导数，导函数，

偏导数，偏导函数

梯度，方向导数

# 导数

求曲线某一点的斜率

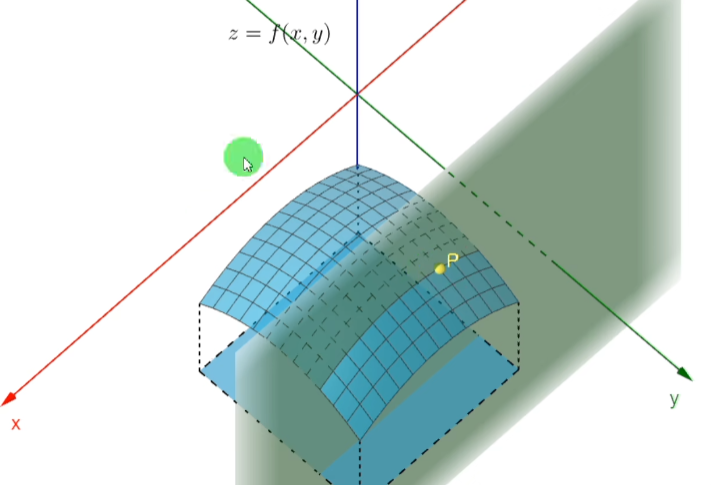
# 导函数

就是曲线函数各个点的斜率的集合

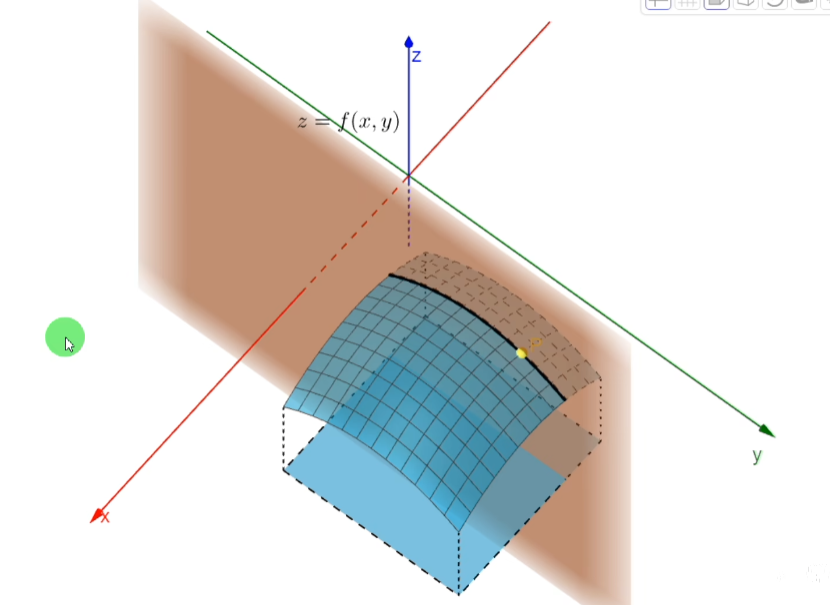
# 偏导数，针对立体空间

空间曲面在某一个方向投影的某个点的斜率

x方向的偏导

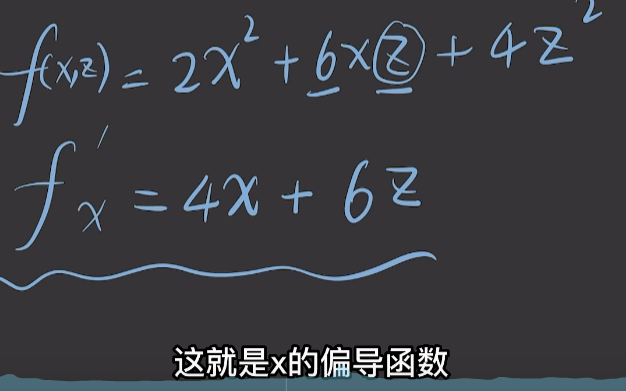


y方向的偏导



# 偏导函数

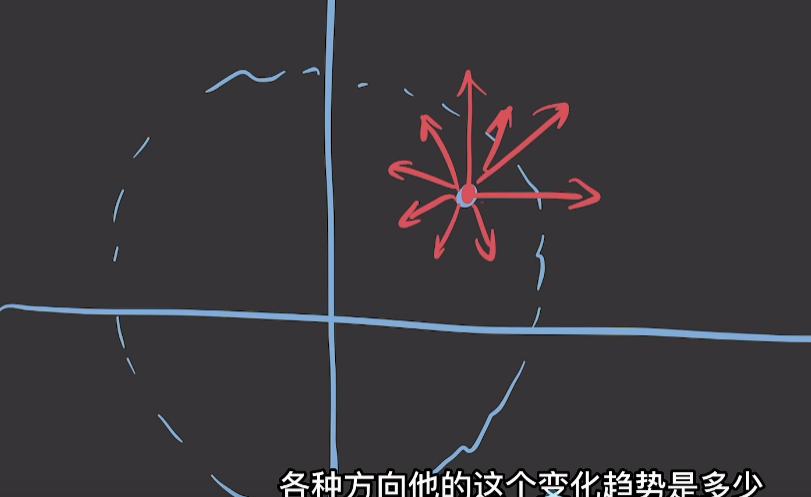
就是拿三维图像的函数导一下



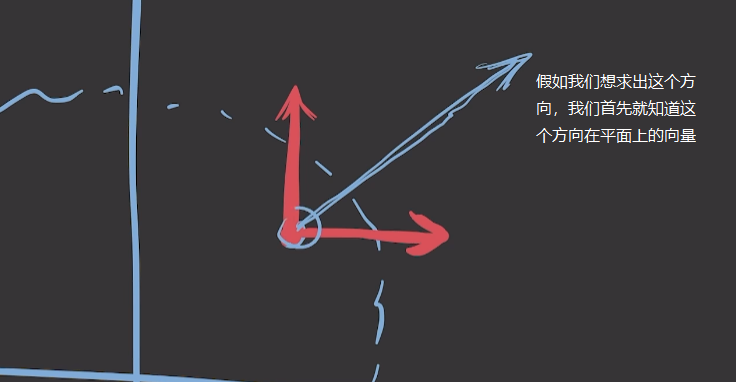
