

수치해석 (1 학기)

김상배 교수

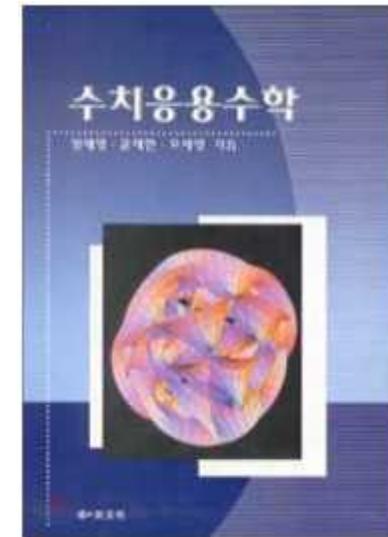
xxx@hnu.kr

<http://sbk.hnu.kr/Lectures>

010-XXXX-7867

문자메세지

카톡(공부내용 질문)



교과서: 수치응용수학, 정세영외2인, 경문사

수치해석 : 해석학적으로 얻기 어려운 문제의 답을 수치적으로 근사값을 구하려는 연구.

예)

Input interpretation:

series	e^{-x}	point	$x = 0$
--------	----------	-------	---------

Series expansion at $x=0$:

$$1 + x + \frac{x^2}{2} + \frac{x^3}{6} + \frac{x^4}{24} + \frac{x^5}{120} + O(x^6)$$

(converges everywhere)

Taylor 급수

함수 $f(x)$ 가 $x = a$ 에서 무한번 미분가능일 때 무한급수

$$f(a) + f'(a)(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \frac{f^{(3)}(a)}{3!}(x-a)^3 + \dots$$

을 $x = a$ 에서 f 의 Taylor급수라 한다.

특히 $a = 0$ 일 때 f 의 Taylor급수를 특히 Maclaurin급수라 한다.

Taylor의 정리

함수 $f(x)$ 가 $x = a$ 을 포함하는 구간에서 $(n + 1)$ 번 미분가능하면,

$$f(x) = f(a) + f'(a)(x-a) + \frac{f''(a)}{2!}(x-a)^2 \dots + \frac{f^{(n)}(a)}{n!}(x-a)^n + R_n(x)$$

이고 $R_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!}(x-a)^{n+1}$ 가 되는 점 ξ 가 a 와 x 사이에 존재한다.

함수들의 멱급수 전개

Maclaurin 급수	수렴구간
$e^x = \sum_{n=0}^{\infty} \frac{x^n}{n!} = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots$	$-\infty < x < +\infty$
$\sin x = \sum_{n=0}^{\infty} (-1)^n \frac{x^{2n+1}}{(2n+1)!} = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \dots$	$-\infty < x < +\infty$
$\cos x = \sum_{n=0}^{\infty} (-1)^n \frac{x^{2n}}{(2n)!} = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + \dots$	$-\infty < x < +\infty$
$\ln(1+x) = \sum_{n=0}^{\infty} (-1)^n \frac{x^{n+1}}{n+1} = x - \frac{x^2}{2} + \frac{x^3}{3} - \frac{x^4}{4} + \dots$	$-1 < x \leq 1$
$\tan^{-1} x = \sum_{n=0}^{\infty} (-1)^n \frac{x^{2n+1}}{2n+1} = x - \frac{x^3}{3} + \frac{x^5}{5} - \dots$	$-1 < x \leq 1$
$\frac{1}{1-x} = \sum_{n=0}^{\infty} x^n = 1 + x + x^2 + x^3 + x^4 + \dots$	$-1 < x < 1$
$\sinh x = \sum_{n=0}^{\infty} \frac{x^{2n+1}}{(2n+1)!} = x + \frac{x^3}{3!} + \frac{x^5}{5!} + \frac{x^7}{7!} + \dots$	$-\infty < x < +\infty$
$\cosh x = \sum_{n=0}^{\infty} \frac{x^{2n}}{(2n)!} = 1 + \frac{x^2}{2!} + \frac{x^4}{4!} + \frac{x^6}{6!} + \dots$	$-\infty < x < +\infty$

[예제] e 의 값을 소수점이하 넷째자리까지 정확히 구하여라.

(풀이)

모든 x 에 대하여

$$e^x = 1 + x + \frac{1}{2!}x^2 + \frac{1}{3!}x^3 + \dots + \frac{1}{n!}x^n + \frac{e^c}{(n+1)!}x^{n+1}, \quad (0 < c < x)$$

이므로 $x = 1$ 이면 $e = 1 + 1 + \frac{1}{2!} + \frac{1}{3!} + \dots + \frac{1}{n!} + \frac{e^c}{(n+1)!}$ 이다.

$c < 1$ 이므로 $|R_n| = \left| \frac{e^c}{(n+1)!} \right| = \frac{e^c}{(n+1)!} < \frac{e^1}{(n+1)!}$ 이다.

여기서 $e < 3$ 이므로 $|R_n| < \frac{3}{(n+1)!}$ 이다.

소수점이하 4째자리까지 정확하려면

$$|R_n| < \frac{3}{(n+1)!} < 0.5 \times 10^{-4} = 0.00005 \text{ 이어야 하고}$$

$n = 8$ 이면 위 식을 만족하므로

$$e \approx 1 + 1 + \frac{1}{2!} + \frac{1}{3!} + \dots + \frac{1}{8!} \approx 2.7183 \text{ 이다.}$$



[예제] $\sin 3^\circ$ 의 값을 소수점이하 다섯째 자리까지 정확히 구하여라.

(풀이)

$f(x) = \sin x$ 로 놓으면 $f(x) = x - \frac{1}{3!} x^3 + \frac{1}{5!} x^5 - \frac{1}{7!} x^7 + \dots$ 이다.

$$f(3^\circ) = f\left(\frac{\pi}{60}\right) = \frac{\pi}{60} - \frac{1}{3!} \left(\frac{\pi}{60}\right)^3 + \frac{1}{5!} \left(\frac{\pi}{60}\right)^5 - \frac{1}{7!} \left(\frac{\pi}{60}\right)^7 + \dots + R_n\left(\frac{\pi}{60}\right)$$

그런데 $|f^{(n+1)}(c)| \leq 1$ 이므로

$$\left| R_n\left(\frac{\pi}{60}\right) \right| = \frac{|f^{(n+1)}(c)|}{(n+1)!} \left(\frac{\pi}{60}\right)^{n+1} \leq \frac{1}{(n+1)!} \left(\frac{\pi}{60}\right)^{n+1}$$

소수점이하 다섯째 자리까지 정확하려면

$$\frac{1}{(n+1)!} \left(\frac{\pi}{60}\right)^{n+1} < 0.5 \times 10^{-5} \text{ 이어야 하고}$$

$n = 3$ 이면 위 식을 만족하므로

$$\sin 3^\circ = \sin \frac{\pi}{60} \approx \frac{\pi}{60} - \frac{1}{3!} \left(\frac{\pi}{60}\right)^3 \approx 0.05234 \text{ 이다.}$$

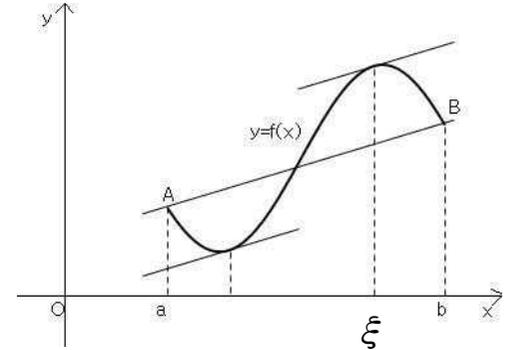
■
한남대학교 수학과 김상배교수

평균값정리

함수 f 가 구간 $[a, b]$ 에서 연속이고, 도함수 $f'(x)$ 가 개구간 (a, b) 에서 존재하면,

$$\frac{f(b) - f(a)}{b - a} = f'(\xi)$$

을 만족하는 $\xi \in (a, b)$ 가 존재한다.



$$f(b) = f(a) + f'(\xi)(b - a)$$

$$f(b) = f(a) + f'(a)(b - a) + \frac{1}{2!} f''(\xi)(b - a)^2$$

$$f(b) = f(a) + f'(a)(b - a) + \frac{1}{2!} f''(a)(b - a)^2 + \frac{1}{3!} f^{(3)}(\xi)(b - a)^3$$

$$f(b) = f(a) + f'(\xi)(b-a)$$

$$f(b) = f(a) + f'(a)(b-a) + \frac{1}{2!} f''(\xi)(b-a)^2$$

$$f(b) = f(a) + f'(a)(b-a) + \frac{1}{2!} f''(a)(b-a)^2 + \frac{1}{3!} f^{(3)}(\xi)(b-a)^3$$

테일러의 정리

함수 $f(x)$ 가 $x = a$ 을 포함하는 구간에서 $(n+1)$ 번 미분가능하면,

$$f(x) = f(a) + f'(a)(x-a) + \frac{f''(a)}{2!}(x-a)^2 \cdots + \frac{f^{(n)}(a)}{n!}(x-a)^n + R_n(x)$$

이고 $R_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!}(x-a)^{n+1}$ 가 되는 점 ξ 가 a 와 x 사이에 존재한다.

적분에 대한 평균값의 정리

(미분에 대한) 평균값 정리와 (어떤 의미에서) 같다.

$$\frac{f(b) - f(a)}{b - a} = f'(\xi)$$

로부터

$$f(b) - f(a) = f'(\xi)(b - a)$$

$g(x) = f'(x)$ 라고 하면

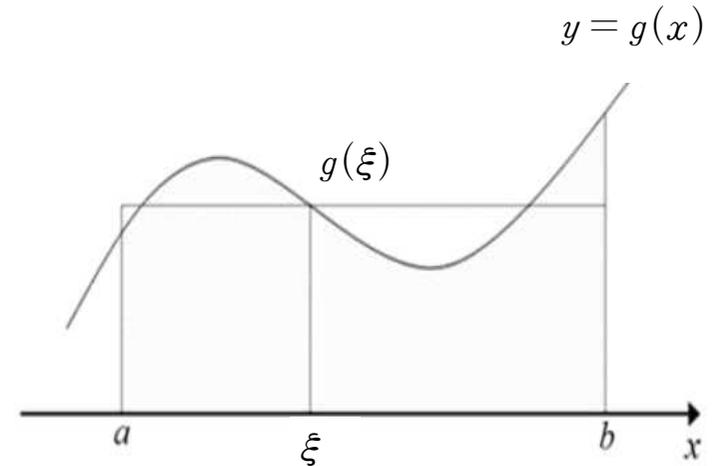
$$\text{왼쪽식 : } f(b) - f(a) = \int_a^b f'(x)dx = \int_a^b g(x)dx$$

$$\text{오른쪽식 : } f'(\xi)(b - a) = g(\xi)(b - a)$$

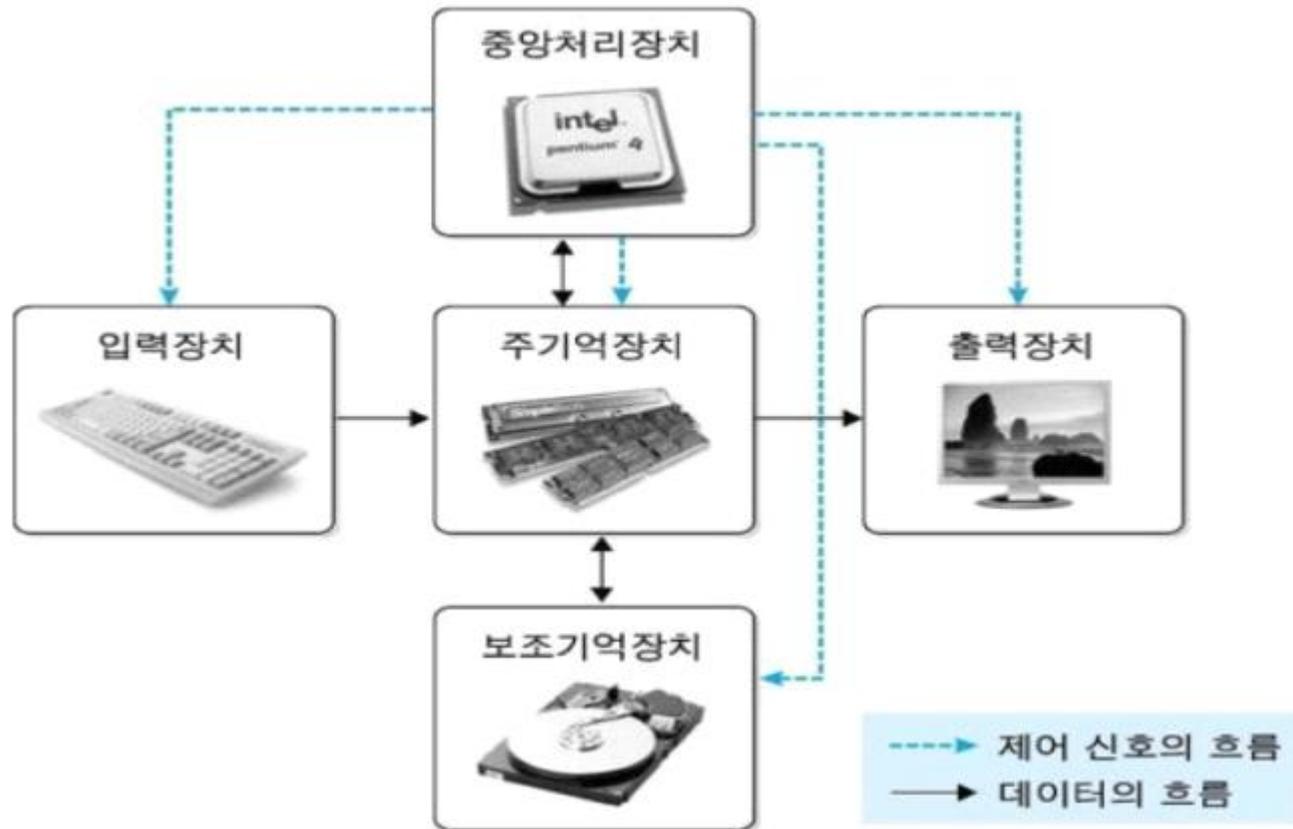
그러므로

$$\int_a^b g(x)dx = g(\xi)(b - a) \text{ 여기서 } \xi \in (a, b)$$

이것이 적분에 대한 평균값 정리이다.



컴퓨터의 구성 : 입력장치 + 중앙처리장치 + 출력장치
(제어장치+연산장치) + 기억장치(주기억+보조기억)



기억장치에 데이터를 저장하는 방법

데이터가 0 과 1 로만 구성됨. 0 과 1 로 숫자, 문자, 그림, 음악 등을 표시한다.

1 bit = 0 또는 1

1 byte = 8 bit = $2^8 = 256$ 개의 문자를 나타낼 수 있다. ASCII코드

Dec	Hx	Oct	Char	Dec	Hx	Oct	Char	Dec	Hx	Oct	Char	Dec	Hx	Oct	Char
0	0	000	NUL (null)	32	20	040	Space	64	40	100	@	96	60	140	`
1	1	001	SOH (start of heading)	33	21	041	!	65	41	101	A	97	61	141	a
2	2	002	STX (start of text)	34	22	042	"	66	42	102	B	98	62	142	b
3	3	003	ETX (end of text)	35	23	043	#	67	43	103	C	99	63	143	c
4	4	004	EOF (end of transmission)	36	24	044	\$	68	44	104	D	100	64	144	d
5	5	005	ENQ (enquiry)	37	25	045	%	69	45	105	E	101	65	145	e
6	6	006	ACK (acknowledge)	38	26	046	&	70	46	106	F	102	66	146	f
7	7	007	BEL (bell)	39	27	047	'	71	47	107	G	103	67	147	g
8	8	010	BS (backspace)	40	28	050	(72	48	110	H	104	68	150	h
9	9	011	TAB (horizontal tab)	41	29	051)	73	49	111	I	105	69	151	i
10	A	012	LF (NL line feed, new line)	42	2A	052	*	74	4A	112	J	106	6A	152	j
11	B	013	VT (vertical tab)	43	2B	053	+	75	4B	113	K	107	6B	153	k
12	C	014	FF (NP form feed, new page)	44	2C	054	,	76	4C	114	L	108	6C	154	l
13	D	015	CR (carriage return)	45	2D	055	-	77	4D	115	M	109	6D	155	m
14	E	016	SO (shift out)	46	2E	056	.	78	4E	116	N	110	6E	156	n
15	F	017	SI (shift in)	47	2F	057	/	79	4F	117	O	111	6F	157	o
16	10	020	DLE (data link escape)	48	30	060	0	80	50	120	P	112	70	160	p
17	11	021	DC1 (device control 1)	49	31	061	1	81	51	121	Q	113	71	161	q
18	12	022	DC2 (device control 2)	50	32	062	2	82	52	122	R	114	72	162	r
19	13	023	DC3 (device control 3)	51	33	063	3	83	53	123	S	115	73	163	s
20	14	024	DC4 (device control 4)	52	34	064	4	84	54	124	T	116	74	164	t
21	15	025	NAK (negative acknowledge)	53	35	065	5	85	55	125	U	117	75	165	u
22	16	026	SYN (synchronous idle)	54	36	066	6	86	56	126	V	118	76	166	v
23	17	027	ETB (end of trans. block)	55	37	067	7	87	57	127	W	119	77	167	w
24	18	030	CAN (cancel)	56	38	070	8	88	58	130	X	120	78	170	x
25	19	031	EM (end of medium)	57	39	071	9	89	59	131	Y	121	79	171	y
26	1A	032	SUB (substitute)	58	3A	072	:	90	5A	132	Z	122	7A	172	z
27	1B	033	ESC (escape)	59	3B	073	;	91	5B	133	[123	7B	173	{
28	1C	034	FS (file separator)	60	3C	074	<	92	5C	134	\	124	7C	174	
29	1D	035	GS (group separator)	61	3D	075	=	93	5D	135]	125	7D	175	}
30	1E	036	RS (record separator)	62	3E	076	>	94	5E	136	^	126	7E	176	~
31	1F	037	US (unit separator)	63	3F	077	?	95	5F	137	_	127	7F	177	DEL

0과 1로 숫자를 나타내는 방법

정수(1배정도) : 4 byte = 32bit = $2 \times 2 \times \dots \times 2 = 2 \times 2^{31} = 2 \times 2,147,483,648$

2,147,483,648(약21억)개의 음수와, 0, 그리고 2,147,483,647(약21억)개의 양수를 사용할 수 있다.

