情報理論第二（8）
誤り検出・誤り訂正符号

人間コミュニケーション学科
梶本　裕之
kajimoto@hc.uec.ac.jp

1

レポート回収

2

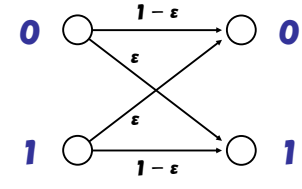
授業の流れ（予定）

第1週（10/ 8） 情報量と情報エントロピー
第2週（10/15） 相互情報量
第3週（10/22） 体育祭のため休講
第4週（10/29） 情報源符号化とデータ圧縮
第5週（11/ 5） ハフマン符号とデータ圧縮
第6週（11/12） 情報源符号化定理
第7週（11/19） 調布祭のため休講
第8週（11/26） 出張のため休講
第9週（12/ 3） マルコフ情報源モデル
第10週（12/10） 通信路のモデル化
第11週（12/17） 誤り検出・誤り訂正符号
第12週（12/24） 出張のため休講
第13週（ 1/ 7） 線形符号
第14週（ 1/14） ハミング符号
第15週（ 1/21） 秘密鍵暗号
第16週（ 1/28） 公開鍵暗号
第17週（ 2/ 4） 出張のため休講

3

（前回の復習）2元対称通信路モデル

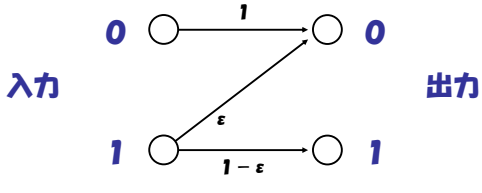
- 1文字あたり確率εの割合でビット反転が起こる（つまりエラー！！）



4

（前回の復習）2通信路モデル

- 1→0（もしくは0→1）の1方向だけ通信時のエラーが生じる通信路
（例：高速光通信，メモリ等）



5

（前回の復習）通信路容量

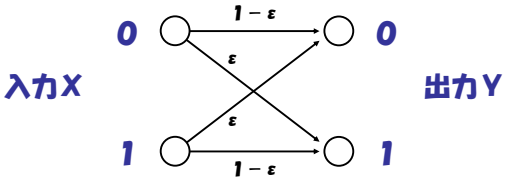
- 入力Xと出力Yの間の相関が高いほど性能はよい（出力Yがランダムだと悪い）
- XとYの間の相互情報量で性能を表す

$$I(X;Y) = H(X) + H(Y) - H(XY)$$
$$= H(Y) - H(Y \mid X)$$

- Xの確率分布が変わるとI(X;Y)も変わる
- I(X;Y)の最大値を特に通信路容量と呼ぶ。

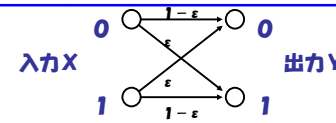
（前回の復習）通信路容量の例

・下記の２元対称通信路における I(X:Y) の最大値（通信路容量）を求めよ．



7

（前回の復習）通信路容量の例



入力X：P(0)=pとする．P(1)=1−p
出力Y：P(0)=qとする．P(1)=1−q．qはpで表せるが

・P(Y|X)：

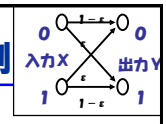
P(0|0)=1−ε，P(0|1)=ε，P(1|0)=ε，P(1|1)=1−ε

・P(Y, X)=P(Y|X)×P(X)：

P(0, 0)=p(1−ε)，P(0, 1)=(1−p)ε，
P(1, 0)=pε，P(1, 1)=(1−p)(1−ε)

8

（前回の復習）通信路容量の例



H(Y)=−q×log(q)−(1−q)×log(1−q)
=h(q)とおく

H(Y|X)=−P(0, 0)×log(P(0|0))−P(0, 1)×log(P(0|1))…
=−(1−ε)×log(1−ε)−ε×log(ε)
= h(ε)