我们求偏导有啥意义呢

# 方向导数

我们只需要知道x，y轴的偏导数，我们就可以求出这个点各个方向的斜率

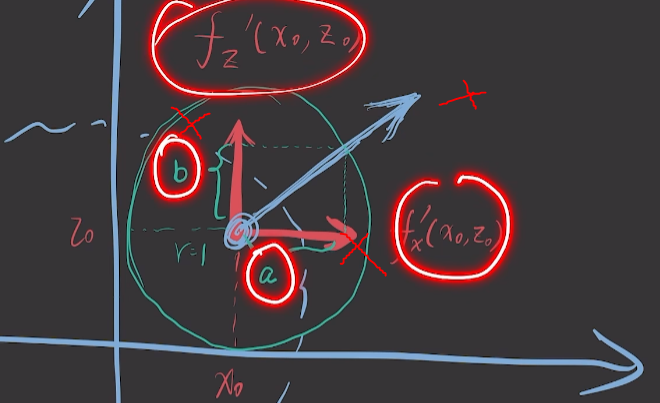


方首先我们知道这个点的坐标，可以求出x，y轴方向上的斜率



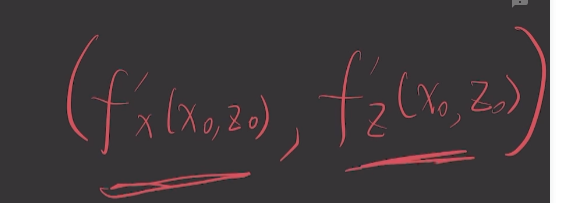
就向下图这样子的公式去求，就可以求出这点的方向导数了，方向导数是一个数值大小

F（x）这两个函数也是数值大小，就是x，y轴的斜率

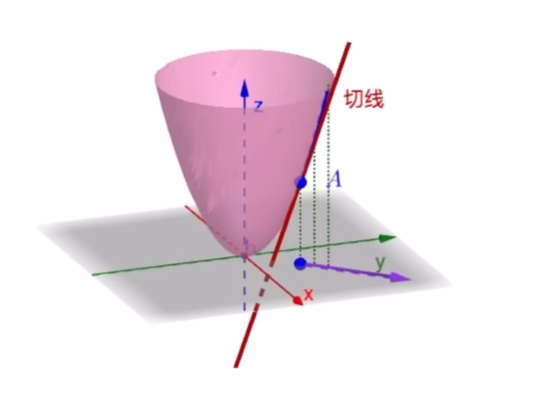


梯度就是我不自己决定走哪个方向了，我就想知道哪个方向是高度上升最快的方向

