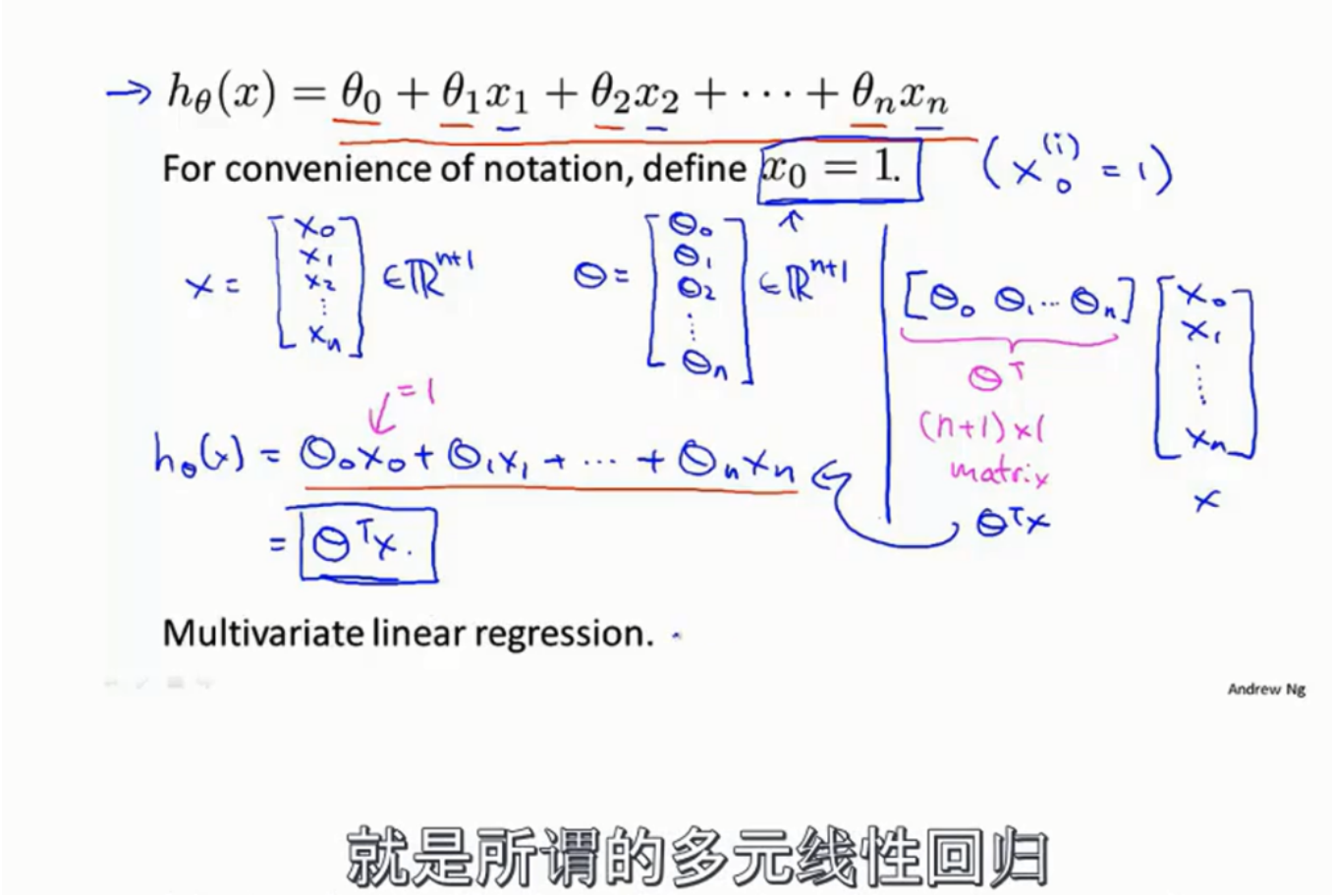
多变量线性回归_吴恩达_机器学习笔记

1、多变量线性回归的表示方法



让我们来理解一下上面的公式：

- 首先，这个公式上最终的结果是

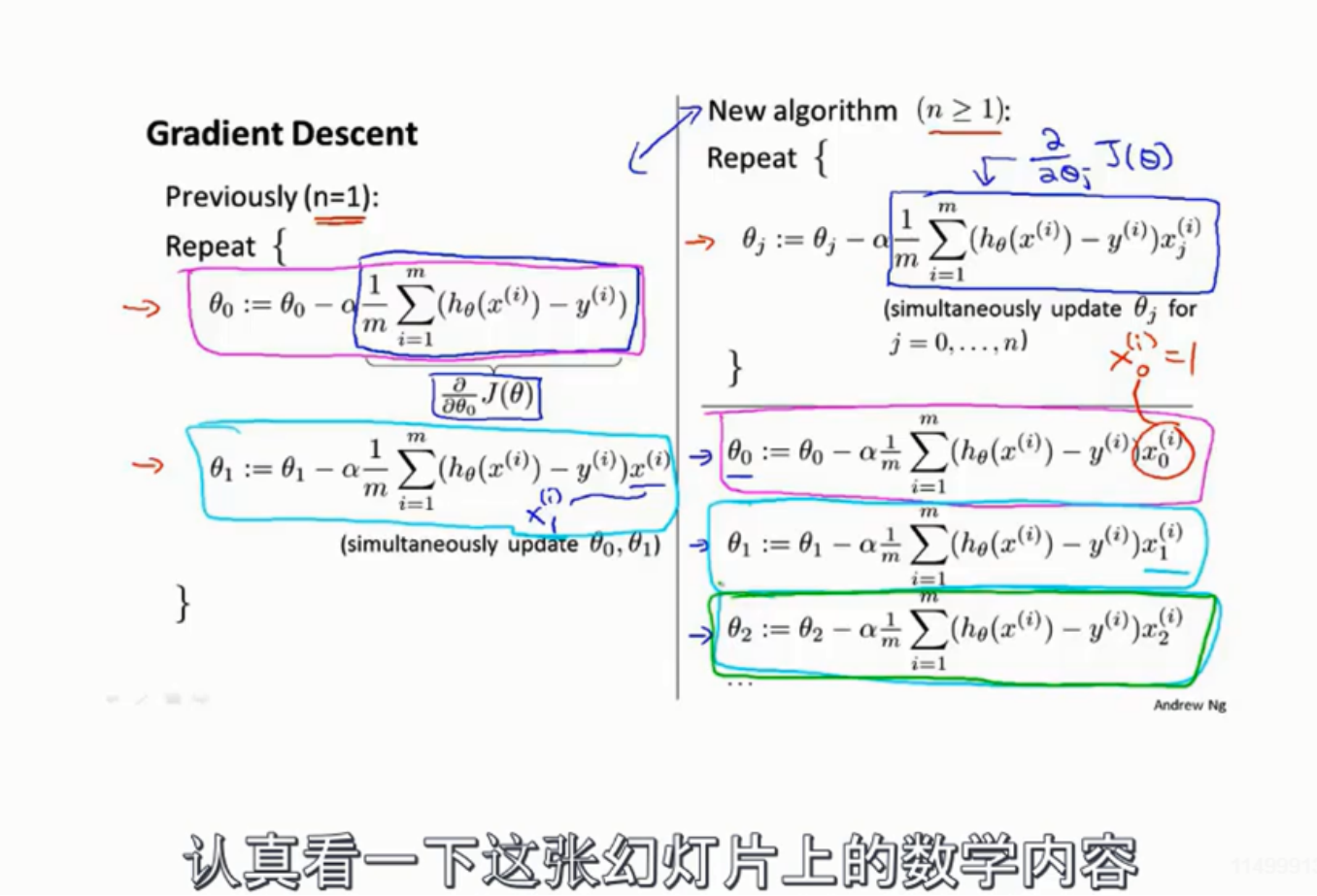
$$h\theta(x) = \theta^T x$$

那么这个公式实际上西塔是一直转置矩阵，也可以称为转置向量，而x实际上也是一个向量，那么这两个向量的乘法就实现了向量的内积。 最终就呈现了上面的结果。

- 其次，这个多变量的线性回归的假设函数，一般是可以很好的和业务进行解释的，例如在吴恩达的案例中，房子的价格与西塔1（面积）呈现正相关，与西塔2（房屋年龄）呈现负相关，很好理解，很直观，业务解释性很强。
- 另外，在西塔0这个地方，一般都是设x1=1，因此，这是一个R的（n+1）个维度的向量，西塔0是常数项。
- 至此，应该熟悉这种表示方法。
- R^n 是指n维向量，**其中n实际上是这个数据集的特征数量。**

