J による時系列解析（Time Series）

M.Shimura
JCD02773@nifty.com

2001.12.8

1　移動平均

1.1　移動平均

移動平均 Moving Analsys	任意の移動平均	m mav n m 次数 n Data

J は、移動平均を簡潔なイディオムで表現できる。

```
mav=: +/\ % [
```

J の infix（\) の機能は強力であり x. で指定した移動平均のためのずらした組み合わせを簡単に作る。

```
] s=. 10 ?. 30

26 7 15 12 20 23 28 24 1 29

   3 <\ s
+-------+------+--------+--------+--------+--------+-------+-------+
|26 7 15|7 15 12|15 12 20|12 20 23|20 23 28|23 28 24|28 24 1|24 1 29|
+-------+------+--------+--------+--------+--------+-------+-------+
   3 +/\ s
48 34 47 55 71 75 53 54
```

```
   3 mav s
16 11.3333 15.6667 18.3333 23.6667 25 17.6667 18
```

1.2 偶数の x.

移動平均は,x.（左パラメーター) が引数の場合には, 原データとの対比のために中心化を行うとよい。`@.` 以下の条件式で偶数奇数の判定を行い、偶数の場合は, 中心化を行っている。

```
ma=:+/\  %  [
ma_c=: 2&ma@ma ‘ ma @. (2&|@[)   NB.Centred   MA

   a=.10?.20
   a,:(2 ma a),:2 ma_c a

 17    4    9     7     12    14    18    13   0  6
10.5  6.5   8     9.5   13    16    15.5  6.5  3  0
 8.5  7.25  8.75 11.25  14.5  15.75 11    4.75 0  0
```

1.3 マトリクスデータへの適用

この mav はランクの指定で, マトリクスデータにも一括して適用できる。

```
     s2=: ?. 5 12 $ 30
   s2
 3 22 13 15  6  1 20 20 28 11 15 24
 1  1 15 20  0 11  2 12 20 17 27 25
15  2 19 12 21 27 22  7  1 22  9 18
22 29 10  7 29 21 22 19  2 18 26  8
13 22 14  7  8 10  4 14 26 27  1 27

6.2 ":   3 mav "1 s2
```

```
 12.67 16.67 11.33  7.33  9.00 13.67 22.67 19.67 18.00 16.67
  5.67 12.00 11.67 10.33  4.33  8.33 11.33 16.33 21.33 23.00
 12.00 11.00 17.33 20.00 23.33 18.67 10.00 10.00 10.67 16.33
 20.33 15.33 15.33 19.00 24.00 20.67 14.33 13.00 15.33 17.33
 16.33 14.33  9.67  8.33  7.33  9.33 14.67 22.33 18.00 18.33
```

1.4 12 ヶ月移動平均

```
mav=: +/\ % [
NB. Season adjustment using 12 month
mav12=: [: 2&mav 12&mav

      2<\ 12 mav i. 24
+-------+-------+-------+-------+--------+---------+---------+---------+
|5.5 6.5|6.5 7.5|7.5 8.5|8.5 9.5|9.5 10.5|10.5 11.5|11.5 12.5|12.5 13.5|
+-------+-------+-------+-------+--------+---------+---------+---------+...
   mav12 i. 24
6 7 8 9 10 11 12 13 14 15 16 17
```

1.5　移動平均の PLOT

移動平均 Moving Analsys	plotcomb [任意の移動平均のグラフ]	m plotcomb n m 次数 n Data

移動平均を求め原データと同時にグラフにあらわす関数を用意した.

・任意の次数（例では３＝四半期）と原系列を plot する

偶数の場合はグラフにあらわすため中心化の処理を行っている。

1.6　経過と解説

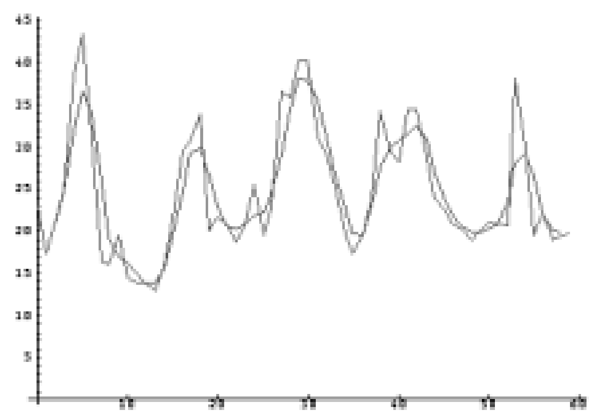

図 1: 3 月移動平均

1.6.1　Script

```
NB. =================================
NB. mav any dimension
NB. ussage: x. plotcomb y.  (x.  dimension y. data)
NB. ==================================
p_sub0=: 4 : 0
NB. make mav anytime odd or even
mav=: +/\ % [
mav_even=:[: 2&mav mav
select. tmp0 =:2 | x.
case. 0 = tmp0 do. goto_odd.
case. 1 = tmp0 do. goto_even.
end.
label_odd.
tmp10=:x. mav y.
tmp2=:<. -: x.
return.
label_even.
tmp10=: x. mav_even y.
tmp2=:>. -: x.
return.
)
p_sub1=: 4 : 0
NB. make plot data by mav
x. p_sub0 y.
tmp30=:(tmp2 # tmp2) ,  tmp2 +i. # tmp10
tmp3=:tmp30,tmp2 #({: tmp30)  NB. x axis
tmp4=:( tmp2 # {. tmp10), tmp10,tmp20=:tmp2 # {: tmp10 NB. y axis
tmp5=:tmp3;tmp4
)

plotcomb=: 4 : 0
NB. usage: x. plotcomb y.
NB. x. jisuu
```

```
NB. y. data
pd 'reset'
pd x. p_sub1 y.
pd y.
pd 'color red'
pd  tmp5
pd 'show'
)
```

