ISTITUTO NAZIONALE FISICA NUCLEARE

INFN/BE - 75/4
5 Dicembre 1975

N. Grion

SVILUPPO ED ESTENSIONE DEL PROGRAMMA G 200 DEL CERN PER L'ANALISI STATISTICA DI DATI SPERIMENTALI

SERVIZIO RIPRODUZIONE DELLA
SEZIONE DI TRIESTE DELL'INFN

SVILUPPO ED ESTENSIONE DEL PROGRAMMA G 200 DEL CERN

PER L'ANALISI STATISTICA DI DATI SPERIMENTALI

N. Grion

Istituto Nazionale Fisica Nucleare, Sezione di Trieste

1. - INTRODUZIONE

Questo programma si ispira al programma G200, "CERN 7090 programme library", 7865/p/cm. Di questo viene utilizzata la struttura matematica che esegue l'analisi statistica dei dati mentre è stata cambiata quella relativa al tipo di fit da effettuare. Sono stati inoltre conservati i nomi delle variabili, ed infine, le subroutines COVAR e POLYNL sono state utilizzate senza nessuna modifica mentre le altre sono state adattate e trasformate al fine di assolvere al compito che questo programma di calcolo si prefigge. Parte dei risultati finali vengono messi in grafico mediante la subroutine PLOTER completamente rifatta.

2. - SCOPO DEL PROGRAMMA

Il programma costruisce, mediante il metodo dei polinomi ortogonali, la curva che meglio si adatta a un numero n di punti, di coordinate (x_i, y_i), ai quali sia eventualmente assegnato un peso w_i, con $i=1,2,\ldots n$; la varianza di y_i è uguale a σ^2/w_i. Il programma inoltre calcola altri parametri tipici dell'analisi statistica (somma dei quadrati, F ratio, ecc.) che possono essere utilizzati per valutare il tipo di fit (lineare, semilogaritmico, doppiologaritmico) e il grado del relativo polinomio che meglio approssima i punti dati. Infine esso calcola la matrice di covarianza.

3. - STRUTTURA DEL PROGRAMMA

Il programma nel suo complesso è costituito da un programma principale denominato LEAST e dalle subroutines POLYNL, POLFIT, ESTEY, STDDEV, COVAR, PLOTER. I compiti svolti rispettivamente dal programma principale e dalle subroutines si possono così riassumere:

1) Programma principale LEAST: legge i dati in entrata così come sono presentati nel paragrafo seguente e prepara x_i, y_i, w_i in base al tipo di fit scelto per il loro uso nelle routines successive.
2) Subroutine POLYNL: azzera le variabili usate nel polinomio scelto per il fit.
3) Subroutine POLFIT: calcola i coefficienti del polinomio scelto per il fit.
4) Subroutine ESTEY: calcola i valori assunti dal polinomio scelto per il fit in corrispondenza alle y_i.

5) Subroutine STDDEV: calcola gli errori associati ai valori dati y_i.

6) Subroutine COVAR: calcola la matrice di covarianza.

7) Subroutine PLOTER: è la routine che sintetizza il lavoro di tutte le altre in quanto fornisce i dati, opportunamente elaborati, alla routine PLOT che ha il compito di metterli in grafico. La PLOTER usa inoltre altre tre routines di base 7): la AXIS, la PLOTS e la SYMBOL.

Il programma è stato concepito in maniera da permettere il calcolo e la messa in grafico di più fit successivi (teoricamente infiniti) in un solo run: esso continua il suo programma di calcolo fintantochè trova gruppi completi di dati (la definizione di gruppo completo di dati è data nel paragrafo successivo).

4. - DATI IN ENTRATA

Rispettando la successione con cui il calcolatore legge le schede dati, in quanto segue vengono riportate le costanti, il loro formato di lettura, il loro significato ed i valori che possono assumere.

<u>Prima scheda</u>: ICASE;FORMAT (45X,I5)

A seconda del valore che si assegna a ICASE, che può essere 1,2 o 3, il programma esegue il fit dei punti dati con curve che si suppone obbediscano a leggi di tipo polinomiale, esponenziale o di potenza.

ICASE = 1: fit lineare. Dati i punti y_i corrispondenti alle ascisse x_i con i = 1,2,.....,n, il programma costruisce secondo un processo iterativo, un polinomio $y = f(x)$ di grado K ($0 \leq K \leq 21$), che risulti nel massimo accordo coi punti dati relativamente al grado del polinomio K ed al caso scelto(*):

$$f(x) = \sum_{n=o}^{K} a_n x^n \quad \text{con } K = 0,1,.....,21$$

(*) Questo argomento è trattato in dettaglio nel paragrafo seguente. In detto paragrafo $f(x) = \sum_{n=o}^{K} a_n x^n$ è scritta nella forma standard $y^{(j)}(x) = \sum_{r=0}^{j} \Theta_{jr} x^n$ per esigenze formali matematiche.

ICASE = 2: fit semilogaritmico. Ai dati iniziali sostituisce i seguenti $Y_i = \ln y_i$. Quindi costruisce una funzione del tipo $Y = f(x)$, cioè $\ln y = f(x)$, che sia nel massimo accordo con i valori Y. Infine esegue la trasformazione dalla forma logaritmica, $\ln y = f(x)$, alla forma esponenziale $y = \exp(f(x))$. Questo tipo di fit è utile per adattare ai punti dati leggi di tipo esponenziale oppure distribuzioni che hanno analogia con la distribuzione normale.

ICASE = 3: fit doppio logaritmico. Ai dati iniziali sostituisce i seguenti $Y_i = \ln y_i$, $X_i = \ln x_i$. Quindi costruisce una funzione del tipo $Y = f(X)$, cioè $\ln y = f(\ln x)$. Infine esegue la trasformazione dalla forma logaritmica alla forma esponenziale $y = \exp(f(\ln x))$. Questo tipo di fit è utile per adattare ai punti dati leggi di tipo di "potenza". Infatti sia $f(x) = a_o + a_1 X + \dots\ a_n X^n = a_o + a_1 \ln x + \dots\ a_n(\ln x)^n$. Passando alla forma esponenziale si ha $y = \exp(a_o + \dots\ a_n(\ln x)^n)$. Troncando il polinomio al primo grado in lnx si ha:

$y = \exp(a_o)\exp(a_1 \ln x) = A_o x^{a_1}$ con $A_o = \exp(a_o)$. Gli eventuali termini di grado superiore al primo sono legati a quello che è lo scarto dei dati da una vera legge di potenza: tanto più piccoli saranno a_2, a_3, ... tanto più la legge di potenza sarà verificata.

<u>Seconda scheda</u>: NUMBER,KDEG,AP,EQWT,WRITER,IPLOT; FORMAT(2I15,3F15.1,I5).

NUMBER: numero dei dati iniziali $(y_1,\dots,y_n)$.

KDEG: massimo grado del polinomio scelto per il fit (nel caso presente KDEG massimo è uguale a 21).

AP: considera l'eventualità che le coordinate $x_1,\dots,x_n$ siano distribuite secondo una progressione aritmetica o meno. Se le x_i sono in progressione aritmetica allora AP = 0. e la terza scheda deve contenere: FIRSTX, DELTAX; FORMAT(2F20.9).

FIRSTX: valore di x_1.

DELTAX: passo della progressione aritmetica.

Se le x_i non sono distribuite secondo una progressione aritmetica AP = 1. e la terza scheda non è quella con le variabili sopra specificate.

EQWT: peso w_i che deve essere assegnato ad ogni punto sperimentale. Se ogni punto sperimentale è stato misurato con lo stesso grado di accuratezza allora il peso associato ad ogni y_i è sempre lo stesso, in questo caso EQWT = 0., ed il programma automaticamente assegna ad ogni y_i il peso w_i = 1.. In caso contrario EQWT = 1..

WRITER: controlla le scritture dei dati in uscita come specificato nel seguente specchietto a seconda dei valori assegnatigli.

0.: stampa i polinomi e la somma dei quadrati,

1.: stampa come per il valore 0., più i valori predetti di y_i,

2.: stampa come per il valore 1., più i residui,

3.: stampa come per il valore 2., più le deviazioni standard,

4.: stampa come per il valore 3., più la matrice di covarianza.

IPLOT: se si vuol far calcolare e mettere in grafico o solamente far calcolare senza mettere in grafico, i punti sperimentali con i rispettivi errori, il corridoio d'errore (opzionale), la curva che risulta nel massimo accordo con i punti dati ed il sistema d'assi cartesiani (opzionali), allora IPLOT = 1, altrimenti è diverso da 1.

Schede successive: i valori di x_i, y_i, w_i andranno punzonati sulle schede di seguito e fino ad esaurimento degli stessi, secondo il seguente schema:

AP	EQWT	SCHEDE SUCCESSIVE	FORMAT
0.0	0.0	$y_1,y_2,\ldots.y_n$	4F20.9
0.0	1.0	$y_1,y_2,\ldots.y_n;w_1,w_2,\ldots.w_n$	4F20.9;4F20.9
1.0	0.0	$x_1,y_1,x_2,y_2,\ldots.x_n,y_n$	4F20.9
1.0	1.0	$x_1,y_1,w_1,x_2,y_2,w_2,\ldots.x_n,y_n,w_n$	3F20.9

$0 \leq x,y$ minore o uguale al massimo numero contenuto nel formato F20.9.