→H(Y|X)は，pに依存しない！！
→ →I(Y:X)=H(Y)−H(Y|X)の最大値は，H(Y) 最大のとき．

H(Y) =h(q)の最大値はq=0.5のときで，1
（第一回レポート）
よって，max(I(Y:X))=1− H(Y|X)
=1 − h(ε)

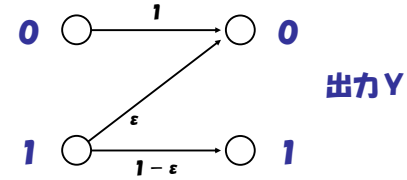
意味
エラー無し：H(Y|X)=0，1bitを確実に送れる
エラーだらけ：ε=0.5，H(Y|X)=1，通信路容量は0

9

前回の小レポート(1)

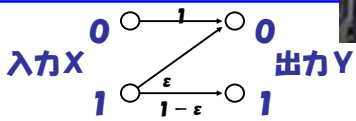
・下記のZ通信路における I(X:Y) の最大値（通信路容量）を求めよ．

（X・Yとも，1の方の確率を変数で表すときれいな形で書ける）



10

前回の小レポート(1)



入力X：P(1)=pとする．P(0)=1−p
出力Y：P(1)=qとする．q=p(1−ε)

・P(Y|X)：

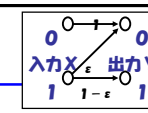
P(0|0)=1，P(0|1)=ε，P(1|0)=0，P(1|1)=1−ε

・P(Y, X)=P(Y|X)×P(X)：

P(0, 0)=(1−p)，P(0, 1)=pε，P(1, 0)=0，P(1, 1)=p(1−ε)

11

前回の小レポート(1)



H(Y)=−q×log(q)−(1−q)×log(1−q)=h(q)

H(Y|X)=−P(0, 0)×log(P(0|0))−…
=−pε×log(ε)−p(1−ε)×log(1−ε)
= p×h(ε)

q=p(1−ε) より，H(Y|X)= q×h(ε)/(1−ε)

I(Y:X) = H(Y)−H(Y|X) = h(q) − q×h(ε)/(1−ε)

極値は∂I/∂q=0の時だから，
∂I/∂q = −log(q)+log(1−q)− h(ε)/(1−ε) = 0

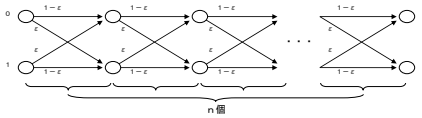
∴q=1/(1+2^{−h(ε)/(1−ε)})

代入して，max(I)=log(1+2^{−h(ε)/(1−ε)})

意味：ε=0で，1(bit)，ε=1で0(bit)．

12

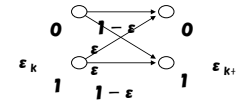
前回の小レポート(2)



- 誤り確率がεである2元対称通信路を，上図のようにn個つなげた通信路を考える．以下の問に答えよ．

1. k個の通信路を通り抜けた時点での誤り確率，すなわち最初に0を入力したときk個目の通信路の出口に1が出力されている（あるいはその逆の）確率をε_kとする．ε_{k+1}をε_kとεを用いて表せ．
2. ある定数とε_kの差分に着目すると，1.で得られた漸化式を等比数列の形に書き直すことができる．この性質を利用してε_nを求めよ．
3. この通信路全体の通信路容量をεとnで表せ．またn→∞の場合，通信路容量はどのようになるかを示せ．

前回の小レポート(2)



1. k個の通信路を通り抜けた時点での誤り確率，すなわち最初に0を入力したときk個目の通信路の出口に1が出力されている（あるいはその逆の）確率をε_kとする．ε_{k+1}をε_kとεを用いて表せ．

$$\varepsilon_{k+1}=(1-\varepsilon)\varepsilon_k+\varepsilon(1-\varepsilon_k)$$
$$=(1-2\varepsilon)\varepsilon_k+\varepsilon$$

前回の小レポート(2)

