非线性最小二乘问题的算法介绍

引言

非线性最小二乘问题（Nonlinear Least Squares，NLS）在科学和工程领域具有广泛的应用，如曲线拟合、参数估计和机器学习等。与线性最小二乘问题相比，非线性最小二乘问题更为复杂，因为残差函数是参数的非线性函数，导致优化问题具有非凸性和多极值点的特征。

在本文中，我们将详细介绍非线性最小二乘问题的数学模型，分析其几何意义，并深入探讨经典的求解算法，如高斯-牛顿法和列文伯格-马夸尔特法。同时，我们将给出这些算法的收敛性定理和证明，并比较两种方法的优缺点。

基本思想

非线性最小二乘问题的目标是找到参数向量 $x \in \mathbb{R}^n$，使得非线性模型函数 $f(x)$ 与观测数据 $y \in \mathbb{R}^m$ 之间的差异最小化。具体地，我们定义残差向量 $r(x) = f(x) - y$，并构建目标函数：

$$\min_{x \in \mathbb{R}^n} \quad F(x) = \frac{1}{2}\|r(x)\|^2 = \frac{1}{2}\sum_{i=1}^{m}[r_i(x)]^2.$$

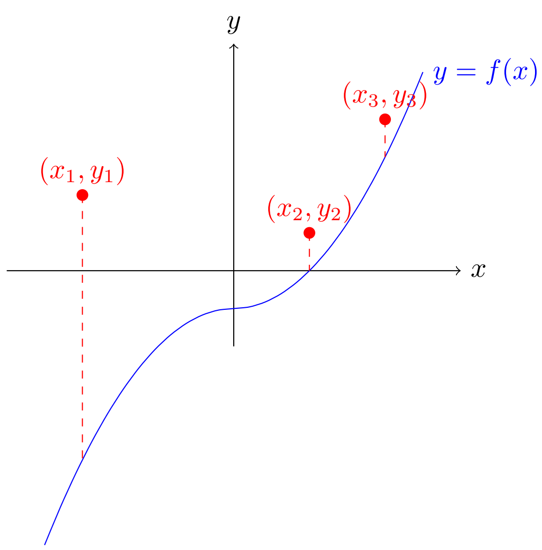
由于 $r_i(x)$ 通常是非线性函数，直接求解最优解 x^* 并不容易。为此，我们借助迭代优化算法，逐步逼近最优解。

几何解释

从几何角度来看，非线性最小二乘问题是寻找参数空间中的一点，使得由参数确定的模型曲面尽可能接近观测数据点。每个参数向量 x 对应一个模型曲面 $f(x)$，我们希望找到使得 $f(x)$ 与 y 之间的欧氏距离最小的 x^*。

几何解释：

非线性最小二乘问题可以视为在高维空间中，将观测数据点 y 投影到模型曲面 $f(x)$ 上的过程。通过调整参数 x，我们使得模型曲面在参数空间中移动，直到找到与数据点最接近的位置。



算法介绍

高斯-牛顿法

高斯-牛顿法是针对非线性最小二乘问题的一种有效迭代算法。其基本思想是在线性近似的基础上，将非线性问题转化为线性最小二乘问题。在第 k 次迭代中，我们对残差函数 $r(x)$ 进行一阶泰勒展开：

$$r(x_k + \Delta x) \approx r(x_k) + J(x_k)\Delta x,$$

其中 $J(x_k) \in \mathbb{R}^{m\times n}$ 是残差函数的雅可比矩阵，定义为：

$$[J(x_k)]_{ij} = \frac{\partial r_i}{\partial x_j}\Big|_{x=x_k}.$$

然后，令增量 Δx 最小化线性近似下的目标函数：

$$\min_{\Delta x} \quad \frac{1}{2}\|r(x_k) + J(x_k)\Delta x\|^2.$$

求解上述问题的正规方程：

$$J(x_k)^T J(x_k)\Delta x = -J(x_k)^T r(x_k).$$

更新参数：

$$x_{k+1} = x_k + \Delta x.$$

列文伯格-马夸尔特法

高斯-牛顿法在残差函数较为线性时表现良好，但在非线性程度较高时可能会出现收敛困难。列文伯格-马夸尔特法（Levenberg-Marquardt method）通过引入阻尼因子，将高斯-牛顿法与梯度下降法结合，提高了算法的鲁棒性。

其更新公式为：

$$\left(J(x_k)^T J(x_k) + \lambda_k I\right)\Delta x = -J(x_k)^T r(x_k),$$

其中 λ_k 为阻尼因子，I 为单位矩阵。当 λ_k 较大时，更新方向接近于梯度下降方向；当 λ_k 较小时，更新方向接近于高斯-牛顿方向。

算法步骤

算法 1：高斯-牛顿法

Algorithm 1 Gauss-Newton Method

```
初始化参数 x_0，设定容差 ε > 0。
for k = 0, 1, 2, ... do
  计算残差 r(x_k) 和雅可比矩阵 J(x_k)。
  求解线性方程组 J(x_k)^T J(x_k)Δx = −J(x_k)^T r(x_k)。
  更新参数 x_{k+1} = x_k + Δx。
  if ‖Δx‖ < ε then
    停止迭代。
  end if
end for
```

