第十二章 优化算法

第 34 讲 无约束优化算法：二阶优化算法

黄定江

DaSE @ ECNU

djhuang@dase.ecnu.edu.cn

1. 34.1 牛顿法
2. 34.2 拟牛顿法

- 从上一讲中可以看出，梯度法仅使用了目标函数的一阶信息。
- 如果函数足够光滑，那么就可以使用更多的信息，例如二阶信息。
- 直观上，可以期望得到更好的优化算法。这就是本讲我们将探究的牛顿类方法。

34.1.1 牛顿法

函数 f 在 $\boldsymbol{x}$ 处的二阶 Taylor 近似 (或模型)$\hat{f}$ 为

$$\hat{f}(\boldsymbol{x}+\boldsymbol{v}) \approx f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})^T \boldsymbol{v} + \frac{1}{2}\boldsymbol{v}^T \nabla^2 f(\boldsymbol{x})\boldsymbol{v} \tag{1}$$

这是 $\boldsymbol{v}$ 的二次凸函数，当 Hessian 矩阵为半正定时，显然在 $\boldsymbol{v} = \Delta \boldsymbol{x}_{nt}$ 处达到最小值。如下图所示：

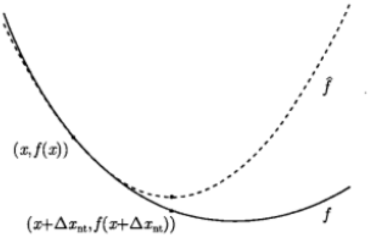

图 1

Newton 步径

若 Hessian 矩阵 $\nabla^2 f(\boldsymbol{x})$ 正定，对二阶近似求极小，可得**牛顿方程** $\nabla^2 f(\boldsymbol{x})\Delta\boldsymbol{x}_{nt} = -\nabla f(\boldsymbol{x})$，并解得：

$$\Delta\boldsymbol{x}_{nt} = -\nabla^2 f(\boldsymbol{x})^{-1}\nabla f(\boldsymbol{x}).$$

它被称之为（f 在 $\boldsymbol{x}$ 处的）**Newton 步径**。

由正定性可知，除非 $\nabla f(\boldsymbol{x}) = 0$，否则就有

$$\nabla f(\boldsymbol{x})^T\Delta\boldsymbol{x}_{nt} = -\nabla f(\boldsymbol{x})^T\nabla^2 f(\boldsymbol{x})^{-1}\nabla f(\boldsymbol{x}) < 0$$

因此，Newton 步径是下降方向 (除非 $\boldsymbol{x}$ 是最优点)。将 $\boldsymbol{x}$ 加上 Newton 步径 $\Delta\boldsymbol{x}_{nt}$ 能够极小化 f 在 $\boldsymbol{x}$ 处的二阶近似。

注： 当 $f(x)$ 就是正定二次函数时，只需要一步便能求出最优解。

Newton 减量估计

通过简单计算，可以得到牛顿法的单步下降量约为 $f(\boldsymbol{x}) - \hat{f}(\boldsymbol{x} + \Delta\boldsymbol{x}_{nt})$，即：

$$f(\boldsymbol{x}) - \hat{f}(\boldsymbol{x} + \Delta\boldsymbol{x}_{nt}) \triangleq \frac{1}{2}\delta(\boldsymbol{x})^2,$$

其中 $\hat{f}$ 仍是 f 在 $\boldsymbol{x}$ 处的二阶近似。

注：这一量可作为牛顿法的终止判定准则。

牛顿法步骤

因此，可以得到**牛顿法**具体步骤如下：

Algorithm 1 牛顿法

1: 给定初始点 $\boldsymbol{x} \in \mathbf{dom}\, f$，误差阈值 $\epsilon > 0$

2: 计算 Newton 步径和减量。

$$\Delta \boldsymbol{x}_{nt} := -\nabla^2 f(\boldsymbol{x})^{-1} \nabla f(\boldsymbol{x}); \quad \delta^2 := \nabla f(\boldsymbol{x})^T \nabla^2 f(\boldsymbol{x})^{-1} \nabla f(\boldsymbol{x})$$

3: 停止准则。如果 $\delta^2/2 \leq \epsilon$，退出。

4: 直线搜索。通过回溯直线搜索确定步长 λ.

5: 改进。$\boldsymbol{x} := \boldsymbol{x} + \lambda \Delta \boldsymbol{x}_{nt}$

6: 重复上述步骤，直至退出。

经典牛顿法有很好的局部收敛性质. 实际上我们有如下定理:

定理 1

（经典牛顿法的收敛性）假设目标函数 f 是二阶连续可微的函数, 且海瑟矩阵在最优值点 $\boldsymbol{x}^*$ 的一个邻域 $N_\delta\left(\boldsymbol{x}^*\right)$ 内是利普希茨连续的, 即存在常数 $L>0$ 使得

$$\left\|\nabla^2 f(\boldsymbol{x})-\nabla^2 f(\boldsymbol{y})\right\| \leqslant L\|\boldsymbol{x}-\boldsymbol{y}\|, \quad \forall \boldsymbol{x}, \boldsymbol{y} \in N_\delta\left(\boldsymbol{x}^*\right)$$

如果函数 $f(\boldsymbol{x})$ 在点 $\boldsymbol{x}^*$ 处满足 $\nabla f\left(\boldsymbol{x}^*\right)=0, \nabla^2 f\left(\boldsymbol{x}^*\right) \succ 0$, 则对于步长为 1 时的迭代有如下结论:

（1） 如果初始点离 $\boldsymbol{x}^*$ 足够近, 则牛顿法产生的迭代点列 $\left\{\boldsymbol{x}^k\right\}$ 收敛到 $\boldsymbol{x}^*$;

（2） $\left\{\boldsymbol{x}^k\right\}$ 收敛到 $\boldsymbol{x}^*$ 的速度是 Q-二次的;

（3） $\left\{\left\|\nabla f\left(\boldsymbol{x}^k\right)\right\|\right\}$ Q-二次收敛到 0 .

