

Goppa & Background Knowledge

<https://youtu.be/BCmEclBZXbg>

최승주

Contents

대수

실수 좌표 공간

해밍 코드

선형 부호

고파 부호



대수(Algebra)

- 정수론(수론): 수학의 한 분야, 각종 수의 성질을 대상으로 함 (가우스가 많은 기여)
- 대수학: 수 대신에 문자를 사용하여 방정식의 풀이 방법이나 대수적 구조를 연구하는 학문

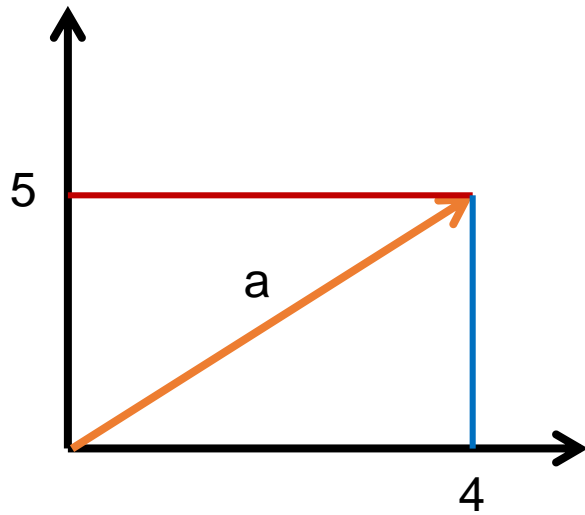
$$\begin{array}{lcl} \text{ex) } 10 * X & = & 5000 \\ 10 * X / 10 & = & 5000 / 10 \end{array}$$

$$X = 500$$

실수 좌표 공간(Real Coordinate Space)

- Vector: 속력 + 방향

\overrightarrow{AB}



$$\vec{a} = \begin{bmatrix} 4 \\ 5 \end{bmatrix} \quad \begin{array}{l} \text{가로} \\ \text{세로} \end{array}$$

실수 좌표 공간(Real Coordinate Space)

- \mathbb{R}^2 \mathbf{R}^2 : 2차원 실수 좌표 공간
 - \mathbb{R} : 실수 좌표 공간
 - 2: 차원
- 가능한 모든 실수의 2 튜플
튜플: 순서가 정해진 숫자들의 리스트
- 실수 2개의 순서 리스트

ex) $\begin{bmatrix} 4 \\ 5 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} \begin{bmatrix} -3 \\ 4 \end{bmatrix}$

실수 좌표 공간(Real Coordinate Space)

- \mathbb{R}^3 \mathbf{R}^3 : 3차원 실수 좌표 공간

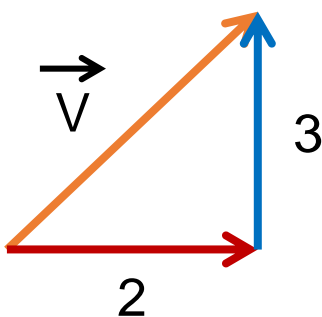
- 실수 3개의 튜플

ex) $\begin{bmatrix} 4 \\ 5 \\ 3 \end{bmatrix} = \vec{x} \quad \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad \begin{bmatrix} 1 \\ -3 \\ 4 \end{bmatrix}$

$\begin{bmatrix} 1 \\ -3 \\ 4 \\ 8 \end{bmatrix} (X)$

실수 좌표 공간(Real Coordinate Space)

- \mathbb{R}^n \mathbf{R}^n : n 차원 실수 좌표 공간
- 단위 벡터(unit vector)
정수의 1과 같은 역할


$$\vec{v} = \begin{bmatrix} 2 \\ 3 \end{bmatrix} \quad (2,3)$$

실수 좌표 공간(Real Coordinate Space)

