### 4.6 正规方程

参考视频: 4 - 6 - Normal Equation (16 min).mkv

到目前为止，我们都在使用梯度下降算法，但是对于某些线性回归问题，正规方程方法是更好的解决方案。如：

图片包含 物体

描述已自动生成

正规方程是通过求解下面的方程来找出使得代价函数最小的参数的： 。 假设我们的训练集特征矩阵为 （包含了 ）并且我们的训练集结果为向量 ，则利用正规方程解出向量

上标**T**代表矩阵转置，上标-1 代表矩阵的逆。设矩阵，则：

以下表示数据为例：

图片包含 文字

描述已自动生成

即：

图片包含 纵横字谜, 墙壁, 障子

描述已自动生成

运用正规方程方法求解参数：

图片包含 设备, 时钟

描述已自动生成

在 **Octave** 中，正规方程写作：

pinv(X'\*X)\*X'\*y

注：对于那些不可逆的矩阵（通常是因为特征之间不独立，如同时包含英尺为单位的尺寸和米为单位的尺寸两个特征，也有可能是特征数量大于训练集的数量），正规方程方法是不能用的。

梯度下降与正规方程的比较：

|  |  |
| --- | --- |
| 梯度下降 | 正规方程 |
| 需要选择学习率 | 不需要 |
| 需要多次迭代 | 一次运算得出 |
| 当特征数量大时也能较好适用 | 需要计算 如果特征数量较大则运算代价大，因为矩阵逆的计算时间复杂度为，通常来说当小于10000 时还是可以接受的 |
| 适用于各种类型的模型 | 只适用于线性模型，不适合逻辑回归模型等其他模型 |

总结一下，只要特征变量的数目并不大，标准方程是一个很好的计算参数的替代方法。具体地说，只要特征变量数量小于一万，我通常使用标准方程法，而不使用梯度下降法。

正规方程的**python**实现：

import numpy as np  
def normalEqn(X, y):  
 theta = np.linalg.inv(X.T@X)@X.T@y #X.T@X等价于X.T.dot(X)  
 return theta

### 4.7 正规方程及不可逆性（选修）

参考视频: 4 - 7 - Normal Equation Noninvertibility (Optional) (6 min).mkv

在这段视频中谈谈正规方程 ( **normal equation** )，以及它们的不可逆性。 由于这是一种较为深入的概念，并且总有人问我有关这方面的问题，因此，我想在这里来讨论它，由于概念较为深入，所以对这段可选材料大家放轻松吧，也许你可能会深入地探索下去，并且会觉得理解以后会非常有用。但即使你没有理解正规方程和线性回归的关系，也没有关系。

我们要讲的问题如下：

备注：本节最后我把推导过程写下。

有些同学曾经问过我，当计算 =inv(X'X ) X'y ，那对于矩阵的结果是不可逆的情况咋办呢?

如果你懂一点线性代数的知识，你或许会知道，有些矩阵可逆，而有些矩阵不可逆。我们称那些不可逆矩阵为奇异或退化矩阵。

问题的重点在于的不可逆的问题很少发生，在**Octave**里，如果你用它来实现的计算，你将会得到一个正常的解。在**Octave**里，有两个函数可以求解矩阵的逆，一个被称为pinv()，另一个是inv()，这两者之间的差异是些许计算过程上的，一个是所谓的伪逆，另一个被称为逆。使用pinv() 函数可以展现数学上的过程，这将计算出的值，即便矩阵是不可逆的。

在pinv() 和 inv() 之间，又有哪些具体区别呢 ?

其中inv() 引入了先进的数值计算的概念。例如，在预测住房价格时，如果是以英尺为尺寸规格计算的房子，是以平方米为尺寸规格计算的房子，同时，你也知道1米等于3.28英尺 ( 四舍五入到两位小数 )，这样，你的这两个特征值将始终满足约束：。

实际上，你可以用这样的一个线性方程，来展示那两个相关联的特征值，矩阵将是不可逆的。

第二个原因是，在你想用大量的特征值，尝试实践你的学习算法的时候，可能会导致矩阵的结果是不可逆的。 具体地说，在小于或等于n的时候，例如，有等于10个的训练样本也有等于100的特征数量。要找到适合的 维参数矢量，这将会变成一个101维的矢量，尝试从10个训练样本中找到满足101个参数的值，这工作可能会让你花上一阵子时间，但这并不总是一个好主意。因为，正如我们所看到你只有10个样本，以适应这100或101个参数，数据还是有些少。

稍后我们将看到，如何使用小数据样本以得到这100或101个参数，通常，我们会使用一种叫做正则化的线性代数方法，通过删除某些特征或者是使用某些技术，来解决当比小的时候的问题。即使你有一个相对较小的训练集，也可使用很多的特征来找到很多合适的参数。 总之当你发现的矩阵的结果是奇异矩阵，或者找到的其它矩阵是不可逆的，我会建议你这么做。

首先，看特征值里是否有一些多余的特征，像这些和是线性相关的，互为线性函数。同时，当有一些多余的特征时，可以删除这两个重复特征里的其中一个，无须两个特征同时保留，将解决不可逆性的问题。因此，首先应该通过观察所有特征检查是否有多余的特征，如果有多余的就删除掉，直到他们不再是多余的为止，如果特征数量实在太多，我会删除些 用较少的特征来反映尽可能多内容，否则我会考虑使用正规化方法。 如果矩阵是不可逆的，（通常来说，不会出现这种情况），如果在**Octave**里，可以用伪逆函数pinv() 来实现。这种使用不同的线性代数库的方法被称为伪逆。即使的结果是不可逆的，但算法执行的流程是正确的。总之，出现不可逆矩阵的情况极少发生，所以在大多数实现线性回归中，出现不可逆的问题不应该过多的关注是不可逆的。

**增加内容：**

的推导过程：

其中：

将向量表达形式转为矩阵表达形式，则有 ，其中为行列的矩阵（为样本个数，为特征个数），为行1列的矩阵，为行1列的矩阵，对进行如下变换:

接下来对偏导，需要用到以下几个矩阵的求导法则:

所以有:

令,

则有