

INFORMATION THEORY

AND CODING

~STUDY GUIDE~

2013年1月

# 第二版前言

经过若干天的修订之后，这个Study Guide的第二版和各位见面了。

第二版以第一版为基础修订。第一版已经讲述了所有必要的基础知识。但是我一直认为有些必要的东西根本就没讲到，比如说香农三大定理。对于编码理论这本书几乎没有涉及，但是这是培养一个信息工程专业出身的人的思维方式的必经之路，所以我加进去了这部分东西。

除此之外，修订的部分还包括原书中的几处严重错误。

下面有必要说明一下这本书的使用方法。实际上我在第一版前言中就已经说过这本书倡导你们积极思考。如果有人想在考试前三天，或者考前一个晚上，甚至考前一小时才根据我这本书开始复习，那么恭喜你：这本书的设计结构从本质上就决定了抱有这种思想的你来年重修。

我一向认为，从哲学层面上理解现象是最为根本的。也因此，这本书会告诉你感性层面的信息论体系结构，会对引导向哲学原理的数学证明加以详细解释，但是不会展示给你例题怎么解。所以，在阅读这本书的同时，读者必须以书上的例题和做过的习题为辅助。特别是在连续信道与连续信源一章，这一章在本书中只给出了一些关键性的数学证明，但是有些题目完全是在利用二元积分，讲解这些东西完全无意义。所以练习的事情就扔给读者了。

在本书的共享和传承问题上，本人放弃使用OneNote而使用Github进行管理。如果有人有兴趣，就可以对我的作品进行继承性创作，直接在Github上fork之即可。该作品的地址是<https://github.com/wattlebird/study_guide.git> 当然，仅限有兴趣且需要对此书进行修改的人编辑。

Ronnie Wang

2013-02-08

# 第一版前言

首先请允许我对前来阅读我写的信息论与编码学习指南（华理专用）的同学表示欢迎。

2012-2013学年第一学期的信息论与编码考试结果令人震惊。在某一个时间段里，我天天都能听到各种各样的噩耗。特别是有些人前三年级从来都没有不及格过的就是这次导致了他们大学生活的遗憾（有人偏要说是“完美”）。有些人在我印象中还很好。我决定要帮助这些可爱的同学们。这个想法我只向一小部分人公布了，初步证明非常受欢迎。他们愿意听我讲而不是听吴耀军讲。

在阅读这本学习指南之前，我要说明清楚这本书的指导原则：

这本书定位为一本学习指南。读者在阅读这本书的时候，大脑应该处于积极思考的状态，目的是建立起抽象物理和数学描述与实际通信过程之间的联系。这种联系应该是一种感性的联系，以至于你作为一个通信工程专业出身的人在今后的生活中都被打下了以信息论的眼光看世界的思维习惯。

这本书同时也继承了我本人的风格：重数学证明而轻机械运算。为什么呢？数学证明能够帮助你从根本上理解我们要用的公式到底是什么意义，是怎么来的。这样就避免了有人在考试时乱套公式，或是遇见新题型而不知道怎么下手。这种釜底抽薪的作风应该是被大力发扬的。当然，由于整本书一直在强调怎么理解概念，对实际解题中出现的形式讲得很少，所以大家仍然有必要把书上例题和习题做一遍，以印证你所学习到的概念。

同时，你会发现这本书使用Microsoft OneNote编辑，这是因为OneNote有很大的灵活性。在正文的旁边，有些地方有说明，这些说明是思维出现矛盾时的火花。为什么思维会出现矛盾？因为现在所学的东西和以往所学的东西在印象中发生了冲突。这时需要比较概念，找出为什么不同。这一点特别体现在离散信源和信道的阐述与连续情况作比较时。我希望你也要把你在学习时遇到的puzzle都记录在书的旁边。关于Microsoft OneNote的使用方法在同一文件夹中另一文档中已有说明，我想十分钟就能看懂。

我对你们的要求就是：通过读我这本书，理解信息论的基本要义，**顺便**通过考试。再强调一遍，考试应该作为你学懂信息论的赠送品。

值得一提的是，本书（应该是——如果技术上没有出问题的话）可供任何人参与编辑的。那么实际上，这本书就成为了一个自由、开放的社区。你可以对我写得不好的地方在旁边进行吐槽，或者你自己作为作者之一对我的话进行改写。如果你认为我还有某些部分没有被添加进来（比如说率失真函数的计算，但是我本人认为这个不是本科所学内容，虽然被有些人标记为考试范围）你可以自己进行添加。这个好处在于本书不仅是对我们这一次参加补考的同学有效，还将对下一届、下下届同学有效，只要他们中间有出色的作者的话。

下面是致谢：

感谢赵嘉俊、孙瑜凯和夏书豪，因为在学习信息论的同时我们之间讨论的问题最多，使我知道了你们的问题都是些什么问题，什么地方需要侧重等等。

感谢夏书豪和包启蒙，因为你们在ComiCup11入场前对我公布的这个企划表示了大力支持。特别再次感谢夏书豪，因为他为了让我写这本书把原来的绿皮课本顺丰包邮送了过来。

感谢在美国的李飒飒，因为我首先公布这个企划是在QQ空间上，你给我点了第一个“赞”（而且也是唯一一个“赞”……）。

Ronnie Wang

2013-01-27

# 版权页

本书的全部文字在知识共享 署名-非商业性使用-相同方式共享 2.5 中国大陆之条款下提供。

**您可以自由：**

* 复制、发行、展览、表演、放映、广播或通过信息网络传播本作品
* 创作演绎作品

**惟须遵守下列条件：**

* **署名** — 您必须按照作者或者许可人指定的方式对作品进行署名。
* **非商业性使用** — 您不得将本作品用于商业目的。
* **相同方式共享** — 如果您改变、转换本作品或者以本作品为基础进行创作，您只能采用与本协议相同的许可协议发布基于本作品的演绎作品。

**With the understanding that:**

* **Waiver** — Any of the above conditions can be [waived](http://web.archive.org/web/20101027021729/http:/creativecommons.org/licenses/by-nc-sa/2.5/cn/) if you get permission from the copyright holder.
* **公共领域** — Where the work or any of its elements is in the [public domain](http://web.archive.org/web/20101027021729/http:/wiki.creativecommons.org/Public_domain) under applicable law, that status is in no way affected by the license.
* **Other Rights** — In no way are any of the following rights affected by the license:
  + Your fair dealing or [fair use](http://web.archive.org/web/20101027021729/http:/wiki.creativecommons.org/Frequently_Asked_Questions#Do_Creative_Commons_licenses_affect_fair_use.2C_fair_dealing_or_other_exceptions_to_copyright.3F) rights, or other applicable copyright exceptions and limitations;
  + The author's [moral](http://web.archive.org/web/20101027021729/http:/wiki.creativecommons.org/Frequently_Asked_Questions#I_don.E2.80.99t_like_the_way_a_person_has_used_my_work_in_a_derivative_work_or_included_it_in_a_collective_work.3B_what_can_I_do.3F) rights;
  + Rights other persons may have either in the work itself or in how the work is used, such as [publicity](http://web.archive.org/web/20101027021729/http:/wiki.creativecommons.org/Frequently_Asked_Questions#When_are_publicity_rights_relevant.3F) or privacy rights.
* **Notice** — 对任何再使用或者发行，您都必须向他人清楚地展示本作品使用的许可协议条款。

目录

[第一章 基本概念 7](#_Toc348086671)

[1.1 信息论研究对象和工具 7](#_Toc348086672)

[1.2由平均信息量H(X)引申出来的其他东西 8](#_Toc348086673)

[1.3 相关定理的数学证明 9](#_Toc348086674)

[第二章 离散信源 11](#_Toc348086675)

[2.1离散无记忆信源的特性 11](#_Toc348086676)

[2.2 简单的马尔可夫过程 12](#_Toc348086677)

[2.3 离散有记忆信源的特性 14](#_Toc348086678)

[2.4 信源编码基础 15](#_Toc348086679)

[课外阅读：输入一个汉字需要敲多少个键——谈谈香农第一定律 16](#_Toc348086680)

[第三章 离散无记忆信道 19](#_Toc348086681)

[3.1 什么是离散无记忆信道 19](#_Toc348086682)

