# 一、Logistics回归

**1、基本原理**

Logistic Regression和Linear Regression的原理是相似的，可以简单的描述为这样的过程：

（1）找一个合适的预测函数（Andrew Ng的公开课中称为hypothesis），一般表示为*h*函数，该函数就是我们需要找的分类函数，它用来预测输入数据的判断结果。这个过程时非常关键的，需要对数据有一定的了解或分析，知道或者猜测预测函数的“大概”形式，比如是线性函数还是非线性函数。

（2）构造一个Cost函数（损失函数），该函数表示预测的输出*(h)*与训练数据类别*(y)*之间的偏差，可以是二者之间的差*(h-y)*或者是其他的形式。综合考虑所有训练数据的“损失”，将Cost求和或者求平均，记为函数，表示所有训练数据预测值与实际类别的偏差。

（3）显然，函数的值越小表示预测函数越准确（即*h*函数越准确），所以这一步需要做的是找到函数的最小值。找函数的最小值有不同的方法，Logistic Regression实现时有的是梯度下降法（Gradient Descent）。

**2、具体原理**

**2.1构造预测函数**

Logistic Regression虽然名字里带“回归”，但是它实际上是一种分类方法，用于两分类问题（即输出只有两种）。需要先找到一个预测函数*(h)*，显然，该函数的输出必须是两个值（分别代表两个类别），所以利用了Logistic函数（或称为Sigmoid函数），函数形式为：

对应的函数图像是一个取值在0和1之间的S型曲线（图1）。

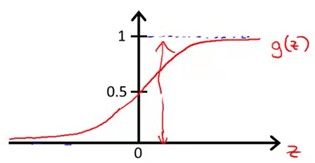


图 1

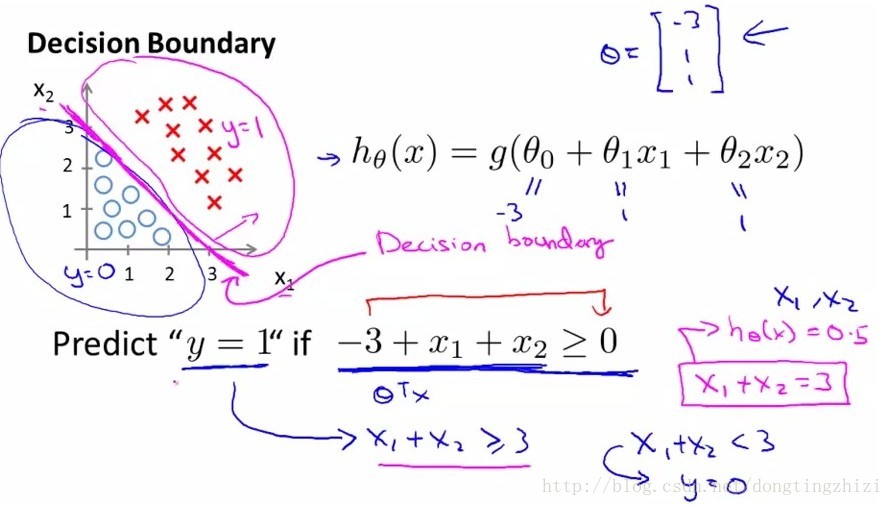


图 2

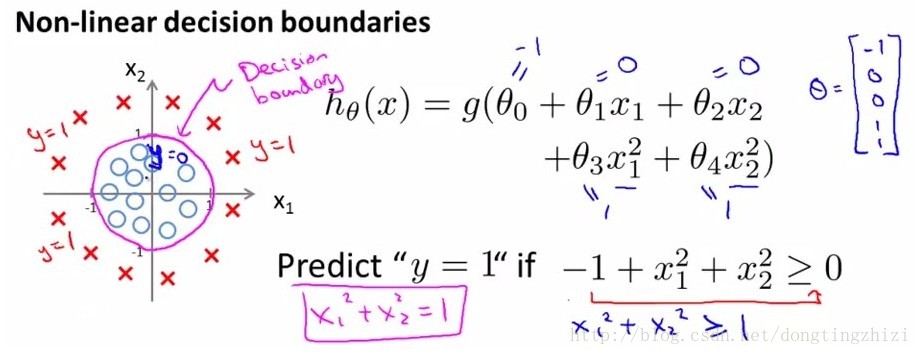


图 3

接下来需要确定数据划分的边界类型，对于图2和图3中的两种数据分布，显然图2需要一个线性的边界，而图3需要一个非线性的边界。接下来我们只讨论线性边界的情况。

对于线性边界的情况，边界形式如下：

构造预测函数为：

函数的值有特殊的含义，它表示结果取1的概率，因此对于输入x分类结果为类别1和类别0的概率分别为：

**2.2构造Cost函数**

Cost函数及函数如式（5）和（6）。

这里的Cost函数和函数是基于最大似然估计推导得到的。下面详细说明推导的过程。（4）式综合起来可以写成：

取似然函数为：

对数似然函数为：

最大似然估计就是要求得使取最大值时的，其实这里可以使用梯度上升法求解，求得的就是要求的最佳参数。但是，在Andrew Ng的课程中将取为（6）式，即：

因为乘了一个负的系数-1/m，所以取最小值时的为要求的最佳参数。

**2.3梯度下降法求*J(θ)*最小值**

求的最小值可以使用梯度下降法，根据梯度下降法可得的更新过程：

式中为学习步长，下面来求偏导：

上式求解过程中用到如下的公式：

因此，（11）式的更新过程可以写成：

因为式中本来为一常量，所以1/m一般将省略，所以最终的更新过程为：

另外，补充一下，2.2节中提到求得取最大值时的也是一样的，用梯度上升法求（9）式的最大值，可得：

**2.4梯度下降过程向量化**

约定训练数据的矩阵形式如下，x的每一行为一条训练样本，而每一列为不同的特称取值：

约定待求的参数的矩阵形式为：

先求并记为：

求并记为：

的参数为一列向量，所以实现g函数时要支持列向量作为参数，并返回列向量。由上式可知可以由一次计算求得。

再来看一下（15）式的更新过程，当j=0时：

同样的可以写出，

综合起来就是：

综上所述，更新的步骤如下：

（1）求；

（2）求；

（3）求，表示矩阵x的转置。

也可以综合起来写成：

其中，1/m是可以省略的。

# 二、线性回归

线性回归是在已有数据集上通过构建一个线性的模型来拟合该数据集特征向量的各个分量之间的关系，对于需要预测结果的新数据，我们利用已经拟合好的线性模型来预测其结果。线性模型在二维空间中就是一条直线，在三维空间是一个平面，高维空间的线性模型不好去描述长什么样子；如果这个数据集能够用一个线性模型来拟合它的数据关系，不管是多少维的数据，构建线性模型的方法都是通用的。

线性方程为：

使用最小二乘法对和进行估计，为了便于计算，我们把和吸入向量形式,相应的把数据集表示为m\*(d+1)大小的矩阵，其中每行对应于一个示例，该行前d个元素对应于示例的d个属性值，最后一个元素置为1，即：

通过建立一个损失函数来衡量估计值和实际之间的误差的大小，我们将最小化损失函数作为一个约束条件来求出参数向量的最优解。损失函数通过平方误差计算。

其中和互为转置矩阵（=），结果为标量，值相等，可以合并。

对求导，

令，即，所以得到如下：