A numeric recipe for Econometrics
part 1

MShimura
JCD02773@nifty.com

2006 年 2 月 28 日

目次

第 1 章

データの詳覧

1.1 データの構成

本書で扱うデータは

のように $X_1X_2X_3 \cdots Y$ の様に縦にデータを並べ、最後の列に Y(被説明変数) を置く型が中心である。データは EXCEL からのインポートや CSV ファイル、データベースとの出し入れを行う。EXCEL,CSV の取り扱いは APPENDIX に記載した。

KD で次のように表示される。名前を *KD* と付ける。	KX　ある果実の直径 *cm* KY 水分含有率 (%) のデータ。 NB. Kanaya p.7

`KD`
 	
`5.6 30`
`5.8 26`
`  6 33`
`6.2 31`
`6.4 33`
`6.4 35`
`6.4 37`
`6.6 36`
`6.8 33`
	`KX=:5.6 5.8 6.0 6.2 6.4 6.4 6.4 6.6 6.8`
	`KY=: 30 26 33 31 33 35 37 36 33`
	`KD=: KX,.KY`

[1]と言う

```
  +/KD    NB. sum
56.2 294

$   KD     NB. size  of matrix
9 2
#   KD     NB. number
9
```

1.1.1　図

散布図はプログラムしなくても簡単に描ける。

```
  load 'plot'
 'symbol' plot {|: KD
   pd 'eps temp\kanaya_0.eps'
```

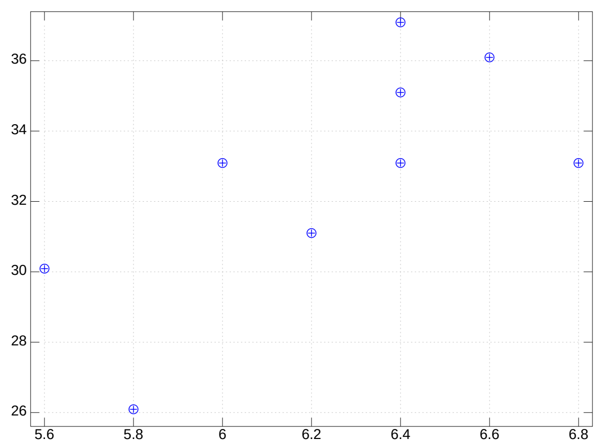

図 1.1　散布図

[1] J では名詞

J 言語ノート

`+/`	$\sum$	合計
`{`	*catalogue*	*box* で囲う
`|:`	*transpose*	縦横変換
NB.	*notebote*	注釈
`$`	*Shape* · 形	マトリクスの型を見る (データのサイズ)
`#`	*Tally* · 小枝	*n* を計る
plot		グラフを描く
simbol		グラフを点で表現する

1.2　平均と比較

1.2.1　平均

平均には日常用いている算術平均の他にも様々な方法がある。

項目	数式	Script	Example

算術平均	$\frac{1}{n}\sum_{i=1}^{n} X_i$	`am=: +/ % #`
		`am2=: # %~ +/`
			`]a=: >: i.10`
			`1 2 3 4 5 6 7 8 9 10`
			`am a`
			`5.5`

幾何平均	$\sqrt[n]{\Pi_{i=1}^{n} X_i}$	`gm=: # %: */`	gm a 4.52873
調和平均	$\frac{n}{\sum_{i=1}^{n}\frac{1}{X_i}}$	`hm=: am &.(%"_)`	逆数の平均値の逆数　比率の平均に用いる hm a 3.41417
common mean		`cm=:[: {.(am,gm)^:_`	cm a 5.00257

J 言語ノート

i.		0 からの順序数を指定個数打ち出す
`>:`		1 を加える
`*`	×	かけ算
%	÷	割り算
`*:`	*square*	2 乗
`%:`	*square – root*	平方根
`{.`	*From*	最初のスカラ 行 列を取り出す
`_`	*infinity*	無限大
`~`	*child*	計算順序を逆にする
`"1`	*Rank*	マトリクスの演算方向
`^:`	*power*	一種のループ
`&.`	*under*	接続詞

1.2.2　各種の平均の応用

1.2.3　四半期データの前期比を年率換算

４半期データの年率換算	$((\frac{y_t}{y_{t-1}})^4 - 1) \times 100$

Working Example **四半期伸び率の年率換算**

```
 qtr_grow 1 1.02      NB. 1 (space) 1.02
8.24322
```

４半期で.1.0 から 1.02 への伸びは年率換算 8.2%

Script

```
qtr_grow=: 3 : '100 * <: ^&4 %/|. y.'
NB. e.g. u. 1 1.02
NB. exchange rate of growth from quarter to year
```

経過と説明

1. 前後 (新旧) を入れ替えて (rotate `|.`) 割る

```
 %/|.  1  1.02
1.02
```

2. 1 を引く (<: は 1 を引く)

```
<: ^&4 %/|.  1  1.02
0.0824322
```

3. 100 倍する ($*$ *times*)

```
100 * <: ^&4 %/|.  1  1.02
8.24322
```

1.2.4　年平均成長率を求める。

５年、１０年などの単位で、平均成長率を求める。
単位は % になっている。

年平均成長率	$\sqrt[n]{\frac{x_{t+n}}{x_t}} - 1 \times 100$

Working Example

```
 5  grow_ave  504827 539160     NB. GDP 1995/2000
 1.32463       (1.32%)

  10  grow_ave  469567 539160   NB. GDP 1990/2000
1.39161        (1.39%)
```

Script

```
grow_ave=: 4 : '100  * <: (x. %: %/ |. y.)'
```

経過と説明

1. 年次を逆にして比を求める。

```
%/ |.  504827 539160     NB. GDP 1995/2000
1.06801
```

2. 5 乗根を求め

```
5 %:  %/ |.  504827 539160     NB. GDP 1995/2000
1.01325
```

3. 100 倍する

```
100* 5 %:  %/ |.  504827 539160     NB. GDP 1995/2000
101.325
```