실수(1배정도) : 4 byte = 32bit = 부호1bit + 지수7bit + 가수24bit

아보가드로수 : 산소16g에 들어 있는 산소원자수 = $+6.022 \times 10^{23}$

7bit의 크기 : $2^7 = 128 = (-64) \rightarrow 0 \rightarrow (+63)$

24bit의 크기 : 10진수 7자리

$$2^{24} = (2^{10})^{2.4} = 1024^{2.4} \approx (10^3)^{2.4} = 10^{(3 \times 2.4)} = 10^{7.2}$$

정수(2배정도) : 8 byte=64bit= $2 \times 2^{63} = 2 \times 9.2 \times 10^{18} =$ 약 -100경에서 +100경

실수(2배정도) : 8 byte=64bit=부호1bit + 지수15bit +가수48bit

15bit = $2^{15} = (-16,384) \rightarrow (+16,383)$

48bit = $2^{48} \approx 1.4 \times 10^{14} =$ 10진수 14자리

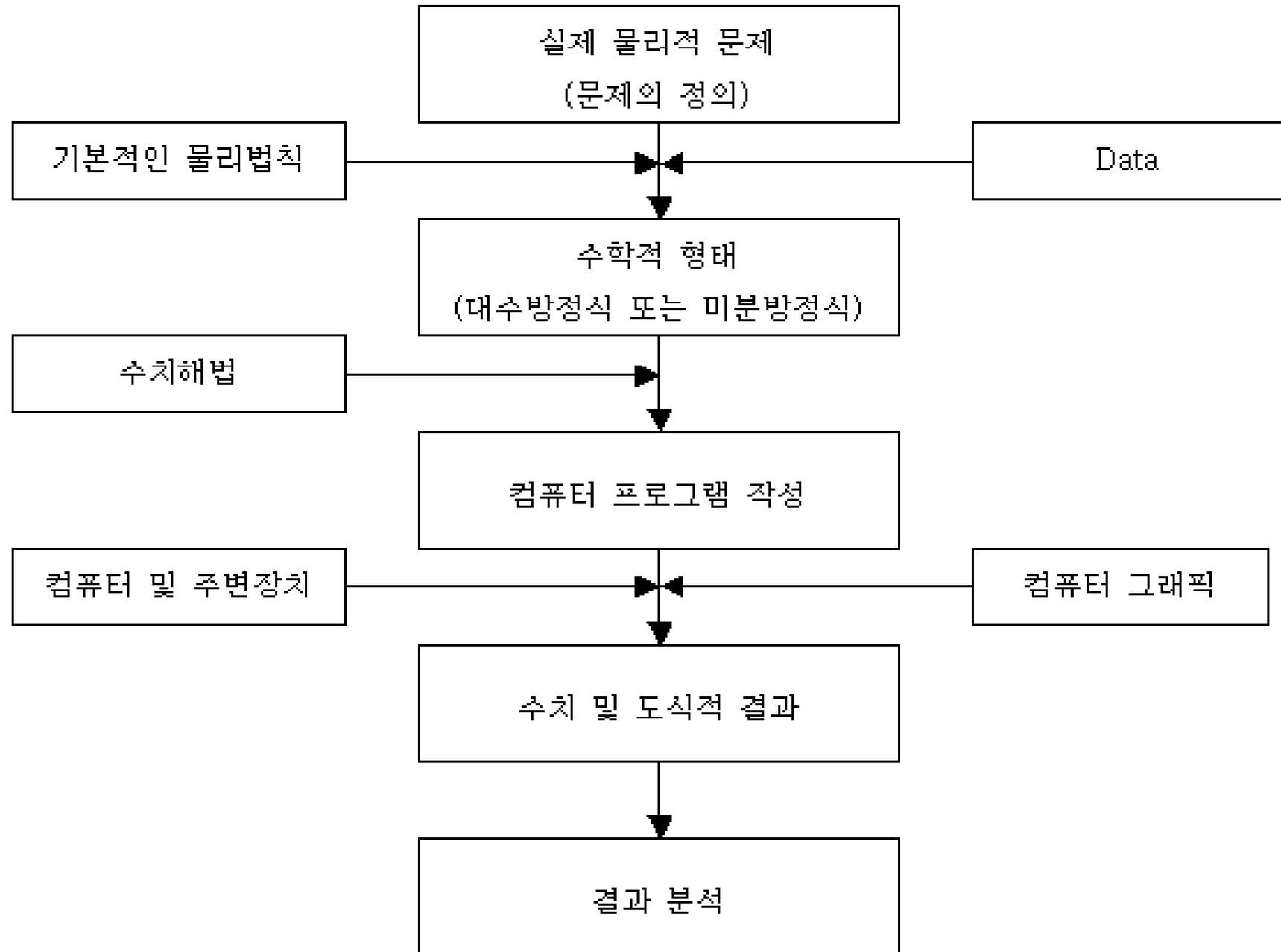
실수의 과학적표기법

$$x = \pm (0.a_1a_2 \cdots a_r)_b \times b^c$$
$$a_1 \neq 0, 0 \leq a_i < b, m \leq c \leq M$$

$(0.a_1a_2 \cdots a_r)_b$: **b진법의 가수(mantissa)** ,
 c :**지수(exponent)**, b :**기저(base)**

[문제] 가수가 3자리이고 $-2 \leq c \leq 2$ (c 는 정수), $b = 5$ 인 경우 컴퓨터가 과학적표기법으로 나타낼 수 있는 실수의 개수는 몇 개인가?

[숙제] 1배정도로 나타낼 수 있는 실수의 개수는 몇 개인가?



오차의 종류

1. 원시오차 : 프로그램으로 수정할 수 없는 근본적인 오차.
 - 1) 초기오차 : 자연현상의 수학적 모델화 과정에서의 오차
무리수를 유한소수로 표현할 때의 오차
 - 2) 전환오차(최초입력) : 10진수를 2진수로 저장할 때 오차

2. 처리오차 : 프로그램으로 수정할 수 있는 오차
 - 1) 절단오차 : 무한합을 유한합으로 계산할 때 오차
 - 2) 저장오차(중간입력) : 기억장치에 저장하기 위하여 반올림 할 때 오차

[예제] 전환오차의 예

$$(0.1)_{10} = \left(\frac{1}{10}\right)_{10} = \left(\frac{1}{1010}\right)_2^* = (0.00011001100 \dots)_2$$

따라서 1/10 을 유한한 가수를 사용하는 컴퓨터에서 전환오차가 생긴다.

[숙제] 등식 * 을 증명해 보아라.

절대오차

참값 x 에 대하여 근사값을 \tilde{x} 라고 할 때, $|x - \tilde{x}|$ 를 **절대오차**라고 한다.

상대오차

참값 x 에 대하여 근사값을 \tilde{x} 라고 할 때, $\frac{|x - \tilde{x}|}{|x|}$ 를 **상대오차**라고 한다.

유효숫자 n자리

참값 x 와 근사값 \tilde{x} 에 대하여, 만약 상대오차가

$$\frac{|x - \tilde{x}|}{|x|} \leq 0.5 \times 10^{-n+1}$$

이면, 근사값 \tilde{x} 는 “**유효숫자 n자리가 일치한다**” 또는
“**n자리의 유효숫자를 갖는다**” 라고 한다.

주의 : n자리수가 정확하게 일치하는 것이 아니고, 비슷하게 일치함.

[예제] (1) $x = 1000, \tilde{x} = 1000.4$ 이면 $\frac{|x - \tilde{x}|}{|x|} \leq 0.5 \times 10^{-4+1}$ 이므로

근사값 \tilde{x} 는 $n=4$ 자리 유효숫자를 갖는다. 즉 근사값의 4자리가 참값과 비슷하게 일치한다고 볼 수있다.

(2) $x = 1000, \tilde{x} = 999.6$ 이면 $\frac{|x - \tilde{x}|}{|x|} \leq 0.5 \times 10^{-4+1}$ 이므로

근사값 \tilde{x} 는 $n=4$ 자리 유효숫자를 갖는다. 즉 근사값의 4자리가 참값과 비슷하게 일치한다고 볼 수있다.

(3) $x = 4000, \tilde{x} = 4001.6$ 이면 $\frac{|x - \tilde{x}|}{|x|} \leq 0.5 \times 10^{-4+1}$ 이므로

근사값 \tilde{x} 는 $n=4$ 자리 유효숫자를 갖는다. 즉 근사값의 4자리가 참값과 비슷하게 일치한다고 볼 수있다.

(4) $x = 4000, \tilde{x} = 3998.4$ 이면 $\frac{|x - \tilde{x}|}{|x|} \leq 0.5 \times 10^{-4+1}$ 이므로

근사값 \tilde{x} 는 $n=4$ 자리 유효숫자를 갖는다. 즉 근사값의 4자리가 참값과 비슷하게 일치한다고 볼 수있다.

덧셈과 뺄셈

유효숫자가 작은 숫자에 맞추어 계산한다. 예를 들어, 유효숫자 세 개인 수 3.14와 유효숫자 5개인 8.9714를 더하면 산술적으로는 12.1114가 나오지만, 유효숫자가 소수점 이하 두 개 이므로 결과는 12.11이 된다.

곱셈과 나눗셈

두 측정치 중 유효숫자가 적은 쪽과 같은 유효숫자를 가진다. 예를 들어, 2.56×12.8690 의 산술적 계산결과는 32.94464이지만, 2.56의 유효숫자가 3개이므로 유효한 결과는 32.9이다.

유효숫자의 자리수 상실

상대적으로 큰수를 곱하거나 작은수로 나누면 overflow 된다.

비슷한 크기의 숫자를 빼면 유효숫자 자리수가 상실된다.

$0.24567 - 0.24512 = 0.00055$ (유효숫자 5자리 -> 2자리)

(해결법) 수학적으로 동치인 다른 식을 변형한다.

[예제]
$$\begin{aligned} \sqrt{x^2 + 1} - 1 &= \frac{(\sqrt{x^2 + 1} - 1)(\sqrt{x^2 + 1} + 1)}{(\sqrt{x^2 + 1} + 1)} \\ &= \frac{(x^2 + 1) - 1^2}{(\sqrt{x^2 + 1} + 1)} \\ &= \frac{x^2}{(\sqrt{x^2 + 1} + 1)} \end{aligned}$$

수열의 비교 ($\{\alpha_n\}, \{\beta_n\}$)

$\alpha_n = O(\beta_n) \iff (\exists n_0 : n > n_0 \implies |\alpha_n| \leq c|\beta_n|)$: 비슷한 속도로 0에 수렴

$\alpha_n = o(\beta_n) \iff \alpha_n/\beta_n \rightarrow 0$: α_n 이 β_n 보다 빠르게 0에 수렴한다.

[예제]

$$\frac{n+1}{n^2} = O\left(\frac{1}{n}\right), \quad \frac{1}{n \ln n} = o\left(\frac{1}{n}\right)$$

수열의 수렴 ($x_n \rightarrow x$)

1차적 수렴 : $|x_{n+1} - x| \leq c|x_n - x|$

2차적 수렴 : $|x_{n+1} - x| \leq c|x_n - x|^2$

p 차적 수렴 : $|x_{n+1} - x| \leq c|x_n - x|^p$

[예제] $x_n = \frac{1}{2^n}$ 은 0으로 1차적으로 수렴한다.

$$\left| \frac{1}{2^{n+1}} - 0 \right| \leq \frac{1}{2} \left| \frac{1}{2^n} - 0 \right|$$

[숙제] $x_n = 5^{-2^n}$ 은 0으로 몇 차적으로 수렴하는지 보여라.

제2장 비선형방정식의 해

선형방정식

$$ax + b = 0$$

$$a_1x_1 + a_2x_2 + \cdots + a_nx_n = a$$

$$b_1x_1 + b_2x_2 + \cdots + b_nx_n = b$$

해가 기하학적으로 선형(직선, 평면, 초평면)으로 나타난다.

해가 없거나, **한 개 이거나**, 무한 개로 나타난다.

비선형방정식

$$ax^2 + b = 0 \quad \sqrt{x} + b = 0 \quad \sin x + x^2 = 0 \quad f(x) = 0$$

해가 없거나, **유한 개이거나**, 무한 개로 나타난다.

5차이상 다항식은 근의 공식 (근과 계수의 관계식)이 없다.

비선형방정식은 일반적으로는 유한 회수로 구할 수 없고,

반복법을 사용하여 근사값의 수열이 참값으로 수렴하도록 한다.

이분법 (p21)

함수 f 가 구간 $[a, b]$ 에서 연속이고 $f(a), f(b)$ 가 부호가 다를 경우
방정식 $f(x) = 0$ 의 해를 찾는 방법

```
입력 :  $a, b, xtol, ftol, N$   
For  $i = 1$  to  $N$   
     $e = \frac{b - a}{2}$   
     $m = \frac{b + a}{2}$   
    if  $e \leq xtol$  or  $|f(m)| \leq ftol$  then stop  
    if  $f(a)f(m) < 0$  then  
         $b = m$   
    else  
         $a = m$   
    endif
```

[예제2.2] 방정식 $e^x - 1.4 - \tan^{-1}x = 0$ 의 근이 구간 $[-7, -5]$ 에 적어도 하나가 존재함을 보이고 이분법을 사용하여 근사값을 구하라.

(풀이)

[중간값의 정리] 함수 $f(x)$ 가 구간 $[a, b]$ 에서 연속이고
 $m = \min\{f(x) \mid a \leq x \leq b\}$, $M = \max\{f(x) \mid a \leq x \leq b\}$
이면 $\forall y \in [m, M], \exists x \in [a, b] : f(x) = y$ 이 성립한다.

함수 $f(x) = e^x - 1.4 - \tan^{-1}x$ 는 $[-7, -5]$ 에서 연속이다.

$f(-7) \approx 0.298$, $f(-5) = -0.0198$ 이므로 $f(-5) \leq 0 \leq f(-7)$

이다. 중간값의 정리에 의하여 $\exists x \in [-7, -5] : f(x) = 0$ 이다.

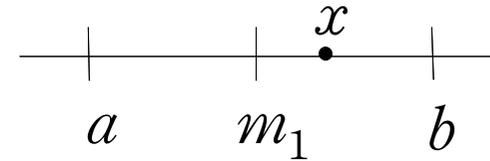
이분법에서 조건을 $N = 50$, $xtol = 1.0E-7$, $ftol = 1.0E-7$ 로 놓고 실행하면 $i = 18$ 근방에서 근사값 $m = -5.682$ 을 구할 수 있다. ■

[숙제] 이분법을 사용하여 방정식 $e^x - \sin x = 0$, $x \in [-4, -2]$ 를 해를 구하라.

[예제2.3] 방정식 $x^3 - x - 2 = 0$ 의 근은 구간 $[1, 2]$ 에 존재함을 보이고,
 절대오차가 10^{-3} 보다 작아지도록 $xtol$ 과 N 을 구하여라.

(풀이)

$f(x) = x^3 - x - 2$ 는 구간 $[1, 2]$ 에서 연속이고,



$$f(1) = -2 < 0 < 4 = f(2)$$

이므로 중간값의 정리에 의하여 구간 $[1, 2]$ 에서 방정식 $f(x) = 0$ 의 근이 존재한다. $|x - m_1| < \frac{b-a}{2}$ 이고

$$|x - m_2| < \frac{b-a}{2^2}, \dots, |x - m_i| < \frac{b-a}{2^i} \text{ 이므로}$$

$$|x - m_N| < \frac{b-a}{2^N} \leq xtol = 10^{-3} \text{ 이 되게 하려면 } \frac{2-1}{2^N} \leq 10^{-3} \text{ 이}$$

되고 양변에 \log 를 취하면 $\log_{10}(2^{-N}) \leq \log_{10}(10^{-3})$ 이 되고

$$N \geq \frac{3}{\log_{10}2} \approx 9.96 \text{ 그러므로 } N = 10 \text{ 이상이면 충분하다. } \blacksquare$$

[예제2.4] 방정식 $e^x - 1.4 - \tan^{-1}x = 0$ 의 근을 구간 $[-7, -5]$ 에서 이분법으로 구할 때 상대오차가 10^{-5} 보다 작게 되려면 최소한 몇 번을 반복해야 하는가?