从上述定理可以看出牛顿法收敛速度快，但牛顿法也有如下缺陷：

- 函数的 Hessian 矩阵本身计算代价大，难以存储。
- Hessian 矩阵还可能面临着不正定的问题，应该如何修正？
- 在高维问题中，求解 Hessian 矩阵的逆（或者是解大规模线性方程组）的计算量更大。
- 能否以较小的代价找到 Hessian 矩阵的一个较好的近似？

这就是接下来要介绍的修正牛顿法和拟牛顿法（变尺度法）。

34.1.2 修正牛顿法

前面也已提及经典牛顿法有如下缺陷:

- 每一步迭代需要求解一个 n 维线性方程组, 这导致在高维问题中计算量很大. 海瑟矩阵 $\nabla^2 f(\boldsymbol{x})$ 既不容易计算又不容易储存.
- 当 $\nabla^2 f(\boldsymbol{x})$ 不正定时, 由牛顿方程给出的牛顿步径 $\Delta \boldsymbol{x}_{nt}$ 的性质通常比较差. 例如可以验证当海瑟矩阵正定时, $\Delta \boldsymbol{x}_{nt}$ 是一个下降方向, 而在其他情况下 $\Delta \boldsymbol{x}_{nt}$ 不一定为下降方向.
- 当迭代点距最优值较远时, 直接选取步长 $\alpha = 1$ 会使得迭代极其不稳定, 在有些情况下迭代点列会发散.

为了克服这些缺陷，这里介绍带线搜索的修正牛顿法，其基本思想是对牛顿方程中的海瑟矩阵 $\nabla^2 f(\boldsymbol{x})$ 进行修正，使其变成正定矩阵. 它的一般框架如下所示：

Algorithm 2 修正牛顿法

1: 给定初始点 $\boldsymbol{x}^0$.
2: **for** $k = 0, 1, 2, \cdots$ **do**
3: 　确定矩阵 $\boldsymbol{E}^k$ 使得矩阵 $\boldsymbol{B}^k \overset{\text{def}}{=} \nabla^2 f(\boldsymbol{x}^k) + \boldsymbol{E}^k$ 正定且条件数较小.
4: 　求解修正的牛顿方程 $\boldsymbol{B}^k \boldsymbol{d}^k = -\nabla f(\boldsymbol{x}^k)$ 得方向 $\boldsymbol{d}^k$.
5: 　使用任意一种线搜索准则确定步长 α_k.
6: 　更新 $\boldsymbol{x}^{k+1} = \boldsymbol{x}^k + \alpha_k \boldsymbol{d}^k$.
7: **end for**

该算法的关键在于修正矩阵 $\boldsymbol{E}^k$ 如何选取.

显式修正

一个最直接的取法是取

$$\boldsymbol{E}^k = \tau_k \boldsymbol{I},$$

即取 $\boldsymbol{E}^k$ 为单位矩阵的常数倍.

- 根据矩阵理论可以知道, 当 τ_k 充分大时, 总可以保证 $\boldsymbol{B}^k$ 是正定矩阵.
- τ_k 不宜取得过大, 这是因为当 τ_k 趋于无穷时, $\boldsymbol{d}^k$ 的方向会接近负梯度方向.
- 比较合适的取法是先估计 $\nabla^2 f(\boldsymbol{x}^k)$ 的最小特征值, 再适当选择 τ_k.

隐式修正

另一种 $\boldsymbol{E}^k$ 的选取是隐式的:

- 它是通过修正 Cholesky 分解的方式来求解牛顿方程.
- 我们知道当海瑟矩阵正定时, 方程组可以用 Cholesky 分解快速求解.
- 当海瑟矩阵不定或条件数较大时, Cholesky 分解会失败. 而修正 Cholesky 分解算法对基本 Cholesky 分解算法进行修正, 且修正后的分解和原矩阵相差不大.
    - 根据 Cholesky 分解的形式, 如果 $\boldsymbol{A}$ 正定且条件数较小, 矩阵 $\boldsymbol{D}$ 的对角线元素不应该太小.
    - 如果计算过程中发现 d_j 过小就应该及时修正. 同时我们需要保证该修正是有界的,

我们首先回顾 Cholesky 分解的定义. 对任意对称正定矩阵 $\boldsymbol{A}=(a_{ij})$, 它的 Cholesky 分解可写作

$$\boldsymbol{A}=\boldsymbol{L}\boldsymbol{D}\boldsymbol{L}^{\mathrm{T}}$$

其中 $\boldsymbol{L}=(l_{ij})$ 是对角线元素均为 1 的下三角矩阵, $\boldsymbol{D}=\mathrm{Diag}\,(d_1,d_2,\cdots,d_n)$ 是对角矩阵且对角线元素均为正.