[3.2 对称信道 21](#_Toc348086683)

[3.3 信道级联 22](#_Toc348086684)

[3.4 信道编码基础 23](#_Toc348086685)

[第四章 连续信源和连续信道 24](#_Toc348086686)

[4.1 连续信源的基本特性 24](#_Toc348086687)

[4.2 连续信源的独立性 26](#_Toc348086688)

[4.3 计算时的注意事项 26](#_Toc348086689)

[4.4 连续信道 27](#_Toc348086690)

[第五章 率失真理论 28](#_Toc348086691)

[5.1 平均失真度 28](#_Toc348086692)

[5.2 率失真函数 28](#_Toc348086693)

[5.3 确定D的定义域 29](#_Toc348086694)

[第六章 信源编码 31](#_Toc348086695)

[6.1 Fano编码 31](#_Toc348086696)

[6.2 Shannon编码 32](#_Toc348086697)

[6.3霍夫曼编码 32](#_Toc348086698)

[6.4编码效率 34](#_Toc348086699)

[第七章 卷积码 35](#_Toc348086700)

[7.1 卷积码编码器 35](#_Toc348086701)

[7.2 卷积码之所以称为卷积码 37](#_Toc348086702)

[7.3 论多项式 38](#_Toc348086703)

[7.4 状态转换图与篱笆图 39](#_Toc348086704)

# 第一章 基本概念

## 1.1 信息论研究对象和工具

大家好，从现在开始，我将努力向大家阐明信息论的基本要义。

信息论用数学的方式去描述信息社会。据香农所述，信息的传递过程可以被抽象化为信源、信宿和两者之间的信道。所传输的东西无外乎为一个个符号，如基本的0和1（当然，二进制符号传输是最简单的情况）。信源是发送符号的那个东西。通常，我们把一个信源记作*X、Y、Z*等。而一个信源X可以发送出符号,,…, ，每个符号又以一定概率出现。实际上，*X、Y、Z*也就是随机变量。

信息论以概率论与数理统计的方法描述世界。下面这个式子在整本书中被频繁使用：

|  |  |
| --- | --- |
|  |  |

H(X)中的X，既可以解释为随机变量，也可以解释为信源。那么这里的H(X)，不是说以随机变量X作为自变量的一个函数，而是一个记号，描述了信源X的生来固有特性：

* 通过观察H(X)的表达式可以知道，它是对的加权平均。而，则为每个符号出现所包含的信息量大小。信息量的大小，为——一个符号出现概率倒数的以2为底的对数所表征，单位是比特(bit)。
* 而把每个符号加权平均后的信息量作为度量信源的最重要属性——*H(X)*，其意义为：这个信源平均每发送一个符号，所发送出去的信息量。这个叫做平均信息量。

还有一点需要特别指出的是：

* *H(X)*的单位是比特每符号(bits/symbol).

从量纲上解释，就是信源平均每发送一个符号，相当于发送多少个比特的信息。比特，成为了信息量大小的衡量单位。

对于一个只发送两个符号的信源*X*，记发送的概率是p，发送的概率是1-p，则信源*X*的平均信息量是

经计算，在p=0.5时*H(X)*达到最大值1 bit/symbol。

计算机生成了可选文字: Figurc1.1.H(P)==H切．1一p)asafunctionofp
1弓一～－一～一－一二
门
H(f
0
0.5
一p

通过看图我们可以知道*H(X)*在与出现概率相同时达到的平均信息量最大。这一现象可以解释为由于各个符号之间出现的机会在概率上没有任何倾向性，在一个时刻，我们无法预知因为哪个符号出现的概率比另一个大就以大概率推测出其中一个可能会收到。这时候的不确定性也就最大，一旦收到符号，消除的不确定性也是最大的。不像某些时候，比如说出现的概率是0.9，这时候如果是对收到哪个符号做赌注的话我想很大一部分人都会赌， 因为以大概率出现，根本就没有赌的价值。

这一现象推广到多个符号的信源，我们可以从直觉上理解为什么信源{*X:X=*,,…,的每个符号等概出现——即每个符号出现的概率均为时信源的平均信息量最大。

这个性质在数学上记作*H(X)*。

*——H(X)*的这个性质只在离散信源下成立。

*H(X)*还有一个性质，那就是*H(X)*.这是因为根据*H(X)*的表达式，其被加权的每一项（即）均大于等于0，所以*H(X)*0.

——同样这个性质只在离散信源下成立。

关于*H(X)*的严格数学证明在后面会加以解释。现在重要的是在感性上保持这一认知。

## 1.2由平均信息量H(X)引申出来的其他东西

从上一节我们知道H(X)是信源X包含的平均信息量，又名平均不确定性。这一节我们将解释什么是H(X,Y)，H(X|Y)和I(X;Y)。

H(X,Y)名为联合平均信息量（joint information measure）。设X是包含符号集的信源，Y是包含符号集的信源。符号和出现的联合概率为。当然，

H(X,Y)代表信源X和Y同时出现时给予我们的平均信息量，或平均不确定性。

H(X,Y)与H(X)和H(Y)有什么关系？根据我所说的感性定义，H(X,Y)=H(X)+H(Y)。初步的理解这样没错，但是在实际生活中，X与Y总有那么一定联系，当收到X时可以在一定程度上确定Y，这时候就称为X与Y不互相独立。H(X,Y)H(X)+H(Y)。

仍然要强调，现在的这个式子只适用于离散信源的条件下。

H(X|Y)名为条件平均信息量（conditional information measure）。记对的条件概率为，对的条件概率为

H(X|Y)代表在已知Y的条件下，X的平均信息量。

由这个定义可知，H(X,Y)=H(X)+H(Y|X)代表的是什么意思了。

有人有可能会问：H(X|Y)的条件平均信息量为什么不这么表示：

这是一个好问题，但是我在这里不会给出解答。我认为你已经理解了平均信息量的定义，这个问题应该很好解答。

根据感性上的定义，H(X|Y)是已知Y时X的平均信息量（平均不确定性），那么也就是说，通过接收Y，我们可以推测出一点有关X的信息。这么一来就很好理解为什么H(X)。

这个性质的严格数学证明将在后面给出。

此外，还应该满足.这是为什么，其原理我在上一节已经讲过了。

I(X;Y)称为X和Y之间的平均互信息量（mutual information measure）。它的定义是：

它和H(X,Y)正好相反，意义是通过接收X消除的有关Y的平均不确定性的大小。

  通过数学推导我们可以轻易得到：I(X;Y)=I(Y;X)。至于这到底是为什么，由于这牵涉到信道有关知识，现在只要在数学上理解它就可以了。

互信息量也应该满足I(X;Y)，这是因为所传递过来的信息必须不为负值。若要严格一点的证明，根据定义即可证明。

至此我认为第一章已经没什么可讲的了。

## 1.3 相关定理的数学证明

我把定理证明都放在这里，不是说这些是可以不看或不重要的内容。恰恰相反，这些内容非常重要，以至于我一定要把它们集中在一起！

在开始证明之前，我发现有一个不等式被大量频繁地使用。这个不等式是

当且仅当a=1时等号成立。

1）证明

证：

2）证明

证：

下面请读者自己试着证明两个等式：

# 第二章 离散信源

## 2.1离散无记忆信源的特性

这一章我们讲离散信源，这包括离散无记忆信源和离散有记忆信源。当然，在连续的情况下，会相应地有连续无记忆信源和连续有记忆信源。

首先要明确的定义是：“无记忆”是一个什么样的概念？信源发出的符号为离散符号，且每个符号之间统计独立，这样的信源就是离散无记忆信源。

相对应地，离散有记忆信源就是发出的符号为离散符号，但所出现的符号之间互相关联。

离散无记忆信源有非常好的特性，那就是符号间统计独立。实际上，在通信系统中，实际的信源传达的信息总是带有冗余。为了消除这部分冗余以减少带宽压力，我们首先就会进行data reduction，提高传输效率。进行过data reduction后的符号均应是统计独立的，这样传输的效率是最高的。有关“统计独立”的详细解释，请参考概率论与数理统计书或 <http://zh.wikipedia.org/wiki/%E7%8B%AC%E7%AB%8B%E6%80%A7> 。