2 差分

差分	`difff=. 3 : '-/( L:0) 2 <\ y.'` `NB. take dfiierrence`	

非定常時系列は一階差分を取ると定常時系列となることが多い。一階差分も infix を用いると用意に作れる。

2.1 経過と解説

シードを固定した乱数

```
   10 ?. 10
8 2 4 3 7 5 1 0 9 6
```

<\ **は** Infix **オーバーラップさせて** 2 **個ずつ組み合わせる。**

```
   2<\ 10 ?. 10
+---+---+---+---+---+---+---+---+---+
|8 2|2 4|4 3|3 7|7 5|5 1|1 0|0 9|9 6|
+---+---+---+---+---+---+---+---+---+
```

ボックスの中での除算

```
     -/ (L:0) 2<\   10 ?. 10
+-+--+-+--+-+-+-+--+-+
|6|_2|1|_4|2|4|1|_9|3|
+-+--+-+--+-+-+-+--+-+
```

```
  plot > df 100 ?. 100
```

```
NB. ==================
```

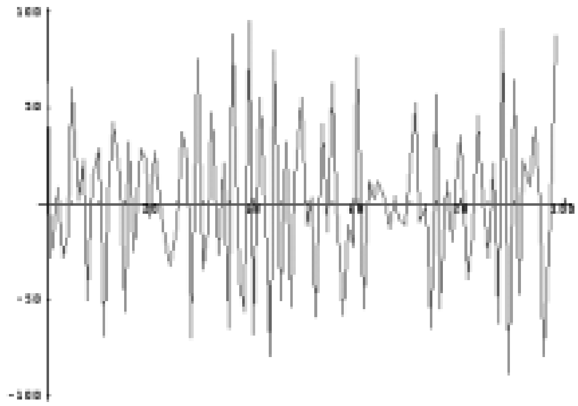

図 2: 差分

```
NB. diff and plot
NB. Usage: diff_plot y.
NB. ==================
diff=:3 : '-/(L:0) 2<\ y.'
diff_plot=: 3 : 0
pd 'reset'
pd > diff y.
pd 'show'
)
```

3　ランダムウオーク

ランドム ウオーク	`rw=: 3 : '+/\ ( ? y. $ 2) { _1 1'`	

0　1 のランダムウオークを時間軸に沿って並べると時系列に似たパターンが現れる。
C.Reiter が作成した, 簡潔にランダムウオークを生成するスクリプトを紹介する。

```
plot +/\ _1 1 {~ ?. 300 $ 2
```

3.1　経過と説明

```
  ?. 10 $ 2
0 1 0 1 0 0 1 1 1 0

  (?. 10 $ 2) { _1 1
_1 1 _1 1 _1 _1 1 1 1 _1

  <\ (?. 10 $ 2) { _1 1 NB. prefix
+--+----+-------+---------+------------+--------------+-----------------+
|_1|_1 1|_1 1 _1|_1 1 _1 1|_1 1 _1 1 _1|_1 1 _1 1 _1 _1|_1 1 _1 1 _1 _1 1|
+--+----+-------+---------+------------+--------------+-----------------+

 +/\ (?. 10 $ 2) { _1 1

_1 0 _1 0 _1 _2 _1 0 1 0
```

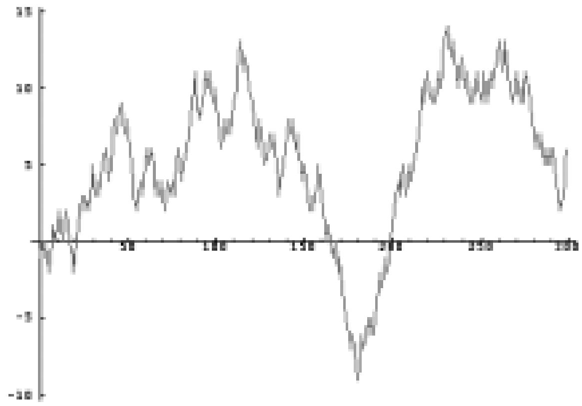

図 3: A Random walk with Steps _1 or 1

3.1.1　正規乱数

randunif (array of psuedo-random real numbers)
単項では０，１間の乱数 2 項では x. で指定する区間の乱数を生成する。

```
randunif=: (?%<:)@:($&2147483647) : ( {. @[+({:-{.)@[*$:@])
rw2=: +/\ _1 1 randunif 300
NB. plot rw2

   randunif 2 3

0.455961 0.341605 0.354983
0.19473  0.83141 0.507221
```

2 項　x.　で指定する間の乱数

```
  _1 2 randunif 2 3

   1.5792  0.54126 0.955131
_0.108855 0.471075 0.363747
```

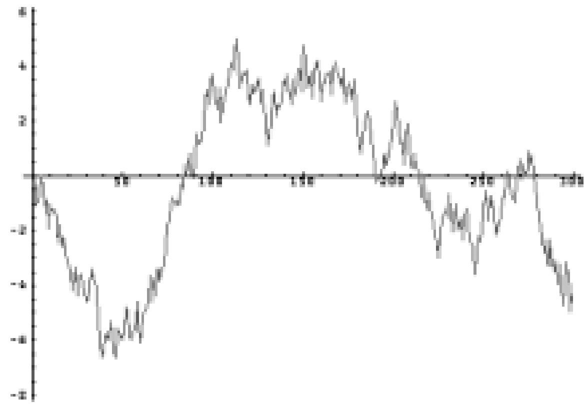

図 4: A Random walk rw2

3.1.2　ランダムウオークと経済学

ランダムウオークは, 経済の分野では, 効率的市場仮説として,「市場は, 全ての情報を完全かつ正しく反映しているとき, 効率的であるといわれ, ・・ある情報セットに基づいて取引を行っても, 経済的利潤を上げることは不可能である。」という結論に帰着する。

この説に従えば, 直前の情報に, 全てが合理的に反映されており, 株価や為替レート, あるいは小麦価格など, 資産価値や商品価格が, ランダムウオークするならば, 過去の価格変動のパターンを精緻な高度な統計的方法を用いて分析しようとしても全く無駄なこととなる。