因此，对修正后的矩阵元素也需要有上界约束. 具体来说, 我们选取两个正参数 δ, β 使得

$$d_j \geqslant \delta, \quad l_{ij}\sqrt{d_j} \leqslant \beta, \quad i = j+1, j+2, \cdots, n.$$

因此，我们只需要修改 Cholesky 分解时 d_j 的更新即可保证上述条件成立. 具体更新方式为

$$d_j = \max\left\{|c_{jj}|, \left(\frac{\theta_j}{\beta}\right)^2, \delta\right\}, \quad \theta_j = \max_{i>j}|c_{ij}|.$$

可以证明, 修正的 Cholesky 分解算法实际上是计算修正矩阵 $\nabla^2 f(\boldsymbol{x}^k) + \boldsymbol{E}^k$ 的 Cholesky 分解, 其中 $\boldsymbol{E}^k$ 是对角矩阵且对角线元素非负. 当 $\nabla^2 f(\boldsymbol{x}^k)$ 正定且条件数足够小时有 $\boldsymbol{E}^k = 0$。

34.1.3 应用：牛顿法求解 Logistic 回归

在前面我们已经介绍了二分类的 Logistic 回归模型：

$$\min_{\boldsymbol{x}} L(\boldsymbol{x}) \overset{\text{def}}{=} \frac{1}{m}\sum_{i=1}^{m} \ln\left(1+\exp\left(-b_i \boldsymbol{a}_i^{\mathrm{T}} \boldsymbol{x}\right)\right) + \lambda\|\boldsymbol{x}\|_2^2.$$

接下来推导求解该问题的牛顿法，这转化为计算目标函数 $L(\boldsymbol{x})$ 的梯度和 Hessian 矩阵的问题。根据第六章介绍的向量值函数求导法，容易算出梯度为

$$\begin{aligned}
\nabla L(\boldsymbol{x}) &= \frac{1}{m}\sum_{i=1}^{m} \frac{1}{1+\exp\left(-b_i \boldsymbol{a}_i^{\mathrm{T}} \boldsymbol{x}\right)} \cdot \exp\left(-b_i \boldsymbol{a}_i^{\mathrm{T}} \boldsymbol{x}\right) \cdot \left(-b_i \boldsymbol{a}_i\right) + 2\lambda \boldsymbol{x} \\
&= -\frac{1}{m}\sum_{i=1}^{m} \left(1-p_i(\boldsymbol{x})\right) b_i \boldsymbol{a}_i + 2\lambda \boldsymbol{x}
\end{aligned}$$

其中 $p_i(\boldsymbol{x}) = \frac{1}{1+\exp\left(-b_i \boldsymbol{a}_i^{\mathrm{T}} \boldsymbol{x}\right)}$。

引入矩阵 $\boldsymbol{A}=[\boldsymbol{a}_1,\boldsymbol{a}_2,\cdots,\boldsymbol{a}_m]^{\mathrm{T}}\in\mathbb{R}^{m\times n}$，向量 $\boldsymbol{b}=(b_1,b_2,\cdots,b_m)^{\mathrm{T}}$，以及

$$\boldsymbol{p}(\boldsymbol{x})=(p_1(\boldsymbol{x}),p_2(\boldsymbol{x}),\cdots,p_m(\boldsymbol{x}))^{\mathrm{T}}.$$

此时梯度可简写为：

$$\nabla L(\boldsymbol{x})=-\frac{1}{m}\boldsymbol{A}^{\mathrm{T}}(\boldsymbol{b}-\boldsymbol{b}\odot\boldsymbol{p}(\boldsymbol{x}))+2\lambda\boldsymbol{x}.$$

再对梯度求导，并写成更为紧凑的矩阵形式，可得到 Hessian 矩阵

$$\nabla^2 L(\boldsymbol{x})=\frac{1}{m}\boldsymbol{A}^{\mathrm{T}}\boldsymbol{P}(\boldsymbol{x})\boldsymbol{A}+2\lambda\boldsymbol{I}$$

其中 $\boldsymbol{P}(\boldsymbol{x})$ 为由 $\{p_i(\boldsymbol{x})\,(1-p_i(\boldsymbol{x}))\}_{i=1}^m$ 生成的对角矩阵。因此，牛顿法可以写作

$$\boldsymbol{x}^{k+1}=\boldsymbol{x}^k+\left(\frac{1}{m}\left(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{P}\left(\boldsymbol{x}^k\right)\boldsymbol{A}+2\lambda\boldsymbol{I}\right)^{-1}\left(\frac{1}{m}\boldsymbol{A}^{\mathrm{T}}\left(\boldsymbol{b}-\boldsymbol{b}\odot\boldsymbol{p}\left(\boldsymbol{x}^k\right)\right)-2\lambda\boldsymbol{x}^k\right)\right).$$

在实际中，λ 经常取为 $\frac{1}{100m}$。另外，当变量规模不是很大时，可以利用正定矩阵的 Cholesky 分解来求解牛顿方程；当变量规模较大时，可以使用共轭梯度法进行不精确求解。这里采用 LIBSVM 网站的数据集，包括：a9a、ijcnn1 和 CINA 数据集。然后使用牛顿法进行求解，其求解结果参见图2。从中可以看出，在精确解附近梯度范数具有 Q- 超线性收敛性。

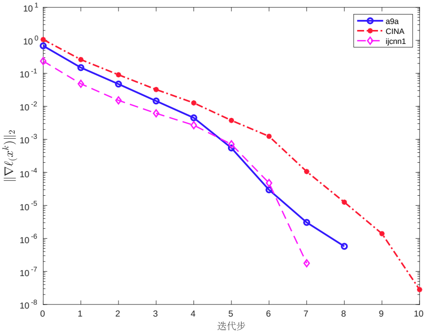

图 2: 牛顿法求解 Logistic 回归模型

- 拟牛顿法（变尺度法）是近 40 多年来发展起来的，它是求解无约束优化问题的一种有效方法。
- 由于它既避免了计算二阶导数矩阵及其求逆过程，又比梯度法的收敛速度快，特别是对高维问题具有显著的优越性，因而使拟牛顿法获得了很高的声誉，至今仍被公认为求解无约束优化问题最有效的算法之一。
- 下面我们就来简要地介绍拟牛顿法的基本原理及其计算过程。

