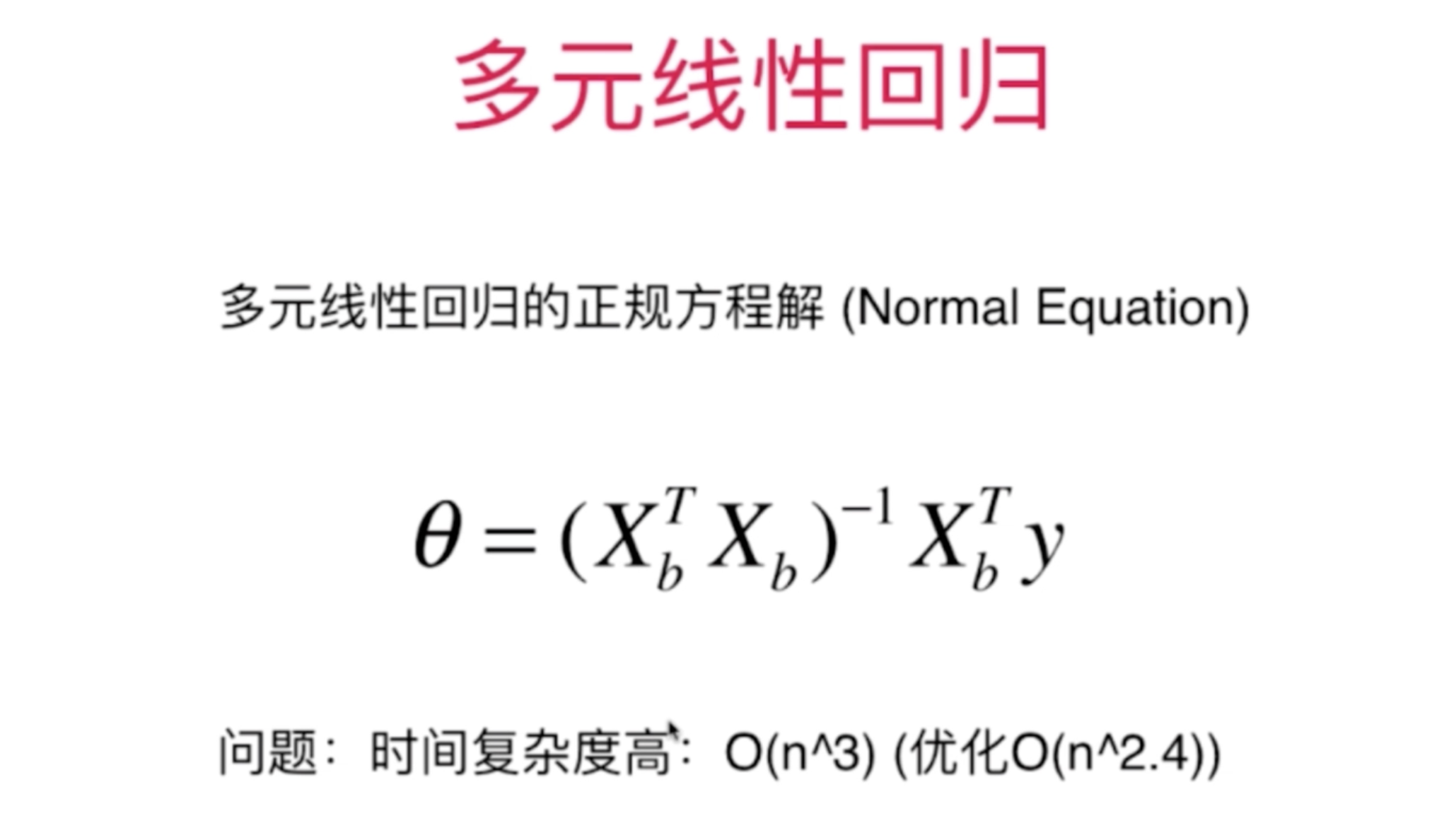


我们上一章中介绍线性回归算法中最后提到：



计算上面的式子，看起来很简单，但是时间复杂度是比较高的，如果我们的数据是巨大的、或者特征数量是巨大的，对于生产环境使用这种线性回归的正规方程求解是没有意义的。