1.2.5　成長率の平均＝幾何平均

成長率のデータしか手許にない場合に、成長率の平均が求められる。

Working Example **成長率の平均**

```
a=:1.25 1.4 1.07143 1.06667 1.125

   gm a
1.17608
```

1.2.6　調和平均

Working Example **平均速度**

３６０ km の区間の最初の１２０ km を時速３０ km で、次の１２０ km を時速４０ km で、最後の１２０ km を６０ km で走った場合の平均速度を求める

```
hm 30 40 60
40

3 % +/ 1r30 1r40 1r60
40
```

J 言語ノート

y.		右引数
`|.`	*Rotate*	ベクトルやマトリクスを指定方向に回転させる
`/`	*Insert*	計算記号 (演算子) を数列の間に挿入する
`<:`	*Decriment*	1 引く
&	接続詞	数字と演算子を繋いで動詞 (関数) を作る
r	分数	1*r*3 は $\frac{1}{3}$

1.3　指数

sample data　 is $ijs/data/s\_data.ijs/L\_1$

1.3.1　ラスパイレス指数

指数は次により求められる。

2000 年を 100 とした指数=$\frac{y_t}{y_{00}}$

幾つかの品目データを一つの指数にする方法に、ラスパイレス、パーシェ、フィッシャーの 3 の方法がありウエイトの置き方で区別される。ラスパイレス指数は自己のデータのウエイトで計算できるので便利である。

Laspiress	$\frac{\sum p_t q_0}{\sum p_0 q_0}$
Parshe	$\frac{\sum p_t q_t}{\sum p_0 q_t}$
Fischer	$\sqrt{\frac{\sum p_t q_0}{\sum p_0 q_0} \times \frac{\sum p_t q_t}{\sum p_0 q_0}}$

財 INDEX	基準時 Price P0	 quantity Q0	比較時 price P1	 quantity Q1
第 1 財	75	50	85	48
第 2 財	60	70	55	72

```
   D0
75 50 85 48
60 70 55 72

  Las:     Par:     Fis:
  101.887  101.515  101.701


   T1
+------+-----+
| 75 50|60 70| NB. 基準時
| 85 48|55 72| NB. 第 1 期
| 95 46|52 75| NB. 第 2 期
|105 44|48 80| NB. 第 3 期
+------+-----+

   lsp_chain0 T1
```

```
1 1.01887 1.05535 1.08302  NB. 基準 第 1 期 第 2 期 第 3 期

   par_chain0 T1
1 1.01515 1.04025 1.04444

   fis_chain0 T1
1 1.01701 1.04777 1.06356
```

(データの出典　日本銀行調査統計局編　計量経済分析の基礎と応用　ｐ２１)

1.3.2　Script

```
lsp_chain0=: 3 : 0
TMP1=:({. "1 L:0 ,.y. ) */ L:0 }.@{. L:0 ,. y.
TMP2=:(;{. TMP1) + ;{: TMP1
; TMP2 %/ {. TMP2
)

par_chain0=: 3 : 0
TMP10=:(}."1 L:0 y. ) */ L:0  {.@{. L:0  y.
TMP11=:;(;{. TMP10) + ;{: TMP10
TMP20=: */"1 L:0 y.
TMP21=:; (;{. TMP20) + ;{: TMP20
,TMP21 %  TMP11
)

fis_chain0=: 3 : '%: (par_chain0 y.)* lsp_chain0 y.'
```

J 言語ノート

n { (or n&{)	*Take*	指定したスカラや行、列の取り出し
'Las'	*Character*	文字列
12.5 ":	*Format*	文字化して出力する (12 桁内少数 5 桁)

1.3.3　関数一覧

算術平均	*am am*2	*am i*.10
幾何平均	*gm*	*gmi*.10
調和平均	*hm*	*hmi*.10

四半期伸び率の年率変換 年平均成長率	*qtr_grow* *grow_ave*	*qtr_grow* 1 1.02 5 *grow_ave* 504827 539160
ラスパイレス指数	*lsp*	*lsp y*.

1.4　データの詳覧

1.4.1　標準化

標準化とは、データを平均０、分散１に変換することであり、単位が異なるデータの回帰や図示に用いる。

標準化	$\dfrac{x-\bar{x}}{\sqrt{\frac{1}{n}\sum(x-\bar{x})^2}}$	データを平均０、分散１に変換する

KX　ある果実の直径 *cm*
KY 水分含有率 (%) のデータ。
	`KD`
	`5.6 30`
	`5.8 26`
	`  6 33`
	`6.2 31`
	`6.4 33`
	`6.4 35`
	`6.4 37`
	`6.6 36`
	`6.8 33`
		`stand KD`
		`  _1.7781 _0.843274`
		` _1.22628  _2.10819`
		`_0.674453  0.105409`
		`_0.122628 _0.527046`
		` 0.429198  0.105409`
		` 0.429198  0.737865`
		` 0.429198   1.37032`
		` 0.981023   1.05409`
		`  1.53285  0.105409`

単位が異なるものはうまく一枚のグラフには表せないことが多いが標準化すると単位が無くなり、同一のグラフに表すことができる。
（データの出典　金谷　「これならわかる応用数学教室」）

```
  plot |: stand  KD
 pd 'eps temp\kanaya_02.eps'
```

Script

```
stand=: dev % "1 sd

dev=: -"1(+/ % #)
sd=. %:@var
var=: # %~ ([:+/[: *: dev)
```

