J.Pharmacobio-Dyn., **8**, 246–256 (1985)

A NONLINEAR MULTIPLE REGRESSION PROGRAM, MULTI2 (BAYES), BASED ON BAYESIAN ALGORITHM FOR MICROCOMPUTERS

KIYOSHI YAMAOKA, TERUMICHI NAKAGAWA, HISASHI TANAKA,* MASATO YASUHARA, KATSUHIKO OKUMURA, AND RYOHEI HORI,**

Faculty of Pharmaceutical Sciences, Kyoto University, Sakyo-ku, Kyoto, 606, Japan and Department of Pharmaceutics, Kyoto University Hospital, Faculty of Medicine, Kyoto University,** Sakyo-ku, Kyoto, 606, Japan*

(Received August 31, 1984)

A nonlinear multiple regression analysis program MULTI2(BAYES) was developed for microcomputers. The Bayesian algorithm which is incorporated in MULTI2 (BAYES) combines the insufficient individual patient data (individual data) with the pharmacokinetic parameters published in literatures (population parameters) to predict the plasma time course of the patient. The program is written in the minimum Microsoft BASIC commands alone to be executable on many personal computers without any modification. The numbers of parameters to estimate, independent variables and dependent variables are not restricted in use of MULTI2(BAYES). The pharmacokinetic models are defined as one pleases by the user. The four nonlinear least squares algorithms, *i.e.* Gauss-Newton method, damping Gauss-Newton method, modified Marquardt method by Fletcher and simplex method can be selected at user's option. MULTI2(BAYES) calculates the confidence limits of time courses at 95% significant level.

Keywords — MULTI2(BAYES); microcomputer; Bayesian algorithm; population pharmacokinetics; nonlinear least square

INTRODUCTION

It is ideal to determine the drug dosage regimen of an individual patient based on the individual pharmacokinetic parameters and on the physical conditions of the patient. Since the frequent blood samplings cast a heavy burden upon the patient, there is a necessity to predict the plasma time course of the patient based on the insufficient clinical data of the individual patient (individual data). Although it is mathematically impossible to evaluate several pharmacokinetic parameters from one or two points of plasma concentration data, the use of pharmacokinetic parameters of a drug published in literatures (population kinetic parameters) can supply the sufficient clinical information.

Sheiner *et al.* introduced a nonlinear least squares method incorporating the Bayesian algorithm[1] from the standpoint of population pharmacokinetics. The purpose of ordinary least squares methods is to evaluate the parameter vector $P(= (p_1, p_2, .. p_m))$ which gives the smallest residual sum of squares SS defined by

$$SS = \sum_{i=1}^{n} W_i (C_i - f(t_i, P))^2 \quad (1)$$

where m is number of parameters, n is number of data points, t_i is independent variable (or time), C_i is dependent variable (or plasma concentration of drug) and $f(t_i, P)$ is a model equation. W_i is the weight of data points and $1, 1/C_i$ and $1/C_i^2$ are often adopted in the pharmacokinetics.

The least squares method based on the Bayesian algorithm estimates the parameters which minimize the following SS.

$$SS = \sum_{i=1}^{n} (C_i - f(t_i, P))^2/\sigma_i^2 + \sum_{j=1}^{m} (\bar{p}_j - p_j)^2/\omega_j^2 \quad (2)$$

where σ_i is the the standard deviation of individual data, and $\bar{p}_j$ and ω_j are the mean and standard deviation of population parameters, respectively.

We reported two nonlinear least squares programs MULTI[2] based on Eq. 1 and MULTI (RUNGE)[3] based on the simultaneous ordinary differential equations. The purpose of the present report is to introduce MULTI2(BAYES) which is based on the Bayesian algorithm.