算法 2：列文伯格-马夸尔特法

Algorithm 2 Levenberg-Marquardt Method

初始化参数 x_0，阻尼因子 $\lambda_0 > 0$，设定容差 $\varepsilon > 0$，参数 $\nu > 1$。
for $k = 0, 1, 2, \ldots$ **do**
  计算残差 $r(x_k)$ 和雅可比矩阵 $J(x_k)$。
  求解线性方程组 $(J(x_k)^T J(x_k) + \lambda_k I)\Delta x = -J(x_k)^T r(x_k)$。
  计算新的目标函数值 $F(x_k + \Delta x)$。
  if $F(x_k + \Delta x) < F(x_k)$ **then**
    更新参数 $x_{k+1} = x_k + \Delta x$。
    令 $\lambda_{k+1} = \lambda_k / \nu$。
  else
    保持参数不变 $x_{k+1} = x_k$。
    令 $\lambda_{k+1} = \lambda_k \times \nu$。
  end if
  if $\|\Delta x\| < \varepsilon$ **then**
    停止迭代。
  end if
end for

收敛性分析

高斯-牛顿法的收敛性

定理 1 (高斯-牛顿法的局部收敛性)**.** 设 $r(x)$ 是关于 x 的二阶连续可微函数，且在最优解 x^* 附近，雅可比矩阵 $J(x)$ 满秩。若初始点 x_0 足够接近 x^*，则高斯-牛顿法产生的迭代序列 $\{x_k\}$ 将以线性甚至二次速率收敛于 x^*。

证明. 由于 $r(x)$ 二阶连续可微，在 x^* 附近可进行二阶泰勒展开：

$$r(x) = r(x^*) + J(x^*)(x - x^*) + \frac{1}{2}H(x^*)(x - x^*)^2 + o(\|x - x^*\|^2),$$

其中 $H(x^*)$ 为残差函数的二阶导数矩阵。由于 $r(x^*) = 0$，代入高斯-牛顿法的更新公式，可得：

$$x_{k+1} - x^* = (I - (J^T J)^{-1} J^T J)(x_k - x^*) + o(\|x_k - x^*\|).$$

由于 $J(x^*)$ 满秩，$J^T J$ 是正定的，故 $I - (J^T J)^{-1} J^T J = 0$。因此，有：

$$x_{k+1} - x^* = o(\|x_k - x^*\|).$$

这表明算法具有至少线性收敛性，若更进一步的条件满足，可以达到二次收敛。 □

列文伯格-马夸尔特法的收敛性

定理 2 (列文伯格-马夸尔特法的全局收敛性)**.** 在适当的条件下，列文伯格-马奎尔特法对非线性最小二乘问题具有全局收敛性，即从任意初始点出发，算法产生的迭代序列 $\{x_k\}$ 至少收敛到一个局部最小点。

证明. 列文伯格-马奎尔特法通过调整阻尼因子 λ_k，保证每次迭代都使目标函数 $F(x)$ 非增。当 $F(x_{k+1}) < F(x_k)$ 时，我们减小 λ_k，使得算法更倾向于高斯-牛顿方向，加快收敛速度；当 $F(x_{k+1}) \geq F(x_k)$ 时，我们增大 λ_k，使更新方向更接近于梯度方向，增强稳定性。

由于目标函数 $F(x)$ 有下界（即非负），且算法保证 $F(x)$ 单调非增，因此 $\{F(x_k)\}$ 收敛。结合参数更新策略和步长控制，可以证明算法的全局收敛性。 □

方法比较

高斯-牛顿法与列文伯格-马夸尔特法的比较:

- **收敛速度**：高斯-牛顿法在残差函数近似线性的情况下，收敛速度较快，可能达到二次收敛；列文伯格-马夸尔特法由于引入了阻尼因子，收敛速度可能略有降低。
- **鲁棒性**：高斯-牛顿法对初始点和非线性程度较为敏感，可能出现收敛困难；列文伯格-马夸尔特法通过阻尼因子调整，增强了算法的鲁棒性。
- **实现难度**：高斯-牛顿法实现较为简单，但需要求解可能病态的线性方程组；列文伯格-马夸尔特法需要设置阻尼因子调整策略，稍微复杂一些。

应用与案例分析

非线性最小二乘问题在实际中有诸多应用，例如：

- **曲线拟合**：通过调整模型参数，使得曲线尽可能贴合实验数据。
- **参数估计**：在统计学中，利用最小二乘方法估计模型参数。
- **机器学习**：在训练非线性模型（如神经网络）时，最小化损失函数。

案例：指数模型拟合

假设我们有一组数据点 $\{(t_i, y_i)\}$，希望拟合指数模型 $y = ae^{bt}$，其中 a 和 b 为待定参数。定义残差：

$$r_i(a,b) = ae^{bt_i} - y_i.$$

构建目标函数：

$$F(a,b) = \frac{1}{2}\sum_{i=1}^{n}[r_i(a,b)]^2.$$

利用列文伯格-马夸尔特法，我们可以迭代求解参数 a 和 b，使得模型曲线最佳拟合数据。

总结

非线性最小二乘问题是优化领域的重要问题，在科学研究和工程实践中有着广泛的应用。通过对高斯-牛顿法和列文伯格-马奎尔特法的深入分析，我们了解了求解非线性最小二乘问题的有效方法。
高斯-牛顿法利用线性近似，在残差函数较为线性的情况下具有快速的收敛速度。列文伯格-马奎尔特法通过引入阻尼因子，增强了算法的鲁棒性，能够适应更广泛的非线性问题。
在实际应用中，合理选择初始参数、阻尼因子调整策略以及高效的数值计算方法，对于算法的成功实施至关重要。未来的研究可以进一步探讨更为复杂的非线性问题，以及在大规模数据和高维参数空间中的应用。