- \mathbb{R}^n \mathbf{R}^n : n차원 실수 좌표 공간

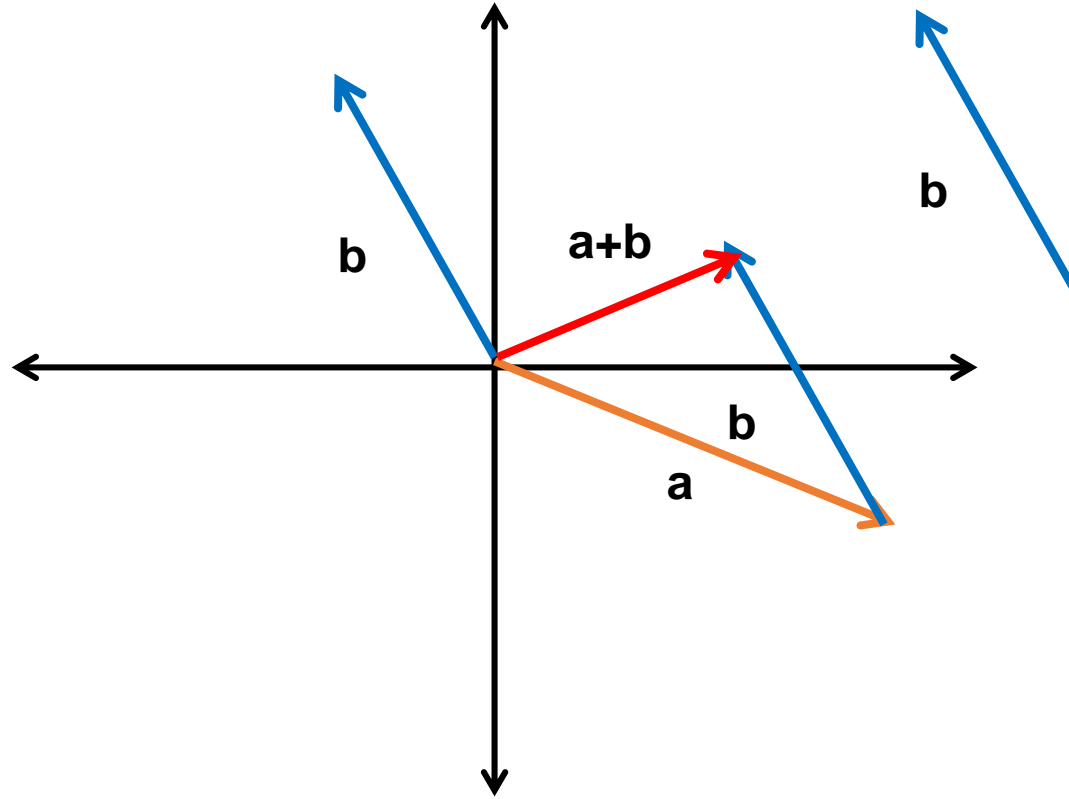
- 단위 벡터(unit vector)

정수의 1과 같은 역할

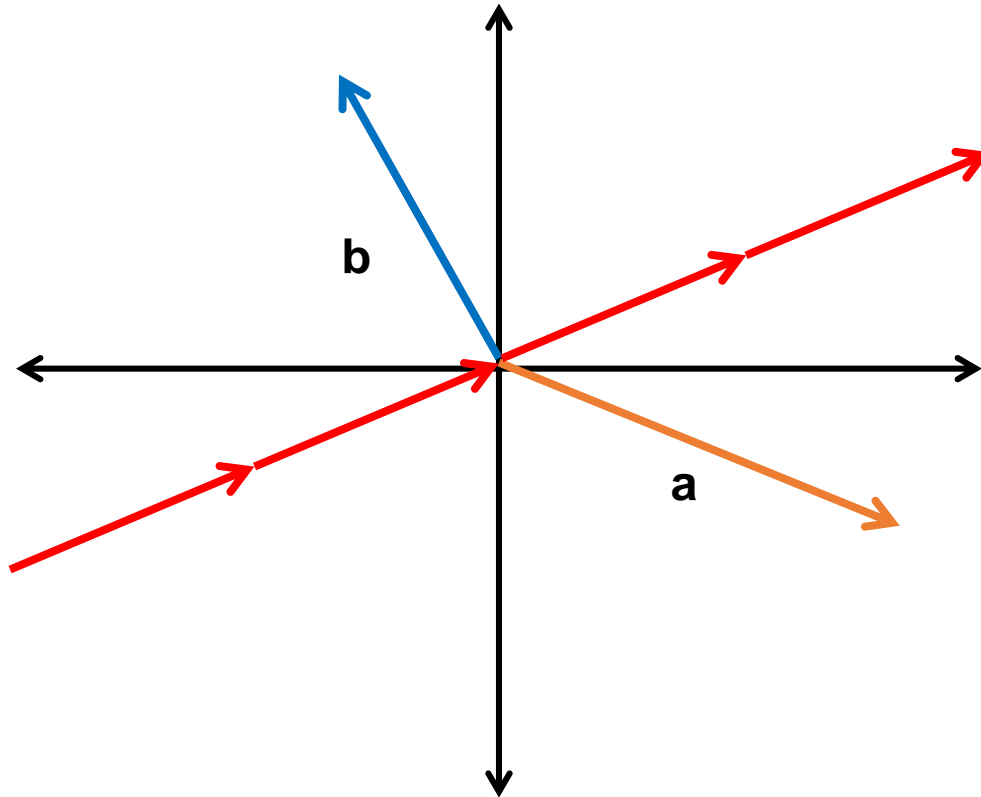
$$\begin{array}{c} \overrightarrow{v} = \begin{bmatrix} 2 \\ 3 \end{bmatrix} \quad (2,3) \end{array} \quad \begin{array}{c} \text{수평} \uparrow \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\ \text{수직} \uparrow \begin{bmatrix} 0 \\ 1 \end{bmatrix} \end{array} \quad \overrightarrow{v} = 2*\hat{i} + 3*\hat{j}$$