Hardware

MULTI2(BAYES) is written in the minimum Microsoft BASIC command alone to be used in many personal computers without any modification of the program. The program occupies about 10 Kbytes for its program list. Though the output device is restricted to the cathode ray tube (CRT), the output to printers is achieved by a simple substitution of PRINT commands with LPRINT or PRINT#1 which is the output command to printer.

```
LIST OF MULTI2(BAYES)
10 PRINT"************************************************"
15 PRINT"*            MULTI-LINES FITTINGS 2            *"
16 PRINT"*              BY BAYESIAN METHOD              *"
17 PRINT"************************************************"
25 DIM TS(30):FOR I=1 TO 30:READ TS(I):NEXT I:REM** STUDENT T-TABLE **
26 DATA 12.706,4.303,3.182,2.776,2.571,2.447,2.365,2.306,2.262,2.228
27 DATA 2.201,2.179,2.160,2.145,2.131,2.120,2.110,2.101,2.093,2.086
28 DATA 2.080,2.074,2.069,2.064,2.060,2.056,2.052,2.048,2.045,2.042
29 PRINT:PRINT"DEFINE EQUATION AT 1000,1100,1200,1300,1400.":PRINT
30 PRINT"CP IS DEPENDENT VARIABLES."
32 PRINT:PRINT"TT(1), TT(2), TT(3), ... ARE INDEPENDENT VARIABLES."
35 PRINT:PRINT"P(1), P(2), ... ARE PARAMETERS TO ESTIMATE."
36 PRINT:PRINT"        GOTO 840 IF DIVERGED."
37 DIM ME$(3):ME$(0)="GAUSS-NEWTON":ME$(1)="DAMPING GAUSS-NEWTON"
38 ME$(2)="MODIFIED MARQUARDT":ME$(3)="SIMPLEX"
40 PRINT:FOR I=0 TO 3:PRINT"(";I;") ";ME$(I);" METHOD":NEXT I
45 PRINT:INPUT"WHICH ALGORITHM DO YOU SELECT";AL
49 PRINT:PRINT"* I BELIEVE YOU HAVE DEFINED EQUATIONS *"
50 INPUT"SUBJECT NAME";N$:INPUT"BAYESIAN(1) OR NORMAL(2)";ME
51 INPUT"NUMBER OF INDEPENDENT VARIABLES(TT)";NV:DIM TT(NV)
52 INPUT"NUMBER OF DEPENDENT VARIABLES(CP)";LN:DIM NL(LN)
55 INPUT"WEIGHT OF DATA(0,1,2)";IW:INPUT"NUMBER OF PARAMETERS";M
56 FOR I=1 TO LN:PRINT"NUMBER OF POINTS(";I;")";:INPUT NL(I):NEXT I
58 N=0:FOR I=1 TO LN:N=N+NL(I):NEXT I:IF ME=1 THEN N=N+M
60 DIM T(N,NV),CP(N),A(M,M+1),P(M),X0(M),X1(M),XT(30),YT(30),XX(30),ZT(30)
62 DIM LS(M,M+1),PM(M),PV(M),WT(N),CS(N,M),GR$(23)
63 NL(0)=0:BS=0:FOR J=1 TO LN:BS=BS+NL(J-1):PRINT
64 FOR I=1 TO NL(J):FOR V=1 TO NV
65 PRINT"TT(";V;") OF";I;"TH POINT ON";J;"TH CURVE";:INPUT T(BS+I,V)
66 NEXT V:PRINT"CP OF";I;"TH POINT ON";J;"TH CURVE";:INPUT CP(BS+I)
67 NEXT I:NEXT J
68 IF ME<>1 THEN FOR I=1 TO N:WT(I)=1/CP(I)^IW:NEXT I:GOTO 75
69 FOR I=1 TO M:PRINT"MEAN, VARIANCE OF P(";I;")";:INPUT PM(I),PV(I):NEXT I
70 PRINT:INPUT"WEIGHT OF PATIENT DATA(FF) (0-1)";FF
71 FOR I=1 TO N-M:WT(I)=FF/CP(I)^IW:NEXT I:FOR I=1 TO M
72 IF PV(I)=0 THEN WT(N-M+I)=(1-FF)/PM(I)^IW:GOTO 74
73 WT(N-M+I)=(1-FF)/PV(I)
74 NEXT I:FOR I=1 TO M:CS(N-M+I,I)=1:NEXT I
75 IF AL=3 THEN 3000
77 PC=.0001:CF=100
78 PRINT:INPUT"DP FOR JACOBIAN(0.1-0.0001)";DP:PRINT:FOR I=1 TO M
80 PRINT"INITIAL P(";I;")=";:INPUT P(I):A(I,0)=P(I):NEXT I:GOSUB 4000:S1=SS
140 PRINT"INITIAL SS=";SS:FOR KK=1 TO 100:GOSUB 7000:GOSUB 7400:GOSUB 6000:JJ=0
490 JJ=JJ+1:IF JJ>25 THEN PRINT"OUT OF DAMPING":GOTO 730
500 FOR I=1 TO M:P(I)=A(I,0)+A(I,M+1):NEXT I:GOSUB 4000
```

```
510 DS=ABS(S1-SS):IF (AL<>2)+(SS=0) THEN 590
515 REM**** FLETCHER MODIFICATION ****
520 PW=0:FOR I=1 TO M:PW=PW+X0(I)*A(I,M+1)+CF*A(I,M+1)*A(I,M+1):NEXT I
530 IF DS>.75*PW THEN CF=CF/2
540 IF DS<.25*PW THEN CF=5*CF
590 IF DS<=PC*S1 THEN 730:REM**** CHECK OF CONVERGENCE ****
595 REM**** DAMPING ****
600 IF (AL=1)*(SS>S1) THEN FOR I=1 TO M:A(I,M+1)=.5*A(I,M+1):NEXT I:GOTO 490
630 FOR I=1 TO M:A(I,0)=P(I):NEXT I:S1=SS:PRINT:PRINT"LOOP=";KK
640 IF AL=1 THEN PRINT"DAMP=";JJ
650 IF AL=2 THEN PRINT"FACTOR=";CF
660 FOR I=1 TO M:PRINT"P(";I;")=";P(I):NEXT I:PRINT"SS=";SS:NEXT KK
730 REM**** LOOP EXIT ****
731 PRINT:PRINT"*";N$;"*  BY ";ME$(AL);" METHOD":PRINT"WEIGHT=1/CP^(";IW;")"
733 PRINT"LOOP=";KK:IF (AL<>3)*(N>M) THEN GOSUB 7000:GOSUB 14000
734 IF SS=0 THEN PRINT"AIC=-INFINITE":GOTO 740
735 PRINT"AIC=";N*LOG(SS)+2*M
740 IF AL=3 THEN PRINT"ALPHA=";AA;"  BETA=";BB;"  GAMMA=";CC
742 IF AL<>3 THEN PRINT"DP=";DP
743 IF ME=1 THEN PRINT"WEIGHT OF PATIENT(FF)=";FF
750 PRINT:FOR I=1 TO M:PRINT"FINAL P(";I;")=";P(I);
760 IF (AL<>3)*(X0(I)>0)*(N>M) THEN PRINT"  S.D.=";SQR(A(I,I)*SS/(N-M));
810 PRINT:NEXT I:PRINT"FINAL SS=";SS:PRINT:GOSUB 9200:BS=0:NL(0)=0
815 FOR J=1 TO LN:BS=BS+NL(J-1):PRINT:XS=0:FOR I=1 TO NL(J)