2、多元梯度下降法

多元梯度下降法中，实际上和一元梯度下降法的表示方法类似，但是实际上需要理解二者的不同，先看如下图：



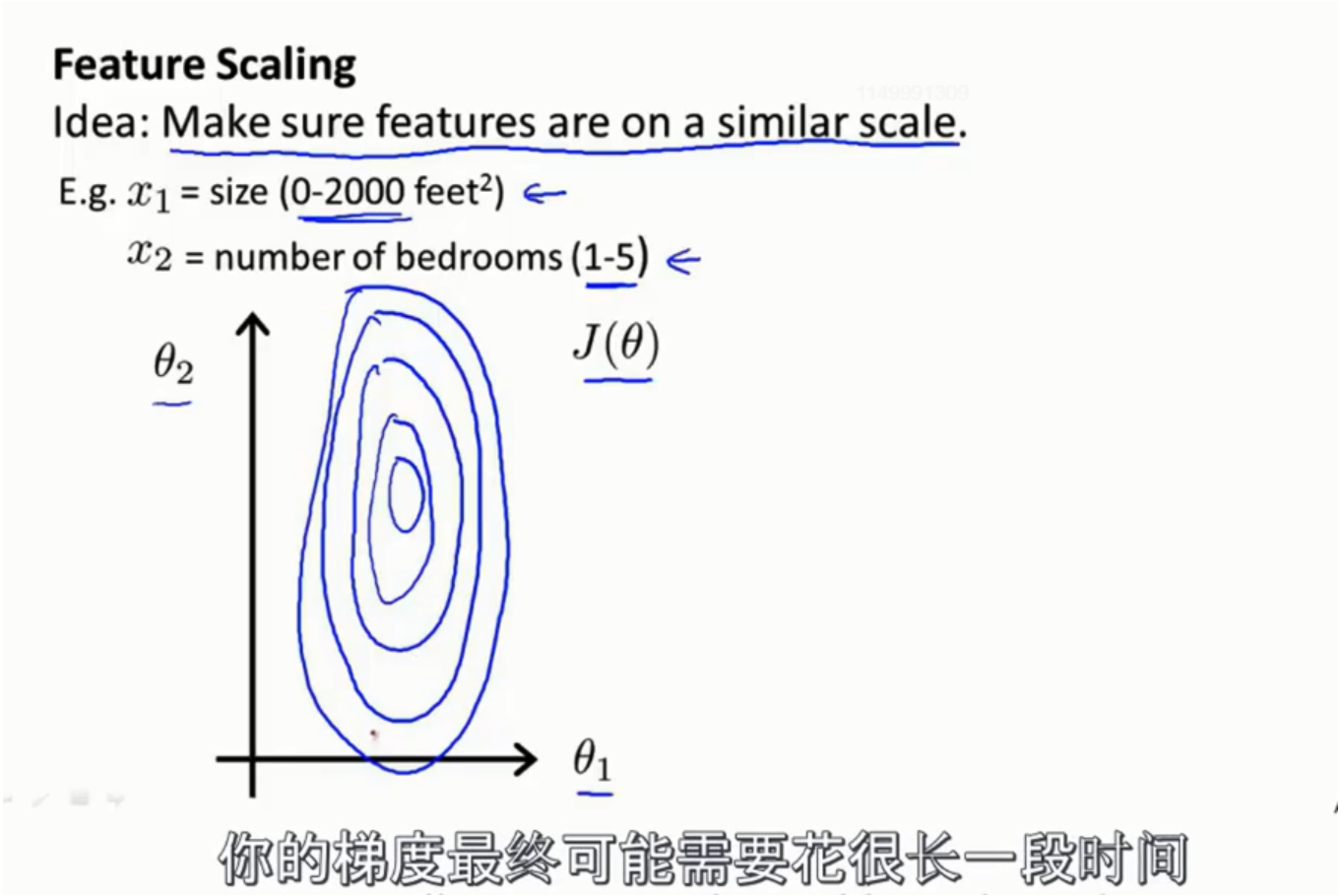
左边是一元梯度下降的更新规则，右边是多远梯度下降的更新规则，我们理解如下：

左边的梯度下降方法上，我们实际上只有一个参数就是θ0和θ1。 而右边的梯度下降方法上，实际上是有多个参数，从θ0、θ1、θ2、θ3……θn。 当出现这么多的参数的时候，应该怎么去求偏导数呢，实际上原理上是一致的，有多少个参数，我们在每一轮迭代的时候，都同时更新如此多的参数的新一轮的值，然后看梯度下降方法是否导致了代价函数取到了比迭代前更小的值，如果是，那么继续更新，如果和上一轮相等或者非常接近一个很小的值，比如1^-10次方这么小的值，那么就停止更新，或者自身设置迭代次数。

