HW1

2018年5月19日　　20:21

1. Machine Learning Problems
第一题主要是关于ML的一些基本概念的理解
(1) Basic Concept:
supervised learning 监督学习
unsupervised learning 无监督学习
classification 分类
regression 回归
clustering 聚类
dimentionality reduction 降维
(2) Tips：
·监督学习和无监督学习最大的区别是训练集有没有label好
·分类和聚类的区别是，分类是已经有了固定的类别，把test的数据扔进去，聚类是完全根据数据自己来学习得到类别
·分类和回归有点像是离散和连续的

2. Bayes Decision Rule
直接扔个例子吧~~

讨论二分类问题，共有Bass和Salman两种鱼，用w1和w2分别表示。有一个特征feature即光照强度，用x表示。现在抓上来一条鱼，判断种类。（请不要说我装逼英语，上课的例子...我也没查是啥鱼...

先验分布 Prior Probability： P(w1) or P(w2) 就是和光照没半毛钱关系，是一种事先知道的既定事实。比如塘主说，这鱼塘本来没鱼，我刚扔了10条Bass和20条Salman进去。于是，P（w1）=10/（10+20）=1/3，P（w2）=20/（10+20）=2/3

似然估计 Likelihood Probability：P(x|w1) or P(x|w2) 就是现在假设x=10强度的时候，比如（x=10|w1)=0.01，也就是说在所有bass里面大部分不适合10强度光照的，但是P(x=10|w2）=0.9，也就是说所有salman里面大部分适合10强度光照的，此时根据概率判断x=10的时候因为P(x|w1)<P(x|w2),所以应该是salmon鱼

后验分布 Posterior Probability：P(w1|x) or P(w2|x)最直观的表示方式，现在我们已知实在x=10光照强度下抓上来一条鱼，问是bass和salman的概率分别是多少呢？

相互之间转换用贝叶斯公式, P(w1|x)=P(x|w1)*P(w1)/P(x)

来自 <https://www.zhihu.com/people/laylalaisy/activities>

在英语语境里，likelihood 和 probability 的日常使用是可以互换的，都表示对机会 (chance) 的同义替代。但在数学中，probability 这一指代是有严格的定义的，即符合柯尔莫果洛夫公理 (Kolmogorov axioms) 的一种**数学对象**（换句话说，**不是**所有的可以用0到1之间的数所表示的对象都能称为概率），而 likelihood (function) 这一概念是由Fisher提出，他采用这个词，也是为了凸显他所要表述的**数学对象**既和 probability 有千丝万缕的联系，但又不完全一样的这一感觉。中文把它们一个翻译为**概率**一个翻译为**似然**也是独具匠心。

先看似然函数的定义，它是给定联合样本值 $\mathbf{x}$ 下关于(未知)参数 θ 的函数： $L(\theta|\mathbf{x}) = f(\mathbf{x}|\theta)$

这里的小 $\mathbf{x}$ 是指联合样本随机变量 $\mathbf{X}$ 取到的值，即 $\mathbf{X} = \mathbf{x}$；

这里的 θ 是指未知参数，它属于参数空间；

这里的 $f(\mathbf{x}|\theta)$ 是一个密度函数，特别地，它表示(给定) θ 下关于联合样本值 $\mathbf{x}$ 的联合密度函数。

这里的 $f(\mathbf{x}|\theta)$ 是一个密度函数，特别地，它表示(给定) θ 下关于联合样本值 $\mathbf{x}$ 的联合密度函数。

所以从定义上，似然函数和密度函数是完全不同的两个**数学对象**：前者是关于 θ 的函数，后者是关于 $\mathbf{x}$ 的函数。所以这里的等号 $=$ 理解为**函数值形式**的相等，而不是两个函数本身是同一函数(根据函数相等的定义，函数相等当且仅当定义域相等并且对应关系相等)。