실수 좌표 공간(Real Coordinate Space)

- \mathbb{R}^n \mathbf{R}^n : n 차원 실수 좌표 공간



실수 좌표 공간(Real Coordinate Space)



$$\vec{v} = 2\hat{i} + 3\hat{j}$$

$$Y = aX + b$$

해밍 코드(Hamming Code)

- 오류 정정 부호의 일종
- 이전 선형 블록
- 1950년 Bell 연구소에서 고안
- 최대 2 비트 오류를 감지하거나 1 비트 오류를 수정할 수 있음
- 신드롬: 오류 검사에 사용되는 유일한 패턴

해밍 코드(Hamming Code)

- 홀수 짝수 패리티 비트 개념
- 1101(1의 개수 3) → 1101¹(짝수)
- 11111(1의 개수 홀수)로 전송이 되면 오류가 발생 했다는 것을 알 수 있음

해밍 코드(Hamming Code)

- 해밍 부호는 어떤 길이의 데이터어(data word)에도 사용 가능
- 해밍 코드는 n 개의 데이터어에 k 개 패리티 비트를 더하여 새로운 코드어(code word)를 생성한다.

해밍 코드

- 오류 감지 및 오류 수정 가능?

Ex)

(7, 4) 해밍 코드

- 4개의 비트 메시지
- 3개의 패리티 비트 사용
- 총 7개의 비트

해밍 코드

- (7, 4) 해밍 코드

메시지 비트: $x_3 x_2 x_1 x_0$

패리티 비트: $p_4 p_2 p_1$

구조

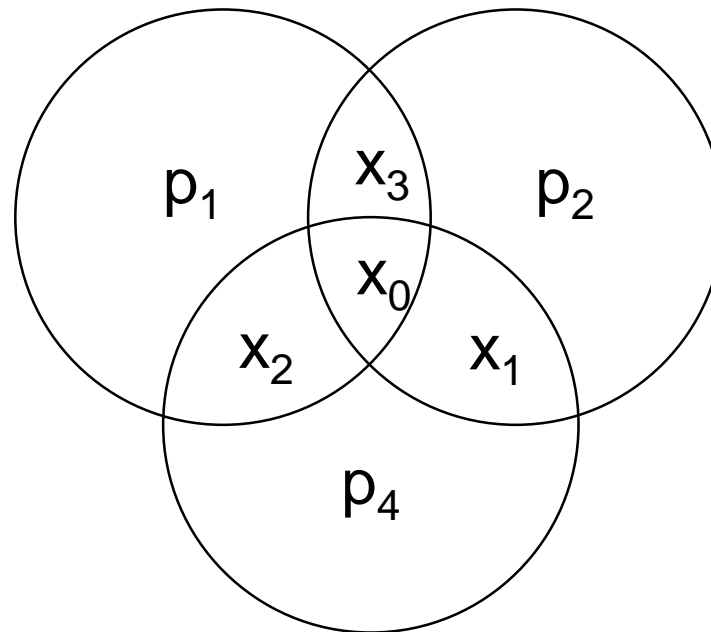
1	2	3	4	5	6	7
p_1	p_2	x_3	p_4	x_2	x_1	x_0

