# 1.数学基础部分

 线性代数：看过的大部分ml算法都是用linear algebra的terminology进行描述的。有线性代数基础，在没有看懂paper的情况下，还是有可能根据伪代码实现文中算法的。

 概率与统计：有助于理解最大似然估计(MLE)，最大后验概率(MAP)，正则化等ML中很重要的概念。所谓的最大似然估计，大多数情况下，就是counting。

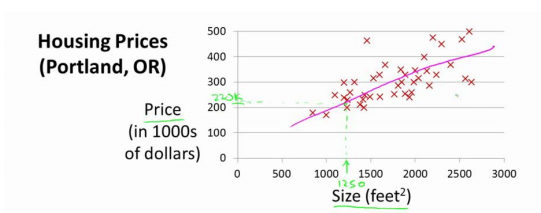
 数值优化(进阶)：注意不是凸优化，就是数值优化。学了这个之后，你就会了解，所谓的(stochastic/online/mini-batch) gradient descent方法，其实就是渣。梯度下降其实就是steepest descent的简化版本，阉割了line search。光是line search，就值得花几节课的时间大讲特讲。更别提像conjugate gradient descent这样高级的一阶算法，以及牛顿法(Newton's method)/拟牛顿法(Quasi-Newton/L-BFGS/OWL-QN)等逆天的二阶算法。

 凸优化：很重要，但是没有系统的学习过，在此不误人子弟。

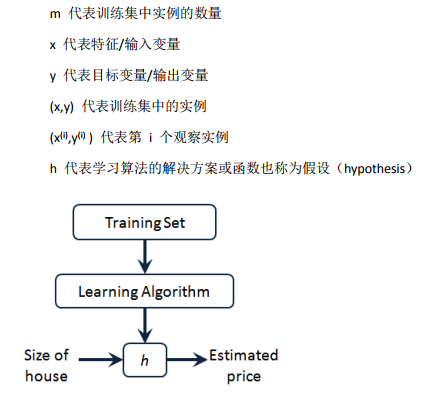
# 二、单变量线性回归

## （一）模型表示

以房屋销售价格为例，在此时，房子大小是唯一变量，即房子的size是决定房屋价格的唯一决定性因素，并且我们已经收集到足够的实例，可以将这些实例化成如下图：



这是个非常典型的监督性学习，因为对于每个自变量size我们都能找到唯一的price来唯一对应，这个单变量线性规划的模型如下图所示：

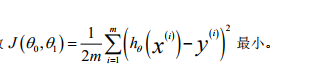


我们最后是需要得到一个h这个h在这个例子里面是一条直线，那么我们假设他是这个样子的：

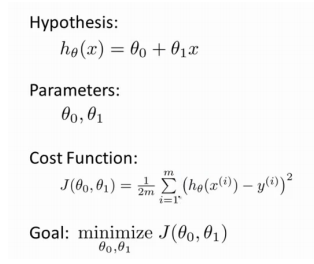
有了假设值，那么我们最终就是要求出两个变量值，使得他能够最好的拟合我们所给的实例，接下来引入代价函数

