第 5 章

牛顿法和BFGS方法

梯度类方法和邻近梯度法只依赖函数和梯度信息来设计，其对所求解的最优化问题的性质要求并不高，很多机器学习应用问题对应的最优化问题可以使用梯度类方法和邻近梯度法来求解。然而在很多机器学习应用中，建模得到的最优化问题的目标函数也可能是充分光滑的，即可以计算其二阶导数（Hessian 矩阵），因此可以利用二阶导数等信息构造下降方向。牛顿法是最典型的二阶方法，其每步迭代的计算复杂度较高，但是收敛速度比一阶算法显著快。本章中，以牛顿法为基础介绍一种拟牛顿方法，即 BFGS 方法，其通过一阶信息近似估计二阶导数，旨在接近牛顿法在收敛速度方面的能力。进一步地，有限内存 BFGS 方法被提出，而有限内存 BFGS 方法更加适合处理机器学习应用问题。

5.1 牛顿法

牛顿法在第3章中已有简单介绍，其利用最优化问题的二阶梯度信息来提升传统梯度下降方法的优化效率。针对无约束最优化问题，即

$$\min_{\boldsymbol{x}} \ \left\{ f(\boldsymbol{x}) \,\middle|\, \boldsymbol{x} \in \mathbb{R}^n \right\}$$

其中，目标函数 f 为二次连续可微函数（Hessian 矩阵记为 $\nabla^2 f(\boldsymbol{x})$）。值得注意的是，该最优化问题的一阶最优性条件为

$$\nabla f(\boldsymbol{x}) = 0$$

如果对 $\nabla f(\boldsymbol{x})$ 函数在 $\boldsymbol{y}$ 处进行泰勒展开，则有

$$\nabla f(\boldsymbol{x}) \approx \nabla f(\boldsymbol{y}) + \nabla^2 f(\boldsymbol{y})\,(\boldsymbol{x} - \boldsymbol{y})$$

如果令 $\boldsymbol{y} = \boldsymbol{x}^k$，且希望找到近似最优解 $\boldsymbol{x}^{k+1}$，满足

$$\nabla f(\boldsymbol{x}^{k+1}) \approx \nabla f(\boldsymbol{x}^k) + \nabla^2 f(\boldsymbol{x}^k)\left(\boldsymbol{x}^{k+1} - \boldsymbol{x}^k\right) = 0$$

即

$$\boldsymbol{x}^{k+1} = \boldsymbol{x}^k - \left[\nabla^2 f(\boldsymbol{x}^k)\right]^{-1} \nabla f(\boldsymbol{x}^k) \tag{5.1}$$

则得到了牛顿法的迭代格式。牛顿法的关键核心是二阶导数Hessian矩阵的逆矩阵的计算，其较高计算难度导致牛顿法每步迭代的计算复杂度高，然而同样因为二阶导数的使用带来了牛顿法快速收敛的优势。以牛顿法为基础，进一步设计拟牛顿方法，其中的典型BFGS方法在5.2节中介绍。

5.2 BFGS方法

这个重要方法是由Broyden、Fletcher、Goldfard和Shanno于1970年提出的一种拟牛顿方法[13–16]，简称为BFGS。BFGS针对无约束最优化问题，其中目标函数$f(\boldsymbol{x})$为二次连续可微函数，该目标函数在迭代点$\boldsymbol{x}^k$的二阶近似记为

$$m_k(\boldsymbol{p}) := f(\boldsymbol{x}^k) + \nabla f(\boldsymbol{x}^k)^{\mathrm{T}}\boldsymbol{p} + \frac{1}{2}\boldsymbol{p}^{\mathrm{T}}\boldsymbol{\mathcal{B}}_k\boldsymbol{p} \tag{5.2}$$

其中，$\boldsymbol{\mathcal{B}}_k \in \mathbb{R}^{n\times n}$表示一个对称正定矩阵, 且其会随着迭代进行更新。由式(5.2)可以看出

$$m_k(\boldsymbol{0}) = f(\boldsymbol{x}^k), \quad \nabla m_k(\boldsymbol{0}) = \nabla f(\boldsymbol{x}^k)$$

因此在$\boldsymbol{x}^k$的附近可以用m_k来近似目标函数$f(\cdot)$。同时对这个二次凸函数$m_k(\boldsymbol{p})$求极小，可以得到

$$\boldsymbol{p}_k = -\boldsymbol{\mathcal{B}}_k^{-1}\nabla f(\boldsymbol{x}^k) \tag{5.3}$$