$0 < w$ minore o uguale al massimo numero contenuto nel formato F20.9.

Valori di $x,y < 0$ possono sempre essere ricondotti al caso di sopra, valori maggiori del massimo ammesso possono essere accettati aumentando il campo del formato.

Se IPLOT = 1 allora bisogna aggiungere ancora cinque schede dati:

Prima scheda: XLENG, XLOW, DELTAX, OX, DIVX; FORMAT(5F16.4).

XLENG: lunghezza totale dell'asse X in centimetri.

XLOW: valore minimo delle x_i sull'asse.

DELTAX: fattore di scala (passo delle x_i per ogni cm. di grafico).

OX: coordinata del punto iniziale dell'asse X.

DIVX: parametro che serve per la subroutine AXIS come specificato in ref. 7).

Seconda scheda: NXTIT, XTITLE; FORMAT(I5,5X,2A10).

NXTIT: numero di caratteri della didascalia sull'asse X.

XTITLE: didascalia (al massimo venti caratteri).

Terza scheda: YLENG, YLOW, DELTAY, OY, DIVY; FORMAT(5F16.4).

Quarta scheda : NYTIT, YTITLE; FORMAT(I5,5X,2A10).

Con analogo significato delle X.

Quinta scheda: PCX, JCP, NOAXIS, NOCOER; FORMAT(F10.2,3I10).

PCX: numero dei punti in cui viene suddiviso l'asse X. In corrispondenza ad ogni x_i con i = 1,....,PCX+1 il polinomio scelto per il fit calcola il rispettivo y_i che poi sarà messo in grafico, quindi tanto maggiore è PCX tanto più continua sarà la curva disegnata.

JCP: passo di NUMBER. 1⩽JCP⩽NUMBER, se JCP = 1 vengono messi in grafico tutti i punti sperimentali (cerchietti) e i rispettivi errori (segmenti), se JCP = NUMBER viene messo in grafico solamente il primo. Se JCP assume un qualsiasi valore intermedio il numero dei punti sperimentali e i rispettivi errori messi in grafico sono la parte intera di NUMBER/JCP+1.

NOAXIS: se è uguale a 1 non vengono tracciati gli assi, se è diverso da 1 sì.

NOCOER: se è uguale a 1 non viene ne tracciato ne calcolato il corridoio d'errore, se è diverso da 1 sì.

Nota: la subroutine PLOTER (IPLOT = 1) può venir chiamata anche nei casi in cui non interessa disegnare la curva (nel qual caso i dati devono venir caricati su nastro) perchè a volte interessano i valori che la funzione assume fuori dall'intervallo in cui cadono i dati sperimentali (la subroutine PLOTER traccia la curva solamente all'interno dell'intervallo). Per chiamare la PLOTER oltre che IPLOT = 1, deve essere WRITER⩾1 (il programma è protetto contro questa ipotesi).

5. - DERIVAZIONE MATEMATICA

5.1 Forma del polinomio di raccordo dei punti dati.

La forma del polinomio di raccordo dei punti dati viene presentata nella seguente forma standard:

$$(5.1) \qquad y^{(j)}(x) = \Theta_{j0} + \Theta_{j1}x + \ldots + \Theta_{jj}x^{j}$$

Fissato K, i coefficienti Θ_{jr} sono stampati sotto la scritta FITTED POLYNOMIAL, per $r = 0,1,\ldots,j$ e per ciascun polinomio di grado $j = 0,1,\ldots,K$.

Il metodo usato per determinare Θ_{jr} è quello dei polinomi ortogonali; il polinomio ortogonale di grado j si può scrivere:

$$(5.2) \qquad \Phi_j(x) = f_{j0} + f_{j1}x + \ldots + x^{j}$$

dove i coefficienti f_{js} vengono calcolati mediante la relazione di ortogonalità pesata:

$$(5.3) \qquad \sum_{i=1}^{n} w_i \Phi_m(x_i)\, \Phi_j(x_i) = \delta_{mj} d_j \qquad 0 \leq m \leq j \qquad 0 \leq j \leq K$$

dove le x_i sono le ascisse degli n punti iniziali.

I coefficienti f_{js} sono stampati sotto la scritta ORTHOGONAL POLYNOMIAL mentre i valori di d_j sono stampati sotto la scritta DIVISOR DJ.

La curva $y^{(j)}(x)$ viene trovata dalla rappresentazione alternativa:

$$(5.4) \qquad y^{(j)}(x) = \sum_{r=0}^{j} c_r \Phi_r(x)$$

I coefficienti c_r si determinano sfruttando la precedente condizione di ortogonalità:

$$(5.5) \qquad c_j \equiv \hat{c}_j = \sum_{i=1}^{n} w_i y_i\, \Phi_j(x_i)/d_j$$

I parametri $\hat{c}_j$ sono stampati sotto la scritta PARAMS CJ.

Vale la seguente formula di ricorrenza:

$$(5.6) \qquad y^{(j+1)}(x) = y^{(j)} + \hat{c}_{j+1}\Phi_{j+1}(x)$$

Essendo

$$\text{Var}(\hat{c}_j) = \sigma^2/d_j \qquad (5.7)$$

è:

$$\begin{aligned} \text{Var}(y^{(j)}(x)) &= \sigma^2 \sum_{r=0}^{j} \Phi_r^2(x)/d_j \\ \text{Var}(y^{(j+1)}(x)) &= \text{Var}(y^{(j)}(x)) + \sigma^2 \Phi_{j+1}^2/d_{j+1} \end{aligned} \qquad (5.8)$$

Sostituendo nella (5.4) al posto di $\Phi_r(x)$ e $\hat{c}_r$ le espressioni date dalle Equazioni (5.2) e (5.5) si ottiene la forma standard

$$y^{(j)}(x) = \sum_{r=0}^{j} \Theta_{jr} x_r \qquad (5.9)$$

dove Θ_{jr} sono funzioni di (x_i, y_i, w_i).

5.2 Determinazione del grado massimo del polinomio

Supponiamo di non conoscere il grado del polinomio che risulti nel massimo accordo con i punti dati. Esso può essere dedotto sulla base dei dati forniti dal programma. Definiamo la grandezza:

$$R_j = \sum_{i=1}^{n} w_i \left[y_i - \sum_{r=0}^{j} \Theta_{jr} x_i^r \right]^2 = \sum_{i=1}^{n} w_i y_i^2 - \hat{c}_o^2 d_o - \ldots - \hat{c}_j^2 d_j \qquad (5.10)$$

dove $j = 0,1,\ldots,K$ e $R_o > R_1 > \ldots > R_K$.

Poniamo $S_j = R_{j-1} - R_j = \hat{c}_j^2 d_j$ che è la somma dei quadrati dei residui quando il grado del polinomio è incrementato da j-1 a j. S_j è stampato sotto la scritta SUM OF SQUARES. Il valore di S_j non è un test significativo anzi può generare delle ambiguità se viene usato per stabilire se è conveniente aumentare il grado del polinomio da j-1 a j: se $S_j = \hat{c}_j^2 d_j$ è "piccolo" tale che $\hat{c}_j$ può essere considerato significativamente prossimo allo zero allora non ha senso aumentare il grado del polinomio da j-1 a j in quanto alla espressione $y^{(j)}(x) = \sum_{r=0}^{j} c_r \Phi_r(x)$ non si aggiunge un contributo significativo; se S_j è così "grande" tale che $R_{j-1} \gg R_j$, ma conteporaneamente R_{j-1} è già "sufficientemente piccolo", allora analogamente al caso precedente, non ha senso incrementare di un'unità il grado del polinomio. Per poter stabilire senza incertezza e con un buon margine di sicurezza il massimo grado del

polinomio che risulti nel massimo accordo con i punti dati bisogna ricorrere alla matematica statistica.
Supponiamo che ciascun y_i abbia una distribuzione con varianza σ^2/w_i. Ciascun S_j ha allora una distribuzione χ^2_1 e quindi R_K ha una distribuzione $\sigma^2\chi^2_{n-K-1}$. R_K è stampato sotto la scritta RES SUM OF SQS assieme al relativo grado di libertà (n-K-1). Il valore $\hat{\sigma}^2 = R_K/(n-K-1)$ è stampato sotto il titolo MEAN SQUARE e fornisce la stima di σ^2. Esso serve inoltre per calcolare $F^{(j)} = S_j/\hat{\sigma}$ che fornisce un criterio per giudicare l'effettiva utilità di incrementare il grado del polinomio da j-1 a j. $F^{(j)}$ ha la distribuzione $F_{1,n-K-1}$ per cui possiamo dire che $\hat{c}_j$ è significativamente diverso da zero se $F^{(j)}$ eccede il valore limite al 95% della distribuzione $F_{1,n-K-1}$. Nella tabella che segue sono riportati alcuni valori di $F^{(j)}$ corrispondenti al valore limite al 95% per certi valori di n-K-1:

n-K-1 =	2	3	4	6	8	12	20	60	120
F(95%) =	18.5	10.1	7.7	6.	5.3	4.7	4.3	4.	3.9

Praticamente per stabilire il miglior valore di KDEG conviene, stabilito ICASE, porre WRITER = 1., e KDEG alto e vedere qual'è il valore di DEG J per cui $F^{(j)}$ sia statisticamente significativo (vedi valori di sopra), ma $F^{(j+1)}, \dots, F^{(K)}$ siano troppo piccole o comunque non eccedano il valore di $F^{(j)}$.