怎么找出方向呢，直接下图，（x轴的偏导，y轴的偏导）

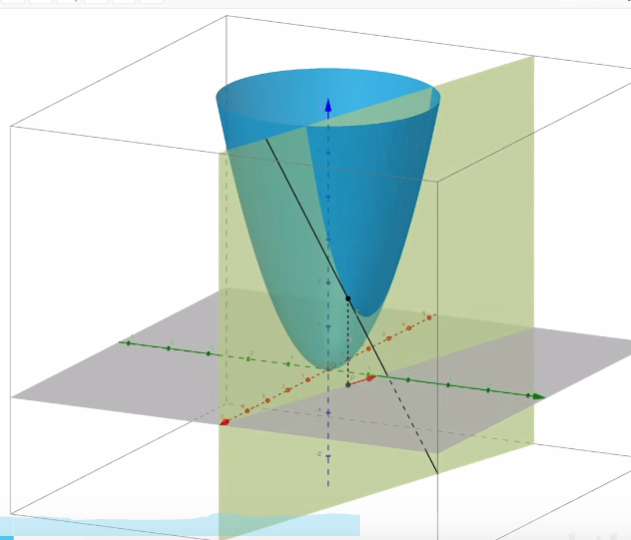


当我的界面沿着紫色方向，不平行x轴，y轴

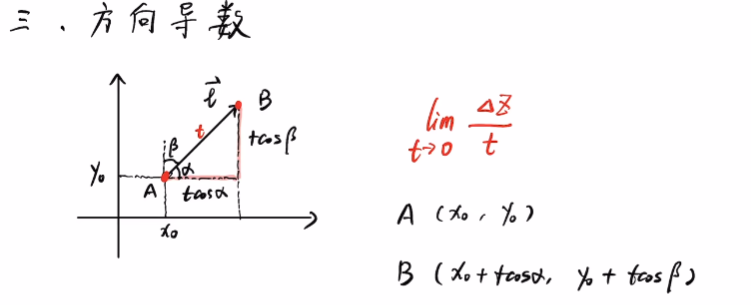


<https://www.bilibili.com/video/BV1uZ4y1L7bB/?spm_id_from=333.788&vd_source=73f0f43dc639135d4ea9acffa3ad6ae0>

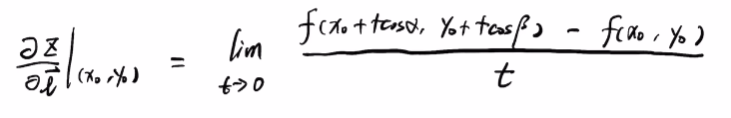
下图底部有一个红色向量，就是我们的界面方向，可以360度旋转，黑色斜线就是斜率，