2. ある定数とε_kの差分に着目すると，1.で得られた漸化式を等比数列の形に書き直すことができる．この性質を利用してε_nを求めよ．

$$\varepsilon_{k+1}=(1-2\varepsilon)\varepsilon_k+\varepsilon$$

ヒントから，$\varepsilon_{k+1}-a=b(\varepsilon_k-a)$とおく．

$$\varepsilon_{k+1}=b\varepsilon_k-ab+a.$$

係数を比較して，$b=(1-2\varepsilon)$，$a=1/2$．

よって，$(\varepsilon_{k+1}-1/2)=(1-2\varepsilon)(\varepsilon_k-1/2)$．

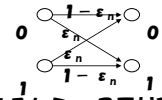
$f_k=\varepsilon_k-1/2$とおく．$f_1=\varepsilon-1/2$．

$f_n=(1-2\varepsilon)f_{n-1}=(1-2\varepsilon)^2f_{n-2}=\cdots=(1-2\varepsilon)^{n-1}(\varepsilon-1/2)$

$\therefore \varepsilon_n=1/2+(1-2\varepsilon)^{n-1}(\varepsilon-1/2)$

前回の小レポート(2)

3. この通信路全体の通信路容量をεとnで表せ．またn→∞の場合，通信路容量はどのようになるかを示せ．



上図のように表せるから，2元対称通信路の通信容量の式に$\varepsilon=\varepsilon_n$を代入して

通信容量$=1-h(\varepsilon_n)$．

ε_nは明らかに1/2に収束．

このとき$h(\varepsilon_n)$は1となり，通信容量は0に収束する．

通信路符号化と誤り訂正・誤り検出符号

通信路容量の意味

- 通信路容量は符号の伝送レートの上界を与える．
- すなわち，

限りなく通信路容量に近い伝送レートの符号がつくれる（＝通信路符号化定理）

- 類似性：情報源エントロピーは平均符号長の下界を与えた（情報源符号化定理）

やりたいこと

- エラーを減らしたい
- もし送信した符号が間違っていたら？
  - 誤っていることを知りたい
  - 誤りを訂正したい
  - 実はこの二つはちょっと違う．

19

エラーを減らす＝冗長性

- 符号化の際に冗長性を与える
- →通信路において誤りが生じても，受信側である程度検出・復元可能にできる
- 例：　0 → 000，1 → 111 と符号化

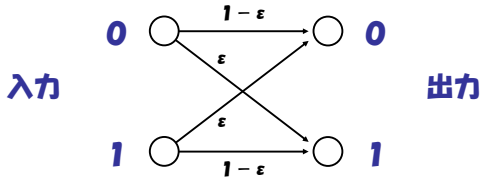
復号化は多数決論理で

000 → 0,
001 → 0,
010 → 0,
100 → 0,
111 → 1,
110 → 1,
101 → 1,
011 → 1

20

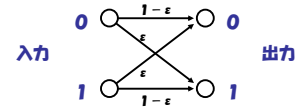
例題

- 以下の通信路において 0→000，1→111 と符号化してデータを送信した場合に，データを正しく復元できる確率を求めよ．



21

例題



コーディング手法：0→000，1→111．

3回中，　回以上正しければ多数決で戻せる．

・000→000の確率：

・000→001,010,100の確率：

合計：

22

誤りの検出と訂正

- ある符号化された語を受信したときに，受信されたビット列を見ただけで

通信途中で誤り（ビットの反転）が混入したことが分かる→ 誤り検出符号

さらに，どの部分がどのように反転したかが分かり，元のビット列を復元できる
→ 誤り訂正符号

23

誤りの検出と訂正

- 実際には「誤り検出符号」「誤り訂正符号」という特別な符号があるのではない．

「この符号化方法なら，

1符号語あたり○ビットの誤りまで検出可能，

□ビットの誤りまで訂正可能」

（○誤り検出・□誤り訂正符号）

という

24

例

- 0 → 000, 1 → 111 と符号化し，
  000, 001, 010, 100 → 0,
  111, 110, 101, 011 → 1 と復号化

