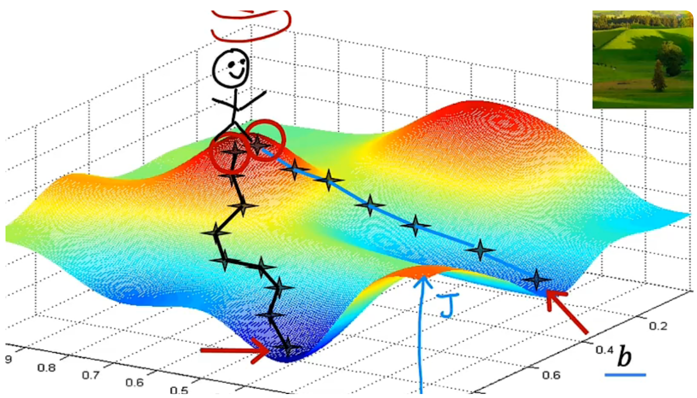
4.梯度下降

4.1 概念

想要找到损失函数J(w,b)的最小值,除了画出函数图像寻找最低点以外,还可以通过梯度下降的方式来求解

整体的思路为

> 从某一个初始的参数开始(比如可以置0)
>
> 不断改变参数使损失函数减小
>
> 直到到达最小值(或足够接近)



其在空间中的表现为,寻找到最快下山的方向,并逐步递进,直到进入山谷之中

4.2 实现

在实现中,梯度下降会不断改变参数的值,公式为:

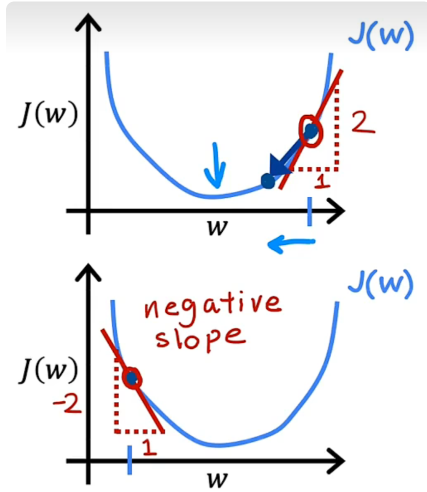
$$w = w - \alpha \frac{\delta}{\delta w} J(w, b)$$
$$b = b - \alpha \frac{\delta}{\delta b} J(w, b)$$

其中, w, b 为被改变的参数, α 为学习率,直观理解就是下降的步长, α 越大,那么单次前进的距离也就越远，而后面的求导部分决定了前进的方向。

因为两个参数在梯度下降时需要用到对方，所以二者应该分别计算出修改值之后，再进行修改，任一方先下降都会影响另一方的下降结果。

4.3 求导的意义

通过求导来决定参数的改变方向。



4.4 学习率

学习率对于梯度下降的影响十分巨大：

> 学习率过小：梯度下降过慢，影响效率
>
> 学习率过大：容易超出范围，导致来回震荡

学习率本身也应当随着训练的进程发生改变，通常为前大后小。

4.5 线性回归的梯度下降

$$w = w - \alpha \frac{\partial}{\partial w} J(w,b) \rightarrow \frac{1}{m}\sum_{i=1}^{m}\left(f_{w,b}\left(x^{(i)}\right) - y^{(i)}\right)x^{(i)}$$

$$b = b - \alpha \frac{\partial}{\partial b} J(w,b) \rightarrow \frac{1}{m}\sum_{i=1}^{m}\left(f_{w,b}\left(x^{(i)}\right) - y^{(i)}\right)$$

$$\}$$

线性回归的中梯度下降的具体展开。

4.6 运行梯度下降

lab1-3：通过实验将本节内容进行了实践