(풀이) 상대 오차 = $\frac{|x - m_i|}{|x|} \leq \frac{b-a}{|x| 2^i} < 10^{-5}$ 이어야 한다.

$$b - a = (-5) - (-7) = 2 \text{ 이고 } x \in [-7, -5] \text{ 이므로 } \frac{1}{|x|} \leq \frac{1}{5}.$$

따라서 $\frac{2}{5 \times 2^i} < 10^{-5}$ 이 되고, $2^{-i+2} < 10^{-4}$ 이 된다. 양변에 로그를

취하여 풀면 $i > 2 + \frac{4}{\log 2} \approx 15.28$ 이므로 최소 16 번 반복해야 한다. ■

[숙제] 이분법을 이용하여 절대오차가 10^{-4} 보다 작은 $\sqrt{2}$ 의 근사값을 구하여라.

가위치법 (p26)

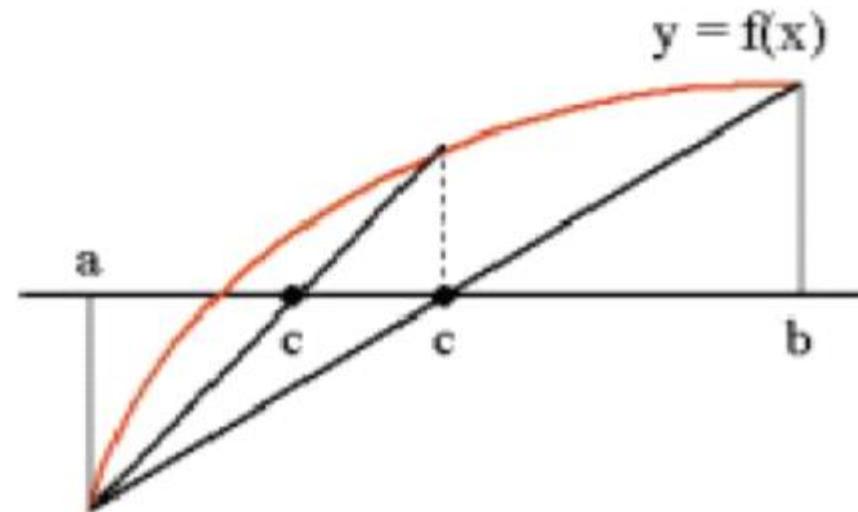
함수 f 가 구간 $[a, b]$ 에서 연속이고, $f(a)$ 와 $f(b)$ 의 부호가 다를 경우, $f(x) = 0$ 의 해를 구하는 방법으로,

두 점 $(a, f(a))$ 와 $(b, f(b))$ 을 잇는 직선의 방정식은

$$y = \frac{f(b) - f(a)}{b - a}(x - a) + f(a)$$

$$0 = \frac{f(b) - f(a)}{b - a}(c - a) + f(a)$$

$$c = a - f(a) \frac{b - a}{f(b) - f(a)}$$



가위치법 알고리즘

입력 : $a, b, xtol, ftol, N$

For $n = 1$ **to** N

$$c = a - f(a) \frac{b - a}{f(b) - f(a)}$$

$$e = |c - c_0|$$

if $e \leq xtol$ **or** $|f(c)| \leq ftol$ **then stop**

if $f(a)f(c) < 0$ **then**

$$b = c$$

else

$$a = c$$

endif

$$c_0 = c$$

[예제2.5] 방정식 $e^x - 1.4 - \tan^{-1}x = 0$ 의 근이 구간 $[-7, -5]$ 에 존재한다.

가위치법을 사용하여 근사값을 구하라.

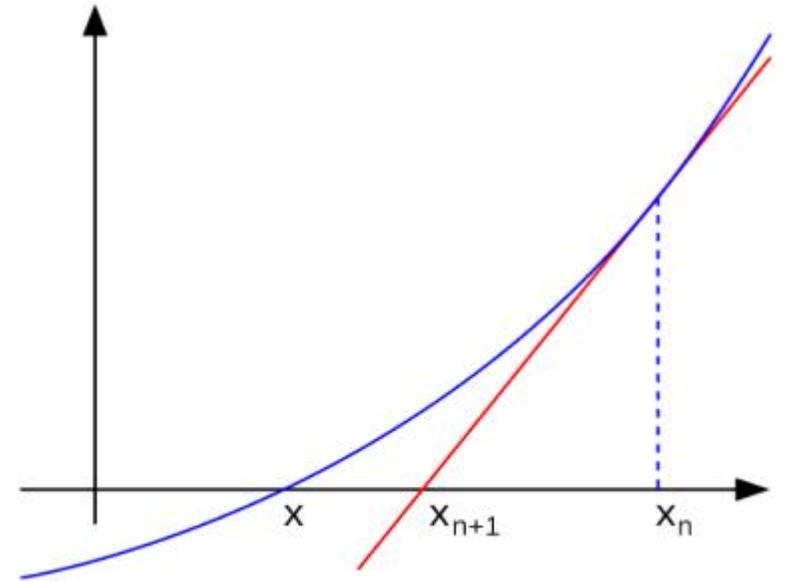
(풀이)

$N = 50$, $xtol = 10^{-7}$, $ftol = 10^{-7}$ 을 사용하여 가위치법의 알고리즘을 사용하여 구하면 대강 8번 근방에서 근사값 $c = -5.6823$ 을 구할 수 있다. 이분법보다 수렴속도가 빠른 것을 알 수 있다. ■

[숙제] 방정식 $e^x - \sin x = 0$, $x \in [-4, -2]$ 의 해를 가위치법으로 구하는 컴퓨터 프로그램을 하여 제출하라.

Newton 방법

뉴턴법은 방정식 $f(x) = 0$ 에서의 함수 f 가 미분가능할 때, 사용할 수 있는 수렴이 빠른 방정식이 수치해법이다. 초기 추측 x_0 에서 시작하여 n 단계의 해의 근사값 x_n 이 주어졌을 때, $(n + 1)$ 단계의 근사값 x_{n+1} 은 x_n 에서의 접선이 x 축과 만나는 지점이다.



x_n 에서의 접선의 방정식

$$y = f'(x_n)(x - x_n) + f(x_n)$$

이 점 $(x_{n+1}, 0)$ 을 만족하므로

$$0 = f'(x_n)(x_{n+1} - x_n) + f(x_n)$$

이 성립하고 x_{n+1} 을 구하면,

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

뉴턴법 알고리즘

입력 : $x_0, xtol, ftol, N$

For $n = 0$ **to** N

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

if $|x_{n+1} - x_n| \leq xtol$ **or** $|f(x_{n+1})| \leq ftol$ **then stop**

[참고] 수렴차수 구하기

$$\begin{cases} e_2 = C e_1^k \\ e_3 = C e_2^k \end{cases}$$
$$\Rightarrow \frac{e_2}{e_3} = \left(\frac{e_1}{e_2} \right)^k \Rightarrow k = \frac{\log(e_2/e_3)}{\log(e_1/e_2)}$$

[예제2.6] 뉴턴법으로 $\sqrt{2}$ 의 근사값을 구하여라.

(풀이) $f(x) = x^2 - 2$ 일 때 $\sqrt{2}$ 는 방정식 $f(x) = 0$ 의 근이다.

$f'(x) = 2x$ 이므로 뉴턴의 반복식은

$$\begin{aligned}x_{n+1} &= x_n - \frac{x_n^2 - 2}{2x_n} \\ &= \frac{x_n}{2} + \frac{1}{x_n}\end{aligned}$$

[예제2.7] 방정식 $e^x - 1.4 - \tan^{-1}x = 0$ 의 근을 뉴턴법으로 구하여라.

초기추측 $x_0 = -7$ 를 사용하여라.

(풀이) 수렴차수가 2가 나온다.

[정리2.1] f'' 이 구간 $[a,b]$ 에서 연속이고 $f(x) = 0$ 의 근 s 가 단일근일 때,
즉 $f(s) = 0 \neq f'(s)$ 이라면, 근의 적당히 작은 근방에서 뉴턴법의
수렴차수는 2이다.

(증명) 테일러 정리로부터 s 와 x_n 사이에 있는 ζ_n 에 대하여,

$$f(s) = f(x_n) + f'(x_n)(s - x_n) + \frac{f''(\zeta_n)}{2!}(s - x_n)^2$$

이 성립하므로, $f(s) = 0$ 를 적용하면,

$$0 = f(x_n) + f'(x_n)(s - x_n) + \frac{f''(\zeta_n)}{2!}(s - x_n)^2$$

$$0 = \frac{f(x_n)}{f'(x_n)} + (s - x_n) + \frac{f''(\zeta_n)}{2 f'(x_n)}(s - x_n)^2$$

$$-\frac{f(x_n)}{f'(x_n)} = (s - x_n) + \frac{f''(\zeta_n)}{2 f'(x_n)}(s - x_n)^2$$

뉴턴법 $x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$ 을 대입하면,

$$x_{n+1} - x_n = s - x_n + \frac{f''(\zeta_n)}{2f'(x_n)}(s - x_n)^2$$

$$s - x_{n+1} = -\frac{f''(\zeta_n)}{2f'(x_n)}(s - x_n)^2$$

$$|s - x_{n+1}| \approx C |s - x_n|^2, \text{ with } C = \left| \frac{f''(s)}{2f'(s)} \right|$$

그러므로 단일근 근처에서 뉴턴법의 수렴차수는 2이다. ■

[정리] $f^{(3)}$ 이 구간 $[a, b]$ 에서 연속이고 $f(x) = 0$ 의 근 s 가 중근일 때,
즉 $f(s) = f'(s) = 0 \neq f''(s)$ 이라면, 뉴턴법의 수렴차수는 1이다.

(증명) 테일러정리로 부터 s 와 x_n 사이에 있는 어떤 ζ_n 에 대하여 다음이 성립한다.

$$f(s) = f(x_n) + f'(x_n)(s - x_n) + \frac{f''(\zeta_n)}{2!}(s - x_n)^2$$

[정리2.1]처럼 뉴턴법을 이용하여

$$s - x_{n+1} = -\frac{f''(\zeta_n)}{2f'(x_n)}(s - x_n)^2$$

$f'(s) = 0$ 을 이용하여

$$s - x_{n+1} = \frac{f''(\zeta_n)}{2 \frac{f'(s) - f'(x_n)}{(s - x_n)}}(s - x_n)$$

평균값의 정리를 이용하여, s 와 x_n 사이에 있는 어떤 ξ_n 에 대하여,

$$s - x_{n+1} = \frac{f''(\zeta_n)}{2f''(\xi_n)}(s - x_n)$$

$$|s - x_{n+1}| \approx C |s - x_n|, \text{ with } C = \left| \frac{f''(s)}{2f''(s)} \right|$$

그러므로 중근 근처에서 뉴턴법의 수렴차수는 1이다. ■

[예제2.8] $f(x) = (x + 1)(x - 5)^2$ 의 2개의 근을 뉴턴법으로 구하고,

각 근으로 수렴하는 근사 수열의 수렴차수를 비교하여라.

(풀이) 초기값으로 단일근 근처에서는 $x_0 = 0$ 으로

중근 근처에서는 $x_0 = 7$ 으로 시작하여 보자. (숙제)

할선법

초기추측 x_0, x_1 에 대하여,

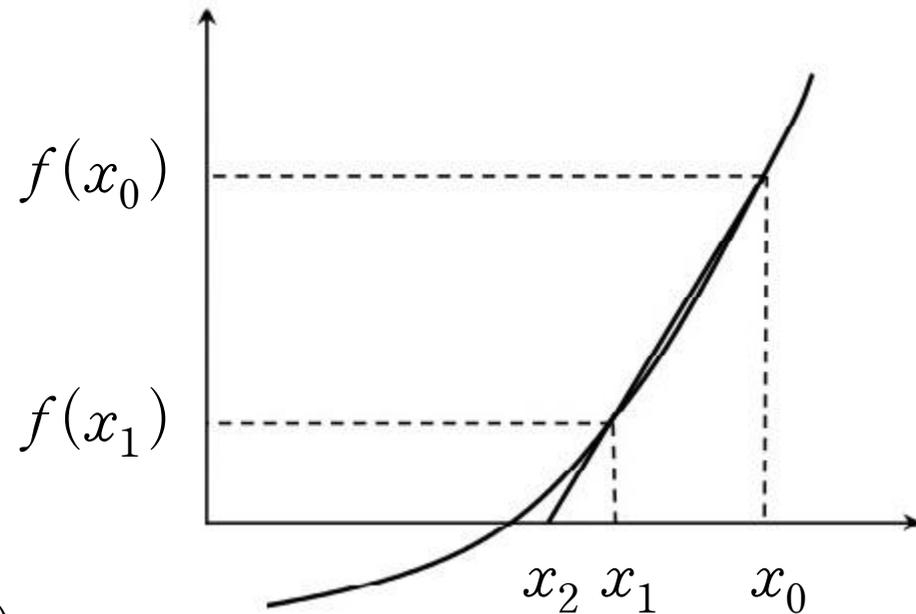
두 점 $(x_0, f(x_0)), (x_1, f(x_1))$ 을 지나는

직선의 방정식

$$y = \frac{f(x_1) - f(x_0)}{x_1 - x_0} (x - x_1) + f(x_1)$$

이 점 $(x_2, 0)$ 을 만족하므로

$$0 = \frac{f(x_1) - f(x_0)}{x_1 - x_0} (x_2 - x_1) + f(x_1)$$



이 성립하고, x_2 을 구하면,

$$x_2 = x_1 - f(x_1) \frac{x_1 - x_0}{f(x_1) - f(x_0)}$$

이것을 일반화 하면

$$x_{n+1} = x_n - f(x_n) \frac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1})}$$



참고로 할선법은 아래와 같이 뉴턴법의 근사식임을 알 수 있다.

$$x_{n+1} = x_n - \frac{f(x_n)}{\frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}}} \approx x_n - \frac{f(x_n)}{f'(x_n)}$$

할선법 알고리즘

입력 : $x_0, x_1, xtol, ftol, N$

For $n = 1$ to N

$$x_{n+1} = x_n - f(x_n) \frac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1})}$$

if $|x_{n+1} - x_n| \leq xtol$ or $|f(x_{n+1})| \leq ftol$ then stop

[정리2.2] f'' 이 구간 $[a,b]$ 에서 연속이고 $f(x) = 0$ 의 근 s 가 단일근일 때, 즉 $f(s) = 0 \neq f'(s)$ 이라면, 근의 적당히 작은 근방에서 할선법의 수렴차수는 1.62 정도 된다.

(증명) $f_n = f(x_n)$ 로 표기하면 할선법은

$$x_{n+1} = x_n - f_n \frac{x_n - x_{n-1}}{f_n - f_{n-1}}$$

양변과 우변 분자에 근 s 를 빼거나 더하면,

$$(x_{n+1} - s) = (x_n - s) - f_n \frac{(x_n - s) - (x_{n-1} - s)}{f_n - f_{n-1}}$$

오차 $e_n = x_n - s$ 을 대입하면,

$$e_{n+1} = e_n - f_n \frac{e_n - e_{n-1}}{f_n - f_{n-1}}$$

$$\begin{aligned}
 e_{n+1} &= e_n - f_n \frac{e_n - e_{n-1}}{f_n - f_{n-1}} \\
 &= \frac{e_n f_n - e_n f_{n-1} - f_n e_n + f_n e_{n-1}}{f_n - f_{n-1}} \\
 &= \frac{f_n e_{n-1} - e_n f_{n-1}}{f_n - f_{n-1}}
 \end{aligned}$$

여기서 테일러정리와 $f(s) = 0$ 를 이용하면,

$$f_n = f(s) + e_n f'(s) + \frac{e_n^2}{2!} f''(\zeta_n) = e_n f'(s) + \frac{e_n^2}{2!} f''(\zeta_n)$$

$$f_{n-1} = f(s) + e_{n-1} f'(s) + \frac{e_{n-1}^2}{2!} f''(\zeta_{n-1}) = e_{n-1} f'(s) + \frac{e_{n-1}^2}{2!} f''(\zeta_{n-1})$$

이 되고 이것을 적용하면,

$$\begin{aligned}
 e_{n+1} &= \frac{f_n e_{n-1} - e_n f_{n-1}}{f_n - f_{n-1}} \\
 &= \frac{e_{n-1} \left(e_n f'(s) + \frac{e_n^2}{2!} f''(\zeta_n) \right) - e_n \left(e_{n-1} f'(s) + \frac{e_{n-1}^2}{2!} f''(\zeta_{n-1}) \right)}{\left(e_n f'(s) + \frac{e_n^2}{2!} f''(\zeta_n) \right) - \left(e_{n-1} f'(s) + \frac{e_{n-1}^2}{2!} f''(\zeta_{n-1}) \right)}
 \end{aligned}$$

여기에 $\zeta_n \approx s \approx \zeta_{n-1}$ 를 적용하고, 분모에만, $e_n^2 \approx 0 \approx e_{n-1}^2$ 을 적용하면,

$$\begin{aligned}
 &\approx \frac{1}{2} \frac{e_{n-1} e_n^2 f''(s) - e_n e_{n-1}^2 f''(s)}{e_n f'(s) - e_{n-1} f'(s)} \\
 &= \frac{1}{2} \frac{f''(s)}{f'(s)} \frac{e_{n-1} e_n^2 - e_n e_{n-1}^2}{e_n - e_{n-1}} = \frac{1}{2} \frac{f''(s)}{f'(s)} e_{n-1} e_n \frac{e_n - e_{n-1}}{e_n - e_{n-1}} \\
 &= \frac{1}{2} \frac{f''(s)}{f'(s)} e_{n-1} e_n
 \end{aligned}$$

그러므로 $e_{n+1} \approx C e_{n-1}e_n$ (여기서 $C = \frac{1}{2} \frac{f''(s)}{f'(s)}$) 를 얻었다.

