# Logitic回归

## 什么是分类任务

训练数据：给定训练数据𝒟 = {𝐱𝑖 ,𝑦𝑖 } ,-其中𝑁为训练样本数目，𝑖为样本索引，𝐱𝑖 为第𝑖个样本的输入特征，𝑦i 为对应的输出/响应。

若yi属于实数域，则就是回归问题

若yi属于C， C={1…，C}，我们称为一个分类任务。

（Logistic回归虽然从名字上来看是回归算法，但其实际上是一个分类算法）

## 两类分类

我们以两类分类为例，在两类分类任务中，样本的输出𝑦 ∈ {0,1}，而在概率分布中，贝努利（Bernoulli）试验的输出为 0,1；所以我们可以使用贝努利分布来描述此类问题。其概率密度函数为：𝑝(𝑦;𝜇) = 𝜇y (1-u)1-y。其中y=0或1，所以方程可写为：

𝑝(𝑦 = 1) = 𝜇, 𝑝(𝑦 = 0) = 1 − 𝜇

在分类任务中，在给定输入𝐱的情况下，输出𝑦用贝努利分布描述：𝑦|𝐱~𝐵𝑒𝑟𝑛𝑜𝑢𝑙𝑙𝑖 (𝜇(𝐱)) ，其中期望𝜇(𝐱) 表示在给定𝐱的情况下，𝑦 = 1的概率。概率密度函数为：

𝑝(𝑦;𝜇) = 𝜇(x)y (1-u(x))1-y 。

𝑝(𝑦=1) = 𝜇(𝐱), 𝑝(𝑦=0) = 1 − 𝜇(𝐱)

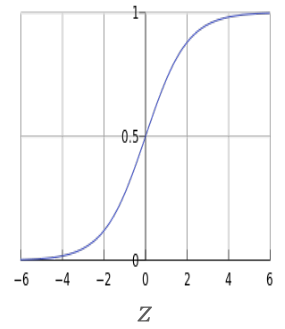
## Logit回归

那么我们需要知道u(x)如何表示？

最简单的就是表示成线性模型：𝜇 (𝐱) = 𝐰T𝐱，但𝐰T𝐱取值范围为负无穷到正无穷，而y取值为u(x)取值应该为[0~1] ，那么我们需要找一个其值范围为0~1函数：

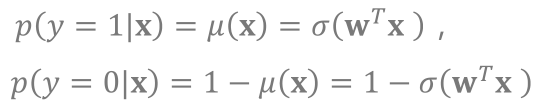


Sigmoid函数图像如下图：

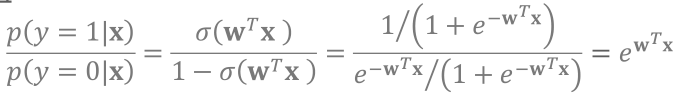


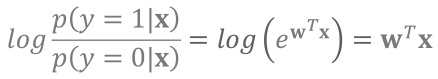
Sigmoid函数也被叫为logistic函数或者logit函数，logistic回归也被称为logit回归。

因此，在Logistic回归模型中：

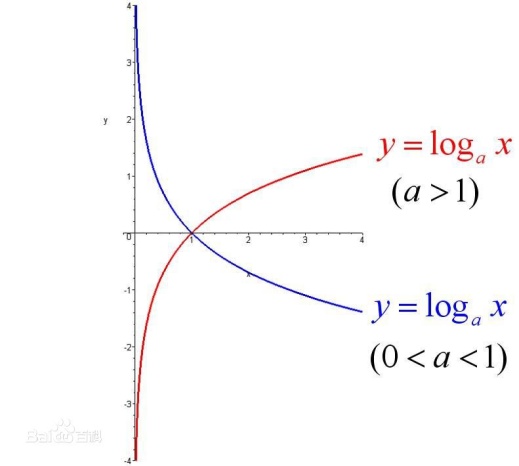


定义一个事件的几率(odds)为该事件发生的概率与不发生的概率的比值:

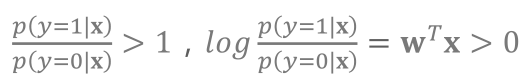


两边同取log运算，得到对数几率: 

Log函数图像如下：



当y=1的概率大于y=0的概率，既𝑝 (𝑦 = 1|𝐱 )> 𝑝 (𝑦 = 0|𝐱) 时，如果取最大后验概率，𝐱的类别取𝑦 = 1：



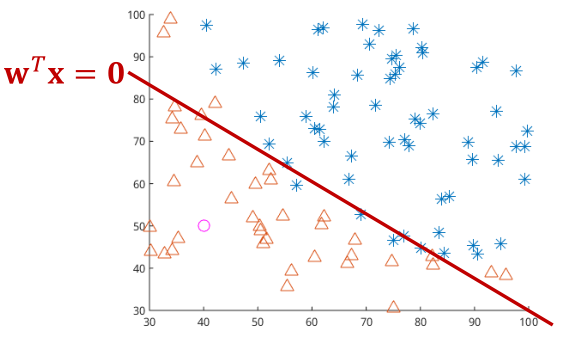
所以我们得出以下结论：

当𝐰T 𝐱 > 0时，𝐱的类别取𝑦 = 1

当𝐰T 𝐱 < 0时，𝐱的类别取𝑦 = 0

当𝐰T 𝐱 = 0时，𝑦 = 1的概率和𝑦 = 0的概率相等，此时𝐱位于决策面上。可将𝐱分类到任意一类，或拒绝作出判断。

由上述结论我们可得决策函数𝑓(𝐱)= 𝐰T 𝐱 ;根据𝐰T 𝐱的符号将输入空间𝐗分出两个区域。𝐰T 𝐱为输入**x**的线性函数，所以Logistic回归模型是一个线性回归分类模型。



