

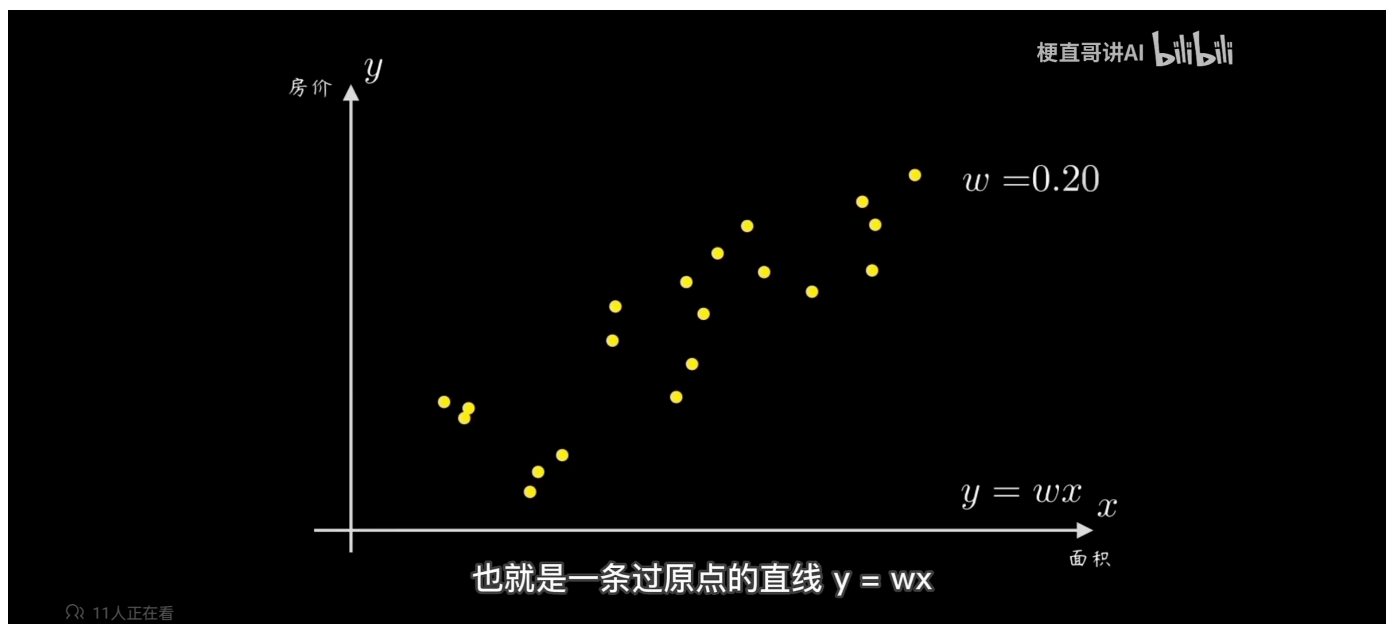
对于梯度下降算法的初步认识

金圣勋

- [参考视频]: 【【梯度下降】3D可视化讲解通俗易懂-哔哩哔哩】 <https://b23.tv/clgQGj5.html>

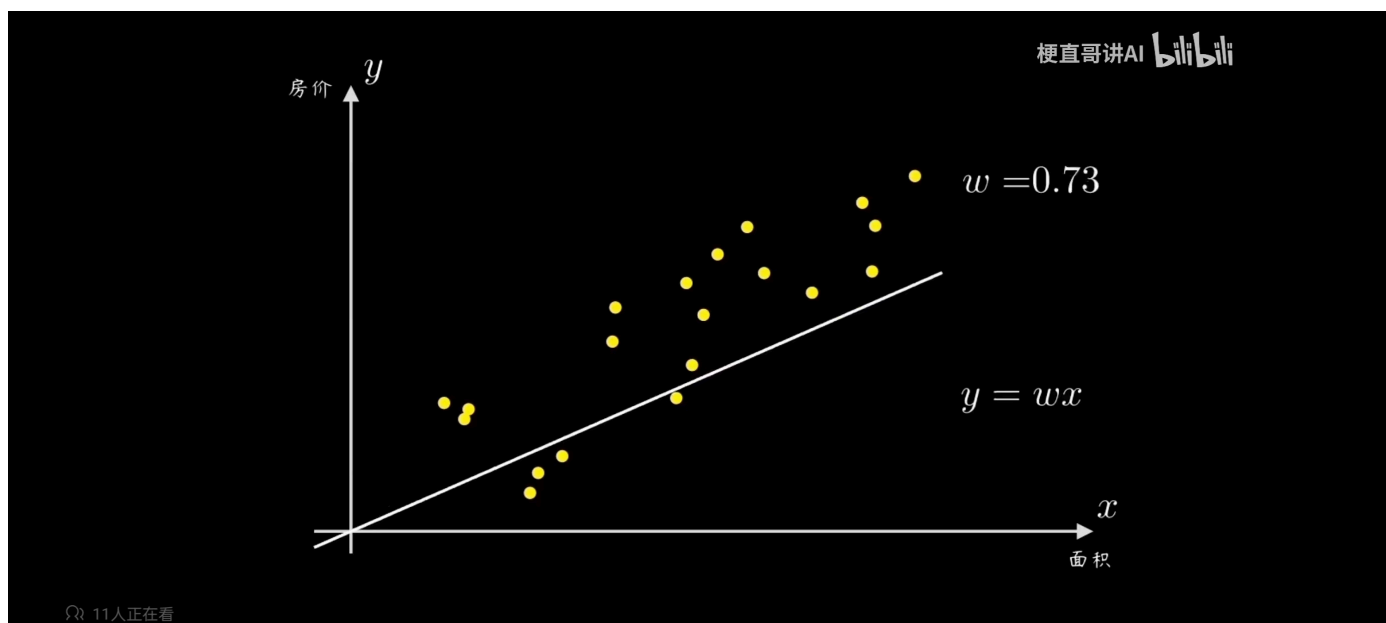
1. 问题的引入

- 如下一个线性的拟合场景



2. 预测函数

- 我们可以用 $y=mx$ 这样一个预测函数对数据点进行拟合,但难以量化拟合效果最好的 m ,所以要构建关于 m 的代价函数



3. 代价函数

- 代价函数的构建过程

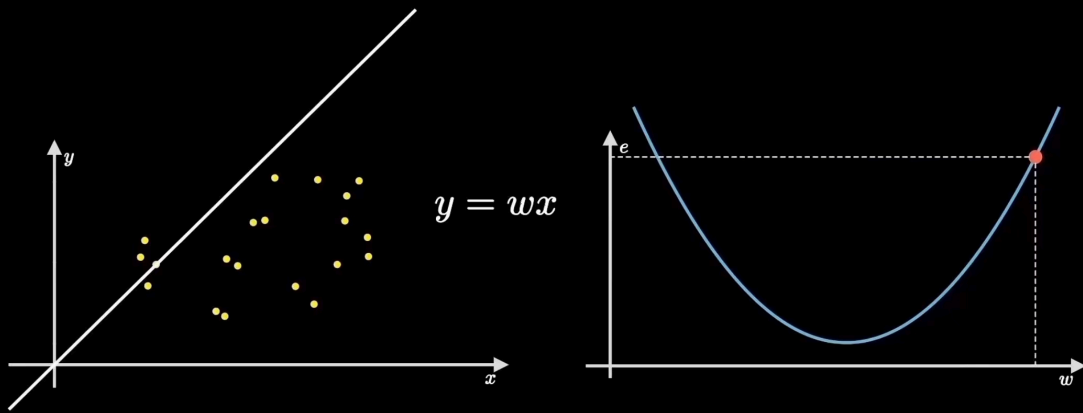
利用均方差，实现样本点拟合过程的转换

梗直哥讲AI bilibili