한편 수렴차수 p 는 $e_{n+1} = C_1 e_n^p$ 로 정의된다.

한번 더 $e_n = C_1 e_{n-1}^p$ 을 적용하면,

$$\begin{aligned}
e_{n+1} &= C_1 (C_1 e_{n-1}^p)^p \\
&= C_1^{p+1} e_{n-1}^{p^2} \text{ ---(1)}
\end{aligned}$$

이 된다. 여기에 $e_{n+1} \approx C e_{n-1}e_n$ 로부터

$$\begin{aligned}
e_{n+1} &\approx C e_{n-1} (C_1 e_{n-1}^p) \\
&= C C_1 e_{n-1}^{p+1} \text{ ---(2)}
\end{aligned}$$

(1),(2)로부터

$$\begin{aligned}
p^2 &= p + 1 \\
p^2 - p - 1 &= 0
\end{aligned}$$

2차방정식의 근의 공식에 따라

$$p = \frac{1 + \sqrt{5}}{2} \approx 1.62 \blacksquare$$

[예제2.9] 방정식 $e^x - 1.4 - \tan^{-1}x = 0$ 의 근을 할선법으로 구하여라.

초기추측 $x_0 = -5, x_1 = -7$ 를 사용하여라.

(풀이) 할선법은 뉴턴법보다 약간 느리다. 수렴차수는 1.62 근방으로 나온다.

[숙제] $x^3 - x - 1 = 0$ 의 근을 할선법을 사용하여 구하고, 수렴차수도 구하여라.

초기추측은 $x_0 = 1, x_1 = 2$ 를 사용하여라.

고정점

실수 s 는 함수 g 의 고정점이다. $\Leftrightarrow s = g(s)$

고정점반복법

방정식 $f(x) = 0$ 을 $x = g(x)$ 형태로 변형한 다음, 위와 같이 반복한다.

$$x_{n+1} = g(x_n), n = 0, 1, 2, \dots$$

고정점반복법 알고리즘

입력 : $x_0, xtol, N$

for $n = 0$ **to** N

$$x_{n+1} = g(x_n)$$

if $|x_{n+1} - x_n| \leq xtol$ **then stop**

[예제2.10]

방정식 $x^2 - x - 3 = 0$ 의 근을 고정점 반복법으로 구하기 위하여
반복함수 $g(x)$ 를 찾아 보아라.

(풀이)

$$(1) x = x^2 - 3 \implies g(x) = x^2 - 3$$

$$(2) \quad x = \frac{x+3}{x} \implies g(x) = \frac{x+3}{x}$$

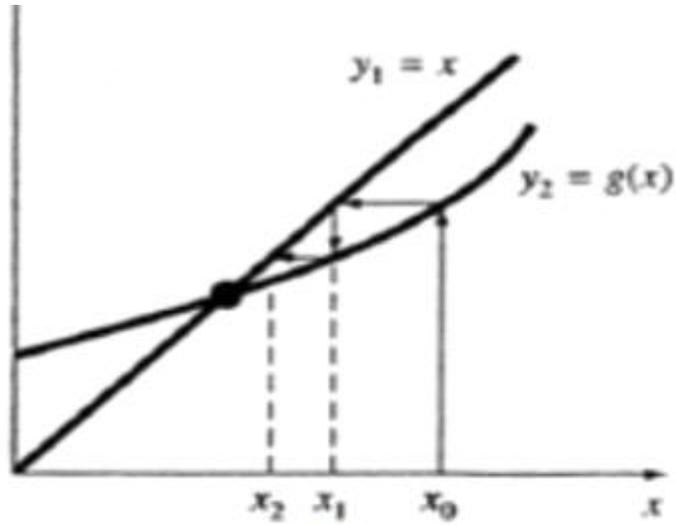
$$(3) \quad x = \sqrt{x+3} \implies g(x) = \sqrt{x+3}$$

$$(4) \quad x = x + 2(x^2 - x - 3) \implies g(x) = x + 2(x^2 - x - 3)$$

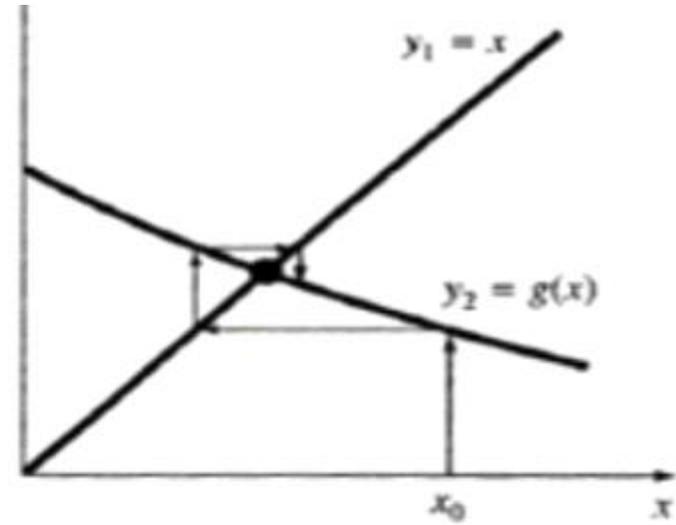
$$(5) \quad x = x - \frac{x^2 - x - 3}{2x - 1} \implies g(x) = x - \frac{x^2 - x - 3}{2x - 1}$$

[예제2.11] 방정식 $x^2 - x - 3 = 0$ 의 근을 예제2.10의 반복함수들을 사용한 고정점 반복법으로 구하여 비교하여라.

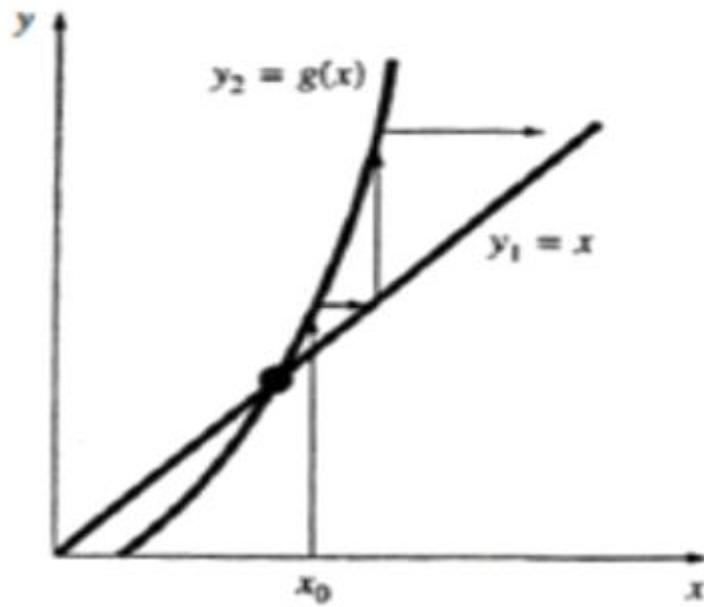
[숙제] p54 #6을 컴퓨터 프로그램을 실행하여 비교하여라.



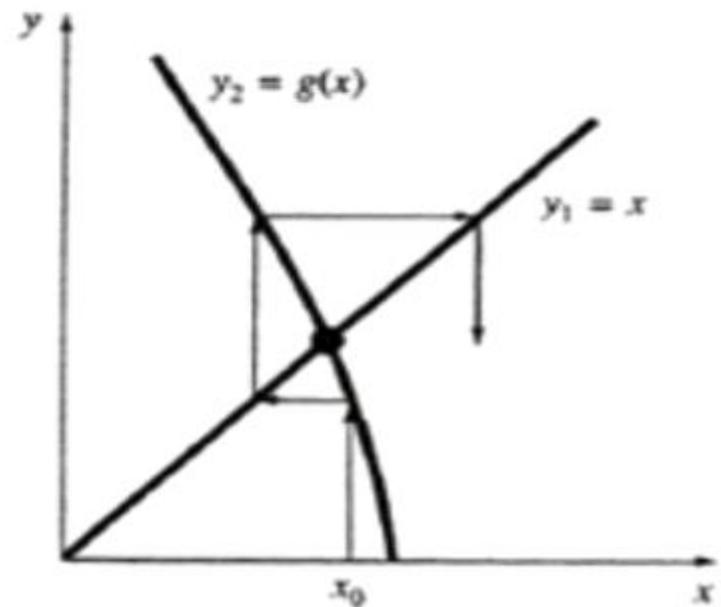
(a)



(b)



(c)



(d)

- [정리 2.3]** (1) $\forall x \in I = [a, b], g(x) \in I$
(2) $\forall x \in I, |g'(x)| \leq r < 1$
(3) 초기값 $x_0 \in I, \forall n \in \mathbb{N}, x_n = g(x_{n-1})$

\Rightarrow (1) 구간 I 안에 함수 g 의 고정점 s 가 존재.

(2) g 의 고정점은 유일하다.

(3) 수열 $\{x_n\}$ 은 고정점으로 수렴.

(4) $|x_n - s| \leq \frac{r^n}{1-r} |x_1 - x_0|$

(증명)

(1) 만약 $g(a) = a$ 또는 $g(b) = b$ 이면 고정점 존재.

$a < g(a) \wedge g(b) < b$ 인 경우, $g'(x)$ 가 존재하면 g 는 연속이므로,

$h(x) \equiv g(x) - x$ 도 연속이다. 또한 $h(a) = g(a) - a > 0$ 이고, 또한

$h(b) = g(b) - b < 0$ 이므로, 중간값의 정리에 의하여, $h(s) = 0$ 을 만족

하는 s 가 I 안에 존재한다. 그러므로 $g(s) - s = h(s) = 0$, 즉 $g(s) = s$ 인 s 가 존재한다.

(2) 서로 다른 두 개의 고정점 s 와 w 가 있다면, 평균값의 정리에 의하여,

$$g(s) - g(w) = g'(c)(s - w)$$

인 c 가 s 와 w 사이에 존재한다. 따라서

$$\begin{aligned} |s - w| &= |g(s) - g(w)| \\ &= |g'(c)| |s - w| \\ &\leq r |s - w| \\ &< |s - w| \end{aligned}$$

그러므로 $|s - w| < |s - w|$ 로 모순이다. 따라서 고정점은 유일하다.

(3) 평균값의 정리에 의하여, $\forall n \in \mathbb{N}$,

$$\begin{aligned} |x_n - s| &= |g(x_{n-1}) - g(s)| \\ &= |g'(c_n)| |x_{n-1} - s| \end{aligned}$$

$$\begin{aligned} &\leq r|x_{n-1} - s| \\ &\leq r^2|x_{n-2} - s| \\ &\leq r^n|x_{n-n} - s| \end{aligned}$$

그러므로

$$|x_n - s| \leq r^n|x_0 - s|$$

여기에서 $r < 1$ 이므로 우변은 0으로 수렴한다.

따라서 x_n 은 s 로 수렴한다.

(4) 삼각부등식에 의하여,

$$\begin{aligned} |s - x_0| &= |s - x_1 + x_1 - x_0| \\ &\leq |s - x_1| + |x_1 - x_0| \\ &\leq r|s - x_0| + |x_1 - x_0| \\ (1 - r)|s - x_0| &\leq |x_1 - x_0| \end{aligned}$$

$$|s - x_0| \leq \frac{1}{1-r} |x_1 - x_0| \text{ ---(\#)}$$

그런데 위 풀이 (3)으로부터,

$$\begin{aligned} |x_n - s| &\leq r^n |x_0 - s| \\ &\leq r^n \left(\frac{1}{1-r} |x_1 - x_0| \right), \text{ by(\#)} \end{aligned}$$

그러므로

$$|x_n - s| \leq \frac{r^n}{1-r} |x_1 - x_0| \quad \blacksquare$$

[명제] 정리2.3의 가정에, $g^{(k)}$ 가 연속이고,

$$\begin{aligned} 0 &= g'(s) = \dots = g^{(k-1)}(s) \\ 0 &\neq g^{(k)}(s) \end{aligned}$$

의 조건을 더하면, 수열 $\{x_n\}$ 의 수렴차수는 k 가 된다.

(증명)

오차를 $e_n = x_n - s$ 라 할 때,

$$\begin{aligned} e_{n+1} &= x_{n+1} - s \\ &= g(x_n) - g(s) \end{aligned}$$

여기에서 함수 $g^{(k)}$ 가 연속이면, by 테일러 정리, x_n 과 s 사이에 ζ_n 이 존재하여,

$$\begin{aligned} &= g'(s) e_n + \frac{g''(s)}{2!} e_n^2 + \dots + \frac{g^{(k-1)}(s)}{(k-1)!} e_n^{k-1} + \frac{g^{(k)}(\zeta_n)}{k!} e_n^k \\ &= \frac{g^{(k)}(\zeta_n)}{k!} e_n^k, \quad \text{by } 0 = g'(s) = \dots = g^{(k-1)}(s) \end{aligned}$$

따라서, $|e_{n+1}| \approx C |e_n|^k$ 여기서 $C = \left| \frac{g^{(k)}(s)}{k!} \right|$.

그러므로 수렴차수는 k 가 된다. ■

Aitken의 Δ^2 방법

고정점반복법을 가속하는 방법으로

$$\hat{x}_n = x_n - \frac{(\Delta x_n)^2}{\Delta^2 x_n}$$

여기에서 $\Delta x_n = x_{n+1} - x_n$

$$\begin{aligned}\Delta^2 x_n &= \Delta x_{n+1} - \Delta x_n \\ &= (x_{n+2} - x_{n+1}) - (x_{n+1} - x_n) \\ &= x_{n+2} - 2x_{n+1} + x_n\end{aligned}$$

(풀이) 고정점 반복법의 근사값 수열 $\{x_n\}$ 이 고정점(참값) s 으로 일차적으로 수렴한다고 하자. 그럼 어떤 상수 C 가 존재하여, 충분히 큰 n 에 대하여,

$$\begin{cases} x_{n+1} - s \approx C(x_n - s) \\ x_{n+2} - s \approx C(x_{n+1} - s) \end{cases}$$

좌변끼리 나누고, 우변끼리 나누면

$$\frac{x_{n+1} - s}{x_{n+2} - s} \approx \frac{x_n - s}{x_{n+1} - s}$$

$$\Rightarrow (x_{n+2} - s)(x_n - s) \approx (x_{n+1} - s)^2$$

$$\Rightarrow x_{n+2}x_n - (x_{n+2} + x_n)s \approx x_{n+1}^2 - 2x_{n+1}s$$

$$\Rightarrow s \approx \frac{x_{n+2}x_n - x_{n+1}^2}{x_{n+2} - 2x_{n+1} + x_n}$$

$$\Rightarrow s \approx \frac{x_n(x_{n+2} - 2x_{n+1} + x_n) - (x_{n+1}^2 - 2x_{n+1}x_n + x_n^2)}{x_{n+2} - 2x_{n+1} + x_n}$$

$$\Rightarrow s \approx x_n - \frac{(x_{n+1} - x_n)^2}{x_{n+2} - 2x_{n+1} + x_n}$$

$$\Rightarrow s \approx x_n - \frac{(\Delta x_n)^2}{\Delta^2 x_n} \quad \blacksquare$$

Steffensen방법 알고리즘

입력 : $x_0, xtol, N$

for $n = 0$ **to** N

$$y = g(x_n)$$

$$z = g(y)$$

$$x_{n+1} = x_n - (y - x_n)^2 / (z - 2y + x_n)$$

if $|x_{n+1} - x_n| \leq xtol$ **then stop**

[예제2.13] 함수 $f(x) = (x - 7)(x - 2 \sin x)^2$ (책오타수정필요), 초기값과 종료조건을 $x_0 = 3$, $xtol = 10^{-10}$ 로 하고, 반복함수 $g(x) \equiv x - \frac{f(x)}{f'(x)}$ 에 Steffensen법을 적용하여 중근에 대한 근사값을 구하고, 뉴턴방법은 중근에 대하여 원래는 1차적을 수렴하는데, Steffensen법으로는 더 빠른 2차적으로 수렴한다.