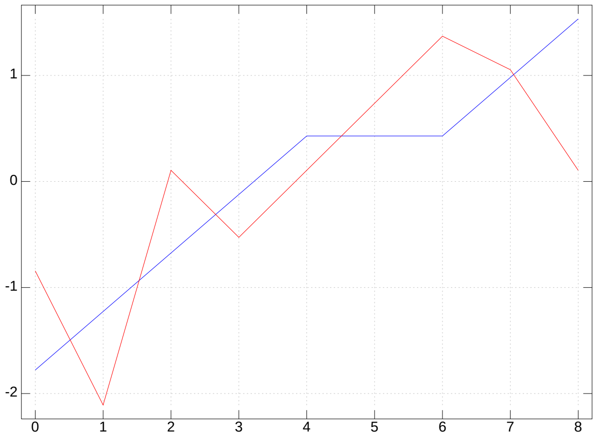

図 1.2　果実の直径 (直線) と水分含有率

J 言語ノート

`*:`	*sqxuare*	2 乗 (`^2` と同じ)
`%:`	$sqxuare_root$	2 乗根 ()
`-"1(+/%#)`	$x-\overline{x}$	残差
`[:`	*Cap*	関数を記述順に実行する指定

1.4.2　平均を引く

平均を引く $x-\overline{x}$	`-(+/%#)`	時系列解析では前もってデータの平均を計算し、データから平均を引いた数を用いることがよく行われる。(最後に戻しておく)

1.4.3　対数化

　重回帰では対数線形を用いることにより、不均一分散が取り除かれる、係数が弾性値を表し解釈しやすい、単位が取捨されるなどの好ましい性質があるので、頻繁に用いられる。 底に *e* を用いる自然対数を用いる。*log* は *ln* とも記述する。J には単項の `^.` で自然対数が、両項で、任意の数を底にとる対数のプリミティブが用意されている。

```
   ^.100
^ 4.60517

^ 4.60517
100
```

<table>
<tr><td>自然対数
$e^y = x$
$logx = y$</td><td>1x1
2.71828

J の 1x1 に e の数値が組み込まれてある。</td><td><pre>
 ^. KD
1.72277 3.4012
1.75786 3.2581
1.79176 3.49651
1.82455 3.43399
 1.8563 3.49651
 1.8563 3.55535
 1.8563 3.61092
1.88707 3.58352
1.91692 3.49651
</pre></td></tr>
</table>

J 言語ノート

`^.`	*naturel log*	自然対数 · 常用対数は `10 ^. 100 is 2`
`^`	e^n	`^ 4.60517 is 100`
1x1	e	e の組込関数

ネイピアと対数

対数とは底を等しくしたときの指数を取り出す関数である。スコットランド人でエディンバラの近くに生まれた *John Napier*(1550 – 1617) が「驚くべき対数法則の記述」を著したのは 1614 年であった。神学と星占術の研究者で数学者ではなかった。勃興してきた天文学の計算の工夫として、対数や e を考案し、天文学者を計算奴隷の仕事から解放した。またネイピアの計算棒といわれる計算器具も開発した。

1.5　共分散行列と相関行列

一度に分散共分散行列と相関行列を求めると、データ構造の見通しが良くなる。

1.5.1　分散と共分散

分散とはデータの平均値からの乖離を 2 乗したものの平均値である。
標準偏差とは、分散の正の平方根である。

<table>
<tr><td>残差平方和（全変動）</td><td>$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} X_i$
$\sum(x - \bar{x})^2$</td><td><pre>
ss=: [: +/ (*:@dev)

mean=: +/ % # NB. am
dev=: - mean
</pre></td></tr>
</table>

分散 Variance	$\frac{\sum(x-\overline{x})^2}{n}$	`var=:ss%#`
標準偏差 Standard deviation	$\sqrt{\frac{\sum(x-\overline{x})^2}{n}}$	`sd=: %:&(ss%#)`
変動係数	$\frac{sd}{\overline{x}}$	`vr=: sd % mean`

共分散	変数の相関の度合いを示す
	$\frac{1}{N}\sum(x-\overline{x})(y-\overline{y})$
		`cov=: # %~ ([: +/ [: */"1 dev)`
		
		`dev=: -"1(+/ % #)`
		`cov=:#  %~ ([: +/ [:*/ "1 (-"1(+/ % #)))`
		(一行で記述)

1.5.2 分散共分散行列と相関行列

多変数の場合、分散共分散行列や相関行列を一度に求めた方がデータの見通しがよい。
$V(X_1+X_2+\cdots+X_n)=\sum_{i=1}^{n}V(X_i+\sum\sum_{i\neq j}Cov(X_i,X_j)$

$$V=\begin{pmatrix} S_{xx} & S_{xy} & S_{xz} \\ S_{yx} & S_{yy} & S_{yz} \\ S_{zx} & S_{zy} & S_{zz} \end{pmatrix}$$

x	y
$x1$	$y1$
$x2$	$y2$
$x3$	$y3$
$x4$	$y4$

$$C=\frac{{}^tXX}{N}=\frac{1}{N}\begin{pmatrix} x_1-\overline{x} & x_2-\overline{x} & \ldots & x_n-\overline{x} \\ y_1-\overline{y} & y_2-\overline{y} & \ldots & y_n-\overline{y} \end{pmatrix}\begin{pmatrix} x_1-\overline{x} & y_1-\overline{y} \\ x_2-\overline{x} & y_2-\overline{y} \\ x_3-\overline{x} & y_3-\overline{y} \\ x_n-\overline{x} & y_n-\overline{y} \end{pmatrix}$$

$$\frac{1}{N}\begin{pmatrix} (x_n-\overline{x})(x_n-\overline{x}) & (x_n-\overline{x})(y_n-\overline{y}) \\ (y_n-\overline{y})(x_n-\overline{x}) & (y_n-\overline{y})(y_n-\overline{y}) \end{pmatrix}=\begin{pmatrix} x\text{の分散} & \text{共分散} \\ \text{共分散} & \text{yの分散} \end{pmatrix}$$

Working Example

```
   vartable s_1
  50 27.5 17.5 42.5
27.5    30    15    25
17.5    15    30    10
42.5    25    10    50
```

経過と説明

<table>
<tr><td>データ ある生徒身長 体重 胸囲 座高

   s_1 NB.(1)
145 30 60 70
145 35 70 75
150 35 65 80
150 40 70 70
155 40 75 75
155 45 70 80
160 40 80 85
160 50 70 90
165 40 65 90
165 45 75 85</td>
<td>列の平均を引く

   dev2 s_1 NB. (2)
_10 _10 _10 _10
_10 _5   0 _5
 _5 _5 _5   0
 _5   0   0 _10
  0   0   5 _5
  0   5   0   0
  5   0 10   5
  5 10   0 10
 10   0 _5 10
 10   5   5   5</td></tr>
<tr><td>内積演算