此时，$\boldsymbol{p}_k$可以认为是拟牛顿方法迭代格式中的搜索方向, 于是拟牛顿方法的迭代格式可以记为

$$\boldsymbol{x}^{k+1} = \boldsymbol{x}^k + \alpha_k\boldsymbol{p}_k \tag{5.4}$$

其中，$\alpha_k > 0$表示步长。明显地，拟牛顿方法的迭代格式和牛顿法十分类似，关键区别在于拟牛顿方法在二阶近似时并没有使用Hessian矩阵, 而是使用一个对称正定的矩阵$\boldsymbol{\mathcal{B}}_k$作为Hessian矩阵的近似。如何在迭代过程中更新$\boldsymbol{\mathcal{B}}_k$自然成为需要解决的核心问题。若每次迭代都从零开始计算矩阵$\boldsymbol{\mathcal{B}}_k$，则需要很高的计算代价。为了减少计算量，可以在前一步$\boldsymbol{\mathcal{B}}_k$的基础上通过“曲率条件”来计算新的$\boldsymbol{\mathcal{B}}_{k+1}$。

下面重点介绍曲率条件。考虑在$\boldsymbol{x}^{k+1}$处的二阶近似模型:

$$m_{k+1}(\boldsymbol{p}) = f(\boldsymbol{x}^{k+1}) + \nabla f(\boldsymbol{x}^{k+1})^{\mathrm{T}}\boldsymbol{p} + \frac{1}{2}\boldsymbol{p}^{\mathrm{T}}\boldsymbol{\mathcal{B}}_{k+1}\boldsymbol{p} \tag{5.5}$$

m_{k+1}在$\boldsymbol{x}^k - \boldsymbol{x}^{k+1}$处函数值近似函数$f$在$\boldsymbol{x}^k$处函数值。因此主要考虑函数$f$在$\boldsymbol{x}^k$的梯度和$m_{k+1}$在$\boldsymbol{x}^k - \boldsymbol{x}^{k+1}$处的梯度，若两者梯度相等则有

$$\nabla m_{k+1}(\boldsymbol{x}^k - \boldsymbol{x}^{k+1}) = \nabla f(\boldsymbol{x}^{k+1}) + \boldsymbol{\mathcal{B}}_{k+1}(\boldsymbol{x}^k - \boldsymbol{x}^{k+1}) = \nabla f(\boldsymbol{x}^k)$$

该式等价于

$$\boldsymbol{\mathcal{B}}_{k+1}\left(\boldsymbol{x}^{k+1} - \boldsymbol{x}^k\right) = \nabla f(\boldsymbol{x}^{k+1}) - \nabla f(\boldsymbol{x}^k)$$

进一步设定 $\boldsymbol{s}_k = \boldsymbol{x}^{k+1} - \boldsymbol{x}^k$，$\boldsymbol{y}_k = \nabla f(\boldsymbol{x}^{k+1}) - \nabla f(\boldsymbol{x}^k)$，可以得到

$$\boldsymbol{\mathcal{B}}_{k+1}\boldsymbol{s}_k = \boldsymbol{y}_k$$

上式被称为割线方程（Secant equation）。在每次迭代过程中，需要保证矩阵 $\boldsymbol{\mathcal{B}}_k$ 的正定性。所以，在上式两边同时左乘 $\boldsymbol{s}_k^{\mathrm{T}}$ 可以得到

$$\boldsymbol{s}_k^{\mathrm{T}}\boldsymbol{\mathcal{B}}_{k+1}\boldsymbol{s}_k = \boldsymbol{s}_k^{\mathrm{T}}\boldsymbol{y}_k$$

因此如下条件

$$\boldsymbol{s}_k^{\mathrm{T}}\boldsymbol{y}_k > \boldsymbol{0} \tag{5.6}$$

是保证矩阵 $\boldsymbol{\mathcal{B}}_{k+1}$ 正定的一个必要条件，也被称为“曲率条件”。如果函数 f 是强凸函数，不等式(5.6)对任意两点 $\boldsymbol{x}^k$ 和 $\boldsymbol{x}^{k+1}$ 都是满足的。但是对于非强凸函数，需要通过线搜索技术满足 Wolfe 条件或者强 Wolfe 条件，以此来保证曲率条件(5.6)成立。

