Arbeidsnotater

STATISTISK SENTRALBYRÅ

Dronningensgt. 16, Oslo-Dep., Oslo 1. Tlf. 41 38 20, 41 36 60

IO 74/11 21. februar 1974

MINSYS, EIT REKNEMASKINPROGRAM FOR ANALYTISK GLATTING AV BEFOLKNINGSRATER

av

ERLING BERGE*

INNHALD

* Eg vil takke Jan M. Hoem og Kjetil Sørlie for kontroll-rekning og for nyttige forslag og kommentarar.

1. INNLEIING

1.1. Dette notatet beskriv eit programsystem for analytisk glatting, korleis det er bygd opp, korleis det skal brukast og korleis det kan modifiserast til å nytte andre funksjonar.

Kapittel 2 gir ein oversikt over problemet og oppbygginga av systemet, kapittel 3 beskriv rutinene som er spesifikke for Hadwiger-funksjonen, kapittel 4 gjer det samme for gammatettheta. I kapittel 5 tar vi for oss bruken av programmet, kva data som krevst, kva parametrar som må settast og til slutt litt om korleis programsystemet kan modifiserast om ein vil nytte andre funksjonar.

I appendiks A har vi ei listing av programsystemet med eksempel på resultata. I appendiks B er gjengitt formlane som må programmerast om beta tettheta skal nyttast til glattingsfunksjon.

1.2. I befolkningsprognosesamanheng må ein nytte dei sikrast mogelege estimat av befolkningsratene. Ved regionale prognoser blir populasjonane ofte for små til at det gir meining å nytte dei observerte ratene direkte. Samtidig er den regionale variasjonen så stor, til dømes når det gjeld fruktbarhet, at ein vanskeleg kan slå saman regionar til observasjonsgrunnlaget blir så stort at dei observerte ratene kan nyttast direkte.

Derfor er det ønskjeleg å estimere fødselsratene ved analytisk glatting.

1.3. Den statistiske teorien for glatting er utvikla av Hoem (1970 og 1973).

Valg av funksjon $f(x, \underset{\sim}{\theta})$ bør motiverast av teoretiske betraktningar over problemet som skal analyserast.

Observasjonsvektoren $\{\hat{\lambda}_x, x = \alpha, \alpha + 1, \ldots, \beta\}$ kan også vere av ein slik art at den gir klare indikasjonar på kva funksjon som bør velgast.

Det programsystemet som her skal omtalast estimerer $\underset{\sim}{\theta}$ når f er anten ein Hadwigerfunksjon eller ei av tetthetene i gammafordelinga og observasjonane er fødselsrater for x-årige kvinner når vi lar x gå fra α til β.

Aldersspesifikke fødselsrater liknar i forma svært på venstreskeive sannsynlighetstettheter. Mellom dei mange mulighetene som finst, har vi konsentrert oppmerksomheta om funksjonar basert på gamma- og Hadwigertetthetene. I arbeidet med å finne ein passande funksjon er det blitt laga fire ulike parameterrepresentasjonar av kvar av funksjonane. Berre den beste utgava er tatt med her.

Resultat fra dette arbeidet finst óg i Hoem & Berge (1974).

Programmet kan imidlertid, ved å velge ein spesiell parameter, ta ein vilkårlig brukarutvikla funksjon.

1.4. Analytisk glatting kan betraktast som ikkje-lineær regresjon.

For ei rad verdiar av x observerer vi utkommet Y slik at vi kjem ut med ei mengd med samhøyrande verdiar $\{(Y, x)\}$. Det vi gjer er å anta at det eksisterer ein funksjonssamanheng mellom Y og x slik at vi kan skrive $Y = f(x, \underset{\sim}{\theta})$ + tilfeldige feil, der $\underset{\sim}{\theta}$ er ein parametervektor som må fastleggast slik at vi i det lange løp gjennomsnittleg vil få små avvik mellom Y_x og $f(x, \underset{\sim}{\theta})$.

2. GRUNNDRAGA I PROGRAMSYSTEMET

2.0. Oppgåva til programsystemet er å glatte befolkningsrater ved hjelp av ulike funksjonstyper.

Systemet kan delast i ein generell del og ein spesiell del. Den generelle delen omfattar dei rutinene som er uavhengige av den valgte glattingsfunksjonen. Den spesielle delen omfattar dei rutinene som er nødvendige for å glatte fødselsrater med ein Hadwigerfunksjon eller ei gammatetthet.

2.1. Generell beskrivelse av problemstillinga.

Problemet som skal løysast av den generelle delen, er å estimere ein parametervektor $\underset{\sim}{\theta}$ slik at vi kan tilnærme observasjonsvektoren $\{\hat{\lambda}_x, x = \alpha, \alpha+1, . , \beta\}$ "best mogeleg" med ein sekvens av funksjonsverdiar $\{f(x, \underset{\sim}{\theta}), x = \alpha, \alpha+1, \ldots, \beta\}$. Den frie variabelen x varierer fra α til og med β i steg på 1 slik at vi alt får $\beta-\alpha+1$ element i observasjonsvektoren. Eit estimat av $\underset{\sim}{\theta}$ finn vi som den parametervektoren $\hat{\underset{\sim}{\theta}}$ som minimerer funksjonen

$$(2.1) \qquad F(\underset{\sim}{\theta}) = \sum_x \{\hat{\lambda}_x - f(x, \underset{\sim}{\theta})\}^2 w_x,$$

der w_x er ein vektfunksjon vi kan velge. Om valg av vektfunksjon og konsekvensar av det, sjå Hoem (1973).

Den noverande versjonen av programmet gir tre mogelege vektfunksjonar:

a) vektoren $\{w_x\}$ blir lest inn fra kort,

b) $w_x = 1$,

og

c) $w_x = L_x/\hat{\lambda}_x$

der L_x er totalt antall personår som ligg til grunn for den observerte raten $\hat{\lambda}_x$ og $x = \alpha, \alpha+1, .., \beta$.

Innlesing av $\{w_x\}$ under alternativ a) er omtala i kapittel 5. Vektfunksjonen b) gir ordinære minste kvadraters estimat av $\underset{\sim}{\Theta}$, mens c) gir eit modifisert minimum kjikvadrat-estimat.

Programmet reknar vidare ut $\hat{\underset{\sim}{\Sigma}}/N$, som er eit estimat for varians - kovariansmatrisa for det endelege parameterestimatet $\hat{\underset{\sim}{\Theta}}$ samt korrelasjonsmatrisa som svarer til $\hat{\underset{\sim}{\Sigma}}/N$. N er det totale antall personar som ligg til grunn for dei observerte ratene $\{\hat{\lambda}_x, x = \alpha, \alpha+1, ..,\beta\}$.

Matrisa $\hat{\underset{\sim}{\Sigma}}$ er gitt som

$$\hat{\underset{\sim}{\Sigma}} = (\hat{\underset{\sim}{J}}^T \underset{\sim}{M} \hat{\underset{\sim}{J}})^{-1} \hat{\underset{\sim}{J}}^T \underset{\sim}{M} \hat{\underset{\sim}{\Sigma}}_0 \underset{\sim}{M} \hat{\underset{\sim}{J}} (\hat{\underset{\sim}{J}}^T \underset{\sim}{M} \hat{\underset{\sim}{J}})^{-1},$$

sjå Hoem (1974).

Her tyder toppskrift ^{T} transponering. $\underset{\sim}{M}$ er ei diagonalmatrise med w_x-ane som diagonalelement.

$\hat{\underset{\sim}{\Sigma}}_0/N$ er eit estimat for varians-kovariansmatrisa til $\{\hat{\lambda}_x, x = \alpha, .., \beta\}$, og

$$\hat{\underset{\sim}{J}} = \begin{bmatrix} \frac{\partial}{\partial\Theta_1} f(\alpha, \hat{\underset{\sim}{\Theta}}), & \dots, & \frac{\partial}{\partial\Theta_r} f(\alpha, \hat{\underset{\sim}{\Theta}}) \\ \vdots & & \vdots \\ \frac{\partial}{\partial\Theta_1} f(\beta, \hat{\underset{\sim}{\Theta}}), & \dots, & \frac{\partial}{\partial\Theta_r} f(\beta, \hat{\underset{\sim}{\Theta}}) \end{bmatrix}$$

I programmet går vi ut fra at $\hat{\underset{\sim}{\Sigma}}_0/N$ er ei diagonalmatrise med diagonalelement lik $\hat{\lambda}_x/L_x$.

Programmet reknar vidare ut eit estimat for variansen til $f(x,\hat{\underset{\sim}{\Theta}})$ for $x = \alpha, ..,\beta$, og skriv den ut saman med $\hat{\lambda}_x$, estimatet $\hat{\lambda}_x/L_x$ for variansen til $\hat{\lambda}_x$, og $f(x,\hat{\underset{\sim}{\Theta}})$. Estimatet for varians-kovariansmatrisa til $\{f(x,\hat{\underset{\sim}{\Theta}}), x = \alpha, .., \beta\}$ finn vi som

$$\hat{\underset{\sim}{J}} (\hat{\underset{\sim}{\Sigma}}/N) \hat{\underset{\sim}{J}}^T.$$

I den spesielle delen av programsystemet finn vi rutinene som reknar ut funksjonsverdiane $f(x,\hat{\underset{\sim}{\Theta}})$ for gitt x og $\underset{\sim}{\Theta}$ og rutiner som reknar ut $\partial f/\partial\Theta_i$ for gitt x og alle i ($i = 1, ..,r$).

Til den spesielle delen av programsystemet kan vi óg rekne styringsprogramma og rutinene for å lese data. Observasjonar blir lest inn anten fra kort eller fra tape.

2.2. Estimeringsmetoden.

I avsnitt 2.1 er det kort sagt at vi finn estimatet av $\underset{\sim}{\theta}$ som den vektoren $\hat{\underset{\sim}{\theta}}$ som minimerer (2.1) med vektfunksjonen w_x.

Dersom $f(x,\underset{\sim}{\theta})$ er noko anna enn lineær i parametrane, må estimeringa til vanleg skje ved numerisk iterasjon. I den spesielle delen av programsystemet finn vi derfor rutiner som på grunnlag av observasjonane reknar ut foreløpig estimat av $\underset{\sim}{\theta}$ som blir brukt som startverdiar i den iterative estimeringa. Det er i litteraturen hovudsakleg to algoritmer som har vore nytta til å finne minimum av ein generell funksjon. Fletcher-Powell algoritmen (Fletcher & Powell, 1963) er implementert av Gruveaus & Jøreskog (1970). Nelder-Mead algoritmen (Nelder & Mead, 1965) er implementert av O'Neill (1971). Vi har funne Nelder-Mead-algoritmen betre eigna i dette programsystemet enn Fletcher-Powell-algoritmen, og har brukt den første av dei.

2.3. Oversikt over rutinene i programsystemet.

Programsystemet består av i alt 18 rutiner, som er lista opp i Appendiks A. Rutinene merka "x" i oversikten under er henta fra IBM SSP IV (1968), mens den som er merka "xx" er henta fra O'Neill (1971). Desse rutinene er utelatt fra listinga i Appendiks A.

Vi brukar desse generelle rutinene:

Funksjonsminimering etter Nelder-Mead algoritmen:

NELMIN **xx**,

utrekning av variansar (sjå avsnitt 2.1):

SIGMA 2 (reknar ut $\hat{\Sigma}/N$ og diagonalen til $\hat{J}(\hat{\Sigma}/N)\hat{J}^T$,

EJNUL (reknar ut $\hat{J}$),

MINV **x** (inverterer ei generell matrise),

utrekning av gammafunksjonen $\Gamma(t) = \int_0^\infty u^{t-1}e^{-u}du$:

DGAMMA **x**,

utrekning av den deriverte av $\ln\Gamma(t)$ (digammafunksjonen):

DGM,

utrekning av minimeringsfunksjonen (2.1):

FN.

Vi brukar desse <u>spesielle</u> rutinene:

Utrekning av Hadwigerfunksjonen for gitt x og $\underset{\sim}{\theta}$:

HAD3,

utrekning av den deriverte av Hadwigerfunksjonen med omsyn på parametrane for gitt x og $\underset{\sim}{\theta}$:

DHAD3,

utrekning av initialverdiar for parametrane i Hadwigerfunksjonen:

STEST1,

utrekning av gammatettheta for gitt x og $\underset{\sim}{\theta}$:

GAM3,

utrekning av den deriverte av gammatettheta med omsyn på parametrane for gitt a og $\underset{\sim}{\theta}$:

DGAM3,

utrekning av initialverdiar for parametrane i gammatettheta:

STEST4,

lesing av data på tape:

TINP,

styringsrutiner:

BLOCK DATA stiller opp felles dataområde ("COMMON"-områda),

MAIN MINSYS les parametrar fra kort, data fra kort eller tape (via TINP), skriv ut resultata av parameteranalysen og data som er lest inn. Den gir også kontrollen over til

NLMAIN som set parameterverdiar spesielle for NELMIN, estimerer startverdiar og set igang estimeringsalgoritmen. Den reknar også ut korrelasjonane mellom dei endelege parameter-estimata og skriv ut resultata omtalt i 2.1.