(|: dev2 s_1) +/ . *
(dev2 s_1) NB. (3)

500 275 175 425
275 300 150 250
175 150 300 100
425 250 100 500</td>
<td>n で割る。$n = 10$

((|: dev2 s_1) +/ . *
(dev2 s_1)) % # s_1 NB. (4)

 50 27.5 17.5 42.5
27.5   30   15   25
17.5   15   30   10
42.5   25   10   50</td></tr>
</table>

Script

```
vartable=:# %~ |:@dev2 +/ .* dev2
```

J 言語ノート

`+/ . *`　　内積

`|:`　　*Transpose*　転置 · 行と列の交換

1.5.3　相関係数と相関行列

先にデータを標準化しておくと同じ手順で相関行列が求められる。

$$R=\begin{pmatrix} 1 & r_{xy} & r_{xz} \\ r_{yx} & 1 & r_{yz} \\ r_{zx} & r_{zy} & 1 \end{pmatrix}$$

$\rho_{xy} = \frac{(\mathbf{x},\mathbf{y})}{\|x\|\cdot\|y\|}$

X, Y の相関係数 $\rho_{XY} = \dfrac{Cov(X,Y)}{\sqrt{V(X)}\cdot\sqrt{V(Y)}}$

Working Example

```
   cortable s_1
       1 0.710047 0.451848     0.85
0.710047        1      0.5 0.645497
0.451848      0.5        1 0.258199
    0.85 0.645497 0.258199        1
```

Script

```
   cortable
3 : 0
NB. dev2=. -"1 mean=:(+/ % #)
ss=. [: +/ [: *: dev2
sd=. %:&(ss%#)
stand=. dev2@]  %"1  sd@]
cor=.  #@] %~ (|:@stand@]  +/ . *  stand@])
cor y.
)
```

J 言語ノート

`@`	*Atop*	動詞と動詞を繋いで一つの動詞とする
`-"1(+/%#)`	`-"1` *is hook*	データから縦の平均を縦方向に引く

1.5.4　偏相関行列

相関行列は, 各変数間の相関係数の行列表示であり, データの特色が一目で見られる。
偏相関行列は, 相関行列の逆行列とその対角行列から求められる。
対角行列の外積をとり, その平方根で相関行列の逆行列 $-r_{ij}$ を割ればよい。

相関行列 R=$\left[r_{ij}\right]$
R の逆行列 R^{-1}　その要素 r^{ij}
$\mathrm{r}_{ij.o} = \dfrac{-r^{ij}}{\sqrt{r^{ii}r^{jj}}}$

例えば X_1と X_2 間の関係を残りの変数の影響を除去して計算したい場合には、偏相関を求める。この偏相関を、相関行列から一度に求めてしまう。

相関行列 R=$\left[r_{ij}\right]$
R の逆行列 R^{-1}　その要素 r^{ij}

$r_{ij.o} = \frac{-r^{ij}}{\sqrt{r^{ii}r^{jj}}}$

<table>
<tr><td>(年間収入 5 分位) の月当たりの所得と費用。</td><td>相関行列</td><td>偏相関行列</td></tr>
<tr><td><pre>
a3 単位円

所得　教育 教養娯楽
198005 2854 17662
254020 5880 24208
290237 11995 30538
348839 19605 36568
452356 23490 48113
</pre></td><td><pre>
cortable a3
       1 0.967571 0.997407
0.967571   1 0.976533
0.997407 0.976533   1
</pre></td><td><pre>
pcor_table cortable a3
        1 _0.414878 0.965817
_0.414878   1 0.631025
 0.965817 0.631025   1
</pre>
教育と教養娯楽の偏相関は 0.63</td></tr>
</table>

1.5.5　ノルム化

$\| x \| = \sqrt{< x \cdot x >} = \sqrt{x_1^2 + x_2^2 + \cdots + x_n^2}$ はベクトルの大きさ (長さ) を表す。

長さの 2 乗の合計が 1 となるように基準化したユークリッド (Euclid) ノルムがよく用いられる。

ノルム化	`norm=:  [: %:  [: +/   *:`

解説

The length(or norm)of **v** is the nonnegative scalar $\| v \|$

$\| x \| = \sqrt{v \cdot v} = \sqrt{x_1^2 + x_2^2 + ... + x_n^2}$

$\| v \|^2 = v \cdot v$

$\| x - y \| = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 \cdots (x_n - y_n)^2}$

は線分 PQ の長さをあらわす。

Example

v=1 -2 2 0

$\| v \| = \sqrt{(1^2 \quad -2^2 \quad 2^2 \quad 0)} = \sqrt{9}$

$u = \frac{1}{\| v \|} v$

```
euc_norm 1 _2 2 0
```

```
0.333333 _0.666667 0.666667 0
```

Script

```
norm=: [: %: [: +/ *:
euc_norm=: 3 : ' y. % norm y.'
```

1.5.6　関数一覧

標準化	*stand*	*stand y.*
統計関数	平均 *mean* 残差 $dev 分散 *var* 標準偏差 *sd* 変動係数 *vr* 共分散 *cov*	
分散共分散行列	*vartable*	*vartable y.*
相関行列	*cortable*	*cortable y.*
偏相関行列	*pcor_table*	*pcor_table cortable y.*
ユークリッドノルム	*euc_norm*	*euc_norm y.*

1.5.7　Reference

山澤成康　実践　計量経済学入門　日本評論社　２００４
日本銀行調査統計局編　[計量経済分析の基礎と応用]　東洋経済新報社　1985
金谷健一 これなら分かる応用数学教室 共立出版 2003

