NLP 总结

李宗军

青岛理工大学 · 理学院

2016 年

要点

1. 基本概念
2. 凸规划
3. 一维搜索问题
4. 无约束最优化
   - 最速下降算法
   - Newton 法
   - 共轭方向法
5. 约束最优化

基本概念

1. NLP 模型
2. NLP 最优解定义
3. 迭代算法框架

NLP 模型

$$\min f(x)$$
$$\begin{cases} g(x) & \le 0 \\ h(x) & = 0 \end{cases}$$

1. 无约束
2. 值含有等式约束
3. 只含有不等式约束
4. 两种约束都含有
5. 线性规划是 NLP 的特殊情况

NLP 最优解的定义

1. 局部最优解
2. 严格局部最优解
3. 全局最优解
4. 严格全局最优解
5. 说明：
   - 求解 NLP 就是获得全局最优解，但是难度太大
   - 可考虑先获得局部最优解，再在多个局部最优解中挑选最好的
   - 某些特殊的 NLP，例如凸规划，其局部最优解就是全局最优解，从而引出了图规划的研究

NLP 迭代算法框架

1. 给定初始点 $x^0, k := 0$
2. 停止计算条件
   - 一般由最优性条件构造，从而引出了各种 NLP 模型的最优性条件
3. 迭代方向 p^k
   - 不同的产生方式导致了各种不同的算法
   - 必须遵守: 是可行下降方向
   - 下降方向必须满足 $-\nabla f(x^k)^T p^k \geq 0$
   - $p^k = -H\nabla f(x^k)$ 是一种一般的取法，其中 H 是半正定矩阵
   - 当 H 是单位矩阵时，就是最速下降法，当 H 是 f 的 Hessian 矩阵的逆矩阵时，就是 Newton 法,...
4. 迭代步长: 一维搜索（精确、非精确）
5. $k := k + 1$, 转向 2.

凸规划 I

1. 凸集：$\forall x_1, x_2 \in S, \forall \lambda \in [0,1], \lambda x_1 + (1-\lambda)x_2 \in S$
2. 凸函数
   - 定义：$\forall x_1, x_2 \in S, \forall \lambda \in [0,1], f(\lambda x_1 + (1-\lambda)x_2) \leq \lambda f(x_1) + (1-\lambda)f(x_2)$
   - 性质:
     - 凸函数的非负数乘仍凸
     - 两个凸函数之和仍凸
     - 两个凸函数之积不一定仍凸
   - 判定
     - 凸函数定义
     - 判定定理（梯度/导数，Hesse 矩阵/二阶导数）
3. 凸集与凸函数的关系定理：凸函数的水平截集仍是凸集
4. 凸规划

凸规划 II

1. 定义：f 是凸函数，可行区域是凸集
2. 性质：任意局部最优解都是全局最优解
3. 判定：NLP 中非线性函数都是凸函数

一维搜索问题

1. 问题模型及其背景
2. 精确搜索算法:0.618,Newton
3. 非精确算法
   - Goldstein:
     - $\varphi(t_k) \leq \varphi(0) + m_1\varphi'(0)t_k$
     - $\varphi(t_k) \geq \varphi(0) + m_2\varphi'(0)t_k$
   - Armijo:
     - $\varphi(t_k) \leq \varphi(0) + m\varphi'(0)t_k$
     - $\varphi(Mt_k) \geq \varphi(0) + m\varphi'(0)Mt_k$

无约束最优化

1. 问题模型
2. 最优性条件
3. 算法
   - 最速下降法
   - Newton 法
   - 共轭方向法（共轭梯度法）

无约束最优性条件

1. 无约束最优化问题：$\min\{f(x) | x \in R^n\}$
2. 最优性条件（四个定理 4.4.1–4.4.4）:
   - 下降方向的判定：$\nabla f(x)^T p < 0$
   - 局部最优解的必要条件：$\nabla f(x^*) = 0$
   - 局部最优解的充分条件：$\nabla f(x^*) = 0, \nabla^2 f(x^*)$ 正定
   - 凸规划的最优解的充要条件：$\nabla f(x^*) = 0$

最速下降算法

1. 算法思想
   迭代方向采用 $-\nabla f(x^k)$,步长采用精确的一维搜索算法。
2. 最优性条件：$\nabla f(x^*) = 0$
3. 迭代终止条件：$\|\nabla f(x^k)\| \leq \varepsilon$
4. 初始点：任意 x^0 都可以

迭代步骤

1. init: 给定初始点 $x^0, k = 0, \varepsilon > 0$
2. 若 $\|\nabla f(x^k)\| \leq \varepsilon$, 则输出近似解 x^k,stop; 否则转型下一步
3. 取 $p^k = -\nabla f(x^k)$, 构造一维搜索问题
   min $\varphi(t) = f(x^k + tp^k)$, 采用 0.618 法求解之， 得 t_k
4. $x^{k+1} = x^k + t_k p^k, k := k + 1$, 转向步骤 2.