割线方程的解不一定是唯一的，可以进一步添加一些限制条件来对解进行约束，比如要求 $\boldsymbol{\mathcal{B}}_{k+1}$ 与 $\boldsymbol{\mathcal{B}}_k$ 的距离是最近的，即

$$\begin{aligned} \min_{\boldsymbol{\mathcal{B}}} \quad & \|\boldsymbol{\mathcal{B}} - \boldsymbol{\mathcal{B}}_k\|_{\boldsymbol{W}} \\ \text{s.t.} \quad & \boldsymbol{\mathcal{B}} = \boldsymbol{\mathcal{B}}^{\mathrm{T}}, \quad \boldsymbol{\mathcal{B}}s_k = \boldsymbol{y}_k \end{aligned} \tag{5.7}$$

其中，$\boldsymbol{s}_k$ 和 $\boldsymbol{y}_k$ 满足曲率条件(5.6)，且 $\boldsymbol{\mathcal{B}}_k$ 是对称正定的。在优化问题(5.7)中，采用加权 Frobenius 范数来定义, 即

$$\|\boldsymbol{\mathcal{B}} - \boldsymbol{\mathcal{B}}_k\|_{\boldsymbol{W}} \equiv \left\|\boldsymbol{W}^{1/2}\left(\boldsymbol{\mathcal{B}} - \boldsymbol{\mathcal{B}}_k\right)\boldsymbol{W}^{1/2}\right\|_{\mathrm{F}} \tag{5.8}$$

其中，Frobenius 范数的定义见附录。若权重矩阵 $\boldsymbol{W}$ 满足 $\boldsymbol{W}\boldsymbol{y}_k = \boldsymbol{s}_k$，使用矩阵范数 $\|\cdot\|_{\boldsymbol{W}}$，则优化问题(5.7)有唯一解，且该解为

$$\boldsymbol{\mathcal{B}}_{k+1} = \left(\boldsymbol{\mathcal{I}} - \gamma_k\boldsymbol{y}_k\boldsymbol{s}_k^{\mathrm{T}}\right)\boldsymbol{\mathcal{B}}_k\left(\boldsymbol{\mathcal{I}} - \gamma_k\boldsymbol{s}_k\boldsymbol{y}_k^{\mathrm{T}}\right) + \gamma_k\boldsymbol{y}_k\boldsymbol{y}_k^{\mathrm{T}} \tag{5.9}$$

其中，

$$\gamma_k = \frac{1}{\boldsymbol{y}_k^{\mathrm{T}}\boldsymbol{s}_k}$$

式(5.9)最初由 Davidon 于 1959 年提出，不久被 Fletcher 和 Powell 推广，因此称为 DFP 公式。如果记 $\boldsymbol{\mathcal{B}}_k^{-1}$ 为 $\boldsymbol{\mathcal{H}}_k$，即 $\boldsymbol{\mathcal{H}}_k = \boldsymbol{\mathcal{B}}_k^{-1}$，进一步使用 Sherman-Morrison-Woodbury 公式，可以得到 Hessian 矩阵的逆矩阵的近似矩阵 $\boldsymbol{H}_k$ 的更新公式，即

$$\boldsymbol{\mathcal{H}}_{k+1} = \boldsymbol{\mathcal{H}}_k - \frac{\boldsymbol{\mathcal{H}}_k\boldsymbol{y}_k\boldsymbol{y}_k^{\mathrm{T}}\boldsymbol{\mathcal{H}}_k}{\boldsymbol{y}_k^{\mathrm{T}}\boldsymbol{\mathcal{H}}_k\boldsymbol{y}_k} + \frac{\boldsymbol{s}_k\boldsymbol{s}_k^{\mathrm{T}}}{\boldsymbol{y}_k^{\mathrm{T}}\boldsymbol{s}_k} \tag{5.10}$$

注意，式(5.10)右边的最后两项都是秩为 1 的矩阵，所以从 $\boldsymbol{\mathcal{H}}_k$ 到 $\boldsymbol{\mathcal{H}}_{k+1}$ 进行了一次秩 2 更新。类似地, 式(5.9)也是对 $\boldsymbol{\mathcal{B}}_k$ 做秩 2 更新。这就是拟牛顿方法更新近似矩阵的基本思想：每次迭代更新时，近似矩阵是结合前几步迭代点的目标函数信息对当前近似矩阵 $\boldsymbol{\mathcal{B}}_k$ 或 $\boldsymbol{\mathcal{H}}_k$ 做简单修正。