3. HADWIGERFUNKSJONEN

3.0. Den forma av Hadwigerfunksjonen som har vore anbefalt for analytisk glatting av fødselsrater, kan vi skrive som

$$h(x) = \frac{RH}{T\sqrt{\pi}} \left(\frac{T}{x+d}\right)^{\frac{3}{2}} \exp\left\{-H^2\left(\frac{T}{x+d} + \frac{x+d}{T} - 2\right)\right\} \text{ for } x > -d. \tag{3.1}$$

Nærmere omtale av Hadwigerfunksjonen fins m.a hos Hoem & Berge (1974).

3.1. Subrutina HAD3.

Vi innfører no størrelsane $W = T/H$, $D = T-d$, $S = W^2/2$, $M = -d+3T\ \{(1+16H^4/9)^{1/2}-1\}/4H^2$ og $Q = \{2(D-M)^2+6S\}/8W(D-M)$.

Vi finn da at $H = Q \pm (Q^2-3/4)^{1/2}$ og vel $H = Q + (Q^2-3/4)^{1/2}$ sidan det ikkje er vanleg med H-verdiar i nærleiken av null.

Vi set så $V = x-D+WH$ og fører dette inn i (3.1.). Det gir

$$(3.2)\qquad h_3(x) = \frac{RH\sqrt{WH}}{\sqrt{\pi}}\, V^{-3/2} \exp\,\{-H(x-D)^2/WV\}$$

for $x>D-T$. Formelen (3.2) er programmert som FUNCTION HAD3(P, A) der A svarar til x og P er parametervektoren (R, S, M, D).

3.2. Subrutina DHAD3.

Subrutina DHAD3 reknar ut vektoren

$$\left(\frac{\partial h_3}{\partial R}, \frac{\partial h_3}{\partial S}, \frac{\partial h_3}{\partial M}, \frac{\partial h_3}{\partial D}\right)$$

for gitt x og parametervektor (R, S, M, D).

Uttrykka som er programmert er

$$\frac{\partial h_3}{\partial R} = h_3(x, R, S, M, D)/R,$$

$$\frac{\partial h_3}{\partial S} = h_3(x, R, S, M, D)\ \{[WV^2 - 6HVS + 2H(x-D)^2\ (V + HW)]\ 8W(D-M)\ (Q^2-3/4)^{1/2} + (x-D)\ [3WV-2H(x-D)^2]\ [6S-2(D-M)^2\ ]\}/\{64S^2V^2(D-M)\ (Q^2-3/4)^{1/2}\},$$

$$\frac{\partial h_3}{\partial M} = h_3(x, R, S, M, D)\quad [3VW-2H(x-D)^2]\ (x-D)\ [6S-2(D-M)^2]/[32SV^2(D-M)^2\ (Q^2-3/4)^{1/2}],$$

og

$$\frac{\partial h_3}{\partial D} = h_3(x, R, S, M, D)\ \{[2H(V+WH)\ (x-D) + 3VW]\ 8W\ (D-M)^2\ (Q^2-3/4)^{1/2} + [3WV-2H(x-D)^2](x-D)\ [2(D-M)^2 - 6S\ ]\}/32SV^2(D-M)^2\ (Q^2-3/4)^{1/2}.$$

Når ein kallar på subrutina, må ein oppgi x, funksjonsverdi h_3 og parametervektor (R, S, M, D).

I CALL DHAD3 (P, A, HA, DH) svarar A til x, HA til $h_3(x, R, S, M, D)$ og P til (R, S, M, D). Svaret får vi tilbake i DH.

3.3. Initialverdiar av parametrane (R, S, M, D).

Subrutina STEST1 reknar ut dei startverdiane som blir nytta i estimeringsalgoritmen.

Vi har basert utrekningane på dei estimatorene som er foreslått av Yntema (1969). (Sjå også Hoem, 1970, pp. 589-591.)

Vi har observasjonane $\{\hat{\lambda}_x,\ x = \alpha, \alpha+1, \ldots, \beta\}$ og set $\hat{R}_0 = \sum_x \hat{\lambda}_x$, $\hat{R}_1 = \sum_x x\,\hat{\lambda}_x$, $\hat{D} = \hat{R}_1/\hat{R}_0$, $\hat{V} = [\hat{U} + 1/2]$ (her tyder $[y]$ heiltallsdelen av y), $\hat{h} = \hat{\lambda}_V$ og $\hat{M} = \min\{x: \hat{\lambda}_x \geq \hat{\lambda}_y$ for alle y$\}$. Då kan vi skrive Yntemaestimatorane for parametrane i (3.1) som

$$\hat{R}_y = \hat{R}_0,$$

$$\hat{T}_y = 3\hat{R}_0^2/4\pi(\hat{D}-\hat{M})\ \hat{h}^2,$$

$$\hat{H}_y = 3R_0/4\ \sqrt{\pi}(\hat{D}-\hat{M})\ \hat{h}$$

og

$$d_y = \hat{T}_y - \hat{D}.$$

Startestimatet for parametrane i (3.2) blir da:

$$\hat{R}_H = \hat{R}_0,$$

$$\hat{S}_H = \hat{R}_0^{\,2}/2\pi\hat{h}^2,$$

$$\hat{M}_H = -\,\hat{d}_y + 3\hat{T}_y\left(\sqrt{1+16\hat{H}_y^{\,4}/9} - 1\right)/4\hat{H}_y^2$$

og

$$\hat{D}_H = \hat{D}.$$

4. GAMMAFUNKSJONEN

4.0. Følgande modifisering av gammatettheta nyttast ofte til analytisk glatting av fødselsrater (Keyfitz (1968), pp. 147):

(4.1) $g(x) = R\,c^k x^{k-1}\,e^{-cx}/\Gamma(k)$ for $x > 0$.

4.1. Subrutina GAM3.

Analogt til det som er gjort med Hadwiger-funksjonen, fører vi inn ein fjerde parameter d, slik at vi får

$$(4.2) \quad g_0(x) = Rc^k(x+d)^{k-1}\exp\{-c\,(x+d)\}/\Gamma(k) \text{ for } x < -d.$$

Vi innfører no $Y = [k/c] - d$, $M = [(k-1)/c] - d$ og $S = k/c^2$.

Det gir $k = S/(Y-M)^2$, $c = 1/(Y-M)$ og $d = S/Y-M$.

Innsett i (4.2) gir dette

$$(4.3) \quad g_3(x) = R(Y-M)^{-\frac{S}{(Y-M)^2}}\{x-Y+S/(Y-M)\}^{\frac{S}{(Y-M)^2}-1} \cdot$$

$$\exp\{-(x-Y+S/(Y-M))/Y-M\}/\Gamma\{S/(Y-M)^2\} \text{ for } x > Y-S/(Y-M).$$

Formelen (4.3) er programmert som FUNCTION GAM3(P, AL) der AL svarar til x og P er parametervektoren (R, Y, S, M).

4.2. Subrutina DGAM3.

Subrutina DGAM3 reknar ut vektoren

$$\left(\frac{\partial g_3}{\partial R}, \frac{\partial g_3}{\partial Y}, \frac{\partial g_3}{\partial S}, \frac{\partial g_3}{\partial M}\right)$$

for gitt x og parametervektor (R, Y, S, M). Utrykka som er programmert, er

$$\frac{\partial g_3}{\partial R} = g_3(x, R, Y, S, M)/R \text{ og}$$

$$\frac{\partial g_3}{\partial Y} = g_3(x, R, Y, S, M)\,\{(Y-M)\,(x-M)\,[S+(Y-M)^2]/[S+(x-Y)\,(Y-M)]$$

$$+ (x-Y)\,(Y-M) - 2S\,[\ln\{(S+(x-Y)\,(Y-M))/(Y-M)^2\}$$

$$- \Psi(S/(Y-M)^2)]\}/(Y-M)^3.$$

Her er $\Psi(y) = d \ln \Gamma(y)/dy$ (digammafunksjonen).

For $1 \leq q \leq 2$ finn vi $\Psi(q)$ ved lineær interpolasjon i Table VIII i Milne (1949) over digammafunksjonen. Vi nyttar så rekursjonsformelen

$$\Psi(q+1) = \Psi(q) + 1/q$$

for å finne den verdien vi er interessert i. Dette skjer i subrutina DGM.

Vi har vidare

$$\frac{\partial g_3}{\partial S} = g_3(x, R, Y, S, M)\ \{\ln\ [(S+(x-Y)\ (Y-M))/(Y-M)^2\ ]$$

$$-\ \Psi\ [S/(Y-M)^2\ ]\ -\ (x-M)\ (Y-M)/[S+(x-Y)\ (Y-M)]\}/(Y-M)^2 \quad \text{og}$$

$$\frac{\partial g_3}{\partial M} = g_3(x, R, Y, S, M)\ \{2S\ [\ln\ ((S+(x-Y)\ (Y-M))/(Y-M)^2)$$

$$-\ \Psi(S/(Y-M)^2)]\ -\ S(x-M)\ (Y-M)/[S\ +$$

$$(x-Y)\ (Y-M)]\ -\ (x-Y)\ (Y-M)\}/(Y-M)^3.$$

Når ein kallar på subrutina DGAM3 må ein oppgi x, funksjonsverdi g_3 og parametervektor (R, S, M, D).

I CALL DGAM3(P, A, HA, DH) svarar A til x, HA til h_3 og P til (R, Y, S, M). Svaret får vi tilbake i DH.

4.3. Initialverdiar for parametrane (R, Y, S, M).

Subrutina STEST4 reknar ut dei startverdiane som blir nytta i estimeringsalgoritmen. Estimatorene er utvikla ved momentmetoden.

Vi set no

$$R_n = \int_{-d}^{\infty} x^n\ g_0(x)\ dx \text{ for } n \geq 0$$

og

$$R_n^* = R_n/R_0 \text{ for } n > 0.$$

For n = 0 gir dette R_0 = R.

Vi finn vidare at

$$R_n^* = \sum_{i=0}^{n} \binom{n}{i} c^i(-d)^{n-i}\ \Gamma(k+i)/\Gamma(k).$$

Av denne relasjonen finn vi R_n^* for n = 1, 2 og 3. Ved å innføre

$$Y = R_1^* \text{ og } S = R_2^* - (R_1^*)^2$$

får vi

$$k = (Y+d)^2/S,$$

$$c = (Y+d)/S$$

og

$$d = -Y + 2S^2/(R_3^*-Y^3-3YS).$$

Størrelsene Y og S er forventning og varians i fordelinga $g_0(x, R, k, c, d)/R$. Modalverdien i denne fordelinga finn vi som $M = (k-1)/c-d$.

Vi har no observasjonane $\{\hat{\lambda}_x, x = \alpha, \alpha+1, .., \beta\}$. Vi finn da start-estimatorar for parametrane (R, Y, S, M) ved å sette

$$\hat{R}_n = \sum_x x^n \hat{\lambda}_x \text{ for } n \geq 0$$

og

$$\hat{R}^*_n = \hat{R}_n/\hat{R}_0 \text{ for } n > 0.$$

Dette gir

$$\hat{R}_G = \hat{R}_0,$$

$$\hat{Y}_G = \hat{R}^*_1,$$

$$\hat{S}_G = \hat{R}^*_2 - (\hat{R}^*_1)^2,$$

$$\hat{d}_G = -\hat{R}^*_1 + 2\hat{S}^2_G/\left[\hat{R}^*_3 - (\hat{R}^*_1)^3 - 3(\hat{R}^*_1)\hat{S}_G\right]$$

og

$$\hat{M}_G = \hat{R}^*_1 - \hat{S}_G/(\hat{R}^*_1+\hat{d}_G).$$

5. OM BRUKEN AV PROGRAMSYSTEMET

5.1. Data til programsystemet.

Data til programsystemet er vanlegvis

(i) ein vektor $\{\hat{\lambda}_x, x = \alpha, \alpha+1, ..., \beta\}$ (ratevektoren) med observerte rater for aldrane $\alpha, \alpha+1, ..., \beta$, og

(ii) ein vektor $\{L_x, x = \alpha, .., \beta\}$ (basisvektoren) med antall personer i nemnarane til ratene.

Ein går ut ifrå at raten er multiplisert med ein mill. og basis med ti og at dei er utan desimalar ut over dette.

Desse data kan lesast inn anten fra kort eller fra tape. Ved lesing fra tape nyttar ein logisk eining nr. 14 (READ (14, 11) ....). Ved lesing fra kort nyttar ein logisk eining nr. 7 (READ (7, 12) ...).

Når data ligg på kort, kan ein utelate vektoren $\{L_x\}$. Ein kan då ikkje få modifisert minimum-kji-kvadratestimat for parametrane. Ein kan også hoppe over lesing av data om ein vil nytte samme data ved ulike estimeringar.

Parameteren K2 styrer dette:

K2 = 1 - bruk data fra førre estimering,

K2 = 2 - les data for $\{\hat{\lambda}_x\}$ og $\{L_x\}$ på kort,

K2 = 3 - les data for $\{\hat{\lambda}_x\}$ og $\{L_x\}$ på tape,

K2 = 4 - les data for berre $\{\hat{\lambda}_x\}$ fra kort, og

K2 = 5 - les data for $\{\hat{\lambda}_x\}$ og $\{L_x\}$ på tape og foreta seleksjonar ifølge liste på kort.

Når data ligg på tape, går programmet ut fra at det først ligg to firesifra identifikasjonsnr., så kjem ratevektoren med sju posisjonar pr. datum, og til slutt basisvektoren også med sju posisjonar pr. datum. Lesinga skjer altså for K2 = 3 og K2 = 5 etter formatet

```
11 FORMAT (2F4.0, 2(150F7.0))
```

(maksimum antall observasjonar er sett til 150).