离散有记忆信源，由于出现的符号总是与已经发出的符号有一定关联，因此称为“有记忆”。描述离散有记忆信源的数学工具是马尔可夫链。

有几个重要的术语需要在此强调一下：

Source alphabet:信源符号，即信源会发出的符号，用*U=*{}来表示

Message 或 word:一组由信源符号组成的码字。

Stationary symbols:出现这个词组的时候，说明每个信源符号出现的概率不随时间改变。在本科阶段，如无特别说明，所有的信源符号都是stationary的！（也是最不需要关心的一个特性）

计算机生成了可选文字: H（切
pi109勿bitslsymbol.我在前一章说过，描述离散信源的一个重要属性是平均信息量H(X)，这也正是整本书的重点。对于一个信源U，其所发送的符号集为{}那么该信源的信息熵/平均信息量/平均不确定性为

式中代表了每个符号出现的概率。

在前一章我说过平均信息量的一个重要特性：在离散信源的情况下，在达到 的条件下，平均信息量达到最大，且为 bits/symbol！（此处当然要读者自己思考。请不要翻到前面。）

计算机生成了可选文字: md=l一
maxH（仍
二l一
H（切
Iogn由此，一个描述信源优劣的指标就出现了：冗余度：

离散无记忆信源的一个重要特性就是：码字的平均信息量等于信源平均信息量乘上码字长度。导致这个现象的根源，就是信源符号间统计独立。为了理解这一特性，请务必把书打开到41页做一下例2.2。

## 2.2 简单的马尔可夫过程

描述离散有记忆信源的数学工具是马尔可夫链。

简单地说，马尔可夫链是在状态间以一定概率进行跳转的过程。我想在高中的时候我们就已经学过天气概率向量和转移矩阵，通过转移矩阵可以预测明天的天气。马尔可夫链的哲学意义是可以预知未来。

在题目中，我们会见到k阶马尔可夫链这个概念。它的意思是，当前信源产生的符号，与信源先前产生的k个符号有关。而马尔可夫链的状态转移矩阵，正是从概率上描述了这一联系。

下面我们通过详解书上一道习题来解释马尔可夫链。

计算机生成了可选文字: 3.2.Aninformationsourecwithalphabct《0,l》gcneratesasecond一order
Markovchain,which15describcdbythefollowingtransitionprobabilities:
P(0l00)=0.8,P(0lll)=0.2,
P(11的）=0.2,P(1111)=0.8,
P(010】）=0.5.P(0110)=0.5.
P(1101)==0.5,P(1110)=0.5.
Sketchthestatediagrambelongingtothischain.
Calculatetheprobabilitiesoftheslates51.
a.b

首先我们来解释P(0|00)是什么意思。

马尔可夫过程是一个与时间紧密相联系的过程。我们不妨假设当前时间为t。于是乎，在时刻t，我们将要从一个有记忆信源发出一个符号：0或1.

题目中说，这个信源是二阶的。也就是说，t时刻发出的符号，与t-2和t-1时刻发出的符号有概率转移的关系。于是P(0|00)的意义就是：

P(0|00)

t t-2 t-1

也就是说，当t-2时刻和t-1时刻所发出来的符号为00时，t时刻发出0的概率为0.8，发出1为0.2.

从转移概率表中，我们可以发现，对于t-2时刻和t-1时刻不同的符号，t时刻发出0的概率也互不相同。这个t-2时刻和t-1时刻的符号组合在一起，形成了四种状态。而所谓马尔可夫链，从这个角度上讲，就是状态之间的不断跳跃转换。

状态是一个很重要的概念，它简化了马尔可夫链的表达。我们把状态记作S，四种不同的状态分别记作状态转换图就可以表示为

计算机生成了可选文字: 夕”·8
51000
S:.01
0.5
－门卜
~～叫
0.5
S3一O
51气产
O‘、、，』〕J了了r内乙
‘飞，、尸J了尹0
,L日Ir户工‘飞，-,
‘曰J了r沪、‘飞、｝』
OJ了声l沪气‘、、、、入
'11叮气r、、，nU
S'=11
D.8

马尔可夫链只给我们提供了状态转移矩阵。但是这意义非凡。一个稳定的马尔可夫链（放心吧，本科阶段所学的马尔可夫链都是稳定的）蕴含着一个惊天大秘密：

它可以求出**在长期观察后**，四个状态出现的概率。

这也正是马尔可夫链的哲学意义：预测未来。

要求出这个状态稳定概率，我们首先假设系统已经达到稳定。这时候，针对四个状态，每个状态列出一个方程：

计算机生成了可选文字: P(0l)=P(0()）·0.2+P(10）·0.5
P(10)=P(0I）·0.5+P(11）·0.2
P(11)=P(01）·0.5+P(11）·0.8

其实只要列出三个状态的方程就可以了。因为通过把这四个线性方程组转化为矩阵后，我们可以发现它不是满秩的。总的规律是，只要列出所有状态数减一个方程就行。

另外，还有一个方程：

计算机生成了可选文字: P(00)+P(01)+P(10)+P(11)=

此四个方程组成的方程组对应的矩阵是满秩的，也就是说，可以求出四个状态稳定后的概率。

这个方程组我建议你自己解一解，因为计算还是比较麻烦的。反正也是做过的习题，答案自己对一下。

## 2.3 离散有记忆信源的特性

由于我想马尔可夫链对于大家来说是非常好理解的，所以我们将少讨论这些定义上的东西，多谈一点有助于理解什么是“有记忆”的东西。

在进行下一讨论之前我们先假设我们知道了转移概率和信源每个符号的长期概率。

首先，如何定义一个离散有记忆信源的信息熵/平均信息量？有人有可能说是

事实是：没那么简单。这里的，不是符号出现的概率。符号出现的概率总是与先前出现的k个符号相关，不是在“符号间统计独立”下的概率。如果在做题中，你求出了那么我告诉你，它是在无穷的马尔可夫过程后稳定下来的符号出现概率。

现在要说明一个离散有记忆信源的信息熵还是一件比较困难的事情。总之，为了说清楚这是怎么回事，我们先定义以下符号：表示在时刻1信源发出的符号。当然，你可以看见这是一个**随机变量**。是在时刻2，依此类推。

我们先考虑最简单的一阶马尔可夫链。通过上一节的描述，我们知道给定了状态转移概率我们就可以求出稳定后每个符号出现的概率。好，我们把这两样东西求准备好。然后听我说：

状态转移过程的信息熵定义为：

计算机生成了可选文字: 川脚
H(UZIU:)＝沉艺P(u:',u方）109P(u2/u.,）·
卜Ij二

式中，代表状态转移概率，而

这与在无记忆信源中的情况不同，时间是确定好的，被标注在下标上。

  它的物理意义，和我在第一章里讲过的一样。但是与无记忆信源的不同在于**存在时间序列**：先后顺序matters。

在**两个时间里**的联合信息熵定义为

计算机生成了可选文字: H(UI,UZ)=
P(uI,,u万）109P(l'l,,l‘万）·
．艺
，乏
‘二l,

到现在为止我们讨论的都是一阶下的情形。同样，这些也可以被推广到高阶情形。例如，如何表示

计算机生成了可选文字: H(“斌口袱l，…，UZ,UI).

稍有常识的人就会看出，一个具有马尔可夫性质的随机序列，我们知道的有关它先前发送的符号越多，我们对下面将会发送什么符号就越有信心，比如说我输入“Microsof”，你可以以99%的概率认为我下面会打的一个字母是“t”而非“v”或“x”等。可以看出信息量在不断减少。这是什么信息量？这就是状态转移过程的信息量。我这句话用数学公式表达就是：

计算机生成了可选文字: H（口冠饰－l，…，UI)SH（口枯11如－2，…，UI)‘…
…‘H(U2/Ul)5H(Ul).

