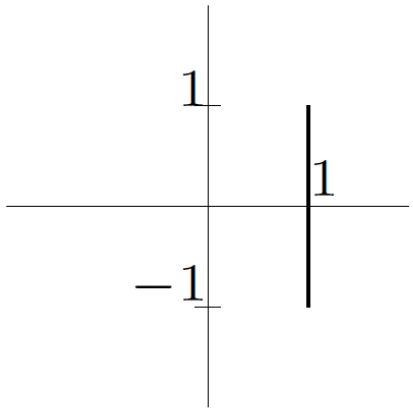
例：ℓ_1-范数

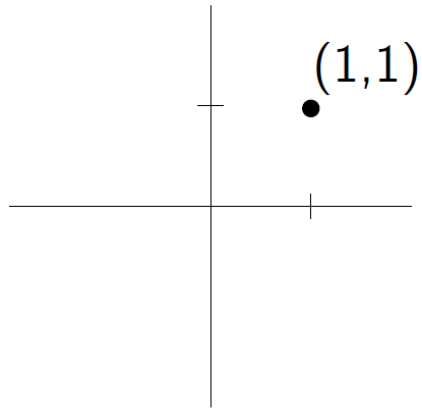
$$f(x) = \|x\|_1 = \max_{s\in\{-1,1\}^n} s^{\mathrm{T}}x$$

- 次微分是区间的乘积

$$\partial f(x) = J_1 \times \cdots \times J_n, \quad J_k = \begin{cases} [-1,1], & x_k = 0 \\ \{1\}, & x_k > 0 \\ \{-1\}, & x_k < 0 \end{cases}$$







$\partial f(0,0) = [-1,1] \times [-1,1]$ $\partial f(1,0) = \{1\} \times [-1,1]$ $\partial f(1,1) = \{(1,1)\}$

逐点上确界函数

设$\{f_\alpha \mid \mathbb{R}^n \to (-\infty, +\infty]\}_{\alpha\in\mathcal{A}}$ 是一族凸函数，令

$$f(x) = \sup_{\alpha\in\mathcal{A}} f_\alpha(x).$$

- 对$x_0 \in \cap_{\alpha\in\mathcal{A}} \mathbf{int\ dom}\, f_\alpha$，定义$I(x_0) = \{\alpha \in \mathcal{A} \mid f_\alpha(x_0) = f(x_0)\}$，则

$$\mathbf{conv} \bigcup_{\alpha\in I(x_0)} \partial f_\alpha(x_0) \subseteq \partial f(x_0).$$

- 如果还有$\mathcal{A}$ 是紧集且f_α 关于α 连续，则

$$\mathbf{conv} \bigcup_{\alpha\in I(x_0)} \partial f_\alpha(x_0) = \partial f(x_0).$$

例：最大特征值函数

$A(x) = A_0 + x_1 A_1 + \cdots + x_n A_n$ 并且系数A_i 对称，令

$$f(x) = \lambda_{\max}(A(x)) = \sup_{\|y\|_2=1} y^{\mathrm{T}} A(x) y$$

计算点$\hat{x}$ 处的一个次梯度：

- 选择特征值$\lambda_{\max}(A(\hat{x}))$ 对应的任一单位特征向量y
- $y^{\mathrm{T}} A(x) y$ 在点$\hat{x}$ 处的梯度是f 的一个次梯度：

$$(y^{\mathrm{T}} A_1 y, \cdots, y^{\mathrm{T}} A_n y) \in \partial f(\hat{x})$$

固定分量的函数极小值

$$f(x) = \inf_y h(x,y), \quad h \text{ 关于 } (x,y) \text{ 联合凸}$$

计算点$\hat{x}$处的一个次梯度：

- 设$\hat{y} \in \mathbb{R}^m$ 满足$h(\hat{x},\hat{y}) = f(\hat{x})$
- 存在$g \in \mathbb{R}^n$ 使得$(g,0) \in \partial h(\hat{x},\hat{y})$，则$g \in \partial f(\hat{x})$

证明：对任意$x \in \mathbb{R}^n, y \in \mathbb{R}^m$

$$\begin{aligned} h(x,y) &\geq h(\hat{x},\hat{y}) + g^{\mathrm{T}}(x-\hat{x}) + 0^{\mathrm{T}}(y-\hat{y}) \\ &= f(\hat{x}) + g^{\mathrm{T}}(x-\hat{x}) \end{aligned}$$

于是

$$f(x) = \inf_y h(x,y) \geq f(\hat{x}) + g^{\mathrm{T}}(x-\hat{x})$$

例: 距离函数

设C是$\mathbb{R}^n$中一闭凸集，令

$$f(x) = \inf_{y \in C} \|x - y\|_2$$

计算点$\hat{x}$处的一个次梯度：

- 若$f(\hat{x}) = 0$，则容易验证$g = 0 \in \partial f(\hat{x})$；
- 若$f(\hat{x}) > 0$，取$\hat{y}$为$\hat{x}$在C上的投影，即$\hat{y} = \mathcal{P}_c(\hat{x})$，
  首先$f(\hat{x}) = \|\hat{x} - \mathcal{P}_c(\hat{x})\| = \|\hat{x} - \hat{y}\|,$；另外，可以验证

$$(g, 0) \in \partial h(\hat{x}, \hat{y}),$$

其中

$$g = \frac{1}{\|\hat{x} - \hat{y}\|_2}(\hat{x} - \hat{y}) = \frac{1}{\|\hat{x} - \mathcal{P}_c(\hat{x})\|_2}(\hat{x} - \mathcal{P}_c(\hat{x}))$$

复合函数

设$f_1, f_2, \cdots, f_m : \mathbb{R}^n \to (-\infty, +\infty]$ 为m 个凸函数，$h : \mathbb{R}^m \to (-\infty, +\infty]$ 为关于各分量单调递增的凸函数，令

$$f(x) = h(f_1(x), f_2(x), \cdots, f_m(x)).$$

计算点$\hat{x}$ 处的一个次梯度:

- $z = (z_1, z_2, \cdots, z_m) \in \partial h(f_1(\hat{x}), f_2(\hat{x}), \cdots, f_m(\hat{x}))$ 以及$g_i \in \partial f_i(\hat{x})$
- $g \overset{\text{def}}{=\!=} z_1 g_1 + z_2 g_2 + \cdots + z_m g_m \in \partial f(\hat{x})$

证明:

$$\begin{aligned} f(x) &\geq h\left(f_1(\hat{x}) + g_1^{\mathrm{T}}(x - \hat{x}), f_2(\hat{x}) + g_2^{\mathrm{T}}(x - \hat{x}), \cdots, f_m(\hat{x}) + g_m^{\mathrm{T}}(x - \hat{x})\right) \\ &\geq h(f_1(\hat{x}), f_2(\hat{x}), \cdots, f_m(\hat{x})) + \sum_{i=1}^{m} z_i g_i^T (x - \hat{x}) \\ &= f(\hat{x}) + g^{\mathrm{T}}(x - \hat{x}), \end{aligned}$$

设函数f_i是凸函数，定义$h(u,v)$为如下凸问题的最优值

$$\begin{aligned} \min_x \quad & f_0(x) \\ \text{s.t.} \quad & f_i(x) \le u_i,\ i=1,\cdots,m \\ & Ax = b + v \end{aligned}$$

计算点$(\hat{u},\hat{v})$处的一个次梯度:

- 假设$h(\hat{u},\hat{v})$有限, 强对偶成立

$$\begin{aligned} \max \quad & \inf_x \left(f_0(x) + \sum_i \lambda_i (f_i(x) - \hat{u}_i) + w^{\mathrm{T}}(Ax - b - \hat{v}) \right) \\ \text{s.t.} \quad & \lambda \ge 0 \end{aligned}$$

- 如果$\hat{\lambda},\hat{w}$是最优对偶变量，那么$(-\hat{\lambda},-\hat{w}) \in \partial h(\hat{u},\hat{v})$

证明:由弱对偶原理可得

$$
\begin{aligned}
h(u,v) &\geq \inf_x \left(f_0(x) + \sum_i \hat{\lambda}_i (f_i(x) - u_i) + \hat{w}^{\mathrm{T}}(Ax - b - v) \right) \\
&= \inf_x \left(f_0(x) + \sum_i \hat{\lambda}_i (f_i(x) - \hat{u}_i) + \hat{w}^{\mathrm{T}}(Ax - b - \hat{v}) \right) \\
&- \hat{\lambda}^{\mathrm{T}}(u - \hat{u}) - \hat{w}^{\mathrm{T}}(v - \hat{v}) \\
&= h(\hat{u}, \hat{v}) - \hat{\lambda}^{\mathrm{T}}(u - \hat{u}) - \hat{w}^{\mathrm{T}}(v - \hat{v})
\end{aligned}
$$

期望函数

u是一个随机变量，h是关于x的凸函数，令

$$f(x) = \mathbb{E}h(x,u)$$

计算点$\hat{x}$处的一个次梯度：

- 选择一个函数g满足$g(u) \in \partial_x h(\hat{x},u)$
- $g = \mathbb{E}_u\, g(u) \in \partial f(\hat{x})$

证明：由h的凸性和$g(u)$的定义，

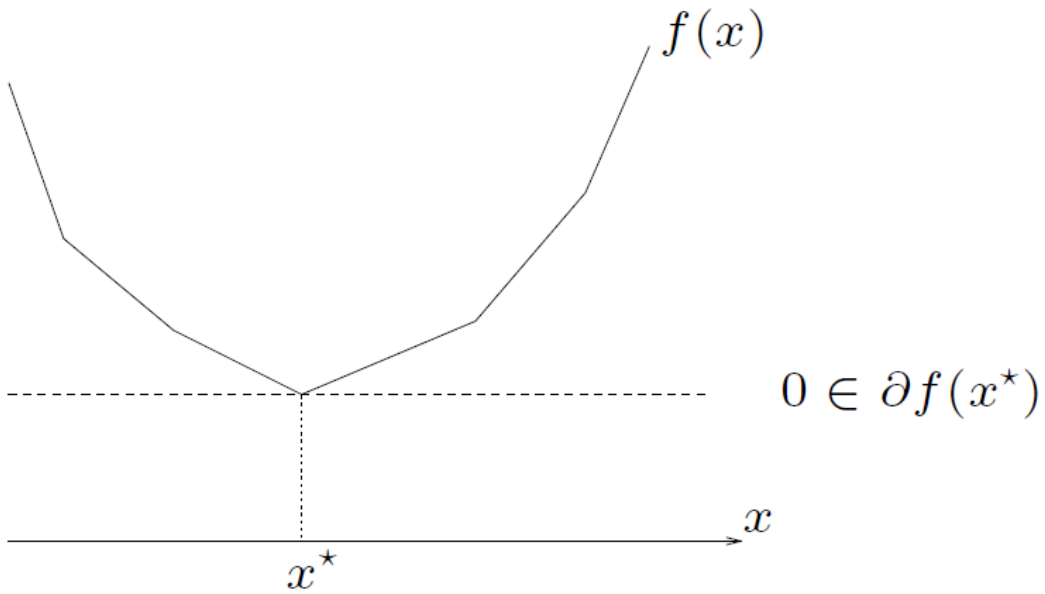
$$\begin{aligned} f(x) &= \mathbb{E}h(x,u) \\ &\geq \mathbb{E}\left(h(\hat{x},u) + g(u)^{\mathrm{T}}(x-\hat{x})\right) \\ &= f(\hat{x}) + g^{\mathrm{T}}(x-\hat{x}) \end{aligned}$$

提纲

定理

x^* 是$f(x)$ 的极小点当且仅当

$$0 \in \partial f(x^*)$$



证明：根据定义

$$f(y) \geq f(x^*) + 0^{\mathrm{T}}(y - x^*),\ \forall y \quad \Leftrightarrow \quad 0 \in \partial f(x^*)$$

定理

$$\begin{aligned} \min \quad & f_0(x) \\ \textit{s.t.} \quad & f_i(x) \leq 0, \quad i = 1, \cdots, m \end{aligned}$$

如果强对偶成立, 那么x^*, λ^* 是最优原始、对偶变量当且仅当

1. x^* 是可行的
2. $\lambda^* \geq 0$
3. $\lambda_i^* f_i(x^*) = 0\,, i = 1, \cdots, m$
4. x^* 是下式的一个极小值点

$$L(x, \lambda^*) = f_0(x) + \sum_{i=1}^{m} \lambda_i^* f_i(x)$$

- 当$\mathbf{dom}\, f_i = \mathbf{R}^n$时，Karush-Kuhn-Tucker 条件为
  条件1, 2, 3 以及

$$0 \in \partial L_x(x^*, \lambda^*) = \partial f_0(x^*) + \sum_{i=1}^{m} \lambda_i^* \partial f_i(x^*).$$

- 对于可微函数f_i, 上式成为

$$0 = \nabla f_0(x^*) + \sum_{i=1}^{m} \lambda_i^* \nabla f_i(x^*)$$

提纲

非光滑优化的例子

- 极小极大问题：

$$\min_{x\in X}\ \max_{1\le i\le m}\ f_i(x)$$

- 求解非线性方程组：

$$f_i(x)=0,\quad i=1,\cdots,m$$

可以把它化为一个极小化问题：

$$\min_{x\in X}\ \|\,(f_1(x),\cdots,f_m(x))\,\|$$

特别地，$\|\cdot\|=\|\cdot\|_1$ 对应 L_1 极小化问题，$\|\cdot\|=\|\cdot\|_\infty$对应切比雪夫近似问题.

- LASSO问题：

$$\min_{x}\ \|Ax-b\|^2+\mu\|x\|_1$$

梯度下降法失败的例子

考虑函数 $f:\mathbb{R}^2\to\mathbb{R}^1, x=(u,v)^{\mathrm{T}}$,

$$f(x)=\max\left[\frac{1}{2}u^2+(v-1)^2,\frac{1}{2}u^2+(v+1)^2\right].$$

- 假设迭代点 x^k 的形式为

$$x^k=\begin{pmatrix}2(1+|\epsilon_k|)\\ \epsilon_k\end{pmatrix},\quad \text{其中}\ \epsilon_k\neq 0.$$

- 可以计算迭代点 x^k 处的梯度：

$$\nabla f\left(x^k\right)=\begin{pmatrix}2(1+|\epsilon_k|)\\ 2(1+|\epsilon_k|)t_k\end{pmatrix}=2(1+|\epsilon_k|)\begin{pmatrix}1\\ t_k\end{pmatrix},$$

其中 $t_k=\operatorname{sign}(\epsilon_k)$.

梯度下降法失败的例子

下面我们考虑直接用梯度下降法进行迭代.

- 在负梯度方向$-\nabla f\left(x^k\right)$上做精确线搜索，可得

$$x^{k+1}=x^k+\alpha_k\left(-\nabla f\left(x^k\right)\right)=\begin{bmatrix}2\left(1+\left|\epsilon_k\right|/3\right)\\ -\epsilon_k/3\end{bmatrix}=\begin{bmatrix}2\left(1+\left|\epsilon_{k+1}\right|\right)\\ \epsilon_{k+1}\end{bmatrix}$$

  其中$\epsilon_{k+1}=-\epsilon_k/3\neq 0$. 所以显然有$\epsilon_k\to 0$.

- 给定一个初始点$x^0=(2+2|\delta|,\delta)^{\mathrm{T}}$，我们有$x^k\to(2,0)^{\mathrm{T}}$.
- 然而$(2,0)^{\mathrm{T}}$并不是稳定点.
- 这表明对非光滑问题直接使用梯度法可能会收敛到一个非稳定点.

提纲

问题设定

假设$f(x)$为凸函数，但不一定可微，考虑如下问题：

$$\min_x f(x)$$

- 一阶充要条件：

$$x^* \text{是一个全局极小点} \quad \Leftrightarrow \quad 0 \in \partial f(x^*)$$

- 因此可以通过计算凸函数的次梯度集合中包含0的点来求解其对应的全局极小点.

次梯度算法结构

为了极小化一个不可微的凸函数f，可类似梯度法构造如下次梯度算法的迭代格式：

$$x^{k+1} = x^k - \alpha_k g^k, \quad g^k \in \partial f(x^k),$$

其中$\alpha_k > 0$为步长. 它通常有如下四种选择：

1. 固定步长$\alpha_k = \alpha$；
2. 固定$\|x^{k+1} - x^k\|$，即$\alpha_k \|g^k\|$为常数；
3. 消失步长$\alpha_k \to 0$且$\sum_{k=0}^{\infty} \alpha_k = +\infty$；
4. 选取α_k使其满足某种线搜索准则.

下面我们讨论在不同步长取法下次梯度算法的收敛性质.

提纲

假设条件

(1) f为凸函数；

(2) f至少存在一个有限的极小值点x^*，且$f(x^*) > -\infty$；

(3) f为利普希茨连续的，即

$$|f(x) - f(y)| \leq G\|x - y\|, \quad \forall\, x, y \in \mathbb{R}^n,$$

其中$G > 0$为利普希茨常数.

我们下面证明这等价于$f(x)$的次梯度是有界的，即

$$\|g\| \leq G, \quad \forall\, g \in \partial f(x), x \in \mathbb{R}^n.$$

证明：

- 充分性：假设$\|g\|_2 \le G, \forall\, g \in \partial f(x)$；取$g_y \in \partial f(y), g_x \in \partial f(x)$:

$$g_x^{\mathrm{T}}(x-y) \ge f(x) - f(y) \ge g_y^{\mathrm{T}}(x-y)$$

再由柯西不等式

$$G\|x-y\|_2 \ge f(x) - f(y) \ge -G\|x-y\|_2$$

- 必要性：反设存在x和$g \in \partial f(x)$，使得$\|g\|_2 > G$；取$y = x + \frac{g}{\|g\|_2}$

$$\begin{aligned} f(y) &\ge f(x) + g^{\mathrm{T}}(y-x) \\ &= f(x) + \|g\|_2 \\ &> f(x) + G \end{aligned}$$

这与$f(x)$是G-利普希茨连续的矛盾.

收敛性分析

- 次梯度方法不是一个下降方法，即无法保证$f(x^{k+1}) < f(x^k)$；
- 收敛性分析的关键是分析$f(x)$历史迭代的最优点所满足的性质.
- 设x^*是$f(x)$的一个全局极小值点，$f^* = f(x^*)$，根据迭代格式，

$$\begin{aligned}\left\|x^{i+1} - x^*\right\|^2 &= \left\|x^i - \alpha_i g^i - x^*\right\|^2 \\ &= \left\|x^i - x^*\right\|^2 - 2\alpha_i \left\langle g^i, x^i - x^*\right\rangle + \alpha_i^2 \left\|g^i\right\|^2 \\ &\leqslant \left\|x^i - x^*\right\|^2 - 2\alpha_i \left(f\left(x^i\right) - f^*\right) + \alpha_i^2 G^2\end{aligned}$$

- 结合$i = 0, \cdots, k$时相应的不等式，并定义$\hat{f}^k = \min_{0 \leqslant i \leqslant k} f\left(x^i\right)$：

$$\begin{aligned}2\left(\sum_{i=0}^{k} \alpha_i\right)\left(\hat{f}^k - f^*\right) &\leqslant \left\|x^0 - x^*\right\|^2 - \left\|x^{k+1} - x^*\right\|^2 + G^2 \sum_{i=0}^{k} \alpha_i^2 \\ &\leqslant \left\|x^0 - x^*\right\|^2 + G^2 \sum_{i=0}^{k} \alpha_i^2\end{aligned}$$

不同步长下的收敛性

(1) 取$\alpha_i = t$为固定步长，则

$$\hat{f}^k - f^* \leq \frac{\|x^0 - x^*\|^2}{2kt} + \frac{G^2 t}{2};$$

- $\hat{f}^k$ 无法保证收敛性
- 当k 足够大时,$\hat{f}^k$ 近似为$G^2 t/2$-次优的

(2) 取α_i使得$\|x^{i+1} - x^i\|$固定，即$\alpha_i \|g^i\| = s$为常数，则

$$\hat{f}^k - f^* \leq \frac{G\|x^0 - x^*\|^2}{2ks} + \frac{Gs}{2};$$

- $\hat{f}^k$ 无法保证收敛性
- 当k 足够大时,$\hat{f}^k$ 近似为$Gs/2$-次优的

不同步长下的收敛性

(3) 取α_i为消失步长，即$\alpha_i \to 0$且$\sum_{i=0}^{\infty} \alpha_i = +\infty$，则

$$\hat{f}^k - f^* \leq \frac{\|x^0 - x^*\|^2 + G^2 \sum_{i=0}^{k} \alpha_i^2}{2 \sum_{i=0}^{k} \alpha_i};$$

进一步可得$\hat{f}^k$收敛到f^*.

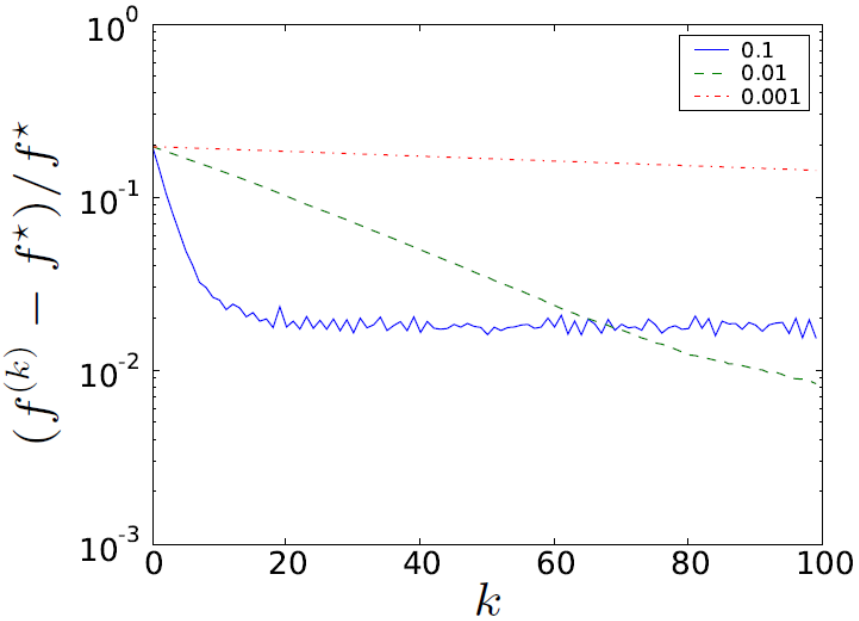
- 和梯度法不同，只有当α_k取消失步长时$\hat{f}^k$才具有收敛性.
- 一个常用的步长取法是$\alpha_k = \frac{1}{k}$.

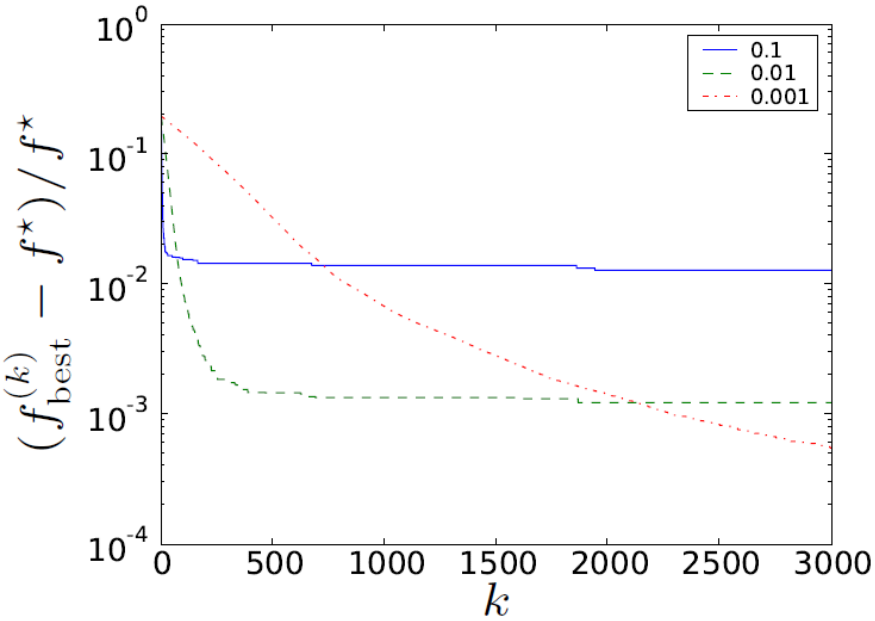
例：ℓ_1-范数极小化问题

$$\min \quad \|Ax-b\|_1 \quad (A \in \mathbb{R}^{500\times 100},\ b \in \mathbb{R}^{500})$$

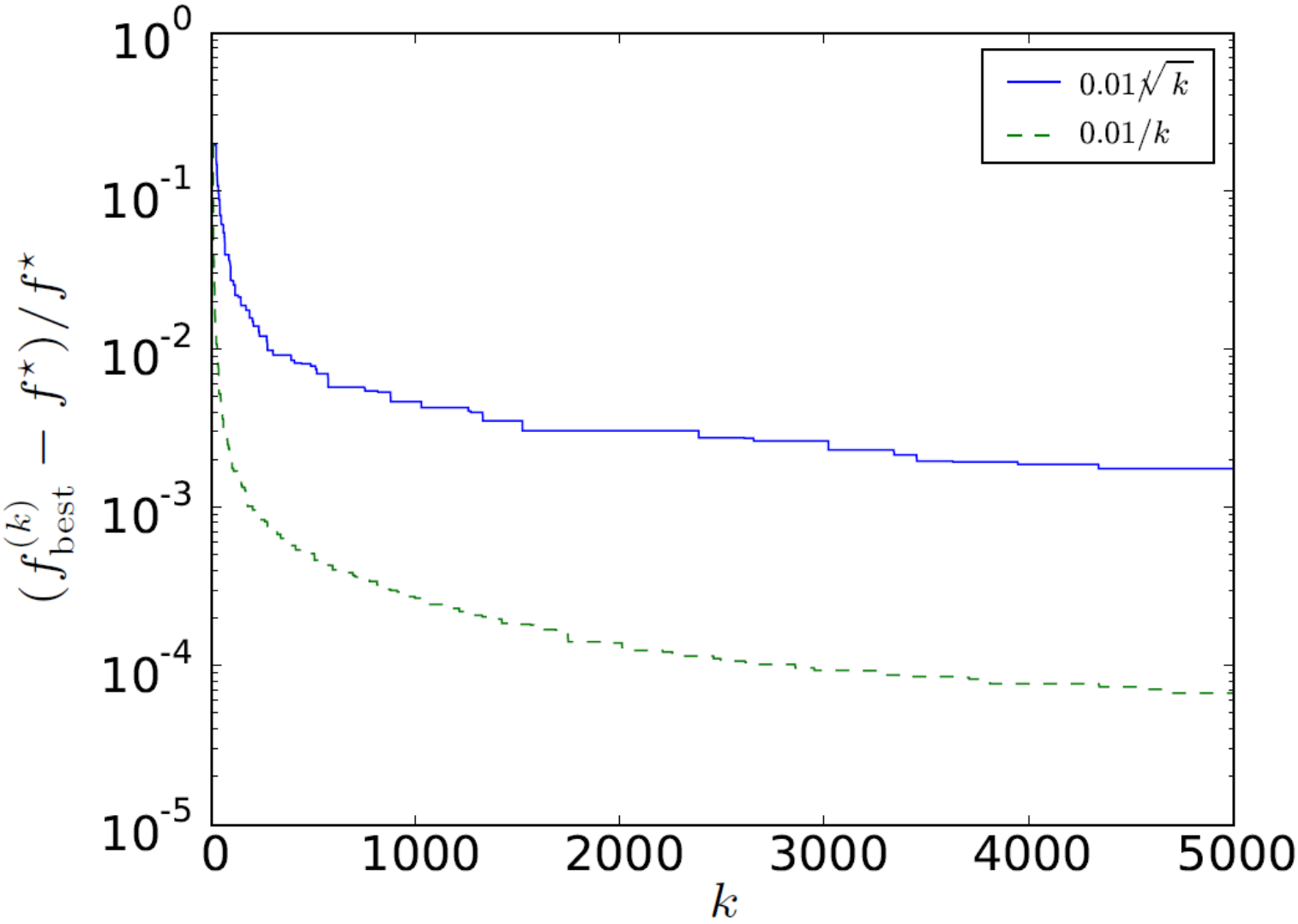
次梯度取为$A^{\mathrm{T}}\mathbf{sign}(Ax-b)$

- 第二类步长策略：$t_k = s/\|g^{(k-1)}\|_2$，$s = 0.1, 0.01, 0.001$





- 第三类步长策略：$t_k = 0.01/\sqrt{k}, t_k = 0.01/k$



固定迭代步数下的最优步长

- 假设$\left\|x^0 - x^*\right\| \leqslant R$，并且总迭代步数$k$是给定的，在固定步长下，

$$\hat{f}^k - f^* \leq \frac{\|x^0 - x^*\|^2}{2kt} + \frac{G^2 t}{2} \leq \frac{R^2}{2kt} + \frac{G^2 t}{2}.$$

- 由平均值不等式知当t满足$\frac{R^2}{2kt} = \frac{G^2 t}{2}$, 即 $t = \frac{R}{G\sqrt{k}}$时，右端达到最小.
- k 步后得到的上界是

$$\hat{f}^k - f^* \leqslant \frac{GR}{\sqrt{k}}$$

- 这表明在$k = O(1/\epsilon^2)$ 步迭代后可以得到$\hat{f}^k - f^* \leq \epsilon$ 的精度
- 类似地可证明第二类步长选取策略下，取$s = \frac{R}{\sqrt{k}}$，可得到估计

$$\hat{f}^k - f^* \leq \frac{GR}{\sqrt{k}}.$$

f^* 已知时的最优步长：Polyak's步长

- 第56页第一个不等式右端在

$$\alpha_i = \frac{f(x^i) - f^*}{\|g^i\|^2}$$

时取到极小.

- 这等价于

$$\frac{(f(x^i) - f^*)^2}{\|g^i\|^2} \leq \|x^i - x^*\|^2 - \|x^{i+1} - x^*\|^2.$$

- 递归地利用上式并结合$\|x^0 - x^*\| \leq R$和$\|g^i\| \leq G$，可以得到

$$\hat{f}^k - f^* \leq \frac{GR}{\sqrt{k}}.$$

练习：计算交集中的一点

为了寻找一个落在m个闭凸集$C_1, \ldots, C_m$的交集中的点：

$$\min \quad f(x) = \max\{d_1(x), \cdots, d_m(x)\}$$

其中$d_j(x) = \inf_{y \in C_j} \|x - y\|_2$表示点$x$到集合$C_j$的欧几里得距离

- 如果交集非空，$f^* = 0$
- 如果$g \in \partial d_j(\hat{x})$并且$C_j$是距离$\hat{x}$最远的集合，那么$g \in \partial f(\hat{x})$
- 次梯度$g \in \partial d_j(\hat{x})$：

$$g = 0 \text{ (如果 } \hat{x} \in C_j), \quad g = \frac{1}{d(\hat{x}, C_j)}(\hat{x} - P_j(\hat{x})) \text{ (如果 } \hat{x} \notin C_j)$$

其中$P_j(\hat{x})$是到C_j的投影；注意当$\hat{x} \notin C_j$时，$\|g\|_2 = 1$

最优步长选取下的次梯度法：

- 当$f^* = 0$且$\|g^{i-1}\|_2 = 1$时，最优步长是$\alpha_i = f(x^{i-1})$.
- 在第k步迭代, 找到距离最远的集合C_j (此时$f(x^{k-1}) = d_j(x^{k-1})$)；取

$$\begin{aligned} x^k &= x^{k-1} - \frac{f(x^{k-1})}{d_j(x^{k-1})}(x^{k-1} - P_j(x^{k-1})) \\ &= P_j(x^{k-1}) \end{aligned}$$

- 是一种交替投影算法
- 每步迭代都将当前迭代点投影到最远的集合上
- 当$m = 2$时, 交替投影到两个集合上

次梯度法的最优性

第62页得出的上界$\hat{f}^k - f^* \leqslant \frac{GR}{\sqrt{k}}$ 可以进一步改进吗?

问题设定:

- f 是凸函数, 有一个极小点x^*
- 已知初始点x^0 满足$\|x^0 - x^*\|_2 \leq R$
- 已知f 在$\{x | \|x - x^0\|_2 \leq R\}$ 上的利普希茨常数G
- f 的定义方式: 给定x, 返回$f(x)$ 及一个次梯度

算法设定: 每步迭代点x^i 都可用任一方法在如下集合中进行选择, 迭代k 步

$$x^0 + \text{span}\{g^0, g^1, \cdots, g^{i-1}\}$$

测试问题

$$f(x) = \max_{i=1,\cdots,k} x_i + \frac{1}{2}\|x\|_2^2, \quad x^0 = 0$$

- 测试问题的解：$x^* = -\frac{1}{k}(\underbrace{1,\cdots,1}_{k},\underbrace{0,\cdots,0}_{n-k})$ 以及$f^* = -\frac{1}{2k}$
- $R = \|x^0 - x^*\|_2 = \frac{1}{\sqrt{k}}$ 且$G = 1 + \frac{1}{\sqrt{k}}$
- 返回的次梯度是$e_{\hat{j}} + x$ 其中$\hat{j} = \min\{j | x_j = \max_{i=1,\ldots,k} x_i\}$

迭代： 当$i = 0,\cdots,k-1$, 分量$x_{i+1}^i,\cdots,x_k^i$ 等于零

$$\hat{f}^k - f^* = \min_{i<k} f(x^i) - f^* \geq -f^* = \frac{GR}{2(1+\sqrt{k})}$$

结论： 上界$O(1/\sqrt{k})$ 无法再改进

锐度条件下的线性收敛速度

我们称函数$f(x)$关于最优值$f(x^*)$满足锐度增长条件，如果存在$c>0$,使得

$$f(x)-f(x^*)\geq c\|x-x^*\|,\quad \forall x.$$

在该条件下，使用Polyak 步长

$$\alpha_i=\frac{f(x^i)-f^*}{\|g^i\|^2}$$

有如下结果

$$\begin{aligned}\left\|x^{i+1}-x^*\right\|^2&=\left\|x^i-\alpha_i g^i-x^*\right\|^2\\&=\left\|x^i-x^*\right\|^2-2\alpha_i\left\langle g^i,x^i-x^*\right\rangle+\alpha_i^2\left\|g^i\right\|^2\\&\leq\left\|x^i-x^*\right\|^2-2\alpha_i\left(f\left(x^i\right)-f^*\right)+\frac{\left(f(x^i)-f^*\right)^2}{\|g^i\|^2}\\&=\left\|x^i-x^*\right\|^2-\frac{\left(f(x^i)-f^*\right)^2}{\|g^i\|^2}\\&\leq\left(1-\frac{c^2}{\|g^i\|^2}\right)\left\|x^i-x^*\right\|^2.\end{aligned}$$

最后一行使用了锐度增长条件。

- 注意，此条件下$\|g^i\| \geq c$. 所以有线性收敛速度。
- 实际中，若不知道f^*, 可以证明存在$\alpha_0 > 0, \gamma_0 \in (0,1)$,采用$\alpha_i = \alpha_0 \gamma^i$得到线性收敛。
- 参考文献： Davis, Damek, et al. "Subgradient methods for sharp weakly convex functions." Journal of Optimization Theory and Applications 179 (2018): 962-982.

总结：次梯度法

- 能够处理一般的不可微凸函数
- 常能推导出非常简单的算法
- 收敛速度可能非常缓慢
- 没有很好的停机准则
- 理论复杂度：迭代$O(1/\epsilon^2)$步，得到ϵ-次优的点
- 一种“最优”的一阶方法：$O(1/\epsilon^2)$无法再改进

投影梯度法和条件梯度法

陈士祥

中国科学技术大学

致谢：本教案部分参考北京大学文再文教授《最优化与建模》

提纲

约束优化问题

在优化和应用数学中，约束优化的梯度方法对于解决广泛的问题至关重要。这些方法将梯度下降——一种用于无约束优化的基本技术——有效地改进到处理约束的情况。

我们将约束问题写为如下一般形式：

$$\min f(x) \quad \text{s.t. } x \in C,$$

其中，约束C是一个闭凸集。

定理 (约束问题最优条件)

考虑上述约束问题，若$x^* \in C$是最优点，当且仅当

$$\langle \nabla f(x^*), x^* - y \rangle \leq 0, \quad \forall y \in C.$$

证明：必要性：记

$$I_C(x) = \begin{cases} 0, & \text{if } x \in C, \\ +\infty, & else. \end{cases}$$

为指示函数。则问题转化为$\min f(x) + I_C(x)$. 由最优条件得$-\nabla f(x^*) \in \partial I_C(x^*)$. 根据次梯度定义，$\forall y \in C$,

$$0 \geq -\nabla f(x^*)^T (y - x^*),$$

得证。充分性：将上述证明反述。

投影梯度法

投影梯度下降方法是梯度下降在约束问题上的直接扩展。给定一个约束集合C，点x到C的投影，表示为$\mathsf{Proj}_C(x)$，定义为：

$$\mathsf{Proj}_C(x) = \arg\min_{z\in C} \|x - z\|^2.$$

投影梯度下降的更新规则是：

$$x_{k+1} = \mathsf{Proj}_C\left(x_k - \alpha_k \nabla f(x_k)\right),$$

其中α_k是第k次迭代的步长。

投影的非扩张性

投影的一个关键属性是其非扩张性：

定理

若集合C是一个闭凸集，那么

$$\|Proj_C(x) - Proj_C(y)\| \leq \|x - y\|,$$

对所有$x, y \in \mathbb{R}^n$成立。

证明：由投影问题的最优条件可知，

$$0 \in \text{Proj}_C(x) - x + \partial I_C(\text{Proj}_C(x)),$$

即$x - \text{Proj}_C(x)$是指示函数I_C在$\text{Proj}_C(x)$点的次梯度。根据次梯度定义可知，$\forall z \in C$, 我们有

$$0 \geq (x - \text{Proj}_C(x))^\top (z - \text{Proj}_C(x)), \quad \forall z \in C.$$

带入$z = \text{Proj}_C(y)$可知

$$0 \geq (x - \text{Proj}_C(x))^\top (\text{Proj}_C(y) - \text{Proj}_C(x)). \tag{1}$$

类似地，我们有

$$0 \geq (y - \text{Proj}_C(y))^\top (\text{Proj}_C(x) - \text{Proj}_C(y)). \tag{2}$$

将(1)和(2)相加可得

$$0 \geq [(y - \text{Proj}_C(y)) - (x - \text{Proj}_C(x))]^\top (\text{Proj}_C(x) - \text{Proj}_C(y)).$$

故，

$$\|\text{Proj}_C(x) - \text{Proj}_C(y)\|^2 \leq (x - y)^\top (\text{Proj}_C(x) - \text{Proj}_C(y)).$$

最后由柯西不等式得证。

投影到仿射集

超平面 $C = \{x | a^T x = b\}$ ($a \neq 0$)

$$P_C(x) = x + \frac{b - a^T x}{\|a\|_2^2} a$$

仿射集 $C = \{x | Ax = b\}$ ($A \in \mathbb{R}^{p \times n}$，且$\text{rank}(A) = p$)

$$P_C(x) = x + A^T (AA^T)^{-1} (b - Ax)$$

当$p \ll n$, 或$AA^T = I, \ldots$ 时，计算成本较低

投影到多面体集

半平面 $C = \{x | a^T x \leq b\}$ $(a \neq 0)$

$$P_C(x) = x + \frac{b - a^T x}{\|a\|_2^2} a \quad if \quad a^T x > b,$$

$$P_C(x) = x \quad if \quad a^T x \leq b$$

矩形： $C = [l, u] = \{x \mid l \leq x \leq u\}$

$$P_C(x)_i = \begin{cases} l_i & x_i \leq l_i \\ x_i & l_i \leq x_i \leq u_i \\ u_i & x_i \geq u_i \end{cases}$$

非负象限： $C = \mathbf{R}_+^n$

$$P_C(x) = x_+ \qquad (x_+ \text{ 表示各分量取} max\{0, x\})$$

概率单纯形：$C = \{x | 1^T x = 1, x \geq 0\}$

$$P_C(x) = (x - \lambda 1)_+$$

其中，λ是下面方程的解：

$$1^T (x - \lambda 1)_+ = \sum_{i=1}^{n} \max\{0, x_k - \lambda\} = 1$$

（一般的）概率单纯形：$C = \{x | a^T x = b, l \leq x \leq u\}$

$$P_c(x) = P_{[l,u]}(x - \lambda a)$$

其中，λ 是下面方程的解：

$$a^T P_{[l,u]}(x - \lambda a) = b$$

投影到范数球

Euclid 球：$C = \{x | \|x\|_2 \leq 1\}$

$$P_C(x) = \frac{1}{\|x\|_2} x \quad if \quad \|x\|_2 > 1,$$

$$P_C(x) = x \quad if \quad \|x\|_2 \leq 1$$

ℓ_1 范数球：$C = \{x | \|x\|_1 \leq 1\}$

$$P_c(x)_k = \begin{cases} x_k - \lambda & x_k > \lambda \\ 0 & -\lambda \leq x_k \leq \lambda \\ x_k + \lambda & x_k < -\lambda \end{cases}$$

若 $\|x\|_1 \leq 1$，则 $\lambda = 0$; 其他情形，λ 是下面方程的解

$$\sum_{k=1}^{n} \max\{|x_k| - \lambda, 0\} = 1$$

投影到简单锥

二阶锥：$C = \{(x,t) \in \mathbf{R}^{n\times 1} | \|x\|_2 \le t\}$

$$P_C(x,t) = (x,t) \quad if \quad \|x\|_2 \le t,$$

$$P_C(x,t) = (0,0) \quad if \quad \|x\|_2 \le -t$$

且

$$P_C(x,t) = \frac{t + \|x\|_2}{2\|x\|_2} \begin{bmatrix} x \\ \|x\|_2 \end{bmatrix} \quad if \ \|x\|_2 \ge |t|, \ x \ne 0$$

半正定锥：$C = \mathbf{S}_+^n$

$$P_C(X) = \sum_{i=1}^{n} \max\{0, \lambda_i\} q_i q_i^T$$

其中，$X = \sum_{i=1}^{n} \lambda_i q_i q_i^T$ 是X 的特征值分解

强凸且光滑问题

对于目标函数f 是μ-强凸和L-光滑的问题，投影梯度下降享有与无约束情况相似的收敛性质。

Fact (强凸问题的收敛性)

假设f 是μ-强凸且L-光滑的，并且最优解x^* 是在C 的内部。如果步长$\alpha_k = \frac{2}{L+\mu}$，则投影梯度下降生成的序列$\{x_k\}$ 满足：

$$\|x_k - x^*\|^2 \leq \left(\frac{L/\mu - 1}{L/\mu + 1}\right)^k \|x_0 - x^*\|^2.$$

该结论需要最优解x^* 是在C 的内部，一般不能成立！

证明：

$$\|x_{k+1} - x^*\|^2 = \|P_C(x_k - \alpha_k \nabla f(x_k)) - P_C(x^*)\|^2 \leq \|x_k - \alpha_k \nabla f(x_k) - x^*\|^2$$

由于x^*在内部，故无约束问题的分析可以直接套用。

一般强凸问题收敛性

定理 (强凸问题的收敛性)

假设f 是μ-强凸且L-光滑的。如果步长$\alpha_k = \frac{1}{L}$，则投影梯度下降生成的序列$\{x_k\}$ 满足：

$$\|x_k - x^*\|^2 \leq (1 - \frac{\mu}{L})^k \|x_0 - x^*\|^2.$$

定理的证明

证明：
令$x^+ := P_C(x - \frac{1}{L}\nabla f(x))$ and $g_C(x) := L(x - x^+)$. 该定理主要基于如下不等式：

$$g_C(x)^\top (x - x^*) \geq \frac{\mu}{2}\|x - x^*\|^2 + \frac{1}{2L}\|g_C(x)\|^2 \tag{3}$$

有了这个基础，投影梯度下降收敛如下：

$$\|x_{k+1} - x^*\|^2 \leq \left(1 - \frac{\mu}{L}\right)\|x_k - x^*\|^2$$

故，我们下面只需要证明(3).
因为

$$\begin{aligned} 0 \leq & f(x^+) - f(x^*) = f(x^+) - f(x) + f(x) - f(x^*) \\ & \leq \nabla f(x) \cdot (x^+ - x) + \frac{L}{2}\|x^+ - x\|^2 + \nabla f(x) \cdot (x - x^*) - \frac{\mu}{2}\|x - x^*\|^2 \\ & = \nabla f(x) \cdot (x^+ - x^*) + \frac{1}{2L}\|g_C(x)\|^2 - \frac{\mu}{2}\|x - x^*\|^2. \end{aligned}$$

因此，如果有下述不等式，那么(3)成立：

$$\nabla f(x) \cdot (x^+ - x^*) \leq g_C(x) \cdot (x^+ - x^*) \quad \text{(projection only makes it better)}$$

$$= g_C(x) \cdot (x - x^*) - \frac{1}{L}\|g_C(x)\|^2$$

因为我们有(1),所以上述不等式成立。

一般$L-$光滑凸问题收敛

若$f(x)$是凸函数，且$\nabla f(x)$是$L-$Lipschitz连续的。那么投影梯度法的收敛结论如下：

$$f(x_k) - f^* \leq \frac{3L\|x_0 - x^*\|^2 + f(x_0) - f(x^*))}{k+1}$$

- 该收敛速度和无约束情况类似。
- 我们将在后面学习近似点梯度法再做证明。

提纲

条件梯度法: 动机

设 C 为凸紧集，考虑优化问题

$$\min_{x\in C} \quad f(x)$$

- 如果使用投影梯度法：

$$x_{k+1} = \mathcal{P}_C\left(x_k - \alpha_k \nabla f\left(x_k\right)\right)$$

这等价于每一步求解如下问题：

$$x_{k+1} = \operatorname*{argmin}_{x\in C} \left\{ f\left(x_k\right) + \left\langle \nabla f\left(x_k\right), x - x_k \right\rangle + \frac{1}{2\alpha_k} \left\| x - x_k \right\|^2 \right\}.$$

- 困难：$\mathcal{P}_C(\cdot)$可能具有昂贵的计算代价

条件梯度法 (CndG or Frank-Wolfe Method)

- 给定 $y_0 = x_0$，以及 $\alpha_k \in (0, 1]$. 条件梯度法的迭代格式为：

$$\begin{aligned} x_k &= \operatorname{argmin}_{x \in C} \langle \nabla f(y_{k-1}), x \rangle, \\ y_k &= (1 - \alpha_k) y_{k-1} + \alpha_k x_k \end{aligned}$$

- x_k的迭代其等价于在y_{k-1}附近最小化线性近似函数：

$$x_k = \operatorname{argmin}_{x \in C} f(y_{k-1}) + \langle \nabla f(y_{k-1}), x - y_{k-1} \rangle$$

  该子问题在一些情况下很容易求解！
- 步长参数 α_k 的选取
  - 消失步长：

$$\alpha_k = \frac{2}{k+1}$$

  - 或通过精确线搜索：

$$\alpha_k = \operatorname*{argmin}_{\alpha \in [0,1]} f\left((1-\alpha) y_{k-1} + \alpha x_k\right)$$

求解子问题：例子

考虑带某一范数$\|\cdot\|$约束的凸优化问题，

$$\min_{x} \quad f(x) \quad \text{s.t.} \quad \|x\| \leq t.$$

用条件梯度法求解该问题时，需要计算子问题，

$$\begin{aligned} x_k &\in \operatorname*{argmin}_{\|x\| \leq t} \langle \nabla f(y_{k-1}), x \rangle \\ &= -t \cdot \left(\arg\max_{\|x\| \leq 1} \langle \nabla f(y_{k-1}), x \rangle \right) \\ &= -t \cdot \partial \|\nabla f(y_{k-1})\|_*. \end{aligned} \tag{4}$$

其中$\|z\|_* = \sup\{z^T x, \|x\| \leq 1\}$是$\|\cdot\|$的对偶范数。注意到(4)条件梯度法的子问题相当于计算一个对偶范数的次梯度。如果计算$\|\cdot\|$范数的次梯度比计算在约束集合$C = \{x \in \mathbb{R}^n : \|x\| \leq t\}$上的投影要简单，条件梯度法比投影梯度法效率更高。

例子: ℓ_1 范数约束问题

由于 ℓ_1 范数的对偶范数是 ℓ_∞ 范数，因此用条件梯度法求解该问题时子问题为:

$$x_k \in -t \cdot \partial \|\nabla f(y_{k-1})\|_\infty.$$

考虑到 ℓ_∞ 范数的次梯度为$\partial \|x\|_\infty = \{v : \langle v, x\rangle = \|x\|_\infty, \|v\|_1 \le 1\}$，子问题等价于，

$$\begin{aligned} i_k &\in \operatorname*{argmax}_{i=1,\dots,n} |\nabla_i f(y_{k-1})| \\ x_k &= -t \cdot \operatorname{sgn}\left[\nabla_{i_k} f(y_{k-1})\right] \cdot e_{i_k}. \end{aligned}$$

其中$\nabla_i f(y_{k-1})$ 表示向量$\nabla f(y_{k-1})$ 的第i 个元素，e_i 表示第i 个元素为1 的单位向量。可以看到计算$\|\cdot\|_\infty$ 的次梯度和计算集合$C := \{x \in \mathbb{R}^n : \|x\|_1 \le t\}$ 上的投影都需要$\mathcal{O}(n)$ 的计算复杂度，但是条件梯度法子问题计算明显要更简单直接。

例子: ℓ_p 范数约束问题, $1 \le p \le \infty$

由于ℓ_p 范数的对偶范数是ℓ_q 范数，其中$1/p + 1/q = 1$，因此用条件梯度法求解该问题时子问题为，

$$x_k \in -t \cdot \partial \|\nabla f(y_{k-1})\|_q.$$

注意到ℓ_q 范数的次梯度为$\partial \|x\|_q = \{v : \langle v, x\rangle = \|x\|_q, \|v\|_p \le 1\}$，子问题等价于，

$$x_k^i = -\beta \cdot \operatorname{sgn}\left[\nabla_i f(y_{k-1})\right] \cdot |\nabla_i f(y_{k-1})|^{p/q}.$$

其中β 是使得$\|x_k\|_q = t$ 的归一化常数。可以看到，除过$p = 1, 2, \infty$ 这些特殊情形，条件梯度法的子问题计算复杂度比直接计算点在集合$C = \{x \in \mathbb{R}^n : \|x\|_p \le t\}$ 上的投影要简单，后者投影计算需要单独解一个优化问题。

例子: 矩阵核范数约束优化问题

矩阵核范数$\|\cdot\|_*$的对偶范数是其谱范数$\|\cdot\|_2$:

$$\|X\|_* = \sum_{i=1}^{\min\{m,n\}} \sigma_i(X), \qquad \|X\|_2 = \max_{i=1,\ldots,\min\{m,n\}} \sigma_i(X).$$

因此条件梯度法的子问题为$X_k \in -t \cdot \partial\|\nabla f(Y_{k-1})\|_2$. 对矩阵范数的次梯度: $\partial\|X\| = \{Y : \langle Y, X\rangle = \|X\|, \|Y\|_* \le 1\}$，设$u, v$分别是矩阵$\nabla f(Y_{k-1})$最大奇异值对应的左、右奇异向量，注意到，

$$\langle uv^T, \nabla f(Y_{k-1})\rangle = u^T \nabla f(Y_{k-1}) v = \sigma_{\max}(\nabla f(Y_{k-1})) = \|\nabla f(Y_{k-1})\|_2.$$

且$\|uv^T\|_* = 1$，因此矩阵$uv^T \in \partial\|\nabla f(Y_{k-1})\|_2$。则条件梯度法子问题等价于，

$$X_k \in -t \cdot uv^T. \tag{5}$$

可以看到，条件梯度法计算子问题时只需要计算矩阵最大的奇异值对应的左、右奇异向量。如果采用投影梯度法，其子问题是计算X到集合$\{X \in \mathbb{R}^{m\times n} : \|X\|_* \le t\}$的投影，需要对矩阵做全奇异值分解，计算量比条件梯度法复杂很多。

注

上面几个例中，可以看出，子问题的解x_k要么是稀疏的，要么是低秩的，这便保证了迭代点非常好的性质。这在许多机器学习应用中非常重要。

收敛性分析

结论：令 $f(x)$ 是凸函数，$\nabla f(x)$ 是L-利普希茨的，$D_C = \sup_{x,y\in C}\|x-y\|$. 则

$$f(y_k) - f(x^*) \le \frac{2L}{k(k+1)}\sum_{i=1}^{k}\|x_i - y_{i-1}\|^2 \le \frac{2L}{k+1}D_C^2.$$

证明：令$\gamma_k = \frac{2}{k+1}$，记$\bar{y}_k = (1-\gamma_k)y_{k-1} + \gamma_k x_k$，则不管

$$\alpha_k = \frac{2}{k+1} \quad 或 \quad \alpha_k = \operatorname*{argmin}_{\alpha\in[0,1]} f((1-\alpha)y_{k-1} + \alpha x_k).$$

对$y_k = (1-\alpha_k)y_{k-1} + \alpha_k x_k$，我们都有$f(y_k) \le f(\bar{y}_k)$。注意到$\bar{y}_k - y_{k-1} = \gamma_k(x_k - y_{k-1})$，由$f(x) \in C_L^{1,1}(C)$有

$$f(y_k) \le f(\bar{y}_k) \le f(y_{k-1}) + \langle \nabla f(y_{k-1}), \bar{y}_k - y_{k-1}\rangle + \frac{L}{2}\|\bar{y}_k - y_{k-1}\|^2 \tag{6}$$

$$\le (1-\gamma_k)[f(y_{k-1}) + \gamma_k[f(y_{k-1}) + \langle \nabla f(y_{k-1}), x - y_{k-1}\rangle] + \frac{L\gamma_k^2}{2}\|x_k - y_{k-1}\|^2 \tag{7}$$

$$\le (1-\gamma_k)f(y_{k-1}) + \gamma_k f(x) + \frac{L\gamma_k^2}{2}\|x_k - y_{k-1}\|^2, \quad 对任意\ x \in C. \tag{8}$$

其中不等式(7)是因为$x_k \in \min_{x\in C}\langle \nabla f(y_{k-1}), x\rangle$，由最优性条件我们可以得到对任意$x \in C$有$\langle x - x_k, \nabla f(y_{k-1})\rangle \ge 0$。

条件梯度法收敛性分析: 引理

令$\gamma_t \in (0,1]$，$t = 1,2,\ldots$，构造序列

$$\Gamma_t = \begin{cases} 1 & t = 1 \\ (1-\gamma_t)\Gamma_{t-1} & t \geq 2 \end{cases}.$$

如果序列$\{\Delta_t\}_{t\geq 0}$满足

$$\Delta_t \leq (1-\gamma_t)\Delta_{t-1} + B_t \quad t = 1,2,\ldots.$$

则对任意的k我们对Δ_k有估计（数学归纳法）

$$\Delta_k \leq \Gamma_k(1-\gamma_1)\Delta_0 + \Gamma_k \sum_{t=1}^{k} \frac{B_t}{\Gamma_t}.$$

收敛性分析

将不等式(8)稍做变换，对任意$x \in C$，

$$f(y_k) - f(x) \leq (1-\gamma_k)[f(y_{k-1}) - f(x)] + \frac{L}{2}\gamma_k^2 \|x_k - y_{k-1}\|^2. \tag{9}$$

由引理可知，

$$f(y_k) - f(x) \leq \Gamma_k(1-\gamma_1)[f(y_0) - f(x)] + \frac{\Gamma_k L}{2}\sum_{i=1}^{k}\frac{\gamma_i^2}{\Gamma_i}\|x_i - y_{i-1}\|^2.$$

由$\gamma_k = \frac{2}{k+1}$，$\gamma_1 = 1$得到$\Gamma_k = \frac{2}{k(k+1)}$，我们可以得到收敛性不等式，

$$f(y_k) - f^* \leq \frac{2L}{k(k+1)}\sum_{i=1}^{k}\|x_i - y_{i-1}\|^2 \leq \frac{2L}{k+1}D_C^2.$$

令$\frac{2L}{k+1}D_C^2 \leq \epsilon$，可以得到分析复杂度结论。

另一种分析：原始-对偶角度

- 由于f是凸函数，对于任意两点$x, y \in \mathcal{D}$，我们有：

$$f(y) \geq f(x) + (y - x)^T \nabla f(x)$$

- 这同样适用于最优解x^*。即$f(x^*) \geq f(x) + (x^* - x)^T \nabla f(x)$。相对于给定点$x$的最佳下界由下式给出

$$f(x^*) \geq f(x) + (x^* - x)^T \nabla f(x) \geq \min_{y \in \mathcal{D}} \{f(x) + (y - x)^T \nabla f(x)\} = f(x) - x^T \nabla f(x) + \min_{y \in \mathcal{D}} y^T \nabla f(x)$$

- 根据条件梯度法每次迭代，因此第k次迭代的方向寻找子问题的解x_k可用于通过设置$l_0 = -\infty$来确定每次迭代中更大（更好）的下界l_k

$$l_k := \max(l_{k-1}, f(y_{k-1}) + \nabla f(y_{k-1})^T (x_k - y_{k-1}))$$

- 这样对未知最优值的下界在实践中是重要的，因为它们可以被用作停止准则，并且在每次迭代中给出一个有效的最优准则，因为总是有$l_k \leq f(x^*) \leq f(y_k)$。
- 已经证明，这个对应于对偶间隙，即$f(y_k)$与下界l_k之间的差值，以相同的收敛速率减少，即$f(y_k) - l_k = O(1/k)$。
- 参考文献：Jaggi, Martin. "Revisiting Frank-Wolfe: Projection-free sparse convex optimization." International conference on machine learning. PMLR, 2013. 讨论了非精确求解子问题时的收敛速度。

关于强凸问题

问题：在存在强凸性的情况下，我们能期望改进Frank-Wolfe算法的收敛保证吗？

- 一般情况下，不可以。
- 在附加条件下，可能可以改进。

一个负面结果

$$\min_{x\in\mathbb{R}^n} \quad \frac{1}{2}x^TQx + b^Tx \quad (Q \succeq 0)$$

$$\text{s.t.} \quad x = [a_1, \ldots, a_k]v, \quad v \geq 0, \quad 1^Tv = 1, \quad \text{即} x \in \text{conv}\{a_1, \ldots, a_k\}$$

记约束为$\mathcal{X}$,假设：

- $int(\mathcal{X}) = \emptyset$
- 最优点x^*位于Ω的边界上并且不是一个极点

定理**(Canon & Cullum '68)**：设$\{x_t\}$为解决上述问题的Frank-Wolfe迭代，步长通过精确线搜索得到。存在一个初始点x_0，使得对于任意$\epsilon > 0$，存在无穷多个t，

$$f(x_t) - f(x^*) \geq \frac{1}{t^{1+\epsilon}}$$

- 示例：选择$x_0 \in int(\mathcal{X})$，使得$f(x_0) < \min_i f(a_i)$
- 一般情况下，不能改进$O(1/t)$的收敛保证

正面结果

为了实现更快的收敛，需要额外的假设：

- 示例：强凸可行集C 设集合C是μ-强凸的，如果对于所有$\lambda \in [0,1]$和所有$x, z \in C$，

$$B\left(\lambda x + (1-\lambda)z, \frac{\mu}{2}\lambda(1-\lambda)\|x-z\|^2\right) \subseteq C$$

其中$B(a,r) := \{y \mid \|y-a\|^2 \leq r\}$，
定理**(Levitin & Polyak '66)**：假设f是凸的且L-平滑的，且C是μ-强凸的。假设对于所有$x \in C$，$\|\nabla f(x)\|^2 \geq c > 0$。那么在温和的条件下，精确线搜索的Frank-Wolfe线性收敛。

- 无需条件$\|\nabla f(x)\|^2 \geq c > 0$，$\mathcal{O}(1/t^2)$收敛。Garber, Dan, and Elad Hazan. "Faster rates for the Frank-Wolfe method over strongly-convex sets." International Conference on Machine Learning. PMLR, 2015.
- 更改迭代，利用away step:Lacoste-Julien, Simon, and Martin Jaggi. "On the global linear convergence of Frank-Wolfe optimization variants." Advances in neural information processing systems 28 (2015).
- 具体约束，$\{X : \|X\|_* \leq 1\}$,线性收敛：Allen-Zhu, Zeyuan, et al. "Linear convergence of a frank-wolfe type algorithm over trace-norm balls." Advances in neural information processing systems 30 (2017).

近似点映射

陈士祥

中国科学技术大学

致谢：本教案部分参考北京大学文再文教授《最优化与建模》

提纲

闭集

一个包含其边界的集合 C 被称为闭集：

$$x^k \in C, \quad x^k \to \bar{x} \qquad \Longrightarrow \qquad \bar{x} \in C$$

保持闭性的操作：

- （有限或无限个）闭集的交集仍是闭集
- 有限个闭集的并集仍是闭集
- 线性映射的原象集: 在 C 闭的情形下，$\{x \mid Ax \in C\}$ 是闭集，

线性映射的像

一个闭集在线性映射下的像不一定是闭的

例: (C 闭, $AC = \{Ax \mid x \in C\}$ 开)

$$C = \{(x_1, x_2) \in \mathbf{R}^2_+ \mid x_1 x_2 \geq 1\}, \quad A = \begin{bmatrix} 1 & 0 \end{bmatrix}, \quad AC = \mathbf{R}_{++}$$

（充分条件） 以下条件成立时，AC 为闭集：

- C 是闭凸集
- A 的零空间中不包含 C 的衰退方向 (recession direction)，即

$$Ay = 0, \quad \hat{x} \in C, \quad \hat{x} + \alpha y \in C \quad \forall \alpha \geq 0 \qquad \Longrightarrow \qquad y = 0$$

特别地，若 C 有界，则 AC 为闭集

闭函数

定义

一个函数被称为闭函数，如果它的上方图是闭集

闭函数的例子：

- $f(x) = -\log(1-x^2)$, $\mathbf{dom}\, f = \{x \mid |x| < 1\}$
- $f(x) = x\log x$, $\mathbf{dom}\, f = \mathbf{R}_+$ 且$f(0) = 0$
- 闭集 C 的示性函数

不是闭函数的例子：

- $f(x) = x\log x$, $\mathbf{dom}\, f = \mathbf{R}_{++}$ 或 $\mathbf{dom}\, f = \mathbf{R}_+$, $f(0) = 1$
- 不是闭集的集合 C 的示性函数

性质

下水平集: f 是闭函数当且仅当 f 的所有 α-下水平集都是闭集

最小值: 如果 f 是闭函数且存在有界的下水平集，则 f 有最小值点

常见的保闭性的操作（凸函数）

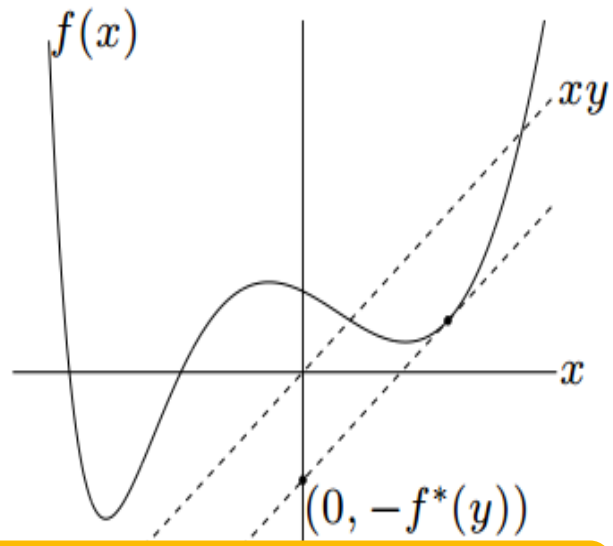
- 加法: $f+g$ 是闭函数，如果 f 和 g 都是闭的 ($\mathbf{dom}\, f \cap \mathbf{dom}\, g \neq \varnothing$)
- 复合线性映射: $f(Ax+b)$是闭函数，如果 f 是闭的
- 取上确界: $\sup_\alpha f_\alpha(x)$ 是闭函数，如果任意函数 f_α是闭的

提纲

共轭函数

函数 f 的共轭函数定义为：

$$f^*(y) = \sup_{x \in \mathbf{dom}\, f} (y^T x - f(x))$$



f^* 恒为闭凸函数

Fenchel 不等式：

$$f(x) + f^*(y) \geq x^T y \quad \forall x, y$$

Proof.

$$f^*(y) = \sup_{x \in \mathbf{dom}\, f} \{y^{\mathrm{T}} x - f(x)\} \geq y^{\mathrm{T}} x - f(x), \quad \forall x \in \mathbf{dom}\, f$$

□

二次函数

考察二次函数 f 的共轭函数：

$$f(x) = \frac{1}{2}x^T Ax + b^T x + c$$

- 强凸情形（$A \succ 0$）

$$f^*(y) = \frac{1}{2}(y-b)^T A^{-1}(y-b) - c$$

- 一般凸情形（$A \succeq 0$）

$$f^*(y) = \frac{1}{2}(y-b)^T A^{\dagger}(y-b) - c, \quad \mathbf{dom}\, f^* = \mathcal{R}(A) + b$$

这里$\mathcal{R}(A)$为A的像空间.

负熵与负对数

- 负熵

$$f(x) = \sum_{i=1}^{n} x_i \log x_i \qquad f^*(y) = \sum_{i=1}^{n} e^{y_i - 1}$$

- 负对数

$$f(x) = -\sum_{i=1}^{n} \log x_i \qquad f^*(y) = -\sum_{i=1}^{n} \log(-y_i) - n$$

- 矩阵对数

$$f(X) = -\log\det X \quad (\mathbf{dom}\, f = \mathbf{S}^n_{++}) \quad f^*(Y) = -\log\det(-Y) - n$$

示性函数与范数

凸集 C 的示性函数：共轭为 C 的支撑函数

$$f(x)=\begin{cases}0, & x\in C\\ +\infty, & x\notin C\end{cases}\qquad f^*(y)=\sup_{x\in C}y^Tx$$

范数：共轭为单位对偶范数球的示性函数

$$f(x)=||x||\qquad f^*(y)=\begin{cases}0, & ||y||_*\le 1\\ +\infty, & ||y||_*>1\end{cases}$$

Proof.

回忆对偶范数的定义：$||y||_*=\sup_{||x||\le 1}x^Ty$

分两类讨论计算$f^*(y)=\sup_x(y^Tx-||x||)$

- 若$||y||_*\le 1$，则 $y^Tx\le||x||\quad\forall x$ (对偶范数的定义)
  $x=0$时等式成立，因此 $\sup_x(y^Tx-||x||)=0$
- 若$||y||_*>1$，则存在一个 x，满足$||x||\le 1, x^Ty>1$，因此有

$$f^*(y)\ge y^T(tx)-||tx||=t(y^Tx-||x||)\to\infty\quad(t\to\infty)$$

□

二次共轭函数

任一函数f的二次共轭函数定义为

$$f^{**}(x) = \sup_{y \in \mathbf{dom}\, f^*} (x^T y - f^*(y))$$

- $f^{**}(x)$ 为闭凸函数
- 由Fenchel不等式，$x^T y - f^*(y) \le f(x)$ 对所有x, y 都成立，推出：

  $$f^{**}(x) \le f(x) \quad \forall x$$

  等价地，$\mathbf{epi}\, f \subseteq \mathbf{epi}\, f^{**}$ (对任意函数f 成立)
- 若f 是闭凸函数，则

  $$f^{**}(x) = f(x) \quad \forall x$$

  等价地，$\mathbf{epi}\, f = \mathbf{epi}\, f^{**}$ (若f 是闭凸函数); 证明在下一面

Proof.

假设$(x, f^{**}(x)) \notin \mathbf{epi}\, f$，则存在严格的分割超平面

$$\begin{bmatrix} a \\ b \end{bmatrix}^T \begin{bmatrix} z - x \\ s - f^{**}(x) \end{bmatrix} \leq c \leq 0 \qquad \forall (z, s) \in \mathbf{epi}\, f$$

其中$a \in \mathbb{R}^n, b, c \in \mathbb{R}$ 且$b \leqslant 0$ (若$b > 0$, 则取 $s \to +\infty$ 可推出矛盾).

- 若$b < 0$,取$s = f(z)$,有$a^{\mathrm{T}} z + bf(z) - a^{\mathrm{T}} x - bf^{**}(x) \leqslant c$

  记$y = a/(-b)$,两边除以$-b$，并将上式左边关于 z 极大化得到

  $$f^*(y) - y^{\mathrm{T}} x + f^{**}(x) \leqslant -\frac{c}{b} < 0$$

  与Fenchel 不等式矛盾.
- 若$b = 0$, 取$\hat{y} \in \mathbf{dom}\, f^*$ 并给$\begin{bmatrix} a \\ b \end{bmatrix}$ 加上一个$\begin{bmatrix} \hat{y} \\ -1 \end{bmatrix}$ 的ε倍，则

  $$\begin{bmatrix} a + \varepsilon \hat{y} \\ -\varepsilon \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} z - x \\ s - f^{**}(x) \end{bmatrix} \leqslant c + \varepsilon \left(f^*(\hat{y}) - x^{\mathrm{T}} \hat{y} + f^{**}(x) \right) < 0$$

  即化为$b < 0$ 的情况，矛盾. □

共轭函数与次梯度

定理

如果 f 是闭凸函数，则

$$y \in \partial f(x) \quad \Leftrightarrow \quad x \in \partial f^*(y) \quad \Leftrightarrow \quad x^T y = f(x) + f^*(y)$$

计算规则

- 可分解的和：

$$f(x_1, x_2) = g(x_1) + h(x_2) \qquad f^*(y_1, y_2) = g^*(y_1) + h^*(y_2)$$

- 数乘：($\alpha > 0$)

$$f(x) = \alpha g(x) \qquad f^*(y) = \alpha g^*(y/\alpha)$$

- 添加线性函数：

$$f(x) = g(x) + a^T x + b \qquad f^*(y) = g^*(y - a) - b$$

- 卷积下确界：

$$f(x) = \inf_{u+v=x} (g(u) + h(v)) \qquad f^*(y) = g^*(y) + h^*(y)$$

提纲

邻近算子

定义邻近算子：

$$\operatorname{prox}_h(x) = \operatorname*{argmin}_u \left(h(u) + \frac{1}{2}\|u - x\|_2^2 \right)$$

直观理解：求解一个距x不算太远的点u，并使函数值$h(u)$也相对较小

定理（邻近算子是良定义的）

如果 h 为闭凸函数，则对任意 x，$\operatorname{prox}_h(x)$存在且唯一

Proof.

首先注意到$h(u) + \frac{1}{2}\|u - x\|_2^2$ 是强凸函数，则

- 存在性：强凸函数的所有α-下水平集有界，故由Weierstrass定理知最小值存在
- 唯一性：强凸函数最小值唯一

□

定理

若h是适当的闭凸函数, 则 $u = \mathrm{prox}_h(x) \iff x - u \in \partial h(u)$

Proof.

若$u = \mathrm{prox}_h(x)$, 则由最优性条件得$0 \in \partial h(u) + (u - x)$, 因此有$x - u \in \partial h(u)$. 反之，若$x - u \in \partial h(u)$ 则由次梯度的定义可得到

$$h(v) \geqslant h(u) + (x - u)^{\mathrm{T}}(v - u), \quad \forall v \in \mathbf{dom}\ h$$

两边同时加$\frac{1}{2}\|v - x\|^2$, 即有

$$\begin{aligned} h(v) + \frac{1}{2}\|v - x\|^2 &\geqslant h(u) + (x - u)^{\mathrm{T}}(v - u) + \frac{1}{2}\|(v - u) - (x - u)\|^2 \\ &\geqslant h(u) + \frac{1}{2}\|u - x\|^2, \quad \forall v \in \mathbf{dom}\ h \end{aligned}$$

根据定义可得$u = \mathrm{prox}_h(x)$. □

邻近算子的例子

在近似点梯度法中，我们关心那些邻近算子prox_{th}容易计算的函数 h

例：ℓ_1 范数

$h(x) = \|x\|_1, \quad \text{prox}_{th}(x) = \text{sign}(x)\max\{|x| - t, 0\}$

Proof.

邻近算子$u = \text{prox}_{th}(x)$ 的最优性条件为

$$x - u \in t\partial\|u\|_1 = \begin{cases} \{t\}, & u > 0 \\ [-t, t], & u = 0 \\ \{-t\}, & u < 0 \end{cases}$$

当$x > t$ 时, $u = x - t$; 当$x < -t$ 时, $u = x + t$; 当$x \in [-t, t]$ 时, $u = 0$,
即有$u = \text{sign}(x)\max\{|x| - t, 0\}$. □

邻近算子的例子

例：ℓ_2 范数

$$h(x) = \|x\|_2, \quad \mathrm{prox}_{th}(x) = \begin{cases} \left(1 - \frac{t}{\|x\|_2}\right) x, & \|x\|_2 \geqslant t, \\ 0, & \text{其他}. \end{cases}$$

Proof.

邻近算子$u = \mathrm{prox}_{th}(x)$ 的最优性条件为

$$x - u \in t\partial\|u\|_2 = \begin{cases} \left\{\frac{tu}{\|u\|_2}\right\}, & u \neq 0, \\ \{w : \|w\|_2 \leqslant t\}, & u = 0, \end{cases}$$

因此, 当$\|x\|_2 > t$ 时, $u = x - \frac{tx}{\|x\|_2}$; 当$\|x\|_2 \leqslant t$ 时, $u = 0$. □

邻近算子的例子

- 二次函数(其中 A 对称正定)

$$h(x) = \frac{1}{2}x^{\mathrm{T}}Ax + b^{\mathrm{T}}x + c, \quad \mathrm{prox}_{th}(x) = (I + tA)^{-1}(x - tb)$$

- 负自然对数的和

$$h(x) = -\sum_{i=1}^{n} \ln x_i, \quad \mathrm{prox}_{th}(x)_i = \frac{x_i + \sqrt{x_i^2 + 4t}}{2}, \quad i = 1, 2, \cdots, n$$

邻近算子的计算规则

- 变量的常数倍放缩以及平移 $(\lambda \neq 0)$：

$$h(x) = g(\lambda x + a), \quad \mathrm{prox}_h(x) = \frac{1}{\lambda}\left(\mathrm{prox}_{\lambda^2 g}(\lambda x + a) - a\right)$$

- 函数（及变量）的常数倍放缩 $(\lambda > 0)$：

$$h(x) = \lambda g\left(\frac{x}{\lambda}\right), \quad \mathrm{prox}_h(x) = \lambda \mathrm{prox}_{\lambda^{-1} g}\left(\frac{x}{\lambda}\right)$$

- 加上线性函数：

$$h(x) = g(x) + a^{\mathrm{T}} x, \quad \mathrm{prox}_h(x) = \mathrm{prox}_g(x - a)$$

- 加上二次项 $(u > 0)$

$$h(x) = g(x) + \frac{u}{2}\|x - a\|_2^2, \quad \mathrm{prox}_h(x) = \mathrm{prox}_{\theta g}(\theta x + (1-\theta)a)$$

其中$\theta = \frac{1}{1+u}$

- 向量函数：

$$h\left(\begin{bmatrix} x \\ y \end{bmatrix}\right) = \varphi_1(x) + \varphi_2(y), \quad \mathrm{prox}_h\left(\begin{bmatrix} x \\ y \end{bmatrix}\right) = \begin{bmatrix} \mathrm{prox}_{\varphi_1}(x) \\ \mathrm{prox}_{\varphi_2}(y) \end{bmatrix}$$

复合仿射映射

- 已知函数$g(x)$和矩阵A，设$h(x) = g(Ax + b)$. 在通常情况下，我们不能使用g的邻近算子直接计算关于h的邻近算子.
- 然而，如果有$AA^{\mathrm{T}} = \frac{1}{\alpha} I$（其中$\alpha$为任意正常数），则

$$\mathrm{prox}_h(x) = (I - \alpha A^{\mathrm{T}} A)x + \alpha A^{\mathrm{T}}(\mathrm{prox}_{\alpha^{-1} g}(Ax + b) - b).$$

- 例如，$h(x_1, x_2, \cdots, x_m) = g(x_1 + x_2 + \cdots + x_m)$ 的邻近算子为

$$\mathrm{prox}_h(x_1, x_2, \cdots, x_m)_i = x_i - \frac{1}{m}\left(\sum_{j=1}^{m} x_j - \mathrm{prox}_{mg}\left(\sum_{j=1}^{m} x_j\right)\right).$$

Proof.

考虑如下优化问题：

$$\begin{aligned} \min_{u,y} \quad & g(y) + \frac{1}{2}\|u - x\|^2, \\ \text{s.t.} \quad & Au + b = y, \end{aligned}$$

则其解中的$u = \mathrm{prox}_h(x)$. 固定y 对于u 求极小值，这是一个到仿射集的投影问题，其解为

$$\begin{aligned} u &= x + A^{\mathrm{T}}(AA^{\mathrm{T}})^{-1}(y - b - Ax) \\ &= (I - \alpha A^{\mathrm{T}}A)x + \alpha A^{\mathrm{T}}(y - b). \end{aligned}$$

将其代入优化问题，将目标函数化为

$$g(y) + \frac{\alpha^2}{2}\|A^{\mathrm{T}}(y - b - Ax)\|^2 = g(y) + \frac{\alpha}{2}\|y - b - Ax\|^2.$$

由此得到$y = \mathrm{prox}_{\alpha^{-1}g}(Ax + b)$，再代入$u$的表达式中即可得到结果. □

提纲

Moreau分解

Moreau分解描述了邻近算子与共轭函数之间的关系：

$$x = \text{prox}_h(x) + \text{prox}_{h^*}(x)$$

- 这来自于共轭函数和次梯度的性质：

$$\begin{aligned} u = \text{prox}_h(x) &\iff x - u \in \partial h(u) \\ &\iff u \in \partial h^*(x-u) \\ &\iff x - u = \text{prox}_{h^*}(x) \end{aligned}$$

- 可以由此推出，子空间的正交投影的广义分解式：

$$x = P_L(x) + P_{L^\perp}(x)$$

L为一个子空间，$L^\perp$是它的正交补
（在Moreau分解中有$h = I_L$, $h^* = I_{L^\perp}$,其中I表示示性函数）

广义Moreau分解

定理

对任意的$\lambda > 0$，我们有广义的Moreau分解式：

$$x = \text{prox}_{\lambda f}(x) + \lambda \text{prox}_{\lambda^{-1} f^*}(x/\lambda)$$

Proof.