Seleksjonane skjer etter dei fire første siffera i identifikasjonsnummeret. For kvar seleksjon må vi punche eit seleksjonskort med identifikasjonsnummeret i dei fire første posisjonane (lesing med formatet 10 FORMAT (I4)). Seleksjonskorta må vere sortert i samme rekkefølge som datasettet er sortert på tapen.

Når data ligg på kort, går ein ut fra at første kortet inneheld ein alfanumerisk identifikasjon på inntil 78 posisjonar (pos. 1-78 på kortet). På neste kort startar ratevektoren som ein tenker seg er puncha fortløpande med ti observasjonar pr. kort og sju posisjonar pr. observasjon. På det kortet der siste rateobservasjon kjem, startar basisvektoren umiddelbart og blir puncha på samme måte som ratevektoren. Lesinga skjer altså for K2 = 2 og K2 = 4 etter formatet

```
12 FORMAT (13A6, 30(/10F7.0)).
```

Dersom ein har data organisert på ein annan måte enn det som programmet går ut ifra, er det enkelt å endre disse to formatstatementa. Format nr. 12 ligg i hovedprogrammet og format nr. 11 ligg i TINP.

5.2. Parametrane i programsystemet.

5.2.0. Ein kan skille mellom to klasser av parametrar. Den eine klassen som ein treng å endre relativt ofte, blir det lest inn verdiar for fra eit parameterkort. Den andre klassen blir gitt verdiar i det programmet blir lagt inn i eit programbibliotek. Vi finn parametrar i hovedprogrammet og i NLMAIN.

5.2.1. Parametrane i parameterkortet.

Parameterkortet blir lest i hovedprogrammet og parameterverdiane blir sjekka om dei er lovlege og innbyrdes konsistente. Dersom det finst ulovlige verdiar blir det gitt default-verdi til parameteren (default-verdi er den verdi som ein normalt ventar vil gjelde). Parametrane i parameterkortet er i

posisjon	1 - 2	K2	for valg av data og innlesingsmåte,
posisjon	3 - 5	NN	som gir antall element i datavektorane,
posisjon	6 - 8	NA	som gir alder ÷ 1 for første observasjon,
posisjon	9 - 10	KT	som set oss i stand til å modifisere aldersgrensene i glattinga,
posisjon	11 - 12	K3	som seier kva for metode vi vil nytte til å estimere glattingsfunksjonen,
posisjon	13 - 14	KL	som seier kva for glattingsfunksjon vi vil nytte,
posisjon	15 - 16	N	som gir antall parametrar i glattingsfunksjonen,
posisjon	17 - 18	I2	som seier om vi må lese inn fra kort dei parametrane som skal nyttast i konstruksjonen av startsimplexet,
posisjon	19 - 21	NK	som gir antall datasett som skal nytte dei samme parametrane,
posisjon	22 - 24	IKN	som seier kva alder utskrifta skal starte ved og i
posisjon	25 - 27	IKM	som seier kva alder utskrifta skal slutte ved.

Parameteren K2 har vi omtala under punkt 5.1. Dafaultverdien for denne er K2 = 2. Parameteren NN kan vere maksimum 150 og minimum 2. Dafaultverdi er 36. NN må gi det faktiske antall element i ratevektoren $\{\hat{\lambda}_x\}$ unntatt når K2 = 1 då vi kan selektere innanfor dei alt innleste data. Parameteren NA kan vere minimum 0 og maksimum 148. Defaultverdi er 14.

Dersom KT = 0, skal vi modifisere grensene for glattinga. Dersom $\hat{\lambda}_x * L_x \leq 5$ for nokon $x = \alpha, \alpha+1, .., \beta$, set vi $\hat{\lambda}_x = 0$ for denne verdi av x. Dersom no $\hat{\lambda}_\alpha = \hat{\lambda}_{\alpha+1} = .. = \hat{\lambda}_{\alpha+i-1} = 0$ og $\hat{\lambda}_{\beta-j+1} = ... = \hat{\lambda}_{\beta-1} = \hat{\lambda}_\beta = 0$, vil glattinga berre skje mellom aldrane fra og med $\alpha+i$ til og med $\beta-j$ (dvs. NA = $\alpha+i-1$ og NN = $\beta-j-\alpha-i+1$). Glattinga vil ifra programmet alltid skje fra og med alder NA+1 til og med alder NA+NN. Dersom KT = 1, vil aldersgrensene ikkje bli modifisert. Men dersom det faktisk er observert ein $\hat{\lambda}_x = 0$ vil denne bli utelatt under estimeringa sjølv om det ikkje kjem til syne i utskriftene. Defaultverdi er KT = 0.

Parameteren K3 gir oss valget mellom å estimere parametervektoren i modellen vår med standard minste kvadraters metode (K3 = 0) eller med ein medifisert minimum-kji-kvadratmetode (K3 = 1). Defaultverdi er K3 = 1.

Skal vi nytte andre metodar må vi lese inn vektfunksjonen $\{w_x\}$ fra kort. Vi må da ha K3 = 2.

Parameteren KL gir valget mellom å glatte ratene med ein brukarutvikla funksjon (KL = 0) kalla FFFF i programmet, eller å nytte ein Hadwigerfunksjon (KL = 1) eller ei modifisert gammatetthet (KL = 2). Dafaultverdi er KL = 1.

Parameteren N gir antall parametrar i glattingsfunksjonen. Dafaultverdi er N = 4. Parameteren I2 seier om vi må lese inn fra kort dei verdiane rutina NELMIN nyttar til å konstruere startsimplexet i minimeringsalgoritmen. Når I2 = 1 les vi verdiane fra kort. Defaultverdiar er STEP(1) = 0.1 og STEP(2) = ... = STEP(N) = 1.0. Når vi har vektfunksjon eller konstruksjonsverdiar for startsimplexet (eller begge) på kort, må korta med verdiar for desse komme mellom parameterkortet og datakorta og med vektfunksjonen først. Begge må punchast med 8 posisjonar pr. verdi. Desimalpunktet må punchast og vi kan ha maksimum 3 desimalar.

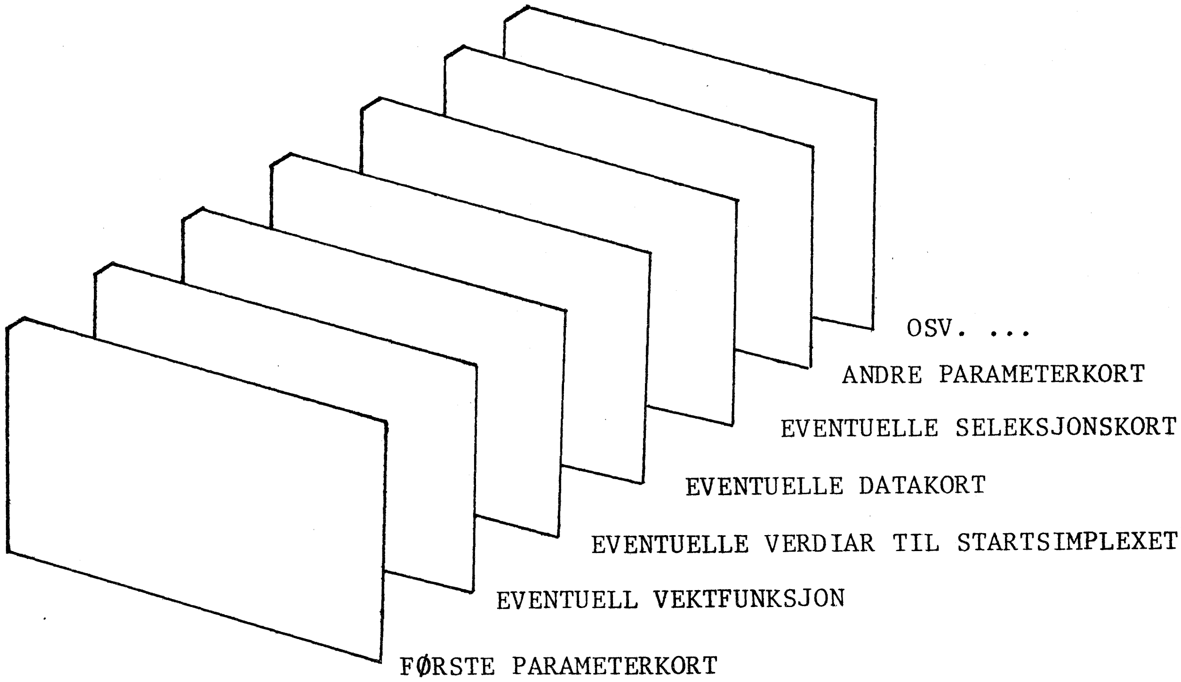
Parameteren NK gir antall glattingar som skal gjerast med konstante parameterverdiar. Defaultverdi er NK = 1. Maksimalverdi er NK = 999. Parametrane IKN og IKM gir første og siste alder vi ønskjer utskrift for. Her kan vi godt ha IKN < NA+1 (alder for første observasjon) eller IKM÷IKN+1 > NN (antall observasjonar). Vi må imidlertid ha at 0 < IKM-IKN+1<150. Dersom denne størrelsen er mindre eller lik null set programmet inn verdiane IKN=16 og IKM=44. Dersom IKM-IKN+ ≥150, set programmet inn verdiane IKN=1 og IKM=50.

5.2.2. Andre parametrar.

I hovedprogrammet har vi ein parameter K1 som gir det maksimale antall parameterkort som kan lesast inn i ein omgang. Her er dette sett til 500. Vi får då maksimalt 999 x 500 glattingar pr. køyring. (999 repetisjonar av samme parameterkortet).

I rutina NLMAIN er det gitt verdiar til nokre av parametrane brukt i minimeringsalgoritmen NELMIN. Meir om desse parametrane finn vi hos O'Neill (1971).

5.2.3. Kortoppsettet ved bruken av programsystemet.



Vi kan ha maksimum 500 parameterkort. Alle kort blir lest i hovedprogrammet unntatt seleksjonskorta som blir lest ifra TINP når data ligg på tape.

Formata som blir nytta under lesinga av dei ulike korta er:
parameterkortet:

30 FORMAT (I2, 2I3, 5I2, 3I3)

vektfunksjon:

13 FORMAT (30(10F8.3))

og for verdiane til startimplexet:

14 FORMAT (10F8.3).

Formatstatementa for lesing av data er gitt i del 5.1.

5.3. Endringar i programsystemet ved bruk av andre glattingsfunksjonar.

Programsystemet slik det er presentert her, gir eit eksempel på ein ikkje-lineær regresjon. Dersom ein kan tenke seg andre samband mellom den frie variabelen (x) og den observerte variabelen ($\hat{\lambda}_x$) enn det Hadwigerfunksjonen eller gammafunksjonen representerer, vil det vere relativt enkelt å nytte systemet til å estimere parametrane i den nye relasjonen. Kravet må då vere at antall parametrar i den nye relasjonen er mindre enn 10 og antall observasjonar i datasettet mindre enn 150. Vi treng da

- ei ny rutine for å rekne ut verdien av den nye funksjonen i punktet x og for parametervektoren P. Vi må ha FUNCTION FFFF (P, X) som første statement i denne rutina.
- ei ny rutine for å rekne ut den deriverte i punktet x av den nye funksjonen med omsyn på kvar av parametrane i P. Første statement i rutina må vere

  SUBROUTINE DFFFF(P, X, HA, DH).

  Her er HA = FFFF(P, X) og resultatet må plasserast i DH.
- ei ny rutine for å rekne ut startestimat til estimeringsalgoritmen. Første statement i denne må vere

  SUBROUTINE FSTEST(N, P)

  der N er antall parametrar og P parametervektoren.

LITTERATUR

Fletcher, R. and Powell, M.J.D. (1964): "A Rapidly Convergent Descent Method for Minimization." The Computer Journal, Vol. 6, pp. 163 ...

Gruvaeus, G.T. and Jöreskog, K.G. (1970): "A Computer Program for Minimizing a Function of Several Variables." Research Bulletin RB-70-14. Educational Testing Service, Princeton, New Jersey.

Hoem, J.M. (1970): "On the Statistical Theory of Analytic Graduation." Proceedings of the Sixth Berkeley Symposium on Mathematical Statistics and Probability. Vol. 1, pp. 569-600. University of California Press, 1972. Artikler, nr. 49, Statistisk Sentralbyrå, Oslo, 1972.

Hoem, J.M. et al.(1974): "Two papers on analytic graduation". Working Paper ANO IO 74/17. Central Bureau of Statistics of Norway.

Hoem, J.M. & Berge, E. (1974): "Some problems in Hadwiger fertility graduation." Working Paper ANO IO 74/5, Statistisk Sentralbyrå, Oslo.

IBM SSP IV (1968): "System /360 Scientific Subroutine Package (360A-CM-03X) Version III." H20-0205-3. IBM, New York.

Keyfitz, N. (1968): "Introduction to the Mathematics of Population." Addison & Wesley, Reading, Mass.

Milne, W.E. (1949): "Numerical Calculus." Princeton University Press, New Jersey.

Nelder, J.A. and Mead, R. (1965): "A Simplex Method for Function Minimization." The Computer Journal, Vol. 7, pp. 308-313.

O'Neill, R. (1971): "Function Minimization using a Simplex Procedure." Journal of the Royal Statistical Society (Series C). Vol. 20, pp. 338-345.