第 2 章

3 本のスクリプト-線形回帰入門

2.1 Matrix divide

重相関回帰、多項式回帰、自己相関 (AR 法) モデルを計算するスクリプトは次の 3 本の一行のスクリプトで表すことができる。

何れも式の中程に `%.` という記号が入っている。これが $matrix - Divide$ という配列計算言語の強力な演算機能を駆使したものである。

単回帰 重回帰 OLS	`regx=: 3 : '({:"1 y. ) %. 1,.}:"1 y.'`
多項式 Polinominal	`poly1=:4 : 'y. %. (>:i. # y.)^/i. >: x.'`
自己回帰 AR	`ar0=:4 : '(x.}.tmp) %.|.("1)}: >x.<\ tmp=:y.-(+/%#)y.'`

2.1.1 逆行列

$AA^{-} = I$ となる行列を逆行列と言う。I は単位行列である。

正方の行列ならば右に同じサイズの単位行列を並べて、掃き出せば逆行列が求まる。

$$\left.\begin{array}{ccc|ccc} 1 & 2 & 1 & 1 & 0 & 0 \\ 1 & 3 & 1 & 0 & 1 & 0 \\ 1 & 1 & 3 & 0 & 0 & 1 \end{array}\right.$$

J 言語ノート

d=. 3 3 $ 2 1 2 1 3 1 1 1 3
 	
`   d`
`2 1 2`
`1 3 1`
`1 1 3`
	J で逆行列を求める
	
	`   %. d`
	` 0.8 _0.1 _0.5`
	`_0.2  0.4    0`
	`_0.2 _0.1  0.5`
		ガウス法による掃き出し。
		
		` gaussj  d,.=/~i.3`
		`1 0 0  0.8 _0.1 _0.5`
		`0 1 0 _0.2  0.4    0`
		`0 0 1 _0.2 _0.1  0.5`

配列計算言語は行列除算機能を持っている。J では *Matrix inverse Matriv dvide* (%.) である。

正規方程式を経ないで生データからいきなり回帰を行うような手法を採るとき、データが正方になることはほとんど期待できないが、J は縦長の行列からダイレクトに逆行列を求めることができる。

数学ノート

除算の歴史現在の人類がアフリカを出て大凡 4 万年を経て、突然エジプトの地にハイテク古代文明が出現した。エッフェル塔に抜かれるまで最高を誇った巨大ピラミッドや神殿、ナイル河の測量、メソポタミアの灌漑設備、フェニキア人の大型船、ローマのアーチ橋.. 石や日干し煉瓦と人力で作り上げる完成された技術があった。

BC4000 年には 30 日 ×12 月 +5 祝日のエジプト暦が作成され、1582 年のグレゴリオ歴までの間精度で優位に立った。ナイル河の測量では水位 0 point は知られていた。

BC1600 年頃の「リンド · パピルス」と言われる数学書が発見されている。

BC2000 年頃のメソポタミアでは、60 進法で計算が行われ、棒で書かれ乾た赤褐色の粘土板には乗法と特殊な逆数を用いた乗法が書き残されている。

暦、交易、課税、戦争、 人の営みは絶え間なく繰り返され、抽象化の進んだ現代数学とは異なる実用の要請があった。

2.1.2　3 のモデル

OLS 単回帰・重回帰モデル

$$
\text{y=}\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \cdots \\ y_n \end{bmatrix} , \begin{bmatrix} 1 & x_{11} & x_{21} & \cdots & x_{k1} \\ 1 & x_{12} & x_{22} & \cdots & x_{k2} \\ 1 & x_{13} & x_{23} & \cdots & x_{k3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{1n} & x_{2n} & \cdots & x_{kn} \end{bmatrix}
$$

多項式モデル

変数 y が t に関する k 次の多項式によって

$y = C_{00} + C_{01}t + \cdots + c_{0k}t^k + \epsilon$

のように表されている場合に、パラメータの最尤推定は次の式を解くことによって得られる。

$$\begin{bmatrix} S_0 & S_1 & \cdots & S_k \\ S_1 & S_2 & \cdots & S_{k+1} \\ S_2 & S_3 & \cdots & S_{k+2} \\ \vdots & \vdots & \ddots & \vdots \\ S_K & S_{K+1} & \cdots & S_{2k} \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ \cdots \\ c_K \end{bmatrix} = \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ \cdots \\ T_k \end{bmatrix}$$

AR モデル

M 次までの AR モデルの最小二乗解をすべて求めるには、次のようにすればよい。この行列を逆行列を用いれば一気に最小二乗解が得られる。

$$\begin{bmatrix} S_{11} & S_{12} & \cdots & S_{1k} \\ S_{12} & S_{22} & \cdots & S_{2k} \\ S_{13} & S_{23} & \cdots & S_{3k} \\ \vdots & \vdots & \ddots & \vdots \\ S_{1K} & S_{2K} & \cdots & S_{kk} \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ \cdots \\ b_K \end{bmatrix} = \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ \cdots \\ T_k \end{bmatrix}$$

$S_{rs} = \sum_{i=k+1}^{n} = t_{i-r} \cdot t_{i-s}$

$T_k = \sum_{i=k+1}^{n} = t_i \cdot t_{i-r}$

$Q_k = T_0 - (b_1T_1 + b_2T_2 + \cdots + b_kT_k$ 残差平方和

$Mll = -\dfrac{n-k}{n}(logQ_k/(n-k))$

$AIC = -2 \times MLL + 2 \times k = (n-k) \times (logQ_k/(n-k)) + 2 \times k$

AIC の値を最小にする次数 k を見つければよい。

2.2 単回帰モデルと重回帰モデル

単回帰の正規方程式と重回帰モデルの正規方程式は次のようにあらわされる。

$(X'X)\hat{\beta} = X'y$

$\hat{\beta} = (X'X)^{-1}X'y$

これらは、更に次のように簡約される。

$(X'X)^{-1}X'y = \dfrac{X'y}{X'X} = \dfrac{y}{X}$

線型数学で表現すれば単回帰も重回帰も同じである。配列計算言語では、単回帰も重回帰も同じスクリプトで記述でき、入力データにより、単回帰、重回帰のどちらにも適用できる。