ランダムウオークは, 自己相関を持たず, 均一分散である。ランダムウオークの検定には, ディッキー・フラーテストや F 値タイプ検定が用いられる。

3.2　Script

```
NB. from Cliford A. Reiter :Fractal Visualization and J (2000)
rw=: +/\ _1 1{ ~? 300 $ 2

NB. plot rw
randunif=: (?%<:)@:($&2147483647) : ( {. @[+({:-{.)@[*$:@])
rw2=: +/\ _1 1 randunif 300
```

```
NB. plot rw2

NB. Apply to any number ,Arrenged by M.Shimura
rw=: 3 : '+/\ ( ? y. $ 2) { _1 1'
rw2=: 3 : '+/\ _1 1 randunif y.'
```

3.3 参考文献

Cliford A. Reiter :Fractal Visualization and J (2000)
蓑谷千凰彦「金融データの統計分析」東洋経済新報社 (2001)

4　自己相関

4.1　自己相関　ACF(k) autocorreration function

ACF(K)		

4.1.1　解説

$$ACF(k) = \frac{\Sigma_{t=k-1}^{n}(Y_t - \overline{Y})(Y_{t-k} - \overline{Y})}{\Sigma_{t=1}^{n}(Y_t - \overline{Y})^2}$$

ACF では $\overline{Y}$ に原系列の平均を用いる。Paerson ではずらした後の各個数の平均を用いる.

定常過程の場合,$X(t)$ の期待値は, 時刻ｔに依存しない。自己共分散 $Cov[X(t_1), X(t_2)]$ も時刻 t_1, t_2 には依存しないで, その時間差 $\tau = t_1 - t_2$ のみに依存する。

本プログラムは, 自己相関関数を一度に求めてしまうものである。（長い系列の場合は, 多くのメモリーが必要である）

4.1.2　経過と説明

```
   a
3 6 8 4 4 8
```

```
C1=: ( <\. y.) ,.  |.@(<\)y.
```

右の欄は上から $Y_t, Y_{t-1}, Y_{t-2}, Y_{t-3}...$ となっている。

左の欄は頭を落とし, 右の欄にサイズを合わせたもの。

```
   C1
+-----------+-----------+
|3 6 8 4 4 8|3 6 8 4 4 8|
+-----------+-----------+
|6 8 4 4 8  |3 6 8 4 4  |
```

```
+-----------+-----------+
|8 4 4 8    |3 6 8 4    |
+-----------+-----------+
|4 4 8      |3 6 8      |
+-----------+-----------+
|4 8        |3 6        |
+-----------+-----------+
|8          |3          |
+-----------+-----------+


   mean   y.
 5.5

   C3   NB.   dev  C1 - (L:0) mean y.
+-------------------------+-------------------------+
|_2.5 0.5 2.5 _1.5 _1.5 2.5|_2.5 0.5 2.5 _1.5 _1.5 2.5|
+-------------------------+-------------------------+
|0.5 2.5 _1.5 _1.5 2.5     |_2.5 0.5 2.5 _1.5 _1.5    |
+-------------------------+-------------------------+
|2.5 _1.5 _1.5 2.5         |_2.5 0.5 2.5 _1.5         |
+-------------------------+-------------------------+
|_1.5 _1.5 2.5             |_2.5 0.5 2.5              |
+-------------------------+-------------------------+
|_1.5 2.5                  |_2.5 0.5                  |
+-------------------------+-------------------------+
|2.5                       |_2.5                      |
+-------------------------+-------------------------+

   C4  =: +/ "1 */ "1 >C3
23.5 _5.25 _14.5 9.25 5 _6.25
```

```
   acf2 a
1 _0.223404 _0.617021 0.393617 0.212766 _0.265957
```

4.1.3 SCRIPT

```
acf2=: 3 : 0
NB. ACF autocorellation coefficients (many times at once)
C1=: ( <\. y.) ,.  |.@(<\)y.
C2=: (mean=. +/ % #) y.
C3=:  C1 - L:0 C2
C4=: +/ "1 */ "2 >C3
NB. C5=:> +/ L:0 ^&2 L:0  {. "1 C3
C4 % ( +/ ^&2 >{. {. C3)
)
```

4.1.4 **出典**

Stephen A. DeLurgio [Forcasting Principles and Applications] McGRAW-HILL 1998

4.2 C.A. Reiter

	`acov=: ([: +/  [ (( -@[ }. ]) * }.) (- (+/ % #))@] ) % #@]`
	`NB. r(t)`
	`acor=: ( acov % 0: acov ]) " 0 1`
	`NB. p(t)`	

4.2.1　解説

Script written by Criford A.Reiter [acor] is same as ACF

$\hat{\gamma}(t) = \frac{1}{T} \sum_{i=1}^{T-t} (Xt + i - \overline{X})(Xi - \overline{X})$

$\rho_{ss} = \frac{R_{ss}(s)}{R_{ss}(0)}$

4.2.2　経過と説明

```
   2 acor a
_0.617021

   3 acor a
0.393617

   acf2 a
1 _0.223404 _0.617021 0.393617 0.212766 _0.265957
```

4.3　Pearsons autocorrelation coefficient

PACF(k)		

4.3.1　解説

$r_{Y_t Y_{t-k}} = \frac{COV(Y_t, Y_{t-k})}{S_{Yt}, S_{Yt-k}}$

4.3.2　経過と説明

a=:3 6 8 4 4 8

共分散　$Cov(Y_t Y_{t-k})$

```
C3=: (+/ "1 */ "2 >  dev L:0 C1) %  n=:> {. "1  <:@# L:0  C1
23.5 _4 _14 9.33333 6 0
```

$S_{Yt} S_{Yt-k}$は　Y_t,　Y_{t-k} の標準偏差

```
   pacf a
1 _0.25 _0.911322 0.802955 1 0
```