## （二）代价函数

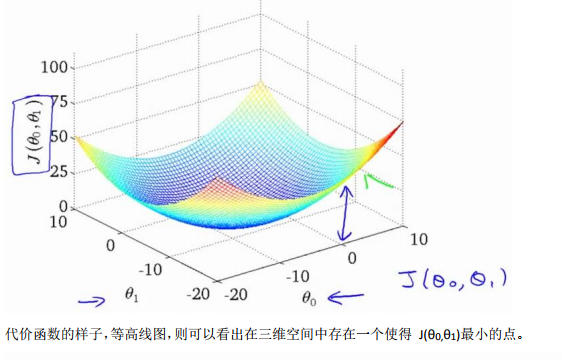
我叫它最小损失函数，就是说每个点到这条拟合线的距离之和最短，



其中m代表总共有m个点，数值点，h(x)代表预测值，y代表实际值。



？似乎是把问题映射到了更高的维度来解决问题？

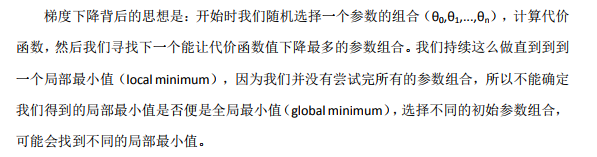


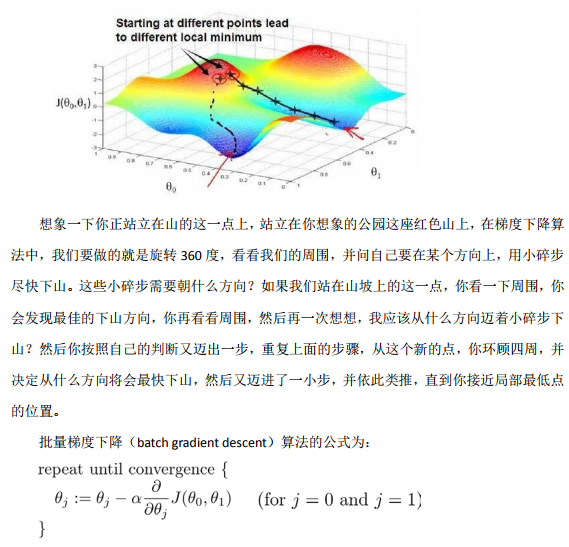
## （三）如何让程序自己测算出来这个未知数的数值

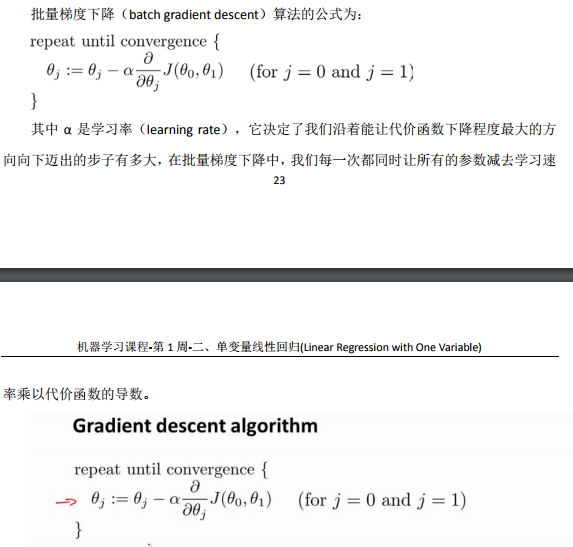
### 1.梯度下降算法

梯度下降算法：可以用来求函数最小值的算法，我们可以使用梯度下降算法来求解两个未知变量和权重。

关于梯度下降算法比较详尽的解释：



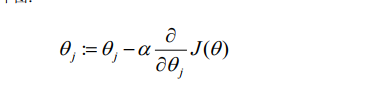




在梯度下降算法中默认的是，同步更新，同时更新。

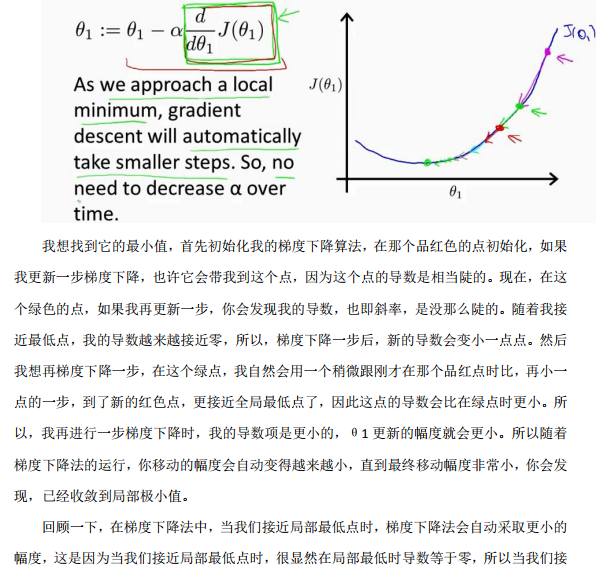