2.2.1　単回帰モデル

Working Example

KX　ある果実の直径 *cm*

KY 水分含有率 (%) のデータ。

（データの出典　金谷　「これならわかる応用数学教室」）

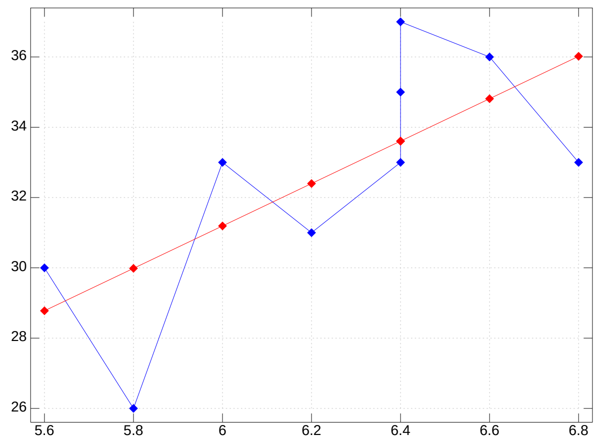

図 2.1　果実の直径と水分含有率

KY KX	1 KX	$(1KX)^{-1}$
	定数項を求めるため X の左に 1 を付加する	逆行列を求める。

`KX,.KY`
`5.6 30`
`5.8 26`
`6 33`
`6.2 31`
`6.4 33`
`6.4 35`
`6.4 37`
`6.6 36`
`6.8 33`
	`1,.KX`
	`1 5.6`
	`1 5.8`
	`1 6`
	`1 6.2`
	`1 6.4`
	`1 6.4`
	`1 6.4`
	`1 6.6`
	`1 6.8`
		`|: %. 1,.KX`
		`3.51504 _0.545113`
		`2.45865 _0.37594`
		`1.40226 _0.206767`
		`0.345865 _0.037594`
		`_0.710526 0.131579`
		`_0.710526 0.131579`
		`_0.710526 0.131579`
		`_1.76692 0.300752`
		`_2.82331 0.469925`

サンプルデータ KX の逆行列を手計算で行おうとすれば大変である。教科書には正方行列の数学ノートに記載した方法は紹介されているが、このような縦長の行列はあまり扱っていない。ここでは、素直に $K.E.Iverson$ の $matrix-divide$(`%.`) に従おう。

$\frac{y}{X}$	$\frac{KY}{1,.KX}$
	両項の `%.` はマトリクスの除算を行う
		`KY %. 1,.KX`
		`_5.01128 6.03383`
		$y = -5.01128 + 6.03383x$

yX^{-1}	$KY\ +/.*(1,.KX)^{-1}$ マトリクスの除算は逆行列の内積と同じである。	`KY +/ . * |: %. 1,.KX` `_5.01128 6.03383`

2.2.2 Linefit

相関係数とモデルのテストは reg_exam_ad を $reg0$ の後ろに連結して行う。
[*1]

```
 reg0 reg_exam_ad KD
+------+----------------+
```

[*1] この様に、動詞 (関数) を引数とする関数を J では副詞と呼ぶ

```
|f=     |_5.01128 6.03383|
+------+----------------+
|corr=:|47.8237         |
+------+----------------+
|AIC:  |18.8684         |
+------+----------------+
|DW=   |2.17534         |
+------+----------------+
|t=:   |_0.33633 2.53299|
+------+----------------+
```

先の散布図とラインフィットは次の関数で作成した。OLS の

```
linefit_reg KD   (KD=: KX,.KY)
```

2.2.3　重相関モデル

線形数学では単相関と重相関は同じ方程式で表すことが出来るた。配列計算言語では、同じプログラムで計算できる。

Warking Example 発電量と経済指数 (1969-1994)

ここではデータが大きいので、EXCEL から OLE で取り出している。
load getexcel.ijs
getexcel ”
e0=. datain 1 1 41 6
reg0 linefit$_r eg$231”1e0

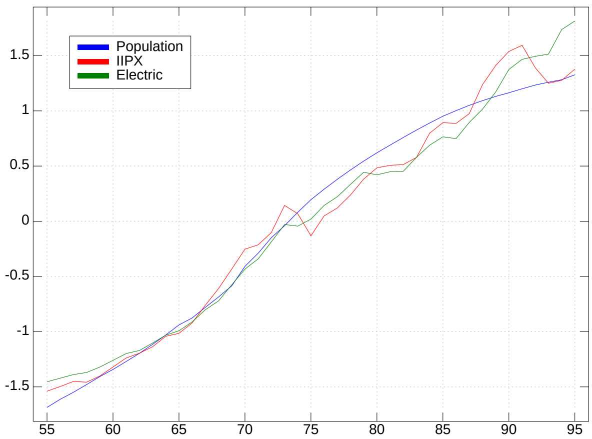

図 2.2　上から発電量 鉱工業生産指数 原油輸入量 人口

y 発電量
x_1 人口
x_2 鉱工業生産指数
x_3 原油輸入量

```
 reg0  2 3 1{"1 e0
_4766.04 5193.81 52.3253
```

$$f = -4766.04 + 5193.81x_1 + 52.3253x_2$$

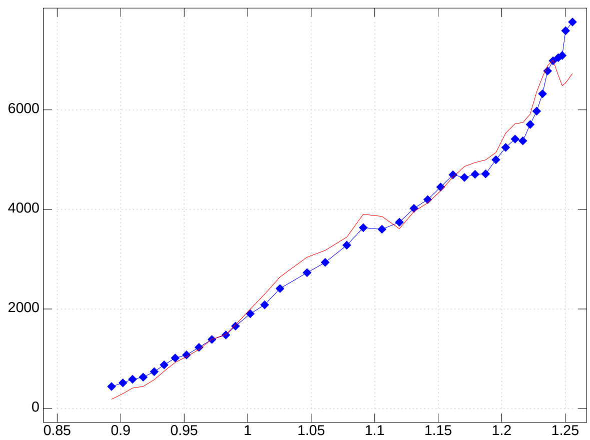

図 2.3　発電量の実数 (青) と推計値 (赤)

t 検定

（スチューデントのｔ検定）　正規分布を使用する平均値の差の検定は，母分散が既知という条件下でしかも，大標本のときだけ信頼できることは知られている．小さな標本から平均値の差の有意性を判定するために考案された平均値の検定がスチューデントのｔ検定である。

近代統計学すなわち確率論と結びついた統計学の幕開けは 1908 年にギネスビールの技師ゴセットによるスチューデントのｔ検定で始まった.