34.2.1 割线方程

若想找到 Hessian 矩阵或它的逆的一个近似，就应当寻找到它需要满足的条件。以便构造近似矩阵，并令其也满足同样的条件即可。

显然，这样的一个近似应当满足 Taylor 展开。根据 Taylor 展开，梯度函数 $\nabla f(\boldsymbol{x})$ 在点 $x^{(k+1)}$ 处的近似为

$$\nabla f(\boldsymbol{x}) \approx \nabla f(\boldsymbol{x}^{(k+1)}) + \nabla^2 f(\boldsymbol{x}^{(k+1)})(\boldsymbol{x} - \boldsymbol{x}^{(k+1)})$$

$\boldsymbol{x} = \boldsymbol{x}^{(k)}$ 即有

$$\nabla f(\boldsymbol{x}^{(k+1)}) - \nabla f(\boldsymbol{x}^{(k)}) = \nabla^2 f(\boldsymbol{x}^{(k+1)})(\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^{(k)}) \tag{2}$$

或

$$\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^{(k)} = \nabla^2 f(\boldsymbol{x}^{(k+1)})^{-1}[\nabla f(\boldsymbol{x}^{(k+1)}) - \nabla f(\boldsymbol{x}^{(k)})] \tag{3}$$

这两个式子并称为**割线方程**，即拟牛顿条件。

割线方程

若令

$$\begin{cases} \Delta \boldsymbol{g}^{(k)} = \nabla f(\boldsymbol{x}^{(k+1)}) - \nabla f(\boldsymbol{x}^{(k)}) \\ \Delta \boldsymbol{x}^{(k)} = \boldsymbol{x}^{(k+1)} - \boldsymbol{x}^{(k)} \end{cases} \tag{4}$$

则式(2) 变为

$$\Delta \boldsymbol{g}^{(k)} = \nabla^2 f(\boldsymbol{x}^{(k+1)}) \Delta \boldsymbol{x}^{(k)}$$

式(3) 变为

$$\Delta \boldsymbol{x}^{(k)} = \nabla^2 f(\boldsymbol{x}^{(k+1)})^{-1} \Delta \boldsymbol{g}^{(k)}$$

34.2.2 拟牛顿法框架

如果得到满足割线方程的 **Hessian** 矩阵的近似（下面均用 $\boldsymbol{H}$ 表示），或者 **Hessian** 矩阵的逆的近似（下面均用 $\overline{\boldsymbol{H}}$ 表示），则可以得到拟牛顿方法的一般求解框架。

Algorithm 3 拟牛顿法计算框架

1: 给定 $\boldsymbol{x}^0 \in \mathbb{R}^n$, 初始矩阵 $\boldsymbol{H}^0 \in \mathbb{R}^{n\times n}$ (或 $\overline{\boldsymbol{H}}^0$), 令 $k=0$.
2: **while** $k=0,1,2,\cdots$ **do**
3: 　计算方向 $\boldsymbol{d}^k = -\left(\boldsymbol{H}^k\right)^{-1}\nabla f(\boldsymbol{x}^k)$ 或 $\boldsymbol{d}^k = -\overline{\boldsymbol{H}}^k\nabla f(\boldsymbol{x}^k)$.
4: 　通过线搜索找到合适的步长 $\lambda_k > 0$, 令 $\boldsymbol{x}^{k+1} = \boldsymbol{x}^k + \lambda_k \boldsymbol{d}^k$.
5: 　更新海瑟矩阵的近似矩阵 $\boldsymbol{H}^{k+1}$ 或其逆矩阵的近似 $\overline{\boldsymbol{H}}^{k+1}$.
6: 　$k \leftarrow k+1$.
7: **end while**

下面，我们将讨论如何借助割线方程（拟牛顿条件）具体的构造 Hessian 矩阵或其逆的近似。这里补充几点说明：

- 基于(2) 式得到 Hessian 矩阵的近似，具有较好的理论性质，迭代序列较为稳定，但仍然可能在大规模问题上是非常耗时的。
- 基于(3) 式得到 Hessian 矩阵的逆的近似，更加实用。
- 上述两种方式之间具有很好的形式对称性，下面仅基于 Hessian 矩阵逆的近似进行探究。

34.2.3 秩一更新

现设 $\overline{\boldsymbol{H}}^{(k)}$ 是第 k 步 Hessian 矩阵的逆的近似，现需构造出 $\overline{\boldsymbol{H}}^{(k+1)}$，则直观的想法是对它进行秩一修正。考虑到对称性，则可设

$$\overline{\boldsymbol{H}}^{(k+1)} = \overline{\boldsymbol{H}}^{(k)} + a\boldsymbol{u}\boldsymbol{u}^T \tag{5}$$

其中 $\boldsymbol{u} \in \mathbb{R}^n$，$a \in \mathbb{R}$ 待定。

秩一更新

利用割线方程，有

$$\begin{aligned}\Delta \boldsymbol{x}^{(k)} &= \overline{\boldsymbol{H}}^{(k+1)} \Delta \boldsymbol{g}^{(k)} \\ &= (\overline{\boldsymbol{H}}^{(k)} + a\boldsymbol{u}\boldsymbol{u}^T)\Delta \boldsymbol{g}^{(k)}\end{aligned} \tag{6}$$

整理得

$$a(\boldsymbol{u}^T \Delta \boldsymbol{g}^{(k)})\boldsymbol{u} = \Delta \boldsymbol{x}^{(k)} - \overline{\boldsymbol{H}}^{(k)} \Delta \boldsymbol{g}^{(k)}.$$