同时我们也知道，无论我们对将要发送的符号是什么多么有信心，我们与真实总是存在一定距离。这句话用数学定理表达就是：

计算机生成了可选文字: 衬试仍＝
11们
柑与
lm
只试U)==!imH(U斌如刁，…，Ul)bit眺ymbol.
IV～心00

这个就是我们无论知道了多少个先前发送的序列也不能判断将要发送什么符号的那点不确定性，这，就是有记忆信源的平均信息量。你可以看出来，相同的符号集放在无记忆信源中，平均信息量会大很多。

## 2.4 信源编码基础

我们到现在为止，都是讨论的信源。在通信原理这门课里，我们可能还记得，信源在通信系统中处于怎样的一个地位。顾名思义，可以知道它是信息产生的源头。我们现在为止都是讨论的离散无记忆信源或是有记忆的信源。离散无记忆信源，正如抛掷一枚均匀或非均匀硬币的结果，或一台输出0～231-1的随机数的计算机。有记忆信源，正如我们写作时所写下去的每一个字符都与上下文相关联。

但是这样的信息，可能还要以数字化的方式处理。这时候就要对信源进行编码。

编码可以对每一个符号一个一个进行编码，也可以把信源符号几个几个组合在一起编码。我们知道ASCII编码，就是对ASCII的码字集里的128个码字用一个7位二进制数表示。这是一个一个编码的例子。我们现在的输入法很先进，可以直接输出一个词，或一个句子。那么，使用26个英文字母编码汉语中词语或句子的方法，就是几个几个组合在一起的编码方式。具体的编码方式，如拼音编码、五笔编码、郑码等。

科学家们对这种编码过程作了诸多研究，他们发现这样一个基本规律：无论信源以怎样的方式组合，单个的也好，多个一块编码也好，用于编码的码元集组成的码字个数，必须大于信源所有可能的码字个数。只有这样，才能保证编码无差错，或差错无穷小。

有人说，这简直就是废话，编码的码字必须和信源的输出一一对应了，不一一对应当然会有差错了！那么用多长的编码码字用于编码信源好呢？这个长度L可以无限长，对于二进制编码来说，长度为L的码字为2L个。这样来说，只要L的长度足够，就能保证无差错编码。

但稍有常识的人就会发现，这是个不现实的想法。L再大也要有个限度。只要够用就行。那么如何描述这个“够用”的概念呢？只要信源的信息量能全部被编码的码字所表达就够了，对不对？

我们已经知道，信源的平均信息量H(U)是信源码字的，平均每一个的，信息量，用比特为单位。这时候，我们知道编码码字集有r个（如用二进制编码就是r=2）。而编码码字的平均长度为L。可见最理想的情况下，编码码字所能带的最大信息量是L·log2r。科学家们发现，其与平均信息量所成的关系只要符合

就能保证立即可译码存在。

这个定理就叫做信源编码定理（书上46页，定理2.2）。值得一提的是，它的特别之处就在于指出了L必须保证的下限，给定了信源平均信息量和编码码字长度的关系。还值得一提的是，为了描述“够用”的含义，不等式左边没有说是信源中信息量最大的那个码字的信息量，而是平均信息量。这相当于放宽了L的筛选条件。而为什么呢？这是因为，立即可译码的充分必要条件——Kraft不等式，在这个定理的证明中起着重要作用。所以这里的编码都必须是立即可译码。

但是编码大业还没有结束。因为我们上面都是以单个符号编码说明的。对于多个信源符号一起编码的情况，我们连提都没提。现在比如说，一个离散无记忆信源输出符号，我们把输出的符号几个几个一起编码，我们会发现，如果联合的码字数越大，在每一个码字里，平均每个码元的信息量就会越接近信源信息熵。比如说，你现在抛一枚均匀硬币，抛个两三下，有可能都是正面。虽然正面反面出现的概率都是0.5，但是现在你抛得的结果就全部都是正面。那么是否有可能，在抛一万次以后，仍然全部都是正面呢？啊，从理论上说，有这种可能，但是实际上就是不会啊！在大量试验后，我们会发现出现正面的概率会无限逼近0.5.比如说，你抛了4995次正面，5005次反面，那么把这一万次的结果进行联合编码，其信息量为

把结果除以10000，可以得到近似的信源平均信息量。在这个例子里当然为1 比特/符号。可能是我的例子举得不够好，现在我抛一枚不均匀的硬币，正面概率为0.4，反面0.6。在抛了10000次后，我们可能会抛出4010次正面，5990次反面。这时候对结果进行联合编码，其结果为

结果除以10000，得到近似的信源平均信息量0.97153 bits/symbol.和按正儿八经的公式计算出来的平均信息量0.97095 bits/symbol相当接近。

这个性质很有用。有人把它总结成了一个定理，名曰“Shannon-McMillan theorem”（见书上57页。但在我看来，只不过是大数定理的推广。

我们编码的目的，正如我在前面所说，是使编码码字集对信源符号够用。这个“够用”的概念，也正是编码有效率的概念。科学家们发现，在联合编码的情况下，如果所联合的码元个数很大，编码就越有效率。

香农指出，实现无差错编码，或是指定一个无穷小误码率上限，只要编码长度满足

就能实现。其中小写l代表信源的联合编码码字长度。同样这个定理的证明，是以Shannon-McMillan theorem为基础之一的。和上面的第一个不等式比较一下，是不是很相像？

这个定理就是无差错信源编码定理，也叫香农第一定理。它给世界各地的编码专家提供了指导性意见，但并未指出如何编码。至于如何编码，请看本书第六章。

## 课外阅读：输入一个汉字需要敲多少个键——谈谈香农第一定律

注：本文摘自樊凌涛在大一暑假时给我们发的课外读物《数学之美与浪潮之巅》。原作者吴军。

今天各种汉字输入法已经很成熟了，随便挑出一种主要的输入法比十几年前最好的输入法都要快、要准。现在抛开具体的输入法，从理论上分析一下，输入汉字到底能有多快。

我们假定常用的汉字在二级国标里面，一共有 6700 个作用的汉字。如果不考虑汉字频率的分布，用键盘上的 26 个字母对汉字编码，两个字母的组合只能对 676 个汉字编码，对 6700 个汉字编码需要用三个字母的组合，即编码长度为三。当然，聪明的读者马上发现了我们可以对常见的字用较短的编码对不常见的字用较长的编码，这样平均起来每个汉字的编码长度可以缩短。我们假定每一个汉字的频率是

p1, p2, p3, ..., p6700

它们编码的长度是

L1, L2, L3, ..., L6700

那么，平均编码长度是

p1×L1 + p2×L2 + ... + p6700×L6700

香农第一定理指出：这个编码的长度的最小值是汉字的信息熵，也就是说任何输入方面不可能突破信息熵给定的极限。当然，香农第一定理是针对所有编码的，不但是汉字输入编码的。这里需要指出的是，如果我们将输入法的字库从二级国标扩展到更大的字库 GBK，由于后面不常见的字频率较短，平均编码长度比针对国标的大不了多少。让我们回忆一下汉字的信息熵，

H = -p1 \* log p1 - ... - p6700 log p6700。

我们如果对每一个字进行统计，而且不考虑上下文相关性，大致可以估算出它的值在十比特以内，当然这取决于用什么语料库来做估计。如果我们假定输入法只能用 26 个字母输入，那么每个字母可以代表 log26=

4.7 比特的信息，也就是说，输入一个汉字平均需要敲 10/4.7= 2.1 次键。

聪明的读者也许一经发现，如果我们把汉字组成词，再以词为单位统计信息熵，那么，每个汉字的平均信息熵将会减少。这样，平均输入一个字可以少敲零点几次键盘。不考虑词的上下文相关性，以词为单位统计，汉字的信息熵大约是8比特作用，也就是说，以词为单位输入一个汉字平均只需要敲 8/4.7=1.7 次键。这就是现在所有输入法都是基于词输入的内在原因。当然，如果我们再考虑上下文的相关性，对汉语建立一个基于词的统计语言模型，我们可以将每个汉字的信息熵降到 6 比特作用，这时，输入一个汉字只要敲 6/4.7=1.3 次键。如果一种输入方法能做到这一点，那么汉字的输入已经比英文快的多了。