说完两者的区别，再说两者的联系。

（1）如果 $\mathbf{X}$ 是离散的随机向量，那么其**概率**密度函数 $f(\mathbf{x}|\theta)$ 可改写为 $f(\mathbf{x}|\theta)=\mathbb{P}_\theta(\mathbf{X}=\mathbf{x})$，即代表了在参数 θ 下随机向量 $\mathbf{X}$ 取到值 $\mathbf{x}$ 的**可能性**；并且，**如果我们发现**

$$L(\theta_1|\mathbf{x})=\mathbb{P}_{\theta_1}(\mathbf{X}=\mathbf{x})>\mathbb{P}_{\theta_2}(\mathbf{X}=\mathbf{x})=L(\theta_2|\mathbf{x})$$

那么**似然**函数就反应出这样一个**朴素推测**：在参数 θ_1 下随机向量 $\mathbf{X}$ 取到值 $\mathbf{x}$ 的**可能性大于** 在参数

可以看看wiki对似然函数的解释，写的很清楚。
zh.wikipedia.org/wiki/%...
文中提到了在已知某个参数 θ 时，事件 X 会发生的概率写作：

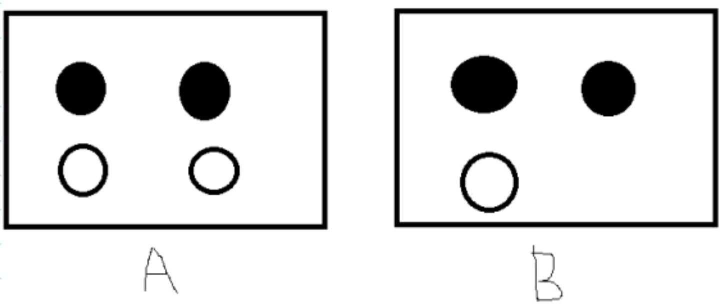
$$P(X|\theta)=\frac{P(X,\theta)}{P(\theta)}$$

然后似然函数是已知 X 对于 θ 的函数，根据贝叶斯定理，上式可以改写成：

$$P(\theta|X)=\frac{P(X|\theta)*P(\theta)}{P(X)}$$

正如@戴玮的答案提到的“先验分布是均匀分布”（其实我觉得说分布不准确，因为在最大似然估计中没有将 θ 视为分布，而是看作一个值，这个时候不同的 θ 出现的概率是相同的。）

(a) 蒙提霍尔问题
<https://zh.wikipedia.org/wiki/%E8%92%99%E6%8F%90%E9%9C%8D%E7%88%BE%E5%95%8F%E9%A1%8C>
参赛者会看见三扇关闭了的门，其中一扇的后面有一辆汽车或者是奖品，选中后面有车的那扇门就可以赢得该汽车或奖品，而另外两扇门后面则各藏有一只山羊或者是后面没有任何东西。当参赛者选定了一扇门，但未去开启它的时候，知道门后情形的节目主持人会开启剩下两扇门的其中一扇，露出其中一只山羊。主持人其后会问参赛者要不要换另一扇仍然关上的门。问题是：换另一扇门会否增加参赛者赢得汽车的机会率？如果严格按照上述的条件的话，答案是会。—换门的话，赢得汽车的机率是2/3。
（1）假设我们有如下的7个球在A,B两个框中，如果我们随便取一个球，已知取到的球来自B框中，那么这个球是白球的概率是多少呢？



对于这个概率，很容易看出是1/3，但是我们也可以通过贝叶斯公式进行计算。将具体的例子代入上式，有P(白球|B框)=P(B框|白球)P(白球)/P(取到B框中的球)=1/3*3/7*7/3=1/3
（2）

Bayesian Statistics

Allright, before we proceed to the solution, let's first have a brief review of Bayesian Statistics.

