情報理論

Information Theory

2012 年後期

（改訂版）

平中幸雄

（図）三浦信一

【授業概要】

・テーマ

「情報の量」と「情報の符号化」についての基礎を学ぶ.

・ねらい

伝送や記録では、データ量をできるだけ減らしたい．一方，通信においては伝送エラーの可能性があり，伝送誤りをできるだけ小さくするための符号が必要となる．これらの課題に対する基本的な考え方と方法を理解する.

・目標

情報量，平均情報量，情報源モデル，情報源符号化，通信路容量，通信路符号化などの重要な概念について理解しておりそれらについて計算や説明ができること.

・キーワード

情報量，平均情報量，相互情報量，情報源，符号化，復号化，通信路

【授業計画】

・授業の方法

講義と２週に１回の小テストで学習を進めていく。

・日程

第１週 授業方法の説明，情報理論とは，ハートレーの情報量
第２週 平均情報量
第３週 情報量の性質，結合平均情報量
第４週 条件付き平均情報量，相互情報量
第５週 情報源モデル，状態遷移確率
第６週 情報源符号化
第７週 情報源符号化定理
第８週 情報源符号法
第９週 通信路モデル
第 10 週 通信路容量
第 11 週 通信路符号
第 12 週 通信路符号化定理
第 13 週 線形符号
第 14 週 エラー検査と訂正
第 15 週 期末試験 の予定.

【学習の方法】

・受講のあり方

授業では要点もしくは分かりにくい点のみ説明するので，事前準備としての予習と充分な理解を得るための復習を必ず行うこと.

・予習のあり方

テキストの該当箇所を読んでおくこと

・復習のあり方

テキストの練習問題に取り組むこと.

【成績評価の方法】

・成績評価基準

総合点で 60 点以上を合格とする．３分の２以上の出席が成績評価の条件.

・方法

小テスト 40 点，期末試験 60 点の配点で評価する.

【テキスト】

テキストを生協販売の予定.

【参考書】

平田廣則，情報理論のエッセンス，昭晃堂，2003.

大石進一，例にもとづく情報理論入門，講談社，1993.

吉田裕亮，情報理論入門，サイエンス社，2009.

南敏：情報理論 第 2 版，産業図書, 1993.

磯道義典，情報理論，コロナ社，1980.

シャノン，ウィーバ(植松訳)，通信の数学的理論，筑摩書房，2009.

韓太舜，小林欣吾著：情報と符号化の数理，培風館，1999.

白木善尚編，情報理論，オーム社，2008.

C.E.Shannon, A Mathematical Theory of Communication, Bell System Technical Journal, vol.25, pp.379-423,623-656, 1948.

【科目の位置付け】

学生便覧の学習・教育目標参照．情報通信の基礎であり，「確率統計学」「マルチメディア入門」「情報通信」「情報計画工学」「暗号とセキュリティ」「情報ネットワーク工学」などと深い関連を有する.

【その他】

・オフィス・アワー

水曜日 1700−1800，9−508

目次

1　情報理論とは

「情報」の量的扱いを数学的に考えた結果、生み出されたのが情報理論である。例えば「情報をデータで表現するときに、データをどこまで少なくできるか」を考えるとき、短縮の限界値があるとすると、それを決めたい（その値が情報量ということになる)。また、実際の通信路は情報量で計算すると、どこまで理論伝送能力があるか求めたい（求まったものが通信容量である)。さらに、伝送中に誤りが発生する可能性があるが、この誤りを訂正しながら、伝送能力を最大にするにはどうしたらよいか（これは通信符号を考えるということになる)。これらに理論的な答えを提供するのが情報理論である。

1A 短く表現したい

肯定か否定のいずれかを求められたとき、「はい」「いいえ」、「yes」、「no」、「○」、「×」などいろんな答え方があって、どれでも同じ情報を伝えている（答え方にニュアンスの違いはないとして)。そして、1 ビットの「1」、「0」で答えるとデータ量が一番少なくなることを諸君は知っているだろう。これは、同じ情報でも、それを表す方法はさまざまで、表現方法によってはデータ量が違うことを示している。

1B 文字種と情報量

上の例では、ある情報が 2 種類のいずれかで表せるなら 1 ビットで表せることを意味する。もし 4 種類なら 2 ビットで表せる。一般的に n 種類だと、$\log_2 n$ ビット(bit)で表せる。例えば、文字で表される情報では、その文字の種類数に依存して、以下のようになる。

1 文字で表される情報量 = $\log_2$[文字の種類数]　（ビット）

情報量の値が整数でないときの考え方は後に説明するが、実際に情報を表すときはビットを整数個使うしかないから、切り上げたビット数が必要な量（情報量以上の値）となる。尚、log の底を 2 でなく、10 にしたときは $\log_{10} n$ デシット（decit)、自然対数の底 e にしたときは $\log_e n$ ナット（nat）と呼ぶ。

2 種類	1 ビット表現	4種類	2 ビット表現
a	0	a	00
b	1	b	01
		c	10
		d	11

	2^m種類	mビット表現
		m
	a	00…00
	b	00…01
	c	00…10
2^m	d	00…11
	・	・
	・	・
	・	11…11

1C ハートレー(Hartley)の情報量

2 文字で表した情報は 1 文字で表した情報の 2 倍の情報量であると考えると、文字数に比例して情報は増えることになる。1 文字で表される情報量 $\log_2 n$ に文字数を乗じたものが文字列全体の情報量となる。これをハートレーの情報量と呼ぶ。

ハートレーの情報量 $= [\text{文字数}] \times \log_2 [\text{文字の種類数}]$　（ビット）

1D 1 ビットより少ない情報量にはどういう意味があるか

1 文字あたりのハートレーの情報量は、文字の種類数 n がちょうど 2 のべき乗でないとき、$\log_2 n$ ビットには少数部分があることになる。例えば、文字の種類数が 3 のとき $\log_2 3 \fallingdotseq 1.585$ ビットである。1 ビットで表せないが、2 ビットで表すより少ない情報であることを示している。

この情報を 2 ついっしょにして表すことを考えると、3 種類からの選択を 2 回繰り返すことになるので、9 種類からの選択となり、合計 9 種類あるので情報量は $\log_2 9 \fallingdotseq 3.17$ ビットとなり、3 ビットより少し多い。

1E 使わないものを加えると情報量が増える？

ハートレーの情報量は、情報として伝える「文字」の種類の数で決まる。すると、「カナ文字の種類数は現在 46 種類であるが、五十音と呼ばれるから 50 種類として計算する」と同じ文章でも情報量が違ってくることになる。文字の種類数をどう想定するかによって、情報量が変化するのでは定義として困る。この問題はシャノン（Shannon）が解決して、情報理論を作り上げた。

1F 有効数字について

有効数字 n 桁の値と有効数字 n 桁の値の乗除の結果は、有効数字 n 桁とするのが基本である。途中結果は一桁多い n+1 桁まで求め、四捨五入して有効数字 n 桁を結果とする。たとえば、4.8 / 3.0 = 1.6 であるが、4.8 は 4.75 から 4.8499…を四捨五入した値で、3.0 は 2.95 から 3.0499…を四捨五入して、それぞれ有効数字 2 桁の値とすると、本当の計算結果は、4.75 / 3.0499..≒1.557 から 4.8499… / 2.95. ≒1.644 の範囲の可能性がある。3 桁目はすべての値(5 から 4)をとりうるので、その値を結果に残す意味がないが、3 桁目の四捨五入して、1.6 と有効数字 2 桁を結果とするのは意味がある。場合によって、2 桁目も変化する場合があるが、変化範囲は 3 桁目ほどではなく、情報として意味のある数字になる。

練習問題

1-1　$\log_2 3$ を有効数字 3 桁で求めよ。ただし、$\log_{10} 2 = 0.3010$、$\log_{10} 3 = 0.4771$ とする。

1-2　情報を英語のアルファベット 26 文字で表すときの、1 文字あたりのハートレーの情報量を有効数字 3 桁で求めよ。ただし、$\log_{10} 13 = 1.1139$ とする。

1-3　ドイツ語のアルファベット 30 文字（英語のアルファベットにａｏｕのウムラウト付きとβの 4 文字を加える）で表すときの、1 文字あたりのハートレーの情報量を有効数字 3 桁で求めよ。

1-4　数字（0 から 9）ですべての情報を表すとき、一つの数字は何ビットの（ハートレーの）情報量を持つと考えられるか。

log の計算

$$\log_2 x = \frac{\log_{10} x}{\log_{10} 2} \approx \frac{\log_2 x}{0.3010}$$

$$\log_2 xy = \log_2 x + \log_2 y$$

$$\log_2 x^a = a \log_2 x$$

$$\log_x y = \frac{1}{\log_y x}$$

$\log_2 n$ は 2 進数表現の桁数（ビット数)、たとえば n=8（0 から 7 まで）だと 3 桁（3 ビット）

$\log_{10} n$ は 10 進数表現での桁数、たとえば n=1000（0 から 999 まで）だと 3 桁

2　情報量はどう決めるとよいか

2A 文字（記号）によって割り当てる 2 進数の数を変える

例えば、文字の情報は 3 種類で、○か×か△で伝えるとする。8 回の伝送内容が、

○○○△×△○×

だったとする。○×△をそれぞれ 2 進数で伝送するとき、一番短く（少ないビット数で）表すには、それぞれの記号にどういう 2 進数を割り当てるとよいか考えてみよう。

○ -> 00　　× -> 11　　△ -> 10

だと、8 × 2 ＝16 ビットとなる。ところが、よく出てくる○だけは 0 を 1 個だけで表すと、12 ビットに減少する。

もちろん、この場合、記号によってビット数が違うと、2 進数を解釈するとき間違えずに元に戻せなくなる可能性があるので、ちゃんと元に戻せる（一意復号可能性を持つ）2 進数を割り当てる必要はある。上の例では、最初に 0 が来れば○、1 がくればその次が 0 か 1 かで、△か×かの判断ができる。

2B 2 進数の長さをどう決めるか

文字（記号）列があるとき、

1) 全文字が同じ割合、例えば、○と×の 2 種類だけのとき、一方を 0、他方を 1 という具合で、それぞれ同じ 1 ビットで表すべきである。○と□と△と×の 4 種類が同じ回数ずつあるとき、それぞれを 2 ビットで表すべきである。同様に文字の種類 n が 2 のべき乗であれば、それぞれ、

$\log_2 n$ ビット

で表すべきである。

2) 2A のように、ある文字の割合が他の文字の 2 倍の割合のとき、例えば、○が 2 回に対して、△と×が 1 回の割合であるとき、○の 1 回は、×と△を一緒にした仮想記号×´と対等と考えられる。すると、○と×´は上の 1)のように、それぞれ同じ 1 ビットで表すべきである。そして、×´の中の 2 つ×と△は、やはり 1 ビットで表すべきであるから、結局×と△は 2 ビットで表すべきである。このとき、×や△を基準に考えた等価的な文字種類数（○は 2 倍で計算する）を n とすると、×と△は $\log_2 n$ ビットで表すべきなので、○は

$\log_2 \frac{n}{2}$ ビット

で表すべきである。

3) 同様にある文字の割合が他より k 倍多いとき、少ない方の文字を基準に考えた等価的な文字種類数を n とすると、各文字は $\log_2 n$ ビットで表すべきなので、k 倍多いほうの文字は、

$\log_2 \frac{n}{k}$ ビット

で表すべきである。

4) 少ない方の文字を基準に考えた等価的な文字種類数が 2 のべき乗でないときは、仮に充分大きい 2 のべき乗数 n を想定し、それぞれの文字の割合を k/n で表せばよい。n->∞とすればいくらでも k/n をそれぞれの文字の出現割合に近づけることができ、そのとき 3)と同じく、

$$\log_2 \frac{n}{k} \quad \text{ビット}$$

で表わせる。

2C 確率を使う

データ中に各文字がどの割合で現れるか事前に知ることはできない。具体的な情報が出てくる前に、どの文字はどういうビット列で表すか決めたいとすると、出現割合の想定値（出現確率）を使う必要がある。出現確率を p とすると、上の k/n が出現割合であるから、k/n=p と変換すれば確率を使って表すことになる。すると、それぞれの文字は以下のビット数で表せば、最少のビット数で表せることになる。これは、文字ごとのデータ量の最小値、つまり情報量を表すことになるので、自己情報量(self-information)と呼ばれる。

自己情報量 = $\log_2 \frac{1}{p}$ ビット

2D 平均情報量

文字ごとに情報量が違うと考えると、2A で考えた文字列の長さは出現文字によって異なることになる。しかし、確率に応じた割合で各文字が出現するので、全体の長さは予測がつく。例えば、充分長い N 個の文字列では、それぞれの文字は Np_i 回現れることが期待されるので、合計の長さは、

$$\sum_{i=1}^{n} Np_i \log_2 \frac{1}{p_i} \quad \text{ビット}$$

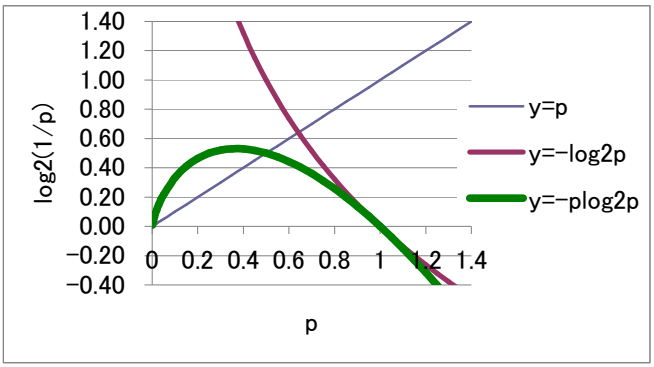
となる。1 文字当たりの（確率の意味での）平均の値は、平均情報量(エントロピー、von Neumann の勧めにより Shannon が entropy と名付けたという)と呼ばれ、

平均情報量=$\sum_{i=1}^{n} p_i \log_2 \frac{1}{p_i}$ ビット

で表わされることになる。

2E 出現確率が 0 の文字の平均情報量への寄与

使わない文字は出現確率が 0 と考えることができる。平均情報量の計算で、出現確率が 0 の文字の寄与はどうなるか考えよう。それは、$p_i \log_2 \frac{1}{p_i}$ で p_i が 0 のときどうなるかである。これは 0 と∞の積なので計算できない。そこで、p_i を段々小さくしていくことを考えると、下図のようになり、0 に近づく。



$p_i \log_2 \frac{1}{p_i} = -p_i \log_2 p_i$ を p_i で微分すると、$-\log_2 p_i - \frac{1}{\log_e 2}$ となり、$p_i \to 0$ で∞となる。つまり、p_i が 0 に近づくと垂直に近い曲線になる。さらに、その値は正のはずだから、$p_i \to 0$ の極限で $p_i \log_2 \frac{1}{p_i}$ は 0 となる。したがって、p_i が 0 に近づくほど、平均情報量に対する寄与は小さくなり、極限では 0 となる。つまり、出現確率が 0 の文字は、平均情報量にとっては存在しないのと同じであり、1E の問題（使わない文字を加えると情報量が増えることになる）は平均情報量として考えれば解決することになる。

練習問題

2-1 毎日の天気を、晴れ/曇り/雨のいずれか一つで表し、それぞれの確率は 1/4、1/2、1/4 であるとすると、28 日分の記録長の期待値は何ビットになるか。