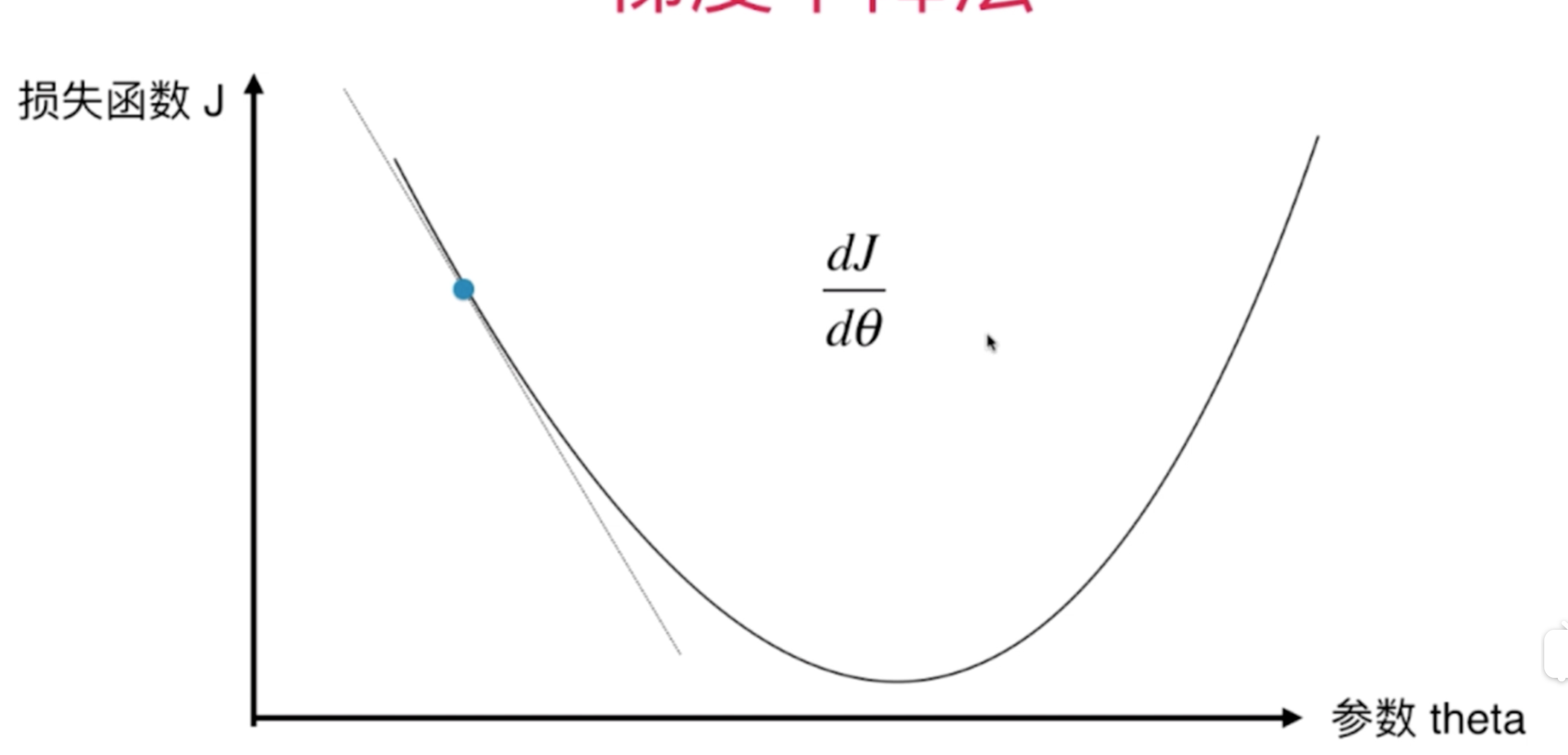
其实上面的线性回归算法最终就是要求救一个线性回归的模型，本质上就是要最小化一个损失函数。而那个公式就是最小化这个函数对应的参数它的数学解，其实在面对普遍的问题时，我们的机器学习的模型是求不到这样带明确公式的数据解的，那么基于这样的模型就要使用一种搜索的策略来找到这个最优解。