Yntema, L. (1969): "On Hadwiger's Fertility Function." Statistical Review of the Swedish National Central Bureau of Statistics, Series III. Vol. 7, pp. 113-117.

APPENDIKS A

PROGRAMLISTING MED EKSEMPEL PÅ UTSKRIFT

Programlistinga er fra implementeringa på ein Honywell 6060 maskin og inneheld ikkje rutinene

NELMIN (O'Neill (1971)),
MINV (IBM SSP IV (1968)) og
DGAMMA (IBM SSP IV (1968)).

Desse rutinene finn vi i dei nemnte publikasjonane, og det er berre DGAMMA som er særleg endra før den er brukt. I publikasjonen heiter den GMMMA og er ei subrutine. Endringane før den går inn i systemet her gjeld følgande statement:

```
kolonne 7                                                   73
        .
        FUNCTION DGAMMA(XX)                                 GMMM0330
        IMPLICIT REAL*8(A-H, P-Z)                           GMMM0331
        COMMON/DAT4/IXI                                     GMMM0332
        REAL*8 XX,DGAMMA                                    GMMM0333
        .
        .
        .
        .
        IXI = IXI+1                                         GMMM0359
        GX = 1.D36                                          GMMM0360
        DGAMMA = GX                                         GMMM0361
        IF(IXI.EQ.11) RETURN                                GMMM0362
        WRITE(8, 1010) IER                                  GMMM0363
        IF(IXI.  EQ.10)WRITE(8,996)                         GMMM0364
        .
        .
        .
        .
        DGAMMA = GX                                         GMMM0651
        .
        .
        .
        DGAMMA = 0.                                         GMMM0671
        IF(IXI.EQ.11) RETURN                                GMMM0672
        IXI = IXI+1                                         GMMM0673
        WRITE(8,1010) IER                                   GMMM0674
        IF(IXI.EQ.10) WRITE(8,996)                          GMMM0675
        RETURN                                              GMMM0680
 996    FORMAT(1H ,42H THIS IS THE LAST TIME THIS
        MESSAGE OCCURS )                                    GMMM0681
1010    FORMAT(1H ,25H SUBROUTINE DGAMMA..IER= , I4)        GMMM0682
        END                                                 GMMM0690
```

I MINV er det gjort berre dei endringane som er nødvendig for at den skal ha dobbel presisjon og gå på H6060-maskinen. Det gjeld følgande statement:

```
kolonne 7                                                       73
        IMPLICIT REAL*8 (A-H, P-Z)                              MINV0331
        REAL*8 A(1), D                                          MINV0332
        DIMENSION L(1), M(1)                                    MINV0340
```

I NELMIN er DOUBLE PRECISION statementet erstatta med dei tre statementa

```
IMPLICIT REAL*8(A-H, P-Z)
REAL*8 START(1), STEP(1), MIN(1), REQMIN, YNEWLO
DIMENSION P(20,21), PSTAR(20), P2STAR(20), PBAR(20), Y(20)
```

Fjorten linjer lenger nede i programlistinga finn vi statementa DN = DFLOAT(N) og DNN = DFLOAT(NN). Desse er erstatte med DN = N og DNN = NN.

Til utskriftene fra programmet skal det berre knyttast nokre få kommentarar. Kodeliste for parametrane finn vi i kap. 5. Antall omstart av NELMIN ganger ti tusen pluss antall ganger NELMIN har rekna ut FN er kalla ICOUNT i utskrifta.

I matrisa med kovariansar, variansar og korrelasjonar mellom dei endelege parameterestimata ligg kovariansane i nedre (venstre) triangelen, korrelasjonane i øvre (høgre) triangelen og variansane langs diagonalen.