2-2 成績が S か A か B か C のどれか 1 文字で表され、それぞれの割合が 1/8、1/4、1/2、1/8 のとき、成績情報は平均何ビットで表せることになるか。

2-3 情報が○か×で表され、○の出現確率が 2/3、×の出現確率が 1/3 とする。それぞれの、自己情報量を求めよ。さらに、その情報 1 個あたりの平均情報量を求めよ。

2-4 出現確率 1 の文字が一つだけあるときは、平均情報量は 0 ということになる。これを納得できるように説明してみよ。

紛らわしい英数字の書き分け方の例

0 o 0 D 1 l 7 2 Z z

x	log2 x	x	log2 x	x	log2 x	x	log2 x
1	0.00000	26	4.70044	51	5.67243	76	6.24793
2	1.00000	27	4.75489	52	5.70044	77	6.26679
3	1.58496	28	4.80735	53	5.72792	78	6.28540
4	2.00000	29	4.85798	54	5.75489	79	6.30378
5	2.32193	30	4.90689	55	5.78136	80	6.32193
6	2.58496	31	4.95420	56	5.80735	81	6.33985
7	2.80735	32	5.00000	57	5.83289	82	6.35755
8	3.00000	33	5.04439	58	5.85798	83	6.37504
9	3.16993	34	5.08746	59	5.88264	84	6.39232
10	3.32193	35	5.12928	60	5.90689	85	6.40939
11	3.45943	36	5.16993	61	5.93074	86	6.42626
12	3.58496	37	5.20945	62	5.95420	87	6.44294
13	3.70044	38	5.24793	63	5.97728	88	6.45943
14	3.80735	39	5.28540	64	6.00000	89	6.47573
15	3.90689	40	5.32193	65	6.02237	90	6.49185
16	4.00000	41	5.35755	66	6.04439	91	6.50779
17	4.08746	42	5.39232	67	6.06609	92	6.52356
18	4.16993	43	5.42626	68	6.08746	93	6.53916
19	4.24793	44	5.45943	69	6.10852	94	6.55459
20	4.32193	45	5.49185	70	6.12928	95	6.56986
21	4.39232	46	5.52356	71	6.14975	96	6.58496
22	4.45943	47	5.55459	72	6.16993	97	6.59991
23	4.52356	48	5.58496	73	6.18982	98	6.61471
24	4.58496	49	5.61471	74	6.20945	99	6.62936
25	4.64386	50	5.64386	75	6.22882	100	6.64386

3　Shannon の情報量の性質

3A 情報量は何を表すか

自己情報量や平均情報量として定義された情報量は、その情報を表すのに最低限必要なデータ量（ビット数）を表す。情報の価値を表すものではない。

たくさんの情報があるとき、それぞれの情報を表すには、たくさんのビット数が必要になる。情報の種類が無限にあると、ビット数も無限に必要となる。しかし、何かを話し、何かを伝えているとき、たいてい情報に範囲がある（種類の数は有限）。例えば、天気は「晴れ」「曇り」「雨」「雪」など聞いて分かる表現は限りがある（「晴れのち雨」などの組み合わせも天気の表現であるが、有限要素の有限個の組み合わせの数もまた有限である）。情報の表し方が有限種類に限定されれば、有限個のビット数で表現され、それぞれの情報の出現確率が分かれば、全体を短くするために、それぞれの情報を表すのに必要なビット数が決まる。

3B 確率事象系

有限個の中のいずれかの現象が起きて、それを情報として伝えるようということになるが、元の現象が確率的に起きる場合、確率事象系と呼び、以下のように記号的に表す。

$$A=\begin{Bmatrix} a_1 & a_2 \\ 1/3 & 2/3 \end{Bmatrix}$$

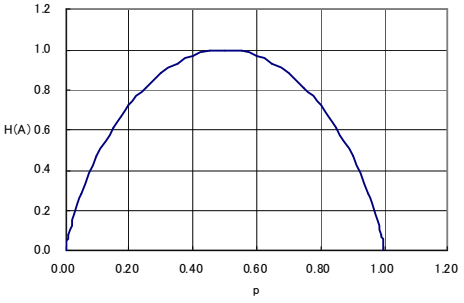
上が事象を表し、下がそれぞれの確率である。確率の合計は１のはずである。

3C 事象が２種類のときの平均情報量の最小値と最大値

平均情報量がどの範囲の値になるか考えてみよう。3B の例のように確率事象系Ａから出てくる情報が２種類のいずれか（起きる事象が２種類）であるときは、一方の確率を p とすると、もう一方が $1-p$ である。平均情報量 H(A)は、

$$H(A)=p\log_2\frac{1}{p}+(1-p)\log_2\frac{1}{1-p}\quad\text{（ビット）}$$

となり、p を変化させたときのグラフは、右図のようになる。最小値は、p が０もしくは１のときの０で、最大値は、p が 1/2 のときの１である。p が０もしくは１というのは、どちらの事象が起きるかが決まっているということで、情報が実質１種類の状況に相当し、情報量はない。一方、p が 1/2 のときは、どちらも対等に出現するということで、ハートレーの情報量で想定する、１ビットで表せる情報量ということになる。



$H(A)=p\log_2\frac{1}{p}+(1-p)\log_2\frac{1}{1-p}$ の図

3D 事象が n 種類のときの平均情報量の最小値と最大値

最小値は、3C の類推で考えると、事象が n 種類あっても、どれかひとつの確率が 1 で他はすべて 0 のとき、平均情報量は 0 となり、これが最小値である。最大値も、3C の類推で言えば、ハートレーの情報量である $\log_2 n$ が予想される。これを確認しよう。平均情報量を H とすると、

$$H-\log_2 n=\sum_{i=1}^{n} p_i \log_2 \frac{1}{p_i}-\log_2 n=\sum_{i=1}^{n} p_i \log_2 \frac{1}{p_i}-\sum_{i=1}^{n} p_i \log_2 n$$

$$=\sum_{i=1}^{n} p_i \log_2 \frac{1}{p_i n}=\sum_{i=1}^{n} \frac{p_i}{\log_e 2}\log_e \frac{1}{p_i n}$$

ここで、右図のように $\log_e x \le x-1$ であるから、

$$\le \sum_{i=1}^{n} \frac{p_i}{\log_e 2}\left(\frac{1}{p_i n}-1\right)=\frac{1}{\log_e 2}\sum_{i=1}^{n}\left(\frac{1}{n}-p_i\right)$$

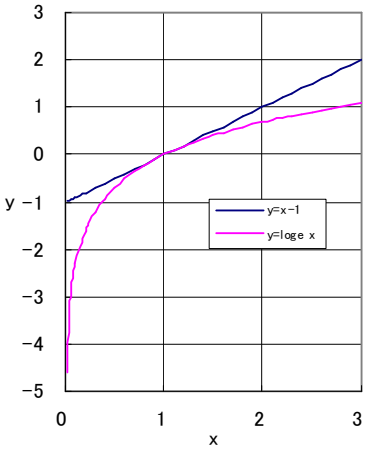
$$=\frac{1}{\log_e 2}(1-1)=0$$

したがって、H の最大値は $\log_2 n$ であることが示された。

最大になるのは、すべての p_i について

$\log_e 1/p_i n = 1/p_i n - 1$ のときのみなので、すべての

$p_i = 1/n$ のときに平均情報量は最大値 $\log_2 n$ となる。



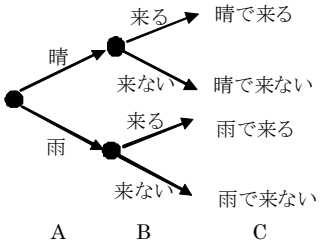
$\log_e x \le x-1$ の図

3E 結合確率と結合平均情報量

天気が確率事象系 A で表され、鈴木さんが来るか来ないかが確率事象系 B で表されるとする。そうすると、天気と鈴木さんが来る/来ないの組み合わせを結合事象 C として考えることができる。このとき、結合確率事象系を考えることになる。事象 a_1 が起こる確率を $p(a_1)$（probability of a_1 と読む）、結合事象 $a_1 b_1$ が起こる結合確率を $p(a_1, b_1)$ と表記することにすると、確率事象系 A、B、C は以下のように表せる。

$$A=\begin{Bmatrix} a_1 & a_2 \\ p(a_1) & p(a_2) \end{Bmatrix},\quad B=\begin{Bmatrix} b_1 & b_2 \\ p(b_1) & p(b_2) \end{Bmatrix},$$

$$C=\begin{Bmatrix} a_1 b_1 & a_2 b_1 & a_1 b_2 & a_2 b_2 \\ p(a_1,b_1) & p(a_2,b_1) & p(a_1,b_2) & p(a_2,b_2) \end{Bmatrix}$$



その結合確率事象の平均情報量は結合平均情報量（joint entropy）と呼ばれ、H(A,B)で表わすと、

$$H(A,B)=H(C)=\sum_k \sum_l p(a_k,b_l)\log_2 \frac{1}{p(a_k,b_l)}$$

となる。ただし、積和の範囲は省略しているが、それぞれの A、B の事象すべてである。

3F 独立確率事象のときの結合平均情報量

3E では、一般性のある表記をしたが、確立事象系Ａと Ｂが独立なとき、つまり天気と鈴木さんが来る/来ないは全く無関係のとき、 $p(a_k,b_l)=p(a_k)\ p(b_l)$ と、単純な積になる。このとき、結合平均情報量は、

$$H(A,B)=\sum_k\sum_l p(a_k)\,p(b_l)\log_2\frac{1}{p(a_k)\,p(b_l)}$$

$$=\sum_k p(a_k)\sum_l p(b_l)\left(\log_2\frac{1}{p(a_k)}+\log_2\frac{1}{p(b_l)}\right)$$

$$=\sum_k p(a_k)\log_2\frac{1}{p(a_k)}\sum_l p(b_l)+\sum_k p(a_k)\sum_l p(b_l)\log_2\frac{1}{p(b_l)}$$

$$=\sum_k p(a_k)\log_2\frac{1}{p(a_k)}+\sum_l p(b_l)\log_2\frac{1}{p(b_l)}$$

$$=H(A)+H(B)$$

つまり、それぞれの平均情報量の和となる。これは、独立な場合、組み合わせの平均情報量は、それぞれの平均情報量の和になることを示す。

練習問題

3-1　２種類のいずれかの事象が起きる場合の平均情報量 H(A)を、一方の確率 p が、1/2、1/3、1/4、1/5 のときについて求め、p が 1/2 から離れると H(A)が小さくなることを確認せよ。

3-2 トランプ（ジョーカを除く）52 枚の中から１枚渡されたとき、その１枚の、スイート（♠ ♡ ♣ ♢ のこと）についての平均情報量、数字（A は１、J は 11、Q は 12、K は 13 とする）についての平均情報量、スイートと数字の組み合わせについて平均情報量を、それぞれ計算せよ。ただし、$\log_2 13=3.700$ とする。

3-3 $H(A)=p\log_2\frac{1}{p}+(1-p)\log_2\frac{1}{1-p}$ が最大となるときの p を、H(A)を p で微分することで求めよ。

3-4 ３つの事象のいずれかが起きるとき、平均情報量を最大にする p,q の値を 3-3 と同様に求めよ。具体的には、まず p を固定して考え H(A)を q で微分し、最大となる q を求め、次に、その q を H(A)に代入し、今度は p を変化させて H(A)最大となる p を求めればよい。

$$A=\begin{Bmatrix}a_1 & a_2 & a_3\\ p & q & 1-p-q\end{Bmatrix},\quad H(A)=p\log_2\frac{1}{p}+q\log_2\frac{1}{q}+(1-p-q)\log_2\frac{1}{1-p-q}$$

4　条件付き平均情報量と相互情報量

4A 条件付確率(conditional probability)

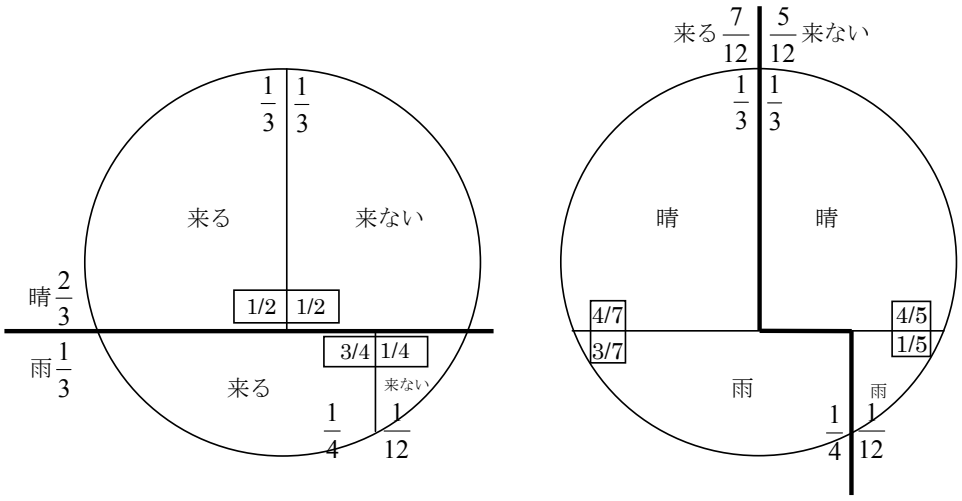
確率事象系ＡとＢの確率変数をそれぞれ a と b とする。結合確率 $p(a,b)$ が $p(a)p(b)$ に等しくないとき、つまりＡとＢが独立事象でないとき、条件付き確率 $p(a|b)$ もしくは $p(b|a)$ を使って、

$$p(a,b)=p(a|b)p(b)=p(b|a)p(a)$$

と表される。たとえば、

$$A=\begin{Bmatrix} a_1 & a_2 \\ p(a_1) & p(a_2) \end{Bmatrix}=\begin{Bmatrix} \text{晴} & \text{雨} \\ 2/3 & 1/3 \end{Bmatrix}$$

$$B=\begin{Bmatrix} b_1(\text{鈴木さんが来る}) & b_2(\text{鈴木さんが来ない}) \\ p(b_1) & p(b_2) \end{Bmatrix}$$



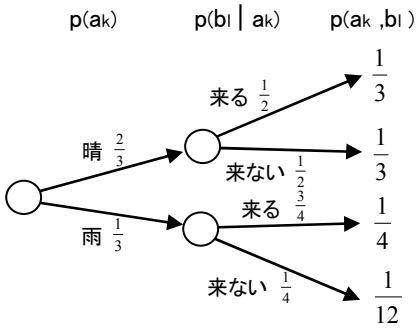
のとき、$p(a_1,b_1)=p(a_1|b_1)p(b_1)$ などと計算される。このときの $p(a_1|b_1)$ は「鈴木さんが来る」とき「晴」である確率である。もちろん、

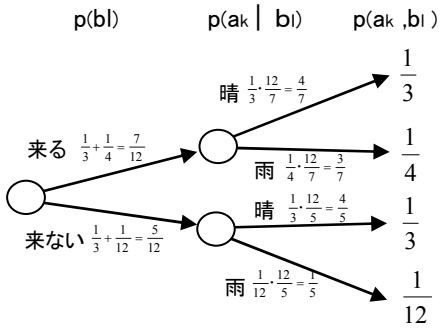
$$p(b_l)=\sum_k p(a_k,b_l),\qquad p(a_k)=\sum_l p(a_k,b_l)$$