梯度下降法就是在机器学习领域最小化损失函数最为常用的方法



这是模型参数和损失函数值得一个函数图

图中的点，如果它的导数！= 0，它就肯定不是一个极值点，就不是我们要找是损失函数最小的theta取值。



那么，我们来看看这个导数的意义：

在直线方程中，导数代表斜率

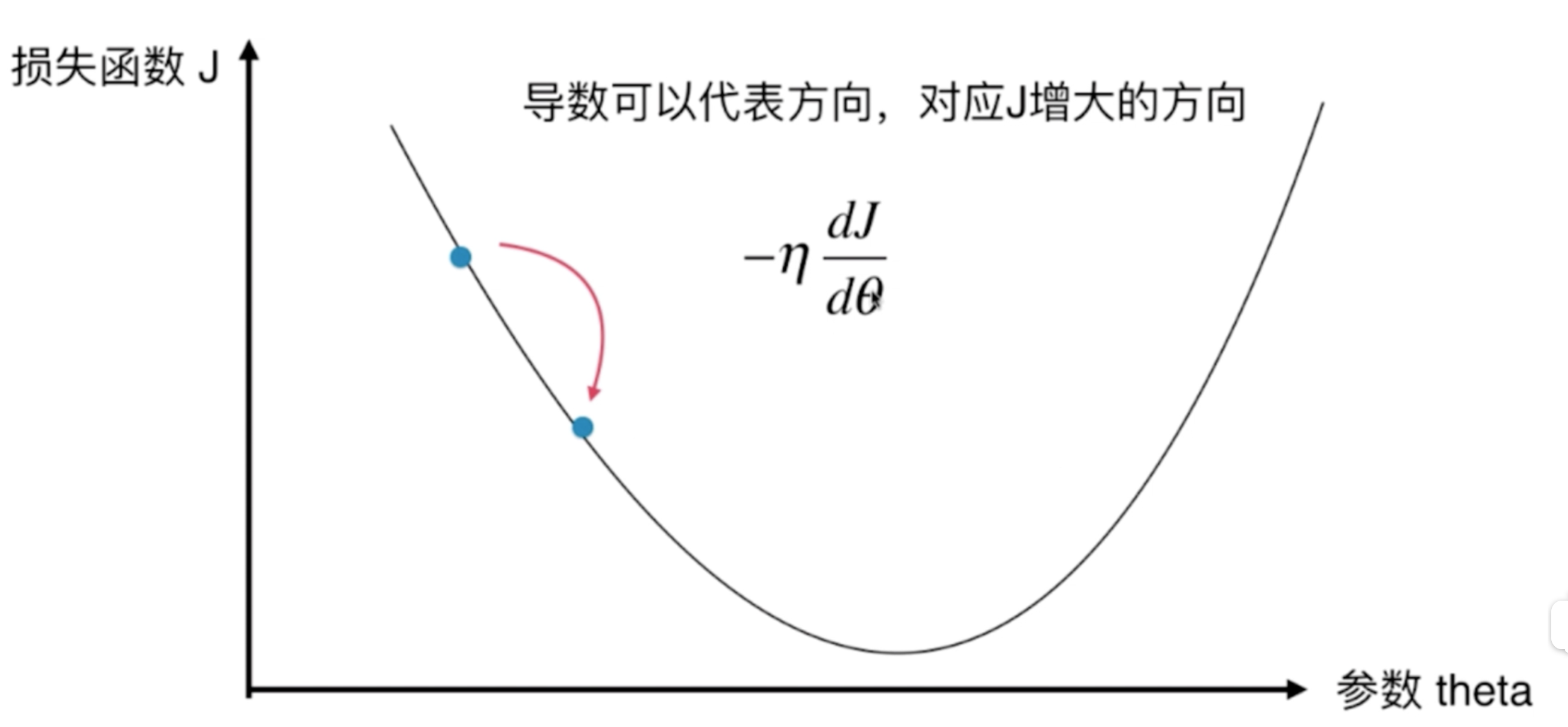
在曲线方程中，导数代表切线的斜率

导数代表：theta单位变化时，J相应的变化

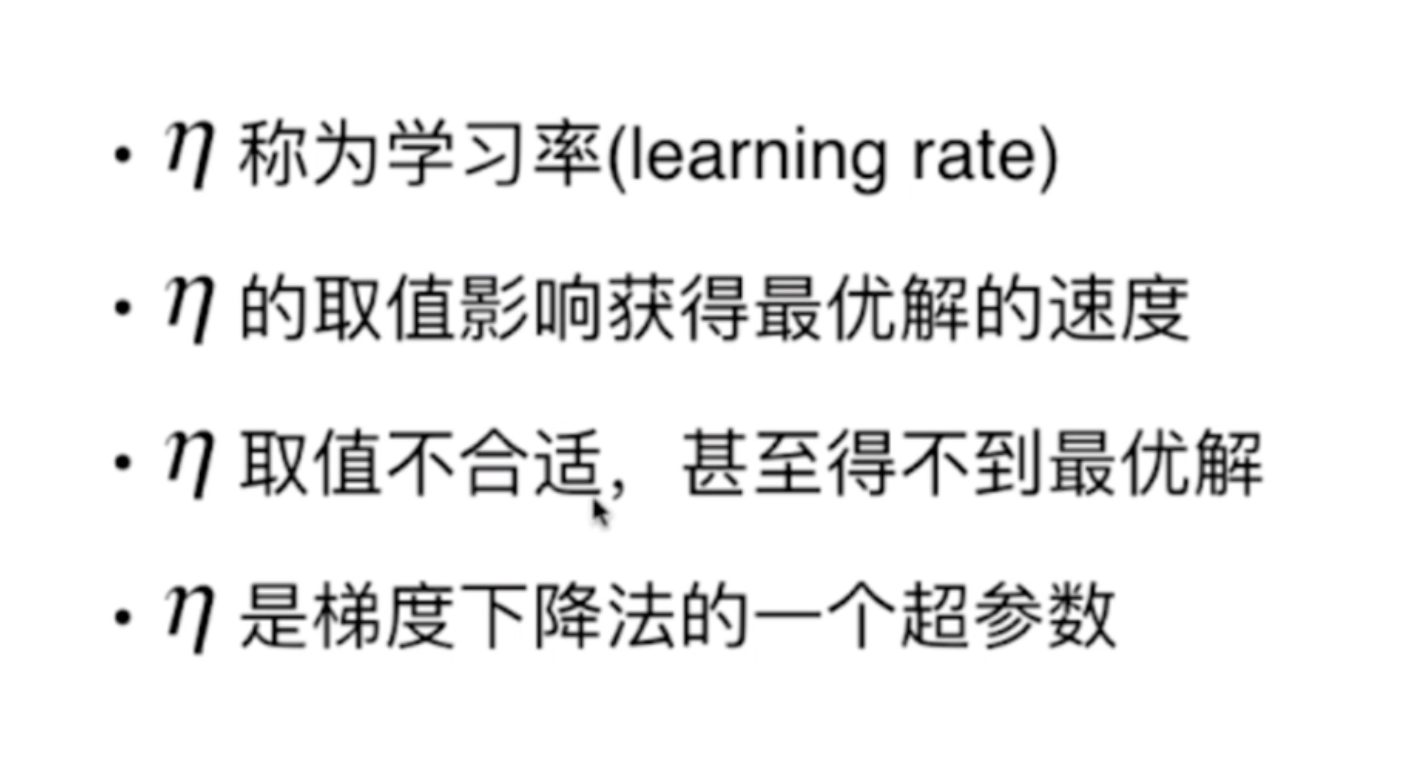