DFP 公式是一个有效的最优化方法，进一步如果对割线方程做一个小小的变形，即

$$\boldsymbol{\mathcal{H}}_{k+1}\boldsymbol{y}_k = \boldsymbol{s}_k$$

类似地，对 Hessian 矩阵的逆矩阵的近似矩阵 $\boldsymbol{\mathcal{H}}_k$ 提出相应的要求，也可以得到 BFGS 公式，其被认为是所有拟牛顿方法中最有效的方法。在 BFGS 方法中，$\boldsymbol{\mathcal{H}}_{k+1}$ 是下面最优化问题的最优解，即

$$\begin{aligned} \min_{\boldsymbol{\mathcal{H}}} \quad & \|\boldsymbol{\mathcal{H}} - \boldsymbol{\mathcal{H}}_k\|_{\boldsymbol{W}} \\ \text{s.t.} \quad & \boldsymbol{\mathcal{H}} = \boldsymbol{\mathcal{H}}^{\mathrm{T}},\ \boldsymbol{\mathcal{H}}\boldsymbol{y}_k = \boldsymbol{s}_k \end{aligned} \tag{5.11}$$

这里的范数同样采用带权重的 Frobenius 范数，权重矩阵 $\boldsymbol{W}$ 满足 $\boldsymbol{W}\boldsymbol{s}_k = \boldsymbol{y}_k$。通过计算可得到最优化问题(5.11)的唯一解是

$$\boldsymbol{\mathcal{H}}_{k+1} = \left(\boldsymbol{\mathcal{I}} - \boldsymbol{\rho}_k \boldsymbol{s}_k \boldsymbol{y}_k^{\mathrm{T}}\right) \boldsymbol{\mathcal{H}}_k \left(\boldsymbol{\mathcal{I}} - \boldsymbol{\rho}_k \boldsymbol{y}_k \boldsymbol{s}_k^{\mathrm{T}}\right) + \boldsymbol{\rho}_k \boldsymbol{s}_k \boldsymbol{s}_k^{\mathrm{T}} \tag{5.12}$$

其中，

$$\boldsymbol{\rho}_k = \frac{1}{\boldsymbol{y}_k^{\mathrm{T}} \boldsymbol{s}_k}$$

同样地, 可以推导出 BFGS 关于 Hessian 矩阵的近似矩阵 $\boldsymbol{\mathcal{B}}_k$ 的更新公式，对式(5.12)使用 Sherman-Morrison-Woodbury 公式就可以得到

$$\boldsymbol{\mathcal{B}}_{k+1} = \boldsymbol{\mathcal{B}}_k - \frac{\boldsymbol{\mathcal{B}}_k \boldsymbol{s}_k \boldsymbol{s}_k^{\mathrm{T}} \boldsymbol{\mathcal{B}}_k}{\boldsymbol{s}_k^{\mathrm{T}} \boldsymbol{\mathcal{B}}_k \boldsymbol{s}_k} + \frac{\boldsymbol{y}_k \boldsymbol{y}_k^{\mathrm{T}}}{\boldsymbol{y}_k^{\mathrm{T}} \boldsymbol{s}_k} \tag{5.13}$$

总结来看，可以得到如下的 BFGS 方法基本框架。

BFGS 方法：

- 给定初始点 $\boldsymbol{x}^0$，误差容忍度 ϵ，初始点处的 Hessian 矩阵的矩阵逆的近似 $\boldsymbol{\mathcal{H}}_0$，$k \leftarrow 0$；
- 如果 $\|\nabla f(\boldsymbol{x}^k)\| > \epsilon$，则进行下一步；
- 计算搜索方向

$$\boldsymbol{p}_k = -\boldsymbol{\mathcal{H}}_k \nabla f(\boldsymbol{x}^k) \tag{5.14}$$