The following equation, or **Bayes' Theorem**, lies at the heart of Bayesian Statistics,

$$P(B \mid A) \times P(A)$$

The following equation, or **Bayes' Theorem**, lies at the heart of Bayesian Statistics,

$$P(A \mid B) = \frac{P(B \mid A) \times P(A)}{P(B)}.$$

It basically is just a reformulation of the condition probability equation,

$$P(A, B) = P(A \mid B) \times P(B) = P(B \mid A) \times P(A).$$

Under independency, the equation just reduces to

$$P(A, B) = P(A) \times P(B),$$

or equivalently,

$$P(A \mid B) = P(A) \quad P(B \mid A) = P(B).$$

As for $P(B)$,

$$P(B) = \int P(b)\mathrm{d}b.$$

OK, now that you are equipped with fundamentals of statistics, let's move on!

The Bayesian Approach

Let's view the Monty Hall problem in its original scenario.

To decide whether we should switch, we should compare probabilities of $P(No.1 = car \mid No.2 = goat)$ and $P(No.3 = car \mid No.2 = goat)$. And these are just conditional probabilites that require the Bayes' Theorem. Note that

$$P(No.3 = car \mid No.2 = goat) = 1 - P(No.1 = car \mid No.2 = goat).$$

Thus, we only need to work out $P(No.1 = car \mid No.2 = goat)$.

Using the Bayes's Theorem,

$$P(No.1 = car \mid No.2 = goat) = \frac{P(No.2 = goat \mid No.1 = car) \times P(No.1 = car)}{P(No.2 = goat)}.$$

The probability of No. 1 containing the car is

$$P(No.1 = car) = \frac{1}{3},$$

while the probability of the door the host opens containing the goat given that your choice contains the car could be easily derived as

$$P(No.2 = goat \mid No.1 = car) = 1.$$

$$P(No.2 = goat \mid No.1 = car) = 1.$$

Well, until now, nothing fancy. But here comes the trickiest part.

What is $P(No.2 = goat)$?

One who thinks little of it might carelessly concludes $\frac{2}{3}$ since the probability of every door containing the car is $\frac{1}{3}$ and that of a goat is just the complement.

Excuse me, but that's not true.

Notice in the statement that the host always opens a door that **does not contain a car**.

Therefore the probability is actually 1.

As a result,

$$P(No.1 = car \mid No.2 = goat) = \frac{1 \times \frac{1}{3}}{1} = \frac{1}{3}.$$

AH! It seems that **we should switch**!

Quite counterintuitive!

The Frequentist Approach

It was actually a bit confusing when I made my first attempt to understand it from a Bayesian view, although it turned out easier if you simply listed all cases and counted the number of wins.

Let me show you.

Behind No. 1	Behind No. 2	Behind No. 3	Result if stick	Result if switch
Car	Goat	Goat	**Win**	Lose
Goat	Car	Goat	Lose	**Win**
Goat	Goat	Car	Lose	**Win**

Table from [1]

As illustrated above, $P(No.1 = car \mid Door = goat) = \frac{1}{3}$.

Naive Generalization

Now that we have worked out the case when there are only 3 doors with only 1 containing the prize, it's natural that we seek the solution when there are more doors and more than 1 prizes.

Similarly, if there is 1 prize and n doors, we should again switch since the Bayes' Theorem tells that the probability of wins remains $\frac{1}{n}$.

the probability of wins remains $\frac{1}{n}$.

But the problem becomes a little bit complicated when there are more than 1 prizes.

Suppose there are m prizes and n doors. Then the probability of winning if you stick turns to be

$$P = \frac{m}{n}.$$

If this is larger than $\frac{1}{2}$ then we stick, otherwise we switch.

References

1. Monty Hall problem - Wikipedia, The Free Encyclopedia

http://wellyzhang.github.io/2016/05/06/monty-hall/

最重要的一个点就是因为总是打开没有bonus的门，所以其实概率是1。

在作业里面相类似，特别要注意因为每次主持人都打开没有bonus的盒子，所以其实P(B2=0)=1

(b)