导数为正：theta增加，J也增加

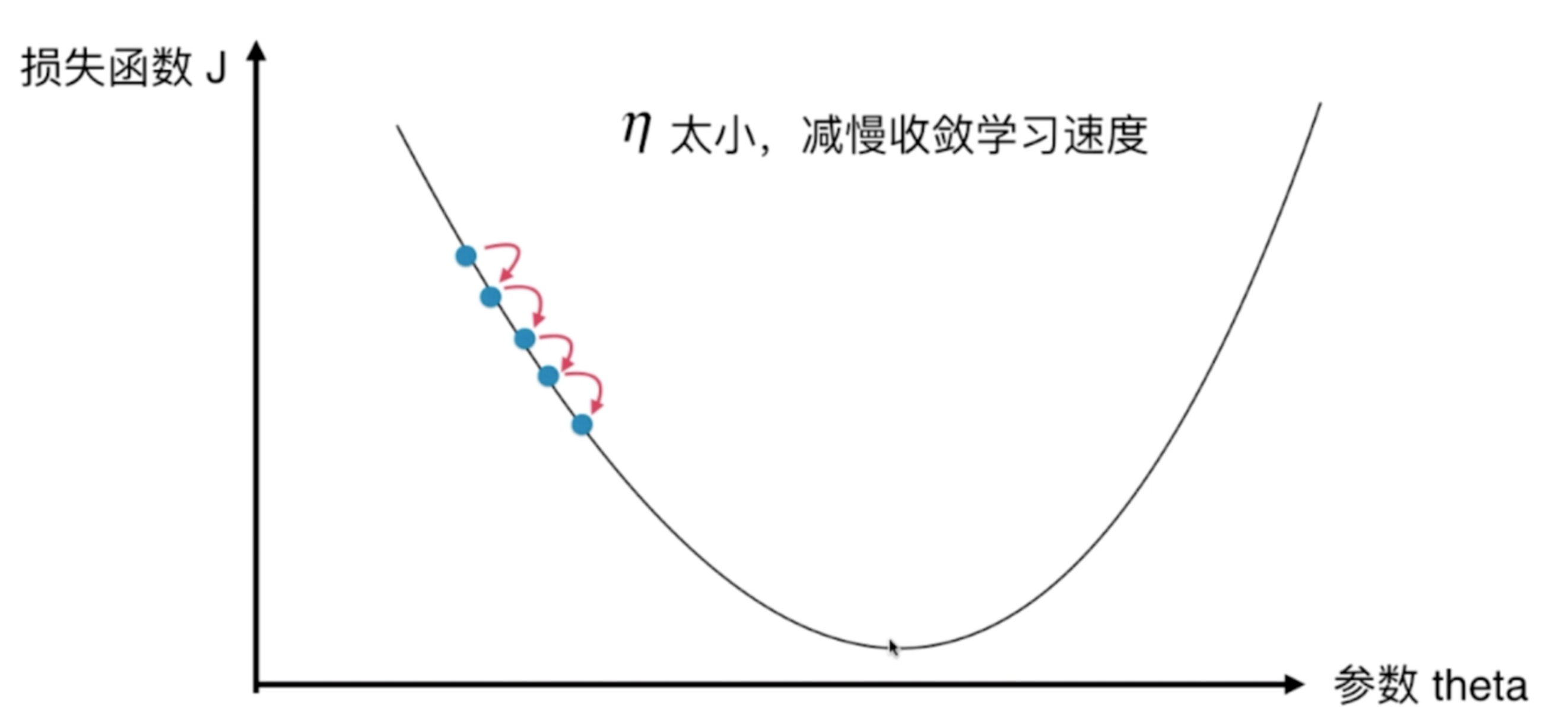
导数为负：theta增加，J减少

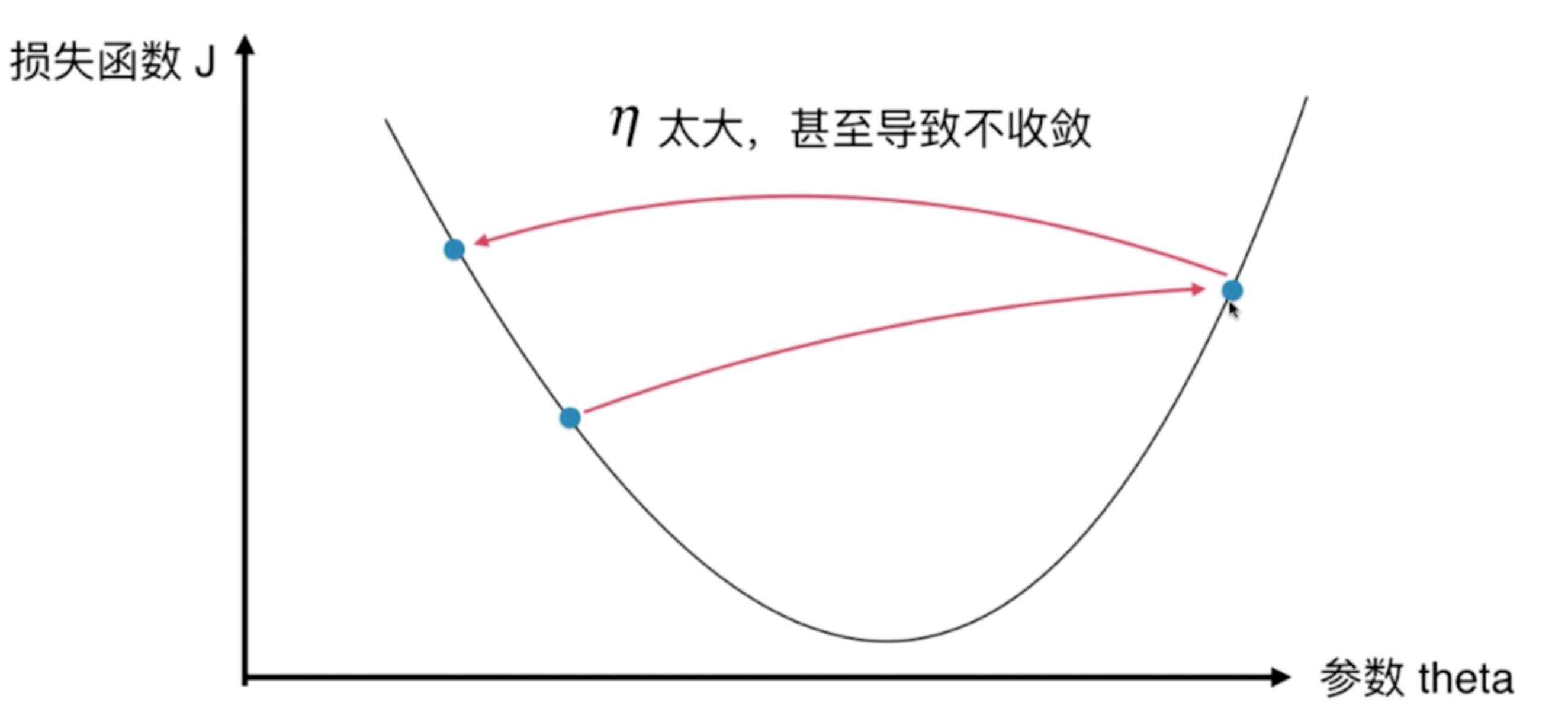
对于一个点来说，我们算出它的导数就知道J 增大和减少的方向，也知道J变化趋势的大小。我们完全可以把导数用于搜索的过程。因为导数大说明变化大，移动的速度也可以快一点，步子可以大一点，如果导数（的绝对值）小说明变化小了，移动的速度要慢下来，步长可以小一点，否则容易跨过最小值点，进入另一侧的上升曲线 。