La scelta del tipo di fit da eseguire è dettata dalla conoscenza della legge che regola il fenomeno. Se questa non è conosciuta o se la distribuzione dei punti dati non dà indicazioni utili per determinarla, conviene provare tutti i tre tipi di fit (ICASE = 1,2,3,) con KDEG calcolato come sopra e stabilire dal confronto delle varie grandezze significative che caratterizzano il caso scelto (massimo residuo, andamento dei valori stimati di y_i, deviazioni standard, ecc.) qual'è fra essi il migliore.
Indicando con $\hat{\Theta}_{Kr}$ i valori dei coefficienti Θ_{Kr} ottenuti quando è stato stabilito il grado del polinomio e il tipo di fit, allora:

$$Y_i = \sum_{r=0}^{K} \hat{\Theta}_{Kr} x_i^r = y^{(K)}(x_i) \tag{5.11}$$

Questi valori sono stampati sotto la scritta EST MEAN OF Y(XI). I residui $y_i - Y_i$ sono stampati se WRITER$\geq$2.. Il massimo residuo è:

$$(5.12)\qquad \underset{i}{\mathrm{Max}}\left(y_i - \sum_{r=0}^{j}\hat{\Theta}_{jr}x_i^r\right) \qquad j = 0,1,\dots\dots,K$$

La curva che rappresenta il best fit dei punti iniziali vienne costruita congiungendo i punti Y_i corrispondenti alle ascisse $x_i = x_{i-1}$ + XLENG$\times$ $\times$DELTAX/PCX e x_1 = XLOW dove $2 \leqslant i \leqslant$ PCX+1. Questi valori sono stampati sotto la scritta FITTED CURVE OF THE EXPERIMENTAL DATA se IPLOT = 1 e WRITER $\geqslant$ 1..

5.3 Limite di confidenza

Sia

$$\mathrm{Var}(Y_i) = z_i^2 = \sigma^2\sum_{j=0}^{K}\Phi_j^2(x_i)/d_j \text{ quindi S.D.}(Y_i) = z_i.$$

Il valore $Y_i \pm 2z_i$ dà l'intervallo di confidenza al 95% per il singolo punto $Y_i = E(y_i)$, essendo questo il valore stimato di y_i. I valori di $2z_i$ sono riportati sotto la scritta 2(STD DEV OF EST MEAN OF Y(XI)). Nel grafico questi errori (segmenti verticali) sono associati ai punti dati $y_i = y(x_i)$ (cerchietti).

Il limite di confidenza al 95% della variazione casuale di $y(x_i)$ attorno al suo valore medio Y_i è: $Y_i \pm 2(z_i^2 + \sigma^2)^{1/2}$. I valori di $2(z_i^2 + \sigma^2)^{1/2}$ sono stampati sotto la scritta 2(STD DEV OF Y(XI) ABOUT EST MEAN); questi errori sono associati alla stima dei punti sperimentali $Y_i = y^{(K)}(x_i)$. Nel grafico essi sono rappresentati mediante le due curve che delimitano il corridoiò d'errore le quali sono due spezzate che congiungono i punti $Y_i \pm 2(z_i^2 + \sigma^2)^{1/2}$. Questi valori sono stampati sotto le scritte UPPER CURVE OF THE ERROR CORRIDOR e LOWER CURVE OF THE ERROR CORRIDOR rispettivamente se IPLOT = 1 e WRITER $\geqslant$ 1..

5.4 Matrice di covarianza

Il programma calcola la matrice di covarianza dei coefficienti a_{Ko}, $a_{K1},\dots\dots, a_{KK}$ del polinomio di grado più elevato se WRITER = 4.. Se poniamo $V = (v_{ij})$ allora $\mathrm{Var}(a_{Ki}) = v_{ii}$ e $\mathrm{Cov}(a_{Ki}, a_{Kj}) = v_{ij}$. Essendo una matrice simmetrica di essa verrà stampata solo la parte sottostante alla diagonale principale.

5.5 Relazione tra il numero di punti iniziali n e il grado del polinomio K.

La distribuzione χ^2 con $\sigma > 0$ è definita solo se il numero di gradi di libertà è uguale a 1,2,.... per $x > 0$ (per $x \leqslant 0$ è nulla). Se il numero di punti iniziali è n e K è il grado massimo del polinomio, il numero di gradi di libertà della distribuzione χ^2 è n-K-1. Dovendo essere quest'ultima grandezza maggiore o tutt'al più uguale a 1 deve valere la relazione $n \geqslant K+2$, quindi tutte le considerazioni fatte in questo paragrafo sono valide se è rispettata questa relazione. Se per ragioni particolari si dovesse scegliere il parametro K in maniera tale che $n<K+2$ il programma di calcolo, non potendo utilizzare il procedimento dell'analisi statistica, si limita a calcolare:

a) i coefficienti dei polinomi se WRITER è uguale a 0.;

b) i coefficienti dei polinomi, e i valori stimati dei punti iniziali se WRITER è uguale a 1.;

c) i coefficienti dei polinomi, i valori stimati dei punti iniziali e il valore dei residui se WRITER è uguale a 2.;

quale sia il valore di IPLOT.

Esso calcola e stampa sotto la scritta N/D il grado di libertà e sotto la scritta RES SUM OF SQS il residuo della somma dei quadrati del sistema, qualsiasi sia il valore di WRITER (compatibilmente con i valori che esso può assumere). I valori di WRITER superiori a 2. sono ridotti automaticamente a WRITER uguale a 2..

