# 求解参数的两种常见方法

## 一、梯度下降法Gradient Descent

### (1)梯度

对多元函数的参数求偏导数，把求得的各个参数的偏导数以向量的形式写出来。从几何意义上讲，是函数变化最快的地方, 更加容易找到函数的极值。

### (2)相关概念

1.步长（Learning rate）：步长决定了在梯度下降迭代的过程中，每一步沿梯度负方向前进的长度。

2.特征（feature）：指的是样本中输入部分。单特征样本，其中特征值为，输出值为

3.假设函数（hypothesis function）：在监督学习中，为了拟合输入样本，而使用的假设函数，记为。单特征可以采用的拟合函数可以为

4.损失函数（loss function）：为了评估模型拟合的好坏，通常用损失函数来度量拟合的程度。损失函数极小化，意味着拟合程度最好，对应的模型参数即为最优参数。在线性回归中，损失函数通常为样本输出和假设函数的差取平方。

### (3)详细算法

1.先决条件：确认假设函数和损失函数

例如，在线性回归中，假设函数为，损失函数为

2.参数初始化：一般初始化为0，步长为1，算法终止距离

3.计算当前位置的梯度：

4.计算当前位置下降距离：

5.确定是否所有的的下降距离都小于算法终止距离，如果小于，则结束，否则继续。

6.同时更新所有的，重新从第3步开始执行

#### (4)算法调优

1. 算法的步长选择。

取决于数据样本，可以多取一些值，从大到小，分别运行算法，看看迭代效果，如果损失函数在变小，说明取值有效，否则要增大步长。步长太大，会导致迭代过快，甚至有可能错过最优解。步长太小，迭代速度太慢，很长时间算法都不能结束。所以算法的步长需要多次运行后才能得到一个较为优的值。

2. 算法参数的初始值选择。

初始值不同，获得的最小值也有可能不同，因此梯度下降求得的只是局部最小值。由于有局部最优解的风险，需要多次用不同初始值运行算法，选择损失函数最小化的初值。

3. 归一化。

由于样本不同特征的取值范围不一样，可能导致迭代很慢，为了减少特征取值的影响，可以对特征数据归一化，也就是对于每个特征x，求出它的期望和标准差，然后转化为：，这样特征的新期望为0，新方差为1，迭代次数可以大大加快。

### (5)分类

1.批量梯度下降法Batch Gradient Descent

更新参数时使用所有样本，速度慢，准确度高

2.随机梯度下降法Stochastic Gradient Descent

更新参数时只使用其中一个样本，速度快，准确度低

3.小批量梯度下降法Mini-batch Gradient Descent

更新参数时使用总样本中一些样本，速度中，准确度中

### (6)梯度下降法和其他无约束优化算法的比较

1.与最小二乘法相比：

梯度下降法需要选择步长，而最小二乘法不需要。梯度下降法是迭代求解，最小二乘法是计算解析解。如果样本量不算很大，且存在解析解，最小二乘法比起梯度下降法要有优势，计算速度很快。但是如果样本量很大，用最小二乘法由于需要求一个超级大的逆矩阵，这时就很难或者很慢才能求解解析解了，使用迭代的梯度下降法比较有优势。

2.与牛顿法/拟牛顿法相比：

两者都是迭代求解，不过梯度下降法是梯度求解，而牛顿法/拟牛顿法是用二阶的海森矩阵的逆矩阵或伪逆矩阵求解。相对而言，使用牛顿法/拟牛顿法收敛更快。但是每次迭代的时间比梯度下降法长。

## 二、最小二乘法

### 1.详细算法

利用损失函数对各个参数求导数，并使导数为0，求出各个参数

拟合函数,损失函数

### 2.局限性

(1) 在特征数量大于样本数量时，逆矩阵求解非常耗时，甚至不可行。此时梯度下降依旧可用，也可以通过主成分分析降低特征的维度后，使特征数量小于等于样本数量时，再使用最小二乘法。

(2) 拟合函数不是线性时，最小二乘法不可用。