对 λf 应用Moreau分解

$$\begin{aligned} x &= \text{prox}_{\lambda f}(x) + \text{prox}_{(\lambda f)^*}(x) \\ &= \text{prox}_{\lambda f}(x) + \lambda \text{prox}_{\lambda^{-1} f^*}(x/\lambda) \end{aligned}$$

第二行运用了共轭函数的性质：$(\lambda f)^*(y) = \lambda f^*(y/\lambda)$ □

支撑函数

支撑函数的共轭（在闭凸集上）是示性函数：

$$f(x) = S_C(x) = \sup_{y \in C} x^T y, \qquad f^*(y) = I_C(y)$$

支撑函数的邻近算子：（应用Moreau 分解）

$$\begin{aligned} \text{prox}_{tf}(x) &= x - t\text{prox}_{t^{-1}f^*}(x/t) \\ &= x - tP_C(x/t) \end{aligned}$$

可以发现，支撑函数的邻近算子可以通过投影计算得到例子：$f(x)$ 是 x 最大的 r 个分量的和，则

$$f(x) = x_{[1]} + \cdots + x_{[r]} = S_C(x), \qquad C = \{y | 0 \le y \le 1, 1^T y = r\}$$

范数

范数的共轭是对偶范数球的共轭函数：

$$f(x) = \|c\|, \qquad f^*(x) = I_B(y) \qquad (B = \{y | \|y\|_* \leq 1\})$$

范数的邻近算子：（应用Moreau 分解）

$$\begin{aligned} \text{prox}_{tf}(x) &= x - t\text{prox}_{t^{-1}f^*}(x/t) \\ &= x - tP_B(x/t) \\ &= x - P_{tB}(x) \end{aligned}$$

当$tB = \{x | \|x\| \leq t\}$ 容易计算时，可以用这个公式高效地计算$\text{prox}_{t\|\cdot\|}$

单点距离

距离 （一般范数意义下）

$$f(x) = \|x - a\|$$

邻近算子：令$g(x) = \|x\|$

$$\begin{aligned}\text{prox}_{tf}(x) &= a + \text{prox}_{tg}(x - a) \\ &= a + x - a - tP_B(\frac{x-a}{t}) \\ &= x - P_{tB}(x - a)\end{aligned}$$

B 定义同上一页

集合的 Euclid 距离

Euclid 距离（对于闭凸集C）

$$d(x) = \inf_{y \in C} \|x - y\|_2$$

距离的邻近算子

$$\mathrm{prox}_{td}(x) = \theta P_C(x) + (1-\theta)x, \qquad \theta = \begin{cases} t/d(x) & d(x) \geq t \\ 1 & otherwise \end{cases}$$

平方距离的邻近算子: $f(x) = d(x)^2/2$

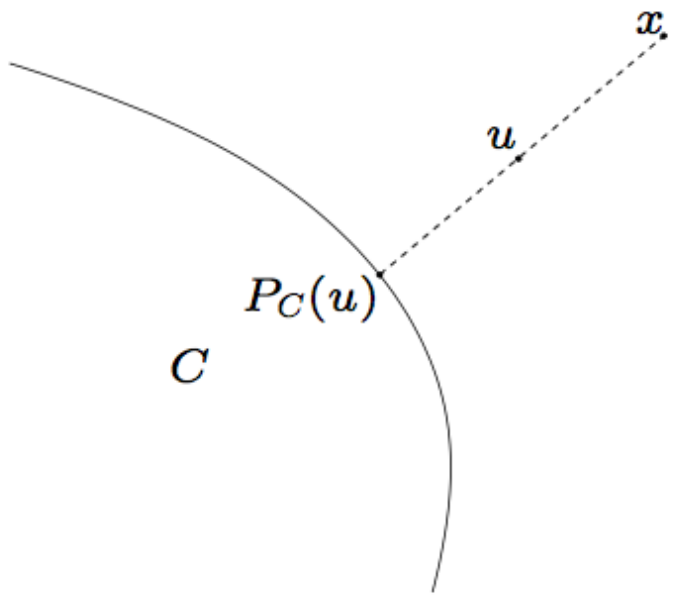
$$\mathrm{prox}_{tf}(x) = \frac{1}{1+t}x + \frac{t}{1+t}P_C(x)$$

证明（关于$\text{prox}_{td}(x)$的表达式）

- 若 $u = \text{prox}_{td}(x) \notin C$, 则有

$$x - u = \frac{t}{d(u)}(u - P_C(u))$$

由此可推出$P_C(u) = P_C(x)$, $d(x) \geq t$, 且 u 是 x 和 $P_C(x)$ 的加权平均



- 若 $u \in C$ 最小化 $d(u) + \frac{1}{2t}\|u - x\|_2^2$, 等价于最小化 $\frac{1}{2t}\|u - x\|_2^2$, 即得 $u = P_C(x)$

证明（关于$\mathrm{prox}_{tf}(x)$的表达式，当$f(x)=d(x)^2/2$时）

$$\begin{aligned}\mathrm{prox}_{tf}(x) &= \arg\min_u \left(\frac{1}{2}d(u)^2+\frac{1}{2t}\|u-x\|_2^2\right)\\ &= \arg\min_u \inf_{v\in C}\left(\frac{1}{2}\|u-v\|_2^2+\frac{1}{2t}\|u-x\|_2^2\right)\end{aligned}$$

最优的 u 可以看成 v 的函数：

$$u=\frac{t}{t+1}v+\frac{1}{t+1}x$$

最优的 v 在集合 C 上极小化

$$\frac{1}{2}\left\|\frac{t}{t+1}v+\frac{1}{t+1}x-v\right\|_2^2+\frac{1}{2t}\left\|\frac{t}{t+1}v+\frac{1}{t+1}x-x\right\|_2^2=\frac{1}{2(1+t)}\|v-x\|_2^2$$

由此即得，$v=P_C(x)$

非凸函数的邻近算子

(适当闭函数的邻近算子) 设 h 是适当闭函数(可以非凸)，且具有有限的下界，即满足$\inf_{x\in\mathbf{dom}h} h(x) > -\infty$, 定义 h 的邻近算子为

$$\mathrm{prox}_h(x) = \underset{u\in\mathbf{dom}\ h}{\arg\min}\left\{h(u) + \frac{1}{2}\|u-x\|^2\right\}.$$

定理

设h 是适当闭函数且$\inf_{x\in\mathbf{dom}h} h(x) > -\infty$, 则$\forall x \in \mathbf{dom}\ h$, $\mathrm{prox}_h(x)$ 是$\mathbb{R}^n$ 上的非空紧集.

Proof.

定义 $g(u) = h(u) + \frac{1}{2}\|u-x\|^2$, 设 $\inf_{x\in\mathbf{dom}h} h(x) = l$.

取$u_0 \in \mathbf{dom}\ h$, 由于$\frac{1}{2}\|u-x\|^2$ 无上界，故$\exists R > 0$, 对$\forall$ 满足$\|u-x\| > R$ 的u, 成立$\frac{1}{2}\|u-x\|^2 > g(u_0) - l$，即$g(u) > g(u_0)$.

这说明下水平集$\{u \mid g(u) \leqslant g(u_0)\}$ 含于球$\|u-x\| \leqslant R$ 内, 即g 有一个非空有界下水平集.

显然$g(u)$ 是闭函数, 由Weierstrass 定理可知, $g(u)$ 的最小值点集合$\mathrm{prox}_h(x)$ 是非空紧集. □

非光滑非凸问题函数的次微分

前面介绍了闭凸函数的邻近算子与次梯度的关系，而对于非凸函数有类似的结论。首先回顾一下非光滑非凸函数的次微分。

次微分

设$f:\mathbb{R}^n\to(-\infty,+\infty]$是适当下半连续函数.

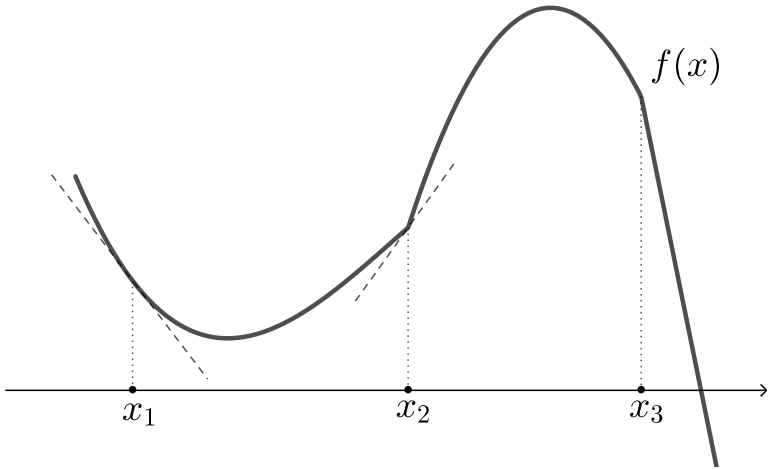
- 对给定的$x\in\mathbf{dom}\,f$, 满足如下条件的所有向量$u\in\mathbb{R}^n$的集合定义为f在点x处的*Fréchet*次微分:
$$\liminf_{y\to x,y\neq x}\frac{f(y)-f(x)-\langle u,y-x\rangle}{\|y-x\|}\geq 0,$$
记为$\hat{\partial}f(x)$.当$x\notin\mathbf{dom}\,f$时, 将$\hat{\partial}f(x)$定义为空集$\varnothing$.

- f在点$x\in\mathbb{R}^n$处的极限次微分(或简称为次微分)定义为
$$\partial f(x)=\{u\in\mathbb{R}^n:\exists\, x^k\to x, f(x^k)\to f(x), u^k\in\hat{\partial}f(x^k)\to u\}.$$
极限次微分通过对x附近的点处的*Fréchet*次微分取极限得到.

- $\hat{\partial} f(x) \subseteq \partial f(x)$, 前者是闭凸集, 后者是闭集.并非在所有的$x \in \mathbf{dom}\, f$ 处都存在*Fréchet* 次微分.
- 凸函数的次梯度要求不等式

$$f(y) \geq f(x) + \langle g, y - x \rangle, \quad g \in \partial f(x)$$

在定义域内全局成立, 而非凸函数只要求在极限意义下成立.
- 当f 是可微函数时, *Fréchet* 次微分和次微分都退化成梯度.



如图，$f(x)$ 在 x_3 处不存在Fréchet 次微分, 但存在次微分

定理

设h 是适当闭函数(可非凸)且有下界, $u \in \mathrm{prox}_h(x)$, 则$x - u \in \partial h(u)$

近似点梯度法

陈士祥

中国科学技术大学

致谢：本教案部分参考北京大学文再文教授《最优化与建模》

提纲

复合优化问题

我们将考虑如下复合优化问题：

$$\min_{x\in\mathbb{R}^n} \quad \psi(x) = f(x) + h(x)$$

- 函数f为可微函数，其定义域$\mathbf{dom}\, f = \mathbb{R}^n$
- 函数h为凸函数，可以是非光滑的，并且邻近算子容易计算
- LASSO问题：$f(x) = \frac{1}{2}\|Ax-b\|^2$，$h(x) = \mu\|x\|_1$
- 次梯度法计算的复杂度：$\mathcal{O}(1/\sqrt{k})$

是否可以设计复杂度为$\mathcal{O}(1/k)$的算法？

邻近算子：回顾

定义

对于一个凸函数h，定义它的邻近算子为

$$\mathrm{prox}_h(x) = \operatorname*{argmin}_u \left(h(u) + \frac{1}{2}\|u - x\|_2^2 \right)$$

例子

- ℓ_1 范数: $h(x) = \|x\|_1, \quad \mathrm{prox}_{th}(x) = \mathrm{sign}(x)\max\{|x| - t, 0\}$
- ℓ_2 范数:$h(x) = \|x\|_2, \quad \mathrm{prox}_{th}(x) = \begin{cases} \left(1 - \frac{t}{\|x\|_2}\right)x, & \|x\|_2 \geqslant t, \\ 0, & \text{其他.} \end{cases}$
- 二次函数(其中 A 对称正定):
  $h(x) = \frac{1}{2}x^{\mathrm{T}}Ax + b^{\mathrm{T}}x + c, \quad \mathrm{prox}_{th}(x) = (I + tA)^{-1}(x - tb)$
- 负自然对数的和: $h(x) = -\sum_{i=1}^n \ln x_i, \quad \mathrm{prox}_{th}(x)_i = \frac{x_i + \sqrt{x_i^2 + 4t}}{2}, \quad i = 1, 2, \cdots, n$

近似点梯度法

对于光滑部分f做梯度下降，对于非光滑部分h使用邻近算子，则近似点梯度法的迭代格式为

$$x^{k+1} = \mathrm{prox}_{t_k h}\left(x^k - t_k \nabla f\left(x^k\right)\right) \tag{1}$$

其中 $t_k > 0$ 为每次迭代的步长，它可以是一个常数或者由线搜索得出.

Algorithm 1 近似点梯度法

1: 输入：函数$f(x), h(x)$, 初始点 x^0.
2: **while** 未达到收敛准则 **do**
3: $\quad x^{k+1} = \mathrm{prox}_{t_k h}\left(x^k - t_k \nabla f\left(x^k\right)\right)$.
4: **end while**

对近似点梯度法的理解

根据定义，(1)式等价于

$$
\begin{aligned}
x^{k+1} &= \arg\min_u \left\{ h(u) + \frac{1}{2t_k} \left\| u - x^k + t_k \nabla f\left(x^k\right) \right\|^2 \right\} \\
&= \arg\min_u \left\{ h(u) + f\left(x^k\right) + \nabla f\left(x^k\right)^{\mathrm{T}} \left(u - x^k\right) + \frac{1}{2t_k} \left\| u - x^k \right\|^2 \right\}
\end{aligned}
$$

根据邻近算子与次梯度的关系, 又可以形式地写成

$$
x^{k+1} = x^k - t_k \nabla f\left(x^k\right) - t_k g^k, \quad g^k \in \partial h\left(x^{k+1}\right).
$$

即对光滑部分做显式的梯度下降，关于非光滑部分做隐式的梯度下降.

步长选取

- 当 f 为梯度L-利普希茨连续函数时，可取固定步长 $t_k = t \leqslant \frac{1}{L}$. 当$L$ 未知时可使用线搜索准则

$$f\left(x^{k+1}\right) \leqslant f\left(x^k\right) + \nabla f\left(x^k\right)^{\mathrm{T}}\left(x^{k+1} - x^k\right) + \frac{1}{2t_k}\left\|x^{k+1} - x^k\right\|^2$$

- 利用BB 步长作为 t_k 的初始估计并用非单调线搜索进行校正：

$$\alpha_{\mathrm{BB1}}^k \overset{\text{def}}{=} \frac{\left(s^{k-1}\right)^{\mathrm{T}} y^{k-1}}{\left(y^{k-1}\right)^{\mathrm{T}} y^{k-1}} \quad \text{或} \quad \alpha_{\mathrm{BB2}}^k \overset{\text{def}}{=} \frac{\left(s^{k-1}\right)^{\mathrm{T}} s^{k-1}}{\left(s^{k-1}\right)^{\mathrm{T}} y^{k-1}},$$

其中$s^{k-1} = x^k - x^{k-1}$ 以及 $y^{k-1} = \nabla f\left(x^k\right) - \nabla f\left(x^{k-1}\right)$.

- 可构造如下适用于近似点梯度法的非单调线搜索准则：

$$\psi(x^{k+1}) \leq C^k - \frac{c_1}{2t_k}\|x^{k+1} - x^k\|^2,$$

$c_1 \in (0,1)$ 为正常数. 注意，定义C^k时需要使用整体函数值$\psi(x^k)$.

提纲

LASSO问题

考虑用近似点梯度法求解 LASSO 问题

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

令 $f(x) = \frac{1}{2}\|Ax-b\|^2, h(x) = \mu\|x\|_1$, 则

$$\nabla f(x) = A^{\mathrm{T}}(Ax-b),$$
$$\mathrm{prox}_{t_k h}(x) = \mathrm{sign}(x)\max\{|x| - t_k\mu, 0\}.$$

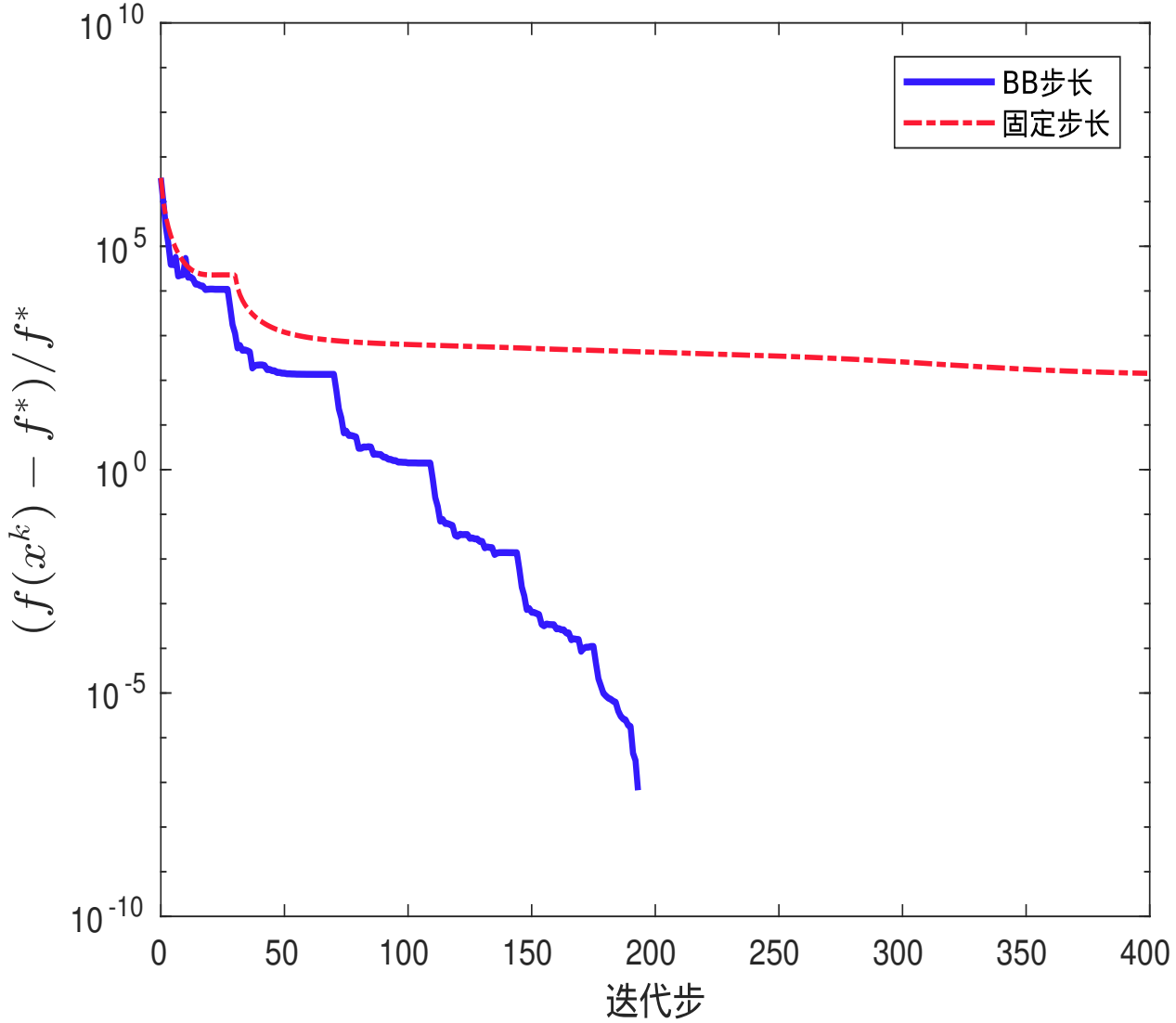
故相应的迭代格式为：

$$y^k = x^k - t_k A^{\mathrm{T}}\left(Ax^k - b\right),$$
$$x^{k+1} = \mathrm{sign}\left(y^k\right)\max\left\{\left|y^k\right| - t_k\mu, 0\right\},$$

即第一步做梯度下降，第二步做收缩

LASSO问题

我们还可以使用BB步长加速收敛



低秩矩阵恢复

考虑低秩矩阵恢复模型:

$$\min_{X\in\mathbb{R}^{m\times n}} \quad \mu\|X\|_* + \frac{1}{2}\sum_{(i,j)\in\Omega}(X_{ij}-M_{ij})^2,$$

其中 M 是想要恢复的低秩矩阵，但是只知道其在下标集 Ω 上的值. 令

$$f(X)=\frac{1}{2}\sum_{(i,j)\in\Omega}(X_{ij}-M_{ij})^2, \quad h(X)=\mu\|X\|_*.$$

定义矩阵$P\in\mathbb{R}^{m\times n}$:

$$P_{ij}=\begin{cases} 1, & (i,j)\in\Omega, \\ 0, & \text{其他}, \end{cases}$$

则

$$f(X)=\frac{1}{2}\|P\odot(X-M)\|_F^2$$

进一步可以得到

$$\nabla f(X) = P \odot (X - M),$$

$$\text{prox}_{t_k h}(X) = U\text{Diag}\left(\max\{|d| - t_k\mu, 0\}\right) V^{\mathrm{T}},$$

其中 $X = U\,\text{Diag}(d)V^{\mathrm{T}}$ 为矩阵 X 的约化的奇异值分解.

由此可以得到近似点梯度法的迭代格式：

$$Y^k = X^k - t_k P \odot \left(X^k - M\right)$$

$$X^{k+1} = \text{prox}_{t_k h}\left(Y^k\right)$$

提纲

收敛性分析

基本假设：

- f 在 $\mathbb{R}^n$ 上是凸的; ∇f 为 L-利普希茨连续，即

$$\|\nabla f(x) - \nabla f(y)\| \leqslant L\|x - y\|, \quad \forall x, y$$

- h 是适当的闭凸函数 (因此 prox_{th} 的定义是合理的);
- 函数 $\psi(x) = f(x) + h(x)$ 的最小值 ψ^* 是有限的，并且在点 x^* 处可取到(并不要求唯一).

梯度映射

在基本假设的基础上，我们定义梯度映射：

$$G_t(x) = \frac{1}{t}\left(x - \mathrm{prox}_{th}(x - t\nabla f(x))\right) \quad (t > 0) \tag{2}$$

不难推出梯度映射具有以下的性质：

- “负搜索方向”：$x^{k+1} = \mathrm{prox}_{th}\left(x^k - t\nabla f\left(x^k\right)\right) = x^k - tG_t\left(x^k\right)$
- 根据邻近算子和次梯度的关系，我们有

$$G_t(x) - \nabla f(x) \in \partial h\left(x - tG_t(x)\right) \tag{3}$$

- 与算法的收敛性的关系：

$$G_t(x) = 0 \iff x \text{ 为 } \psi(x) = f(x) + h(x) \text{ 的最小值点}$$

$$G_t(x) = 0 \iff x \text{ 为 } \psi(x) = f(x) + h(x) \text{ 的最小值点}$$

证明：

$$\begin{aligned} 0 \in \nabla f(x) + \partial h(x) &\iff x - t\nabla f(x) \in x + t\partial h(x) \\ &\iff x - (x - t\nabla f(x)) \in t\partial h(x) \\ &\iff x = \operatorname{prox}_{th}(x - t\nabla f(x)) \\ &\iff G_t(x) = 0. \end{aligned}$$

定理 1(固定步长近似点梯度法的收敛性)

取定步长为 $t_k = t \in \left(0, \frac{1}{L}\right]$，设 $\{x^k\}$ 由迭代格式(1)产生，则

$$\psi\left(x^k\right) - \psi^* \leqslant \frac{1}{2kt}\left\|x^0 - x^*\right\|^2$$

定理证明

证明　根据利普希茨连续“二次上界”的性质，得到

$$f(y) \leqslant f(x) + \nabla f(x)^{\mathrm{T}}(y-x) + \frac{L}{2}\|y-x\|^2, \quad \forall x, y \in \mathbb{R}^n$$

令 $y = x - tG_t(x)$, 有

$$\begin{aligned} f\left(x - tG_t(x)\right) &\leqslant f(x) - t\nabla f(x)^{\mathrm{T}} G_t(x) + \frac{t^2 L}{2}\left\|G_t(x)\right\|^2 \\ &\leqslant f(x) - t\nabla f(x)^{\mathrm{T}} G_t(x) + \frac{t}{2}\left\|G_t(x)\right\|^2 \end{aligned} \tag{4}$$

此外，由 $f(x), h(x)$ 为凸函数，结合(4)式我们有

$$h\left(x - tG_t(x)\right) \leqslant h(z) - \left(G_t(x) - \nabla f(x)\right)^{\mathrm{T}}\left(z - x + tG_t(x)\right) \tag{5}$$

$$f(x) \leqslant f(z) - \nabla f(x)^{\mathrm{T}}(z-x) \tag{6}$$

将(4)(5)(6)式相加可得对任意 $z \in \mathbf{dom}\ \psi$ 有

$$\psi\left(x - tG_t(x)\right) \leqslant \psi(z) + G_t(x)^{\mathrm{T}}(x-z) - \frac{t}{2}\left\|G_t(x)\right\|^2 \tag{7}$$

定理证明

由$x^i = x^{i-1} - tG_t(x^{i-1})$，在不等式(7) 中，取 $z = x^*, x = x^{i-1}$ 得到

$$\begin{aligned}\psi(x^i) - \psi^* &\leqslant G_t(x^{i-1})^{\mathrm{T}}\left(x^{i-1} - x^*\right) - \frac{t}{2}\left\|G_t(x^{i-1})\right\|^2 \\ &= \frac{1}{2t}\left(\left\|x^{i-1} - x^*\right\|^2 - \left\|x^{i-1} - x^* - tG_t(x^{i-1})\right\|^2\right) \\ &= \frac{1}{2t}\left(\left\|x^{i-1} - x^*\right\|^2 - \left\|x^i - x^*\right\|^2\right)\end{aligned} \tag{8}$$

取 $i = 1, 2, \cdots, k$ 并累加，得

$$\begin{aligned}\sum_{i=1}^{k}\left(\psi\left(x^i\right) - \psi^*\right) &\leqslant \frac{1}{2t}\sum_{i=1}^{k}\left(\left\|x^{i-1} - x^*\right\|^2 - \left\|x^i - x^*\right\|^2\right) \\ &= \frac{1}{2t}\left(\left\|x^0 - x^*\right\|^2 - \left\|x^k - x^*\right\|^2\right) \\ &\leqslant \frac{1}{2t}\left\|x^0 - x^*\right\|^2.\end{aligned}$$

定理证明

注意到在不等式(7)中，取 $z = x^{i-1}$ 即得：

$$\psi(x^i) \leqslant \psi(x^{i-1}) - \frac{t}{2}\left\|G_t(x^{i-1})\right\|^2$$

即 $\psi\left(x^i\right)$ 不增，因此

$$\psi\left(x^k\right) - \psi^* \leqslant \frac{1}{k}\sum_{i=1}^{k}\left(\psi\left(x^i\right) - \psi^*\right) \leqslant \frac{1}{2kt}\left\|x^0 - x^*\right\|^2$$

步长选取

定理**1**中要求 $t \leq \frac{1}{L}$，而根据定理证明的过程，也可以用线搜索准则：

- 从某个 $t = \hat{t} > 0$ 开始进行回溯$(t \leftarrow \beta t)$, 直到满足不等式

$$f\left(x - tG_t(x)\right) \leqslant f(x) - t\nabla f(x)^{\mathrm{T}} G_t(x) + \frac{t}{2}\left\|G_t(x)\right\|^2 \tag{9}$$

- 这等价于算法部分提到的线搜索准则：

$$f\left(x^{k+1}\right) \leqslant f\left(x^k\right) + \nabla f\left(x^k\right)^{\mathrm{T}}\left(x^{k+1} - x^k\right) + \frac{1}{2t_k}\left\|x^{k+1} - x^k\right\|^2$$

至此我们解释了该线搜索准则的合理性

定理 2 (非固定步长的近似点梯度法的收敛性)

从某个 $t=\hat{t}>0$ 开始进行回溯$(t \leftarrow \beta t)$ 直到满足不等式(9), 设 $\left\{x^k\right\}$ 是由迭代格式(1) 产生的序列，则

$$\psi\left(x^k\right)-\psi^* \leqslant \frac{1}{2k \min\{\hat{t}, \beta/L\}}\left\|x^0-x^*\right\|^2$$

Proof.

由定理**1**的证明,当$0<t \leqslant \frac{1}{L}$ 时,不等式(9)成立,故由线搜索所得的步长 t 应满足 $t \geqslant t_{\min }=\min \left\{\hat{t}, \frac{\beta}{L}\right\}$. 同理,我们有$\psi\left(x^i\right)$单调不增,且

$$\psi\left(x^i\right)-\psi^* \leqslant \frac{1}{2t_{\min }}\left(\left\|x^{i-1}-x^*\right\|^2-\left\|x^i-x^*\right\|^2\right)$$

取$i=1,2, \cdots, k$并累加，并利用$\psi\left(x^i\right)$ 不增，可得

$$\psi\left(x^k\right)-\psi^* \leqslant \frac{1}{2kt_{\min }}\left\|x^0-x^*\right\|^2$$

□

提纲

近似点映射的非扩张性

定理**4.1.** 考虑一个适当的、闭凸函数$h:\mathbb{R}^n \to \mathbb{R}$。集合$\text{prox}_h(x)$在每个点$x \in E$是唯一的。此外，对任意的点$x, y \in E$，估计如下成立：

$$\|\text{prox}_h(x) - \text{prox}_h(y)\|^2 \leq \langle \text{prox}_h(x) - \text{prox}_h(y), x - y \rangle.$$

特别地，近端映射$x \mapsto \text{prox}_h(x)$是1-Lipschitz 连续的。

证明

证明: 注意到对于每个点x，函数$z \mapsto h(z) + \frac{1}{2}\|z - x\|^2$ 是适当的、封闭的并且1-强凸的，因此有一个唯一的极小值点。因此近端集合prox_h 在$\mathbb{R}^n$ 上是唯一的。接着，考虑$\mathbb{R}^n$ 中的任意两点x, y 并定义$x^+ = \text{prox}_h(x)$ 和$y^+ = \text{prox}_h(y)$。根据定义，x^+ 是函数$h + \frac{1}{2}\|\cdot - x\|^2$ 的极小值点，y^+ 是$h + \frac{1}{2}\|\cdot - y\|^2$ 的极小值点。
我们有关系

$$x - x^+ \in \partial h(x^+), \quad y - y^+ \in \partial h(y^+).$$

根据次梯度的单调性，我们有

$$\langle x^+ - y^+, g_1 - g_2 \rangle \geq 0, \forall g_1 \in \partial h(x^+), g_2 \in \partial h(y^+)$$

带入上面的关系，

$$\langle x^+ - y^+, x - x^+ - (y - y^+) \rangle \geq 0,$$

化简即

$$\|\text{prox}_h(x) - \text{prox}_h(y)\|^2 \leq \langle \text{prox}_h(x) - \text{prox}_h(y), x - y \rangle.$$

使用Cauchy 不等式，得到

$$\|\text{prox}_h(x) - \text{prox}_h(y)\| \leq \|x - y\|.$$

故近端映射$x \mapsto \text{prox}_h(x)$ 是1-Lipschitz 连续的。

Moreau envelope

我们定义Moreau包络函数如下：

$$h_\alpha(x) := \min_y h(y) + \frac{1}{2\alpha}\|x-y\|^2.$$

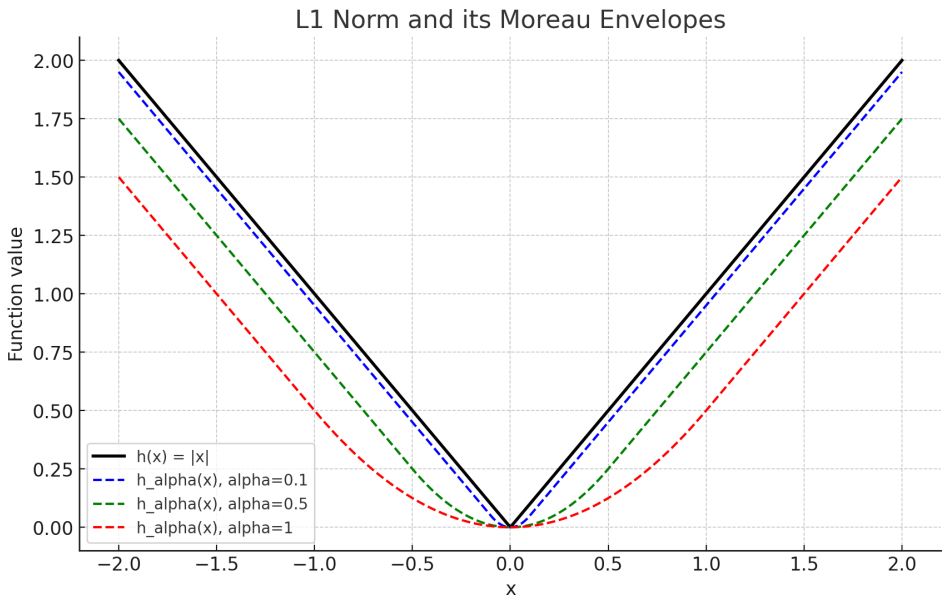


Figure: 函数$f(x)=|x|$和其Moreau包络函数。Moreau包络函数可以看成是对非光滑函数的逼近。

定理： 对于任何适当的、闭的、凸函数$h:\mathbb{R}^n\to\mathbb{R}$，Moreau 包络$h_\alpha$ 在$\mathbb{R}^n$ 上连续可微，其梯度为

$$\nabla h_\alpha(x) = \alpha^{-1}(x-\text{prox}_{\alpha h}(x)).$$

定理证明：

首先假设$\alpha = 1$，给定任意点$x \in \mathbb{R}^n$。我们的目标是说明f_α 的次梯度在每一个点处都是单点集。为此，我们依次推导出等价关系

$$
\begin{aligned}
& z \in \partial f_\alpha(x) \\
\Leftrightarrow\ & x \in \partial(f \square \tfrac{1}{2}\|\cdot\|^2)(z) \quad (4.1) \\
\Leftrightarrow\ & x \in \partial(f^* + \tfrac{1}{2}\|\cdot\|^2)(z) \quad (4.2) \\
\Leftrightarrow\ & x \in \partial f^*(z) + z \\
\Leftrightarrow\ & 0 \in \partial(f^* + \tfrac{1}{2}\|\cdot - x\|^2)(z) \\
\Leftrightarrow\ & z = \mathrm{prox}_{f^*}(x) \\
\Leftrightarrow\ & z = x - \mathrm{prox}_{\alpha f}(x), \quad (4.3)
\end{aligned}
$$

其中(4.1)的符号$f \square \frac{1}{2}\|\cdot\|^2$表示卷积。第二个等价关系(4.2) 使用了共轭公式（lecture 9-第15页）。最后一个等价关系(4.3) 来自于Moreau分解定理（lecture 9-第28页）。因此我们推出f_α 的次梯度是一个单射。因此，f_α 的包络是可微的，并且$\nabla f_\alpha(x) = x - \mathrm{prox}_f(x)$。因此从定理4.1 推出$\nabla f_\alpha$ 是1-Lipschitz 连续的。回到一般设置$\alpha \neq 1$，观察$\alpha \cdot f$ 是Moreau-Yosida 包络用参数1。应用我们已经对$\alpha = 1$ 证明的情况来完成证明。

该定理说明，近似点梯度法求解非光滑优化问题$\min h(x)$时，可以看成是$h_\alpha(x)$的梯度法。因为

$$x_{k+1} = \mathrm{prox}_{\alpha h}(x_k) = x_k - \alpha \nabla h_\alpha(x_k)$$

非凸函数的近似点梯度法

(适当闭函数的邻近算子) 设 h 是适当闭函数(可以非凸)，且具有有限的下界，即满足$\inf_{x\in\mathbf{dom}h} h(x) > -\infty$, 定义 h 的邻近算子为

$$\mathrm{prox}_h(x) = \underset{u\in\mathbf{dom}\ h}{\arg\min}\left\{h(u) + \frac{1}{2}\|u-x\|^2\right\}.$$

- $\mathrm{prox}_h(x)$良定义，且是$\mathbb{R}^n$ 上的非空紧集
- 对$u \in \mathrm{prox}_h(x)$,有$x-u \in \partial h(u)$. ∂h表示h(包括非凸情形)的次微分

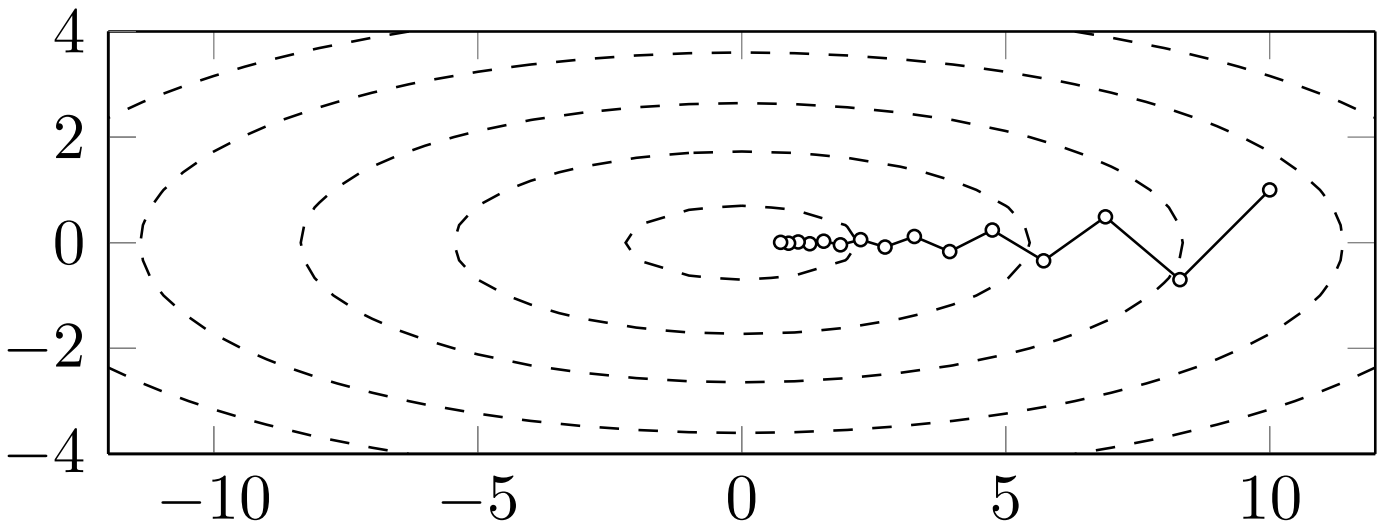
对于复合优化问题$\min\ \psi(x) = f(x) + h(x)$, f 可微, h 为适当闭函数(可非凸)。与一般的近似点梯度法类似，有迭代格式：

$$x^{k+1} = \mathrm{prox}_{t_k h}\left(x^k - t_k\nabla f\left(x^k\right)\right)$$

迭代时往往选取 $\mathrm{prox}_{t_k h}$ 中的一个元素，此时算法也具有收敛性。

近似梯度法的缺点

一般梯度法的缺点：L_2距离作为逼近不够精确，其并没有反应函数曲率的变化。
例如，二次函数，$f(x,y)=\frac{1}{2}x^TQx$, Q是正定矩阵



梯度法的收敛速度与条件数$\kappa=\frac{\lambda_{\max}(Q)}{\lambda_{\min}(Q)}$相关。

Bregman 散度

定义：设$r: C \to \mathbb{R}$在C上是严格凸的且可微的，则

$$D_r(x,z) := r(x) - r(z) - \nabla r(z) \cdot (x - z)$$

为Bregman 散度。

Bregman 散度与平方欧几里得距离有一些相似性：如果$r(x) = \frac{1}{2}\|x\|^2$，那么$D_r(x,z) = \frac{1}{2}\|x - z\|^2$。
Bregman 散度是一种局部二次度量，可以将其视为

$$D_r(x,z) = \frac{1}{2}(x - z)^T \nabla^2 r(\xi)(x - z)$$

对于某些依赖于x和z的ξ。
r的严格凸性确保了当$x = z$时，$D_r(x,z) = 0$。

例子：KL散度

设$D_r(x,z)=\mathsf{KL}(x,z):=\sum_i x_i\log\frac{x_i}{z_i}$，它由下面的函数$r:C\to\mathbb{R}$生成：

$$r(x)=\sum_i x_i\log x_i\ （负熵）$$

如果$C=\Delta:=\{x\in\mathbb{R}^n_+\mid\sum_i x_i=1\}$是概率单纯形。
证明：

$$D_r(x,z)=r(x)-r(z)-\nabla r(z)\cdot(x-z)$$

$$=\sum_i x_i\log x_i-\sum_i z_i\log z_i-\left(\sum_i\log z_i+1\right)(x_i-z_i)$$

$$=-\sum_i x_i+\sum_i z_i+\sum_i x_i\log\frac{x_i}{z_i}=\mathsf{KL}(x,z)$$

例子：平方马氏（Mahalanobis）距离

设$D_r(x,z)=\frac{1}{2}(x-z)^TQ(x-z)$ 对于某个矩阵Q，由生成函数

$$r(x)=\frac{1}{2}x^TQx$$

得到。
证明：

$$\begin{aligned}D_r(x,z)&=r(x)-r(z)-\nabla r(z)\cdot(x-z)\\&=\frac{1}{2}x^TQx-\frac{1}{2}z^TQz-z^TQ(x-z)\\&=\frac{1}{2}(x-z)^TQ(x-z).\end{aligned}$$

例子：冯·诺伊曼（von Neumann）散度

如果$C = S_+^n$（正定锥），则由特征值的广义负熵

$$r(X) = \sum_{i=1}^{n} \lambda_i(X)\log(\lambda_i(X)) - \lambda_i(X) := \mathsf{Tr}(X\log X - X)$$

生成的是冯·诺伊曼散度（在量子力学中常用）。
$D_r(X,Z)$ 定义为

$$\begin{aligned} D_r(X,Z) &= \mathsf{Tr}(X\log X - X) - \mathsf{Tr}(Z\log Z - Z) - \mathsf{Tr}((X-Z)\log Z) \\ &= \mathsf{Tr}(X\log X - \log Z) - X + Z). \end{aligned}$$

使用了$\nabla r(X) = \log X$ 的事实。

Bregman 散度的常见族

这个表列出了常见的Bregman 散度函数及其名称。

Function Name	$\varphi(x)$	dom φ	$D_\varphi(x;y)$
Squared norm	$\frac{1}{2}x^2$	$(-\infty,+\infty)$	$\frac{1}{2}(x-y)^2$
Shannon entropy	$x\log x - x$	$[0,+\infty)$	$x\log\frac{x}{y} - x + y$
Bit entropy	$x\log x + (1-x)\log(1-x)$	$[0,1]$	$x\log\frac{x}{y} + (1-x)\log\frac{1-x}{1-y}$
Burg entropy	$-\log x$	$(0,+\infty)$	$\frac{x}{y} - \log\frac{x}{y} - 1$
Hellinger	$-\sqrt{1-x^2}$	$[-1,1]$	$(1-xy)(1-y^2)^{-1/2} - (1-x^2)^{1/2}$
ℓ_p quasi-norm	$-x^p \quad (0<p<1)$	$[0,+\infty)$	$-x^p + p\,xy^{p-1} - (p-1)\,y^p$
ℓ_p norm	$\lvert x\rvert^p \quad (1<p<\infty)$	$(-\infty,+\infty)$	$\lvert x\rvert^p - p\,x\,\operatorname{sgn} y\,\lvert y\rvert^{p-1} + (p-1)\,\lvert y\rvert^p$
Exponential	$\exp x$	$(-\infty,+\infty)$	$\exp x - (x-y+1)\exp y$
Inverse	$1/x$	$(0,+\infty)$	$1/x + x/y^2 - 2/y$

Figure: 常见散度， from I. Dhillon & J. Tropp, 2007。

、

镜像下降算法

考虑如下的凸优化问题：

$$\min \quad f(x), \quad \text{s.t.} \quad x \in C,$$

f 为凸函数，C 是 $\mathbf{dom}\, f$ 的凸子集，且 f 在 C 上存在次梯度.

Bregman距离: 令r为可微凸函数，由r产生的 **Bregman距离**为：

$$D_r(y,x) = r(y) - r(x) - \nabla r(x)^T(y-x)$$

镜像(非线性)次梯度方法：

1. 取次梯度$g^k \in \partial f(x^k)$
2. 更新迭代格式：

$$x^{k+1} = \operatorname*{argmin}_{x \in C} \left\{ (g^k)^T(x - x^k) + \frac{1}{\alpha_k} D_r(x, x^k) \right\}.$$

取$r(x) = \frac{1}{2}||x||_2^2$时，这就是投影次梯度法。

例：KL散度，子问题

当$D_r(x,z)=\mathbf{KL}(x||z)$，$C=\Delta$，且$f$可微分时，镜像下降（**MD**）具有闭合形式：
如果$C=\Delta:=\{x\in\mathbb{R}^n_+\mid\sum_i x_i=1\}$是概率单纯形，且$f$是可微分的，那么镜像下降更新可以写为

$$x_i^{k+1}=\frac{x_i^k\exp(-\alpha_k\nabla f(x^k)_i)}{\sum_{j=1}^n x_j^k\exp(-\alpha_k\nabla f(x^k)_j)}$$

其中$1\le i\le n$，并且α_k是步长。
这个更新规则通常被称为指数梯度下降、熵下降或者乘法权重更新（MWU）。

镜像梯度法另一种理解

使用Bregman散度，也可以这样描述MD：

$$\nabla r(y_{k+1}) = \nabla r(x_k) - \eta_k g_k \quad \text{其中} g_k \in \partial f(x_k) \tag{5.1}$$

$$x_{k+1} \in \mathcal{P}_{C,r}(y_{k+1}) = \arg\min_{z \in C} D_r(z, y_{k+1}) \tag{5.2}$$

$\mathcal{P}_{C,r}$表示“广义”距离下的投影算子。

- 在某个“对偶”空间执行梯度下降

通过查看最优条件可以看出等价性

- (5.2)的最优条件给出

$$\begin{aligned} 0 &\in \nabla r(x_{k+1}) - \nabla r(y_{k+1}) + N_C(x_{k+1}) \quad \text{(参见Bertsekas '16)} \\ &= \nabla r(x_{k+1}) - \nabla r(x_k) + \eta_k g_k + N_C(x_{k+1}), \end{aligned} \tag{10}$$

  其中，$N_C(x_{k+1})$表示约束C的正则法锥，其等于指示函数I_C的次微分集合。

- 原始的MD迭代中，子问题的最优条件是

$$0 \in g_k + \frac{1}{\eta_k}(\nabla r(x_{k+1}) - \nabla r(x_k) + N_C(x_{k+1})) \quad \text{(参见Bertsekas '16)} \tag{11}$$

- 这两个条件显然是相同的

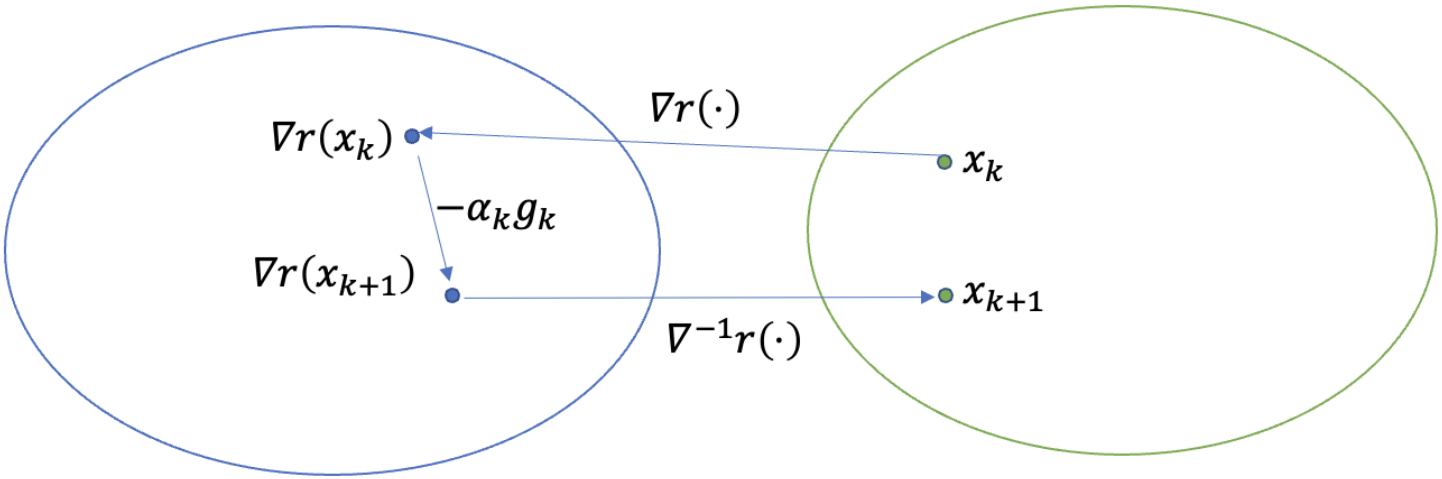
$C = \mathbb{R}^n$的情形

为了简化，假设$C = \mathbb{R}^n$，则镜像梯度法的另一种形式是

$$x_{k+1} = \nabla r^*(\nabla r(x_k) - \eta g_k) \tag{5.3}$$

其中$r^*(x) := \sup_z\{z^T x - r(z)\}$是$r$的Fenchel共轭

- 这是Nemirovski和Yudin在1983年最初提出的版本



当$C = \mathbb{R}^n$时，(5.1)-(5.2)简化为

$$x_{k+1} = y_{k+1} = (\nabla r)^{-1}(\nabla r(x_k) - \eta g_k)$$

因此，只需证明

$$(\nabla r)^{-1} = \nabla r^* \tag{5.4}$$

假设$y = \nabla r(x_k)$。根据共轭子梯度定理，这等价于（作业2.5），

$$x_k = \nabla r^*(y) \tag{12}$$

这意味着

$$x_k = \nabla r^*(y) = \nabla r^* \nabla r(x_k) \tag{13}$$

因此，$\nabla r^* = (\nabla r)^{-1}$。

函数 r 需要满足的性质：在范数$||\cdot||$ 的意义下强凸，即

$$r(y) \geq r(x) + \nabla r(x)^T(y-x) + \frac{1}{2}||x-y||^2.$$

考虑计算与任意点(包括最优点)处函数值的距离：对于任意的 $x^* \in C$,

$$\begin{aligned} f(x^k) - f(x^*) &\leq (g^k)^T(x^k - x^*) \\ &= (g^k)^T(x^{k+1} - x^*) + (g^k)^T(x^k - x^{k+1}) \end{aligned}$$

根据 x^{k+1} 点处最优性的必要条件:

$$(\alpha_k g^k + \nabla r(x^{k+1}) - \nabla r(x^k))^T(y - x^{k+1}) \geq 0, \forall y \in C$$

因此，我们取 $y = x^*$，得

$$(g^k)^T(x^{k+1} - x^*) \leq \frac{1}{\alpha_k}(\nabla r(x^{k+1}) - \nabla r(x^k))^T(x^* - x^{k+1})$$

收敛性分析(续)

进一步，

$$
\begin{aligned}
&(\nabla r(x^{k+1}) - \nabla r(x^k))^T (x^* - x^{k+1}) \\
&= D_r(x^*, x^k) - D_r(x^*, x^{k+1}) - D_r(x^k, x^{k+1})
\end{aligned}
$$

综合以上各式，对任意 $x^* \in C$,

$$
\begin{aligned}
f(x^k) - f(x^*) &\leq (g^k)^T (x^{k+1} - x^*) + (g^k)^T (x^k - x^{k+1}) \\
&\leq \frac{1}{\alpha_k} \left[D_r(x^*, x^k) - D_r(x^*, x^{k+1}) \right] - \frac{1}{\alpha_k} D_r(x^k, x^{k+1}) \\
&\quad + (g^k)^T (x^k - x^{k+1}).
\end{aligned}
$$

应用Fenchel-Young不等式 ($x^T y \leq \frac{1}{2\alpha}||x||^2 + \frac{\alpha}{2}||y||_*^2$)

$$
\begin{aligned}
&\leq \frac{1}{\alpha_k} \left[D_r(x^*, x^k) - D_r(x^*, x^{k+1}) \right] - \frac{1}{\alpha_k} D_r(x^k, x^{k+1}) \\
&\quad + \frac{\alpha_k}{2} ||g^k||_*^2 + \frac{1}{2\alpha_k} ||x^k - x^{k+1}||^2 \\
&\leq \frac{1}{\alpha_k} \left[D_r(x^*, x^k) - D_r(x^*, x^{k+1}) \right] + \frac{\alpha_k}{2} ||g^k||_*^2
\end{aligned}
$$

保证收敛的条件

取固定步长 $\alpha_k = \alpha$,并在上面的不等式中对$1, \ldots, k$求和，得

$$\frac{1}{k}\sum_{i=1}^{k} f(x^i) - f(x^*) \leq \frac{1}{\alpha k} D_h(x^*, x^1) + \frac{\alpha}{2} \max_i ||g^i||_*^2$$

一般而言，

$$f^{\text{best},k} - f^* \leq \frac{D_h\left(x^*, x^1\right) + \frac{1}{2}\sum_{i=1}^{k} \alpha_i^2 \max_i ||g^i||_*^2}{\sum_{i=1}^{k} \alpha_i}$$

当以下条件满足时，算法可以保证收敛

- $D_h(x^*, x^1) < \infty$
- $\sum_k \alpha_k = \infty$ 且$\alpha_k \to 0$ (消失步长)
- 对于任意$g \in \partial f(x)$ 和$x \in C$, 有$||g||_* \leq G$恒成立，其中$G < \infty$

镜像下降算法的例子

- 通常的(投影)次梯度法:取 $r(x) = \frac{1}{2}||x||_2^2$
- 使用单纯形约束，$C = \{x \in R_+^n | \mathbf{1}^T x = 1\}$，并使用负熵函数

$$h(x) = \sum_{i=1}^{n} x_i \log x_i$$

1. ℓ_1范数意义下为强凸函数
2. 对于初始点$x^1 = 1/n$, 有$D_r(x^*, x^1) \leq \log n$ 对任意$x^* \in C$ 成立
3. 若$G_\infty \geq ||g||_\infty$ 对任意$g \in \partial f(x)$，$x \in C$成立, 即存在有限上界，则

$$f_{best}^k - f^* \leq \frac{\log n}{\alpha k} + \frac{\alpha}{2k} G_\infty$$

4. 比通常的次梯度算法表现好很多

加速梯度算法

陈士祥

中国科学技术大学

致谢：本教案部分参考北京大学文再文教授《最优化与建模》

提纲

（近端）梯度方法的迭代复杂度

复杂度：要求某种收敛准则小于ϵ所需要的迭代步数.

- 强凸且光滑问题：$O\left(\kappa \log \frac{1}{\epsilon}\right)$
- 凸且光滑问题：$O\left(\frac{1}{\epsilon}\right)$

是否还有希望进一步加速收敛?

问题及可能的解决方案

问题：

- 梯度下降（GD）关注于改进每次迭代的成本，这有时可能太“短视”。
- GD有时可能会出现锯齿状或经历突然变化。

解决方案：

- 利用历史信息（即过去的迭代）。
- 添加缓冲（如动量）以产生更平滑的轨迹。

重球法（heavy ball method）

求解无约束光滑问题

$$\min_{x\in\mathbb{R}^n} f(x)$$

Heavy ball 算法：

$$x_{k+1} = x_k - \eta_k \nabla f(x_k) + \theta_k (x_k - x_{k-1})$$

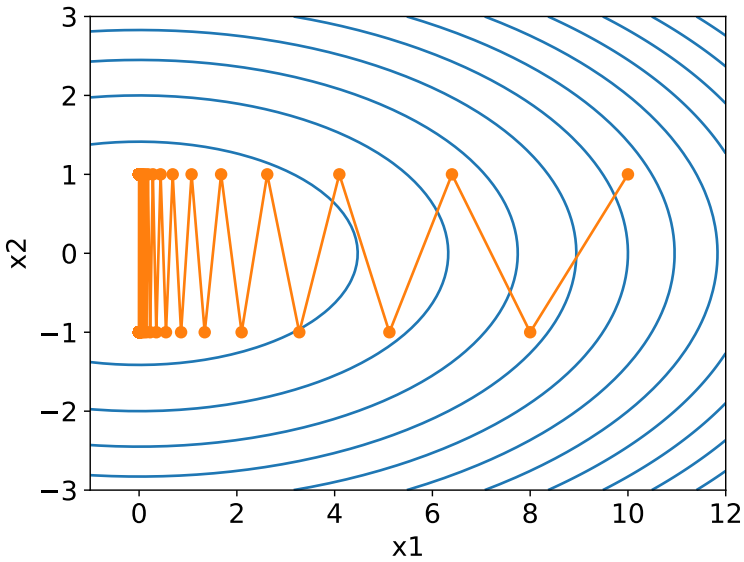
其中$\theta_k(x_k - x_{k-1})$ 是动量项。$x_{-1} = x_0$.

- Polyak '64 提出重球方法，也被称为Polyak 动量法（momentum method）。
- 参考文献：B. Polyak. Some methods of speeding up the convergence of iteration methods. USSR Computational Mathematics and Mathematical Physics, 4(5):1–17, 1964.
- 为“球”添加惯性（即包含一个动量项）以减轻锯齿状变化。
- 利用差分$\ddot{x}(t) \approx \frac{x_{k+1}-2x_k+x_{k-1}}{h_1}, \dot{x}(t) \approx \frac{x_k - x_{k-1}}{h_2}$，重球法可以理解为二阶微分方程：

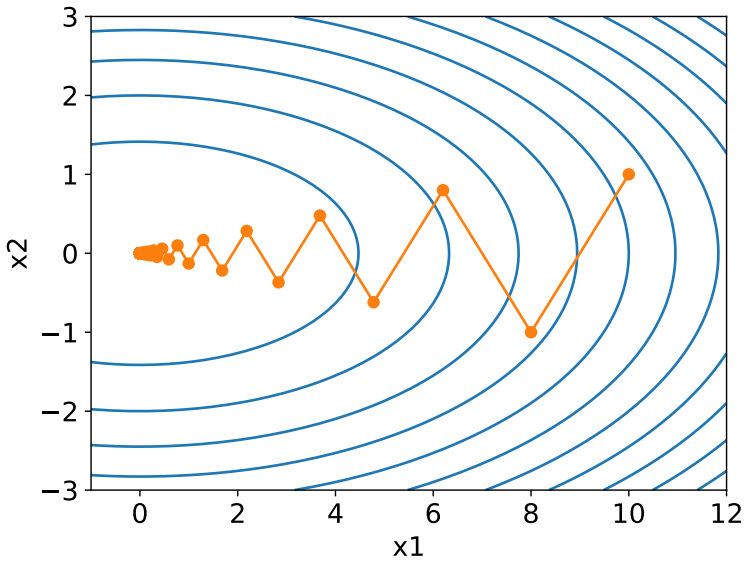
$$\beta\ddot{x}(t) + \dot{x}(t) = -\nabla f(x(t)), \text{利用} \eta_k = h_1/\beta, \theta_k = \frac{h_1}{\beta h_2} - 1.$$

β可以理解为物体的质量，$\ddot{x}(t)$为物体加速度，$\dot{x}(t)$为速度。二阶项$\beta\ddot{x}(t)$可以理解为物体的“惯性”，因此系统更加平滑。

图1(b)展示的是动量梯度法求解函数$x_1^2 + 10x_2^2$最小值的迭代轨迹。与图1(a)中的梯度法相对比，动量梯度法轨迹在x_2更加缓和，因为动量随机梯度法在历史相同的梯度方向累积起来形成动量，而梯度变化快的方向相互抵消，缓解了振荡现象。总体上来说，动量梯度法只需添加一个参数θ_k, 其余参数设定方式可与SGD相似。



(a) 梯度法迭代轨迹示意图



(b) 动量梯度法迭代轨迹示意图

Figure: 梯度法和重球法在函数$x_1^2 + 10x_2^2$迭代轨迹

二次问题的直观理解

考虑如下二次正定问题

$$\min_{x\in\mathbb{R}^n} \frac{1}{2}(x-x^*)^T Q(x-x^*)$$

其中$Q \succ 0$有条件数κ

重球法可以写为如下的系统：

$$\begin{bmatrix} x_{k+1} \\ x_k \end{bmatrix} = \begin{bmatrix} (1+\theta_k)I & -\theta_k I \\ I & 0 \end{bmatrix} \begin{bmatrix} x_k \\ x_{k-1} \end{bmatrix} - \begin{bmatrix} \eta_k \nabla f(x_k) \\ 0 \end{bmatrix}$$

或等价地，

$$\begin{aligned} \begin{bmatrix} x_{k+1} - x^* \\ x_k - x^* \end{bmatrix} &= \begin{bmatrix} (1+\theta_k)I & -\theta_k I \\ I & 0 \end{bmatrix} \begin{bmatrix} x_k - x^* \\ x_{k-1} - x^* \end{bmatrix} - \begin{bmatrix} \eta_k \nabla f(x_k) \\ 0 \end{bmatrix} \\ &= \begin{bmatrix} (1+\theta_k)I - \eta_k Q & -\theta_k I \\ I & 0 \end{bmatrix} \begin{bmatrix} x_k - x^* \\ x_{k-1} - x^* \end{bmatrix} \end{aligned}$$

系统矩阵

定义系统矩阵H_k为

$$H_k := \begin{bmatrix} (1+\theta_k)I - \eta_k Q & -\theta_k I \\ I & 0 \end{bmatrix}$$

则

$$\begin{bmatrix} x_{k+1} - x^* \\ x_k - x^* \end{bmatrix} = H_k \begin{bmatrix} x_k - x^* \\ x_{k-1} - x^* \end{bmatrix}$$

意义：重球方法的收敛依赖于系统矩阵H_k的谱范数。
关键思想：找到适当的步长η_k和动量系数θ_k来控制H_k的谱范数。

对于二次问题的收敛性

定理**11.1**（二次函数的重球方法的收敛性）.
假设f 是一个L-光滑且μ-强凸的二次函数。令$\eta_k \equiv \frac{4}{(\sqrt{L}+\sqrt{\mu})^2}$,
$\theta_k \equiv \max\left\{\left|1-\sqrt{\eta_k L}\right|, \left|1-\sqrt{\eta_k \mu}\right|\right\}^2$，且$\kappa = L/\mu$。那么

$$\left\|\begin{bmatrix} x_{k+1}-x^* \\ x_k - x^* \end{bmatrix}\right\|_2 \leq \left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^k \left\|\begin{bmatrix} x_1-x^* \\ x_0 - x^* \end{bmatrix}\right\|_2$$

- 迭代复杂度：$O\left(\sqrt{\kappa}\log\frac{1}{\epsilon}\right)$
- 相比GD 有显著改善：$O\left(\sqrt{\kappa}\log\frac{1}{\epsilon}\right)$ vs. $O\left(\kappa\log\frac{1}{\epsilon}\right)$
- 参数设置需要预先知道L 和μ

证明

定理11.1的证明考虑系统矩阵H_k 的谱来控制收敛。让λ_i 是Q 的第i 个特征值，设

$$\Lambda := \mathrm{Diag}(\begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_n \end{bmatrix}),$$

则H_k 的谱半径（用$\rho(\cdot)$ 表示）满足

$$\rho(H_k) = \rho\left(\begin{bmatrix} (1+\theta_k)I - \eta_k\Lambda & -\theta_k I \\ I & 0 \end{bmatrix}\right) \leq \max_{1\leq i\leq n} \rho\left(\begin{bmatrix} 1+\theta_k-\eta_k\lambda_i & -\theta_k \\ 1 & 0 \end{bmatrix}\right)$$

要完成证明，只需说明

$$\max_i \rho\left(\begin{bmatrix} 1+\theta_k-\eta_k\lambda_i & -\theta_k \\ 1 & 0 \end{bmatrix}\right) \leq \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}$$

证明(续)

为了证明上述不等式，注意到

$$\begin{bmatrix} 1+\theta_k-\eta_k\lambda_i & -\theta_k \\ 1 & 0 \end{bmatrix}$$

的两个特征值是方程$z^2-(1+\theta_k-\eta_k\lambda_i)z+\theta_k=0$的根。
如果$(1+\theta_k-\eta_k\lambda_i)^2\le 4\theta_k$，则这个方程的根具有相同的模长$\sqrt{\theta_k}$（不管它们是共轭的还是只有一个根）。
此外，可以说明，如果

$$\theta_k\in\left[\left(1-\sqrt{\eta_k\lambda_i}\right)^2,\left(1+\sqrt{\eta_k\lambda_i}\right)^2\right],$$

则满足$(1+\theta_k-\eta_k\lambda_i)^2\le 4\theta_k$，此时只需要令$\theta_k=\max\left\{\left(1-\sqrt{\eta_k L}\right)^2,\left(1-\sqrt{\eta_k\mu}\right)^2\right\}$。

使用这样的θ_k，我们有（从上述和两个特征值具有相同幅度的事实）

$$\rho(H_k) \leq \sqrt{\theta_k}$$

最终，设置$\eta_k = \frac{4}{(\sqrt{L}+\sqrt{\mu})^2}$ 确保$1 - \sqrt{\eta_k L} = -(1 - \sqrt{\eta_k \mu})$，从而得到

$$\theta_k = \max\left\{\left(1 - \frac{2\sqrt{L}}{\sqrt{L}+\sqrt{\mu}}\right)^2, \left(1 - \frac{2\sqrt{\mu}}{\sqrt{L}+\sqrt{\mu}}\right)^2\right\} = \left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^2$$

这进一步证明了

$$\rho(H_k) \leq \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}$$

一般情况下的收敛结果

- 二阶可微时，若$f(x)$是L-光滑，$\mu-$强凸，那么局部可以证明复杂度：$O\left(\sqrt{\kappa}\log\frac{1}{\epsilon}\right)$ - Polyak 1964.
- 一阶可微，算法复杂度和梯度法类似。参考文献：Ghadimi, Euhanna, Hamid Reza Feyzmahdavian, and Mikael Johansson. "Global convergence of the heavy-ball method for convex optimization." 2015 European control conference (ECC). IEEE, 2015.

提纲

Nesterov加速算法

Yurri Nesterov 在1983年提出的算法: 任意给定x_0，$y_0 = x_0, k \geq 1$时，有如下迭代

$$\begin{aligned} x_k &= y_{k-1} - s\nabla f(y_{k-1}) \\ y_k &= x_k + \frac{k-1}{k+2}(x_k - x_{k-1}) \end{aligned} \tag{Nesterov}$$

- $s > 0$是步长，一般选$s \leq 1/L$.
- 交替进行梯度更新和适当的外推。类似地采用了动量的思想。
- 每次迭代的成本几乎与GD相同，仅需要计算一次梯度。
- 不是下降方法（即我们可能不会有$f(x_{k+1}) \leq f(x_k)$）。
- 系数$\frac{k-1}{k+2}$的选取很神秘, 早期并没有非常直观的理解。

Nesterov加速梯度方法的收敛性

假设f是凸的且L-光滑的。如果$s = \frac{1}{L}$，那么

$$f(x_k) - f^* \leq \frac{2L\|x_0 - x^*\|_2^2}{(k+1)^2}$$

- 迭代复杂度：$O\left(\frac{1}{\sqrt{\epsilon}}\right)$
- 比梯度方法快得多。
- 我们稍后将为（更一般的）近似梯度版本的证明。

梯度下降法:一阶方法的收敛下界

一阶方法：任何选择x_{k+1}在集合中的迭代算法

$$x_0 + \text{span}\{\nabla f(x_0), \nabla f(x_1), \ldots, \nabla f(x_k)\}$$

问题类：满足L式光滑和凸假设的任何函数。
定理**(Nesterov)**：对于每个整数$k \leq \frac{n-1}{2}$和每个x_0，存在在问题类中的函数，对于任何一阶方法

$$f(x_k) - f^* \geq \frac{3L\|x_0 - x^*\|^2}{32(k+1)^2}$$

- 该下界是前k次迭代的收敛下界，并不说明$k \to \infty$无更快速度。
- 表明梯度方法的$\frac{1}{k}$速率不是最优的。
- Nesterov's 加速梯度方法有$\frac{1}{k^2}$的收敛性，达到了最优收敛速度。

该定理见Yu. Nesterov, Lectures on Convex Optimization (2018), section 2.1. (Theorem 2.1.7 in the book.) Nesterov加速梯度算法.

微分方程的角度理解Nesterov加速算法-Su, Boyd, Candes '2016

首先，Nesterov的更新规则(Nesterov)等价于

$$\frac{x_{k+1}-x_k}{\sqrt{s}}=\frac{k-1}{k+2}\frac{(x_k-x_{k-1})}{\sqrt{s}}-\sqrt{s}\nabla f(y_k) \tag{1}$$

假设有一条光滑曲线$X(\tau),\tau\geq 0$. 我们拟设$x_k\approx X(k\sqrt{s})$.
令$k=\tau/\sqrt{s}$,则$X(\tau)\approx x_{\tau/\sqrt{s}}=x_k$ 和$X(\tau+\sqrt{s})\approx x_{(\tau+\sqrt{s})/\sqrt{s}}=x_{k+1}$。然后泰勒展开给出

$$\frac{x_{k+1}-x_k}{\sqrt{s}}=\dot{X}(\tau)+\frac{1}{2}\ddot{X}(\tau)\sqrt{s}+o(\sqrt{s})$$

$$\frac{x_k-x_{k-1}}{\sqrt{s}}=\dot{X}(\tau)-\frac{1}{2}\ddot{X}(\tau)\sqrt{s}+o(\sqrt{s})$$

$$\sqrt{s}\nabla f(y_k)=\sqrt{s}\nabla f(x_k)+o(\sqrt{s}).$$

再利用关系

$$\frac{k-1}{k+2}=1-\frac{3}{k}+O(1/k^2)\approx 1-\frac{3}{k} \tag{2}$$

带入(1)中，得到

$$\dot{X}(\tau)+\frac{1}{2}\ddot{X}(\tau)\sqrt{s}+o(\sqrt{s})=(1-\frac{3\sqrt{s}}{\tau})\left(\dot{X}(\tau)-\frac{1}{2}\ddot{X}(\tau)\sqrt{s}+o(\sqrt{s})\right)-\sqrt{s}\nabla f(X(\tau))+o(\sqrt{s})$$

令$s \to 0$, 比较上一页最后等式中$\sqrt{s}$的系数，得到如下常微分方程（ODE）

$$\ddot{X} + \frac{3}{\tau}\dot{X} + \nabla f(X) = 0 \tag{3}$$

初始条件：

$$X(0) = x_0,$$

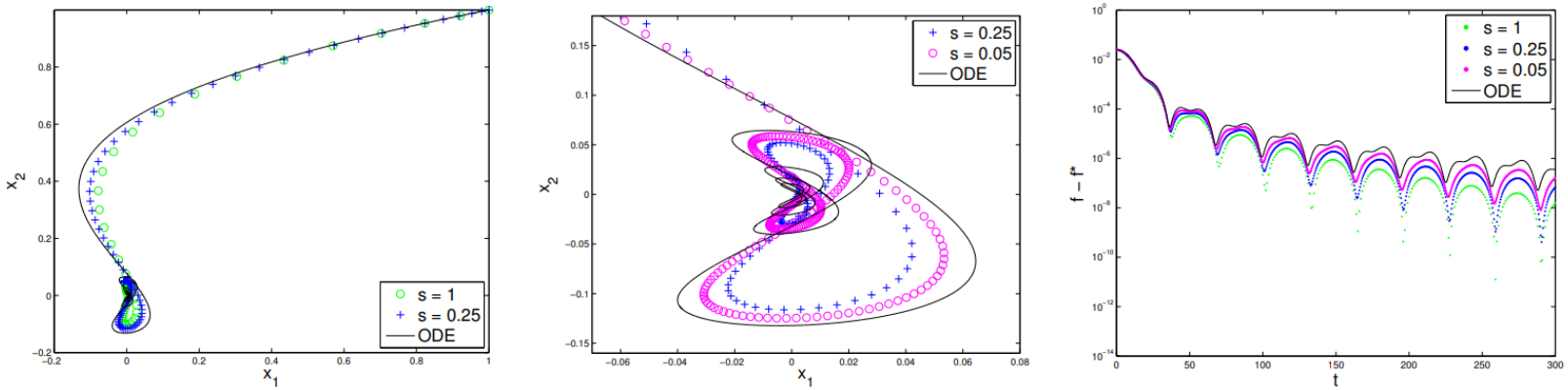
和

$$\dot{X}(0) = 0,$$

这是因为$k = 1$时，$(x_2 - x_1)/\sqrt{s} = -\sqrt{s}\nabla f(y_1) = o(1)$.

注

推导过程依赖于关系式(2),这也是*Nesterov*加速算法最奇妙的地方。



(a) Trajectories. (b) Zoomed trajectories. (c) Errors $f - f^\star$.

Figure 1: Minimizing $f = 2 \times 10^{-2} x_1^2 + 5 \times 10^{-3} x_2^2$, starting from $x_0 = (1,\ 1)$. The black and solid curves correspond to the solution to the ODE. In (c), for the x-axis we use the identification between time and iterations, $t = k\sqrt{s}$.

Figure: ODE曲线和Nesterov算法迭代轨迹, Fig. From Su, Boyd, Candes '2016

ODE的收敛率

ODE理论揭示

$$f(X(\tau)) - f^* \leq \frac{2\|x_0 - x^*\|^2}{\tau^2} \tag{4}$$

这在某种程度上解释了Nesterov的$O\left(\frac{1}{k^2}\right)$收敛。

若要将Nesterov算法中的迭代$\{x_k\}$和ODE序列$\{X(\tau)\}$严格联系起来，请见下述文献中的定理2： Su, Weijie, Stephen Boyd, and Emmanuel J. Candes. "A differential equation for modeling Nesterov's accelerated gradient method: Theory and insights." Journal of Machine Learning Research 17.153 (2016): 1-43.

证明(4)

定义李雅普诺夫函数如下：
$E(\tau) := \tau^2(f(X) - f^*) + 2\|X + \tau\dot{X}/2 - X^*\|_2^2$。我们有

$$\begin{aligned}
\dot{E} &= 2\tau(f(X) - f^*) + \tau^2\langle\nabla f(X), \dot{X}\rangle + 4\langle X + \frac{\tau}{2}\dot{X} - X^*, \frac{3}{2}\dot{X} + \frac{\tau}{2}\ddot{X}\rangle \\
&= 2\tau(f(X) - f^*) + \tau^2\langle\nabla f(X), \dot{X}\rangle + 4\langle X + \frac{\tau}{2}\dot{X} - X^*, -\frac{\tau}{2}\nabla f(X)\rangle \\
&= 2\tau(f(X) - f^*) - 2\tau\langle\nabla f(X), X - X^*\rangle \\
&\leq 0.
\end{aligned}$$

其中，第二个等式利用了(3), 最后一行用了凸函数性质。

这意味着E是关于τ的非递增的，注意到$\|X + \tau\dot{X}/2 - X^*\|_2^2$的非负性,因此

$$f(X(\tau)) - f^* \leq \frac{E(\tau)}{\tau^2} \leq \frac{E(0)}{\tau^2} = \frac{2\|x_0 - x^*\|^2}{\tau^2}.$$

魔术数字3

$$\ddot{X} + \frac{r}{\tau}\dot{X} + \nabla f(X) = 0$$

- $r = 3$是保证$O\left(\frac{1}{\tau^2}\right)$收敛的最小常数，可以用任何其他$r \geq 3$替换。
- 在某种意义上，3最小化了收敛界$O\left(\frac{1}{\tau^2}\right)$中的前常数（见Su, Boyd, Candes '14）。

提纲

典型问题形式

考虑如下复合优化问题：

$$\min_{x\in\mathbb{R}^n} \quad \psi(x) = f(x) + h(x) \tag{5}$$

- $f(x)$是连续可微的凸函数，且梯度是利普西茨连续的：

$$\|\nabla f(x) - \nabla f(y)\| \le L\|x - y\|;$$

- $h(x)$是适当的闭凸函数，且临近算子

$$\mathrm{prox}_h(x) = \operatorname*{argmin}_{u\in\mathrm{dom}h} \left\{ h(u) + \frac{1}{2}\|x - u\|^2 \right\}$$

  容易计算.

- 对于上述问题，近似点梯度法

$$x^{k+1} = \mathrm{prox}_{t_k h}(x^k - t_k \nabla f(x^k))$$

  在步长取常数$t_k = 1/L$时，收敛速度为$\mathcal{O}(1/k)$.

Nesterov加速算法简史

- Nesterov分别在1983年、1988年和2005年提出了三种改进的一阶算法，收敛速度能达到$\mathcal{O}\left(\frac{1}{k^2}\right)$. 实际上，这三种算法都可以应用到近似点梯度算法上.
- 在Nesterov加速算法刚提出的时候，由于牛顿算法有更快的收敛速度，Nesterov加速算法在当时并没有引起太多的关注. 但近年来，随着数据量的增大，牛顿型方法由于其过大的计算复杂度，不便于有效地应用到实际中，Nesterov加速算法作为一种快速的一阶算法重新被挖掘出来并迅速流行起来.
- Beck和Teboulle就在2008年给出了Nesterov在1983年提出的算法的近似点梯度法版本——FISTA. 由于其应用于图像处理中，使得Nesterov加速法进一步得到重视。参考文献：Beck, Amir, and Marc Teboulle. "A fast iterative shrinkage-thresholding algorithm for linear inverse problems." SIAM journal on imaging sciences 2.1 (2009): 183-202.

- FISTA算法由两步组成：第一步沿着前两步的计算方向计算一个新点，第二步在该新点处做一步近似点梯度迭代（如图所示）.

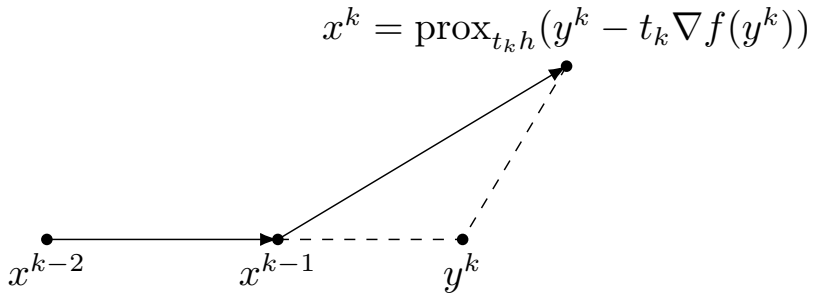


Figure: FISTA算法图示

- 完整的FISTA见算法6：

$$\begin{aligned} y^k &= x^{k-1} + \frac{k-2}{k+1}(x^{k-1} - x^{k-2}) \\ x^k &= \text{prox}_{t_k h}(y^k - t_k \nabla f(y^k)) \end{aligned} \tag{6}$$

- 可以看出，相对于(Nesterov), 仅需将梯度迭代变为近似点梯度迭代公式。

FISTA的等价形式

- 系数$\frac{k-2}{k+1}$被称为动量系数，有更一般的取值，可以替换为$\gamma_k(\frac{1}{\gamma_{k-1}}-1)$，这里$\{\gamma_k\}$满足下一页中的不等式(9)
- $\gamma_k=\frac{2}{k+1}$即有$\gamma_k(\frac{1}{\gamma_{k-1}}-1)=\frac{k-2}{k+1}$
- 算法7给出了FISTA的一个等价变形：

$$\begin{aligned} y^k &= (1-\gamma_k)x^{k-1}+\gamma_k v^{k-1} \\ x^k &= \mathrm{prox}_{t_k h}(y^k - t_k\nabla f(y^k)) \\ v^k &= x^{k-1}+\frac{1}{\gamma_k}(x^k-x^{k-1}) \end{aligned} \tag{7}$$

FISTA的收敛条件

下面给出算法7以$\mathcal{O}\left(\frac{1}{k^2}\right)$的速度收敛的条件：

$$f(x^k) \leq f(y^k) + \langle \nabla f(y^k), x^k - y^k \rangle + \frac{1}{2t_k}\|x^k - y^k\|_2^2, \tag{8}$$

$$\gamma_1 = 1, \quad \frac{(1-\gamma_k)t_k}{\gamma_k^2} \leq \frac{t_{k-1}}{\gamma_{k-1}^2}, \quad k > 1, \tag{9}$$

$$\frac{\gamma_k^2}{t_k} = \mathcal{O}\left(\frac{1}{k^2}\right). \tag{10}$$

- 可以看到当取$t_k = \frac{1}{L}$，$\gamma_k = \frac{2}{k+1}$时，以上条件满足.
- γ_k的选取并不唯一，例如我们可以采取

$$\gamma_1 = 1, \quad \frac{1}{\gamma_k} = \frac{1}{2}\left(1 + \sqrt{1 + \frac{4}{\gamma_{k-1}}}\right).$$

线搜索方法一

- 算法6和算法7都要求步长满足$t_k \leq \frac{1}{L}$，此时条件(8)满足.
- 对绝大多数问题我们不知道函数∇f的利普希茨常数. 为了在这种情况下条件(8)依然能满足，需要使用线搜索来确定合适的t_k.
- 方法一在算法7的第2行中加入线搜索，并取$\gamma_k = \frac{2}{k+1}$，以回溯的方式找到满足条件(8)的t_k. 该算法的具体过程见算法11.

$$\text{重复} \quad \begin{cases} t_k \leftarrow \rho t_k \\ x^k \leftarrow \operatorname{prox}_{t_k h}(y^k - t_k \nabla f(y^k)) \end{cases} \quad \text{直到(8)满足} \tag{11}$$

- 当t_k足够小时，条件(8)是一定会得到满足的，因此不会出现线搜索无法终止的情况.
- 容易验证其他两个条件(9)(10)在迭代过程中也得到满足.

