$$\left(\sum p_{ij}\delta_{j} - D_i\right)^2 \quad i = 1, \dots N$$ to be minimized w.r.t. δ_K

$$2\, p_{iK}\left(\sum p_{ij}\delta_j - D_i\right) = 0$$

$$p_{iK}\sum p_{ij}\delta_j = p_{iK} D_i$$

Summation over all i, if w_i are the weights

$$\sum_i w_i p_{iK} \sum_j p_{ij}\delta_j = \sum_i w_i p_{iK} D_i \qquad K = 1, \dots M$$

$$[w_1 D_1 \cdots w_N D_N]_{1\times N} \begin{bmatrix} p_{1K} \\ \vdots \\ p_{NK} \end{bmatrix}_{N\times M}$$

~~Symmetric matrix~~

$$\sum_j \delta_j \cancel{\sum_i p_{iK} p_{ij} w_i}$$

$$\sum p_{ij}$$

$$[\ \ ]_{1\times M} \begin{pmatrix} \cancel{p_{K1} \quad p_{KN}} \end{pmatrix}_{M\times N} \begin{bmatrix} \\ \\ \end{bmatrix}$$

$$\sum_i p_{iK} w_i \sum_j p_{ij}\delta_j = \sum_i w_i p_{iK} D_i$$

$$\sum_i p_{iK} p_{ij} w_i = \sum_i p_{iK} w_i D_i$$

1 Jan 89

$(F - G)^2$ to be min.

$$2(F-G)\frac{\partial}{\partial m}(F-G) = 0$$

$$= 2(F-G)\cancel{\frac{\partial}{\partial m}}\left(0 - \frac{\partial G}{\partial m}\right) = 0$$

$$\therefore\ 2(F-G)\frac{\partial G}{\partial m} = 0$$

$$\therefore\ \frac{\partial G}{\partial m} = 0$$

$\sum w_i (F_i - G_i)^2$ to be min.

$$\cancel{2\sum w_i} \quad \cancel{}2\sum w_i (F_i - G_i)\left(0 - \frac{\partial G_i}{\partial m}\right) = 0$$

$$\therefore\ -\cancel{2}\sum w_i (F_i - G_i)\frac{\partial G_i}{\partial m} = 0$$

$$\frac{\partial G_i}{\partial m} = (H$$

N

$$P_{N\times M}\cdot\Delta_{M\times 1}=D_{N\times 1}$$

$$P'_{M\times N}\cdot P_{N\times M}\cdot\Delta_{M\times 1}=P'_{M\times N}\cdot D_{N\times M}$$

$$\left(P_{N\times M}\cdot\Delta_{M\times 1}-D_{N\times 1}\right)\cancel{\left(\Delta'_{1\times M}P\right.}$$

$$\left(\Delta'_{1\times M}P'_{M\times N}-D'_{1\times N}\right)\left(P_{N\times M}\cdot\Delta_{M\times 1}-D_{N\times 1}\right)$$

$$\Delta'_{1\times M}P'_{M\times N}P_{N\times M}\Delta_{M\times 1}-\Delta'_{1\times M}P'_{M\times N}D_{N\times 1}$$

$$-D'_{1\times N}\cdot P_{N\times M}\cdot\Delta_{M\times 1}+D'_{1\times N}\cdot D_{N\times 1}$$

$i=1,\dots N$

$$\Sigma P_{ij}\delta_j=D_i$$

$$\sum_i\omega_i\left(\Sigma P_{ij}\delta_j-D_i\right)^2$$

$$\sum_i P_{ij}\left(\omega_i\Sigma P_{ij}\delta_j-D_i\right)=0$$

$$\sum_i P_{ij}\,w\Sigma P_{ij}\delta_j=\sum_i w_iP_{ij}D_i$$

GAUSS POONA01 9 Sept. 89

1. Interactive Programs

(i) statements between F4 and F2 are executed.

(ii) F1 key brings back the previous screen. {ctrl F1 will bring the previously edited batch pgm}

(iii) ctrl E saves the screen. [as a screen]

---

2. Batch programs

(i) Ascii files.

(ii) run ~~a: pgm~~ a: testprg F4F2

(iii) Use EDIT a: testprg F4F2 to ~~edit~~ edit a program

(iv) Use F2 to run a program being edited. This will also save it

(v) use F1 to save file and come to command mode.

(vi) CTRL F will prompt for a new file name.