Il gruppo di schede dei dati relative al caso $n<K+2$ devono sempre essere messe in coda ad altri eventuali gruppi, perchè quando il programma lo incontra esegue i calcoli e li stampa come specificato sopra e poi si ferma definitivamente, ignorando i successivi gruppi di dati.

```
      PROGRAM LEAST(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE4)
C     LEAST SQUARES FIT OF POLYNOMIAL, WITH STATISTICAL ANALYSIS
C
      DIMENSION X(512),Y(512),WT(512)
      COMMON X,Y,WT
      COMMON /COMPLO/ ICASE,NUMBER,IPLOT,XLENG,XLOW,DELTAX,OX,DIVX,
     1NXTIT,XTITLE(2),YLENG,YLOW,DELTAY,OY,DIVY,NYTIT,YTITLE(2),PCX,JCP,
     2NOAXIS,NOCOER
C
    1 READ (5,750) ICASE
      IF(EOF,5)9999,337
  750 FORMAT (45X,I5)
  337 WRITE(6,106)
  106 FORMAT  (1H1)
      WRITE(6,751) ICASE
  751 FORMAT(1X,27H CONSIDERED CASE IS ICASE =,I5)
      READ(5,100) NUMBER,KDEG,AP,EQWT,WRITER,IPLOT
  100 FORMAT(2I15,3F15.1,I5)
      WRITE(6,161) NUMBER,KDEG,AP,EQWT,WRITER ,IPLOT
  161 FORMAT(1H0,1X,8HNUMBER =,I5,4X,6HKDEG =,I5,4X,4HAP =,F6.1,4X,
     16HEQWT =,F6.1,4X,8HWRITER =,F6.1,4X,7HIPLOT =,I5)
      IF(AP-0.5)4,30,30
    4 READ(5,101) FIRSTX,DELTAX
  101 FORMAT(2F20.9)
      WRITE(6,171) FIRSTX,DELTAX
  171 FORMAT(1H0,1X,8HFIRSTX =,F10.3,5X,8HDELTAX =,F10.3)
      READ(5,103)(Y(I),I=1,NUMBER)
  103 FORMAT(4F20.9)
      DO 9 I=1,NUMBER
      X(I)=FIRST X
      FIRST X=FIRST X + DELTA X
    9 CONTINUE
      IF(EQWT.EQ.0.0) GO TO 1001
      READ(5,103) (WT(I),I=1,NUMBER)
      GO TO 573
 1001 DO 999 I=1,NUMBER
      WT(I)=1.0
  999 CONTINUE
      GO TO 573
   30 IF(EQ WT-0.5)21,20,20
   20 READ(5,102) (X(I),Y(I),WT(I),I=1,NUMBER)
  102 FORMAT(3F20.9)
      GO TO 573
   21 DO 1002 I=1,NUMBER
 1002 WT(I)=1.0
      READ(5,103) (X(I),Y(I),I=1,NUMBER)
  573 WRITE(6,181)
  181 FORMAT(1H0)
      WRITE(6,182)(I,X(I),I,Y(I),I,WT(I),I=1,NUMBER)
  182 FORMAT (1X,2(1X,2HX(,I3,2H)=,E11.4,2X,2HY(,I3,2H)=,E11.4,2X,
     13HWT(,I3,2H)=,E11.4,4X))
      DO 99 I=1,NUMBER
      GO TO (701,702,703),ICASE
  703 IF(X(I).EQ.0.0) X(I)=1.E-04
      X(I)=ALOG(X(I))
  702 IF(Y(I).EQ.0.0) Y(I)=1.E-04
      IF(Y(I).EQ.1.0) Y(I)=1.+1.E-04
      Y(I)=ALOG(Y(I))
      IF(EQWT.EQ.0.0) GO TO 99
      WT(I)=WT(I)/Y(I)
      WT(I)=1./WT(I)**2
   99 CONTINUE
  701 IF(IPLOT.EQ.1)  GO TO 50
    5 CALL POLYNL(NUMBER,KDEG,WRITER,ICASE)
      GO TO 1
   50 READ(5,110) XLENG,XLOW,DELTAX,OX,DIVX
```

```
      READ(5,111) NXTIT,XTITLE
      READ(5,110) YLENG,YLOW,DELTAY,OY,DIVY
      READ(5,111) NYTIT,YTITLE
  110 FORMAT(5F16.4)
  111 FORMAT(I5,5X,2A10)
      READ(5,112) PCX,JCP,NOAXIS,NOCOER
  112 FORMAT(F10.2,3I10)
      WRITE(6,114) XLENG,XLOW,DELTAX,OX,DIVX,NXTIT,XTITLE
  114 FORMAT(//,2X,7HXLENG =,F7.2,3X,6HXLOW =,F9.3,3X,8HDELTAX =,F10.4,
     13X,4HOX =,F7.2,3X,6HDIVX =,F5.1,3X,7HNXTIT =,I3,3X,8HXTITLE =,
     22A10)
      WRITE(6,115) YLENG,YLOW,DELTAY,OY,DIVY,NYTIT,YTITLE
  115 FORMAT(1H0,1X,7HYLENG =,F7.2,3X,6HYLOW =,F9.3,3X,8HDELTAY =,F10.4,
     13X,4HOY =,F7.2,3X,6HDIVY =,F5.1,3X,7HNYTIT =,I3,3X,8HYTITLE =,
     22A10)
      WRITE(6,116) PCX,JCP,NOAXIS,NOCOER
  116 FORMAT(1H0,1X,5HPCX =,F10.2,5X,5HJCP =,I5,5X,8HNOAXIS =,I5,5X,
     18HNOCOER =,I5)
      GO TO 5
 9999 IF(IPLOT.EQ.1)  CALL  PLOT(0.,0.,999)
      STOP0000
       END


      SUBROUTINE POLYNL(NUMBER,KDEG,WRITER,ICASE)
      DIMENSION X(512),Y(512),WT(512),RESID(512),U(512),V(512),
     1VAR(512),FITPOL(21),F(21),ORTPOL(21),H(21),SSQS(21),
     2C(21),RESMAX(21),NOMAX(21)
      COMMON X,Y,WT,RESID,U,V,VAR,FIT POL,F,ORT POL,H,S SQS,C,RES MAX,
     1 NO MAX
C
C     ≠RESID(I)≠ HOLDS THE CURRENT DIFFERENCE BETWEEN THE DATA Y(I) AND
C     THE FITTED VALUE, AND IS ALTERED AFTER EACH INCREASE OF THE DEGREE
C     OF THE POLYNOMIAL. ≠U(I)≠ AND ≠V(I)≠ ARE USED IN THE RECURRENCE
C     RELATION TO CALCULATE THE VALUE OF AN ORTHOGONAL POLYNOMIAL IN
C     TERMS OF THE PREVIOUS TWO ORTHOGONAL POLYNOMIALS.
C     ≠VAR(I)≠ CONTAINS THE VARIANCE OF A FITTED Y VALUE, AND INCREASES
C     WITH EACH INCREASE OF DEGREE OF THE FITTED POLYNOMIAL.
      DO 1 I=1,NUMBER
      RESID(I)=Y(I)
      U(I)=1.0
      V(I)=0.0
    1 VAR(I)=0.0
C
C     IN ORDER TO AVOID ZERO SUBSCRIPTS, THE J≠TH POLYNOMIAL IS REFERREDP
C     TO BY THE INDEX (J+1) OR (J+2). ≠FIT POL≠ AND ≠ORT POL≠ REFER TO
C     THE FITTED AND THE ORTHOGONAL POLYNOMIALS RESPECTIVELY. ≠C(J+2)≠
C     IS THE LEAST SQUARES ESTIMATE OF THE BEST MULTIPLE OF THE J≠TH
C     ORTHOGONAL POLYNOMIAL.
      K2 = KDEG + 3
      DO 22 J=1,K2
      FIT POL(J) =0.0
      F(J)=0.0
      ORT POL(J) = 0.0
   22 H(J)=0.0
      ORT POL(2) =1.
      C(1)=0.0
      WRITE(6,101)
  101 FORMAT(62H1POWER        FITTED                ORTHOGONAL      DEGREE    D
     1IVISOR /59H OF   X     POLYNOMIAL             POLYNOMIAL
     2  DJ)
C
      CALL POLFIT(KDEG,WRITER)
C
      RETURN
       END
```

```
      SUBROUTINE POLFIT(KDEG,WRITER)
      DIMENSION X(512),Y(512),WT(512),RESID(512),U(512),V(512),
     1VAR(512),FITPOL(21),F(21),ORTPOL(21),H(21),SSQS(21),
     2C(21),RESMAX(21),NOMAX(21),DINV(21),EL(210)
      COMMON X,Y,WT,RESID,U,V,VAR,FIT POL,F,ORT POL,H,S SQS,C,RES MAX,
     1 NO MAX,DINV,EL
      COMMON /COMPLO/ ICASE,NUMBER,IPLOT,XLENG,XLOW,DELTAX,OX,DIVX,
     1NXTIT,XTITLE(2),YLENG,YLOW,DELTAY,OY,DIVY,NYTIT,YTITLE(2),PCX,JCP,
     2NOAXIS,NOCOER
C
C     ≠DIV≠ IS THE DIVISOR USED TO CALCULATE VAR(C).
      DIV=1.0
      DINV(1)=0.0
      A=0.0
      B=-1.0
      KPLUS=KDEG+1
      DO 60 J=1,KPLUS
      EX=0.0
      EY=0.0
      Z=0.0
      BIG RES =0.0
      NO RES = 0
      DO 61 I=1,NUMBER
      IF(J-1)49,49,6
    6 VAR(I)=VAR(I)+V(I)*V(I)*DINV(J-1)
      RESID(I)=RESID(I)-C(J)*V(I)