Maximum Likelihood Decision Rule：

Likelihood

似然度/可能性：已知分类的情况下判断这种分类中事件x出现的概率

- Suppose now we have a measurement or feature on the state of nature - say the fish lightness value
- $P(x|\omega_1)$ and $P(x|\omega_2)$ describe the difference in lightness feature between populations of sea bass and salmon
- $P(x|\omega_j)$ is called the **likelihood** of ω_j *with respect to x; the category* ω_j *for which* $P(x \mid \omega_j)$ is large is more likely to be the true category
- **Maximum likelihood decision**
  - Assign input pattern x to class ω_1 if
    $P(x \mid \omega_1) > P(x \mid \omega_2)$, otherwise ω_2

尤其要注意这里是P(x|wj)的概率，就是假设这边有两个类别分贝是bass和salman鱼，即bass是w1，salmon是w2，然后x对应光照，假设x=10强度的时候，比如P（x=10|w1)=0.01,也就是说在所有bass里面大部分不适合10强度光照的，但是P（x=10|w2）=0.9，也就是说所有salmon里面大部分适合10强度光照的，此时根据概率判断x=10的时候应该是salmon鱼

Optimal Bayes Decision Rule

- $P(\omega_1 \mid x)$ is the probability of the state of nature being ω_1 given that feature value x has been observed
- Decision given the posterior probabilities, Optimal Bayes Decision rule

X is an observation for which:

if $P(\omega_1 \mid x) > P(\omega_2 \mid x)$ ➔ True state of nature = ω_1

if $P(\omega_1 \mid x) < P(\omega_2 \mid x)$ ➔ True state of nature = ω_2

Bayes decision rule minimizes the probability of error, that is the term

Bayes decision rule minimizes the probability of error, that is the term Optimal comes from. But why? Can you prove it?

Bayes Risk

- Conditional risk

$$R(\alpha_i|\boldsymbol{x}) = \sum_{j=1}^{c} \lambda(\alpha_i|\omega_j)P(\omega_j|\boldsymbol{x})$$

一共有c种class，然后将每种class时的lost乘以该class概率后相加

- Select the action for which the conditional risk $R(\alpha_i|\boldsymbol{x})$ is *minimum*

$$R = \sum_{over\ \boldsymbol{x}} R(\alpha_i|\boldsymbol{x})$$

- Risk R is minimum and R in this case is called the
  - Bayes risk = best performance that can be achieved!

Minimal Total Risk: 简单来说就是对于每一个每种x情况下，action选择一个分类，假如按题目二分类，这种时候会出现两种可能的错误：首先action是认为只要出现x情况，就选择分类1/分类2，这是会出现选择分类1但其实是分类2/选择分类2其实是分类1的，将概率*cost得到risk，然后选择risk较小的那种action

There are two actions $\{\alpha_1, \alpha_2\}$ we can take, with their loss matrix below. Show the minimal total risk ($R = \sum_x \min_i R(\alpha_i|x)$) we can get.

risk=2.44

$\lambda(\alpha_i\|\omega_j)$	$j=1$	$j=2$
$i=1$	0	1
$i=2$	2	0

cat()

https://www.cnblogs.com/tornadomeet/p/3515535.html

转 MATLAB的cat()函数

原文地址：MATLAB的cat()函数 作者：工程师

cat：用来联结数组

用法：C = cat(dim, A, B) 按dim来联结A和B两个数组。

C = cat(dim, A1, A2, A3, ...) 按dim联结所有输入的数组。

e.g.

a=cat(3,A,B) 左括号后的3表示构造出的矩阵维数；在新的矩阵中第1、2维就是A和B这两个矩阵的行数和列数，第3维是A和B这两个矩阵的矩阵个数，即为2

cat(2, A, B)相当于[A, B]；

cat(1, A, B)相当于[A; B].