(viii) Any variable name preceded by ^ indicates that it is not a literal variable

x = "FIL"

output file = ^x on

3. Synatax

(i) $x = 5$; $Z = \text{rndn}(3,3)$; $Y = x + Z$ (P1) (P2)

is a complete program

(ii) s = " this is a string "

(iii) /* this is a comment */

(iv) let x[2,2] = 1 ~~,~~ 2, 3, 4 ;

(v) save a: xm = x will save the matrix in a file xm.FMT ⟷ IN BINARY

(vi) Load ~~x~~ xp[2,2] = a:xm will load the matrix.

(vii) load ~~Z~~ Z[ ] = a:xm will load it ~~as a column vector. if~~ as a matrix which was saved; if it is xm.FMT

if xm is an ascii file, it will be read as a column vector.

I/O operation

t file = X· ASC ON ;

will be sent to the file.

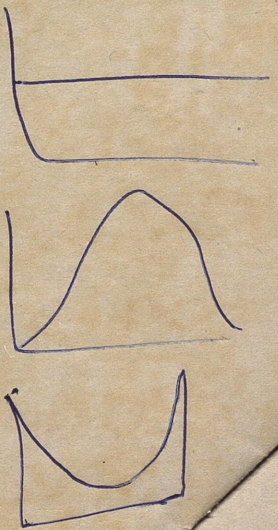
take place

SC RESET ;

file.

the file.

d.



F 8.2

ll print a string.

x, mask, fmt) → X is the matrix
to be printed. N×K

matrix, with 0 for char, 1 for numer

t 1 × 3 matrix, each row specifies format

a colom.

4. I/O operation

(i) output file = X.ASC ON;

all printouts will be sent to the file.

Appending will take place

(ii) output file = X.ASC RESET;

will overwrite the file.

(iii) output off will close the file.

(iv) screen off can also be used.

(v) FORMAT 8, 2 ⇒ F 8.2

(vi) Print $x will print a string.

(vii) PRINTFM (x, mask, fmt) ⟶ X is the matrix to be printed. N x K

mask = matrix, with 0 for char, 1 for numer

fmt 1 x 3 matrix, each row specifies format of colum.

(5) Some useful functions

(i) y = rndu (20, 20) ⟶ 20,20 matrix of random uniform vars

rndn

(ii) y = seqa (start, inc, #entries)

$\quad\quad\quad\quad\quad\quad\quad 1 \quad\; 2 \quad\;\; 5$

gives a column vector 1 3 5 7 9.

(iii) y = ones (5,5)
zeros (4,4)
eye (7)
gives corresponding matrices

(iv) y = rows (x) gives # rows in x

(v) y = meanc (x) ⟶ mean of columns
stdc (x) ⟶ std

maxc ( )

minc ( )

: moment (x, 0) gives moment $\sum_{i}^{n} x_{ij} x_{ik}$

(6) ~~All~~ Matrix operations

(i) $z = x ! y$ will put y below ~~z~~ x

(ii) $z = x \sim y$ " y to the right of x

(iii) Z = ~~R~~RESHAPE(x, r, c)

will make a new matrix with r rows & c cols.

(iv) Z = VEC(x) — makes a column vector

(v) $y = x[1:3, 5:8]$

will make a new matrix with 1–3 rows & 5-8 cols.

(vi) $Y = x .* Z$ — element by element multiplication

(7) Editiy

F5 bgn line

F6 end line

F9 Top of file (screen)

F10 bottom file (screen

cursor movement

^7 Delete line

Printfm contd.

FMT matrix N,1 = format stry "*.*lf"

N,2 = width

N,3 = Precision

E.G.

Y[4,3]= AGE	1	2
PAY	3	5
SEX	7	8
JOB	20	80

mask [1, 3] = 0, 1, 1

Then

FMT (3,3)

= "_*.*s" 8 8

"*.*lf" 10 . 3

"*.*le" 12 4

d = PRINTFM (x, mant, fmt)

will sho

AGE	1.000	2.0000E+00
PA1	3.000	5. E+.
SEX	7.000	
JOB	20.000	

etc.

---

x = cons ← get a string variable.
from KBD

---

y = con(3,2) → get a 3×2 matrix
keyboard.

---

print A B C ;
will print ~~a b c~~ values of A, B C.

8 Some statistical ~~operat~~ computations based on matrices.

$$\cancel{\bar{x}} \;\; \overline{xy} - \bar{x}\,\bar{y}$$

$$\cancel{xy - nx} \quad \frac{\Sigma xy - n\bar{x}\bar{y}}{\cancel{\Sigma x}}$$

(i) ST = date (ST = 4×1 vector)

x = RNDN (1000, 8)

ET = Date

TRQ = TIMEHSEC (ST, ET) / 100.0 .