3、多元梯度下降法1 — 特征缩放

本节主要学习多元梯度下降法的一些实用方法，其中之一就是特征缩放。为何要进行特征缩放，具体看下面的案例：

- 特征缩放以后，梯度下降算法能够更快的收敛。案例如下：



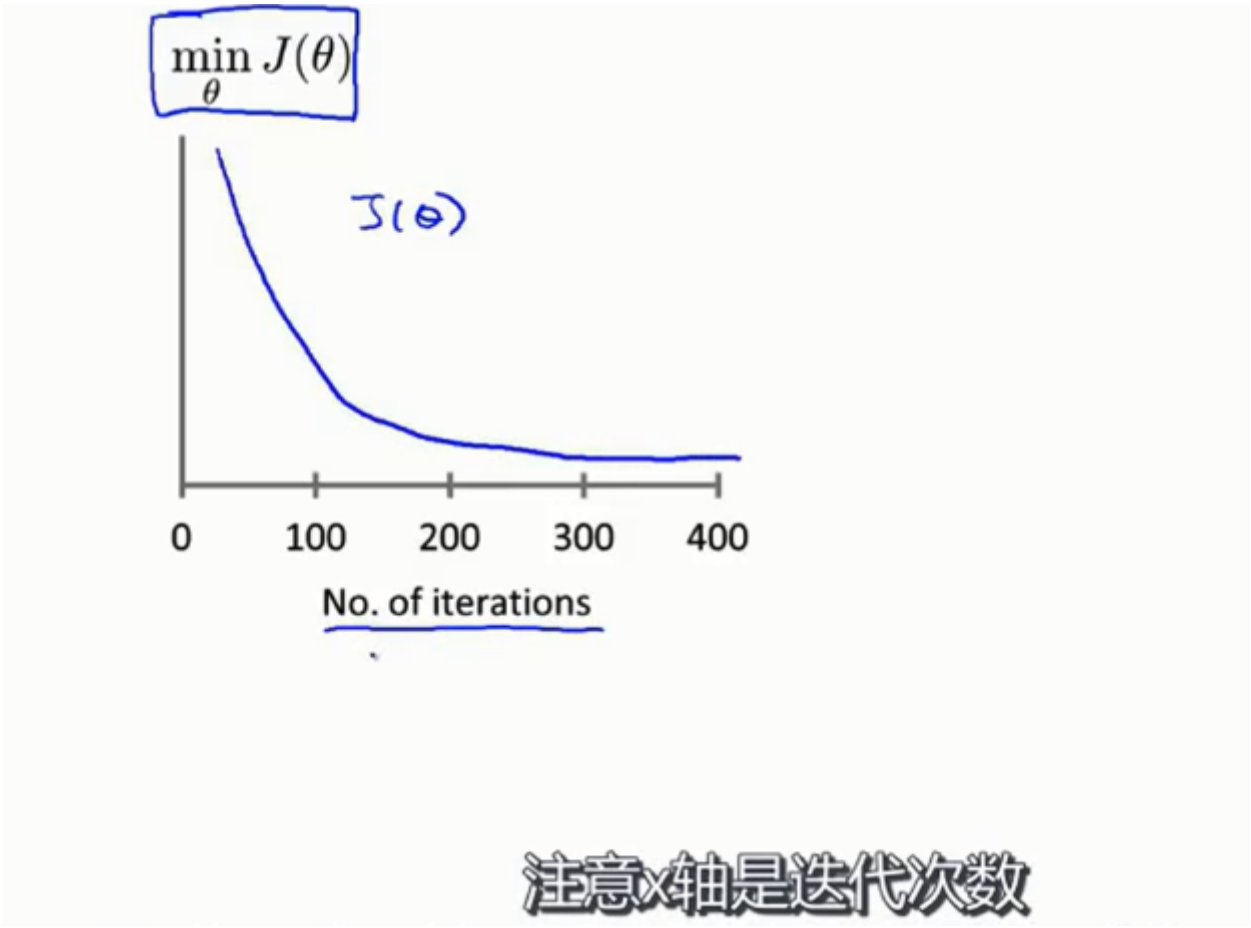
在这个案例中，假设只有θ1和θ2两个参数，假如θ2的取值在0-2000范围，而θ1的取值在1-5的范围，那么可以想象一下，这个等高线图，实际上比如图所示的范围将更加的狭长。这样一个数据中，梯度下降算法将需要非常多次的迭代才能够下降到极值点，同时，由于过于狭长，可以想象，这给设置学习率将会带来更加深刻的麻烦，学习率将更加难以设置并不能保证能够下降到极值点，可能发生震荡。

