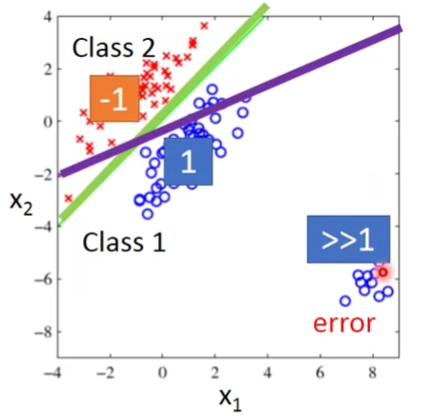
# 机器学习笔记

李宏毅教授机器学习(视频地址：https://www.bilibili.com/video/BV1JE411g7XF?p=3&vd\_source=0a27b8c0284e7cd3b72bc0c89478f2d7)

# 第三课：Classification

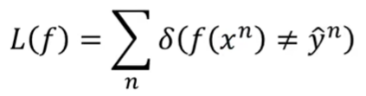
Classification(分类)，就是我们给一个x，通过一个function，给出这个x所属的类型n。

而我们是如何给我们的输入进行分类的呢，第一种想法是用做regression的方法来进行分类，比如class 1代表target是-1，class 2代表target是1，我们将输入进行regression后，接近1的就是class 2，接近-1的就是class 1，但是这是有问题的，如下图所示

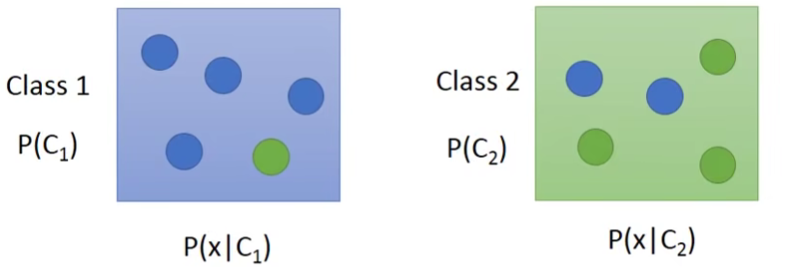


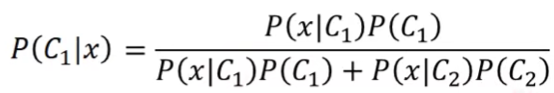
那些“太正确”的x反而对regression来说是错误的，于是我们在做regression的时候会将直线偏一些，变成紫色那条线，但实际上我们用眼睛看是可以看出绿色那条才是更好的，所以不适合用做regression的方法来进行classification。

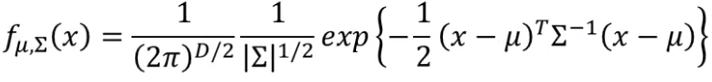
那么我们该怎么做呢？

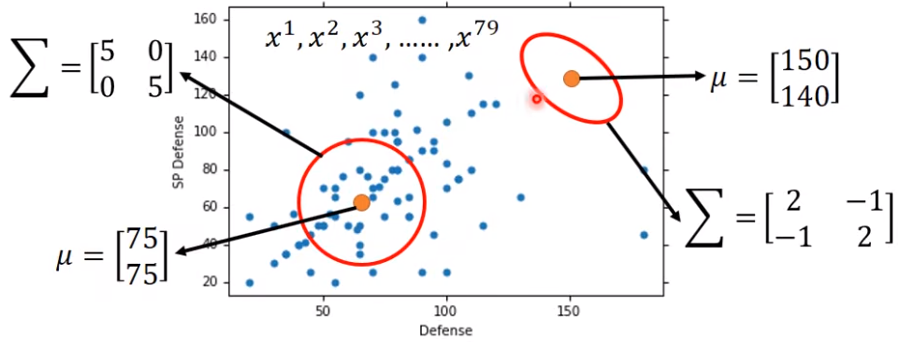
现在考虑二元分类的问题，我们需要一个function g，做到给一个x，如果g(x)>0，那么就x就属于class 1，否则就属于class 2，此时loss function就定义成，这个式子代表某个function预测分类错误的次数。

我们分类的做法就是用概率去进行分类，比如我们现在有两个class



那么x属于class 1的概率就是，然后我们把x属于各个class的概率都求出来，其中概率最大的那个class就是x的class，其中P(C1)和P(C2)概率的计算就是用对应class里面元素的个数除以总个数得出。

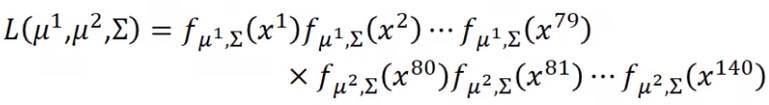
而P(x|C1)和P(x|C2)的求法就是用高斯分布来得出，高斯分布的公式如图所示，其分布由两部分决定，一部分是mean μ，另一部分是covariance matrix Σ。



这个高斯分布就代表其sample出数据的几率，越接近这个分布几率就越大，但是每个地方的几率都不为0，我们用一个L(μ,Σ)来代表这个分布sample出所有点的概率，其公式如图所示，代表每个数据独立sample出来的几率的乘积。



我们需要做的事情就是找到一组()，使得其满足，其求法分别是，，其中是的转置矩阵。

但是这个做法可能无法得出比较好的分类，所以我们对其进行改进，我们给每一个class共用同一个Σ，于是我们的L又写成，我们的目标依然是找到一组，使得这个式子的值最大，而这个Σ也很好求，就是，我们用这种方法求出来的分类，他们的边界是一条直线。

总的来说，classification有三个步骤：

第一步是Function Set，也就是设定一个model。

第二步是Goodness of a function，判断一个function的好坏程度。

第三步是找到最好的一个function。