线搜索方法二

- 第二种线搜索方法不仅改变步长t_k而且改变γ_k
- 该算法的具体过程见算法12.

$$\text{重复}\quad \begin{cases} \text{取}\gamma_k\text{为}t_{k-1}\gamma^2 = t_k\gamma_{k-1}^2(1-\gamma)\text{的正根} \\ y^k \leftarrow (1-\gamma_k)x^{k-1} + \gamma_k v^{k-1} \\ x^k \leftarrow \operatorname{prox}_{t_k h}(y^k - t_k\nabla f(y^k)) \\ t_k \leftarrow \rho t_k \end{cases} \quad \text{直到(8)成立} \tag{12}$$

- 算法12的执行过程比算法11的复杂. 由于它同时改变了t_k和γ_k，迭代点x^k和参照点y^k在线搜索的过程中都发生了变化，点y^k处的梯度也需要重新计算.
- 但此算法给我们带来的好处就是步长t_k不再单调下降，在迭代后期也可以取较大值，这会进一步加快收敛.

- 由算法12，γ_k满足条件(9)且有$0 < \gamma_k \leq 1$，且t_k有下界$t_{\min}$.
- 由$\sqrt{1-x}$在点$x = 0$处的凹性，

$$\frac{\sqrt{t_{k-1}}}{\gamma_{k-1}} = \frac{\sqrt{(1-\gamma_k)t_k}}{\gamma_k} \leq \frac{\sqrt{t_k}}{\gamma_k} - \frac{\sqrt{t_k}}{2},$$

- 反复利用上式可得

$$\frac{\sqrt{t_k}}{\gamma_k} \geq \sqrt{t_1} + \frac{1}{2}\sum_{i=2}^{k}\sqrt{t_i},$$

- 因此

$$\frac{\gamma_k^2}{t_k} \leq \frac{1}{(\sqrt{t_1} + \frac{1}{2}\sum_{i=2}^{k}\sqrt{t_i})^2} \leq \frac{4}{t_{min}(k+1)^2} = \mathcal{O}\left(\frac{1}{k^2}\right). \tag{13}$$

- 以上的分析说明条件(9)和(10)在算法12的执行中也得到满足.

强凸情况下的动量系数选取

若f是$L-$光滑，$\mu-$强凸的，令条件数为$\kappa = L/\mu$.

$$x_{k+1} = \text{prox}_{t_k h}\left(y_k - t_k \nabla f(y_k)\right)$$

$$y_{k+1} = x_{k+1} + \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}(x_{k+1} - x_k)$$

定理**11.2**（加速近端梯度方法对于强凸情况的收敛）
假设f是μ-强凸且L-光滑的。如果$t_k \equiv \frac{1}{L}$，那么

$$F(x_k) - F^* \leq \left(1 - \frac{1}{\sqrt{\kappa}}\right)^k \left(F(x_0) - F^* + \frac{\mu\|x_0 - x^*\|_2^2}{2}\right)$$

- 表明强凸情况下，收敛复杂度为$\mathcal{O}(\sqrt{\kappa}\log(1/\epsilon))$. 这也是最优的收敛速度。
- 缺点：需要同时知晓L, μ，L可以通过线搜索估计，但是μ更难以估计。
- 错误估计κ，可能使得算法震荡，可以使用重启动方法修正。参考文献："Adaptive restart for accelerated gradient schemes," B. O'donoghue, and E. Candes, Foundations of computational mathematics, 2012

FISTA算法小结

- 总的来说，固定步长的FISTA算法对于步长的选取是较为保守的，为了保证收敛，有时不得不选取一个很小的步长，这使得固定步长的FISTA算法收敛较慢.
- 如果采用线搜索，则在算法执行过程中会有很大机会选择符合条件的较大步长，因此线搜索可能加快算法的收敛，但代价就是每一步迭代的复杂度变高.
- 在实际的FISTA算法中，需要权衡固定步长和线搜索算法的利弊，从而选择针对特定问题的高效算法.

下降FISTA算法

- 原始的FISTA算法不是一个下降算法，这里给出一个FISTA的下降算法变形.
- 只需要对算法7的第2步进行修改. 在计算邻近算子之后，我们并不立即选取此点作为新的迭代点，而是检查函数值在当前点处是否下降，只有当函数值下降时才更新迭代点.
- 假设经过近似点映射之后的点为u，则对当前点x^k做如下更新：

$$x^k = \begin{cases} u, & \psi(u) \leq \psi(x^{k-1}), \\ x^{k-1}, & \psi(u) > \psi(x^{k-1}). \end{cases} \tag{14}$$

- 由于步长或γ_k会随着k变化，(14)式中的$\psi(u) > \psi(x^{k-1})$不会一直成立，即算法不会停留在某个x^{k-1}而不进行更新.
- 步长和γ_k的选取只需使用固定步长$t_k \leq \frac{1}{L}$, $\gamma_k = \frac{2}{k+1}$或者使用前述的任意一种线搜索方法均可.

提纲

第二类Nesterov加速算法

- 对于复合优化问题(5)，我们给出第二类Nesterov加速算法：

$$\begin{aligned} z^k &= (1-\gamma_k)x^{k-1} + \gamma_k y^{k-1} \\ y^k &= \mathrm{prox}_{(t_k/\gamma_k)h}\left(y^{k-1} - \frac{t_k}{\gamma_k}\nabla f(z^k)\right) \\ x^k &= (1-\gamma_k)x^{k-1} + \gamma_k y^k \end{aligned} \tag{15}$$

- 和经典FISTA 算法的一个重要区别在于，第二类Nesterov 加速算法中的三个序列$\{x^k\}$，$\{y^k\}$和$\{z^k\}$都可以保证在定义域内．而FISTA 算法中的序列$\{y^k\}$不一定在定义域内．

第二类Nesterov加速算法

- 第二类Nesterov 加速算法的一步迭代可参考下图.

$$y^k = \mathrm{prox}_{(t_k/\gamma_k)h}(y^{k-1} - (t_k/\gamma_k)\nabla f(z^k))$$

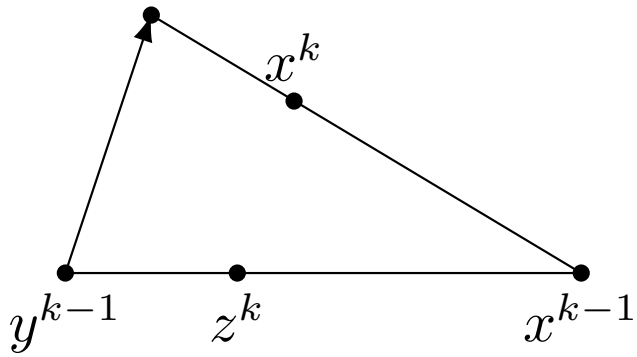


Figure: 第二类Nesterov加速算法的一步迭代

参考论文：Nesterov, Y. (2013). Gradient methods for minimizing composite objective function. Mathematical Programming, 140(1), 125-161.

第三类Nesterov加速算法

- 针对问题(5)的第三类Nesterov加速算法框架为：

$$\begin{aligned} z^k &= (1-\gamma_k)x^{k-1} + \gamma_k y^{k-1} \\ y^k &= \mathrm{prox}_{(t_k \sum_{i=1}^k 1/\gamma_i)h}\left(-t_k \sum_{i=1}^k \frac{1}{\gamma_i}\nabla f(z^i)\right) \\ x^k &= (1-\gamma_k)x^{k-1} + \gamma_k y^k \end{aligned} \tag{16}$$

- 该算法和第二类Nesterov加速算法（算法15）的区别仅仅在于y^k的更新：第三类Nesterov加速算法计算y^k时需要利用全部已有的$\{\nabla f(z^i)\}, i = 1, 2, \cdots, k$.
- 同样地，该算法取$\gamma_k = \frac{2}{k+1}$，$t_k = \frac{1}{L}$时，也有$\mathcal{O}\left(\frac{1}{k^2}\right)$的收敛速度.

其他方式理解加速梯度算法

- 梯度法结合镜像梯度法的线性耦合：Allen-Zhu, Zeyuan, and Lorenzo Orecchia. "Linear Coupling: An Ultimate Unification of Gradient and Mirror Descent." 8th Innovations in Theoretical Computer Science Conference (ITCS 2017). Schloss-Dagstuhl-Leibniz Zentrum für Informatik, 2017.
- 椭圆法的拓展: Bubeck S, Lee YT, Singh M. A geometric alternative to Nesterov's accelerated gradient descent. arXiv preprint arXiv:1506.08187. 2015 Jun 26.
  - 该方法的proximal 版本：Chen, Shixiang, Shiqian Ma, and Wei Liu. "Geometric descent method for convex composite minimization." Advances in Neural Information Processing Systems 30 (2017).
- 高精度的ODE方程法：Shi, Bin, et al. "Understanding the acceleration phenomenon via high-resolution differential equations." Mathematical Programming (2022): 1-70.

针对非凸问题的Nesterov加速算法

- 仍然考虑问题(5)的形式，这里并不要求f是凸的，但是要求其是可微的且梯度是利普希茨连续的，h与之前的要求相同.
- 算法17给出非凸复合优化问题的加速梯度法框架.

$$\begin{aligned} z^k &= \gamma_k y^{k-1} + (1-\gamma_k)x^{k-1} \\ y^k &= \mathrm{prox}_{\lambda_k h}\left(y^{k-1} - \lambda_k \nabla f(z^k)\right) \\ x^k &= \mathrm{prox}_{t_k h}\left(z^k - t_k \nabla f(z^k)\right) \end{aligned} \tag{17}$$

针对非凸问题的Nesterov加速算法

- 从形式上看，算法17和之前介绍的任何一种算法都不相同，但可以证明当λ_k和t_k取特定值时，它等价于第二类Nesterov加速算法.
- 在非凸函数情形下，一阶算法一般只能保证收敛到一个稳定点，并不能保证收敛到最优解，因此无法用函数值与最优值的差来衡量优化算法解的精度. 对于非凸复合函数(5)，我们利用梯度映射

$$G_t(x) = \frac{1}{t}(x - \mathrm{prox}_{th}(x - t\nabla f(x)))$$

来判断算法是否收敛. 注意到$G_t(x) = 0$是优化问题(5)的一阶必要条件，因此利用$\|G_{t_k}(x^k)\|$来刻画算法17的收敛速度.
- 可以证明，当f为凸函数时，算法17的收敛速度与FISTA算法相同，两者都为$\mathcal{O}\left(\frac{1}{k^2}\right)$；当$f$为非凸函数时，算法17也收敛，且收敛速度为$\mathcal{O}\left(\frac{1}{k}\right)$.

提纲

LASSO问题求解

- LASSO问题为

$$\min_{x} \quad \frac{1}{2}\|Ax-b\|_2^2+\mu\|x\|_1 \tag{18}$$

- 求解LASSO问题(18)的FISTA算法可以由下面的迭代格式给出：

$$\begin{aligned} y^k &= x^{k-1}+\frac{k-2}{k+1}(x^{k-1}-x^{k-2}),\\ w^k &= y^k - t_k A^T(Ay^k-b),\\ x^k &= \mathrm{sign}(w^k)\max\{|w^k|-t_k\mu,0\}. \end{aligned}$$

- 与近似点梯度算法相同，由于最后一步将w^k中绝对值小于$t_k\mu$的分量置零，该算法能够保证迭代过程中解具有稀疏结构.

- 我们也给出第二类Nesterov加速算法：

$$
\begin{aligned}
z^k &= (1-\gamma_k)x^{k-1} + \gamma_k y^{k-1}, \\
w^k &= y^{k-1} - \frac{t_k}{\gamma_k} A^T(Az^k - b), \\
y^k &= \operatorname{sign}(w^k)\max\left\{|w^k| - \frac{t_k}{\gamma_k}\mu, 0\right\}, \\
x^k &= (1-\gamma_k)x^{k-1} + \gamma_k y^k,
\end{aligned}
$$

- 和第三类Nesterov加速算法：

$$z^k = (1-\gamma_k)x^{k-1} + \gamma_k y^{k-1},$$

$$w^k = -t_k \sum_{i=1}^{k} \frac{1}{\gamma_i} A^T(Az^i - b),$$

$$y^k = \mathrm{sign}(w^k)\max\left\{|w^k| - t_k \sum_{i=1}^{k} \frac{1}{\gamma_i}\mu, 0\right\},$$

$$x^k = (1-\gamma_k)x^{k-1} + \gamma_k y^k.$$

- 取$\mu = 10^{-3}$，分别利用连续化近似点梯度法、连续化FISTA加速算法、连续化第二类Nesterov算法来求解问题
- 分别取固定步长$t = \frac{1}{L}$，这里$L = \lambda_{\max}(A^TA)$，和结合线搜索的BB步长.
- 结果如下图：

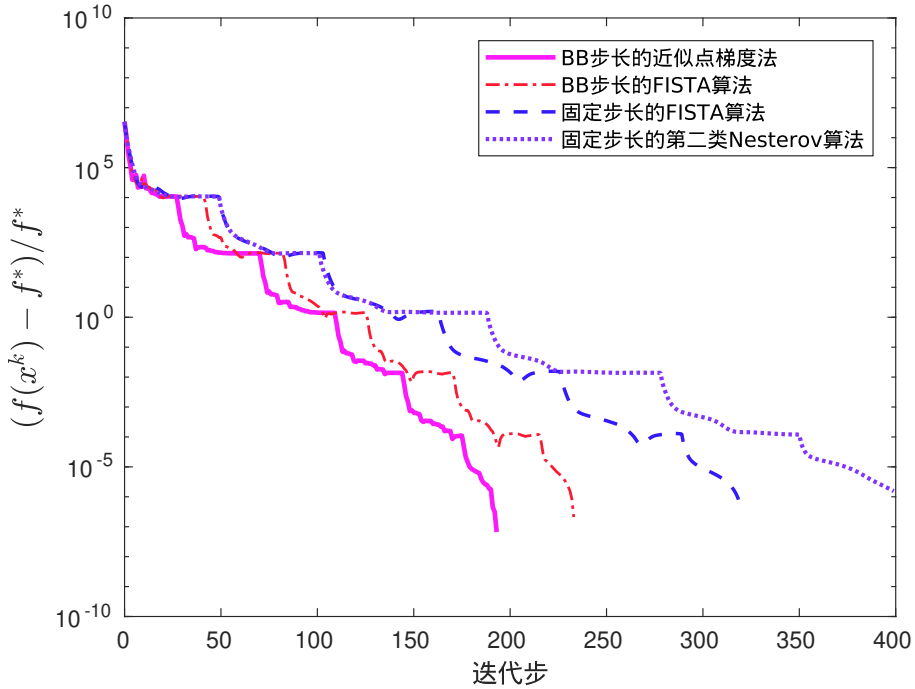


Figure: 使用近似点梯度法以及不同的加速算法求解LASSO 问题

LASSO问题求解（续）

可以看到：

- 就固定步长而言，FISTA算法相较于第二类Nesterov加速算法收敛得略快一些；
- 注意到FISTA算法是非单调算法.
- BB步长和线搜索技巧可以加速算法的收敛速度.
- 带线搜索的近似点梯度法可以比带线搜索的FISTA算法更快收敛.

提纲

收敛性假设

- f 在其定义域$\mathbf{dom}\, f = \mathbb{R}^n$内为凸的，$\nabla f$在常数$L$意义下利普西茨连续，即

$$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|, \qquad \forall x, y;$$

- h是适当的闭凸函数；
- $\psi(x)$的最小值ψ^*是有限的，并且在点x^*处可以取到.

固定步长FISTA算法收敛速度

定理 (固定步长FISTA算法收敛速度)

在上述收敛性假设的条件下，当用算法7求解凸复合优化问题(5)时，若取固定步长$t_k = \frac{1}{L}$，则

$$\psi(x^k) - \psi(x^*) \leq \frac{2L}{(k+1)^2}\|x^0 - x^*\|^2. \tag{19}$$

FISTA收敛定理的证明

- 根据$x^k = \text{prox}_{t_k h}(y^k - t_k \nabla f(y^k))$，可知

$$-x^k + y^k - t_k \nabla f(y^k) \in t_k \partial h(x^k).$$

故对于任意的x，有

$$t_k h(x) \geq t_k h(x^k) + \langle -x^k + y^k - t_k \nabla f(y^k), x - x^k \rangle. \tag{20}$$

- 由f 的凸性、梯度利普希茨连续和$t_k = \frac{1}{L}$可以得到

$$f(x^k) \leq f(y^k) + \langle \nabla f(y^k), x^k - y^k \rangle + \frac{1}{2t_k} \|x^k - y^k\|^2. \tag{21}$$

- 结合以上两个不等式，对于任意的x有

$$\begin{aligned}\psi(x^k) &= f(x^k) + h(x^k) \\ &\le h(x) + f(y^k) + \langle \nabla f(y^k), x - y^k \rangle + \frac{1}{t_k}\langle x^k - y^k, x - x^k \rangle \\ &\quad + \frac{1}{2t_k}\left\| x^k - y^k \right\|^2 \\ &\le h(x) + f(x) + \frac{1}{t_k}\langle x^k - y^k, x - x^k \rangle + \frac{1}{2t_k}\left\| x^k - y^k \right\|^2 \\ &= \psi(x) + \frac{1}{t_k}\langle x^k - y^k, x - x^k \rangle + \frac{1}{2t_k}\left\| x^k - y^k \right\|^2 .\end{aligned} \tag{22}$$

- 在(22)式中分别取$x = x^{k-1}$和$x = x^*$，并记$\psi(x^*) = \psi^*$，再分别乘$1 - \gamma_k$和γ_k并相加得到

$$\begin{aligned}&\psi(x^k) - \psi^* - (1 - \gamma_k)(\psi(x^{k-1}) - \psi^*) \\ \le &\frac{1}{t_k}\langle x^k - y^k, (1 - \gamma_k)x^{k-1} + \gamma_k x^* - x^k \rangle + \frac{1}{2t_k}\left\| x^k - y^k \right\|^2 .\end{aligned} \tag{23}$$

- 结合迭代式

$$v^k = x^{k-1} + \frac{1}{\gamma_k}(x^k - x^{k-1}),$$
$$y^k = (1-\gamma_k)x^{k-1} + \gamma_k v^{k-1},$$

不等式(23)可以化为

$$\begin{aligned}
&\psi(x^k) - \psi^* - (1-\gamma_k)(\psi(x^{k-1}) - \psi^*) \\
\leq &\frac{1}{2t_k}(\|y^k - (1-\gamma_k)x^{k-1} - \gamma_k x^*\|^2 - \|x^k - (1-\gamma_k)x^{k-1} - \gamma_k x^*\|^2) \\
= &\frac{\gamma_k^2}{2t_k}(\|v^{k-1} - x^*\|^2 - \|v^k - x^*\|^2).
\end{aligned} \tag{24}$$

- t_k, γ_k的取法满足不等式

$$\frac{1-\gamma_k}{\gamma_k^2} t_k \leq \frac{1}{\gamma_{k-1}^2} t_{k-1}, \tag{25}$$

可以得到一个有关相邻两步迭代的不等式

$$\frac{t_k}{\gamma_k^2}(\psi(x^k)-\psi^*)+\frac{1}{2}\|v^k-x^*\|^2 \leq \frac{t_{k-1}}{\gamma_{k-1}^2}(\psi(x^{k-1})-\psi^*)+\frac{1}{2}\|v^{k-1}-x^*\|^2. \tag{26}$$

- 反复利用(26)式，我们有

$$\frac{t_k}{\gamma_k^2}(\psi(x^k)-\psi^*)+\frac{1}{2}\|v^k-x^*\|^2 \leq \frac{t_1}{\gamma_1^2}(\psi(x^1)-\psi^*)+\frac{1}{2}\|v^1-x^*\|^2. \tag{27}$$

- 对$k=1$，注意到$\gamma_1=1, v^0=x^0$，再次利用(24)式可得

$$\begin{aligned}&\frac{t_1}{\gamma_1^2}(\psi(x^1)-\psi^*)+\frac{1}{2}\|v^1-x^*\|^2\\ \leq&\frac{(1-\gamma_1)t_1}{\gamma_1^2}(\psi(x^0)-\psi^*)+\frac{1}{2}\|v^0-x^*\|^2=\frac{1}{2}\|x^0-x^*\|^2.\end{aligned}\tag{28}$$

- 结合(27)式和(28)式可以得到(19)式.

证明思路

- 证明中关键的一步在于建立(26)式，而建立这个递归关系并不需要$t = 1/L, \gamma_k = 2/(k+1)$这一具体条件，我们只需要保证条件(8)和条件(9)成立即可.
- 条件(8)主要依赖于$f(x)$的梯度利普希茨连续性；而(9)的成立依赖于γ_k和t_k的选取.
- 条件(10)的成立保证了算法7的收敛速度达到$\mathcal{O}\left(\frac{1}{k^2}\right)$.
- 如果抽取条件(8)-(10) 作为算法收敛的一般条件，则可以证明一大类FISTA算法的变形都具有$\mathcal{O}\left(\frac{1}{k^2}\right)$的收敛速度.

一般FISTA算法的收敛速度

推论 (一般FISTA算法的收敛速度)

在收敛性假设条件下，当用算法7求解凸复合优化问题(5)时，若迭代点x^k, y^k，步长t_k以及组合系数γ_k满足条件(8)-(10)，则

$$\psi(x^k) - \psi(x^*) \leq \frac{C}{k^2}, \tag{29}$$

其中C仅与函数f，初始点x^0的选取有关. 特别地，采用线搜索算法11和算法12的FISTA算法具有$\mathcal{O}\left(\frac{1}{k^2}\right)$的收敛速度.

虽然已经抽象出了t_k, γ_k满足的条件，但我们无法再找到其他的t_k, γ_k来进一步改善FISTA算法的收敛速度，即$\mathcal{O}\left(\frac{1}{k^2}\right)$是FISTA算法所能达到的最高的收敛速度.

牛顿法

陈士祥

中国科学技术大学

致谢：本教案部分参考北京大学文再文教授《最优化与建模》

梯度法的困难

考虑无约束优化问题

$$\min_{x\in\mathbb{R}^n} \quad f(x), \tag{1}$$

使用梯度下降法, 给出的迭代格式是

$$x^{k+1} = x^k - \alpha_k \nabla f\left(x^k\right).$$

由于梯度下降的基本策略是沿一阶最速下降方向迭代。当$\nabla^2 f(x)$ 的条件数较大时, 它的收敛速度比较缓慢(只用到一阶信息).

如果$f(x)$足够光滑, 我们可以利用$f(x)$的二阶信息改进下降方向, 以加速算法的迭代.

提纲

经典牛顿法

- 对于可微二次函数$f(x)$, 考虑目标函数f在点x_k的二阶泰勒近似

$$f\left(x^k+d^k\right)=f\left(x^k\right)+\nabla f\left(x^k\right)^{\mathrm{T}}d^k+\frac{1}{2}\left(d^k\right)^{\mathrm{T}}\nabla^2 f\left(x^k\right)d^k+o\left(\left\|d^k\right\|^2\right).$$

忽略高阶项$o\left(\left\|d^k\right\|^2\right)$, 并将等式右边视作$d^k$的函数并极小化, 得

$$\nabla^2 f\left(x^k\right)d^k=-\nabla f\left(x^k\right). \tag{2}$$

方程(2)被称为牛顿方程, d^k被称为牛顿方向.

- 若$\nabla^2 f\left(x^k\right)$非奇异, 可构造迭代格式

$$x^{k+1}=x^k-\alpha_k\nabla^2 f\left(x^k\right)^{-1}\nabla f\left(x^k\right). \tag{3}$$

当步长$\alpha_k=1$时迭代格式(3)被称为经典牛顿法.

牛顿法求解等式问题

牛顿法最初是为了求解一般等式问题。设$F:\mathbb{R}^n \to \mathbb{R}^n$, 考虑如下问题：

$$F(x) = 0.$$

迭代为

$$x_{k+1} = x_k - JF(x_k)^{-1}F(x_k). \tag{4}$$

对于凸问题(1)来说，求解其最小值等价于求解下面的等式：

$$\nabla f(x) = 0.$$

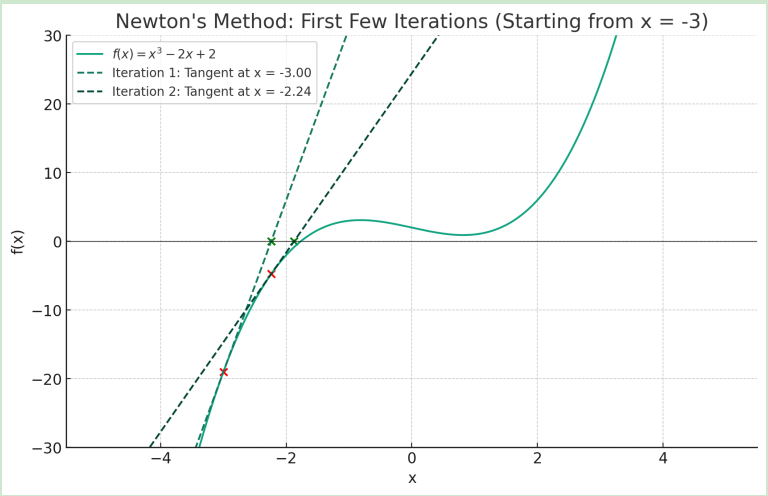
记$F = \nabla f(x)$, 则(4)与(3)相同。

牛顿法非全局收敛

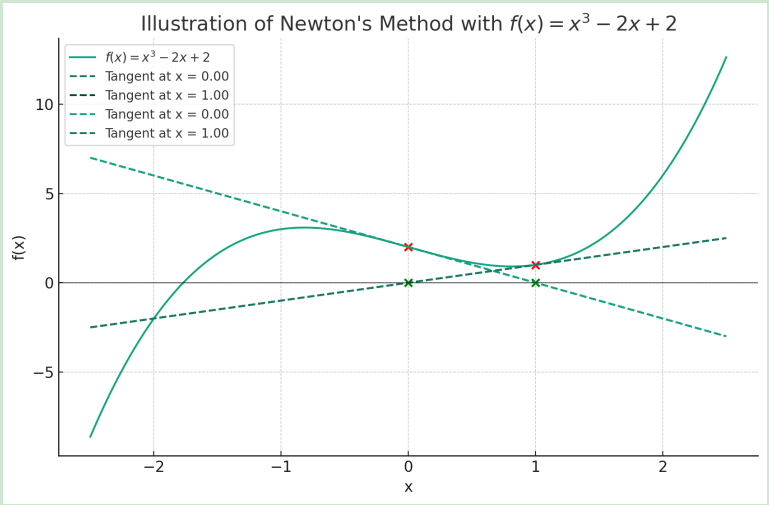
对于一维问题，即$F:\mathbb{R}\to\mathbb{R}$，下面的例子展示牛顿法的迭代过程。

Example

用牛顿法求解$F(x)=x^3-2x+2=0$的根。在迭代点x_k处，作出函数图像的切线$l(y)=F(x_k)+F'(x_k)(y-x_k)$，与$x$轴的交点得到下一个迭代点$x_{k+1}$，即$x_{k+1}=x_k-\frac{F(x_k)}{F'(x_k)}$. 从初始点$x_0=-3$和$x_0=1$出发,牛顿法迭代分别如图1(a)和1. 从$x_0=1$出发的点，由于离$F(x)=0$的根太远，牛顿法不收敛。



(a) 牛顿法1

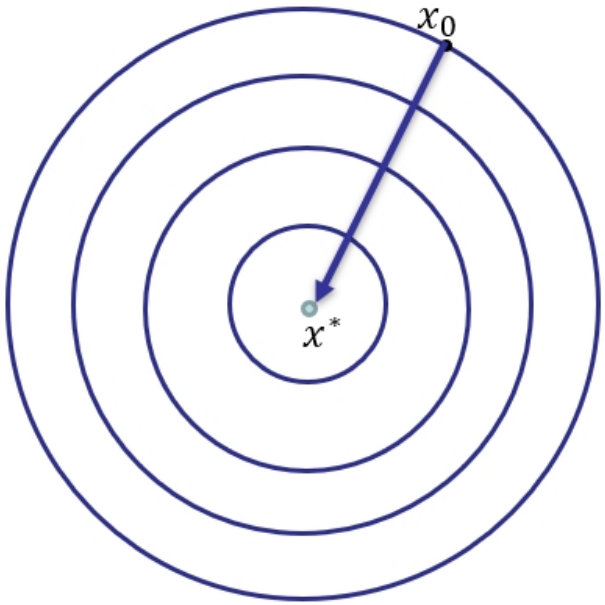


(b) 牛顿法2

Figure: 图(b),从$x_0=1$出发，牛顿法不收敛,迭代点困于0，1两点。

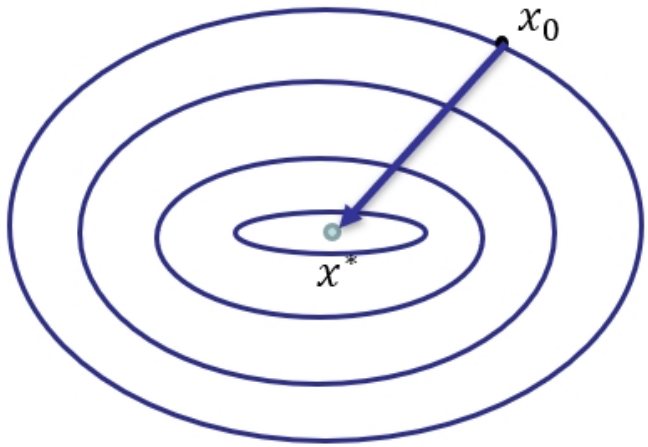
牛顿法为什么好?

对于正定二次函数而言，牛顿法一步即可达到最优解。



$$f(x) = \frac{1}{2}||x||^2$$

(a) 牛顿法1



$$f(x) = \frac{1}{2}||Qx||^2$$

(b) 牛顿法2

Figure: 牛顿法对于正定二次问题，可以一步得到最优解。

对于非二次函数，牛顿法并不能保证经有限次迭代求得最优解。但由于目标函数在极小点附近可用二次函数较好地近似，故当初始点靠近极小点时，牛顿法的收敛速度一般会很快。

仿射不变性(affine-invariant):

令$A \in \mathbb{R}^{n\times n}$为一个可逆矩阵。$f(x)$为$\mathbb{R}^n$上的一个函数。考虑如下函数

$$\phi(y) = f(Ay).$$

即对于原来的函数f，我们选择了$\mathbb{R}^n$新的一组基底A，得到新坐标下的函数$\phi(y)$. 牛顿法的关键性质可由下面的结论说明。

结论： 令$\{x_k\}$是牛顿法对于$f(x)$的序列，即

$$x_{k+1} = x_k - \nabla^2 f(x_k)^{-1}\nabla f(x_k);$$

令$\{y_k\}$是牛顿法对于$\phi(y)$的序列，即

$$y_{k+1} = y_k - \nabla^2 \phi(y_k)^{-1}\nabla \phi(y_k);$$

若$y_0 = A^{-1}x_0$, 则对于任意$k \geq 1$, $y_k = A^{-1}x_k$ 。

提纲

收敛速度分析

牛顿法收敛速度非常快, 但实际使用中会存在若干限制因素.

- 初始点x^0 需要距离最优解充分近(因为局部收敛性)
  应用时常以梯度类算法先求得较低精度的解, 后用牛顿法加速.
- $\nabla^2 f(x^*)$ 需正定, 半正定条件下可能退化到Q-线性收敛.
- $\nabla^2 f$ 的条件数较高时, 将对初值的选择作出较严苛的要求.

经典牛顿法的收敛性

定理

经典牛顿法的收敛性 假设 f 二阶连续可微,且存在 x^* 的一个邻域 $N_\delta(x^*)$ 及常数 $L>0$ 使得

$$\left\|\nabla^2 f(x)-\nabla^2 f(y)\right\| \leqslant L\|x-y\|, \quad \forall x, y \in N_\delta(x^*)$$

如果 $f(x)$ 满足 $\nabla f(x^*)=0, \nabla^2 f(x^*) \succ 0$, 则对于迭代格式(3)有:

- 如果初始点离 x^* 足够近,则迭代点列 $\{x^k\}$ 收敛到 x^*;
- $\{x^k\}$ Q-二次收敛到 x^*;
- $\{\|\nabla f(x^k)\|\}$ Q-二次收敛到 0.

定理证明

根据经典牛顿法定义以及$\nabla f\left(x^*\right)=0$,得

$$\begin{aligned} x^{k+1}-x^* &= x^k-\nabla^2 f\left(x^k\right)^{-1}\nabla f\left(x^k\right)-x^* \\ &= \nabla^2 f\left(x^k\right)^{-1}\left[\nabla^2 f\left(x^k\right)\left(x^k-x^*\right)-\left(\nabla f\left(x^k\right)-\nabla f\left(x^*\right)\right)\right], \end{aligned} \tag{5}$$

注意到

$$\nabla f\left(x^k\right)-\nabla f\left(x^*\right)=\int_0^1 \nabla^2 f\left(x^k+t\left(x^*-x^k\right)\right)\left(x^k-x^*\right)\mathrm{d}t,$$

由此

$$\begin{aligned} &\left\|\nabla^2 f\left(x^k\right)\left(x^k-x^*\right)-\left(\nabla f\left(x^k\right)-\nabla f\left(x^*\right)\right)\right\| \\ =&\left\|\int_0^1\left[\nabla^2 f\left(x^k+t\left(x^*-x^k\right)\right)-\nabla^2 f\left(x^k\right)\right]\left(x^k-x^*\right)\mathrm{d}t\right\| \\ \leqslant&\int_0^1\left\|\nabla^2 f\left(x^k+t\left(x^*-x^k\right)\right)-\nabla^2 f\left(x^k\right)\right\|\left\|x^k-x^*\right\|\mathrm{d}t \\ \leqslant&\left\|x^k-x^*\right\|^2\int_0^1 Lt\,\mathrm{d}t(\text{Lip. 连续性})=\frac{L}{2}\left\|x^k-x^*\right\|^2. \end{aligned} \tag{6}$$

注意到, $\exists r>0$, 当 $\|x-x^*\| \leqslant r$ 时有 $\left\|\nabla^2 f(x)^{-1}\right\| \leqslant 2\left\|\nabla^2 f\left(x^*\right)^{-1}\right\|$ 成立 (请思考为何?), 故结合(5)及(6),得到

$$
\begin{aligned}
&\left\|x^{k+1}-x^*\right\| \\
\leqslant &\left\|\nabla^2 f\left(x^k\right)^{-1}\right\|\left\|\nabla^2 f\left(x^k\right)\left(x^k-x^*\right)-\left(\nabla f\left(x^k\right)-\nabla f\left(x^*\right)\right)\right\| \\
\leqslant &\left\|\nabla^2 f\left(x^k\right)^{-1}\right\| \cdot \frac{L}{2}\left\|x^k-x^*\right\|^2 \\
\leqslant &L\left\|\nabla^2 f\left(x^*\right)^{-1}\right\|\left\|x^k-x^*\right\|^2 .
\end{aligned}
\tag{7}
$$

当初始点 x^0 满足 $\left\|x^0-x^*\right\| \leqslant \min\left\{\delta, r, \frac{1}{2L\left\|\nabla^2 f\left(x^*\right)^{-1}\right\|}\right\}$ 时,迭代点列一直处于邻域 $N_{\hat{\delta}}\left(x^*\right)$ 中,故 $\left\{x^k\right\}$ Q-二次收敛到 x^*.

另一方面,由牛顿方程(2)可知

$$\begin{aligned}
\left\|\nabla f\left(x^{k+1}\right)\right\| &= \left\|\nabla f\left(x^{k+1}\right)-\nabla f\left(x^{k}\right)-\nabla^2 f\left(x^{k}\right)d^k\right\| \\
&= \left\|\int_0^1 \nabla^2 f\left(x^k+td^k\right)d^k dt-\nabla^2 f\left(x^k\right)d^k\right\| \\
&\leqslant \int_0^1 \left\|\nabla^2 f\left(x^k+td^k\right)-\nabla^2 f\left(x^k\right)\right\|\left\|d^k\right\| dt \\
&\leqslant \frac{L}{2}\left\|d^k\right\|^2 \leqslant \frac{1}{2}L\left\|\nabla^2 f\left(x^k\right)^{-1}\right\|^2\left\|\nabla f\left(x^k\right)\right\|^2 \\
&\leqslant 2L\left\|\nabla^2 f\left(x^*\right)^{-1}\right\|^2\left\|\nabla f\left(x^k\right)\right\|^2.
\end{aligned}$$

这证明梯度的范数Q-二次收敛到0.

提纲

修正牛顿法

经典牛顿法的基本格式如下:

$$x^{k+1} = x^k - \nabla^2 f\left(x^k\right)^{-1} \nabla f\left(x^k\right).$$

除了计算、存储代价昂贵之外, 经典牛顿法还存在如下问题:

- 海瑟矩阵可能非正定,导致牛顿方向其实并非下降方向;
- 初始点离最优值点较远时候迭代不稳定(步长固定), 因而算法可能不收敛.

为提高算法的稳定性, 从以上两方面考虑, 应该:

- 对$\nabla^2 f(x)$进行修正,使其正定(所有特征值大于0);
- 用线搜索确定步长来增加算法的稳定性(Wolfe, Goldstein, Armijo).

综上考虑, 我们提出下面带线搜索的牛顿方法, 并称其为修正牛顿方法.

带线搜索的修正牛顿法

Algorithm 1 带线搜索的修正牛顿法

1: 给定初始点x^0.
2: **for** $k = 0, 1, 2, \cdots$ **do**
3: 确定矩阵E^k使得矩阵$B^k \overset{\text{def}}{=} \nabla^2 f\left(x^k\right) + E^k$正定且条件数较小.
4: 求解修正的牛顿方程$B^k d^k = -\nabla f\left(x^k\right)$得方向$d^k$.
5: 使用任意一种线搜索准则确定步长α_k.
6: 更新$x^{k+1} = x^k + \alpha_k d^k$.
7: **end for**

在上述算法中, B^k (即E^k) 的选取较为关键, 需要注意:

- B^k应具有较低的条件数(原因见收敛性定理);
- 对$\nabla^2 f(x)$的改动较小, 以保存二阶信息;
- B^k本身的计算代价不应太高.

修正牛顿法的全局收敛性

定理

修正牛顿法全局收敛性定理 令 f 在开集 $\mathcal{D}$ 上二阶连续可微,且初始点 x^0 满足 $\{x \in \mathcal{D} : f(x) \leq f(x_0)\}$ 为紧集.若算法**1**中 B^k 的条件数上界存在,即

$$\kappa(B_k) = \|B_k\| \left\|B_k^{-1}\right\| \leq C, \quad \exists C > 0, \forall k = 0, 1, 2, \cdots \tag{8}$$

则

$$\lim_{k \to \infty} \nabla f(x_k) = 0,$$

即算法具有全局收敛性.

上述定理的证明冗长, 读者可具体参考: Newton's method, in Studies in Numerical Analysis, vol. 24 of MAA Studies in Mathematics, The Mathematical Association of America, 1984, pp. 29–82.

修正矩阵E^k的显式选取

进一步地, 在修正步长后, 为还使 B^k 正定, 我们修正海瑟矩阵的特征值.
首先做特征分解

$$\nabla^2 f(x_k) = Q\Lambda Q^T,$$

其中,Q 为正交矩阵,Λ 为对角矩阵.

取$E^k = \tau_k I$,即单位阵的常数倍, 代入上式, 则有

$$B^k = Q(\Lambda + \tau_k I)Q^T.$$

当正数τ_k 足够大的时候, 可保证 B^k 正定, 但 d^k 会接近负梯度方向, 退化为一阶方法.
一种简单的技巧是对τ_k做截断, 即

$$\tau_k = \max\left\{0, \delta - \lambda_{\min}(\nabla^2 f\left(x^k\right))\right\}, \quad (\text{给定}\delta > 0)$$

然而, 此时$\lambda_{\min}(\nabla^2 f\left(x^k\right))$ 通常难以计算. 我们通常使用Cholesky分解试探性取τ_k, 从而避免这个问题. 具体可参考书《最优化：建模、算法与理论》第237页 。

提纲

非精确牛顿法

当变量维数很大时，牛顿法可能有如下困难:

- 海瑟矩阵$\nabla^2 f(x)$ 本身的计算、存储存在困难;
- 对$\nabla^2 f(x)$求逆或者做Cholesky分解的代价很高.

非精确牛顿法

- 使用迭代法(如共轭梯度法)求解牛顿方程，在一定的精度下提前停机, 以提高求解效率.
- 引入向量r^k 来表示残差, 将上述方程记为

$$\nabla^2 f\left(x^k\right) d^k = -\nabla f\left(x^k\right) + r^k. \tag{9}$$

因此终止条件可设置为

$$\left\| r^k \right\| \leqslant \eta_k \left\| \nabla f\left(x^k\right) \right\|. \tag{10}$$

- 不同的$\{\eta_k\}$将导致不同的精度要求, 使算法有不同的收敛速度.

收敛性分析

我们叙述非精确牛顿法的局部收敛定理.

定理

非精确牛顿法的收敛定理 设函数$f(x)$二阶连续可微, 且$\nabla^2 f(x^*)$正定, 则在非精确牛顿法中,

(1) 若$\exists t < 1$,使得η_k满足$0 < \eta_k < t, k = 1, 2, \cdots$, 且起始点$x_0$充分靠近$x^*$并迭代最终收敛到$x^*$, 则梯度$\nabla f(x^k)$以*Q*-线性收敛速度收敛;

(2) 若$\lim_{k\to\infty} \eta_k = 0$成立, 则梯度$\nabla f(x^k)$以*Q*-超线性收敛速度收敛;

(3) 若*(1)*或*(2)*成立, 且$\nabla^2 f$在x^*附近*Lip.*连续, $\eta_k = O\left(\left\|\nabla f(x^k)\right\|\right)$, 则梯度$\nabla f(x^k)$以*Q*-二次收敛速度收敛.

定理证明

我们只给出一个不严格的证明, 请读者主要体会证明思路.

注意到$\nabla^2 f(x)$在x^*处正定, 在x^*附近连续, 故存在正常数L, 使得

$$\left\| \left(\nabla^2 f(x^k) \right)^{-1} \right\| \leq L, \quad (\forall x^k \text{同} x^* \text{足够接近})$$

代入(9), 得(第二个不等式用到了$\eta_k < 1$)

$$\left\| d^k \right\| \leq L \left(\left\| \nabla f(x^k) \right\| + \left\| r^k \right\| \right) \leq 2L \left\| \nabla f(x^k) \right\|.$$

利用Taylor展式和$\nabla^2 f$的连续性, 得到

$$\begin{aligned}
\nabla f(x^{k+1}) &= \nabla f(x^k) + \nabla^2 f(x^k) d^k + \int_0^1 \left[\nabla^2 f\left(x^k + t d^k \right) - \nabla^2 f\left(x^k \right) \right] d^k dt \\
&= \nabla f(x^k) + \nabla^2 f(x^k) d^k + o\left(\left\| d^k \right\| \right) \\
&= \nabla f(x^k) - \left(\nabla f(x^k) - r^k \right) + o\left(\left\| \nabla f(x^k) \right\| \right) \\
&= r^k + o\left(\left\| \nabla f(x^k) \right\| \right).
\end{aligned}$$

定理证明

上式中两边取范数, 结合精度控制式(10), 得到

$$\left\|\nabla f(x^{k+1})\right\| \leq \eta_k \left\|\nabla f(x^k)\right\| + o\left(\left\|\nabla f(x^k)\right\|\right) \leq \left(\eta_k + o(1)\right) \left\|\nabla f(x^k)\right\|.$$

当x^k足够接近x^*时,$o(1)$项可被$(1-t)/2$控制,则

$$\left\|\nabla f(x^{k+1})\right\| \leq (\eta_k + (1-t)/2) \left\|\nabla f(x^k)\right\| \leq \frac{1+t}{2} \left\|\nabla f(x^k)\right\|.$$

由于$t < 1$,故梯度梯度$\nabla f(x^k)$以Q-线性收敛速度收敛.

推论

从以上证明过程可以看出如下的结果:

$$\frac{\left\|\nabla f(x^{k+1})\right\|}{\|\nabla f(x^k)\|} \leq \eta_k + o(1).$$

若$\lim_{k\to\infty} \eta_k = 0$成立,可有Q-超线性收敛的结论

$$\lim_{k\to\infty} \frac{\left\|\nabla f(x^{k+1})\right\|}{\|\nabla f(x^k)\|} = 0.$$

若$\nabla^2 f$ 在x^* 附近Lip.连续, 令$\eta_k = O\left(\left\|\nabla f(x^k)\right\|\right)$, 可有二次收敛的结论

$$\left\|\nabla f(x^{k+1})\right\| = O\left(\left\|\nabla f(x^k)\right\|^2\right).$$

因此, 在实际应用时:

- 取$\eta_k = \min\left(0.5, \sqrt{\|\nabla f(x^k)\|}\right)$, 可成立局部的超线性收敛;
- 取$\eta_k = \min\left(0.5, \left\|\nabla f(x^k)\right\|\right)$, 可成立局部的二次收敛.

*线搜索Newton-CG法

最后我们讨论求解牛顿方程的方法.
一种广泛应用的方法是线搜索Newton-CG法, 它利用共轭梯度迭代求解牛顿方程.

- 由于共轭梯度法要求被解方程的系数矩阵正定, 但$\nabla^2 f$ 可能非正定, 因此在迭代时需要辨别$\nabla^2 f$的正定性以保证算法有效.

具体算法在下一面给出. 其中B_k 表示$\nabla^2 f(x^k)$; $\{z_j\}$ 表示CG法从零向量开始的迭代序列, 最终收敛到牛顿方向.
我们采用的迭代终止条件是$\eta_k = \min\left(0.5, \sqrt{\|\nabla f(x^k)\|}\right)$.

CG法解牛顿方程的优势是只用到海瑟矩阵-向量积而无需求出$\nabla^2 f$, 故可以被设计成*Hessian-free*方法, 在大规模问题下解决Hessian矩阵计算或存储的困难.

*线搜索Newton-CG法:算法

Algorithm 2 线搜索Newton-CG法

给定初始点x^0.
for $k = 0, 1, 2, \cdots$ **do**
  令$\epsilon_k = \eta_k \left\|\nabla f(x^k)\right\|$, $z_0 = 0, r_0 = \nabla f(x^k), p_0 = -r_0 = -\nabla f(x^k)$.
  for $j = 0, 1, 2, \cdots$ **do**
    if $p_j^T B_k p_j \leq 0$ (判断是否是负曲率方向) **then**
      终止循环.$j = 0$ 则返回$d^k = -\nabla f$,否则返回$d^k = z_j$.
    end if
    $\alpha_j = r_j^T r_j / p_j^T B_k p_j$, $z_{j+1} = z_j + \alpha_j p_j$, $r_{j+1} = r_j + \alpha_j B_k p_j$
    if $r_j < \epsilon_k$ (判断是否达到收敛条件) **then**
      终止循环. 返回$d^k = z_{j+1}$.
    end if
    $\beta_{j+1} = r_{j+1}^T r_{j+1} / r_j^T r_j$, $d_{j+1} = -r_{j+1} + \beta_{j+1} d_j$
  end for
  线搜索确定步长α_k(符合条件则取$\alpha_k = 1$),更新$x^{k+1} = x^k + \alpha_k d^k$.
end for

提纲

逻辑回归模型

考虑二分类的逻辑回归模型

$$\min_x \quad \ell(x) \overset{\text{def}}{=} \frac{1}{m}\sum_{i=1}^{m} \ln\left(1+\exp\left(-b_i a_i^{\mathrm{T}} x\right)\right) + \lambda\|x\|_2^2.$$

为使用牛顿法, 需要计算目标函数的梯度与海瑟矩阵:

$$\begin{aligned}\nabla \ell(x) &= \frac{1}{m}\sum_{i=1}^{m} \frac{1}{1+\exp\left(-b_i a_i^{\mathrm{T}} x\right)} \cdot \exp\left(-b_i a_i^{\mathrm{T}} x\right) \cdot \left(-b_i a_i\right) + 2\lambda x \\ &= -\frac{1}{m}\sum_{i=1}^{m} \left(1-p_i(x)\right) b_i a_i + 2\lambda x,\end{aligned}$$

其中$p_i(x) = \frac{1}{1+\exp\left(-b_i a_i^{\mathrm{T}} x\right)}$.

进一步对$\nabla \ell(x)$求导, 成立

$$\begin{aligned}
\nabla^2 \ell(x) &= \frac{1}{m}\sum_{i=1}^{m} b_i \cdot \nabla p_i(x) a_i^{\mathrm{T}} + 2\lambda I \\
&= \frac{1}{m}\sum_{i=1}^{m} b_i \frac{-1}{\left(1+\exp\left(-b_i a_i^{\mathrm{T}} x\right)\right)^2} \cdot \exp\left(-b_i a_i^{\mathrm{T}} x\right) \cdot \left(-b_i a_i a_i^{\mathrm{T}}\right) + 2\lambda I \\
&= \frac{1}{m}\sum_{i=1}^{m} \left(1-p_i(x)\right) p_i(x) a_i a_i^{\mathrm{T}} + 2\lambda I \quad \left(b_i^2 = 1\right).
\end{aligned}$$

逻辑回归模型

引入矩阵$A = [a_1, a_2, \cdots, a_m]^{\mathrm{T}} \in \mathbb{R}^{m\times n}$, 向量$b = (b_1, b_2, \cdots, b_m)^{\mathrm{T}}$, 以及

$$p(x) = (p_1(x), p_2(x), \cdots, p_m(x))^{\mathrm{T}},$$

则可重写梯度和海瑟矩阵为

$$\nabla \ell(x) = -\frac{1}{m} A^{\mathrm{T}}(b - b \odot p(x)) + 2\lambda x,$$
$$\nabla^2 \ell(x) = \frac{1}{m} A^{\mathrm{T}} W(x) A + 2\lambda I,$$

其中$W(x)$ 为由$\{p_i(x)\,(1 - p_i(x))\}_{i=1}^m$ 生成的对角矩阵.

则最终牛顿法迭代格式可以写作:

$$x^{k+1} = x^k + \left(\frac{1}{m} A^{\mathrm{T}} W\left(x^k\right) A + 2\lambda I\right)^{-1} \left(\frac{1}{m} A^{\mathrm{T}}\left(b - b \odot p\left(x^k\right)\right) - 2\lambda x^k\right).$$

逻辑回归模型

我们得到了牛顿法的迭代格式, 因此可以调用牛顿法直接求解逻辑回归问题. 正如我们对牛顿方程处理思路的不同, 若变量规模不大, 则可尝试利用正定矩阵的Cholesky分解求解牛顿方程; 若变量规模较大, 则可以使用共轭梯度法对方程进行不精确的求解.

我们使用LIBSVM网站的数据集(具体数据集见下表), 对不同的数据集均调用非精确CG-牛顿法求解, 设置精度条件为

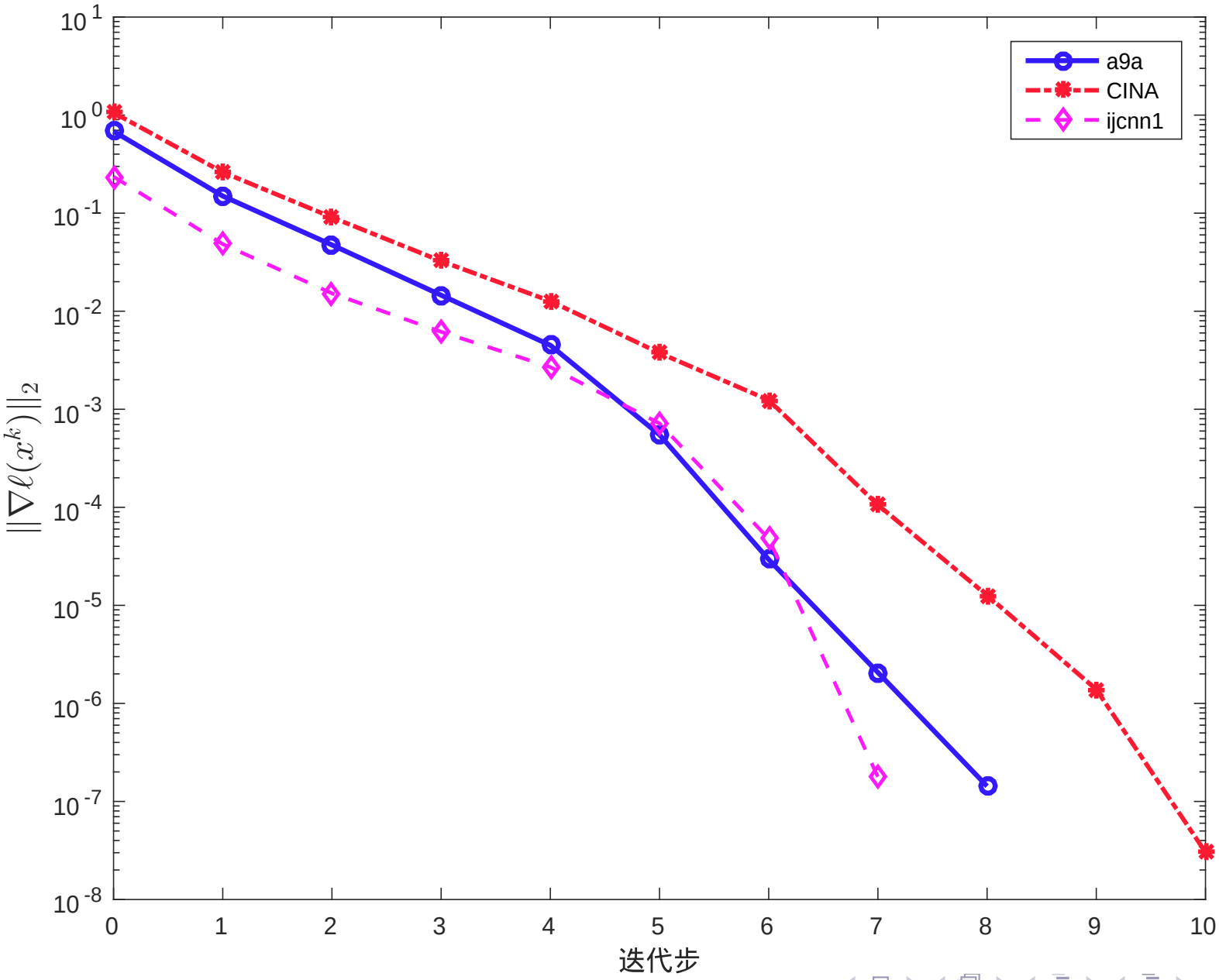
$$\left\|\nabla^2 \ell\left(x^k\right) d^k+\nabla \ell\left(x^k\right)\right\|_2 \leqslant \min\left\{\left\|\nabla \ell\left(x^k\right)\right\|_2^2, 0.1\left\|\nabla \ell\left(x^k\right)\right\|_2\right\},$$

Table: LIBSVM数据集(部分)

名称 \ 维数	m	n
a9a	16281	122
*ijcnn*1	91701	22
CINA	3206	132

在数据集中进行算法测试, 数值结果如下图所示. 从图中可以看到, 精确解附近梯度范数具有Q-超线性收敛性.

数值结果


a9a
CINA
ijcnn1
$\|\nabla \ell(x^k)\|_2$
迭代步
10^{1}
10^{0}
10^{-1}
10^{-2}
10^{-3}
10^{-4}
10^{-5}
10^{-6}
10^{-7}
10^{-8}
0
1
2
3
4
5
6
7
8
9
10

参考文献

[1] Nocedal J, Wright S. Numerical optimization[M]. Springer Science and Business Media, 2006.

[2] Hu J, Milzarek A, Wen Z, et al. Adaptive quadratically regularized Newton method for Riemannian optimization[J]. SIAM Journal on Matrix Analysis and Applications, 2018, 39(3): 1181-1207.

拟牛顿法

陈士祥

中国科学技术大学

致谢：本教案部分参考北京大学文再文教授《最优化与建模》

提纲

割线方程的推导

设$f(x)$是二阶连续可微函数. 对$\nabla f(x)$在点x^{k+1}处一阶泰勒近似, 得

$$\nabla f(x) = \nabla f(x^{k+1}) + \nabla^2 f(x^{k+1})(x - x^{k+1}) + \mathcal{O}(\left\|x - x^{k+1}\right\|^2),$$

令$x = x^k$, 且$s^k = x^{k+1} - x^k$为点差, $y^k = \nabla f(x^{k+1}) - \nabla f(x^k)$为梯度差, 得

$$\nabla^2 f(x^{k+1})s^k + \mathcal{O}(\left\|s^k\right\|^2) = y^k.$$

现忽略高阶项$\left\|s^k\right\|^2$, 只希望近似海瑟矩阵的矩阵B^{k+1}满足方程

$$B^{k+1}s^k = y^k,$$

或其逆矩阵H^{k+1}满足

$$H^{k+1}y^k = s^k.$$

上述两个方程即称为割线方程.

曲率条件

由于近似矩阵必须保证迭代收敛, 正如牛顿法要求海瑟矩阵正定, B^k正定也是必须的, 即有必要条件

$$\left(s^k\right)^{\mathrm{T}} B^{k+1} s^k > 0 \Longrightarrow \left(s^k\right)^{\mathrm{T}} y^k > 0,$$

定义

曲率条件　在迭代过程中满足$\left(s^k\right)^{\mathrm{T}} y^k > 0, \forall k \in \mathbb{N}^+$.

如果线搜索使用Wolfe准则:

$$\nabla f(x^k + \alpha d^k)^{\mathrm{T}} d^k \geqslant c_2 \nabla f(x^k)^{\mathrm{T}} d^k,$$

其中$c_2 \in (0,1)$. 上式即$\nabla f(x^{k+1})^{\mathrm{T}} s^k \geqslant c_2 \nabla f(x^k)^{\mathrm{T}} s^k$. 在不等式两边同时减去$\nabla f(x^k)^{\mathrm{T}} s^k$, 由于$c_2 - 1 < 0$且$s^k$是下降方向, 因此最终有

$$(y^k)^{\mathrm{T}} s^k \geqslant (c_2 - 1) \nabla f(x^k)^{\mathrm{T}} s^k > 0.$$

拟牛顿算法的基本框架

拟牛顿算法的基本框架为:

算法 1 拟牛顿算法框架

Require: 初始坐标$x^0 \in \mathbb{R}^n$, 初始矩阵$B^0 \in \mathbb{R}^{n\times n}$(或$H^0$), $k = 0$.
Ensure: x^K, B^K(或H^K).

1: 检查初始元素.
2: **while** 未达到停机准则 **do**
3: 计算方向$d^k = -(B^k)^{-1}\nabla f(x^k)$或$d^k = -H^k\nabla f(x^k)$.
4: 通过线搜索(Wolfe)产生步长$\alpha_k > 0$, 令$x^{k+1} = x^k + \alpha_k d^k$.
5: 更新海瑟矩阵的近似矩阵B^{k+1}或其逆矩阵H^{k+1}.
6: $k \leftarrow k + 1$.
7: **end while**

秩一更新(SR1)

定义

秩一更新　对于拟牛顿矩阵$B^k \in \mathbb{R}^{n\times n}$, 设$0 \neq u \in \mathbb{R}^n$且$a \in \mathbb{R}$待定, 则$uu^{\mathrm{T}}$是秩一矩阵, 且有秩一更新

$$B^{k+1} = B^k + auu^{\mathrm{T}}.$$

根据割线方程$B^{k+1}s^k = y^k$, 代入秩一更新的结果, 得到

$$(B^k + auu^{\mathrm{T}})s^k = y^k,$$

整理得

$$auu^{\mathrm{T}}s^k = (a \cdot u^{\mathrm{T}}s^k)u = y^k - B^k s^k.$$

由于$a \cdot u^{\mathrm{T}}s^k$是标量, 因此上式表明$u$和$y^k - B^k s^k$同向. 简单考虑不妨就令$u$和$y^k - B^k s^k$相等, 即$u = y^k - B^k s^k$. 代入上式得

$$(a \cdot (y^k - B^k s^k)^{\mathrm{T}} s^k)(y^k - B^k s^k) = y^k - B^k s^k.$$

秩一更新公式

再令$(a\cdot(y^k-B^ks^k)^{\mathrm{T}}s^k)\neq 0$, 则可以确定$a$为

$$a=\frac{1}{(y^k-B^ks^k)^{\mathrm{T}}s^k}.$$

由同样的过程可以推出基于H^k的秩一更新公式.

定理

拟牛顿算法的秩一更新公式 拟牛顿矩阵B^k的秩一更新公式为

$$B^{k+1}=B^k+\frac{uu^{\mathrm{T}}}{u^{\mathrm{T}}s^k},\quad u=y^k-B^ks^k,$$

拟牛顿矩阵H^k的秩一更新公式为

$$H^{k+1}=H^k+\frac{vv^{\mathrm{T}}}{v^{\mathrm{T}}y^k},\quad v=s^k-H^ky^k.$$

B^k和H^k的公式在形式上互为对偶. 实际上$H^k=(B^k)^{-1}$, 利用秩一更新的SMW公式即可推出基于H^k的公式, 反之亦然.

秩一更新公式的缺陷

即使B^k正定，由秩一公式更新的B^{k+1}无法保证正定.

定理

秩一更新公式使B^{k+1}正定的充分条件 使用秩一更新公式从B^k更新B^{k+1}, B^{k+1}正定的充分条件可以是:

(1) B^k正定;

(2) $u^{\mathrm{T}}s^k > 0$.

证明: 设$0 \neq w \in \mathbb{R}^n$, 则

$$w^{\mathrm{T}}B^{k+1}w = w^{\mathrm{T}}B^k w + \frac{w^{\mathrm{T}}uu^{\mathrm{T}}w}{u^{\mathrm{T}}s^k} = w^{\mathrm{T}}B^k w + \frac{(u^{\mathrm{T}}w)^2}{u^{\mathrm{T}}s^k} > 0.$$

同样地, 将上述定理中B换成H, $u^{\mathrm{T}}s^k$换成$v^{\mathrm{T}}y^k$, 仍然成立. 因此, 由于无法保证$u^{\mathrm{T}}s^k$或$v^{\mathrm{T}}y^k$恒大于0, 上述的秩一更新公式一般不用.

BFGS(Broyden-Fletcher-Goldfarb-Shanno) 公式

BFGS公式的核心思想是对B^k进行秩二更新.

定义

秩二更新　对于拟牛顿矩阵$B^k \in \mathbb{R}^{n\times n}$, 设$0 \neq u, v \in \mathbb{R}^n$且$a, b \in \mathbb{R}$待定, 则有秩二更新形式

$$B^{k+1} = B^k + auu^{\mathrm{T}} + bvv^{\mathrm{T}}.$$

根据割线方程, 将秩二更新的待定参量式代入, 得

$$B^{k+1}s^k = (B^k + auu^{\mathrm{T}} + bvv^{\mathrm{T}})s^k = y^k,$$

整理可得

$$(a \cdot u^{\mathrm{T}}s^k)u + (b \cdot v^{\mathrm{T}}s^k)v = y^k - B^k s^k.$$

简单的取法是令$(a \cdot u^{\mathrm{T}}s^k)u$对应$y^k$相等, $(b \cdot v^{\mathrm{T}}s^k)v$对应$-B^k s^k$相等, 即有

$$a \cdot u^{\mathrm{T}}s^k = 1, \quad u = y^k, \quad b \cdot v^{\mathrm{T}}s^k = -1, \quad v = B^k s^k.$$

BFGS公式

将上述参量代入割线方程, 即得BFGS更新公式

$$B^{k+1} = B^k + \frac{uu^{\mathrm{T}}}{(s^k)^{\mathrm{T}}u} - \frac{vv^{\mathrm{T}}}{(s^k)^{\mathrm{T}}v}.$$

利用SMW公式以及$H^k = (B^k)^{-1}$, 可以推出关于H^k的BFGS公式.

定义

BFGS公式 在拟牛顿类算法中, 基于B^k的*BFGS*公式为

$$B^{k+1} = B^k + \frac{y^k \left(y^k\right)^{\mathrm{T}}}{\left(s^k\right)^{\mathrm{T}} y^k} - \frac{B^k s^k \left(B^k s^k\right)^{\mathrm{T}}}{\left(s^k\right)^{\mathrm{T}} B^k s^k},$$

基于H^k的*BFGS*公式为

$$H^{k+1} = \left(I - \frac{s^k \left(y^k\right)^{\mathrm{T}}}{\left(s^k\right)^{\mathrm{T}} y^k}\right)^{\mathrm{T}} H^k \left(I - \frac{s^k \left(y^k\right)^{\mathrm{T}}}{\left(s^k\right)^{\mathrm{T}} y^k}\right) + \frac{s^k \left(s^k\right)^{\mathrm{T}}}{\left(s^k\right)^{\mathrm{T}} y^k}.$$

Sherman-Morrison-Woodbury (SMW)公式

秩一校正的求逆公式

Sherman-Morrison-Woodbury公式： 设$A \in \mathbb{R}^{n\times n}$是非奇异阵，$u, v \in \mathbb{R}^n$是任意向量。若$1 + v^T A^{-1} u \neq 0$, 则$A$的秩一校正$A + uv^T$非奇异，且其逆可以表示为

$$(A + uv^T)^{-1} = A^{-1} - \frac{A^{-1}uv^T A^{-1}}{1 + v^T A^{-1} u}. \tag{1}$$

Sherman-Morrison-Woodbury推广公式： 设$A \in \mathbb{R}^{n\times n}$是非奇异阵，$U \in \mathbb{R}^{n\times k}, V \in \mathbb{R}^{n\times k}$是任意矩阵。若$I_k + V^T A^{-1} U$可逆, 则$A + UV^T$非奇异，且其逆可以表示为

$$(A + UV^T)^{-1} = A^{-1} - A^{-1}U(I_k + V^T A^{-1} U)^{-1} V^T A^{-1}. \tag{2}$$

推导H^k的BFGS公式之提示

对于可逆矩阵$B \in \mathbb{R}^{n\times n}$与矩阵$U \in \mathbb{R}^{n\times m}, V \in \mathbb{R}^{n\times m}$，SMW公式为：

$$\left(B + UV^{\mathrm{T}}\right)^{-1} = B^{-1} - B^{-1}U\left(I + V^{\mathrm{T}}B^{-1}U\right)^{-1}V^{\mathrm{T}}B^{-1}.$$

在BFGS的推导中，关于B^k的更新公式为：

$$B_{k+1} = B_k + \frac{y_k y_k^{\mathrm{T}}}{s_k^{\mathrm{T}} y_k} - \frac{B_k s_k \left(B_k s_k\right)^{\mathrm{T}}}{s_k^{\mathrm{T}} B_k s_k} = B_k + \begin{pmatrix} -\frac{B_k s_k}{s_k^{\mathrm{T}} B_k s_k} & \frac{y_k}{s_k^{\mathrm{T}} y_k} \end{pmatrix} \begin{pmatrix} s_k^{\mathrm{T}} B_k \\ y_k^{\mathrm{T}} \end{pmatrix}.$$

对照SMW公式，令式中$B = B_k$，且

$$U_k = \begin{pmatrix} -\frac{B_k s_k}{s_k^{\mathrm{T}} B_k s_k} & \frac{y_k}{s_k^{\mathrm{T}} y_k} \end{pmatrix}, \quad V_k = \begin{pmatrix} B_k s_k & y_k \end{pmatrix},$$

此时公式的左端就等于B_{k+1}^{-1}，且右端只需计算一个2阶矩阵的逆．假设$B_k^{-1} = H_k$，由SMW公式就得到

$$H_{k+1} = \left(B_k + U_k V_k^{\mathrm{T}}\right)^{-1} = \left(I - \frac{s_k y_k^{\mathrm{T}}}{s_k^{\mathrm{T}} y_k}\right) H_k \left(I - \frac{y_k s_k^{\mathrm{T}}}{s_k^{\mathrm{T}} y_k}\right) + \frac{s_k s_k^{\mathrm{T}}}{s_k^{\mathrm{T}} y_k}.$$

BFGS公式的有效性

BFGS公式产生的B^{k+1}或H^{k+1}是否正定呢?

定理

BFGS公式使拟牛顿矩阵正定的充分条件 使用秩二更新公式从B^k或H^k更新B^{k+1}或H^{k+1}, 拟牛顿矩阵正定的充分条件可以是:

(1) B^k或H^k正定;

(2) 满足曲率条件$\left(s^k\right)^{\mathrm{T}} y^k > 0, \forall k \in \mathbb{N}^+$.

证明上述定理, 只需要从基于H^k的BFGS公式分析即可, 从而得到H^{k+1}和其逆B^{k+1}均正定.

因为在确定步长时使用某一Wolfe准则线搜索即可满足曲率条件, 因此BFGS公式产生的拟牛顿矩阵有望保持正定, 是有效算法.

从优化意义理解BFGS格式

基于H^k的BFGS格式恰好是优化问题

$$\begin{aligned} \min_{H} \quad & \mathbf{OPT} = \left\| H - H^k \right\|_W, \\ s.t. \quad & H = H^{\mathrm{T}}, \\ & H y^k = s^k. \end{aligned}$$

的解. 上式中$\|\cdot\|_W$是加权范数, 定义为

$$\|H\|_W = \left\| W^{1/2} H W^{1/2} \right\|_F,$$

且W满足割线方程, 即$W s^k = y^k$.

- $\|\cdot\|_W$可以让得到的拟牛顿公式在一定条件下同样满足仿射不变性（请回顾：“牛顿法为什么好”-牛顿法的仿射不变性质）。
- 注意$H y^k = s^k$是割线方程, 因此优化问题的意义是在满足割线方程的对称矩阵中找到距离H^k最近的矩阵H作为H^{k+1}. 因此我们可以进一步认知, BFGS格式更新的拟牛顿矩阵是正定对称的, 且在满足割线方程的条件下采取的是最佳逼近策略.

DFP公式

DFP公式利用与BFGS公式类似的推导方法, 不同的是其以割线方程$H^{k+1}y^k = s^k$为基础进行对H^k的秩二更新.
基于H^k满足的DFP公式, 利用SMW公式以及$B^k = (H^k)^{-1}$, 可以推出关于B^k的DFP公式.
(关键的推导步骤仍然可以参考推导BFGS公式时给出的提示)

定义

DFP公式 基于H^k的*DFP*更新公式为

$$H^{k+1} = H^k - \frac{H^k y^k (H^k y^k)^{\mathrm{T}}}{(y^k)^{\mathrm{T}} H^k y^k} + \frac{s^k (s^k)^{\mathrm{T}}}{(y^k)^{\mathrm{T}} s^k},$$

基于B^k的*DFP*更新公式为

$$B^{k+1} = (I - \frac{y^k (s^k)^{\mathrm{T}}}{(s^k)^{\mathrm{T}} y^k})^{\mathrm{T}} B^k (I - \frac{y^k (s^k)^{\mathrm{T}}}{(s^k)^{\mathrm{T}} y^k}) + \frac{y^k (y^k)^{\mathrm{T}}}{(s^k)^{\mathrm{T}} y^k}.$$

从优化意义上理解DFP公式

有了BFGS公式的优化意义做铺垫, 讨论DFP公式的优化意义显得十分简单. 利用对偶性质, 基于B^k的DFP格式将是优化问题

$$\begin{aligned} \min_{B} \quad & \mathbf{OPT} = \left\| B - B^k \right\|_W, \\ s.t. \quad & B = B^{\mathrm{T}}, \\ & Bs^k = y^k. \end{aligned}$$

的解. 上式中$\|\cdot\|_W$是加权范数, 定义为

$$\|B\|_W = \left\| W^{1/2} B W^{1/2} \right\|_F,$$

且W满足另一割线方程, 即$Wy^k = s^k$.
注意$Bs^k = y^k$是另一割线方程, 因此优化问题的意义是在满足割线方程的对称矩阵中找到距离B^k最近的矩阵B作为B^{k+1}.

BFGS另一种解释

$X = B_{k+1}$是下面优化问题的最优解：

$$\begin{aligned} &\min_X Tr(B_k^{-1}X) - \log\det(B_k^{-1}X) - n \\ &\text{s.t.} \quad Xs_k = y_k, X^T = X. \end{aligned} \tag{3}$$

上述问题中的目标函数，是概率分布$N(0,X)$和$N(0,B_k)$的相对熵。

对于**DFP**公式：$X = B_{k+1}$是下面优化问题的最优解：

$$\begin{aligned} &\min_X Tr(H_k^{-1}X) - \log\det(H_k^{-1}X) - n \\ &\text{s.t.} \quad Xy_k = s_k, X^T = X. \end{aligned} \tag{4}$$

上述问题中的目标函数，设$Y = X^{-1}$有

$$Tr(H_k^{-1}X) - \log\det(H_k^{-1}X) - n = Tr(B_kY^{-1}) - \log\det(B_kY^{-1}) - n$$

是概率分布$N(0,B_k)$和$N(0,Y)$的相对熵。

DFP公式的缺陷

尽管DFP格式与BFGS对偶, 但从实际效果而言, DFP格式的求解效率整体上不如BFGS格式. M.J.D. Powell曾求解问题

$$\min_{x\in\mathbb{R}^2} \quad f(x) = \frac{1}{2}\|x\|_2^2 .$$

设置初始值

$$B^0 = \begin{pmatrix} 1 & 0 \\ 0 & \lambda \end{pmatrix}, \quad x_1 = \begin{pmatrix} \cos\psi \\ \sin\psi \end{pmatrix},$$

其中$\tan^2\psi = \lambda$. 当误差阈$\epsilon = 10^{-4}$时, 分别取λ为不同的值, 使用BFGS算法与DFP算法所产生的迭代步数分别如下表(见下页)所示. 由此看出, 在本问题中, BFGS算法的求解效率要远高于DFP算法.

(参考文献: Powell M J D. How bad are the BFGS and DFP methods when the objective function is quadratic?[J]. Mathematical Programming, 1986, 34(1): 34-47.)

DFP公式的缺陷

Table: BFGS方法的迭代次数

λ \ ϵ	0.1	0.01	10^{-4}	10^{-8}
10	5	6	8	10
100	7	8	10	12
10^4	12	13	15	17
10^6	17	18	20	22
10^9	24	25	27	29

Table: DFP方法的迭代次数

λ \ ϵ	0.1	0.01	10^{-4}	10^{-8}
10	10	13	16	19
30	25	32	37	40
100	80	99	107	111
300	237	290	307	313
10^3	787	958	1006	1014

提纲

BFGS全局收敛性

根据对BFGS格式有效性的分析, 我们先确保初始矩阵B^0是对称正定的.

定理

BFGS全局收敛性 设初始矩阵B^0是对称正定矩阵, 目标函数$f(x)$是二阶连续可微函数, 下水平集

$$\mathcal{L} = \{x \in \mathbb{R}^n | f(x) \leqslant f(x^0)\}$$

凸, 且存在$m, M \in \mathbb{R}^+$使得对$\forall z \in \mathbb{R}^n, x \in \mathcal{L}$满足

$$m \|z\|^2 \leqslant z^{\mathrm{T}} \nabla^2 f(x) z \leqslant M \|z\|^2$$

(即$z^{\mathrm{T}} \nabla^2 f(x) z$被$\|z\|$控制), 那么*BFGS*格式结合*Wolfe*线搜索的拟牛顿算法全局收敛到$f(x)$的极小值点x^*.

BFGS局部收敛性与R-收敛速度

上述全局定理说明在一定假设下, 使用BFGS格式确定下降方向, 搭配Wolfe线搜索确定步长后是全局收敛的. 下面这个定理从局部收敛性给出了其收敛速度.

定理

BFGS局部收敛性 设$f(x)$二阶连续可微, 点列$\{x_k\}$是由*BFGS*格式产生的, 并收敛于x^*, $\nabla^2 f(x^*)$是对称正定矩阵. 设初始矩阵B^0为任意的对称正定矩阵, 那么存在$0 \leqslant c < 1$, $K \in \mathbb{N}^+$, 使得对$\forall k > N$, 成立

$$f\left(x^{k+1}\right) - f\left(x^*\right) \leqslant c^{k-K+1}\left(f\left(x^K\right) - f\left(x^*\right)\right),$$

$$\sum_{k=0}^{\infty} \left\| x^k - x^* \right\| < \infty.$$

上述定理表明, 序列$\left\{f\left(x^k\right) - f\left(x^*\right)\right\}_k$是压缩的, 收敛将具有Q-线性收敛速度.

BFGS格式的Q-收敛速度

由局部收敛性定理, 可以进一步推出BFGS的Q-收敛速度.

定理

BFGS的Q-超线性收敛速度 除要求*BFGS*局部收敛性的假设外, 再要求f的海瑟矩阵在x^*处*Lip*-连续, 则有

$$\lim_{k\to\infty}\frac{\left\|x^{k+1}-x^*\right\|}{\left\|x^k-x^*\right\|}=0.$$

即$\left\{x^k\right\}$为*Q*-超线性收敛到x^*.

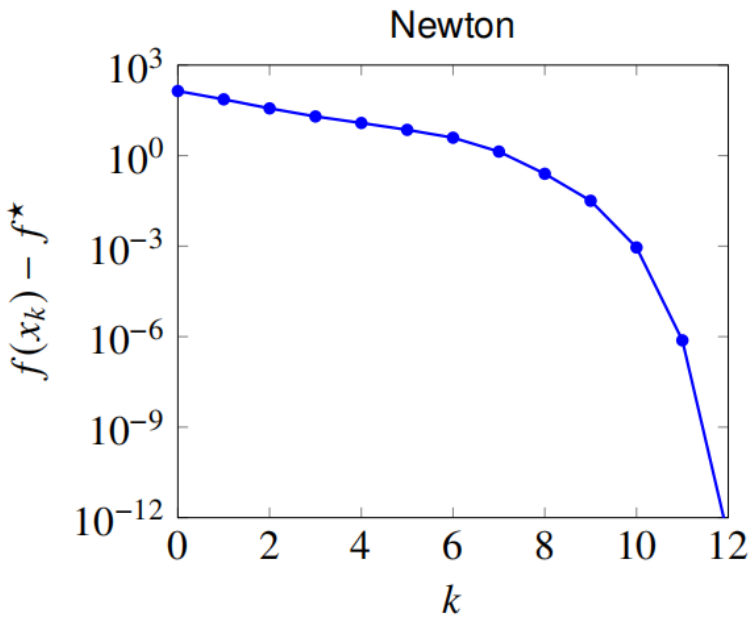
以BFGS格式为代表的拟牛顿类算法由于仅仅使用了海瑟矩阵的近似, 因此很难达到二阶收敛速度, 最多只能达到Q-超线性收敛速度. 但是, 由于拟牛顿方法对近似矩阵的更新代价可能远小于牛顿方法计算海瑟矩阵的代价, 因此它在大规模问题中的开销可能远小于牛顿算法, 较为实用.

