1. **凸函数的概念，性质**
2. **梯度下降法：**
3. 梯度方向表示了函数增长速度最快的方向，那么和它相反的方向就是函数减少速度最快的方向了
4. 对于机器学习模型优化的问题，当我们需要求解最小值的时候，朝着梯度下降的方向走，就能找到最优值了
5. 最简单的梯度下降算法由两个函数，三个变量组成：

函数1：待求的函数

函数2：待求函数的导数

变量1：当前找到的变量，这个变量是“我们认为”当前找到的最好的变量，可以是函数达到最优值（这里是最小值）。

变量2：梯度，对于绝大多数的函数来说，这个就是函数的负导数。

变量3：步长，也就是沿着梯度下降方向行进的步长。也是这篇文章的主角。

1. 梯度下降法的learning rate有个阙值:当大于这个数的时候，会发散，小于这个数的时候，会收敛；rate取这个值的时候，会在最低点附近来回摇摆。
2. **最速下降法：**
   1. 办法不去计算这个learning rate，又能让剃度下降的每一步尽可能地走好
   2. 最速下降法。所谓的最速下降法，就是在确定下降方向后，从下降方向中找到下降程度最大的一点进行下降。
   3. 最速下降法在求出梯度之后，要进行另外一个小优化问题的求解过程，那就是选择最合适的learning rate，使得函数值最小，如果待求的函数为f(x)，当前的迭代轮数为t，那么当前函数的优化的参数解是，这一点的梯度为，于是我们小优化问题就变成了：
   4. 最速下降法的优化方向有一个特点，那就是相邻两轮迭代的梯度相互正交。这个特点对于推导最终的算法十分重要，我们首先来证明一下。
   5. 对于被第t轮更新后来说，在这个方向上应该是最小的了，换句话说在这个点上，关于的方向梯度应该为0，那么[公式]就不应该有[公式]方向的分量。如果有就说明上一轮迭代并没有做到最优，和我们的假设矛盾，所以假设不成立，我们最终可以认定，[公式]和[公式]是正交的。