算法分析

1. 具有全局收敛性
2. 每次迭代，计算量较小
3. 迭代速度较慢
4. 改进：用非精确搜索代替精确搜索

Newton 法

1. 算法思想

$$f(x) \approx f(x^k) + \nabla f(x^k)^T(x - x^k) + \frac{1}{2}(x - x^k)^T \nabla^2 f(x^k)(x - x^k)$$
$$= g(x)$$

let $\nabla g(x) = 0$, 解得 $x^{k+1} = x^k - \nabla^2 f(x^k)^{-1} \nabla f(x^k)$

2. Newton 法迭代步骤
   1. init: 给定充分靠近最优解的初始点 x^0, 误差 $\varepsilon > 0, k = 0$
   2. 若 $\|\nabla f(x^k)^T\| \leq \varepsilon$, 则输出 x^k， stop
   3. $x^{k+1} = x^k - \left[\nabla^2 f(x^k)\right]^{-1} \nabla f(x^k)$
   4. $k = k + 1$, 转向步骤 2.

Newton 法特点

同一维搜索的 Newton 法

1. 具有局部收敛性
2. 只有当初始点充分靠近最优解时，才能保证算法的收敛性
   1. 改进：$x^{k+1} = x^k - t_k \left[\nabla^2 f(x^k)\right]^{-1} \nabla f(x^k), t_k$ 采用一维搜索
   2. 即迭代方向 $p^k = -\left[\nabla^2 f(x^k)\right]^{-1} \nabla f(x^k)$
3. 具有二阶收敛速度
4. 每次迭代步的计算量较大：$\left[\nabla^2 f(x^k)\right]^{-1}$

共轭方向法

简介:

1. 计算量介于最速下降法和 Newton 法之间
2. 为二次规划设计的
3. 调整后，可适用于非线性规划，依据是在最优解附近，非线性函数近似于二次函数
4. 迭代方向采用共轭方向
5. 步长采用精确一维搜素

共轭方向

1. 共轭方向：$p, q \neq 0, p^T A q = 0$, 其中 A 是实对称矩阵。
   - 注：正交是共轭的特殊情况。
2. 共轭方向组：对 $p^1, \cdots, p^n$ 中的任何两个不同的向量，满足 $(p^i)^T A p^j = 0, i \neq j$
3. 共轭方向组的性质
   - 共轭方向组是线性无关组（Th4.4.5）

共轭方向法定理

对二次严格凸函数 $\min\ f(x) = \frac{1}{2}x^TAx + b^Tx$，其中 A 是 n 阶实对称正定矩阵：
依次沿着任意一组共轭方向：$p^1, \cdots, p^n$, 采用精确一维搜索，则最多经过 n 轮迭代就可获得整体最优解。

迭代步骤

1. 给定 $x^0, \varepsilon > 0$
2. 计算 $\nabla f(x^k)$, 若 $\left\|\nabla f(x^k)\right\| \leq \varepsilon$,stop, 输出 x^k
3. 令 $k = 0$,$p^k = -\nabla f(x^k)$
4. 采用一维搜索，求解 $\min\ \varphi(t) = f(x^k + t\,p^k)$, 解得 t_k.
5. $x^{k+1} = x^k + t_k p^k$,k++
6. While $k \leq n$:
    1. 若 $\nabla f(x^k)$, 若 $\left\|\nabla f(x^k)\right\| \leq \varepsilon$, *stop*, 输出$x^k$
    2. $\lambda_k = \frac{\left\|\nabla f(x^k)\right\|^2}{\|p^k\|^2}$,$p^{k+1} = -\nabla f(x^k) + \lambda_k p^k$
    3. 采用一维搜索，求解 $\min\ \varphi(t) = f(x^k + t\,p^{k+1})$, 解得 t_k.$x^{k+1} = x^k + t_k p^{k+1}$
    4. k++
7. 转向步骤 2.