- 更新迭代点: $\boldsymbol{x}^{k+1} = \boldsymbol{x}^k + \alpha_k \boldsymbol{p}_k$，其中，步长 α_k 通过 Wolfe 线搜索准则得到；
- 计算 $\boldsymbol{s}_k = \boldsymbol{x}^{k+1} - \boldsymbol{x}^k$ 和 $\boldsymbol{y}_k = \nabla f(\boldsymbol{x}^{k+1}) - \nabla f(\boldsymbol{x}^k)$；
- 由式(5.12)计算 $\boldsymbol{\mathcal{H}}_{k+1}$；
- $k \leftarrow k+1$。

BFGS 收敛性质：假设初始矩阵 $\boldsymbol{\mathcal{B}}_0$ 是对称正定矩阵，目标函数 $f(\boldsymbol{x})$ 是二阶连续可微的，且下水平集

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

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

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

结合Wolfe线搜索的BFGS方法全局收敛到$f(\boldsymbol{x})$的极小值点$\boldsymbol{x}^*$。如果进一步在最优点$\boldsymbol{x}^*$的一个邻域内Hessian矩阵李普希茨连续，且迭代点列$\{\boldsymbol{x}^k\}$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$\boldsymbol{x}^*$。该部分理论性质需要大家了解，其证明过程无须掌握。

5.3　有限内存的 BFGS 方法

以BFGS为代表的拟牛顿方法虽然克服了计算Hessian矩阵的困难，但是其仍然难以应用在大规模最优化问题中。例如，机器学习应用中的逻辑回归问题，因其数据规模较大, 采用拟牛顿方法仍然需要很高的计算代价。具体来看，拟牛顿矩阵$\boldsymbol{\mathcal{B}}_k$或者$\boldsymbol{\mathcal{H}}_k$均为稠密矩阵，而存储稠密矩阵要消耗大量的内存，这对于大规模最优化问题来说是不可能实现的。本节重点介绍有限内存BFGS方法（Limited memory BFGS, L-BFGS）来解决存储问题。L-BFGS方法可以认为是BFGS方法的扩展, 其中外部的存储被用来加速收敛。该方法适用于大规模最优化问题，因为其所需的存储内存（以及迭代的成本）可以由用户自主控制。另外，L-BFGS方法可以被看作是拟牛顿方法的高效简易实现，主要原因之一是其不需要知道Hessian矩阵的稀疏性，同样也不需要了解目标函数的可分性，从而可以非常简单地进行编程实现。

Liu和Nocedal于1980年提出了有限内存BFGS方法（L-BFGS）[17]，该方法在实现上几乎与众所周知的BFGS方法相同，唯一的区别是其在矩阵更新方面。L-BFGS方法与传统BFGS方法单独进行存储修正不同，当可用的存储空间用完时，前面的修正被删除以便为新的修正腾出存储空间。后续迭代遵循这种格式：用户指定要保留数量为m的BFGS修正，提供稀疏的对称正定矩阵$\boldsymbol{\mathcal{H}}_0$并用其近似函数$f$的Hessian矩阵的逆。在前$m$次迭代中，L-BFGS方法与BFGS方法相同。对于$k>m$，$\boldsymbol{\mathcal{H}}_k$基于前面$m$次迭代的信息通过对$\boldsymbol{\mathcal{H}}_0$进行$m$次BFGS迭代格式更新得到。

L-BFGS方法是根据BFGS方法变形而来，为了推导方便，我们以$\boldsymbol{\mathcal{H}}_k$的更新方式(5.12)来推导L-BFGS。首先BFGS方法可以重写成以下形式

$$\boldsymbol{\mathcal{H}}_{k+1}=\boldsymbol{V}_k^{\mathrm{T}}\boldsymbol{\mathcal{H}}_k\boldsymbol{V}_k+\boldsymbol{\rho}_k\boldsymbol{s}_k\boldsymbol{s}_k^{\mathrm{T}}\tag{5.15}$$

其中，

$$\boldsymbol{\rho}_k=\frac{1}{\boldsymbol{y}_k^{\mathrm{T}}\boldsymbol{s}_k},\quad\boldsymbol{V}_k=\boldsymbol{\mathcal{I}}-\boldsymbol{\rho}_k\boldsymbol{y}_k\boldsymbol{s}_k^{\mathrm{T}}$$