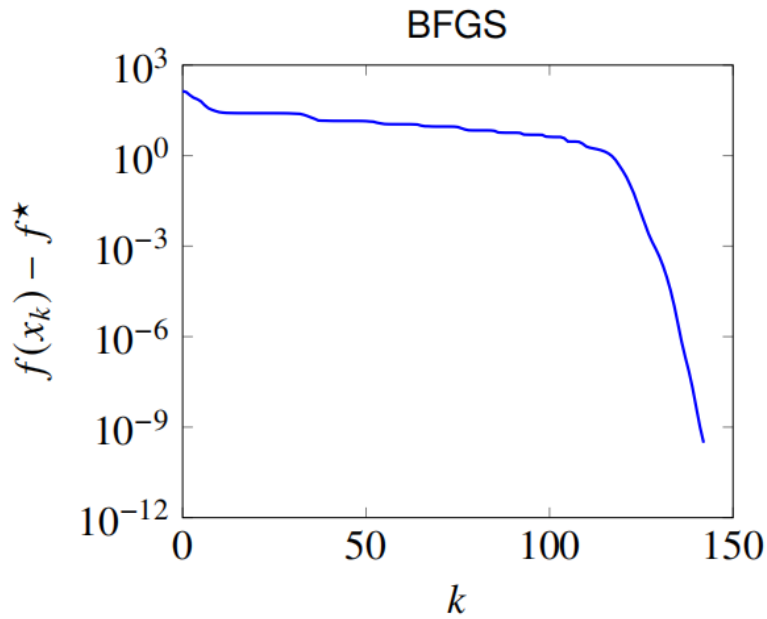
BFGS方法的收敛速度之例

例　考虑极小化问题

$$\min_{x\in\mathbb{R}^{100}} \quad c^{\mathrm{T}}x - \sum_{i=1}^{500} \ln\left(b_i - a_i^{\mathrm{T}}x\right),$$

下图展示了误差$f(x_k) - f^*$与迭代次数k之间的关系(k是迭代次数). 虽然 BFGS方法的迭代次数显著得多, 但由于牛顿法每次迭代的计算代价为$\mathcal{O}(n^3)$加上计算海瑟矩阵的代价, 而BFGS方法的每步计算代价仅为$\mathcal{O}(n^2)$, 因此BFGS算法可能更快取得优势解.





提纲

有限内存方法的基本思路

基本思路　标准的拟牛顿近似矩阵的更新公式可以记为

$$B^{k+1} = g(B^k, s^k, y^k),\ s^k = x^{k+1} - x^k, y^k = \nabla f(x^{k+1}) - \nabla f(x^k).$$

如果只保存最近的m组数据, 那么迭代公式可以写成

$$B^{k+1} = g(g(\cdots g(B^{k-m+1}, s^{k-m+1}, y^{k-m+1}))).$$

考虑BFGS方法：

$$d^k = -(B^k)^{-1}\nabla f(x^k) = -H^k \nabla f(x^k).$$

重写BFGS更新公式为

$$H^{k+1} = (V^k)^{\mathrm{T}} H^k V^k + \rho_k s^k (s^k)^{\mathrm{T}},$$

其中

$$\rho_k = \frac{1}{(y^k)^{\mathrm{T}} s^k}, \quad V^k = I_{n\times n} - \rho_k y^k (s^k)^{\mathrm{T}}.$$

将上式递归地展开m次, 即

$$H^k = \left(\prod_{j=k-m}^{k-1} V^j\right)^{\mathrm{T}} H^{k-m} \left(\prod_{j=k-m}^{k-1} V^j\right) +$$
$$\rho_{k-m} \left(\prod_{j=k-m+1}^{k-1} V^j\right)^{\mathrm{T}} s^{k-m} \left(s^{k-m}\right)^{\mathrm{T}} \left(\prod_{j=k-m}^{k-1} V^j\right) + \cdots +$$
$$\rho_{k-1} s^{k-1} \left(s^{k-1}\right)^{\mathrm{T}}.$$

为了节省内存, 我们只展开m次, 利用H^{k-m}进行计算, 即可求出H^{k+1}.

下面介绍一种不计算H^k, 只利用展开式计算$d^k = -H^k \nabla f(x^k)$的巧妙算法: 双循环递归算法. 它利用迭代式的结构尽量节省计算d^k的开销.

将等式两边同右乘$\nabla f(x^k)$, 则等式左侧为$-d^k$. 观察等式右侧需要计算

$$V^{k-1}\nabla f(x^k), \cdots, V^{k-m}\cdots V^{k-1}\nabla f(x^k).$$

这些计算可以递归地进行. 同时在计算$V^{k-l}\cdots V^{k-1}\nabla f(x^k)$的过程中, 可以计算上一步的$\rho_{k-l}(s^{k-l})^{\mathrm{T}}[V^{k-l+1}\cdots V^{k-1}\nabla f(x^k)]$, 这是一个标量. 记

$$q = V^{k-m}\cdots V^{k-1}\nabla f(x^k),$$

$$\alpha_i = \rho_{k-l}(s^{k-l})^{\mathrm{T}}[V^{k-l+1}\cdots V^{k-1}\nabla f(x^k)],$$

因此递归公式可化为如下的形式:

$$H^k\nabla f(x^k) = \left(\prod_{j=k-m}^{k-1} V^j\right)^{\mathrm{T}} H^{k-m}q + \left(\prod_{j=k-m+1}^{k-1} V^j\right)^{\mathrm{T}} s^{k-m}\alpha_{k-m} + \cdots + s^{k-1}\alpha_{k-1}.$$

有限内存BFGS方法

在双循环递归算法中, 除了上述第一个循环递归过程(自下而上)外, 还有以下第二个循环递归过程. 我们需要在公式中自上而下合并每一项. 以前两项为例, 它们有公共的因子$(V^{k-m+1}\cdots V^{k-1})^{\mathrm{T}}$, 提取后可以将前两项写为(注意将$V^{k-m}$的定义回代)

$$\begin{aligned}&\left(V^{k-m+1}\cdots V^{k-1}\right)^{\mathrm{T}}\left[\left(V^{k-m}\right)^{\mathrm{T}}r+\alpha_{k-m}s^{k-m}\right]\\&=\left(V^{k-m+1}\cdots V^{k-1}\right)^{\mathrm{T}}\left(r+\left(\alpha_{k-m}-\beta\right)s^{k-m}\right),\end{aligned}$$

这正是第二个循环的迭代格式. 注意合并后原递归式的结构仍不变, 因此可以递归地计算下去. 最后, 变量r就是我们期望的结果$H^k\nabla f(x^k)$.

L-BFGS双循环递归算法

拟牛顿算法的基本框架为:

算法 2 L-BFGS双循环递归

Require: 初始化$q \leftarrow \nabla f(x^k)$.
Ensure: r, 即$H^k \nabla f(x^k)$.

检查初始元素.
for $i = k-1, \cdots, k-m$ **do**
    计算并保存$\alpha_i \leftarrow \rho_i (s^i)^{\mathrm{T}} q$.
    更新$q \leftarrow q - \alpha_i y^i$.
end for
初始化$r \leftarrow \hat{H}^{k-m} q$, 其中$\hat{H}^{k-m}$是$H^{k-m}$的近似矩阵.
for $i = k-m, \cdots, k-1$ **do**
    计算$\beta \leftarrow \rho_i (y^i)^{\mathrm{T}} r$.
    更新$r \leftarrow r + (\alpha_i - \beta) s^i$.
end for

算法分析

L-BFGS双循环递归算法约需要$4mn$次乘法运算, $2mn$次加法运算; 若近似矩阵$\hat{H}^{k-m}$是对角矩阵, 则额外需要n次乘法运算. 由于m不会很大, 因此算法的复杂度是$\mathcal{O}(mn)$. 算法需要的额外存储为临时变量α_i, 其大小是$\mathcal{O}(m)$.

$\hat{H}^{k-m}$的一种取法可以是取对角矩阵

$$\hat{H}^{k-m} = \gamma_k I_{n\times n} \triangleq \frac{\left(s^{k-1}\right)^{\mathrm{T}} y^{k-1}}{\left(y^{k-1}\right)^{\mathrm{T}} y^{k-1}} I_{n\times n}.$$

这恰好是BB方法的第一个步长.

Algorithm 3 L-BFGS方法

Input: 选择初始点x^0, 参数$m > 0$, $k \leftarrow 0$.
Output: 达到收敛准则后的x^{k+1}.

检查初始元素 **while** 未达到收敛准则 **do**
  选取近似矩阵$\hat{H}^{k-m}$;
  使用算法2(L-BFGS双循环递归算法)计算$d^k = -H^k \nabla f(x^k)$;
  使用满足Wolfe准则的线搜索算法确定步长α_k;
  更新$x^{k+1} = x^k + \alpha_k d^k$.
  if $k > m$ **then**
    从内存空间中删除s^{k-m}, y^{k-m}.
  计算并保存$s^k = x^{k+1} - x^k, y^k = \nabla f(x^{k+1}) - \nabla f(x^k)$.
  $k \leftarrow k + 1$.

对偶算法

陈士祥

中国科学技术大学

致谢：本教案部分参考北京大学文再文教授《最优化与建模》

提纲

对偶方法

动机：前面课程的算法直接求解原问题，我们也可也求解对偶问题。

- 次梯度法：速度慢，步长选择困难
- 梯度法：需要对偶函数可微
- 对偶函数可能不可微，或定义域非平凡
- 对原始函数加小的强凸项，将对偶函数光滑化

近似点梯度法（本讲）：对偶函数分裂成两项

- 一项是梯度利普希茨连续函数
- 另一项有方便计算的近似点算子

问题形式

设f，h是闭凸函数，考虑如下形式的问题:

$$(\mathrm{P}) \quad \min_{x\in\mathbb{R}^n} \quad \psi(x) = f(x) + h(Ax).$$

该问题即使h有简单的近似点映射，但$h(Ax)$并无简单近似点映射。
引入新变量$y = Ax$，我们将f, h拆分，考虑问题:

$$(\mathrm{P}) \quad \min_{x\in\mathbb{R}^n} \quad \psi(x) = f(x) + h(y), \ \text{s.t. } Ax = y.$$

拉格朗日函数为:

$$L(x, y, z) = f(x) + g(y) + z^T(Ax - y)$$

对偶问题

$$(\mathrm{D}) \quad \max_{z} \quad \phi(z) = -f^*(-A^{\mathrm{T}}z) - h^*(z).$$

对偶方法

使用一阶算法求解对偶问题：

$$\max \quad -f^*(-A^T z) - h^*(z)$$

对偶问题容易使用一阶算法求解的原因：

- 对偶问题无约束，或者约束很简单
- 对偶问题可微，或者带有简单的非光滑项
- 对偶问题可以具有“可分”结构，易于分解为更小的问题

强凸函数共轭函数的性质

定理：设$f(x)$是适当且闭的强凸函数，强凸参数为$\mu > 0$，则$f^*(y)$在全空间$\mathbb{R}^n$上有定义，$f^*(y)$是梯度$\frac{1}{\mu}$-利普希茨连续的可微函数.

证明：

- 对任意的$y \in \mathbb{R}^n$，$f(x) - x^{\mathrm{T}}y$是强凸函数，因此对任意的$y \in \mathbb{R}^n$，存在唯一的$x \in \mathbf{dom}\, f$，使得$f^*(y) = x^{\mathrm{T}}y - f(x)$. 根据最优性条件

$$y \in \partial f(x) \Leftrightarrow f^*(y) = x^{\mathrm{T}}y - f(x).$$

- 由于$f(x)$是闭凸函数，二次共轭为其本身，于是对同一组x, y有

$$x^{\mathrm{T}}y - f^*(y) = f(x) = f^{**}(x) = \sup_y \left\{x^{\mathrm{T}}y - f^*(y)\right\}.$$

- 这说明y也使得$x^{\mathrm{T}}y - f^*(y)$取到最大值. 根据一阶最优性条件，

$$x \in \partial f^*(y).$$

- 再根据x的唯一性容易推出$\partial f^*(y)$中只含一个元素，故$f^*(y)$可微.

- 下证$f^*(y)$为梯度$\frac{1}{\mu}$-利普希茨连续的. 对任意的y_1, y_2，存在唯一的$x_1, x_2 \in \mathbf{dom}\, f$使得

$$y_1 \in \partial f(x_1), \quad y_2 \in \partial f(x_2).$$

- 根据次梯度性质以及$f(x) - \frac{\mu}{2}\|x\|^2$是凸函数，

$$f(x_2) \geq f(x_1) + (y_1 - \mu x_1)^{\mathrm{T}}(x_2 - x_1),$$
$$f(x_1) \geq f(x_2) + (y_2 - \mu x_2)^{\mathrm{T}}(x_1 - x_2),$$

- 将上述两式相加得

$$(y_1 - y_2)^{\mathrm{T}}(x_1 - x_2) \geq \mu\|x_1 - x_2\|^2.$$

- 根据x和y的关系我们有$x_1 = \nabla f^*(y_1), x_2 = \nabla f^*(y_2)$，代入上式可得

$$(y_1 - y_2)^{\mathrm{T}}(\nabla f^*(y_1) - \nabla f^*(y_2)) \geq \mu\|\nabla f^*(y_1) - \nabla f^*(y_2)\|^2.$$

这正是$\nabla f^*(y)$的余强制性，可知$\nabla f^*(y)$是$\frac{1}{\mu}$-利普希茨连续的.

等式约束的情况

$$\begin{array}{ll}\min & f(x)\\ \text{s.t.} & Ax=b\end{array} \qquad \min \quad f^*(-A^Tz)+b^Tz$$

对偶次梯度上升算法：(假设**dom** $f^*=\mathbb{R}^n$):

$$\hat{x}=\operatorname*{argmin}_x(f(x)+z^TAx),\quad z^+=z+t(A\hat{x}-b)$$

- 由于对偶问题是求解最大化问题，故是次梯度上升法。
- $\hat{x}$ 是f^* 在点$-A^Tz$处的次梯度，（$i.e.,\hat{x}\in\partial f^*(-A^Tz)$）
- $b-A\hat{x}$ 是$f^*(-A^Tz)+b^Tz$ 在点z处的次梯度
- 若f强凸，则变为“对偶梯度上升算法”

当计算$\hat{x}$很容易时，该算法很有效。(for example, f is separable)

f可分时：Dual decomposition

convex problem with separable objective

$$\begin{array}{ll} \min & f_1(x_1)+f_2(x_2) \\ \text{s.t.} & A_1x_1+A_2x_2 \le b \end{array}$$

constraint is *complicating or coupling* constraint

dual problem

$$\begin{array}{ll} \max & -f_1^*(-A_1^Tz)-f_2^*(-A_2^Tz)-b^Tz \\ \text{s.t.} & z \ge 0 \end{array}$$

can be solved by (sub-)gradient projection if $z \ge 0$ is the only constraint

subproblems: to calculate $f_j^*(-A_j^T z)$ and a (sub-) gradient for it,

$$\min \text{ (over } x_j) \;\; f_j(x_j) + z^T A_j x_j$$

optimal value is $f_j^*(-A_j^T z)$; minimizer $\hat{x}_j$ is in $\partial f_j^*(-A_j^T z)$

对偶投影次梯度法

$$\begin{aligned} \hat{x}_j &= \operatorname*{argmin}_{x_j} (f_j(x_j) + z^T A_j x_j), \;\; j = 1, 2 \\ z^+ &= (z + t(A_1 \hat{x}_1 + A_2 \hat{x}_2 - b))_+ \end{aligned}$$

- 可分性的作用：minimization problems over x_1, x_2 are independent
- z-update is projected subgradient step ($u_+ = \max\{u, 0\}$ elementwise)

对偶问题中的复合结构

为了在对偶问题上使用近似点梯度法，$\phi(z) = -f^*(-A^\top z) - h^*(z)$ 需要满足“可微函数+凸函数”的复合形式：

- h 或h^*的近似点算子容易计算（有闭形式或简单算法）
- f 是闭的强凸函数

  我们可以说明$f^*(-A^{\mathrm{T}}z)$ 的梯度是利普希茨连续的：

$$\|A\nabla f^*(-A^{\mathrm{T}}z_1) - A\nabla f^*(-A^{\mathrm{T}}z_2)\|_2 \leq \frac{\|A\|_2^2}{\mu}\|z_1 - z_2\|_2$$

对偶近似点梯度法

考虑在对偶问题上应用近似点梯度算法，每次迭代更新如下：

$$z^{k+1} = \mathrm{prox}_{th^*}\left(z^k + tA\nabla f^*\left(-A^{\mathrm{T}}z^k\right)\right)$$

对偶问题是取最大值，因此邻近算子内部应该取上升方向.
进一步引入变量$x^{k+1} = \nabla f^*(-A^{\mathrm{T}}z^k)$，迭代格式等价于

$$x^{k+1} = \arg\min_x \left\{f(x) + \left(A^{\mathrm{T}}z^k\right)^{\mathrm{T}}x\right\}, \quad z^{k+1} = \mathrm{prox}_{th^*}\left(z^k + tAx^{k+1}\right)$$

- 如果f 可分, x^{k+1} 的计算可分解为多个独立的问题
- 步长t 可取常数或采取回溯线搜索法
- 也可使用加速近似点梯度法，例如FISTA

下面我们将提供另一种角度来理解对偶近似点梯度法.

回顾：Moreau 分解

- 设f是定义在$\mathbb{R}^n$上的适当的闭凸函数，则对任意的$x \in \mathbb{R}^n$，

$$x = \mathrm{prox}_f(x) + \mathrm{prox}_{f^*}(x);$$

- 或更一般地，

$$x = \mathrm{prox}_{\lambda f}(x) + \lambda \mathrm{prox}_{\lambda^{-1} f^*}\left(\frac{x}{\lambda}\right),$$

其中$\lambda > 0$为任意正实数.

- Moreau 分解的结论表明：对任意的闭凸函数f，空间$\mathbb{R}^n$上的恒等映射总可以分解成两个函数f与f^*邻近算子的和.

对偶近似点梯度法：交替极小的解释

- 取$\lambda = t, f = h^*$，并注意到$h^{**} = h$，我们有

$$\begin{aligned} z^k + tAx^{k+1} &= \mathrm{prox}_{th^*}(z^k + tAx^{k+1}) + t\mathrm{prox}_{t^{-1}h}\left(\frac{z^k}{t} + Ax^{k+1}\right) \\ &= z^{k+1} + t\mathrm{prox}_{t^{-1}h}(\frac{z^k}{t} + Ax^{k+1}), \end{aligned}$$

- 由此给出对偶近似点梯度法等价的针对原始问题的更新格式：

$$\begin{aligned} x^{k+1} &= \underset{x}{\mathrm{argmin}} \left\{ f(x) + (z^k)^{\mathrm{T}} Ax \right\}, \\ y^{k+1} &= \mathrm{prox}_{t^{-1}h}\left(\frac{z^k}{t} + Ax^{k+1}\right) \\ &= \underset{y}{\mathrm{argmin}} \left\{ h(y) - (z^k)^{\mathrm{T}}(y - Ax^{k+1}) + \frac{t}{2}\|Ax^{k+1} - y\|_2^2 \right\}, \\ z^{k+1} &= z^k + t(Ax^{k+1} - y^{k+1}). \end{aligned}$$

交替极小方法

- 考虑等价问题:

$$\min_{x,y} f(x)+h(y), \quad \text{s.t.} \quad y=Ax$$

- 定义拉格朗日函数和增广拉格朗日函数:

$$L(x,y,z)=f(x)+h(y)-z^{\mathrm{T}}(y-Ax)$$

$$L_t(x,y,z)=f(x)+h(y)-z^{\mathrm{T}}(y-Ax)+\frac{t}{2}\|y-Ax\|^2$$

- 等价的交替极小格式是

$$\begin{aligned} x^{k+1} &= \arg\min_x L(x,y^k,z^k) \\ y^{k+1} &= \arg\min_y L_t(x^{k+1},y,z^k) \\ z^{k+1} &= z^k+t(Ax^{k+1}-y^{k+1}) \end{aligned}$$

- 对偶近似点梯度法等价于对原始约束问题使用上述交替极小化方法
- 选择何种形式，取决于函数h或h^*中，哪一个近似点映射易求

提纲

正则化范数近似

假设f是强凸函数，$\|\cdot\|$是任意一种范数，考虑

$$\min\ f(x) + \|Ax - b\|$$

对应原始问题我们有$h(y) = \|y - b\|$

$$h^*(z) = \begin{cases} b^{\mathrm{T}} z & \|z\|_* \le 1 \\ +\infty & \text{其他} \end{cases} \qquad \mathrm{prox}_{th^*}(x) = \mathcal{P}_{\|z\|_* \le 1}(x - tb)$$

其中$\|\cdot\|_*$表示$\|\cdot\|$的对偶范数. 从而对偶问题为：

$$\max_{\|z\|_* \le 1} \quad -f^*(-A^{\mathrm{T}} z) - b^{\mathrm{T}} z$$

应用对偶近似点梯度法，更新如下：

$$x^{k+1} = \operatorname*{argmin}_x \left\{ f(x) + (A^{\mathrm{T}} z^k)^{\mathrm{T}} x \right\}$$
$$z^{k+1} = \mathcal{P}_{\|z\|_* \le 1}(z^k + t(Ax^{k+1} - b))$$

正则化范数近似

考虑等价问题

$$\min_{x,y} f(x) + \|y\|, \quad \text{s.t. } Ax - b = y$$

交替极小化格式是

$$x^{k+1} = \operatorname*{argmin}_x f(x) + \|y^k\| + (z^k)^{\mathrm{T}}(Ax - b - y^k)$$

$$y^{k+1} = \operatorname*{argmin}_y f(x^{k+1}) + \|y\| + (z^k)^{\mathrm{T}}(Ax^{k+1} - b - y) + \frac{t}{2}\|Ax^{k+1} - b - y\|_2^2$$

$$z^{k+1} = z^k + t(Ax^{k+1} - b - y^{k+1})$$

例

假设f是强凸函数，考虑

$$\min \quad f(x) + \sum_{i=1}^{p} \|B_i x\|_2,$$

其中$B_i \in \mathbb{R}^{m_i \times n}$. 令$h(y_1, y_2, \cdots, y_p) = \sum_{i=1}^{p} \|y_i\|_2$, $y_i \in \mathbb{R}^{m_i}$，以及

$$A = \begin{bmatrix} B_1^{\mathrm{T}} & B_1^{\mathrm{T}} & \cdots & B_p^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}.$$

根据$\|\cdot\|_2$的共轭函数定义，对偶问题形式如下：

$$\max_{\|z_i\|_2 \le 1} \quad -f^* \left(-\sum_{i=1}^{p} B_i^{\mathrm{T}} z_i \right),$$

记C_i是$\mathbb{R}_{m_i}$中的单位欧几里得球，对偶近似点梯度法更新如下：

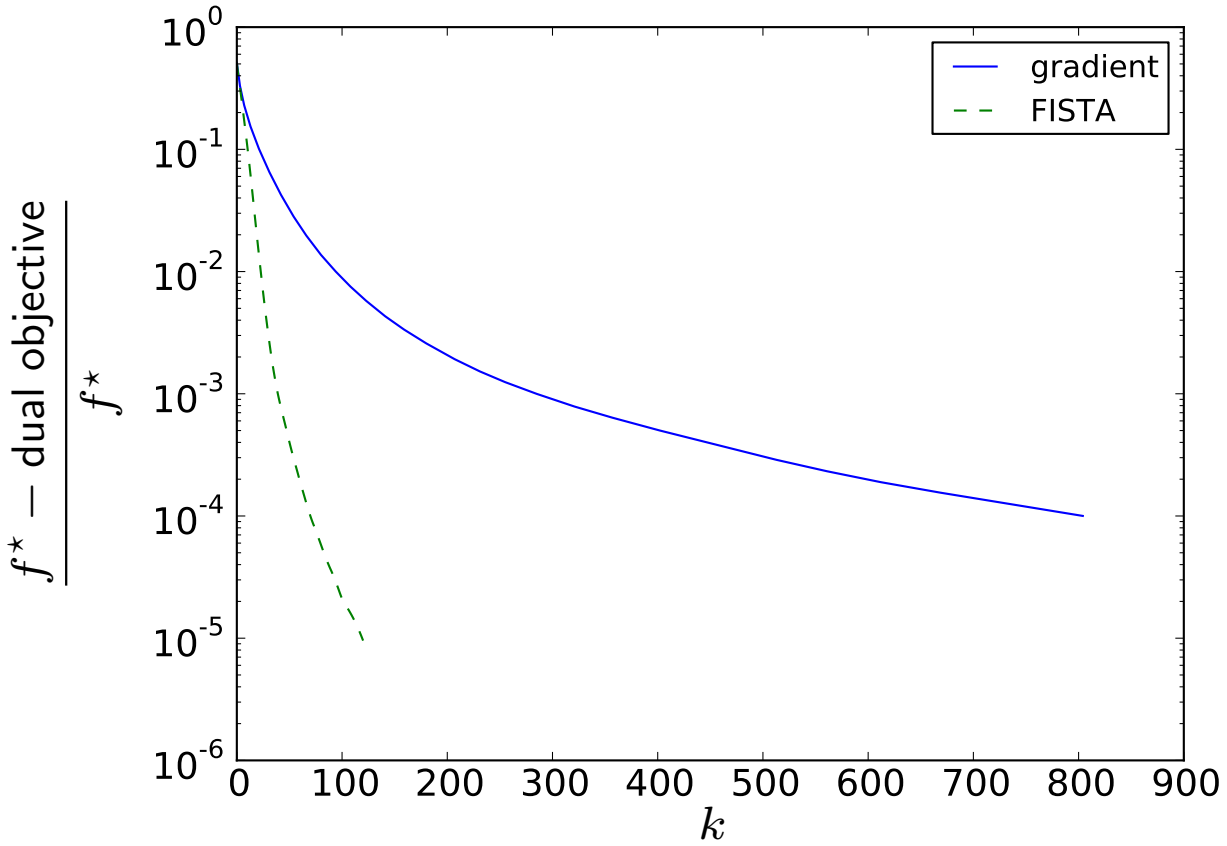
$$x^{k+1} = \operatorname*{argmin}_x \left\{ f(x) + (\sum_{i=1}^{p} B_i^{\mathrm{T}} z_i)^{\mathrm{T}} x \right\},$$
$$z_i^{k+1} = \mathcal{P}_{C_i}(z_i^k + t B_i x^{k+1}), i = 1, 2, \cdots, p.$$

数值实验

上页问题中，令

$$f(x) = \frac{1}{2}||Cx - d||_2^2$$

随机生成$C \in \mathbb{R}^{2000\times 1000}$, $B_i \in \mathbb{R}^{10\times 1000}$, $p = 500$



在凸集交上的极小化

假设f是强凸函数，集合C_i为闭凸集，且易于计算投影，考虑

$$\begin{aligned} \min \quad & f(x), \\ \text{s.t.} \quad & x \in C_1 \cap C_2 \cap \cdots \cap C_m, \end{aligned}$$

我们有$h(y_1, y_2, \cdots, y_m) = \sum_{i=1}^{m} I_{C_i}(y_i)$，$A = \begin{bmatrix} I & I & \cdots & I \end{bmatrix}^{\mathrm{T}}$，对偶问题为

$$\max_{z_i \in C_i} \quad -f^*\left(-\sum_{i=1}^{m} z_i\right) - \sum_{i=1}^{m} I_{C_i}^*(z_i),$$

$I_{C_i}^*(z_i)$是集合C_i的支撑函数，其显式表达式不易求出．因此我们利用Moreau 分解将迭代格式写成交替极小化方法的形式：

$$\begin{aligned} x^{k+1} &= \operatorname*{argmin}_x \left\{ f(x) + \left(\sum_{i=1}^{m} z_i\right)^{\mathrm{T}} x \right\}, \\ y_i^{k+1} &= \mathcal{P}_{C_i}\left(\frac{z_i^k}{t} + x^{k+1}\right), \quad i = 1, 2, \cdots, m, \\ z_i^{k+1} &= z_i^k + t(x^{k+1} - y_i^{k+1}), \quad i = 1, 2, \cdots, m. \end{aligned}$$

更一般情况：可分问题的拆分

假设f_i是强凸函数，h_i^*有易于计算的邻近算子．考虑

$$\min \quad \sum_{j=1}^{n} f_j(x_j) + \sum_{i=1}^{m} h_i(A_{i1}x_1 + A_{i2}x_2 + \cdots + A_{in}x_N),$$

其对偶问题形式如下：

$$\max \quad -\sum_{i=1}^{m} h_i^*(z_i) - \sum_{j=1}^{n} f_j^*(-A_{1j}^{\mathrm{T}}z_1 - A_{2j}^{\mathrm{T}}z_2 - \cdots - A_{mj}^{\mathrm{T}}z_m).$$

对偶近似点梯度法更新如下：

$$x_j^{k+1} = \operatorname*{argmin}_{x_j} \left\{ f_j(x_j) + (\sum_{i=1}^{m} A_{ij}z_i^k)^{\mathrm{T}}x_j \right\}, \quad j = 1, 2, \cdots, n,$$

$$z_i^{k+1} = \mathrm{prox}_{th_i^*}\left(z_i + t\sum_{j=1}^{n} A_{ij}x_j^{k+1} \right), \quad i = 1, 2, \cdots, m.$$

提纲

鞍点问题

令f, h是适当的闭凸函数. 考虑原始问题：

$$\min \quad f(x) + h(Ax),$$

- 由于h有自共轭性，我们将问题变形为

$$(\mathrm{L_{PD}}) \quad \min_x \max_z \quad \psi_{PD}(x,z) \overset{\text{def}}{=\!=} f(x) - h^*(z) + z^{\mathrm{T}}Ax. \tag{1}$$

  可以看到此时问题变成了一个极小– 极大问题，即关于变量x求极小，关于变量z求极大，这是一个典型的鞍点问题.

- 另一种常用的鞍点问题定义方式构造拉格朗日函数. 问题

$$\min_{x\in\mathbb{R}^n, y\in\mathbb{R}^m} \quad f(x) + h(y), \quad \text{s.t.} \quad y = Ax.$$

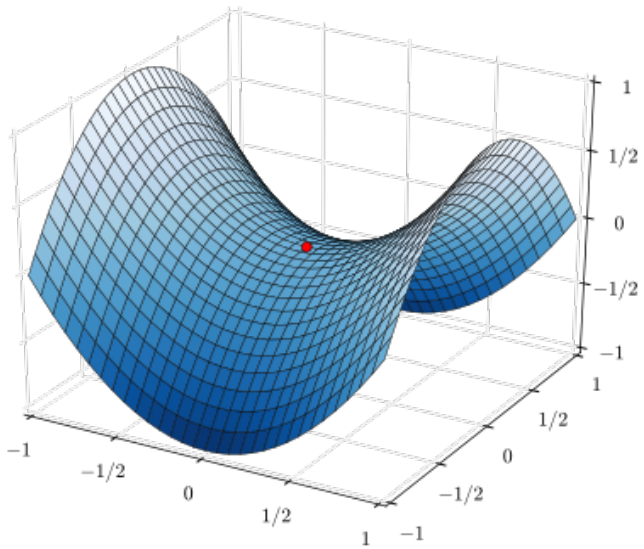
  相应的鞍点问题形式如下：

$$(\mathrm{L_P}) \quad \min_{x,y} \max_z \quad f(x) + h(y) + z^{\mathrm{T}}(Ax - y). \tag{2}$$

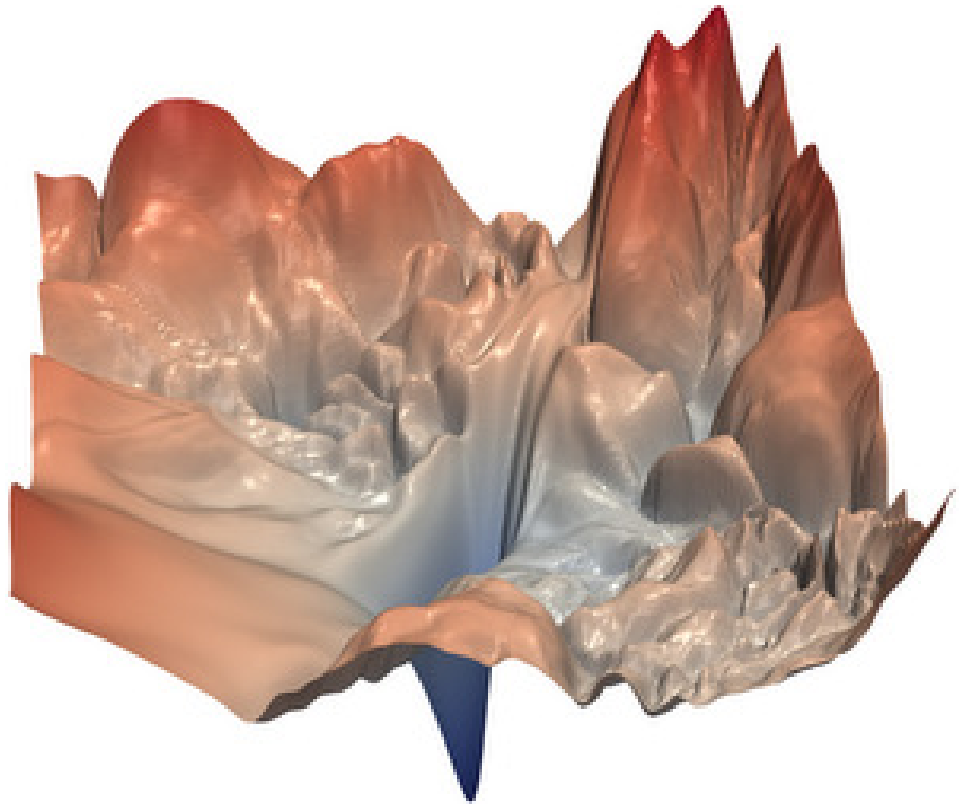
回顾：鞍点和局部极小

- 如果海瑟矩阵$\nabla^2 L(x,y,z)$既有正特征值又有负特征值，我们称稳定点(x,y,z)为一个鞍点.
- 对于凸问题，若强对偶成立，那么求解f的最小值等价于求解L的鞍点。
- 求解鞍点问题也称为$\min-\max$问题，是困难的一类问题。

非凸问题的鞍点问题比较复杂（右图所示，即使原问题非凸，也会带来非常复杂的情况），我们仅考虑$f(x), h(x)$均为凸问题的情况。



(a) $x^2 - y^2$的鞍点图示



(b) （可能不对？）神经网络ResNet56的损失函数嵌入到低维图示

《三体：死神永生》

(c) AD 1453。君士坦丁堡和奥斯曼帝国战争

(d) 女巫狄奥伦娜得到四维时空碎片

(e) 取出了密封在圣索菲亚大教堂地基深处的圣杯，把圣杯换成了一串新鲜的葡萄

Figure: generated by DALL·E

原始对偶混合梯度（PDHG）算法

- PDHG 算法的思想就是分别对两类变量应用近似点梯度算法.
- 以求解问题(1) 为例，PDHG 算法交替更新对偶变量以及原始变量，其迭代格式如下：

$$
\begin{aligned}
z^{k+1} &= \operatorname*{argmax}_{z} \left\{ -h^*(z) + \left\langle Ax^k, z - z^k \right\rangle - \frac{1}{2\delta_k} \|z - z^k\|_2^2 \right\} \\
&= \operatorname{prox}_{\delta_k h^*}(z^k + \delta_k A x^k), \\
x^{k+1} &= \operatorname*{argmin}_{x} \left\{ f(x) + (z^{k+1})^{\mathrm{T}} A(x - x^k) + \frac{1}{2\alpha_k} \|x - x^k\|_2^2 \right\} \\
&= \operatorname{prox}_{\alpha_k f}(x^k - \alpha_k A^{\mathrm{T}} z^{k+1}),
\end{aligned}
$$

其中α_k, δ_k分别为原始变量和对偶变量的更新步长.

- 它在第一步固定原始变量x^k针对对偶变量做(近似点)梯度上升，在第二步固定更新后的对偶变量z^{k+1}针对原始变量做(近似点)梯度下降. 在这里注意，原始变量和对偶变量的更新顺序是无关紧要的，若先更新原始变量，其等价于在另一初值下先更新对偶变量.

Chambolle-Pock算法

- PDHG 算法的收敛性需要比较强的条件，有些情形下未必收敛.
- Chambolle-Pock算法与PDHG 算法的区别在于多了一个外推步
- 具体的迭代格式如下：

$$
\begin{aligned}
z^{k+1} &= \operatorname{prox}_{\delta_k h^*}(z^k + \delta_k A y^k), \\
x^{k+1} &= \operatorname{prox}_{\alpha_k f}(x^k - \alpha_k A^{\mathrm{T}} z^{k+1}), \\
y^{k+1} &= 2x^{k+1} - x^k = x^{k+1} + \theta_k (x^{k+1} - x^k).
\end{aligned}
$$

提纲

LASSO问题求解

考虑LASSO问题

$$\min_{x\in\mathbb{R}^n} \quad \psi(x) \overset{\text{def}}{=\!=} \mu\|x\|_1 + \frac{1}{2}\|Ax-b\|_2^2.$$

取$f(x)=\mu\|x\|_1$ 和$h(x)=\frac{1}{2}\|x-b\|_2^2$，相应的鞍点问题：

$$\min_{x\in\mathbb{R}^n}\max_{z\in\mathbb{R}^m} \quad f(x)-h^*(z)+z^{\mathrm{T}}Ax.$$

根据共轭函数的定义，

$$h^*(z)=\sup_{y\in\mathbb{R}^m}\quad \left\{y^{\mathrm{T}}z-\frac{1}{2}\|y-b\|_2^2\right\}=\frac{1}{2}\|z\|_2^2+b^{\mathrm{T}}z.$$

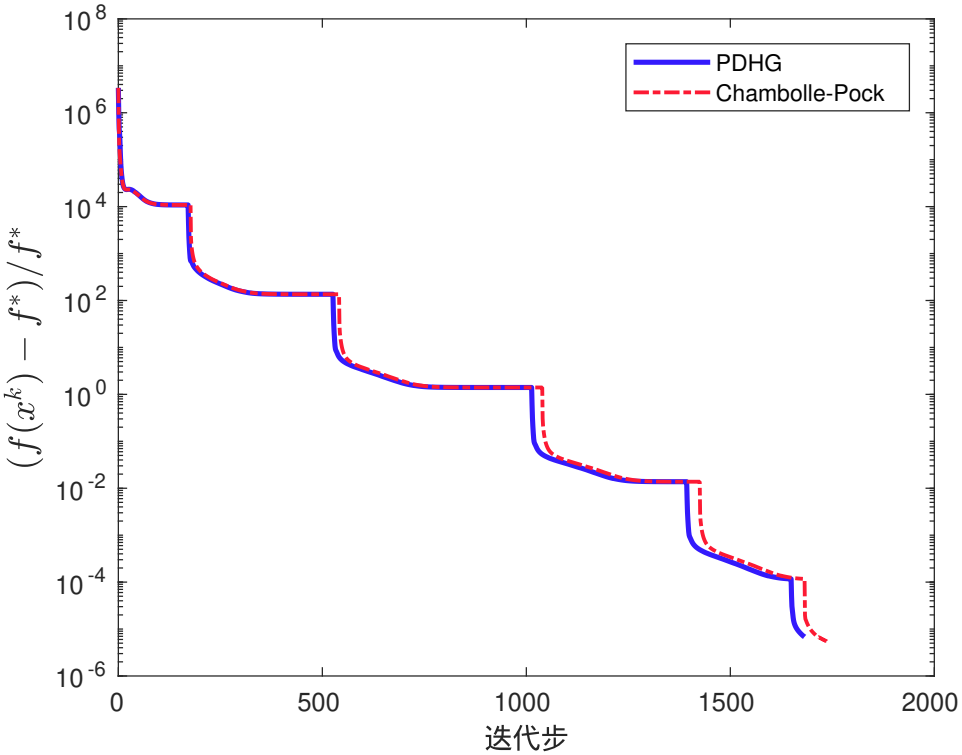
应用PDHG算法，x^{k+1} 和z^{k+1} 的更新格式分别为

$$\begin{aligned} z^{k+1} &= \mathrm{prox}_{\delta_k h^*}(z^k+\delta_k Ax^k)=\frac{1}{\delta_k+1}\left(z^k+\delta_k Ax^k-\delta_k b\right),\\ x^{k+1} &= \mathrm{prox}_{\alpha_k\mu\|\cdot\|_1}(x^k-\alpha_k A^{\mathrm{T}}z^{k+1}). \end{aligned}$$

这里δ_k,α_k 为步长.

Chambolle-Pock算法格式为

$$
\begin{aligned}
z^{k+1} &= \frac{1}{\delta_k + 1}\left(z^k + \delta_k A y^k - \delta_k b\right), \\
x^{k+1} &= \mathrm{prox}_{\alpha_k \mu \|\cdot\|_1}(x^k - \alpha_k A^{\mathrm{T}} z^{k+1}), \\
y^{k+1} &= 2x^{k+1} - x^k.
\end{aligned}
$$



TV-L^1模型

考虑去噪情形下的TV-L^1模型（即$\mathcal{A}$为矩阵空间的恒等算子）：

$$\min_{U\in\mathbb{R}^{n\times n}} \quad \|U\|_{TV} + \lambda\|U-B\|_1,$$

其中$\|U\|_{TV}$为全变差，即可以用离散的梯度（线性）算子$D:\mathbb{R}^{n\times n}\to\mathbb{R}^{n\times n\times 2}$表示为

$$\|U\|_{TV} = \sum_{1\le i,j\le n} \|(DU)_{ij}\|_2.$$

对任意的$W,V\in\mathbb{R}^{n\times n\times 2}$，记

$$\|W\| = \sum_{1\le i,j\le n} \|w_{ij}\|_2, \quad \langle W,V\rangle = \sum_{1\le i,j\le n,1\le k\le 2} w_{i,j,k}v_{i,j,k},$$

其中$w_{ij}\in\mathbb{R}^2$且$\|\cdot\|$定义了$\mathbb{R}^{n\times n\times 2}$上的一种范数．利用$\|\cdot\|$的定义，有

$$\|U\|_{TV} = \|DU\|.$$

参考文献：Zhu, Mingqiang, and Tony Chan. "An efficient primal-dual hybrid gradient algorithm for total variation image restoration." Ucla Cam Report 34.2 (2008).

TV-L^1模型

我们取D为相应的线性算子，并取

$$f(U) = \lambda\|U - B\|_1, U \in \mathbb{R}^{n\times n}, \quad h(W) = \|W\|, \quad W \in \mathbb{R}^{n\times n\times 2}.$$

相应的鞍点问题(1)如下：

$$(\mathrm{L_{PD}}) \quad \min_{U\in\mathbb{R}^{n\times n}} \max_{V\in\mathbb{R}^{n\times n\times 2}} f(U) - h^*(V) + \langle V, DU\rangle .$$

根据共轭函数的定义，

$$h^*(V) = \sup_{U\in\mathbb{R}^{n\times n\times 2}} \{\langle U, V\rangle - \|U\|\} = \begin{cases} 0, & \max\limits_{i,j} \|v_{ij}\|_2 \le 1, \\ +\infty, & \text{其他}. \end{cases}$$

记$\mathcal{V} = \{V \in \mathbb{R}^{n\times n\times 2} : \max\limits_{ij} \|v_{ij}\|_2 \le 1\}$，其示性函数记为$I_{\mathcal{V}}(V)$，则问题$(\mathrm{L_{PD}})$可以整理为

$$\min_U \max_V \quad f(U) + \langle V, DU\rangle - I_{\mathcal{V}}(V).$$

应用PDHG算法，则V^{k+1}的更新为

$$V^{k+1} = \text{prox}_{sI_{\mathcal{V}}}(V^k + sDU^k) = \mathcal{P}_{\mathcal{V}}(V^k + sDU^k), \tag{3}$$

即$V^k + sDU^k$在$\mathcal{V}$上的投影，而U^{k+1}的更新如下：

$$\begin{aligned} U^{k+1} &= \text{prox}_{tf}(U^k + tGV^{k+1}) \\ &= \underset{U}{\text{argmin}} \left\{ \lambda \|U - B\|_1 + \left\langle V^{k+1}, DU \right\rangle + \frac{1}{2t}\|U - U^k\|_F^2 \right\} \end{aligned}$$

其中$G : \mathbb{R}^{n\times n\times 2} \to \mathbb{R}^{n\times n}$为离散的散度算子(负的伴随算子)，其满足

$$\langle V, DU\rangle = -\langle GV, U\rangle, \quad \forall\, U \in \mathbb{R}^{n\times n},\ V \in \mathbb{R}^{n\times n\times 2}.$$

若应用Chambolle-Pock算法，那么U^{k+1}的更新保持不变，仅需调整V^{k+1}的更新为$V^k + sD(2U^{k+1} - U^k)$在$\mathcal{V}$上的投影.

图像反卷积模型

考虑问题

$$\min_{U\in\mathbb{R}^{n\times n}} \quad \|U\|_{TV} + \frac{\lambda}{2}\|\mathcal{A}U - B\|_F^2,$$

其中$\mathcal{A}U = K_{\mathcal{A}} * U$为卷积算子，且$K_{\mathcal{A}}$是$\mathcal{A}$的卷积核对应的矩阵. 由$\mathcal{A}$的特殊性，我们不需要对$f(U)$做任何变换。

类似于TV-L^1模型中的分析，取D为相应的线性算子，并取

$$f(U) = \frac{\lambda}{2}\|\mathcal{A}U - B\|_F^2,\ U \in \mathbb{R}^{n\times n},\ h(W) = \|W\|,\ W \in \mathbb{R}^{n\times n\times 2}.$$

类似地，一般的鞍点问题叙述如下：

$$(\mathrm{L_{PD}}) \quad \min_U \max_V \qquad f(U) + \langle V, DU\rangle - I_{\mathcal{V}}(V),$$

其中$\mathcal{V}$与TV-L^1模型中的定义一致.

应用PDHG算法，则V^{k+1}的更新仍为(3) 式，而U^{k+1}的更新为：

$$\begin{aligned} U^{k+1} &= \operatorname{prox}_{tf}(U^k + tGV^{k+1}) \\ &= \operatorname*{argmin}_U \left\{ \frac{\lambda}{2}\|\mathcal{A}U - B\|_F^2 + \frac{1}{2t}\|U - (U^k + tGV^{k+1})\|_F^2 \right\}, \end{aligned}$$

其中G为离散的散度算子. 可知U^{k+1}满足如下方程：

$$\lambda \mathcal{A}^*(\mathcal{A}U^{k+1} - B) + \frac{1}{t}(U^{k+1} - (U^k + tGV^{k+1})) = 0,$$

其中$\mathcal{A}^*$是$\mathcal{A}$的共轭算子，且其卷积核对应的矩阵为$K_{\mathcal{A}^*}$. 由于$\mathcal{A}U = K_{\mathcal{A}} * U$具有卷积的形式，我们可以利用快速傅里叶变换$\mathcal{F}$和其逆变换$\mathcal{F}^{-1}$来快速求解上面的线性方程组.

根据

$$\mathcal{F}(\mathcal{A}U) = \mathcal{F}(K_{\mathcal{A}} * U) = \mathcal{F}(K_{\mathcal{A}}) \odot \mathcal{F}(U),$$

其中$\odot$表示逐分量相乘, 我们有

$$\mathcal{F}(K_{\mathcal{A}^*}) \odot \Big(\mathcal{F}(K_{\mathcal{A}}) \odot \mathcal{F}(U^{k+1}) - \mathcal{F}(B)\Big) + \frac{1}{t\lambda}\mathcal{F}(U^{k+1} - (U^k + tGV^{k+1})) = 0.$$

利用关系式$\mathcal{F}(K_{\mathcal{A}^*}) = \overline{\mathcal{F}(K_{\mathcal{A}})}$，可得$U^{k+1}$ 的显式表达式

$$U^{k+1} = \mathcal{F}^{-1}\left(\frac{\mathcal{F}(U^k + tGV^{k+1}) + t\lambda\mathcal{F}(B) \odot \overline{\mathcal{F}(K_{\mathcal{A}})}}{1 + t\lambda|\mathcal{F}(K_{\mathcal{A}})|^2}\right),$$

以上表达式中除$\mathcal{F}, \mathcal{F}^{-1}, G$外，其余均为逐分量的运算

提纲

Chambolle-Pock 算法的收敛性

- 设X, Z分别为变量x, z的取值空间，若点$(\hat{x}, \hat{z})$满足

$$\psi_{\mathrm{PD}}(x, \hat{z}) \geq \psi_{\mathrm{PD}}(\hat{x}, \hat{z}) \geq \psi_{\mathrm{PD}}(\hat{x}, z), \quad \forall\, x \in X, z \in Z,$$

  称$(\hat{x}, \hat{z})$是问题(1)的一个鞍点，其中ψ_{PD}的定义见(2).

- 对任意子集$B_1 \times B_2 \subset X \times Z$，定义部分原始– 对偶间隙为

$$\mathcal{G}_{B_1 \times B_2}(x, z) = \max_{z' \in B_2} \psi_{\mathrm{PD}}(x, z') - \min_{x' \in B_1} \psi_{\mathrm{PD}}(x', z).$$

不难验证，只要鞍点$(\hat{x}, \hat{z}) \in B_1 \times B_2$，就有

$$\begin{aligned} \mathcal{G}_{B_1 \times B_2}(x, z) &\geq \psi_{\mathrm{PD}}(x, \hat{z}) - \psi_{\mathrm{PD}}(\hat{x}, z) \\ &= (\psi_{\mathrm{PD}}(x, \hat{z}) - \psi_{\mathrm{PD}}(\hat{x}, \hat{z})) + (\psi_{\mathrm{PD}}(\hat{x}, \hat{z}) - \psi_{\mathrm{PD}}(\hat{x}, z)) \geq 0, \end{aligned}$$

并且在鞍点处$\mathcal{G}_{B_1 \times B_2}(\hat{x}, \hat{z}) = 0$. 此外，容易验证当点$(\hat{x}, \hat{z}) \in \mathrm{int}(B_1 \times B_2)$且满足$\mathcal{G}_{B_1 \times B_2}(\hat{x}, \hat{z}) = 0$ 时，$(\hat{x}, \hat{z})$是一个鞍点.

Chambolle-Pock 算法的收敛性

设f, h为闭凸函数，原问题存在鞍点$(\hat{x}, \hat{z})$. 在Chambolle-Pock迭代格式中取步长$\alpha_k = t, \delta_k = s$，且满足$st < \frac{1}{L}$（$L = \|A\|_2^2$），则序列$\{(x^k, z^k)\}$具有：

(a) 令常数$C \leq (1 - Lst)^{-1}$. $\forall k$，(x^k, z^k)有界，且满足

$$\frac{\|x^k - \hat{x}\|^2}{2t} + \frac{\|z^k - \hat{z}\|^2}{2s} \leq C\left(\frac{\|x^0 - \hat{x}\|^2}{2t} + \frac{\|z^0 - \hat{z}\|^2}{2s}\right),$$

(b) 记$\bar{x}_N = \frac{1}{N}\sum_{k=1}^{N} x^k$，$\bar{z}_N = \frac{1}{N}\sum_{k=1}^{N} z^k$，则对$B_1 \times B_2 \subset X \times Z$，有

$$\mathcal{G}_{B_1 \times B_2}(\bar{x}_N, \bar{z}_N) \leq \frac{D(B_1, B_2)}{N}, \tag{4}$$

其中$D(B_1, B_2) = \sup_{(x,z)\in B_1 \times B_2}\left\{\frac{\|x - x^0\|^2}{2t} + \frac{\|z - z^0\|^2}{2s}\right\}$;

进一步地，序列$\{(\bar{x}_N, \bar{z}_N)\}_{N=1}^{\infty}$的聚点为问题(1)的一个鞍点；

(c) 存在问题(1)一个鞍点(x^*, z^*)使得$x^k \to x^*$, $z^k \to z^*$.

收敛性分析

为了方便推导，首先考虑算法的一般格式：

$$\begin{aligned} z^{k+1} =& \operatorname{prox}_{sh^*}(z^k + sA\bar{x}), \\ x^{k+1} =& \operatorname{prox}_{tf}(x^k - tA^{\mathrm{T}}\bar{z}). \end{aligned}$$

这里和Chambolle-Pock算法不同的是，我们使用$\bar{x}, \bar{z}$来表示更新x, z时的参考点．当它们取特定值时，以上格式可以为PDHG 算法或Chambolle-Pock 算法．根据邻近算子的性质，

$$\begin{aligned} -A^{\mathrm{T}}\bar{z} + \frac{x^k - x^{k+1}}{t} &\in \partial f(x^{k+1}), \\ A\bar{x} + \frac{z^k - z^{k+1}}{s} &\in \partial h^*(z^{k+1}). \end{aligned}$$

根据次梯度的定义，对于任意的$(x, z) \in X \times Z$有

$$\begin{aligned} f(x) \geq& f(x^{k+1}) + \frac{1}{t}(x - x^{k+1})^{\mathrm{T}}(x^k - x^{k+1}) - (x - x^{k+1})^{\mathrm{T}}A^{\mathrm{T}}\bar{z}, \\ h^*(z) \geq& h^*(z^{k+1}) + \frac{1}{s}(z - z^{k+1})^{\mathrm{T}}(z^k - z^{k+1}) + (z - z^{k+1})^{\mathrm{T}}A\bar{x}. \end{aligned}$$

将上述两个不等式相加，并引入二次项可整理得到

$$\begin{aligned}
&\frac{\|x-x^k\|^2}{2t}+\frac{\|z-z^k\|^2}{2s}-\frac{\|x-x^{k+1}\|^2}{2t}-\frac{\|z-z^{k+1}\|^2}{2s}\\
\geq&\left[f(x^{k+1})-h^*(z)+(x^{k+1})^{\mathrm{T}}A^{\mathrm{T}}z\right]-\left[f(x)-h^*(z^{k+1})+x^{\mathrm{T}}A^{\mathrm{T}}z^{k+1}\right]\\
&+\frac{\|x^k-x^{k+1}\|^2}{2t}+\frac{\|z^k-z^{k+1}\|^2}{2s}\\
&+(x^{k+1}-\bar{x})^{\mathrm{T}}A^{\mathrm{T}}(z^{k+1}-z)-(x^{k+1}-x)^{\mathrm{T}}A^{\mathrm{T}}(z^{k+1}-\bar{z}).
\end{aligned}\tag{5}$$

将Chambolle-Pock格式代入(5)，即取$\bar{x}=2x^k-x^{k-1},\bar{z}=z^{k+1}$，那么

$$\begin{aligned}
&(x^{k+1}-\bar{x})^{\mathrm{T}}A^{\mathrm{T}}(z^{k+1}-z)-(x^{k+1}-x)^{\mathrm{T}}A^{\mathrm{T}}(z^{k+1}-\bar{z})\\
=&(x^{k+1}-x^k-(x^k-x^{k-1}))^{\mathrm{T}}A^{\mathrm{T}}(z^{k+1}-z)\\
=&(x^{k+1}-x^k)^{\mathrm{T}}A^{\mathrm{T}}(z^{k+1}-z)-(x^k-x^{k-1})^{\mathrm{T}}A^{\mathrm{T}}(z^k-z)\\
&-(x^k-x^{k-1})^{\mathrm{T}}A^{\mathrm{T}}(z^{k+1}-z^k)\\
\geq&(x^{k+1}-x^k)^{\mathrm{T}}A^{\mathrm{T}}(z^{k+1}-z)-(x^k-x^{k-1})^{\mathrm{T}}A^{\mathrm{T}}(z^k-z)\\
&-\sqrt{L}\|x^k-x^{k-1}\|\|z^{k+1}-z^k\|,
\end{aligned}\tag{6}$$

应用柯西不等式即得到最后的不等号

又利用$2ab \leq \alpha a^2 + \frac{b^2}{\alpha}$对任意的$\alpha > 0$均成立，有

$$\begin{aligned}&\sqrt{L}\|x^k - x^{k-1}\|\|z^{k+1} - z^k\| \\ \leq &\frac{\sqrt{L}\alpha t}{2t}\|x^k - x^{k-1}\|^2 + \frac{\sqrt{L}s}{2\alpha s}\|z^{k+1} - z^k\|^2,\end{aligned}$$

取$\alpha = \sqrt{\frac{s}{t}}$，则

$$\sqrt{L}\alpha t = \sqrt{L}\frac{s}{\alpha} = \sqrt{Lst} < 1,$$

从而合并(5) 式和(6) 式得到，对于任意的$(x, z) \in X \times Z$，

$$\begin{aligned}&\frac{\|x - x^k\|^2}{2t} + \frac{\|z - z^k\|^2}{2s} - \frac{\|x - x^{k+1}\|^2}{2t} - \frac{\|z - z^{k+1}\|^2}{2s} \\ \geq &\left[f(x^{k+1}) - h^*(z) + (x^{k+1})^{\mathrm{T}}A^{\mathrm{T}}z\right] - \left[f(x) - h^*(z^{k+1}) + x^{\mathrm{T}}A^{\mathrm{T}}z^{k+1}\right] \\ &+ (1 - \sqrt{Lst})\frac{\|z^k - z^{k+1}\|^2}{2s} + \frac{\|x^k - x^{k+1}\|^2}{2t} - \sqrt{Lst}\frac{\|x^{k-1} - x^k\|^2}{2t} \\ &+ (x^{k+1} - x^k)^{\mathrm{T}}A^{\mathrm{T}}(z^{k+1} - z) - (x^k - x^{k-1})^{\mathrm{T}}A^{\mathrm{T}}(z^k - z).\end{aligned} \tag{7}$$

将上述不等式中的k从0遍历至$N-1$并求和，消掉不等式两边共同项后有

$$\begin{aligned}
&\sum_{k=1}^{N}\left\{\left[f(x^k)-h^*(z)+(x^k)^{\mathrm{T}}A^{\mathrm{T}}z\right]-\left[f(x)-h^*(z^k)+x^{\mathrm{T}}A^{\mathrm{T}}z^k\right]\right\}\\
&+\frac{\|x-x^N\|^2}{2t}+\frac{\|z-z^N\|^2}{2s}+(1-\sqrt{Lst})\sum_{k=1}^{N}\frac{\|z^k-z^{k-1}\|^2}{2s}\\
&+(1-\sqrt{Lst})\sum_{k=1}^{N-1}\frac{\|x^k-x^{k-1}\|^2}{2t}+\frac{\|x^N-x^{N-1}\|^2}{2t}\\
\le&\frac{\|x-x^0\|^2}{2t}+\frac{\|z-z^0\|^2}{2s}+(x^N-x^{N-1})^{\mathrm{T}}A^{\mathrm{T}}(z^N-z),
\end{aligned}\tag{8}$$

其中约定$x^{-1}=x^0$. 再一次应用柯西不等式，以及$2ab\le\alpha a^2+\frac{b^2}{\alpha}$对任意的$\alpha>0$均成立，可以得到

$$\begin{aligned}
(x^N-x^{N-1})^{\mathrm{T}}A^{\mathrm{T}}(z^N-z)&\le\|x^N-x^{N-1}\|(\sqrt{L}\|z^N-z\|)\\
&\le\frac{\|x^N-x^{N-1}\|^2}{2t}+\frac{Lst\|z-z^N\|^2}{2s}.
\end{aligned}$$

不等式(8)可进一步整理为

$$\begin{aligned}
&\sum_{k=1}^{N}\left\{\left[f(x^k)-h^*(z)+(x^k)^{\mathrm{T}}A^{\mathrm{T}}z\right]-\left[f(x)-h^*(z^k)+x^{\mathrm{T}}A^{\mathrm{T}}z^k\right]\right\}\\
&+\frac{\|x-x^N\|^2}{2t}+(1-Lst)\frac{\|z-z^N\|^2}{2s}+(1-\sqrt{Lst})\sum_{k=1}^{N}\frac{\|z^k-z^{k-1}\|^2}{2s}\\
&+(1-\sqrt{Lst})\sum_{k=1}^{N-1}\frac{\|x^k-x^{k-1}\|^2}{2t}\\
\le&\frac{\|x-x^0\|^2}{2t}+\frac{\|z-z^0\|^2}{2s}.
\end{aligned}\tag{9}$$

若取$(x,z)=(\hat{x},\hat{z})$，则由鞍点性质可知

$$[f(x^k)-h^*(\hat{z})+(x^k)^{\mathrm{T}}A^{\mathrm{T}}\hat{z}]-[f(\hat{x})-h^*(z^k)+\hat{x}^{\mathrm{T}}A^{\mathrm{T}}z^k]\ge 0.$$

进而(9)左边每一项都是正的，结论(a)成立.

收敛性分析

从(9)出发，利用f, h^*的凸性，以及$\bar{x}_N, \bar{z}_N$的定义，有

$$\begin{aligned}
&\left[f(\bar{x}_N) - h^*(z) + (\bar{x}_N)^{\mathrm{T}} A^{\mathrm{T}} z\right] - \left[f(x) - h^*(\bar{z}_N) + x^{\mathrm{T}} A^{\mathrm{T}} \bar{z}_N\right] \\
\leq & \frac{1}{N} \sum_{k=1}^{N} \left\{ \left[f(x^k) - h^*(z) + (x^k)^{\mathrm{T}} A^{\mathrm{T}} z\right] - \left[f(x) - h^*(z^k) + x^{\mathrm{T}} A^{\mathrm{T}} z^k\right] \right\} \\
\leq & \frac{1}{N} \left(\frac{\|x - x^0\|^2}{2t} + \frac{\|z - z^0\|^2}{2s} \right).
\end{aligned} \tag{10}$$

从而结论(b)中(4)式成立. 由(1) 知$\{(x^k, z^k)\}$是有界序列，因此其均值列$\{(\bar{x}_N, \bar{z}_N)\}$也为有界序列. 记$(x^\sharp, z^\sharp)$为序列$\{(\bar{x}_N, \bar{z}_N)\}$的聚点，利用$f, h^*$ 的凸性以及闭性（下半连续性），对(10) 式左右同时取下极限，可知对任意的$(x, z) \in X \times Z$，

$$\left[f(x^\sharp) - h^*(z) + (x^\sharp)^{\mathrm{T}} A^{\mathrm{T}} z\right] - \left[f(x) - h^*(z^\sharp) + x^{\mathrm{T}} A^{\mathrm{T}} z^\sharp\right] \leq 0.$$

从而$(x^\sharp, z^\sharp)$也是问题(1)的一个鞍点.

收敛性分析

为了证明$\{(x^k, z^k)\}$全序列收敛到问题(1)的鞍点，我们采用的大致思路为：先说明其子列收敛，然后再利用(7)式估计序列中其他点到子列极限点的误差（进而证明全序列收敛），最后说明该极限点是鞍点. 根据结论(1)，$\{(x^k, z^k)\}$是有界点列，因此存在子列$\{(x^{k_l}, z^{k_l})\}$收敛于(x^*, z^*). 在(7)式中令$(x, z) = (x^*, z^*)$，并将k从k_l取至$N-1, N > k_l$并求和，有

$$
\begin{aligned}
&\frac{\|x^* - x^N\|^2}{2t} + \frac{\|z^* - z^N\|^2}{2s} \\
&+ (1 - \sqrt{Lst}) \sum_{k=k_l+1}^{N} \frac{\|z^k - z^{k-1}\|^2}{2s} - \frac{\|x^{k_l} - x^{k_l-1}\|^2}{2t} \\
&+ (1 - \sqrt{Lst}) \sum_{k=k_l}^{N-1} \frac{\|x^k - x^{k-1}\|^2}{2t} + \frac{\|x^N - x^{N-1}\|^2}{2t} \\
&+ (x^N - x^{N-1})^{\mathrm{T}} A^{\mathrm{T}} (z^N - z^*) - (x^{k_l} - x^{k_l-1})^{\mathrm{T}} A^{\mathrm{T}} (z^{k_l} - z^*) \\
\leq &\frac{\|x^* - x^{k_l}\|^2}{2t} + \frac{\|z^* - z^{k_l}\|^2}{2s}.
\end{aligned}
$$

收敛性分析

去掉上式中不等式左边的求和项（正项），我们有如下估计：

$$\begin{aligned}
&\frac{\|x^* - x^N\|^2}{2t} + \frac{\|z^* - z^N\|^2}{2s} \\
\le &\frac{\|x^* - x^{k_l}\|^2}{2t} + \frac{\|z^* - z^{k_l}\|^2}{2s} + \frac{\|x^{k_l} - x^{k_l-1}\|^2}{2t} - \frac{\|x^N - x^{N-1}\|^2}{2t} \\
&+ (x^{k_l} - x^{k_l-1})^{\mathrm{T}} A^{\mathrm{T}} (z^{k_l} - z^*) - (x^N - x^{N-1})^{\mathrm{T}} A^{\mathrm{T}} (z^N - z^*).
\end{aligned}$$

注意到

$$\begin{aligned}
x^{k_l} &\to x^*, && (x^{k_l} \text{ 的定义}) \\
x^N - x^{N-1} &\to 0, && (\text{由(9) 式推出}) \\
\{z^k\} &\text{有界}, && (\text{本定理中(a) 的结论})
\end{aligned}$$

所以当$N \to \infty$时有，$x^N \to x^*$, $z^N \to z^*$，全序列收敛性得证．最后，由全序列收敛可知均值$(\bar{x}_N, \bar{z}_N)$也收敛到(x^*, z^*)，根据(a) 的结论和极限的唯一性立即得到$(x^\sharp, z^\sharp) = (x^*, z^*)$，即收敛到问题(1)的一个鞍点

ALM:增广拉格朗日函数法

陈士祥

中国科学技术大学

致谢：本教案部分参考北京大学文再文教授《最优化与建模》

提纲

二次罚函数法及困难

对于等式约束问题:

$$\begin{aligned} \min_{x} \quad & f(x) \\ \text{s.t.} \quad & c_i(x) = 0, \quad i \in \mathcal{E} \end{aligned} \tag{1}$$

传统的二次罚函数法需要求解最小化罚函数的子问题:

$$\min_{x} \quad P_E(x, \sigma) = f(x) + \frac{1}{2}\sigma \sum_{i \in \mathcal{E}} c_i^2(x).$$

由$c_i\left(x^{k+1}\right) \approx -\frac{\lambda_i^*}{\sigma_k}$, 为了满足可行性条件, 必须使罚因子$\sigma_k$趋于$\infty$, 这造成了子问题求解的数值困难.

我们接下来介绍的增广拉格朗日函数法可以利用有限的罚因子逼近最优解, 从而避免了上述必须使罚因子迅速膨胀的数值困难.

提纲

等式约束问题的增广拉格朗日函数法

增广拉格朗日函数法的每步都需要构造增广拉格朗日函数. 根据不同的约束, 增广拉格朗日函数的形式也不同, 因此我们分别论述.

定义

等式约束问题的增广拉格朗日函数

对于等式约束问题(1), 定义增广拉格朗日函数为:

$$L_\sigma(x,\lambda) = f(x) + \sum_{i\in\mathcal{E}} \lambda_i c_i(x) + \frac{1}{2}\sigma \sum_{i\in\mathcal{E}} c_i^2(x).$$

这即是在拉格朗日函数的基础上添加等式约束的二次罚函数.

由定义可得, 在第k步迭代, 给定罚因子σ_k和乘子λ^k, $L_{\sigma_k}\left(x,\lambda^k\right)$的最小值点$x^{k+1}$应满足梯度条件

$$\nabla_x L_{\sigma_k}\left(x^{k+1},\lambda^k\right) = \nabla f\left(x^{k+1}\right) + \sum_{i\in\mathcal{E}} \left(\lambda_i^k + \sigma_k c_i\left(x^{k+1}\right)\right) \nabla c_i\left(x^{k+1}\right) = 0. \tag{2}$$

我们将(2)式对比优化问题(1)满足的KKT条件(对最优解(x^*, λ^*)的梯度条件)

$$\nabla f\left(x^*\right)+\sum_{i \in \mathcal{E}} \lambda_i^* \nabla c_i\left(x^*\right)=0, \tag{3}$$

为保证(2)和(3)式在最优解处的一致性, 对充分大的k, 应满足:

$$\lambda_i^* \approx \lambda_i^k+\sigma_k c_i\left(x^{k+1}\right), \quad \forall i \in \mathcal{E}, \tag{4}$$

即等价于

$$c_i\left(x^{k+1}\right) \approx \frac{1}{\sigma_k}\left(\lambda_i^*-\lambda_i^k\right).$$

由此得出我们希望设计的增广拉格朗日算法具有如下的特性.

性质

- 增广拉格朗日函数法通过合理更新乘子, 即通过控制$\lambda_i^* - \lambda_i^k$降低约束违反度. 因为根据约束违反度满足的公式, 当λ_i^k足够接近λ_i^*时, $c_i(x^{k+1})$将远小于$1/\sigma_k$.
- (4)式的一个截断近似可以写为:

$$\lambda_i^{k+1} = \lambda_i^k + \sigma_k c_i\left(x^{k+1}\right), \quad \forall i \in \mathcal{E},$$

  这可以作为算法中乘子的更新方式.

根据如上讨论, 并对$c(x), \nabla c(x)$沿用罚函数法的定义, 我们将在下文写出等式约束问题增广拉格朗日函数法的具体算法.

使用增广拉格朗日函数法的实例

我们考虑优化问题

$$\begin{aligned} \min \quad & x+\sqrt{3}y, \\ \text{s.t.} \quad & x^2+y^2=1. \end{aligned}$$

容易求得最优解为$x^*=\left(-\frac{1}{2},-\frac{\sqrt{3}}{2}\right)^{\mathrm{T}}$, 相应的拉格朗日乘子$\lambda^*=1$.

根据增广拉格朗日函数的形式, 写出本问题的增广拉格朗日函数:

$$L_\sigma(x,y,\lambda)=x+\sqrt{3}y+\lambda\left(x^2+y^2-1\right)+\frac{\sigma}{2}\left(x^2+y^2-1\right)^2,$$

并在下图中绘制$L_2(x,y,0.9)$的等高线.

二次罚函数法与增广拉格朗日函数法求解的等高线

图中标“$*$”的点为原问题的最优解x^*
标“$\circ$”的点为罚函数或增广拉格朗日函数的最优解

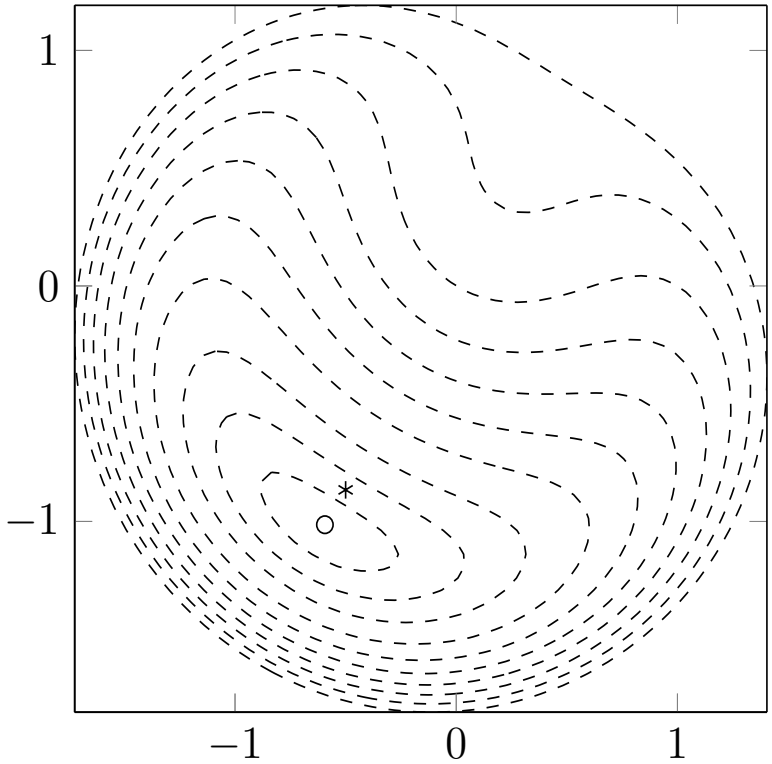


Figure: (a) 二次罚函数

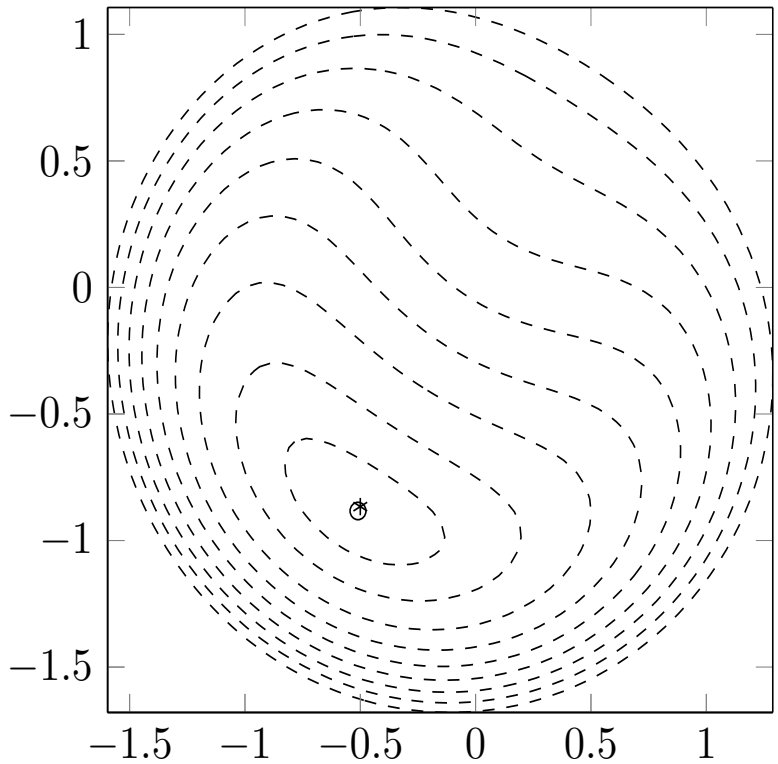


Figure: (b) 增广拉格朗日函数

比较与结论

我们比较二次罚函数和增广拉格朗日函数在最优解探寻方面的有效性.

- 二次罚函数法求出的最优解为$(-0.5957, -1.0319)$, 与最优解的欧氏距离约0.1915, 约束违反度为0.4197.
- 增广拉格朗日罚函数法求出的最优解为$(-0.5100, -0.8833)$, 与最优解的欧氏距离约0.02, 约束违反度为0.0403.

由此可见, 成立如下的经验性结论.

性质

增广拉格朗日函数法可具有比二次罚函数法更精确的寻优能力, 且约束违反度一般更低.

算法 1 增广拉格朗日函数法

Require: 初始坐标$x^0 \in \mathbb{R}^n$, 乘子λ^0, 罚因子$\sigma_0 > 0$, 约束违反度常数 $\varepsilon > 0$, 精度$\eta_k > 0$, 迭代步$k = 0$.

Ensure: x^{k+1}, λ^k.

检查初始元素.
for $k = 0, 1, 2, \cdots$ do **do**
  以x^k为初始点, 求解$\min_x \quad L_{\sigma_k}\left(x, \lambda^k\right)$, 得到满足需求的精度条件$\left\|\nabla_x L_{\sigma_k}\left(x, \lambda^k\right)\right\| \leqslant \eta_k$ 的解x^{k+1}.
  if $\left\|c\left(x^{k+1}\right)\right\| \leqslant \varepsilon$ **then**
    返回近似解$\left(x^{k+1}, \lambda_k\right)$, 终止迭代.
  end if
  更新乘子: $\lambda^{k+1} = \lambda^k + \sigma_k c\left(x^{k+1}\right)$.
  更新罚因子: $\sigma_{k+1} = \rho\sigma_k$.
end for

ρ与σ_k的取值指导

在每次迭代确定σ_k时, 应考虑如下的问题.

- σ_k不应增长过快:
  (1)随着罚因子σ_k的增大, 可见$L_{\sigma_k}(x, \lambda^k)$关于$x$的海瑟矩阵的条件数也将增大, 这将导致数值困难;

  (2)σ_k与σ_{k+1}接近时, x^k可以作为求解x^{k+1}的初始点, 以加快收敛.
- σ_k不应增长过慢: 算法整体的收敛速度将变慢(惩罚不足).

因此在实际中, 我们应该控制σ_k的增长维持在一个合理的速度区间内. 一个简单的方法是维持$\rho \in [2, 10]$, 不过近年来也有学者设计了更合理的自适应方法.

提纲

一般约束问题的增广拉格朗日函数法

一般的约束优化问题可以写成

$$\begin{aligned} \min \quad & f(x), \\ \text{s.t.} \quad & c_i(x) = 0, i \in \mathcal{E}, \\ & c_i(x) \leqslant 0, i \in \mathcal{I}. \end{aligned} \tag{5}$$

对于问题(5), 我们一般引入松弛变量, 得到如下等价形式:

$$\begin{aligned} \min_{x,s} \quad & f(x), \\ \text{s.t.} \quad & c_i(x) = 0, i \in \mathcal{E}, \\ & c_i(x) + s_i = 0, i \in \mathcal{I}, \\ & s_i \geqslant 0, i \in \mathcal{I}. \end{aligned} \tag{6}$$

这样的做法我们已经用过多次了, 应熟练掌握.

构造增广拉格朗日函数

保留非负约束, 可以构造拉格朗日函数

$$L(x,s,\lambda,\mu)=f(x)+\sum_{i\in\mathcal{E}}\lambda_i c_i(x)+\sum_{i\in\mathcal{I}}\mu_i\left(c_i(x)+s_i\right),s_i\geqslant 0,i\in\mathcal{I}.$$

记问题(6)中等式约束的二次罚函数为$p(x,s)$, 即

$$p(x,s)=\sum_{i\in\mathcal{E}}c_i^2(x)+\sum_{i\in\mathcal{I}}\left(c_i(x)+s_i\right)^2,$$

那么可以同样构造增广拉格朗日函数如下:

$$\begin{aligned}L_\sigma(x,s,\lambda,\mu)=&f(x)+\sum_{i\in\mathcal{E}}\lambda_i c_i(x)+\sum_{i\in\mathcal{I}}\mu_i\left(c_i(x)+s_i\right)+\frac{\sigma}{2}p(x,s),\\&s_i\geqslant 0,i\in\mathcal{I}.\end{aligned}$$

消元法求解子问题

在第 k 步迭代中, 给定乘子λ^k, μ^k 和罚因子σ_k, 需要求解如下问题:

$$\min_{x,s} \quad L_{\sigma_k}\left(x, s, \lambda^k, \mu^k\right), \quad \text{s.t.} \quad s \geqslant 0, \tag{7}$$

以得到 x^{k+1}, s^{k+1}.

我们现在介绍一种基于消元的方法, 即考虑消去 s, 求解只关于 x 的优化问题.