      ABS RES= ABS (RESID(I))
      IF(ABS RES - BIG RES)49,49,62
   62 BIG RES = ABS RES
      NO RES = I
   49 W=(X(I)-A)*V(I) - B*U(I)
      U(I)=V(I)
      V(I)=W
      W=WT(I)*V(I)*V(I)
      EX=EX+W
      W=W*X(I)
      EY=EY+W
      W=RESID(I)*WT(I)*V(I)
   61 Z=Z+W
C
C     ≠RES MAX(J)≠ AND ≠NO MAX(J)≠ CONTAIN THE BIGGEST RESIDUAL AND ITS
C     NUMBER AFTER FITTING THE POLYNOMIAL OF DEGREE (J-1). ≠S SQS≠ IS
C     THE REDUCTION IN THE SUM OF SQUARES OF THE RESIDUALS.
    8 A=EY/EX
      B=EX/DIV
      DIV=EX
      RES MAX(J) = BIG RES
      NO MAX(J) = NO RES
      C(J+1)=Z/EX
      S SQS(J) = Z*C(J+1)
      J POWER = J-1
      WRITE(6,101)
  101 FORMAT(1H0)
      DO 66 I=1,J
      NT= I+1
      FIT POL(NT) = FIT POL(NT) + C(J+1)*ORT POL(NT)
      I POWER = I-1
      IF(I-J)9,65,65
    9 WRITE(6,102) IPOWER,FITPOL(NT),ORTPOL(NT)
  102 FORMAT(1H ,I3,2E19.8)
      GO TO 66
   65 WRITE(6,103) IPOWER,FITPOL(NT),JPOWER,DIV
  103 FORMAT(1H ,I3, E19.8,19H     0.10000000E 01,I6,E18.8)
   66 CONTINUE
      DINV(J)=1.0/DIV
      IF(WRITER-3.5)5,5,11
```

```
   11 IF(J-1)3,10,3
C     THE COEFFICIENTS OF THE ORTHOGONAL POLYNOMIALS ARE STORED IN THE
C     ARRAY ≠EL(210)≠, FOR USE IN SUBROUTINE COVAR.
   10 EL(1)=1.0
      GO TO 5
    3 N1=(J*(J-1))/2
      DO 2 I=1,J
      L=N1+I
    2 EL(L)=ORTPOL(I+1)
    5 DO 63 I=1,NT
      NS=I+1
   63 H(NS) = ORT POL(I) - A*ORT POL(NS) - B*F(NS)
      DO 60 I=1,NT
      NS=I+1
      F(NS) = ORT POL(NS)
   60 ORT POL(NS) = H(NS)
      DET=DINV(KPLUS)
C
      CALL EST EY(NUMBER,KDEG,WRITER,KPLUS,DET,ICASE)
C
      IF(IPLOT.EQ.1.AND.WRITER.GE.1.0) CALL PLOTER(J)
      RETURN
      END



      SUBROUTINE EST EY (NUMBER,KDEG,WRITER,KPLUS,DET,ICASE)
      DIMENSION X(512),Y(512),WT(512),RESID(512),U(512),V(512),
     1VAR(512),FITPOL(21),F(21),ORTPOL(21),H(21),SSQS(21),
     2C(21),RESMAX(21),NOMAX(21),DINV(21),EL(210),URCA(512),
     3BURCA(512),WURCA(512),RESTY(512)
      COMMON X,Y,WT,RESID,U,V,VAR,FIT POL,F,ORT POL,H,S SQS,C,RES MAX,
     1NO MAX,DINV,EL,WURCA,BURCA,URCA,RESTY
C
      IF(WRITER - 0.5)43,43,56
   56 WRITE(6,104)
  104 FORMAT(126H1  NO     LOGX(I)      X(I)              LOGY(I)         Y(
     1I)          E.M.LOGY(I)       E.M.OFY(I)       RES.LOGY(I)       RESID.
     2Y(I)   )
      WRITE(6,105)
  105 FORMAT(1H0)
C
C     THE RESIDUAL SUM OF SQUARES IS STORED IN ≠RES SQS≠, AND THE NUMBER
C     OF DEGREES OF FREEDOM IN ≠NDF≠. ≠EST Y≠ IS THE FITTED Y VALUE.
   43 RES SQS = 0.0
      BIG RES = 0.0
      NDF = -KPLUS
      K2=KDEG+2
   14 DO 4 I=1,NUMBER
      RESID(I)=RESID(I) - V(I)*C(K2)
      VAR(I)=V(I)*V(I)*DINV(KPLUS)+VAR(I)
      EST Y = Y(I) - RESID(I)
      RESTY(I)=ESTY
      NDF=NDF+1
      ABS RES = ABS (RESID(I))
      IF(ABS RES - BIG RES)50,50,58
   58 BIG RES = ABS RES
      NO RES = I
   50 IF(WRITER-0.5)4,4,11
   11 GO TO (5,150,151) ICASE
    5 IF(WRITER.LT.2.0) GO TO 222
      WRITE(6,100) I,X(I),Y(I),ESTY,RESID(I)
  100 FORMAT(1H ,I4,12X,E12.4,3(16X,E16.6))
      GO TO 4
  222 WRITE(6,101) I,X(I),Y(I),ESTY
  101 FORMAT(1H ,I4,12X,E12.4,2(16X,E16.6))
```

```
      GO TO 4
  150 WURCA(I)=EXP(Y(I))
      URCA(I)=EXP(ESTY)
      EXPRES=WURCA(I)-URCA(I)
      IF(WRITER.LT.2.0) GO TO 223
      WRITE(6,110) I,X(I),Y(I),WURCA(I),ESTY,URCA(I),RESID(I),EXPRES
  110 FORMAT(1H ,I4,12X,E12.4,6E16.6)
      GO TO 4
  223 WRITE(6,111) I,X(I),Y(I),WURCA(I),ESTY,URCA(I)
  111 FORMAT(1H ,I4,12X,E12.4,4E16.6)
      GO TO 4
  151 WURCA(I)=EXP(Y(I))
      URCA(I)=EXP(ESTY)
      BURCA(I)=EXP(X(I))
      EXPRES=WURCA(I)-URCA(I)
      IF(WRITER.LT.2.0) GO TO 224
      WRITE(6,12) I,X(I),BURCA(I),Y(I),WURCA(I),ESTY,URCA(I),RESID(I),
     ≠EXPRES
   12 FORMAT(1H ,I4,2E12.4,6E16.6)
      GO TO 4
  224 WRITE(6,121)I,X(I),BURCA(I),Y(I),WURCA(I),ESTY,URCA(I)
  121 FORMAT(1H ,I4,2E12.4,4E16.6)
    4 RES SQS = RES SQS + WT(I)*RESID(I)*RESID(I)
      RES MAX(K2) = BIG RES
      NO MAX(K2) = NO RES
C
      CALL STD DEV(NUMBER,KPLUS,WRITER,NDF,RES SQS,ICASE)
C
      RETURN
      END



      SUBROUTINE STD DEV(NUMBER,KPLUS,WRITER,NDF,RES SQS,ICASE)
      DIMENSION X(512),Y(512),WT(512),RESID(512),U(512),V(512),
     1VAR(512),FITPOL(21),F(21),ORTPOL(21),H(21),SSQS(21),
     2C(21),RESMAX(21),NOMAX(21),DINV(21),EL(210),URCA(512),SDY(512),
     3BURCA(512),SDEY(512),WURCA(512),RESTY(512)
      COMMON X,Y,WT,RESID,U,V,VAR,FIT POL,F,ORT POL,H,S SQS,C,RES MAX,
     1NO MAX,DINV,EL,WURCA,BURCA,URCA,RESTY,SDEY,SDY
C
C     ≠AVE SQ≠ IS THE MEAN SQUARE RESIDUAL. ≠SD EY≠ CONTAINS TWICE THE
C     STANDARD DEVIATION OF ≠EST Y≠. THE LATTER IS ESSENTIALLY THE
C     ESTIMATED MEAN OF A RANDOM VARIABLE Y(X). THE RANDOM VARIATION OF
C     Y(X) ABOUT ITS MEAN IS PREDICTED BY ≠SD Y≠.
      IF(NDF-0)17,17,37
   37 AVE SQ = RES SQS/FLOAT (NDF)
      IF(WRITER-2.5)8,8,30
   30 WRITE(6,110)
  110 FORMAT(46H1   NO  2(STD DEV OF EST    2(STD DEV OF Y(XI)/
     1        45H          MEAN OF Y(XI))        ABOUT EST MEAN)/1H )
C
    8 DO 200 I=1,NUMBER
      VAR(I) = AVE SQ * VAR(I)
      SDEY(I)=SQRT(VAR(I))*2.0
      SD Y(I) = SQRT (VAR(I) + AVE SQ) * 2.0
      GO TO (200,201,202),ICASE
  201 SDY(I)=URCA(I)*SDY(I)
      SDEY(I)=WURCA(I)*SDEY(I)
      GO TO 200
  202 SDY(I)=URCA(I)*X(I)*SDY(I)
      SDEY(I)=WURCA(I)*SDEY(I)*X(I)
      SDY(I)=ABS(SDY(I))
      SDEY(I)=ABS(SDEY(I))
  200 CONTINUE
      IF(WRITER-2.5)40,40,50
```

```
   50 WRITE(6,111)(I,SDEY(I),SDY(I),I=1,NUMBER)
  111 FORMAT(1H ,I5,E15.4,E21.4)
   40 WRITE(6,112)
  112 FORMAT(74H1 DEG J    SUM OF SQUARES    F RATIO      PARAMS  CJ
     1MAX RESIDUAL    NO /1H )
      A = 1.0/AVE SQ
C
C     THE ≠F RATIO≠ PROVIDES A STATISTICAL MEASURE OF THE SIGNIFICANCE