4.3.3　Script

```
pacf=: 3 : 0
NB. Pearsons autocorellations  (many times at once)
C1=: ( <\. y.) ,.  |.@(<\)y.
dev=. - (+/ % #)
NB. Cov(Yt Yt-k)
C3=: (C2=:+/ "1 */ "2 >  dev L:0 C1) %  n=:> {. "1  <:@# L:0  C1
NB. Syt Syt-k
C4=: +/ L:0 *:  L:0 dev L:0 C1
C5=:*/ "1 > %: L:0  C4 % L:0  n2=:<:@# L:0 C1
C3 % C5
)
```

4.3.4　**出典**

Stephen A. DeLurgio [Forcasting Principles and Applications] McGRAW-HILL 1998

5　Autoregression　**自己回帰**

AR（Autoregression) の解法には Yule-Walker 法, バーク法, ハウスホルダー法がある。

5.1　Yule-Walker 法

$C_M a_M = c_M$
$c(k) = \frac{1}{N} \sum_{N-k}^{s=1} x(s)x(s+h)(0)$

$$(k = 0, 1, 2, ..., M)$$

$$C_M = \begin{vmatrix} c(0) & c(1) & \dots & c(M-2) & c(M-1) \\ c(1) & c(0) & c(1) & \dots & c(M-2) \\ \dots & \dots & \dots & \dots & \dots \\ c(M-2) & \dots & c(1) & c(0) & c(1) \\ c(M-1) & c(M-2) & \dots & c(1) & c(0) \end{vmatrix}$$

$c_M = (c(1), c(2), ...c(M))^T$

自己相関　ACF(k) を用いて C_M を作れば, この方程式をクラメール法やガウス法で解くことにより, 直接 AR の係数が求められる。（レビンソンのアルゴリズムは用いない。）

5.1.1　**経過と結果**

```
  acf2 a
1 _0.223404 _0.617021 0.393617 0.212766 _0.265957
```

ACF(3)　0 次から 2 次の C_M の右に,c_M を連結したもの

```
  Y4
        1 _0.223404 _0.617021 _0.223404
_0.223404         1 _0.223404 _0.617021
_0.617021 _0.223404         1  0.393617
```

上の拡大行列をクラメール法で解いたもの。 cr=: %.}:"1 NB. Cramer Method

```
  3 yw a
         1 _8.88178e_16 6.66134e_16   _0.37704
_2.22045e_16            1 4.44089e_16   _0.70024
_6.66134e_16 _8.32667e_16           1 0.00453847
```

$$X(t) = -0.37704x(t-1) - 0.70024x(t-2) + 0.00453847x(t-3)$$

5.1.2 Script

```
  cmtest=:4 : 0
NB.  Yule- Walker sub
NB. make C(m) Matrix
NB. for univariate AR
T1=: (|.}. M) , M=:  y.
T2=: x. # ,: T1
T3=: ({. TMP), <: 1{ TMP=: $ T2
T4=:(-x.) {. ("1) TMP2=: T3 $ , T2
)

 yw=:4 : 0
NB. Yule Walker(Univariate)
NB. Data type is Yoko list
NB. X(t)=X1(t-1)+ X2(t-2) + X3(t-3)......
NB. Usage: x. jisuu // y. Data
Y1=: acf2 y.
Y2=:(>: x.) {. Y1
Y3=:}. (>: x.) {. Y1
Y4=: (x. cmtest Y2),. Y3
```

```
Y5=: (cr=: %.}:"1) Y4
)
```

5.1.3 AIC

```
29 aictest corn
68.6085
   29 plot_ar corn
```

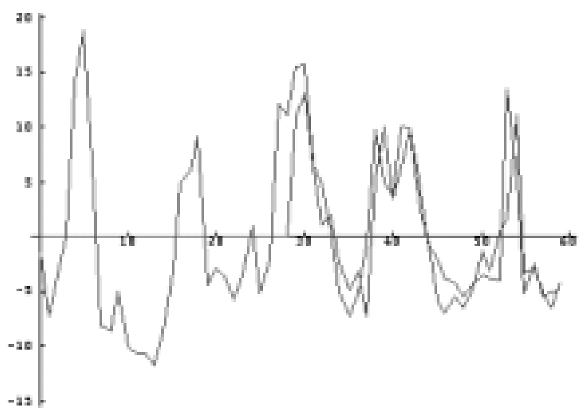

図 5: A Random walk with Steps _1 or 1

```
NB. AR  1JI
NB. ****************************************
aictest=: 4 : 0
NB. find AIC value
NB. Usage: x. aictest y. // x. is jisuu // y. is data
y=: dev y.
A0=:}: A00=:x. <\ y NB.
A1=: {:"1  x. yw y NB. tmp is reg
A2=: (tmp0=:|."1 >A0) +/ . * A1 NB. (|."1 ) is rotate A0
A3=:  (tmp1=:(x. }. y)) - A2  NB.
```

```
A4=:(tmp2=: +/ *: A3) % n=: (# A3) - x. NB. tmp2 is Q
A5=:(n * ^. A4)++:x.
)
```

5.1.4 回帰と予測

回帰と予測は, 次のような組み合わせを作ると $x(t-4)x(t-3)x(t-2)x(t-1)$ になっており, 回帰は, これに,$A(m)$ を掛ける。最下行は,1 期先の予測に用いる。

```
|."1 > 4 <\ a=. >: i. 12
 4  3  2 1 | 5
 5  4  3 2 | 6
 6  5  4 3 | 7
 7  6  5 4 | 8
 8  7  6 5 | 9
 9  8  7 6 | 10
10  9  8 7 | 11
11 10  9 8 | 12
12 11 10 9 | x
```