- 首先固定 x, 关于 s 的子问题化为

$$\min_{s \geqslant 0} \quad \sum_{i \in \mathcal{I}} \mu_i \left(c_i(x) + s_i\right) + \frac{\sigma_k}{2} \sum_{i \in \mathcal{I}} \left(c_i(x) + s_i\right)^2.$$

  容易直接解得使子问题最优且满足非负约束的s_i为

$$s_i = \max\left\{-\frac{\mu_i}{\sigma_k} - c_i(x), 0\right\}, \quad i \in \mathcal{I}. \tag{8}$$

- 将 s_i 的表达式代入 L_{σ_k} 我们有

$$L_{\sigma_k}\left(x,\lambda^k,\mu^k\right)=f(x)+\sum_{i\in\mathcal{E}}\lambda_i c_i(x)+\frac{\sigma_k}{2}\sum_{i\in\mathcal{E}}c_i^2(x)+$$
$$\frac{\sigma_k}{2}\sum_{i\in\mathcal{I}}\left(\max\left\{\frac{\mu_i}{\sigma_k}+c_i(x),0\right\}^2-\frac{\mu_i^2}{\sigma_k^2}\right).$$

其为关于x的连续可微函数(假设$f(x),c_i(x),i\in\mathcal{I}\cup\mathcal{E}$连续可微). 因此, 问题(7)等价于

$$\min_{x\in\mathbb{R}^n}\quad L_{\sigma_k}\left(x,\lambda^k,\mu^k\right).$$

并可以利用梯度法进行求解.

注意: 这里, 我们消去了变量s, 因此可以只考虑关于x的优化问题.

更新乘子

对于问题(6), 其最优解x^*, s^* 和乘子λ^*, μ^* 需满足KKT 条件:

$$\begin{aligned} &0 = \nabla f\left(x^*\right) + \sum_{i\in\mathcal{E}} \lambda_i^* \nabla c_i\left(x^*\right) + \sum_{i\in\mathcal{I}} \mu_i^* \nabla c_i\left(x^*\right), \\ &\mu_i^* \geqslant 0,\ s_i^* \geqslant 0, i \in \mathcal{I} \end{aligned}$$

问题(7)的最优解x^{k+1}, s^{k+1} 满足

$$\begin{aligned} 0 = \nabla f\left(x^{k+1}\right) + \sum_{i\in\mathcal{E}} \left(\lambda_i^k + \sigma_k c_i\left(x^{k+1}\right)\right) \nabla c_i\left(x^{k+1}\right) + \\ \sum_{i\in\mathcal{I}} \left(\mu_i^k + \sigma_k \left(c_i\left(x^{k+1}\right) + s_i^{k+1}\right)\right) \nabla c_i\left(x^{k+1}\right), \end{aligned}$$

$$s_i^{k+1} = \max\left\{-\frac{\mu_i^k}{\sigma_k} - c_i\left(x^{k+1}\right), 0\right\}, \quad i \in \mathcal{I}.$$

对比问题(6)和问题(7)的KKT 条件, 易知乘子的更新格式为

$$\begin{aligned} \lambda_i^{k+1} &= \lambda_i^k + \sigma_k c_i\left(x^{k+1}\right), \quad i \in \mathcal{E}, \\ \mu_i^{k+1} &= \max\left\{\mu_i^k + \sigma_k c_i\left(x^{k+1}\right), 0\right\}, \quad i \in \mathcal{I}. \end{aligned} \tag{9}$$

考虑凸优化问题(不等式形式)

$$\begin{aligned} \min_{x\in\mathbb{R}^n} \quad & f(x), \\ \text{s.t.} \quad & c_i(x) \leqslant 0, i = 1, 2, \cdots, m. \end{aligned} \tag{10}$$

根据定义, 写出问题(10)的增广拉格朗日函数:

$$L_\sigma(x,\lambda) = f(x) + \frac{\sigma}{2}\sum_{i=1}^{m}\left(\max\left\{\frac{\lambda_i}{\sigma} + c_i(x), 0\right\}^2 - \frac{\lambda_i^2}{\sigma^2}\right).$$

给定一列单调递增的乘子$\sigma_k \uparrow \sigma_\infty$, 以及初始乘子$\lambda^0$, 结合(9)式, 问题(10)的增广拉格朗日函数法为

$$\begin{cases} x^{k+1} \approx \arg\min\limits_{x\in\mathbb{R}^n} L_{\sigma_k}\left(x, \lambda^k\right), \\ \lambda^{k+1} = \max\left\{0, \lambda^k + \sigma_k c\left(x^{k+1}\right)\right\}. \end{cases} \tag{11}$$

不精确条件

定义$\phi_k(x) = L_{\sigma_k}\left(x, \lambda^k\right)$. 由于 $\phi_k(x)$ 的最小值点的显式表达式通常是未知的, 我们往往调用迭代算法求其一个近似解. 为保证收敛性, 我们要求该近似解至少满足不精确条件. 例如:

$$\phi_k\left(x^{k+1}\right) - \inf \phi_k \leqslant \frac{\varepsilon_k^2}{2\sigma_k}, \quad \varepsilon_k \geqslant 0, \quad \sum_{k=1}^{\infty} \varepsilon_k < +\infty. \tag{12}$$

由于 $\inf \phi_k$ 是未知的,直接验证(12) 式是数值上不可行的. 但是, 如果ϕ_k 是α -强凸函数, 则有

$$\phi_k(x) - \inf \phi_k \leqslant \frac{1}{2\alpha} \operatorname{dist}^2\left(0, \partial \phi_k(x)\right) \tag{13}$$

根据(13)式, 可以进一步构造如下数值可验证的不精确条件:

$$\operatorname{dist}\left(0, \partial \phi_k\left(x^{k+1}\right)\right) \leqslant \sqrt{\frac{\alpha}{\sigma_k}} \varepsilon_k, \quad \varepsilon_k \geqslant 0, \quad \sum_{k=1}^{\infty} \varepsilon_k < +\infty. \tag{14}$$

凸问题的增广拉格朗日函数法的收敛性

下面我们给出不精确条件下增广拉格朗日函数法的收敛性定理.

定理

凸问题的增广拉格朗日函数法的收敛性

假设$\{x^k\}, \{\lambda^k\}$为问题(10)通过(11)式生成的序列, x^{k+1}满足不精确条件(12). 如果问题(10)的*Slater*约束品性成立, 那么序列$\{\lambda^k\}$是有界序列且收敛到λ^* (λ^*为对偶问题的一个最优解).

如果存在一个γ, 使得下水平集$\{x \in \mathcal{X} \mid f(x) \leqslant \gamma\}$是非空有界的, 那么序列$\{x^k\}$也是有界的, 并且其所有的聚点都是问题(10)的最优解.

提纲

基追踪问题(BP)

考虑一类简单的基追踪问题. 设 $A \in \mathbb{R}^{m\times n}(m \leqslant n)$, $b \in \mathbb{R}^m$, $x \in \mathbb{R}^n$, 基追踪问题被描述为

$$\min_{x\in\mathbb{R}^n} \quad \|x\|_1, \qquad \text{s.t.} \qquad Ax = b. \tag{15}$$

考虑其对偶问题:

$$\min_{y\in\mathbb{R}^m} \quad b^{\mathrm{T}}y, \qquad \text{s.t.} \qquad \left\|A^{\mathrm{T}}y\right\|_\infty \leqslant 1. \tag{16}$$

通过引入变量s, 对偶问题可以等价地写成

$$\min_{y\in\mathbb{R}^m,\ s\in\mathbb{R}^n} b^{\mathrm{T}}y, \qquad \text{s.t.} \qquad A^{\mathrm{T}}y - s = 0,\ \|s\|_\infty \leqslant 1. \tag{17}$$

根据问题(15)的形式, 引入罚因子σ和乘子λ, 其增广拉格朗日函数为

$$L_\sigma(x,\lambda) = \|x\|_1 + \lambda^{\mathrm{T}}(Ax-b) + \frac{\sigma}{2}\|Ax-b\|_2^2. \tag{18}$$

固定σ, 第k步迭代更新格式为

$$\begin{cases} x^{k+1} = \arg\min\limits_{x\in\mathbb{R}^n} \left\{ \|x\|_1 + \frac{\sigma}{2}\left\|Ax-b+\frac{\lambda^k}{\sigma}\right\|_2^2 \right\}, \\ \lambda^{k+1} = \lambda^k + \sigma\left(Ax^{k+1}-b\right). \end{cases} \tag{19}$$

设迭代初始点$x^0=\lambda^0=0$, 考虑格式(19)中的第一步, 并假设 x^{k+1}为 $L_\sigma(x,\lambda^k)$的一个全局极小解, 则对$L_\sigma(x,\lambda^k)$利用极小性条件得

$$0 \in \partial\left\|x^{k+1}\right\|_1 + \sigma A^{\mathrm{T}}\left(Ax^{k+1}-b+\frac{\lambda^k}{\sigma}\right).$$

因此成立

$$-A^{\mathrm{T}}\lambda^{k+1} \in \partial\left\|x^{k+1}\right\|_1. \tag{20}$$

BP问题的实例与解

例　简单基追踪问题的增广拉格朗日函数解法

考虑标准基追踪问题, 其中A是512×1024规模的随机矩阵(每个元素从标准正态分布中抽样), b定义为

$$b = Au,$$

其中$u \in \mathbb{R}^{1024}$是服从正态分布随机稀疏向量, 设其稀疏度$r = 0.1$或0.2, 即分别约具有102/205个服从正态分布的非零分量.

进一步地, 我们固定罚因子σ, 采用近似点梯度法作为求解器, 不精确地求解关于x的子问题以得到x^{k+1}.

我们演示的算法中设置了求解精度$\eta_k = 10^{-k}$, 并使用**BB**步长作为线搜索的初始步长. 下图展示了算法产生的迭代点与最优点的欧式距离的变化, 以及它们约束违反度的变化趋势. 由图可知:

性质

对于本例中的标准基追踪问题, 固定的罚因子σ也可以使增广拉格朗日函数法收敛.

BP问题的实例与解

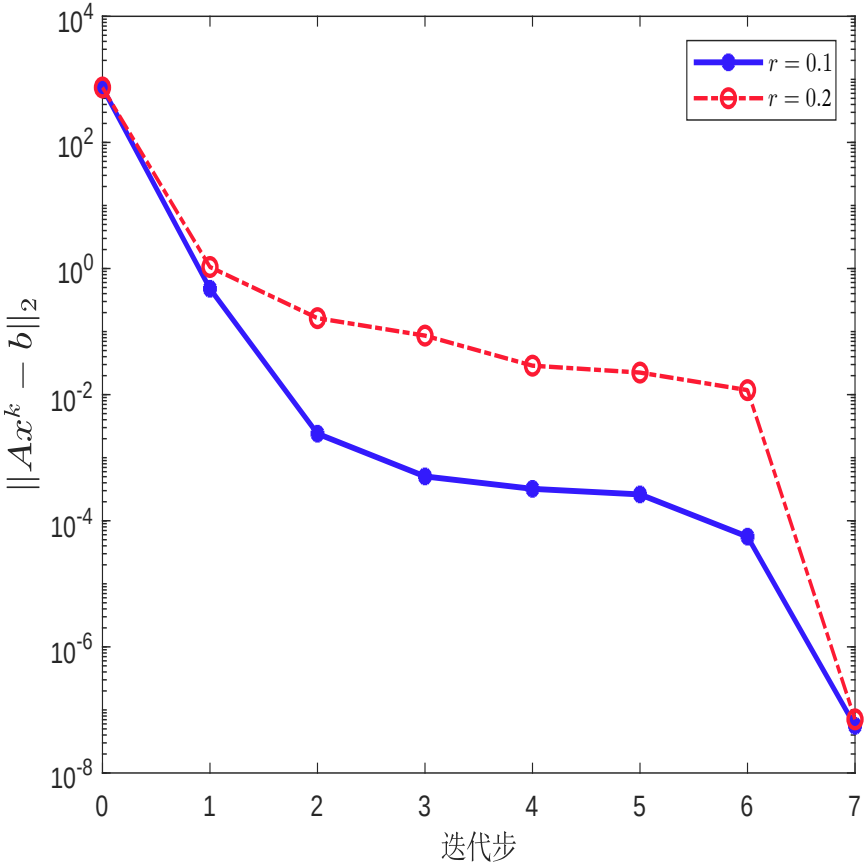


Figure: (a) 约束违反度

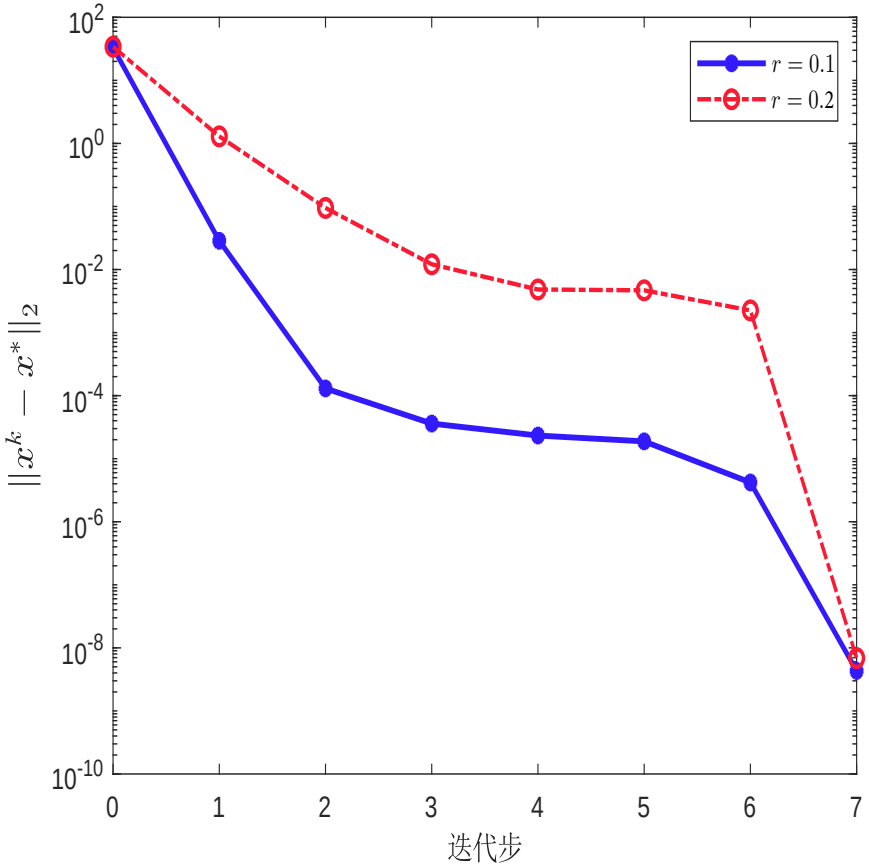


Figure: (b) 与最优点的距离

基追踪问题的收敛性定理

定理

简单基追踪问题的收敛性定理

假设问题(15)的可行域非空, 迭代序列$\{x^k\}, \{\lambda^k\}$ 是由迭代格式(19) 从初始点 $x^0 = \lambda^0 = 0$ 产生的, 则存在正整数 K 使得任意的$x^k, k \geqslant K$ 是问题(15)的解.

提纲

现在我们考虑另一重要的问题. 设对偶问题(17):

$$\min_{y\in\mathbb{R}^m, s\in\mathbb{R}^n} \quad b^{\mathrm{T}}y, \quad \text{s.t.} \quad A^{\mathrm{T}}y - s = 0, \quad \|s\|_\infty \leqslant 1.$$

引入拉格朗日乘子λ和罚因子σ, 作增广拉格朗日函数

$$L_\sigma(y, s, \lambda) = b^{\mathrm{T}}y + \lambda^{\mathrm{T}}\left(A^{\mathrm{T}}y - s\right) + \frac{\sigma}{2}\left\|A^{\mathrm{T}}y - s\right\|_2^2, \quad \|s\|_\infty \leqslant 1.$$

增广拉格朗日函数法的迭代格式为($\rho > 1$ 和 $\bar{\sigma} < +\infty$ 为算法参数):

$$\begin{cases} \left(y^{k+1}, s^{k+1}\right) = \arg\min\limits_{y,\|s\|_\infty\leqslant 1} \left\{ b^{\mathrm{T}}y + \dfrac{\sigma_k}{2}\left\|A^{\mathrm{T}}y - s + \dfrac{\lambda}{\sigma_k}\right\|_2^2 \right\}, \\ \lambda^{k+1} = \lambda^k + \sigma_k\left(A^{\mathrm{T}}y^{k+1} - s^{k+1}\right), \\ \sigma_{k+1} = \min\left\{\rho\sigma_k, \bar{\sigma}\right\}. \end{cases}$$

其中$\left(y^{k+1}, s^{k+1}\right)$的显式表达式未知, 需要迭代求解.

消元法求解子问题

除了利用投影梯度法求解关于 (y, s) 的联合最小化问题外, 还可以利用最优性条件将 s 用 y 来表示, 转而求解只关于 y 的最小化问题.

关于s的极小化问题为

$$\min_{s} \quad \frac{\sigma}{2}\left\|A^{\mathrm{T}}y - s + \frac{\lambda}{\sigma}\right\|_2^2, \qquad \mathbf{s.t.} \quad \|s\|_\infty \leqslant 1.$$

这是一个关于s的二次型函数, 因此问题的解为

$$s = \mathcal{P}_{\|s\|_\infty \leqslant 1}(A^{\mathrm{T}}y + \frac{\lambda}{\sigma}),$$

其中$\mathcal{P}_{\|s\|_\infty \leqslant 1}(z)$为集合$\{s|\ \|s\|_\infty \leqslant 1\}$的投影算子, 即

$$\mathcal{P}_{\|s\|_\infty \leqslant 1}(z) = \max\left\{\min\left\{z,\ 1\right\},\ -1\right\}.$$

消元法求解子问题

将上述s的表达式代入的增广拉格朗日函数法的迭代格式, 得

$$\begin{cases} y^{k+1} = \arg\min\limits_{y} \left\{ b^{\mathrm{T}} y + \frac{\sigma}{2} \left\| \psi\left(A^{\mathrm{T}} y + \frac{\lambda}{\sigma} \right) \right\|_2^2 \right\}, \\ \lambda^{k+1} = \sigma_k \psi\left(A^{\mathrm{T}} y^{k+1} + \frac{\lambda^k}{\sigma_k} \right), \\ \sigma_{k+1} = \min\left\{ \rho\sigma_k, \bar{\sigma} \right\}. \end{cases} \tag{21}$$

其中$\psi(x) = \mathrm{sign}(x)\max\{|x| - 1, 0\}$.

我们不能得到关于 y^{k+1} 的显式表达式. 但是由于 $L_{\sigma_k}\left(y, \lambda^k\right)$ 关于 y 连续可微, 故可以利用梯度法求解.

提纲

一般线性规划问题的增广拉格朗日函数法

我们介绍基追踪问题及其对偶问题的增广拉格朗日函数法不仅仅是为了向读者展示该方法在一类问题中的应用, 更有实际意义的是, 它展示了一类处理一般线性规划问题的基本思路.

一般的线性规划问题可描述为

$$\min_x \quad c^{\mathrm{T}}x, \qquad \mathbf{s.t.} \quad Ax = b, \qquad x \geqslant 0.$$

其对偶问题为

$$\max_y \quad b^{\mathrm{T}}y, \qquad \mathbf{s.t.} \quad A^{\mathrm{T}}y + s = c, \qquad s \geqslant 0.$$

以上问题中$A \in \mathbb{R}^{m\times n}, x \in \mathbb{R}^n, s \in \mathbb{R}^n, y \in \mathbb{R}^m$.
首先还是先写出对偶问题的增广拉格朗日函数

$$L_{\sigma_k}(y, s, \lambda) = -b^{\mathrm{T}}y + \left(\lambda^k\right)^{\mathrm{T}}\left(A^{\mathrm{T}}y + s - c\right) + \frac{\sigma_k}{2}\left\|A^{\mathrm{T}}y + s - c\right\|_2^2, \quad s \geqslant 0.$$

因此根据对偶问题的增广拉格朗日函数, 设计迭代算法为

$$\begin{cases} \left(y^{k+1}, s^{k+1}\right) = \arg\min\limits_{y\in\mathbb{R}^n, s\geqslant 0} L_{\sigma_k}\left(y, s, \lambda^k\right) \\ \qquad\qquad = \arg\min\limits_{y\in\mathbb{R}^n, s\geqslant 0} \left\{-b^{\mathrm{T}}y + \dfrac{\sigma_k}{2}\left\|A^{\mathrm{T}}y + s - c + \dfrac{\lambda^k}{\sigma_k}\right\|_2^2\right\}, \\ \lambda^{k+1} = \lambda^k + \sigma_k\left(A^{\mathrm{T}}y^{k+1} + s^{k+1} - c\right), \\ \sigma_{k+1} = \min\left\{\rho\sigma_k, \bar{\sigma}\right\}. \end{cases} \tag{22}$$

并且还是用最优性条件将s用y表示(消元法), 得到上述关于s的二次型函数中使得s极小化的解为

$$s = \operatorname{argmin}_{s\geqslant 0}\left\|A^{\mathrm{T}}y + s - c + \frac{\lambda}{\sigma}\right\|_2^2 = \mathcal{P}_{\mathbb{R}^n_+}\left(c - A^{\mathrm{T}}y - \frac{\lambda}{\sigma}\right).$$

将上述s解的表达式代入上述迭代算法, 可得约减式

$$\begin{cases} y^{k+1} = \underset{y\in\mathbb{R}^n}{\operatorname{argmin}} \left\{ -b^{\mathrm{T}}y + \frac{\sigma_k}{2} \left\| \psi\left(y, \lambda^k, \sigma_k\right) \right\|_2^2 \right\}, \\ \lambda^{k+1} = \sigma_k \psi\left(y^{k+1}, \lambda^k, \sigma_k\right), \\ \sigma_{k+1} = \min\left\{\rho\sigma_k, \bar{\sigma}\right\}. \end{cases}$$

其中$\psi\left(y, \lambda^k, \sigma_k\right) = \mathcal{P}_{\mathbb{R}^n_+}\left(A^{\mathrm{T}}y + \frac{\lambda^k}{\sigma_k} - c\right)$.

这样, 我们仍利用梯度法或半光滑牛顿法即可求解上述的迭代算法, 以最终解决一般线性规划问题.

提纲

半定规划问题

在线性规划问题后, 我们考虑半定规划问题:

$$\begin{aligned} \min_{X\in\mathcal{S}^n} \quad & \langle C, X\rangle, \\ \text{s.t.} \quad & \langle A_i, X\rangle = b_i, i = 1, 2, \cdots, m, \\ & X \succeq 0. \end{aligned} \tag{23}$$

引入乘子$\lambda \in \mathbb{R}^m$, 罚因子σ, 记$\mathcal{A}(X) = \left(\langle A_1, X\rangle, \langle A_2, X\rangle, \cdots, \langle A_m, X\rangle\right)^{\mathrm{T}}$, 则增广拉格朗日函数为

$$L_\sigma(X, \lambda) = \langle C, X\rangle - \lambda^{\mathrm{T}}(\mathcal{A}(X) - b) + \frac{\sigma}{2}\|\mathcal{A}(X) - b\|_2^2, \quad X \succeq 0.$$

根据增广拉格朗日函数, 写出算法的迭代格式为

$$\begin{cases} X^{k+1} \approx \underset{X\in\mathcal{S}_+^n}{\operatorname{argmin}} L_{\sigma_k}\left(X, \lambda^k\right), \\ \lambda^{k+1} = \lambda^k - \sigma_k\left(\mathcal{A}\left(X^{k+1}\right) - b\right), \\ \sigma_{k+1} = \min\left\{\rho\sigma_k, \bar{\sigma}\right\}. \end{cases}$$

当迭代收敛时, X^k 和λ^k 分别收敛到原问题和对偶问题的解.

考虑问题(23)的对偶问题:

$$\begin{aligned} \min_{y\in\mathbb{R}^m} \quad & -b^{\mathrm{T}}y, \\ \text{s.t.} \quad & \sum_{i=1}^{m} y_i A_i \preceq C. \end{aligned} \tag{24}$$

引入松他变量$S \succeq 0$, 乘子$\Lambda \in \mathcal{S}^n$以及罚因子$\sigma$, 则增广拉格朗日函数为

$$L_\sigma(y,S,\Lambda) = -b^{\mathrm{T}}y + \left\langle \Lambda, \sum_{i=1}^{m} y_i A_i + S - C \right\rangle + \frac{\sigma}{2}\left\| \sum_{i=1}^{m} y_i A_i + S - C \right\|_F^2.$$

根据上述函数, 增广拉格朗日函数法的迭代格式为

$$\left(y^{k+1}, S^{k+1}\right) \approx \arg\min_{y\in\mathbb{R}^m} L_{\sigma_k}\left(y, S, \Lambda^k\right),$$

$$\Lambda^{k+1} = \Lambda^k + \sum_{i=1}^{m} y_i^{k+1} A_i + S^{k+1} - C,$$

$$\sigma_{k+1} = \min\left\{\rho\sigma_k, \bar{\sigma}\right\}.$$

类似地, 我们利用消元法, 由最优性条件消去S, 得

$$L_\sigma(y,\Lambda)=-b^{\mathrm{T}}y+\frac{\sigma}{2}\left(\left\|\mathcal{P}_{\mathcal{S}_+^n}\left(\sum_{i=1}^m y_iA_i-C+\frac{\Lambda}{\sigma}\right)\right\|_F^2-\frac{\|\Lambda\|_F^2}{\sigma^2}\right).$$

其中$\mathcal{P}_{\mathcal{S}_+^n}$ 为到半定锥集$\mathcal{S}_+^n$ 的投影算子.

因此迭代算法的格式变为

$$\begin{aligned}
y^{k+1}&\approx\arg\min_{y\in\mathbb{R}^m}L_{\sigma_k}\left(y,\Lambda^k\right),\\
\Lambda^{k+1}&=\sigma\mathcal{P}_{\mathcal{S}_+^n}\left(\sum_{i=1}^m y_i^{k+1}A_i-C+\frac{\Lambda^k}{\sigma_k}\right),\\
\sigma_{k+1}&=\min\left\{\rho\sigma_k,\bar{\sigma}\right\}.
\end{aligned}$$

基于原问题和对偶问题的算法比较

我们对比半定规划及其对偶问题的增广拉格朗日函数法.

- 半定规划问题的迭代格式中X^{k+1}是在半正定锥$\mathcal{S}^n_+$中求解, 还是约束优化;
- 对偶问题中y^{k+1}在$\mathbb{R}^m$中求解, 对应于可微的无约束优化问题.

注: 容易验证$L_{\sigma_k}(y,\Lambda^k)$关于$y$连续可微, 因此可用梯度法求$y^{k+1}$.
因此, 若m较小, 我们推荐考虑对偶问题的增广拉格朗日函数法, 以求解半定规划问题.

更进一步, 我们指出, 实际上$L_{\sigma_k}(y,\Lambda^k)$关于$y$还是强半光滑的, 因此我们考虑对偶问题时还可以用半光滑牛顿法更快速地求解半定规划问题.

提纲

考虑具有如下形式的优化问题：

$$\min_{x\in\mathbb{R}^n} \quad f(x)+h(Ax) \tag{25}$$

其中 f,h 为适当的闭凸函数, $A\in\mathbb{R}^{m\times n}$

回顾Lecture 14, 优化问题(25)的一些常见例子如下：

- 当 h 是单点集 $\{b\}$ 的示性函数时，等价于线性等式约束优化问题

$$\min_{x\in\mathbb{R}^n} \quad f(x), \qquad \text{s.t. } Ax=b$$

- 当h 是凸集C 上的示性函数时，等价于凸集约束问题

$$\min \quad f(x) \qquad \text{s.t. } Ax\in C$$

- 当$h(y)=\|y-b\|$ 时，等价于正则优化问题

$$\min \quad f(x)+\|Ax-b\|$$

对偶问题

写出上述优化问题的拉格朗日函数：

$$L(x,y,z) = f(x) + h(y) + z^T(Ax - y)$$

则对偶问题为：

$$\max \quad \psi(z) = \inf_{x,y} L(x,y,z) = -f^*(-A^T z) - h^*(z) \tag{26}$$

我们有以下结论：

使用近似点算法求解对偶问题(26) $\Longleftrightarrow$ 对原问题(25)用增广拉格朗日函数法

对偶函数的邻近算子

对于对偶问题(26)，近似点算法迭代格式如下

$$z^{k+1} = \text{prox}_{t\psi}\left(z^k\right) = \arg\min_z \left\{ f^*\left(-A^{\mathrm{T}} z\right) + h^*(z) + \frac{1}{2t_k}\left\| z - z^k \right\|_2^2 \right\}$$

事实上也可以写成：$z^{k+1} = \text{prox}_{t\psi}(z^k) = z^k + t^k(A\hat{x}^k - \hat{y})$, 其中

$$(\hat{x}, \hat{y}) = \underset{x,y}{\text{argmin}} \left(f(x) + h(y) + z^T(Ax - y) + \frac{t}{2}||Ax - y||_2^2 \right) \tag{27}$$

也就是说，$\hat{x}, \hat{y}$ 最小化增广拉格朗日函数，近似点算法迭代格式对应增广拉格朗日函数法中的乘子更新。

等价性证明

首先改写(27)为:

$$\begin{aligned} \min_{x,y,w} \quad & f(x)+h(y)+\frac{t}{2}||w||_2^2 \\ \text{s.t.} \quad & Ax-y+z/t=w \end{aligned}$$

对约束$Ax-y+\frac{z}{t}=w$ 引入乘子 u, 由最优性条件有:

$$A\hat{x}-\hat{y}+\frac{z}{t}=w, \quad -A^{\mathrm{T}}u\in\partial f(\hat{x}), \quad u\in\partial h(\hat{y}), \quad tw=u$$

消去 w 得 $u=z+t(Ax-y)$. 又根据共轭函数性质（作业2.5）得:

$$\hat{x}\in\partial f^*\left(-A^{\mathrm{T}}u\right), \quad \hat{y}\in\partial h^*(u)$$

代入等式$Ax-y+\frac{z}{t}=w$中，可得$0\in -A\partial f^*(-A^Tu)+\partial h^*(u)+\frac{1}{t}(u-z)$, 这正是 $u=\mathrm{prox}_{t\psi}(z)$ 的最优性条件.

另一方面，若有$u=\mathrm{prox}_{t\psi}(z)$, 则选取 $\hat{x}\in\partial f^*\left(-A^{\mathrm{T}}u\right)$ 及 $\hat{y}\in\partial h^*(u)$, 即可恢复出增广拉格朗日函数法中的变量

总结：增广拉格朗日函数法

选择初始点 $z^{(0)}$ 并迭代以下步骤:

1. 最小化增广拉格朗日函数

$$(\hat{x},\hat{y}) = \underset{x,y}{\operatorname{argmin}} \left(f(x) + h(y) + \frac{t_k}{2} ||Ax - y + \frac{1}{t_k} z^k||_2^2 \right)$$

2. 乘子更新

$$z^{k+1} = z^k + t_k(A\hat{x} - \hat{y})$$

- 这等价于对对偶问题使用近似点算法
- (对偶问题)可以使用加速版本的近似点算法来加快收敛速度
- 通常第一步先求解一个较不精确的最优值

参考文献

- BERTSEKAS D P. Constrained optimization and Lagrange multiplier methods[M]. Salt Lake: Academic press, 2014.
- ROCKAFELLAR R T. Augmented Lagrangians and applications of the proximal point algorithm in convex programming[J]. Mathematics of Operations Research, 1976, 1(2): 97-116.
- YANG L, SUN D, TOH K C. SDPNAL+: a majorized semismooth Newton-CG augmented Lagrangian method for semidefinite programming with nonnegative constraints[J]. Mathematical Programming Computation, 2015, 7(3): 331-366.
- ZHAO X Y, SUN D, TOH K C. A Newton-CG augmented Lagrangian method for semidefinite programming[J]. SIAM Journal on Optimization, 2010, 20(4): 1737-1765.

参考文献

- YIN W, OSHER S, GOLDFARB D, et al. Bregman iterative algorithms for 1-minimization with applications to compressed sensing[J]. SIAM Journal on Imaging Sciences, 2008, 1(1): 143-168.
- LI X, SUN D, TOH K C. An asymptotically superlinearly convergent semismooth Newton augmented Lagrangian method for linear programming[J]. ArXiv:1903.09546, 2019.
- LI X, SUN D, TOH K C. A highly efficient semismooth Newton augmented Lagrangian method for solving Lasso problems[J]. SIAM Journal on Optimization, 2018, 28(1): 433-458.

ADMM:交替方向乘子法

陈士祥

中国科学技术大学

致谢：本教案部分参考北京大学文再文教授《最优化与建模》

提纲

典型问题形式

考虑如下凸问题：

$$\begin{aligned} \min_{x_1,x_2} \quad & f_1(x_1)+f_2(x_2), \\ \text{s.t.} \quad & A_1x_1+A_2x_2=b, \end{aligned} \tag{1}$$

- f_1, f_2是适当的闭凸函数，但不要求是光滑的，$x_1 \in \mathbb{R}^n, x_2 \in \mathbb{R}^m$, $A_1 \in \mathbb{R}^{p\times n}, A_2 \in \mathbb{R}^{p\times m}, b \in \mathbb{R}^p$.
- 问题特点：目标函数可以分成彼此分离的两块，但是变量被线性约束结合在一起. 常见的一些无约束和带约束的优化问题都可以表示成这一形式。

问题形式举例

- 可以分成两块的无约束优化问题

$$\min_{x} \quad f_1(x) + f_2(x).$$

引入一个新的变量z并令$x = z$，将问题转化为

$$\begin{aligned} \min_{x,z} \quad & f_1(x) + f_2(z), \\ \text{s.t.} \quad & x - z = 0. \end{aligned}$$

- 带线性变换的无约束优化问题

$$\min_{x} \quad f_1(x) + f_2(Ax).$$

可以引入一个新的变量z，令$z = Ax$，则问题变为

$$\begin{aligned} \min_{x,z} \quad & f_1(x) + f_2(z), \\ \text{s.t.} \quad & Ax - z = 0. \end{aligned}$$

问题形式举例

- 凸集$C \subset \mathbb{R}^n$上的约束优化问题

$$\begin{aligned} \min_{x} \quad & f(x), \\ \text{s.t.} \quad & Ax \in C, \end{aligned}$$

$I_C(z)$是集合C的示性函数，引入约束$z = Ax$，那么问题转化为

$$\begin{aligned} \min_{x,z} \quad & f(x) + I_C(z), \\ \text{s.t.} \quad & Ax - z = 0. \end{aligned}$$

- 全局一致性问题

$$\min_{x} \quad \sum_{i=1}^{N} \phi_i(x).$$

令$x = z$，并将x复制N份，分别为x_i，那么问题转化为

$$\begin{aligned} \min_{x_i,z} \quad & \sum_{i=1}^{N} \phi_i(x_i), \\ \text{s.t.} \quad & x_i - z = 0,\ i = 1, 2, \cdots, N. \end{aligned}$$

- 首先写出问题(1)的增广拉格朗日函数

$$\begin{aligned} L_\rho(x_1, x_2, y) =& f_1(x_1) + f_2(x_2) + y^{\mathrm{T}}(A_1x_1 + A_2x_2 - b) \\ &+ \frac{\rho}{2}\|A_1x_1 + A_2x_2 - b\|_2^2, \end{aligned} \tag{2}$$

其中$\rho > 0$是二次罚项的系数.

- 常见的求解带约束问题的增广拉格朗日函数法为如下更新:

$$(x_1^{k+1}, x_2^{k+1}) = \operatorname*{argmin}_{x_1, x_2} L_\rho(x_1, x_2, y^k), \tag{3}$$

$$y^{k+1} = y^k + \tau\rho(A_1x_1^{k+1} + A_2x_2^{k+1} - b), \tag{4}$$

其中τ为步长.

Alternating direction method of multipliers, ADMM

- 交替方向乘子法的基本思路: 第一步迭代(3)同时对x_1和x_2进行优化有时候比较困难，而固定一个变量求解关于另一个变量的极小问题可能比较简单，因此我们可以考虑对x_1和x_2交替求极小
- 其迭代格式可以总结如下：

$$x_1^{k+1} = \operatorname*{argmin}_{x_1} L_\rho(x_1, x_2^k, y^k), \tag{5}$$

$$x_2^{k+1} = \operatorname*{argmin}_{x_2} L_\rho(x_1^{k+1}, x_2, y^k), \tag{6}$$

$$y^{k+1} = y^k + \tau\rho(A_1x_1^{k+1} + A_2x_2^{k+1} - b), \tag{7}$$

其中τ为步长，通常取值于$\left(0, \frac{1+\sqrt{5}}{2}\right]$

原问题最优性条件

- 因为f_1, f_2均为闭凸函数，约束为线性约束，所以当Slater 条件成立时，可以使用凸优化问题的KKT 条件来作为交替方向乘子法的收敛准则. 问题(1)的拉格朗日函数为

$$L(x_1, x_2, y) = f_1(x_1) + f_2(x_2) + y^{\mathrm{T}}(A_1x_1 + A_2x_2 - b).$$

- 根据最优性条件定理，若x_1^*, x_2^*为问题(1)的最优解，y^*为对应的拉格朗日乘子，则以下条件满足：

$$0 \in \partial_{x_1} L(x_1^*, x_2^*, y^*) = \partial f_1(x_1^*) + A_1^{\mathrm{T}} y^*, \tag{8a}$$

$$0 \in \partial_{x_2} L(x_1^*, x_2^*, y^*) = \partial f_2(x_2^*) + A_2^{\mathrm{T}} y^*, \tag{8b}$$

$$A_1x_1^* + A_2x_2^* = b. \tag{8c}$$

在这里条件(8c)又称为原始可行性条件，条件(8a)和条件(8b)又称为对偶可行性条件.

ADMM单步迭代最优性条件

- 由x_2的更新步骤

$$x_2^k = \underset{x}{\operatorname{argmin}} \left\{ f_2(x) + \frac{\rho}{2} \left\| A_1 x_1^k + A_2 x - b + \frac{y^{k-1}}{\rho} \right\|^2 \right\},$$

根据最优性条件不难推出

$$0 \in \partial f_2(x_2^k) + A_2^{\mathrm{T}}[y^{k-1} + \rho(A_1 x_1^k + A_2 x_2^k - b)]. \tag{9}$$

当$\tau = 1$时，根据(7)可知上式方括号中的表达式就是y^k，最终有

$$0 \in \partial f_2(x_2^k) + A_2^{\mathrm{T}} y^k,$$

- 由x_1的更新公式

$$x_1^k = \underset{x}{\operatorname{argmin}} \left\{ f_1(x) + \frac{\rho}{2} \| A_1 x + A_2 x_2^{k-1} - b + \frac{y^{k-1}}{\rho} \|^2 \right\},$$

假设子问题能精确求解，根据最优性条件

$$0 \in \partial f_1(x_1^k) + A_1^{\mathrm{T}}[\rho(A_1 x_1^k + A_2 x_2^{k-1} - b) + y^{k-1}].$$

ADMM单步迭代最优性条件

- 根据ADMM 的第三式(7)取$\tau = 1$有

$$0 \in \partial f_1(x_1^k) + A_1^{\mathrm{T}}(y^k + \rho A_2(x_2^{k-1} - x_2^k)). \tag{10}$$

对比条件(8a)可知多出来的项为$A_1^{\mathrm{T}}A_2(x_2^{k-1} - x_2^k)$。因此要检测对偶可行性只需要检测残差

$$s^k = A_1^{\mathrm{T}}A_2(x_2^{k-1} - x_2^k)$$

- 综上当x_2更新取到精确解且$\tau = 1$时，判断ADMM 是否收敛只需要检测前述两个残差r^k, s^k是否充分小：

$$\begin{aligned} 0 \approx \|r^k\| &= \|A_1x_1^k + A_2x_2^k - b\| \quad \text{（原始可行性）}, \\ 0 \approx \|s^k\| &= \|A_1^{\mathrm{T}}A_2(x_2^{k-1} - x_2^k)\| \quad \text{（对偶可行性）}. \end{aligned} \tag{11}$$

提纲

线性化

- 线性化技巧使用近似点项对子问题目标函数进行二次近似.
- 不失一般性，我们考虑第一个子问题，即

$$\min_{x_1} \quad f_1(x_1) + \frac{\rho}{2}\|A_1x_1 - v^k\|^2, \tag{12}$$

其中$v^k = b - A_2x_2^k - \frac{1}{\rho}y^k$.

- 当子问题目标函数可微时，线性化将问题(12)变为

$$x_1^{k+1} = \operatorname*{argmin}_{x_1} \left\{ \left(\nabla f_1(x_1^k) + \rho A_1^{\mathrm{T}}\left(A_1x_1^k - v^k\right)\right)^{\mathrm{T}} x_1 + \frac{1}{2\eta_k}\|x_1 - x^k\|_2^2 \right\},$$

其中η_k是步长参数，这等价于做一步梯度下降.

- 当目标函数不可微时，可以考虑只将二次项线性化，即

$$x_1^{k+1} = \operatorname*{argmin}_{x_1} \left\{ f_1(x_1) + \rho\left(A_1^{\mathrm{T}}(A_1x_1^k - v^k)\right)^{\mathrm{T}} x_1 + \frac{1}{2\eta_k}\|x_1 - x^k\|_2^2 \right\},$$

这等价于做一步近似点梯度步.

缓存分解

- 如果目标函数中含二次函数，例如$f_1(x_1) = \frac{1}{2}\|Cx_1 - d\|_2^2$，那么针对$x_1$的更新(5)等价于求解线性方程组
$$(C^{\mathrm{T}}C + \rho A_1^{\mathrm{T}}A_1)x_1 = C^{\mathrm{T}}d + \rho A_1^{\mathrm{T}}v^k.$$
- 虽然子问题有显式解，但是每步求解的复杂度仍然比较高，这时候可以考虑用缓存分解的方法. 首先对$C^{\mathrm{T}}C + \rho A_1^{\mathrm{T}}A_1$进行Cholesky 分解并缓存分解的结果，在每步迭代中只需要求解简单的三角形方程组
- 当ρ发生更新时，就要重新进行分解. 特别地，当$C^{\mathrm{T}}C + \rho A_1^{\mathrm{T}}A_1$一部分容易求逆，另一部分是低秩的情形时，可以用SMW公式来求逆.

- 有时候为了方便求解子问题，可以用一个性质好的矩阵D近似二次项$A_1^{\mathrm{T}}A_1$, 此时子问题(12)替换为

$$\begin{aligned} x_1^{k+1} \quad = \quad & \operatorname*{argmin}_{x_1} \left\{ f_1(x_1) + \frac{\rho}{2}\|A_1x_1 - v^k\|_2^2 \right. \\ & \left. + \frac{\rho}{2}(x_1 - x^k)^{\mathrm{T}}(D - A_1^{\mathrm{T}}A_1)(x_1 - x^k) \right\}. \end{aligned}$$

  这种方法也称为优化转移.

- 通过选取合适的D，当计算$\operatorname*{argmin}_{x_1} \left\{ f_1(x_1) + \frac{\rho}{2}x_1^{\mathrm{T}}Dx_1 \right\}$明显比计算$\operatorname*{argmin}_{x_1}\{f_1(x_1) + \frac{\rho}{2}x_1^{\mathrm{T}}A_1^{\mathrm{T}}A_1x_1\}$要容易时，优化转移可以极大地简化子问题的计算.

  特别地，当$D = \frac{\eta_k}{\rho}I$时，优化转移等价于做单步的近似点梯度步.

二次罚项系数的动态调节

- 原始可行性和对偶可行性分别用$\|r^k\|$和$\|s^k\|$度量.
- 求解过程中二次罚项系数ρ太大会导致原始可行性$\|r^k\|$下降很快，但是对偶可行性$\|s^k\|$下降很慢；二次罚项系数太小，则会有相反的效果. 这样都会导致收敛比较慢或得到的解的可行性很差.
- 一个自然的想法是在每次迭代时动态调节惩罚系数ρ的大小，从而使得原始可行性和对偶可行性能够以比较一致的速度下降到零. 一个简单有效的方式是令

$$\rho^{k+1} = \begin{cases} \gamma_p \rho^k, & \|r^k\| > \mu \|s^k\|, \\ \rho^k / \gamma_d & \|s^k\| > \mu \|r^k\|, \\ \rho^k, & \text{其他}, \end{cases}$$

其中$\mu > 1, \gamma_p > 1, \gamma_d > 1$是参数, 常见的选择为$\mu = 10, \gamma_p = \gamma_d = 2$. 在迭代过程中将原始可行性$\|r^k\|$和对偶可行性$\|s^k\|$保持在彼此的$\mu$倍内. 如果发现$\|r^k\|$或$\|s^k\|$下降过慢就应该相应增大或减小二次罚项系数$\rho^k$.

- 考虑有多块变量的情形

$$\begin{aligned} \min_{x_1,x_2,\cdots,x_N} \quad & f_1(x_1)+f_2(x_2)+\cdots+f_N(x_N), \\ \text{s.t.} \quad & A_1x_1+A_2x_2+\cdots+A_Nx_N=b. \end{aligned} \tag{13}$$

这里$f_i(x_i)$是闭凸函数，$x_i\in\mathbb{R}^{n_i}, A_i\in\mathbb{R}^{m\times n_i}$.

- 同样写出增广拉格朗日函数$L_\rho(x_1,x_2,\cdots,x_N,y)$，相应的多块ADMM 迭代格式为

$$\begin{aligned} x_1^{k+1} &= \operatorname*{argmin}_x L_\rho(x,x_2^k,\cdots,x_N^k,y^k), \\ x_2^{k+1} &= \operatorname*{argmin}_x L_\rho(x_1^{k+1},x,\cdots,x_N^k,y^k), \\ &\cdots\cdots\cdots\cdots \\ x_N^{k+1} &= \operatorname*{argmin}_x L_\rho(x_1^{k+1},x_2^{k+1},\cdots,x,y^k), \\ y^{k+1} &= y^k+\tau\rho(A_1x_1^{k+1}+A_2x_2^{k+1}+\cdots+A_Nx_N^{k+1}-b), \end{aligned}$$

其中$\tau\in(0,(\sqrt{5}+1)/2)$为步长参数.

提纲

LASSO 问题的Primal 形式

- LASSO 问题

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

转换为标准问题形式：

$$\begin{aligned} \min_{x,z} \quad & \frac{1}{2}\|Ax-b\|^2 + \mu\|z\|_1, \\ \text{s.t.} \quad & x = z. \end{aligned}$$

- 交替方向乘子法迭代格式为

$$\begin{aligned} x^{k+1} &= \operatorname*{argmin}_x \left\{ \frac{1}{2}\|Ax-b\|^2 + \frac{\rho}{2}\|x - z^k + y^k/\rho\|_2^2 \right\}, \\ &= (A^{\mathrm{T}}A + \rho I)^{-1}(A^{\mathrm{T}}b + \rho z^k - y^k), \\ z^{k+1} &= \operatorname*{argmin}_z \left\{ \mu\|z\|_1 + \frac{\rho}{2}\|x^{k+1} - z + y^k/\rho\|^2 \right\}, \\ &= \operatorname{prox}_{(\mu/\rho)\|\cdot\|_1}\left(x^{k+1} + y^k/\rho\right), \\ y^{k+1} &= y^k + \tau\rho(x^{k+1} - z^{k+1}). \end{aligned}$$

LASSO 问题的Primal 形式

- 注意，因为$\rho > 0$，所以$A^{\mathrm{T}}A + \rho I$总是可逆的．x迭代本质上是计算一个岭回归问题（ℓ_2范数平方正则化的最小二乘问题）；而对z的更新为ℓ_1范数的邻近算子，同样有显式解．在求解x迭代时，若使用固定的罚因子ρ，我们可以缓存矩阵$A^{\mathrm{T}}A + \rho I$的初始分解，从而减小后续迭代中的计算量．

- 需要注意的是，在LASSO 问题中，矩阵$A \in \mathbb{R}^{m\times n}$通常有较多的列（即$m \ll n$），因此$A^{\mathrm{T}}A \in \mathbb{R}^{n\times n}$是一个低秩矩阵，二次罚项的作用就是将$A^{\mathrm{T}}A$增加了一个正定项．该ADMM 主要运算量来自更新$x$变量时求解线性方程组，复杂度为$O(n^3)$

- 如果A 低秩，可以利用SMW公式计算

$$(A^{\mathrm{T}}A + \rho I)^{-1} = \rho^{-1}I - \rho^{-1}A^{\mathrm{T}}(\rho I + AA^{\mathrm{T}})^{-1}A$$

LASSO 问题的对偶形式

- 考虑LASSO 问题的对偶问题

$$\begin{aligned} \min \quad & b^{\mathrm{T}}y + \tfrac{1}{2}\|y\|^2, \\ \text{s.t.} \quad & \|A^{\mathrm{T}}y\|_\infty \le \mu. \end{aligned} \tag{14}$$

- 引入约束$A^{\mathrm{T}}y + z = 0$，可以得到如下等价问题：

$$\begin{aligned} \min \quad & \underbrace{b^{\mathrm{T}}y + \frac{1}{2}\|y\|^2}_{f(y)} + \underbrace{I_{\|z\|_\infty \le \mu}(z)}_{h(z)}, \\ \text{s.t.} \quad & A^{\mathrm{T}}y + z = 0. \end{aligned} \tag{15}$$

- 对约束$A^{\mathrm{T}}y + z = 0$引入乘子x，对偶问题的增广拉格朗日函数为

$$L_\rho(y, z, x) = b^{\mathrm{T}}y + \frac{1}{2}\|y\|^2 + I_{\|z\|_\infty \le \mu}(z) - x^{\mathrm{T}}(A^{\mathrm{T}}y + z) + \frac{\rho}{2}\|A^{\mathrm{T}}y + z\|^2.$$

LASSO 问题的对偶形式

- 当固定y, x时，对z的更新即向无穷范数球$\{z \mid \|z\|_\infty \le \mu\}$做欧几里得投影，即将每个分量截断在区间$[-\mu, \mu]$中；当固定$z, x$时，对$y$的更新即求解线性方程组

$$(I + \rho AA^{\mathrm{T}})y = A(x^k - \rho z^{k+1}) - b.$$

- 因此得到ADMM 迭代格式为

$$\begin{aligned} z^{k+1} &= \mathcal{P}_{\|z\|_\infty \le \mu}\left(x^k/\rho - A^{\mathrm{T}}y^k\right), \\ y^{k+1} &= (I + \rho AA^{\mathrm{T}})^{-1}\left(A(x^k - \rho z^{k+1}) - b\right), \\ x^{k+1} &= x^k - \tau\rho(A^{\mathrm{T}}y^{k+1} + z^{k+1}). \end{aligned}$$

- 虽然ADMM 应用于对偶问题也需要求解一个线性方程组，但由于LASSO 问题的特殊性（$m \ll n$），求解y更新的线性方程组需要的计算量是$O(m^3)$，使用缓存分解技巧后可进一步降低至$O(m^2)$，这大大小于针对原始问题的ADMM .

- 对许多问题x本身不稀疏，但在某种变换下是稀疏的:

$$\min_x \quad \mu\|Fx\|_1 + \frac{1}{2}\|Ax-b\|^2. \tag{16}$$

- 一个重要的例子是当$F \in \mathbb{R}^{(n-1)\times n}$是一阶差分矩阵

$$F_{ij} = \begin{cases} 1, & j = i+1, \\ -1, & j = i, \\ 0, & \text{其他}, \end{cases}$$

且$A = I$时，广义LASSO问题为

$$\min_x \quad \frac{1}{2}\|x-b\|^2 + \mu\sum_{i=1}^{n-1}|x_{i+1}-x_i|,$$

这个问题就是图像去噪问题的TV 模型；当$A = I$且F是二阶差分矩阵时，问题(16)被称为一范数趋势滤波.

- 通过引入约束$Fx = z$:

$$\begin{aligned} \min_{x,z} \quad & \frac{1}{2}\|Ax - b\|^2 + \mu\|z\|_1, \\ \text{s.t.} \quad & Fx - z = 0, \end{aligned} \tag{17}$$

- 引入乘子y，其增广拉格朗日函数为

$$L_\rho(x, z, y) = \frac{1}{2}\|Ax - b\|^2 + \mu\|z\|_1 + y^{\mathrm{T}}(Fx - z) + \frac{\rho}{2}\|Fx - z\|^2.$$

- 此问题的x迭代是求解方程组

$$(A^{\mathrm{T}}A + \rho F^{\mathrm{T}}F)x = A^{\mathrm{T}}b + \rho F^{\mathrm{T}}\left(z^k - \frac{y^k}{\rho}\right),$$

而z迭代依然通过ℓ_1范数的邻近算子.

- 因此交替方向乘子法所产生的迭代为

$$
\begin{aligned}
x^{k+1} &= (A^{\mathrm{T}}A + \rho F^{\mathrm{T}}F)^{-1}\left(A^{\mathrm{T}}b + \rho F^{\mathrm{T}}\left(z^k - \frac{y^k}{\rho}\right)\right),\\
z^{k+1} &= \mathrm{prox}_{(\mu/\rho)\|\cdot\|_1}\left(Fx^{k+1} + \frac{y^k}{\rho}\right),\\
y^{k+1} &= y^k + \tau\rho(Fx^{k+1} - z^{k+1}).
\end{aligned}
$$

- 对于全变差去噪问题，$A^{\mathrm{T}}A + \rho F^{\mathrm{T}}F$是三对角矩阵，所以此时$x$迭代可以在$\mathcal{O}(n)$的时间复杂度内解决；对于图像去模糊问题，$A$是卷积算子，则利用傅里叶变换可将求解方程组的复杂度降低至$O(n\log n)$；对于一范数趋势滤波问题，$A^{\mathrm{T}}A + \rho F^{\mathrm{T}}F$是五对角矩阵，所以$x$迭代仍可以在$\mathcal{O}(n)$的时间复杂度内解决

半定规划问题

考虑

$$\begin{aligned} \min_{X\in S^n} \quad & \langle C, X\rangle \\ \text{s.t.} \quad & \left\langle A^{(i)}, X\right\rangle = b_i, \quad i = 1, \cdots, m, \\ & X \succeq 0 \end{aligned}$$

对偶问题为

$$(D) \quad \begin{cases} \min\limits_{y\in\mathbb{R}^m, S\in S^n} & -b^\top y \\ \text{s.t.} & \mathcal{A}^*(y) + S = C, \quad S \succeq 0, \end{cases}$$

增广拉格朗日函数为

$$\mathcal{L}_\mu(X, y, S) = -b^\top y + \langle X, \mathcal{A}^*(y) + S - C\rangle + \frac{1}{2\mu}\|\mathcal{A}^*(y) + S - C\|_F^2.$$

ADMM迭代格式为

$$\begin{aligned} y^{k+1} &:= \arg\min_{y\in\mathbb{R}^m} \mathcal{L}_\mu(X^k,\ y,\ S^k), \\ &= -(\mathcal{A}\mathcal{A}^*)^{-1}\left(\mu(\mathcal{A}(X^k)-b)+\mathcal{A}(S^k-C)\right) \\ S^{k+1} &:= \arg\min_{S\in\mathcal{S}^n} \mathcal{L}_\mu(X^k,\ y^{k+1},\ S), \quad S\succeq 0, \\ X^{k+1} &:= X^k + \frac{\mathcal{A}^*(y^{k+1})+S^{k+1}-C}{\mu}. \end{aligned}$$

- 关于S的子问题:

$$\min_{S\in\mathcal{S}^n} \quad \left\|S-V^{k+1}\right\|_F^2, \quad S\succeq 0,$$

其中$V^{k+1} := V(S^k,X^k) = C-\mathcal{A}^*(y(S^k,X^k))-\mu X^k$.
显示解为

$$S^{k+1} := V_\dagger^{k+1} := Q_\dagger \Sigma_+ Q_\dagger^\top$$

其中$V^{k+1} = Q\Sigma Q^\top = \begin{pmatrix} Q_\dagger & Q_\ddagger \end{pmatrix}\begin{pmatrix} \Sigma_+ & 0 \\ 0 & \Sigma_- \end{pmatrix}\begin{pmatrix} Q_\dagger^\top \\ Q_\ddagger^\top \end{pmatrix}$

更新拉格朗日乘子X^{k+1}

- 更新格式:

$$X^{k+1} := X^k + \frac{\mathcal{A}^*(y^{k+1}) + S^{k+1} - C}{\mu}$$

- 等价形式:

$$X^{k+1} = \frac{1}{\mu}(S^{k+1} - V^{k+1}) = \frac{1}{\mu}V_{\ddagger}^{k+1},$$

其中$V_{\ddagger}^{k+1} := -Q_{\ddagger}\Sigma_{-}Q_{\ddagger}$.

- 注意到X^{k+1} 也是如下优化问题的最优解

$$\min_{X \in S^n} \quad \left\| \mu X + V^{k+1} \right\|_F^2, \quad X \succeq 0.$$

- 该问题的基本形式是

$$\min_{X} \quad \langle S, X\rangle - \ln\det X + \mu\|X\|_1, \tag{18}$$

其中S是已知的对称矩阵，通常由样本协方差矩阵得到. 变量$X \in \mathcal{S}^n_{++}$，$\|\cdot\|_1$定义为矩阵所有元素绝对值的和.

- 目标函数由光滑项和非光滑项组成，因此引入约束$X = Z$将问题的两部分分离：

$$\begin{aligned} \min \quad & \underbrace{\langle S, X\rangle - \ln\det X}_{f(X)} + \underbrace{\mu\|Z\|_1}_{h(Z)}, \\ \text{s.t.} \quad & X = Z. \end{aligned}$$

引入乘子U作用在约束$X - Z = 0$上，可得增广拉格朗日函数为

$$L_\rho(X, Z, U) = \langle S, X\rangle - \ln\det X + \mu\|Z\|_1 + \langle U, X - Z\rangle + \frac{\rho}{2}\|X - Z\|_F^2.$$

- 首先，固定Z^k, U^k，则X子问题是凸光滑问题，对X求矩阵导数并令其为零，

$$S - X^{-1} + U^k + \rho(X - Z^k) = 0.$$

这是一个关于X的矩阵方程，可以求出满足上述矩阵方程的唯一正定的X为

$$X^{k+1} = Q\mathrm{Diag}(x_1, x_2, \cdots, x_n)Q^{\mathrm{T}},$$

其中Q包含矩阵$S - \rho Z^k + U^k$的所有特征向量，x_i的表达式为

$$x_i = \frac{-d_i + \sqrt{d_i^2 + 4\rho}}{2\rho},$$

d_i为矩阵$S - \rho Z^k + U^k$的第i个特征值.

- 固定X^{k+1}, U^k，则Z的更新为矩阵ℓ_1 范数的邻近算子.
- 最后是常规的乘子更新.

- 考虑矩阵分离问题：

$$\begin{aligned} \min_{X,S} \quad & \|X\|_* + \mu\|S\|_1, \\ \text{s.t.} \quad & X + S = M, \end{aligned} \tag{19}$$

其中$\|\cdot\|_1$与$\|\cdot\|_*$分别表示矩阵ℓ_1范数与核范数.

- 引入乘子Y作用在约束$X+S=M$上，我们可以得到此问题的增广拉格朗日函数

$$L_\rho(X,S,Y) = \|X\|_* + \mu\|S\|_1 + \langle Y, X+S-M\rangle + \frac{\rho}{2}\|X+S-M\|_F^2. \tag{20}$$

- 对于X子问题，

$$\begin{aligned} X^{k+1} &= \operatorname*{argmin}_X \ L_\rho(X, S^k, Y^k) \\ &= \operatorname*{argmin}_X \left\{ \|X\|_* + \frac{\rho}{2} \left\| X + S^k - M + \frac{Y^k}{\rho} \right\|_F^2 \right\}, \\ &= \operatorname*{argmin}_X \left\{ \frac{1}{\rho}\|X\|_* + \frac{1}{2} \left\| X + S^k - M + \frac{Y^k}{\rho} \right\|_F^2 \right\}, \\ &= U\mathrm{Diag}\Big(\mathrm{prox}_{(1/\rho)\|\cdot\|_1}(\sigma(A))\Big)V^{\mathrm{T}}, \end{aligned}$$

其中$A = M - S^k - \frac{Y^k}{\rho}$，$\sigma(A)$为$A$的所有非零奇异值构成的向量并且$U\mathrm{Diag}(\sigma(A))V^{\mathrm{T}}$为$A$的约化奇异值分解.

- 对于S子问题，

$$\begin{aligned} S^{k+1} &= \operatorname*{argmin}_S \ L_\rho(X^{k+1}, S, Y^k) \\ &= \operatorname*{argmin}_S \left\{ \mu\|S\|_1 + \frac{\rho}{2}\left\| X^{k+1} + S - M + \frac{Y^k}{\rho} \right\|_F^2 \right\} \\ &= \operatorname{prox}_{(\mu/\rho)\|\cdot\|_1}\left(M - X^{k+1} - \frac{Y^k}{\rho} \right). \end{aligned}$$

- 那么交替方向乘子法的迭代格式为

$$\begin{aligned} X^{k+1} &= U\operatorname{Diag}\left(\operatorname{prox}_{(1/\rho)\|\cdot\|_1}(\sigma(A))\right)V^{\mathrm{T}}, \\ S^{k+1} &= \operatorname{prox}_{(\mu/\rho)\|\cdot\|_1}\left(M - L^{k+1} - \frac{Y^k}{\rho} \right), \\ Y^{k+1} &= Y^k + \tau\rho(X^{k+1} + S^{k+1} - M). \end{aligned}$$

图像去噪模型

$$b = Kx_t + w$$

- x_t 为未知图像
- b 为观察到的图像，模糊且有噪声；w 为噪声
- $N \times N$的像素点按列储存为长为N^2的向量

模糊矩阵K

- 表示一个2维的卷积，是有空间不动点的扩散函数
- 满足周期边界条件，有循环块(circulant blocks)
- 可对角化，即存在酉的2维离散傅立叶变换矩阵W，使得

$$K = W^H \mathbf{diag}(\lambda) W.$$

系数矩阵为$I + K^T K$ 的线性方程组可在$O(N^2 \log N)$ 的时间内求解。

$$\begin{aligned} \min \quad & \|Kx-b\|_1+\gamma\|Dx\|_{tv} \\ \text{s.t.} \quad & 0\le x\le 1 \end{aligned}$$

目标函数的第二项称为全变差罚函数

- Dx 是离散化的水平和垂直的方向导数

$$\begin{pmatrix} I\otimes D_1 \\ D_1\otimes I \end{pmatrix}, \quad \begin{pmatrix} -1 & 0 & 0 & \cdots & 0 & 0 & 1 \\ 1 & -1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 1 & -1 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & -1 & 0 & 0 \\ 0 & 0 & 0 & \cdots & 1 & -1 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 1 & -1 \end{pmatrix}$$

- $\|\cdot\|_{tv}$是欧式距离的和：$\|(u,v)\|_{tv}=\sum_{i=1}^{n}\sqrt{u_i^2+v_i^2}$

ADMM求解图像去模糊问题

考虑对原问题进行拆分:

$$\min \ \|u\|_1 + \gamma\|v\|_{tv}, \ \text{ s.t. } u = Kx - b, \ v = Dx, y = x, 0 \le y \le 1$$

ADMM 算法要求:

- 将$\|u\|_1$ 和$\|v\|_{tv}$的proximal算子以及在集合C上的投影这三部分分离
- 在$O(N^2 \log N)$的时间内求解系数矩阵为$I + K^T K + D^T D$ 的线性方程组

图像去模糊的实例

- 1024 $\times$ 1024的图像，满足周期边界条件
- 高斯模糊
- 椒盐噪声(salt-and-pepper noise)：50% 的像素点被随机替换为0/1

original

noisy/blurred

restored

$$\begin{aligned} \min_{x_i,z} \quad & \sum_{i=1}^{N} \phi_i(x_i), \\ \text{s.t.} \quad & x_i - z = 0,\ i = 1, 2, \cdots, N. \end{aligned}$$

- 增广拉格朗日函数为

$$L_\rho(x_1, \cdots, x_N, z, y_1, \cdots, y_N) = \sum_{i=1}^{N} \phi_i(x_i) + \sum_{i=1}^{N} y_i^{\mathrm{T}}(x_i - z) + \frac{\rho}{2} \sum_{i=1}^{N} \|x_i - z\|^2.$$

- 固定z^k, y_i^k，更新x_i的公式为

$$x_i^{k+1} = \operatorname*{argmin}_x \left\{ \phi_i(x) + \frac{\rho}{2} \left\| x - z^k + y_i^k/\rho \right\|^2 \right\}. \tag{21}$$

- 注意：虽然表面上看增广拉格朗日函数有$(N+1)$个变量块，但本质上还是两个变量块. 这是因为在更新某个x_i时并没有利用其他x_i，所有x_i可以看成一个整体. 相应地，所有乘子y_i也可以看成一个整体.

- 迭代式(21)的具体计算依赖于ϕ_i的形式，在一般情况下更新x_i的表达式为

$$x_i^{k+1} = \text{prox}_{\phi_i/\rho}\left(z^k - y_i^k/\rho\right).$$

- 固定x_i^{k+1}, y_i^k，问题关于z是二次函数，因此可以直接写出显式解：

$$z^{k+1} = \frac{1}{N}\sum_{i=1}^{N}\left(x_i{}^{k+1} + y_i^k/\rho\right).$$

- 综上，该问题的交替方向乘子法迭代格式为

$$\begin{aligned}
x_i^{k+1} &= \text{prox}_{\phi_i/\rho}\left(z^k - y_i^k/\rho\right),\ i = 1, 2, \cdots, N,\\
z^{k+1} &= \frac{1}{N}\sum_{i=1}^{N}\left(x_i^{k+1} + y_i^k/\rho\right),\\
y_i^{k+1} &= y_i{}^k + \tau\rho(x_i{}^{k+1} - z^{k+1}),\ i = 1, 2, \cdots, N.
\end{aligned}$$

形式一：考虑f是不可分(inseparable)函数，g是可分(separable)函数

$$\min_{\mathbf{x},\mathbf{z}} \sum_{l=1}^{L} (f_l(\mathbf{x}) + g_l(\mathbf{z}_l)), \text{ s.t. } \mathbf{A}\mathbf{x} + \mathbf{z} = \mathbf{b}$$

- 将$\mathbf{x}$复制L 份: $\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_L$
- 拆分矩阵和向量

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_1 \\ \vdots \\ \mathbf{A}_L \end{bmatrix}, \mathbf{z} = \begin{bmatrix} \mathbf{z}_1 \\ \vdots \\ \mathbf{z}_L \end{bmatrix}, \mathbf{b} = \begin{bmatrix} \mathbf{b}_1 \\ \vdots \\ \mathbf{b}_L \end{bmatrix}$$

将$\mathbf{A}\mathbf{x} + \mathbf{z} = \mathbf{0}$转化为

$$\mathbf{A}_l\mathbf{x}_l + \mathbf{z}_l = \mathbf{b}_l, \mathbf{x}_l - \mathbf{x} = \mathbf{0}, l = 1, \cdots, L.$$

模型一：

$$\begin{aligned} \min_{\mathbf{x},\{\mathbf{x}_l\},\mathbf{z}} \quad & \sum_{l=1}^{L}(f_l(\mathbf{x}_l)+g_l(\mathbf{z}_l)) \\ \text{s.t.} \quad & \mathbf{A}_l\mathbf{x}_l+\mathbf{z}_l=\mathbf{b}_l, \mathbf{x}_l-\mathbf{x}=\mathbf{0}, l=1,\cdots,L. \end{aligned}$$

- $\mathbf{x}_l$ 是$\mathbf{x}$ 的备份
- $\mathbf{z}_l$是$\mathbf{z}$的一部分
- 将$\{\mathbf{x}_l\},\mathbf{z},\mathbf{x}$分为两块
  - $\{\mathbf{x}_l\}$: 给定$\mathbf{z}$ 和$\mathbf{x}$，$\mathbf{x}_l$的更新是可分的
  - $(\mathbf{z},\mathbf{x})$: 给定$\{\mathbf{x}_l\}$，$\mathbf{z}_l$ 和$\mathbf{x}$ 的更新也是可分的
    因此可以使用标准的2块的ADMM算法
- 还可以在目标函数中加入简单的正则项$h(\mathbf{x})$

分布式ADMM I

考虑用MPI建立L个计算节点。

- $\mathbf{A}_l$ 是仅储存在节点l 上的局部数据
- $\mathbf{x}_l, \mathbf{z}_l$ 是局部变量，$\mathbf{x}_l$ 仅在节点l上 储存和更新
- $\mathbf{x}$ 是全局变量，由MPI指派和计算
- $\mathbf{y}_l, \bar{\mathbf{y}}_l$ 分别是$\mathbf{A}_l\mathbf{x}_l + \mathbf{z}_l = \mathbf{b}_l$ 和$\mathbf{x}_l - \mathbf{x} = \mathbf{0}$对应的拉格朗日乘子，仅在节点$l$上 储存和更新

每次迭代中，

- 每个节点l 各自使用数据$\mathbf{A}_l$计算$\mathbf{x}_l^{k+1}$
- 每个节点l各自计算$\mathbf{z}_l^{k+1}$，并准备$\mathbf{P}_l = (\cdots)$
- MPI 收集$\mathbf{P}_l$ 并将他们的平均值$\mathbf{x}^{k+1}$ 分发到各节点上
- 每个节点l 各自计算乘子$\mathbf{y}_l^{k+1}, \bar{\mathbf{y}}_l^{k+1}$

形式二：考虑f和g均是可分的函数

$$\min \sum_{j=1}^{N} f_j(\mathbf{x}_j) + \sum_{i=1}^{M} g_i(\mathbf{z}_i), \text{ s.t. } \mathbf{A}\mathbf{x} + \mathbf{z} = \mathbf{b},$$

其中

$$\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N), \mathbf{z} = (\mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_M).$$

将$\mathbf{A}$分解为

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} & \cdots & \mathbf{A}_{1N} \\ \mathbf{A}_{21} & \mathbf{A}_{22} & \cdots & \mathbf{A}_{2N} \\ & & \cdots & \\ \mathbf{A}_{M1} & \mathbf{A}_{M2} & \cdots & \mathbf{A}_{MN} \end{bmatrix}, \ \mathbf{b} = \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \\ \vdots \\ \mathbf{b}_M \end{bmatrix}.$$

我们可以得到类似的模型

$$\min \sum_{j=1}^{N} f_j(\mathbf{x}_j) + \sum_{i=1}^{M} g_i(\mathbf{z}_i), \text{ s.t. } \sum_{j=1}^{N} \mathbf{A}_{ij}\mathbf{x}_j + \mathbf{z}_i = \mathbf{b}_i, i = 1, \cdots, M.$$

分布式ADMM II

注意到$\mathbf{A}_{ij}\mathbf{x}_j'$在约束中是耦合的，因此令

$$\mathbf{p}_{ij} = \mathbf{A}_{ij}\mathbf{x}_j,$$

模型二：

$$\min \sum_{j=1}^{N} f_j(\mathbf{x}_j) + \sum_{i=1}^{M} g_i(\mathbf{z}_i), \text{ s.t. } \begin{array}{l} \sum_{j=1}^{N} \mathbf{p}_{ij} + \mathbf{z}_i = \mathbf{b}_i, \forall i, \\ \mathbf{p}_{ij} - \mathbf{A}_{ij}\mathbf{x}_j = 0, \forall i, j. \end{array}$$

ADMM迭代

- 交替更新$\{\mathbf{p}_{ij}\}$ 和$(\{\mathbf{x}_j\}, \{\mathbf{z}_i\})$
- 关于$\mathbf{p}_{ij}$ 的子问题有闭形式解
- 关于$(\{\mathbf{x}_j\}, \{\mathbf{z}_i\})$的子问题对$\mathbf{x}_j$ 和$\mathbf{z}_i$是可分的
  - $\mathbf{x}_j$的更新需要f_j 和$\mathbf{A}_{1j}^T\mathbf{A}_{1j}, \cdots, \mathbf{A}_{Mj}^T\mathbf{A}_{Mj}$;
  - $\mathbf{z}_i$的更新需要g_i.

思考：如何进一步将f_j和$\mathbf{A}_{1j}^T\mathbf{A}_{1j}, \cdots, \mathbf{A}_{Mj}^T\mathbf{A}_{Mj}$ 去耦合?

分布式ADMM III

对每个$\mathbf{x}_j$，作M 个独立的备份$\mathbf{x}_{1j}, \mathbf{x}_{2j}, \cdots, \mathbf{x}_{Mj}$.
模型三：

$$\min \sum_{j=1}^{N} f_j(\mathbf{x}_j) + \sum_{i=1}^{M} g_i(\mathbf{z}_i), \text{ s.t. } \begin{array}{ll} \sum_{j=1}^{N} \mathbf{p}_{ij} + \mathbf{z}_i = \mathbf{b}_i, & \forall i, \\ \mathbf{p}_{ij} - \mathbf{A}_{ij}\mathbf{x}_{ij} = \mathbf{0}, & \forall i, j, \\ \mathbf{x}_j - \mathbf{x}_{ij} = \mathbf{0}, & \forall i, j. \end{array}$$

ADMM

- 交替更新$(\{\mathbf{x}_j\}, \{\mathbf{p}_{ij}\})$ 和$(\{\mathbf{x}_j\}, \{\mathbf{z}_i\})$
- 关于$(\{\mathbf{x}_j\}, \{\mathbf{p}_{ij}\})$的子问题是可分的
  - $\mathbf{x}_j$的更新只需要f_j，即只需计算prox_{f_j}
  - $\mathbf{p}_{ij}$的更新有闭形式解
- 关于$(\{\mathbf{x}_{ij}\}, \{\mathbf{z}_i\})$的子问题也是可分的
  - $\mathbf{x}_{ij}$的更新需要$(\alpha I + \beta \mathbf{A}_{ij}^T \mathbf{A}_{ij})$;
  - $\mathbf{y}_i$的更新只需要g_i，即只需计算prox_{g_i}.

去中心化ADMM

对$\mathbf{x}$进行局部备份得到$\mathbf{x}_i$，不使用约束

$$\mathbf{x}_i - \mathbf{x} = 0, i = 1, \cdots, M,$$

而是考虑图$\mathcal{G} = (\mathcal{V}, \varepsilon)$其中$\mathcal{V}$为顶点集，$\varepsilon$为边集。

根据图$\mathcal{G}$建立约束

$$\begin{aligned} \mathbf{x}_i - \mathbf{x}_j &= \mathbf{0}, && \forall (i,j) \in \varepsilon, && \text{or} \\ \mathbf{x}_i - \mathbf{z}_{ij} &= \mathbf{0}, \mathbf{x}_j - \mathbf{z}_{ij} = \mathbf{0} && \forall (i,j) \in \varepsilon, && \text{or} \\ \text{mean}\{\mathbf{x}_j : (i,j) \in \varepsilon\} - \mathbf{x}_i &= \mathbf{0}, && \forall i \in \mathcal{V}. \end{aligned}$$

去中心化ADMM

- 去中心化ADMM在互相连接的网络上运行
- 没有控制和分发数据的中心
- 应用:
  - wireless sensor networks
  - collaborative learning
- 去中心化ADMM交替进行如下步骤
  - 每个节点进行局部计算
  - 相邻节点交流或传播信息
- 因为数据没有共享或储存于某个中心，因此数据安全性得到保证
- 算法的收敛速度受如下条件影响
  - 问题的性质，如目标函数的凸性，问题的条件数等
  - 图的规模，连通性，谱性质等

V. Chandrasekaran, P.Parrilo, A. Willsky

包含正则项的极大似然模型

$$\min_{R,S,L} \langle R, \hat{\Sigma}_X \rangle - \log\det(R) + \alpha\|S\|_1 + \beta Tr(L), \text{ s.t. } R = S - L, R \succ 0, L \succeq 0,$$

其中X为观察到的变量, $\Sigma_X^{-1} \approx R = S - L$, S稀疏, L低秩. 目标函数的前两项来自对数极大似然函数

$$l(K;\Sigma) = \log\det(K) - \mathrm{tr}(K\Sigma).$$

引入指示函数

$$\mathcal{I}(L \succeq 0) := \begin{cases} 0, & \text{if } L \succeq 0 \\ +\infty, & \text{otherwise.} \end{cases}$$

模型可以改写为3项的ADMM算法

$$\min_{R,S,L} \langle R, \hat{\Sigma}_X \rangle - \log\det(R) + \alpha\|S\|_1 + \beta \mathrm{Tr}(L) + \mathcal{I}(L \succeq 0), \text{ s.t. } R - S + L = 0.$$

稳定主成分追踪(PCP)

模型一：

$$\begin{aligned} \min_{L,S,Z} \quad & \|L\|_* + \rho\|S\|_1 \\ \text{s.t.} \quad & L + S + Z = M \\ & \|Z\|_F \le \sigma, \end{aligned}$$

注：$M =$ 低秩矩阵+ 稀疏矩阵+ 噪声.
在图像处理中还要加入非负约束$L \ge 0$.
模型二：

$$\begin{aligned} \min_{L,S,Z,K} \quad & \|L\|_* + \rho\|S\|_1 + \mathcal{I}(\|Z\|_F \le \sigma) + \mathcal{I}(K \ge 0) \\ \text{s.t.} \quad & L + S + Z = M \\ & L - K = 0. \end{aligned}$$

约束可以整合为

$$\begin{pmatrix} I & I \\ I & 0 \end{pmatrix} \begin{pmatrix} L \\ S \end{pmatrix} + \begin{pmatrix} I & 0 \\ 0 & -I \end{pmatrix} \begin{pmatrix} Z \\ K \end{pmatrix} = \begin{pmatrix} M \\ 0 \end{pmatrix}.$$

混合全变差和l_1正则

原始模型：

$$\min_x \ TV(x) + \alpha\|Wx\|_1, \ \text{s.t.} \ \|Rx - b\|_2 \leq \sigma.$$

新模型：

$$\begin{aligned} \min_x \quad & \Sigma_i \|z_i\|_2 + \alpha\|Wx\|_1 + \mathcal{I}(\|y\|_2 \leq \sigma) \\ \text{s.t.} \quad & z_i = D_i x, \forall i = 1, \cdots, N \\ & y = Rx - b. \end{aligned}$$

将变量分为两块：x 和$(y, \{z_i\})$

$$\begin{pmatrix} R \\ D_1 \\ \vdots \\ D_N \end{pmatrix} x - \begin{pmatrix} y \\ z_1 \\ \vdots \\ z_N \end{pmatrix} = \begin{pmatrix} b \\ 0 \\ \vdots \\ 0 \end{pmatrix},$$

x的子问题不易求解.