因此，在梯度下降算法中，我们需要优先对数据进行缩放在0-1这个区间内或者其他的-3到3的区间均可，其实公式比较简单，具体参照特征缩放或者均值归一化的公式 （a - 某特征的平均值）/(特征的最大值-特征的最小值)。分母也可以是数据的方差之类的。这里不具体赘述。

这里需要思考一点，就是什么样的特征之间的差距是可以接受的呢，按照视频里的经验，实际上，不同的特征的取值范围，相互之间足够接近即可，在具体的建模过程中，相差不太大就行，同时处理特征能够保证离群值不成为影响boosting之类的组合模型算法的关键因素，确保模型可用性。

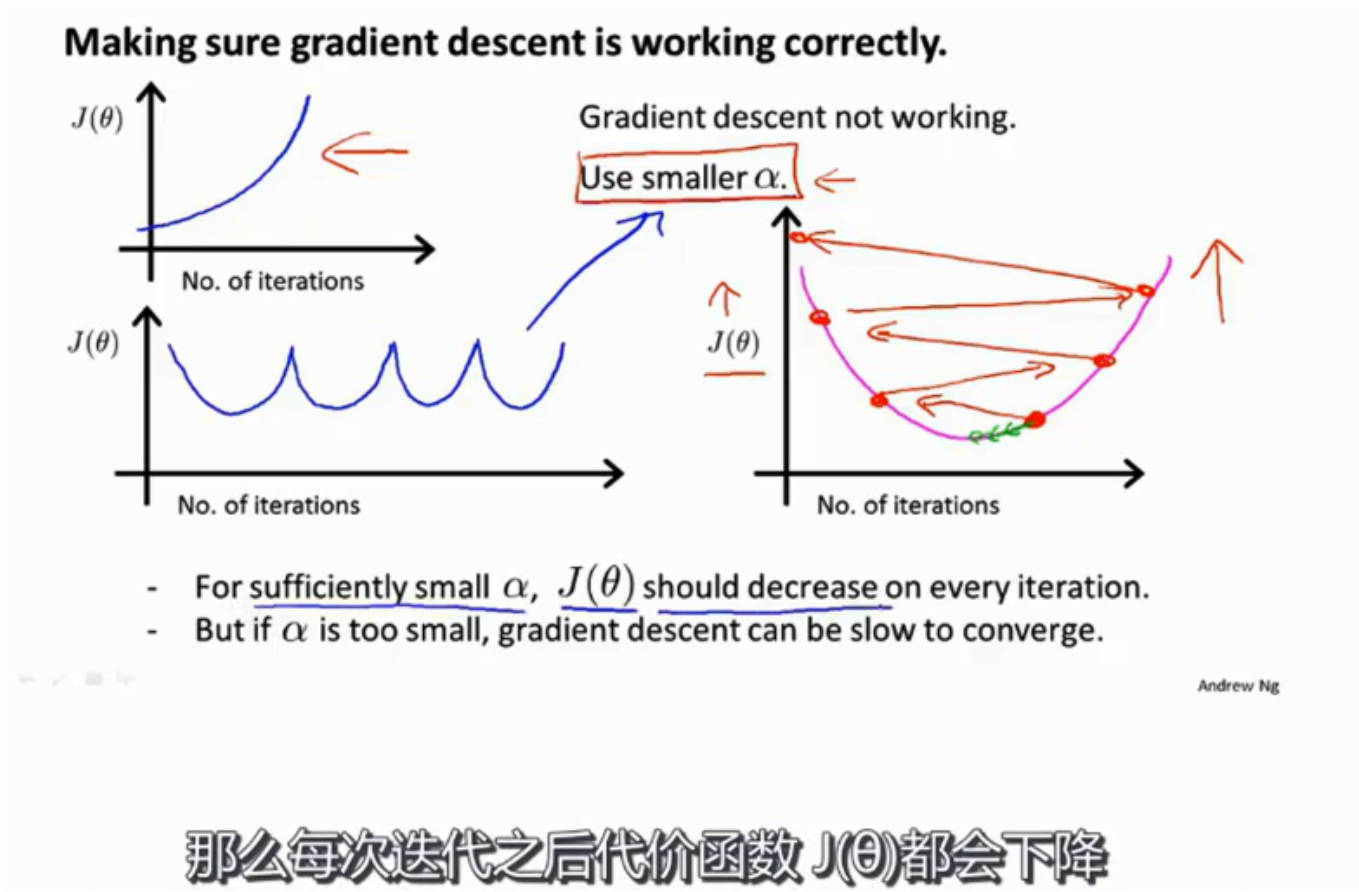
4、多元梯度下降法2 — 学习率