ｎ個の観測値の標本平均ｍと母平均μの差（距離）を不偏標本分散の平方根ｕで割った統計量　　ｔ $= \dfrac{(n-u)}{}$

ｕ$\sqrt{n}$ の分布が自由度ｎ－１のｔ分布に従うことはゴセット（筆名：スチューデント）が最初に発見し，フィッシャーが厳密に証明したことは歴史的事実として有名です.

先の発電量に関する重相関モデルの採点結果である。

```
  reg0  reg_exam_ad 2 3 1{"1 e0
+-------+------------------------+
|f=     |_4766.04 5193.81 52.3253|
+-------+------------------------+
```

```
|corr=:|97.9651                |
+------+-----------------------+
|AIC:  |478.907                |
+------+-----------------------+
|DW=   |0.202234               |
+------+-----------------------+
|t=:   |_2.01828 1.95848 5.00342|
+------+-----------------------+
```

2.3　多項式モデル

```
                 4 poly1 s2

_2.01542 5.65633 _0.889141 0.0485201 _0.000848678
```

$y = -2.01542 + 5.65633x - 0.889141x^2 + 0.0485201x^3 - 0.000848678x^4$
データが長いと紙面を取るので、説明のため、10 個の乱数データを用意した。

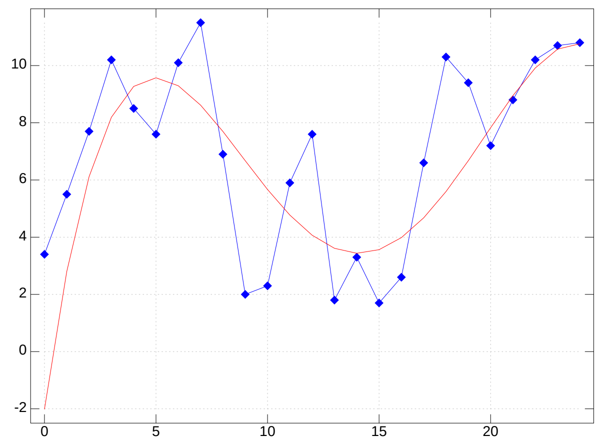

図 2.4　東京の冬期の最低気温 (青) と推計値 (赤)4 次

X, .a X は単に順序数でオリジンを 1 とした。	X の 0 1 2 3 乗	左の逆行列

<table>
<tr>
<td>20 までの 10 個の乱数</td>
<td>データの個数を求め、1 から順にナンバーを付け、これを 3 次の場合なら 0 1 2 3 乗する。0 乗が定数項になる。</td>
<td>左の逆行列</td>
</tr>
<tr>
<td>(>:i. # a) ,.a
 1 6
 2 3
 3 19
 4 15
 5 10
 6 14
 7 0
 8 7
 9 12
10 17</td>
<td>(>:i. # a)^/ i. >: 3
1 1   1   1
1 2   4   8
1 3   9   27
1 4 16   64
1 5 25 125
1 6 36 216
1 7 49 343
1 8 64 512
1 9 81 729
1 10 100 1000</td>
<td>6.2": |: %. 1,.(>: i. # a)^/ >: i.3
  1.60 _0.93 0.16 _0.01
  0.27 0.08 _0.04 0.00
 _0.40 0.53 _0.12 0.01
 _0.57 0.57 _0.11 0.01
 _0.40 0.34 _0.05 0.00
 _0.07 0.00 0.02 0.00
  0.27 _0.32 0.09 _0.01
  0.43 _0.45 0.11 _0.01
  0.27 _0.25 0.05 0.00
 _0.40 0.43 _0.11 0.01</td>
</tr>
</table>

```
  4 poly0 s2
_2.01542 5.65633 _0.889141 0.0485201 _0.000848678
```

$f = -2.01542 + 5.65633t_{-1} - 0.889141t_{-2} - 0.0485201t_{-3} - 0.000848678t_{-4}$

多項式の最適次数も AIC の最小の値で求める。

東京都の 3 月の最低気温のデータでモデル選択を行う。4 次の AIC が最小である。

```
  3 poly_exam S1
co. of det.(%):  39.09
value of AIC  :  53.52
Value of MLE:   4.896   1.133  _0.141   0.004

  4 poly_exam S1
co. of det.(%):  62.51
value of AIC  :  43.38
Value of MLE:  _2.015   5.656  _0.889   0.049  _0.001

  5 poly_exam S1
co. of det.(%):  62.51
value of AIC  :  45.38
Value of MLE:  _1.936   5.585  _0.871   0.047  _0.001   0.000
```

$linefit\_polt$ でグラフを比較してみる。

```
 4 linefit_poly S1
  pd 'eps temp\poly_0.eps'
```

2.4 自己回帰モデル AR

自己回帰モデル (AR ·Auto Regressive Model) は離散型の時系列解析の代表的な方法であって、時系列の現時点における値が、自己の過去の時点の線形結合で表される。

自己回帰のアルゴリズムには $Yull - Walker$ 法、$Burk$ 法、$Householder$ 法があるが、ここでは、$Householder$ のデータ構成を用いるが $Householder$ 変換を経由しないで、逆行列を用いて、一度に計算してしまうシンプルな方法を用いている。

なお、3 方式のアルゴリズムによる回帰係数には若干の差があり、$Burk$ 法はピークに強く、$Yull - Walker$ 法は谷に強い。$householder$ 法は中立的であると言われている。