将变量去耦合的方法

- 使用线性化技巧简化proximal算子，进行非精确更新
- 引入新变量作为桥梁，类似分布式ADMM

例如考虑如下问题

$$\min_{\mathbf{x}_1,\mathbf{x}_2,\mathbf{y}} (f_1(\mathbf{x}_1)+f_2(\mathbf{x}_2))+g(\mathbf{y}),\ \text{s.t.}\ (\mathbf{A}_1\mathbf{x}_1+\mathbf{A}_2\mathbf{x}_2)+\mathbf{B}\mathbf{y}=\mathbf{b}.$$

在ADMM迭代中$(\mathbf{x}_1,\mathbf{x}_2)$的子问题中$\mathbf{x}_1$ 和$\mathbf{x}_2$ 是耦合的，但是进行线性化后$\mathbf{x}_1$ 和$\mathbf{x}_2$的子问题是独立的.

$$\min_{\mathbf{x}_1,\mathbf{x}_2}(f_1(\mathbf{x}_1)+f_2(\mathbf{x}_2))+\left\langle \begin{bmatrix} g_1 \\ g_2 \end{bmatrix}, \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} \right\rangle + \frac{1}{2t}\left\| \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} - \begin{bmatrix} \mathbf{x}_1^k \\ \mathbf{x}_2^k \end{bmatrix} \right\|_2^2.$$

非凸约束问题

考虑如下约束优化问题：

$$\begin{aligned} \min_{\boldsymbol{x}} \quad & f(\boldsymbol{x}), \\ s.t. \quad & \boldsymbol{x} \in \mathcal{S}, \end{aligned}$$

其中f是凸的，但是$\mathcal{S}$是非凸的。可以将上述问题改写为：

$$\begin{aligned} \min_{\boldsymbol{x}} \quad & f(\boldsymbol{x}) + \mathbb{I}_{\mathcal{S}}(\boldsymbol{z}), \\ s.t. \quad & \boldsymbol{x} - \boldsymbol{z} = \boldsymbol{0}, \end{aligned}$$

交替方向乘子法产生如下迭代：

$$\begin{aligned} \boldsymbol{x}^{k+1} &= \operatorname*{argmin}_{\boldsymbol{x}} \left(f(\boldsymbol{x}) + (\rho/2) \|\boldsymbol{x} - \boldsymbol{z}^k + \boldsymbol{u}^k\|_2^2 \right), \\ \boldsymbol{z}^{k+1} &= \Pi_{\mathcal{S}}(\boldsymbol{x}^{k+1} + \boldsymbol{u}^k), \\ \boldsymbol{u}^{k+1} &= \boldsymbol{u}^k + (\boldsymbol{x}^{k+1} - \boldsymbol{z}^{k+1}) \end{aligned}$$

其中，$\Pi_{\mathcal{S}}(\boldsymbol{z})$是将$\boldsymbol{z}$投影到集合$\mathcal{S}$中。因为$f$是凸的，所以上述$\boldsymbol{x}$-极小化步是凸问题，但是$\boldsymbol{z}$-极小化步是向一个非凸集合的投影。

一般来说，这种投影很难计算，但是在下面列出的这些特殊情形中可以精确求解。

- 基数：如果$\mathcal{S}=\{\boldsymbol{x}|\boldsymbol{card}(\boldsymbol{x})\leq c\}$，其中$\boldsymbol{card}(\boldsymbol{v})$表示非零元素的数目，那么$\Pi_{\mathcal{S}}(\boldsymbol{v})$保持前$c$大的元素不变，其他元素变为0。
  例如回归选择（也叫特征选择）问题：

$$\begin{aligned}\min_{\boldsymbol{x}} \quad & \|\boldsymbol{A}\boldsymbol{x}-\boldsymbol{b}\|_2^2,\\ s.t. \quad & \boldsymbol{card}(\boldsymbol{x})\leq c.\end{aligned}$$

- 秩：如果$\mathcal{S}$是秩为c的矩阵的集合，那么$\boldsymbol{card}(\boldsymbol{V})$可以通过对$\boldsymbol{V}$做奇异值分解，$\boldsymbol{V}=\sum_i \sigma_i \boldsymbol{u}_i\boldsymbol{u}_i^T$，然后保留前$c$大的奇异值及奇异向量，
  即$\Pi_{\mathcal{S}}(\boldsymbol{V})=\sum_{i=1}^{c}\sigma_i\boldsymbol{u}_i\boldsymbol{u}_i^T$。
- 布尔约束：如果$\mathcal{S}=\{\boldsymbol{x}|\boldsymbol{x}_i\in\{0,1\}\}$，那么$\Pi_{\mathcal{S}}(\boldsymbol{v})$就是简单地把每个元素变为0和1中离它更近的数。

非负矩阵分解和补全

非负矩阵分解和补全问题可以写成如下形式：

$$\begin{aligned} \min_{\boldsymbol{X},\boldsymbol{Y}} \quad & \|\mathcal{P}_{\Omega}(\boldsymbol{X}\boldsymbol{Y}-\boldsymbol{M})\|_F^2, \\ s.t. \quad & \boldsymbol{X}_{ij} \geq 0, \boldsymbol{Y}_{ij} \geq 0, \forall i,j, \end{aligned}$$

其中，Ω表示矩阵$\boldsymbol{M}$中的已知元素的下标集合，$\mathcal{P}_{\Omega}(\boldsymbol{A})$表示得到一个新的矩阵$\boldsymbol{A}'$，其下标在集合$\Omega$中的所对应的元素等于矩阵$\boldsymbol{A}$的对应元素，其下标不在集合$\Omega$中的所对应的元素为0。注意到，这个问题是非凸的。

为了利用交替方向乘子法的优势，我们考虑如下的等价形式：

$$\begin{aligned} \min_{\boldsymbol{U},\boldsymbol{V},\boldsymbol{X},\boldsymbol{Y},\boldsymbol{Z}} \quad & \frac{1}{2}\|\boldsymbol{X}\boldsymbol{Y}-\boldsymbol{Z}\|_F^2, \\ s.t. \quad & \boldsymbol{X}=\boldsymbol{U}, \boldsymbol{Y}=\boldsymbol{V}, \\ & \boldsymbol{U} \geq 0, \boldsymbol{V} \geq 0, \\ & \mathcal{P}_{\Omega}(\boldsymbol{Z}-\boldsymbol{M})=0. \end{aligned}$$

$$L_{\alpha,\beta}(\boldsymbol{X},\boldsymbol{Y},\boldsymbol{Z},\boldsymbol{U},\boldsymbol{V},\boldsymbol{\Lambda},\boldsymbol{\Pi}) = \frac{1}{2}\|\boldsymbol{XY}-\boldsymbol{Z}\|_F^2 + \boldsymbol{\Lambda}\bullet(\boldsymbol{X}-\boldsymbol{U}) + \boldsymbol{\Pi}\bullet(\boldsymbol{Y}-\boldsymbol{V}) + \frac{\alpha}{2}\|\boldsymbol{X}-\boldsymbol{U}\|_F^2 + \frac{\beta}{2}\|\boldsymbol{Y}-\boldsymbol{V}\|_F^2,$$

$$\begin{aligned}
\boldsymbol{X}^{k+1} &= \operatorname*{argmin}_{\boldsymbol{X}} \; L_{\alpha,\beta}(\boldsymbol{X},\boldsymbol{Y}^{\boldsymbol{k}},\boldsymbol{Z}^{\boldsymbol{k}},\boldsymbol{U}^{\boldsymbol{k}},\boldsymbol{V}^{\boldsymbol{k}},\boldsymbol{\Lambda}^{\boldsymbol{k}},\boldsymbol{\Pi}^{\boldsymbol{k}}),\\
\boldsymbol{Y}^{k+1} &= \operatorname*{argmin}_{\boldsymbol{Y}} \; L_{\alpha,\beta}(\boldsymbol{X}^{\boldsymbol{k+1}},\boldsymbol{Y},\boldsymbol{Z}^{\boldsymbol{k}},\boldsymbol{U}^{\boldsymbol{k}},\boldsymbol{V}^{\boldsymbol{k}},\boldsymbol{\Lambda}^{\boldsymbol{k}},\boldsymbol{\Pi}^{\boldsymbol{k}}),\\
\boldsymbol{Z}^{k+1} &= \operatorname*{argmin}_{\mathcal{P}_\Omega(\boldsymbol{Z}-\boldsymbol{M})=0} \; L_{\alpha,\beta}(\boldsymbol{X}^{\boldsymbol{k+1}},\boldsymbol{Y}^{\boldsymbol{k+1}},\boldsymbol{Z},\boldsymbol{U}^{\boldsymbol{k}},\boldsymbol{V}^{\boldsymbol{k}},\boldsymbol{\Lambda}^{\boldsymbol{k}},\boldsymbol{\Pi}^{\boldsymbol{k}}),\\
\boldsymbol{U}^{k+1} &= \operatorname*{argmin}_{\boldsymbol{U}\geq 0} \; L_{\alpha,\beta}(\boldsymbol{X}^{\boldsymbol{k+1}},\boldsymbol{Y}^{\boldsymbol{k+1}},\boldsymbol{Z}^{\boldsymbol{k+1}},\boldsymbol{U},\boldsymbol{V}^{\boldsymbol{k}},\boldsymbol{\Lambda}^{\boldsymbol{k}},\boldsymbol{\Pi}^{\boldsymbol{k}}),\\
\boldsymbol{V}^{k+1} &= \operatorname*{argmin}_{\boldsymbol{V}\geq 0} \; L_{\alpha,\beta}(\boldsymbol{X}^{\boldsymbol{k+1}},\boldsymbol{Y}^{\boldsymbol{k+1}},\boldsymbol{Z}^{\boldsymbol{k+1}},\boldsymbol{U}^{\boldsymbol{k+1}},\boldsymbol{V},\boldsymbol{\Lambda}^{\boldsymbol{k}},\boldsymbol{\Pi}^{\boldsymbol{k}}),\\
\Lambda^{k+1} &= \Lambda^k + \tau\alpha(\boldsymbol{X}^{k+1}-\boldsymbol{U}^{k+1}),\\
\Pi^{k+1} &= \Pi^k + \tau\beta(\boldsymbol{Y}^{k+1}-\boldsymbol{V}^{k+1}).
\end{aligned}$$

提纲

Douglas-Rachford splitting算法

考虑复合优化问题

$$\min \quad f(x) = g(x) + h(x)$$

其中g, h 是闭凸函数.

Douglas-Rachford迭代: 从任意初始点z^0开始，

$$\begin{aligned} x^k &= \text{prox}_{th}(z^{k-1}) \\ y^k &= \text{prox}_{tg}(2x^k - z^{k-1}) \\ z^k &= z^{k-1} + y^k - x^k \end{aligned}$$

- t为正常数
- 通常用于g, h 的proximal算子计算代价较小的场景
- 在较弱的条件下(如极小点存在)，迭代点列x^k收敛

- 从y的更新开始

$$y^+ = \text{prox}_{tg}(2x - z); \quad z^+ = z + y^+ - x; \quad x^+ = \text{prox}_{th}(z^+)$$

- 交换z和x的更新顺序

$$y^+ = \text{prox}_{tg}(2x - z); \quad x^+ = \text{prox}_{th}(z + y^+ - x); \quad z^+ = z + y^+ - x$$

- 作变量替换$w = z - x$

DR 迭代的等价形式**:** 从任意初始点$x^0 \in \mathbf{dom}\ h, w^0 \in t\partial h(x^0)$开始

$$\begin{aligned} y^+ &= \text{prox}_{tg}(x - w) \\ x^+ &= \text{prox}_{th}(y^+ + w) \\ w^+ &= w + y^+ - x^+ \end{aligned}$$

DR迭代和不动点迭代

Douglas-Rachford迭代可以写成不动点迭代的形式

$$z^k = F(z^{k-1}),$$

其中$F(z) = z + \text{prox}_{tg}(2\text{prox}_{th}(z) - z) - \text{prox}_{th}(z)$

DR迭代和不动点迭代的关系

- 若z是不动点，则$x = \text{prox}_{th}(z)$是目标函数的极小点

$$\begin{aligned} z = F(z), \quad x = prox_{th}(z) &\Rightarrow \text{prox}_{tg}(2x - z) = x = \text{prox}_{th}(z) \\ &\Rightarrow x - z \in t\partial g(x); z - x \in t\partial h(x) \\ &\Rightarrow 0 \in t\partial g(x) + t\partial h(x) \end{aligned}$$

- 若x是目标函数的极小点，$u \in t\partial g(x) \cap -t\partial h(x)$，则$x - u = F(x - u)$

Douglas-Rachford迭代的松弛形式

- 不动点迭代的松弛形式

$$z^+ = z + \rho(F(z) - z)$$

$1 < \rho < 2$ 称为超松弛，$0 < \rho < 1$称为次松弛

- DR迭代的松弛形式一

$$\begin{aligned} x^+ &= \mathrm{prox}_{th}(z) \\ y^+ &= \mathrm{prox}_{tg}(2x^+ - z) \\ z^+ &= z + \rho(y^+ - x^+) \end{aligned}$$

- DR迭代的松弛形式二

$$\begin{aligned} y^+ &= \mathrm{prox}_{tg}(x - w) \\ x^+ &= \mathrm{prox}_{th}((1-\rho)x + \rho y^+ + w) \\ w^+ &= w + \rho y^+ + (1-\rho)x - x^+ \end{aligned}$$

凸的原始问题

$$\begin{aligned} \min \quad & f_1(x_1) + f_2(x_2) \\ \text{s.t.} \quad & A_1 x_1 + A_2 x_2 = b \end{aligned}$$

对偶问题

$$\max \quad -b^T z - f_1^*(-A_1^T z) - f_2^*(-A_2^T z)$$

对对偶问题应用Douglas-Rachford 迭代：

$$\underbrace{b^T z + f_1^*(-A_1^T z)}_{g(z)} + \underbrace{f_2^*(-A_2^T z)}_{h(z)}$$

$$
\begin{aligned}
y^{k+1} &= \mathrm{prox}_{tg}(x^k - w^k),\\
x^{k+1} &= \mathrm{prox}_{th}(w^k + y^{k+1}),\\
w^{k+1} &= w^k + y^{k+1} - x^{k+1}.
\end{aligned}
$$

- 第一式的最优性条件为

$$0 \in tb - tA_1\partial f_1^*(-A_1^{\mathrm{T}}y^{k+1}) - x^k + w^k + y^{k+1},$$

上式等价于存在$x_1^k \in \partial f_1^*(-A_1^{\mathrm{T}}y^{k+1})$，使得

$$y^{k+1} = x^k - w^k + t(A_1x_1^k - b).$$

这就是如下更新的最优性条件.

$$x_1^k = \underset{x_1}{\mathrm{argmin}} \left\{ f_1(x_1) + (x^k)^{\mathrm{T}}(A_1x_1 - b) + \frac{t}{2}\|A_1x_1 - b - w^k/t\|_2^2 \right\}.$$

- 类似地，第二式的最优性条件为

$$0 \in tA_2 \partial f_2^*(-A_2^{\mathrm{T}} x^{k+1}) + w^k + y^{k+1} - x^{k+1},$$

其等价于存在$x_2^k \in \partial f_2^*(-A_2^{\mathrm{T}} x^{k+1})$，使得

$$x^{k+1} = x^k + t(A_1 x_1^k + A_2 x_2^k - b).$$

进一步等价于如下更新的最优性条件.

$$x_2^k = \underset{x_2}{\operatorname{argmin}} \left\{ f_2(x_2) + (x^k)^{\mathrm{T}}(A_2 x_2) + \frac{t}{2}\|A_1 x_1^k + A_2 x_2 - b\|_2^2 \right\}.$$

- 由第一、第二式的等价形式整理可得$w^{k+1} = -tA_2 x_2^k$.

定义增广拉格朗日函数

$$L_t(x_1,x_2,z)=f_1(x_1)+f_2(x_2)+z^T(A_1x_1+A_2x_2-b)+\frac{t}{2}\|A_1x_1+A_2x_2-b\|_2^2$$

1. 关于x_1极小化增广拉格朗日函数

$$\begin{aligned} x_1^k &= \operatorname*{argmin}_{x_1} L_t(x_1,x_2^{k-1},z^{k-1}) \\ &= \operatorname*{argmin}_{x_1}\left(f_1(x_1)+(z^{k-1})^TA_1x_1+\frac{t}{2}\|A_1x_1+A_2x_2^{k-1}-b\|_2^2\right) \end{aligned}$$

2. 关于x_2极小化增广拉格朗日函数

$$\begin{aligned} x_2^k &= \operatorname*{argmin}_{x_2} L_t(x_1^k,x_2,z^{k-1}) \\ &= \operatorname*{argmin}_{x_2}\left(f_2(x_2)+(z^{k-1})^TA_2x_2+\frac{t}{2}\|A_1x_1^k+A_2x_2-b\|_2^2\right) \end{aligned}$$

3. 对偶变量更新$z^k=z^{k-1}+t(A_1x_1^k+A_2x_2^k-b)$

非扩张函数

定义

函数f称为非扩张的*(non-expansive)*, 若

$$\|x-y\|^2 \geq \|f(x)-f(y)\|^2;$$

函数f称为固定非扩张的*(firmly non-expansive)*, 若

$$(f(x)-f(y))^\top (x-y) \geq \|f(x)-f(y)\|^2.$$

定理：prox_h 是固定非扩张的，进而也是非扩张的，因而是Lipschitz连续的(常数为1).

- 设$u=\text{prox}_h(x), v=\text{prox}_h(y)$，由proximal算子的性质

$$x-u \in \partial h(u), y-v \in \partial h(v) \quad \Rightarrow \quad (x-u-y+v)^\top (u-v) \geq 0$$

- 由柯西不等式, $\|u-v\|^2 \leq (x-y)^\top (u-v) \leq \|x-y\|\|u-v\|$, 因此

$$\|\text{prox}_h(x)-\text{prox}_h(y)\|_2 \leq \|x-y\|_2.$$

Douglas-Rachford迭代映射

定义迭代映射F, G如下

$$\begin{aligned} F(z) &= z + \text{prox}_{tg}(2\text{prox}_{th}(z) - z) - \text{prox}_{th}(z) \\ G(z) &= z - F(z) \\ &= \text{prox}_{th}(z) - \text{prox}_{tg}(2\text{prox}_{th}(z) - z) \end{aligned}$$

- F 是固定非扩张的

$$(F(z) - F(\hat{z}))^T (z - \hat{z}) \geq \|F(z) - F(\hat{z})\|_2^2 \quad \forall z, \hat{z}$$

- 进而G 也是固定非扩张的

$$\begin{aligned} & (G(z) - G(\hat{z}))^T (z - \hat{z}) \\ = \quad & \|G(z) - G(\hat{z})\|_2^2 + (F(z) - F(\hat{z}))^T (z - \hat{z}) - \|F(z) - F(\hat{z})\|_2^2 \\ \geq \quad & \|G(z) - G(\hat{z})\|_2^2 \end{aligned}$$

F固定非扩张性的证明.

- 设$x = \text{prox}_{th}(z), \hat{x} = \text{prox}_{th}(\hat{z})$, 以及

$$y = \text{prox}_{tg}(2x - z), \quad \hat{y} = \text{prox}_{tg}(2\hat{x} - \hat{z})$$

- 代入$F(z) = z + y - x$和$F(\hat{z}) = \hat{z} + \hat{y} - \hat{x}$:

$$\begin{aligned}
&(F(z) - F(\hat{z}))^T (z - \hat{z}) \\
&\geq (z + y - x - \hat{z} - \hat{y} + \hat{x})^T (z - \hat{z}) - (x - \hat{z})^T (z - \hat{z}) + \|x - \hat{x}\|_2^2 \\
&= (y - \hat{y})^T (z - \hat{z}) + \|z - x - \hat{z} + \hat{x}\|_2^2 \\
&= (y - \hat{y})^T (2x - z - 2\hat{x} + \hat{z}) - \|y - \hat{y}\|_2^2 + \|F(z) - F(\hat{z})\|_2^2 \\
&\geq \|F(z) - F(\hat{z})\|_2^2
\end{aligned}$$

其中用到了prox_{th}和prox_{tg} 的固定非扩张性

$$(x - \hat{x})^T (z - \hat{z}) \geq \|x - \hat{x}\|_2^2, \quad (2x - z - 2\hat{x} + \hat{z})^T (y - \hat{y}) \geq \|y - \hat{y}\|_2^2$$

□

DR迭代的收敛性分析

不动点迭代格式

$$\begin{aligned} z^k &= (1-\rho_k)z^{k-1} + \rho_k F(z^{k-1}) \\ &= z^{k-1} - \rho_k G(z^{k-1}) \end{aligned}$$

假设

- 最优值$f^* = \inf_x(g(x)+h(x))$ 有些且可取到
- $\rho_k \in [\rho_{\min}, \rho_{\max}]$, 其中$0 < \rho_{\min} < \rho_{\max} < 2$

结论

- z^k 收敛到F 的不动点z^*
- $x^k = \mathrm{prox}_{th}(z^{k-1})$ 收敛到原目标函数的极小点$x^* = \mathrm{prox}_{th}(z^*)$
  (利用prox_{th} 的连续性)

Proof.

设z^* 为$F(z)$ 的任意不动点，即为$G(z)$的零点. 考虑第k步迭代，
记$z = z^{k-1}, \rho = \rho_k, z^+ = z^k$, 则

$$\begin{aligned} \|z^+ - z^*\|_2^2 - \|z - z^*\|_2^2 &= 2(z^+ - z)^T(z - z^*) + \|z^+ - z\|_2^2 \\ &= -2\rho G(z)^T(z - z^*) + \rho^2\|G(z)\|_2^2 \\ &\le -\rho(2-\rho)\|G(z)\|_2^2 \\ &\le -M\|G(z)\|_2^2 \end{aligned} \tag{22}$$

其中$M = \rho_{\min}(2 - \rho_{\max})$, 第三行用到了$G$的固定非扩张性.

- (22) 可以推出

$$M\sum_{k=0}^{\infty}\|G(z^k)\|_2^2 \le \|z^0 - z^*\|_2^2, \quad \|G(z^k)\|_2 \to 0$$

- 并且$\|z^k - z^*\|_2$不增；进而z^k 有界
- 因为$\|z^k - z^*\|_2$ 不增, 因此极限$\lim_{k\to\infty}\|z^k - z^*\|_2$ 存在

Proof.

- 因为迭代点列z^k 有界，因此它有收敛子列
- 设$\bar{z}_k$ 收敛子列，极限为$\bar{z}$; 由G的连续性,

$$0 = \lim_{k\to\infty} G(\bar{z}_k) = G(\bar{z})$$

  因此$\bar{z}$ 是G 的零点，且极限$\lim_{k\to\infty} \|z^k - \bar{z}\|_2$ 存在
- 设存在两个子列分别收敛于$\bar{z}_1, \bar{z}_2$, 即极限

$$\lim_{k\to\infty} \|z^{k_{j_1}} - \bar{z}_1\|_2, \quad \lim_{k\to\infty} \|z^{k_{j_2}} - \bar{z}_2\|_2$$

  均存在, 则由z^k 的收敛性知

$$\|\bar{z}_2 - \bar{z}_1\|_2 = \lim_{k\to\infty} \|z^k - \bar{z}_1\|_2 = \lim_{k\to\infty} \|z^k - \bar{z}_2\|_2 = 0$$

□

提纲

假设

我们先引入一些必要的假设.

- $f_1(x), f_2(x)$均为闭凸函数，且每个ADMM 迭代子问题存在唯一解；
- 原始问题的解集非空，且Slater 条件满足.

注：假设给出的条件是很基本的.

- f_1和f_2的凸性保证了要求解的问题是凸问题，每个子问题存在唯一解是为了保证迭代的良定义
- 在Slater 条件满足的情况下，原始问题的KKT 对和最优解是对应的，因此可以很方便地使用KKT 条件来讨论收敛性.

由于原始问题解集非空，不妨设(x_1^*, x_2^*, y^*)是KKT 对，即满足条件

$$-A_1^{\mathrm{T}} y^* \in \partial f_1(x_1^*), \quad -A_2^{\mathrm{T}} y^* \in \partial f_2(x_2^*), \quad A_1 x_1^* + A_2 x_2^* = b.$$

我们最终的目的是证明ADMM 迭代序列$\{(x_1^k, x_2^k, y^k)\}$收敛到原始问题的一个KKT 对，因此引入如下记号来表示当前迭代点和KKT 对的误差：

$$(e_1^k, e_2^k, e_y^k) \overset{\text{def}}{=\!=} (x_1^k, x_2^k, y^k) - (x_1^*, x_2^*, y^*).$$

我们进一步引入如下辅助变量来简化之后的证明：

$$\begin{aligned}
u^k &= -A_1{}^{\mathrm{T}}[y^k + (1-\tau)\rho(A_1 e_1^k + A_2 e_2^k) + \rho A_2(x_2^{k-1} - x_2^k)], \\
v^k &= -A_2{}^{\mathrm{T}}[y^k + (1-\tau)\rho(A_1 e_1^k + A_2 e_2^k)], \\
\Psi_k &= \frac{1}{\tau\rho}\|e_y^k\|^2 + \rho\|A_2 e_2^k\|^2, \\
\Phi_k &= \Psi_k + \max(1-\tau, 1-\tau^{-1})\rho\|A_1 e_1^k + A_2 e_2^k\|^2.
\end{aligned} \tag{23}$$

引理

假设$\{(x_1^k, x_2^k, y^k)\}$为交替方向乘子法产生一个迭代序列，那么，对任意的$k \geq 1$有

$$u^k \in \partial f_1(x_1^k),\ v^k \in \partial f_2(x_2^k), \tag{24}$$

$$\begin{aligned}\Phi_k - \Phi_{k+1} \geq & \min(\tau, 1+\tau-\tau^2)\rho\|A_2(x_2^k - x_2^{k+1})\|^2 \\ & + \min(1, 1+\tau^{-1}-\tau)\rho\|A_1 e_1^{k+1} + A_2 e_2^{k+1}\|^2.\end{aligned} \tag{25}$$

注：只有当$\tau \in \left(0, \frac{1+\sqrt{5}}{2}\right)$时，(25) 式中不等号右侧的项才为非负.

Proof.

先证明(24)式的两个结论. 根据交替方向乘子法的迭代过程，对x_1^{k+1}我们有

$$0 \in \partial f_1(x_1^{k+1}) + A_1^{\mathrm{T}} y^k + \rho A_1^{\mathrm{T}}(A_1 x_1^{k+1} + A_2 x_2^k - b).$$

将$y^k = y^{k+1} - \tau\rho(A_1 x_1^{k+1} + A_2 x_2^{k+1} - b)$代入上式，消去$y^k$就有

$$-A_1^{\mathrm{T}}\Big(y^{k+1} + (1-\tau)\rho(A_1 x_1^{k+1} + A_2 x_2^{k+1} - b) + \rho A_2(x_2^k - x_2^{k+1})\Big) \in \partial f_1(x_1^{k+1}).$$

根据u^k的定义自然有$u^k \in \partial f_1(x_1^k)$ ($注意代回 b = A_1 x_1^* + A_2 x_2^*$). 类似地，对x_2^{k+1}我们有

$$0 \in \partial f_2(x_2^{k+1}) + A_2^{\mathrm{T}} y^k + \rho A_2^{\mathrm{T}}(A_1 x_1^{k+1} + A_2 x_2^{k+1} - b),$$

同样利用y^k的表达式消去y^k，得到

$$-A_2^{\mathrm{T}}\Big(y^{k+1} + (1-\tau)\rho(A_1 x_1^{k+1} + A_2 x_2^{k+1} - b)\Big) \in \partial f_2(x_2^{k+1}).$$

Proof.

根据v^k的定义自然有$v^k \in \partial f_2(x_2^k)$.
接下来证明不等式(25). 首先根据(x_1^*, x_2^*, y^*)的最优性条件以及关系式(24),

$$u^{k+1} \in \partial f_1(x_1^{k+1}), \quad -A_1^{\mathrm{T}} y^* \in \partial f_1(x_1^*),$$
$$v^{k+1} \in \partial f_2(x_2^{k+1}), \quad -A_2^{\mathrm{T}} y^* \in \partial f_2(x_2^*).$$

根据凸函数的单调性,

$$\left\langle u^{k+1} + A_1^{\mathrm{T}} y^*, x_1^{k+1} - x_1^* \right\rangle \geq 0,$$
$$\left\langle v^{k+1} + A_2^{\mathrm{T}} y^*, x_2^{k+1} - x_2^* \right\rangle \geq 0.$$

将上述两个不等式相加，结合u^{k+1}, v^{k+1}的定义，并注意到恒等式

$$A_1 x_1^{k+1} + A_2 x_2^{k+1} - b = (\tau\rho)^{-1}(y^{k+1} - y^k) = (\tau\rho)^{-1}(e_y^{k+1} - e_y^k), \tag{26}$$

Proof.

我们可以得到

$$\begin{aligned}&\frac{1}{\tau\rho}\left\langle e_y^{k+1}, e_y^k - e_y^{k+1}\right\rangle - (1-\tau)\rho\|A_1x_1^{k+1} + A_2x_2^{k+1} - b\|^2\\&+\rho\left\langle A_2(x_2^{k+1} - x_2^k), A_1x_1^{k+1} + A_2x_2^{k+1} - b\right\rangle\\&-\rho\left\langle A_2(x_2^{k+1} - x_2^k), A_2e_2^{k+1}\right\rangle \geq 0.\end{aligned} \tag{27}$$

不等式(27)的形式和不等式(25)还有一定差异，主要的差别就在

$$\rho\left\langle A_2(x_2^{k+1} - x_2^k), A_1x_1^{k+1} + A_2x_2^{k+1} - b\right\rangle$$

这一项上．我们接下来估计这一项的上界．

Proof.

引入新符号

$$\nu^{k+1} = y^{k+1} + (1-\tau)\rho(A_1x_1^{k+1} + A_2x_2^{k+1} - b),$$
$$M^{k+1} = (1-\tau)\rho\left\langle A_2(x_2^{k+1} - x_2^k), A_1x_1^k + A_2x_2^k - b\right\rangle,$$

则$-A_2^{\mathrm{T}}\nu^{k+1} \in \partial f_2(x_2^{k+1})$以及$-A_2^{\mathrm{T}}\nu^k \in \partial f_2(x_2^k)$. 再利用单调性知

$$\left\langle -A_2^{\mathrm{T}}(\nu^{k+1} - \nu^k), x_2^{k+1} - x_2^k\right\rangle \geq 0. \tag{28}$$

根据这些不等式关系我们最终得到

$$\begin{aligned}
&\rho\left\langle A_2(x_2^{k+1} - x_2^k), A_1x_1^{k+1} + A_2x_2^{k+1} - b\right\rangle \\
&= (1-\tau)\rho\left\langle A_2(x_2^{k+1} - x_2^k), A_1x_1^{k+1} + A_2x_2^{k+1} - b\right\rangle \\
&\quad + \left\langle A_2(x_2^{k+1} - x_2^k), y^{k+1} - y^k\right\rangle \\
&= M^{k+1} + \left\langle \nu^{k+1} - \nu^k, A_2(x_2^{k+1} - x_2^k)\right\rangle \leq M^{k+1}.
\end{aligned}$$

Proof.

估计完这一项之后，不等式(27)可以放缩成

$$\frac{1}{\tau\rho}\left\langle e_y^{k+1}, e_y^k - e_y^{k+1}\right\rangle - (1-\tau)\rho\|A_1x_1^{k+1} + A_2x_2^{k+1} - b\|^2$$
$$+ M^{k+1} - \rho\left\langle A_2(x_2^{k+1} - x_2^k), A_2e_2^{k+1}\right\rangle \geq 0.$$

上式中含有内积项，利用恒等式

$$\langle a, b\rangle = \frac{1}{2}(\|a\|^2 + \|b\|^2 - \|a-b\|^2) = \frac{1}{2}(\|a+b\|^2 - \|a\|^2 - \|b\|^2),$$

进一步得到

$$\frac{1}{\tau\rho}(\|e_y^k\|^2 - \|e_y^{k+1}\|^2) - (2-\tau)\rho\|A_1x_1^{k+1} + A_2x_2^{k+1} - b\|^2$$
$$+ 2M^{k+1} - \rho\|A_2(x_2^{k+1} - x_2^k)\|^2 - \rho\|A_2e_2^{k+1}\|^2 + \rho\|A_2e_2^k\|^2 \geq 0. \tag{29}$$

Proof.

此时除了M^{k+1}中的项，(29)中的其他项均在不等式(25)中出现. 由于M^{k+1}的符号和τ的取法有关，下面我们针对τ的两种取法进行讨论.
情形一　$\tau \in (0,1]$，此时$M^{k+1} \geq 0$，根据基本不等式，

$$\begin{aligned} & 2\left\langle A_2(x_2^{k+1}-x_2^k), A_1x_1^k + A_2x_2^k - b\right\rangle \\ \leq & \|A_2(x_2^{k+1}-x_2^k)\|^2 + \|A_1x_1^k + A_2x_2^k - b\|^2. \end{aligned}$$

代入不等式(29)得到

$$\begin{aligned} & \frac{1}{\tau\rho}\|e_y^k\|^2 + \rho\|A_2e_2^k\|^2 + (1-\tau)\rho\|A_1e_1^k + A_2e_2^k\|^2 \\ & - \left[\frac{1}{\tau\rho}\|e_y^{k+1}\|^2 + \rho\|A_2e_2^{k+1}\|^2 + (1-\tau)\rho\|A_1e_1^{k+1} + A_2e_2^{k+1}\|^2\right] \\ \geq & \rho\|A_1x_1^{k+1} + A_2x_2^{k+1} - b\|^2 + \tau\rho\|A_2(x_2^{k+1}-x_2^k)\|^2. \end{aligned} \tag{30}$$

Proof.

情形二 $\tau > 1$，此时$M^{k+1} < 0$，根据基本不等式，

$$\begin{aligned}
&-2\left\langle A_2(x_2^{k+1}-x_2^k), A_1x_1^k + A_2x_2^k - b\right\rangle \\
\leq &\tau\|A_2(x_2^{k+1}-x_2^k)\|^2 + \frac{1}{\tau}\|A_1x_1^k + A_2x_2^k - b\|^2.
\end{aligned}$$

同样代入不等式(29)可以得到

$$\begin{aligned}
&\frac{1}{\tau\rho}\|e_y^k\|^2 + \rho\|A_2e_2^k\|^2 + \left(1-\frac{1}{\tau}\right)\rho\|A_1e_1^k + A_2e_2^k\|^2 \\
&-\left[\frac{1}{\tau\rho}\|e_y^{k+1}\|^2 + \rho\|A_2e_2^{k+1}\|^2 + \left(1-\frac{1}{\tau}\right)\rho\|A_1e_1^{k+1} + A_2e_2^{k+1}\|^2\right] \\
\geq &\left(1+\frac{1}{\tau}-\tau\right)\rho\|A_1x_1^{k+1} + A_2x_2^{k+1} - b\|^2 \\
&+(1+\tau-\tau^2)\rho\|A_2(x_2^{k+1}-x_2^k)\|^2.
\end{aligned} \tag{31}$$

□

定理

在假设的条件下，进一步假定A_1, A_2列满秩. 如果$\tau \in \left(0, \frac{1+\sqrt{5}}{2}\right)$，则序列$\{(x_1^k, x_2^k, y^k)\}$收敛到原始问题的一个*KKT*对.

Proof.

前证引理表明Φ_k都是有界列，根据Φ_k的定义(23)可知

$$\|e_y^k\|, \quad \|A_2 e_2^k\|, \quad \|A_1 e_1^k + A_2 e_2^k\|$$

均有界. 根据不等式

$$\|A_1 e_1^k\| \leq \|A_1 e_1^k + A_2 e_2^k\| + \|A_2 e_2^k\|,$$

可以进一步推出$\{\|A_1 e_1^k\|\}$也是有界序列. 注意到$A_1^{\mathrm{T}} A_1 \succ 0, A_2^{\mathrm{T}} A_2 \succ 0$，因此以上有界性也等价于$\{(x_1^k, x_2^k, y^k)\}$是有界序列.

Proof.

另一个直接结果是无穷级数

$$\sum_{k=0}^{\infty} \|A_1 e_1^k + A_2 e_2^k\|^2, \ \sum_{k=0}^{\infty} \|A_2(x_2^{k+1} - x_2^k)\|^2$$

都是收敛的，这表明

$$\begin{aligned} \|A_1 e_1^k + A_2 e_2^k\| = \|A_1 x_1^k + A_2 x_2^k - b\| \to 0, \\ \|A_2(x_2^{k+1} - x_2^k)\| \to 0. \end{aligned} \tag{32}$$

利用这些结果我们就可以推导收敛性了．首先证明迭代点子列的收敛性．由于$\{(x_1^k, x_2^k, y^k)\}$是有界序列，因此它存在一个收敛子列，设

$$(x_1^{k_j}, x_2^{k_j}, y^{k_j}) \to (x_1^\infty, x_2^\infty, y^\infty).$$

由(23)式中的u^k, v^k的定义及(32)式可得$\{u^k\}$与$\{v^k\}$相应的子列也收敛.

Proof.

记

$$u^{\infty} \overset{\text{def}}{=\!=} \lim_{j\to\infty} u^{k_j} = -A_1^{\mathrm{T}} y^{\infty}, \quad v^{\infty} = \lim_{j\to\infty} v^{k_j} = -A_2^{\mathrm{T}} y^{\infty}. \tag{33}$$

从(24)式我们知道对于任意的$k \geq 1$，有$u^k \in \partial f_1(x_1^k)$, $v^k \in \partial f_2(x_2^k)$. 由次梯度映射的图像是闭集可知

$$-A_1 y^{\infty} \in \partial f_1(x_1^{\infty}), \quad -A_2 y^{\infty} \in \partial f_2(x_2^{\infty}).$$

由(32)的第一式可知

$$\lim_{j\to\infty} \|A_1 x_1^{k_j} + A_2 x_2^{k_j} - b\| = \|A_1 x_1^{\infty} + A_2 x_2^{\infty} - b\| = 0.$$

这表明$(x_1^{\infty}, x_2^{\infty}, y^{\infty})$是原始问题的一个KKT 对. 因此上述分析中的$(x_1^*, x_2^*, y^*)$均可替换为$(x_1^{\infty}, x_2^{\infty}, y^{\infty})$.

Proof.

注意到Φ_k是单调下降的，且对子列$\{\Phi_{k_j}\}$有

$$\begin{aligned}&\lim_{j\to\infty}\Phi_{k_j}\\ =&\lim_{j\to\infty}\left(\frac{1}{\tau\rho}\|e_y^{k_j}\|^2+\rho\|A_2e_2^{k_j}\|^2+\max\left\{1-\tau,1-\frac{1}{\tau}\right\}\rho\|A_1e_1^{k_j}+A_2e_2^{k_j}\|^2\right)\\ =&0.\end{aligned}$$

这说明

$$\|e_y^k\|\to 0,\quad \|A_2e_2^k\|\to 0,\quad \|A_1e_1^k+A_2e_2^k\|\to 0,$$

进一步有

$$0\le\limsup_{k\to\infty}\|A_1e_1^k\|\le\lim_{k\to\infty}\left(\|A_2e_2^k\|+\|A_1e_1^k+A_2e_2^k\|\right)=0.$$

注意到$A_1^{\mathrm{T}}A_1\succ 0$，$A_2^{\mathrm{T}}A_2\succ 0$，所以最终我们得到全序列收敛. □

多块ADMM收敛性反例

- 考虑最优化问题

$$\begin{aligned} \min \quad & 0, \\ \text{s.t.} \quad & A_1x_1 + A_2x_2 + A_3x_3 = 0, \end{aligned} \tag{34}$$

其中$A_i \in \mathbb{R}^3$, $i = 1, 2, 3$为三维空间中的非零向量，$x_i \in \mathbb{R}$, $i = 1, 2, 3$是自变量. 该问题实际上就是求解三维空间中的线性方程组，若A_1, A_2, A_3之间线性无关，则问题(34) 只有零解. 此时容易计算出最优解对应的乘子为$y = (0, 0, 0)^{\mathrm{T}}$.

- 增广拉格朗日函数为

$$L_\rho(x, y) = 0 + y^{\mathrm{T}}(A_1x_1 + A_2x_2 + A_3x_3) + \frac{\rho}{2}\|A_1x_1 + A_2x_2 + A_3x_3\|^2.$$

- 当固定x_2, x_3, y时，对x_1求最小可推出

$$A_1^{\mathrm{T}} y + \rho A_1^{\mathrm{T}}(A_1 x_1 + A_2 x_2 + A_3 x_3) = 0,$$

整理可得

$$x_1 = -\frac{1}{\|A_1\|^2}\left(A_1^{\mathrm{T}}\left(\frac{y}{\rho} + A_2 x_2 + A_3 x_3\right)\right).$$

可类似地计算x_2, x_3的表达式

- 因此多块交替方向乘子法的迭代格式可以写为

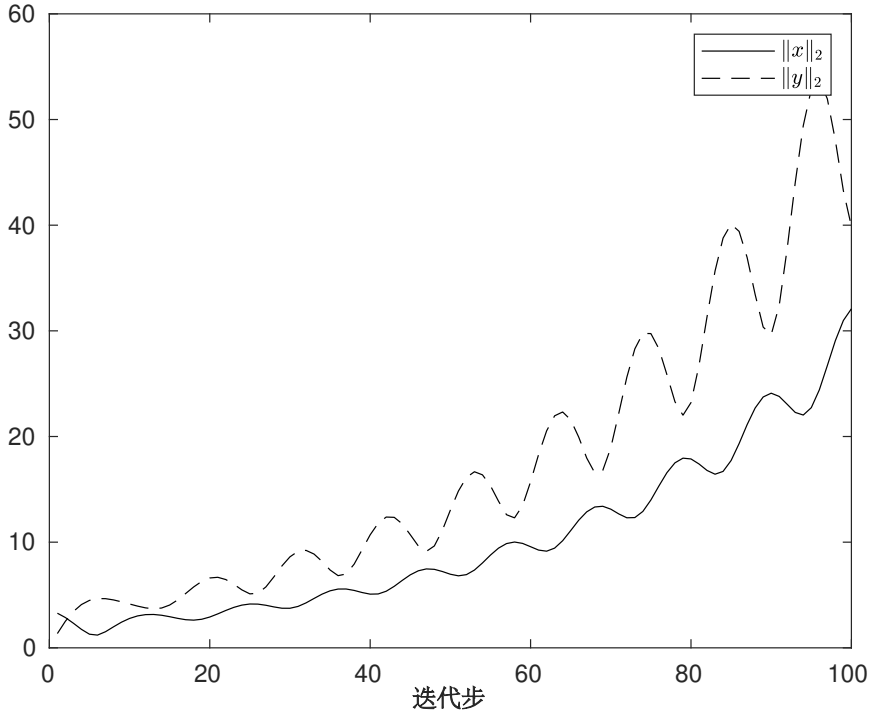
$$\begin{aligned}
x_1^{k+1} &= -\frac{1}{\|A_1\|^2} A_1^{\mathrm{T}}\left(\frac{y^k}{\rho} + A_2 x_2^k + A_3 x_3^k\right), \\
x_2^{k+1} &= -\frac{1}{\|A_2\|^2} A_2^{\mathrm{T}}\left(\frac{y^k}{\rho} + A_1 x_1^{k+1} + A_3 x_3^k\right), \\
x_3^{k+1} &= -\frac{1}{\|A_3\|^2} A_3^{\mathrm{T}}\left(\frac{y^k}{\rho} + A_1 x_1^{k+1} + A_2 x_2^{k+1}\right), \\
y^{k+1} &= y^k + \rho(A_1 x_1^{k+1} + A_2 x_2^{k+1} + A_3 x_3^{k+1}).
\end{aligned} \tag{35}$$

多块ADMM收敛性反例

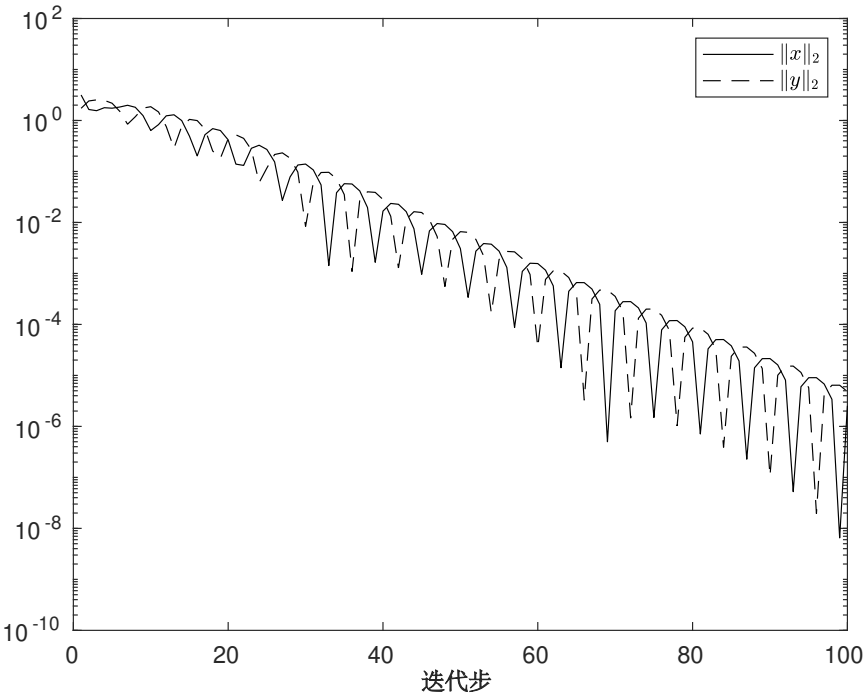
- 自变量初值初值选为$(1,1,1)$，乘子选为$(0,0,0)$. 选取A为

$$\widetilde{A}=\begin{bmatrix}1&1&2\\0&1&1\\0&0&1\end{bmatrix}\quad 或\quad \widehat{A}=\begin{bmatrix}1&1&1\\1&1&2\\1&2&2\end{bmatrix}.$$

- 下图记录了在不同A下x和y的ℓ_2范数随迭代的变化过程.



(a) 系数矩阵为$\widetilde{A}$



(b) 系数矩阵为$\widehat{A}$

Figure: 选取不同A时的数值结果

References

Douglas-Rachford method, ADMM, Spingarn's method

- J. E. Spingarn, *Applications of the method of partial inverses to convex programming: decomposition*, Mathematical Programming (1985)
- J. Eckstein and D. Bertsekas, *On the Douglas-Rachford splitting method and the proximal algorithm for maximal monotone operators*, Mathematical Programming (1992)
- P.L. Combettes and J.-C. Pesquet, *A Douglas-Rachford splitting approach to nonsmooth convex variational signal recovery*, IEEE Journal of Selected Topics in Signal Procesing (2007)
- S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, *Distributed optimization and statistical learning via the alternating direction method of multipliers* (2010)
- N. Parikh, S. Boyd, *Block splitting for distributed optimization* (2013)

image deblurring: the example is taken from
D. O'Connor and L. Vandenberghe, *Primal-dual decomposition by operator splitting and applications to image deblurring* (2014)

BCD:分块坐标下降法

陈士祥

中国科学技术大学

致谢：本教案部分参考北京大学文再文教授《最优化与建模》

本章需要掌握的知识点

1. 坐标下降法能求解的问题：如何拆分优化问题
2. 分块坐标下降法基本格式
3. 对偶坐标上升法：对偶问题的推导

回顾：对偶方法和ADMM

对偶方法操作在具有以下形式的问题的对偶上:

$$\min_x \quad f(x) \quad \text{受约束于} \quad Ax = b$$

对于凸函数f。对偶（次）梯度方法选择一个初始的$u(0)$，并重复以下步骤对于$k = 1, 2, 3, \ldots$:

$$x^{(k)} \in \arg\min_x \left[f(x) + (u^{(k-1)})^T Ax \right]$$

$$u^{(k)} = u^{(k-1)} + t_k (Ax^{(k-1)} - b)$$

其中t_k是以标准方式选择的步长。

- 优点: 第一步可分解性。
- 缺点: 收敛性质较差。

改进收敛性的方法: 通过增加拉格朗日函数，即在第一步加上项$\frac{\rho}{2}\|Ax - b\|^2$。执行分块最小化ADMM。

提纲

坐标下降法Coordinate descent

在前面的课程中，我们学习了求解可分问题的几个较为复杂的优化算法。
考虑

$$\min_{x\in\mathbb{R}^n} f(x)$$

坐标下降法是一种简单而高效的技术，也称为坐标逐次最小化算法。

$$x_1^{(k)} \in \arg\min_{x_1} f\left(x_1, x_2^{(k-1)}, x_3^{(k-1)}, \ldots, x_n^{(k-1)}\right)$$

$$x_2^{(k)} \in \arg\min_{x_2} f\left(x_1^{(k)}, x_2, x_3^{(k-1)}, \ldots, x_n^{(k-1)}\right)$$

$$x_3^{(k)} \in \arg\min_{x_3} f\left(x_1^{(k)}, x_2^{(k)}, x_3, \ldots, x_n^{(k-1)}\right)$$

$$\vdots$$

$$x_n^{(k)} \in \arg\min_{x_n} f\left(x_1^{(k)}, x_2^{(k)}, x_3^{(k)}, \ldots, x_n\right)$$

对于$k = 1, 2, 3, \ldots$，请注意，在我们解决$x_i^{(k)}$之后，我们将从那时起使用其新值！

假设1.

坐标下降法可以收敛到一个点x, 在该点处函数沿着各个坐标轴方向达到极小。[a]

[a]值得注意的是，该假设在函数非凸时不成立。

基于该假设，让我们考虑如下3个问题。

问题**1:** 给定一个凸且可微的函数$f:\mathbb{R}^n\to\mathbb{R}$, 如果我们在某一点$x$ 处，使得$f(x)$ 沿每个坐标轴都已最小化，那么我们是否已找到全局最小值?

即，是否有$f(x+\delta e_i)\geq f(x)$ 对所有δ,i，从而有$f(x)=\min_z f(z)$?

(这里$e_i=(0,\ldots,1,\ldots,0)\in\mathbb{R}^n$ 是第i 个标准基向量)

答案**:** 是的！证明如下: 各个坐标方向均有偏导为0，即

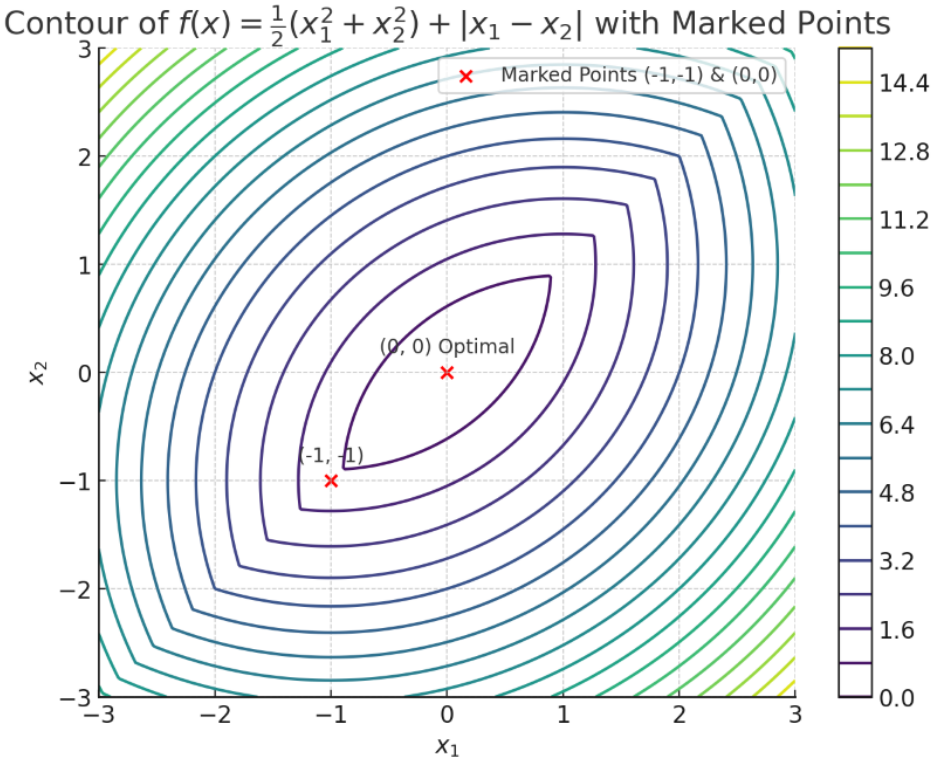
$$\nabla f(x)=\left(\frac{\partial f}{\partial x_1}(x),\ldots,\frac{\partial f}{\partial x_n}(x)\right)=0$$

问题**2:** 相同的问题，但对于不可微的凸函数f 如何?

答案**:** 不一定！看以下反例。

- 此时坐标下降法不收敛到最小值！
- 如何修正？见后面的交替近似线性极小化算法！

$$f(x) = \frac{1}{2}(x_1^2 + x_2^2) + |x_1 - x_2|$$



在点$(-1,-1)$处，如果沿着x轴或者y轴看，均为最小值，但不是全局最小。

问题**3:** 同样的问题，但现在$F(x) = f(x) + \sum_{i=1}^{n} r_i(x_i)$, 其中$f$ 是凸的且可微的，每个r_i 也是凸的.(这里的非光滑部分称为可分的)

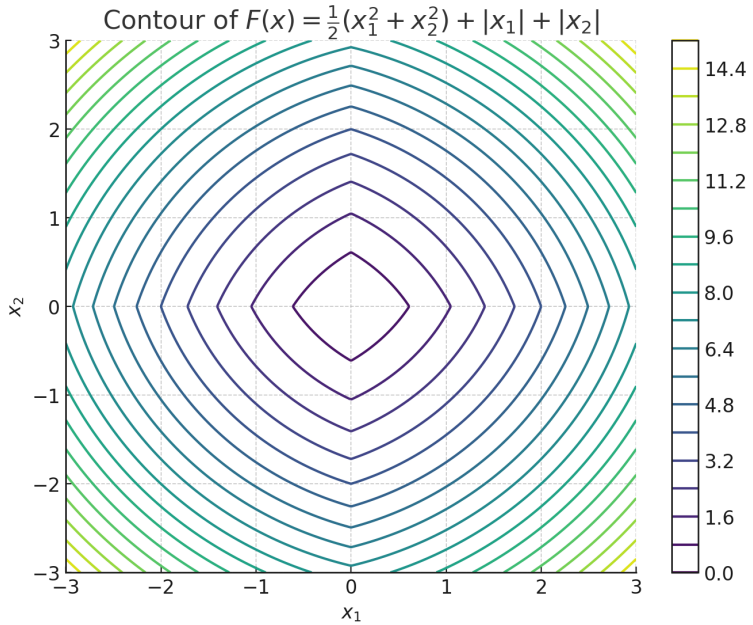
答案**:** 是的！ 此时坐标下降法收敛， 见Tseng. 2001: Convergence of a Block Coordinate Descent Method for Nondifferentiable Minimization.

证明: 由各个坐标轴最小，各个坐标方向导数均非负。对于任何y,

$$F(y) - F(x) \geq \nabla f(x)^T (y - x) + \sum_{i=1}^{n} [r_i(y_i) - r_i(x_i)] = \sum_{i=1}^{n} [\nabla_i f(x)(y_i - x_i) + r_i(y_i) - r_i(x_i)]$$

$$\geq \sum_{i=1}^{n} \left[(\nabla_i f(x) + r_i'(x_i))(y_i - x_i) \right] \geq 0$$

例： $F(x) = \frac{1}{2}(x_1^2 + x_2^2) + |x_1| + |x_2|$



更一般的分块问题形式

结合上面3个问题的回答，我们只考虑具有如下形式的问题：

$$\min_{x\in\mathcal{X}} \quad F(x_1,x_2,\cdots,x_s)=f(x_1,x_2,\cdots,x_s)+\sum_{i=1}^{s} r_i(x_i),$$

- $\mathcal{X}$是函数的可行域，自变量x拆分成s个变量块$x_1,x_2,\cdots,x_s$，每个变量块$x_i\in\mathbb{R}^{n_i}$.
- 函数f是关于x的可微函数，每个$r_i(x_i)$关于x_i是适当的闭凸函数，但不一定可微.
- 目标函数F的性质体现在f，每个r_i以及自变量的分块上. 通常情况下，f对于所有变量块x_i不可分，但单独考虑每一块自变量时，f有简单结构；r_i只和第i个自变量块有关，因此r_i在目标函数中是一个可分项.
- 求解该问题的难点在于如何利用分块结构处理不可分的函数f.

问题形式

- 分组LASSO 模型：参数$x = (x_1, x_2, \cdots, x_G) \in \mathbb{R}^p$ 可以分成G 组，且$\{x_i\}_{i=1}^G$中只有少数的非零向量.

$$\min_x \quad \frac{1}{2n}\|b - Ax\|_2^2 + \lambda \sum_{i=1}^{G} \sqrt{p_i}\|x_i\|_2.$$

- K-均值聚类问题的等价形式：

$$\begin{aligned} \min_{\Phi, H} \quad & \|A - \Phi H\|_F^2, \\ \text{s.t.} \quad & \Phi \in \mathbb{R}^{n\times k}, \text{每一行只有一个元素为1，其余为0}, \\ & H \in \mathbb{R}^{k\times p}. \end{aligned}$$

- 低秩矩阵恢复: 设$b \in \mathbb{R}^m$是已知的观测向量，$\mathcal{A}$是线性映射.

$$\min_{X,Y} \quad \frac{1}{2}\|\mathcal{A}(XY) - b\|_2^2 + \alpha\|X\|_F^2 + \beta\|Y\|_F^2,$$

其中$\alpha, \beta > 0$为正则化参数.

问题形式

- 非负矩阵分解：设$\mathcal{M}$是已知张量，考虑求解如下极小化问题：

$$\min_{A_1,A_2,\cdots,A_N\geq 0} \quad \frac{1}{2}\|\mathcal{M}-A_1\circ A_2\circ\cdots\circ A_N\|_F^2+\sum_{i=1}^{N}\lambda_i r_i(A_i),$$

其中“$\circ$”表示张量的外积运算.

- 字典学习：设$A\in\mathbb{R}^{m\times n}$为$n$个观测，每个观测的信号维数是$m$，现在我们要从$A$中学习出一个字典$D\in\mathbb{R}^{m\times k}$和系数矩阵$X\in\mathbb{R}^{k\times n}$：

$$\begin{aligned}\min_{D,X} \quad & \tfrac{1}{2n}\|DX-A\|_F^2+\lambda\|X\|_1,\\ \text{s.t.} \quad & \|D\|_F\leq 1.\end{aligned}$$

在这里自变量有两块，分别为D和X，此外对D还存在球约束$\|D\|_F\leq 1$.

挑战和难点

- 函数f关于变量全体一般是非凸的，这使得问题求解具有挑战性
- 应用在非凸问题上的算法收敛性不易分析，很多针对凸问题设计的算法通常会失效（例如后面第18页的反例）
- 目标函数的整体结构十分复杂，变量的更新需要很大计算量
- 目标: 发展一种更新方式简单且有全局收敛性（收敛到稳定点）的有效算法

变量划分

- 分块坐标下降法更新方式：按照$x_1, x_2, \cdots, x_s$的次序依次固定其他$(s-1)$块变量极小化F，完成一块变量的极小化后，它的值便立即被更新到变量空间中，更新下一块变量时将使用每个变量最新的值.
- 变量划分

$$\mathcal{X}_i^k = \{x \in \mathbb{R}^{n_i} \mid (x_1^k, \cdots, x_{i-1}^k, x, x_{i+1}^{k-1}, \cdots, x_s^{k-1}) \in \mathcal{X}\}.$$

- 辅助函数

$$f_i^k(x_i) = f(x_1^k, \cdots, x_{i-1}^k, x_i, x_{i+1}^{k-1}, \cdots, x_s^{k-1}),$$

其中x_j^k表示在第k次迭代中第j块自变量的值，函数f_i^k 表示在第k次迭代更新第i块变量时所需要考虑的目标函数的光滑部分.

变量更新方式

在每一步更新中，通常使用以下三种更新格式之一：

$$x_i^k = \underset{x_i \in \mathcal{X}_i^k}{\operatorname{argmin}} \left\{ f_i^k(x_i) + r_i(x_i) \right\}, \tag{1}$$

$$x_i^k = \underset{x_i \in \mathcal{X}_i^k}{\operatorname{argmin}} \left\{ f_i^k(x_i) + \frac{L_i^{k-1}}{2} \|x_i - x_i^{k-1}\|_2^2 + r_i(x_i) \right\}, \tag{2}$$

$$x_i^k = \underset{x_i \in \mathcal{X}_i^k}{\operatorname{argmin}} \left\{ \langle \hat{g}_i^k, x_i - \hat{x}_i^{k-1} \rangle + \frac{L_i^{k-1}}{2} \|x_i - \hat{x}_i^{k-1}\|_2^2 + r_i(x_i) \right\}, \tag{3}$$

- $L_i^k > 0$为常数
- 在更新格式(3)中，$\hat{x}_i^{k-1}$可以采用外推定义：

$$\hat{x}_i^{k-1} = x_i^{k-1} + \omega_i^{k-1}(x_i^{k-1} - x_i^{k-2}), \tag{4}$$

  其中$\omega_i^k \geq 0$为外推的权重，$\hat{g}_i^k \overset{\text{def}}{=\!=} \nabla f_i^k(\hat{x}_i^{k-1})$为外推点处的梯度.
- 在(4)式中取权重$\omega_i^k = 0$即可得到不带外推的更新格式，此时计算(3)等价于进行一次近似点梯度法的更新. 在(3)式使用外推是为了加快分块坐标下降法的收敛速度.

Algorithm 1 分块坐标下降法Block Coordinate descent(BCD)

初始化：选择两组初始点$(x_1^{-1}, x_2^{-1}, \cdots, x_s^{-1}) = (x_1^0, x_2^0, \cdots, x_s^0)$.
for $k = 1, 2, \cdots$ **do**
  for $i = 1, 2, \cdots$ **do**
    使用格式(1) 或(2) 或(3) 更新x_i^k.
  end for
  if 满足停机条件 **then**
    返回$(x_1^k, x_2^k, \cdots, x_s^k)$，算法终止.
  end if
end for

- 三种格式都有其适用的问题，特别是子问题是否可写出显式解
- 在每一步更新中，三种迭代格式对不同自变量块可以混合使用，不必仅仅局限于一种.

算法格式

- BCD算法的子问题可采用三种不同的更新格式，这三种格式可能会产生不同的迭代序列，可能会收敛到不同的解，坐标下降算法的数值表现也不相同.
- 格式(1)是最直接的更新方式，称为交替极小化算法，它严格保证了整个迭代过程的目标函数值是下降的. 然而由于f的形式复杂，子问题求解难度较大. 在收敛性方面，格式(1)在强凸问题上可保证目标函数收敛到极小值，但在非光滑（如第7页）非凸问题上（第18页）不一定收敛.
- 格式(2) (3) 则是对格式(1)的修正，称为交替近似极小化算法[1]，不保证迭代过程目标函数的单调性，但可以改善收敛性结果. 使用格式(2)可使得算法收敛性在函数F为非严格凸时有所改善.
- 格式(3)，称为交替近似线性极小化算法[2],实质上为目标函数的一阶泰勒展开近似，在一些测试问题上有更好的表现，可能的原因是使用一阶近似可以避开一些局部极小值点. 此外，格式(3)的计算量很小，比较容易实现. 而且收敛条件很弱，非凸非光滑（可分时）均可收敛。

[1] Attouch et al. 2013:Proximal alternating minimization and projection methods for nonconvex problems
[2] Bolte et al. 2014:Proximal alternating linearized minimization for nonconvex and nonsmooth problems

例子：二元二次函数

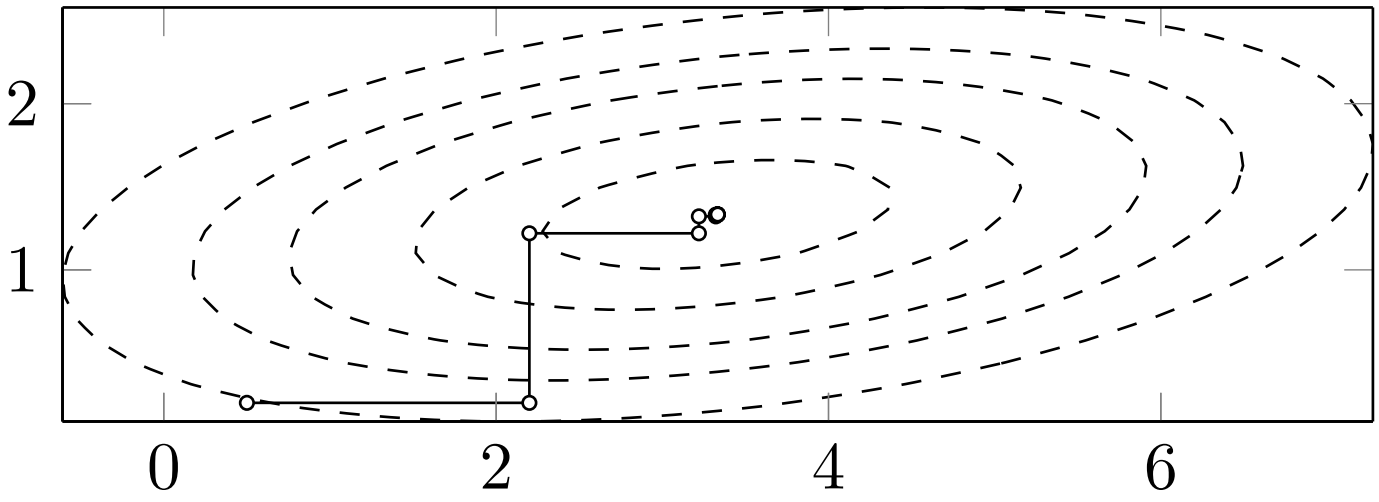
考虑二元二次函数的优化问题

$$\min \quad f(x,y) = x^2 - 2xy + 10y^2 - 4x - 20y.$$

故采用格式(1)的分块坐标下降法为

$$x^{k+1} = 2 + y^k, \quad y^{k+1} = 1 + \frac{x^{k+1}}{10}.$$

下图描绘了当初始点为 $(x,y) = (0.5, 0.2)$时的迭代点轨迹，可以看到在进行了7 次迭代后迭代点与最优解已充分接近.



迭代格式(1) 不收敛反例

值得注意的是，对于非凸函数$f(x)$，分块坐标下降法可能失效. Powell 在1973 年就给出了一个使用格式(1)但不收敛的例子:

$$F(x_1,x_2,x_3) = -x_1x_2 - x_2x_3 - x_3x_1 + \sum_{i=1}^{3}[(x_i-1)_+^2 + (-x_i-1)_+^2],$$

其中$(x_i-1)_+^2$的含义为先对(x_i-1)取正部再平方. 设$\varepsilon > 0$，初始点取为

$$x^0 = \left(-1-\varepsilon, 1+\frac{\varepsilon}{2}, -1-\frac{\varepsilon}{4}\right),$$

容易验证迭代序列满足

$$x^k = (-1)^k \cdot (-1,1,-1) + (-\frac{1}{8})^k \cdot \left(-\varepsilon, \frac{\varepsilon}{2}, -\frac{\varepsilon}{4}\right),$$

这个迭代序列有两个聚点$(-1,1,-1)$与$(1,-1,1)$，但这两个点都不是F的稳定点.

注1.

该例子失效的原因是，假设*1*不成立。如果要满足假设*1*,迭代格式(1)需要每个坐标方向的严格拟凸性[a]。

[a]Grippo, Luigi, and Marco Sciandrone. "On the convergence of the block nonlinear Gauss–Seidel method under convex constraints." Operations research letters 26.3 (2000): 127-136.

提纲

LASSO 问题求解

下面介绍如何使用分块坐标下降法来求解LASSO 问题

$$\min_x \mu\|x\|_1 + \frac{1}{2}\|Ax-b\|^2.$$

将自变量x记为$x=\left[x_i, \bar{x}_i^\top\right]^\top$，其中$\bar{x}_i$为$x$去掉第$i$个分量而形成的列向量．而相应地，矩阵$A$在第$i$块的更新记为$A=\begin{bmatrix} a_i & \bar{A}_i \end{bmatrix}$，其中$\bar{A}_i$为矩阵$A$去掉第$i$列而形成的矩阵．
在第i块的更新中考虑格式(1)。做替换$c_i = b - \bar{A}_i\bar{x}_i$，原问题等价于

$$\min_{x_i} \quad f_i(x_i) \overset{\text{def}}{=\!=} \mu|x_i| + \frac{1}{2}\|a_i\|^2 {x_i}^2 - a_i^{\mathrm{T}} c_i x_i.$$

可直接写出它的最小值点

$$x_i^k = \operatorname*{argmin}_{x_i} f_i(x_i) = \begin{cases} \frac{a_i^{\mathrm{T}} c_i - \mu_i}{\|a_i\|^2}, & a_i^{\mathrm{T}} c_i > \mu, \\ \frac{a_i^{\mathrm{T}} c_i + \mu_i}{\|a_i\|^2}, & a_i^{\mathrm{T}} c_i < -\mu, \\ 0, & \text{其他}. \end{cases}$$

考虑最基本的非负矩阵分解问题

$$\min_{X,Y\geq 0} \quad f(X,Y)=\frac{1}{2}\|XY-M\|_F^2.$$

可以计算梯度

$$\frac{\partial f}{\partial X}=(XY-M)Y^{\mathrm{T}},\quad \frac{\partial f}{\partial Y}=X^{\mathrm{T}}(XY-M).$$

注意到在格式(3)中，当$r_i(X)$为凸集示性函数时即是求解到该集合的投影，因此得到分块坐标下降法如下：

$$X^{k+1}=\max\{X^k-t_k^x(X^kY^k-M)(Y^k)^{\mathrm{T}},0\},$$
$$Y^{k+1}=\max\{Y^k-t_k^y(X^k)^{\mathrm{T}}(X^kY^k-M),0\},$$

其中t_k^x,t_k^y是步长，

$$\min_{D,X} \quad \frac{1}{2n}\|DX-A\|_F^2+\lambda\|X\|_1+\frac{\mu}{2}\|D\|_F^2.$$

- 当固定变量D时，考虑函数

$$f_D(X)=\frac{1}{2n}\|DX-A\|_F^2+\lambda\|X\|_1.$$

使用格式(3)．通过直接计算可得$f_D(X)$中光滑部分的梯度为

$$G=\frac{1}{n}D^{\mathrm{T}}(DX-A),$$

因此格式(3)等价于

$$X^{k+1}=\mathrm{prox}_{t_k\lambda\|\cdot\|_1}\left(X^k-\frac{t_k}{n}(D^k)^{\mathrm{T}}(D^kX^k-A)\right),$$

其中t_k为步长．

$$\min_{D,X} \quad \frac{1}{2n}\|DX - A\|_F^2 + \lambda\|X\|_1 + \frac{\mu}{2}\|D\|_F^2.$$

- 当固定变量X时，考虑函数

$$f_X(D) = \frac{1}{2n}\|DX - A\|_F^2 + \frac{\mu}{2}\|D\|_F^2.$$

使用格式(1)．计算关于D^{T}的梯度为

$$\nabla_{D^{\mathrm{T}}} f_X(D) = \frac{1}{n}X(X^{\mathrm{T}}D^{\mathrm{T}} - A^{\mathrm{T}}) + \mu D^{\mathrm{T}},$$

令梯度为零向量，可得

$$D = AX^{\mathrm{T}}(XX^{\mathrm{T}} + n\mu I)^{-1}.$$

因为$X \in \mathbb{R}^{k\times n}$，其中$k \ll n$，所以$XX^{\mathrm{T}}$是一个比较小的矩阵，可以方便地求出它的逆．故格式(1)等价于

$$D^{k+1} = A(X^{k+1})^{\mathrm{T}}(X^{k+1}(X^{k+1})^{\mathrm{T}} + n\mu I)^{-1}.$$

最大割问题的非凸松弛

最大割问题

$$
\begin{aligned}
\text{(半定松弛)} \quad & \min \quad \langle C, X \rangle, \\
& \text{s.t.} \quad X_{ii} = 1,\ i = 1, 2, \cdots, n, \\
& \qquad\ \ X \succeq 0. \\
\text{(非凸松弛)} \quad & \min \quad \left\langle C, V^{\mathrm{T}} V \right\rangle, \\
& \text{s.t.} \quad v_i \in \mathbb{R}^p,\ \|v_i\| = 1,\ i = 1, 2, \cdots, n, \\
& \qquad\ \ V = [v_1, v_2, \cdots, v_n].
\end{aligned}
$$

- 比较两种松弛方式可知，非凸松弛通过引入分解$X = V^{\mathrm{T}} V$并限制V的每一列的ℓ_2范数为1，将半定松弛中的X对角线元素为1以及X半正定的约束消去了.
- 这两个问题一般不等价，当p充分大时二者等价. 实际计算中通常选取一个较小的p.

矩阵V是按列分成n块的，考虑格式(1)为例，取定i，固定其余v_j

$$\mathrm{Tr}\left(\begin{bmatrix} C_{11} & \cdots & C_{1i} & \cdots & C_{1n} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ C_{i1} & \cdots & C_{ii} & \cdots & C_{in} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ C_{n1} & \cdots & C_{ni} & \cdots & C_{nn} \end{bmatrix} \begin{bmatrix} v_1^{\mathrm{T}}v_1 & \cdots & v_1^{\mathrm{T}}v_i & \cdots & v_1^{\mathrm{T}}v_n \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ v_i^{\mathrm{T}}v_1 & \cdots & v_i^{\mathrm{T}}v_i & \cdots & v_i^{\mathrm{T}}v_n \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ v_n^{\mathrm{T}}v_1 & \cdots & v_n^{\mathrm{T}}v_i & \cdots & v_n^{\mathrm{T}}v_n \end{bmatrix}\right),$$

根据以上矩阵分块示意图可知和v_i有关的部分为

$$C_{ii}v_i^{\mathrm{T}}v_i + \sum_{j\neq i}(C_{ij}+C_{ji})v_i^{\mathrm{T}}v_j.$$

由于约束$\|v_i\| = 1$，上式中第一项是常数。最终在第i步子问题是：

$$\min f_i(v_i) = \left(\sum_{j\neq i} C_{ji}v_j^{\mathrm{T}}\right)v_i, \mathrm{s.t.}\|v_i\| = 1.$$

其解为：$v_i = -\left(\sum_{j\neq i} C_{ji}v_j\right) \Big/ \left\|\sum_{j\neq i} C_{ji}v_j\right\|.$

提纲

问题定义

原问题：

$$\min_{w\in\mathbb{R}^d} P(w) = \frac{1}{n}\sum_{i=1}^{n}\phi_i(w^T x_i) + \frac{\lambda}{2}\|w\|^2$$

这里$x_1,\ldots,x_n$是$\mathbb{R}^d$给定的向量，$\phi_1,\ldots,\phi_n$是单变量凸函数, and $\lambda > 0$是给定的正则化参数。

例子：给定$\{y_i\}_{i=1}^n$, $y_i \in \mathbb{R}$是对应数据x_i的标签，

- **SVMs:** $\phi_i(a) = \max\{0, 1 - y_i a\}$ (L-Lipschitz)
- **Regularized logistic regression:** $\phi_i(a) = \log(1 + \exp(-y_i a))$
- **Ridge regression:** $\phi_i(a) = (a - y_i)^2$ (smooth)
- **Regression:** $\phi_i(a) = |a - y_i|$
- **Support vector regression:** $\phi_i(a) = \max\{0, |a - y_i| - \nu\}$

对偶问题

原问题：

$$\min_{w\in\mathbb{R}^d} P(w) = \frac{1}{n}\sum_{i=1}^{n}\phi_i(w^T x_i) + \frac{\lambda}{2}\|w\|^2$$

对偶问题表述为:

$$\max_{\alpha\in\mathbb{R}^n} D(\alpha) = \frac{1}{n}\sum_{i=1}^{n} -\phi_i^*(-\alpha_i) - \frac{\lambda}{2}\left\|\frac{1}{\lambda n}\sum_{i=1}^{n}\alpha_i x_i\right\|^2$$

其中$\phi_i^*(u) = \max_z(zu - \phi_i(z))$ 是ϕ_i 的共轭函数。

不同的共轭函数ϕ_i^*对应不同训练数据y_i. 他们关于对偶变量是坐标可分的。

结论：如果凸函数ϕ_i的梯度$\nabla\phi_i$是$L-$Lipschitz 连续的，那么ϕ_i^*是$1/L$强凸的。

对偶间隙

原问题：

$$\min_{w\in\mathbb{R}^d} P(w) = \frac{1}{n}\sum_{i=1}^{n}\phi_i(w^T x_i) + \frac{\lambda}{2}\|w\|^2$$

对偶问题：

$$\max_{\alpha\in\mathbb{R}^n} D(\alpha) = \frac{1}{n}\sum_{i=1}^{n} -\phi_i^*(-\alpha_i) - \frac{\lambda}{2}\left\|\frac{1}{\lambda n}\sum_{i=1}^{n}\alpha_i x_i\right\|^2$$

- 设原问题最优解为w^*, 对偶问题最优解为α^*
- 定义$w(\alpha) = \frac{1}{\lambda n}\sum_{i=1}^{n}\alpha_i x_i$，已知$w(\alpha^*) = w^*$.
- 由对偶定理可知：

$$P(w^*) = D(\alpha^*), \text{ 且}\forall w, \alpha, \quad P(w) \geq D(\alpha).$$

- 定义对偶间隙：

$$P(w(\alpha)) - D(\alpha)$$

SDCA算法[3]

SDCA算法步骤如下:

1. 随机选择一个训练样本i.
2. 对偶问题进行精确线搜索，找到$\Delta\alpha_i$:

$$\max -\phi_i^*\left(-(\alpha_{t-1,i}+\Delta\alpha_i)\right)-\frac{\lambda n}{2}\left\|w_{t-1}+\frac{1}{\lambda n}\Delta\alpha_i x_i\right\|^2$$

3. 更新对偶变量α_t 和原始变量w_t:

$$\alpha_t \leftarrow \alpha_{t-1}+\Delta\alpha_i e_i, \quad w_t \leftarrow w_{t-1}+\frac{1}{\lambda n}\Delta\alpha_i x_i$$

4. 当对偶间隙足够小时终止。

- 第一步选取指标i，可以按照顺序选取,此时该算法为，求解对偶问题的分块坐标上升法。
- 第2步的子问题，关于$\Delta\alpha_i$是强凸的，很多情况有闭式解。

[3]Shalev-Shwartz, Shai, and Tong Zhang. "Stochastic dual coordinate ascent methods for regularized loss minimization." Journal of Machine Learning Research 14.1 (2013).

该算法相对于随机梯度下降法(SGD)（后一讲）有如下优势：

- 如果原始问题是光滑的，SDCA在对偶间隙上比SGD有更快的线性收敛速度。
- 如果原始问题是非光滑的，SDCA的对偶间隙收敛速度是次线性的。

随机优化算法

陈士祥

中国科学技术大学

致谢：本教案部分参考北京大学文再文教授《最优化与建模》

本节要点

- 随机梯度法的基本格式，如何应用到有限求和形式问题上. 第24、28-30页
- 无放回采样的训练方式：29页
- 一般凸问题的SGD算法收敛,第48页

提纲

动机

- 机器学习一词通常用于描述任何涉及数据操作的任务。
- 机器学习的确切含义难以明确，因为相关的术语如数据挖掘、数据分析、人工智能或大数据也指涉到处理数据和学习过程。
- 本节中强调数据相关任务与优化之间的联系，尽管主要关注机器学习，但更广义上属于数据科学范畴。
- 机器学习的双重目标：
  1. 模式提取：从数据中提取模式，着眼于数据的统计特性。
  2. 信息应用：利用这些数据模式来进行推断或对未知数据进行预测。

机器学习：一般形式

- 关注于将输入映射到准确的输出：寻找合适的映射ϕ：输入$\rightarrow$输出。
- 强调通过风险最小化来近似预期风险。
- 损失函数有助于处理模型输出与真实输出之间的差异。

图像处理：

$\phi($  $)=$狗

自然语言处理：

$\phi($ “好好学习，天天向上” $)=$study hard and make progress every day

监督学习模型

- 假定(a, y)服从概率分布P，其中a为输入，y为标签.
- 假设我们数据集中的样本来自联合分布$p(a, y)$。我们寻求一个预测函数ϕ，使得期望风险较小，其中：

$$R(\phi) := P(\phi(a) \neq y) = \mathbb{E}[1(\phi(a) \neq y)]$$

这里$1(\cdot)$表示事件的指示函数。在实际中，我们很少知道数据的分布，只能访问分布的一个样本集$\{(a_i, y_i)\}_{i=1}^N$。在这种情况下，我们可以通过考虑定义为

$$R_N(\phi) := \frac{1}{N} \sum_{i=1}^{N} 1(\phi(a_i) \neq y_i)$$

的经验风险来量化我们在这个数据集上的预测效果。与期望风险函数不同，经验风险函数通常可以计算，因为它对应于可用的数据点。通过基于大数定律的论证，可以确保在样本足够多的情况下，经验风险与期望风险之间的差异可以以高概率被控制。

- 然而上述损失函数$R_N(\phi)$是非连续函数，难以优化。
- 我们引入$L(\cdot,\cdot)$表示损失函数，使得

$$1(\phi(a) \neq y) \approx L(\phi(a), y)$$

- L的例子如
  - ℓ_2损失函数

$$L(\hat{y}, y) = \frac{1}{2}\|\hat{y} - y\|_2^2$$

  - 若$\hat{y}, y \in \mathbb{R}^d$为概率分布（即各分量和为1的向量），则可定义互熵损失函数

$$L(\hat{y}, y) = \sum_{i=1}^{d} \hat{y}_i \log \frac{\hat{y}_i}{y_i}$$

实际中为了缩小目标函数的范围，需要将$\phi(\cdot)$参数化为$\phi(\cdot;x,b)$. ϕ参数化的例子如

- 线性函数

$$\phi(a) = a^\top x + b;$$

  线性函数的参数为x, b.