使用“,”分割列表的语法，cat(dim, C{:})或cat(dim, C.field)，这是将包含数据矩阵的cell或结构数组联合为一个数组的方便方式。

例如：

```
>> A = [1 2; 3 4];
>> B = [5 6; 7 8];
```

例如：

```
>> A = [1 2; 3 4];
>> B = [5 6; 7 8];
>> A
```



```
    1     2
    3     4
>> B
B =
    5     6
    7     8
>> cat(1, A, B) %按列连接（列数相同）
ans =
    1     2
    3     4
    5     6
    7     8
>> cat(2, A, B) %按行连接（行数相同）
ans =
    1     2     5     6
    3     4     7     8
>> cat(3, A, B) %合成效果如下图
ans(:,:,1) =
    1     2
    3     4
ans(:,:,2) =
    5     6
    7     8
```

```
a = magic(3)
b = pascal(3)
```

k=1,合并后形如 [a;b],行添加矩阵（要求a,b的列数相等才能合并）；

```
>> c = cat(1,a,b)
c =
    8     1     6
    3     5     7
    4     9     2
    1     1     1
    1     2     3
    1     3     6
```

k=2,合并后形如[a,b],列添加矩阵（要求a,b的行数相等才能合并）

```
>> c = cat(2,a,b)
c =
    8     1     6     1     1     1
    3     5     7     1     2     3
    4     9     2     1     3     6
>> c = cat(3,a,b)
c(:,:,1) =
```

```
>> c = cat(3,a,b)
c(:,:,1) =

     8     1     6
     3     5     7
     4     9     2

c(:,:,2) =

     1     1     1
     1     2     3
     1     3     6
```

n维的矩阵合并，要求n-1维维数相等才可以.

```
>> c = cat(4,a,b)
c(:,:,1,1) =

     8     1     6
     3     5     7
     4     9     2

c(:,:,1,2) =

     1     1     1
     1     2     3
     1     3     6
```

3. Gaussian Discriminant Analysis and MLE

(a)

1. 二维正态分布

定律定义

编辑

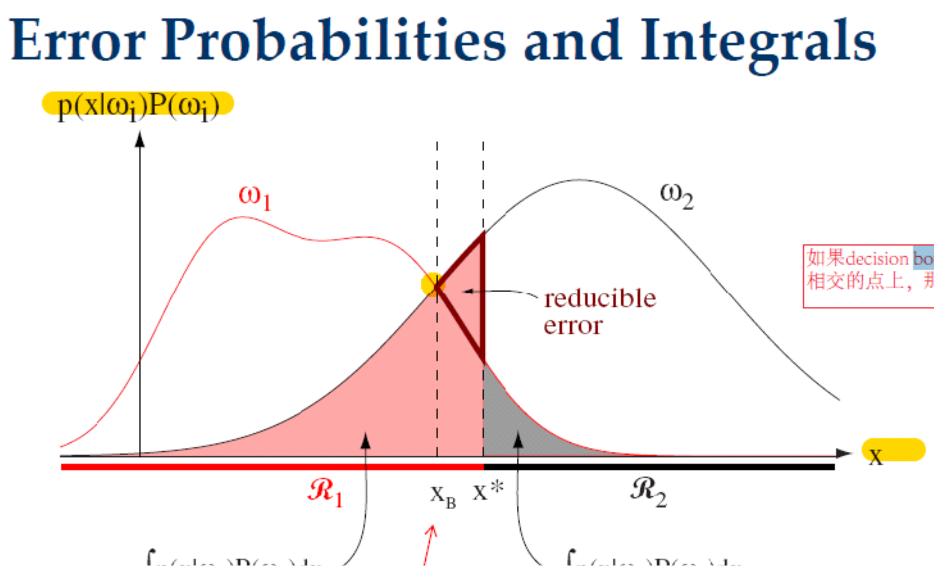
满足下述的概率密度分布的随机变量分布叫做二维正态分布

$$f(x,y)=\left(2\pi\sigma_1\sigma_2\sqrt{1-\rho^2}\right)^{-1}exp\left[-\frac{1}{2(1-\rho^2)}\left(\frac{(x-\mu_1)^2}{\sigma_1^2}-\frac{2\rho(x-\mu_1)(y-\mu_2)}{\sigma_1\sigma_2}+\frac{(y-\mu_2)^2}{\sigma_2^2}\right)\right]$$