如果确保设置的学习率是在正常工作确保梯度下降算法的迭代能够持续的优化代价函数的极值呢？常用的办法就是去观察迭代次数X和代价函数之间的持续变化的可视化图像。具体示例如下：



一般而言，通过迭代次数300次以后，我们需要计算的是θ值，然后通过θ值去计算代价函数的值是否在持续的变小。

当然，也可以看本次迭代后的结果与上次迭代后的结果的差值，是否已经小于某一个很小的值，比如小于0.0001。入股小于，那么也就停止迭代，当然保险的方法，还是直接可视化观测。



反正学习率足够小的情况下，梯度下降一定会收敛到极值。

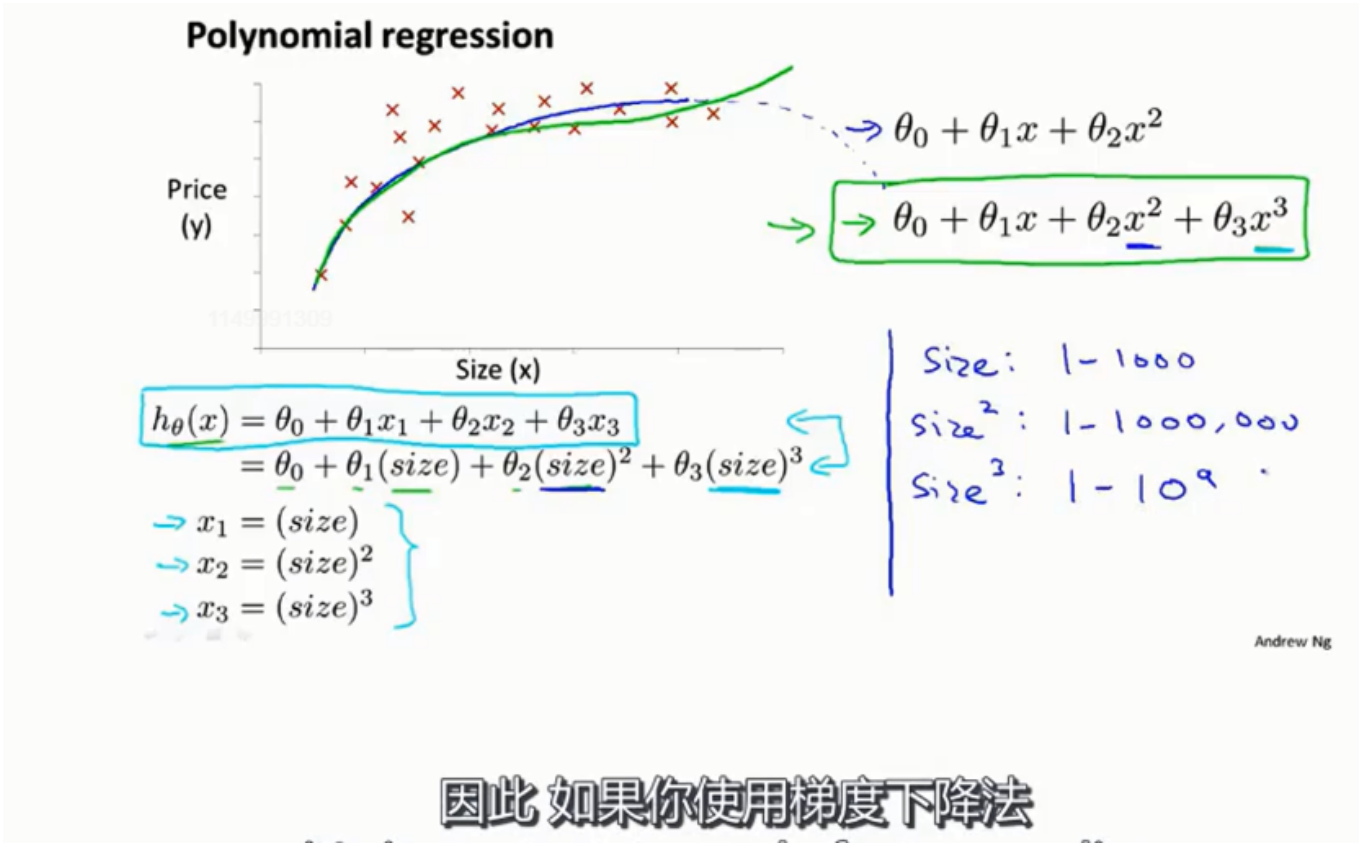
但是如果学习率过大，震荡的结果，如上图中左边，最后的代价函数，始终在一个值的区间反复，但是绝对这个值离我们想要优化到的极值点是比较远的，不符合目标的。