- 深度神经网络

$$\begin{aligned}
&\phi_0(a) = a \\
&\phi_l(a) = W_l\phi_{l-1}(a) + b_l, \\
&\phi(a) = \phi_L(a).
\end{aligned}$$

  其中$\phi(\cdot)$为非线性激活函数. 深度神经网络的参数为W_l与b_l.

例：线性模型

通过最小二乘法解决最佳拟合问题。给定数据$\{(a_i, y_i)\}_{i=1,\ldots,N}$，其中$a_i \in \mathbb{R}^d, y_i \in \mathbb{R}$

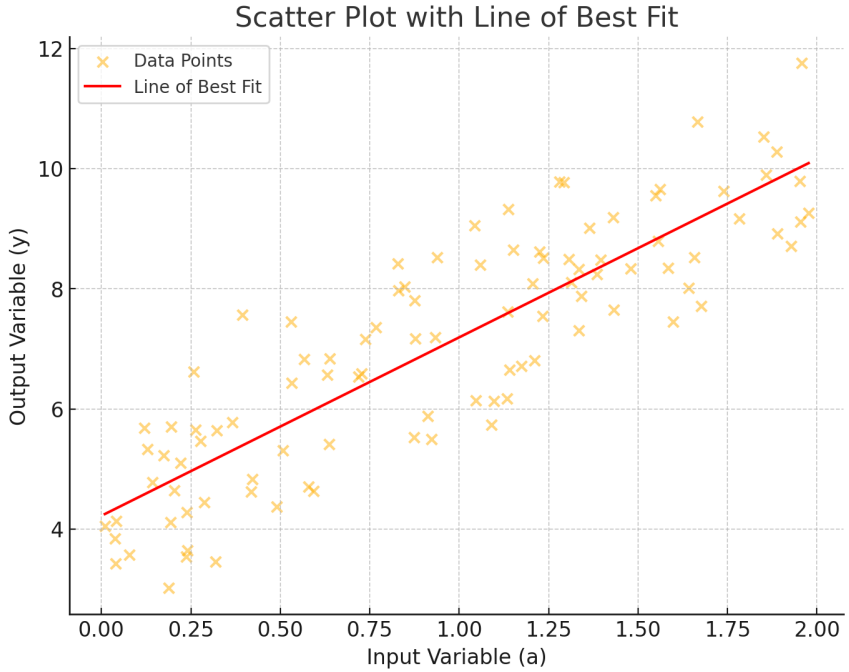


Figure: 线性模型数据散点图

- 我们定义线性模型使得对任何$a \in \mathbb{R}^d$，我们有$\phi(a) = a^T x + b$。
- 通过平方误差$(a_i^T x + b - y_i)^2$ 评估数据拟合度，并通过优化算法求解模型。损失函数为：

$$\frac{1}{N}\sum_{i=1}^{N}(a_i^T x + b - y_i)^2$$

然而，线性模型太简单了，复杂问题需要更精巧的模型。

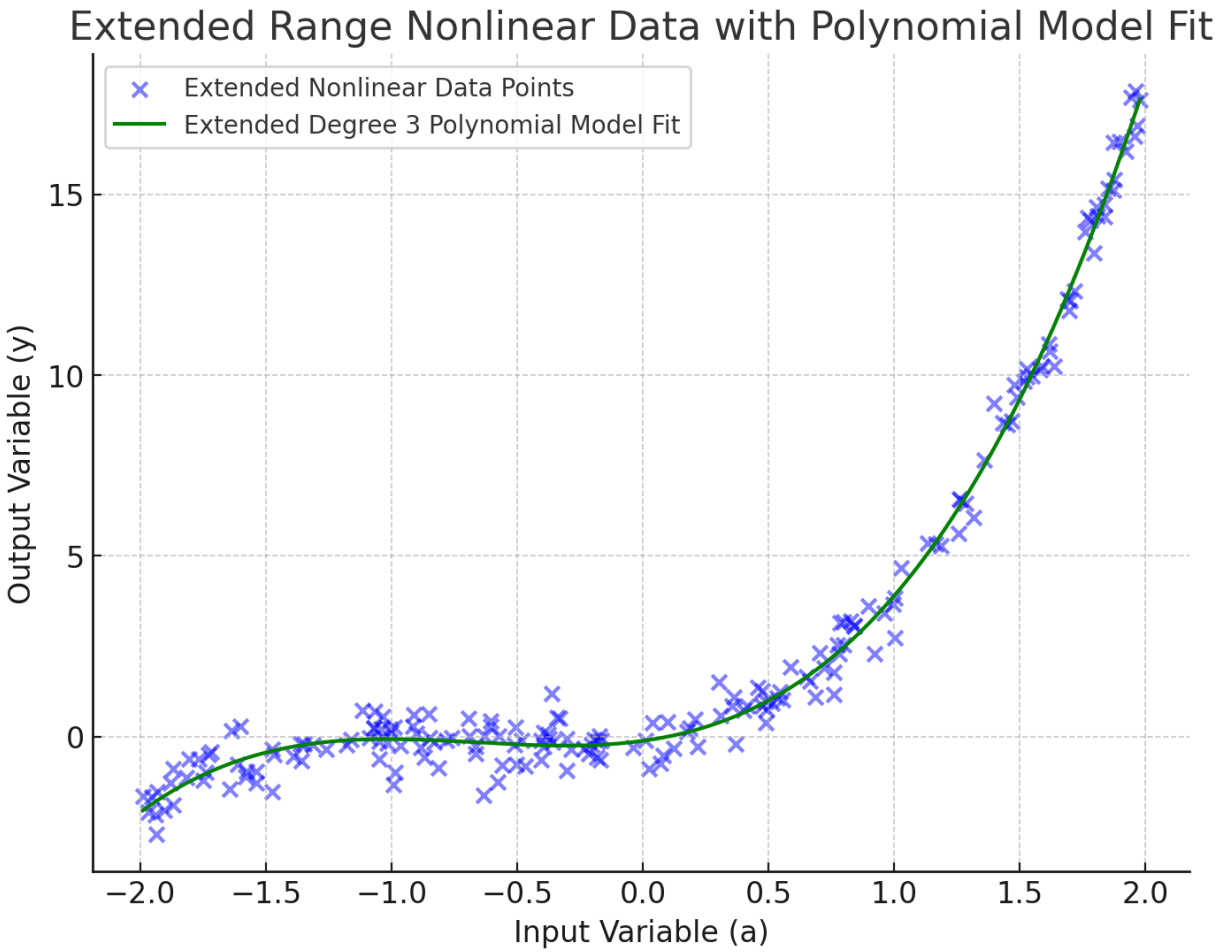
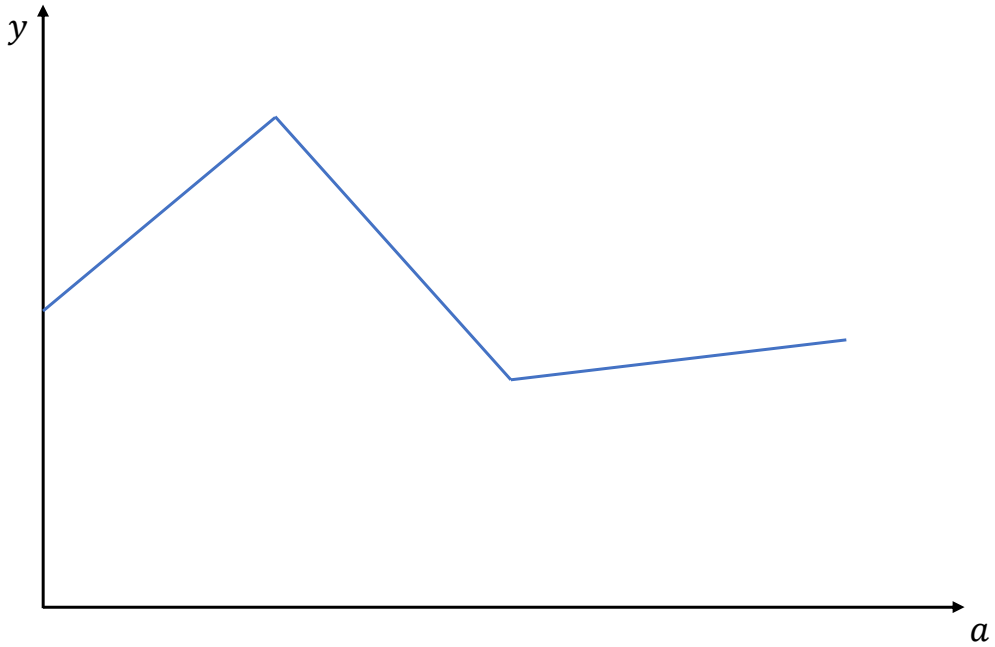


Figure: 非线性模型数据散点图

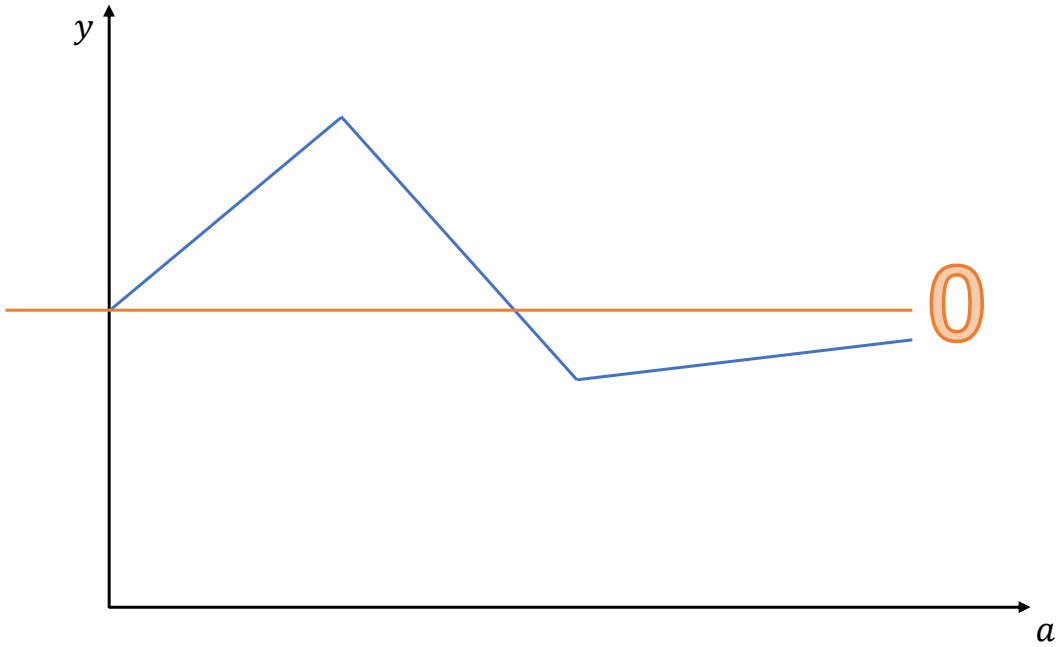
例：神经网络-分段线性函数逼近

有界区间上的分段线性函数 = 常数+一些类似于  函数的组合



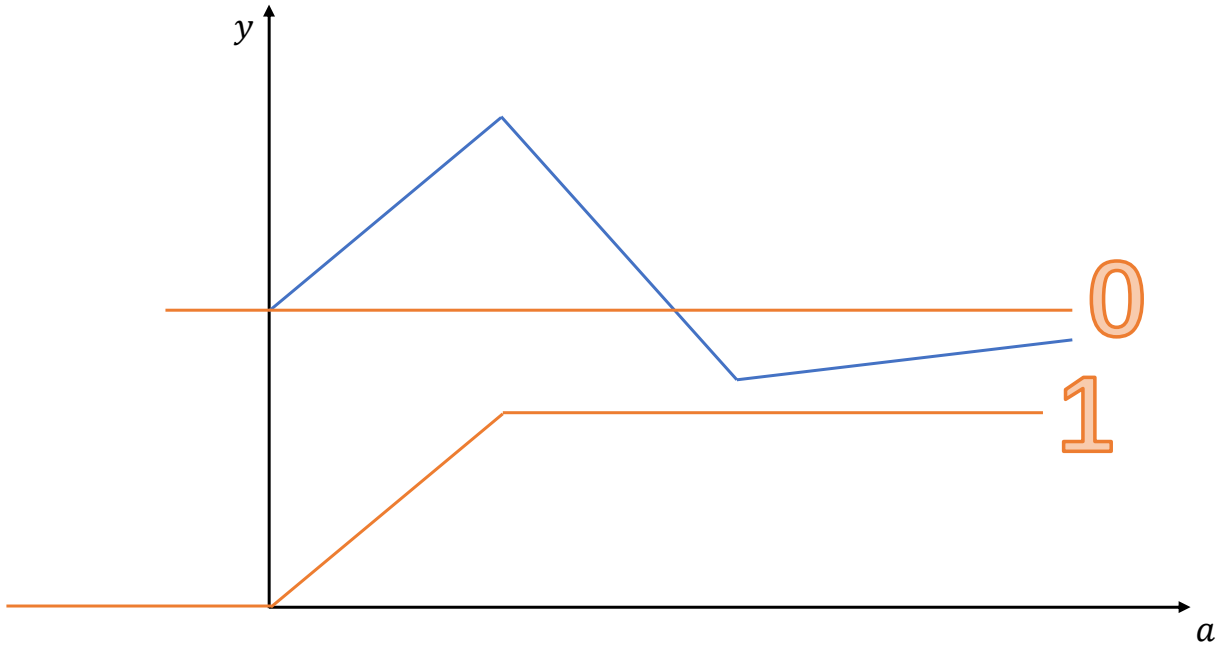
例：神经网络-分段线性函数逼近

有界区间上的分段线性函数 = 常数+一些类似于  函数的组合



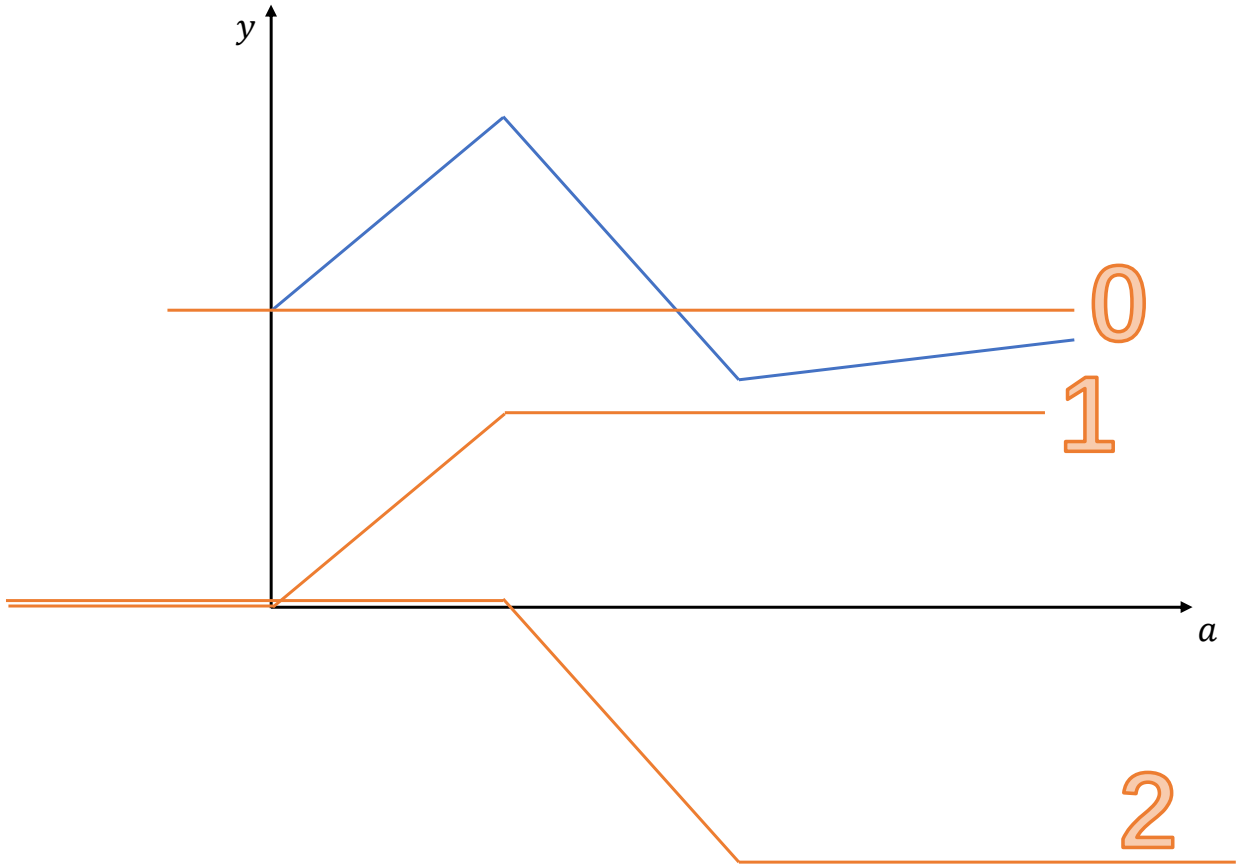
例：神经网络-分段线性函数逼近

有界区间上的分段线性函数 = 常数+一些类似于  函数的组合



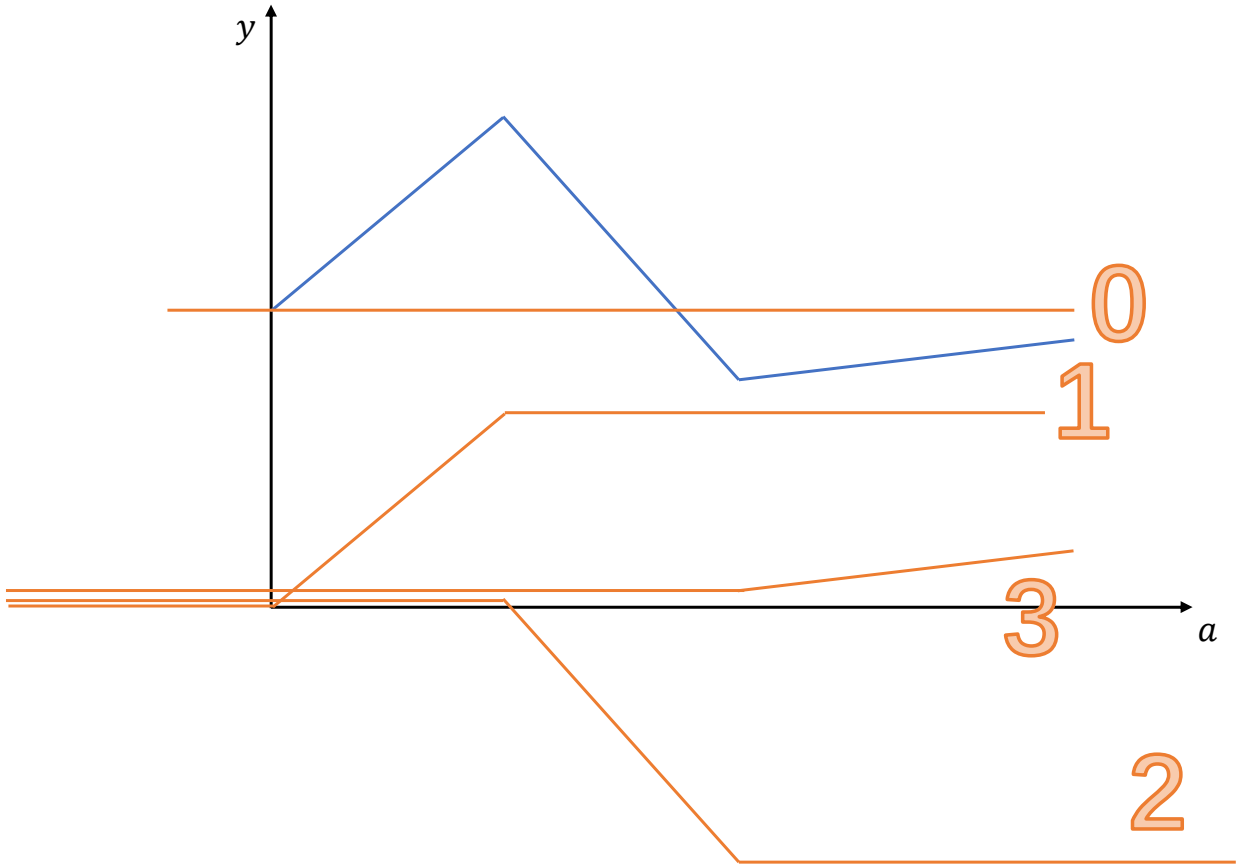
例：神经网络-分段线性函数逼近

有界区间上的分段线性函数 = 常数+一些类似于 函数的组合



例：神经网络-分段线性函数逼近

有界区间上的分段线性函数 = 常数+一些类似于 函数的组合

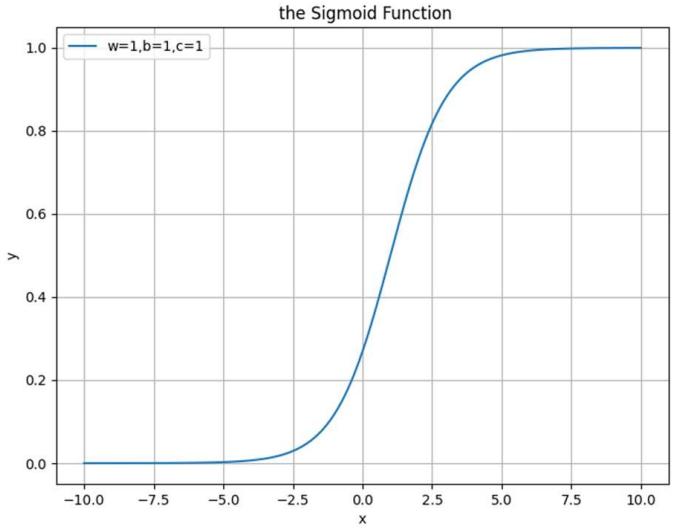


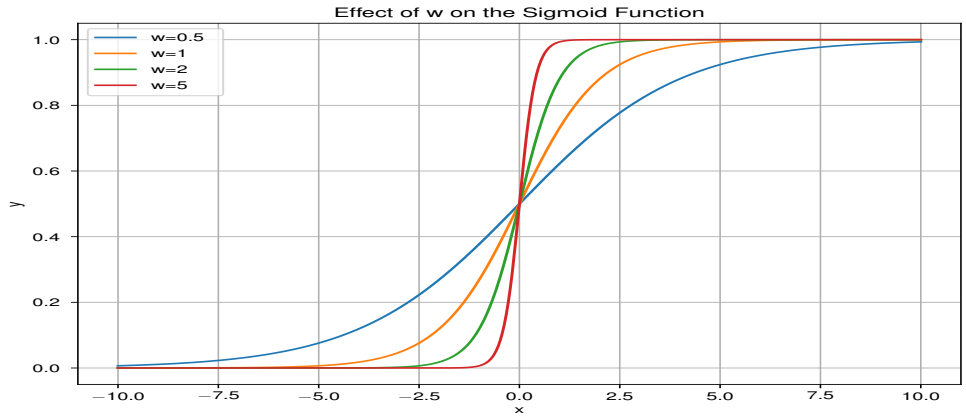
例：神经网络-Sigmoid 函数

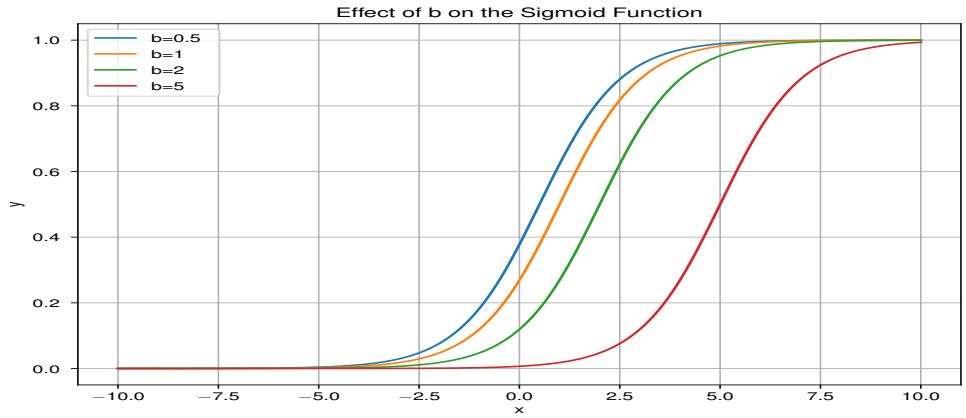
何种函数形如  ？

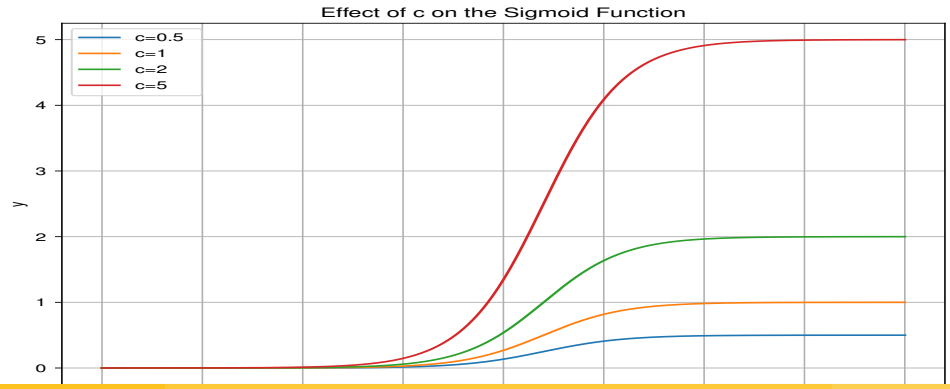
Sigmoid 函数复合线性变换：

$$y = c\frac{1}{1+\exp(-(wa+b))} = c\,\text{sigmoid}(wa+b)$$




Effect of w on the Sigmoid Function
w=0.5
w=1
w=2
w=5
1.0
0.8
0.6
0.4
0.2
0.0
y
−10.0
−7.5
−5.0
−2.5
0.0
2.5
5.0
7.5
10.0
x


Effect of b on the Sigmoid Function
b=0.5
b=1
b=2
b=5
1.0
0.8
0.6
0.4
0.2
0.0
y
−10.0
−7.5
−5.0
−2.5
0.0
2.5
5.0
7.5
10.0
x


Effect of c on the Sigmoid Function
c=0.5
c=1
c=2
c=5
5
4
3
2
1
0
y

神经网络：总结

一维线性模型：

$$\hat{y} = b + wa$$

更为复杂的分段线性模型：(n称之为神经元个数，为神经网络的“宽度”)

$$\hat{y} = b + \sum_{i=1}^{n} c_i \text{sigmoid}(w_i a + b_i)$$

多维线性模型：

$$\hat{y} = b + w^\top a, w \in \mathbb{R}^d$$

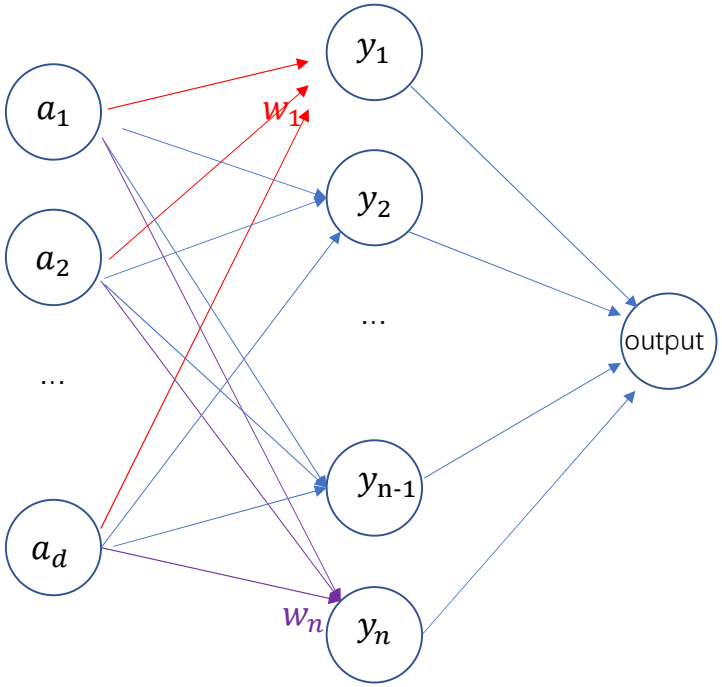
更为复杂的分段线性模型：

$$\hat{y} = b + \sum_{i=1}^{n} c_i \text{sigmoid}(w_i^\top a + b_i), w_i \in \mathbb{R}^d$$

定理 (Universal approximation Theorem; G. Cybenko,1989)

Let $C([0,1]^d)$ denote the set of all continous function $[0,1]^d \to \mathbb{R}$, let σ be any sigmoidal activation function then the finite sum of the form $f(x) = \sum_{i=1}^{n} c_i \sigma(w_i^\top a + b_i)$ is dense in $C([0,1]^d)$.

例：神经网络-单层神经网络



单层神经网络:

$$y_i = \text{sigmoid}(w_i^T a + b_i)$$

上述结论中sigmoid函数可以被替换为更一般的激活函数， 同时可以构造更“深层”的复合函数逼近函数。更深层的网络可以减小网络的“宽度”。上述过程可用下图表示:

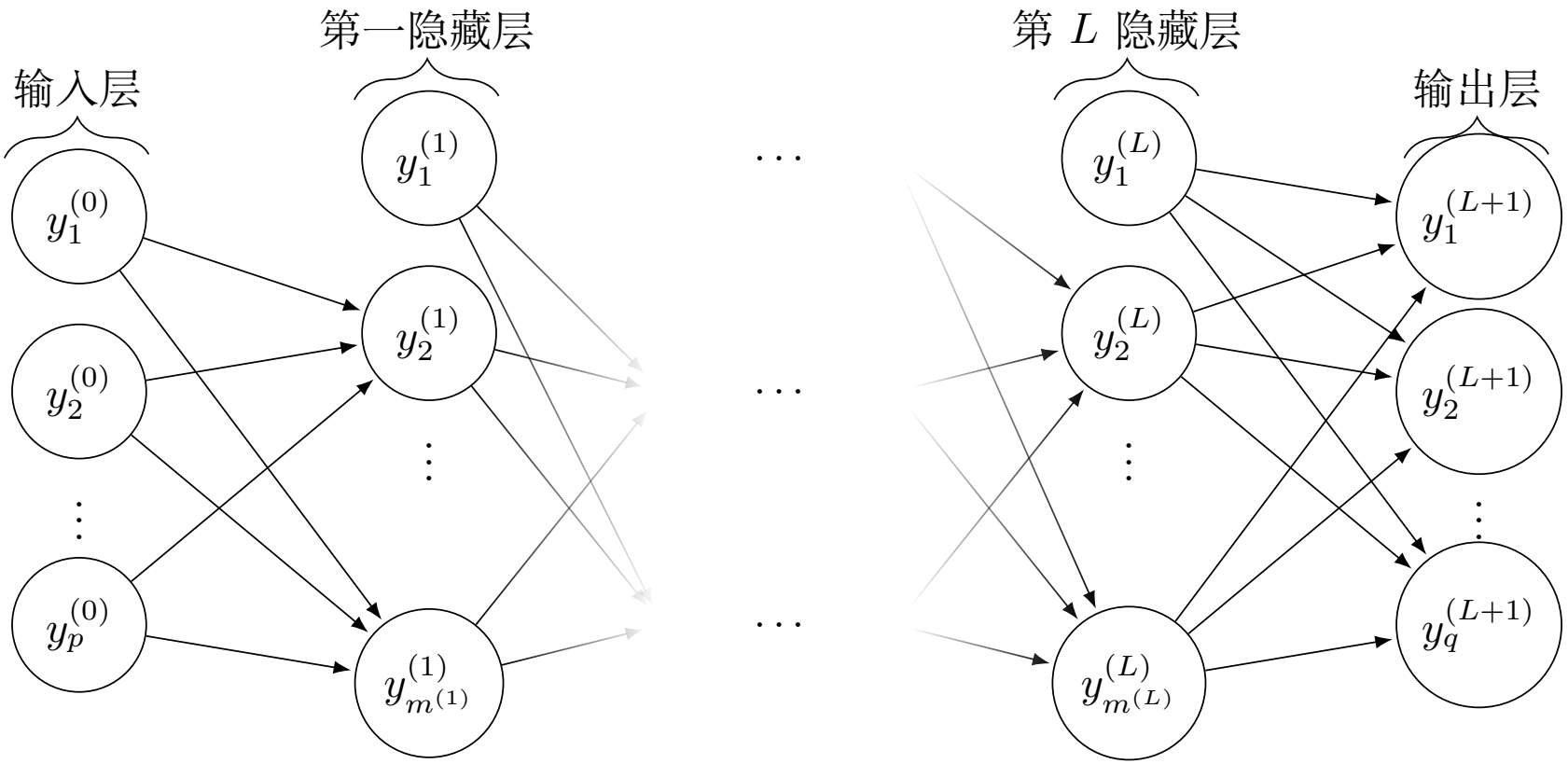


Figure: 带p个输入单元和q个输出单位的$(L+2)$层感知机的网络图， 第l 个隐藏层包含$m^{(l)}$个神经元.

神经网络的损失函数如下：

- 第l层，给定前一层的激活值$y^{(l-1)}$：

$$y^{(l)} = \phi(W^{(l)}, b^{(l)}; y^{(l-1)}) = \phi(W^{(l)} y^{(l-1)} + b^{(l)}) \tag{1}$$

- 也可使用ReLU作为激活函数，得到第l层的激活值

$$y^{(l)} = \text{ReLU}(W^{(l)} y^{(l-1)} + b^{(l)}) \tag{2}$$

- 对于最后一层，我们使用softmax函数得到类概率：

$$\text{softmax}(y_i^{(L+1)}) = \frac{e^{y_i^{(L+1)}}}{\sum_{j=1}^{q} e^{y_j^{(L+1)}}} \tag{3}$$

其中，q是输出类别的数量。

- 在所有数据上，计算网络预测与实际标签之间差异的交叉熵损失函数：

$$\text{CE}(y, \hat{y}) = -\frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{q} y_{i,j} \log(\text{softmax}(y_{i,j}^{(L)})) \tag{4}$$

神经网络模型具有以下两个特征：（1）参数维度非常大；（2）训练样本数量N非常大。表1 和表2 给出了一些经典的数据集和模型大小。模型规模和数据集大小制约了训练效率，大模型的训练有可能花费数天甚至数月时间。优化器的选择直接影响模型的训练效率。本课程中，我们将探讨常用的模型训练优化器。

Table: 大数据集数据量

数据集	数据集大小
Cifar10, Cifar100	60000 张图像
ImageNet	约1400万张图像
MS Coco	约33万张图像
GPT-3	45TB

Table: 大模型参数量

模型名称	参数量
VGG16	1.4亿
ResNet50	2500万
DenseNet-201	2000万
GPT-3	1750亿
GPT-4	1.8万亿

网络上爆料的GPT-4训练成本：一次的训练的成本为6300万美元，OpenAI训练GPT-4的FLOPS约为2.15e25，在大约25000个A100上训练了90到100天，利用率在32%到36%之间。

- 用经验风险来近似期望风险，即要求解下面的极小化问题：

$$\min_x \quad \frac{1}{N}\sum_{i=1}^{N} L(\phi(a_i;x),y_i) \approx \mathbb{E}_{(a,y)\sim\hat{P}}[L(\phi(a;x),y)]. \tag{5}$$

- 记

$$f_i(x) = L(\phi(a_i;x),y_i)$$

则只需考虑如下随机优化问题：

$$\min_{x\in\mathbb{R}^n} \quad f(x) \overset{\text{def}}{=\!=} \frac{1}{N}\sum_{i=1}^{N} f_i(x), \tag{6}$$

问题(6)也称为随机优化问题的有限和形式.

- 随机算法主要思路： 由于数据规模巨大，计算目标函数的梯度非常困难，但是通过采样的方式只计算部分样本的梯度来进行梯度下降，也能达到非常好的数值表现，而每步的运算量却得到了极大的减小.

提纲

梯度下降算法

- 下面为了讨论方便，先假设(6)中每一个$f_i(x)$是凸的、可微的.
- 可以运用梯度下降算法

$$x^{k+1} = x^k - \alpha_k \nabla f(x^k), \tag{7}$$

来求解原始的优化问题.

- 在迭代格式(7)中，

$$\nabla f(x^k) = \frac{1}{N} \sum_{i=1}^{N} \nabla f_i(x^k).$$

要计算这个梯度必须计算出所有的$\nabla f_i(x^k)$然后将它们相加.

- 然而在机器学习中，采集到的样本量是巨大的，因此计算$\nabla f(x^k)$需要非常大的计算量. 使用传统的梯度法求解机器学习问题并不是一个很好的做法. 更别提牛顿算法这样的二阶算法。

提纲

随机梯度下降算法(SGD)

- SGD的基本迭代格式为

$$x^{k+1} = x^k - \alpha_k \nabla f_{s_k}(x^k), \tag{8}$$

其中s_k是从$\{1, 2, \cdots, N\}$中随机等可能地抽取的一个样本，α_k称为步长. 在机器学习和深度学习领域中，更多的时候被称为学习率(learning rate).

- 通过对比(7)式和(8)式可知，随机梯度算法不去计算全梯度$\nabla f(x^k)$，而是从众多样本中随机抽出一个样本s_i，然后仅仅计算这个样本处的梯度$\nabla f_{s_k}(x^k)$，以此作为$\nabla f(x^k)$的近似.
- 在全梯度$\nabla f(x^k)$的表达式中含系数$1/N$，而迭代格式(8)中不含$1/N$. 这是因为我们要保证随机梯度的条件期望恰好是全梯度，即

$$\mathbb{E}_{s_k}[\nabla f_{s_k}(x^k)|x^k] = \nabla f(x^k).$$

小批量随机梯度法：Mini-batch SGD

- 实际计算中每次只抽取一个样本s_k的做法比较极端，常用的形式是小批量（mini-batch）随机梯度法.
- 每次迭代中，随机选择一个元素个数为B的集合$\mathcal{I}_k \subset \{1, 2, \cdots, N\}$，然后执行迭代格式

$$x^{k+1} = x^k - \frac{\alpha_k}{B} \sum_{s \in \mathcal{I}_k} \nabla f_s(x^k),$$

  其中B为$\mathcal{I}_k$中的元素个数. 该选取方式为有放回采样。
- 实际中，通常采用无放回采样方式选取小批量，该方案比有放回采样效果更好。
  1. 无放回采样也叫做随机洗牌（random shuffling），因为可以先将所有的数据集随机排序，即下标$1, 2, \ldots, N$有一个新的排序$\sigma_1, \ldots, \sigma_N$, 每批次训练数据按照新的排序，依次选择长度为$B$的子集。
  2. 过完所有的数据，叫做一个epoch。因此训练中1个epoch可以使得模型见完所有数据。
  3. 理论上也可以证明random shuffling(RS) 收敛速度更快。[1][2][3]

[1] Shamir, Ohad. "Without-replacement sampling for stochastic gradient methods." Advances in neural information processing systems 29 (2016). 证明RS不比无放回慢

[2] Gürbüzbalaban, Mert, Asu Ozdaglar, and Pablo A. Parrilo. "Why random reshuffling beats stochastic gradient descent." Mathematical Programming 186 (2021): 49-84.证明RS渐进地比无放回快

[3] Haochen, Jeff, and Suvrit Sra. "Random shuffling beats SGD after finite epochs." International Conference on Machine Learning. PMLR, 2019.证明有限步后RS比无放回快。

随机次梯度法

- 当$f_i(x)$是凸函数但不一定可微时，我们可以用$f_i(x)$的次梯度代替梯度进行迭代．这就是随机次梯度算法．
- 它的迭代格式为

$$x^{k+1} = x^k - \alpha_k g^k, \tag{9}$$

其中α_k为步长，$g^k \in \partial f_{s_k}(x^k)$为随机次梯度，其期望为真实的次梯度．

提纲

动量方法

- 传统的梯度法在问题比较病态时收敛速度非常慢，随机梯度下降法也有类似的问题. 为了克服这一缺陷，可以将重球法拓展到随机梯度法中。其思想是在算法迭代时一定程度上保留之前更新的方向，同时利用当前计算的梯度调整最终的更新方向.
- 动量方法的具体迭代格式如下：

$$v^{k+1} = \mu_k v^k - \alpha_k \nabla f_{s_k}(x^k), \tag{10}$$

$$x^{k+1} = x^k + v^{k+1}. \tag{11}$$

在计算当前点的随机梯度$\nabla f_{s_i}(x^k)$后，我们并不是直接将其更新到变量x^k上，而是将其和上一步更新方向v^k做线性组合来得到新的更新方向v^{k+1}.

动量方法

- 由动量方法迭代格式立即得出当$\mu_k = 0$时该方法退化成随机梯度下降法. 在动量方法中，参数μ_k的范围是$[0, 1)$，通常取$\mu_k \geq 0.5$, 神经网络训练中通常$\mu_k = 0.9$，其含义为迭代点带有较大惯性，每次迭代会在原始迭代方向的基础上做一个小的修正.
- 在普通的梯度法中，每一步迭代只用到了当前点的梯度估计，动量方法的更新方向还使用了之前的梯度信息.
- 当许多连续的梯度指向相同的方向时，步长就会很大，这从直观上看也是非常合理的.

图11比较了梯度法和动量方法的表现. 可以看到普通梯度法生成的点列会在椭圆的短轴方向上来回移动，而动量方法生成的点列更快收敛到了最小值点.

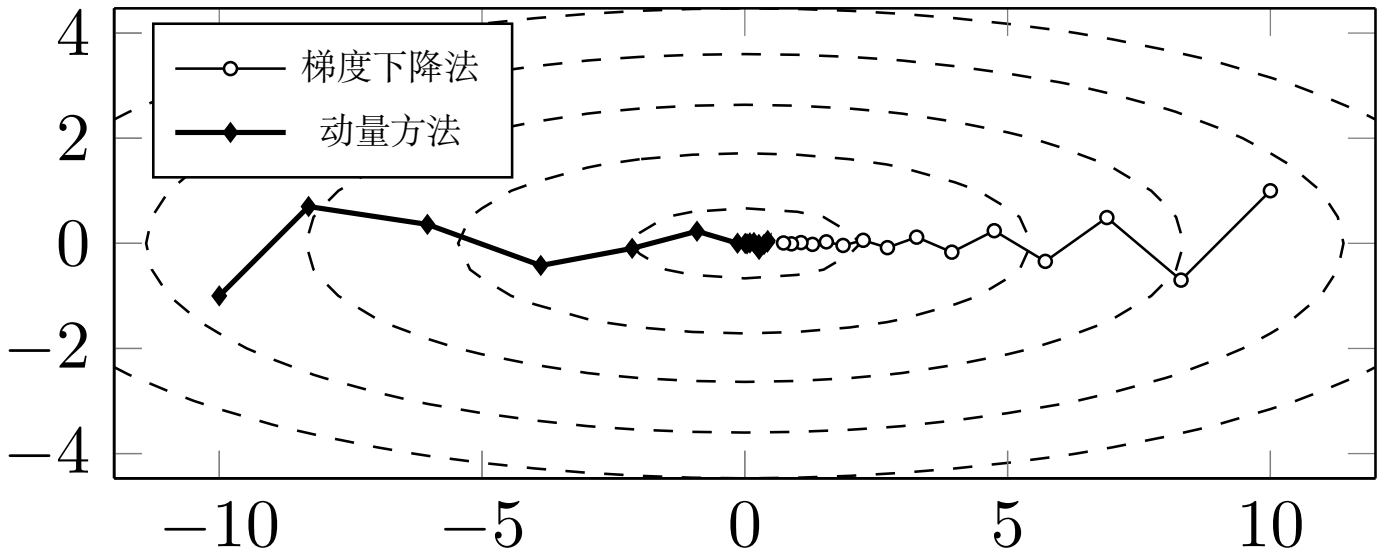


Figure: 动量方法在海瑟矩阵病态条件下的表现

动量方法

在梯度类算法中，很容易陷入局部最优，虽然有学者认为在高维情况中很难遇到局部最优，但动量法可能会是一个跳出局部最优的解决方案。
《三体：死神永生》

(a) AD 1453。君士坦丁堡和奥斯曼帝国战争

(b) 女巫狄奥伦娜得到四维时空碎片

(c) 取出了密封在圣索菲亚大教堂地基深处的圣杯，把圣杯换成了一串新鲜的葡萄

Figure: generated by DALL·E

高维度的情况我们无法计算！

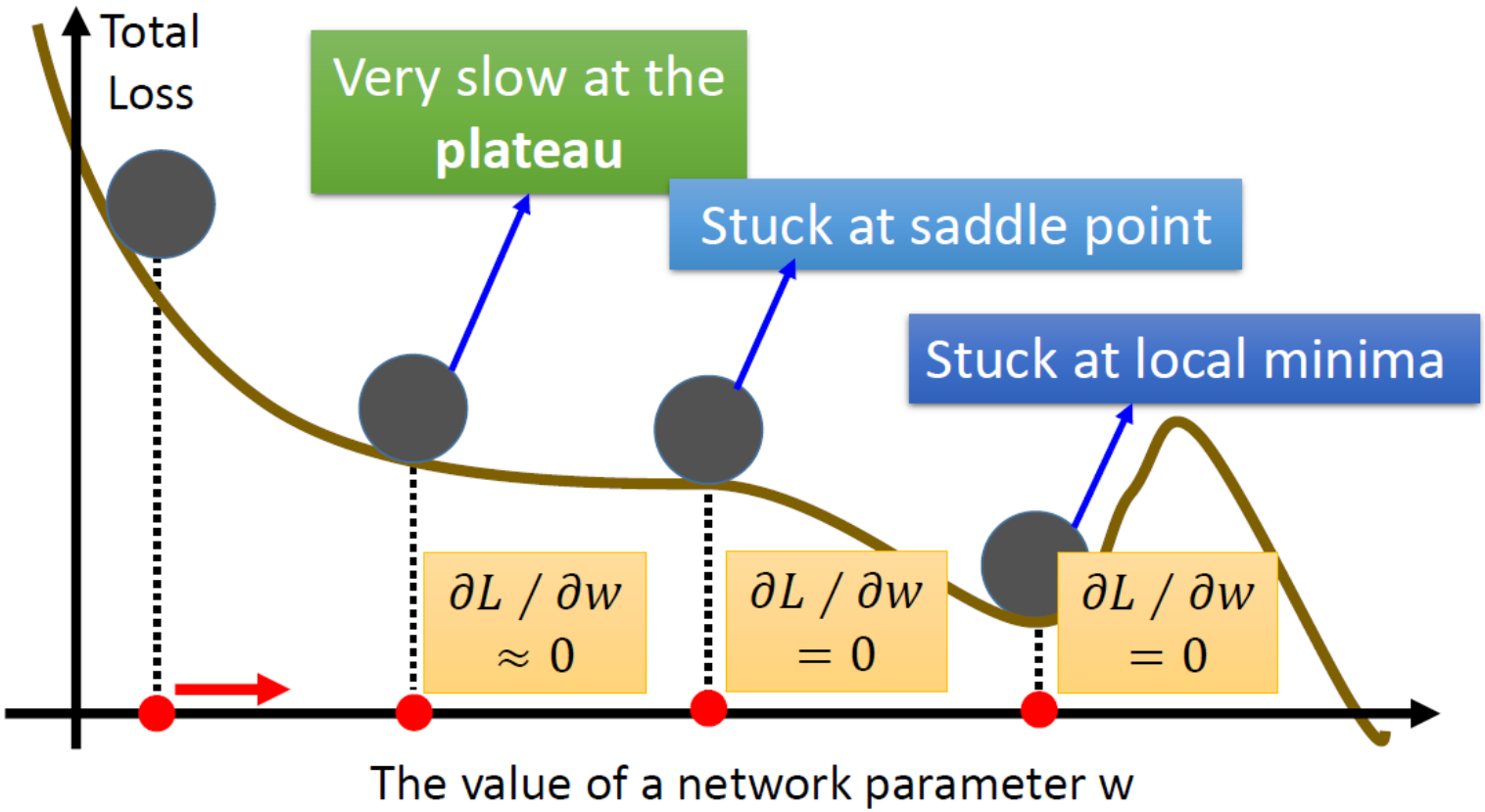


Figure: 图片来源：Hung-yi Lee , Machine Learning(2017)：Gradient Descent、Tips for training DNN

Nesterov加速算法

- 假设$f(x)$为光滑的凸函数. 针对凸问题的Nesterov加速算法为

$$y^{k+1} = x^k + \mu_k(x^k - x^{k-1})$$
$$x^{k+1} = y^k - \alpha_k \nabla f(y^k)$$

- 针对光滑问题的Nesterov加速算法迭代的随机版本为

$$y^{k+1} = x^k + \mu_k(x^k - x^{k-1}), \tag{12}$$
$$x^{k+1} = y^{k+1} - \alpha_k \nabla f_{s_k}(y^{k+1}), \tag{13}$$

其中$\mu_k = \frac{k-1}{k+2}$，步长α_k是一个固定值或者由线搜索确定.

- 可以看出，二者的唯一区别为随即版本将全梯度$\nabla f(y^k)$替换为随机梯度$\nabla f_{s_k}(y^{k+1})$.

Nesterov加速算法与动量方法的联系

- 若在第k步迭代引入速度变量$v^k = x^k - x^{k-1}$，再合并原始Nesterov 加速算法的两步迭代可以得到

$$x^{k+1} = x^k + \mu_k(x^k - x^{k-1}) - \alpha_k \nabla f_k(x^k + \mu_k(x^k - x^{k-1})).$$

- 定义有关v^{k+1}的迭代式

$$v^{k+1} = \mu_k v^k - \alpha_k \nabla f_k(x^k + \mu_k v^k),$$

- 于是得到关于x^k和v^k的等价迭代：

$$\begin{aligned} v^{k+1} &= \mu_k v^k - \alpha_k \nabla f_{s_k}(x^k + \mu_k v^k), \\ x^{k+1} &= x^k + v^{k+1}. \end{aligned}$$

- 二者的主要差别在梯度的计算上．Nesterov加速算法先对点施加速度的作用，再求梯度，可以理解为对标准动量方法做了校正．

提纲

提纲

假设

考虑最小化问题$\min_x F(x) := \mathbb{E}[f(x;\xi)]$，其中：

- F 是μ-强凸且L-光滑的。
- $g(x_t;\xi_t)$ 是给定$\{\xi_0,\ldots,\xi_{t-1}\}$ 的$\nabla F(x_t)$ 的无偏估计。
- 对于所有的x，随机梯度的方差是有界的：

$$\mathbb{E}\left[\|g(x;\xi)\|_2^2\right] \le \sigma_g^2 + c_g\|\nabla F(x)\|_2^2$$

固定步长的收敛性

定理**18.1 (**强凸问题的**SGD**收敛性；固定步长**):** 在上一页的假设下，如果$\eta_t \equiv \eta \leq \frac{1}{Lc_g}$，那么SGD的迭代序列$\{x_t\}_{t=1,\dots}$满足:

$$\mathbb{E}[F(x_t) - F(x^*)] \leq \frac{\eta L \sigma_g^2}{2\mu} + (1 - \eta\mu)^t [F(x_0) - F(x^*)] \tag{14}$$

请参考Bottou, Curtis, Nocedal '18 [4] (定理4.6) 以获得证明。

[4] Bottou, Léon, Frank E. Curtis, and Jorge Nocedal. "Optimization methods for large-scale machine learning." SIAM review 60.2 (2018): 223-311.

固定步长SGD的影响

$$\mathbb{E}[F(x_t) - F(x^*)] \leq \frac{\eta L \sigma_g^2}{2\mu} + (1 - \eta\mu)^t [F(x_0) - F(x^*)]$$

在一开始有快速（线性）收敛：

- 接近x^* 的某个邻域— 梯度的噪声阻碍了进一步的收敛。
- 当梯度计算无噪声（即$\sigma_g = 0$）时，它线性地收敛到最优点。
- 较小的步长η 提供了更好的收敛点。

解决方案：运行固定步长的SGD；当收敛停滞时，减小步长并继续SGD。

具有递减步长的SGD的收敛性

定理**18.2 (**强凸问题的**SGD**收敛性；递减步长**):** 假设F是μ-强凸的，并且满足条件(14)且$c_g = 0$。如果$\eta_t = \frac{\theta}{t+1}$对某个$\theta > \frac{1}{2\mu}$，那么SGD:

$$\mathbb{E}[\|x_t - x^*\|_2^2] \leq \frac{c_\theta}{t+1}$$

其中$c_\theta = \max\left\{\frac{2\theta^2\sigma_g^2}{2\mu\theta - 1}, \|x_0 - x^*\|_2^2\right\}$.

该定理说明使用步长$O(1/t)$时，收敛速度为$O(1/t)$.

定理18.2的证明:

使用SGD更新规则，我们有：

$$\|x_{t+1}-x^*\|_2^2 = \|x_t-\eta_t g(x_t;\xi_t)-x^*\|_2^2 = \|x_t-x^*\|_2^2-2\eta_t(x_t-x^*)^\top g(x_t;\xi_t)+\eta_t^2\|g(x_t;\xi_t)\|_2^2 \quad (15)$$

因为x_t 独立于ξ_t，应用双期望定理得到：

$$\begin{aligned}\mathbb{E}[(x_t-x^*)^\top g(x_t;\xi_t)] &= \mathbb{E}[\mathbb{E}[(x_t-x^*)^\top g(x_t;\xi_t)|\xi_1,\ldots,\xi_{t-1}]] \quad (16)\\ &= \mathbb{E}[(x_t-x^*)^\top \nabla F(x_t)]\end{aligned}$$

由强凸性质，

$$(x_t-x^*)^\top(\nabla F(x_t)-\nabla F(x^*)) \geq \mu\|x_t-x^*\|^2 \quad (17)$$

结合(15), (16) 和(17) 以及(14) (当$c_g=0$) 得到：

$$\mathbb{E}[\|x_{t+1}-x^*\|_2^2] \leq (1-2\mu\eta_t)\mathbb{E}[\|x_t-x^*\|_2^2]+\eta_t^2\sigma_g^2 \quad (18)$$

采用$\eta_t=\frac{\theta}{t+1}$ 并用归纳法完成证明（练习！）

SGD的强凸时的最优性

根据Nemirovski, Yudin '83[5], Agarwal 等'11[6] 和Raginsky, Rakhlin '11[7] 的研究：在最小化强凸函数时，没有算法在进行t 次查询到含噪声的一阶函数信息能够实现比$\frac{1}{t}$ 更好的精确度。

结论： 使用递减步长的$\eta_t \approx \frac{1}{t}$ 是最优的。

[5]"Problem complexity and method efficiency in optimization," A. Nemirovski, D. Yudin, Wiley, 1983.

[6]"Information-theoretic lower bounds on the oracle complexity of stochastic convex optimization," A. Agarwal, P. Bartlett, P. Ravikumar, M. Wainwright, IEEE Transactions on Information Theory, 2011.

[7]"Information-based complexity, feedback and dynamics in convex programming," M. Raginsky, A. Rakhlin, IEEE Transactions on Information Theory, 2011

凸问题的随机梯度方法

当我们失去强凸性时情况如何?

- 最小化$F(x) := \mathbb{E}[f(x;\xi)]$，其中$F$是凸的。
- 对于所有的x，有$\mathbb{E}[\|g(x;\xi)\|_2^2] \leq \sigma_g^2$。
- $g(x_t;\xi_t)$是给定$\{\xi_0,\ldots,\xi_{t-1}\}$的$\nabla F(x_t)$的无偏估计。

假设我们返回一个加权平均$\tilde{x}_t := \frac{\sum_{k=0}^t \eta_k x_k}{\sum_{j=0}^t \eta_j}$。

定理**18.3**：在第上一页的假设下，有：

$$\mathbb{E}[F(\tilde{x}_t) - F(x^*)] \leq \frac{\frac{1}{2}\mathbb{E}[\|x_0 - x^*\|_2^2] + \frac{1}{2}\sigma_g^2 \sum_{k=0}^{t} \eta_k^2}{\sum_{k=0}^{t} \eta_k}$$

如果$\eta_t \approx \frac{1}{\sqrt{t}}$，则：

$$\mathbb{E}[F(\tilde{x}_t) - F(x^*)] \approx \frac{\log t}{\sqrt{t}}$$

定理18.3 的证明

备注：这与次梯度方法的收敛性分析非常相似。
由F 的凸性，我们有$F(x) \geq F(x_t) + (x - x_t)^\top \nabla F(x_t)$。
这与(15)和(16)一起表明：

$$2\eta_k \mathbb{E}[F(x_k) - F(x^*)] \leq \mathbb{E}[\|x_k - x^*\|_2^2] - \mathbb{E}[\|x_{k+1} - x^*\|_2^2] + \eta_k^2 \sigma_g^2$$

对$k = 0, \ldots, t$ 求和得到：

$$\sum_{k=0}^{t} 2\eta_k \mathbb{E}[F(x_k) - F(x^*)] \leq \mathbb{E}[\|x_0 - x^*\|_2^2] - \mathbb{E}[\|x_{t+1} - x^*\|_2^2] + \sigma_g^2 \sum_{k=0}^{t} \eta_k^2 \leq \mathbb{E}[\|x_0 - x^*\|_2^2] + \sigma_g^2 \sum_{k=0}^{t} \eta_k^2$$

令$v_t = \frac{\eta_t}{\sum_{k=0}^{t} \eta_k}$，得到：

$$\sum_{k=0}^{t} v_k \mathbb{E}[F(x_k) - F(x^*)] \leq \frac{\frac{1}{2}\mathbb{E}[\|x_0 - x^*\|_2^2] + \frac{1}{2}\sigma_g^2 \sum_{k=0}^{t} \eta_k^2}{\sum_{k=0}^{t} \eta_k}$$

由F 的凸性，我们得出：

$$\mathbb{E}[F(\tilde{x}_t) - F(x^*)] \leq \frac{\frac{1}{2}\mathbb{E}[\|x_0 - x^*\|_2^2] + \frac{1}{2}\sigma_g^2 \sum_{k=0}^{t} \eta_k^2}{\sum_{k=0}^{t} \eta_k}$$

这证明了使用加权平均$\tilde{x}_t$ 可以在凸问题中实现有效的收敛。

随机梯度算法的收敛性

上述定理11.2表明对于强凸函数，随机梯度下降法的收敛速度可以达到$\mathcal{O}(1/K)$. 定理11.3表明非光滑凸函数有$\mathcal{O}(1/\sqrt{K})$.对于一般的凸函数随机梯度算法也有一定的收敛性，为此我们在下表比较随机算法和普通算法的复杂度.

Table: 梯度下降法的算法复杂度

	f凸(次梯度算法)	f可微强凸	f可微强凸且L-光滑
随机算法	$\mathcal{O}\left(\frac{1}{\varepsilon^2}\right)$	$\mathcal{O}\left(\frac{1}{\varepsilon}\right)$	$\mathcal{O}\left(\frac{1}{\varepsilon}\right)$
普通算法	$\mathcal{O}\left(\frac{N}{\varepsilon^2}\right)$	$\mathcal{O}\left(\frac{N}{\varepsilon}\right)$	$\mathcal{O}\left(N\ln\left(\frac{1}{\varepsilon}\right)\right)$

- 在前两种情况下，即f凸(次梯度算法)f可微强凸, SGD 比梯度法要更快。这是因为每次仅需计算一个样本的梯度，达到相同的收敛阶。
- 在f可微强凸且L-光滑时，如果求解精度不需太高，

$$\frac{1}{\varepsilon} < N\ln\left(\frac{1}{\varepsilon}\right)$$

时，SGD也比梯度法更快。该条件很容易在大数据场景下满足.

随机梯度法-2

陈士祥

中国科学技术大学

本节要点

- 了解Adam优化器的思想
- 理解SAG、SAGA、SVRG降低方差的原理
- 了解随机梯度法的收敛结果，相较于全梯度法的优势

Outline

AdaGrad

- 在一般的随机梯度法中，调参是一个很大的难点．我们希望算法能在运行的过程中，根据当前情况自发地调整参数．
- 对无约束光滑凸优化问题，点x是问题的解等价于该点处梯度为零向量．但梯度的每个分量收敛到零的速度是不同的．传统梯度算法只有一个统一的步长α_k来调节每一步迭代，它没有针对每一个分量考虑．
- 当梯度的某个分量较大时，可以推断出在该方向上函数变化比较剧烈，要用小步长；当梯度的某个分量较小时，在该方向上函数比较平缓，要用大步长．AdaGrad[1]就是根据这个思想设计的．

[1] Duchi, J., Hazan, E., & Singer, Y. (2011). Adaptive subgradient methods for online learning and stochastic optimization. Journal of Machine Learning Research, 12(Jul), 2121-2159.

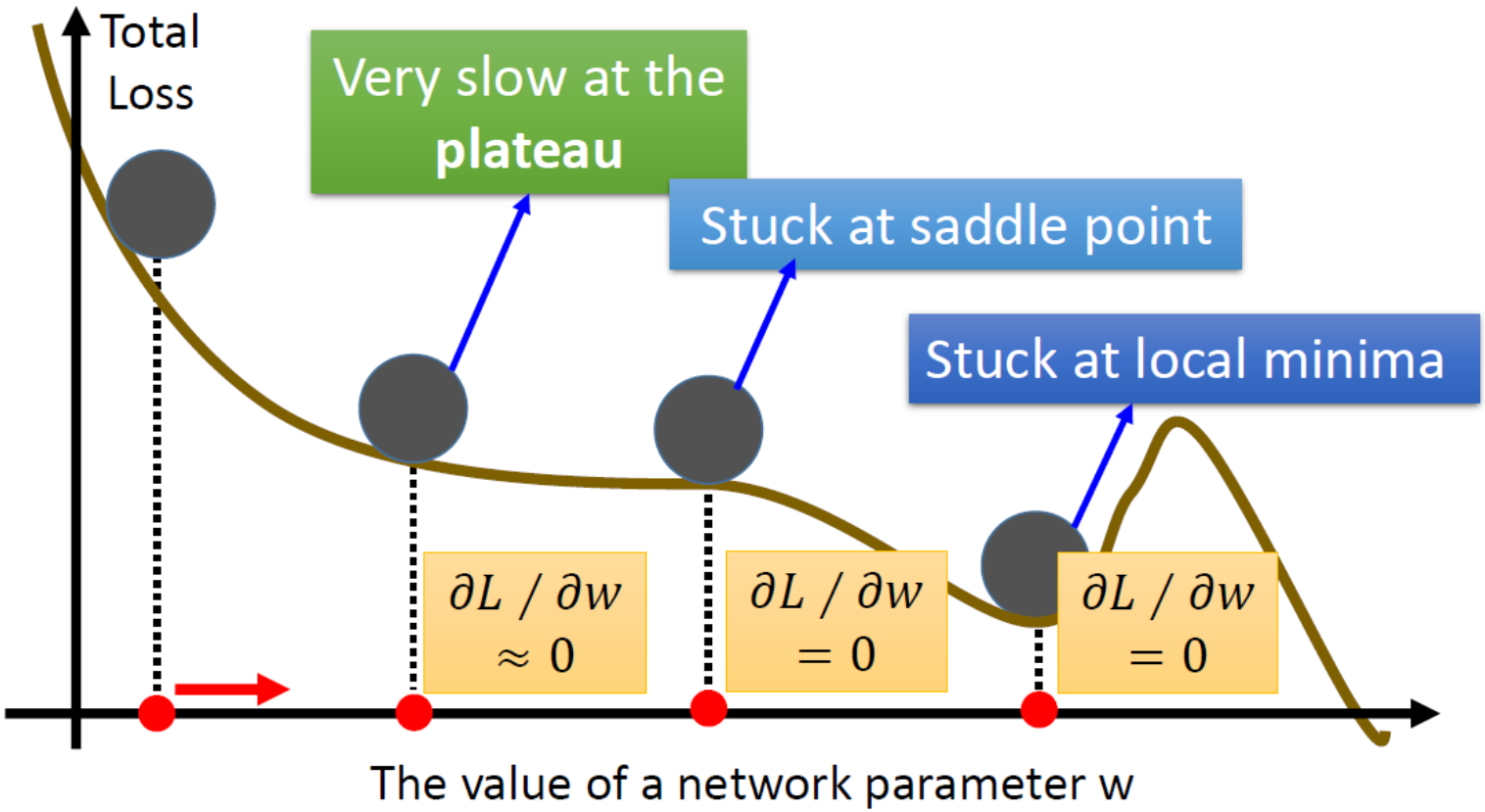


Figure: 图片来源：Hung-yi Lee , Machine Learning(2017)：Gradient Descent、Tips for training DNN

AdaGrad

- 令$g^k = \nabla f_{s_k}(x^k)$，为了记录整个迭代过程中梯度各个分量的累积情况，引入向量

$$G^k = \sum_{i=1}^{k} g^i \odot g^i.$$

从G^k的定义可知G^k的每个分量表示在迭代过程中，梯度在该分量处的累积平方和. 当G^k的某分量较大时，我们认为该分量变化比较剧烈，因此应采用小步长，反之亦然.

- 因此AdaGrad的迭代格式为

$$x^{k+1} = x^k - \frac{\alpha}{\sqrt{G^k + \varepsilon \mathbf{1}_n}} \odot g^k, \tag{1}$$

$$G^{k+1} = G^k + g^{k+1} \odot g^{k+1}, \tag{2}$$

这里$\frac{\alpha}{\sqrt{G^k+\varepsilon \mathbf{1}_n}}$中的除法和求根运算都是对向量每个分量分别操作的（下同），α为初始步长，引入$\varepsilon \mathbf{1}_n$这一项是为了防止除零运算.

- $\varepsilon = 10^{-6}, 10^{-8}$是通常的选择。

AdaGrad

- 优点：可以看到AdaGrad的步长大致反比于历史梯度累计值的算术平方根，所以梯度较大时步长下降很快，反之则下降较慢，这样做的效果是在参数空间更平缓的方向上，前后两次迭代的距离较大，因此适合梯度是稀疏的情况.
- 缺点：在凸优化问题中AdaGrad有比较好的理论性质，但实际应用中也发现在训练深度神经网络模型时，从训练开始就积累梯度平方会导致步长过早或过多减小. 从而使得训练过早停止。

AdaGrad的收敛阶

- 如果在AdaGrad中使用真实梯度$\nabla f(x^k)$，那么AdaGrad也可以看成是一种介于一阶和二阶的优化算法.
- 考虑$f(x)$在点x^k处的二阶泰勒展开：

$$f(x) \approx f(x^k) + \nabla f(x^k)^\top (x - x^k) + \frac{1}{2}(x - x^k)^\top B^k (x - x^k),$$

我们知道选取不同的B^k可以导出不同的优化算法. AdaGrad是使用一个对角矩阵来作为B^k. 具体地，取

$$B^k = \frac{1}{\alpha}\mathrm{Diag}(\sqrt{G^k + \varepsilon \mathbf{1}_n})$$

时导出的算法就是AdaGrad.

RMSProp

- RMSProp[2]（root mean square propagation）是对AdaGrad的一个改进，该方法在非凸问题上可能表现更好. AdaGrad会累加之前所有的梯度分量平方，这就导致步长是单调递减的，因此在训练后期步长会非常小，计算的开销也较大.
- RMSProp提出只需使用离当前迭代点比较近的项，同时引入衰减参数ρ. 具体地，令

$$M^{k+1} = \rho M^k + (1-\rho) g^{k+1} \odot g^{k+1},$$

再对其每个分量分别求根，就得到均方根(root mean square)

$$R^k = \sqrt{M^k + \varepsilon \mathbf{1}_n}, \tag{3}$$

最后将均方根的倒数作为每个分量步长的修正.

[2]Tieleman, T., & Hinton, G. (2012). Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural networks for machine learning, 4(2), 26-31.

RMSProp

- RMSProp迭代格式为：

$$x^{k+1} = x^k - \frac{\alpha}{R^k} \odot g^k, \tag{4}$$

$$M^{k+1} = \rho M^k + (1-\rho) g^{k+1} \odot g^{k+1}. \tag{5}$$

  引入参数ε同样是为了防止分母为0的情况发生. 一般取$\rho = 0.9, \alpha = 0.001$.

- 可以看到RMSProp 和AdaGrad 的唯一区别是将G^k替换成了M^k.

Adam

- Adam[3]选择了一个动量项进行更新：

$$S^k = \rho_1 S^{k-1} + (1-\rho_1) g^k.$$

- 类似RMSProp，Adam也会记录梯度的二阶矩：

$$M^k = \rho_2 M^{k-1} + (1-\rho_2) g^k \odot g^k.$$

- 与原始动量方法和RMSProp的区别是，由于S^k和M^k本身带有偏差，Adam在更新前先对其进行修正：

$$\hat{S}^k = \frac{S^k}{1-\rho_1^k}, \quad \hat{M}^k = \frac{M^k}{1-\rho_2^k},$$

这里ρ_1^k, ρ_2^k分别表示ρ_1, ρ_2的k次方.

- Adam最终使用修正后的一阶矩和二阶矩进行迭代点的更新.

$$x^{k+1} = x^k - \frac{\alpha}{\sqrt{\hat{M}^k + \varepsilon \mathbf{1}_n}} \odot \hat{S}^k.$$

[3] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Proceedings of 3rd International Conference on Learning Representations, 2015.

AMSGrad

- Adam 是目前最为常用的优化器,然而对某些凸优化问题其并不收敛[4]
- AMSGrad 对Adam进行了修正，得到了有收敛保证的算法。不过在实际大部分神经网络训练中，Adam的表现仍然最优。
- 此外，还有其他算法对Adam进行改进，例如AdaFactor[5]
  - NLP训练中，抛弃Adam里边的动量。CV模型很多时候要靠“SGD+动量”来炼出最优效果来，自适应学习率优化器通常训练不出最好的效果。但对于NLP模型来说，情况有点相反
  - 对额外内存需求的矩阵M^k做低秩矩阵分解，进一步减小内存占用。

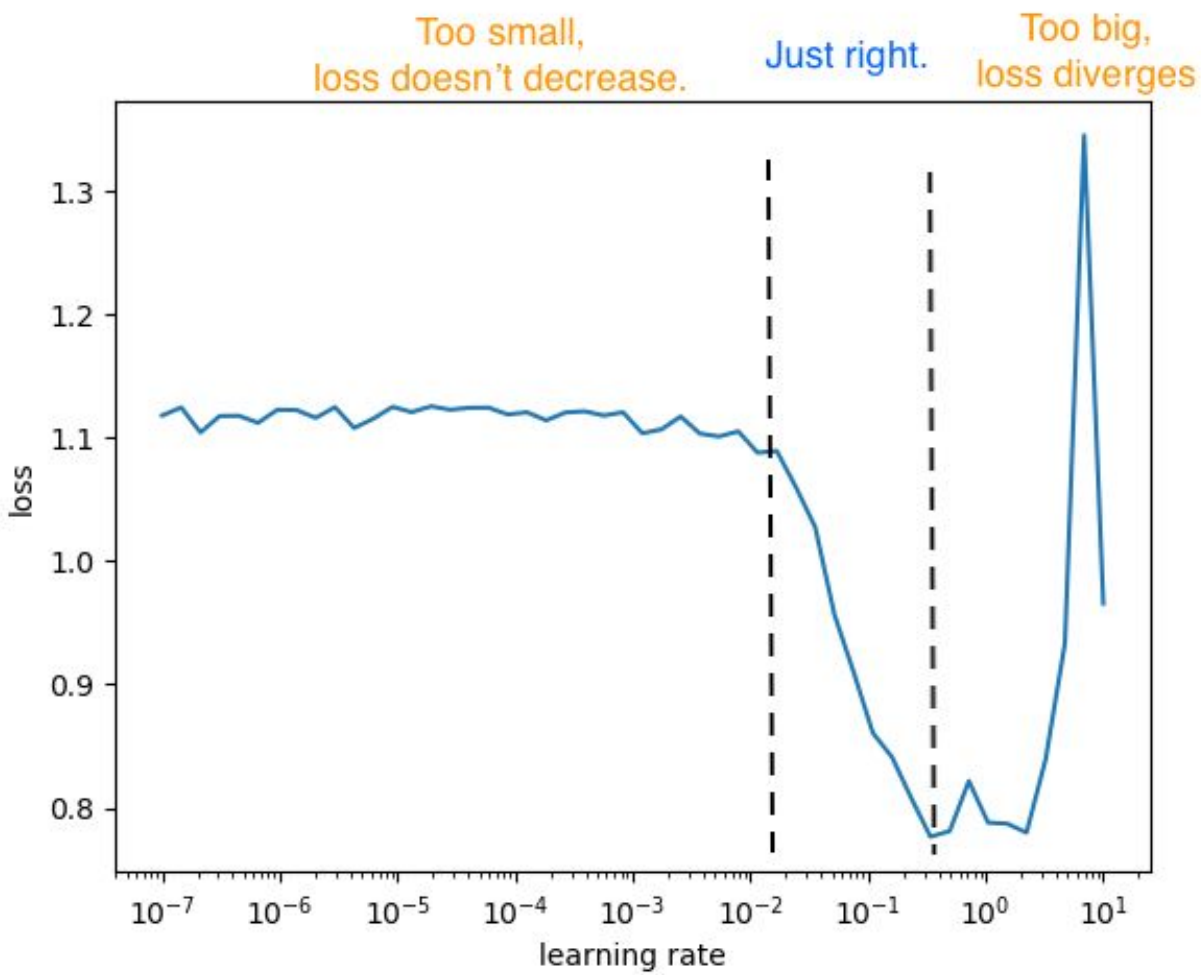
[4] Reddi, Sashank J., Satyen Kale, and Sanjiv Kumar. "On the Convergence of Adam and Beyond." International Conference on Learning Representations. 2018.

[5] Shazeer, Noam, and Mitchell Stern. "Adafactor: Adaptive learning rates with sublinear memory cost." International Conference on Machine Learning. PMLR, 2018.

步长/学习率设置

虽然AdaGrad、Adam等算法一定程度缓解了步长的设置，但是实际中步长参数仍然很大程度影响训练效果。

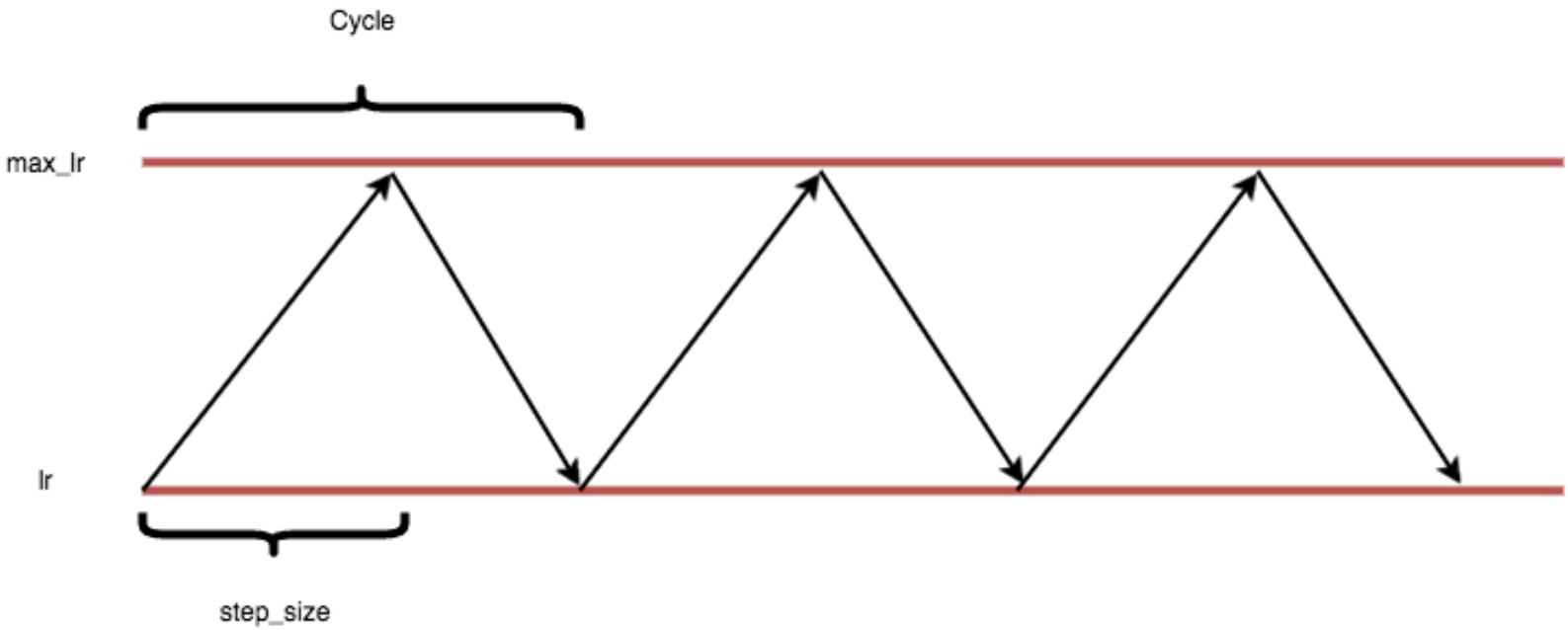
Cyclical 学习率[6]



LR Range Test :第一个区域中学习率太小以至于损失几乎没有减少，第二个区域里损失收敛很快，最后一个区域中学习率太大以至于损失开始发散。

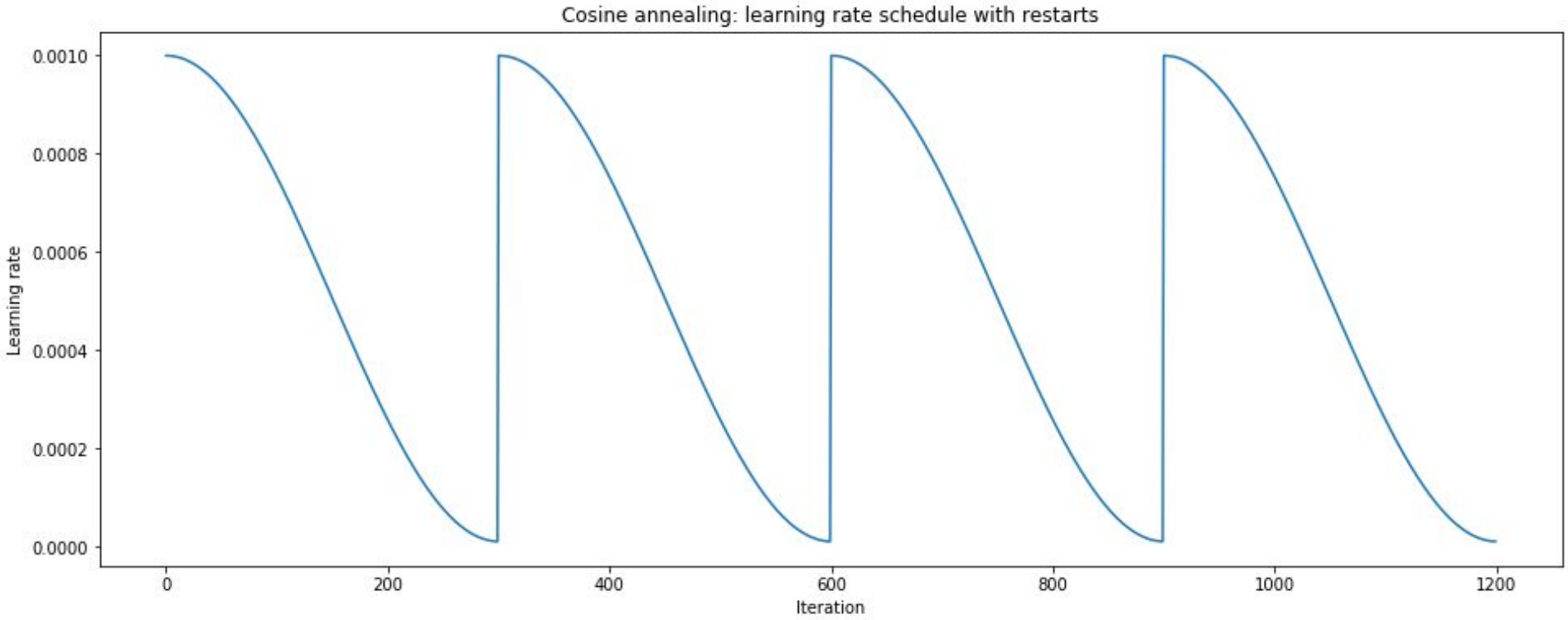
[6]Smith, Leslie N. "Cyclical learning rates for training neural networks." 2017 IEEE winter conference on applications of computer vision (WACV). IEEE, 2017.