约束最优化

1. 约束最优化模型
2. 最优性条件：K-T 条件
3. 算法
   - 简约梯度法
   - 罚函数法
   - 内点障碍函数法

K-T 条件 I

$$\nabla f(x) + \sum_{i \in I(x^*)} \lambda_i \nabla g_i(x) + \sum_{j \in J} u_j \nabla h_j(x) = 0, \lambda_i \geq 0, i \in I$$

等价于

$$\begin{aligned} \nabla f(x) + \sum_{i \in I} \lambda_i \nabla g_i(x) + \sum_{j \in J} u_j \nabla h_j(x) &= 0 \\ \lambda_i g_i(x) &= 0, i \in I \\ \lambda_i &\geq 0, i \in I \end{aligned} \tag{1}$$

其中 $\lambda_i g_i(x) = 0$ 称为互补松紧条件。

K-T 条件 II

1. K-T 条件是 NLP 最优解的必要条件!
2. 凸规划的充要条件：定理 4.5.2
3. K-T 条件的作用是：把最优化问题，转换为方程组问题，然后运用方程组的求解理论方法来解决。
4. 注意：K-T 条件适用于 LP 问题（熟练写出 LP 三种模型的 K-T 条件）

简约梯度法

1. 考虑的问题

$$\min f(x)$$

$$st.\begin{cases} Ax & = b \\ x & \geq 0 \end{cases}$$

2. 算法思想

   仿照 LP 中的单纯形法，在每一轮迭代的当前点 x^k 处，把 x^k 分为基变量部分和非基变量部分，则 $f(x) = F(x_N)$, 采用 $F(x_N)$ 的负梯度方向构造可行下降方向 p，迭代步长采用一维搜索方法。

定理 4.5.3

1. 若 $p \neq 0$, 则 p 是 $f(x)$ 在 x^k 处的可行下降方向;
2. $p = 0$ 的充要条件是 x^k 是 K-T 点。

迭代步骤 I

1. 算法初始化，给定初始点 $x^0, \varepsilon > 0, k := 0$
2. 设 I_B^k 是 x^k 的 m 个最大分量的下标集，对矩阵 A 进行分解 $A = (B_k, N_k)$
3. 计算 $\nabla f(x^k) = \left(\nabla_B f(x^k), \nabla_N f(x^k)\right)^T$，然后计算简约梯度 $r_N^k = \left(B_k^{-1} N_k\right)^T \nabla_B f(x^k) - \nabla_N f(x^k)$. 记 r_N^k 的第 i 个分量为 r_i^k
4. 构造 $p^k = (p_B^k, p_N^k)^T$, 其中
$$p_N^k = (p_i^k) = \begin{cases} r_i^k, & r_i^k \leq 0 \\ x_i^k r_i^k, & r_i^k > 0 \end{cases}, p_B^k = -B_k^{-1} N_k p_N^k.$$
若 $\left\|p^k\right\| \leq \varepsilon$, 停止迭代，输出 x^k.
5. 采用一维搜索，求解 $\min\left\{f\left(x^k + tp^k\right) | 0 \leq t \leq t_{\max}^k\right\}$, 得到最优解 t_k, 其中 $t_{\max}^k = \begin{cases} +\infty, & p^k \geq 0 \\ \min\left\{-\frac{x_i^k}{p_i^k} | p_i^k < 0\right\}, & \text{else} \end{cases}$

迭代步骤 II

6. $x^{k+1} = x^k + t_k p^k, k := k + 1$, 转向 2.

算法说明

1. 数值实验表明：对于大规模的线性约束的 NLP，简约梯度法是最好的
2. 对约束最优化而言，迭代方向必须是可行下降方向，此类算法统称为可行下降算法。
3. 由定理 4.5.3（$p^k = 0$ 的充要条件是 x^k 是 K-T 点。），从而迭代终止条件采用简单的 $\|p^k\| \le \varepsilon$，而不是采用复杂的 K-T 条件。
4. 初始点，可采用 LP 中的两阶段法获得。

罚函数法

1. 算法思想
   利用问题中的约束，构造适当的带有参数的惩罚函数，然后在转移到原来的目标函数上，以构造增广目标函数，把约束 NLP 问题转化为求解一系列无约束 NLP 问题。
2. 序列无约束极小化方法
   这种把约束 NLP 的求解转换为一系列无约束优化问题求解的方法，统称为序列无约束极小化方法。
3. 增广函数构造
   构造增广函数：$F(x) = f(x) + M_k p(x)$

要点

1. 基本概念
2. 凸规划
3. 一维搜索问题
4. 无约束最优化
   - 最速下降算法
   - Newton 法
   - 共轭方向法
5. 约束最优化

1. 基本概念
2. 凸规划
3. 一维搜索问题
4. 无约束最优化
   - 最速下降算法
   - Newton 法
   - 共轭方向法
5. 约束最优化