秩一更新

因此 $\boldsymbol{u}$ 与 $\Delta \boldsymbol{x}^{(k)} - \overline{\boldsymbol{H}}^{(k)} \Delta \boldsymbol{g}^{(k)}$ 共线。不妨令 $\boldsymbol{u} = \Delta \boldsymbol{x}^{(k)} - \overline{\boldsymbol{H}}^{(k)} \Delta \boldsymbol{g}^{(k)}$，则代入可得

$$a = \frac{1}{(\Delta \boldsymbol{x}^{(k)} - \overline{\boldsymbol{H}}^{(k)} \Delta \boldsymbol{g}^{(k)})^T \Delta \boldsymbol{g}^{(k)}}.$$

从而，得到**秩一更新**公式：

$$\overline{\boldsymbol{H}}^{(k+1)} = \overline{\boldsymbol{H}}^{(k)} + \frac{(\Delta \boldsymbol{x}^{(k)} - \overline{\boldsymbol{H}}^{(k)} \Delta \boldsymbol{g}^{(k)})(\Delta \boldsymbol{x}^{(k)} - \overline{\boldsymbol{H}}^{(k)} \Delta \boldsymbol{g}^{(k)})^T}{(\Delta \boldsymbol{x}^{(k)} - \overline{\boldsymbol{H}}^{(k)} \Delta \boldsymbol{g}^{(k)})^T \Delta \boldsymbol{g}^{(k)}}.$$

上述矩阵也称之为**尺度矩阵**。

秩一更新

如果考虑的是 Hessian 矩阵本身的近似，则同理可得对应的秩一更新公式如下：

$$\boldsymbol{H}^{(k+1)} = \boldsymbol{H}^{(k)} + \frac{(\Delta \boldsymbol{g}^{(k)} - \boldsymbol{H}^{(k)}\Delta \boldsymbol{x}^{(k)})(\Delta \boldsymbol{g}^{(k)} - \boldsymbol{H}^{(k)}\Delta \boldsymbol{x}^{(k)})^T}{(\Delta \boldsymbol{g}^{(k)} - \boldsymbol{H}^{(k)}\Delta \boldsymbol{x}^{(k)})^T \Delta \boldsymbol{x}^{(k)}}.$$

通过对比发现，实际上二者之间只是做了如下形式的替换：

$$\overline{\boldsymbol{H}} \to \boldsymbol{H}, \quad \Delta \boldsymbol{x}^{(k)} \leftrightarrow \Delta \boldsymbol{g}^{(k)}.$$

注： 这样的形式对称性对我们了解拟牛顿法大有裨益。

秩一更新

秩一更新存在的重大缺陷：

- 秩一更新虽然结构简单，易计算；
- 但不能保证在迭代过程中保持正定。
- 需要寻求更好的近似。

34.2.4 DFP

为克服秩一更新的缺陷，直观的改进是对它进行秩二修正。同样地考虑对称性，则可设

$$\overline{\boldsymbol{H}}^{(k+1)} = \overline{\boldsymbol{H}}^{(k)} + a\boldsymbol{u}\boldsymbol{u}^T + b\boldsymbol{v}\boldsymbol{v}^T \tag{7}$$

其中 $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^n$，$a, b \in \mathbb{R}$ 待定。

DFP

利用割线方程，有

$$\begin{aligned}\Delta \boldsymbol{x}^{(k)} &= \overline{\boldsymbol{H}}^{(k+1)} \Delta \boldsymbol{g}^{(k)} \\ &= (\overline{\boldsymbol{H}}^{(k)} + a\boldsymbol{u}\boldsymbol{u}^T + b\boldsymbol{v}\boldsymbol{v}^T)\Delta \boldsymbol{g}^{(k)}\end{aligned} \tag{8}$$

整理得

$$a(\boldsymbol{u}^T\Delta \boldsymbol{g}^{(k)})\boldsymbol{u} + b(\boldsymbol{v}^T\Delta \boldsymbol{g}^{(k)})\boldsymbol{v} = \Delta \boldsymbol{x}^{(k)} - \overline{\boldsymbol{H}}^{(k)}\Delta \boldsymbol{g}^{(k)}.$$

因此 $\boldsymbol{u}, \boldsymbol{v}$ 的线性组合等于 $\Delta \boldsymbol{x}^{(k)} - \overline{\boldsymbol{H}}^{(k)}\Delta \boldsymbol{g}^{(k)}$。同样地，不妨令 $\boldsymbol{u} = \Delta \boldsymbol{x}^{(k)}$，$\boldsymbol{v} = \overline{\boldsymbol{H}}^{(k)}\Delta \boldsymbol{g}^{(k)}$，则代入可得

DFP

$$a = \frac{1}{(\Delta \boldsymbol{x}^{(k)})^T \Delta \boldsymbol{g}^{(k)}},$$

$$b = -\frac{1}{(\Delta \boldsymbol{g}^{(k)})^T \overline{\boldsymbol{H}}^{(k)} \Delta \boldsymbol{g}^{(k)}}.$$

从而，得到更新公式：

$$\overline{\boldsymbol{H}}^{(k+1)} = \overline{\boldsymbol{H}}^{(k)} + \frac{\Delta \boldsymbol{x}^{(k)})\Delta \boldsymbol{x}^{(k)})^T}{(\Delta \boldsymbol{x}^{(k)})^T \Delta \boldsymbol{g}^{(k)}} - \frac{\overline{\boldsymbol{H}}^{(k)} \Delta \boldsymbol{g}^{(k)} (\overline{\boldsymbol{H}}^{(k)} \Delta \boldsymbol{g}^{(k)})^T}{(\Delta \boldsymbol{g}^{(k)})^T \overline{\boldsymbol{H}}^{(k)} \Delta \boldsymbol{g}^{(k)}}. \tag{9}$$