C     OF THE GIVEN DEGREE OF THE POLYNOMIAL.
      DO 31 J=1,KPLUS
      J POWER = J-1
      J PLUS = J+1
      F RATIO = A * S SQS(J)
   31 WRITE(6,113) JPOWER,SSQS(J),FRATIO,C(JPLUS),RESMAX(JPLUS),
     1NOMAX(JPLUS)
  113 FORMAT(1H ,I4,E19.6,F12.2,E16.6,E16.6,I5)
   17 WRITE(6,114)
  114 FORMAT(22H0 D/F   RES SUM OF SQS)
      IF(NDF-0)44,44,34
   34 WRITE(6,115)
  115 FORMAT(1H+,25X,29HMEAN  SQUARE           ROOT M.S. )
   44 WRITE(6,117) NDF,RESSQS
  117 FORMAT(1H0,I4,E16.6)
      IF(NDF.LE.0) STOP 1111
   32 ROOT MS = SQRT (AVE SQ)
      WRITE(6,118) AVESQ,ROOTMS
  118 FORMAT(1H+,20X,E17.6,E17.6)
      IF(WRITER-3.5)42,42,2
C
    2 CALL COVAR(KPLUS,AVESQ,ICASE)
C
   42 RETURN
      END


      SUBROUTINE COVAR(KPLUS,AVESQ,ICASE)
      DIMENSION X(512),Y(512),WT(512),RESID(512),U(512),V(512),
     1VAR(512),FITPOL(21),F(21),ORTPOL(21),H(21),SSQS(21),
     2C(21),RESMAX(21),NOMAX(21),DINV(21),EL(210),COV(210),URCA(512),
     3BURCA(512),SDEY(512),WURCA(512),SDY(512),RESTY(512)
      COMMON X,Y,WT,RESID,U,V,VAR,FIT POL,F,ORT POL,H,S SQS,C,RES MAX,
     1NO MAX,DINV,EL,WURCA,BURCA,URCA,RESTY,SDEY,SDY
C
C  COVARIANCE MATRIX COV IS STORED IN ONE DIMENSIONAL ARRAY , I.E.
C  ACROSS THE ROWS OF A LOWER TRIANGLE.  SIMILARLY FOR EL.  DINV IS A
C  DIAGONAL MATRIX STORED IN ONE DIMENSIONAL ARRAY.  THIS ARRANGEMENT
C  SAVES STORAGE SPACE.
C              THEN  COV = EL.DINV.EL
C
      XINDF(I,J)=(I*(I-1))/2+J
      KMAX=(KPLUS*(KPLUS-1))/2+KPLUS
      DO 3 I=1,KMAX
    3 COV(I)=0.0
      DO 1 I=1,KPLUS
      DO 1 J=1,I
      DO 1 L=I,KPLUS
      N=XINDF(I,J)
      N1=XINDF(L,I)
      N2=XINDF(L,J)
    1 COV(N)=EL(N1)*EL(N2)*DINV(L)        +COV(N)
      DO 4 I=1,KMAX
    4 COV(I)=COV(I)*AVESQ
      WRITE(6,10)
   10 FORMAT (40H1COVARIANCE MATRIX FOR FITTED POLYNOMIAL//)
      DO 2 I=1,KPLUS
```

```
      N3=(I*(I-1))/2+1
      N4=N3+I-1
      J=I-1
    2 WRITE(6,20) J,(COV(K),K=N3,N4)
   20 FORMAT (1H I2,(1X,6E20.6))
      RETURN
       END



      SUBROUTINE PLOTER(JJ)
      DIMENSION X(512),Y(512),WT(512),RESID(512),U(512),V(512),
     1VAR(512),FITPOL(21),F(21),ORTPOL(21),H(21),SSQS(21),
     2C(21),RESMAX(21),NOMAX(21),DINV(21),EL(210),WURCA(512),URCA(512),
     3BURCA(512),SDEY(512),SDY(512),RESTY(512)
      COMMON    X,Y,WT,RESID,U,V,VAR,FIT POL,F,ORT POL,H,S SQS,C,RES MAX,
     1NO MAX,DINV,EL,WURCA,BURCA,URCA,RESTY,SDEY,SDY
      COMMON   /COMPLO/  ICASE,NBR,IPLOT,XLENG,XLOW,DELTAX,OX,DIVX,
     1NXTIT,XTITLE(2),YLENG,YLOW,DELTAY,OY,DIVY,NYTIT,YTITLE(2),PCX,JCP,
     2NOAXIS,NOCOER
      DIMENSION BUF(3000),XX(513),YY(513),WW(513)
      INTEGER XTITLE,YTITLE
      CALL PLOTS(BUF(10),2000,4)
      IF(NOAXIS.EQ.1) GO TO 10
      CALL AXIS(0.,0.,XTITLE,-NXTIT,XLENG,0.,XLOW,DELTAX,DIVX)
      CALL AXIS(0.,0.,YTITLE,NYTIT,YLENG,90.,YLOW,DELTAY,DIVY)
   10 FAC=1.
      DO 11 I=1,NBR,JCP
      GO TO(77,88,99)ICASE
   77 XX(I)=(X(I)-XLOW)/DELTAX
      YY(I)=(Y(I)-YLOW)/DELTAY
      WW(I)=SDEY(I)/(2.0*DELTAY)
      GO TO 1111
   88 XX(I)=(X(I)-XLOW)/DELTAX
      YY(I)=(WURCA(I)-YLOW)/DELTAY
      WW(I)=SDEY(I)/(2.0*DELTAY)
      GO TO 1111
   99 XX(I)=(BURCA(I)-XLOW)/DELTAX
      YY(I)=(WURCA(I)-YLOW)/DELTAY
      WW(I)=SDEY(I)/(2.0*DELTAY)
 1111 XP=XX(I)-0.1
      YP=YY(I)+WW(I)*FAC
      CALL PLOT(XP,YP,3)
      XP=XP+0.2
      CALL PLOT(XP,YP,2)
      CALL PLOT(XX(I),YP,3)
      CALL SYMBOL(XX(I),YY(I),0.1,1,0.0,-2)
      FAC=-FAC
      YP=YY(I)+WW(I)*FAC
      CALL PLOT(XX(I),YP,2)
      XP=XX(I)-0.1
      CALL PLOT(XP,YP,3)
      XP=XP+0.2
      CALL PLOT(XP,YP,2)
   11 CONTINUE
      IF(NOCOER.EQ.1) GO TO 20
      WRITE (6,4400)
 4400 FORMAT (1H1)
      WRITE(6,1008)
 1008 FORMAT(48X,33HUPPER CURVE OF THE ERROR CORRIDOR)
      DO 7 K=1,2
      DO 4 I=1,NBR
      GO TO (71,71,72) ICASE
   71 XX(I)=(X(I)-XLOW)/DELTAX
      GO TO 9
   72 XX(I)=(BURCA(I)-XLOW)/DELTAX
```

```
    9 IF (ICASE.EQ.1) GO TO 5
      YY(I)=URCA(I)+SDY(I)/2.0
      GO TO 3
    5 YY(I)=RESTY(I)+SDY(I)/2.0
    3 A=XX(I)
      D=(YY(I)-YLOW)/DELTAY
      IF(I.EQ.1) CALL PLOT(A,D,3)
      CALL PLOT(A,D,2)
    4 CONTINUE
      GO TO(1,7) K
    1 WRITE(6,1012)
      WRITE(6,1013) (I,YY(I),I=1,NBR)
      DO 6 I=1,NBR
      GO TO (61,62,62) ICASE
   62 SDY(I)=-SDY(I)
      GO TO 6
   61 SDY(I) = -2.*SDY(I)
    6 CONTINUE
    7 CONTINUE
      WRITE(6,1012)
      WRITE(6,1009)
 1009 FORMAT(48X,33HLOWER CURVE OF THE ERROR CORRIDOR)
      WRITE(6,1012)
      WRITE(6,1013) (I,YY(I),I=1,NBR)
   20 DX=XLENG*DELTAX/PCX
      XX(1)=XLOW
      LL=PCX+1.
      DO 21 I=2,LL
      XX(I)=XX(I-1)+DX
      IF(XX(I).GT.(-DX/10.).AND.XX(I).LT.(DX/10.))  XX(I)=0.0
   21 CONTINUE
      GO TO (25,26,26) ICASE
   25 BIG=Y(1)
      DO 27 I=2,NBR
      IF(Y(I).GT.BIG)      BIG = Y(I)
   27 CONTINUE
      GO TO 29
   26 BIG = URCA(1)
      DO 28 I=2,NBR
      IF(URCA(I).GT.BIG)   BIG = URCA(I)
   28 CONTINUE
   29 DO 24 I=1,LL
      YY(I)=0.0
      DO 22 J=1,JJ
      IPOWER=J-1
      NT=J+1
      IF(IPOWER.EQ.0)  GO TO 23
      GO TO (237,238,239)ICASE
  237 YY(I)=YY(I)+FITPOL(NT)*XX(I)**IPOWER
      IF(J.EQ.JJ) YY(I)=YY(I)-YLOW
      GO TO 22
  238 YY(I)=YY(I)+FITPOL(NT)*XX(I)**IPOWER
      IF(J.EQ.JJ.AND.YY(I).GT.20.)      YY(I)=20.0
      IF(J.EQ.JJ)    YY(I)=(EXP(YY(I))-YLOW)
      GO TO 22
  239 IF(J.EQ.2.AND.XX(I).LT.1.E-4)      XX(I)=1.E-4
      IF(J.EQ.2)       XX(I)=ALOG(XX(I))
      YY(I)=YY(I)+FITPOL(NT)*XX(I)**IPOWER
      IF(J.NE.JJ) GO TO 22
      IF(J.EQ.JJ.AND.YY(I).GT.20.)      YY(I)=20.0
      YY(I)=(EXP(YY(I))-YLOW)
      XX(I)=EXP(XX(I))
      GO TO 22
   23 YY(I)=YY(I)+FITPOL(NT)
   22 CONTINUE
      IF((YY(I)+YLOW).GT.1.3*BIG)         YY(I)=BIG
   24 CONTINUE
```