この符号は
1符号語あたり2ビットの誤りまで検出可能，
1符号語あたり1ビットの誤りまで訂正可能
（2誤り検出・1誤り訂正符号）

25

例題

- 0 → 00…0, 1 → 11…1 と，それぞれn個の繰り返しを符号語とした場合，この符号の誤り検出・誤り訂正能力はいくらになるか．

（回答）
- bitまで誤りを検出できる．
- bit まで誤りを訂正できる．

26

誤り訂正・誤り検出符号とハミング距離

「冗長性」をもっと詳しく．

- 符号化の際に冗長性を与える
- →通信路において誤りが生じても，受信側である程度検出・復元可能にできる

なぜか？

28

Richard W. Hamming (1915-1998)

"Error detecting and error correcting codes"
The Bell Sys. Tech. J. 29: 147-160, 1950.

- ベル研における Shannon の同僚
- 誤り検出・誤り訂正符号の理論を確立
- ハミング窓(Fourier変換)
- マンハッタン計画（原爆が大気を燃やし尽くさないことを計算）

ハミング距離

- 同じ長さを持つ2つのビット列がどの程度はなれているかをあらわす量

$a = a_1a_2\ldots a_n$, $b = b_1b_2\ldots b_n$ について

$$d_H(a, b) = \Sigma_{i=1\sim n} | a_i - b_i |$$

a と b の各桁を比較したときに異なっているビットの数

30

例題

- 以下のビット列それぞれの間のハミング距離はいくらか.

(a) 10010100

(b) 01110101

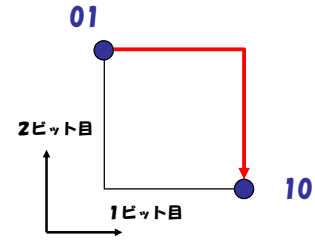
(c) 01100111

(回答) $d_H(a,b)$ = $d_H(b,c)$ = $d_H(a,c)$ =

31

ハミング距離の視覚的イメージ

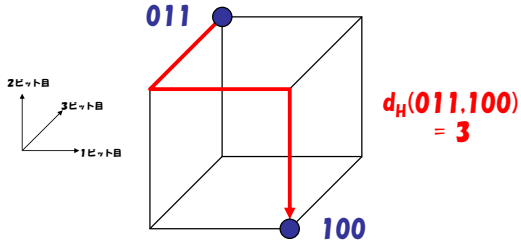
- 2ビットの場合：平面で考えると



32

ハミング距離の視覚的イメージ

- 3ビットの場合：空間で考えると



ハミング距離＝n次元立方体上で何回で到達できるか

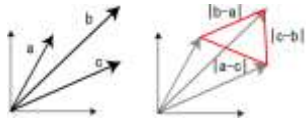
33

ハミング距離は「距離」なのか

- 数学的に「距離」と呼ばれるものは以下の4条件を満たす：
  - $f(a, b) \geqq 0$
  - $f(a, b) = 0 \Leftrightarrow a = b$
  - $f(a, b) = f(b, a)$
  - $f(a, b) + f(b, c) \geqq f(a, c)$ （三角不等式）

34

参考：ユークリッド距離

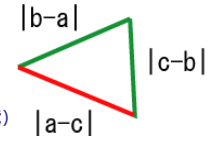


a,b,c：位置ベクトル

ユークリッド距離 $f(a,b)=|a-b|=\sqrt{(a_x-b_x)^2+(a_y-b_y)^2}$

意味：ベクトルa-bの長さ(ユークリッド長)

$f(a, b) \geqq 0$
- $f(a, b) = 0 \Leftrightarrow a = b$
- $f(a, b) = f(b, a)$
- $f(a, b) + f(b, c) \geqq f(a, c)$ （三角不等式）



ハミング距離は「距離」なのか

- ハミング距離 $d_H(a, b)$ も上記の性質すべてを満たす（つまり数学的な意味での「距離」である）
  - $f(a, b) \geqq 0$
  - $f(a, b) = 0 \Leftrightarrow a = b$
  - $f(a, b) = f(b, a)$
  - $f(a, b) + f(b, c) \geqq f(a, c)$ （三角不等式）