820 FOR V=1 TO NV:TT(V)=T(BS+I,V):PRINT"TT(";V;") OF";J;"TH CURVE=";TT(V)
825 NEXT V:GOSUB 900:XT(I)=TT(1):YT(I)=CP(BS+I)
830 PRINT"CP OF";J;"TH CURVE=";CP;" (OBSERVED=";CP(BS+I);")"
832 NEXT I:PRINT:GOSUB 9200
834 DD=XT(NL(J))*1.5/31:FOR I=1 TO 30:XX(I)=(I-1)*DD:NEXT I
835 FOR I=1 TO 30:TT(1)=XX(I):GOSUB 900:ZT(I)=CP:NEXT I:XT$="TT1("+STR$(J)+")"
836 YT$="CP("+STR$(J)+")":XL$="0":YL$="0":IF ME=2 THEN XL$="N":YL$="N"
837 NS=NL(J):NP=30:GOSUB 10000
838 GOSUB 9200:NEXT J
840 PRINT:PRINT"** SELECTION OF ALGORITHM **"
842 FOR I=0 TO 3:PRINT"(";I;") ";ME$(I);" METHOD":NEXT I
844 PRINT:PRINT"** THE OTHER PROCEDURES **"
846 PRINT"(-3) CALCULATION OF SIGNIFICANT BOUNDARY OF CP AT 95% LEVEL"
847 PRINT"(-2) CHANGE OF FF VALUE"
848 PRINT"(-1) END":PRINT
850 INPUT"WHICH DO YOU SELECT";A1:IF (A1=-2)*(ME=2)+(A1=-3)*(AL=3) THEN 850
851 IF A1=-3 THEN GOTO 8100
852 IF A1=-2 THEN 70
855 IF A1=-1 THEN END
857 AL=A1:PRINT:PRINT"**** ";ME$(AL);" METHOD ****"
860 GOTO 75:REM**********************************************************
900 ON J GOTO 1000,1100,1200,1300,1400
1000 DEF FN CP=F1(TT(J),P(I)):RETURN
1100 DEF FN CP=F2(TT(J),P(I)):RETURN
1200 DEF FN CP=F3(TT(J),P(I)):RETURN
1300 DEF FN CP=F4(TT(J),P(I)):RETURN
1400 DEF FN CP=F5(TT(J),P(I)):RETURN
1999 REM JACOBIAN *******************************************************
2000 FOR JS=1 TO M:PT=P(JS):P(JS)=PT+DP:GOSUB 900
2020 DD=CP:P(JS)=PT-DP:GOSUB 900
2030 CS(BS+I,JS)=(DD-CP)/2/DP:P(JS)=PT:NEXT JS:RETURN
2999 REM**** SIMPLEX METHOD ****
3000 AA=1:BB=.5:CC=2:SG=1E+10:PC=.00001
3025 PRINT:FOR I=1 TO M:PRINT"INITIAL P(";I;")";:INPUT A(I,1):NEXT I
3030 FOR J=2 TO M+1:FOR I=1 TO M:A(I,J)=2*RND(1)*A(I,1)+.01*(RND(1)-.5)
3035 NEXT I:NEXT J:KK=0
3040 FOR K=1 TO M+1:FOR I=1 TO M:P(I)=A(I,K):NEXT I:GOSUB 4000:A(0,K)=SS:NEXT K
3070 KK=KK+1:IF KK>=500 THEN 730
3075 PRINT:PRINT"LOOP=";KK:FOR I=1 TO M+1:PRINT"SS";I;"=";A(0,I):NEXT I
3077 GOTO 5000
3080 SR=0:SL=1E+10:FOR J=1 TO M+1:IF SR<A(0,J) THEN JH=J:SR=A(0,J)
```

```
3090 IF SL>A(0,J) THEN JL=J:SL=A(0,J)
3100 NEXT J:SR=0:FOR J=1 TO M+1:IF (J<>JH)*(SR<A(0,J)) THEN JS=J:SR=A(0,J)
3110 NEXT J:FOR I=1 TO M:X0(I)=0:FOR J=1 TO M+1
3115 IF J<>JH THEN X0(I)=X0(I)+A(I,J)
3120 NEXT J:X0(I)=X0(I)/M:NEXT I:FOR I=1 TO M:A(I,0)=(1+AA)*X0(I)-AA*A(I,JH)
3130 P(I)=A(I,0):NEXT I:GOSUB 4000:SR=SS:IF SR<=A(0,JS) THEN 3300
3160 IF SR<A(0,JH) THEN FOR I=1 TO M:A(I,JH)=A(I,0):NEXT I:A(0,JH)=SR
3170 FOR I=1 TO M:A(I,0)=BB*A(I,JH)+(1-BB)*X0(I)
3180 P(I)=A(I,0):NEXT I:GOSUB 4000:SR=SS
3190 IF SR<=A(0,JH) THEN FOR I=1 TO M:A(I,JH)=A(I,0):NEXT I:A(0,JH)=SR:GOTO 3070
3200 FOR K=1 TO M+1:FOR I=1 TO M:A(I,K)=(A(I,K)+A(I,JL))/2:P(I)=A(I,K):NEXT I
3210 GOSUB 4000:A(0,K)=SS:NEXT K:GOTO 3070
3300 IF SR<A(0,JL) THEN 3500
3320 FOR I=1 TO M:A(I,JH)=A(I,0):NEXT I:A(0,JH)=SR:GOTO 3070
3500 FOR I=1 TO M:X1(I)=CC*A(I,0)+(1-CC)*X0(I)
3505 P(I)=X1(I):NEXT I:GOSUB 4000:SL=SS
3510 IF SL<SR THEN FOR I=1 TO M:A(I,JH)=X1(I):NEXT I:A(0,JH)=SL:GOTO 3070
3520 GOTO 3320
3999 REM**** CALCULATION OF SS ****
4000 SS=0:BS=0:FOR J=1 TO LN:BS=BS+NL(J-1):FOR I=1 TO NL(J)
4010 FOR V=1 TO NV:TT(V)=T(BS+I,V):NEXT V
4020 GOSUB 900:SS=SS+(CP(BS+I)-CP)^2*WT(BS+I):NEXT I:NEXT J:IF ME<>1 THEN RETURN
4030 S2=0:BS=BS+NL(LN):FOR I=1 TO M
4040 S2=S2+(PM(I)-P(I))^2*WT(BS+I):NEXT I:SS=SS+S2:RETURN
4999 REM**** CHECK OF CONVERGENCE ****
5000 SR=0:FOR I=1 TO M+1:SR=SR+A(0,I):NEXT I
5030 IF ABS(SR-SG)>PC*SG THEN SG=SR:GOTO 3080
5040 FOR I=1 TO M:P(I)=A(I,JL):NEXT I:SS=A(0,JL):GOTO 730
5999 REM**** MODIFIED CHOLESKY METHOD ****
6000 IF NS=1 THEN A(1,2)=A(1,2)/A(1,1):RETURN
6020 FOR JS=1 TO NS:LS(1,JS)=A(1,JS):NEXT JS
6030 FOR IS=2 TO NS:FOR JS=IS TO NS:DS=0:FOR KS=1 TO IS-1
6040 DS=DS+LS(KS,IS)*LS(KS,JS)/LS(KS,KS)
6050 NEXT KS:LS(IS,JS)=A(IS,JS)-DS:NEXT JS:NEXT IS
6060 LS(1,NS+1)=A(1,NS+1):FOR IS=2 TO NS:DS=0
6070 FOR KS=1 TO IS-1:DS=DS+LS(KS,IS)*LS(KS,NS+1)/LS(KS,KS):NEXT KS
6080 LS(IS,NS+1)=A(IS,NS+1)-DS:NEXT IS
6090 A(NS,NS+1)=LS(NS,NS+1)/LS(NS,NS)
6100 FOR IS=NS-1 TO 1 STEP -1:DS=0:FOR KS=IS+1 TO NS
6110 DS=DS+LS(IS,KS)*A(KS,NS+1):NEXT KS
6120 A(IS,NS+1)=(LS(IS,NS+1)-DS)/LS(IS,IS):NEXT IS:RETURN
6999 REM**** NORMAL EQUATION ****
7000 NL(0)=0:BS=0:FOR J=1 TO LN:BS=BS+NL(J-1):FOR I=1 TO NL(J):FOR V=1 TO NV
7100 TT(V)=T(BS+I,V):NEXT V:GOSUB 900:CS(BS+I,0)=CP(BS+I)-CP:GOSUB 2000