해밍 코드

- (7, 4) 해밍 코드

구조

1	2	3	4	5	6	7
p_1	p_2	x_3	p_4	x_2	x_1	x_0



해밍 코드

- (7, 4) 해밍 코드

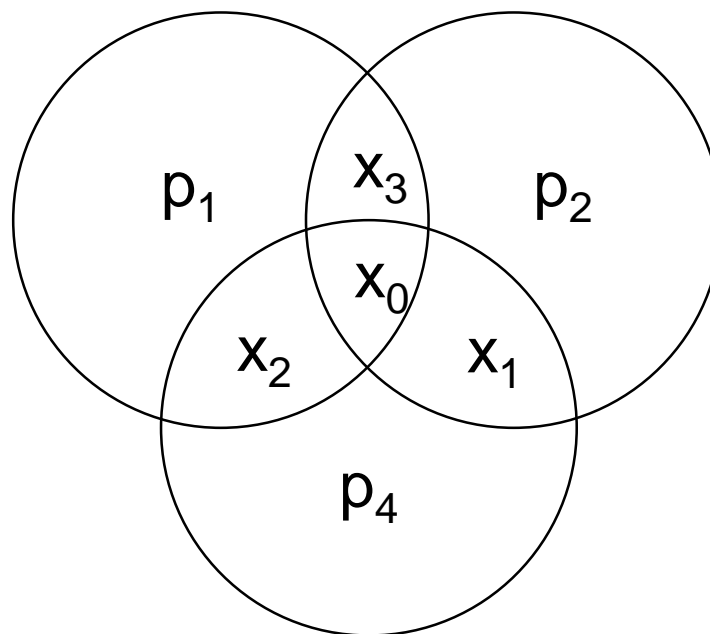
구조

1	2	3	4	5	6	7
p_1	p_2	x_3	p_4	x_2	x_1	x_0

- $P1 \rightarrow 1, 3, 5, 7$ 담당

$$P1 = x_3 \oplus x_2 \oplus x_0$$

1	001
3	011
5	101
7	111



해밍 코드

- (7, 4) 해밍 코드

구조

1	2	3	4	5	6	7
p_1	p_2	x_3	p_4	x_2	x_1	x_0

- $P1 \rightarrow 1, 3, 5, 7$

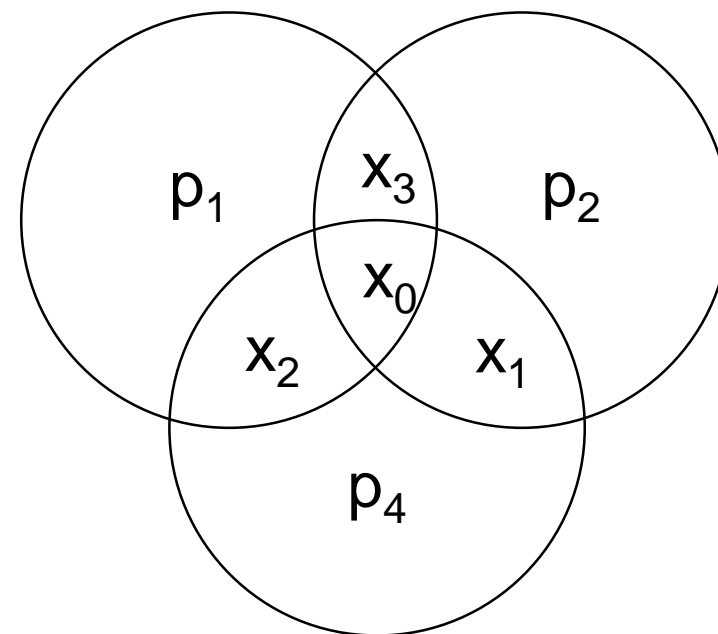
$$P1 = x_3 \oplus x_2 \oplus x_0$$

1	001
3	011
5	101
7	111

- $p2 \rightarrow 2, 3, 6, 7$

$$P2 = x_3 \oplus x_1 \oplus x_0$$

2	010
3	011
6	110
7	111

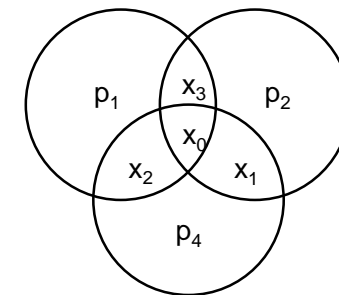


해밍 코드

- (7, 4) 해밍 코드

구조

1	2	3	4	5	6	7
p_1	p_2	x_3	p_4	x_2	x_1	x_0



- $P1 \rightarrow 1, 3, 5, 7$

$$P1 = x_3 \oplus x_2 \oplus x_0$$

1	001
3	011
5	101
7	111

$p2 \rightarrow 2, 3, 6, 7$

$$P2 = x_3 \oplus x_1 \oplus x_0$$

2	010
3	011
6	110
7	111

$p4 \rightarrow 4, 5, 6, 7$

$$P4 = x_2 \oplus x_1 \oplus x_0$$

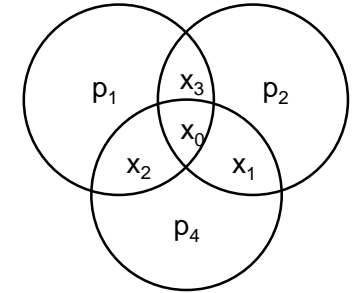
4	100
5	101
6	110
7	111

해밍 코드

- (7, 4) 해밍 코드

구조

1	2	3	4	5	6	7
p ₁	p ₂	x ₃	p ₄	x ₂	x ₁	x ₀



- P1 → 1, 3, 5, 7

$$P1 = x_3 \oplus x_2 \oplus x_0$$

1	001
3	011
5	101
7	111

p2 → 2, 3, 6, 7

$$P2 = x_3 \oplus x_1 \oplus x_0$$

2	010
3	011
6	110
7	111

p4 → 4, 5, 6, 7

$$P4 = x_2 \oplus x_1 \oplus x_0$$

4	100
5	101
6	110
7	111

해밍 코드

- $C_1 = P1 + x_3 + x_2 + x_0$

- $C_2 = P2 + x_3 + x_1 + x_0$

- $C_4 = P4 + x_2 + x_1 + x_0$

C의 결과로 어디에 에러가 발생했는지 확인 가능

해밍 코드

- $C_1 = P1 + x_3 + x_2 + x_0$
- $C_2 = P2 + x_3 + x_1 + x_0$
- $C_4 = P4 + x_2 + x_1 + x_0$

Ex)

1	2	3	4	5	6	7