学习率的取值一般是0.1，0.01等，每次学习率调参不要过大，可以多次尝试。

5、特征和多项式回归

多项式回归，是指模型有时候需要拟合的数据中，需要使用到特征与特征相乘的结果才能够更好的拟合数据，比如一个预测房价的案例中，特征中有x1是房屋的长，x2是房屋的宽，那么实际上这时候做一个数据处理，产生一个x1*x2的面积数据，可能会更好的拟合数据，那么实际上，这个面积，就是多项式。

更好的案例如下图：



上图中，我们的数据集如果用一个简单的一元线性函数来拟合数据的话，那么实际上这个数据是不能很好的拟合的，可能平方误差会比较大。

- 如果我们尝试在拟合的数据上加上平方项呢？可能结果仍然不尽理想，因为平方项的二次函数的拟合可能存在下降的情况。
- 但是拟合的数据上再加上一个立方项呢？嗯，结果可能会更好一些，那么这个就是多项式回归的一个很直观的案例。

那么，这里有一个问题？什么时候应该用多项式回归来拟合数据呢？ 吴恩达并没有给出很明确的答案，但是，对业务的熟悉程度能够解决这个问题。这个问题的思考其实应该更加深入。待后续搜索到答案再补充。

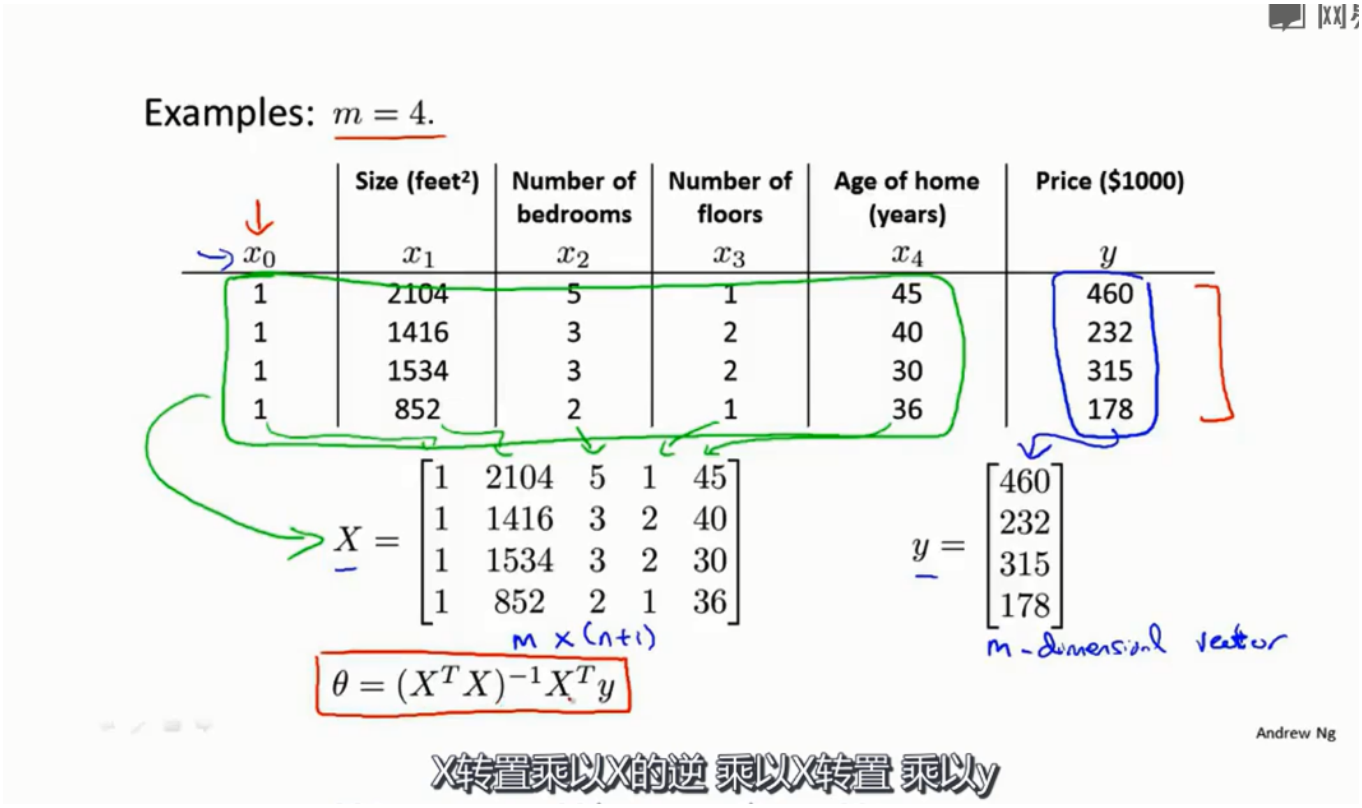
上面问题的答案，应该说，特征选择的时候，是有很多方案可以实施的，比如正则化，比如算法自带的特征选择。但是创造新的特征需要根据业务环境考虑。

还是看上面那个图，那么如果size是房屋的面积的话，实际上，第二步就是房屋面积的平方，第三个就是房屋面积的立方特征。

6、正规方程求解最优化问题（区别梯度下降法等迭代优化）