Cyclical 学习率



- 让学习率在合理范围内$[lr, \max lr]$进行周期性变化，这样实际上能以更少的步骤提高模型的准确率。
- 鞍点位置的梯度较小，因此小的学习率使模型在训练后期遍历这些鞍点时会很慢。通过在后期提高学习率，可以帮助模型更有效地摆脱鞍点。

SGDR[7]



- Transformer模型训练常用的策略
- 重启动是优化中常见的方法，例如无梯度算法、共轭梯度法、Nesterov加速法
- SGDR周期性的热启动学习率，

[7] Loshchilov, Ilya, and Frank Hutter. "SGDR: Stochastic Gradient Descent with Warm Restarts." International Conference on Learning Representations. 2016.

Outline

批量大小

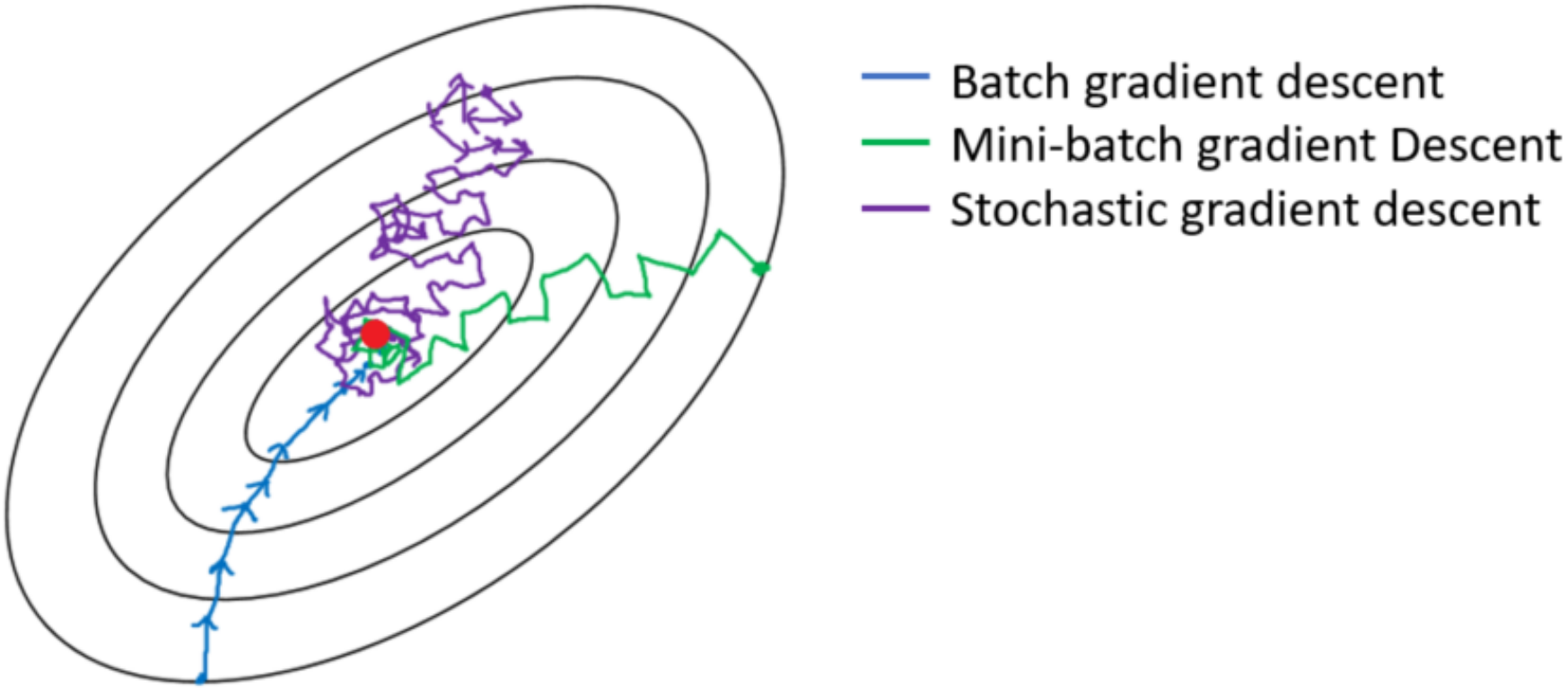


Figure: 来源：https://towardsdatascience.com/gradient-descent-algorithm-and-its-variants-10f652806a3

- 批量越小，噪音越大

Figure: batchsize=$\{8, 16, \ldots, 2048\}$测试误差以及训练时间。来源[8]

- 批量越大，训练越快，达到计算资源并行瓶颈
- 批量越大，测试误差越大。
- 一般认为，小批量所携带的噪声，有助于逃离鞍点，以及泛化性能较差的局部极小值（锐度较大的点）

[8] https://wandb.ai/ayush-thakur/dl-question-bank/reports/What-s-the-Optimal-Batch-Size-to-Train-a-Neural-Network—VmlldzoyMDkyNDU

Small Batch v.s. Large Batch

	Small	Large
Speed for one update (no parallel)	Faster	Slower
Speed for one update (with parallel)	Same	Same (not too large)
Time for one epoch	Slower	Faster WIN
Gradient	Noisy	Stable
Optimization	Better WIN	Worse
Generalization	Better WIN	Worse

Batch size is a hyperparameter you have to decide.

Figure: 小批量和大批量对比。来源[9]

[9] https://speech.ee.ntu.edu.tw/ hylee/ml/ml2021-course-data/small-gradient-v7.pdf

大批量训练方案:调整步长

Table 4: The L2 norm of layer weights and gradients for using AlexNet to train ImageNet dataset. The data is collected at 1st iteration. The batch size is 4096. conv means convolutional layer, and means fully-connected layer. x.0 means the layer weight, x.1 means the layer bias.

Layer	conv1.1	conv1.0	conv2.1	conv2.0	conv3.1	conv3.0	conv4.0	conv4.1
$\|\|w\|\|_2$	1.86	0.098	5.546	0.16	9.40	0.196	8.15	0.196
$\|\|\nabla w\|\|_2$	0.22	0.017	0.165	0.002	0.135	0.0015	0.109	0.0013
$\frac{\|\|w\|\|_2}{\|\|\nabla w\|\|_2}$	8.48	**5.76**	33.6	83.5	69.9	127	74.6	148
Layer	conv5.1	conv5.0	fc6.1	fc6.0	fc7.1	fc7.0	fc8.1	fc8.0
$\|\|w\|\|_2$	6.65	0.16	30.7	6.4	20.5	6.4	20.2	0.316
$\|\|\nabla w\|\|_2$	0.09	0.0002	0.26	0.005	0.30	0.013	0.22	0.016
$\frac{\|\|w\|\|_2}{\|\|\nabla w\|\|_2}$	73.6	69	117	**1345**	68	489	93	19

Figure: 不同层的参数与梯度比值不同，调整不同层的学习率$\eta = \gamma \frac{\|w\|}{\|\nabla L(w)\|}$：[10]

- Adam版本，Lamb算法[11]
- 各种技巧[12] [13] [14]

[10] You, Yang, Igor Gitman, and Boris Ginsburg. "Scaling sgd batch size to 32k for imagenet training." arXiv preprint arXiv:1708.03888 6.12 (2017): 6.

[11] Large Batch Optimization for Deep Learning: Training BERT in 76 minutes (ICLR 2020)

[12] Extremely Large Minibatch SGD: Training ResNet-50 on ImageNet in 15 Minutes (https://arxiv.org/abs/1711.04325)

[13] Stochastic Weight Averaging in Parallel: Large-Batch Training That Generalizes Well (https://arxiv.org/abs/2001.02312)

[14] Accurate, large minibatch sgd: Training imagenet in 1 hour (https://arxiv.org/abs/1706.02677)

Outline

梯度下降法与随机梯度下降法的比较

回顾分析梯度下降法与随机梯度下降法的主要区别：

- 在强凸性假设下，对梯度下降法有

$$\begin{aligned}\Delta_{k+1}^2 &= \|x^{k+1} - x^*\|^2 = \|x^k - \alpha\nabla f(x^k) - x^*\|^2 \\ &= \Delta_k^2 - 2\alpha\langle\nabla f(x^k), x^k - x^*\rangle + \alpha^2\|\nabla f(x^k)\|^2 \\ &\leq (1 - 2\alpha\mu)\Delta_k^2 + \alpha^2\|\nabla f(x^k)\|_2^2 \qquad (\mu\text{-强凸}) \\ &\leq (1 - 2\alpha\mu + \alpha^2 L^2)\Delta_k^2. \qquad (L\text{-光滑})\end{aligned} \tag{6}$$

梯度下降法与随机梯度下降法的比较

- 对随机梯度下降法，利用条件期望的性质有

$$
\begin{aligned}
\mathbb{E}[\Delta_{k+1}^2] &= \mathbb{E}[\|x^{k+1}-x^*\|_2^2] = \mathbb{E}[\|x^k-\alpha\nabla f_{s_k}(x^k)-x^*\|]^2 \\
&= \mathbb{E}[\Delta_k^2] - 2\alpha\mathbb{E}[\langle\nabla f_{s_k}(x^k), x^k-x^*\rangle] + \alpha^2\mathbb{E}[\|\nabla f_{s_k}(x^k)\|^2] \\
&= \mathbb{E}[\Delta_k^2] - 2\alpha\mathbb{E}[\langle\nabla f(x^k), x^k-x^*\rangle] + \alpha^2\mathbb{E}[\|\nabla f_{s_k}(x^k)\|^2] \\
&\le (1-2\alpha\mu)\mathbb{E}[\Delta_k^2] + \alpha^2\mathbb{E}[\|\nabla f_{s_k}(x^k)\|^2] \qquad (\mu\text{-强凸}) \\
&= (1-2\alpha\mu)\mathbb{E}[\Delta_k^2] + \alpha^2\mathbb{E}[\|\nabla f_{s_k}(x^k)-\nabla f(x^k)+\nabla f(x^k)\|^2] \\
&\le \underbrace{(1-2\alpha\mu+\alpha^2L^2)\mathbb{E}[\Delta_k^2]}_{A} + \underbrace{\alpha^2\mathbb{E}[\|\nabla f_{s_k}(x^k)-\nabla f(x^k)\|^2]}_{B}.
\end{aligned}
\tag{7}
$$

- 可以看到两种算法的主要差别就在B项上，也就是梯度估计的某种方差．它导致了随机梯度算法只能有$\mathcal{O}(1/k)$的收敛速度．

方差减小技术

- 在许多机器学习的应用中，随机梯度算法的收敛速度更快一些.
- 这主要是因为许多应用对解的精度要求不太高，而在开始部分方差较小，即有$B \ll A$，那么我们会观察到近似Q-线性收敛速度；
- 而随着迭代步数增多，方差增大，最终的收敛速度为$\mathcal{O}(1/k)$.
- 为了能获得比较快的渐进收敛速度，我们的主要目标即减少方差项B. 下面介绍三种减小方差的算法：
  - SAG (stochastic average gradient)
  - SAGA
  - SVRG (stochastic variance reduced gradient)

基本思想

假设我们想使用蒙特卡罗方法来估计某随机变量X的均值$\mathbb{E}X$，我们可以很容易的采样一个样本X,并且可以有效地计算另一个与X 高度相关的随机变量Y 的期望$\mathbb{E}Y$。一种降低方差的方法是使用以下估计器θ_γ 来近似$\mathbb{E}X$:

$$\theta_\gamma = \gamma(X - Y) + \mathbb{E}Y \quad \gamma \in [0, 1].$$

我们有

- $$\mathbb{E}\theta_\gamma = \gamma\mathbb{E}X + (1 - \gamma)\mathbb{E}Y \tag{8}$$
- $$\mathrm{Var}\theta_\gamma = \gamma^2[\mathrm{Var}(X) + \mathrm{Var}(Y) - 2\mathrm{Cov}(X, Y)]$$
- 这表明，当γ 从0 增大到1, 方差增大，但是偏差降低。

随机梯度法中，我们每次仅采样$\mathbb{E}X$的一个估计$\nabla f_{s_k}(x_k)$. 这对应在(8)中令$\gamma = 1$,是均值$\mathbb{E}X = \frac{1}{N}\sum_i \nabla f_i(x_k)$的无偏估计，但是方差较大。

SAG算法

- 当迭代接近收敛时，上一步的随机梯度也是当前迭代点处梯度的一个很好的估计. 随机平均梯度法(SAG)[15]就是基于这一想法.
- 在迭代中，SAG算法记录所有之前计算过的随机梯度，再与当前新计算的随机梯度求平均，最终作为下一步的梯度估计.
- 具体来说，SAG算法在内存中开辟了存储N个随机梯度的空间

$$[g_1^k, g_2^k, \cdots, g_N^k],$$

分别用于记录和第i个样本相关的最新的随机梯度. 在第k步更新时，若抽取的样本点下标为s_k，则计算随机梯度后将$g_{s_k}^k$的值更新为当前的随机梯度值，而其他未抽取到的下标对应的g_i^k保持不变. 每次SAG 算法更新使用的梯度方向是所有g_i^k的平均值.

[15] M. Schmidt, N. Le Roux, and F. Bach, Minimizing finite sums with the stochastic average gradient," Mathematical Programming, vol. 162, no. 1-2, pp. 83112, 2017.

SAG算法

- 换言之，在每步迭代中，我们先抽样$\mathbb{E}X_k$的随机无偏估计$\nabla f_{s_k}(x^k)$, 储存的历史随机梯度视为随机变量Y, 则

$$\mathbb{E}Y_k = \frac{1}{N}\sum_{i}^{N} g_i^{k-1}$$

- 在(8)中令$\gamma = \frac{1}{N}$, 则得到了一个梯度估计：

$$\theta_\gamma^k = \frac{1}{N}(\nabla f_{s_k}(x^k) - g_{s_k}^{k-1}) + \frac{1}{N}\sum_{i=1}^{N} g_i^{k-1}, \tag{9}$$

- SAG算法的迭代格式为

$$x^{k+1} = x^k - \alpha_k \theta_\gamma^k = x^k - \frac{\alpha_k}{N}\sum_{i=1}^{N} g_i^k$$

其中g_i^k的更新方式为

$$g_i^k = \begin{cases} \nabla f_{s_k}(x^k), & i = s_k, \\ g_i^{k-1}, & \text{其他}, \end{cases} \tag{10}$$

这里s_k是第k次迭代随机抽取的样本.

SAG算法

- $\{g_i^k\}$的初值可简单地取为0或中心化的随机梯度向量，
- SAG算法每次使用的随机梯度的条件期望并不是真实梯度$\nabla f(x^k)$，因为

$$\mathbb{E}[\theta_\gamma^k \mid \textit{everything prior to } x^k] = \frac{1}{N}\nabla f(x^k) + (1 - \frac{1}{N})\sum_{i=1}^{N} g_i^{k-1}$$

- 但随着迭代进行，随机梯度的期望和真实梯度的偏差会越来越小.

SAG算法的收敛性

定理 (SAG算法的收敛性)

在强凸性收敛性假设的条件下，取固定步长$\alpha_k = \frac{1}{16L}$，g_i^k的初值取为0，则对任意的k，有

$$\mathbb{E}[f(x^k)] - f(x^*) \leq \left(1 - \min\left\{\frac{\mu}{16L}, \frac{1}{8N}\right\}\right)^k C_0,$$

其中常数C_0为与k无关的常数.

- 优点：SAG每次迭代仅需计算一个随机梯度，有Q-线性收敛速度.
- 缺点：SAG算法在于需要存储N个梯度向量，当样本量N很大时，这是一个很大的开销. 因此SAG算法在实际中很少使用.

SAGA算法

- SAGA算法[16]和SAG算法一样，储存了历史随机梯度。但是(8)中$\gamma = 1$.
- SAGA算法的迭代方式为

$$x^{k+1} = x^k - \alpha_k \left(\nabla f_{s_k}(x^k) - g_{s_k}^{k-1} + \frac{1}{N} \sum_{i=1}^{N} g_i^{k-1} \right). \tag{11}$$

- 每次迭代使用的梯度方向都是无偏的，即

$$\mathbb{E}\left[\nabla f_{s_k}(x^k) - g_{s_k}^{k-1} + \frac{1}{N} \sum_{i=1}^{N} g_i^{k-1} \,\middle|\, x^k \right] = \nabla f(x^k).$$

- 但是SAGA对比SAG的方差更大。

[16] A. Defazio, F. Bach, and S. Lacoste-Julien, SAGA: A fast incremental gradient method with support for non-strongly convex composite objectives," in Advances in neural information processing systems,pp. 1646-1654, 2014.

SAGA算法的收敛性

SAGA算法同样有Q-线性收敛速度：

定理 (SAGA算法的收敛性)

在强凸性收敛性假设的条件下，取固定步长$\alpha_k = \frac{1}{2(\mu N+L)}$. 定义$\Delta_k = \|x^k - x^*\|$，则对任意的$k \geq 1$有

$$\mathbb{E}[\Delta_k^2] \leq \left(1 - \frac{\mu}{2(\mu N + L)}\right)^k \left(\Delta_1^2 + \frac{N(f(x^1) - f(x^*))}{\mu N + L}\right). \tag{12}$$

如果强凸的参数μ是未知的，也可以取$\alpha = \frac{1}{3L}$，有类似的收敛结果.

Outline

SVRG算法

- 与SAG算法和SAGA算法不同，SVRG算法[17]通过周期性缓存全梯度的方法来减小方差.
- 具体做法是在随机梯度下降方法中，每经过m次迭代就设置一个检查点，计算一次全梯度，在之后的m次迭代中，将这个全梯度作为参考点来达到减小方差的目的.
- 令$\tilde{x}^j$是第j个检查点，则我们需要计算点$\tilde{x}^j$处的全梯度

$$\nabla f(\tilde{x}^j) = \frac{1}{N}\sum_{i=1}^{N} \nabla f_i(\tilde{x}^j),$$

在之后的迭代中使用方向v^k作为更新方向：

$$v^k = \nabla f_{s_k}(x^k) - (\nabla f_{s_k}(\tilde{x}^j) - \nabla f(\tilde{x}^j)), \tag{13}$$

其中$s_k \in \{1, 2, \cdots, N\}$是随机选取的一个样本.

[17] Accelerating stochastic gradient descent using predictive variance reduction," R. Johnson, T. Zhang, NIPS, 2013.

Algorithm SVRG for finite-sum optimization

for $j = 1, 2, \ldots$ **do**

  $\tilde{x}^j \leftarrow x_m^{j-1}$, and compute $\nabla F(\tilde{x}^j)$ `// full gradient`

1 Initialize $x_0^j \leftarrow \tilde{x}^j$

2 **for** $k = 0, \ldots, m-1$ **do**

    `// each epoch contains` m `iterations`

3   Choose s_k uniformly from $\{1, \ldots, N\}$

4    $x_{k+1}^j = x_k^j - \alpha \left(\nabla f_{s_k}(x_k^j) - \nabla f_{s_k}(\tilde{x}^j) + \nabla F(\tilde{x}^j) \right)$

5 **end**

end

- 算法中，每m步重新计算一个完整的梯度。内层循环完成m步称为一个epoch。
- 每个epoch计算$2m + n$个小梯度。
- 相对比SGD，如果$m \geq n$,那么计算量只差常数倍，但是收敛速度提高一个阶。
- 对比SAG和SAGA，不需要储存梯度表格。

- 注意到给定$s_1, s_2, \cdots, s_{k-1}$时$x^k, \tilde{x}^j$均为定值，由$v^k$的表达式可知

$$\begin{aligned}
&\mathbb{E}[v^k | s_1, s_2, \cdots, s_{k-1}] \\
&= \mathbb{E}[\nabla f_{s_k}(x^k)|x^k] - \mathbb{E}[\nabla f_{s_k}(\tilde{x}^j) - \nabla f(\tilde{x}^j)|s_1, s_2, \cdots, s_{k-1}] \\
&= \nabla f(x^k) - 0 = \nabla f(x^k),
\end{aligned}$$

- 当$x^k \approx \tilde{x}^j \approx x^*$时，$v^k$也趋于0.

SVRG算法-方差

- 假设

$$\|\nabla f_i(x) - \nabla f_i(y)\| \leq L\|x-y\|, \quad i = 1, 2, \cdots, N.$$

- 令$y = \tilde{x}^j$，x^*为$f(x)$的最小值点，$\Delta_k = \|x^k - x^*\|$，则

$$\begin{aligned}
&\mathbb{E}\left[\|v^k\|^2\right] = \mathbb{E}\left[\|\nabla f_{s_k}(x^k) - (\nabla f_{s_k}(y) - \nabla f(y))\|^2\right] \\
&= \mathbb{E}\left[\|\nabla f_{s_k}(x^k) - \nabla f_{s_k}(y) + \nabla f(y) + \nabla f_{s_k}(x^*) - \nabla f_{s_k}(x^*)\|^2\right] \\
&\leq 2\mathbb{E}\left[\|\nabla f_{s_k}(x^k) - \nabla f_{s_k}(x^*)\|^2\right] + 2\mathbb{E}\left[\|\nabla f_{s_k}(y) - \nabla f(y) - \nabla f_{s_k}(x^*)\|^2\right] \qquad (14) \\
&\leq 2L^2\mathbb{E}\left[\Delta_k^2\right] + 2\mathbb{E}\left[\|\nabla f_{s_k}(y) - \nabla f_{s_k}(x^*)\|^2\right] \\
&\leq 2L^2\mathbb{E}\left[\Delta_k^2\right] + 2L^2\mathbb{E}\left[\|y - x^*\|^2\right].
\end{aligned}$$

其中第一个不等式是因为$\|a+b\|^2 \leq 2\|a\|^2 + 2\|b\|^2$，第二个不等式使用了有关二阶矩的不等式

$$\mathbb{E}[\|\xi - \mathbb{E}\xi\|^2] \leq \mathbb{E}[\|\xi\|^2].$$

- 所以方差可以被到最优点距离控制

SVRG算法

- 从(14)式看出，若x^k和$y = \tilde{x}^j$非常接近x^*，梯度估计的方差就很小.
- 频繁地更新y可以使得方差更小，但也增加了计算全梯度的次数.

SVRG算法的收敛性

下面给出SVRG算法的收敛性. 这里的收敛性是针对参考点序列$\{\tilde{x}^j\}$而言的.

定理 (SVRG算法的收敛性)

设m为利用每个$\tilde{x}^j$更新的次数. 设每个$f_i(x)$可微, 且梯度L-利普希茨连续; 函数$f(x)$强凸, 强凸参数为μ. 取步长$\alpha \in \left(0, \frac{1}{2L}\right]$, 并且$m$充分大使得

$$\rho = \frac{1}{\mu\alpha(1-2L\alpha)m} + \frac{2L\alpha}{1-2L\alpha} < 1, \tag{15}$$

则*SVRG*算法对于参考点$\tilde{x}^j$在函数值期望的意义下有*Q*-线性收敛速度:

$$\mathbb{E}f(\tilde{x}^j) - f(x^*) \leq \rho\mathbb{E}[f(\tilde{x}^{j-1}) - f(x^*)]. \tag{16}$$

- 若$m \underset{\sim}{>} \frac{L}{\mu} = \kappa$, 使用常数步长$\alpha = \mathcal{O}(1/L)$, 那么$\rho < 1/2$. 收敛到$\mathbb{E}f(\tilde{x}^j) - f(x^*) \leq \epsilon$ 需要计算量为
$$\mathcal{O}((\kappa + n)\log\frac{1}{\epsilon}).$$
- 回顾全梯度法需要
$$\mathcal{O}(n\kappa\log\frac{1}{\epsilon}).$$
这表明n很大时, SVRG更优。

定理的证明

- 定义$\Delta_k = \|x^k - x^*\|$.
- 对于内层循环，

$$\begin{aligned}
\mathbb{E}[\Delta_{k+1}^2] &= [\|x^{k+1} - x^*\|^2] = \mathbb{E}[\|x^k - \alpha v^k - x^*\|^2] \\
&= \mathbb{E}[\Delta_k^2] - 2\alpha\mathbb{E}[\langle v^k, x^k - x^*\rangle] + \alpha^2\mathbb{E}[\|v^k\|^2] \\
&= \mathbb{E}[\Delta_k^2] - 2\alpha\mathbb{E}[\langle \nabla f(x^k), x^k - x^*\rangle] + \alpha^2\mathbb{E}[\|v^k\|^2] \\
&\leq \mathbb{E}[\Delta_k^2] - 2\alpha\mathbb{E}[(f(x^k) - f(x^*))] + \alpha^2\mathbb{E}[\|v^k\|^2].
\end{aligned}$$

- 根据f_i是凸函数且梯度$L-$Lipschitz连续，

$$\|\nabla f_i(x) - \nabla f_i(x^*)\|^2 \leq 2L[f_i(x) - f_i(x^*) - \nabla f_i(x^*)^\top (x - x^*)].$$

- 对i从1到N进行求和，注意$\nabla f(x^*) = 0$：

$$\frac{1}{N}\sum_{i=1}^{N} \|\nabla f_i(x) - \nabla f_i(x^*)\|^2 \leq 2L[f(x) - f(x^*)], \quad \forall x. \tag{17}$$

- 利用(14)式的推导过程可得

$$\mathbb{E}[\|v^k\|^2] \leq 2\mathbb{E}[\|\nabla f_{s_k}(x^k) - \nabla f_{s_k}(x^*)\|^2] + 2\mathbb{E}[\|\nabla f_{s_k}(\tilde{x}^{j-1}) - \nabla f_{s_k}(x^*)\|^2].$$

对上式右侧第一项，有

$$\begin{aligned}
&\mathbb{E}[\|\nabla f_{s_k}(x^k) - \nabla f_{s_k}(x^*)\|^2] \\
=\ &\mathbb{E}[\mathbb{E}[\|\nabla f_{s_k}(x^k) - \nabla f_{s_k}(x^*)\|^2 | s_1, s_2, \cdots, s_{k-1}]] \\
=\ &\mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N} \|\nabla f_i(x^k) - \nabla f_i(x^*)\|^2\right] \leq 2L\mathbb{E}[f(x^k) - f(x^*)],
\end{aligned}$$

- 类似地，对右侧第二项，有

$$\mathbb{E}[\|\nabla f_{s_k}(\tilde{x}^{j-1}) - \nabla f_{s_k}(x^*)\|^2] \leq 2L\mathbb{E}[f(\tilde{x}^{j-1}) - f(x^*)].$$

- 最终可得对$\mathbb{E}[\|v^k\|^2]$的估计：

$$\mathbb{E}[\|v^k\|^2] \leq 4L(\mathbb{E}[f(x^k) - f(x^*)] + \mathbb{E}[f(\tilde{x}^{j-1}) - f(x^*)]).$$

- 将$\mathbb{E}[\|v^k\|^2]$的上界代入对$\mathbb{E}[\Delta_{k+1}^2]$的估计，就有

$$\begin{aligned}\mathbb{E}[\Delta_{k+1}^2] &\leq \mathbb{E}[\Delta_k^2] - 2\alpha\mathbb{E}[f(x^k) - f(x^*)] + \alpha^2\mathbb{E}[\|v^k\|^2] \\ &\leq \mathbb{E}[\Delta_k^2] - 2\alpha(1 - 2\alpha L)\mathbb{E}[f(x^k) - f(x^*)] \\ &\quad + 4L\alpha^2\mathbb{E}[f(\tilde{x}^{j-1}) - f(x^*)].\end{aligned}$$

- 对k从1到m求和，并且注意到$x^1 = \tilde{x}^{j-1}$就可以得到

$$\begin{aligned}
&\mathbb{E}[\Delta_{m+1}^2] + 2\alpha(1-2\alpha L)\sum_{k=1}^{m}\mathbb{E}[f(x^k) - f(x^*)] \\
&\leq \mathbb{E}[\|\tilde{x}^{j-1} - x^*\|^2] + 4L\alpha^2 m\mathbb{E}[f(\tilde{x}^{j-1}) - f(x^*)] \\
&\leq \frac{2}{\mu}\mathbb{E}[f(\tilde{x}^{j-1}) - f(x^*)] + 4L\alpha^2 m\mathbb{E}[f(\tilde{x}^{j-1}) - f(x^*)],
\end{aligned}$$

- 注意到$\tilde{x}^j = \frac{1}{m}\sum_{k=1}^{m} x^k$，所以

$$\begin{aligned}
&\mathbb{E}[f(\tilde{x}^j) - f(x^*)] \\
&\leq \frac{1}{m}\sum_{k=1}^{m}\mathbb{E}[f(x^k) - f(x^*)] \\
&\leq \frac{1}{2\alpha(1-2\alpha L)m}\left(\frac{2}{\mu} + 4mL\alpha^2\right)\mathbb{E}[f(\tilde{x}^{j-1}) - f(x^*)] \\
&= \rho\mathbb{E}[f(\tilde{x}^{j-1}) - f(x^*)].
\end{aligned}$$

Outline

回顾：原问题与对偶问题

原问题：（给定数据$x_i, i = 1, \ldots, n$）

$$\min_{w \in \mathbb{R}^d} P(w) = \frac{1}{n} \sum_{i=1}^{n} \phi_i(w^T x_i) + \frac{\lambda}{2} \|w\|^2$$

对偶问题：

$$\max_{\alpha \in \mathbb{R}^n} D(\alpha) = \frac{1}{n} \sum_{i=1}^{n} -\phi_i^*(-\alpha_i) - \frac{\lambda}{2} \left\| \frac{1}{\lambda n} \sum_{i=1}^{n} \alpha_i x_i \right\|^2$$

- 设原问题最优解为w^*, 对偶问题最优解为α^*
- 定义$w(\alpha) = \frac{1}{\lambda n} \sum_{i=1}^{n} \alpha_i x_i$，已知$w(\alpha^*) = w^*$.
- 由对偶定理可知：

$$P(w^*) = D(\alpha^*), \text{ 且} \forall w, \alpha, \quad P(w) \geq D(\alpha).$$

- 定义对偶间隙：

$$P(w(\alpha)) - D(\alpha)$$

SDCA算法

SDCA算法步骤如下:

1. 随机选择一个训练样本i.
2. 对偶问题进行精确线搜索，找到$\Delta\alpha_i$:

$$\max -\phi_i^*\left(-(\alpha_{t-1,i}+\Delta\alpha_i)\right)-\frac{\lambda n}{2}\left\|w_{t-1}+\frac{1}{\lambda n}\Delta\alpha_i x_i\right\|^2$$

3. 更新对偶变量α_t和原始变量w_t:

$$\alpha_t \leftarrow \alpha_{t-1}+\Delta\alpha_i e_i, \quad w_t \leftarrow w_{t-1}+\frac{1}{\lambda n}\Delta\alpha_i x_i$$

4. 当对偶间隙足够小时终止。

SDCA-dual free [18]

当函数ϕ不易计算其共轭函数ϕ^*时，SDCA并不易用。SDCA-dual free是一种无需对偶函数的新算法。令

$$\varphi_i(w) = \phi(w^T x_i), \forall i.$$

Algorithm SDCA 无对偶版本

Input: 初始化$w_0 = \frac{1}{\lambda n}\sum_{i=1}^n v_{0,i}$

for $k = 0, 1, \ldots$ **do**

  从$\{1, \ldots, n\}$ 中随机等概率选择i_k

  $\Delta_k \leftarrow -\eta\lambda n \left(\nabla\varphi_{i_k}(w_k) + \nu_{k,i_k}\right)$

  $\nu_{k+1,i} \leftarrow \nu_{k,i} + \Delta_k 1\{i = i_k\}$ // 只更新第i_k 个坐标

  $w_{k+1} \leftarrow w_k + \frac{1}{\lambda n}\Delta_k$

由初始化方式，以及更新公式，在新算法中，我们仍有原始-对偶变量的关系：

$$w_k = \frac{1}{\lambda n}\sum_{i=1}^n \nu_{k,i}, \quad \forall k \geq 0 \tag{18}$$

[18] "SDCA without duality, regularization, and individual convexity," S. Shalev-Shwartz, ICML, 2016.

直观理解

- 原问题最优条件：每个ν_i^* 都等于$-\nabla\varphi_i(w^*)$，对所有i 成立。
- 通过修改的更新规则，对对偶变量和随机梯度进行凸组合：

$$\nu_{k+1,i_k} \leftarrow (1-\eta\lambda n)\nu_{k,i_k} + \eta\lambda n\left(-\nabla\varphi_{i_k}(x_k)\right)$$

  当它收敛时，它将满足上述最优条件。
- SDCA（无对偶性）的更新规则可以整理如下：

$$w_{k+1} = w_k - \eta\left(\nabla\varphi_{i_k}(w_k) + \nu_{k,i_k}\right)$$

- 可以直接验证这是一个无偏的梯度估计。
  期望梯度为：

$$E[\nabla\varphi_{i_k}(w_k)+\nu_{k,i_k}]=\frac{1}{n}\sum_{i=1}^{n}(\nabla\varphi_i(w_k)+\nu_{i,k})\overset{(18)}{=}\nabla P(w_k)$$

- 随着我们逐渐接近最优点，可以证明$\nabla\varphi_i(w_k)+\nu_{i,k}$的方差将趋于0。
- 故SDCA-dual free 也是一种方差缩减的随机梯度法。
- 缺点：仍需要存储n个向量$v_{k,i}, i=1,\ldots,n.$
- 优点：在强凸时可以证明与SVRG相似的复杂度，也可拓展到求解非凸光滑问题。

Outline

递归随机梯度估计SARAH[19]

递归/自适应更新梯度估计的关键思想：

$$g_t = \nabla f_i(x_t) - \nabla f_i(x_{t-1}) + g_{t-1} \tag{19}$$

$$x_{t+1} = x_t - \eta g_t \tag{20}$$

需要计算2个随机梯度。

与SVRG的比较（在整个epoch中使用固定点处的梯度）：

$$g_t = \nabla f_i(x_t) - \nabla f_i(x_{\text{old}}) + \nabla F(x_{\text{old}})$$

[19]"SARAH: A novel method for machine learning problems using stochastic recursive gradient," L. Nguyen, J. Liu, K. Scheinberg, M. Takac, ICML, 2017

SARAH

对于许多（例如强凸）问题，递归梯度估计g_t可能迅速衰减（方差下降；偏差上升）。

- g_t可能会迅速偏离目标梯度$\nabla F(x_t)$
- g_t不能保证足够的下降，导致进展停滞

解决方案：每隔几次迭代重置g_t以与真实批梯度校准。

Algorithm 随机递归梯度算法（SARAH）

初始化：设定迭代次数S

for $s = 1, 2, \ldots, S$ **do**

  设置初始点$x_0^s \leftarrow x_{m+1}^{s-1}$并计算$g_0^s = \nabla F(x_0^s)$

  $x_1^s = x_0^s - \eta g_0^s$

  for $t = 1, \ldots, m$ **do**

    从$\{1, \ldots, n\}$中均匀选择i_t

    $g_t^s = \nabla f_{i_t}(x_t^s) - \nabla f_{i_t}(x_{t-1}^s) + g_{t-1}^s$

    $x_{t+1}^s = x_t^s - \eta g_t^s$

  end

end

梯度估计的偏差

与SVRG 不同，g_t 并不是$\nabla F(x_t)$ 的无偏估计。
期望公式如下：

$$E[g_t \mid 在x_t 之前的所有信息] = \nabla F(x_t) - \nabla F(x_{t-1}) + g_{t-1}$$

虽然这不等于$\nabla F(x_t)$，但如果我们平均掉所有随机性，我们有（数学归纳法）：

$$E[g_t] = E[\nabla F(x_t)]$$

这表示，在平均意义上，g_t 可以看作是对$\nabla F(x_t)$的无偏估计。所以SARAH与SAG、SAGA也不相同。

定理**19.1**（Nguyen 等，2019[20]）
假设每个f_i是L-平滑的。那么当$\eta \leq \frac{1}{L\sqrt{m}}$时，SARAH 算法满足：

$$\frac{1}{(m+1)S}\sum_{s=1}^{S}\sum_{t=0}^{m} E\left[\|\nabla F(x_t^s)\|^2\right] \leq \frac{2}{\eta(m+1)S}\left[F(x_0^0) - F(x^*)\right]$$

- 其中，(设置$m \approx n$，$\eta \approx \frac{1}{L\sqrt{m}}$)寻找$\varepsilon$-近似稳定点的迭代复杂度（即$\|\nabla F(x)\| \leq \varepsilon$）为：

$$O\left(n + \frac{L\sqrt{n}}{\varepsilon^2}\right)$$

  是有限求和形式非凸情况下的最优复杂度。
- 该结果比全梯度法和SGD相比更优($O(\frac{Ln}{\varepsilon^2})$)，此结果也由Fang 等人在2018 年针对一个类似SARAH 的算法“Spider”推导出，并由Wang 等人在2019 年针对“SpiderBoost”进行了改进。

[20]Nguyen, Lam M., et al. "Finite-sum smooth optimization with SARAH." Computational Optimization and Applications 82.3 (2022): 561-593.

对于有限求和问题

$$\min f(x) = \frac{1}{N}\sum_{i=1}^{N} f_i(x)$$

计算单个梯度的收敛复杂度如下表：

Table: 梯度类算法复杂度

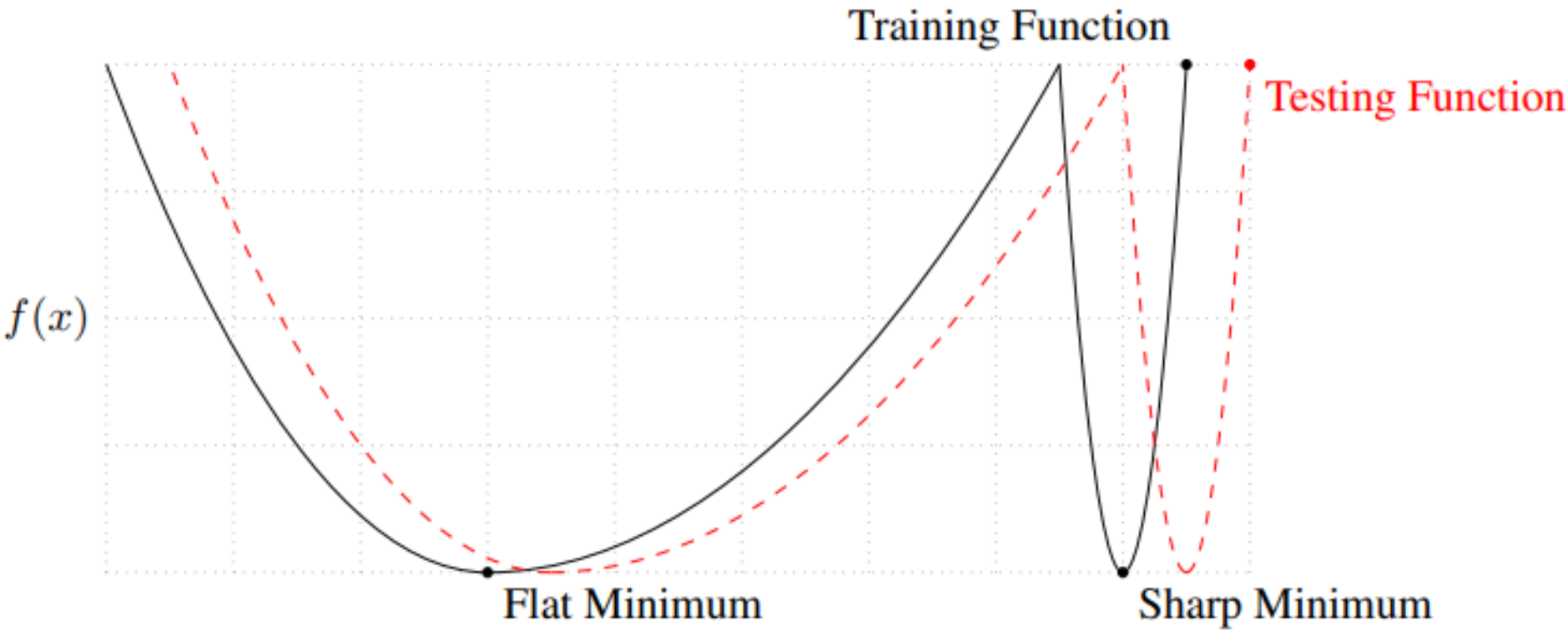
	f凸(次梯度算法)[21]	f可微强凸且L-光滑[22]	f非凸且L-光滑[23]
SGD	$\mathcal{O}\left(\frac{1}{\varepsilon^2}\right)$	$\mathcal{O}\left(\frac{1}{\varepsilon}\right)$	$\mathcal{O}(\frac{1}{\epsilon^4})$
全梯度（GD）	$\mathcal{O}\left(\frac{N}{\varepsilon^2}\right)$	$\mathcal{O}\left(N\kappa \ln\left(\frac{1}{\varepsilon}\right)\right)$	$\mathcal{O}(N\frac{1}{\epsilon^2})$
Nesterov 加速梯度	-	$\mathcal{O}\left(N\sqrt{\kappa} \ln\left(\frac{1}{\varepsilon}\right)\right)$	-
SAG/SAGA	-	$\mathcal{O}((\kappa+N)\ln(1/\epsilon))$	-
SVRG	-	$\mathcal{O}((\kappa+N)\ln(1/\epsilon))$	$\mathcal{O}(N+N^{2/3}/\epsilon^2)$
SARAH/Spider	-	-	$\mathcal{O}(N+\sqrt{N}/\epsilon^2)$

[21] $f(x)-f(x^*)\leq\epsilon$
[22] $f(x)-f(x^*)\leq\epsilon$
[23] $\|\nabla f(x)\|\leq\epsilon$

Outline

SAM:[24] Sharpness Aware Minimization

- 谷歌团队提出的SAM优化器，旨在提高训练结果的泛化能力。
- 研究者认为，局部极小点如果锐度很高，那么其泛化性较差。



[24] Foret, Pierre, et al. "Sharpness-aware Minimization for Efficiently Improving Generalization." International Conference on Learning Representations. 2020.

- 假设期望损失为$L_{\mathcal{D}}(w)$, 经验损失为$L_{\mathcal{S}}(w)$. 那么以高概率有[25]

$$L_{\mathcal{D}}(w) \leq \max_{\|\epsilon\|_2 \leq \rho} L_{\mathcal{S}}(w+\epsilon) + h(\frac{\|w\|_2^2}{\rho})$$

  其中h为某严格单调递增函数, ρ为参数，通常取0.05 或0.1 。
- 可将上式右侧改写为

$$[\underbrace{\max_{\|\epsilon\|_2 \leq \rho} L_{\mathcal{S}}(w+\epsilon) - L_{\mathcal{S}}(w)}_{\text{衡量sharpness}}] + L_{\mathcal{S}}(w) + h(\frac{\|w\|_2^2}{\rho})$$

- 故，我们最小化右端，用$\lambda\|w\|_2^2$代替$h(\frac{\|w\|_2^2}{\rho})$，得到如下min-max 问题(可以将l2球变为一般的lp球) :

$$\min_w L_{\mathcal{S}}^{SAM}(w) + \lambda\|w\|_2^2, \quad \text{这里} \quad L_{\mathcal{S}}^{SAM} := \max_{\|\epsilon\|_p \leq \rho} L_{\mathcal{S}}(w+\epsilon)$$

[25] 见Foret, Pierre, et al. 2020, 定理1

- 关于ϵ极大化问题难以求解，做一阶泰勒展开

$$\epsilon^*(w) = \arg\max_{\|\epsilon\|_p \le \rho} L_{\mathcal{S}}(w+\epsilon) \approx \arg\max_{\|\epsilon\|_p \le \rho} L_{\mathcal{S}}(w) + \epsilon^T \nabla_w L_{\mathcal{S}}(w)$$

- 故有近似解

$$\hat{\epsilon}(w) = \rho \,\mathrm{sign}(\nabla_w L_{\mathcal{S}}(w)) \frac{|\nabla_w L_{\mathcal{S}}(w)|^{q-1}}{(\|\nabla_w L_{\mathcal{S}}(w)\|_q^q)^{1/p}}$$

- 令$p=2$,则SAM算法迭代如下：

$$\begin{cases} \epsilon_t(w) = \rho \frac{\nabla_w L_{\mathcal{S}}(w_t)}{\|\nabla_w L_{\mathcal{S}}(w_t)\|_2} \\ w_{t+1} = w_t - \alpha_t(\nabla L_{\mathcal{S}}(w_t + \epsilon_t(w)) + \lambda w_t) \end{cases} \tag{21}$$

随机梯度法，亦可使用*Adam*等

其中$\nabla L_{\mathcal{S}}(w_t + \epsilon_t(w)) \approx \nabla_w L_{\mathcal{S}}(w)\,|_{w_t+\epsilon_t(w)}$

- 优点：SAM可以提高泛化能力(通常提高1%左右)；缺点：每次迭代需要计算2次梯度。

Outline

LION[26](EvoLved Sign Momentum)

- 谷歌团队将优化算法的发现定义为程序搜索。他们在一个既无限又稀疏的程序空间内部署高级搜索技术，利用进化算法以及如热启动和重启的技术提高效率。

Program 2: An example training loop, where the optimization algorithm that we are searching for is encoded within the `train` function. The main inputs are the weight (`w`), gradient (`g`) and learning rate schedule (`lr`). The main output is the `update` to the weight. `v1` and `v2` are two additional variables for collecting historical information.

```
w = weight_initialize()
v1 = zero_initialize()
v2 = zero_initialize()
for i in range(num_train_steps):
  lr = learning_rate_schedule(i)
  g = compute_gradient(w, get_batch(i))
  update, v1, v2 = train(w, g, v1, v2, lr)
  w = w - update
```

Program 3: Initial program (AdamW). The bias correction and ϵ are omitted for simplicity.

```
def train(w, g, m, v, lr):
  g2 = square(g)
  m = interp(g, m, 0.9)
  v = interp(g2, v, 0.999)
  sqrt_v = sqrt(v)
  update = m / sqrt_v
  wd = w * 0.01
  update = update + wd
  lr = lr * 0.001
  update = update * lr
  return update, m, v
```

Program 4: Discovered program after search, selection and removing redundancies in the raw Program 8. Some variables are renamed for clarity.

```
def train(w, g, m, v, lr):
  g = clip(g, lr)
  g = arcsin(g)
  m = interp(g, v, 0.899)
  m2 = m * m
  v = interp(g, m, 1.109)
  abs_m = sqrt(m2)
  update = m / abs_m
  wd = w * 0.4602
  update = update + wd
  lr = lr * 0.0002
  m = cosh(update)
  update = update * lr
  return update, m, v
```

[26]Chen, Xiangning, et al. "Symbolic discovery of optimization algorithms." Advances in Neural Information Processing Systems 36 (2024).

LION

Algorithm 1 AdamW Optimizer

given $\beta_1, \beta_2, \epsilon, \lambda, \eta, f$
initialize $\theta_0, m_0 \leftarrow 0, v_0 \leftarrow 0, t \leftarrow 0$
while θ_t not converged **do**
  $t \leftarrow t + 1$
  $g_t \leftarrow \nabla_\theta f(\theta_{t-1})$
  update EMA of g_t **and** g_t^2
  $m_t \leftarrow \beta_1 m_{t-1} + (1 - \beta_1) g_t$
  $v_t \leftarrow \beta_2 v_{t-1} + (1 - \beta_2) g_t^2$
  bias correction
  $\hat{m}_t \leftarrow m_t / (1 - \beta_1^t)$
  $\hat{v}_t \leftarrow v_t / (1 - \beta_2^t)$
  update model parameters
  $\theta_t \leftarrow \theta_{t-1} - \eta_t (\hat{m}_t / (\sqrt{\hat{v}_t} + \epsilon) + \lambda \theta_{t-1})$
end while
return θ_t

Algorithm 2 Lion Optimizer (ours)

given $\beta_1, \beta_2, \lambda, \eta, f$
initialize $\theta_0, m_0 \leftarrow 0$
while θ_t not converged **do**
  $g_t \leftarrow \nabla_\theta f(\theta_{t-1})$
  update model parameters
  $c_t \leftarrow \beta_1 m_{t-1} + (1 - \beta_1) g_t$
  $\theta_t \leftarrow \theta_{t-1} - \eta_t (\text{sign}(c_t) + \lambda \theta_{t-1})$
  update EMA of g_t
  $m_t \leftarrow \beta_2 m_{t-1} + (1 - \beta_2) g_t$
end while
return θ_t

- LION（演化符号动量）旨在比传统优化器（如Adam）更节省内存，因为它主要跟踪动量并使用符号操作进行更新。它提供所有维度上均匀的更新幅度，与自适应优化器不同，后者的更新可能因参数而异。
- 在多个领域（包括图像分类、扩散模型和语言建模）中展示了卓越性能，包括在视觉任务（如ImageNet分类）中准确度提高2%，以及在训练中显著的计算节省（在某些情况下计算减少高达5倍）。

学习优化: Learning to optimize

陈士祥

中国科学技术大学

Acknowlegement：This lecture note is based on Prof. Wotao Yin's tutorial 《Learning to Optimize》

Outline

机器学习**(ML)**	优化**(OPT)**
答案作为现有数据给出	没有给出答案，但我们知道如何评估答案。
ML 从数据中学习，以在未来给出答案	OPT 找到具有最佳评估的答案

L2O (Learning to optimize)使用经验来“更快地优化”或“ 生成更好的解决方案”。

Classic Optimization vs Learning-to-Optimize

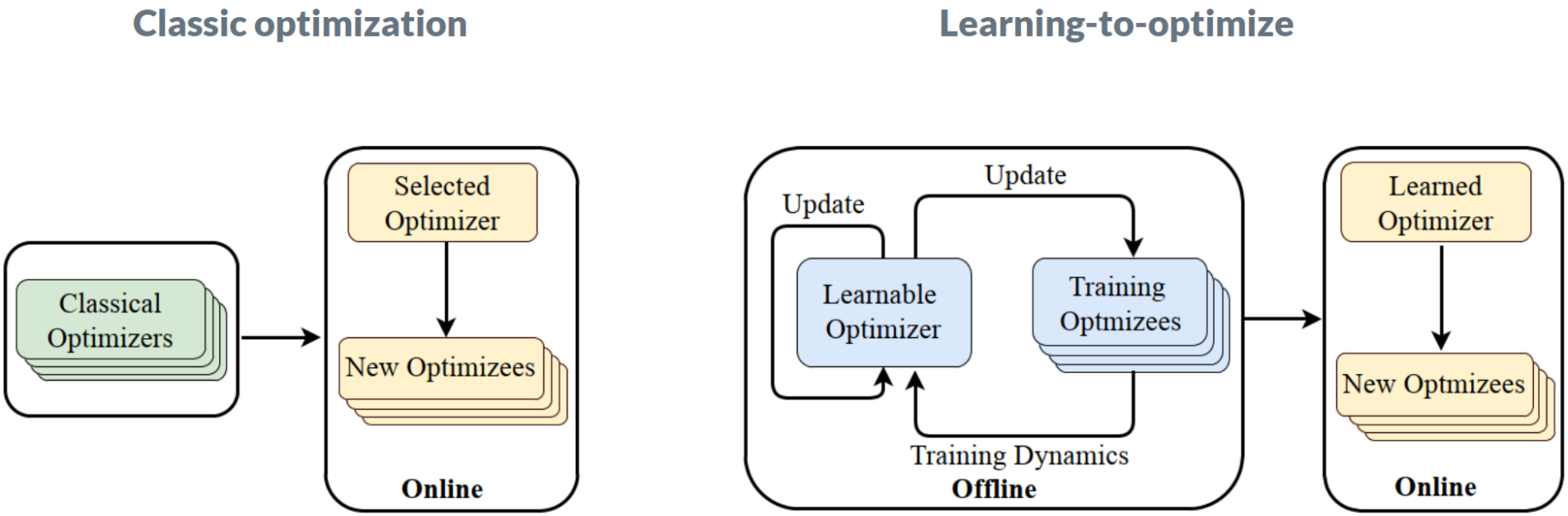


Figure: 传统优化器和学习优化对比，图片来源[1]

[1] Chen, Tianlong, et al. "Learning to optimize: A primer and a benchmark." Journal of Machine Learning Research 23.189 (2022): 1-59.

何时考虑L2O

情况一	情况二
已有很多好的解决方案样例，但很难写出好的解析模型 (例如，逆问题)	有很多好的解决方案样例，或者反复解决类似的优化问题, 实际问题服从某个分布$\mathcal{T}$.
$\min f(x)$ 但是目标函数建模不够好	$\min f(x), f \sim \mathcal{T}$
L2O 通过学习模型或方法找到更好的解决方案。	L2O 通过采取“快速捷径”找到相似的解决方案。

基本形式

传统优化算法考虑目标函数：

$$\min f(x)$$

- 考虑梯度下降(GD) 迭代：$x_{t+1} = x_t - \alpha f(x_t)$
- 引入自由参数：
  - z_t表示t时刻的输入，它包括迭代到t 时的所有迭代点和梯度；
  - $x_{t+1} = x_t - g(z_t, \phi)$, 这里使用$\phi$ 参数化g, 其表示某个神经网络，例如LSTM
- L2O中的目标函数多种多样，例如(Andrychowicz et al'NIPS16[2]):

$$\min \mathbf{E}_{f \in \mathcal{T}} \left[\sum_{t=1}^{T} w_t f(x_t) \right]$$

$$x_{t+1} = x_t - g(z_t, \phi), t = 1, \ldots, T-1.$$

$f \in \mathcal{T}$表示从优化问题的分布$\mathcal{T}$中采样出函数，w_t是权重。

[2]Learning to learn by gradient descent by gradient descent. In Advances in neural information processing systems, pages 3981–3989, 2016.

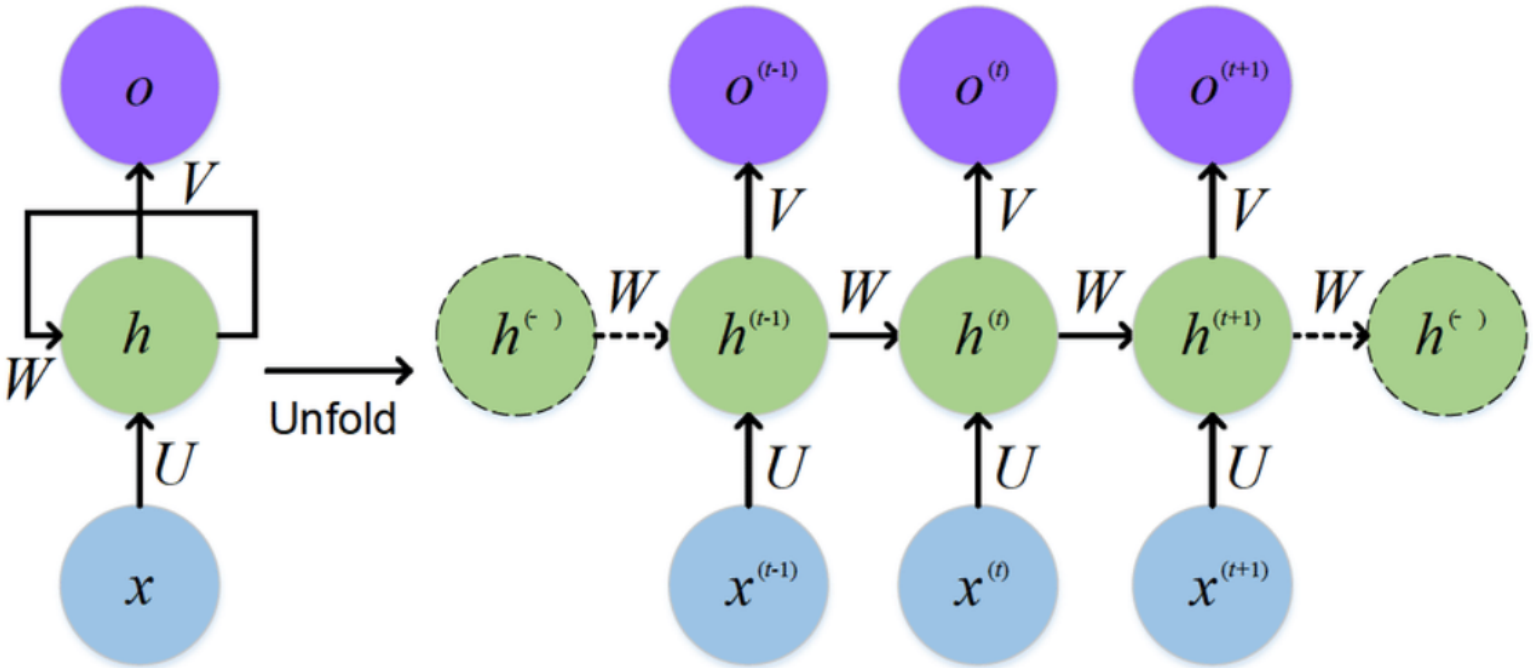
基于模型vs 非基于模型

基于模型(**model-based**)	非基于模型(**model-free**)
g以已有的优化方法为基本更新格式或作为起点	g 基于神经网络的通用逼近能力，例如多层神经网络或循环神经网络
L2O 搜索某些参数的最佳值	L2O 设置为发现完全新的优化算法规则，而不参考任何现有更新格式
可以将此L2O 以各种方式与经典优化方法结合	

此处的“模型”，并非机器学习中的模型，而是指传统优化算法的某种框架、结构。

Outline

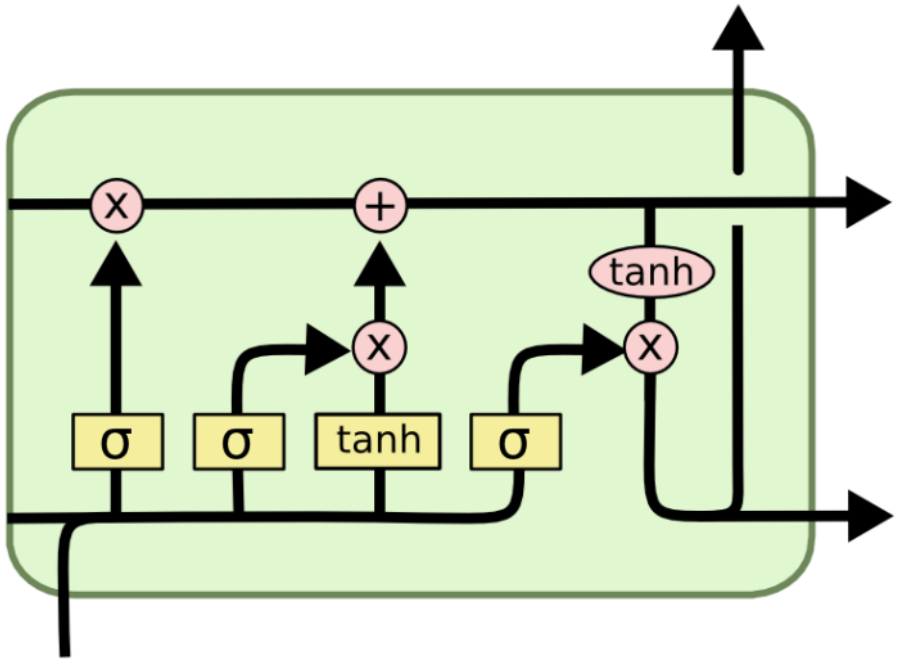
model-free L2O, 基于RNN，特别是LSTM



Wichrowska et al'17; Metz et al'ICML19; Li-Malik'ICLR17; Bello et al'ICC17; Jiang et al'18;

Figure: RNN

许多无模型L2O 使用LSTM



Andrychowicz et al'NIPS16; Chen et al'ICML17; Lv-Jiang-Li'17; Cao et al'NeurIPS19; Xiong-Hsieh'20

Figure: model-free L2O

Optimizer Architecture	Input Feature	Meta Training Objective	Additional Technique	Evaluation Metric
LSTM	Gradient	Meta Loss	Transform input gradient ∇ into $\log(\nabla)$ and $\text{sign}(\nabla)$	Training Loss
LSTM	Objective Value	Objective Value	N/A	Objective Value
LSTM	Gradient	Meta Loss	Random Scaling Combination with Convex Functions	Training Loss
Hierarchical RNNs	Scaled averaged gradients, relative log gradient magnitudes, relative log learning rate	Log Meta Loss	Gradient History Attention Nesterov Momentum	Training Loss
MLP	Gradient	Meta Loss	Unbiased Gradient Estimators	Training Loss Testing Loss
RNN Controller	Loss, Gradient	Meta Loss	Coordinate Groups	Training Loss
Searched Mathematical Rule by Primitive Functions	Scaled averaged gradients	Meta Loss	N/A	Testing Accuracy
Multiple LSTMs	Gradient, momentum, particle's velocity and attraction	Meta Loss and Entropy Regularizer	Sample- and Feature- Attention	Training Loss
RNN	Input Images, Input Gradient	Meta Loss	N/A	Standard and Robust Test Accuracies
LSTM	Input Gradient	Meta Loss	N/A	Training Loss and Robust Test Accuracy

Figure: model-free L2O

model-free L2O的局限性

model-free L2O中，网络层数是影响效果的重要变量，然而：

- 深层网络具有高内存成本
- 浅层网络无法运行更多迭代

Outline

model-based L2O

- model-free L2O 完全利用神经网络寻找新的优化器
- model-based L2O受传统优化算法的迭代格式启发，设计独有的L2O
- 利用神经网络训练出剩余的参数，因此可以被看成为“半参数化”
- 常见的分类：Unrolling, Safeguarding, Plug-and-Play, 等
- 优点：传统优化算法可能要上百次或几千次迭代收敛，而L2O只需几十层。

Unrolling

- 例：LASSO

$$\min \frac{1}{2}\|b - Ax\|^2 + \lambda\|x\|_1,$$

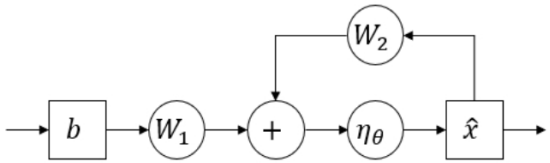
传统方法，需要调整参数λ,使用近似点梯度法（ISTA: iterative shrinkage and Thresholding Algorithm):

$$x_{k+1} = \eta_{\lambda/L}(x_k + \frac{1}{L}A^T(b - Ax_k))$$

- 将ISTA改写为：

$$x_{k+1} = \eta_\theta(W_1 b + W_2 x_k),$$

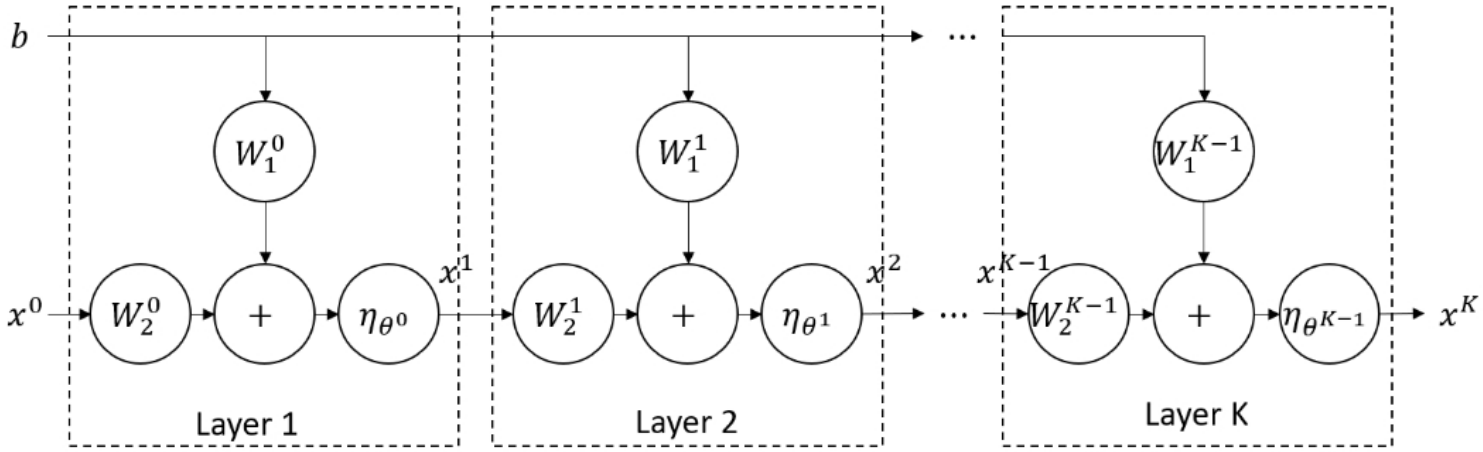
where $W_1 = \frac{1}{L}A^T$, $W_2 = I_n - \frac{1}{L}A^T A$, $\theta = \lambda/L$.



(a) RNN structure of ISTA.

Figure: From [3]

[3] Chen, Xiaohan, et al. "Theoretical linear convergence of unfolded ISTA and its practical weights and thresholds." Advances in Neural Information Processing Systems 31 (2018).



(b) Unfolded learned ISTA Network.

Figure: From [4]

- 将传统算法做截断，限制为K次迭代，端到端训练
- 可以减少参数而不会影响性能(Chen et al. NeurIPS'18 & Liu et al. ICLR'19)
- 在逆问题、PDE 和图模型中已很常见，且取得了成功

[4] Chen, Xiaohan, et al. "Theoretical linear convergence of unfolded ISTA and its practical weights and thresholds." Advances in Neural Information Processing Systems 31 (2018).

挑战：

- 展开长度：更多层K带来更好性能但训练困难；
- 容量：仅在非常少的情况下可以理论证明其效果(Liu et al'ICLR18)；
- 可训练性：缺乏性能保证；
- 泛化：但测试问题的数据，不服从训练数据分布时，L2O会失效。当L2O失败时该怎么办?

Safe-guarding

- (Heaton et al.20) L2O 收敛性可通过引入“能量”E来保证

$$x^{k+1} = \begin{cases} \text{L2O update } z^k & \text{if } E^t(z^k) \leq E^t(x^k) \\ \text{Classic update } T(x^k) & \text{otherwise} \end{cases}$$

- 当L2O未能降低能量时，传统的优化算法迭代T 将在该迭代中接管

测试案例：从噪声测量中恢复稀疏向量

给定观测b_q, 假设其由如下方式生成

$$b_q = Ax_q^* + \epsilon_q$$

其中，x_q^*是稀疏的真实信号，ϵ_q是噪声

- 固定A；改变稀疏向量和噪声
- 训练损失是相对于真实信号的平方损失：$\|x_K - x_q^*\|^2$
- 展开至16层
- 定义归一均方误差：NMSE(Normalized Mean Squared Error):

$$\mathrm{NMSE}_{dB}(x_K, x_q^*) = 10\log_{10}(\|x_K - x_q^*\|^2 / \|x_q^*\|^2)$$

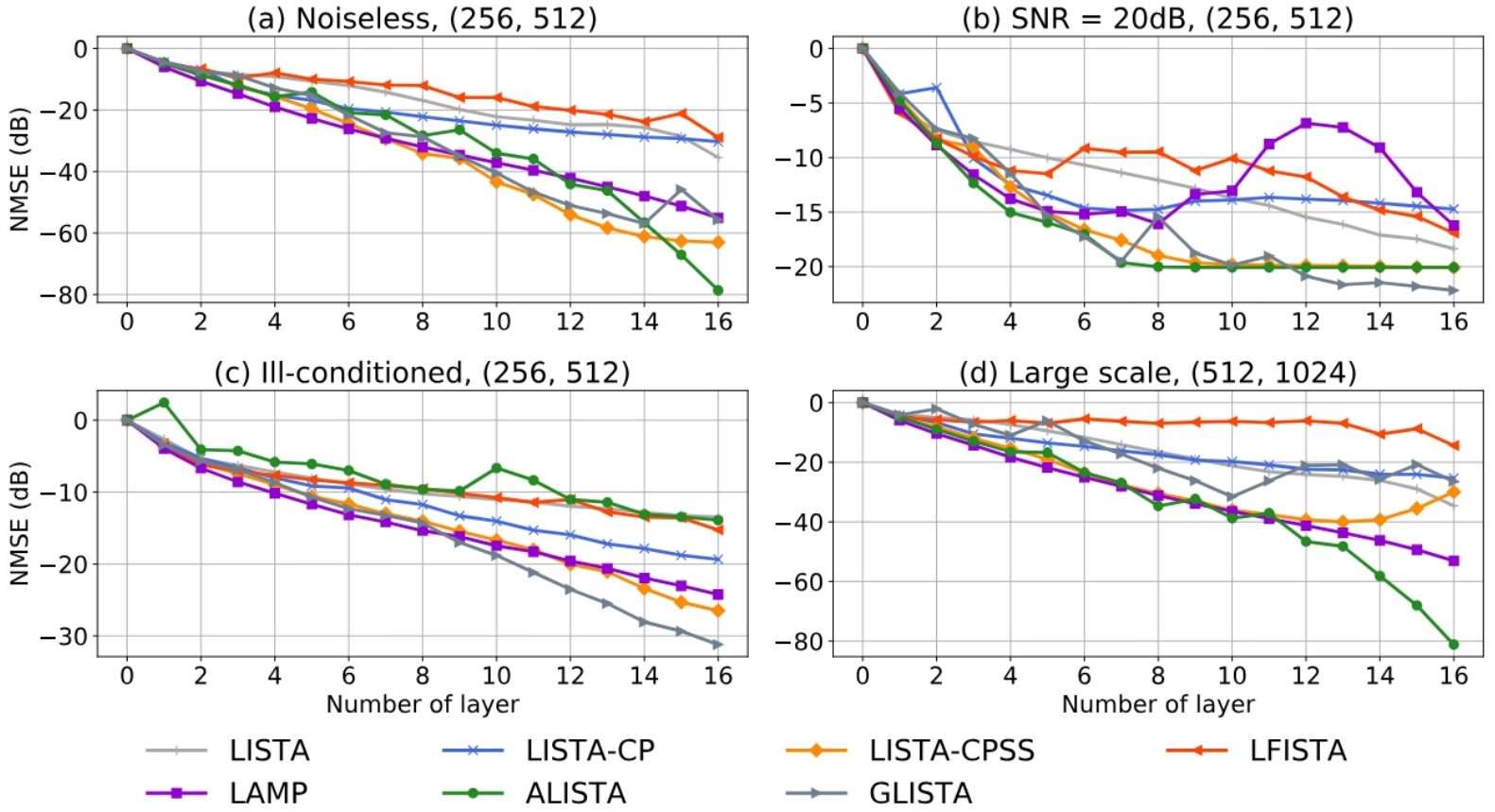


Figure 5: Results of sparse recovery in four different settings: (a) noiseless with $(m, n) = (256, 512)$; (b) additive Gaussian measurement noises of SNR=20dB and $(m, n) = (256, 512)$; (c) coherent dictionary with $(m, n) = (256, 512)$; (d) larger scale with $(m, n) = (512, 1024)$. The x-axis counts the layers and the y-axis is the NMSE of the recovery.

Figure: 图片来源[5]

[5]Chen, Tianlong, et al. "Learning to optimize: A primer and a benchmark." Journal of Machine Learning Research 23.189 (2022): 1-59.

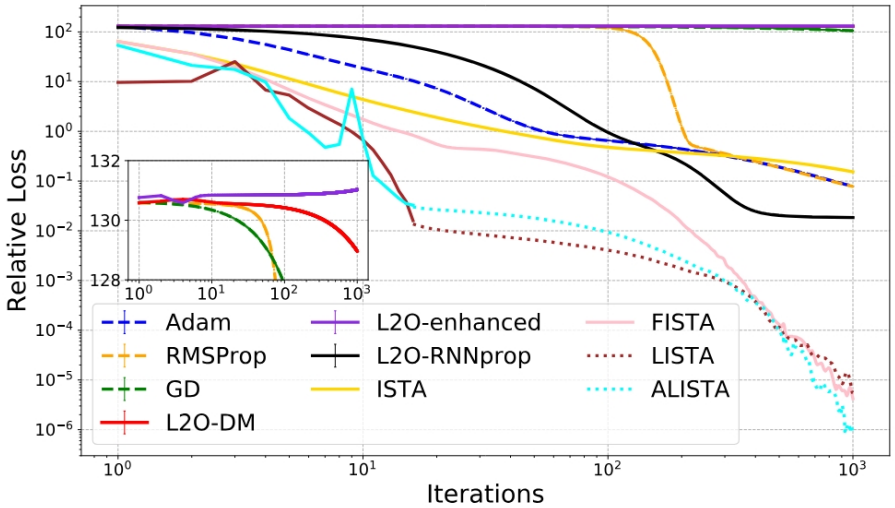
与传统优化方法比较

- 传统LASSO模型的解记为：

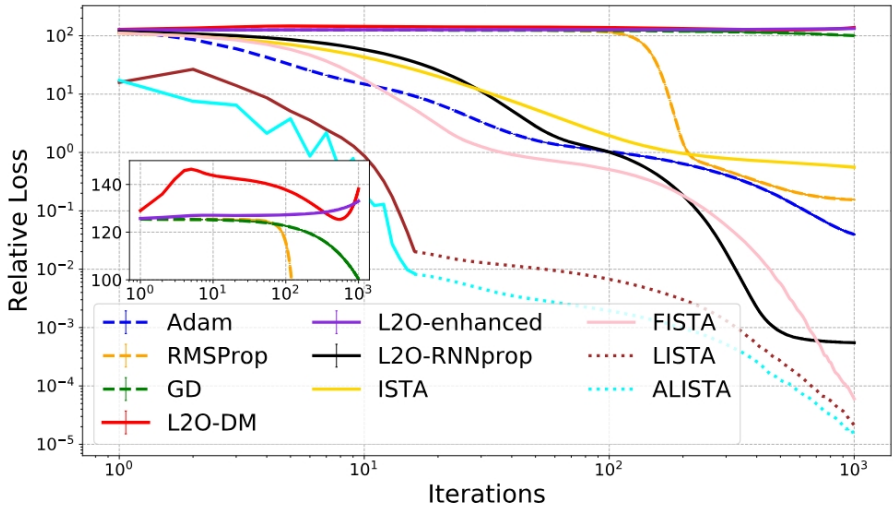
$$x_q^{\text{Lasso}} = \arg\min_x f_q(x), \quad \text{where } f_q(x) = \frac{1}{2}\|b - Ax\|^2 + \lambda\|x\|_1.$$

- 定义相对误差

$$R_{f,Q}(x) = \frac{\mathbf{E}_{q\sim Q}[f_q(x) - f_q^*]}{\mathbf{E}_{q\sim Q}[f_q^*]}$$



(a) $(m,n)=(5,10)$



(b) $(m,n)=(25,50)$

Figure 6: Evaluation comparisons among analytic, model-based L2O, and model-free L2O optimizers on Lasso. y-axis represents the modified relative loss (21), and x-axis denotes the number of iterations, both in the logarithmic scale.

Figure: 图片来源[6]

[6]Chen, Tianlong, et al. "Learning to optimize: A primer and a benchmark." Journal of Machine Learning Research 23.189 (2022): 1-59.

Outline

Plug-and-Play：背景

- 考虑
$$\min_x f(x) + \gamma g(x)$$
- 改写为
$$\min_{x,y} f(x) + \gamma g(y), \text{ s.t. } x = y.$$
- ADMM:
$$\begin{aligned} x^{k+1} &= \text{Prox}_{\beta g}\left(y^k - u^k\right) \\ y^{k+1} &= \text{Prox}_{\alpha f}\left(x^{k+1} + u^k\right) \\ u^{k+1} &= u^k + x^{k+1} - y^{k+1}. \end{aligned}$$
- 在成像中的解释
  - 步骤1：带噪声图像$\rightarrow$减少噪声的图像. $\text{Prox}_{\beta g}$可以看为去噪器
  - 步骤2：不一致$\rightarrow$更一致的数据

非邻近去噪器

- SOTA去噪器是某些函数的代理操作：
  - NLM, BM3D, CNN
- 但仍有解释：

$$H_\sigma : \textit{noisy image} \to \textit{less noisy image}$$

- 问题：如何集成到如ADMM的迭代中？

Plug-and-Play

- (Venkatakrishnan et al'GlobalSIP13) PnP ADMM：

$$\begin{aligned} x^{k+1} &= \mathrm{Prox}_{\beta g}\left(y^k - u^k\right) \\ y^{k+1} &= \mathrm{Prox}_{\alpha f}\left(x^{k+1} + u^k\right) \\ u^{k+1} &= u^k + x^{k+1} - y^{k+1} \end{aligned} \qquad \Rightarrow \qquad \begin{aligned} x^{k+1} &= H_\sigma\left(y^k - u^k\right) \\ y^{k+1} &= \mathrm{Prox}_{\alpha f}\left(x^{k+1} + u^k\right) \\ u^{k+1} &= u^k + x^{k+1} - y^{k+1}. \end{aligned}$$

  - 即插：训练好的去噪器H_σ插入已有的算法中。
  - 即用：在新的任务中直接使用。
- 实际表现惊人地出色！

示例：超分辨率

Low resolution input

Other method

Other method

Other method

Other method

Other method

PnP-ADMM with BM3D

(Chen-Wang-Elgendy'17)

- 优点：
  - 无限深度（展开长度不是问题）
  - 具有收敛保证(Ryu et al'ICML19) 如果
    - $I - H_\sigma$ 是Lipschitz
    - f 是强凸的
- 局限：
  - 先训练去噪器H_σ在插入（训练非端到端）
  - 好的泛化性无法解释

Outline

总结

- 学习优化是一种新型优化范式, 由数据驱动。
- 在以下情况下有用：
  - 使用数据改进建模和方法
  - 使用数据找到优化捷径