这种迭代公式由 Davidon 发现，并由 Flecher 以及 Powell 进一步发展。因此被称为 **DFP 公式**。

34.2.5 BFGS

利用前面提及的形式对称性，可得如下更新公式：

$$\boldsymbol{H}^{(k+1)} = \boldsymbol{H}^{(k)} + \frac{\Delta \boldsymbol{g}^{(k)})\Delta \boldsymbol{g}^{(k)})^T}{(\Delta \boldsymbol{g}^{(k)})^T \Delta \boldsymbol{x}^{(k)}} - \frac{\boldsymbol{H}^{(k)} \Delta \boldsymbol{x}^{(k)} (\boldsymbol{H}^{(k)} \Delta \boldsymbol{x}^{(k)})^T}{(\Delta \boldsymbol{x}^{(k)})^T \boldsymbol{H}^{(k)} \Delta \boldsymbol{x}^{(k)}}.$$

这种迭代格式就是著名的 **BFGS 公式**，它是由 Broyden, Fletcher, Goldfarb 和 Shanno 四人的名字首字母组成。

尽管 DFP 格式和 BFGS 格式存在这种对偶关系，但实际上，BFGS 格式效果更好些。因此，在实际中 BFGS 格式被使用的更多。

34.2.6 拟牛顿法计算步骤

综上，可将拟牛顿法（以 DFP 为例）的计算步骤总结如下：

1. 给定初始点 $\boldsymbol{x}^{(0)}$ 及梯度允许误差 $\varepsilon > 0$；
2. 若
$$\|\nabla f(\boldsymbol{x}^{(0)})\|^2 \neq \varepsilon$$
则 $\boldsymbol{x}^{(0)}$ 即为近似极小点，停止迭代。否则，转向下一步；

拟牛顿法计算步骤

3. 令

$$\overline{\boldsymbol{H}}^{(0)} = \boldsymbol{I}(\text{单位阵})$$

$$\boldsymbol{p}^{(0)} = -\overline{\boldsymbol{H}}^{(0)} \nabla f(\boldsymbol{x}^{(0)})$$

在 $\boldsymbol{p}^{(0)}$ 方向进行一维搜索，确定最佳步长 λ_0

$$\min_{\lambda} f(\boldsymbol{x}^{(0)} + \lambda \boldsymbol{p}^{(0)}) = f(\boldsymbol{x}^{(0)} + \lambda_0 \boldsymbol{p}^{(0)})$$

如此可得下一个近似点

$$\boldsymbol{x}^{(1)} = \boldsymbol{x}^{(0)} + \lambda_0 \boldsymbol{p}^{(0)}$$

拟牛顿法计算步骤

4. 一般地，设已得到近似点 $\boldsymbol{x}^{(k)}$，算出 $\nabla f(\boldsymbol{x}^{(k)})$，若

$$\|\nabla f(\boldsymbol{x}^{(k)})\|^2 \leq \varepsilon$$

则 $\boldsymbol{x}^{(k)}$ 即为所求的近似解，停止迭代；否则，按式(9) 计算 $\overline{\boldsymbol{H}}^{(k)}$，并令

$$\boldsymbol{p}^{(k)} = -\overline{\boldsymbol{H}}^{(k)}\nabla f(\boldsymbol{x}^{(k)})$$

在 $\boldsymbol{p}^{(k)}$ 方向进行一维搜索，确定最佳最长 λ_k

$$\min_{\lambda} f(\boldsymbol{x}^{(k)} + \lambda \boldsymbol{p}^{(k)}) = f(\boldsymbol{x}^{(k)} + \lambda_k \boldsymbol{p}^{(k)})$$

其下一个近似点为

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + \lambda_k \boldsymbol{p}^{(k)}$$

拟牛顿法计算步骤

5. 若 $\boldsymbol{x}^{(k+1)}$ 点满足精度要求，则 $\boldsymbol{x}^{(k+1)}$ 即为所求的近似解。否则，转回第 (4) 步，直到求出某点满足精度要求为止。

注：与共轭梯度法相类似，如果迭代 n 次仍不收敛，则以 $\boldsymbol{x}^{(n)}$ 为新的 $\boldsymbol{x}^{(0)}$，以这时的 $\boldsymbol{x}^{(0)}$ 为起点重新开始一轮新的迭代。

对于拟牛顿法，我们也可以得到其基本的收敛性以及收敛速度.

定理 2

（*BFGS* 全局收敛性）假设初始矩阵 $\boldsymbol{H}^0$ 是对称正定矩阵，目标函数 $f(\boldsymbol{x})$ 是二阶连续可微函数，且下水平集

$$\mathcal{L} = \left\{\boldsymbol{x} \in \mathbb{R}^n \mid f(\boldsymbol{x}) \leqslant f\left(\boldsymbol{x}^0\right)\right\}$$

是凸的，并且存在正数 m 以及 M 使得对于任意的 $\boldsymbol{z} \in \mathbb{R}^n$ 以及任意的 $\boldsymbol{x} \in \mathcal{L}$ 有

$$m\|\boldsymbol{z}\|^2 \leqslant \boldsymbol{z}^{\mathrm{T}} \nabla^2 f(\boldsymbol{x}) \boldsymbol{z} \leqslant M\|\boldsymbol{z}\|^2,$$

则采用 *BFGS* 格式并结合 *Wolfe* 线搜索的拟牛顿算法全局收敛到 $f(\boldsymbol{x})$ 的极小值点 $\boldsymbol{x}^*$.