进一步将式(5.12)递推地展开m次，其中，m是一个自主给定的整数，可以得到

$$\begin{aligned}\boldsymbol{\mathcal{H}}_k = &\left(\boldsymbol{V}_{k-1}^{\mathrm{T}}\boldsymbol{V}_{k-2}^{\mathrm{T}}\cdots\boldsymbol{V}_{k-m}^{\mathrm{T}}\right)\boldsymbol{\mathcal{H}}_{k-m}\left(\boldsymbol{V}_{k-m}\boldsymbol{V}_{k-2}\cdots\boldsymbol{V}_{k-1}\right)\\&+\boldsymbol{\rho}_{k-m}\left(\boldsymbol{V}_{k-1}^{\mathrm{T}}\boldsymbol{V}_{k-2}^{\mathrm{T}}\cdots\boldsymbol{V}_{k-m+1}^{\mathrm{T}}\right)\boldsymbol{s}_{k-m}\boldsymbol{s}_{k-m}^{\mathrm{T}}\left(\boldsymbol{V}_{k-m+1}\boldsymbol{V}_{k-2}\cdots\boldsymbol{V}_{k-1}\right)\\&+\boldsymbol{\rho}_{k-m+1}\left(\boldsymbol{V}_{k-1}^{\mathrm{T}}\boldsymbol{V}_{k-2}^{\mathrm{T}}\cdots\boldsymbol{V}_{k-m+2}^{\mathrm{T}}\right)\boldsymbol{s}_{k-m+1}\boldsymbol{s}_{k-m+1}^{\mathrm{T}}\left(\boldsymbol{V}_{k-m+2}\boldsymbol{V}_{k-2}\cdots\boldsymbol{V}_{k-1}\right)\\&+\cdots+\boldsymbol{\rho}_{k-1}\boldsymbol{s}_{k-1}\boldsymbol{s}_{k-1}^{\mathrm{T}}\end{aligned}\tag{5.16}$$

值得注意的是，为了节省内存，上式不可能无限展开下去，但此时$\boldsymbol{\mathcal{H}}_{k-m}$仍然还是无法显式求解得到，一个自然的想法就是用$\boldsymbol{\mathcal{H}}_{k-m}$的近似矩阵来代替$\boldsymbol{\mathcal{H}}_{k-m}$进行计算，近似矩阵的选取方式非常多，而最常见的一种选取方式是选取对角矩阵$\hat{\boldsymbol{\mathcal{H}}}_{k-m}=\gamma_k\boldsymbol{\mathcal{I}}$来代替$\boldsymbol{\mathcal{H}}_{k-m}$，而$\gamma_k$选取为BB步长[18]，即

$$\gamma_k=\frac{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}}{\boldsymbol{y}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}}$$

综上所述，可以得到如下的L-BFGS方法框架。

L-BFGS方法：

- 给定初始点$\boldsymbol{x}^0$，$0<\beta'<\dfrac{1}{2}$，$\beta'<\beta<1$，误差容忍度ϵ，初始点处的Hessian矩阵的矩阵逆的近似$\boldsymbol{\mathcal{H}}_0$，$k\leftarrow 0$；
- 如果$\|\nabla f(\boldsymbol{x}^k)\|>\epsilon$，则进行下一步；
- 计算搜索方向

$$\boldsymbol{d}_k=-\boldsymbol{\mathcal{H}}_k\nabla f(\boldsymbol{x}^k)\tag{5.17}$$

- 更新迭代点：$\boldsymbol{x}^{k+1}=\boldsymbol{x}^k+\alpha_k\boldsymbol{d}_k$，其中步长$\alpha_k$（初始步长选为1）通过Wolfe线搜索准则得到，即满足

$$f(\boldsymbol{x}^k+\alpha_k\boldsymbol{d}_k)\leqslant f(\boldsymbol{x}^k)+\beta'\alpha_k\nabla f(\boldsymbol{x}^k)^{\mathrm{T}}\boldsymbol{d}_k$$

$$\nabla f(\boldsymbol{x}^k+\alpha_k\boldsymbol{d}_k)^{\mathrm{T}}\boldsymbol{d}_k\geqslant\beta\nabla f(\boldsymbol{x}^k)^{\mathrm{T}}\boldsymbol{d}_k$$

- 计算$\boldsymbol{s}_k=\boldsymbol{x}^{k+1}-\boldsymbol{x}^k$和$\boldsymbol{y}_k=\nabla f(\boldsymbol{x}^{k+1})-\nabla f(\boldsymbol{x}^k)$；
- 由式(5.15)和式(5.16)计算$\boldsymbol{\mathcal{H}}_{k+1}$；
- $k\leftarrow k+1$。