I utskrifta finn vi ein verdi for minimaliseringsfunksjonen. Denne verdien er for K3 = 1 lik kji-kvadratverdien for tilpasninga mellom aldersgrensene som er nytta i tilpasninga. For K3 = 0 får vi ut sum kvadratavvik mellom dei samme aldersgrensene. Til slutt i utskrifta får vi skrive ut sum kvadratavvik og kji-kvadratverdi for tilpasninga mellom aldrane IKN og IKM og mellom aldrane 20 og 30.

```
      BLOCK DATA
      IMPLICIT REAL*8(A-H,P-Z),LOGICAL*1(O)
      REAL*4 A(190),B(190)
      COMMON/DAT/A,NN,KL,K3
      COMMON/DAT1/N,I2
      COMMON/DAT2/B
      COMMON/DAT3/Q
      COMMON/DAT4/IXI
      COMMON/DAT5/S(150),WX(150)
      COMMON/DAT6/NA,IN,IM
      COMMON/DAT7/STEP(20)
      COMMON/DAT8/IKN,IKM,IS1,IS2,IS3
      COMMON/DAT9/NAS,NNS,IKK,C(190)
      DATA IXI/0/
      END
```

```
C     PROGRAM MAIN MINSYS
      IMPLICIT REAL*8(A-H,P-Z),LOGICAL*1(O)
      REAL*4 A(190),B(190)
      COMMON/DAT/A,NN,KL,K3
      COMMON/DAT1/N,I2
      COMMON/DAT2/B
      COMMON/DAT5/S(150),WX(150)
      COMMON/DAT6/NA,IN,IM
      COMMON/DAT7/STEP(20)
      COMMON/DAT8/IKN,IKM,IS1,IS2,IS3
      COMMON/DAT9/NAS,NNS,IKK,C(190)
      DIMENSION P(20),TAGG(13)
C
C     DESCRIPTION OF PARAMETERS
C
C     TYPE OF METHOD
C
C     MP=2 NELDER-MEAD
C
C
C     REPETITION OF PARAMETERCARD
C     NK GIVES NUMBER OF CASES WITH EQUAL PARAMETERS
C     DEFAULT IS NK=1
C
C     K1 GIVES
C     NUMBER OF OBSERVATIONSETS(CURVES)
C     DEFAULT IS K1=500
C
C     N GIVES
C     NUMBER OF VARIABLES(PARAMETERS)
C     DEFAULT IS N=4
C
C     AGE BOUNDARIES OF THE GRADUATION
C
C     NA GIVES AGE MINUS ONE AT WHICH GRADUATION IS TO START
C     DEFAULT IS NA=14
C
C     NN GIVES
C     NUMBER OF OBSERVATIONS(AGEGROUPS)
C     FIRST OBSERVATION IS AT AGE NA+1
```

```
C      DEFAULT IS NN=36
C
C
C      IKN GIVS AGE AT WHICH PRINTOUT WILL START
C      IKM GIVS AGE AT WHICH PRINTOUT WILL STOP
C
C      IF IKM-IKN>150 THE PROGRAM ASUMES THAT IKN=1 AND IKM=50
C
C
C      KT=0 ESTIMATE NEW BOUNDARIES
C      KT=1 DO NOT ESTIMATE NEW BOUNDARIES
C      DEFAULT IS KT=0
C
C
C      TYPE OF INPUT
C
C      K2=1 REPEAT DATA FROM LAST RUN
C      K2=2 OBSERVATIONS ON CARDS
C      K2=3 OBSERVATIONS ON TAPE
C      K2=4 OBSERVATIONS ON CARDS(BASE IS MISSING)
C      K2=5 OBSERVATIONS ON TAPE WITH SELECTIONS
C      DEFAULT IS K2=2
C
C
C
C      FUNCTIONTYPE OR
C      TYPE OF PARAMETERREPRESENTATION
C
C      KL=0 FFFF GENERAL FUNCTION SUPPLIED BY USER
C      KL=1 HAD3 R,S,M,Y
C      KL=2 GAM3 R,Y,S,M
C      DEFAULT IS KL=1
C
C
C      CONSTRUCTION OF STARTSIMPLEX
C
C      I2=0 USE VALUES SUPPLIED BY PROGRAM
C      I2=1 READ VALUES FROM CARDS
C      DEFAULT IS I2=0
C
```

```
C
C
C
C      TYPE OF MINIMIZATION
C      K3=0 STANDARD LEAST SQUARES
C      K3=1 MODIFIED MINIMUM CHI-SQUARES
C      K3=2 WEIGHT FUNCTION SUPPLIED BY USER
C      DEFAULT IS K3=1
C
C
C      KC GIVES NUMBER OF ENTRIES TO TINP
C
C
C
C      ******************************************
C
C      PARAMETERSECTION
C
C      SEE ALSO BLOCK DATA SUBPROGRAM
C
       K1=500
       KC=0
       MP=2
C
C
       DO 1 J1=1,K1
       READ(7,30,END=9)K2,NN,NA,KT,K3,KL,N,I2,NK,IKN,IKM
       WRITE(8,11)J1
       WRITE(8,32)     K2,NN,NA,KT,K3,KL,N,I2,NK,IKN,IKM
C
C
    56 IF(I2)58,59,57
    58 I2=0
       GO TO 59
    57 IF(I2-1)58,59,58
    59 IF(K2-1)62,61,60
    61 IF(J1-1)62,62,65
    62 K2=2
       GO TO 65
    60 IF(K2-5)65,65,62
```

```
   65 IF(NN-1)67,67,68
   67 NN=36
      GO TO 69
   68 IF(NN-150)69,69,67
   69 IF(NA)70,72,71
   70 NA=14
      GO TO 72
   71 IF(NA-149)72,70,70
   72 NNA=NN+NA
      IF(NNA-150)87,87,86
   86 NN=36
      NA=14
   87 IF(KT)73,74,75
   73 KT=0
      GO TO 74
   75 IF(KT-1)74,74,73
   74 IF(K3)76,77,78
   76 K3=1
      GO TO 77
   78 IF(K3-1)76,77,88
   88 IF(K3-2)76,77,76
   77 IF(KL)79,80,84
   84 IF(KL-1)79,80,81
   79 KL=1
      GO TO 80
   81 IF(KL-2)79,80,79
   80 IF(N-1) 82,83,85
   82 N=4
      GO TO 83
   85 IF(N-10)83,83,82
   83 IF (NK) 89,89,90
   89 NK=1
   90 IKK=IKM-IKN+1
      IF(IKK-150)95,95,94
   94 IKN=1
      IKM=50
   95 IF(IKK-1)96,96,97
   96 IKN=16
      IKN=44
   97 CONTINUE
```

```
      IF (K2.EQ.1) NK=1
      IF(K2.EQ.4)K3=0
      IF(K2.EQ.4)KT=1
      WRITE(8,33)     K2,NN,NA,KT,K3,KL,N,I2,NK,IKN,IKM
      KL=KL*3
      K3=K3*3
      NNS=NN
      NAS=NA
      IF(K2-1)91,92,91
   91 NN1=NN
      NA1=NA
   92 CONTINUE
      DO 93 I=1,IKK
   93 WX(I)=0.
C
C     DATAINPUT
C
      IF(K3.EQ.6)READ(7,13,END=9)(WX(I),I=1,NN)
      IF(I2.EQ.1)READ(7,14,END=9)(STEP(I),I=1,N)
      DO 1 K4=1,NK
      NN=NN1
      NA=NA1
      IF(K2.EQ.1) GO TO 31
      IF (K2.EQ.2) READ (7,12,END=9) (TAGG(I),I=1,13), (A(I),I=1,NN),
     1(B(I),I=1,NN)
      IF (K2.EQ.4) READ (7,12,END=9) (TAGG(I),I=1,13), (A(I),I=1,NN)
      IF(K2-5)7,4,7
    7 IF(K2-3)3,4,3
    4 CALL TINP(KC,K2)
    3 CONTINUE
      DO 41 I=1,NN
      C(I)=A(I)*1.D-6
      B(I)=B(I)*1.D-1
   41 A(I)=A(I)*1.D-6
      IF(K2-2)45,46,45
   45 IF(K2-4)47,46,47
   46 WRITE(8,17)(TAGG(I),I=1,13)
      GO TO 48
   47 WRITE(8,18)A(NN+1),A(NN+2)
   48 CONTINUE
```

```
      DO 25 I=1,NN
   25 WRITE(8,19)(I+NA),A(I),B(I)
      GO TO 43
   31 DO 42 I=1,NN
   42 A(I)=C(I)
      NA=NAS
      NN=NNS
      NAS=NA1
      NNS=NN1
   43 IF(KT-1)44,29,44
   44 CONTINUE
C
C
C     BOUNDARIES ARE ESTIMATED
C
      WRITE(8,21)
      DO 23 I=1,NN
      IF(K2.EQ.4)B(I)=100000.
      SX=A(I)*B(I)
      IF(SX-5.)6,6,23
    6 A(I)=0.
      WRITE(8,22)(I+NA)
   23 CONTINUE
      NB=NN/2
      KO=0
      DO 24 I=1,NB
      IF(A(I))27,27,26
   27 NA=NA+1
      KO=KO+1
   24 NN=NN-1
   26 NO=NN+KO+1
      DO 28 I=1,NB
      IF(A(NO-I))28,28,29
   28 NN=NN-1
   29 WRITE(8,34)NA+1,NA+NN
C
      IN=NA-NA1+1
      IM=IN+NN-1
    2 IF (MP.EQ.2) CALL NLMAIN (P,K2)
    1 CONTINUE
```

```
      9 STOP
     11 FORMAT(1H1,9HTABLE NO ,I4/)
     12 FORMAT (13A6, 30(/10F7.0))
     13 FORMAT(30(/10F8.3))
     14 FORMAT(10F8.3)
     17 FORMAT (1H1,23H OBSERVATIONS ARE FROM ,13A6/10H       AGE ,
       110H      RATE ,10H   BASE    )
     18 FORMAT(1H1,23H OBSERVATIONS ARE FROM ,F4.0,A5/10H       AGE ,
       110H      RATE ,10H   BASE    )
     19 FORMAT(4X,I5,F10.6,F10.1)
     21 FORMAT(1H0,58H RATE IS SET TO ZERO IF BASE*RATE<=5 THIS IS DONE FO
       1R AGES//)
     22 FORMAT(I5)
     30 FORMAT(I2,2I3,5I2,3I3)
     32 FORMAT(1H ,41H PARAMETER CARD READ FROM SYSIN CONTAINS ,I2,2I3,5I3
       1,3I3)
     33 FORMAT(//1H ,83HAFTER ANALYSIS OF PARAMETER CARD THE FOLLOWING PAR
       1AMETERS ARE SUPPOSED TO BE USED   //1X,
       215H TYPE OF INPUT ,I2/1X,
       330H NUMBER OF OBSERVATION POINTS ,I3/1X,
       431H AGE-1 WHERE GRADUATION STARTS ,I3/1X,
       544H ESTIMATION OF BOUNDARIES OF THE GRADUATION ,I2/1X,
       629H TYPE OF PARAMETER ESTIMATES ,I2/1X,
       723H TYPE OF FUNCTION USED ,I2/1X,
       843H METHOD OF MINIMIZATION IS NELDER-MEAD      /1X,
       931H NUMBER OF FUNCTION PARAMETERS ,I2/1X,
       A36H VALUES FOR CONSTRUCTION OF SIMPLEX ,I2/1X,
       B39H NUMBER OF CASES WITH EQUAL PARAMETERS ,I3/1X,
       C24H PRINTOUT STARTS AT AGE ,I3/1X,
       D24H PRINTOUT STOPS AT AGE  ,I3/1X)
     34 FORMAT(1H ,41H LIMITS OF THE GRADUATION ARE TO BE FROM ,I3,4H TO ,
       1I3)
        END
```

```
      SUBROUTINE NLMAIN(P,K2)
      IMPLICIT REAL*8(A-H,P-Z)
      REAL*4 A(190),B(190)
      REAL*8 P(1),MIN(20)
      COMMON/DAT/A,NN,KL,K3
      COMMON/DAT1/N,I2
      COMMON/DAT2/B
      COMMON/DAT5/S(150),WX(150)
      COMMON/DAT6/NA,IN,IM
      COMMON/DAT7/STEP(20)
      COMMON/DAT8/IKN,IKM,IS1,IS2,IS3
      COMMON/DAT9/NAS,NNS,IKK,C(190)
      DIMENSION S2(150),Z(10,10)
C
C     PARAMETERS
C
C     SEE LISTING OF NELMIN FOR DESCRIPTION OF THEESE PARAMETERS
C
      REQMIN=1.D-16
      KONVGE=5
      ICOUNT=1000
      IF(I2.EQ.1) GO TO 24
      STEP(1)=.1
      DO 23 I=2,N
   23 STEP(I)=1.
   24 CONTINUE
C
C     **********************************
C
      DO 6 I=1,IKK
      S2(I)=0.
    6 S(I)=0.
      DO 5 I=IN,IM
      IF(K3-6)7,5,7
    7 WX(I)=1.
      IF(K3-3)5,8,5
    8 WX(I)=0.
      IF(A(I))5,5,9
    9 WX(I)=B(I)/A(I)
    5 S(I)=A(I)/B(I)
```

```
C
C     STARTESTIMATION
C
      IF(KL.EQ.0) CALL FSTEST(N,P)
      IF(KL.EQ.3) CALL STEST1(N,P)
      IF(KL.EQ.6) CALL STEST4(N,P)
C
C
C
C
C     START MINIMIZATION
C
      CALL NELMIN(N,P,MIN,YNEWLO,REQMIN,STEP,KONVGE,ICOUNT)
C
C     WRITE OUT RESULTS
C
      IF(ICOUNT)4,3,3
    4 WRITE(8,12)ICOUNT
      STOP
    3 WRITE(8,13)ICOUNT,YNEWLO,(MIN(I),I=1,N)
C
C     COMPUTE VARIANCES
C
      IF(K2.EQ.4) GO TO 20
      CALL SIGMA2(MIN,Z,S2)
      DO 21 I=1,N-1
      DO 21 J=I+1,N
      IF(Z(I,I))25,21,25
   25 IF(Z(J,J))26,21,26
   26 Z(I,J)=Z(I,J)/DSQRT(Z(I,I)*Z(J,J))
   21 CONTINUE
      WRITE(8,11)
      DO 22 I=1,N
      P(I)=MIN(I)
   22 WRITE(8,16)(Z(I,J),J=1,N)
   20 CONTINUE
      WRITE(8,15)
      SQ=0.
      CQ=0.
      SQC=0.
```

```
      CQC=0.
      I4=NNS+3
      C(I4)=0.
      DO 1 I3=IN,IM
      AL=NA+I3
      I4=AL-NAS
      IF(KL.EQ.0)HA=FFFF(MIN,AL)
      IF(KL.EQ.3) HA=HAD3(MIN,AL)
      IF(KL.EQ.6) HA=GAM3(MIN,AL)
      IF(I4)32,32,33
   33 IF(NNS-I4)32,36,36
   32 I4=NNS+3
   36 IF(C(I4))27,27,28
   27 VK1=0.
      SVAR=0.
      WXT=0.
      GO TO 29
   28 SVAR=C(I4)/B(I4)
      VK1=DSQRT(SVAR/(C(I4)*C(I4)))
      WXT=B(I4)/C(I4)
   29 IF(HA)30,30,31
   30 VK2=0.
      GO TO 37
   31 CONTINUE
      VK2=DSQRT(S2(I3)/(HA*HA))
   37 DSQ=C(I4)-HA
      SQ=SQ+DSQ*DSQ
      CQ=CQ+DSQ*DSQ*WXT
      IF((AL.GT.30).OR.(AL.LT.20)) GO TO 1
      SQC=SQC+DSQ*DSQ
      CQC=CQC+DSQ*DSQ*WXT
    1 WRITE(8,14)AL,C(I4),HA,SVAR,S2(I3),VK1,VK2
      PSQ=(100*SQC)/SQ
      PCQ=(100*CQC)/CQ
      WRITE(8,17)SQ,CQ,SQC,CQC,PSQ,PCQ
      IN=IS1
      IM=IS2
      NA=IS3
      RETURN
   11 FORMAT(/1H1,69H COVARIANCES,VARIANCES AND CORRELATIONS OF FINAL PA
```

```
     1RAMETER ESTIMATES )
 12 FORMAT(1H ,32HERROR IN THE PARAMETERS ICOUNT=,I6)
 13 FORMAT(/1H ,53H NUMBER OF RESTARTS AND FUNCTION EVALUATIONS.ICOUNT
     1= ,I8//1X,
     232H VALUE OF MINIMIZATION FUNCTION ,D18.7//1X,
     327H FINAL PARAMETER ESTIMATES ,10F15.7)
 14 FORMAT(1H ,F3.0,1X,2F15.6,6X,2D15.6,2X,2F15.6)
 15 FORMAT(//1H ,3HAGE,6X,3(10H  OBSERVED,5X,10H  COMPUTED,9X)/1X,10X,
     12(10H  VALUE   ,5X),3X,2(10H  VARIANCE,5X),4X,2(15H  KOEFF OF VAR.
     1))
 16 FORMAT(1H ,10F15.7)
 17 FORMAT(/1H ,43H SUM OF SQUARED DEVIATIONS BETWEEN OBSERVED /1X,
     153H AND COMPUTED VALUES AS THEY ARE WRITTEN OUT HERE IS ,D18.7/1X,
     224H THE CHISQUARE VALUE IS ,D18.7/1X,
     A26H SUM OF SQUARES AGE 20-30 ,D18.7/1X,
     B27H CHISQUARE VALUE AGE 20-30 ,D18.7/1X,
     347H SUM OF SQUARES AGE 20-30 IN PER CENT OF TOTAL ,F5.2/1X,
     442H CHISQUARE AGE 20-30 IN PER CENT OF TOTAL ,F5.2/1X)
      END
```

```
      FUNCTION FN(P)
      IMPLICIT REAL*8(A-H,P-Z)
      REAL*4 A(190),B(190)
      REAL*8 P(1),FN
      COMMON/DAT/A,NN,KL,K3
      COMMON/DAT1/N,I2
      COMMON/DAT2/B
      COMMON/DAT5/S(150),WX(150)
      COMMON/DAT6/NA,IN,IM
      FN=0.D0
      DO 1 I=1,N
      IF((P(I).LT.1.D-5).AND.(P(I).GT.-1.D-5)) P(I)=0.D0
    1 CONTINUE
      DO 2 I=IN,IM
      AL=I+NA-IN+1
      IF(KL.EQ.0)HA=FFFF(P,AL)
      IF(KL.EQ.3)HA=HAD3(P,AL)
      IF(KL.EQ.6)HA=GAM3(P,AL)
      W=A(I)-HA
      IF(FN-1.D38)2,2,3
    2 FN=FN+W*W*WX(I)
      RETURN
    3 WRITE(8,10)
      RETURN
   10 FORMAT(1H0,25HVALUE OF FN EXCEEDS 1.D38)
      END
```

```
      FUNCTION HAD3(P,A)
      IMPLICIT REAL*8(A-H,P-Z),LOGICAL*1(O)
      REAL*8 P(1),A,HAD3
      IF(P(1))9,9,1
    1 IF(P(2))9,9,2
    2 DM=P(4)-P(3)
      S=P(2)
      W=DSQRT(2.*S)
      IF(DM)3,9,3
    3 Q=(2.*DM*DM+6.*S)/(8.*W*DM)
      IF(Q)9,9,7
    7 QSQ=Q*Q-.75
      IF(QSQ)9,9,4
    4 CONTINUE
      H=Q+DSQRT(QSQ)
      RWH=P(1)*DSQRT(W*H)*H
      AD=A-P(4)
      ADH=AD+W*H
      IF(ADH)9,9,5
    5 DIV=ADH*DSQRT(ADH)*1.772453850905516
      EXP=(-H*AD*AD)/(W*AD+2.*S*H)
      IF(EXP-174.)6,9,9
    6 HAD3=(RWH*DEXP(EXP))/DIV
      RETURN
    9 HAD3=0.D0
      RETURN
      END
```

```
      SUBROUTINE DHAD3(P,A,HA,DH)
      IMPLICIT REAL*8(A-H,P-Z),LOGICAL*1(O)
      REAL*8 P(1),DH(1),A,HA
      DH(1)=HA/P(1)
      DM=P(4)-P(3)
      S=P(2)
      W=DSQRT(2.*S)
      Q=(2.*DM*DM+6.*S)/(8.*W*DM)
      QS=DSQRT(Q*Q-.75)
      H=Q+QS
      WH=W*H
      AD=A-P(4)
      V=AD+WH
      VW=V*W
      SV=S*V
      VWH=V+WH
      DV1=(VW*V-6.*H*SV+2.*H*AD*AD*VWH)*8.*W*DM*QS
      DV2=AD*(3.*VW-2.*H*AD*AD)*(6.*S-2.*DM*DM)
      DH(2)=(HA*(DV1+DV2))/(64.*SV*SV*DM*QS)
      DV1=HA*(3.*VW-2.*H*AD*AD)*AD*(6.*S-2.*DM*DM)
      DH(3)= DV1/(32.*SV*V*DM*DM*QS)
      DV1=(2.*H*VWH*AD+3.*VW)*8.*W*DM*DM*QS
      DV2=(3.*VW-2.*H*AD*AD)*AD*(2.*DM*DM-6.*S)
      DH(4)=(HA*(DV1+DV2))/(32.*SV*V*DM*DM*QS)
      RETURN
      END
```

```
      SUBROUTINE STEST1(N,P)
      IMPLICIT REAL*8(A-H,P-Z),LOGICAL*1(O)
      REAL*4 A(190)
      REAL*8 P(1)
      COMMON/DAT/A,NN,KL,K3
      COMMON/DAT6/NA,IN,IM
      R=0.
      D=0.
      DO 1 I=IN,IM
      AL=I+NA-IN+1
      R=R+A(I)
    1 D=D+AL*A(I)
      IF(R.LT.1.D-5) GO TO 9
      D=D/R
      P(1)=R
      H=A(IN)
      M=IN
      DO 4 I=IN+1,IM
      IF(H-A(I))3,4,4
    3 H=A(I)
      M=I
    4 CONTINUE
      M=M+NA-IN+1
      V=D+.5
      IV=V
      IF((IV-NA+IN-1).LT.1)IV=M
      HH=A(IV-NA+IN-1)
      IF(HH.LT..1D-5)HH=A(M-NA+IN-1)
      IF(D-M)20,20,21
   20 M=D-1.
      WRITE(8,12)
   21 CONTINUE
      DIV=4.*(D-M)*HH*HH*3.141592653589793
      P(3)=(3.*R*R)/DIV
      P(4)=P(3)-D
      P(2)=(HH*P(3)*1.772453850905516)/R
      DM0=-P(4)+(3.*P(3)*(DSQRT(1.+(16.*(P(2)**4.))/9.D0)-1.))/(4.*P(2)
     1*P(2))
      P(3)=DM0
      S2=(R*R)/(HH*HH*2.*3.141592653589793)
```

```
      P(2)=S2
      P(4)=D
      WRITE(8,10)(P(I),I=1,N)
      RETURN
    9 WRITE(8,11)
      STOP
   10 FORMAT(/1H ,32H START ESTIMATES(YNTEMA.R,S,M,Y),4F12.6/)
   11 FORMAT(//1H ,30HESTIMATION IMPOSSIBLE R<=1.D-5)
   12 FORMAT(1H ,32H SUB.STEST1..D-M<=0..NOW M=D-1. )
      END
```

```
      FUNCTION GAM3(P,A)
      IMPLICIT REAL*8(A-H,P-Z),LOGICAL*1(O)
      REAL*8 P(1),A,GAM3
      YM=P(2)-P(4)
      IF(YM)10,10,1
   10 P(4)=P(2)-1.
      YM=1.
      WRITE(8,11)
    1 E=P(3)/(YM*YM)
      TL=DLOG(YM)*DABS(E)
      IF(DABS(TL)-87.)4,4,12
   12 WRITE(8,13)
      D1=1.D38
      GO TO 6
    4 D1=(P(1)*(YM**(-E)))
    6 CONTINUE
      D2=(A-P(2)+(P(3)/YM))
      IF(D2)9,9,3
    3 TL=DLOG(D2)*DABS(E-1.)
      IF(TL-88.)5,5,9
    5 D3=D2**(E-1.)
      EX=-D2/YM
      IF(DABS(EX)-88.)2,2,9
    2 GAM3=(D1*D3*DEXP(EX))/DGAMMA(E)
      RETURN
    9 GAM3=0.
      RETURN
   11 FORMAT(1H ,42H SUB.GAM3..P(2)-P(4)<=0..NOW P(4)=P(2)-1. )
   13 FORMAT(1H ,51H SUB.GAM3..P(2)-P(4) TOO SMALL.EXPRESSION=10**38    )
      END
```

```
      SUBROUTINE DGAM3(P,A,HA,DH)
      IMPLICIT REAL*8(A-H,P-Z),LOGICAL*1(O)
      REAL*8 P(1),DH(1),A,HA
      DH(1)=HA/P(1)
      Y=P(2)
      S=P(3)
      YM=Y-P(4)
      YM2=YM*YM
      B=S/YM2
      GDG=DGM(B)
      XY=A-Y
      XM=A-P(4)
      XMYM=XM*YM
      XYYM=XY*YM
      SXYYM=S+XYYM
      SXMYM=S*XMYM
      IF(SXYYM)1,1,2
    1 FLNF=0.
      WRITE(8,10)
      GO TO 3
    2 CONTINUE
      FLNF=DLOG(SXYYM/YM2)
    3 CONTINUE
      DH(2)=(HA*((XMYM*(S+YM2))/SXYYM+XYYM-2.*S*(FLNF-GDG)))/(YM*YM2)
      DH(3)=(HA*(FLNF-GDG-XMYM/SXYYM))/YM2
      DH(4)=(HA*(2.*S*(FLNF-GDG)-SXMYM/SXYYM-XYYM))/(YM*YM2)
      RETURN
   10 FORMAT(1H ,58H SUB.DGAM3..S2+(AGE-D)*(D-M)<=0 IN FUNCT DLOG..RETUR
     1NS 0. )
      END
```

```
      SUBROUTINE STEST4(N,P)
      IMPLICIT REAL*8(A-H,P-Z),LOGICAL(O)
      REAL*4 A(190)
      REAL*8 P(1)
      COMMON/DAT/A,NN,KL,K3
      COMMON/DAT6/NA,IN,IM
      DIMENSION R(4)
      DO 1 I=1,4
    1 R(I)=0.
      DO 2 I=IN,IM
      AL=I+NA-IN+1
      R(1)=R(1)+A(I)
      R(2)=R(2)+AL*A(I)
      DO 2 J=3,4
    2 R(J)=R(J)+AL**(J-1)*A(I)
      IF(R(1).LT.1.D-5) GO TO 9
      YM=R(2)/R(1)
      S2=R(3)/R(1)-YM*YM
      R3S=R(4)/R(1)
      DYM=(2.*S2*S2)/(R3S-YM*YM*YM-3.*YM*S2)
      P(1)=R(1)
      P(2)=YM
      P(3)=S2
      P(4)=YM-S2/DYM
      WRITE(8,10)(P(I),I=1,N)
      RETURN
    9 WRITE(8,11)
      STOP
   10 FORMAT(/1H ,32H START ESTIMATES(MOMENT.R,Y,S,M),4F12.6)
   11 FORMAT(//1H ,30HESTIMATION IMPOSSIBLE R<=1.D-5)
      END
```

```
      SUBROUTINE TINP(KC,K2)
      IMPLICIT REAL*8(A-H,P-Z),LOGICAL*1(O)
      REAL*4 A(190),B(190)
      COMMON/DAT/A,NN,KL,K3
      COMMON/DAT2/B
      KC=KC+1
      IF(K2.EQ.3) GO TO 1
      READ(7,10,END=3)IS
    1 READ(14,11,END=3)A(NN+1),A(NN+2),(A(I),I=1,NN),(B(I),I=1,NN)
      IF(K2.EQ.3) RETURN
      IQ=A(NN+1)
      IF(IS.EQ.IQ)RETURN
      GO TO 1
    3 WRITE(8,13)KC
      STOP
   10 FORMAT(I4)
   11 FORMAT(F4.0,A4,2(150F7.0))
   13 FORMAT(/1H ,20H EOF ON DATAFILE 14 /1X,
     128H NUMBER OF RECORDS READ ARE ,F5.0)
      END
```

```
      SUBROUTINE EJNUL(P,EJ)
      IMPLICIT REAL*8(A-H,P-Z),LOGICAL*1(O)
      REAL*4 A(190)
      REAL*8 P(1),EJ(150,10)
      COMMON/DAT/A,NN,KL,K3
      COMMON/DAT1/N,I2
      COMMON/DAT6/NA,IN,IM
      DIMENSION DH(10)
      DO 1 I=IN,IM
      AL=I+NA-IN+1
      IF(KL)6,7,6
    7 HA=FFFF(P,AL)
      CALL DFFFF(P,AL,HA,DH)
    6 IF(KL-3)3,2,3
    2 HA=HAD3(P,AL)
      CALL DHAD3(P,AL,HA,DH)
    3 IF(KL-6)5,4,5
    4 HA=GAM3(P,AL)
      CALL DGAM3(P,AL,HA,DH)
    5 DO 1 J=1,N
    1 EJ(I,J)=DH(J)
      RETURN
      END
```

```
      SUBROUTINE SIGMA2(P,Z,S2)
      IMPLICIT REAL*8(A-H,P-Z),LOGICAL*1(O)
      REAL*4 A(190),B(190)
      REAL*8 P(1),S2(1),Z(10,10)
      COMMON/DAT/A,NN,KL,K3
      COMMON/DAT1/N,I2
      COMMON/DAT2/B
      COMMON/DAT5/S(150),WX(150)
      COMMON/DAT6/NA,IN,IM
      COMMON/DAT8/IKN,IKM,IS1,IS2,IS3
      DIMENSION EM(150,10),EJ(150,10),EJM(10,150),LV(10),MV(10),AX(100)
      CALL EJNUL(P,EJ)
C
C     JOT*MO
C
      DO 2 I=1,N
      DO 2 J=IN,IM
    2 EJM(I,J)=EJ(J,I)*WX(J)
C
C     JOT*MO*JO
C
      DO 3 I=1,N
      DO 3 J=1,N
      EM(I,J)=0.
      DO 3 K=IN,IM
    3 EM(I,J)=EM(I,J)+EJM(I,K)*EJ(K,J)
C
C     (JOT*MO*JO)-1
C
      K=0
      DO 31 J=1,N
      DO 31 I=1,N
      K=K+1
      Z(I,J)=0.
   31 AX(K)=EM(I,J)
      CALL MINV(AX,N,DET,LV,MV)
      IF(DET)33,6,33
   33 K=0
      DO 32 J=1,N
      DO 32 I=1,N
```

```
      K=K+1
   32 Z(I,J)=AX(K)
C
C     AT=((JOT*MO*JO)-1*JOT*MO)T
C
      DO 4 I=1,N
      DO 4 J=IN,IM
      EM(J,I)=0.
      DO 4 K=1,N
    4 EM(J,I)=EM(J,I)+Z(I,K)*EJM(K,J)
C
C     Z=A*ZD*AT
C
      DO 5 I=1,N
      DO 5 J=1,N
      Z(I,J)=0.
      DO 5 K=IN,IM
    5 Z(I,J)=Z(I,J)+S(K)*EM(K,I)*EM(K,J)
C
C     JO*Z
C
      IS1=IN
      IS2=IM
      IS3=NA
      IN=1
      IM=IKM-IKN+1
      NA=IKN-1
      CALL EJNUL(P,EJ)
      DO 7 I=IN,IM
      DO 7 J=1,N
      EJM(J,I)=0.
      DO 7 K=1,N
    7 EJM(J,I)=EJM(J,I)+EJ(I,K)*Z(K,J)
C
C     JO*Z*JOT
C
      DO 8 I=IN,IM
      S2(I)=0.
      DO 8 K=1,N
    8 S2(I)=S2(I)+EJM(K,I)*EJ(I,K)
```

```
C
      RETURN
    6 WRITE(8,10)
      RETURN
   10 FORMAT(1H ,56H SINGULAR MATRIX  CALCULATION OF VARIANCES DISCONTIN
     1UED )
      END
```

```
      FUNCTION DGM(P)
      IMPLICIT REAL*8(A-H,P-Z),LOGICAL*1(O)
      REAL*4 PSITAB(51)
      REAL*8 P,DGM
      COMMON/DAT3/Q
      DATA PSITAB/-.57722,-.54480,-.51327,-.48263,-.45280,-.42375,-.3954
     16,-.36787,-.34095,-.31469,-.28904,-.26398,-.23949,-.21555,-.19212,
     2-.16919,-.14674,-.12475,-.10321,-.08209,-.06138,-.04107,-.02114,-.
     300158,.01763,.03649,.05502,.07323,.09114,.10874,.12605,.14308,
     4.15983,.17633,.19256,.20855,.22429,.23980,.25509,.27015,.28499,
     5.29963,.31406,.32829,.34233,.35618,.36985,.38334,.39666,.40980,
     6.42278/
      IG=0
      N=P
      Q=P-N
      IF(P)9,9,8
    8 IF(P-1.)6,6,5
    5 CONTINUE
      IF(Q.LT..1D-4) GO TO 1
      IF(Q.GT..99999) GO TO 4
      KQ=50*Q+1
      DIFF=PSITAB(KQ+1)-PSITAB(KQ)
      DQ=Q-(KQ-1)*.02
      DGM=PSITAB(KQ)+DIFF*DQ*50.
      N=N-1
      Q=Q+1.
      IF(IG)7,3,7
    3 CONTINUE
      IF(N.EQ.0)RETURN
      Q=Q-1.
      DO 2 I=1,N
      BN=I+Q
    2 DGM=DGM+1./BN
      RETURN
    1 Q=1.
      N=N-1
      DGM=-0.577215664901533
      GO TO 3
    4 Q=1.
      DGM=-0.577215664901533
```

```
      GO TO 3
    6 IG=1
      GO TO 5
    7 Q=Q-1.
      DGM=DGM-1./Q
      RETURN
    9 WRITE(8,10)
      RETURN
   10 FORMAT(1H ,46H FUNCTION DGM CANNOT HANDLE VALUES <= 0.          )
      END
```