但是，事实上没有一种输入方法接近这个效率。这里面主要有两个原因。首先，要接近信息论给的这个极限，就要对汉字的词组根据其词频进行特殊编码。事实上像王码这类的输入方法就是这么做到，只不过它们第一没有对词组统一编码，第二没有有效的语言模型。这种编码方法理论上讲有效，实际上不实用。原因有两个，第一，很难学；第二，从认知科学的角度上讲，人一心无二用，人们在没有稿子边想边写的情况下不太可能在回忆每个词复杂的编码的同时又不中断思维。我们过去在研究语言识别时做过很多用户测试，发现使用各种复杂编码输入法的人在脱稿打字时的速度只有他在看稿打字时的一半到四分之一。因此，虽然每个字平均敲键次数少，但是打键盘的速度也慢了很多，总的并不快。这也就是为什么基于拼音的简单输入法占统治地位的原因。事实上，汉语全拼的平均长度为 2.98，只要基于拼音的输入法能利用上下文彻底解决一音多字的问题，平均每个汉字输入的敲键次数应该在三次左右，每分钟输入 100 个字完全有可能达到。

另外一个不容易达到信息论极限的输入速度的原因在于，这个理论值是根据一个很多的语言模型计算出来的。在产品中，我们不可能占有用户太多的内存空间，因此各种输入方法提供给用户的是一个压缩的很厉害的语音模型，而有的输入方法为了减小内存占用，根本没有语言模型。拼音输入法的好坏关键在准确而有效的语言模型。

另一方面，由于现有输入方法离信息论给的极限还有很大的差距，汉语输入方法可提升的空间很大，会有越来越好用的输入方法不断涌现。当然，输入速度只是输入法的一项而不是唯一的衡量标准。我们也会努力把谷歌的输入法做的越来越好。大家不妨先试试现在的版本，http://tools.google.com/pinyin/，半年后再看看我们有没有提高。

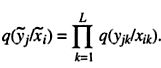
# 第三章 离散无记忆信道

## 3.1 什么是离散无记忆信道

离散无记忆信道，所传输的符号为离散符号，且传输符号的每个过程之间相互独立。这到底是什么意思呢？在数学上，码字传输的转移概率为

计算机生成了可选文字: q汤属）=q伪．,y,=，…，yj少‘.，…办

而离散无记忆信道能够保持这样的特性，使这个转移概率简单地拆分为每个符号传递的转移概率的乘积：



简单地说，若噪声独立地随机地影响每个码元，导致接收的码字中码元错误是独立随机出现的，那么这种信道就是无记忆的。

若噪声、干扰是前后相关的，错误是重复出现的，那么这种信道就是有记忆的。

有一点需要说明的是，在本科阶段，只考虑简单的情况——就是只谈“离散无记忆信道”而不谈“离散有记忆信道”。如果对我上面这一段叙述感冒的请直接无视。

在这一章里，我们把信源记作X，把信宿记作Y。信源符号集为{}，信宿符号集为。信源的符号是如何变为信宿的符号的，中间存在一个转移矩阵Q：

计算机生成了可选文字: q11qZI
q12q22
qlmqZ用…

就代表，就代表。均为条件概率。**注意这里的简写与完整表达的下标对应关系。**

我们知道信源符号出现的概率为，那么我们就可以求出信宿符号出现的概率：

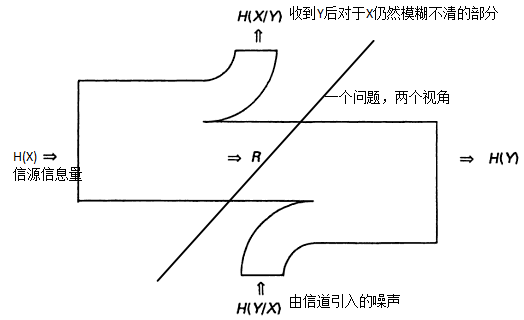
符号集从变到的过程，就是被抽象化了的通信过程。

回想一下我们在第一章里所学的信息熵/平均不确定性概念。在一个通信过程中，在遵循上面表达的前提下，H(X)代表信源的信息熵，H(X|Y)代表信宿收到信宿应该收到的符号后对信源发送的符号遗留的不确定性。这里书上有一个大词专门称呼H(X|Y)，叫equivocation。相应的动词equivocate专门描述一个人讲话含糊其辞，避实就虚。请自行脑补equivocation作为H(X|Y)等价词语的内涵。

那么，H(X)-H(X|Y)就是通信过程中清楚传递的那一部分信息。在第一章中，我们也对此做过定义：I(Y;X)，专业术语为互信息量。实际上，你可以看出来，它表示了信道能够传输的信息量大小（每符号）。于是我们在此章专门有一个符号R来表示互信息量。R=I(Y;X)。单位显然是bits/symbol.但是有时候也会出现bits/second,这时候往往是规定了信元传输速率的时候。当题目中给定了信源符号传输速率的时候，请各位务必注意要把bits/symbol转换成bits/second。

以信宿为中心观察，H(Y)表示信宿收到符号应该带来的信息量。同时H(Y|X)表示X发出后Y还参与的不确定性。书上有一个词称呼H(Y|X)，名曰irrelevance。值得注意的是，这一部分“不相关性”是由信道的噪声带来的。

这四个量之间的关系可以用下图表示：



显然是由信道噪声引起。

在数学上我们已经证明

计算机生成了可选文字: I(X；均＝H（均一H（竹幻
=H(x）一H(XI均．

实际上这就是以信道传输量R为核心的两个视角。

实际上，本章主要讨论的问题是信道容量。有人说R已经代表了信道传输的能力。但是，通过解析R的表达式可以看出，只要知道转移概率和信源符号概率分布就可以知道R的大小。

现在请读者试着解析R的表达式，证明R只与信道转移概率和信源符号概率有关。

## 3.2 对称信道

表征信道特征的信道转移矩阵，如果每行每列的元素都相同，那么称该信道为对称信道。在英文中，这样的信道是symmetric的。

简单的对称信道如。左边这个信道还叫做binary symmetric channel，BSC。

如果你要求这个信道的信道容量，我想书上已经说了一大堆了，不熟的同学可以再看看例4.3和定义4.2.我这里想讨论一下对称信道的特性。那么究竟是什么特性使得对称信道如此迷人呢？

我在上一节的末尾说过，确定一个信道的信道容量，只需要知道信源的符号概率分布和信道转移概率（矩阵）。但是对一个对称信道，只需要知道信道转移概率矩阵就可以确定其信道容量。

还有一个特性，那就是，如果输入的符号均为等概分布，则输出的符号也为等概分布。

下面让我们来推导这个重大性质。现有对称信道Q：

Q具有如下两个重要性质：

稍有常识的人就会看出，这是把Q的每行每列相加的必然结果。

上面这个推导说明了对称信道条件熵与信源统计特性无关，仅由信道特性决定。这样，在求一个对称信道信道容量的时候，

这个定理在书上4.2也有证明。

至于对称信道输入符号等概则输出符号也等概这一现象，可以这样证明：

## 3.3 信道级联

在通信系统中往往会出现信道级联的现象。

计算机生成了可选文字: R
一一．一．一一．..............．目．.．心卜
X
R1
Rz

R，是信道传输速率。级联的信道满足R.

需要明确的是，这样的信道并不简单地存在H(Z|X)=H(Y|X)+H(Z|Y)的关系。这句话的意思是，有人可能认为，由第一个信道引起的irrelevance和第二个信道引起的irrelevance加起来就是全部irrelevance。实际上这是错误的。要计算全部信道的irrelevance或equivocation，必须先算出整个信道的信道转移矩阵。但是，整个信道的irrelevance，不会低于第一个信道的irrelevance。

如果要求整个一个新信道的信道转移概率，则计算矩阵即可。

相关的例题是例4.4，习题是4.5。

## 3.4 信道编码基础

所谓信道编码，就是给经过信源编码的码字加上冗余，形成差错控制，以便在接收端能完整恢复出原始的信源编码。

比如说，一个被信源编码为1和0两个码字的信源，现在要进行传输。但是由于1有可能会被传输为0，0有可能会被传输为1.为了大概率避免这种悲剧发生，有一种原始的方法：在信道编码模块中，把1编码为111111111，把0编码为000000000.在接收端，如果接收的结果是111111101，那么就判定为1，101010101，那么就判定为1，如果是010101010，那么就判定为0.这就叫择多判决逻辑。