29 pred corn
20.0508

```
plot_ar =: 4 : 0
NB. plotAR
x. aictest y.
pd 'reset'
pd 'color green'
pd  dev y.
pd 'color red'
pd (x. # 0),A2
pd 'show'
```

```
)

pred=: 4 : 0
NB. prediction AR
NB. Usage: x. pred y.
x. aictest y.
A09=: (+/ % #) y.
A10=:|.  {: > A00
A11=: A10 +/ . * A1
A12=: A09 + A11
)
```

5.1.5　参考文献

石黒真木夫「予測と AR モデル」（尾崎・北川編　時系列解析の方法　朝倉書店）１９９８

6　レビンソンのアルゴリズム

レビンソンのアルゴリズムは最初に偏相関係数を求め, これをもとに回帰係数を計算する方法である。一変数の場合には,J は直接ユールウオーカー方程式を解くことができるが、ここでは, 多変量自己回帰係数を求めるアルゴリズムの理解のために, 作成した。

6.1　一変数

[0 次]

$\mathrm{K}_0 = C_0$

[1 次]

$K_1 = \frac{C_1}{V_0}$

$reg_1 = K_1$

[2 次]

$K_{2(2)} = \frac{1}{V_1}(C_2 - K_1 C_1)$

$K_{2(1)} = K_{1(1)} - K_{2(2)} K_{1(1)}$

$reg_2 = K_{2(1)} + K_{2(2)}$

$V_2 = V_1(1 - (K_{2(2)})^2)$

[3 次]

$K_{3(3)} = \frac{1}{V_2}\left(C_3 - \sum\left(\begin{array}{c} K_{2(1)}C_2 \\ K_{2(2)}C_1 \end{array}\right)\right)$

$K_{3(1)} = K_{2(1)} - K_{3(3)} K_{2(2)}$

$K_{3(2)} = K_{2(2)} - K_{3(3)} K_{2(2)}$

$reg_3 = K_{3(1)} + K_{3(2)} + K_{3(3)}$

$V_3 = V_2(1 - (K_{3(3)})^2)$

[4 次]

$K_{4(4)} = \frac{1}{V_4}\left(C_4 - \sum\left(\begin{array}{c} K_{3(1)}C_3 \\ K_{3(2)}C_2 \\ K_{3(3)}C_1 \end{array}\right)\right)$

$K_{4(1)} = K_{3(1)} - K_{4(4)} K_{3(3)}$

$K_{4(2)} = K_{3(2)} - K_{4(4)} K_{3(2)}$

$K_{4(3)} = K_{3(3)} - K_{4(4)} K_{3(1)}$

$reg_4 = K_{4(1)} + K_{4(2)} + K_{4(3)} + K_{4(4)}$

$V_4 = V_3(1 - (K_{4(4)})^2)$

```
   C
9.33354 7.87931 4.50071 0.512286 _2.90688 _4.87683 _5.10031 _3.82648

   5 levin C
       0         0          0          0         0  9.33354
0.844193         0          0          0         0  2.68189
 1.52126 _0.802026          0          0         0 0.956775
 1.50865 _0.778118 _0.0157164          0         0 0.956538
 1.50777 _0.821906  0.0691834 _0.0562752         0 0.953509
 1.51126 _0.826198   0.120175  _0.149817 0.0620402 0.949839
```

右の欄は分散 σ^2。回帰係数は, 上から 0 次, １次.. ５次（=x.) まで同時に求めている。

6.1.1　参考文献

廣田薫・生駒哲一「確率過程の数理（数理工学基礎シリーズ)] 朝倉書店　２００１

6.2 Householder 法

自己回帰 Autoregression	aregb [回帰係数を求める] areg [平均値・決定係数・AIC 値・回帰係数]	m aregb n m areg n m 次数 n Data

ハウスホルダー法による AR　ARMA は鈴木教授がスクリプトを公開しておられるので, 簡単に再掲した。

```
   3 aregc tmp3
750.25 0.946196 341.705 1.37654 _0.236394 _0.221991
```

平均値　決定係数 AIC 値　回帰係数

Yule-Walker 法との比較

```
   3 yw tmp3
           1 _1.01918e_13 _1.01252e_13    1.09991
 1.84741e_13            1   2.22045e_13 0.0260682
_1.05693e_13 _1.24345e_13             1 _0.238416

Householder f(x)= 1.37654Xt-1 _0.236394Xt-2 _0.221991Xt-3
Yule-Walker f(x)=1.09991Xt-1 + 0.0260682Xt-2 -0.238416Xt-3

NB.  n=48 monthly values of a Common Stock Price
NB.   data from  DeLurgio   SERIESB.DAT
```

6.2.1 Script

```
NB. ======================================
NB. Auto Regression by G. suzuki
NB. =====================================
adep=:[}.]-(+/%#)@]
aindep=: 4 : 0
|:(0,<:x.)}.((x.),<:#y.)$y.-(+/%#)y.
)

aregb=:([adep])%.[aindep]
aregc=: 4 : 0
y=:(k=:x.)}.t=:y. - m=:(+/%#)y.
x=:|:(0,<:k)}.(k,<:n=:#t)$t
q=:+/*:y-x+/ . * b=:y%.x
c=:1-q%+/*:y-(+/%#)y
m,c,((n*^.q%n=:n-k)++:k),b
NB. ave r2 AIC ar
)
```

7　自己回帰移動平均：ARMA

鈴木教授の ARMA のスクリプトの再掲

```
NB. ================================================
NB. ARMA
NB. script written by Suzuki , 30 Sept. 2000
NB. ===============================================

mav=:(([:+/[+/\"_1(}:,:}.)@])%2:*[)‘(+/\%[)@.(2:|[)
sac=: 3 : 0
NB. Seasonal Adjustment Coefficient
y=. (+/12+/\"_1(}:,:}.)y.)%24 NB. moving average
(+/%#) ((<:(#y.)%12),12)$(6}. _6 }.y.) %y
)

aregm=: 4 : 0
NB. autoregressive model
b=. (p=. x.}.t)%.q=.}: x. ]\t=.y.-m=.(+/%#)y.
c=.1-(r=.+/*:p-q+/ . * b)%+/*:p
m,c,(((#t)*^.r%#p)++:x.),b
)

trend=: 4 : 0
b=. x. (}.%.[: }: ]\)t=. y-m=.(+/%#)y=.}.y.
'r t'=. '';(-x.){.t
while. (#r)<{.y.
 do. r=. r, m+q [t=. (}. t),q=.+/b*t
end.
)
```

```
predict=: 4 : 0
NB. prediction by ARMA MODEL
z=.(+/12+/\"_1(}: ,: }.)y=.}. y.) % 24
s=.(+/%#)((<:(#y)%12),12)$(6}. _6}. y)%z
(k$s)*x. trend (k=. {. y.),z
)
```

