

2009 年度春期

# 応用数学Ⅰ

(数学科)

# 情報数学特論

(理工学専攻情報学領域)

(担当: 角皆)

## 本講義の概要

- 情報通信の数理

- ★ 情報理論

- ★ 符号理論

- ★ 暗号理論

- それを支える数学

- ★ 有限体とその上の線型代数・Galois 理論

- ★ 情報量の理論・計算量の理論

## 情報通信を行なう際の要請

- 効率的に → 情報理論
- 確実に → 符号理論
- 安全に → 暗号理論

## 情報通信を行なう際の要請

- 効率的に → 情報理論
- 確実に → 符号理論
- 安全に → 暗号理論

## 情報理論 (情報源符号化・情報量の理論)

- 伝えるべき情報をより効率良く伝えるには
- 「効率の良さ」を計る
  - ★ 伝えるべき「情報の量」を計る
  - ★ 伝える為の「手間」を計る

→ Shannon

「情報の量は伝えるのに必要な手間と一致」

## 情報理論 (情報源符号化・情報量の理論)

- 伝えるべき情報をより効率良く伝えるには
- 「効率の良さ」を計る
  - ★ 伝えるべき「情報の量」を計る
  - ★ 伝える為の「手間」を計る

→ Shannon

「情報の量は伝えるのに必要な手間と一致」

## 情報理論 (情報源符号化・情報量の理論)

- 伝えるべき情報をより効率良く伝えるには
- 「効率の良さ」を計る
  - ★ 伝えるべき「情報の量」を計る
  - ★ 伝える為の「手間」を計る

→ **Shannon**

「情報の量は伝えるのに必要な手間と一致」

## 符号理論 (誤り訂正符号)

- 通信路での雑音による誤りを  
検出・訂正するための符号方式

- 誤りを検出・訂正するには

- ★ 「冗長性」を持たせる
- ★ しかしなるべく効率良く

→ 効率の良い符号の構成のために  
様々な代数的性質を利用  
(線型符号・代数幾何符号など)

## 符号理論 (誤り訂正符号)

- 通信路での雑音による誤りを  
検出・訂正するための符号方式
- 誤りを検出・訂正するには
  - ★ 「冗長性」を持たせる
  - ★ しかしなるべく効率良く

→ 効率の良い符号の構成のために  
様々な代数的性質を利用  
(線型符号・代数幾何符号など)

## 符号理論 (誤り訂正符号)

- 通信路での雑音による誤りを  
検出・訂正するための符号方式
- 誤りを検出・訂正するには
  - ★ 「冗長性」を持たせる
  - ★ しかしなるべく効率良く

→ 効率の良い符号の構成のために  
様々な代数的性質を利用  
(線型符号・代数幾何符号など)

## 暗号理論 (共通鍵・公開鍵暗号)

- 安全な情報生活の為に
  - ★ 秘密通信
  - ★ デジタル認証・署名
  - ★ 秘密分散
  - ★ 鍵共有
- 安全な暗号の実現  
(RSA 暗号・楕円曲線暗号など)
- 安全性を計る (計算量の理論)

## 暗号理論 (共通鍵・公開鍵暗号)

- 安全な情報生活の為に
  - ★ 秘密通信
  - ★ デジタル認証・署名
  - ★ 秘密分散
  - ★ 鍵共有
- 安全な暗号の実現  
(RSA 暗号・楕円曲線暗号など)
- 安全性を計る (計算量の理論)

## 基礎となる数理の予備知識

代表的には、例えば次のような事柄

	基礎編	初級編
情報理論	微分積分・線型代数・確率論	
符号理論	有限体上の 線型代数	整数論・群論・ 代数幾何の初歩
暗号理論	初等整数論 (素数の話)	

他に、計算の理論 (計算可能性・計算量) など

## 情報通信を行なう際の要請

- 効率的に → 情報理論
- 確実に → 符号理論
- 安全に → 暗号理論

## 情報理論 (情報源符号化・情報量の理論)

- 伝えるべき情報をより効率良く伝えるには
- 「効率の良さ」を計る
  - ★ 伝えるべき「情報の量」を計る
  - ★ 伝える為の「手間」を計る

→ **Shannon**

「情報の量は伝えるのに必要な手間と一致」

## 情報の符号化

アナログデータ (連続データ)

↓ サンプルングなど

デジタルデータ (離散・有限データ)

↑ ここで扱う “情報” (元データ・情報源)

↓ ← ここで扱う “符号化”

通信用の形式のデジタルデータ  
(特定 (一般には少数) の種類の文字の列)

## 情報の符号化

アナログデータ (連続データ)

↓ サンプルングなど

デジタルデータ (離散・有限データ)

↑ ここで扱う “情報” (元データ・情報源)

↓ ← ここで扱う “符号化”

通信用の形式のデジタルデータ

(特定 (一般には少数) の種類の文字の列)

## 文字データ → **ASCII code**

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	(control characters)															
1	(control characters)															
2		!	"	#	\$	%	&	'	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	'	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	

**7 bit** で 1 文字を表す (実装上は **8 bit** にする)

## ASCII code

<b>A</b>	01000001	<b>J</b>	01001010	<b>S</b>	01010011
<b>B</b>	01000010	<b>K</b>	01001011	<b>T</b>	01010100
<b>C</b>	01000011	<b>L</b>	01001100	<b>U</b>	01010101
<b>D</b>	01000100	<b>M</b>	01001101	<b>V</b>	01010110
<b>E</b>	01000101	<b>N</b>	01001110	<b>W</b>	01010111
<b>F</b>	01000110	<b>O</b>	01001111	<b>X</b>	01011000
<b>G</b>	01000111	<b>P</b>	01010000	<b>Y</b>	01011001
<b>H</b>	01001000	<b>Q</b>	01010001	<b>Z</b>	01011010
<b>I</b>	01001001	<b>R</b>	01010010		

## モールス符号 (Morse code)

<b>A</b>	· -	<b>J</b>	· - - -	<b>S</b>	...
<b>B</b>	- ...	<b>K</b>	- · -	<b>T</b>	-
<b>C</b>	- · - ·	<b>L</b>	· - ..	<b>U</b>	.. -
<b>D</b>	- ..	<b>M</b>	- -	<b>V</b>	... -
<b>E</b>	·	<b>N</b>	- ·	<b>W</b>	· - -
<b>F</b>	.. - ·	<b>O</b>	- - -	<b>X</b>	- .. -
<b>G</b>	- - ·	<b>P</b>	· - - ·	<b>Y</b>	- · - -
<b>H</b>	....	<b>Q</b>	- - · -	<b>Z</b>	- - ..
<b>I</b>	..	<b>R</b>	· - ·		