$$e = \frac{1}{n} \underbrace{(x_1^2 + \dots + x_n^2)}_a * w^2 + \underbrace{(-2 * x_1 * y_1 - \dots - 2 * x_n * y_n)}_b * w + \underbrace{(y_1^2 + \dots + y_n^2)}_c$$

我们可以大大简化最终的式子

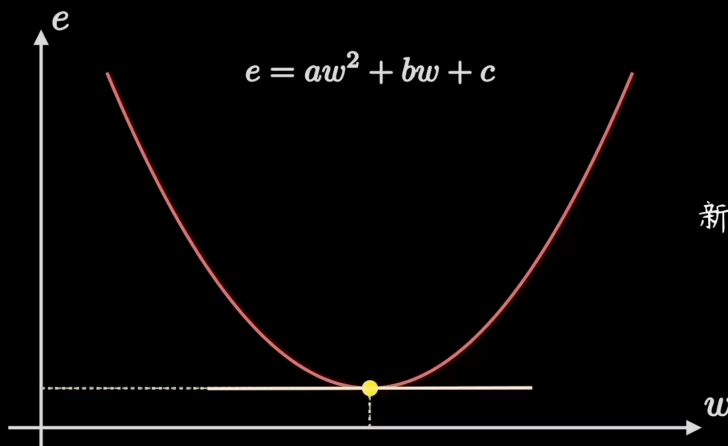
12人正在看



预测函数 代价函数
你和过程映射到了一个函数图像上

9人正在看

= 梯度下降法



新 w = 旧 w - 斜率 * 学习率

等于旧 w 减去斜率乘以学习率

7人正在看

- 广义上来说，图中斜率应为梯度
- 这种方式保证了 w 向着梯度减小的方向更新,而且能够在 w 越过梯度最小点时得以“矫正”回来
- 学习率的大小选择也是关键，太大来回震荡无法满足要求，太小无法满足要求，所以学习率一般是随迭代次数而衰减的