正规方程实际上是面临小类型数据所能够实施的一种求解最优化问题的方案，由于整个公式中涉及到求矩阵的逆矩阵，或者近似逆矩阵。如果特征向量非常的大，那么这个逆矩阵的计算将会产生非常大的时间开销，这时候，使用梯度下降方法，恐怕是更能够快速求解的一种选择。

正规方程的整体表达式如下：



通过这个算法，那么就可以求导最小化的θ参数向量。

使用正规方程计算参数向量的时候，是不需要进行特征缩放的，仍然可以求解。

正规方程一般在1万条数据以内的数据，建议使用，如果超过这个量，建议使用梯度下降、拟牛顿法等优化类算法进行极值求解。

7、正规方程在矩阵不可逆下的解决方法

当矩阵不可逆的情况发生时，那么可以通过奇异值分解来求矩阵的逆矩阵。奇异值分解通常被成为SVD，具体的学习文档参见： https://blog.csdn.net/zhongkejingwang/article/details/43053513

如果特征与特征之间存在某种直接的函数映射关系，那么这个特征矩阵将无法求得逆矩阵。

另外一种情况，就是特征数量，远远大于样本量的时候，特征矩阵是不可逆的，比如10个样本数据，但是有100个特征的极端情况。想象一下，要拟合这样的一个数据，需要找到100维或者101维的参数向量，来拟合数据，这个实现起来，因为数据量太小的问题，实际上几乎是很难的。

这里也给到了矩阵不可逆的常规解决办法：

- 从业务场景入手，删除冗余的重复的特征。
- 使用正则化等特征选择方法，做特征选择，去掉不需要的特征然后再进行建模。

8、本章总结和编程技巧

虽然，大多数程序语言，都实现了矩阵的奇异值分解函数，但是，仍然有必要去学习和了解奇异值分解的原理。