其中 $\mu_1,\mu_2,\sigma_1,\sigma_2,\rho$ 都是常数，我们称 (X_1,X_2) 服从参数为 $\mu_1,\mu_2,\sigma_1,\sigma_2,\rho$ 的二维正态分布，常把这个分布记作 $N\left(\mu_1,\mu_2,\sigma_1^2,\sigma_2^2,\rho\right)$ 。 $\mu_1,\mu_2,\sigma_1,\sigma_2,\rho$ 的范围分别为 $-\infty<\mu_1<+\infty;-\infty<\mu_2<+\infty;-1<\rho<1\ \sigma_1\geq 0;\sigma_2\geq 0$ 。这个函数在三维空间中的图像是一个椭圆切面的钟倒扣在 Ox_1x_2 平面上，其中心在（ μ_1,μ_2 ）点。

2. Decision Boundary

当两者相同的时候为decision boundary，即此时会使error最小



如果decision boundary 就在两个相交的点上，那么error最小

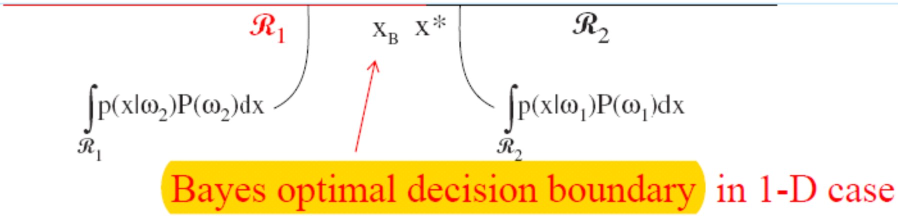
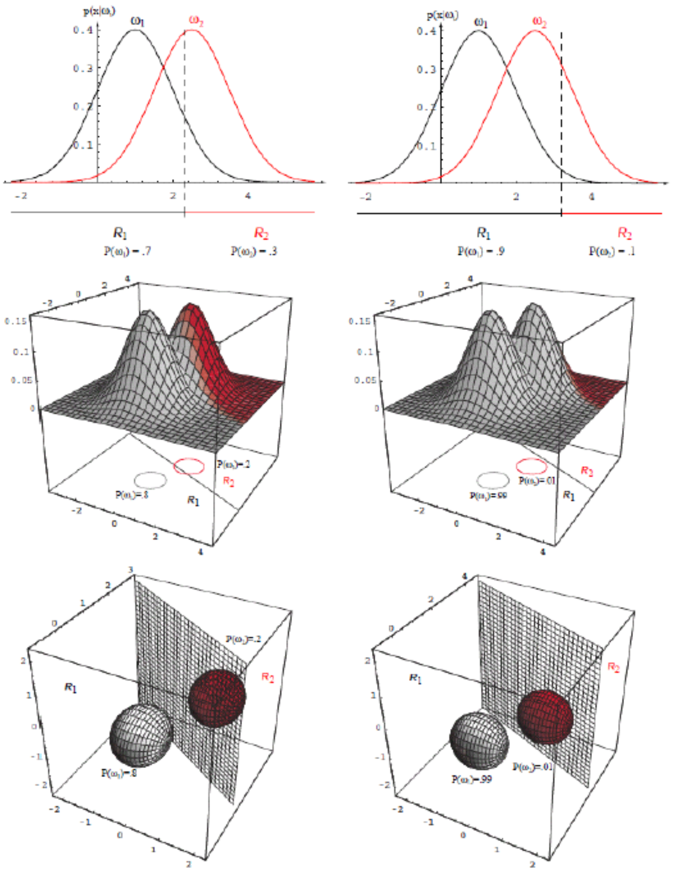
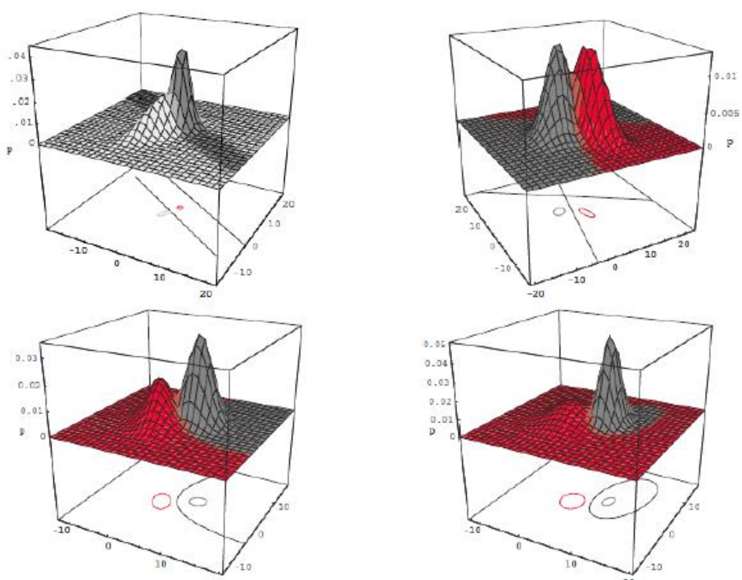
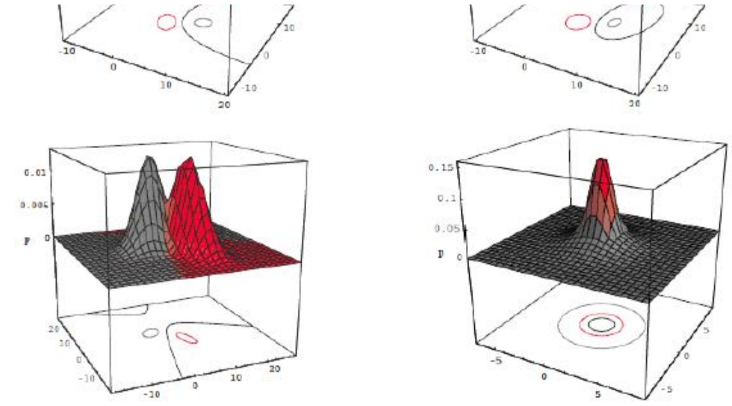