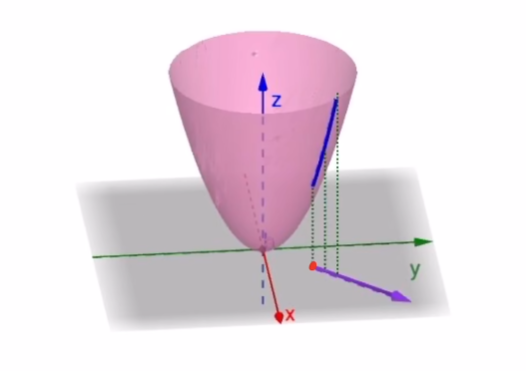
下图t使我们定义的单位向量，知道t，A的坐标，既可以知道b的坐标



那么我们下图公式



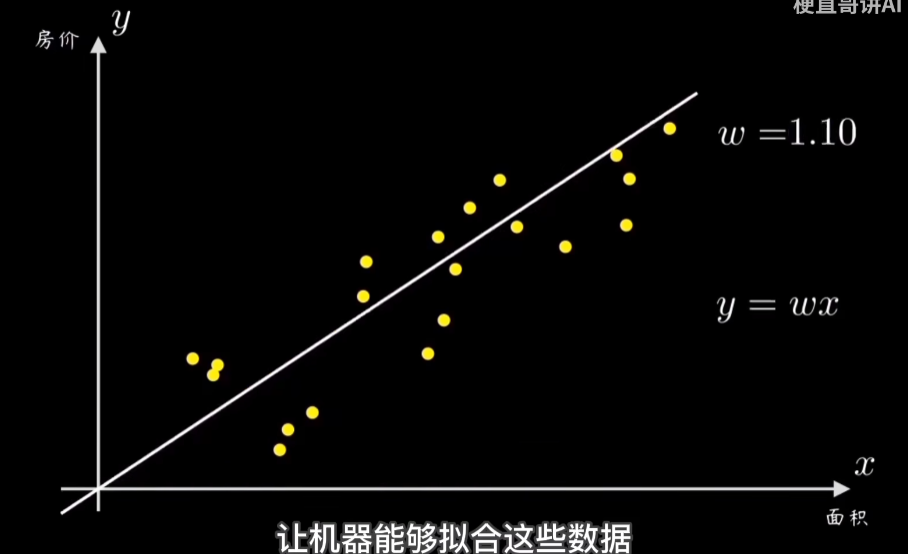
对应下面的图的 蓝线的高度差/紫线的长度



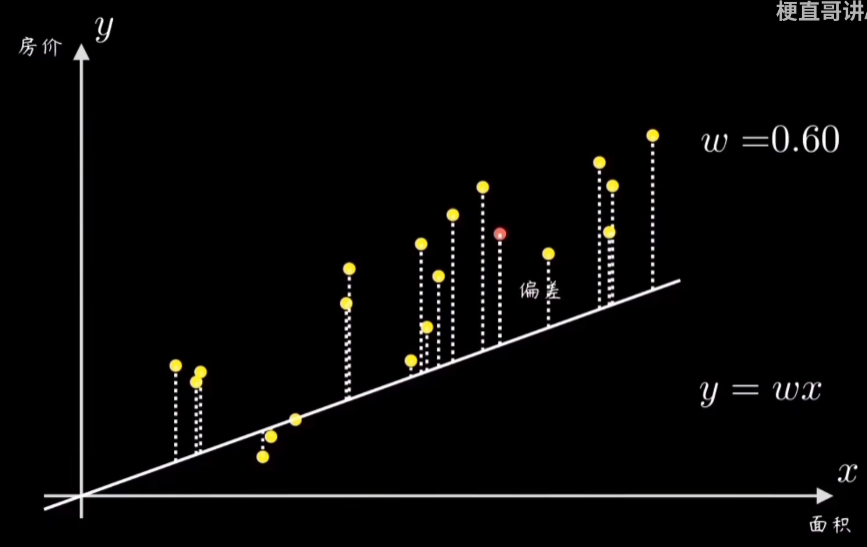
# 梯度

每走一段路，就看下那个方向的斜率最陡，就走哪个方向

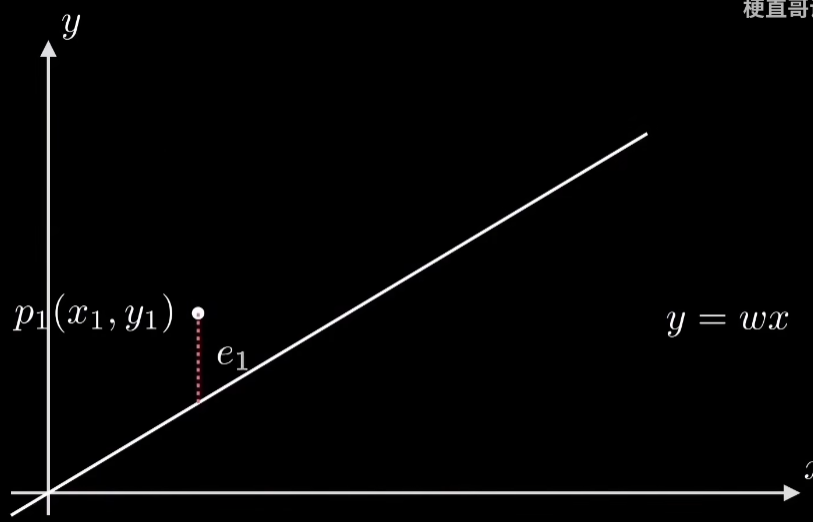
设计一个算法算出w，使这条直线能够拟合这些点，按照下图，我们调整是斜率w，在神经网络中对应的就是权重。调整到一个合适的斜率，使得均方误差最小。