である。たとえば、

$$p(b_1|a_1)=1/2,\ p(b_2|a_1)=1/2,\ p(b_1|a_2)=3/4,\ p(b_2|a_2)=1/4$$

のとき、$p(b_1),\ p(b_2)$ などを求めると、以下のようになる。





$\sum_k p(a_k)=1$, $\sum_l p(b_l)=1$, $\sum_k\sum_l p(a_k,b_l)=1$, $p(a_k\,|\,b_l)=p(a_k,b_l)/\,p(b_l)$ である。

4B 条件付き平均情報量(conditional entropy)

確率事象系ＡとＢが独立なとき、$p(a,b)=p(a)p(b)$ と表され、

$$H(A,B)=H(A)+H(B)$$

であった。独立でないとき、$p(a,b)=p(a\,|\,b)p(b)$ なので、$p(a\,|\,b)$ の平均情報量を $H(A\,|\,B)$ と表すと

$$H(A,B)=H(A\,|\,B)+H(B)$$

となり、$p(a,b)=p(b\,|\,a)p(a)$ なので、$p(b\,|\,a)$ の平均情報量を $H(B\,|\,A)$ と表すと

$$H(A,B)=H(B\,|\,A)+H(A)$$

となることが予想され、その結果

$$H(A,B)=H(A\,|\,B)+H(B)=H(B\,|\,A)+H(A)$$

となるはずである。$H(A\,|\,B)$ は、$p(a\,|\,b)$ の自己情報量をすべての $p(a,b)$ について考えた値（期待値）なので、具体的に以下のように書ける。

$$H(A\,|\,B)=\sum_k\sum_l p(a_k,b_l)\log_2\frac{1}{p(a_k\,|\,b_l)}$$

そして、

$$\begin{aligned}
H(A,B)&=\sum_k\sum_l p(a_k,b_l)\log_2\frac{1}{p(a_k,b_l)}\\
&=\sum_k\sum_l p(a_k,b_l)\left\{\log_2\frac{1}{p(a_k\,|\,b_l)}+\log_2\frac{1}{p(b_l)}\right\}\\
&=\sum_k\sum_l p(a_k,b_l)\log_2\frac{1}{p(a_k\,|\,b_l)}+\sum_k\sum_l p(a_k,b_l)\log_2\frac{1}{p(b_l)}\\
&=\sum_k\sum_l p(a_k,b_l)\log_2\frac{1}{p(a_k\,|\,b_l)}+\sum_l p(b_l)\log_2\frac{1}{p(b_l)}\\
&=H(A\,|\,B)+H(B)
\end{aligned}$$

となる。ＡとＢを入れ替えると、$H(A,B)=H(B,A)$ で、$H(A,B)=H(B\,|\,A)+H(A)$ となる。

4C 条件付きと条件付きでない平均情報量の関係

$$H(A|B)-H(A)=\sum_k\sum_l p(a_k,b_l)\log_2\frac{1}{p(a_k|b_l)}-\sum_k p(a_k)\log_2\frac{1}{p(a_k)}$$

$$=\sum_k\sum_l p(a_k,b_l)\log_2\frac{p(a_k)}{p(a_k|b_l)}=\sum_k\sum_l p(a_k,b_l)\log_2\frac{p(a_k)p(b_l)}{p(a_k,b_l)}$$

ここで、$\log_e x\le x-1$を使うと

$$\le\sum_k\sum_l\frac{p(a_k,b_l)}{\log_e 2}\left\{\frac{p(a_k)p(b_l)}{p(a_k,b_l)}-1\right\}=\sum_k\sum_l\frac{1}{\log_e 2}\{p(a_k)p(b_l)-p(a_k,b_l)\}=0$$

したがって、

$$H(A|B)\le H(A)$$

である。同様に

$$H(B|A)\le H(B)$$

となる。つまり、条件付平均情報量は、条件無しの平均情報量以下になる。

4D 結合平均情報量と個別平均情報量の合計の関係

上の結果から、

$$H(A,B)=H(A|B)+H(B)\le H(A)+H(B)$$

となる。つまり、結合平均情報量は、それぞれの平均情報量の和以下の値になり、ＡとＢが独立でないときは、独立なときより少ないビット数でその情報を表せることを意味する。

4E 相互情報量(mutual information)

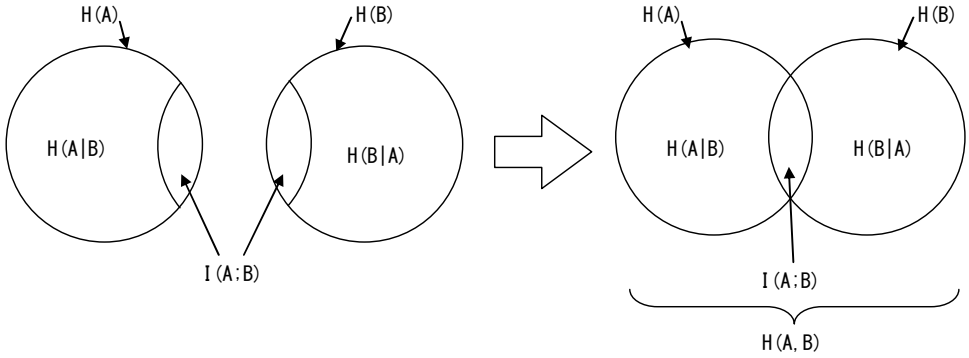
上の結果は、確率事象ＡとＢが独立でないとき、Ｂが分かるとＡを表すビット数が少なくて済む、つまりＢを表すビットがＡに関する情報も表していることを示す。これは、ＢによってＡに関する情報が得られるとも解釈できる。このとき得られる情報量を相互情報量と呼びI(A;B)で表す。I(A;B)は、個別平均情報量の合計から結合平均情報量を引いたものであるから、

$$I(A;B)=H(A)+H(B)-H(A,B)$$

$$=H(A)-H(A|B)$$

$$=H(B)-H(B|A)$$

となる。$I(A;B)$は$I(B;A)$と等しく、結合平均情報量が独立なときに比べて少なくなる量を表し、Ｂが表しているＡに関する情報量は、Ａが表しているＢに関する情報量と等しいことになる。



練習問題

4-1 本文中にある数値を使って、天気（晴/雨）と鈴木さんが来るか来ないかを確率事象ＡとＢと考え、H(B), H(B|A), I(A;B)を求めよ。

4-2 あるサッカーチームは、ホームでの勝率は 3/4、アウェイでの勝率は 1/4、ホームとアウェイ合わせての勝率は 1/2、ホームとアウェイの比率は 1:1 とするとき、ホームかアウェイかの情報(B)と勝ち負けの情報(A)を合わせた情報を表すのに必要な情報量は何ビットか。ただし、引き分けは無いものとする。また、ホームかアウェイかが分かると、勝ち負けについて何ビットの情報を得られるか。

確率と平均情報量のメモ

$p(a)$ a となる確率

$p(b)$ b となる確率

$p(a,b)$ a でかつ b となる確率（a と b が独立なとき、$p(a)\ p(b)$ に等しい）

$p(a|b)$ b であるときに a となる確率

$$p(b_l)=\sum_k p(a_k,b_l)$$

$$p(a_k)=\sum_l p(a_k,b_l)$$

$$H(A,B)=H(A|B)+H(B)$$
$$=H(B|A)+H(A)$$

$H(A,B)=H(A)+H(B)$ （独立なとき）

$H(B|A)\le H(B)$

$$I(A;B)=H(A)+H(B)-H(A,B)$$
$$=H(A)-H(A|B)$$
$$=H(B)-H(B|A)$$

5　情報源のモデル

5A 連続情報源と離散情報源

音声などのように連続信号で表されるアナログ情報を発生する情報源を連続情報源（continuous information source）と呼び、文字などのようにディジタル情報を発生する情報源を離散情報源（discrete information source）と呼ぶ。本講義では、離散情報源のみを扱うが、連続情報源は離散情報源の情報の種類を増やし時間間隔を狭めていったときの極限として考えることができる（詳しくは南敏著の参考書などを見よ）。

5B 離散情報源の出力

離散情報源の出力を、文字を含む広い意味の記号（symbol）と考える。記号の集合（情報源アルファベットとも呼ばれる）Ｓは、記号が英文字である場合、以下のように表される。

$$S=\{a,b,c,...,z\}$$

複数回の出力記号を並べ、記号系列として扱うとき、通報（message）とも呼ばれ、

$$\mathbf{s}^n = s_1 s_2 s_3 \cdots s_n$$

のように表わされる。

5C 無記憶情報源

過去の履歴に依存せずに、記号ごとの生起確率が決まる場合、記号の下にその確率を書くと、次のように確率事象系として定義できる。

$$S=\begin{Bmatrix} ○ & × \\ 1/2 & 1/2 \end{Bmatrix}$$

5D マルコフ情報源

過去の履歴に多少なりとも依存して記号が出力される場合も多い。その依存関係がマルコフ連鎖で表されるものをマルコフ情報源と呼ぶ。マルコフ連鎖は、各記号の生起確率が、過去の履歴を条件にした条件付き確率で表されるものである。たとえば、1 から m までの過去の記号出力に依存して m+1 個目での記号 s_{m+1} の生起確率が決まるとき、

$$p(s_{m+1}\,|\,\mathbf{s}^m) \text{ もしくは } p(s_{m+1}\,|\,s_1 s_2 s_3 \cdots s_m)$$

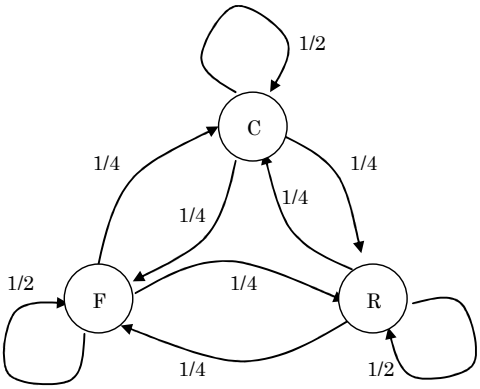
で表される。これは、m 重マルコフ情報源と呼ばれる。過去に依存するということは、過去の情報を（部分的かもしれないが）内部に記憶している（内部状態を持つ）ということを意味する。

5E 状態遷移図（シャノン図）

１重マルコフ連鎖（単純マルコフ連鎖とも呼ぶ）のとき、例えば、日ごとの天気（晴、曇、雨の３種だけとする）の移り変わりが前日の天気を条件とする条件付き確率で決まるとき、内部状態と出力記号を以下のような状態遷移図で表すと分かりやすい。○が状態、矢印が状態遷移、矢印横の値は遷

移確率を表す。各遷移確率は、晴（fair）をF、曇（cloudy）をC、雨（rainy）をRで略記すると、以下の条件付確率で表せる。

$$p(F|F)=1/2,\ p(C|F)=1/4,\ p(R|F)=1/4,$$
$$p(F|C)=1/4,\ p(C|C)=1/2,\ p(R|C)=1/4,$$
$$p(F|R)=1/4,\ p(C|R)=1/4,\ p(R|R)=1/2,$$



5F 遷移確率行列

ある時刻tでのF,C,Rの状態確率を以下の横ベクトル$\mathbf{u}(t)$で表す。

$$\mathbf{u}(t)=(F\text{である確率},C\text{である確率},R\text{である確率})$$

次に、5Eの遷移確率を、まとめて遷移確率行列として表現すると、以下のようになる。ただし、縦横方向ともにF,C,Rの順で、縦方向がt時点での天気で、横方向がt+1時点での天気である。

$$\mathbf{P}=\begin{bmatrix} p(F|F) & p(C|F) & p(R|F) \\ p(F|C) & p(C|C) & p(R|C) \\ p(F|R) & p(C|R) & p(R|R) \end{bmatrix}=\begin{bmatrix} 1/2 & 1/4 & 1/4 \\ 1/4 & 1/2 & 1/4 \\ 1/4 & 1/4 & 1/2 \end{bmatrix}$$

上の$\mathbf{u}(t)$に遷移確率行列$\mathbf{P}$を右から掛けると、時刻t+1の状態確率ベクトル$\mathbf{u}(t+1)$が求まる。

$$\mathbf{u}(t+1)=\mathbf{u}(t)\mathbf{P}$$

このように、情報理論では、シャノンの論文に倣い、状態確率を横ベクトルで表し、状態遷移確率行列を右から掛けるのが慣行である。

5G 状態遷移計算の簡単な例

$\mathbf{u}(t)$が(1,0,0)とすると、5Fの$\mathbf{P}$のとき、$\mathbf{u}(t+1)=(1/2,1/4,1/4)$となり、晴の条件付確率と一致する。同様に、$\mathbf{u}(t)$を(0,1,0)、(0,0,1)として、$\mathbf{u}(t+1)$を計算してみると、行列演算の意味が分かるであろう。一般には、$\mathbf{u}(t)$はさまざまな値をとる可能性があり、$\mathbf{u}(t)\mathbf{P}$で次の時刻での状態確率が計算できることになる。

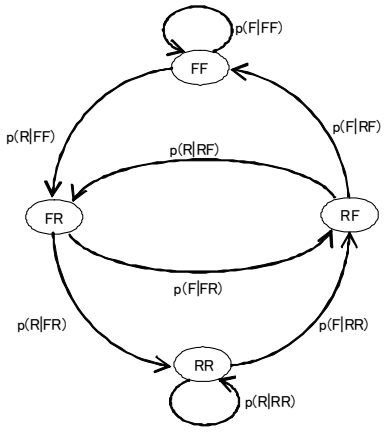
5H 状態遷移を繰り返すとどうなるか

$\mathbf{u}(t)$に右から$\mathbf{P}$を掛けることを繰り返していく（$\mathbf{u}(t)\mathbf{P}^n$のnを大きくしていく）と、ある値（定常確率）に収束する（$\mathbf{uP}=\mathbf{u}$を満たす値$\mathbf{u}$に近づく）場合がある。その場合、正規マルコフ情報源とよ

ぶ。振動するなどして収束しないが、時間平均の極限をとると $\mathbf{uP}=\mathbf{u}$ を満たす値に収束するときエルゴード（時間平均と集合平均が一致する）マルコフ情報源と呼ぶ。

5I ｍ重マルコフ連鎖の状態遷移図

２重以上のマルコフ連鎖を図に表すには、条件となる m 回の記号列を１回前の状態に見立てた単純マルコフ連鎖に相当する状態遷移図を作ればよい。n 種類の記号があると、n^m 個の状態を記載する必要がある。右の図は、F と R の２つの状態について、前２回の状態により、次の状態の確率が決まることを表す図である。



5J ｍ重マルコフ情報源の平均情報量

m 回分全体の平均情報量 $H(S^m)$ と m 回目までが分かっているときの m+1 回目に関する平均情報量 $H(S^{m+1}\mid S^m)$ から、m+1 回分全体の平均情報量 $H(S^{m+1})$ を書き表してみる。