Figure 2.17: Components of the probability of error for equal priors and (non-optimal) decision point x^*. The pink area corresponds to the probability of errors for deciding ω_1 when the state of nature is in fact ω_2; the gray area represents the converse, as given in Eq. 68. If the decision boundary is instead at the point of equal posterior probabilities, x_B, then this reducible error is eliminated and the total shaded area is the minimum possible — this is the Bayes decision and gives the Bayes error rate.

(C) 二维正态分布omiga各个参数的影响：







https://blog.csdn.net/xfijun/article/details/53822490

（d）极大似然值估计

极大似然估计

以前多次接触过极大似然估计，但一直都不太明白到底什么原理，最近在看贝叶斯分类，对极大似然估计有了新的认识，总结如下：

贝叶斯决策

首先来看贝叶斯分类，我们都知道经典的贝叶斯公式：

$$P(w|x)=\frac{p(x|w)p(w)}{p(x)}$$

其中：p(w)：为先验概率，表示每种类别分布的概率；$p(x|w)$：类条件概率，表示在某种类别前提下，某事发生的概率；而$P(w|x)$为后验概率，表示某事发生了，并且它属于某一类别的概率，有了这个后验概率，我们就可以对样本进行分类。后验概率越大，说明某事物属于这个类别的可能性越大，我们越有理由把它归到这个类别下。

我们来看一个直观的例子：**已知：**在夏季，某公园男性穿凉鞋的概率为1/2，女性穿凉鞋的概率为2/3，并且该公园中男女比例通常为2:1，**问题：**若你在公园中随机遇到一个穿凉鞋的人，请问他的性别为男性或女性的概率分别为多少？