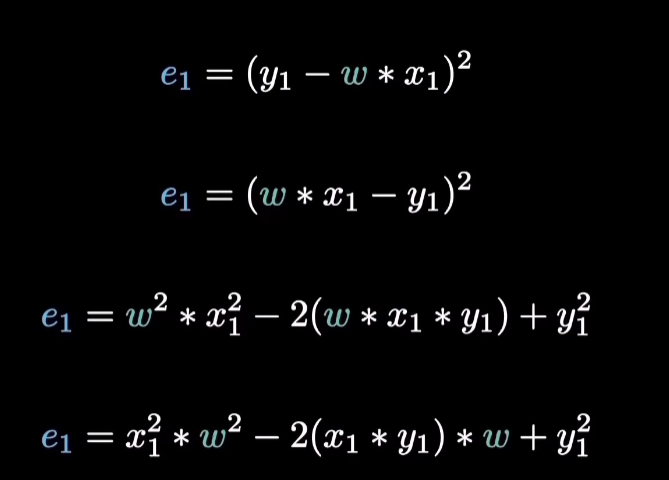
找出每个点距离直线的误差，我们知道每个点的坐标



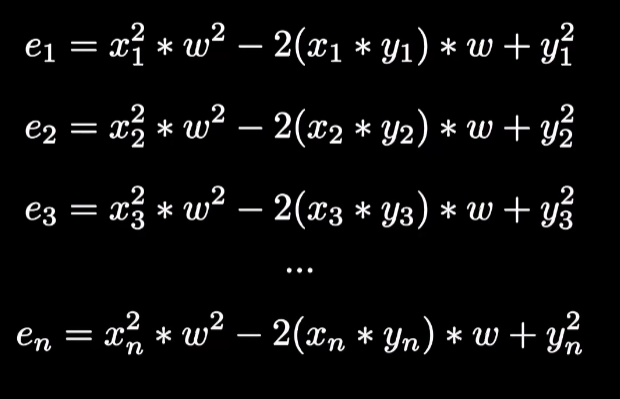
x1和y1，还有知道函数y=wx，就可以知道e1



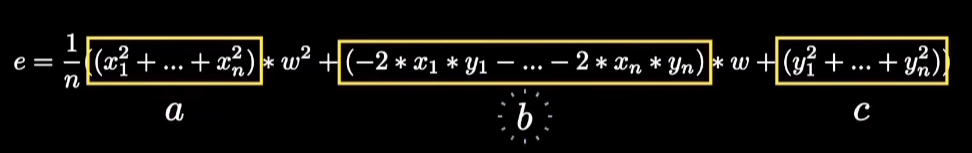
计算公式为，下面是一个点的代入



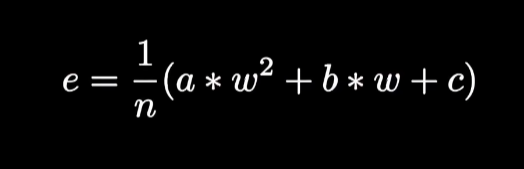
然后我们有很多个点



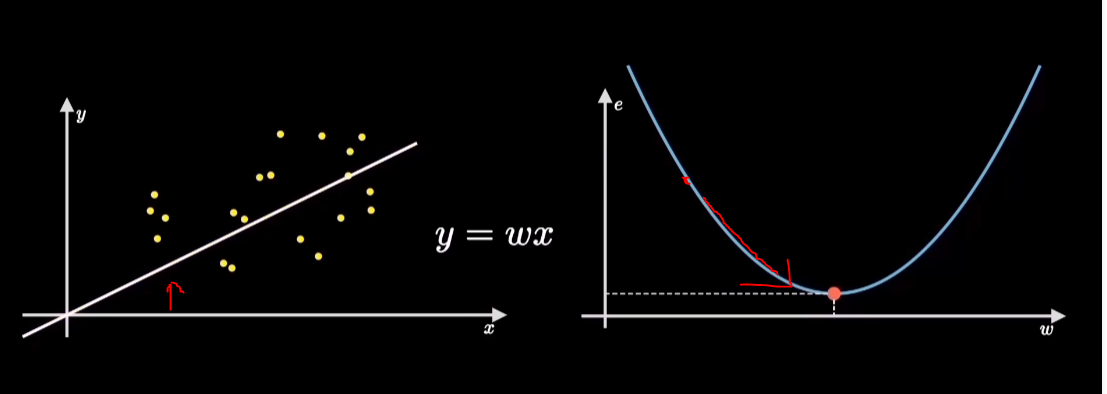
我们目标是求所有e的平均值，我们将所有的e相加