[숙제] 방정식 $x - e^{-x} = 0$ 의 구간 $[0, 1]$ 위 근의 근사값을 Steffensen 방법으로 구하라. 종료조건 $xtol = 10^{-4}$ 을 사용하라.

제3장 선형연립방정식의 해법

선형대수 요약

행렬, 상삼각행렬, 하삼각행렬, 항등행렬, 전치행렬, 대칭행렬, 교대행렬

역행렬, 정칙행렬, 특이행렬, 행렬식(determinant)

정방행렬의 열공간 = 열벡터들이 (일차결합들로)생성하는 공간
= 선형변환 $y = Ax$ 의 치역

$rank(A)$ = 열공간의 차원

$Ker(A)$ = 핵공간 = $\{x \in \mathbb{R}^n \mid Ax = 0\}$

$null(A)$ = 핵공간의 차원

일차독립, 일차종속.

고유치(λ), 고유벡터(x) : $Ax = \lambda x, x \neq 0$

[정리3.2] [정리3.5] $n \times n$ 정방행렬에 대하여 다음은 동치이다.

- (1) A 가 정칙행렬이다 (즉 $\exists A^{-1}$)
- (2) $Ax = 0$ 는 단 하나의 해 $x = 0$ 을 갖는다.
- (3) $Ax = b$ 가 유일한 해를 갖는다.
- (4) $\det(A) \neq 0$
- (5) $\text{rank}(A) = n$
- (6) $\text{null}(A) = 0$
- (7) 선형변환 $F(x) = Ax$ 는 단사(1-1) 함수이다.
- (8) A 가 일차독립인 n 개의 열들을 가진다.
- (9) A 가 일차독립인 n 개의 행들을 가진다.

[정리3.7] 실수(또는 복소수) λ 는 $n \times n$ 행렬 A 의 고유치이다.

$$\Leftrightarrow \det(A - \lambda I) = 0$$

Gauss 소거법

[예제3.1] 다음 연립방정식을 Gauss소거법으로 풀어라.

(1) 상삼각화

$$\begin{cases} 2x_1 + 4x_2 - 2x_3 = 6 \\ x_1 - x_2 + 5x_3 = 0 \\ 4x_1 + x_2 - 2x_3 = 2 \end{cases} \quad \left[\begin{array}{ccc|c} 2 & 4 & -2 & 6 \\ 1 & -1 & 5 & 0 \\ 4 & 1 & -2 & 2 \end{array} \right] : \text{증가행렬}$$

$$\Rightarrow \begin{cases} 2x_1 + 4x_2 - 2x_3 = 6 \\ -3x_2 + 6x_3 = -3 \\ 4x_1 + x_2 + 2x_3 = 2 \end{cases} \quad \left[\begin{array}{ccc|c} 2 & 4 & -2 & 6 \\ [1/2] & -3 & 6 & -3 \\ 4 & 1 & -2 & 2 \end{array} \right]$$

$$\Rightarrow \begin{cases} 2x_1 + 4x_2 - 2x_3 = 6 \\ -3x_2 + 6x_3 = -3 \\ -7x_2 + 2x_3 = -10 \end{cases} \quad \left[\begin{array}{ccc|c} 2 & 4 & -2 & 6 \\ [1/2] & -3 & 6 & -3 \\ [2] & -7 & 2 & -10 \end{array} \right]$$

$$\Rightarrow \begin{cases} 2x_1 + 4x_2 - 2x_3 = 6 \\ -3x_2 + 6x_3 = -3 \\ -12x_3 = -3 \end{cases} \quad \left[\begin{array}{ccc|c} 2 & 4 & -2 & 6 \\ [1/2] & -3 & 6 & -3 \\ [2] & [7/3] & -12 & -3 \end{array} \right]$$

(2) 후진대입법

$$x_3 = (-3)/(-12) = 1/4$$

$$x_2 = (-3 - 6x_3)/(-3) = 3/2$$

$$x_1 = (6 - 4x_2 + 2x_3)/2 = 1/4$$



Gauss소거법 알고리즘

입력 : A, x, b, n

#상삼각화

for $k = 1$ to $n - 1$

for $i = k + 1$ to n

$$l_{ik} = a_{ik}/a_{kk}$$

for $j = k + 1$ to n

$$a_{ij} = a_{ij} - l_{ik}a_{kj}$$

end

$$b_i = b_i - l_{ik}b_k$$

end

end

#후진대입법

```
x = b
for i = n to 1
  s = 0
  for j = i + 1 to n
    s = s + aijxj
  end
  xi = (xi - s) / aii
end
```

[숙제] p92#1(2) Gauss소거법

피벗원소(pivot element) : Gauss소거법에서 대각선에 나타난 원소

[주의] 피벗원소가 0 과 가까우면 계산에서 오차가 커지고, 0 이 되면 Gauss소거가 불가능하게 된다. 그런 경우 피하기 위하여, 피벗원소가 있는 행을 절대값이 큰 원소를 가진 다른 행과 교환하여 소거를 진행한다. 이것을 **피벗팅(pivoting)**이라고 한다. **전체피벗**과 **부분피벗**이 있는데, 전체피벗은 시간이 많이 걸리므로 부분피벗이 많이 이용된다.

[예제3.3]
$$\begin{cases} \varepsilon x_1 + x_2 = 2 \\ x_1 + x_2 = 3 \end{cases}$$
 여기서 ε 는 매우 작은 양수이다.

(풀이) (1) 피벗원소가 ε 인 경우

$$\begin{cases} \varepsilon x_1 + x_2 = 2 \\ (1 - 1/\varepsilon)x_2 = 3 - 2/\varepsilon \end{cases}$$

컴퓨터의 오차 계산에 따라

$$\begin{cases} x_2 = \frac{3 - 2/\varepsilon}{(1 - 1/\varepsilon)} \approx \frac{-2/\varepsilon}{-1/\varepsilon} = 2 \\ x_1 = 0 \end{cases}$$

(2) 피벗원소가 1인 경우 (1행과 2행을 교환하여)

$$\begin{cases} x_1 + x_2 = 3 \\ \varepsilon x_1 + x_2 = 2 \end{cases}$$

$$\begin{cases} x_1 + x_2 = 3 \\ (1 - \varepsilon)x_2 = 2 - 3\varepsilon \end{cases}$$

컴퓨터의 오차 계산에 따라

$$\begin{cases} x_2 = \frac{2-3\varepsilon}{(1-\varepsilon)} \approx \frac{2}{1} = 2 \\ x_1 = 1 \end{cases}$$

한편, 위 연립방정식의 정확한 해는 $x_1 = \frac{1}{1-\varepsilon}$, $x_2 = \frac{2-3\varepsilon}{1-\varepsilon}$ 이므로

참값은 $x_1 \approx 1$, $x_2 \approx 2$ 이다. 즉 (2)가 바른 방법이다. ■

(숙제) 부분피벗Guass소거법 (필기로 풀기) (예제3.4참조)

$$\begin{bmatrix} 1 & -1 & 2 \\ -2 & 1 & -1 \\ 4 & -1 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} -2 \\ 2 \\ -1 \end{bmatrix}$$

행렬의 기본행연산

- (1) 서로 다른 두 행을 바꾼다. (row exchange)
- (2) 한 행을 $k(\neq 0)$ 배 한다.
- (3) 한 행에 다른 행의 k 배를 더한다.

행렬의 행동치(동치관계)

행렬에 기본행연산을 유한번 행하여 얻은 행렬은 원래 행렬과 행동치이다.

Gauss소거 연립선형방정식 해법

증가행렬과 행동치인 상삼각행렬을 만들어 후진대입법으로 푼다.

기본행렬(3가지 타입)

항등행렬에 기본행연산(3가지 타입)을 행하여 만든 행렬.

(예) $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -2 & 0 & 1 \end{bmatrix}$ 3×3 단위행렬의 첫행에 -2 를 곱하여 3째행에 더한다.

기본행렬의 성질

(1) 행렬에 기본행연산을 하는 것이나 그 기본행연산을 항등행렬에 행하여 얻어지는 기본행렬을 그 행렬의 왼쪽에서 곱한 것이나 같은 결과를 준다.

(2) 기본행렬의 역행렬

$$\begin{bmatrix} 1 & 0 & 0 \\ m & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ -m & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

(3) 기본행렬들의 곱

$$\begin{bmatrix} 1 & 0 & 0 \\ m & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ n & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ m & 1 & 0 \\ n & 0 & 1 \end{bmatrix}$$

(숙제) p102#8 기본행렬들의 곱

LU 분해법

$$A = \begin{bmatrix} 2 & 1 & 1 \\ -1 & 2 & 1 \\ 4 & 0 & 4 \end{bmatrix}$$

$$L_1 A = \begin{bmatrix} 1 & 0 & 0 \\ 1/2 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 2 & 1 & 1 \\ -1 & 2 & 1 \\ 4 & 0 & 4 \end{bmatrix} = \begin{bmatrix} 2 & 1 & 1 \\ 0 & 5/2 & 3/2 \\ 4 & 0 & 4 \end{bmatrix}$$

$$L_2 L_1 A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -2 & 0 & 1 \end{bmatrix} \begin{bmatrix} 2 & 1 & 1 \\ 0 & 5/2 & 3/2 \\ 4 & 0 & 4 \end{bmatrix} = \begin{bmatrix} 2 & 1 & 1 \\ 0 & 5/2 & 3/2 \\ 0 & -2 & 2 \end{bmatrix}$$

$$L_3 L_2 L_1 A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 4/5 & 1 \end{bmatrix} \begin{bmatrix} 2 & 1 & 1 \\ 0 & 5/2 & 3/2 \\ 0 & -2 & 2 \end{bmatrix} = \begin{bmatrix} 2 & 1 & 1 \\ 0 & 5/2 & 3/2 \\ 0 & 0 & 16/5 \end{bmatrix}$$

$$L_3 L_2 L_1 A = \begin{bmatrix} 2 & 1 & 1 \\ 0 & 5/2 & 3/2 \\ 0 & 0 & 16/5 \end{bmatrix}$$

$$\begin{aligned}
A &= L_1^{-1} L_2^{-1} L_3^{-1} \begin{bmatrix} 2 & 1 & 1 \\ 0 & 5/2 & 3/2 \\ 0 & 0 & 16/5 \end{bmatrix} \\
&= \begin{bmatrix} 1 & 0 & 0 \\ -1/2 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 2 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -4/5 & 1 \end{bmatrix} \begin{bmatrix} 2 & 1 & 1 \\ 0 & 5/2 & 3/2 \\ 0 & 0 & 16/5 \end{bmatrix} \\
&= \begin{bmatrix} 1 & 0 & 0 \\ -1/2 & 1 & 0 \\ 2 & -4/5 & 1 \end{bmatrix} \begin{bmatrix} 2 & 1 & 1 \\ 0 & 5/2 & 3/2 \\ 0 & 0 & 16/5 \end{bmatrix} \\
&= LU
\end{aligned}$$

LU분해후 연립선형방정식 해법

$$\begin{aligned}
Ax = b &\Leftrightarrow L U x = b \\
&\Leftrightarrow Ly = b \wedge Ux = y
\end{aligned}$$

[예제 3.6]

$$Ax = b \Leftrightarrow \begin{bmatrix} 2 & 1 & 1 \\ -1 & 2 & 1 \\ 4 & 0 & 4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}$$

$$\Leftrightarrow \begin{bmatrix} 1 & 0 & 0 \\ -1/2 & 1 & 0 \\ 2 & -4/5 & 1 \end{bmatrix} \begin{bmatrix} 2 & 1 & 1 \\ 0 & 5/2 & 3/2 \\ 0 & 0 & 16/5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}$$

먼저, $\begin{bmatrix} 1 & 0 & 0 \\ -1/2 & 1 & 0 \\ 2 & -4/5 & 1 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}$ 를

전진대입법으로 풀면 $y_1 = 1, y_2 = -1/2, y_3 = -7/5$

다음으로 $\begin{bmatrix} 2 & 1 & 1 \\ 0 & 5/2 & 3/2 \\ 0 & 0 & 16/5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1 \\ -1/2 \\ -7/5 \end{bmatrix}$ 를

후진대입법으로 풀면 $x_3 = -7/16, x_2 = 1/16, x_1 = 11/16$ ■

[정의3.15]

벡터공간 V 에 대하여, 함수 $\| \cdot \| : V \rightarrow \mathbf{R}$ 를 **벡터노름(norm)**이라 한다.

정의

$$\Leftrightarrow (1) \quad \forall x \in V, \|x\| \geq 0$$

$$(2) \quad \forall \alpha \in \mathbf{R}, \forall x \in V, \|\alpha x\| = |\alpha| \|x\|$$

$$(3) \quad \forall x, y \in V, \|x + y\| \leq \|x\| + \|y\| \quad (\text{삼각부등식})$$

$$(4) \quad (\|x\| = 0 \Leftrightarrow x = 0)$$

정의

[정의] 벡터공간에서 $x_n \rightarrow x \Leftrightarrow \|x_n - x\| \rightarrow 0$

노름(norm)의 예

벡터 공간 $V = \mathbf{R}^n$ 의 원소 $x = [x_1, x_2, \dots, x_n]^T$ 에 대하여 다음 정의되는 함수들은 노름이 된다.

$$\|x\|_1 = \left(\sum_{i=1}^n |x_i| \right)$$

$$\|x\|_2 = \left(\sum_{i=1}^n x_i^2 \right)^{1/2}$$

$$\|x\|_p = \left(\sum_{i=1}^n |x_i|^p \right)^{1/p}$$

$$\|x\|_\infty = \max_{1 \leq i \leq n} |x_i|$$

[예제3.9] 벡터 $x = [3, -2, 4]^T$ 에 대하여, $\|x\|_1, \|x\|_2, \|x\|_\infty$ 를 구하여라.

행렬의 노름

$$\|A\| \stackrel{\text{정의}}{=} \max_{x \neq 0} \frac{\|Ax\|}{\|x\|}$$

벡터노름에 따라

$$\|A\|_1 = \max_{x \neq 0} \frac{\|Ax\|_1}{\|x\|_1}, \quad \|A\|_p = \max_{x \neq 0} \frac{\|Ax\|_p}{\|x\|_p}, \quad \|A\|_\infty = \max_{x \neq 0} \frac{\|Ax\|_\infty}{\|x\|_\infty}$$

행렬노름의 성질

- (1) $\|A\| \geq 0$
- (2) $\|\alpha A\| = |\alpha| \|A\|$
- (3) $\|A + B\| \leq \|A\| + \|B\|$
- (4) $\|A\| = 0 \Leftrightarrow A = 0$
- (5) $\|Ax\| \leq \|A\| \|x\|$
- (6) $\|AB\| \leq \|A\| \|B\|$
- (7) $\|I\| = 1, I$ 는 항등행렬

(증명) (5) $\|A\| = \max_{x \neq 0} \frac{\|Ax\|}{\|x\|}$ 이므로

모든 $x \neq 0$ 에 대하여

$$\frac{\|Ax\|}{\|x\|} \leq \|A\| \quad \text{이다. 그러므로 } \|Ax\| \leq \|A\| \|x\| \quad \text{이다.}$$

모든 $x = 0$ 에 대하여, $Ax = 0$ 이고 $\|0\| = 0$ 이므로

$$\|Ax\| = 0 = \|A\| \|x\| \quad \text{이므로 역시 성립한다.} \quad \blacksquare$$

[정리3.18] $n \times n$ 행렬 $A = [a_{ij}]$ 에 대하여,

$$(1) \|A\|_{\infty} = \max_{1 \leq i \leq n} \sum_{j=1}^n |a_{ij}| \quad (2) \|A\|_1 = \max_{1 \leq j \leq n} \sum_{i=1}^n |a_{ij}|$$

(증명)

$$(1) \|Ax\|_{\infty} = \left\| \left[\sum_{j=1}^n a_{ij} x_j \right] \right\|_{\infty}$$

$$\begin{aligned} &= \max_{1 \leq i \leq n} \left| \sum_{j=1}^n a_{ij} x_j \right| \\ &\leq \max_{1 \leq i \leq n} \left| \sum_{j=1}^n a_{ij} \right| |x_j| \\ &\leq \left(\max_{1 \leq i \leq n} \left| \sum_{j=1}^n a_{ij} \right| \right) \|x\|_\infty \end{aligned}$$

$x \neq 0$ 에 대하여,

$$\frac{\|Ax\|_\infty}{\|x\|_\infty} \leq \left(\max_{1 \leq i \leq n} \left| \sum_{j=1}^n a_{ij} \right| \right)$$

그러므로 $\|A\|_\infty = \max_{x \neq 0} \frac{\|Ax\|_\infty}{\|x\|_\infty} \leq \left(\max_{1 \leq i \leq n} \left| \sum_{j=1}^n a_{ij} \right| \right)$ --[1]

반대방향의 부등호를 보이기 위하여 특별한 x 를 선택하여 보자.