上述定理叙述了 BFGS 格式的全局收敛性，但没有说明以什么速度收敛．下面这个定理介绍了在一定条件下 BFGS 格式会达到 Q-超线性收敛速度．这里仍然只给出定理结果，感兴趣的读者可以查阅相关文献，了解详细的证明过程。

定理 3

（*BFGS* 收敛速度）设 $f(\boldsymbol{x})$ 二阶连续可微，在最优点 $\boldsymbol{x}^*$ 的一个邻域内海瑟矩阵利普希茨连续，且使用 *BFGS* 迭代格式收敛到 f 的最优值点 $\boldsymbol{x}^*$．若迭代点列 $\{\boldsymbol{x}^k\}$ 满足

$$\sum_{k=1}^{\infty} \left\| \boldsymbol{x}^k - \boldsymbol{x}^* \right\| < +\infty,$$

则 $\{\boldsymbol{x}^k\}$ 以 Q-超线性收敛到 $\boldsymbol{x}^*$.

例 1

试用 *DFP* 法重新计算下述二次函数的极小点

$$f(\boldsymbol{x}) = \frac{3}{2}x_1^2 + \frac{1}{2}x_2^2 - x_1x_2 - 2x_1$$

解

我们从 $\boldsymbol{x}^{(0)} = (-2, 4)^T$ 开始，并取

$$\overline{\boldsymbol{H}}^{(0)} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

$$\nabla f(\boldsymbol{x}) = [(3x_1 - x_2 - 2), (x_2 - x_1)]^T$$

$$\nabla f(\boldsymbol{x}^{(0)}) = (-12, 6)^T$$

$$\boldsymbol{p}^{(0)} = -\overline{\boldsymbol{H}}^{(0)}\nabla f(\boldsymbol{x}^{(0)}) = -\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} -12 \\ 6 \end{pmatrix} = \begin{pmatrix} 12 \\ -6 \end{pmatrix}$$

利用一维搜索，即 $\min_\lambda f(\boldsymbol{x}^{(0)} + \lambda \boldsymbol{p}^{(0)})$，可算得

$$\lambda_0 = \frac{5}{17}$$

$$\boldsymbol{x}^{(1)} = \boldsymbol{x}^{(0)} + \lambda_0 \boldsymbol{p}^{(0)} = \begin{pmatrix} -2 \\ 4 \end{pmatrix} + \frac{5}{17}\begin{pmatrix} 12 \\ -6 \end{pmatrix} = \left(\frac{26}{17}, \frac{38}{17}\right)^T$$

$$\nabla f(\boldsymbol{x}^{(1)}) = \left(\frac{6}{17}, \frac{12}{17}\right)^T$$

$$\Delta \boldsymbol{x}^{(0)} = \boldsymbol{x}^{(1)} - \boldsymbol{x}^{(0)} = \left(\frac{26}{17}, \frac{38}{17}\right)^T - (-2, 4)^T = \left(\frac{60}{17}, -\frac{30}{17}\right)^T$$

$$\Delta \boldsymbol{g}^{(0)} = \nabla f(\boldsymbol{x}^{(1)}) - \nabla f(\boldsymbol{x}^{(0)}) = \left(\frac{6}{17}, \frac{12}{17}\right)^T - (-12, 6)^T = \left(\frac{210}{17}, -\frac{90}{17}\right)^T$$

$$\overline{\boldsymbol{H}}^{(1)} = \overline{\boldsymbol{H}}^{(0)} + \frac{\Delta \boldsymbol{x}^{(0)}(\Delta \boldsymbol{x}^{(0)})^T}{(\Delta \boldsymbol{g}^{(0)})^T \Delta \boldsymbol{x}^{(0)}} - \frac{\overline{\boldsymbol{H}}^{(0)} \Delta \boldsymbol{g}^{(0)} (\Delta \boldsymbol{g}^{(0)})^T \overline{\boldsymbol{H}}^{(0)}}{(\Delta \boldsymbol{g}^{(0)})^T \overline{\boldsymbol{H}}^{(0)} \Delta \boldsymbol{g}^{(0)}}$$

$$= \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + \frac{\left(\frac{60}{17}, -\frac{30}{17}\right)^T \left(\frac{60}{17}, -\frac{30}{17}\right)}{\left(\frac{210}{17}, -\frac{90}{17}\right)\left(\frac{60}{17}, -\frac{30}{17}\right)^T} - \frac{\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}\left(\frac{210}{17}, -\frac{90}{17}\right)^T \left(\frac{210}{17}, -\frac{90}{17}\right)\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}}{\left(\frac{210}{17}, -\frac{90}{17}\right)\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}\left(\frac{210}{17}, -\frac{90}{17}\right)^T}$$

$$= \frac{1}{986}\begin{pmatrix} 385 & 241 \\ 241 & 891 \end{pmatrix}$$

$$p^{(1)} = -\overline{H}^{(1)}\nabla f(x^{(1)}) = -\frac{1}{986}\begin{pmatrix} 385 & 241 \\ 241 & 891 \end{pmatrix}\begin{pmatrix} \frac{6}{17} \\ \frac{12}{17} \end{pmatrix} = -\begin{pmatrix} \frac{9}{29} \\ \frac{21}{29} \end{pmatrix}$$