从问题看，就是上面讲的，某事发生了，它属于某一类别的概率是多少？即后验概率。

设：$w_1=$男性，$w_2=$女性，$x=$穿凉鞋。

由已知可得：

$$先验概率\ p(w_1)=2/3，\ p(w_2)=1/3$$

$$类条件概率\ p(x|w_1)=1/2，\ p(x|w_2)=2/3$$

男性和女性穿凉鞋相互独立，所以 $p(x)=p(x|w_1)p(w_1)+p(x|w_2)p(w_2)=5/9$

（若只考虑分类问题，只需要比较后验概率的大小，的取值并不重要）。

$$P(w_1|x)=\frac{p(x|w_1)p(w_1)}{p(x)}=\frac{1/2\times 2/3}{5/9}=\frac{3}{5}$$

$$P(w_2|x)=\frac{p(x|w_2)p(w_2)}{\cdots}=\frac{2/3\times 1/3}{\cdots}=\frac{2}{\cdots}$$

由贝叶斯公式算出: $$P(w_2 \mid x)=\frac{p(x \mid w_2)\,p(w_2)}{p(x)}=\frac{2/3\times 1/3}{5/9}=\frac{2}{5}$$

4. Text Classification with Naive Bayes
https://blog.csdn.net/qq_35083093/article/details/75340848

（1）拉普拉斯平滑处理 laplace smoothing

背景:为什么要做平滑处理?

零概率问题，就是在计算实例的概率时，如果某个量x，在观察样本库（训练集）中没有出现过，会导致整个实例的概率结果是0。在文本分类的问题中，当一个词语没有在训练样本中出现，该词语调概率为0，使用连乘计算文本出现概率时也为0。这是不合理的，不能因为一个事件没有观察到就武断的认为该事件的概率是0。

拉普拉斯的理论支撑

为了解决零概率的问题，法国数学家拉普拉斯最早提出用加1的方法估计没有出现过的现象的概率，所以加法平滑也叫做拉普拉斯平滑。

假定训练样本很大时，每个分量x的计数加1造成的估计概率变化可以忽略不计，但可以方便有效的避免零概率问题。

应用举例

假设在文本分类中，有3个类，C1、C2、C3，在指定的训练样本中，某个词语K1，在各个类中观测计数分别为0，990，10，K1的概率为0，0.99，0.01，对这三个量使用拉普拉斯平滑的计算方法如下：

1/1003 = 0.001，991/1003=0.988，11/1003=0.011

在实际的使用中也经常使用加 lambda（1≥lambda≥0）来代替简单加1。如果对N个计数都加上lambda，这时分母也要记得加上N*lambda。

来自 <https://www.cnblogs.com/bqtang/p/3693827.html>

（2）spconver()
https://ww2.mathworks.cn/help/matlab/ref/spconvert.html
将一个稀疏矩阵进行压缩，就本来比如有一个10000*1000的矩阵，但是只有一个非零的值[500,500]=1 这时候就不要保存一个10000*1000的矩阵了而是保存500 500 1 其中前面两个表示非零的位置，后面一个表示非零的值，从而可以大大压缩矩阵

（3）找最大的10个数的坐标
https://zhidao.baidu.com/question/115035565.html

（4）disp

disp(X)显示一个数组时不需要输出它的名称。如果X包含一个文本字符串，则显示字符串。

另一种在电脑屏幕上显示数组的方法是输出数组的名称，但在输出结果之前还有一个抬头，通常是不需要的。

disp函数仅接受一个输入。为了显示多于一个数组和字符串，需要将函数sprintf或fprintf串联起来。

注意disp不显示空数组。

下面的例子使用disp来显示带有标签的矩阵：

下面的例子使用disp来显示带有标签的矩阵：

```
disp('    Corn      Oats      Hay')
x = gallery('uniformdata',[5 3],0);
disp(x)
```

结果是：

```
    Corn      Oats      Hay
   0.9501    0.7621    0.6154
   0.2311    0.4565    0.7919
   0.6068    0.0185    0.9218
   0.4860    0.8214    0.7382
   0.8913    0.4447    0.1763
```

（5） sparse矩阵
https://jingyan.baidu.com/article/6181c3e08a4d6f152ef15333.html