$\left| \sum_{j=1}^n a_{ij} \right|$ 들 중에서 $i = k$ 에서 최대값을 가진다고 할 때, 즉

$$\max_{1 \leq i \leq n} \left| \sum_{j=1}^n a_{ij} \right| = \left| \sum_{j=1}^n a_{kj} \right|$$

라고 할 때, $x = [x_j]$ 를 다음과 같이 정의하자.

$$x_j = \begin{cases} \frac{a_{kj}}{|a_{kj}|}, & a_{kj} \neq 0 \\ 1, & a_{kj} = 0 \end{cases}$$

그러면 모든 $|x_j| = 1$ 이 되어 $\|x\|_\infty = 1$ 임을 알 수 있다. 그래서

$$\begin{aligned} \|Ax\|_\infty &= \left\| \left[\sum_{j=1}^n a_{ij} x_j \right] \right\|_\infty = \max_{1 \leq i \leq n} \left| \sum_{j=1}^n a_{ij} x_j \right| \\ &\geq \left| \sum_{j=1}^n a_{kj} x_j \right| \geq \left| \sum_{\substack{j=1 \\ a_{kj} \neq 0}}^n a_{kj} \frac{a_{kj}}{|a_{kj}|} \right| \end{aligned}$$

$$\geq \left| \sum_{\substack{j=1 \\ a_{kj} \neq 0}}^n \frac{|a_{kj}|^2}{|a_{kj}|} \right| = \sum_{j=1}^n |a_{kj}| = \max_{1 \leq i \leq n} \left| \sum_{j=1}^n a_{ij} \right|$$

그러므로

$$\|Ax\|_{\infty} \geq \max_{1 \leq i \leq n} \left| \sum_{j=1}^n a_{ij} \right|$$

그런데 $\|x\|_{\infty} = 1$ 이므로

$$\frac{\|Ax\|_{\infty}}{\|x\|_{\infty}} \geq \max_{1 \leq i \leq n} \left| \sum_{j=1}^n a_{ij} \right|$$

따라서

$$\|A\|_{\infty} = \max_{x \neq 0} \frac{\|Ax\|_{\infty}}{\|x\|_{\infty}} \geq \left(\max_{1 \leq i \leq n} \left| \sum_{j=1}^n a_{ij} \right| \right) \text{ --[2]}$$

[1],[2] 에 의하여 결론이 증명되었다. ■

[예제3.10] 다음 행렬 A 에 대하여 $\|A\|_\infty$ 와 $\|A\|_1$ 을 구하여라.

$$A = \begin{bmatrix} 4 & -3 & 2 \\ -1 & 0 & 5 \\ 2 & 6 & -2 \end{bmatrix}$$

(풀이)

$$(1) \|A\|_\infty = \max_{1 \leq i \leq n} \sum_{j=1}^n |a_{ij}| = \max\{9, 6, 10\} = 10$$

$$(2) \|A\|_1 = \max_{1 \leq j \leq n} \sum_{i=1}^n |a_{ij}| = \max\{7, 9, 9\} = 9 \quad \blacksquare$$

조건상수

A 가 정칙행렬(=역행렬를 가진 행렬)일 때,

$$\kappa(A) = \|A\| \|A^{-1}\|$$

을 A 의 **조건상수**라고 한다.

[예제3.11] 다음 행렬의 조건상수를 $\|\cdot\|_\infty$ 을 사용하여 구하여라.

$$A_\varepsilon = \begin{bmatrix} 1 & 1+\varepsilon \\ 1-\varepsilon & 1 \end{bmatrix}, \quad (0 < \varepsilon < 1)$$

(풀이)

$$A_\varepsilon^{-1} = \frac{1}{\varepsilon^2} \begin{bmatrix} 1 & -1-\varepsilon \\ -1+\varepsilon & 1 \end{bmatrix} \text{ 이므로}$$

$$\|A_\varepsilon\|_\infty = 2 + \varepsilon, \quad \|A_\varepsilon^{-1}\|_\infty = \frac{1}{\varepsilon^2} (2 + \varepsilon) \text{ 이다 따라서}$$

$$\kappa(A) = \|A\|_\infty \|A^{-1}\|_\infty = \left(1 + \frac{2}{\varepsilon}\right)^2 \quad \blacksquare$$

[참고]

예제3.11에서 ε 이 0 (zero)로 가면 행렬 A_ε 는 비정칙행렬로 가까이 간다. 그 때 조건상수는 무한히 커진다. 한편 A 가 정칙 행렬이면

$$1 = \|I\| = \|A A^{-1}\| \leq \|A\| \|A^{-1}\| = \kappa(A) \text{ 이다.}$$

잉여벡터(residual vector) : $b - A\tilde{x}$

[정리3.20] A 가 정칙행렬일때 선형연립방정식 $Ax = b$ ($b \neq 0$)의 참해를 x^* 라고 하고, 근사해를 \tilde{x} 할 때, 다음이 성립한다.

$$\frac{1}{\kappa(A)} \frac{\|b - A\tilde{x}\|}{\|b\|} \leq \frac{\|\tilde{x} - x^*\|}{\|x^*\|} \leq \kappa(A) \frac{\|b - A\tilde{x}\|}{\|b\|}$$

(둘째 부등식 증명)

$$\begin{aligned} b - A\tilde{x} &= Ax^* - A\tilde{x} \\ &= A(x^* - \tilde{x}) \text{ 이므로} \end{aligned}$$

$$x^* - \tilde{x} = A^{-1}(b - A\tilde{x}) \text{ 이다.}$$

따라서

$$\begin{aligned} \|x^* - \tilde{x}\| &= \|A^{-1}(b - A\tilde{x})\| \\ &\leq \|A^{-1}\| \|b - A\tilde{x}\| \quad \text{--[1]} \end{aligned}$$

한편 $b = Ax^*$ 로부터,

$$\begin{aligned} \|b\| &= \|Ax^*\| \\ &\leq \|A\| \|x^*\| \end{aligned}$$

따라서

$$\frac{1}{\|x^*\|} \leq \frac{\|A\|}{\|b\|} \quad \text{--[2]}$$

[1], [2] 로부터

$$\begin{aligned} \frac{\|x^* - \tilde{x}\|}{\|x^*\|} &\leq \|A\| \|A^{-1}\| \frac{\|(b - A\tilde{x})\|}{\|b\|} \\ &= \kappa(A) \frac{\|(b - A\tilde{x})\|}{\|b\|} \quad \blacksquare \end{aligned}$$

(숙제) 위 예제3.20 의 첫 번째 부등식 증명

불량조건행렬

정리3.20 으로부터 조건상수가 크면 잉여벡터의 크기 $\|b - A\tilde{x}\|$ 가 아무리 작더라도 근사해 \tilde{x} 의 참해 x^* 에 대한 상대오차 $\frac{\|x^* - \tilde{x}\|}{\|x^*\|}$ 가 커질수 있다 것을 알 수 있다. 이와 같이 조건상수가 큰 행렬을 **불량조건행렬**(ill-conditioned matrix)라고 한다.

그러니까 불량조건이면 벡터 b (보통 실험 관찰에서 얻어지는 수치) 의 조그만 오차에도 근사해의 오차가 커질 수 있다는 점을 알 수 있다.

선형연립방정식($Ax = b$)해법

- 1) **직접법** : 가우스소거법(LU분해법) - 많은 연산량과 기억량 필요
- 2) **간접법** : 반복법 - 적은 연산량과 기억량 필요.

행렬(A) 분할

$A = M - N$ 여기에서 M 은 정칙행렬($\exists M^{-1}$).

반복법(iterative method)

$$Ax = b$$

$$(M - N)x = b$$

$$Mx = Nx + b$$

초기(추측)벡터 $x^{(0)}$

k 단계 근사수열 $x^{(k)}$: $Mx^{(k)} = Nx^{(k-1)} + b, k = 1, 2, 3, \dots$

[정리 3.22] 반복법 수렴의 조건

행렬의 분할 $A = M - N$ 에 대하여, 만약 $\|M^{-1}N\| \leq \alpha < 1$ 이면,

임의의 초기벡터에 $x^{(0)}$ 에 대하여, 반복법 $Mx^{(k)} = Nx^{(k-1)} + b$ 에

의하여 주어진 수열 $\langle x^{(k)} \rangle$ 는 $Ax = b$ 의 정확한 해 x^* 에 수렴한다

또한

$$\|x^{(k)} - x^*\| \leq \alpha^k \|x^{(0)} - x^*\|$$

$$\|x^{(k)} - x^*\| \leq \frac{\alpha^k}{1 - \alpha} \|x^{(0)} - x^{(1)}\|$$

(증명)

정확한 해 x^* 는 방정식 $Ax = b$ 을 만족하므로 $Ax^* = b$ 이다.

그러므로

$$Ax^* = b \Rightarrow (M - N)x^* = b$$

$$\Rightarrow Mx^* = Nx^* + b$$

$$\Rightarrow M(x^{(k)} - x^*) = Mx^{(k)} - Mx^*$$

$$= (Nx^{(k-1)} + b) - (Nx^* + b)$$

$$= N(x^{(k-1)} - x^*)$$

$$\Rightarrow (x^{(k)} - x^*) = M^{-1}N(x^{(k-1)} - x^*)$$

$$\begin{aligned}\Rightarrow \|x^{(k)} - x^*\| &= \|M^{-1}N(x^{(k-1)} - x^*)\| \\ &\leq \|M^{-1}N\| \|x^{(k-1)} - x^*\| \\ &\leq \alpha \|x^{(k-1)} - x^*\|\end{aligned}$$

$$\begin{aligned}\Rightarrow \|x^{(k)} - x^*\| &\leq \alpha \|x^{(k-1)} - x^*\| \text{ -- [1]} \\ &\leq \alpha \alpha \|x^{(k-2)} - x^*\| \text{ by [1]} \\ &\leq \alpha^2 \|x^{(k-2)} - x^*\| \\ &\leq \dots \\ &\leq \alpha^k \|x^{(0)} - x^*\|\end{aligned}$$

$$\Rightarrow \|x^{(k)} - x^*\| \leq \alpha^k \|x^{(0)} - x^*\| \text{ -- [2]}$$

$$\Rightarrow \|x^{(k)} - x^*\| \rightarrow 0 \text{ since } \alpha^k \rightarrow 0 \text{ as } k \rightarrow \infty$$

$$\Rightarrow x^{(k)} \rightarrow x^*$$

두 번째 부등식을 증명해 보자.

$$\begin{aligned} \|x^{(0)} - x^*\| &= \|x^{(0)} - x^{(1)} + x^{(1)} - x^*\| \\ &\leq \|x^{(0)} - x^{(1)}\| + \|x^{(1)} - x^*\|, \text{ by 삼각부등식} \\ &\leq \|x^{(0)} - x^{(1)}\| + \alpha \|x^{(0)} - x^*\| \end{aligned}$$

$$\Rightarrow (1 - \alpha) \|x^{(0)} - x^*\| \leq \|x^{(0)} - x^{(1)}\|$$

$$\Rightarrow \|x^{(0)} - x^*\| \leq \frac{1}{(1 - \alpha)} \|x^{(0)} - x^{(1)}\| \text{ --[3]}$$

[2]로부터, $\|x^{(k)} - x^*\| \leq \alpha^k \|x^{(0)} - x^*\|$

$$\leq \frac{\alpha^k}{(1 - \alpha)} \|x^{(0)} - x^{(1)}\| \text{ by [3]}$$

그러므로

$$\|x^{(k)} - x^*\| \leq \frac{\alpha^k}{(1 - \alpha)} \|x^{(0)} - x^{(1)}\| \quad \blacksquare$$

[설명]

정리3.22 로부터, 수렴하기 위한 충분조건으로 $\|M^{-1}N\| < 1$ 인 분할 $A = M - N$ 을 찾아야 하고, $Mx^{(k)} = Nx^{(k-1)} + b$ 와 같이 각 단계에서 $x^{(k)}$ 를 얻으려면 M 을 계수로 갖는 연립방정식을 쉽게 풀 수 있어야 한다.

Recharadson 방법

$M = I$ 으로 선택하면, $N = I - A$ 가 된다. 그러므로

$$x^{(k)} = (I - A)x^{(k-1)} + b, \quad k = 1, 2, 3, \dots$$

Jacobi 방법

$M = D$ 으로 선택하면, $N = -L - U$ 가 된다. 그러므로

$$Dx^{(k)} = (-L - U)x^{(k-1)} + b, \quad k = 1, 2, 3, \dots$$

여기서 $A = L + D + U$, D :대각행렬, L :하삼각행렬, U :상삼각행렬

Gauss-Seidel 방법

$M = L + D$ 으로 선택하면, $N = -U$ 가 된다. 그러므로

$$(D + L)x^{(k)} = -Ux^{(k-1)} + b, \quad k = 1, 2, 3, \dots$$

여기서 $A = L + D + U$, D :대각행렬, L :하삼각행렬, U :상삼각행렬

[정의3.23]

$n \times n$ 행렬 A 는 강대각지배행렬이다.

정의

$$\Leftrightarrow 1 \leq \forall i \leq n, \left(\sum_{j=1, j \neq i}^n |a_{ij}| \right) < |a_{ii}|$$

[정리3.24]

$n \times n$ 행렬 A 는 강대각지배행렬이면, 임의의 초기벡터 $x^{(0)}$ 에 대하여, Jacobi 방법은 항상 $Ax = b$ 의 해에 수렴한다.

(증명)

정리3.22에 의하여 $\|D^{-1}(-L-U)\| < 1$ 를 보이면 된다.

노름은 아무나 선택해도 되므로, $\|D^{-1}(-L-U)\|_{\infty} < 1$ 을 보이자.

$$\begin{aligned}
 & D^{-1}(-L-U) \\
 &= \begin{bmatrix} a_{11} & 0 & 0 & \cdots & 0 \\ 0 & a_{22} & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & 0 \\ 0 & 0 & \cdots & \cdots & a_{nn} \end{bmatrix}^{-1} \begin{bmatrix} 0 & -a_{12} & -a_{13} & \cdots & -a_{1n} \\ -a_{21} & 0 & -a_{23} & \cdots & -a_{2n} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ -a_{n1} & -a_{n2} & -a_{n3} & \cdots & 0 \end{bmatrix} \\
 &= \begin{bmatrix} 1/a_{11} & 0 & 0 & \cdots & 0 \\ 0 & 1/a_{22} & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & 0 \\ 0 & 0 & \cdots & \cdots & 1/a_{nn} \end{bmatrix} \begin{bmatrix} 0 & -a_{12} & -a_{13} & \cdots & -a_{1n} \\ -a_{21} & 0 & -a_{23} & \cdots & -a_{2n} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ -a_{n1} & -a_{n2} & -a_{n3} & \cdots & 0 \end{bmatrix}
 \end{aligned}$$

$$= \begin{bmatrix} 0 & -a_{12}/a_{11} & -a_{13}/a_{11} & \cdots & -a_{1n}/a_{11} \\ -a_{21}/a_{22} & 0 & -a_{23}/a_{22} & \cdots & -a_{2n}/a_{22} \\ \cdots & \cdots & 0 & \cdots & \cdots \\ -a_{n1}/a_{nn} & -a_{n2}/a_{nn} & -a_{n3}/a_{nn} & \cdots & 0 \end{bmatrix}$$

$$\|D^{-1}(-L-U)\|_{\infty} = \max_{1 \leq i \leq n} \left(\sum_{j=1, j \neq i}^n \left| \frac{a_{ij}}{a_{ii}} \right| \right) \quad \text{by 정리3.18}$$

그런데 A 가 강대각지배행렬이므로, $1 \leq \forall i \leq n$,

$$\left(\sum_{j=1, j \neq i}^n |a_{ij}| \right) < |a_{ii}|$$

따라서,

$$\left(\sum_{j=1, j \neq i}^n \frac{|a_{ij}|}{|a_{ii}|} \right) < 1$$

그러므로

$$\|D^{-1}(-L-U)\|_{\infty} < 1 \quad \blacksquare$$

[정리3.29]

$n \times n$ 행렬 A 는 강대각지배행렬이면, 임의의 초기벡터 $x^{(0)}$ 에 대하여, Gauss-Seidel 방법은 항상 $Ax = b$ 의 해에 수렴한다.