7105 NEXT I:NEXT J:IF ME<>1 THEN 7310
7107 FOR I=1 TO M:CS(N-M+I,0)=PM(I)-P(I):NEXT I
7310 FOR I=1 TO M:FOR J=1 TO M:A(I,J)=0:FOR L=1 TO N
7390 A(I,J)=A(I,J)+CS(L,I)*CS(L,J)*WT(L):NEXT L:A(J,I)=A(I,J):NEXT J:NEXT I
7395 RETURN
7400 FOR I=1 TO M:A(I,M+1)=0:FOR J=1 TO N
7440 A(I,M+1)=A(I,M+1)+CS(J,I)*CS(J,0)*WT(J):NEXT J:NEXT I:NS=M
7450 REM**** MARQUARDT MODIFICATION ****
7460 IF AL=2 THEN FOR I=1 TO M:A(I,I)=A(I,I)+CF:X0(I)=A(I,M+1):NEXT I
7470 RETURN
8100 REM**** S.D. OF CP ****
8110 PRINT:PRINT"WHICH CURVE( 1 -";LN;")";:INPUT JP
8120 GOSUB 7000:NS=M:GOSUB 14000
8125 PRINT:J=JP:FOR V=1 TO NV:PRINT"TT(";V;") ON";J;"TH CURVE (OR 9999)";
8127 INPUT TT(V):IF TT(V)=9999 THEN 840
8130 NEXT V:FOR IS=1 TO M:PT=P(IS):P(IS)=PT+DP:GOSUB 900
8140 DD=CP:P(IS)=PT-DP:GOSUB 900:X0(IS)=(DD-CP)/2/DP:P(IS)=PT:NEXT IS
8160 PT=0:FOR IS=1 TO M:FOR JS=1 TO M:PT=PT+A(IS,JS)*X0(IS)*X0(JS)
8170 NEXT JS:NEXT IS:GOSUB 900:PRINT"CP=";CP;"(+/-)";
8180 PRINT TS(N-M)*SQR(SS/(N-M)*PT);"AT 95% LEVEL OF T-TEST":GOTO 8125
9200 REM**** TEMPORARY STOP ****
```

```
9205 INPUT"DO YOU CONTINUE(Y/N)";CO$
9210 IF CO$="N" THEN END
9220 RETURN
10000 REM**** GRAPH SUBROUTINE ****
10010 Y0=YT(1):Y9=YT(1):X0=XT(1):X9=XT(1)
10020 FOR GI=2 TO NS
10030 IF XT(GI)>X9 THEN X9=XT(GI)
10040 IF XT(GI)<X0 THEN X0=XT(GI)
10050 IF YT(GI)>Y9 THEN Y9=YT(GI)
10060 IF YT(GI)<Y0 THEN Y0=YT(GI)
10062 NEXT GI:FOR GI=1 TO NP
10064 IF ZT(GI)>Y9 THEN Y9=ZT(GI)
10066 IF ZT(GI)<Y0 THEN Y0=ZT(GI)
10067 IF XX(GI)>X9 THEN X9=XX(GI)
10068 IF XX(GI)<X0 THEN X0=XX(GI)
10070 NEXT GI
10080 IF YL$="0" THEN Y0=0
10090 IF XL$="0" THEN X0=0
10110 SP$="":FOR GI=1 TO 39:SP$=SP$+" ":NEXT GI
10120 FOR GI=1 TO 23:GR$(GI)=SP$:NEXT GI
10130 G1=5:G2=4:G3=5:G4=6:G5=5:XM=30:YM=18:PT=1
10140 GY=(Y9-Y0)/(G2-1):GG$="T"
10150 GP=0:IF (GY>.1)*(Y9<1000)*(Y9>-1000) THEN GT$="    "+YT$:GOTO 10170
10160 GP=INT(.4343*LOG(GY)):GT$=YT$+"X10^"+STR$(GP)+")"
10170 GR$(1)=GT$+RIGHT$(GR$(1),LEN(GR$(1))-LEN(GT$))
10180 PT=PT+1
10190 FOR GI=1 TO G2:G$=STR$((Y9-GY*(GI-1))/10^GP):IF LEN(G$)<G1 THEN G$=G$+SP$
10200 G$=LEFT$(G$,G1)+GG$:GR$(PT)=G$+RIGHT$(GR$(PT),LEN(GR$(PT))-LEN(G$))
10210 PT=PT+1:IF GI=G2 THEN 10250
10220 FOR GJ=1 TO G3:GT$=LEFT$(SP$,G1)+"I"
10230 GR$(PT)=GT$+RIGHT$(GR$(PT),LEN(GR$(PT))-LEN(GT$)):PT=PT+1
10240 NEXT GJ:GG$="+":NEXT GI
10250 G$="+":FOR GI=1 TO G4-1:G$=G$+"+-----":NEXT GI:G$=G$+"I"
10260 GT$=LEFT$(SP$,G1)+G$:GR$(PT)=GT$+RIGHT$(GR$(PT),LEN(GR$(PT))-LEN(GT$))
10270 PT=PT+1
10280 GX=(X9-X0)/(G4-1):GG$=LEFT$(SP$,G1)
10290 GP=0:IF (GX>.1)*(X9<1000)*(X9>-1000) THEN 10310
10300 GP=INT(.4343*LOG(GX)):XT$=XT$+" (X10^"+STR$(GP)+")"
10310 FOR GI=1 TO G4:G$=STR$((X0+GX*(GI-1))/10^GP):IF LEN(G$)<4 THEN G$=G$+SP$
10320 GG$=GG$+LEFT$(G$,4):IF GI<G4 THEN GG$=GG$+LEFT$(SP$,G5-3)
10330 NEXT GI:GR$(PT)=GG$+RIGHT$(GR$(PT),LEN(GR$(PT))-LEN(GG$)):PT=PT+1
10340 GT$=LEFT$(SP$,INT(G1+(XM-LEN(XT$))/2))+XT$
10350 GR$(PT)=GT$+RIGHT$(GR$(PT),LEN(GR$(PT))-LEN(GT$))
10370 REM---- PLOT ROUTINE ----
10380 FOR GI=1 TO NP
10400 GX=INT(XM*(XX(GI)-X0)/(X9-X0)+.5)+G1+1
10410 GY=INT(YM*(Y9-ZT(GI))/(Y9-Y0)+.5)+1:PL$="*"
10420 GR$(GY+1)=LEFT$(GR$(GY+1),GX)+PL$+RIGHT$(GR$(GY+1),LEN(GR$(GY+1))-GX-1)
10430 NEXT GI:FOR GI=1 TO NS
10440 GX=INT(XM*(XT(GI)-X0)/(X9-X0)+.5)+G1+1
10450 GY=INT(YM*(Y9-YT(GI))/(Y9-Y0)+.5)+1:PL$="O"
10460 GR$(GY+1)=LEFT$(GR$(GY+1),GX)+PL$+RIGHT$(GR$(GY+1),LEN(GR$(GY+1))-GX-1)
10470 NEXT GI
10480 PRINT:FOR GI=1 TO 23:PRINT GR$(GI):NEXT GI:RETURN
14000 REM**** GAUSS-JORDAN METHOD ****
14010 FOR IS=1 TO NS:X0(IS)=IS:NEXT IS
14020 DE=1:FOR KS=1 TO NS:PI=0:FOR JS=KS TO NS
14030 IF ABS(PI)<ABS(A(KS,JS)) THEN LS=JS:PI=A(KS,JS)
14040 NEXT JS:IF PI=0 THEN PRINT"MATRIX IS SINGULAR.":DE=0:RETURN
14060 IF LS=KS THEN 14090
14070 WI=X0(KS):X0(KS)=X0(LS):X0(LS)=WI:FOR IS=1 TO NS:W=A(IS,KS)
14080 A(IS,KS)=A(IS,LS):A(IS,LS)=W:NEXT IS
14090 A(KS,KS)=1:DE=DE*PI:FOR JS=1 TO NS+1:A(KS,JS)=A(KS,JS)/PI:NEXT JS
14100 FOR IS=1 TO NS:IF IS=KS THEN 14120
14110 AI=A(IS,KS):A(IS,KS)=0:FOR JS=1 TO NS+1:A(IS,JS)=A(IS,JS)-AI*A(KS,JS)
```

```
14115 NEXT JS
14120 NEXT IS:NEXT KS
14130 FOR IS=1 TO NS-1
14140 KS=X0(IS):IF KS=IS THEN 14170
14150 WI=X0(KS):X0(KS)=X0(IS):X0(IS)=WI:FOR JS=1 TO NS+1:W=A(KS,JS)
14160 A(KS,JS)=A(IS,JS):A(IS,JS)=W:NEXT JS:GOTO 14140
14170 NEXT IS:RETURN
```