(ii) Y = RNDNS (Row, col, seed) e.g. IS = 39378

RNDUS (Row, col, seed) Y = RNDNS (5, 10, IS)

(iii) r = POLYROOT (c)

c = n+1 dim. $c1 * x^n + c2\, x^{n-1} + \cdot$ (n+1

r = N×2 vector of roots

(iv) c = POLYCHAR(X)

= Char. polynomial of ~~A~~ square matrix

(v) b = OLSQR (Y, X, name)

Y = N×1 Dep. var.

X = N×K indep. var.

Name = string. (if 'v', the VP & VR will contain predicted values & residuals

bols = K×1 vectr for least sq. estimate of regression.

(vi) (a) If A n×n x = $B_{n\times1}$

Then x = B/A will solve the eqn.

(b) If $A_{10\times5}$ ~~x = B~~ 1×5 ; $A_{n\times2} \cdot x_{2\times1} = b_{n\times1}$

L.S. estimate.

(vii) $Y = SORT(X, k)$ → Y = Matrix x sorted on column k

(viii) $Y = MININDC(X)$ $X = N \times K$

$Y = K \times 1$ matrix, containing index of smallest value.

Also MINC → smallest value
MAXC, MAXINDC

(ix) $Y = INV(X)$ → any
INVPD(X) → symmetric +ve definite.

~~(x)~~ Thus if ~~x~~ $y_i =$ ~~$m_1 x_{i1} + c + m$~~ $m_1 x_{i1} + m_2 x_{i2} + c$

~~We~~ Then $X = [\ ]_{n \times 2}$ matrix, make $X1 = X$ ~ones(3,1)

Then $Z = INVPD(X1'X1) * X1'Y$ will give least sq. values in Z $\begin{matrix} m_1 \\ m_2 \\ c \end{matrix}$

(x) Y = EXEC (pgm, command line)

string string.

eg Y = EXEC("A.EXE", "FILE1")

Y is the exit code

(xi) Y = EIG (X, Q)

EIGSYM( ) → symmetric real.

Y = N×1 or N×2 evals.

Q a string such that if Q = "AA"

Then AAR, AAI will contain real & img. parts of Evals. for EIG ~~SYM~~ ; AA will contain Evec for ~~Q itself contains~~ EIGSYM.

(xii) DOS DIR ↵

DOS ↵ → goes to DOS shell; return to gauss with EXIT.

(xiii) Y = DET(X)

DETL ⇒ determinant of the last matrix selected to.

Fast operation

XI = INV(X)
XD = DETL

} If both inv. & determinant are needed.

~~X~~ CALL CHOL(X)
XD = DETL

} If only det. is needed.

(xiv) ~~E~~Y = CUMPRODC(X)

cummulative product.

also CUMSUMC(X)

(xv) Y = COUNTS(X, V)

creates a distribution.

GAUSS DATASETS 16 Sept. 89

Arranged as matrices, any # rows & max 8190 cols.
Row = observation, cols = variable.
USE SAVEXTOD to store matrix in
memory to Data sets.

USE READDATA to read small
matrices into memory.

~~FIN1 = 'DAT1"~~
ROWSF COLSF

OPEN F1 = DAT1 for READ
X = ROWSF(DAT1)
Y = COLSF(DAT1)
F1 = CLOSE(F1) 0 for success.

OPEN F1 = DAT1
~~R1~~ X = READR(F1, 100)

Use CONVERT

to convert an ASCII file directly

into gauss data set.

```
PROC · MEANCD(DATASET)          LOCAL
  OPEN F1 = ^ dataset           F1, dta, sums, means
                                nr, ~~obs~~
   SUMS = 0
   NR = FLOOR(8190/COLSF(F1))

 DO UNTIL EOF(F1)
    DTA = readr(f1, nr)
   SUMS = SUMS + SUMC(DTA)
    ENDO
   means = SUMS/ROWSF(F1)
   F1 = CLOSE(F1)

  Retp(means)
     endp;
```

CREATE . FP/= MYDATA WITH VAR, 4, 2

will create MYDATA.DAT
MYDATA.DHT

with names VAR1, VAR2, VAR3, VAR4,

and 2 byte integers.