TABLE NO 1
PARAMETER CARD READ FROM SYSIN CONTAINS 3 36 14 0 1 2 4 0 7 16 44

AFTER ANALYSIS OF PARAMETER CARD THE FOLLOWING PARAMETERS ARE SUPPOSED TO BE USED

TYPE OF INPUT 3
NUMBER OF OBSERVATION POINTS 36
AGE-1 WHERE GRADUATION STARTS 14
ESTIMATION OF BOUNDARIES OF THE GRADUATION 0
TYPE OF PARAMETER ESTIMATES 1
TYPE OF FUNCTION USED 2
METHOD OF MINIMIZATION IS NELDER-MEAD
NUMBER OF FUNCTION PARAMETERS 4
VALUES FOR CONSTRUCTION OF SIMPLEX 0
NUMBER OF CASES WITH EQUAL PARAMETERS 7
PRINTOUT STARTS AT AGE 16
PRINTOUT STOPS AT AGE 44

OBSERVATIONS ARE FROM 1968

AGE	RATE	BASE
15	0.000667	29973.0
16	0.003298	29410.5
17	0.019512	28342.0
18	0.050812	29225.5
19	0.087830	29375.0
20	0.128356	30150.5
21	0.157226	31337.0
22	0.175834	32644.5
23	0.191731	29447.5
24	0.195725	28141.5
25	0.192218	25211.0
26	0.188863	23345.0
27	0.172577	20698.0
28	0.161848	21304.0
29	0.148437	21019.0
30	0.130609	20458.0
31	0.114562	19622.5
32	0.103915	19015.5
33	0.094327	18605.5
34	0.079484	18695.5
35	0.068799	18619.5
36	0.061196	20279.0
37	0.052837	20516.0
38	0.042596	21082.0
39	0.037367	20927.5
40	0.029771	21732.5
41	0.022185	21726.0
42	0.015969	23044.0
43	0.010880	23069.0
44	0.006947	24184.0
45	0.004008	25450.0
46	0.001781	25824.5
47	0.000948	26361.0
48	0.000459	28343.5
49	0.000085	23634.5
50	0.000040	25263.0