FIG. 1. *Program List of MULTI2(BAYES)*

The Algorithms for Nonlinear Least Squares

Fig. 1 is the program list of MULTI2 (BAYES). The classical Gauss-Newton method, the damping Gauss-Newton method, the modified Marquardt method by Fletcher and the simplex method can be selected for the nonlinear curve fitting in MULTI2(BAYES), as in MULTI.[2] MULTI2(BAYES) estimates the pharmacokinetic parameters which minimize the following residual sum of sequares *SS* instead of Eq. 2.

$$SS=FF\sum_{i=1}^{n}(C_i-f(T_i,P))^2/C_i^{IW}+(1-FF)\sum_{j=1}^{m}(\bar{p}_j-p_j)^2/\omega_j^2 \quad (3)$$

where C_i is the value of the time course of patient, $f(T_i,P)$ is the model equation for ith time course curve, T_i is a vector of independent variables and FF which takes a value from 0 to 1 is overall weight of patient data. Eq. 3 is derived from Eq. 2 by substituting $FF/(1-FF)\cdot 1/C_i^{IW}$ into σ_i^2 in Eq. 2. The parameter IW is the exponent of the measured data C_i. The data weights of 1, $1/C_i$ or $1/C_i^2$ are selected for $IW=0$, $IW=1$ or $IW=2$, respectively. The selection of IW is discussed in the subsequent section. As FF increases to unity, the weight of individual patient data increases against that of population kinetic parameters. The reason why MULTI2(BAYES) adopted Eq. 3 instead of Eq. 2 is that it is difficult to estimate the σ_i of individual patient data from the inadequate plasma data of patient, although σ_i and ω_j in Eq. 2 can be simultaneously evaluated by two stage method (TS method) or NONMEM method[4-6] in case that the sufficient data of many patients are available. The improvements of MULTI2(BAYES) over MULTI are that model equations with several independent variables can be used in MULTI2 (BAYES)(while, one independent variable in MULTI) and that the Bayesian algorithm can also be selected in MULTI2(BAYES) (lines from 68 through 74). The Gauss elimination method in MULTI to solve the normal equation is replaced by the modified Cholesky method[7] to improve the computing speed in MULTI2 (BAYES) (lines from 6000 through 6120). Furthermore, this program calculates the confidence limits of time course curves at 95% significant level (lines from 25 through 28, from 8100 through 8180 and from 14000 through 14170).