### （四）梯度下降算法中的数学原理

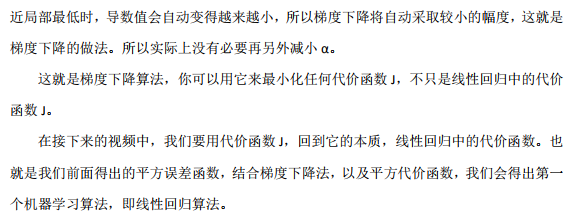
**1、最主要的公式如下：**



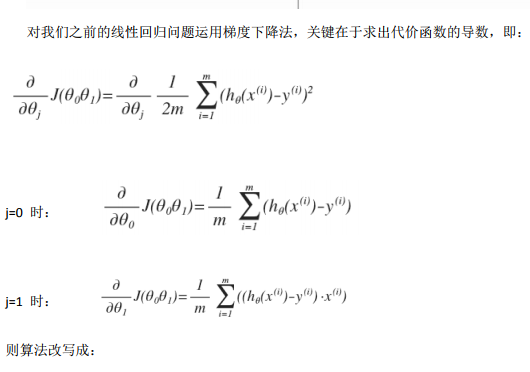
这个算法可以说是第一个接触到的很美的算法了。

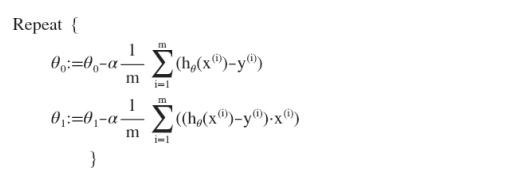
他的原理是这样的，





**2、梯度下降算法在最小损失函数中的应用**





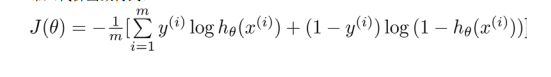
强调的就是同时更新同步更新，批量更新。

# 三、逻辑回归

适用于标签y取值零散的情况

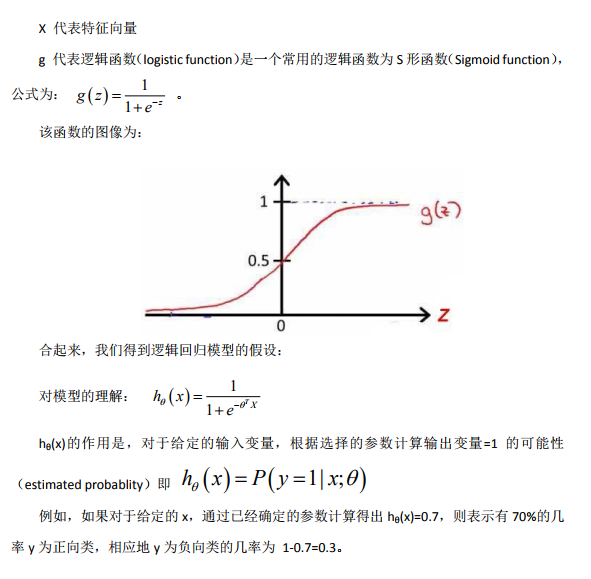
## 分类问题

**1.为什么逻辑回归的损失函数变成如下形式？：**

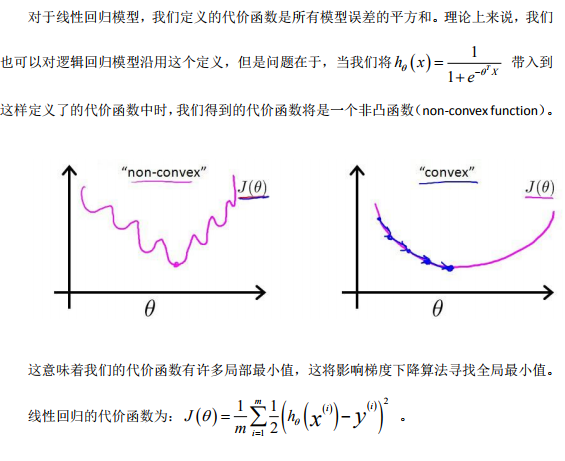


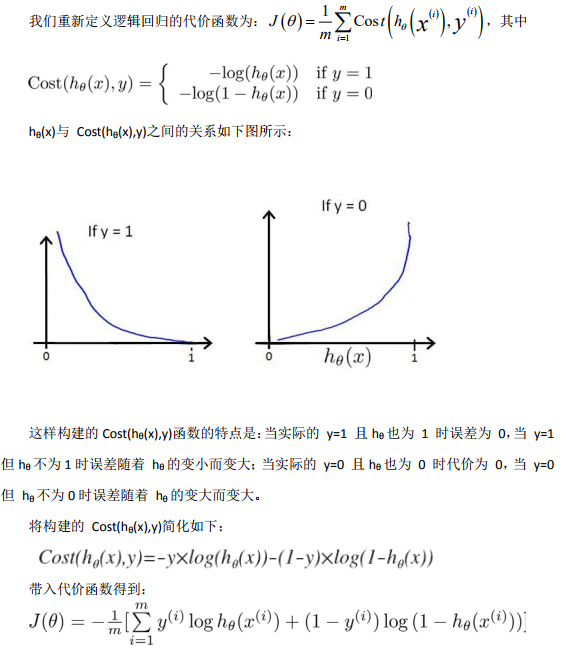
逻辑回归与线性回归的区别在于，线性回归作用于连续数值但是逻辑回归作用域离散数值；