e1+e2+e3...+en=

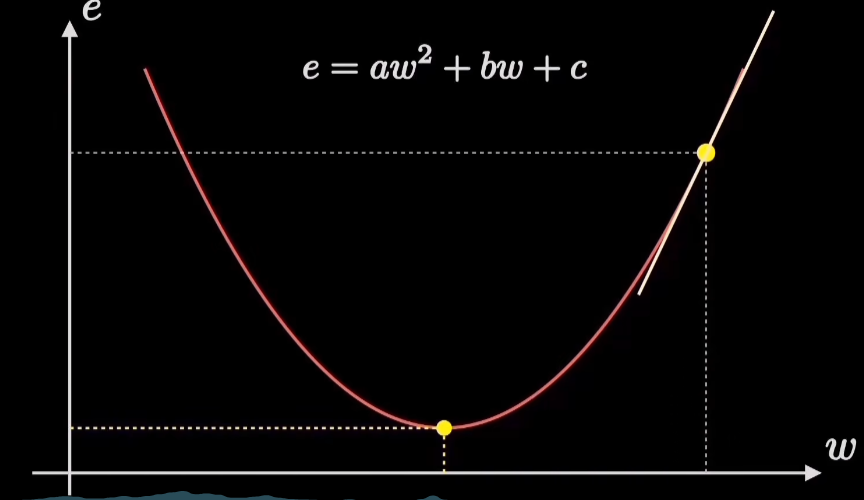
将所有x，y代入算出常数，得出一个一元二次方程，函数表示误差和权重的关系



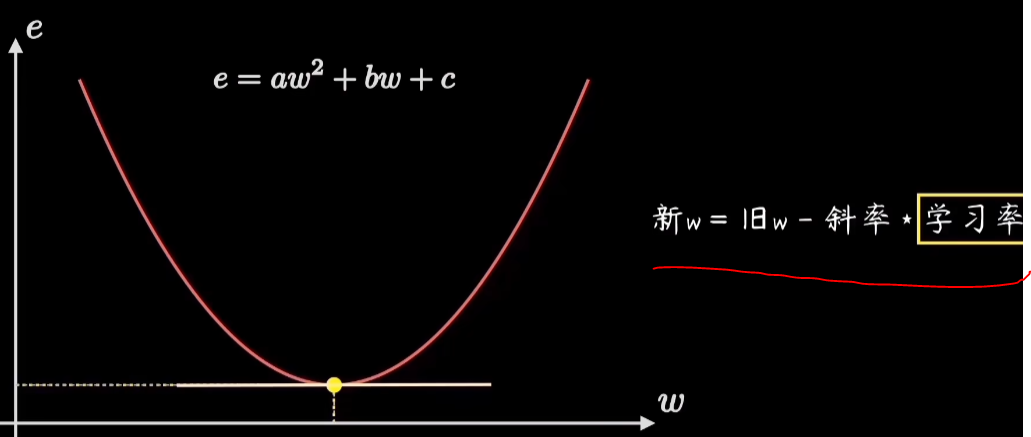
右边的抛物线就叫做代价函数，当左边的斜率变大，对应右边曲线的红点往左移



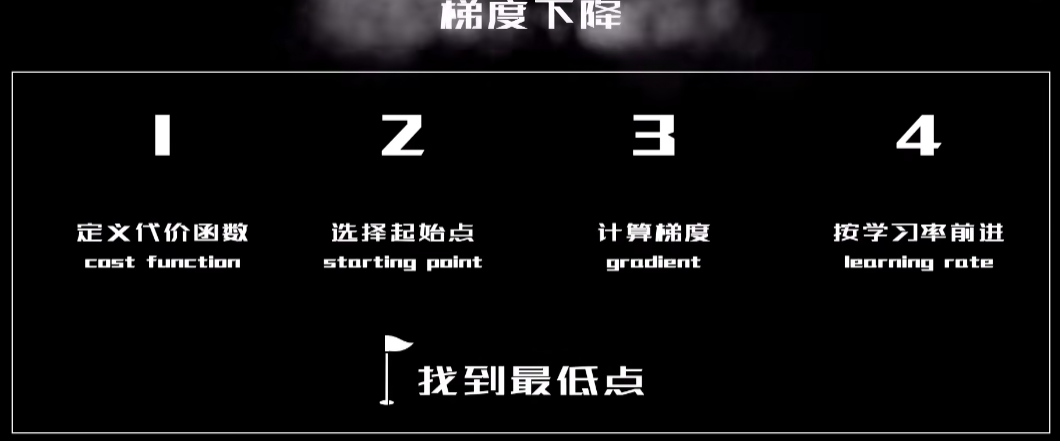
当我们在曲线的某一个点，根据导数推断出斜率，知道斜率我们就知道往哪个方向可以降到最低点。



现在我们就需要往左移动，设置步伐为1，我们发现有可能无法到达最低点，可能会在最低点反复横跳。