择多判决逻辑是一种比较没有技术含量的、原始的差错控制方法。科学家们对差错控制编码进行研究，开发出了各种不同的编码方式，把编码分成线型码与非线型码、分组码与卷积码、系统码与非系统码。至于如何编码，其具体方式在第七章讲述了一个卷积码。但总而言之不变的是，通过一些精巧的方法加入额外的码字，就可以实现差错控制。而且加得越多，错误的概率就越低。

现在有一个概念“编码速率”，或简称“码率”，用来描述这种概念。简单地说，码率就是信道编码时有效信息位与全部编码长度之比。码率越低，就意味着所加入的冗余码字越多，可靠性越高；码率越高，加入的冗余码字越少，可靠性越低。

香农的信道编码定理（也叫香农第二定理）指出：对于一个给定的有干扰信道，如信道容量为C，只要发送端以低于C的速率R发送信息(R为编码器输入的二元码元速率)，则一定存在一种编码方法，使编码错误概率pe随着码长n的增加，按指数下降到任意小的值。这就是说，可以通过编码使通信过程实际上不发生错误，或者使错误控制在允许的数值之下。

这个香农信道编码定理指出，对于传输速率小于信道容量的信源，总存在一种信道编码方法，使得差错概率降到无穷小——大不了就是无限增大冗余码字，牺牲码率罢了。同时若是超过了信道容量这个界限，那么将不可避免地出现误码。

所要注意的是，香农只是从理论上指出信道容量C是一个界限，至于如何信道编码以达到这个极限，还需要编码专家研究各种不同的编码方法。

# 第四章 连续信源和连续信道

## 4.1 连续信源的基本特性

我们到现在为止，已经在离散信源和离散信道上面花了不少时间。我想，只要搞懂了离散情况，就很容易推广到连续情况。

离散与连续最主要的不同，就是离散情况下使用概率分布来描述符号出现的概率。但是连续情况下有无数个可能的值。这时候就要使用概率密度函数来描述连续情况下的信息量。

连续信源的信息熵定义和离散信源的信息熵定义本质上有相通之处：

于是，把连续情形中的条件信息熵、互信息熵、联合信息熵的定义都与离散情形下对应起来，就很方便地得到他们的公式。（我这里省了一大段话，因为正如上文所说，连续情况是离散情况的对应）

在第一章中，我说过，离散信源的信息熵满足这两个性质：

但是在连续信源中，这两个性质是否还成立呢？

第一个性质，在离散信源中，我曾经说过，由于0p(x)，所以，故H(X)在离散信源下大于等于0.

但是在连续信源中无法保证这一点，因为在一个连续区间中，密度函数p(x)有可能大于1.所以，有同学在连续情况下求出的信息熵如果小于0而且按照公式反复对照检查无误，结果有可能是正确的。

第二个性质，说明了在离散情形下什么时候信息熵达到最大。当然，在那里，我们曾经说过，当各个符号出现的概率均相等时也就说明了符号的出现无倾向性。这时候信息熵达到最大。那么，在连续世界里，是不是也是这样呢？即连续信源服从均匀分布，信息熵达到最大？事实告诉我们，并非如此。

答案是这样的：信号幅值受限时，服从均匀分布信息熵最大；信号功率恒定时，服从正态分布信息熵最大；信号期望恒定时，服从指数分布，信息熵达到最大。

下面我们就幅值受限情形做详细证明，就功率恒定情形做引导性证明，就期望恒定情形请读者自己证明。

1）幅值受限

幅值受限在数学上的表达是。记住，这是约束条件。

我们的目标就是求出在满足这个约束条件下的最大值，顺便也把p(x)的形态求出来。

对于这种问题，高数下在多元微分一章中阐述过拉格朗日乘数法。求函数在条件下的极值点，可以按以下方法进行：

构造拉格朗日函数。

求对的偏导数，并建立方程组：

解此方程组得到的极值点就是可能的极值点。

拉格朗日极值法可以推广到多于两个变量的多元函数以及约束条件多于一个的情形。

对于求幅值受限下H(X)最大值的问题，亦可利用此法。作拉格朗日函数：

可以看见，这个拉格朗日函数已经去掉了积分符号。下面要对此微分，不过注意，是对p(x)微分。

由此可以解出（k为常数）。实际上，这已经确定了为常数。

再根据，可以立即得出。此时H(X)=。

2）功率受限

在功率受限的条件下，限制条件有两个：

于是拉格朗日函数可以写为：

。注意这里有两个限制条件。

对p(x)微分后，可得 。

由此求出p(x)，注意其中哪个是常量，哪个是变量。

将求得的p(x)代入限制条件，可解出p(x)= 。

对应的H(X)= bits/symbol.

3）期望受限

在期望受限的条件下，限制条件有两个：

解出来对应的H(X)=。X服从指数分布。

## 4.2 连续信源的独立性

虽然通过理解离散信源的独立性我们也可以推知连续信源的独立性如何表达，如H(X|Y)=H(X)，I(X;Y)等。但是我们有时候会见到一些莫名其妙的表达式，如

这里的看上去应该是随机变量的取值，那么是个什么东西呢？它应该是同一个信源里的随机变量，那么同一个信源为什么还要用多个随机变量呢？

实际上，那些下标与时间有关。在采样时刻1得到的值就是随机变量，在采样时刻2得到的值就是随机变量……以此类推。

在邻近的时间上采样，随机变量之间会有很大的相关性，于是这时候相关性的研究就有了意义。

若是有人明确指出，这个信源是无记忆信源，那么就有以下关系：

## 4.3 计算时的注意事项

连续信源这一章没什么好讲的，因为几乎所有的习题（当然，除了我上一节讲的证明题）都是套公式。

有一点需要指出的是，记住一些常见的积分结果有助于加快解题速度。这些结果包括但不限于：

为了保证您能拿到高分，请务必自己做一做书上的例题和习题，以熟悉过程。

## 4.4 连续信道

连续信道只考虑很特殊的情况，因为普遍的情况都十分复杂。也就是说，这一章也十分之简单。

这一节的信道只考虑加性信道。也就是说，噪声在信道中被加入，并与输入的符号统计独立。

在数学上把我这句话表达出来，就是

这样做的直接结果，就是H(Y|X)=H(N)

实际上我认为只要理解那三行推导是什么意思这一章就完成了。我在这里也实在想不出来还要讲些什么。不过为了让您理解“加性”的意思，请做一遍例6.1。

# 第五章 率失真理论

## 5.1 平均失真度

本章所讲内容均以离散无记忆信源和离散无记忆信道为基础。

在本章里，信源符号集为，信宿符号集为。稍后如果有兴致，我会解释为什么会这样。

描述一个信道，在前面相关章节里我已经阐述过，用的是转移矩阵：

转移矩阵的每一个元素都是信源符号转移成信宿符号的概率。

现在，有一个失真矩阵，它与转移矩阵相同大小。它用来描述信源符号被转移成错误的信宿符号的危害性。**通常**（也就是说还有例外的情况，以题目为准），如果信源符号被转移成正确的信宿符号，则率失真矩阵里的相应元素为0.

这个失真矩阵为：

而平均失真度，就是描述这个信道平均传输错符号带来的危害性。这个函数记为

计算机生成了可选文字: d(Q)＝艺艺，U),(klJ)pU,k）·
jk

通常在题目中，失真矩阵是给定的，信源也是给定，但是信道是变化的——信道转移概率矩阵是未知的。于是我们把所有满足的信道转移矩阵集合记为。

## 5.2 率失真函数

假设我们已经给定了平均失真度的上限D，相应的也已经确定。那么率失真函数定义为：

率失真函数是保证信号恢复时平均失真度小于预定失真上限D所需的最小平均互信息量。

平均互信息量的定义是，由于信源给定，变量只有代表信道的转移概率。而信道实际上又受到D的牵制，所以率失真函数实际上变成了D的函数。

那么研究率失真函数的意义到底是什么呢？我们知道，失真是由各种各样的因素造成的，但有时候失真很有用处。比如说，在数据压缩的时候，其过程就是通过信道加入失真使得信宿的信息熵减少，从而达到压缩效果。科学家的任务，就是要在给定的平均失真度上限D下，研究出压缩编码。率失真函数指出在这个D下的最小传输速率。在低于这个最小传输速率的界限之下，压缩编码是不可实现的。

压缩的过程在书中用reproduce表达，信宿的码元符号叫做reproducing alphabet.