8　相互相関関数　Cross-Correlation Coefficience

8.1　CCF(k) crosscorreration function

$$CCF(k) = \frac{\Sigma_{t=1+k}^{n}(Y_t - \bar{Y})(X_{t-k} - \bar{X})}{\sqrt{\Sigma_{t=1}^{n}(Y_t - \bar{Y})^2 \Sigma_{t=1}^{n}(X_t - \bar{X})^2}}$$

または,

$$COV_{xy}(k) = \frac{1}{n} \sum_{t=1+k}^{n}(Y_t - \bar{Y})(X_{t-k} - \bar{X}), k \geq 0$$

$$COV_{xy}(k) = \frac{1}{n} \sum_{t=1-k}^{n}(Y_{t-k} - \bar{Y})(X_t - \bar{X}), k \geq 0$$

$$CCF(k) = \frac{COV_{xy}(k)}{S_x S_y}$$

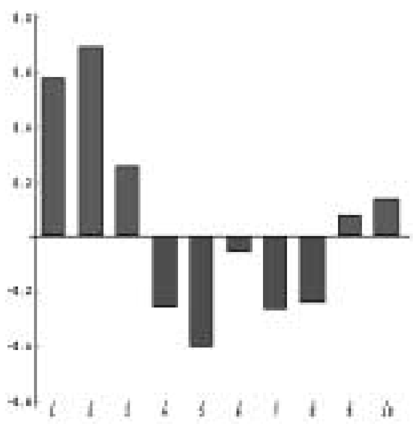

図 6: Cross-Correlation NY-LONDON Stock price

$$\rho_{xy}(k) = \gamma_{xy}(k) / \sqrt{\gamma_{xx}(0)\gamma_{yy}(0)}$$

$$\gamma_{xy}(-k) = E[(X_t - \mu_x)(Y_{t-k} - \mu_y)]$$

8.2

相互相関		

8.2.1 解説

いくつかの時系列のデータを相互にラグを取った場合の相関係数である。時間の推移とともに相関の強弱が変化する過程が plot によくあらわされる。

IBM 株の NY London 取引所のデータの最初の１０ピース（１９９３）

```
 Cross Correlations between NY & London  IBM Stock Prices

 NY     London
38.625   56.5
38.879 57.625
39.125     59
39.375   59.5
40.063   58.5
39.375 58.875
39.438  59.25
39.625 58.625
 39.25 58.125
39.063  58.75
```

```
   ccf q
-5|   0.580839   0.691107  0.259779 _0.257538 _0.404517
 0|  _0.0553644 _0.266541 _0.242612  0.0788315 0.136927
```

```
      ccf2 dat
      0              1                  2                   3
+-----------------+-----------------+--------------------+--------------------
|0.455316 0.580839|0.165699 0.691107|0.00810851  0.259779|_0.0459359 _0.257538|
|0.580839  2.19628|0.148197 0.660269|_0.0206628 _0.518274| _0.173093 _0.305578|
```

```
+-----------------+-----------------+--------------------+--------------------
 NY 自己相関   LN -k
 LN   +k   LN   自己相関
```

8.2.2 Script

```
ccf=: 3 : 0
NB. CCF  (2 variate )
dev=: -"1 (mean=: +/ % #)
C0=: mean y.
C1=: ( <\. 0{"1 y.) ,.  |.@(<\) 1{"1 y.
C2=:((0{"1 C1) - (L:0) 0{C0),.((1{"1 C1) - (L:0) 1{C0)
NB.Cov xy
C4=: ( +/ "1 */ "2 >C2) % # y.
NB. Sx * Sy (use dev)
NB. C5=:*/"1 > %: L:0 tmp_0=: ([: +/  ^&2)  L:0 C1   %  L:0  ( # L:0 C1)
C5=: */ %: (([: +/ ^&2) dev y.) % # y. NB. (devide by n type)
NB. C5=: %: +/ */ "1 y.
C4 %   C5
)

ccf2=: 3 : 0
NB. Correlation coefficience   (multi variate )
NB. divide Sxx Syy
NB. Usage: ccf2 n (data matrix is tate type)
C2=: (dev=: - "1 +/ % #) y.
C1=: (|. <\  C2) ,:  <&|: \.  C2
C4=:(1{C1) +/ . * (L:0) 0{C1
NB. C5=:*/  %: (   +/  ^&2 y.) % # y.
```

```
C5=:*/  %: (   +/  ^&2 C2) % N=: # y.  NB. each SD
( C4 % L:0 N) % L:0 C5 NB. Cov / SD
)

ccf3=: 3 : 0
NB.Covariance coefficient  (multi variate )
NB. divide only by n
NB. Usage: ccf3 n (data matrix is longitude type)
C2=: (dev=: - "1 +/ % #) y.
C1=: (|. <\  C2) ,:  <&|: \.  C2
C4=:(1{C1) +/ . * (L:0) 0{C1
C4 % L:0 0{# y.
)
```