$H(A,B)=H(A\mid B)+H(B)$ で、S^m をＢとし、S^m のその次の記号 S' を A とすると、

$$H(S',S^m)=H(S'\mid S^m)+H(S^m)$$

となるが、S' と S^m を合わせたものが S^{m+1} で、S^m のときに S' になるのと S^{m+1} になるのは同じことだから、

$$H(S^{m+1})=H(S'\mid S^m)+H(S^m)$$

となる。$H(S'\mid S^m)$ で得られるのは、m+1 回目に関する平均情報量だが、実質的に１回分の情報量と同じと考えて単純に $H(S\mid S^m)$ と表すと、記号系列を $\mathbf{s}^m$ として、以下のように計算される。

$$H(S\mid S^m)=\sum_{k=1}^{n}\sum_{l=1}^{n^m}p(s_k,\mathbf{s}_l^m)\log_2\frac{1}{p(s_k\mid\mathbf{s}_l^m)}$$

4C で示したように、

$$H(S\mid S^m)\le H(S)$$

であり、過去の情報から推測できる分だけ、１回分を表すのに必要な平均情報量は少なくなる（少ないビット数で表現できる）。

練習問題

5-1　３つのクラス（A、B、C）がある。１日経つと、クラス A に属する者の 1/2 はそのままだが残りの 1/2 は B に下がる、クラス B に属する者の 1/8 は A に上がり 1/2 は B のままで 3/8 は C に下がる、クラス C に属する者の 1/16 は B に上がるが残りは C のままである。

(1) 状態遷移図を描け。

(2) 遷移確率行列を示せ。

(3) 充分時間が経ったあとの３つの状態それぞれの確率（定常確率）を求めよ。

ヒント：$\mathbf{u}=\mathbf{uP}$ を満たす $\mathbf{u}=(p(\mathrm{A}),p(\mathrm{B}),p(\mathrm{C}))=(\alpha,\beta,1-\alpha-\beta)$ を連立方程式を解いて求めればよい。

5-2　遷移確率行列が以下のとき

$$\mathbf{P}=\begin{bmatrix}1/2 & 1/2 & 0\\ 1/3 & 0 & 2/3\\ 2/3 & 0 & 1/3\end{bmatrix}$$

(1) 状態遷移図を描け。３つの状態は、A, B, C としてよい。

(2) ３つの状態それぞれの定常確率を求めよ。

(3) $H(S\,|\,S^1)$ を計算し、定常確率を使って $H(S)$ を求めよ。

5-3　遷移確率行列が以下のとき

$$\mathbf{P}=\begin{bmatrix}0 & 1\\ 1 & 0\end{bmatrix}$$

(1) 状態遷移図を描け。２つの状態は、A, B としてよい。

(2) ２つの状態それぞれの初期確率を p と q として、それぞれの状態にある確率の時間平均を求めよ。もちろん、q=1-p である。

(3) 任意の時点で、２つの状態の確率は、それぞれいくつと考えればよいか。（これは、集合平均に相当する。）

6　情報源符号化

6A 符号化

情報源から発生する各記号に、符号（code）を割り当てて送信するとき、できるだけ短い符号にしたい。情報源を表す確率事象系が以下のとき、

$$S=\begin{Bmatrix} a & b & c \\ p(a) & p(b) & p(c) \end{Bmatrix}$$

それぞれの記号に対し、２元符号（０か１で符号を構成する）の符号組を

$$C=\{00 \quad 01 \quad 10\}$$

などと決める。

6B 一意復号可能性

よく考えずに符号を決めると困ることがある。ひとつは、一意復号可能（uniquely decodable）かどうかである。たとえば、上の場合で、一部の符号を短くして、

$$C_1=\{0 \quad 01 \quad 10\}$$

としたとすると、010 を受け取ったとき、ac か ba か、どちらか分からない。このような場合を一意復号不可能という。もちろん、２つ以上の記号に同じ符号を割り当てた場合、例えば、

$$C_2=\{0 \quad 1 \quad 1\}$$

の場合も一意復号不可能であるが、このように別々の記号に同じ符号を割り当てたものを、特異符号（singular code）と呼ぶ。

6C 瞬時復号可能性

例えば、以下のように割り当てた場合、

$$C_3=\{0 \quad 01 \quad 011\}$$

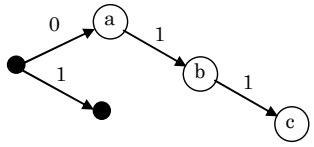
必ず０が先頭にあるので０が現れたときに符号が判定でき、一意復号可能である。しかし、送られてきた符号が 01 のとき、その次のビットが１か０かまで見ないと、その符号が b か c かの判断ができない。このように、次の符号部分かもしれない部分まで調べて初めて復号が可能な符号を、非瞬時符号とよぶ。

$$C_4=\{0 \quad 10 \quad 11\}$$

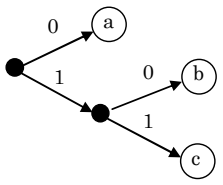
ならば、一意復号可能で、その記号を表す符号全体を受け取った段階で送信記号が判断できるので**瞬時符号**（instantaneously decodable code）である。

6D 符号木

瞬時復号可能かどうかは、以下の図のような符号木をつくると分かる。図を書くときは、０か１かで、上は０下は１という具合に二股に分け、符号を割り当てたノードには対応する記号を書き込む。すべての割り当てた符号が符号木の末端（葉のところ）に割り振られていれば、瞬時符号である。ある符号が別の符号の語頭（prefix）になっていると非瞬時符号の可能性がある（一意復号可能でないときは、非瞬時符号とも呼べない)。



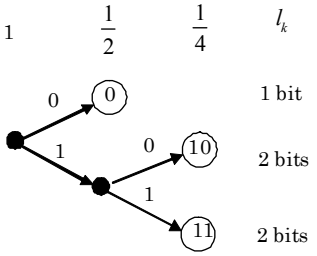
非瞬時符号



瞬時符号

6E クラフトの不等式

瞬時符号を符号木で表すと、すべての符号は末端の葉に割り当てられることになる。このとき、ノードごとの値を想定し、始点を１に、１回枝分かれすると 1/2、２回枝分かれすると 1/4 と 1/2 ずつした値とする。すべての葉に符号が割り当てられているとき、２つ合わせて上位の値に合算していくと、1/2 していったものを集めているだけなので、最後に１になるはずである。枝分かれの回数は、符号の長さ（ビット数）に等しいので、個々の符号の長さをl_kとすると、



$$\sum_{k=1}^{n} 2^{-l_k} = 1$$

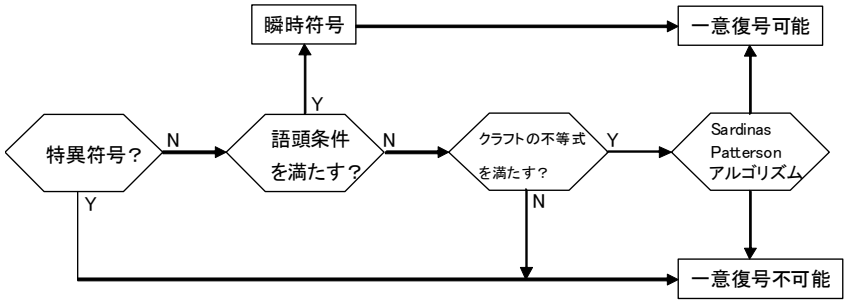
となる。もし、葉の数が余って、符号に割り当てられていないものがあると、１より小さくなるので、その場合も合わせて表現すると、

$$\sum_{k=1}^{n} 2^{-l_k} \le 1$$

となる。これをクラフト（Kraft）の不等式と呼び、瞬時符号のときには成立する。逆にクラフトの不等式が成立していれば、瞬時符号である可能性がある。

6F 一意復号可能符号かどうかの判定法

一意復号可能かどうか調べるには、まず、**語頭条件**（どの符号も別の符号の語頭（prefix）になっていないという条件）を満たしているか調べる。語頭条件を満たしていれば、頭から調べていけば、どれかひとつとしか一致しないはずなので、特異符号（同じ符号を別の情報に使っている）でない限り一意復号可能符号である。語頭条件を満たしていない場合、クラフトの不等式を満たしていなければ、一意復号可能でない（McMillan が示した（参考：大石 2.6 節））。一意復号可能か可能でないかは、次の図の手順で判定できる。



語頭条件を満たさず、クラフトの不等式を満たす場合、以下の Sardinas-Patterson アルゴリズムで調べる。以下の符号 C についての検討を例に説明する。

$$C=\{00 \quad 001 \quad 0010\}$$

まず、語頭が別の符号と同じものを探す。この場合、00 と 001、00 と 0010、001 と 0010 がある。00 と 001 について、001 に何らかの符号が連なったとき、00 の後に別の何らかの符号がつながったときと同じになるものがあるか調べればよい。両方、先頭が 00 なので、その残りの 1 だけを考えればよい。1 の後に何かが連なったものが、別の符号になるかどうか調べればよい。このとき、1 で始まる符号はないので、そういうことはない。つまり、00 と 001 が先頭になるものの間では、全体が同じ符号になるものはない。同様に、00 と 0010 を調べると、共通の 00 を外した残りの 10 は他の符号の頭になっていないので、その後に何を連ねても同じ符号になることはない。

001 と 0010 については、共通部分を外すと、一方に 0 が残る。この 0 の後に何かを連ねれば別の符号と一致するか調べる。つまり 0 と他の符号を試す。３符号とも先頭が 0 なので、その先頭の 0 を外した残り、つまり 0 か 01 か 010 が先頭である符号があれば、こちら側の続きになる。01 と 010 については、それで始まる符号はない。残りの 0 は、ひとつ前の状態と同じだから、同じ調査を繰り返しても同じ結果になる。つまり、別の符号の組み合わせと一致することはない。一般には残り部分は何種類かある可能性があるが、いくら試してもそれらのいずれかの符号しか出てこなくなったら、それ以上繰り返しても意味はない。このようになれば、ちょうど別の符号に一致することはないということになるので、一意復号可能であると言える。

逆に、どこかの段階で、残りの部分がいずれかの符号に一致すれば、同じ符号が２つ以上の符号の組み合わせで表されることになって、一意復号不可能であることがわかる。

6G 平均符号長と個別符号長の決め方

すべての符号について、平均符号長（確率的な意味での平均で期待値のこと）L を求めると、

$$L=\sum_{k=1}^{n} p(s_k) l_k$$

と表せる。ただし、s_k は k 番目の記号を表し、l_k は k 番目の符号長を表すものとする。各 s_k に対して、

$$\log_2 \frac{1}{p(s_k)} \le l_k < \log_2 \frac{1}{p(s_k)} + 1$$

となるよう l_k（整数）を定めたとき、各辺を $\sum_{k=1}^{n} p(s_k)$ で加算すると、

$$H(S) \le L < H(S)+1$$

となる。ただし、$H(S)$ は s_k に関する平均情報量である。これは、上の l_k の決め方で、平均符号長が平均情報量＋１ビット未満に納めることができることを意味する。さらに、２つ上の式の左側から、

$$\frac{1}{p(s_k)} \le 2^{l_k}$$

であるから、$p(s_k) \ge 2^{-l_k}$。したがって、

$$\sum_{k=1}^{n} 2^{-l_k} \le \sum_{k=1}^{n} p(s_k) = 1$$

となって、クラフトの不等式も満たす。その結果、上の l_k の条件を満たすとき、瞬時符号が作れることになる。

6H 拡大情報源

いくつかの情報をまとめて、ひとつの符号を割り当てることを考えてみる。例えば、以下の無記憶確率事象系 S があるとすると、

$$S = \begin{Bmatrix} a & b \\ 1/4 & 3/4 \end{Bmatrix}, \qquad C = \{0 \quad 1\}, \qquad L = 1$$

$$H(S) = \frac{1}{4}\log_2 4 + \frac{3}{4}(\log_2 4 - \log_2 3) = \log_2 4 - \frac{3}{4}\log_2 3 = 2 - \frac{3}{4}\times 1.585 = 0.811 \text{ ビット}$$

この事象２回分を合わせた確率事象系 S^2 を考え、以下の符号化をすると、

$$S^2 = \begin{Bmatrix} aa & ab & ba & bb \\ \frac{1}{16} & \frac{3}{16} & \frac{3}{16} & \frac{9}{16} \end{Bmatrix}, \qquad C_2 = \{000 \quad 001 \quad 01 \quad 1\},$$

$$L_2 = 3\times\frac{1}{16} + 3\times\frac{3}{16} + 2\times\frac{3}{16} + 1\times\frac{9}{16} = \frac{27}{16} = 1.6875 \text{ ビット}$$

２個分が平均 1.688 ビットなので、１個分は平均 0.844 ビットで符号化していることになる。このように、複数まとめて符号化すると、1 個あたりの平均符号長を短くできる。このときの S^2 で表される情報源は、S で表される情報源に対する２次の拡大情報源と呼ぶ。

6I 情報源符号化定理

S^2 と同様に N 個の情報をまとめて確率事象系 S^N を考えて、ひとつの符号を割り当てると、6G と同様にして、

$$H(S^N) \le L_N < H(S^N)+1$$

と書ける。ただし、L_N は N 個をまとめて符号化したときの平均符号長である。まとめる事象がすべて独立なとき、L を一個あたりの平均符号長とすると、

$$NH(S) \le NL < NH(S)+1$$
$$H(S) \le L < H(S)+1/N$$

となる。Nを∞に近づけていくと、

$$L \to H(S)$$

つまり、**平均符号長を平均情報量にいくらでも近づけて、瞬時符号を作ることができる**ことになる。2元（2進数で表す場合）以外の任意のr元符号に対して厳密に表現すると、以下のようになる。

情報源符号化定理(Information Coding Theorem)

無記憶情報源Sの拡大情報源を考えると、次の条件を満たす1情報源記号あたりの平均符号長Lを持つr元瞬時符号が構成できる。

$$\forall \varepsilon > 0, \quad \frac{H(S)}{\log_2 r} \le L < \frac{H(S)}{\log_2 r} + \varepsilon$$

6J 符号の効率と冗長度

実際の符号の平均符号長Lは平均情報量H(S)以上の長さになる。符号の効率(efficiency)は

$$e = \frac{H(S)}{L}$$

で表され、無駄になる割合を冗長度(redundancy)と呼び、以下で表される。

$$1 - e = \frac{L - H(S)}{L}$$

6K 平均符号長の下限

平均符号長Lは平均情報量$H(S)$より小さくすることはできない。それは、$H(S)-L$を以下のように評価することで確認できる。$p(s_k)$をp_kと略記し、$l_k = \log_2 q_k$となるq_kを導入すると、

$$H(S) - L = \sum_{k=1}^{n} p_k \log_2 \frac{1}{p_k} - \sum_{k=1}^{n} p_k l_k = \sum_{k=1}^{n} p_k (\log_2 \frac{1}{p_k} - \log_2 \frac{1}{q_k})$$
$$= \sum_{k=1}^{n} p_k \log_2 \frac{q_k}{p_k} \le \sum_{k=1}^{n} \frac{p_k}{\log_e 2} \left(\frac{q_k}{p_k} - 1 \right) = \frac{1}{\log_e 2} \sum_{k=1}^{n} (q_k - p_k) = \frac{1}{\log_e 2} (1-1) = 0$$