## 5.3 确定D的定义域

确定D的定义域，本质上就是要确定和。

1）如何确定

在题目中，信源概率和失真矩阵是给定的，未知的是转移矩阵。在知道这一点后，我们来开始写的表达式：

所求的为的最小值。那么这个最小值怎么求呢？很简单。因为我们已知，所以我们在对应的j值下，找出最小的。也就是在失真矩阵的每一行找到那一行的最小值。

然后，强制命令那一行最小值对应的转移概率为1，其余的都为零。

这样一来，的结果就是第j行失真矩阵的最小值。

有的时候，在第j行失真矩阵的最小值不止一个，那么没有关系，只要强令那些最小值对应的转移概率之和为零就能达到同样的结果。

然后的结果就是

这时候有一点需要说明，由于很多典型的失真矩阵对正确传输的符号的失真值均为0，所以大多数情况下，就是0.

2）如何确定

在D=时，=0——率失真函数如是指示。这就意味着，信源符号与信宿符号之间没有任何关联。用数学语言表示，那就是。

现在所求的，就是在满足的情况下的最小平均失真度。

式中，中括号里的是可以确定下来的值。它的意思就是将信源概率向量与失真矩阵每一列的向量进行内积（不清楚的，可以自己画画看），得到k个值。

在这k个值中，选出最小的值，并强令此时对应的q(k)=1,其余的等于0.

有的时候，这k个值中最小的值不止一个。此时只要将相应的q(k)之和等于1，其余的等于0就可以了。

这样的出来的结果。

值得一提的是，由于我们在确定和的时候已经对转移概率加以了强限制，很多时候也就可以求出相应的转移概率矩阵。关于这一点，书上的例7.3作了很好的说明。也请大家用此例题检验我上面所说的话。

# 第六章 信源编码

## 6.1 Fano编码

Fano编码的编码步骤为：

1. 对信源符号在概率上从大到小排序。
2. 根据给定的编码符号个数r，在概率上把信源符号尽可能等分成r组。
3. 每一组加上相应的编码符号。
4. 若这一组只有一个符号，那么这一组分完；若下一组还有若干符号，则重复步骤2.

有可能我说话表达不清楚，但是通过下面这道例题我将尽可能详细解释这是怎么一回事。

计算机生成了可选文字: 公训叩Ie2.10
symbolProbabilitycodel(r==2)codeZ(r=3)codc3(r=4)
3025121010050404
00000000
uluZu3u4usu6幻u8

这是书上50页的例2.10.信源的符号已经依概率从大到小排序排好了。

r=2的时候，就是要以0和1两个符号对信源进行编码。这时候应尽可能地把信源分成概率为0.5和0.5两大块。中间不可以有跳跃。显而易见，u1和u2分在一起时能尽可能满足要求。于是第一步使得编码变成了

计算机生成了可选文字: mbab
codeZ(
001111111
3025121010050404
00000000
uluZu3u4usu6幻u8

现在，第一组有两个符号，还没分完。第二组还有6个符号，还没分完。那么第一组再尽可能地二等分，即u1为一组，u2为一组。第二组的六个符号要尽可能地二等分，即u3u4为一组（概率和为0.22），u5-u8为一组（概率和为0.23）。此时编码变成了

计算机生成了可选文字: 乙扭仰le2.10
symbolprobabilitycodcl(r=2)codeZ(r=3)codc3(r==4)
u10·30吸
uZo·25处
u30.12IQ
u40·101旦
us0.1011!
u60.0511
u,0.0411
us0.0411

此时再看先前的两组，已经只分到一个符号，就不再分下去了，也就是说u1u2编码完成。至于剩下的工作，我想你也应该懂怎么做了。

## 6.2 Shannon编码

由于Shannon编码在例2.12和习题2.12的解答之间存在冲突，在此不作介绍。给您带来不便深表抱歉。（不过我想这个编码是一个不很流行的编码）

## 6.3霍夫曼编码

首先有一个好消息告诉大家：霍夫曼编码是最有效率的不等长编码（撒花～

霍夫曼编码的实现步骤：

首先，假设有L个信源符号，而编码符号有D个。并且DL。

1. 把L个信源符号按照概率从大到小排列。
2. 用D个编码符号标注概率最小的D个信源符号。然后合并这D个最小的信源符号合成一个新的信源符号，从而得到一个包含L-D+1个符号的新信源集（可以自己试试）
3. 把新信源集的符号按照概率大小，从大到小排列。然后再用D个编码符号标注最小的D个新信源符号，并且合并之，从而得到一个包含L-2(D-1)个符号的新信源集。
4. 依照上述方法不断进行信源缩减。若经过a次缩减，信源的符号刚好剩下D个，则用D个编码符号标注着最后D个信源符号。
5. 从最后一级缩减开始，沿码字符号授予的路线，向前返回，即可得到信源个符号对应的码字，完成编码过程。
6. 若经a次缩减后的信源的符号数小于D，则应在原有信源符号的基础上增加若干概率为0的虚假信源符号。

我一开始也没看懂这是什么一个过程。不过通过一道例题的叙述，我想您也将很快明白这是什么一个过程。

例：信源S的符号空间和概率场为：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |
|  | 0.24 | 0.20 | 0.18 | 0.16 | 0.14 | 0.08 |

试用码元集{0,1,2}对S进行编码。

解：首先估计是否要增加虚假符号。6-a(3-1)=2<D,故需要增加一个虚假符号，使得最后信源符号能够刚好分完。

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 码字 | 信源符号 | 概率 |  |  |  |  |  |
|  |  |  |  |  |  | 0.54 | 0 |
| 1 |  | 0.24 |  | 0.24 |  | 0.24 | 1 |
|  |  |  |  | 0.22 |  | 0.22 | 2 |
| 00 |  | 0.20 |  | 0.20 | 0 |  |  |
| 01 |  | 0.18 |  | 0.18 | 1 |  |  |
| 02 |  | 0.16 |  | 0.16 | 2 |  |  |
| 20 |  | 0.14 | 0 |  |  |  |  |
| 21 |  | 0.08 | 1 |  |  |  |  |
| 22 |  | 0 | 2 |  |  |  |  |

大家只要沿着线段从右往左看，就能发现规律。

现在可以做一下书上的练习和习题，以试验一下效果如何。

有人问过我这么一个问题：如果合并后的新信源符号与原信源集的一些符号概率相同，那么这个新信源符号是放在概率相同的符号的上面还是下面？深入的研究表明，这样的编码结果的确不同，但是方法都是正确的。编码的问题本来就是多解的。放在上面还是下面依照个人喜好决定。但是，在一道题目中不要这次合并的时候放在上面、那次合并的时候放在下面……

## 6.4编码效率

我们至今为止已经讨论了怎么编码，但是对编码背后的种种理论没有加以研究。科学家们研究出各种编码方法的目的就是提高编码效率。

信源编码定理指出，编码的平均长度与信源信息熵有着密切联系：

平均码长L是由码字长度与出现概率加权而得。

编码效率定义为

这里，的r是指编码符号个数。而，我们前面讲过，就是一个r个符号的离散信源符号等概率分布时的平均信息量。这时候信源的平均信息量是最大的。而前面乘上一个L，代表L个编码符号组成的码字最大平均信息量。实际上，也就是编码符号的最大编码能力。如果信源的平均信息量越接近这个极限值，则说明编码效率越高。

# 第七章 卷积码

## 7.1 卷积码编码器

下图是一个常见的教学用卷积码编码器（说是“教学用”是因为它已经够简单的了）：

计算机生成了可选文字: UlFi傲coded't
InputDatabits
m-
Outputcodedbits
UZsecondcodedbit

卷积码把一连串的序列流变成一个单独的码字.

一个卷积码编码器有k, n, K三大属性：

1. k为每个时刻输入的比特流个数，通常为1.
2. n代表输出端的个数。上图为2.
3. K代表寄存器个数。上图为3.