- 对于考核的思考

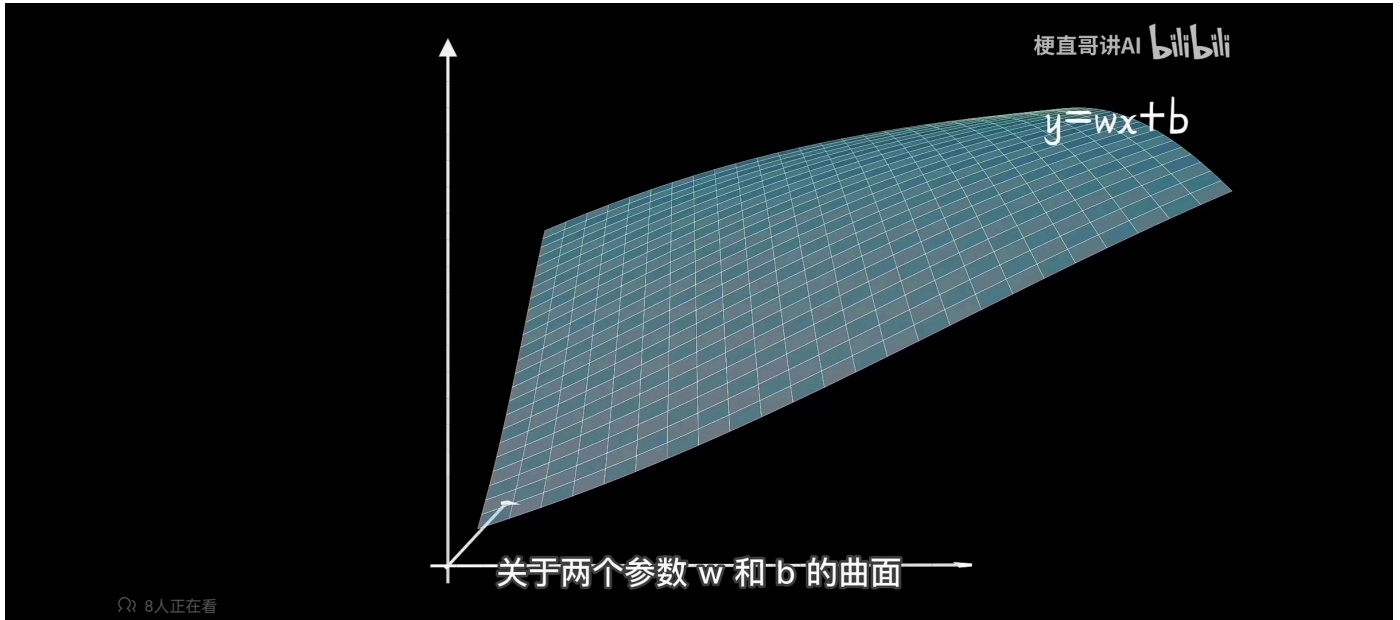
大概 $\|grad(Z_{actual}, \theta)\|$ 随着不断迭代而下降

$f(n) = |nZ_0 - \sum_n Z_{actual}|$ 是减函数，当 $\lim_{n \rightarrow \infty} f(n) = 0$