Definition of Model Equations to Estimate

Table I presents population parameters and individual patient data of pralidoxime methanesulfonate.[8] The patient plasma data were generated using the pharmacokinetic parameters of subjects following 2 g dose of drug (FD=800 mg, k_a=1.02 h^{-1}, k_e=0.470 h^{-1} and V_d=88.9 l). The example use of MULTI2(BAYES) is shown using the data in Table I. The adopted model is the following one-compartment model with first order absorption.

$$C_p=FD\cdot k_a/V_d/(k_e-k_a)\cdot(\exp(-k_a\cdot t)-\exp(-k_e\cdot t)) \quad (4)$$

where C_p is plasma concentration, FD is drug amount absorbed, k_a is absorption rate constant, k_e is elimination rate constant, and V_d is volume of distribution. The pharmacokinetic equations to be simultaneously fitted are defined at 1000, 1100, .. 1400. For example, the defined equation for Eq. 4 is given as follows.

```
1000 CP=TT(2)*P(1)/P(2)/(P(1)-
        P(3))*(EXP(-P(3)*TT(1))
        -EXP(-P(1)*TT(1))):RETURN
```

where CP is a dependent variable (for example, plasma concentration), TT(1) and TT(2) are independent variables, P(1), P(2) and P(3) are pharmacokinetic parameters to estimate. In this example, CP is C_p, TT(2) is *FD*, P(1) is k_a, P(2) is V_d and P(3) is k_e.

TABLE I. *Population Parameters and Individual Plasma Data of Pralidoxime Methanesulfonate*
The patient plasma data (two points) were generated using the pharmacokinetic parameters of subjects after a 2 g dose of the drug in reference (FD=800 mg, $k_a=1.02\ h^{-1}$, $k_e=0.47\ h^{-1}$ and $V_d=88.9$ l).

Population parameters

Kinetic parameters	Mean	Variance
k_a	0.81 (h^{-1})	0.090 (h^{-2})
V_d	96 (l)	380 (l^2)
k_e	0.46 (h^{-1})	0.012 (h^{-2})

Individual data

Time (h)	C_p (μg/ml)
1.2	4.6
4.0	2.3

70 kg of body weight is supposed. The estimated absorbed amount (FD) is 800 mg.

```
EX.  OF  MULTI2(BAYES)

900 ON J GOTO 1000,1100,1200,1300,1400
1000 CP=TT(2)*P(1)/P(2)/(P(1)-P(3))*(EXP(-P(3)*TT(1))-EXP(-P(1)*TT(1))):RETURN
1100 DEF FN CP=F2(TT(J),P(I)):RETURN
1200 DEF FN CP=F3(TT(J),P(I)):RETURN
1300 DEF FN CP=F4(TT(J),P(I)):RETURN
1400 DEF FN CP=F5(TT(J),P(I)):RETURN

RUN
************************************************
*            MULTI-LINES FITTINGS 2            *
*              BY BAYESIAN METHOD              *
************************************************

DEFINE EQUATION AT 1000,1100,1200,1300,1400.

CP IS DEPENDENT VARIABLES.

TT(1), TT(2), TT(3), ... ARE INDEPENDENT VARIABLES.

P(1), P(2), ... ARE PARAMETERS TO ESTIMATE.

        GOTO 840 IF DIVERGED.

( 0 ) GAUSS-NEWTON METHOD
( 1 ) DAMPING GAUSS-NEWTON METHOD
( 2 ) MODIFIED MARQUARDT METHOD
( 3 ) SIMPLEX METHOD
```

```
WHICH ALGORITHM DO YOU SELECT 1

* I BELIEVE YOU HAVE DEFINED EQUATIONS *
SUBJECT NAMESUBJECT 1
BAYESIAN(1) OR NORMAL(2) 1
NUMBER OF INDEPENDENT VARIABLES(TT) 2
NUMBER OF DEPENDENT VARIABLES(CP) 1

WEIGHT OF DATA(0,1,2) 0
NUMBER OF PARAMETERS 3
NUMBER OF POINTS( 1 ) 2

TT( 1 ) OF 1 TH POINT ON 1 TH CURVE 1.2
TT( 2 ) OF 1 TH POINT ON 1 TH CURVE 800
CP OF 1 TH POINT ON 1 TH CURVE 4.6
TT( 1 ) OF 2 TH POINT ON 1 TH CURVE 4
TT( 2 ) OF 2 TH POINT ON 1 TH CURVE 800
CP OF 2 TH POINT ON 1 TH CURVE 2.3
MEAN, VARIANCE OF P( 1 ) .81 , .09
MEAN, VARIANCE OF P( 2 ) 96 , 380
MEAN, VARIANCE OF P( 3 ) .46 , .012

WEIGHT OF PATIENT DATA(FF) (0-1) .5

DP FOR JACOBIAN(0.1-0.0001) .001

INITIAL P( 1 )= .81
INITIAL P( 2 )= 96
INITIAL P( 3 )= .46
INITIAL SS= .313362

LOOP= 1
DAMP= 1
P( 1 )= .899565
P( 2 )= 91.3035
P( 3 )= .455026
SS= .141085

LOOP= 2
DAMP= 1
P( 1 )= .892611
P( 2 )= 90.8188
P( 3 )= .455755
SS= .140404

*SUBJECT 1*  BY DAMPING GAUSS-NEWTON METHOD
WEIGHT=1/CP^( 0 )
LOOP= 3
AIC=-3.81627
DP= .001
WEIGHT OF PATIENT(FF)= .5

FINAL P( 1 )= .893416    S.D.= .0937316
FINAL P( 2 )= 90.8348    S.D.= 5.64349
FINAL P( 3 )= .455746    S.D.= .0353038
FINAL SS= .140401

DO YOU CONTINUE(Y/N)Y

TT( 1 ) OF 1 TH CURVE= 1.2
TT( 2 ) OF 1 TH CURVE= 800
CP OF 1 TH CURVE= 4.25101   (OBSERVED= 4.6 )
TT( 1 ) OF 1 TH CURVE= 4
TT( 2 ) OF 1 TH CURVE= 800
CP OF 1 TH CURVE= 2.3999   (OBSERVED= 2.3 )
```

```
DO YOU CONTINUE(Y/N)Y

   CP( 1)
 4.6 T       0
     I       *****
     I      *     **
     I            *
     I     *       *
     I               **
 3.06+    *            *
     I                  **
     I                    *
     I   *                0**
     I                       *
     I                        **
 1.53+                          **
     I *                          **
     I
     I
     I
     I
 0   +*
     ++-----+-----+-----+-----+-----I
      0      1.1   2.2   3.3   4.4   5.6
                TT1( 1)
DO YOU CONTINUE(Y/N)Y

** SELECTION OF ALGORITHM **
( 0 ) GAUSS-NEWTON METHOD
( 1 ) DAMPING GAUSS-NEWTON METHOD
( 2 ) MODIFIED MARQUARDT METHOD
( 3 ) SIMPLEX METHOD

** THE OTHER PROCEDURES **
(-3) CALCULATION OF SIGNIFICANT BOUNDARY OF CP AT 95% LEVEL
(-2) CHANGE OF FF VALUE
(-1) END

WHICH DO YOU SELECT-3

WHICH CURVE( 1 - 1 ) 1

TT( 1 ) ON 1 TH CURVE (OR 9999) 2
TT( 2 ) ON 1 TH CURVE (OR 9999) 800
CP= 4.21469 (+/-) 1.08231 AT 95% LEVEL OF T-TEST

TT( 1 ) ON 1 TH CURVE (OR 9999) 9999

** SELECTION OF ALGORITHM **
( 0 ) GAUSS-NEWTON METHOD
( 1 ) DAMPING GAUSS-NEWTON METHOD
( 2 ) MODIFIED MARQUARDT METHOD
( 3 ) SIMPLEX METHOD

** THE OTHER PROCEDURES **
(-3) CALCULATION OF SIGNIFICANT BOUNDARY OF CP AT 95% LEVEL
(-2) CHANGE OF FF VALUE
(-1) END

WHICH DO YOU SELECT-1
OK
```