**2、逻辑回归需要控制在0-1之间，这种情况下0.5比较适合作为分界条件，并且，sigmoid函数具有很好的性质便于我们计算**

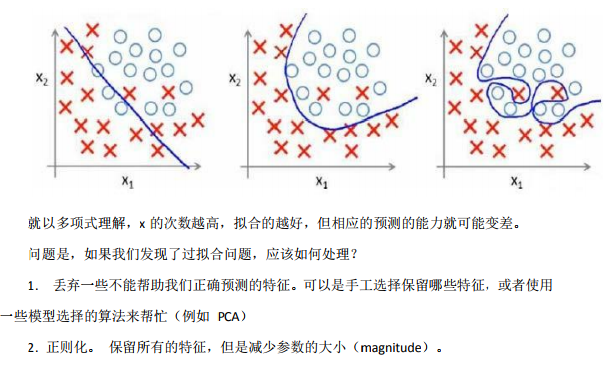


**3、为什么不要用线性回归模型的损失函数？**





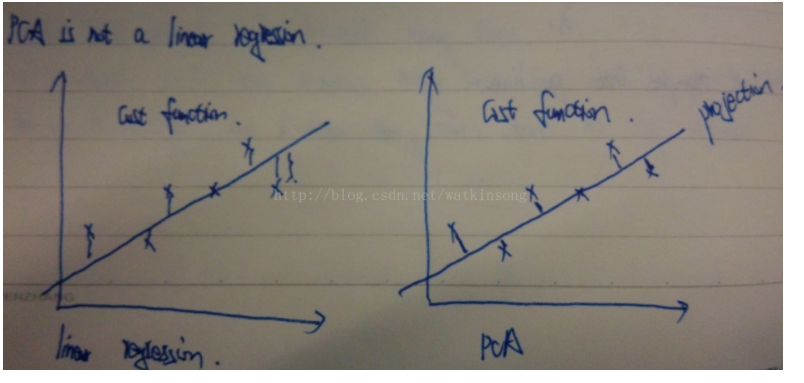
**4.过拟合问题的存在和解决办法？**



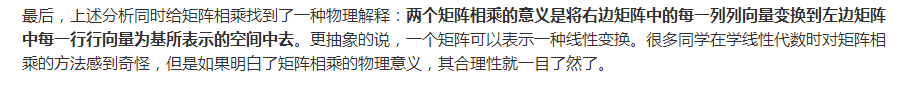
1. PCA：（http://www.360doc.com/content/13/1124/02/9482\_331688889.shtml）

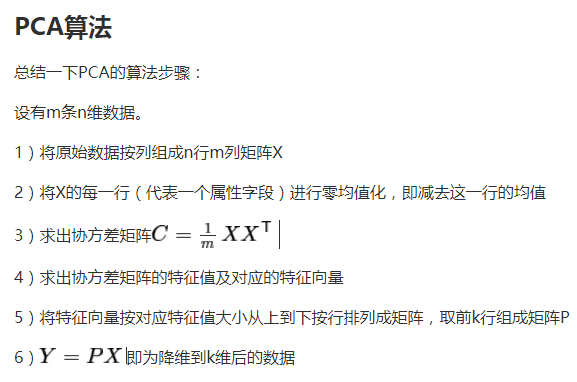
需要学习**矩阵分析**的知识，所谓的PCA其实就是一个多次幂降幂的过程，例如，三维降成二维，二维降成一维数据，二维的坐标可以用一条横线来拟合，三维的可以用一个平面来拟合，但这个过程要注意计算损失值，就是拟合差值，注意这个差值和linear regression是不一样的，

这个公式：



**矩阵相乘的意义：？**





CV样本或者测试样本

对称矩阵，对角矩阵

协方差矩阵

实对称矩阵：特征向量，特征值，协方差矩阵特征向量的标准化