~~X=~~

Q = WRITER(FP, Z)

~~CLOSE~~ FP = CLOSE(FP)

$$\frac{e^{-x_i^{c_1}} - e^{-x_i^{c_2}}}{x_i \ln x_i} \quad Y_i$$

$Y_i = 1 \quad$ or $\quad -1$

$x_i > 1 \qquad >$

$C_2 > C1$

$a \cancel{>} 1$

25 Sep '95

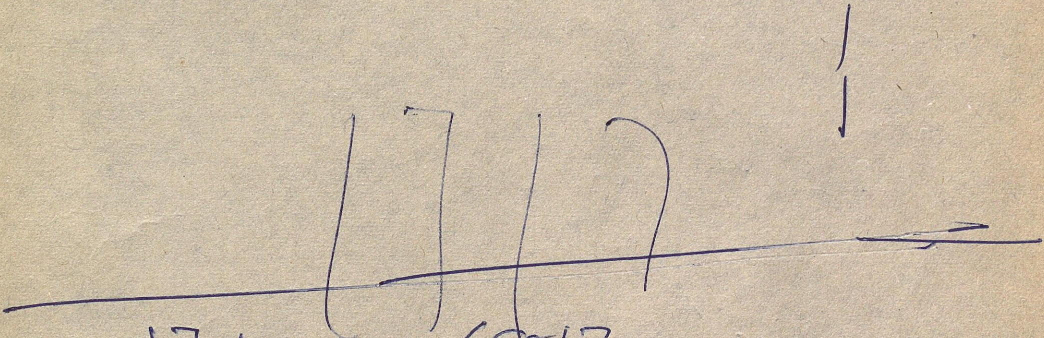
xi —

lxi — ln x

c [2, 1] c1 & c2

$x_1 = c_1 + lxi$

$x_2 = c_2 + lxi$



0.0171 — 6.9547

1

```
CLEAR ~~xI, dxi~~ , ~~x1, x2~~,
      xI , LxI , XLXI,
```

```
LOCAL ~~VN~~ X1, X2, VN, VAL
```

```
load ~~E~~xi[ ] = do1.val
 dxi = ln(xi)
 xdxi = xi .* lxi
```

```
x1 = xi .^X[1,1]
x2 = xi .^X[2,1]
VN = exp(-x1) - exp(-x2)
VAL = VN./xdxi
RETP (VAL)
```

OPEN "COM1: speed, parity, data,
stop, RS, CS n, DS n, CD(n), LF"

AS # filenum [LEN = numb]

speed 75, 110, 300 9600

parity → Space ~~Odd~~ Mark
0 bit 1 bit

E even N none.
del.

data, 4, 5, 6, 7 or 8
del

stop 1 or 2

filenum = 1, 2, et.

~~len~~ num = ~~128~~ del. 128

RS suppress RTS — Request to send

CS 1 ⇒ contro CTS — Clear to send — Con 1 millisecond

DS n contro Data Set Red

– 8 data bits = ~~8 bits~~ N parity

OPEN "COM1:1200, N, 8" as #1

COPEN "COM1: 300, N, 8, 2, CS, DS, CD 1000" as #1

ON ERROR GOTO 0

CLOSE #1

GOTO 100

100 RESUME

e^{-x^c}

$-e^{-x} \cdot c x^{c-1}$ Num $=0$

$x \ln x$. $\ln x - 1$ $\longrightarrow 0$

~~$x=0$~~ $e^{-x^{c_1}}$ $e^{-x^{c_1}} - e^{-x^{c_2}}$

$\dfrac{x \ln x}{}$ $x \to 0$

$x = 0$. Num $= 0$

$x = 1$ Num $= 0$

$e^{-x^{c_1}}$

$1 - x^{c_1} - 1 + x^{c_2}$

Near $x = 0 \rightarrow \dfrac{x^{c_2} - x^{c_1}}{x \ln x}$

TEST1 28/9/89

(i) Use CONVERT
to chnge tdat.VAL into gam data set

(ii) dataset = "TDAT"

(iii) introduce dstatd (dataset, 0)

$Rd = 2$

$x = dta$

$1 - 4x(1-x)$

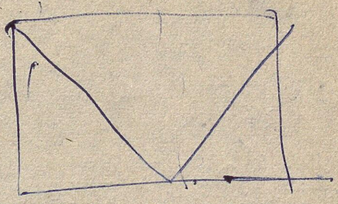
$1 - x + x^2$

$x^2 \mp x + 1$

$\cancel{x^2 = 4x+4}$

$$\int_0 1 - 4x + 4x^2$$

-2 ''