p_1	p_2	x_3	p_4	x_2	x_1	x_0
1	1	0	0	1	1	0
1	0	0	0	1	1	0

해밍 코드

- $C_1 = P1 + x_3 + x_2 + x_0$
- $C_2 = P2 + x_3 + x_1 + x_0$
- $C_4 = P4 + x_2 + x_1 + x_0$

C1: 0

C2: 1

C4: 0

010 → 위치 2번에 문제 발생

Ex)

1	2	3	4	5	6	7
p ₁	p ₂	x ₃	p ₄	x ₂	x ₁	x ₀
1	1	0	0	1	1	0
1	0	0	0	1	1	0

해밍 코드

- $C_1 = P1 + x_3 + x_2 + x_0$
- $C_2 = P2 + x_3 + x_1 + x_0$
- $C_4 = P4 + x_2 + x_1 + x_0$

Ex)

1	2	3	4	5	6	7
p_1	p_2	x_3	p_4	x_2	x_1	x_0
1	1	0	0	1	1	0
1	1	0	0	0	1	0

해밍 코드

- $C_1 = P_1 + x_3 + x_2 + x_0$
- $C_2 = P_2 + x_3 + x_1 + x_0$
- $C_4 = P_4 + x_2 + x_1 + x_0$

C1: 1

C2: 0

C4: 1 101 → 위치 5번에 문제 발생

Ex)

1	2	3	4	5	6	7
p_1	p_2	x_3	p_4	x_2	x_1	x_0
1	1	0	0	1	1	0
1	1	0	0	0	1	0

선형 부호(Linear Code)

- $[n, k]$ code를 갖고 진행
n: 코드의 길이(words)
k: 차원

- $G = \text{Generating Matrix} = \underbrace{[I_k \mid P]}_n \Big]_k$

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- $H = \text{Parity Check Matrix} = \underbrace{[-P^T \mid I_{n-k}]}_n \Big]_{n-k}$

선형 부호(Linear Code)

- $[n, k]$ code

- $G = \begin{bmatrix} 1001 \\ 0110 \end{bmatrix} \quad [I_k \mid P] \rightarrow \begin{bmatrix} 1001 \\ 0110 \end{bmatrix}$

- $H = [-P^T \mid I_{n-k}] \rightarrow \begin{bmatrix} 0110 \\ 1001 \end{bmatrix} \rightarrow \begin{bmatrix} -0 & -1 & 1 & 0 \\ -1 & -0 & 0 & 1 \end{bmatrix}$



$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \quad A^T = \begin{bmatrix} a & c \\ b & d \end{bmatrix}$$

$$\begin{bmatrix} 0110 \\ 1001 \end{bmatrix}$$

고파 부호

- 고파 부호 링크 참조

<https://www.youtube.com/watch?v=u4y3YehFivA&list=PLdOq9g7U6Pdt5ZIWeffEU-ViDUS6j-jFS&index=21&t=487s>

Q & A