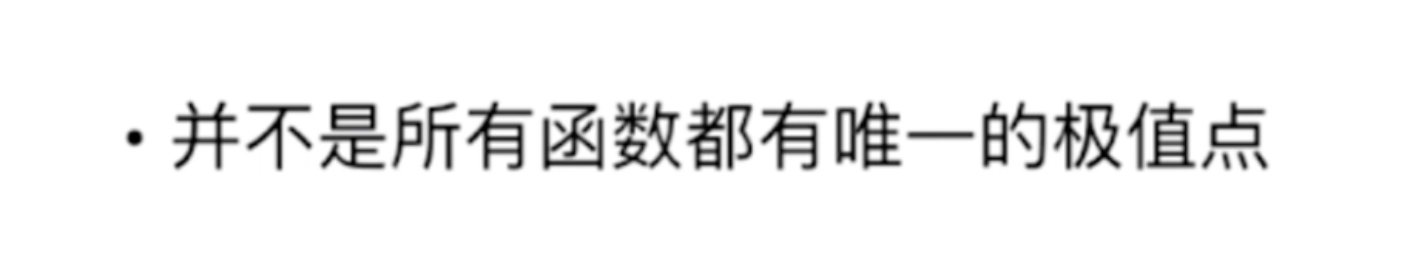
比如上图的第一个点的导数是负数，所以theta增加时J减少，所以我们在第一个点的theta基础上加上（-导数），就是我们想要的下降的方向，为了调整移动的大小，加入一个步长因子，读作Eta， 。



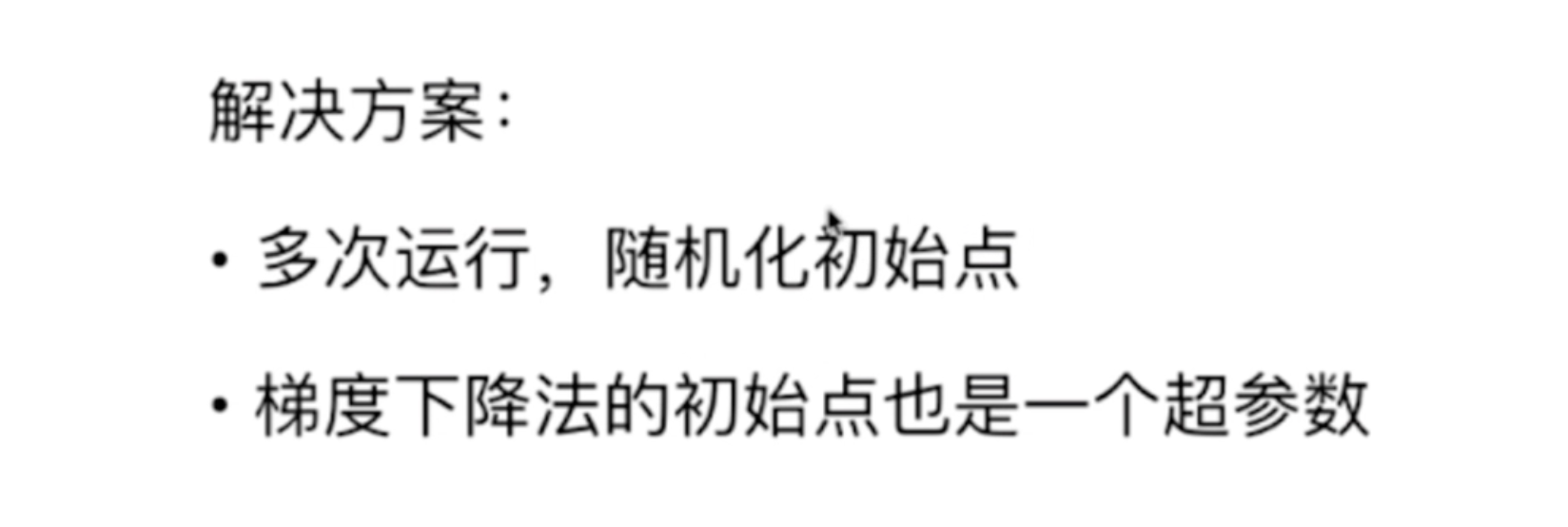




什么是梯度？ 对于一元函数是求导数，对于多元函数就是对各个方向分量分别求导最终得到的方向就是梯度。







这里学习的案例是线性回归的只有一个极值点