9　多変量自己回帰・VAR

自己回帰 (Autoregression) を多変量に拡張した多変量自己回帰の方法にも, ユルウオーカー法、バーク法、ハウスホルダー法がある。

9.0.3　ユールウオーカー法・Yule-Walker Method

$C(m) = \frac{1}{N}\sum_{N=m+1}^{N} x(n)x(n-m)'$

C(m) は自己共分散関数の推定値で $k \times k$ 行列。

Yule-Walker 方程式は次のようになる。

$$\begin{bmatrix} C(0) & C(-1) & \dots & C(1-M) \\ C(-1) & C(0) & \dots & C(2-M) \\ \dots & & & \\ C(1-M) & C(2-M) & \dots & C(0) \end{bmatrix} \begin{bmatrix} A(1) \\ A(2) \\ \dots \\ A(M) \end{bmatrix} = \begin{bmatrix} C(1) \\ C(2) \\ \dots \\ C(M) \end{bmatrix}$$

9.0.4　経過と説明

```
dat=: ?. 10 2 $ 20
```

相互共分散行列の生成

```
 ccf3 dat
+----------+----------+-----------+-----------+------------+
|34.61   11|4.631 _9.57|_2.128 _3.11|_0.137 _1.53|_10.116 11.81|
|   11 27.2|_2.28 _15.5|  3.39   5.2| _4.74  _8.7|   4.64     8|
+----------+----------+-----------+-----------+------------+
```

`Yule-Walker　Matrix` **の生成**

```
 3 mgmain dat
+-----------+----------+-----------+-----------+
|34.61   11  |4.631 _9.57|_2.128 _3.11|4.631 _9.57 |
|   11 27.2  |_2.28 _15.5|  3.39   5.2|_2.28 _15.5 |
+-----------+----------+-----------+-----------+
```

```
|4.631 _9.57 |34.61    11 |4.631 _9.57 |_2.128 _3.11|
|_2.28 _15.5 |    11 27.2 |_2.28 _15.5 |  3.39    5.2|
+------------+-----------+------------+------------+
|_2.128 _3.11|4.631 _9.57|34.61    11  |_0.137 _1.53|
|  3.39    5.2|_2.28 _15.5|    11 27.2  | _4.74  _8.7|
+------------+-----------+------------+------------+
```

9.0.5 **レビンソンのアルゴリズム**・Levinson Algolism

J はマトリックスを要素とするマトリクスの逆行列をいきなり求めることが出来ない。

多変数の Yule-Walker 方程式の解法として, レビンソンのアルゴリズム（多変数版）があり、一変数と同様に偏相関係数を求めることにより, 高次の回帰係数を順次求めることができる。

(1) 初期値の設定 ($m = 0$)

$W_0 = Z_0 = C(0)$

$AIC(0) = Nlog|W_0|$

(2) レビンソンのアルゴリズム ($m = 1, 2, \ldots, M$)

$E(m) = C(m) - \sum_{i=1}^{m-1} A_{m-1}(i)C(m-i)$

$$A_m(m) = E(m)Z_{m-1}^{-1}$$

$$B_m(m) = E(m)^T W_{m-1}^{-1}$$

$A_m(i) = A_{m-1}(i) - A_m(m)B_{m-1}(m-i), (i = 1, \ldots, m-1)$

$B_m(i) = B_{m-1}(i) - B_m(m)A_{m-1}(m-i), (i = 1, \ldots, m-1)$

$$W_m = C(0) - \sum_{i=1}^{m} A_m(i)C(i)^T$$

$$Z_m = C(0) - \sum_{n=1}^{m} B_m(i)C(i)$$

$$AIC(m) = Nlog|W_m| + 2mk^2$$

9.0.6　経過と結果

[0 次]

$W_0 = Z_0 = C(0)$

[1 次]

$E(1) = C(1)$

$A_1(1) = E(1)Z_0^{-1}$

$B_1(1) = E(1)^T W_0^{-1}$

$W_1 = C(0) - A_1(1)C(1)^T$

$V_1 = C(0) - B_1(1)C(1)$

[2 次]

$E(2) = C(2) - A_1(1)C(1)$

$A_2(2) = E(2)Z_1^{-1}$

$B_2(2) = E(2)^T W_1^{-1}$

$A_2(1) = A_1(1) - A_2(2)B_1(1)$

$B_2(1) = B_1(1) - B_2(2)A_1(1)$

$$W_2 = C(0) - \sum \begin{pmatrix} A_2(1)C(1)^T \\ A_2(2)C(2)^T \end{pmatrix}$$

$$V_2 = C(0) - \sum \begin{pmatrix} B_2(1)C(1) \\ B_2(2)C(2) \end{pmatrix}$$

[3 次]

$$E(3) = C(3) - \sum \begin{pmatrix} A_2(1)C(2) \\ A_2(2)C(1) \end{pmatrix}$$

$A_3(3) = E(3)Z_2^{-1}$

$B_3(3) = E(3)^T W_2^{-1}$

$$\begin{pmatrix} A_3(1) = A_2(1) - A_3(3)B_2(2) \\ A_3(2) = A_2(2) - A_3(3)B_2(1) \end{pmatrix}$$

$$\begin{pmatrix} B_3(1) = B_2(1) - B_3(3)A_2(2) \\ B_3(2) = B_2(2) - B_3(3)A_2(1) \end{pmatrix}$$

$$W_3 = C(0) - \sum \begin{pmatrix} A_3(1)C(1)^T \\ A_3(2)C(2)^T \\ A_3(3)C(3)^T \end{pmatrix}$$

$$V_3 = Z_3 = C(0) - \sum \begin{pmatrix} B_3(1)C(1) \\ B_3(2)C(2) \\ B_3(3)C(3) \end{pmatrix}$$

[４次]

$$E(4) = C(4) - \sum \begin{pmatrix} A_3(1)C(3) \\ A_3(2)C(2) \\ A_3(3)C(1) \end{pmatrix}$$