RATE IS SET TO ZERO IF BASE*RATE<=5 THIS IS DONE FOR AGES

49
50
LIMITS OF THE GRADUATION ARE TO BE FROM 15 TO 48

START ESTIMATES(MOMENT.R,Y,S,M)   2.753574   27.295380   33.845156   25.556532

NUMBER OF RESTARTS AND FUNCTION EVALUATIONS.ICOUNT=   184

VALUE OF MINIMIZATION FUNCTION   0.1242105D 04

FINAL PARAMETER ESTIMATES   2.6992238   27.0628912   32.6628208   24.5524850

COVARIANCES, VARIANCES AND CORRELATIONS OF FINAL PARAMETER ESTIMATES

0.0001135	0.1571860	0.0321205	0.1501661
0.0000378	0.0005097	0.5285961	0.6182245
0.0000729	0.0025410	0.0453385	-0.2347319
0.0000497	0.0004332	-0.0015514	0.0009635

AGE	OBSERVED VALUE	COMPUTED VALUE	OBSERVED VARIANCE	COMPUTED VARIANCE	OBSERVED KOEFF OF VAR.	COMPUTED KOEFF OF VAR.
16.	0.003298	0.005400	0.112137D-06	0.600108D-07	0.101537	0.045362
17.	0.019512	0.020514	0.688448D-06	0.157938D-06	0.042524	0.019373
18.	0.050812	0.046733	0.173862D-05	0.314549D-06	0.025950	0.012001
19.	0.087830	0.080679	0.298996D-05	0.557251D-06	0.019687	0.009253
20.	0.128356	0.116986	0.425718D-05	0.786824D-06	0.016075	0.007582
21.	0.157226	0.150462	0.501726D-05	0.901070D-06	0.014247	0.006309
22.	0.175834	0.177295	0.538633D-05	0.909314D-06	0.013199	0.005379
23.	0.191731	0.195424	0.651094D-05	0.886633D-06	0.013309	0.004818
24.	0.195725	0.204375	0.695503D-05	0.885356D-06	0.013474	0.004604
25.	0.192218	0.204847	0.762437D-05	0.904758D-06	0.014365	0.004643
26.	0.188863	0.198250	0.809008D-05	0.914553D-06	0.015060	0.004824
27.	0.172577	0.186306	0.833786D-05	0.888049D-06	0.016732	0.005058
28.	0.161848	0.170754	0.759707D-05	0.818252D-06	0.017030	0.005298
29.	0.148437	0.153163	0.706204D-05	0.715817D-06	0.017903	0.005524
30.	0.130609	0.134831	0.638425D-05	0.599014D-06	0.019346	0.005740
31.	0.114562	0.116753	0.583830D-05	0.484637D-06	0.021091	0.005963
32.	0.103915	0.099635	0.546475D-05	0.383480D-06	0.022496	0.006215
33.	0.094327	0.083928	0.506984D-05	0.299937D-06	0.023870	0.006525
34.	0.079484	0.069878	0.425150D-05	0.233788D-06	0.025941	0.006919
35.	0.068799	0.057570	0.369500D-05	0.182441D-06	0.027940	0.007419
36.	0.061196	0.046980	0.301770D-05	0.142662D-06	0.028387	0.008040
37.	0.052837	0.038006	0.257540D-05	0.111556D-06	0.030373	0.008788
38.	0.042596	0.030503	0.202049D-05	0.869139D-07	0.033370	0.009665
39.	0.037367	0.024303	0.178555D-05	0.672016D-07	0.035760	0.010667
40.	0.029771	0.019233	0.136988D-05	0.513898D-07	0.039314	0.011787
41.	0.022185	0.015127	0.102113D-05	0.387684D-07	0.045549	0.013017
42.	0.015969	0.011829	0.692979D-06	0.288056D-07	0.052129	0.014348
43.	0.010880	0.009200	0.471629D-06	0.210621D-07	0.063121	0.015774
44.	0.006947	0.007120	0.287256D-06	0.151508D-07	0.077150	0.017287

SUM OF SQUARED DEVIATIONS BETWEEN OBSERVED
AND COMPUTED VALUES AS THEY ARE WRITTEN OUT HERE IS 0.2163866D-02
THE CHISQUARE VALUE IS 0.7534567D 03
SUM OF SQUARES AGE 20-30 0.8211884D-03
CHISQUARE VALUE AGE 20-30 0.1235447D 03
SUM OF SQUARES AGE 20-30 IN PER CENT OF TOTAL 37.95
CHISQUARE AGE 20-30 IN PER CENT OF TOTAL 16.40

OBSERVATIONS ARE FROM 1972

AGE	RATE	BASE
15	0.000699	30058.5
16	0.004798	30637.5
17	0.023804	30078.5
18	0.057574	29892.0
19	0.092316	29984.0
20	0.124422	29400.0
21	0.140950	28208.5
22	0.161626	29011.5
23	0.171963	29204.0
24	0.174367	29954.0
25	0.173612	31098.0
26	0.166425	32417.0
27	0.153412	29235.0
28	0.143130	28002.5
29	0.125224	25107.0
30	0.110037	23319.5
31	0.096600	20662.5
32	0.083954	21357.0
33	0.071517	21044.0
34	0.059688	20490.0
35	0.052582	19645.5
36	0.043049	19025.0
37	0.039344	18630.5
38	0.030581	18672.0
39	0.023871	18600.0
40	0.019395	20211.5
41	0.014611	20533.0
42	0.009837	21043.0
43	0.006038	20868.5
44	0.003880	21648.5
45	0.002171	21650.0
46	0.001350	22961.5
47	0.000305	22952.0
48	0.000166	24025.0
49	0.000119	25222.0
50	0.	25601.0

RATE IS SET TO ZERO IF BASE*RATE<=5 THIS IS DONE FOR AGES

48
49
50

LIMITS OF THE GRADUATION ARE TO BE FROM 15 TO 47

START ESTIMATES(MOMENT.R,Y,S,M)   2.383132   26.678369   31.394413   24.931354

NUMBER OF RESTARTS AND FUNCTION EVALUATIONS.ICOUNT=   234

VALUE OF MINIMIZATION FUNCTION   0.7434540D 03

FINAL PARAMETER ESTIMATES   2.3420655   26.5373318   31.1067173   24.0743473

COVARIANCES,VARIANCES AND CORRELATIONS OF FINAL PARAMETER ESTIMATES

0.0000883	0.1413634	0.1129694	0.0610804
0.0000319	0.0005760	0.5745894	0.5947987
0.0002393	0.0031085	0.0508136	-0.2186986
0.0000181	0.0004497	-0.0015529	0.0009923

AGE	OBSERVED VALUE	COMPUTED VALUE	OBSERVED VARIANCE	COMPUTED VARIANCE	OBSERVED KOEFF OF VAR.	COMPUTED KOEFF OF VAR.
16.	0.004798	0.007116	0.156605D-06	0.703190D-07	0.082479	0.037263
17.	0.023804	0.023781	0.791396D-06	0.170696D-06	0.037372	0.017373
18.	0.057574	0.050371	0.192607D-05	0.350899D-06	0.024105	0.011760
19.	0.092316	0.082751	0.307884D-05	0.596784D-06	0.019007	0.009335
20.	0.124422	0.115574	0.423204D-05	0.782557D-06	0.016534	0.007654
21.	0.140950	0.144208	0.499672D-05	0.837634D-06	0.015859	0.006347
22.	0.161626	0.165622	0.557110D-05	0.804491D-06	0.014604	0.005416
23.	0.171963	0.178510	0.588834D-05	0.758193D-06	0.014111	0.004878
24.	0.174367	0.182974	0.582116D-05	0.734206D-06	0.013837	0.004683
25.	0.173612	0.180065	0.558274D-05	0.723112D-06	0.013610	0.004723
26.	0.166425	0.171333	0.513388D-05	0.700573D-06	0.013615	0.004885
27.	0.153412	0.158471	0.524755D-05	0.652110D-06	0.014932	0.005096
28.	0.143130	0.143077	0.511133D-05	0.579259D-06	0.015796	0.005319
29.	0.125224	0.126514	0.498761D-05	0.493419D-06	0.017834	0.005552
30.	0.110037	0.109856	0.471867D-05	0.407367D-06	0.019741	0.005810
31.	0.096600	0.093879	0.467514D-05	0.330026D-06	0.022383	0.006119
32.	0.083954	0.079099	0.393098D-05	0.265241D-06	0.023616	0.006511
33.	0.071517	0.065809	0.339845D-05	0.212999D-06	0.025777	0.007013
34.	0.059688	0.054134	0.291303D-05	0.171322D-06	0.028595	0.007646
35.	0.052582	0.044078	0.267654D-05	0.137787D-06	0.031114	0.008421
36.	0.043049	0.035557	0.226276D-05	0.110349D-06	0.034943	0.009342
37.	0.039344	0.028442	0.211181D-05	0.875851D-07	0.036936	0.010405
38.	0.030581	0.022575	0.163780D-05	0.686102D-07	0.041848	0.011603
39.	0.023871	0.017791	0.128339D-05	0.528859D-07	0.047458	0.012926
40.	0.019395	0.013929	0.959602D-06	0.400396D-07	0.050507	0.014365
41.	0.014611	0.010840	0.711586D-06	0.297484D-07	0.057734	0.015911
42.	0.009837	0.008388	0.467471D-06	0.216868D-07	0.069505	0.017556
43.	0.006038	0.006457	0.289336D-06	0.155177D-07	0.089086	0.019291
44.	0.003880	0.004947	0.179227D-06	0.109051D-07	0.109111	0.021111

SUM OF SQUARED DEVIATIONS BETWEEN OBSERVED
AND COMPUTED VALUES AS THEY ARE WRITTEN OUT HERE IS 0.9538533D-03
THE CHISQUARE VALUE IS 0.4181914D 03
SUM OF SQUARES AGE 20-30 0.3148408D-03
CHISQUARE VALUE AGE 20-30 0.6086460D 02
SUM OF SQUARES AGE 20-30 IN PER CENT OF TOTAL 33.01
CHISQUARE AGE 20-30 IN PER CENT OF TOTAL 14.55

APPENDIKS B

BETA-TETTHETA SOM GRUNNLAG FOR EIN GLATTINGSFUNKSJON

B.0. Ved sida av Hadwigertettheta og gammatettheta har beta-tettheta vore brukt som utgangspunkt for å lage glattingsfunksjonar. I funksjonane vi har nytta framanfor har vi funne at fire parametrar er tilstrekkeleg. Vi skal her gi eit eksempel på ein glattingsfunksjon med fem parametrar basert på beta-tettheta.

B.1. La X ha tettheta

$$b_0(x) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} x^{\alpha-1}(1-x)^{\beta-1} \text{ for } 0 < x < 1,$$

$\alpha > 1$ og $\beta = 1$. Da er

$$EX = \frac{\alpha}{\alpha+\beta},$$

og

$$\text{var } X = \frac{\alpha\beta}{(\alpha+\beta)^2\,(\alpha+\beta+1)},$$

medan modalverdien i fordelinga er

$$\text{mod } X = \frac{\alpha-1}{\alpha+\beta-2}.$$

B.2. Vi innfører no transformasjonen

$$Y = A + (B-A)\,X.$$

Y vil da ha ei tetthet $b_1(\cdot)$ over intervallet (A, B), der

$$b_1(y) = \frac{\Gamma(\alpha+\beta)\,(y-A)^{\alpha-1}\,(B-y)^{\beta-1}}{\Gamma(\alpha)\,\Gamma(\beta)\,(B-A)^{\alpha+\beta-1}}\text{ ; med } \alpha > 1 \text{ og } \beta > 1. \tag{1}$$

For Y finn vi

$$EY = \frac{B\alpha+A\beta}{\alpha+\beta},$$

$$\text{var } Y = \frac{(B-A)^2\,\alpha\beta}{(\alpha+\beta)^2\,(\alpha+\beta+1)}$$

og

$$\text{mod } Y = \frac{B(\alpha-1)+A(\beta-1)}{\alpha+\beta-2}.$$

B.3. Vi set no $EY = \eta$, mod $Y = \Theta$, var $Y = \tau^2$. Det gir

(2) $$A = \frac{\tau^2\,(3\Theta-\eta-B) + \eta\,(B-\eta)\,(\eta-\Theta)}{\tau^2 + (B-\eta)\,(\eta-\Theta)},$$

(3) $$\beta = \frac{(B+\eta-2\Theta)\,(B-\eta) - \tau^2}{(\eta-\Theta)\,(B-\eta)^2 + (2B+\eta-3\Theta)\,\tau^2}\,(B-\eta)$$

og

(4) $$\alpha = \frac{\left[(B+\eta-2\Theta)\,(B-\eta)-\tau^2\right](B+2\eta-3\Theta)\,\tau^2}{\left[(\eta-\Theta)\,(B-\eta)^2+(2B+\eta-3\Theta)\tau^2\right]\left[(\eta-\Theta)\,(B-\eta)+\tau^2\right]}$$

$$= \frac{\beta}{B-\eta}\cdot\frac{B+2\eta-3\Theta}{(\eta-\Theta)\,(B-\eta)+\tau^2}\,\tau^2.$$

B.4. Vi fører inn (2), (3) og (4) i (1) og multipliserer samtidig med R. Vi får da ein funksjon med fem parametrar: R, Θ, η, τ^2 og B. Funksjonenblir

(5) $$b_2(x) = \frac{R\Gamma(\alpha+\beta)\,(x-A)^{\alpha-1}\,(B-x)^{\beta-1}}{\Gamma(\alpha)\,\Gamma(\beta)\,(B-A)^{\alpha+\beta-1}}$$

for $A < x < B$, $\alpha > 1$, $\beta > 1$ og A, α og β gitt ved (2), (3) og (4).