ここで、$\log_e x \le x - 1$を使っている。したがって、必ず

$$L \ge H(S)$$

で、平均符号長の下限は平均情報量となる。

6L より分かりやすい情報源符号化定理

次章のシャノン・ファノ符号化は、以下の条件を満たすように個別符号長を具体的に決める方法となっている。ただし、s_kはk番目の記号を表し、l_kはk番目の符号長を表す整数値とする。

$$\log_2 \frac{1}{p(s_k)} \le l_k < \log_2 \frac{1}{p(s_k)} + 1$$

6I では、存在の可能性だけ説明したが、具体的に作り方が示されているので、6I の定理の条件にあう瞬時符合が存在すると言える。以上の結果を、分かりやすくまとめると以下になる。

無記憶情報源 S に対して

(1) 平均符号長 L の下限は平均情報量 $H(S)$ である。

$$L \geq H(S)$$

(2)拡大情報源の符号化を行うことで、平均符号長 L を任意に $H(S)$ に近付けた瞬時符号（したがって一意復号可能な符号）を作ることができる。

$$\forall \varepsilon > 0, \quad H(S) + \varepsilon > L \geq H(S)$$

練習問題

6-1　情報源の出力記号 $S = \{a \quad b \quad c \quad d \quad e\}$ に対して、下表の 6 種類（C_1 から C_6）の符号化を検討する。以下の問いに答えよ。

	C_1	C_2	C_3	C_4	C_5	C_6
a	000	0	0	1	00	0
b	100	10	01	01	10	10
c	101	110	011	000	1011	1011
d	110	1110	0111	0010	110	110
e	111	11110	01111	0011	1111	1111

（１）それぞれの符号木を書け。

（２）それぞれクラフトの不等式が成り立つか調べよ。

（３）一意復号可能な符号はどれか。

（４）瞬時符号はどれか。

6-2　情報源 $S = \begin{Bmatrix} a & b \\ 0.9 & 0.1 \end{Bmatrix}$ に対して１個だけ符号化するとき、H(S)、（２進で）最短の平均符号長 L_1、冗長度を求めよ。次に、２個の組み合わせ S^2 に対して符号化するとき、$H(S^2)$、２進で最短の符号を考えてその平均符号長 L_2、冗長度を求めよ。

7　情報源符号化法

7A シャノン・ファノ符号（Shannon-Fano Code）

情報源符号化定理に沿って、以下の式を満たす符号長l_kを求める。

$$\log_2 \frac{1}{p(s_k)} \le l_k < \log_2 \frac{1}{p(s_k)} + 1$$

この式を満たすような一意復号可能符号が作れれば、6G に述べたように、それは瞬時符号となる。例えば、送信すべき情報が以下の確率事象系 S で決まり、確率の順に並んでいるとすると、

$$S = \begin{Bmatrix} s_1 & s_2 & s_3 & s_4 \\ 0.4 & 0.3 & 0.2 & 0.1 \end{Bmatrix}$$

$$l_1 \ge \log_2 \frac{1}{0.4} = \log_2 10 - \log_2 4 = 3.21 - 2 = 1.21$$

$$l_2 \ge \log_2 \frac{1}{0.3} = \log_2 10 - \log_2 3 = 3.21 - 1.58 = 1.63$$

$$l_3 \ge \log_2 \frac{1}{0.2} = \log_2 5 = 2.32$$

$$l_4 \ge \log_2 \frac{1}{0.1} = \log_2 10 = 3.21$$

なので、l_1、l_2、l_3、l_4はそれぞれ、2、2、3、4 ビットとなる。符号の決め方として、(1)それぞれに異なる数値を割り当てる、(2)その数値を２進表現する、(3)その２進表現から上に決めた符号長だけ切り出す、という方法を採用する。ただし、(1)の数値は、(2)と(3)の手順で語頭条件（ある符号が別の符号の語頭になっていない）を満たし、瞬時符号になるようにする。具体的には、以下のよう累積確率P_iを決める。

$$P_1 = 0, P_2 = P_1 + p(s_1) = 0.4, P_3 = P_2 + p(s_2) = 0.7, P_4 = P_3 + p(s_3) = 0.9$$

これで、順に大きくなる異なる値が得られ、それぞれ（小数点以下部分）を２進数にすると、

$$P_1 = 0.00000\cdots, P_2 = 0.01100\cdots, P_3 = 0.10110\cdots, P_4 = 0.11100\cdots$$

その小数点下から、2、2、3、4 ビットを切り出すと、以下の符号が得られ、符号木を書くと瞬時符号であることが分かる。

$$C = \{c_1 \quad c_2 \quad c_3 \quad c_4\} = \{00 \quad 01 \quad 101 \quad 1110\},$$

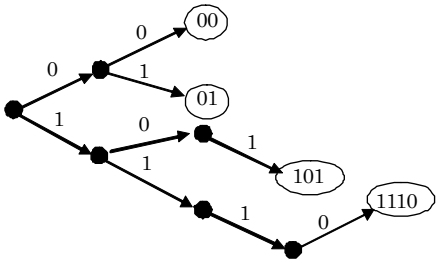
この方法で語頭条件を満たすことは、以下のように確認できる。6G にある通り、

$$p(s_k) \ge 2^{-l_k}$$

なので、例えば、

$$P_2 - P_1 = p(s_1) = 0.4 \ge 2^{-2} = 2^{-l_1}$$

となっていて、P_1とP_2の差は割り当てたビット数（l_1）



で表せる最小値（$2^{-2}=0.25$）以上である。したがって、P_1とP_2はl_1のビット数の範囲で１以上違う値のはずである。P_2とP_3の差も同様に2^{-l_2}以上である。P_3とP_4も同じで、結局、短い符号は、長い符号の語頭になっていることはない。ただし、符号木を見れば分かるように、この符号は冗長である。101 は 10 でもよく、1110 は 11 にすれば短くなる。

7B ハフマン符号（Huffman code）

シャノン・ファノ符号は平均で 1 ビット近く平均情報量より長い可能性がある。より短い符号の作り方をハフマンが見つけた。さらに、ハフマンの符号は、それぞれの確率値が分かっているとき、最短符号（compact code）となることが分かっている。その作り方を例で示す。情報源Sで、確率の小さい下位２つを合わせてs_3'として記号を１個減らしたS'を作る。その中で、確率の小さい下位２つを合わせてs_2'としてS''を作り、最終的に記号の数を２個まで集約していく。

$$S=\begin{Bmatrix} s_1 & s_2 & s_3 & s_4 \\ 0.4 & 0.3 & 0.2 & 0.1 \end{Bmatrix} \quad \Rightarrow \quad S'=\begin{Bmatrix} s_1 & s_2 & s_3' \\ 0.4 & 0.3 & 0.3 \end{Bmatrix} \quad \Rightarrow \quad S''=\begin{Bmatrix} s_2' & s_1 \\ 0.6 & 0.4 \end{Bmatrix}$$

$$C=\{1 \quad 00 \quad 010 \quad 011\} \quad \Leftarrow \quad C'=\{1 \quad 00 \quad 01\} \quad \Leftarrow \quad C''=\{0 \quad 1\}$$

２つのときは、一方に０を他方に１を割り当てるのが最短符号なので、S''の符号をC''とする。s_2'を元の２つに戻すと、s_2'の符号０にもう１個０か１を付けて元の情報s_2とs_3'を区別すると、C'になる。s_3'を元の２つに戻し、s_3'の符号 01 にもう１個０か１を付けて元の情報s_3とs_4を区別すると、Cになる。これで元の４個に対する符号ができた。

S'やS''でも確率の順に並べているが、確率が同じときは、どちらを先にしてもよい。記号ごとの符号長は違ってくる場合もあるが、確率が同じなので、平均符号長に違いはない。また、２つに分けるときに、０と１をどちらに割り当ててもよい。

ハフマン符号の作成手順

(1) 各記号を確率の大きい方から並べる
(2) 確率値最下位 2 個を合わせたものに、新たな記号を与え、確率値を計算する
(3) 記号の数が 2 個になるまで(1)と(2)を繰り返す
(4) 2 個のそれぞれに 0 と 1 の符号を割り当てる
(5) (2)の合わせた順を逆にたどり、2 つに分けるたび、それぞれに 0 と 1 を符号に追加する
(6) 元の記号になるまで(5)を繰り返す

7C ハフマン符号は最短符号

このことの証明の大筋は以下のようになる。

(1) 記号が２つだけのとき、一方を 0、他方を１とし、１ビットを割り当てるのが最短符号である。
(2) 記号が３つのとき、平均符号長が最短になるには、３つのうち２つだけ２ビットにした場合である（３つとも２ビット以上にすると長くなる）。どれを２ビットにするとよいかを考えてみると、

$$S=\begin{Bmatrix} s_1 & s_2 & s_3 \\ p_1 & p_2 & p_3 \end{Bmatrix}, \qquad p_1+p_2+p_3=1$$

のとき、s_1に１ビットを割り当て、残りの２個に２ビットを割り当てると（３個のときの）平均符号長L_3は、

$$L_3=1\times p_1+2\times p_2+2\times p_3=1+p_2+p_3 \quad \text{ビット}$$

となる。L_3が最小になるは、p_2+p_3の値が一番小さいときで、それはp_2とp_3が３つの中で小さい方の２つのときである。つまり、確率の小さいもの２つに２ビットを割り当てると最短符号になる。

(3) 同様に、まとめてn個のとき最短平均符号長L_nとなる符号があり、それをn+1個に戻したときの平均符号長L_{n+1}は、どれか一つをs_iとs_jに分割して、それぞれの符号として１ビット付加することになり、分割する記号の確率p_iとp_jで、

$$L_{n+1}=L_n+p_i+p_j$$

と表される。L_{n+1}が最小となるのはp_iとp_jが最小のとき、つまり最下位２個のときである。

(4) 確率最小の記号2個をまとめてひとつにすることを繰り返して、記号は2個だけになった状態は、上の(1)である。そこから(3)の元の最小確率 2 個の分割を繰り返して、元の情報源まで戻すとき、平均符号長は常にそれぞれの時点での最短なので、元の状態に戻ったときも最短符号である。このとき、すべての分割で、必ず他の記号より小さい

この作業を最後まで繰り返してできる符号は最短符号である。つまり、常に、最小の２個を分割して１ビット追加する場合が最短符号になる。これを逆に準備するには、最小の２個を合わせて１個にする作業をしていくことになる。これを行っているのが、上のハフマン符号の作り方である。

7D その他の情報源符号

各記号の確率が予め分かっているときの最短符号はハフマン符号であるが、ハフマン符号は符号表を送信側と受信側で持たないといけない。確率が小さいものが多いと、符号長も長くなり符号表も大きくなる場合がある。そこで、符号表を持たなくてよい符号法としてイライアス(Elias)符号、算術符号がある。また、各記号の確率が予め分かっていないときのための（漸近的に最適な）符号は、ユニバーサル符号と呼ばれ、Lempel-Ziv 符号、ブロックソート法や文法圧縮法による符号などがある。

練習問題

7-1　次の確率事象系について、以下の問いに答えよ。

$$S_1=\begin{Bmatrix} a & b & c \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{Bmatrix} \qquad S_2=\begin{Bmatrix} a & b & c & d \\ 0.8 & 0.1 & 0.06 & 0.04 \end{Bmatrix} \qquad S_3=\begin{Bmatrix} a & b & c & d & e \\ \frac{1}{2} & \frac{1}{4} & \frac{1}{8} & \frac{1}{16} & \frac{1}{16} \end{Bmatrix}$$

(1) シャノン・ファノ符号とその平均符号長を求めよ。

(2) ハフマン符号とその平均符号長を求めよ。

(3) 平均情報量と各平均符号長を比較せよ。

練習問題 7-2　情報源 $S=\begin{Bmatrix} a & b & c \\ p_a & p_b & p_c \end{Bmatrix}$ に対してハフマン符号を作成したとき、その平均符号長は $H(S)$ 以上であることを確認せよ。

ヒント:このときの平均符号長は、$p_a \geq p_b \geq p_c$ とすると、7C の通り $1+p_b+p_c=2-p_a$ と表される。p_a を固定したときの $H(S)$ の最大値は $p_b=p_c$ のときで、その値を p_a で表してみよ。

練習問題 7-3　アルファベットの出現確率(%)が以下のとき、それぞれのハフマン符号と、その平均符号長を求めよ。ただし、‘と”、;と:は同じ文字として扱ってよい。

(www7.plala.or.jp/dvorakjp/hinshutu.html の数値)

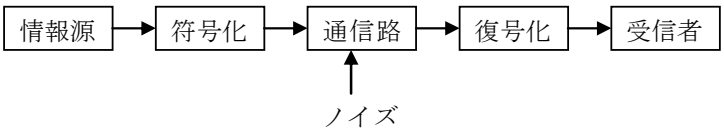
	11.41
a	8.45
t	8.18
i	7.13
o	7.05
s	6.97
n	6.77
r	6.22
h	4.26
l	3.87
d	3.87
c	e 3.20
u	2.95
m	2.67
p	2.02
f	1.98
g	1.82
y	1.79
w	1.70
b	1.64
,	1.62
.	1.08
v	1.00
k	0.90
‘ “	0.82
j	0.21
x	0.18
q	0.08
z	0.08
; :	0.08

8　通信路と相互情報量

これまでは情報を短く表現することを情報源符号化として考えてきた。しかし、短くすると１ビットでもエラーがあると大きな違いになる場合もある。エラーを考えると少し冗長に送らないといけない。どの程度冗長に送ればよいか。これを考えてみよう。

8A 通信路モデル

情報は符号化され、通信路を通過後、復号化され、元の情報に戻った後、受信者に渡る。通信路ではノイズ（雑音、外乱）により、正しく伝えられるとは限らない。つまり、誤って伝わる可能性がある。ノイズが大きいと、送信符号が別の符号に変化する確率が大きくなる。



8B 送信記号と受信記号

符号化された情報は、送信記号（symbol）A を使って伝送される。送信記号は、情報を表す記号とは別のものでもよい。送信記号は、アナログ信号で表された送信信号として伝送される。

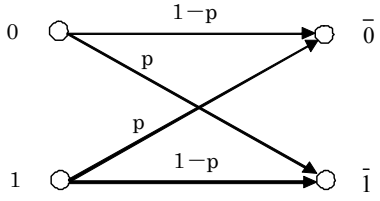
例えば、２進符号は、0 と 1 を表す電圧値や光の強さなどで伝送される。ノイズがあると、その電圧や光の強さは変化し、0 が 1 に間違って判断される可能性がある。受信側での判断結果を受信記号 B とし以下のように表す。受信記号 B は、混乱を避けるため、送信側の 0/1 とは違う $\bar{0}/\bar{1}$ で表記している。この例のように、２つの記号のいずれかで情報伝送が行われる通信路を２元通信路と呼ぶ。

$$A=\begin{Bmatrix} 0 & 1 \\ p(0) & p(1) \end{Bmatrix}, \qquad B=\begin{Bmatrix} \bar{0} & \bar{1} \\ p(\bar{0}) & p(\bar{1}) \end{Bmatrix}$$