4.梯度下降算法的必要性

- 对开始的例子稍加改动

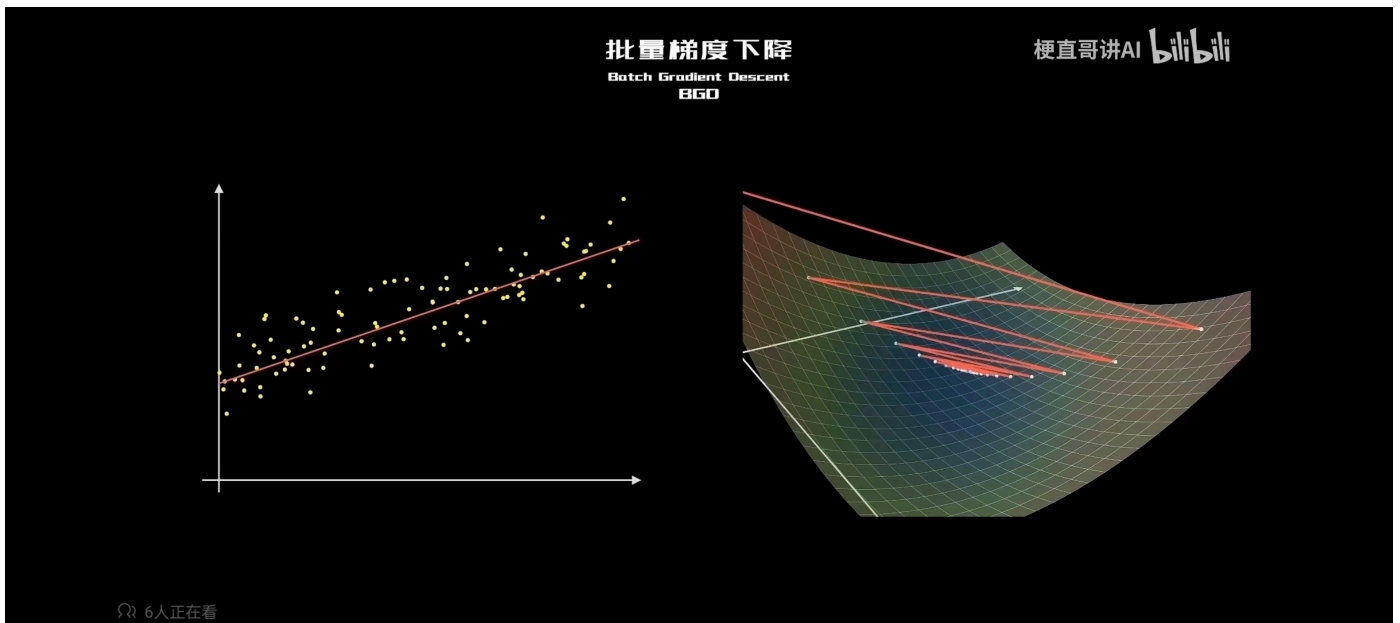
拟合直线为 $y=wx+b$,代价函数会变成关于 w 和 b 的二次曲面



- 或者 y 不是线性或一元函,代价函数都会更复杂,一般的求解难以求平均误差最小值

5.三种常用的梯度算法

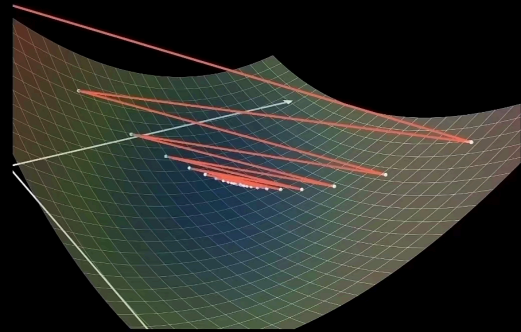
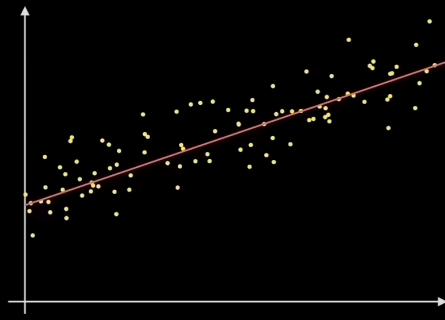
- BGD(每次用所有的数据,效果好但速度慢)



- SGD(每次只用一组数据,效果好但速度快)

批量梯度下降 Batch Gradient Descent BGD

梗直哥讲AI bilibili

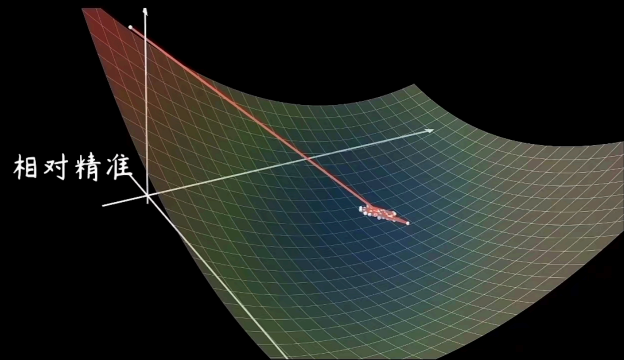
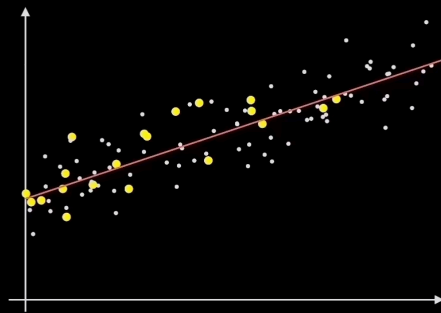


6人正在看

- MGD(每次用一部分数据,兼具前两者优点)

小批量梯度下降 Mini-Batch Gradient Descent MBGD

梗直哥讲AI bilibili



虽然没有二哥速度快

7人正在看