

初めてさんの J 言語 (PART VI)

統計数理研究所 (名誉教授) 鈴木義一郎

【論理演算の概要】

【 (「+.」 Or) . (「+;」 Not-Or) . (「\*.」 And) . (「\*:」 Not-And) : 両側形】

a=:0 0 1 1	a +.b	a +:b	a *.b	a *: b
b=:0 1 0 1	0 1 1 1	1 0 0 0	0 0 0 1	1 1 1 0
基本的な 4 つの論理演算 (論理和と論理積)				

【論理和 (Or +.) : ブール代数】

a +.b	a (0:<+)b	a (0:`1:@.*@+)b	左右の引数が共に「0」のときだけ「0」 を出力し、でなければ「1」を出力。
0 1 1 1	0 1 1 1	0 1 1 1	

【否定論理和 (Not-Or +:) : ブール代数】

a +:b	a (0:+=)b	a (1:`0:@.*@+)b	左右の引数が共に「0」のときだけ「1」 を出力し、でなければ「0」を出力。
1 0 0 0	1 0 0 0	1 0 0 0	

【論理積 (And \*.) : ブール代数】

a *.b	a (1:=*)b	a (0:`1:@.*@*)b	左右の引数が共に「1」のときだけ「1」 を出力し、でなければ「0」を出力。
0 0 0 1	0 0 0 1	0 0 0 1	

【否定論理積 (Not-And \*:.) : ブール代数】

a *:b	a (0:=*)b	a (1:`0:@.*@*)b	左右の引数が共に「1」のときだけ「0」 を出力し、でなければ「1」を出力。
1 1 1 0	1 1 1 0	1 1 1 0	

【 (「-:」 Match) : (「=」 (Equal) ) : (「~:」 Not equal) : 論理演算】

20 30 40 = 40 30 20 0 1 0	20 30 40 ~: 40 30 20 1 0 1	「=(~:)」は等しければ 1 (0), 等しくなければ 0 (1)
20 30 40 -: 40 30 20 0	20 30 40 -: 20 30 40 1	「-: Match」は「形」まで含めてソ ックリ同じときだけ「1」
左右の要素毎に等しいときに「0」等しくないときに「1」を出力する。「~:」は 'not equal'		

**【 [b.] Boolean . Basic】**

<code>+./~ 1 0</code> 1 1 1 0	<code>]S=:7 b./~ 1 0</code> 1 1 1 0	<code>]T=: :S</code> 1 1 1 0	<code>]t=: . ,T</code> 0 1 1 1	<code>#. t</code> 7
<code>+:/~ 1 0</code> 0 0 0 1	<code>]SN=:8 b./~ 1</code> 0 0 0 0 1	<code>]U=: :SN</code> 0 0 0 1	<code>]u=: . ,U</code> 1 0 0 0	<code>#. u</code> 8
<code>*./~ 1 0</code> 1 0 0 0	<code>]P=:1 b./~ 1 0</code> 1 0 0 0	<code>]Q=: :P</code> 1 0 0 0	<code>]q=: . ,Q</code> 0 0 0 1	<code>#. q</code> 1
<code>*:/~ 1 0</code> 0 1 1 1	<code>]PN=:14 b./~ 1</code> 0 0 1 1 1	<code>]R=: :PN</code> 0 1 1 1	<code>]r=: . ,R</code> 1 1 1 0	<code>#. r</code> 14

<code>boolean=:3 :'#. ., :y b.~/~1</code> 0'		<code>boolean"0 (7 8 1 14)</code> 7 8 1 14	
<code>1 0 +./ 0 1</code> 1 1 0 1	<code>11 b./~1 0</code> 1 1 0 1	<code>1 0 +:/ 0 1</code> 0 0 1 0	<code>4 b./~1 0</code> 0 0 1 0
<code>1 0 *./ 0 1</code> 0 1 0 0	<code>2 b./~1 0</code> 0 1 0 0	<code>1 0 */: 0 1</code> 1 0 1 1	<code>13 b./~1 0</code> 1 0 1 1
<code>boolean"0 (11 4 2 13)</code> 11 4 2 13		<code>boolean"0 &gt;:i.15</code> 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15	
<code>^ b._1</code> ^.	<code>^ b.0</code> 0 0 0	<code>^ b.1</code> \$&1@({).@\$)	

**【negative indexing】**

`(_1 b./~;_3 b./~;_15 b./~)d`

1 1	1 1	0
1 1	0 1	0
		0
		1

`negativ=:4 :'(16+x)b./~y'`

<code>_1 negativ d</code> 1 1 1 1	<code>_3 negativ d</code> 1 1 0 1	<code>_15 negativ d</code> 0 0 0 1
---	---	--

bit 12345 000000000000000000001100000011100 1 bit 67890 00000000000000001000010010011001 0		bit=:3 : '((32\$2)#:y){0 1'
]s=:_17{bit 12345 00011000000111001	s*.t 000000000000110000	#. (h#:b)*.h#:a 48
]t=:_17{bit 67890 10000100100110010	#. s *. t 48	a (17 b.) b 48
bit_and=:4 : '#.(bit y)*. bit x'		bit_or=:4 : '#.(bit y)< bit x'
s > t 00011000000001001	#. s > t 12297	12345 bit_or 67890 12297
12345 (18 b.)67890 12297	_12345 (18 b.)67890 _80187	_12345 (23 b.)67890 _12297