8C 通信路行列と通信路線図

ノイズがなければ、0 は $\bar{0}$ に、1 は $\bar{1}$ に受信されるが、ノイズがあると、0 が $\bar{1}$ に受信されたり、1 が $\bar{0}$ に受信されることになる。その関係を、左縦に送信記号、上横に受信記号を割り当てた通信路行列 T で表すと以下のように表せる。右端は例で、誤って伝わる確率を、２つとも p としている。このように、いずれの方向も同じ確率で表される場合を、対称通信路と呼ぶ。通信路行列を表す右の図を通信路線図と呼ぶ。

$$T=\begin{bmatrix} p(\bar{0}\,|\,0) & p(\bar{1}\,|\,0) \\ p(\bar{0}\,|\,1) & p(\bar{1}\,|\,1) \end{bmatrix}=\begin{bmatrix} 1-p & p \\ p & 1-p \end{bmatrix}$$



$$(p(\bar{0}),p(\bar{1}))=(p(0),p(1))\begin{bmatrix} p(\bar{0}\,|\,0) & p(\bar{1}\,|\,0) \\ p(\bar{0}\,|\,1) & p(\bar{1}\,|\,1) \end{bmatrix}$$

8D 通信路で伝えられる情報量

先の記号 A、B を確率事象系ととらえると、送られる平均情報量を H(A)で、受信される平均情報量は H(B)である。もし、ノイズが無く、送られた記号が正しく、伝えられれば、H(B)=H(A)となる。そして、H(A)を大きくすれば、それだけのたくさん情報が伝えられる。例えば、２進記号でなく、もっとたくさんの記号（電圧であればたくさんの値のいずれか）で伝えるとよい。伝えられる情報量は、受信記号 B を受け取ることで得られる A に関する平均情報量、つまり以下の相互情報量となる。

$$I(A;B) = H(A) - H(A|B)$$

$$H(A|B) = \sum_k \sum_l p(a_k, b_l) \log_2 \frac{1}{p(a_k|b_l)}$$

この式で、ノイズがないとき、$p(a_k|b_l)$ は１か０であるので、H(A|B)=0 となり、I(A;B)=H(A)である。しかし、実際の通信路ではノイズが必ずあり、H(A|B)は 0 ではなくなり、送った情報量 H(A)より受け取る情報量 H(B)は小さくなる。

8C の通信路線図では、$p(b_l|a_k)$ が 1－p か p であることを表している。この値から $p(a_k|b_l)$（起こった結果から原因を推定するので事後確率と呼ばれる）を求めるにはベーズの定理を使う。

$$p(a_k|b_l) = \frac{p(b_l|a_k)p(a_k)}{\sum_i p(b_l|a_i)p(a_i)}$$

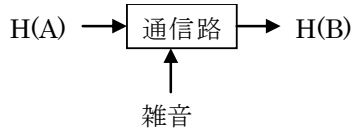
分母は $p(b_l)$ の計算値で、分子は $p(a_k, b_l)$ と同じである。8C の通信路線図に対して、以下のような計算になる。

$$p(0|\bar{0}) = \frac{p(\bar{0}|0)p(0)}{p(\bar{0}|0)p(0) + p(\bar{0}|1)p(1)}$$

p(0)と p(1)（=1－p(0)）が決まれば、$p(0|\bar{0})$ が求まる。

一方、$I(A;B) = H(B) - H(B|A)$ でもあるから、$p(b_l|a_k)$ から H(B|A)を計算すると事後確率を求めなくてもよいが、H(B)を決めるために $p(\bar{0})$ と $p(\bar{1})$（＝1－ $p(\bar{0})$）を求める必要がある。そして、受信した記号の平均情報量からノイズにより送信記号に等価的に加わった平均情報量を差し引く、という計算で $I(A;B)$ を求めることになる。

整理し直してみると、通信路に平均情報量 H(A)の記号をひとつ送ったときに、通信路行列 T で表される雑音の影響を受けた後、平均情報量 H(B)の記号のひとつとして受信される。



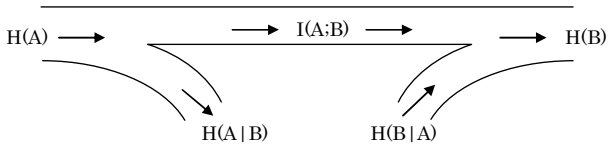
２進数の通信路では以下のように表される。

$$A = \begin{Bmatrix} 0 & 1 \\ p(0) & p(1) \end{Bmatrix}, \qquad B = \begin{Bmatrix} \bar{0} & \bar{1} \\ p(\bar{0}) & p(\bar{1}) \end{Bmatrix}, \qquad T = \begin{bmatrix} p(\bar{0}|0) & p(\bar{1}|0) \\ p(\bar{0}|1) & p(\bar{1}|1) \end{bmatrix}$$

このとき、通信路を通して伝わった平均情報量は、相互情報量 I(A;B)で表される。

$$I(A;B) = H(A) - H(A|B) = H(B) - H(B|A)$$

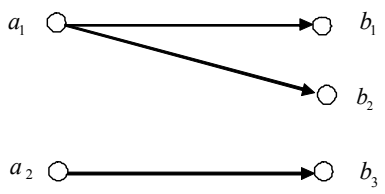
図解すると下図のようになり、通信路で消えてしまう情報量が $H(A|B)$、受信側に入り込むノイズが $H(B|A)$ と表現されていることになる。



H(A|B)は、B を知った上で、まだ A に関して残っている平均情報量を表し、H(B|A)は、A を知った上で、まだ B に関して残っている平均情報量を表す。その結果、H(A)－H(A|B)は送信情報量から、受信記号を元に送信記号を推定して推定しきれない分の情報量を差し引く計算である。そして、H(B)－H(B|A)は受信記号の情報量から送信記号が正しく伝えられない分の情報量を差し引く計算を表している。

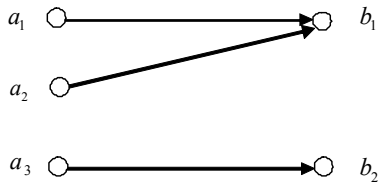
8E ノイズのない通信路

一般的には、送信記号の種類と受信記号の種類は一致しない可能性がある。しかし、ノイズの影響がなければ、受信記号から送信記号を特定できる。その場合の通信路線図を書くと、右図のように受信記号は、どの送信記号に対応するかが一意に決まっていることになる。このとき、H(A|B)=0 で、$I(A;B) = H(A) \le H(B)$ である.



8F 確定的通信路

受信記号が送信記号より少ないとき、受信能力が送信能力に対応できないことを意味する。送信記号に対する受信記号が決まっているという意味で確定的通信路と呼ばれ、右図のように表せる。このとき、H(B|A)=0 で、$I(A;B) = H(B) \le H(A)$ である.



8G 通信路容量

確定的通信路では、通信路で伝えられる相互情報量 I(A;B)は、受信記号で表される受信の仕方で制約される。また、ノイズのある通信路では、電圧などのわずかな違いを利用して送信記号を増やしても、違いが小さいと正しく伝えられなくなる。それらの結果として、通信路で伝えらる情報量の最大値を**通信路容量** C（channel capacity）と呼び、相互情報量の最大値として求める。送信記号と通信路行列を固定した場合は、送信記号それぞれの送信確率を変えたときの最大値として、以下のように max の下に変数等を示して表記される。

$$C = \max_{p(a_k)} I(A;B)$$

（ただし、これは 1 個の記号送信あたりの通信容量で、実用的には、秒当たり記号送信回数を乗じて、ビット/秒の単位で表現される。）

8H 通信路容量の計算例

送信受信とも２進記号で、以下のように設定したときの通信路容量を求めてみよう。 p は、１ビットのデータが誤る確率を意味するので、**ビットエラー率**(BER bit error rate)と呼ばれる。

$$A=\begin{Bmatrix}0 & 1\\ x & 1-x\end{Bmatrix},\quad B=\begin{Bmatrix}\bar{0} & \bar{1}\\ p(\bar{0}) & p(\bar{1})\end{Bmatrix},\qquad T=\begin{bmatrix}1-p & p\\ p & 1-p\end{bmatrix}$$

この場合、送信受信は２進記号で通信路行列も固定しているので、送信側の 0/1 の比率を変えて、相互情報量の最大値を求めることになる。

$$C=\max_x I(A;B)=\max_x\left\{H(B)-H(B\mid A)\right\}$$

を計算すればよい。

$$\begin{aligned}p(\bar{0}) &= p(\bar{0}\mid 0)p(0)+p(\bar{0}\mid 1)p(1)\\ &= (1-p)x+p(1-x)=x+p-2xp\ (=\alpha\text{とおく})\end{aligned}$$

$$\begin{aligned}p(\bar{1}) &= p(\bar{1}\mid 0)p(0)+p(\bar{1}\mid 1)p(1)\\ &= px+(1-p)(1-x)=1-x-p+2xp=1-\alpha\end{aligned}$$

$$H(B)=\alpha\log_2\frac{1}{\alpha}+(1-\alpha)\log_2\frac{1}{1-\alpha}$$

$$\begin{aligned}H(B\mid A) &= \sum_k\sum_l p(a_k,b_l)\log_2\frac{1}{p(b_l\mid a_k)}\\ &= \sum_k\sum_l p(b_l\mid a_k)p(a_k)\log_2\frac{1}{p(b_l\mid a_k)}\\ &= (1-p)x\log_2\frac{1}{1-p}+px\log_2\frac{1}{p}+p(1-x)\log_2\frac{1}{p}+(1-p)(1-x)\log_2\frac{1}{1-p}\\ &= p\log_2\frac{1}{p}+(1-p)\log_2\frac{1}{1-p}\end{aligned}$$

となる。H(B|A)は x に依存しないので、

$$C=\max_x\left\{H(B)-H(B\mid A)\right\}=\max_x\left\{H(B)\right\}-H(B\mid A)$$

$0\le x\le 1$ で $p\le\alpha\le 1-p$ （ $p\le 1/2$ として）なので、H(B)の最大値は１ビットとなり、

$$C=1-\left\{p\log_2\frac{1}{p}+(1-p)\log_2\frac{1}{1-p}\right\}$$

として、ビットエラー率 p で決まる値となる。

練習問題

8-1　送信記号 A、受信記号 B、通信路行列 T を以下で表すとき、

$$A=\begin{Bmatrix}0 & 1\\ 0.5 & 0.5\end{Bmatrix},\ B=\begin{Bmatrix}\bar{0} & \bar{1}\\ p(\bar{0}) & p(\bar{1})\end{Bmatrix},\ T=\begin{bmatrix}1-p & p\\ q & 1-q\end{bmatrix}$$

通信路線図を描き、I(A;B)を求めよ。

8-2　問題 8-1 で、$p=q=0.1$ のときの I(A;B)を求めよ。

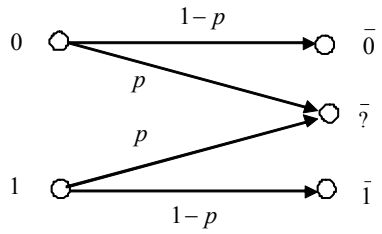
8-3　問題 8-1 で、$p=0.1$、$q=0.2$ のときの I(A;B)を求めよ。

8-4　送信記号 A、受信記号 B、通信路行列を以下で表すとき、

$$A=\begin{Bmatrix}0 & 1\\ x & 1-x\end{Bmatrix},\ B=\begin{Bmatrix}\bar{0} & \bar{1}\\ p(\bar{0}) & p(\bar{1})\end{Bmatrix},\ T=\begin{bmatrix}3/4 & 1/4\\ 1/4 & 3/4\end{bmatrix}$$

(1) 受信記号中に含まれるノイズ分の情報量を求めよ。
(2) 送信記号から受信記号への相互情報量を表す式を求めよ。
(3) 通信路容量を C を求めよ。

8-5　２元通信路（1 か 0 のいずれかで送信、受信する）であるが、受信側で 0/1 のどちらとも判定できず、?(不明)として判定する場合を考える。下図のような確率になっているとき、通信路容量を求めよ（送信側の確率を変化させて相互情報量の最大値を求めよ)。



ヒント：受信記号を３つと考えて、以下で定式化し、8-1 と同様に計算してみよ。

$$A=\begin{Bmatrix}0 & 1\\ x & 1-x\end{Bmatrix},\ B=\begin{Bmatrix}\bar{0} & \bar{?} & \bar{1}\\ p(\bar{0}) & p(\bar{?}) & p(\bar{1})\end{Bmatrix},\ T=\begin{bmatrix}1-p & p & 0\\ 0 & p & 1-p\end{bmatrix}$$

9　通信路符号

9A エラー検出と訂正

ノイズのある通信路では、送信記号がそのまま受信されない可能性がある。正しくない記号を受け取るということは、誤った情報を受け取るということになる。音声や画像では部分的に誤っても大きな問題ではないかもしれないが、情報によっては大問題となる。そこで、ディジタル情報の伝送では、エラー検出方法や、エラーを自動訂正できる方法を使う。

受信した情報にエラーかあるかどうか判定するためには、判定するための情報も受け取る必要がある。その結果、通信路に送信する符号は、本来の伝えたい情報とエラー検出・訂正用のデータを合わせたものになる。

通信路符号 ＝ 伝えたい情報 ✜ エラー検出・訂正用データ

9B パリティ検査

簡単なエラー検出法として、ASCII 文字伝送などを対象として**パリティ検査法**(parity check)がある。ASCII 文字は 7 ビットであるが、エラー検査用 1 ビットを合わせて 8 ビットとして伝送する。検査用ビットは、偶数パリティ（符号中の 1 の数が偶数になるように、検査用ビットを作る）もしくは、奇数パリティ（符号中の 1 の数が偶数になるように、検査用ビットを作る）が基本である。

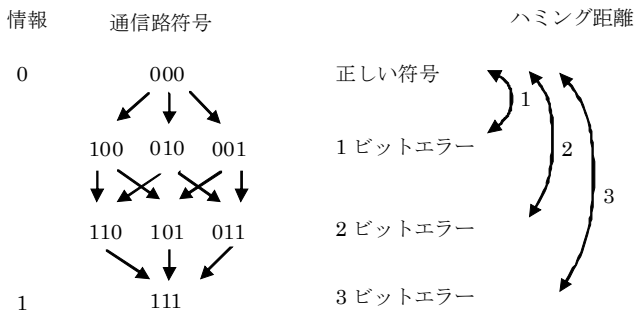
通信路符号：11100001 ＝ 偶数パリティ：1 ✜ 情報'a'：1100001

通信路符号：01100001 ＝ 奇数パリティ：0 ✜ 情報'a'：1100001

このパリティ法では、伝送符号のいずれかのビットで 1 ビットだけエラーがあれば、必ず分かることになる。もし、2 ビットのエラーがあると分からない。さらに言うと、奇数個のエラーであれば検出できるが、偶数個のエラーだとエラーがあっても検出できないことになる。したがって、エラーが起きても 1 ビットだけという場合に使えるエラー検出法である。

9C 多数決符号

エラー検出だけでなく、エラー訂正もできる通信路符号を考えてみよう。簡単な例として、3 ビットの多数決符号がある。1 ビットの伝送を 3 ビット繰り返して送るもので、1 ビットのエラーがあっても、2 ビットのエラーがあっても、受信側はエラーがあることが分かる。