5.4 本章小结

本书重点介绍面向机器学习的最优化方法，通常二阶方法并不适用于机器学习中的最优化问题的求解。本章以牛顿法的介绍为基础，重点介绍了BFGS方法，其为重要的拟牛顿方法也可以看作是二阶方法。作为原始BFGS方法改进的有限内存BFGS方法在

机器学习领域被广泛使用，并可以高效计算大规模机器学习问题。二阶方法有其优点也有缺点，合理利用二阶方法的优点可以增加其对于大规模机器学习问题的适应性。

5.5　习题

1. 设 $f(\boldsymbol{x}) = \frac{3}{2}x_1^2 + \frac{1}{2}x_2^2 - x_1x_2 - 2x_1$，设初始点 $\boldsymbol{x}^0 = (-2, 4)^{\mathrm{T}}$，试用牛顿法求极小化 $f(\boldsymbol{x})$ 的最优解。
2. 设函数 $f(\boldsymbol{x}) = \|\boldsymbol{x}\|^{\beta}$, 其中 $\beta > 0$ 为给定的常数。考虑使用经典牛顿法 (5.1) 对 $f(\boldsymbol{x})$ 进行极小化，初值 $\boldsymbol{x}^0 \neq \boldsymbol{0}$。证明：
   (1) 若 $\beta > 1$ 且 $\beta \neq 2$，则 $\boldsymbol{x}^k$ 收敛到 0 的速度是 $\boldsymbol{Q}$-线性的。
   (2) 若 $0 < \beta < 1$，则牛顿法发散。
3. 分别用牛顿法和拟牛顿法极小化 Rosenbrock 函数
$$f(\boldsymbol{x}) = (x_1 + 10x_2)^2 + 5(x_3 - x_4)^2 + (x_2 - 2x_3)^4 + 10(x_1 - x_4)^4$$
初始点 $\boldsymbol{x}^0 = (3, -1, 0, 1)^{\mathrm{T}}$，解为 $\boldsymbol{x}^* = (0, 0, 0, 0), f(\boldsymbol{x}^*) = 0$。
4. 证明拟牛顿方向是椭球范数 $\|\cdot\|_{B_k}$ 意义下的最速下降方向。
5. (1) 如果 f 是强凸函数，证明不等式 (5.6) 对任意两点 x^k 和 x^{k+1} 都是满足的。
(2) 给出一个满足 $f(0) = -1, f(1) = -\frac{1}{4}$ 的单变量函数，证明在这种情况下不等式 (5.6) 是不成立的。
6. 记 $\{x_k\}$、$\{p_k\}$、$\{H_k\}$ 为采用 BFGS 法解决最小化正定二次问题时生成的迭代序列。目标函数为
$$f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^{\mathrm{T}}\boldsymbol{Q}\boldsymbol{x} - \boldsymbol{b}^{\mathrm{T}}\boldsymbol{x}$$
选取 α_k 满足
$$f(\boldsymbol{x}^k + \alpha_k\boldsymbol{p}_k) = \min_{\alpha} f(\boldsymbol{x}^k + \alpha\boldsymbol{p}_k)$$
假设向量 $\boldsymbol{x}^0, \boldsymbol{x}^1, \cdots, \boldsymbol{x}^{k-1}$ 均不是最优值，则：
   (1) 向量 $\boldsymbol{p}_0, \boldsymbol{p}_1, \cdots, \boldsymbol{p}_{k-1}$ 是 $\boldsymbol{Q}$-共轭的。
   (2) 证明：$\boldsymbol{D}^k = \boldsymbol{Q}^{-1}$。
7. 设函数 $f(\boldsymbol{x})$ 是连续可微的凸函数，且 $\nabla f(\boldsymbol{x})$ 满足李普希茨条件
$$\|\nabla f(\boldsymbol{y}) - \nabla f(\boldsymbol{z})\| \leqslant M\|\boldsymbol{y} - \boldsymbol{z}\|$$
则由 L-BFGS 方法 ($m = 0$) 产生的点列 x^k 必满足 $\lim_{k\to\infty} f(\boldsymbol{x}^k) = -\infty$ 或者 $\lim_{k\to\infty} \nabla f(\boldsymbol{x}^k) = 0$。