12345 (23 b.)67890 80187	12345(4 : '#.(bit y)+.bit x')67890 80187	
12345 (20 b.)67890 67842	#. s < t 67842	

20 b./~ i.10 0123456789 0022446688 0101454589 0000444488 0123012389 0022002288 0101010189 0000000088 0123456701 0022446600	0(20 b.)i.10 0123456789 1(20 b.)i.10 0022446688 2(20 b.)i.10 0101454589 3(20 b.)i.10 0000444488 4(20 b.)i.10 0123012389	5(20 b.)i.10 0022002288 6(20 b.)i.10 0101010189 7(20 b.)i.10 0000000088 8(20 b.)i.10 0123456701 9(20 b.)i.10 0022446600
--	--	--

<pre> b20_1=:3 :',(&lt;.\^"1)5 2\$y' b20_2=:3 :',(&lt;.\^"2)2 2 2\$y),_2{.y' b20_3=:3 :',(&lt;.\^"1)2 4\$y),&lt;.\^_2{.y' b20_4=:3 :',(&lt;.\^2 4\$y),_2{.y' b20_5=:3 :',(&lt;.\^"1(&lt;.\^"3)2 2 2\$y),&lt;.\^_2{.y' b20_6=:3 :',(&lt;.\^4 2\$y),_2{.y' </pre>		
<pre> 1(20 b.)i.10 0 0 2 2 4 4 6 6 8 8 2(20 b.)i.10 0 1 0 1 4 5 4 5 8 9 3(20 b.)i.10 0 0 0 0 4 4 4 4 8 8 4(20 b.)i.10 0 1 2 3 0 1 2 3 8 9 5(20 b.)i.10 0 0 2 2 0 0 2 2 8 8 6(20 b.)i.10 0 1 0 1 0 1 0 1 8 9 </pre>	<pre> 1 (20 b.)i.10 0 0 2 2 4 4 6 6 8 8 2 (20 b.)i.10 0 1 0 1 4 5 4 5 8 9 3 (20 b.)i.10 0 0 0 0 4 4 4 4 8 8 4 (20 b.)i.10 0 1 2 3 0 1 2 3 8 9 5 (20 b.)i.10 0 0 2 2 0 0 2 2 8 8 6 (20 b.)i.10 0 1 0 1 0 1 0 1 8 9 </pre>	

<pre> 23 b.\ 2^i.10 1 3 7 15 31 63 127 255 511 1023 _1+2^}.i.11 1 3 7 15 31 63 127 255 511 1023 </pre>	<pre> 1 (b32=:4 :'+:^:x)y') 12345 24690 1 2 3 (b32 ぢ"0) 12345 24690 49380 98760 </pre>
--	--

<pre> b33=:4 :',(&lt;.&amp;-:)^:(-x)y' _5 b33 12345 385 (s=:_5 _4 _3 _2 _1) (33 b."0) 12345 385 771 1543 3086 6172 s (b33"0) 12345 385 771 1543 3086 6172 </pre>	<pre> half=:&lt;.&amp;-: half 12345 6172 half^:2(12345) 3086 half^:3(12345) 1543 half^:4(12345) 771 half^:5(12345) 385 </pre>
--	---

【 ( 「 \_ 」 Negative Sign/Infinity ) . ( 「 \_ . 」 Indefinite ) . ( 「 \_ : 」 Indefinity ) . ( 「 ” \_ 」 ) 】

$2 \%$ 0	$2 \%$ 0	$- + -$ -	「_」は $\infty$ (無限大)の名詞 「_」は $-\infty$ (マイナス無限大)
$- - -$ -	$3 + -$ -	$2 * -$ -	「_ .」は 不定形(indeterminate)の名詞
$- : ' '$ -	$( ) \% _ : ) 2$ 0		「_ :」は無限大( $\infty$ )を出力する動詞
$( ) + 1 " _ ) i . 3$ 1 2 3	$( ) + 1 : ) i . 3$ 1 2 3		「 ” _ 」は数値につけて “動詞化” する副詞。