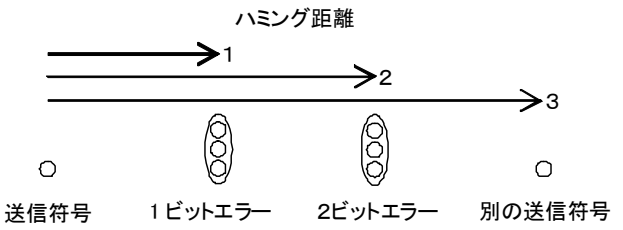


9D 多数決符号のエラー確率と訂正

上の 3 ビット多数決符号で、ビットエラー率を p とすると、

	符号	確率	$p<<1$のとき
エラー無し	000	$(1-p)^3$	≈ 1
１ビットエラー	100 010 001	$3p(1-p)^2$	$\approx 3p$
２ビットエラー	110 101 011	$3p^2(1-p)$	$\approx 3p^2$
３ビットエラー	111	p^3	$\approx p^3$

となる。p が小さい、例えば一般有線通信での最大エラー率の目安である $p=10^{-6}$ のとき、p^2 や p^3 はすごく小さくなる。そのため、000 が 100/010/001 になる確率は、111 が 100/010/001 になる確率よりずっと大きい。したがって、100/010/001 が受信された場合、正しくは 000 である場合がほとんどと考えられる。つまり、100/010/001 はエラーを含むことがわかるが､元は 000 であると考えても、ほとんど正しいことになる。つまり、確率の大きい方への訂正できると考えられる。これは､受信記号を１ビット訂正して 000 とすることに相当する。



9E ハミング距離(Hamming distance)

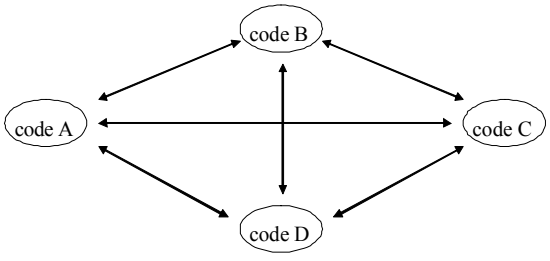
上のエラー訂正は、符号の違いが何ビットあるかを利用している。この違いのあるビットの数を、符号間の**ハミング距離**と呼ぶ。ハミング距離を機械的に求めるには、２符号の XOR(排他的論理和)をビットごとに計算して、結果の１の数を数えればよい。XOR を⊕で表すと、

0 ⊕ 0 = 0
0 ⊕ 1 = 1
1 ⊕ 0 = 1
1 ⊕ 1 = 0
10001 ⊕ 01101 = 11100（ハミング距離３）

9F ハミング距離とエラー検出・訂正能力

エラーが１ビットあると、ハミング距離が１の別の符号になる。したがって、エラーを検出するには、ハミング距離が 2 以上の符号で送信しないといけない。相互のハミング距離が２である符号だけを使って（そのいずれかで送信すると)、１ビットエラーのある符号は、存在しない符号になり、エラー検出ができる。相互のハミング距離が３の符号だけを使うと、２ビットまでのエラーが検出できる。

通信路符号を決めるとき、ハミング距離が一定値の符号の組を選ぶとよいが、すべての符号の距離を同じにするのが難しい場合は、必要なエラー検出・訂正能力を得られるよう、符号間ハミング距離の最小値 Dmin を設定すればよい。そのとき、すべての符号ペアに対して、エラー検出可能な最大ビット数は Dmin－1 となる。



4符号の場合、6つの符号間距離の最小値が重要

最小ハミング距離が３の符号組を使うと、間に１ビットもしくは２ビットのエラーで発生する符号

が存在する。**9D** で述べたように、ハミング距離の近い１ビットエラーは、エラー無しに訂正し、遠い２ビットエラーはそれに近い符号に訂正すると、１ビットエラー訂正ができる。ハミング距離が２ずつ増えるたびに、訂正ビット数を１増やすことができる。その結果、エラー訂正ができる最大ビット数 t_{max} は相互ハミング距離によって、以下の式で表せる。

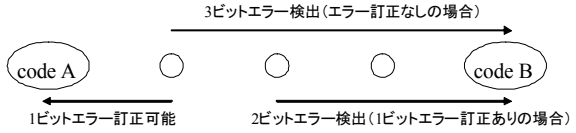
$$t_{\max} = \mathrm{int}\left[\frac{D\min - 1}{2}\right]$$

ここで、int[ ] は、整数部分である（余りがあるときは、切捨てを意味する）。

訂正を限界まででではなく、t ビットまで行うと、隣の正しい符号からハミング距離で t ビット手前までが隣の符号と判定される領域になるので、そのもうひとつ手前まで、エラーが起きたことを判断できる領域になり、エラー検出可能ビット数は、以下の e となる。

$$e = D_{\min} - 1 - t$$

例：Dmin=4 だと、t_{max} = 1 となり、１ビットのエラー訂正が可能である。１ビットのエラー訂正を行うと、2 ビットまでのエラーが検出可能である。もし、訂正を行わなければ、3 ビットまでのエラーが検出可能である。



練習問題

9-1　２ビットの情報を以下のような通信路符号で送るとき、

情報	通信路符号
00	00000
01	00111
10	11100
11	11011

(1) 最小ハミング距離を求めよ。

(2) 最大何ビットまでのエラーを検出できるか。

(3) 最大何ビットまでのエラーを訂正できるか。

9-2　最小ハミング距離が４で、２ビットの情報を送信するための符号組を作れ。そのとき、最大何ビットまでのエラーが検出でき、最大何ビットまでのエラーが訂正できるか。

10　通信路符号化定理

10A エラー訂正符号の訂正後エラー確率と符号効率

多数決符号で伝送エラーを訂正するときの、訂正後のエラー率を考えてみよう。9C/9D の例では、ビットエラー率を p としたとき、1 ビット訂正しても残るエラー率は、2 ビットエラーもしくは 3 ビットエラーが発生していたときの確率で、合計は

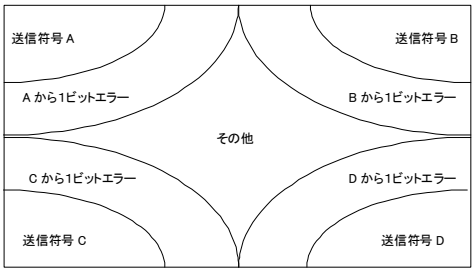
$$3p^2(1-p)+p^3=p^2(3+2p)\approx 3p^2$$

である。したがって、1 ビットで送ったときのエラー率 $p(<<1)$ より、結果的に小さいエラー率で伝送できることになる。さらに 5 ビット多数決、7 ビット多数決としていけば、最終的な（訂正後の）エラー率をいくらでも小さくできる。しかし、5 ビットで 1 ビットの情報、7 ビット送って 1 ビットの情報しか送れなくなるので、**符号効率**（伝送するビット数に対する情報ビット数の割合）は悪くなっていく。

10B k ビット情報に対する 1 ビットエラー訂正符号

1 ビットの情報を多数決符号とするのではなく、情報源符号化定理のときと同じように、何ビットかの情報の組み合わせに対して、多数決符号に相当するエラー訂正符号を使うことを考えてみよう。具体的には、k ビットの情報にエラー訂正のためのビットを加えて、合計 n ビットで伝送することを考える。全部で 2^n 種類の 2 進符号が受信される可能性があるが、その中に 2^k 種類の情報を表す符号があり、残りの 2^n-2^k 種類の符号はすべて伝送エラーで発生する符号ということになる。

そのエラー符号の中で、正しい符号にハミング距離が近いものは、訂正して解釈することにする。このとき、正しい符号のひとつからハミング距離が 1 のものは ${}_nC_1$ 個ある。正しい符号間の最小ハミング距離が 3 になるように符号を選べたとすると、すべての正しい符号からハミング距離 1 のものは、他の正しい符号からハミング距離が 1 のものとオーバラップしないはずである。元の符号と 1 ビットエラーの符号で合わせて $(1+{}_nC_1)$ 個の組が 2^k 個あり、それは全体で 2^n 以下であるはずなので、



$$(1+{}_nC_1)2^k=(1+n)2^k\le 2^n$$

である。したがって、

$$2^{n-k}\ge n+1\text{、つまり }n-k\ge\log_2(n+1)$$

であるはずである。この式は、1 ビットエラー訂正可能にするための、k ビットの情報に対するエラー訂正符号全体のビット数 n の下限を決めている。下限であるので、等号を満たす n よりひとつ以上大きい n を採用する必要があるかもしれない。

具体的に、k=1 では $n\ge 3$（k=1,n=3 は 3 ビット多数決符号である）、k=2 では $n\ge 5$、k=3 では $n\ge 6$ となる。k を大きくしていくと、$n-k$ も大きくせざるを得ないが、k の大きくなり方より、ゆっくりで $\log_2(n+1)$ 位である。その結果、k を無限に近づけると、符号効率 k/n は 1 に近づく。つまり、エラー訂正には余分なビット数を使うことになるが、たくさんを組み合わせた送り方をすれば、その余分な分は、全体に対する割合としては無視できるくらいまで小さくできる。

10C nビット符号での1ビット訂正の場合

k ビットの情報を n ビットで伝送したときの訂正後のエラー確率は、ビットエラー率 p に対して、2ビットエラー確率は $nC_2\,p^2(1-p)^{n-2}$、3ビットエラー確率は $nC_3\,p^3(1-p)^{n-3}$ で、合計おおよそ $n(n-1)p^2/2$ である。10A の値より大きくなるが、エラー訂正により、結果としての元の p よりエラー率を下げることは変わりがなく、符号効率は 1/3 より大きい値 k/n にできる。

10D nビット符号でのtビット訂正後の場合

t ビットまで訂正したいときは、ハミング距離を 2t+1 以上にすればよい。そのとき、正しい符号のひとつからハミング距離が1のものは ${}_nC_1$ 個、ハミング距離2のものは ${}_nC_2$ 個、ハミング距離 t のものは ${}_nC_t$ 個ある。正しい符号間の最小ハミング距離が 2t+1 になるように符号を選べたとすると、すべての正しい符号からハミング距離 t のものは、他の正しい符号からハミング距離が t のものとオーバラップしないはずである。したがって、

$$(1+{}_nC_1+\cdots+{}_nC_t)2^k \le 2^n$$

なので、送りたい情報 k ビットに対して、この式を満たすように n を決めればよい。10B と同様、

$$n-k \ge \log_2(1+{}_nC_1+\cdots+{}_nC_t) \approx \log_2{}_nC_t \approx t\log_2(n/t)$$

となり、k を大きくしていっても、$n-k$ の大きくなり方は k の大きくなり方より小さい。したがって、k を∞に近づけた極限では、符号効率 k/n は1に近づく。t ビットエラー訂正後のエラー率は、t+1 ビット以上エラーがある確率の合計なので、$nC_{t+1}\,p^{t+1}(1-p)^{n-t-1}$ 程度と見積もれる。

10E 通信路符号化定理

実際の伝送路では、1秒間に何ビット送れるかを bps(bit per second)もしくはビット/秒で表し、通信路容量と呼ぶ（8G と同じ名前であるが、こちらは時間も考慮した能力である)。これは、1回の伝送ビット数に秒当たり伝送回数を乗じたものである。

前節までの検討のように、最小ハミング距離を大きくすることで、訂正後のエラー確率を任意に小さくでき、かつ、符号効率も1に近い伝送ができることになる。このことを、通信路容量に対応して表現すると、**「通信路容量の範囲内で情報を送りながら、いくらでもエラー確率を小さくすることができる」**ということになり、**通信路符号化定理**と呼ばれる。完全にエラー率を0にすることはできないが、エラー率を限り無く0に近いエラー訂正符号を使いながら、通信路容量 C に限りなく近い情報を伝送することができることを意味する。

通信路符号化定理（Channel Coding Theorem）

ノイズの影響を受ける離散通信路の通信容量を C bps とし、ある情報源からの情報を R bps の速度で符号化して、この通信路を通して伝送する。このとき、

(1) もし $R \le C$ であれば、この情報源の情報をいくらでも小さいあいまい度（$H(\text{送信記号}\,|\,\text{受信記号})$）で伝送することができるよう符号化することができる。

(2) $R > C$ であれば、この情報を $R-C+\varepsilon$ より小さいあいまい度で伝送することができる。ただし、ε は任意の小さな正数である

(3) また、$R > C$ の場合、この情報を $R - C$ より小さいあいまい度で伝送することができない。

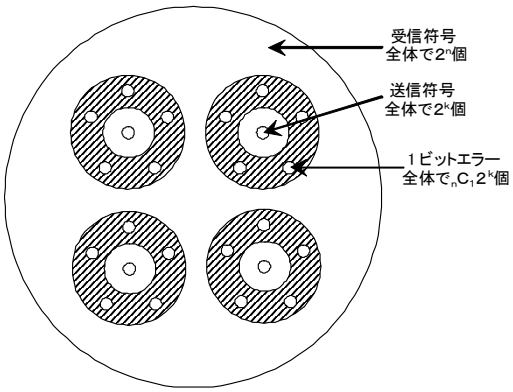
10F 通信路符号作成手順

1) ビットエラー率 p を測定などで求める
2) 必要とする訂正後エラー率 ε を決める
3) ε を実現するための符号間最小ハミング距離を求める
4) 期待する通信路符号の効率を決める
5) それに合う、1 符号に含める情報ビット数 k と符号長 n を決める
6) 符号の組み（2^k 個）を見つける

ただし、この方法で決めた符号を使うときは、符号を変換テーブルに記憶しておく必要があり、情報ビット数が増えて符号数が多いと記憶場所や処理の課題がある。より実用的なエラー訂正符号としては、BCH 符号、リード・ソロモン符号、畳み込み符号、ターボ符号、LDPC 符号などがある。

10G 通信路符号化定理のまとめ

(1) ハミング距離を大きくすることで、訂正後のエラー率を任意に下げることができる。
(2) エラー訂正のために余分に必要になるビット数は、$\log_2[\text{訂正範囲の符号数}]$ 程度である。
(3) その余分に必要なビット数は全符号長 n に対して $\log_2[n\text{の多項式}]$ のオーダであるため、n を無限大に近づけると、情報を表すビット数 k に対して k/n は 1 に近づく。つまり、エラー訂正のために余分に必要なビット数は、全体の中で無視できるくらい小さくでき、エラー訂正符号の効率は、いくらでも良くできる。k が 12,000 のとき（イーサネットの最大サイズ 1500B の場合）、n は 12,014 程度という具合で、ビット数（対数）で評価する n−k は全体からするとすごく小さい割合にできる。



練習問題

10-1 通信容量が 9000bps の通信路で、ビットエラー率が 10^{-6} のとき、

(1) 情報ビット 1 ビットをそのまま伝送するときビット当たりのエラー率と、情報の伝送速度（bps）を求めよ。
(2) 情報ビット 1 ビットを 3 ビット多数決符号で伝送するときのビット当たりのエラー率（概算値）と、情報の伝送速度（bps）を求めよ。
(3) 情報ビット 2 ビットを練習問題 9-1 に記載のエラー訂正符号で伝送するときの情報ビット当たりのエラー率（概算値）と、情報の伝送速度（bps）を求めよ。

10-2　２ビットの情報を２ビットエラー訂正可能符号で伝送したい。どのような符号にすればよいか考えよ。符号組が見つかったら、訂正後エラー確率（情報ビット当たりの概算値）と符号効率を求めよ。