再由一维搜索 $\min_\lambda f(x^{(1)} + \lambda p^{(1)})$，得

$$\lambda_1 = \frac{29}{17}$$

从而

$$x^{(2)} = x^{(1)} + \lambda_1 p^{(1)} = \begin{pmatrix} \frac{26}{17} \\ \frac{38}{17} \end{pmatrix} + \frac{29}{17}\begin{pmatrix} -\frac{9}{29} \\ -\frac{21}{29} \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

$$\nabla f(x^{(2)}) = (0, 0)^T$$

可知 $x^{(2)} = (1, 1)^T$ 为极小点。

- 在以上讨论中，若取第一个尺度矩阵 $\overline{\boldsymbol{H}}^{(0)}$ 为对称正定阵，则可以证明，由式(9) 迭代形成的尺度矩阵均为对称正定阵。
- 由此可知其搜索方向 $\boldsymbol{p}^{(k)} = -\overline{\boldsymbol{H}}^{(k)}\nabla f(\boldsymbol{x}^{(k)})$ 为下降方向，这就可以保证每次迭代均能使目标函数值有所改善。
- 当把 DFP 拟牛顿法用于正定二次函数时，产生的搜索方向为共轭方向，因而也具有有限步收敛的性质。若将初始尺度矩阵也取为单位矩阵，对这种函数来说，DFP 法就与共轭梯度法一样了。

34.2.7 应用：拟牛顿法求解压缩感知问题

考虑压缩感知问题:

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} \|\boldsymbol{x}\|_1, \quad \text{s.t.} \quad \boldsymbol{A}\boldsymbol{x}=\boldsymbol{b}, \tag{10}$$

其中 $\boldsymbol{A}\in\mathbb{R}^{m\times n}, \boldsymbol{b}\in\mathbb{R}^m$ 为给定的矩阵和向量. 这是一个约束优化问题, 如何将其转化为一个无约束优化问题呢? 自然地, 我们可以考虑其对偶问题. 由于问题 (10) 的对偶问题的无约束优化形式不是可微的, 即无法计算梯度 (读者可以自行验证)。

我们考虑如下正则化问题:

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} \|\boldsymbol{x}\|_1 + \frac{1}{2\alpha}\|\boldsymbol{x}\|_2^2, \quad \text{s.t.} \quad \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}, \tag{11}$$

这里 $\alpha > 0$ 为正则化参数. 显然, 当 α 趋于无穷大时, 问题 (11) 的解会逼近 (10) 的解. 由于问题 (11) 的目标函数是强凸的, 其对偶问题的无约束优化形式的目标函数是可微的. 具体地, 问题 (11) 的对偶问题为

$$\min_{\boldsymbol{y}\in\mathbb{R}^m} f(\boldsymbol{y}) = -\boldsymbol{b}^{\mathrm{T}}\boldsymbol{y} + \frac{\alpha}{2}\left\|\boldsymbol{A}^{\mathrm{T}}\boldsymbol{y} - \mathcal{P}_{[-1,1]^n}\left(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{y}\right)\right\|_2^2, \tag{12}$$

其中 $\mathcal{P}_{[-1,1]^n}(\boldsymbol{x})$ 为 $\boldsymbol{x}$ 到集合 $[-1,1]^n$ 的投影.

通过简单计算, 可知

$$\nabla f(\boldsymbol{y}) = -\boldsymbol{b} + \alpha \boldsymbol{A} \left(\boldsymbol{A}^{\mathrm{T}} \boldsymbol{y} - \mathcal{P}_{[-1,1]^n} \left(\boldsymbol{A}^{\mathrm{T}} \boldsymbol{y} \right) \right)$$

那么, 我们可以利用 L-BFGS 方法来求解问题 (12). 在得到该问题的解 $\boldsymbol{y}^*$ 之后, 问题 (11) 的解 $\boldsymbol{x}^*$ 可通过下式近似得到:

$$\boldsymbol{x}^* \approx \alpha \left(\boldsymbol{A}^{\mathrm{T}} \boldsymbol{y}^* - \mathcal{P}_{[-1,1]^n} \left(\boldsymbol{A}^{\mathrm{T}} \boldsymbol{y}^* \right) \right)$$

进一步地, 当 α 充分大时, 问题 (11) 的解就是原问题 (10) 的解. 因此, 我们可以通过选取合适的 α, 通过求解问题 (12) 来得到问题 (10) 的解.

我们用 LASSO 问题中的 $\boldsymbol{A}$ 和 $\boldsymbol{b}$, 分别选取 $\alpha = 5, 10$, 调用 BFGS 方法[1]求解问题 (12), 其中内存长度取为 5. 迭代收敛过程见图 3. 从图中我们可以看到, 当靠近最优解时, BFGS 方法的迭代点列呈 Q-线性收敛.

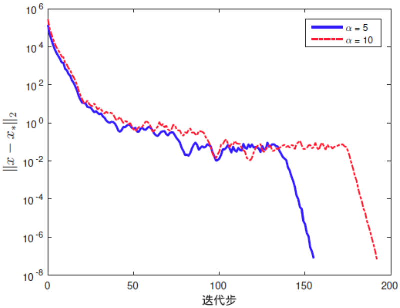

图 3: 压缩感知问题

[1]实际中，使用的是有限内存 BFGS，感兴趣的同学可以查阅相关文献。

本讲小结

牛顿法

- 二阶近似
- 牛顿步径
- 牛顿法、修正的牛顿法

拟牛顿法（变尺度法）

- 割线方程
- 秩一更新
- DFP、BFGS

注意，牛顿法中还有非精确牛顿法以及拟牛顿法中还有有限内存 BFGS 方法等改进方法，本课程没有详细介绍。另外，本节我们主要讨论了无约束优化问题的二阶求解算法。那么我们是如何处理约束优化问题的呢？