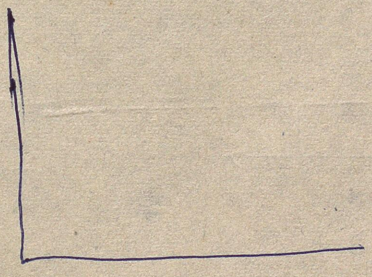
$$\left(x - 2x^2 + \frac{4x^3}{3} \right)$$

0 1 − $1 - 2 + 4/3 =$

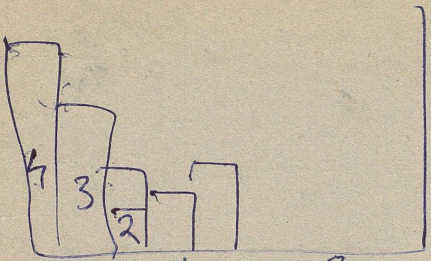
$\cancel{3x(1}$ $3x - 6x^2 + 4x^3 - 1$

$\cancel{3 - 6 +}$

$$\frac{\cancel{8}\int 1 - 2x,}{2x - 1}$$



$(x - x^2)$, $(x^2 - x)$



1 2 3 4 5 6 7 8

4 3 2 1 1 2 3 1 ✓

5 4 2 4 5 ✓

1 2 3 4 5

6 3 2 1 3 6 ✓

1 2 3 4 5 6

6 2 2 2 2 6

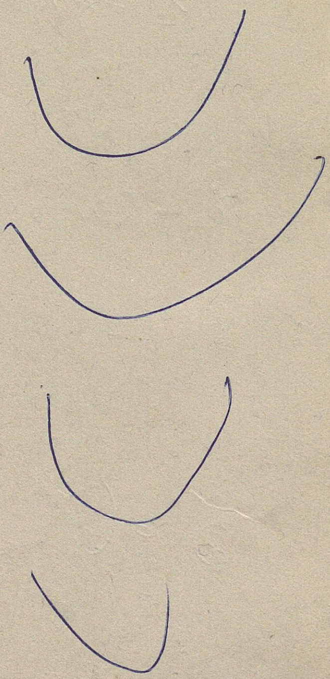
1 2 3 4 5 6

7 2 1 1 2 7

1 2 3 4 5 6

8 2 2 8

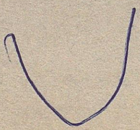
1 2 3 4



Rand U → −0.16

No −0.14

inves noi



3oct 89

```
PROC  MYFUNC(x, a);
retp ( x.^a[.,1] .* exp(x.*x./a[.,2]);
endp.
lb=0   ub=2 ;  a~ -2,13
y= intquad(&myfunc, lb, ub, a, 16)
```

23 Sept. 84

Genetic Code

(i) Non banded Energy ?
Correlation between various properties

(ii) Probability of 1 mutation is p
2 p^2

weighting !

(iii) Distances → not all values independent
check dimensionality of space !

(iv) Degeneracy of code
Monte Carlo calculation for alternative genetic code.

ALA

24 sept 89

26/Sept/8[illegible]

ALA →

Genetic code

A C G U

0 1 2 3

Codon no. = $i*16 + j*4 + K + 1$

48
12
3
1
64

① ② ③

~~IC1/16~~ $i = IC1/16$

$j = IC1 - i*16$

I

DIM IA(64), IB(64), IC(64), KA(20,~~10~~4) KB(~~20,4~~) (20,10)

~~DIMB~~

Y 1

X IA) 2

IABS (~~A~~ 3

Genetic Code 26 Sept 89

(1) How should interconversion Prob. be calculated?

(2) Look at ~~ex~~ M&G dist. between AA who can be interconverted with 0, 1, 2 & 3 mutations.

(3) What is the ~~dt~~ intercov. prob between GLY & GLY. (i.e. any a.a. with itself)

(4) Calculate the corr. between tran. prob. & dist by permuting the properties.

(5) For each AA, calculate the mean charge & st.dev. of charge with 1 mutation.

(6) Do it for each codon?

(7) What are the optimal assignment of STOP

and START codons?

(8) Distances between codons can be 1 2 or 3

Distance between AA can be 0 ⟶ 3

Transform these to $(1/9)^d$, or exp(d) etc.

& look at corr. with distances.

(9) What is done when AA changes to STOP or START codon?

---

(10) Use the average 'stability' of CODONS with the frequency of their use?

(11) Transform the properties ~~to~~ to standard Normal form, and repeat the exercise.

S.G.

Cluster Analysis of rainfall over 59 stations.

Pentads to be divided into 12 months & cluster analysis to be done

Submitted on 13.6 Express ✓ error

Submitted on 19.6 " ✓

Express 25.6 ✓