修正牛顿法（阻尼牛顿法和L-M算法）

黑塞矩阵H可能不是正定的，也不一定是可逆的，而牛顿法则要求H是正定的。当初始值较远时，误差函数往往是高度非线性的，局部二次近似的误差较大，计算不准确。

阻尼牛顿法

加入了对步长的线搜索，与最速下降法中的搜索方式相同，找到最小的α，使得f(xk+αd)最小。

3.2 阻尼牛顿法

由于不能保证 Hessian 矩阵 **正定**，故 **牛顿方向** $-\nabla^2 f(x^k)^{-1}\nabla f(x^k)$ 不一定下降。所以，引入了 **阻尼牛顿法**，在 **牛顿方向上加上线搜索**，确定步长 α，以此来保证搜索方向下降：

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

当 **步长** α 取固定值为 1，则退化为一般的 **牛顿法（经典牛顿法）**。

阻尼牛顿法基本过程：

- 任选初始点 $x^0 \in \mathbb{R}^n$，$k = 0$
- 计算 $\nabla f(x^k)$，若目标函数满足终止条件 $||\nabla f(x^k)|| < \varepsilon$ (或者其他终止条件)，则令 $x^* = x^k$ 为最终解，结束整个算法
- 计算 Hessian 矩阵 $\nabla^2 f(x^k)$，
- 计算搜索方向 $d^k = -\nabla^2 f(x^k)^{-1}\nabla f(x^k)$
- 沿着搜索方向 d^k 进行 **线搜索**，确定 **步长** α_k
- 迭代更新：$x^{k+1} = x^k + d^k = x^k - \alpha_k \nabla^2 f(x^k)^{-1}\nabla(x^k)$

L-M算法

通过一个系数让算法在最速下降法和牛顿法之间做权衡

3.3 L-M算法

牛顿法中在迭代点 x^k 处的 Hessian 矩阵 $\nabla^2 f(x^k)$ 可能是 **奇异、非正定** 等情况，**无法保证** Hessian **矩阵可逆**。

故可以为 Hessian 矩阵加上一个 λI 来 **强行让** Hessian **矩阵变得正定**：

$$[\nabla^2 f(x^k) + \lambda I]d^k = -\nabla f(x^k)$$

即使得 $[\nabla^2 f(x^k) + \lambda I]^{-1}$ 存在，所以有 L-M **算法下的牛顿方向**：

$$d^k = -[\nabla^2 f(x^k) + \lambda I]^{-1}\nabla f(x^k)$$

其中 $\lambda \in \mathbb{R}^+$ 且 I 为单位阵。

在机器学习邻域的线性回归问题中，岭回归也是采用这种方法，使得协方差矩阵的逆一定存在。

在 L-M 算法中，只要 λ 足够大，一定可以使得 Hessian 矩阵 的所有特征值都大于 0，即 **保证** Hessian **矩阵一定正定** 。