36

小レポート(1)

- ハミング距離の間に三角不等式

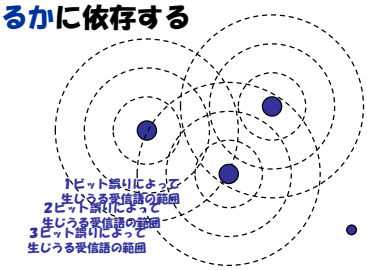
$$f(a, b) + f(b, c) \geqq f(a, c)$$

が成立する理由を考えよ.

37

符号語間のハミング距離と誤り

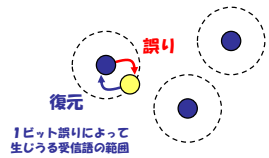
- 符号の誤り検出・誤り訂正能力は，各符号語間のハミング距離がどれほど離れているかに依存する



38

誤り訂正

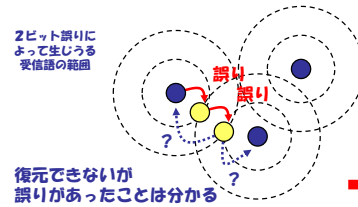
- 誤りによって広がった各符号語の受信語の範囲が互いにオーバーラップしない限り，元の符号語が何であったかを特定し，復元できる



39

誤り検出

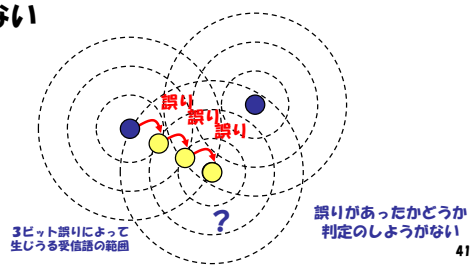
- 誤りによって広がった各符号語の受信語の範囲が他の符号語を含まない限り，何らかの誤りが生じたことを検出できる



40

誤り訂正も検出もできない場合

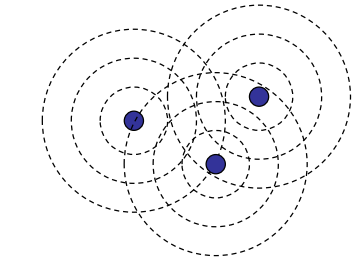
- 受信語の範囲に他の符号語が入った場合，それが誤りによるものかどうかを特定できない



41

従って．．．

- 下図の例は「2誤り検出可能・1誤り訂正可能」の例である

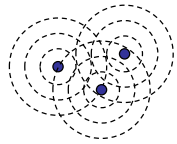


42

ハミング距離と符号

- 一般に，各符号語間のハミング距離が最低でもKと保証されているならば，その符号は最低でも
  - k より少ない誤りについて検出可能
  - k/2 より少ない誤りについて訂正可能

となる



43

例：パリティビット方式

- 元の語に含まれる1の数が偶数なら0，奇数なら1を末尾に加える．
- →1の数は常に偶数個となる

例：011011
→0110110
110111
→1101111

誤り検出：1bit

誤り訂正：出来ない

エラーの起きる確率が極めて低いときに利用

44

パリティビット方式のHamming距離

- 正しい符号は常に1の数が偶数個
  - ○101011
  - ×101010
  - ○101000
- つまり，Hamming距離は2．

- 公式を適用：
  - 誤り検出：
  - 誤り訂正：

45

(参考) データの「消失」に関して

- パリティ符号は，データの「反転」を訂正できないが，「消失」は訂正できる．

1010?1

↓

101011

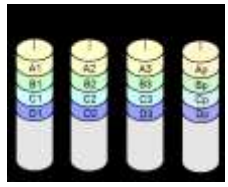
46

(参考)ハードディスクとRAID

- ハードディスクが壊れても復旧するには？
- RAID1：二つのハードディスクに同じデータをコピー
- RAID4,5：パリティビットを使用
- ハードディスクは壊れるときは全体が壊れる（データの反転ではなく「消失」）
- →パリティビットからデータを復旧可能