6. - ESEMPIO

Consideriamo come punti iniziali quelli di uno spettro di altezza dello ^{90}Sr rivelato mediante uno scintillatore plastico. I dati in entrata vengono stampati come mostrato a pag. 26: questa operazione è utile per controllare che i dati effettivamente letti dal computer (nel nostro caso su schede) siano proprio quelli forniti. Nelle pagine successive sono stampati i risultati del calcolo completo. L'ultima pagina mostra un esempio di rappresentazione grafica. Il disegno grande rappresenta lo spettro dello ^{90}Sr: i cerchietti individuano i punti sperimentali (nel nostro caso dN/dE, E), i trattini verticali rappresentano i relativi errori, la curva mediana il best fit e le due curve estreme delimitano il corridoio d'errore. Gli assi vengono costruiti mediante la subroutine AXIS 7). Il primo grafico piccolo a partire dall'alto è un esempio, in scala ridotta, del caso precedente; l'unica differenza è che ora non vengono rappresentati gli assi e i punti sperimentali sono riportati uno ogni tre (JCP = 3). Per costruire il best fit è stato scelto ICASE = 2, il migliore tra i tre, e KDEG = 4. Il secondo grafico (sempre a partire dall'alto) è lo stesso del caso di sopra a meno del corridoio d'errore che ora non viene rappresentato.

Nel terzo è mostrato come aumenta l'errore associato ad ogni punto sperimentale quando il caso scelto (ICASE = 3) non è consono alla natura intrinsica della distribuzione dei punti sperimentali. Infine l'ultimo grafico rappresenta il best fit degli stessi punti sperimentali quando il grado del polinomio (KDEG) è diverso da quello suggerito dal programma, nonostante che il caso scelto (ICASE = 1) sia accettabile. I cinque grafici sono stati eseguiti in un solo run. L'origine degli assi (anche se non vengono rappresentati) per il primo grafico è fissata dall'operatore.

L' origine degli assi per il secondo grafico è riferita all'origine degli assi del primo ed è specificata nei dati in entrata relativi a questo.

A sua volta l'origine degli assi per il terzo grafico è riferita all'origine degli assi del secondo grafico e così via.

NO	2(STD DEV OF EST MEAN OF Y(XI))	2(STD DEV OF Y(XI) ABOUT EST MEAN)
1	2.1752E+01	3.6777E+01
2	2.4648E+01	6.0583E+01
3	4.1874E+01	9.8457E+01
4	6.3010E+01	1.5192E+02
5	8.7253E+01	2.1719E+02
6	1.1827E+02	2.8620E+02
7	1.2950E+02	3.4939E+02
8	1.2859E+02	3.9868E+02
9	1.2779E+02	4.2913E+02
10	1.3533E+02	4.3899E+02
11	1.3114E+02	4.2916E+02
12	1.2356E+02	4.0250E+02
13	1.1925E+02	3.6337E+02
14	1.0361E+02	3.1699E+02
15	8.5638E+01	2.6853E+02
16	7.4158E+01	2.2227E+02
17	6.0685E+01	1.8108E+02
18	5.2285E+01	1.4633E+02
19	4.1566E+01	1.1815E+02
20	3.6037E+01	9.5843E+01
21	3.0977E+01	7.8402E+01
22	2.7770E+01	6.4802E+01
23	1.8170E+01	5.4239E+01
24	1.3819E+01	4.6248E+01
25	1.2786E+01	4.0838E+01
26	1.8466E+01	3.8693E+01
27	3.0487E+01	4.1332E+01

DEG J	SUM OF SQUARES	F RATIO	PARAMS CJ	MAX RESIDUAL	NO
0	1.100834E+03	89117.94	6.385267E+00	1.556953E+00	1
1	4.281138E+00	346.58	-1.022475E-02	2.221562E+00	1
2	1.432120E+01	1159.37	-5.379841E-04	9.023099E-01	27
3	3.148327E+00	254.87	7.415255E-06	2.291936E-01	24
4	2.138599E-07	.00	-5.744819E-11	2.292913E-01	24

D/F	RES SUM OF SQS	MEAN SQUARE	ROOT M.S.
22	2.717562E-01	1.235255E-02	1.111420E-01

COVARIANCE MATRIX FOR FITTED POLYNOMIAL

0	4.204127E-02				
1	-3.275529E-03	2.819477E-04			
2	7.734282E-05	-7.016711E-06	1.812260E-07		
3	-7.074245E-07	6.625515E-08	-1.756611E-09	1.737200E-11	
4	2.203354E-09	-2.106748E-10	5.693564E-12	-5.718754E-14	1.906251E-16

UPPER CURVE OF THE ERROR CORRIDOR

```
YY(  1) = 1.486656E+02   YY(  2) = 2.722120E+02   YY(  3) = 4.582902E+02   YY(  4) = 7.092978E+02   YY(  5) = 1.011487E+03
YY(  6) = 1.334900E+03   YY(  7) = 1.639433E+03   YY(  8) = 1.884820E+03   YY(  9) = 2.040604E+03   YY( 10) = 2.092503E+03
YY( 11) = 2.043632E+03   YY( 12) = 1.911217E+03   YY( 13) = 1.720696E+03   YY( 14) = 1.499414E+03   YY( 15) = 1.271592E+03
YY( 16) = 1.055426E+03   YY( 17) = 8.622949E+02   YY( 18) = 6.975035E+02   YY( 19) = 5.618090E+02   YY( 20) = 4.531087E+02
YY( 21) = 3.678853E+02   YY( 22) = 3.022514E+02   YY( 23) = 2.525952E+02   YY( 24) = 2.159307E+02   YY( 25) = 1.900873E+02
YY( 26) = 1.738535E+02   YY( 27) = 1.670773E+02   YY(
```

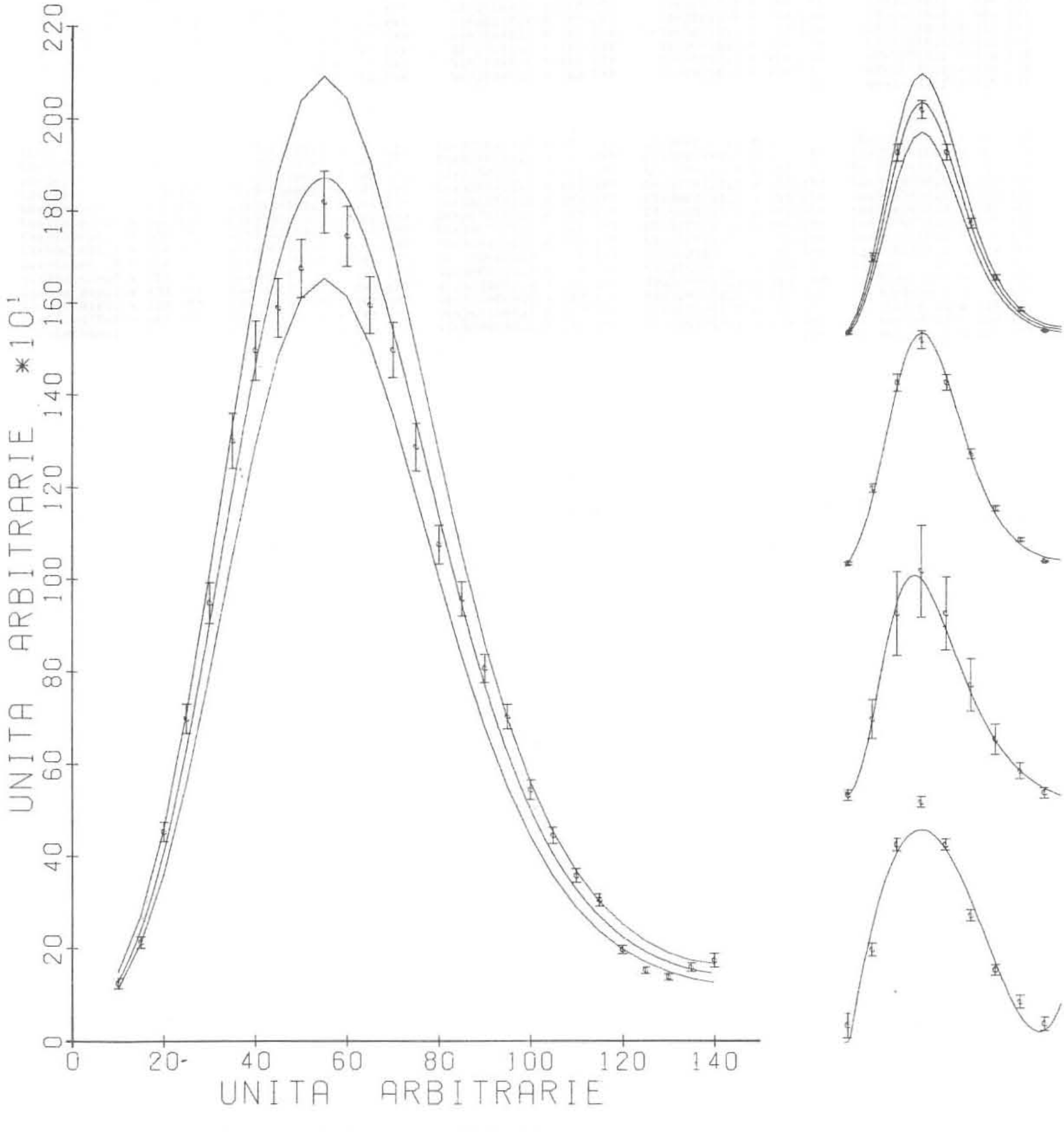
LOWER CURVE OF THE ERROR CORRIDOR

```
YY(  1) = 1.118888E+02   YY(  2) = 2.116288E+02   YY(  3) = 3.598328E+02   YY(  4) = 5.573808E+02   YY(  5) = 7.942945E+02
YY(  6) = 1.048701E+03   YY(  7) = 1.290046E+03   YY(  8) = 1.486138E+03   YY(  9) = 1.611477E+03   YY( 10) = 1.653508E+03
YY( 11) = 1.614472E+03   YY( 12) = 1.508720E+03   YY( 13) = 1.357323E+03   YY( 14) = 1.182419E+03   YY( 15) = 1.003060E+03
YY( 16) = 8.331561E+02   YY( 17) = 6.812140E+02   YY( 18) = 5.511713E+02   YY( 19) = 4.436639E+02   YY( 20) = 3.572661E+02
YY( 21) = 2.894836E+02   YY( 22) = 2.374494E+02   YY( 23) = 1.983565E+02   YY( 24) = 1.696827E+02   YY( 25) = 1.492496E+02
YY( 26) = 1.351608E+02   YY( 27) = 1.257458E.02   YY(
```