【複数組の関数を同時に定義するワザ】

<pre>stat_reg=:3 :0 regb=:[%.1:,.] regp=:(1:,.)+/ .*regb regq=:[:+/[*:[-regp regcd=:100"*_1:-regq%[:+/[*:(-+/%#)@[ mat=:[:%.( :+/ .**)@(1:,.] resvar=:regq%[:-/[:\$1:,.] regt=:regb%[:%:resvar*[:(&lt;1 0)&amp; :mat@] m11=:&gt;:@^.@((o.2)"*_regq%#@[)*#@[%_2: regaic=:+:@(1:+#@(1:,:)))-2:*m11 'program set of regression model' )</pre>	<p>左のように、回帰分析に必要な関数群を「大局定義」で行えば、</p> <pre>stat_reg'' program set of regression model   2 5 \$ namelist 3</pre> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>mat</td><td>m11</td><td>regaic</td><td>regb</td><td>regcd</td></tr> <tr> <td>regp</td><td>regq</td><td>regt</td><td>resvar</td><td>stat_reg</td></tr> </table> <p>のように「stat_reg」という関数を実行することにより、必要に応じて複数個の関数をまとめて定義することができる。</p> <p>「namelist 3」は定義されている(代)動詞のセットをボックスの形で出力する。</p>	mat	m11	regaic	regb	regcd	regp	regq	regt	resvar	stat_reg
mat	m11	regaic	regb	regcd							
regp	regq	regt	resvar	stat_reg							

【観測地点の標高と 07 年 8 月の平均気温】

観測地点	標高 (AL : m)	平均気温 (TM : °C)
A (甲 府)	273	27.7
B (勝 沼)	394	26.7
C (小 関)	552	24.9
D (河口湖)	860	23.3
E (山 中)	992	21.7
F (富士山頂)	3776	6.4

AL=:273 394 552 860 992 3776 TM=:27.7 26.7 24.9 23.3 21.7 6.4	標高が高くなると気温が下がると考えられる。 このデータで、6合目(2500m)の気温を推定	
《1次式のモデル》 ]b1=:TM regb AL	《2次式のモデル》 ]b2=:TM regb AL^/1 2	《コメント》 回帰係数が算出されている。

28.5486 _0.00592836	29.9715 _0.00882574 6.84209e_7	
TM regcd AL 99.3547	TM regcd AL^/1 2 99.8732	決定係数の値(%表示)
TM regaic AL 16.3951	TM regaic AL^/1 2 6.63166	情報量規準(AIC)の値で2次式のモデルを選択
+/b1*1 2500 13.7277	+/b2*1,2500^/1 2 12.1835	6合目での予想気温

年次	1950	55	60	65	70	75	80	85	90	95
男子自殺者数(百人)	98	138	115	83	88	117	128	154	123	142
完全失業率(%)	1.2	2.5	1.7	1.2	1.1	1.9	2.0	2.6	2.1	3.1

L=:1.2 2.5 1.7 1.2 1.1 1.9 2.0 2.6 2.1 3.1 S=:98 138 115 83 88 117 128 154 123 142	完全失業率と男子自殺者数のデータを(L)(S)に入力している。	
《1次式のモデル》	《2次式のモデル》	《コメント》
]b1=:S regb L 54.878 32.8464 S regaic L 75.8121	]b2=:S regb L^/1 2 7.82459 84.6602 _12.8461 S regaic L^/1 2 70.7389	回帰係数と情報量規準(AIC)を求めている。「AIC」の値は2次モデルが小さいが、大きいところを予測するのは危険
+/b1*2 2\$1 1 4 5 186.264 219.11	失業率が4%、5%になった場合の男子自殺者数を予測していて、それぞれ18626人、21911人になるものと予想される。	

mean=:3 :'+/y)%#y' [ sum=:3 :'+/y' MEAN=:+/%# [ SUM=:+/ var=:3 :'mean *:dev y.' [ dev=:3 :'y-mean' VAR=:[:MEAN[:*:DEV=: -MEAN sdev=:3 :'%:var y' [ mdev=:3 :'mean dev y.' SDEV=:[:%:VAR [ MDEV=:[:MEAN[: DEV classify=:3 :'t, :+/"1(t=..:/~y)=/y' meanc=:3 :'+/*y)%+/{:y' varc=:3 :'mean({:y)#*:({:y)-meanc y' rsk1=:[:+ / sub=:+/\&.   rsk2=:[:+ /[:}.sub^:2	NB. 平均と総和(Explicit) NB. 平均と総和(Tacit) NB. 偏差と分散(Explicit) NB. 偏差と分散(Tacit) NB. 平均偏差と標準誤差 NB. 平均偏差と標準誤差 NB. データを分類する関数 NB. 分類されたデータの平均 NB. 分類されたデータの分散 NB. 第1累積度数の和 NB. 第2累積度数の和 NB. 累積度数法による平均
--	--

```

meanr=:4 : '{.x)+({:x)*<:(rsk1%+/)y'
varr=:4 : '{(*:x)*((+:rsk2 y)+s-(*:s=.rsk1
y)%n)%n=.+/y'
each=:&>.
div=:3 : '(0=t|y)#t=:1+i.y'
gcd=:4 : '{:/:~(a e. div y)#a=.div x'
lcm=:4 : '{.(b
e.a=.x*m)#b=.y*m=.1+i.>.x<.y'

```

NB. 累積度数法による分散

NB. イーチ (副詞)

NB. 約数を全て出力

NB. 最大公約数 (GCD)

NB. 最小公倍数 (LCM)