Jacobi 방법 실행

$$Ax = b$$

$$(L + D + U)x = b$$

$$Dx = (-L - U)x + b$$

$$Dx^{(k)} = (-L - U)x^{(k-1)} + b, \quad k = 1, 2, 3, \dots$$

$$\begin{bmatrix} a_{11} & 0 & 0 & \cdots & 0 \\ 0 & a_{22} & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & 0 \\ 0 & 0 & \cdots & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1^{(k)} \\ x_2^{(k)} \\ \vdots \\ x_n^{(k)} \end{bmatrix} = \begin{bmatrix} 0 & -a_{12} & -a_{13} & \cdots & -a_{1n} \\ -a_{21} & 0 & -a_{23} & \cdots & -a_{2n} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ -a_{n1} & -a_{n2} & -a_{n3} & \cdots & 0 \end{bmatrix} \begin{bmatrix} x_1^{(k-1)} \\ x_2^{(k-1)} \\ \vdots \\ x_n^{(k-1)} \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}$$

$$\begin{aligned}
 a_{11}x_1^{(k)} &= 0 - a_{12}x_2^{(k-1)} - a_{13}x_3^{(k-1)} \cdots - a_{1n}x_n^{(k-1)} + b_1 \\
 a_{22}x_2^{(k)} &= -a_{21}x_1^{(k-1)} + 0 - a_{23}x_3^{(k-1)} \cdots - a_{2n}x_n^{(k-1)} + b_2 \\
 \cdots &= \cdots \cdots 0 \cdots \cdots \cdots \\
 a_{nn}x_n^{(k)} &= -a_{n1}x_1^{(k-1)} - a_{n2}x_2^{(k-1)} - a_{n3}x_3^{(k-1)} \cdots 0 + b_n
 \end{aligned}$$

Jacobi 알고리즘

```

입력 :  $n, A, x^{(0)}, b, kmax$ 
for  $k = 1$  to  $kmax$ 
  for  $i = 1$  to  $n$ 
     $s = 0$ 
    for  $j = 1$  to  $n$ 
      if ( $j \neq i$ )  $s = s - a_{ij}x_j$ 
    end
     $y_i = (s + b_i)/a_{ii}$ 
  end
   $x = y$ 
end

```

(숙제) 다음 주어진 초기벡터 $x^{(0)}$ 를 이용하여 Jacobi 방법으로 다음 선형연립 방정식을 풀어라.

$$\begin{bmatrix} 4 & 1 & -2 \\ -1 & 4 & 1 \\ 1 & 0 & 4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1 \\ -4 \\ 7 \end{bmatrix}, \quad x^{(0)} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

- (1) 지필 숙제로 3회 반복하여 근사해를 구하여라.
- (2) 컴퓨터 숙제로 $\|x^{(k-1)} - x^{(k)}\| < 1.0E-2$ 제한조건으로 실행하라.

Gauss-Seidel 방법 실행

$$Ax = b$$

$$(L + D + U)x = b$$

$$(D + L)x = -Ux + b$$

$$(D + L)x^{(k)} = -Ux^{(k-1)} + b, \quad k = 1, 2, 3, \dots$$

$$Dx^{(k)} = -Lx^{(k)} - Ux^{(k-1)} + b, \quad k = 1, 2, 3, \dots$$

$$\begin{bmatrix} a_{11} & 0 & 0 & \cdots & 0 \\ 0 & a_{22} & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & 0 \\ 0 & 0 & \cdots & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1^{(k)} \\ x_2^{(k)} \\ \vdots \\ x_n^{(k)} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 \\ -a_{21} & 0 & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ -a_{n1} & -a_{n2} & -a_{n3} & \cdots & 0 \end{bmatrix} \begin{bmatrix} x_1^{(k)} \\ x_2^{(k)} \\ \vdots \\ x_n^{(k)} \end{bmatrix} + \begin{bmatrix} 0 & -a_{12} & -a_{13} & \cdots & -a_{1n} \\ 0 & 0 & -a_{23} & \cdots & -a_{2n} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix} \begin{bmatrix} x_1^{(k-1)} \\ x_2^{(k-1)} \\ \vdots \\ x_n^{(k-1)} \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}$$

$$\begin{aligned}
a_{11}x_1^{(k)} &= 0 - a_{12}x_2^{(k-1)} - a_{13}x_3^{(k-1)} - a_{14}x_4^{(k-1)} \cdots - a_{1n}x_n^{(k-1)} + b_1 \\
a_{22}x_2^{(k)} &= -a_{21}x_1^{(k)} + 0 - a_{23}x_3^{(k-1)} - a_{24}x_4^{(k-1)} \cdots - a_{2n}x_n^{(k-1)} + b_2 \\
a_{33}x_3^{(k)} &= -a_{31}x_1^{(k)} - a_{32}x_2^{(k)} + 0 - a_{34}x_4^{(k-1)} \cdots - a_{3n}x_n^{(k-1)} + b_3 \\
&\dots = \dots \quad \dots \quad \dots \quad \dots \quad 0 \quad \dots \quad \dots \quad \dots \\
a_{nn}x_n^{(k)} &= -a_{n1}x_1^{(k)} - a_{n2}x_2^{(k)} - a_{n3}x_3^{(k)} \cdots \cdots 0 + b_n
\end{aligned}$$

Gauss-Seidel 알고리즘

```

입력 :  $n, A, x^{(0)}, b, kmax$ 
for  $k = 1$  to  $kmax$ 
  for  $i = 1$  to  $n$ 
     $s = 0$ 
    for  $j = 1$  to  $n$ 
      if ( $j \neq i$ )  $s = s - a_{ij}x_j$ 
    end
     $x_i = (s + b_i) / a_{ii}$ 
  end
end

```

[숙제] Jacobi 속제와 동일

제4장 행렬의 고유치 문제

고유벡터와 고유치

$n \times n$ 정방행렬 A 에 대하여 n 차원 벡터 $\mathbf{x} (\neq 0)$ 를 A 의 **고유벡터 (eigen vector)**라고 한다.

정의

$$\Leftrightarrow (\exists \text{ 상수 } \lambda, \text{ 벡터 } \mathbf{x} \neq 0 : A\mathbf{x} = \lambda\mathbf{x})$$

정의

$$\Leftrightarrow ((\mathbf{x} \neq 0, \mathbf{y} = A\mathbf{x}) \Rightarrow \mathbf{x} // \mathbf{y})$$

: 행렬로 선형변환하여도 방향을 바꾸지 않은 벡터.

여기에서 상수 λ 를 A 의 **고유치(eigen value)**라고 한다.

고유치와 고유벡터를 묶은 쌍을 (λ, \mathbf{x}) 를 A 의 **고유쌍(eigen pair)**라고 한다.

고유방정식 (특성방정식)

λ 에 관한 방정식 $A\mathbf{x} = \lambda\mathbf{x}$ 을 풀기 위하여 $A\mathbf{x} = \lambda\mathbf{x}$ 을 변형하면

$$(A - \lambda I)\mathbf{x} = 0$$

가 되고, 0 이 아닌 벡터 \mathbf{x} 가 위 식의 영이 아닌 해가 되기 위한 필요충분조건은

$$\det(A - \lambda I) = 0$$

이다. 위 방정식을 **고유방정식**(eigen equation) 또는 **특성방정식**(characteristic equation)이라고도 한다.

고유치 해법

$n \times n$ 정방행렬 A 에 대한 고유방정식은 λ 에 대한 n 차 대수 방정식이므로

중근이 없다면 최대 n 개의 고유치가 있을 수 있다. 고유치를 구하려면

n 차방정식을 풀면 되지만, 5차이상은 근의 공식이 없고, 고차방정식은 계수들의

작은 오차에도 근의 근사값의 오차가 크다. 따라서 고유치를 구하기 위하여는

반복법과 같은 다른 접근이 필요하다.

고유벡터의 비유일성과 고유공간

행렬 A 의 고유벡터 \mathbf{x} 의 스칼라 k 배인 $k\mathbf{x}$ 도 A 의 고유벡터가 된다. 왜냐하면,

$$A(k\mathbf{x}) = kA\mathbf{x} = k\lambda\mathbf{x} = \lambda(k\mathbf{x})$$

$$\text{즉 } A(k\mathbf{x}) = \lambda(k\mathbf{x})$$

이다. 어떤 고유치 λ 에 대응하는 모든 고유벡터들의 집합을 **고유공간**(eigen space)이라고 한다.

고유벡터의 일차독립성

어떤 행렬 A 의 서로 다른 고유치에 해당하는 고유벡터는 일차독립이다.

(증명) $(\lambda_1, \mathbf{x}_1), (\lambda_2, \mathbf{x}_2), \dots, (\lambda_n, \mathbf{x}_n), (\forall i, j, \lambda_i \neq \lambda_j)$ 를 고유쌍들이라고 하자.

만약 $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_j, j < n$ 가 일차독립인 최대갯수라고 가정하면

모두는 0이 아닌 상수들 $\alpha_1, \alpha_2, \dots, \alpha_j$ 가 존재하여

$$\mathbf{x}_{j+1} = \alpha_1\mathbf{x}_1 + \alpha_2\mathbf{x}_2 + \dots + \alpha_j\mathbf{x}_j \text{ --(1)}$$

로 표시된다. 양변에 행렬 A 를 곱하여

$$A\mathbf{x}_{j+1} = A(\alpha_1\mathbf{x}_1 + \alpha_2\mathbf{x}_2 + \cdots + \alpha_j\mathbf{x}_j)$$

$$\lambda_{j+1}\mathbf{x}_{j+1} = \alpha_1 A\mathbf{x}_1 + \alpha_2 A\mathbf{x}_2 + \cdots + \alpha_j A\mathbf{x}_j$$

$$\lambda_{j+1}(\alpha_1\mathbf{x}_1 + \alpha_2\mathbf{x}_2 + \cdots + \alpha_j\mathbf{x}_j) = \alpha_1\lambda_1\mathbf{x}_1 + \alpha_2\lambda_2\mathbf{x}_2 + \cdots + \alpha_j\lambda_j\mathbf{x}_j$$

$$\alpha_1(\lambda_{j+1} - \lambda_1)\mathbf{x}_1 + \alpha_2(\lambda_{j+1} - \lambda_2)\mathbf{x}_2 + \cdots + \alpha_j(\lambda_{j+1} - \lambda_j)\mathbf{x}_j = \mathbf{0}$$

$$\beta_1\mathbf{x}_1 + \beta_2\mathbf{x}_2 + \cdots + \beta_j\mathbf{x}_j = \mathbf{0} \quad \text{여기에서 } \beta_k = \alpha_k(\lambda_{j+1} - \lambda_k).$$

그런데 (1)에서 $\exists \alpha_k \neq 0$ 이므로 $\exists \beta_k = \alpha_k(\lambda_{j+1} - \lambda_k) \neq 0$ 이다.

이것은 $\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_j$ 가 일차독립이라는 사실에 모순이다. 따라서

$\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n$ 들 중에서 일차독립인 최대갯수 j 는 $j < n$ 이면 안된다.

즉 $j = n$ 이어야 한다. ■

[주의] 서로 다른 고유치의 수와 대응하는 고유벡터 수는 일치하지 않는다.

[예제 4.1]

행렬 $A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$ 의 고유치 λ_1, λ_2 는 모두 0 이지만, 대응하는 고유벡터는 $\mathbf{x} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$ 뿐이다.

행렬 $B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ 의 고유치 λ_1, λ_2 는 모두 1 이지만, 대응하는 고유벡터는 $\mathbf{x}_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$, $\mathbf{x}_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$ 이다.

[숙제] p127 #2

유사행렬(similar matrix)

$n \times n$ 정방행렬 A, B 에 대하여,

B 는 A 의 유사변환이다.

정의

$$\Leftrightarrow \exists S : B = S A S^{-1}$$

정의

$$\Leftrightarrow A, B \text{ 는 유사하다. } (A \overset{s}{\sim} B)$$

(주의) $\overset{s}{\sim}$ 는 $n \times n$ 정방행렬들에서 동치관계이다.

그러므로 B 가 A 의 유사변환이면 A 도 B 의 유사변환이다.

[정리4.2] 행렬 A 의 고유쌍이 (λ, \mathbf{x}) 이고 $B = S A S^{-1}$ 이면, $(\lambda, S\mathbf{x})$ 는 B 의 고유쌍이 된다.

(증명) $A\mathbf{x} = \lambda\mathbf{x} \Rightarrow S^{-1}BS\mathbf{x} = \lambda\mathbf{x}$
 $\Rightarrow BS\mathbf{x} = S\lambda\mathbf{x}$ 양변에 왼쪽에 S 를 곱하여
 $= \lambda S\mathbf{x}$
 $\Rightarrow B(S\mathbf{x}) = \lambda(S\mathbf{x})$

[정리4.3] Schur 정리

모든 $n \times n$ 행렬은 삼각행렬과 유사하다.

[주의] Schur 정리는 삼각행렬의 존재성만 보여주고, 삼각행렬을 계산하는 방법은 제공하지 않는다. (상, 하)삼각행렬과 대각행렬의 고유치는 그 대각원소의 값들이 된다.

벡터의 직교

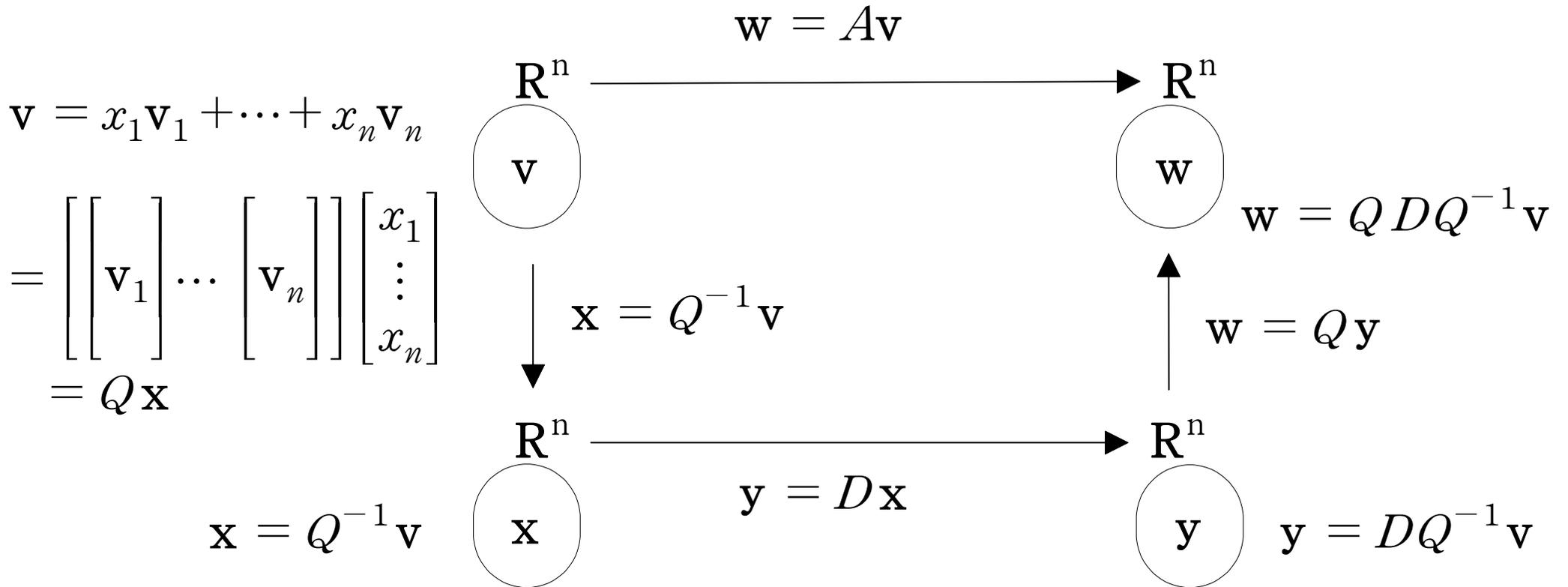
두 벡터 x, y 가 $x^T y = 0$ 를 만족하면 두 벡터는 **직교**(orthogonal)하다.

직교행렬

정방행렬 Q 가 $Q^T Q = Q Q^T = I$ 를 만족 하면 행렬 Q 를 **직교행렬** (orthogonal matrix)라고 한다.

[정리4.5] $n \times n$ 행렬 A 이 **대칭행렬**이면,

- (1) 모든 고유치 $\lambda_1, \lambda_2, \dots, \lambda_n$ 가 실수이다.
- (2) 직교하는 n 개의 단위 고유벡터 v_1, v_2, \dots, v_n 들이 존재한다.
- (3) 고유벡터 v_1, v_2, \dots, v_n 들이 기저를 이룬다. 즉 모든 벡터들은 고유벡터들의 일차결합으로 표시된다.
- (4) 대각행렬 $D = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_n)$, $Q = [v_1, v_2, \dots, v_n]$ 라고 하면 $A[v_1, v_2, \dots, v_n] = [\lambda_1 v_1, \lambda_2 v_2, \dots, \lambda_n v_n]$ 즉 $AQ = QD$ 그러므로 $A = QDQ^T$ 또는 $D = Q^T A Q$ 가 된다.



그러므로 $A = QDQ^{-1}$ 또는 $D = Q^{-1}AQ$ 이다.

(주의) Q 는 직교행렬이므로 $Q^{-1} = Q^T$ 이다.

[숙제] p132 예제 4.2 (필기 숙제)

거듭제곱법(역방법)

$n \times n$ 실수행렬 A 이 n 개의 고유치 $\lambda_1, \dots, \lambda_n$ 와 일차독립인 고유벡터 v_1, \dots, v_n 를 가지고

$$|\lambda_1| > |\lambda_2| \geq \dots \geq |\lambda_n| \geq 0$$

를 만족한다고 가정하자. 임의의 초기벡터 x_0 에 대하여, 반복법

$$x_k = A^k x_0, \quad k = 1, 2, \dots$$

을 **거듭제곱법**(power method)라고 한다. 실제 실행할 때, x_k 의 크기가 너무 크거나 작아지면, 오차 발생이 커지므로, 도중에 적당히 크기 조절을 해준다.

[정리] 거듭제곱방법에서 벡터 x_k 는 λ_1 절대치가 가장 큰 고유치 에 대응하는 벡터 v_1 에 수렴한다.