FITTED CURVE OF THE EXPERIMENTAL DATA

```
XX(  1) = 0.              XX(  2) = 2.0000E+00      XX(  3) = 4.0000E+00      XX(  4) = 6.0000E+00      XX(  5) = 8.0000E+00
XX(  6) = 1.0000E+01      XX(  7) = 1.2000E+01      XX(  8) = 1.4000E+01      XX(  9) = 1.6000E+01      XX( 10) = 1.8000E+01
XX( 11) = 2.0000E+01      XX( 12) = 2.2000E+01      XX( 13) = 2.4000E+01      XX( 14) = 2.6000E+01      XX( 15) = 2.8000E+01
XX( 16) = 3.0000E+01      XX( 17) = 3.2000E+01      XX( 18) = 3.4000E+01      XX( 19) = 3.6000E+01      XX( 20) = 3.8000E+01
XX( 21) = 4.0000E+01      XX( 22) = 4.2000E+01      XX( 23) = 4.4000E+01      XX( 24) = 4.6000E+01      XX( 25) = 4.8000E+01
XX( 26) = 5.0000E+01      XX( 27) = 5.2000E+01      XX( 28) = 5.4000E+01      XX( 29) = 5.6000E+01      XX( 30) = 5.8000E+01
XX( 31) = 6.0000E+01      XX( 32) = 6.2000E+01      XX( 33) = 6.4000E+01      XX( 34) = 6.6000E+01      XX( 35) = 6.8000E+01
XX( 36) = 7.0000E+01      XX( 37) = 7.2000E+01      XX( 38) = 7.4000E+01      XX( 39) = 7.6000E+01      XX( 40) = 7.8000E+01
XX( 41) = 8.0000E+01      XX( 42) = 8.2000E+01      XX( 43) = 8.4000E+01      XX( 44) = 8.6000E+01      XX( 45) = 8.8000E+01
XX( 46) = 9.0000E+01      XX( 47) = 9.2000E+01      XX( 48) = 9.4000E+01      XX( 49) = 9.6000E+01      XX( 50) = 9.8000E+01
XX( 51) = 1.0000E+02      XX( 52) = 1.0200E+02      XX( 53) = 1.0400E+02      XX( 54) = 1.0600E+02      XX( 55) = 1.0800E+02
XX( 56) = 1.1000E+02      XX( 57) = 1.1200E+02      XX( 58) = 1.1400E+02      XX( 59) = 1.1600E+02      XX( 60) = 1.1800E+02
XX( 61) = 1.2000E+02      XX( 62) = 1.2200E+02      XX( 63) = 1.2400E+02      XX( 64) = 1.2600E+02      XX( 65) = 1.2800E+02
XX( 66) = 1.3000E+02      XX( 67) = 1.3200E+02      XX( 68) = 1.3400E+02      XX( 69) = 1.3600E+02      XX( 70) = 1.3800E+02
XX( 71) = 1.4000E+02      XX(


YY(  1) = 2.789433E+01    YY(  2) = 3.927448E+01    YY(  3) = 5.434855E+01    YY(  4) = 7.394410E+01    YY(  5) = 9.894914E+01
YY(  6) = 1.302772E+02    YY(  7) = 1.688214E+02    YY(  8) = 2.153992E+02    YY(  9) = 2.706907E+02    YY( 10) = 3.351736E+02
YY( 11) = 4.090615E+02    YY( 12) = 4.922485E+02    YY( 13) = 5.842681E+02    YY( 14) = 6.842687E+02    YY( 15) = 7.910114E+02
YY( 16) = 9.028906E+02    YY( 17) = 1.017979E+03    YY( 18) = 1.134092E+03    YY( 19) = 1.248874E+03    YY( 20) = 1.359891E+03
YY( 21) = 1.464739E+03    YY( 22) = 1.561138E+03    YY( 23) = 1.647032E+03    YY( 24) = 1.720670E+03    YY( 25) = 1.780664E+03
YY( 26) = 1.826041E+03    YY( 27) = 1.856255E+03    YY( 28) = 1.871190E+03    YY( 29) = 1.871141E+03    YY( 30) = 1.856768E+03
YY( 31) = 1.829052E+03    YY( 32) = 1.789231E+03    YY( 33) = 1.738736E+03    YY( 34) = 1.679121E+03    YY( 35) = 1.612005E+03
YY( 36) = 1.539010E+03    YY( 37) = 1.461711E+03    YY( 38) = 1.381596E+03    YY( 39) = 1.300036E+03    YY( 40) = 1.218256E+03
YY( 41) = 1.137326E+03    YY( 42) = 1.058156E+03    YY( 43) = 9.814918E+02    YY( 44) = 9.079271E+02    YY( 45) = 8.379092E+02
YY( 46) = 7.717544E+02    YY( 47) = 7.096627E+02    YY( 48) = 6.517335E+02    YY( 49) = 5.979816E+02    YY( 50) = 5.483525E+02
YY( 51) = 5.027365E+02    YY( 52) = 4.609822E+02    YY( 53) = 4.229075E+02    YY( 54) = 3.883102E+02    YY( 55) = 3.569756E+02
YY( 56) = 3.286845E+02    YY( 57) = 3.032181E+02    YY( 58) = 2.803631E+02    YY( 59) = 2.599145E+02    YY( 60) = 2.416790E+02
YY( 61) = 2.254758E+02    YY( 62) = 2.111389E+02    YY( 63) = 1.985173E+02    YY( 64) = 1.874754E+02    YY( 65) = 1.778936E+02
YY( 66) = 1.696685E+02    YY( 67) = 1.627125E+02    YY( 68) = 1.569546E+02    YY( 69) = 1.523402E+02    YY( 70) = 1.488322E+02
YY( 71) = 1.464116E+02    YY(
```


UNITA ARBITRARIE *10¹
0
20
40
60
80
100
120
140
160
180
200
220
0
20
40
60
80
100
120
140
UNITA ARBITRARIE

BIBLIOGRAFIA

1) Forsythe G. "Generation and Use of Orthogonal Polynomials for Data Fitting with a Digital Computer". J. Soc. Industr. and Appl. Math. Vol.5, p. 74 (1957).

2) Plackett R. "Principles of Regression Analysis". Oxford, Claredon Press.

3) Cadwell J. "A Least Square Surface Fitting Programme". Comp. Journal Vol. 3, p. 266 (1961).

4) Cadwell J. and Williams D. "Some Orthogonal Methods of Curve and Surface Fitting" Computer J., Vol. 4, p. 260 (1961).

5) Courrant R. and Hilbert D. "Methods of Mathematical Physics". Vol. 1 Interscience, N.Y., (1952).

6) Cadwell J. and D. Hudson. Programme G200 CERN 7090 Programme Library (Questo programma, a sua volta, è una estensione fatta da D. Hudson del programma R1000 di J. Cadwell).

7) A. Dovier e C. De Focatiis Comunicazione N.9: Aggiornamenti delle routines relative al plotter calcomp 563; Università degli Studi di Trieste, Centro di Calcolo.
M. HMELJAK Communicazione N.4: Uso del tracciatore calcomp plotter 563 comandato dall'unita nastro off-line 760; Università degli Studi di Trieste, Centro di Calcolo.