(1)1 0 1 1
(2)1 1 0 1
(3)1 1 0 0
(p)1 0 1 0

(1)1 0 1 1
(2)x y z w
(3)1 1 0 0
(p)1 0 1 0



x=parity(1,1,1)=1
y=parity (0,1,0)=1
z=parity (1,0,1)=0
w=parity (1,0,0)=1

47

例：2重パリティビット方式

- m×n ビットからなる元の語を行列にして，行ベクトル列ベクトルおよび全体のそれぞれにパリティビットを付ける方式

100101 → 100 / 101 → 1001 / 1010 / 0011 → 100110100011

48

例：2重パリティビット方式

- ある場所が間違うと．．．

(1) 1001 1010 0011　(2) 1001 1000 0011　(3) 1001 1000 0011　(4) 1001 1010 0011

(2) 一箇所間違える
(3)間違った行，列がそれぞれ特定できる
(4)その交点の符号を訂正

49

例：2重パリティビット方式

- 二箇所間違うと．．．

(1) 1001 1010 0011　(2) 1001 1100 0011　(3) 1001 1100 0011

(2) 2箇所間違える
(3)おかしい列だけ特定できる
どこを訂正したらよいかは分からない

50

2重パリティビット方式のHamming距離

1001
1010
0011

1001 → 1001
1000 → 1000
0011 → 0011

1001 → 1001
1100 → 1100
0011 → 0011

1011 → 1011
1100 → 1100
0011 → 0011

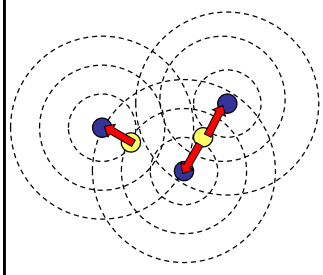
1111
1100
0011

- 4箇所間違うと「正しい」符合になる
- すなわちHamming距離=4
- 公式より
- 誤り検出：3bit
- 誤り訂正：1bit

51

復号と誤訂正

最小距離復号法



1. 受信したビット列を全ての符号語と比較，ハミング距離を測る
2. 得られた距離の中で1つだけ最小のものがあったら，それを採用
3. 最小のものが複数あったら，訂正不能とする

53

実際には．．．

- 確率的にふるまう通信路では，1符号語あたりの誤りの数の上限は限定できないので，「誤りが1個以内である」「2個以内である」などと信じて誤り訂正を行うしかない → まれに誤訂正が起こる
- 逆に，符号語間の最小距離よりも大きな誤りがあっても，符号語によっては訂正可能な場合もある

54

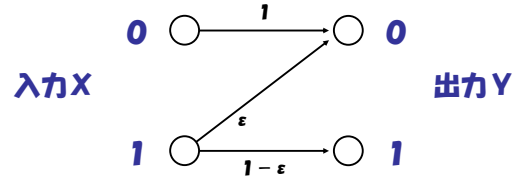
他の復号化方法：最尤復号法

- 通信路の誤り発生確率があらかじめ判っているならば，**各符号語が受信語のように変化する確率**を計算して「**最もありえる（最尤）**」元の語を選択する方法
  - 例えば誤りの起こり方にかたよりがある場合，確率的に最もありえる符号語とハミング距離最小の符号語とは必ずしも一致しない

55

例題

- Z通信路において 0 → 000，1 → 111 なる符号を用いた場合，受信語 010 はどのように復号すればよいか．



- （回答）

56

小レポート(2)

- 1ビットごとの誤り発生率が ε であるような2元対称通信路でnビットの符号語を送ることを考える．この符号語間の最小距離が 2k+1 (k は整数) であった場合，実際に正しく復号できる確率，および誤訂正が生じる確率を，それぞれ式で表せ．

（ただし符号語空間は目一杯混雑していて，k+1回以上の誤りが発生した場合は常に他の符号語の誤り訂正範囲に入ってしまうものと仮定する）

57