(증명)

고유벡터 v_1, \dots, v_n 들이 일차독립이므로 초기벡터 x_0 는 고유벡터들의

일차결합으로 표시될 수 있다. 즉

$$\exists c_1, \dots, c_n : \mathbf{x}_0 = c_1 \mathbf{v}_1 + \dots + c_n \mathbf{v}_n$$

이다. 그러면

$$\begin{aligned} \mathbf{x}_k &= A^k \mathbf{x}_0 \\ &= A^k (c_1 \mathbf{v}_1 + \dots + c_n \mathbf{v}_n) \\ &= c_1 A^k \mathbf{v}_1 + \dots + c_n A^k \mathbf{v}_n \\ &= c_1 \lambda_1^k \mathbf{v}_1 + \dots + c_n \lambda_n^k \mathbf{v}_n \\ &= \lambda_1^k \left(c_1 \mathbf{v}_1 + c_2 \left(\frac{\lambda_2}{\lambda_1} \right)^k \mathbf{v}_2 + \dots + c_n \left(\frac{\lambda_n}{\lambda_1} \right)^k \mathbf{v}_n \right) \end{aligned}$$

그런데, $|\lambda_1| > |\lambda_2| \geq \dots \geq |\lambda_n| \geq 0$ 로부터

$$1 > \frac{|\lambda_2|}{|\lambda_1|} \geq \frac{|\lambda_3|}{|\lambda_1|} \geq \dots \geq \frac{|\lambda_n|}{|\lambda_1|}$$

이므로 $k \rightarrow \infty$ 에 따라, $\left| \frac{\lambda_i}{\lambda_1} \right| \rightarrow 0$ ($2 \leq \forall i \leq n$) 이므로

$$\mathbf{x}_k \rightarrow C \mathbf{v}_1$$

와 같이 수렴한다. ■

거듭제곱법에서 고유치구하기

$$\begin{aligned} \mathbf{x}_{k+1} &= A \mathbf{x}_k \\ &= \lambda_1 \mathbf{x}_k \end{aligned}$$

이므로 $1 \leq i \leq n$ 인 어떤 i 에 대하여 $(\mathbf{x}_k)_i$ 는 \mathbf{x}_k 의 i 번째 성분이고,
 고유치의 근사값은 $\lambda_1 = (\mathbf{x}_{k+1})_i / (\mathbf{x}_k)_i$ 로 계산될 수 있다.

[예제4.4] 행렬 $A = \begin{bmatrix} 2 & 5 \\ -1 & -4 \end{bmatrix}$ 에 초기벡터 $\mathbf{x}_0 = \begin{bmatrix} 1 \\ 3 \end{bmatrix}$ 를 이용하여 거듭제곱
 방법을 실행하여 절대치가 가장 큰 고유치와 대응하는 고유벡터를 구하여라.

[숙제] p149 1-(1) : 거듭제곱법

역거듭제곱법

정칙행렬 A 의 고유치 $\lambda_i (\neq 0)$ 와 고유벡터 \mathbf{v}_i 에 대하여 고유방정식

$A\mathbf{v}_i = \lambda_i\mathbf{v}_i$ 의 양변에 역행렬 A^{-1} 을 곱하면

$$\mathbf{v}_i = A^{-1}(\lambda_i\mathbf{v}_i)$$

$$\mathbf{v}_i = \lambda_i A^{-1}(\mathbf{v}_i)$$

$$\frac{1}{\lambda_i}\mathbf{v}_i = A^{-1}(\mathbf{v}_i)$$

$$A^{-1}(\mathbf{v}_i) = \frac{1}{\lambda_i}\mathbf{v}_i$$

따라서 역행렬 A^{-1} 은 고유치 $\frac{1}{\lambda_i}$ 와 고유벡터 \mathbf{v}_i 를 갖는다. 즉 역행렬에 거듭제곱방법을 적용하면 절대치가 가장 작은 고유치를 구할 수 있다. 이 방법을 **역거듭제곱법**(inverse power method)라고 한다.

[예제] 역거듭제곱법으로 절대치가 가장 작은 고유치와 그 고유벡터를 구하라.

$$A = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 2 & 0 \\ 1 & 0 & 3 \end{bmatrix}$$

이동된역거듭제곱법(shifted inverse power method)

고유치가 아닌 특정상수 a 에 대하여 고유방정식 $A\mathbf{v}_i = \lambda_i\mathbf{v}_i$ 으로부터 $aI\mathbf{v}_i = a\mathbf{v}_i$ 를 양변에서 빼면

$$\begin{aligned}
A\mathbf{v}_i - aI\mathbf{v}_i &= \lambda_i\mathbf{v}_i - a\mathbf{v}_i \\
(A - aI)\mathbf{v}_i &= (\lambda_i - a)\mathbf{v}_i \\
(A - aI)^{-1}\mathbf{v}_i &= \frac{1}{\lambda_i - a}\mathbf{v}_i
\end{aligned}$$

그러므로 행렬 $(A - aI)^{-1}$ 에 거듭제곱방법을 적용하면 고유치

$$\frac{1}{\lambda_1 - a}, \frac{1}{\lambda_2 - a}, \dots, \frac{1}{\lambda_n - a}$$

중에서 절대치가 가장 큰 고유치 $\frac{1}{\lambda_k - a}$ 와 그에 대응하는 고유벡터 \mathbf{v}_k 를

구할 수 있다. 그것으로부터 행렬 A 의 고유치 중에서 특정상수 a 에 가장

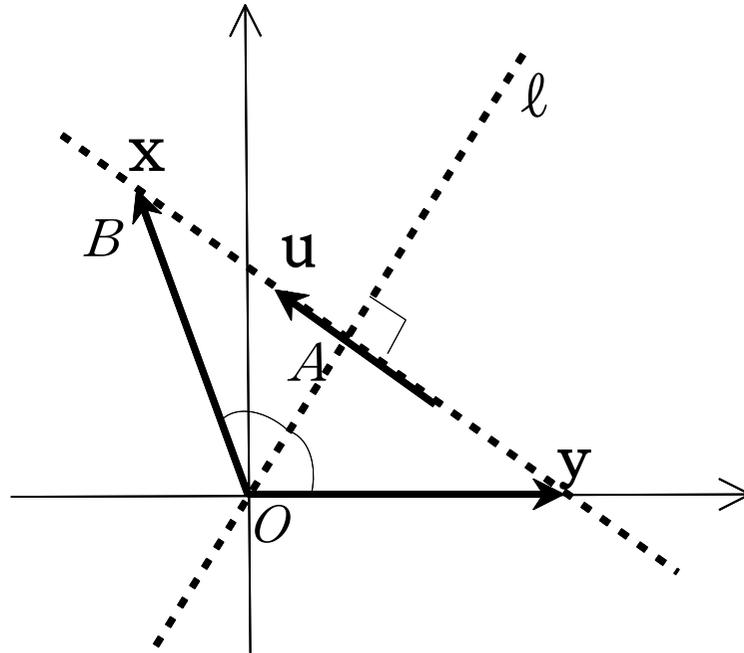
가까운 고유치 λ_k 와 그에 대응하는 고유벡터 \mathbf{v}_k 를 구할 수 있다. 이 방법을

이동된역거듭제곱법(shifted inverse power method)라고 한다.

[숙제] p148 예제4.5 : 이동된역거듭제곱법 ($a = 7.0$)

Householder 변환

크기가 같은 두 벡터 x 와 y 의 중점을 잇는 선분의 수직이등분선 ℓ 을 기준선으로 벡터들을 대칭(거울)변환하는 것을 **Householder 변환**이라고 한다. 대칭변환이기 때문에 한번 더 변환하면 원래로 돌아온다. 즉 $HH = I$ 가 된다.



선분 \overline{AB} 의 길이는 단위벡터 $u = \frac{x-y}{\|x-y\|}$ 를 이용하여 내적 $x \cdot u$ 로

주어지고, 벡터 $(x-y)$ 는 $(x-y) = 2(x \cdot u)u$ 로 주어진다.

$$\begin{aligned}
\text{그러므로 } y &= \mathbf{x} - 2(\mathbf{x} \cdot \mathbf{u})\mathbf{u} \\
&= \mathbf{x} - 2(\mathbf{u} \cdot \mathbf{x})\mathbf{u} = \mathbf{x} - 2\mathbf{u}(\mathbf{u} \cdot \mathbf{x}) \\
&= \mathbf{x} - 2\mathbf{u}(\mathbf{u}^T \mathbf{x}) = \mathbf{x} - 2(\mathbf{u}\mathbf{u}^T)\mathbf{x} \\
&= (I - 2\mathbf{u}\mathbf{u}^T)\mathbf{x} \stackrel{\text{정의}}{=} H\mathbf{x}
\end{aligned}$$

여기에서 행렬 $H = (I - 2\mathbf{u}\mathbf{u}^T)$ 를 **Householder 행렬**이라고 한다. 이 행렬을 이용하면 어떤 벡터를 원하는 좌표축이나 좌표(초)평면으로 대칭변환하여, 최소 1 개를 제외한 나머지 성분들이 0 (제로)가 되도록 변환할 수 있다. 그런데 위에서 벡터 y 의 종점을, 양의 좌표축 위에만 아니라, 음의 좌표축 위에도 선택할 수 있는데, 컴퓨터 계산 중 뿔셈에 의한 오차를 줄이기 위해서는, \mathbf{x} 의 종점에서 먼 방향의 좌표축 위에 위치하도록 y 를 선택하는 것이 좋다.

[예제4.6] 벡터 $\mathbf{x} = [4, 1, -2, 2]^T$ 를 길이가 같고, 3번과 4번째 성분이 0 이 되는 벡터로 변환하는 Householder행렬을 찾아라.

(풀이) 벡터 \mathbf{x} 와 길이가 같고, 3번째와 4번째 성분이 0이 되는 벡터 \mathbf{y} 를

$\mathbf{y} = [4, a, 0, 0]^T$ 라 하면,

$$\begin{aligned}\|\mathbf{x}\| = \|\mathbf{y}\| &\Rightarrow \sqrt{4^2 + 1^2 + (-2)^2 + 2^2} = \sqrt{4^2 + a^2} \\ &\Rightarrow a = \pm \sqrt{1^2 + (-2)^2 + 2^2} \\ &= \pm 3\end{aligned}$$

\mathbf{y} 의 2번째 성분 a 의 부호를 \mathbf{x} 의 2번째 성분 1의 부호와 반대 쪽을 택하면 $a = -3$ 이 된다. 그러므로

$$\begin{aligned}(\mathbf{x} - \mathbf{y}) &= [4, 1, -2, 2]^T - [4, -3, 0, 0]^T \\ &= [0, 4, -2, 2]^T \\ \Rightarrow \|\mathbf{x} - \mathbf{y}\| &= \sqrt{24} \\ \Rightarrow \mathbf{u} &= \frac{1}{\sqrt{24}} [0, 4, -2, 2]^T\end{aligned}$$

그러므로 Householder 행렬은

$$H = (I - 2uu^T) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -\frac{1}{3} & \frac{2}{3} & -\frac{2}{3} \\ 0 & \frac{2}{3} & \frac{2}{3} & \frac{1}{3} \\ 0 & -\frac{2}{3} & \frac{1}{3} & \frac{2}{3} \end{bmatrix}$$

이다. ■

[숙제] 예제4.6에서 첫 번 성분만 제외하고 모두 0 (제로)가 되는 벡터, 즉 $y = [a, 0, 0, 0]$ 로 변환하는 Householder 행렬을 구하라.

제2고유치 구하기(수축방법)

행렬 A 의 고유치 λ 에 대응하는 고유벡터 $v = [v_1, v_2, \dots, v_n]$ 를 크기가 같고, 첫 번 성분을 제외한 나머지 모든 성분이 모두 0 (제로)가 되고, 첫 성분이 v_1 와 반대 부호를 갖는 벡터로 변환하는 Householder 행렬을 H 라고 하자.

즉

$$Hv = a e_1 \quad (a = -\text{sign}(v_1) \|v\|) \quad \text{--(1)}$$

이다. 고유 방정식 $Av = \lambda v$ 의 양변에 행렬 H 를 곱하여

$$\begin{aligned} HAv &= H\lambda v \\ &= \lambda Hv \text{ by 선형성} \\ &= \lambda a e_1 \text{ by (1)} \\ &= a \lambda e_1 \end{aligned}$$

$$\begin{aligned} \Rightarrow a \lambda e_1 &= HAv \\ &= HA(HH)v \text{ by } HH = I \\ &= HAH(Hv) \text{ by 결합법칙} \\ &= HAH(a e_1) \text{ by (1)} \\ &= a HAH(e_1) \text{ by 선형성} \end{aligned}$$

$$\begin{aligned} \Rightarrow (HAH)e_1 &= \lambda e_1 \text{ by } a \neq 0 \\ &\text{(HAH의 고유(치, 벡터)가 } \lambda, e_1 \text{ 이 됨)} \end{aligned}$$

$$\Rightarrow (HAH) \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} = \begin{bmatrix} \lambda \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

그러므로 위 우변은 행렬 HAH 의 첫 열벡터이다. 따라서

$$HAH = \begin{bmatrix} \lambda & r_1 & \cdots & r_{n-1} \\ 0 & \begin{bmatrix} & \\ & B \end{bmatrix} \\ \vdots & \\ 0 & \end{bmatrix}$$

행렬 HAH 는 A 와 유사행렬이므로 HAH 의 고유치들은 A 의 고유치들과 같고, 행렬 B 는 λ 를 제외한 A 의 모든 고유치들을 갖고 있다. (정리4.7참조) 행렬 B 를 A 의 수축행렬이라고 한다. 수축행렬 B 에 거듭제곱법을 행하면 A 의 제2의 고유치를 구할 수 있다. ■

제2고유벡터 구하기

수축행렬 B 에 거듭제곱법을 행하여 얻은 고유치 λ_1 과 고유벡터 w 로부터 행렬 HAH 의 고유벡터 x 를 거쳐, 행렬 A 의 고유벡터 v_1 를 구해보자.

우선 고유치 λ_1 에 대응하는 행렬 HAH 의 고유벡터를 x 라고 하자. 즉

$$(HAH)x = \lambda_1 x$$

$$\begin{bmatrix} \lambda & r_1 & \cdots & r_{n-1} \\ 0 & \begin{bmatrix} & \\ & B \end{bmatrix} \\ \vdots & & & \\ 0 & & & \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \lambda_1 \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \quad \text{--(1)}$$

위 식의 2행부터 n 행까지 방정식을 모으면 다음을 얻는다.

$$\begin{bmatrix} & \\ & B \end{bmatrix} \begin{bmatrix} x_2 \\ \vdots \\ x_n \end{bmatrix} = \lambda_1 \begin{bmatrix} x_2 \\ \vdots \\ x_n \end{bmatrix}$$

여기에서, B 의 고유치 λ_1 에 대응하는 고유벡터가 $\mathbf{w} = [x_2, x_3, \dots, x_n]^T$ 임을 알 수 있다. 즉 HAH 행렬의 고유벡터의 성분 x_2, x_3, \dots, x_n 은 B 에 거듭제곱법을 실시하여 얻는 고유벡터 \mathbf{w} 로부터 얻을 수 있다. 나머지 성분 x_1 을 구하자면, 식 (1)에서 1행만 모으면

$$\begin{aligned} \lambda x_1 + \mathbf{r}^T \mathbf{w} &= \lambda_1 x_1 \\ \Rightarrow x_1 &= \frac{\mathbf{r}^T \mathbf{w}}{\lambda_1 - \lambda}, \quad \mathbf{r} = [r_1, r_2, \dots, r_{n-1}]^T \end{aligned}$$

이렇게 구해진 HAH 의 고유벡터 \mathbf{x} 를 이용하여, 이제 행렬 A 의 고유벡터 \mathbf{v}_1 을 구해보자.

$$\begin{aligned} (HAH)\mathbf{x} &= \lambda_1 \mathbf{x} \\ \Rightarrow H(HAH\mathbf{x}) &= H(\lambda_1 \mathbf{x}) \quad \text{by 양변에 } H \text{ 곱하기} \\ \Rightarrow AH\mathbf{x} &= \lambda_1 H\mathbf{x} \quad \text{by } HH=I \text{ 와 선형성} \end{aligned}$$

$$\Rightarrow A(H\mathbf{x}) = \lambda_1(H\mathbf{x}) \quad \text{by 결합법칙}$$

$$\Rightarrow A\mathbf{v}_1 = \lambda_1\mathbf{v}_1 \quad \text{with } \mathbf{v}_1 = H\mathbf{x}$$

그러므로 행렬 A 의 고유벡터 \mathbf{v}_1 는 $\mathbf{v}_1 = H\mathbf{x}$ 로 계산할 수 있다. ■

[예제4.7] 행렬 A 가 고유치 $\lambda = 4$ 와 고유벡터 $\mathbf{v} = [1, -2, 2]^T$ 을 갖고 있을 때 수축방법을 이용하여 제2의 고유치 λ_1 과 대응하는 고유벡터 \mathbf{v}_1 를 구하여라. (컴퓨터 사용)

$$A = \begin{bmatrix} 4 & 1 & 1 \\ -2 & 3 & 0 \\ 2 & 0 & 3 \end{bmatrix}$$

[숙제] p161 #9 (컴퓨터 사용)