最初の 10 個のデータで自己回帰の仕組みを再現してみよう。

$x_{t-1}, x_{t-2}, x_{t-3}, \cdots, x_{t-n}$ という型のデータの組み合わせ (X) を作る。

次数相当分のデータ (y) が組み合わせの結果減少するので、先頭から落として個数を合わせる。

$\frac{y}{X}$ を計算する。 X の逆行列が求まるので、簡単なマトリクスの除算で係数が求まる。

<table>
<tr>
<td>データの最初から 12 個
NR, y</td>
<td>$y, x_{t-0}, x_{t-1}, x_{t-2}, x_{t-3}$</td>
<td></td>
</tr>
<tr>
<td>1 3.4
2 5.5
3 7.7
4 10.2
5 8.5
6 7.6
7 10.1
8 11.5
9 6.9
10 2
......
25 10.7
26 10.8</td>
<td>(2}.s2),.|."1 >3<\s2
<table>
<tr><th>Y</th><th>x_{t-1}</th><th>x_{t-2}</th><th>x_{t-3}</th></tr>
<tr><td>3.4</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>5.5</td><td>3.4</td><td>0</td><td>0</td></tr>
<tr><td>7.7</td><td>5.5</td><td>3.4</td><td>0</td></tr>
<tr><td>10.2</td><td>7.7</td><td>5.5</td><td>3.4</td></tr>
<tr><td>8.5</td><td>10.2</td><td>7.7</td><td>5.5</td></tr>
<tr><td>7.6</td><td>8.5</td><td>10.2</td><td>7.7</td></tr>
<tr><td>10.1</td><td>7.6</td><td>8.5</td><td>10.2</td></tr>
<tr><td>11.5</td><td>10.1</td><td>7.6</td><td>8.5</td></tr>
<tr><td>6.9</td><td>11.5</td><td>10.1</td><td>7.6</td></tr>
<tr><td>2</td><td>6.9</td><td>11.5</td><td>10.1</td></tr>
<tr><td>...</td><td>...</td><td>...</td><td>...</td></tr>
<tr><td>10.7</td><td>10.2</td><td>8.8</td><td>7.2</td></tr>
<tr><td>10.8</td><td>10.7</td><td>10.2</td><td>8.8</td></tr>
</table>
</td>
<td>$Inefix$(\) の強力な機能で X を作成する。一つずつづらした組み合わせを Box(<) とInfix(\) で作りOpen(¿)で開く。
$x_{t-1}, x_{t-2}, x_{t-3}$ の並びになるように縦に回転 $Rotate$(|.”1) させる。
Y の時点と個数を合わせるため古い方から指定次数分をを落とす。
最後の 1 行は回帰には用いない。虎の子として予測に用いる。

3<\ >:i.5
+-----+-----+-----+
|1 2 3|2 3 4|3 4 5|
+-----+-----+-----+
>3<\ >:i.5
1 2 3
2 3 4
3 4 5
|. "1>3<\ >:i.5
3 2 1
4 3 2
5 4 3

この形になれば重相関と同じようにマトリクスの除算で解が求められる。</td>
</tr>
</table>

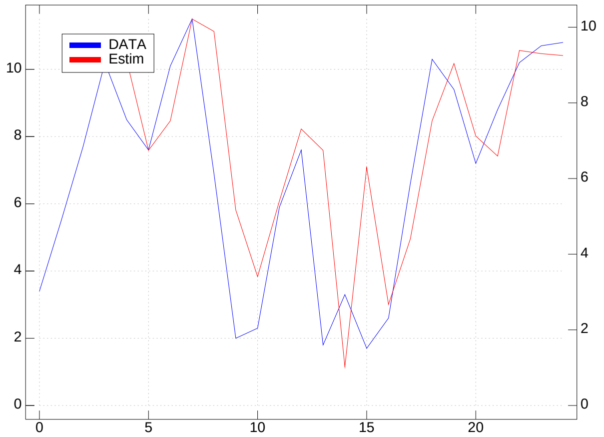

図 2.5　東京の冬期の最低気温

AIC

$Mll = -\frac{n-k}{n}(logQ_k/(n-k))$

$AIC = -2 \times MLL + 2 \times k = (n-k) \times (logQ_k/(n-k)) + 2 \times k$

AIC の値を最小にする次数 k を見つければよい。

```
   3 ar_exam T
+------------------------------+
|mean=6.904                    |
+------------------------------+
|R^2=0.535289                  |
+------------------------------+
|AIC=41.5126                   |
+------------------------------+
|reg=1.00361 _0.586829 0.246381|
+------------------------------+
   4 ar_exam T
+----------------------------------------+
|mean=6.904                              |
+----------------------------------------+
|R^2=0.546771                            |
+----------------------------------------+
|AIC=41.4362                             |
+----------------------------------------+
|reg=0.973496 _0.548562 0.232687 0.0641936|
+----------------------------------------+
   5 ar_exam T
```

```
+-------------------------------------------------+
|mean=6.904                                       |
+-------------------------------------------------+
|R^2=0.564681                                     |
+-------------------------------------------------+
|AIC=41.7886                                      |
+-------------------------------------------------+
|reg=1.00576 _0.525087 0.122851 0.244503 _0.21142|
+-------------------------------------------------+
```

```
 4 linefit_ar T
 pd 'eps temp\ar_0.eps'
```

2.4.1　関数一覧

単回帰 $|cdot$ 重回帰	$reg0$ reg_exam_ad reg_estim_ad $linefit\_reg$	 reg0 reg_exam_ad 2 3 1 {"1 e0 reg0 linefit_reg 2 3 1 {"1 e0
多項式回帰	$poly1$ poly_exam poly_estm linefit_poly	 4 poly_exam s20 4 poly_estm s2 4 linefit_poly s2
自己回帰	$ar0$ ar_exam ar_estm linefit_ar	 4 ar_exam s20 4 ar_estm s2 4 linefit_ar s2
掃き出し	gaussj	using numeric.ijs