FIG. 2. *Example Run of MULTI2(BAYES)*

Since CP is plotted *versus* TT(1) in MULTI2(BAYES), time t should usually be assigned to TT(1). When several model equations are simultaneously fitted, the model equations are, in the same way, defined at 1100, 1200, 1300 and 1400. If the simultaneous curve fitting uses more than 5 curves, the line 900 must be changed, for example, as

900 ON J GOTO 1000,1100,..,
1400,1500,1600,,..

Fig. 2 is the example run of MULTI2(BAYES). The underlines in Fig. 2 specify the input into the computer through the keyboard. The lines without underline are responses on CRT from MULTI2(BAYES). The selected nonlinear algorithm is the damping Gauss-Newton method with the Bayesian algorithm, *IW* and *FF* in Eq. 3 are 0 and 0.5, respectively, and DP which is used to numerically calculate the Jacobian matrix is 0.001. The other data for input are given in Table I. The means of population parameters are used as the initial parameters for P(1), P(2) and P(3).

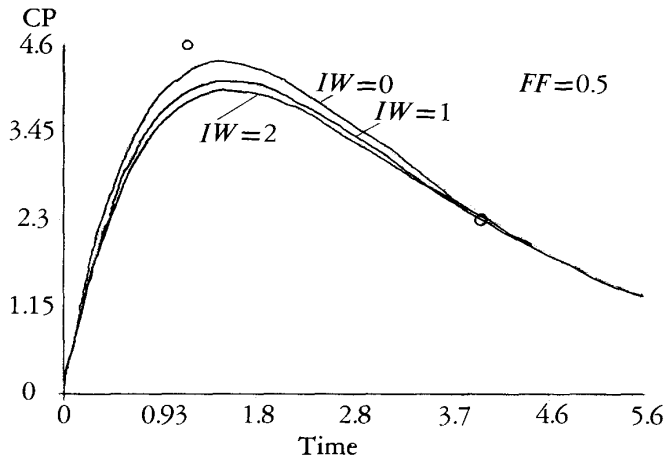

FIG. 3. *Two Individual Dta Points and The Predicted Time Course Curves for IW=0, 1, and 2 When FF is 0.5*
The other computing conditions are the same as in Fig. 2.

DISCUSSIONS

Fig. 3 shows two data points of patient and the predicted time courses by MULTI2 (BAYES) when *IW* is 0, 1 and 2. The selected *FF* is 0.5. The other computing conditions are the same as in Fig. 2. The choice of *IW* does not so much affect the estimated time course in this example, though the data point around peak plasma concentration becomes apart from the predicted value by the least squares. Fig. 4 show

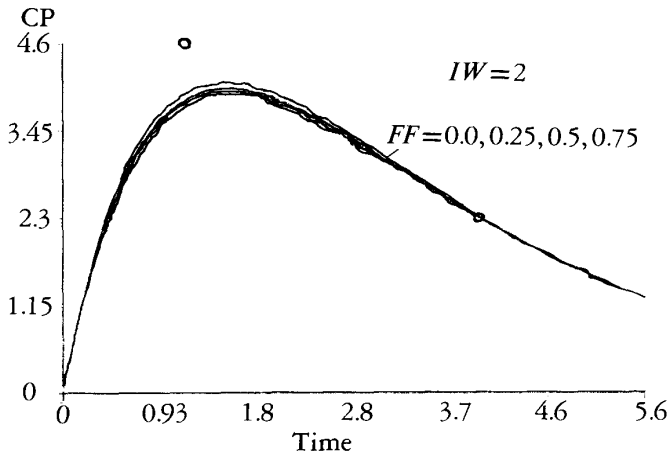

FIG. 4. *Two Individual Data Points and The Predicted Time Course Curves for FF=0.0, 0.25, 0.5 and 0.75 When IW is 2*
The other computing conditions are the same as in Fig. 2.