上图这个卷积码编码器可以被表示成（2，1）卷积码编码器，其限制长度为3.而且，这个编码器的**码率**为

我们现在关心的问题是，卷积码是如何编出来的。注意到图中这个符号代表“异或”。在每个时刻，输出端u1会对三个寄存器的结果进行异或然后输出，输出端u2会对第一个和第三个寄存器的结果进行异或然后输出。喵

在一开始的时候，寄存器全为0。这时候我们打算对比特流“101”进行编码。

当输入为1时，因此此时输出为“11”。计算机生成了可选文字: 一一协｝0}O
一一扭｝0}0
t1me坛：(ul,uZ)=(1,1)time坛：(ul，处）=(o,0)
m=(101)
U=(1110001011)计算机生成了可选文字: 1
O}O
0}1}O
t1me
tl:(ul,uZ)=(1,1)
t1me
如：(ul，吻）=(1,0)
1}0
＼心
到丫＼一
一一补｝0}1}0
丫一
公
!
t1me
几：(ul,uZ)=(O,O)
t1me
t4:(ul，处）=(1,0)
J、O厂

以上就是所有的编码过程。（实际上时刻2-5就同理而省略了）

## 7.2 卷积码之所以称为卷积码

在看懂了上一节卷积码编码器是怎样编码之后，有人不免会抱怨整个编码过程太麻烦，对应于不同的输入要画出这5个过程。但是，卷积码编码器有一个很好的性质，能帮助我们快速编码。

观察到每一个输出是对寄存器内数字的异或运算。如果有人还熟悉数字滤波器的基本构造，我想她/他就会发现这实际上是一个数字滤波器。

然后这代表什么意思呢？这说明卷积码编码器实际上是一个LTI系统。LTI系统之所以能够深入分析的原因之一在于该类系统具有叠加性质。这样，如果能够将LTI系统的输入用一组基本信号的线性组合来表示，就可以根据该系统对这些基本信号的响应，然后利用叠加性质求得整个系统的输出。

单位冲激函数的重要特性之一就是一般信号都可以表示为延迟冲激的线性组合。这个事实，再与叠加性和时不变性结合起来，就能够用LTI的单位冲激响应来完全表征一个LTI系统特性。这样一种表示在离散系统下叫做卷积和，在分析LTI系统时提供了极大的方便性。

下面我们对上一阶的卷积码编码器输入“单位冲激”“100”：

计算机生成了可选文字: 1}O}0
t1me
ti:(ul,uZ)=(1,1)

计算机生成了可选文字: 一一知｝0}1}0
t1metZ:(ul,uZ)=(1,o)

计算机生成了可选文字: 0}0
t1met3:
(ul,uZ)=(l,1)

这时候可以看出它的编码结果是（11 10 11 00 00），但是这不是重点。我们现在观察到，u1的输出序列是11100，u2的输出序列是10100.

也就是说，u1的单位冲激响应是11100，u2的单位冲激响应是10100.那么如何把这个结果用在编码当中呢？

还是以101为例。我们注意到101可以拆分为时刻1的单位冲激输入和时刻3开始的的单位冲激输入。这个在时间序列上就意味着：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 1 | 1 | 1 | 0 | 0 |
| + |  |  | 1 | 1 | 1 |
|  | 1 | 1 | 0 | 1 | 1 |

表中第三行是对第一行和第二行的“模二加”（实际上为异或），也就是u1的输出序列。大家可以翻到第一节验证一下。

同理，u2也是一个LTI系统，按照上面的方法计算可以得到

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 1 | 0 | 1 | 0 | 0 |
| + |  |  | 1 | 0 | 1 |
|  | 1 | 0 | 0 | 0 | 1 |

即u2的输出结果是10001.大家可以翻到第一节去验证一下。

在书上也有相关的示意。如果觉得我在此处叙述比较模糊，可以翻开到6.2.1.1。

也就是说，我们只要知道在卷积码编码器的“单位冲激响应”，就可以得到任何输入的结果。

## 7.3 论多项式

我在这里要告诉大家一种新型表示码字的方式，这是用多项式来表示码字。

假设，我们现在有一个n位码字。其相应的码多项式为

比如说，10001这个码的码多项式是

还有一个非常重要的内容：所有的对码多项式的运算（加、减、乘、除）都是在模二规则之下运作。具体来说就是：

加运算：0+0=0 0+1=1 1+0=1 1+1=0

乘运算：0\*0=0 0\*1=0 1\*0=0 1\*1=1

实际上加运算就是异或运算，乘运算就是与运算。**加运算和乘运算都没有进位退位之说！**

然后我们将这个多项式运用到上一节所讲的卷积码编码器的“单位冲激响应”。在上一节，u1的“单位冲激响应”就是111，用码多项式表示就是。u2的“单位冲激响应”就是101，用码多项式表示就是

由于这两个码多项式在上一节中有着重要的作用，我们把它们叫做“生成多项式”。记作：

现在我告诉你：**多项式乘多项式有着时域卷积的功能**。现在，我们在此输入比特序列101.这个比特序列的码多项式是。回想起LTI系统中求任意输入的输出时，我们会把输入和单位冲激响应做卷积运算。这里我们只要把输入的码多项式和生成多项式分别相乘，就可以得到每个输出端的输出结果。

u1输出端：

（请各位自己试着乘一乘，以搞懂我上面说的“模二规则”是怎么一回事）

u2输出端：

这分别是两个输出端的输出结果。大家可以与上一节的计算结果相比较，看是否相符。

## 7.4 状态转换图与篱笆图

一个编码器的“状态”，指的是编码器的寄存器靠右K-1位的值。

在开始讲下去之前，我建议你把该编码器所有的输入对应的输出都写出来，写成一张表。

现在我们研究卷积码编码器的“状态转换”。

计算机生成了可选文字: 0j00
11
、～~~~
瑞
口
1/11,'
沪沪
．口，..-.....
1/00
,2
l0
操
0/10
0/01
凡n
、
1/or、
11,
,
...．曰．
‘、
I吐．

如图所示，这个就是那个编码器的状态转换图。在斜杠/左边的，是此时此刻输入的值，也就是左边第一个寄存器的值。

不同的输入会有不同的输出，这些在斜杠/右边明确地写了出来。100的输出是11，这个是在第一节就已经明确了的。

箭头，代表在下一个时刻，状态会变成什么样子。比如说，现在寄存器里的值是100.在下一时刻，寄存器右移，变成了?10。这个?代表此时的新输入的值，可能是0，也有可能是1.因此，你会发现，每个状态总是有两个输出箭头，代表新输入的值是0还是1.

由此可见，用状态图表示能够展现状态的迁移过程，并能把所有的编码信息展示出来。

现在讨论另一种通过状态转换表示编码过程的方法。篱笆图：

计算机生成了可选文字: State
S0=
又＝=
0/00
、1月以
0
,FO『
二．
n卜UO
0l
S1=0lr
八产
、
}+
口口工T矛落
、．一
及＝11
U全介
、
、
口口口口口口
心

篱笆图的优势就是在时间上把编码过程展现出来。在上图中，每一个点代表一个状态，从每一个点出发有两条线，而决定是向哪条线发展到下一个方向的因素，是输入信号为0还是1.

和状态图一样，斜杠左边写明了输入，斜杠右边写明了本次状态下的输出。

为了展示清楚，我们这次再次对该编码器输入序列101.编码器完整的Trellis图为：

计算机生成了可选文字: Examplesequence:
几jlbits
玩putbits一一一一一一一一上坐巴竺一一一一
!l}o}i!o}o…
outPutbits
0100
0/00
0/00
0/00
0/00
卜Ull
、Ull
、创11
、Ull
、Ull
1、二入
1月0'
[l、〕久
,1月0、
1、士．
1/00
沪、
汤
，尹
，汤尹、、
尹、
1毋1
井’(\
1毋1
少狱
分二、、
1毋1
少狱
，》沪
尹、
1/.01
火
1/.01
t,
t2
几
t4
t,
t6

沿着10100000000000...的顺序输入，只突出编码器经过的路径，其结果就是这样：

计算机生成了可选文字: 乃jlbits
Invutbits一一一一一一一止三些些二一一一一
…，…”}1}0}0…
OutPutblts
像、
、Ull
0110
.，久．卿1.
,,，功”\\,_/

关于这一部分内容，书上6.2.2节已经做了详细解释。

# 参考文献

[1] Jan C a Van Der Lubbe 著，Information Theory，Cambridge University Press，1997

[2] 冯穗力 等编著，《数字通信原理》，电子工业出版社，2012