我们研究出一种新的办法，就是根据斜率去设定步伐，发现更大了，所以我们会规定一个学习率例如0.01，斜率\*0.01来缩小斜率的值。

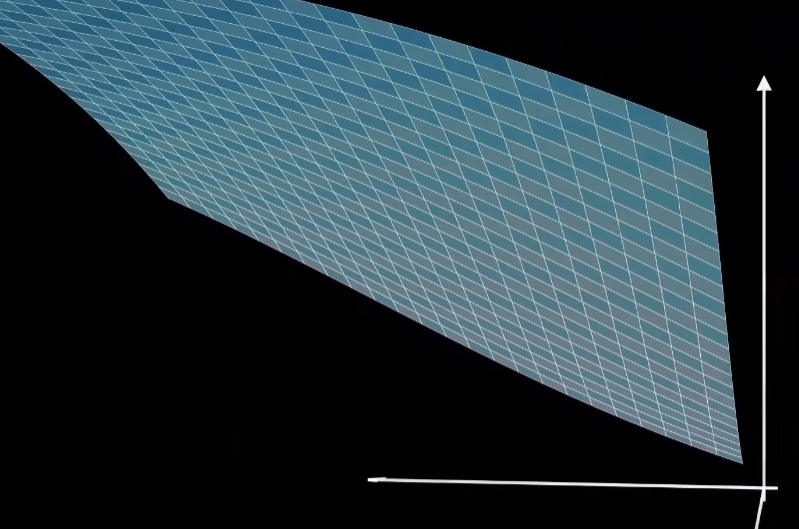


一开始是1到4，后面就是重复3,4



但是在实际的训练过程中，不可能这么简单，上面我们讲到的是只有一个变量X去映射y的值。但在生活中我们的变量不可能只有x，例如房价的预测，房价的预测不仅仅包括土地面积，也包括城市，楼层，区域，朝向，政策，一个变量就代表一个维度。

假如我们价格只跟土地面积和楼层有关，那我们我们的函数就变成了一个三维图像。x就代表土地面积,Y就代表楼层,z就代表价格。



代价函数还有可能代表曲线