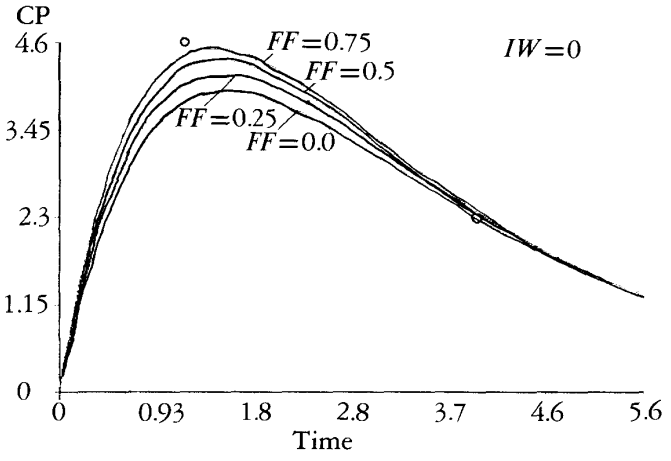

FIG. 5. *Two Individual Data Points and The Predicted Time Course Curves for FF=0.0, 0.25, 0.5 and 0.75 When IW is 0*
The other computing conditions are the same as in Fig. 2.

the data points and the time courses when $IW=2$. The time course curves overlap each other when FF is changed from 0.0 to 0.75. Fig. 5 presents the two data points and predected time courses by MULTI2(BAYES) when FF is 0, 0.25, 0.5 and 0.75. The selected IW is 0. Since the individual patient data are ignored in case $FF=0$, the converged kinetic parameters coincide with the means of population parameters. As FF increases, the predicted value by MULTI2(BAYES) approaches to the data point of the patient around the peak plasma concentration. However, the choice of FF would not be definite, considering that the data of the individual patient contains an error of quantitative analysis and the physical condition of patient can often be fluctuated. FF can statistically be evaluated in case sufficient data of many patients are available. Eq. 2 is reduced to Eq. 5 on the assumption that σ_i^2 is proportional to C_i^{IW} (*i.e.* $\sigma_i^2=\sigma^2 C_i^{IW}$), where $IW=0$, 1 or 2.

$$SS=1/\sigma^2\sum_{i=1}^{n}(C_i-f(t_i,P)^2/C_i^{IW}+\sum_{j=1}^{m}(\bar{p}_j-p_j)^2/\omega^2 \quad (5)$$

σ^2 is calculated by the NOMMEM method or the two stage method (TS method).[2,5] In the TS method, the variance of experimental error σ^2 are calculated by the following equation.

$$\sigma^2=1/(N-m\times k)\sum_{i=1}^{k}SS_i \quad (6)$$

where N is total number of data points of several subjects. m is number of parameters of model equation, k is number of subjects and SS_i is residual sum of squares obtained by curve fitting of data points of each subject. By comparing Eq. 5 with Eq. 3,

$$\sigma^{-2}=FF/(1-FF) \quad (7)$$

By rearranging Eq. 7 with respect to FF,

$$FF=1/(1+\sigma^2) \quad (8)$$

When the individual patient data include large experimental errors, σ^2 becomes large and FF consequently becomes small. The selection of IW is also important to estimate the pharmacokinetic constants of the individual patient. Peck *et al.* discussed the weight of data.[9] The choice of $IW=0$ means that all data points are measured with equal certainty. The use of $IW=2$ means that the data points give a constant coefficient of variation. The selection of $IW=1$ is the hybrid of the above choices. As Peck *et al.* point out, it is indisputable that the selection of weights is often subjective and somewhat arbitrary. However, if there is no definite information that the experimental points have various different variances, there seem to be no objection to the choice of $IW=0$.

REFERENCES

1) L. B. Sheiner, S. Beal, B. Rosenberg and V. V. Marathe: Forecasting individual pharmacokinetics, *Chem. Pharmacol. Ther.*, **26**, 294–305 (1979).
2) K. Yamaoka, Y. Tanigawara, T. Nakagawa and T. Uno: A pharmacokinetic analysis program (MULTI) for microcomputer, *J. Pharamcobio-Dyn.*, **4**, 879–885 (1981).
3) K. Yamaoka and T. Nakagawa: A nonlinear least squares program based on differential equation, MULTI(RUNGE), for microcomputers, *J. Pharmacobio-Dyn.*, **6**, 595–606 (1983).
4) L. B. Sheiner, B. Rosenberg and V. V. Marathe: Estimation of population characteristics of pharmacokinetic parameters from routine clinical data, *J. Pharmacokinet. Biopharm.*, **5**, 445–479 (1977).
5) L. B. Sheiner and S. L. Beal: Evaluation of methods for estimating population pharmacokinetic parameters. I. Michaelis-Menten model: routine clinical pharmacokinetic data, *J. Pharmacokinet. Biopharm.*, **8**, 553–571 (1980).
6) L. B. Sheiner and S. L. Beal: Evaluation of methods for estimating population pharmacokinetic parameters. III. monoexponential model: routine clinical pharmacokinetic data, *J. Pharmacokinet. Biopharm.*, **11**, 303–319 (1983).
7) R. J. Melosh and R. M. Bamford: Efficient solution of lead-deflection equations, *J. Struct. Div. Proc. ASCE*, 661–676 (1969).
8) F. R. Sidel, W. A. Groff and A. Kaminskis: Pradidoxime mathanesulfonate: plasma levels and pharmacokinetics after oral administration to man, *J. Pharm. Sci.*, **61**, 1136–1140 (1972).
9) C. C. Peck, L. B. Sheiner and A. I. Nichols: The Problem of choosing weights in nonlinear regression analysis of pharmacokinetic data, *Drug Metab. Rev.*, **15**, 133–148 (1984).