11　線形符号

11A 組織符号

10F で求めた符号を使うとき、符号をすべて記録しておいて送信時に変換し、受信時も記録符号と比較して判断しないといけない。ビット数が多くなると符号の記憶場所も変換時間も問題になる。そこで、送信すべき情報から、ある計算式で求めたものを、エラー検査・エラー訂正用データ（以下では検査データと略す）としてを付加できるとよい。受信したデータも、計算式に当てはめて、エラー判断、訂正を行えるとよい。そのための方式が種々考案されてきた。その場合、

送信符号(n ビット)＝送信情報(k ビット)＋検査データ(n-k ビット)

のように、情報と検査データが明確に区別されるので、組織符号（systematic code）と呼ばれる。

11B 線形符号

検査データの計算に線形演算を行うものを線形符号と呼ぶ。線形演算は、定数（行列を含む）の乗算を行う計算である。情報を要素数 k の横ベクトル $\mathbf{x}=(x_1, x_2, \ldots, x_k)$、検査データを要素数 n－k の横ベクトル $\mathbf{p}=(p_1, p_2, \ldots, p_{n-k})$、$\mathbf{x}$ から $\mathbf{p}$ を計算する定数値行列 $\mathbf{C}$ を以下の行列（情報・検査ビット関連行列）とすると、

$$\mathbf{C}=\begin{bmatrix} c_{11} & \cdot & c_{1\,n-k} \\ c_{21} & \cdot & c_{2\,n-k} \\ \cdot & \cdot & \cdot \\ c_{k1} & \cdot & c_{k\,n-k} \end{bmatrix}$$

$$\mathbf{p}=\mathbf{xC}$$

と表せる。この式は、以下の連立式を意味する。

$$p_1 = c_{11}x_1 + c_{21}x_2, \ldots, c_{k1}x_k$$
$$p_2 = c_{12}x_1 + c_{22}x_2, \ldots, c_{k2}x_k$$
$$\cdots$$
$$p_{n-k} = c_{1\,n-k}x_1 + c_{2\,n-k}x_2, \ldots, c_{k\,n-k}x_k$$

符号の議論では 0 と 1 が並ぶような符号を考えることが多く、横書きに納まりやすいように横ベクトルで符号を表すことが多い。このとき、$\mathbf{x}$ の要素も $\mathbf{p}$ の要素も 1 か 0 であるので、$\mathbf{C}$ の要素も 1 か 0 で考えればよい。さらに行列計算の結果が、0 か 1 のいずれかになるように、$\mathbf{xC}$ の各乗算後の加算では、XOR を使う。つまり、0+0=0、0+1=1、1+0=1、1+1=0 として計算する。このようにしても、$(\mathbf{x}+\mathbf{y})\mathbf{C}=\mathbf{xC}+\mathbf{yC}$ の関係は維持され、線形性は失われない。

送信する符号は、$\mathbf{x}$ と $\mathbf{p}$ をつないだもので、横ベクトル $\mathbf{u}$ で表すと、

$$\mathbf{u}=(\mathbf{x},\mathbf{p})=(x_1, x_2, \ldots, x_k, p_1, p_2, \ldots, p_{n-k})$$

と表記される。

11C 生成行列

送信符号 **u** を情報 **x** から一気に計算する式に変形すると、

$$\mathbf{u} = (\mathbf{x}, \mathbf{p}) = (\mathbf{x}, \mathbf{xC}) = \mathbf{x}[\mathbf{I}_k, \mathbf{C}] = \mathbf{xG}$$

$$= (x_1, x_2, \cdots, x_k, p_1, \cdots, p_{n-k}) = (x_1, x_2, \cdots, x_k) \begin{bmatrix} 1 & & 0 & c_{11} & . & c_{1\,n-k} \\ & \ddots & & . & . & . \\ 0 & & 1 & c_{k1} & . & c_{k\,n-k} \end{bmatrix}$$

と表せる。**G** は送信符号を生成する行列であるので、**生成行列**と呼ぶ。$\mathbf{I}_k$ は縦横 k 要素の単位行列（対角部分だけが 1）である。

$$\mathbf{I}_k = \begin{bmatrix} 1 & 0 & . & 0 \\ 0 & 1 & . & 0 \\ . & . & . & . \\ 0 & 0 & . & 1 \end{bmatrix}$$

11D エラー検査

エラー検査も線形計算で行う。具体的に、受信符号 $\mathbf{y} = (y_1, y_2, \cdots, y_n)$ に対して、次の計算を行う。後にわかるように、エラー検査が実現するので、**H** は**検査行列**と呼ばれる。

$$\mathbf{s} = \mathbf{yH} = \mathbf{y} \begin{bmatrix} \mathbf{C} \\ \mathbf{I}_{n-k} \end{bmatrix}$$

$$= (s_1, s_2, \cdots, s_{n-k}) = (y_1, y_2, \cdots, y_n) \begin{bmatrix} c_{11} & . & c_{1\,n-k} \\ . & . & . \\ c_{k1} & . & c_{k\,n-k} \\ 1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

エラーがなければ、受信符号は送信符号と同じになるので、$\mathbf{y} = \mathbf{u} = \mathbf{x}[\mathbf{I}_k, \mathbf{C}]$ である。これを上の式に代入すると、

$$\mathbf{s} = \mathbf{yH} = \mathbf{x}[\mathbf{I}_k, \mathbf{C}] \begin{bmatrix} \mathbf{C} \\ \mathbf{I}_{n-k} \end{bmatrix}$$

$$= (x_1, x_2, \cdots, x_k) \begin{bmatrix} 1 & & 0 & c_{11} & . & c_{1\,n-k} \\ & \ddots & & . & . & . \\ 0 & & 1 & c_{k1} & . & c_{k\,n-k} \end{bmatrix} \begin{bmatrix} c_{11} & . & c_{1\,n-k} \\ . & . & . \\ c_{k1} & . & c_{k\,n-k} \\ 1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$= (x_1, x_2, \cdots, x_k) \begin{bmatrix} c_{11} + c_{11} & c_{1\,n-k} + c_{1\,n-k} \\ & \\ c_{k1} + c_{k1} & c_{k\,n-k} + c_{k\,n-k} \end{bmatrix} = (0, 0, \cdots, 0) = \mathbf{0}$$

となって、左辺のベクトルは$\mathbf{0}$ベクトルになる。

もし、エラーがあるときは、そのエラーをまとめて表すベクトルを$\mathbf{e} = (e_1, \cdots, e_i, \cdots, e_n)$と表すと、$\mathbf{u} = (u_1, \cdots, u_i, \cdots, u_n)$に対して、

$$\mathbf{y} = \mathbf{u} + \mathbf{e} = (u_1 + e_1, \cdots, u_i + e_i, \cdots, u_n + e_n)$$

と表せる。エラーのあるビットでは要素e_iが１となり、XOR の計算から、そのビットのy_iはu_iを反転したものとなる。このとき、上の検査行列の計算をすると、

$$\mathbf{s} = (\mathbf{u} + \mathbf{e})\mathbf{H} = \mathbf{u}\mathbf{H} + \mathbf{e}\mathbf{H} = \mathbf{0} + \mathbf{e}\mathbf{H} = \mathbf{e}\mathbf{H}$$

となって、エラーベクトル$\mathbf{e}$と検査行列$\mathbf{H}$の積になる。１ビットだけのエラーであれば、$\mathbf{e}$のどれかのビットが１になり、$\mathbf{H}$の対応する行が$\mathbf{s}$となって出力される。$\mathbf{H}$のどの行も要素の少なくともひとつは０ではないように決めておけば、$\mathbf{s}$は$\mathbf{0}$ではなくなり、エラーがあることが分かる。$\mathbf{s}$はエラーの兆候を示すので、**シンドローム**と呼ばれる。

11E １ビットエラー検査符号の例

１ビット情報に偶数パリティビットを付加するときの生成行列$\mathbf{G}$は、１行２列の行列として、以下のように書ける。

$$\mathbf{G} = \begin{bmatrix} 1 & 1 \end{bmatrix}$$

送信情報が０のときは(0, 0)、１のときは(1, 1)、つまり、送信情報を２回ずつ繰り返す送信符号が生成される。検査行列$\mathbf{H}$は、

$$\mathbf{H} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

と表せ、受信信号$\mathbf{y}$が (0, 0)もしくは(1, 1)では、シンドローム$\mathbf{s} = \mathbf{y}\mathbf{H}$は０となる。１ビットエラーの(0, 1)もしくは(1, 0)では１となり、エラーを表す。

２ビット情報に偶数パリティビットを付加するときの生成行列$\mathbf{G}$と検査行列$\mathbf{H}$は、

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}, \quad \mathbf{H} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

で、２ビットの送信情報に対して、以下のように送信符号が生成される。

$$\begin{array}{ll} (0,0) & (0,0,0) \\ (0,1) & (0,1,1) \\ (1,0) & (1,0,1) \\ (1,1) & (1,1,0) \end{array} \quad \Rightarrow$$

１の数が偶数になるように、元の情報に１ビットが付加されている。この４つに対して、シンドローム $\mathbf{s}=\mathbf{yH}$ はすべて 0 となる。送信符号に対して１ビットエラーの(0, 0, 1)、(0, 1, 0)、(1, 0, 0)、(1, 1, 1)では、すべて１となる。

k ビットデータに偶数パリティビットを付加するときの生成行列 $\mathbf{G}$ は最後の列がすべて１で、k ビット中の１の XOR をとったものになる。一方、検査行列 $\mathbf{H}$ はすべて１の列が１つだけで、受信符号の１の XOR をとったものになるので、受信符号に１が偶数個あれば０となり、奇数個であれば１となる。

$$\mathbf{G}=\begin{bmatrix}1 & 0 & . & 0 & 1\\ 0 & 1 & . & 0 & 1\\ . & . & . & . & .\\ 0 & 0 & . & 1 & 1\end{bmatrix},\qquad \mathbf{H}=\begin{bmatrix}1\\ 1\\ 1\\ .\\ 1\end{bmatrix}$$

11F ２ビット以上のエラー検査

２ビットエラーがあれば、$\mathbf{s}=\mathbf{eH}$ にしたがい検査行列の２行の XOR がシンドロームになる。どの２行の XOR 結果もゼロベクトルにならないのであれば、２ビットエラーは必ず検出できることになる。同様にすべての m 行の XOR の結果がゼロベクトルにならないのであれば、m ビットまでのエラーが検出できることになる。

11G １ビットエラー訂正符号の例

２ビット情報を１ビットエラー訂正可能にする送信符号は、以下の生成行列 $\mathbf{G}$ で生成できる。

$$\mathbf{G}=\begin{bmatrix}1 & 0 & 1 & 0 & 1\\ 0 & 1 & 0 & 1 & 1\end{bmatrix},\qquad \begin{aligned}(0,0)G&=(0,0,0,0,0)\\ (0,1)G&=(0,1,0,1,1)\\ (1,0)G&=(1,0,1,0,1)\\ (1,1)G&=(1,1,1,1,0)\end{aligned}$$

ここで、(0, 1)の送信符号は $\mathbf{G}$ の下行と一致し、(1, 0)の送信符号は $\mathbf{G}$ の上行と一致している。そして、(1, 1)の送信符号は、それらの送信符号の XOR に一致する。検査行列 $\mathbf{H}$ は、

$$\mathbf{H}=\begin{bmatrix}1 & 0 & 1\\ 0 & 1 & 1\\ 1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 1\end{bmatrix}$$

と表せ、上の４つの送信符号ではシンドローム $\mathbf{s}=\mathbf{yH}$ は、すべて(0, 0, 0)となる。その他の受信符号は１ビットもしくは２ビットエラーに相当し、下図のように、シンドロームがゼロベクトルではなくなる。１ビットエラーであれば、シンドロームは検査行列 $\mathbf{H}$ の対応する行と同じ並びになる。これは、１ビットエラーであると仮定すれば、訂正可能であることを示す。

00000 -> 000		01011 -> 000
10000 -> 101	11000 -> 110	11011 -> 101
01000 -> 011	10010 -> 111	00011 -> 011
00100 -> 100	01100 -> 111	01111 -> 100
00010 -> 010	00110 -> 110	01001 -> 010
00001 -> 001		01010 -> 001
00101 -> 101	00111 -> 111	01110 -> 101
11101 -> 011	01101 -> 110	10110 -> 011
10001 -> 100	11001 -> 110	11010 -> 100
10111 -> 010	10011 -> 111	11100 -> 010
10100 -> 001		11111 -> 001
10101 -> 000		11110 -> 000

11H エラー訂正可能な条件

上の例では、エラーのあるビットが１ビットだけならば、検査行列のエラービットに対応する行がシンドロームになる。その結果、受信符号と検査行列から求めたシンドローム**s**が**0**でないとき、そのベクトルと一致する検査行列の列を探せば、エラーのあったビットが分かる。つまり、そのエラーを訂正することができる。

この方法が使えるためには、検査行列のすべての行は、他の行と一致していていてはいけない。１ビットのエラー訂正ができるためには、送信符号間のハミング距離は３以上で、２ビットまでのエラーは必ず検出できる。そのため、検査行列の中の２行までの行の XOR はすべてゼロベクトルにはならない。したがって、検査行列のどの列も、他の列と一致しないはずである。

２ビット以上のエラー訂正についても、11D の式、

$$\mathbf{s} = \mathbf{eH}$$

でシンドローム**s**から２ビット以上のエラーベクトル**e**を一意に決定できるような**H**となっていれば可能である。

練習問題

11-1 情報・検査ビット関連行列**C**を以下の値とするとき、

$$\mathbf{C} = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix}$$

(1) 生成行列**G**を具体的に書け。

(2) ３ビットの情報それぞれに対する、送信符号を求めよ。

(3) 検査行列を具体的に書け。

(4) (2)で求めた送信符号に対して、シンドロームはすべて**0**となることを確認せよ。

(5) 情報 000 と 111 に対する受信符号で､各ビットにエラーがあったときのシンドロームを求めよ。

11-2 ３ビットの情報に偶数パリティビットを付加して送信符号を作るときの生成行列$\mathbf{G}$を求め、すべての３ビット情報に対応する送信符号を書け。次に検査行列$\mathbf{H}$を求め、１ビットのエラーを含む受信符号で、シンドロームが１になることを確認せよ。

11-3 情報・検査ビット関連行列$\mathbf{C}$が次のように与えられたとき、

$$\mathbf{C}=\begin{bmatrix}1 & 1 & 0 & 1\\ 1 & 0 & 1 & 1\\ 0 & 1 & 1 & 1\end{bmatrix}$$

(1) 生成行列$\mathbf{G}$を求めよ。

(2) 送信符号をすべて求め、最小ハミング距離を求めよ。

(3) 検査行列$\mathbf{H}$を求めよ。

(4) 受信符号が 1011011 のとき、エラーがあったかどうか調べよ。エラーがあったとき、エラーが１ビットと仮定して、正しい符号を求めよ。

11-4 生成行列$\mathbf{G}$と検査行列$\mathbf{H}$が以下のとき、何ビットのエラーまで検出できるか。

$$\mathbf{G}=\begin{bmatrix}1 & 0 & 1 & 1 & 1 & 0\\ 0 & 1 & 0 & 1 & 1 & 1\end{bmatrix},\qquad \mathbf{H}=\begin{bmatrix}1 & 1 & 1 & 0\\ 0 & 1 & 1 & 1\\ 1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}$$