## 决策边界

更一般地：根据需要划分的类别，分类器将输入空间X划分为一些互不相交的区域。这些区域的边界叫做决策边界(decision boundaries)。

预测函数的形式不同，会使得决策面或光滑，或粗糙。

决策面是输入的线性函数，称为线性决策面，对应的分类器就是线性分类器。

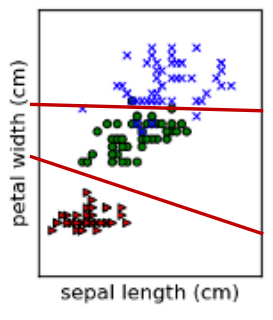
分类器为每个类别分配一个判别函数，根据判别函数来判断一个新样本属于该类别的可能性。

假设有C个类别，则有得到𝐶个判别函数：𝛿c(𝐱) ,c ∈ {1,…,C} 。

对一个新的样本**X**，一般是找到最大的𝛿c(𝐱) , 即该样本的类别为：

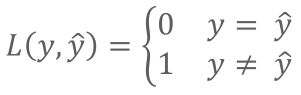


判别函数𝛿(𝐱) 和𝛿k(𝐱) 相等的点的集合，就是类𝑐和类𝑘之间的决策面:

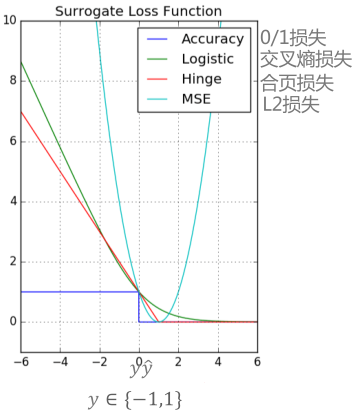


# Logistic损失函数

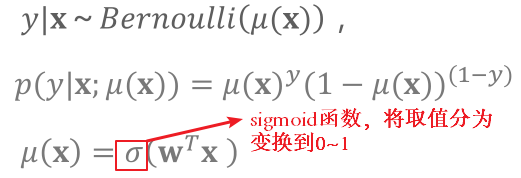
0/1 损失：预测类别正确损失为0，否则为1，记为：



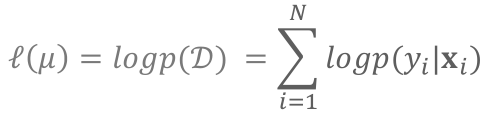
0/1 损失不连续，优化计算不方便，所以需要寻找其他替代损失函数（Surrogate Loss Function），通常是凸函数，计算方便且和0/1损失是一致的。下图为几种Logistic损失函数的图像。

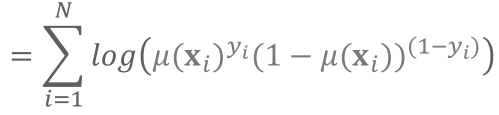


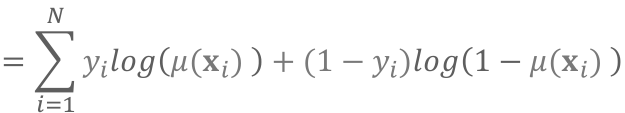
Logistic回归模型：



log似然函数为：





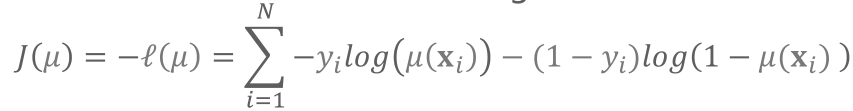


log运算： 、、

(关于极大似然函数和为什么要取对数可以参考下面链接的文章：<https://blog.csdn.net/u014182497/article/details/82252456>)

定义负log似然损失为：

则极大似然估计等价于最小训练集上的负log似然损失：



* log似然估计函数的极大值表示：事件在不同条件下发生的可能性，似然函数的值越大说明该事件在对应的条件下发生结果X的可能性越大
* 所以加负号就是极小值时发生的可能性越大，而越大则越不可能发生结果X
* 损失函数的定义就是越小越接近真实值，越大越远离真实值，和负log似然函数描述一致，所以拿来当做损失函数

负log似然损失亦被称为Logistic损失，Logistic损失亦被称为交叉熵损失（Cross Entropy Loss）。

交叉熵损失：两个分布之间的差异（已知真实分布情况下，预测分布与真实分布之间的差异）

<https://www.leiphone.com/news/201801/T9JlyTOAMxFZvWly.html>

[Understanding Feature Engineering](https://towardsdatascience.com/understanding-feature-engineering-part-1-continuous-numeric-data-da4e47099a7b)