$A_4(4) = E(4)Z_3^{-1}$
$B_4(4) = E^T(4)^T W_3^{-1}$

$$\begin{pmatrix} A_4(1) = A_3(1) - A_4(4)B_3(3) \\ A_4(2) = A_3(2) - A_4(4)B_3(2) \\ A_4(3) = A_3(3) - A_4(4)B_3(1) \end{pmatrix}$$

$$\begin{pmatrix} B_4(1) = B_3(1) - B_4(4)A_3(3) \\ B_4(2) = B_3(2) - B_4(4)A_3(2) \\ B_4(3) = B_3(3) - B_4(4)A_3(1) \end{pmatrix}$$

$$W_4 = C(0) - \sum \begin{pmatrix} A_4(1)C(1)^T \\ A_4(2)C(2)^T \\ A_4(3)C(3)^T \\ A_4(4)C(4)^T \end{pmatrix}$$

$$V_4 = Z_4 = C(0) - \sum \begin{pmatrix} B_4(1)C(1) \\ B_4(2)C(2) \\ B_4(3)C(3) \\ B_4(4)C(4) \end{pmatrix}$$

9.1　TIMSAC **のデータで**

統計数理研究所の時系列解析プログラム TIMSAC が公表されている。TIMSAC の多変量時系列解析のサンプルデータを用いて見よう。(n=500)

```
  adat2
713 5508 652
709 5511 650
709 5509 686
710 5506 697
712 5503 701
714 5502 706
716 5500 709
```

```
716 5499 710
718 5499 710
720 5496 708
721 5491 713
724 5490 714
726 5485 714
728 5484 735
730 5482 736

(n=  500)
```

自己相関係数

```
 8.5 ": L:0  ccf2 adat2
+-----------------------+-----------------------+-----------------------+
| 0.01343 0.00407 0.01425| 0.01325 0.00395 0.01387| 0.01284 0.00384 0.01342|
| 0.00407 0.02163 0.00098| 0.00422 0.02145 0.00057| 0.00438 0.02120 0.00112|
| 0.01425 0.00098 0.10175| 0.01449 0.00161 0.06427| 0.01432 0.00189 0.00243|
+-----------------------+-----------------------+-----------------------+...


   4   varmain   ccf2   adat2
+-----------------------+-----------------------+-----------------------+
| 1.68205_0.00067 0.00241|_0.75117 0.01197_0.00056| 0.07704_0.00961 0.00140|
|_0.02651 1.09003_0.00637| 0.01613 0.01840 0.00937| 0.03608 0.03156 0.00262|
| 0.57314_0.19702 0.91130| 0.01740 0.12938_0.51502| 0.82392_0.55123_0.22514|
+-----------------------+-----------------------+-----------------------+
| 1.66845_0.02353 0.00439|_0.73853 0.01990_0.00389| 0.08229 0.02055 0.00249|
|_0.04814 1.11082 0.01118| 0.06069 0.00909 0.00293|_0.03334 0.02255_0.00886|
| 1.33946_0.86264 0.90411|_0.78035 0.12683_0.50662|_0.08226_0.00679_0.21875|
+-----------------------+-----------------------+-----------------------+
| 0.00018 0.00000_0.00005| 0.00018_0.00001 0.00011|                       |
| 0.00000 0.00032 0.00001|_0.00001 0.00033 0.00017|                       |
```

```
|_0.00005 0.00001 0.02935| 0.00011 0.00017 0.02905|                       |
+-----------------------+-----------------------+-----------------------+
```

上段　Amm　**３次まで掲載**

中断　Bmm　３**次まで掲載**

下段　W　V

4 **次の多変量自己回帰の推定値の算定**

```
4 reg_v3 adat2
```

```
...
1435.04 10973.7 1449.99
 1424.5 10964.7 1449.02
1429.35 10964.4 1448.25
 1468.2 10962.9 1447.96
1383.21 10972.4 1440.14
1389.19 10975.8 1437.75
1359.61 10973.5 1434.74
1412.03 10970.3 1435.39
1427.85 10967.2 1436.14
1431.04   10966  1441.6
1437.29 10964.2 1439.65
 1441.7 10963.8  1441.8
1442.97 10963.8 1444.66
 1444.9 10960.9 1442.98

(n=495)
```

9.1.1　**参考文献**

北川源四郎「多変量時系列モデル£・時系列解析の方法　朝倉書店　1998

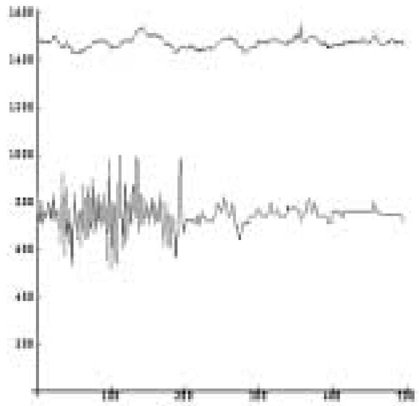

図 7: 変数１ー実数と回帰

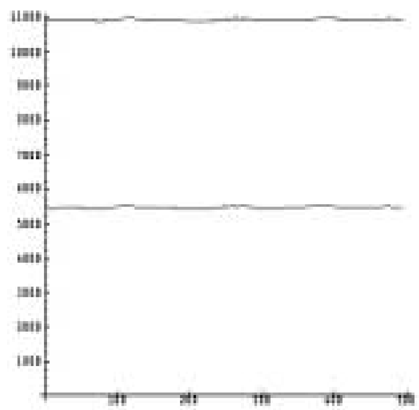

図 8: 変数２ー実数と回帰

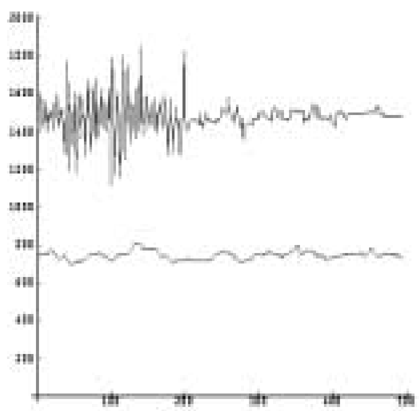

図 9: 変数３ー実数と回帰