Denne funksjonen programmerer vi som FUNCTION FFFF(P,X), der P er parametervektoren (R,Θ,η,τ^2,B).

B.5. Anta at vi har observert ratene $\{\hat{\lambda}_x,\ x = a_0,\ a_0+1,\ \ldots,\ b_0\}$ og at persongrunnlaget for ratene er $\{L_x,\ x = a_0,\ a_0+1,\ \ldots,\ b_0\}$. L_x er altså nemnaren i den brøken som definerer $\hat{\lambda}_x$. Vi finn da startestimatorar for parametrane ved å sette

$$\hat{R}_n = \sum_x x^n \hat{\lambda}_x \text{ for } n \geq 0$$

og

$$\hat{R}_n^{*} = \hat{R}_n/\hat{R} \quad \text{for } n > 0.$$

Dette gir

$$\hat{R}_B = \hat{R}_0,$$

$$\hat{\eta}_B = \hat{R}_1^{*}$$

og

$$\hat{\tau}_B^2 = \hat{R}_2^{*} - (\hat{R}_1^{*})^2.$$

Sett no

$$\hat{B}_B = \max\left\{x:\hat{\lambda}_x \times L_x > 5 \;\&\; \hat{\lambda}_y \times L_y \leq 5 \text{ for } y > x\right\} + 1$$

og

$$\hat{A}_B = \min\left\{x:\hat{\lambda}_x \times L_x > 5 \;\&\; \hat{\lambda}_y \times L_y \leq 5 \text{ for } y < x\right\} - 1.$$

Av (2) finn vi

$$\hat{\Theta}_B = \frac{\hat{\eta}_B\,(\hat{B}_B - \hat{\eta}_B)\,(\hat{\eta}_B - \hat{A}_B) - (\hat{A}_B + \hat{B}_B + \hat{\eta}_B)\,\hat{\tau}_B^2}{(\hat{\eta}_B - \hat{A}_B)\,(\hat{B}_B - \hat{\eta}_B) - 3\hat{\tau}_B^2}$$

Desse uttrykka programmerer vi som SUBROUTINE FSTEST(N,P), der N er antall parametrar og P er parametervektoren $(R, \Theta, \eta, \tau^2, B)$.

B.6. Vi skal no finne dei deriverte av $b_2(x)$ med omsyn på parametrane R, Θ, η, τ^2 og B. Det blir for R

$$(6) \qquad \frac{\partial b_2(x)}{\partial R} = b_2(x)/R.$$

For dei andre parametrane nyttar vi relasjonane

$$(7) \qquad \frac{\partial b_2(x)}{\partial \Theta} = \frac{\partial b_2(x)}{\partial \alpha}\frac{\partial \alpha}{\partial \Theta} + \frac{\partial b_2(x)}{\partial \beta}\frac{\partial \beta}{\partial \Theta} + \frac{\partial b_2(x)}{\partial A}\frac{\partial A}{\partial \Theta} + \frac{\partial b_2(x)}{\partial B}\frac{\partial B}{\partial \Theta},$$

$$(8) \qquad \frac{\partial b_2(x)}{\partial \eta} = \frac{\partial b_2(x)}{\partial \alpha}\frac{\partial \alpha}{\partial \eta} + \frac{\partial b_2(x)}{\partial \beta}\frac{\partial \beta}{\partial \eta} + \frac{\partial b_2(x)}{\partial A}\frac{\partial A}{\partial \eta} + \frac{\partial b_2(x)}{\partial B}\frac{\partial B}{\partial \eta},$$

$$(9) \qquad \frac{\partial b_2(x)}{\partial \tau^2} = \frac{\partial b_2(x)}{\partial \alpha}\frac{\partial \alpha}{\partial \tau^2} + \frac{\partial b_2(x)}{\partial \beta}\frac{\partial \beta}{\partial \tau^2} + \frac{\partial b_2(x)}{\partial A}\frac{\partial A}{\partial \tau^2} + \frac{\partial b_2(x)}{\partial B}\frac{\partial B}{\partial \tau^2}$$

og

$$(10) \qquad \frac{\partial b_2(x)}{\partial B} = \frac{\partial b_2(x)}{\partial \alpha}\frac{\partial \beta}{\partial B} + \frac{\partial b_2(x)}{\partial \beta}\frac{\partial \beta}{\partial B} + \frac{\partial b_2(x)}{\partial A}\frac{\partial A}{\partial B} + \frac{\partial b_2(x)}{\partial B}.$$

Vi vil her ha $\frac{\partial B}{\partial \Theta} = \frac{\partial B}{\partial \eta} = \frac{\partial B}{\partial \tau^2} = 0.$

Vi set no $\frac{d}{dx} \ln \Gamma(x) = \Psi(x).$

Vi finn da for størrelsane brukt i (7), (8), (9) og (10) følgande uttrykk:

$$(11) \qquad \frac{\partial b_2(x)}{\partial \alpha} = b_2(x)\left[\Psi(\alpha+\beta) - \Psi(\alpha) + \ln\frac{x-A}{B-A}\right],$$

$$(12) \qquad \frac{\partial b_2(x)}{\partial \beta} = b_2(x)\left[\Psi(\alpha+\beta) - \Psi(\beta) + \ln\frac{B-x}{B-A}\right],$$

$$(13) \qquad \frac{\partial b_2(x)}{\partial A} = b_2(x)\left[\frac{(\alpha+\beta-1)x - \beta A - (\alpha-1)B}{(B-A)\ (x-A)}\right],$$

$$(14) \qquad \frac{\partial b_2(x)}{\partial B} = b_2(x)\left[\frac{(\alpha+\beta-1)\ x-\alpha B - (\beta-1)A}{(B-A)\ (B-x)}\right],$$

$$(15) \qquad \frac{\partial A}{\partial \Theta} = \frac{3\tau^2 - \eta(B-\eta) + A(B-\eta)}{\tau^2 + (\eta-\Theta)\ (B-\eta)},$$

$$(16) \qquad \frac{\partial A}{\partial \eta} = \frac{2\eta(B+\Theta) - 3\eta^2 - \Theta B - A(B+\Theta-2\eta) - \tau^2}{\tau^2 + (\eta-\Theta)\ (B-\eta)},$$

$$(17) \qquad \frac{\partial A}{\partial \tau^2} = \frac{3\Theta - B - A - \eta}{\tau^2 + (\eta-\Theta)\ (B-\eta)},$$

$$(18) \qquad \frac{\partial A}{\partial B} = \frac{(\eta-\Theta)\ (\eta-A) - \tau^2}{\tau^2 + (\eta-\Theta)\ (B-\eta)},$$

$$(19) \qquad \frac{\partial \beta}{\partial \Theta} = \frac{(\beta-2)\ (B-\eta)^2 + 3\tau^2\beta}{(\eta-\Theta)\ (B-\eta)^2 + (2B+\eta-3\Theta)\tau^2},$$

$$(20) \qquad \frac{\partial \beta}{\partial \eta} = \frac{\tau^2 - (B-\eta)\ (B+3\eta-4\Theta) - \beta\left[\tau^2+(B-\eta)\ (B-3\eta+2\Theta)\right]}{(\eta-\Theta)\ (B-\eta)^2 + (2B+\eta-3\Theta)\tau^2},$$

$$(21) \qquad \frac{\partial \beta}{\partial \tau^2} = \frac{\eta - B - \beta(2B+\eta-3\Theta)}{(\eta-\Theta)\ (B-\eta)^2 + (2B+\eta-3\Theta)\tau^2},$$

$$(22) \qquad \frac{\partial \beta}{\partial B} = \frac{(B-\eta)\ (rB+\eta-4\Theta) - \tau^2 - 2\beta\left[\tau^2 + (B-\eta)\ (\eta-\Theta)\right]}{(\eta-\Theta)\ (B-\eta)^2 + (2B+\eta-3\Theta)\tau^2},$$

$$(23) \qquad \frac{\partial \alpha}{\partial \Theta} = \frac{\partial \beta}{\partial \Theta}\left(\frac{\alpha}{\beta}\right) + \frac{(B-\eta)^2\alpha - 3\beta\tau^2}{(B-\eta)\ \left[(\eta-\Theta)\ (B-\eta) + \tau^2\right]},$$

$$(24) \qquad \frac{\partial \alpha}{\partial \eta} = \frac{\partial \beta}{\partial \eta}\left(\frac{\alpha}{\beta}\right) + \frac{(2\beta+\alpha)\tau^2 - \alpha(B-\eta)\ (B-3\eta+2\Theta)}{(B-\eta)\ \left[(\eta-\Theta)\ (B-\eta) + \tau^2\right]},$$

$$(25) \qquad \frac{\partial \alpha}{\partial \tau^2} = \frac{\partial \beta}{\partial \tau^2}\left(\frac{\alpha}{\beta}\right) + \frac{(\eta-\Theta)\ (B-\eta)\ \alpha}{\left[(\eta-\Theta)\ (B-\eta) + \tau^2\right]\ \tau^2}$$

og

$$(26) \qquad \frac{\partial \alpha}{\partial B} = \frac{\partial \beta}{\partial B}\left(\frac{\alpha}{\beta}\right) + \frac{(\beta-\alpha)\tau^2 - 2\alpha(\eta-\Theta)\ (B-\eta)}{(B-\eta)\ \left[(\eta-\Theta)\ (B-\eta) + \tau^2\right]}.$$

Vi programmerer desse uttrykka som SUBROUTINE DFFFF(P,X,HA,DH), der vi har at P er parametervektoren (R,Θ,η,τ^2,B), X er alderen og HA = FFFF(P,X). Resultatet

$$\left(\frac{\partial b_2(x)}{\partial R}, \frac{\partial b_2(x)}{\partial \Theta}, \frac{\partial b_2(x)}{\partial \eta}, \frac{\partial b_2(x)}{\partial \tau^2}, \frac{\partial b_2(x)}{\partial B}\right)$$

plasserer vi i DH.

Dersom vi ønskjer å nytte berre fire parametrar, set vi B fast t.d. lik 50 og må da sette $\frac{\partial b_2(x)}{\partial B} = 0$.

APPENDIKS C

OVERSETTING AV SYMBOL FRA HOEM & BERGE (1974) TIL SYMBOL NYTTA HER

I Hoem og Berge er det nytta ein del andre symbol enn i dette notatet. Vi måtte i programbeskrivelsen her ta mest omsyn til dei symbol som vart nytta under programmeringa. Til hjelp for dei som vil lese begge notata er det laga følgande oversettingsliste:

Symbol i Hoem & Berge (1974):		Symbol i Berge (1974):
Hadwigerfunksjonen:		Hadwigerfunksjonen:
R	=	R,
H	=	H,
T	=	T,
D	=	-d,
U	=	D,
M	=	M,
S^2	=	S.
Gammafunksjonen:		Gammafunksjonen:
R	=	R,
b	=	k,
c	=	1/c,
d	=	-d,
μ	=	Y,
m	=	M
σ^2	=	S.

APPENDIKS D

EKSEMPEL PÅ BRUK AV RUTINENE FFFF, DFFFF OG FSTEST.

A. Innlesing av startverdiar fra kort.

Dersom ein ønskjer å glatte ufullstendige data (t.d. kohortrater), vil det vere behov for å nytte andre startestimat enn dei den interne rutina gjev.

Ein kan da lese dei inn fra kort. Vi går da fram slik:

(i) Sett programparameteren KL lik 0.

(ii) Punch opp følgande kort og legg dei bakerst i programdekket. Hugs å fullføre FORMAT 10.

```
      FUNCTION FFFF(P, A)
      IMPLICIT REAL*8(A-H, P-Z)
      REAL*8 P(1), A, FFFF
      FFFF = HAD3(P, A)
      RETURN
      END

      SUBROUTINE DFFFF(P, A, HA, DH)
      IMPLICIT REAL*8(A-H, P-Z)
      REAL*8 P(1), DH(1), A, HA
      CALL DHAD3(P, A, HA, DH)
      RETURN
      END

      SUBROUTINE FSTEST(N, P)
      IMPLICIT REAL*8(A-H, P-Z)
      REAL*8 P(1)
      READ (7, 10) (P (I), I = 1, N)
      WRITE (8, 11) (P (I), I = 1, N)
      RETURN
10    FORMAT (.......)
11    FORMAT (/1H ,32H START ESTIMATES (.......R, S, M, Y), 4F12.6/)
      END
```

(iii) Kort med startestimat legg vi bakerst i kortoppsettet som høyrer til eit parameterkort (sjå figur side 15) berre framfor eventuelle seleksjonskort. Det er viktig at vi puncher parametrane i rekkefølga R, S, M og Y når vi nyttar Hadwigerfunksjonen.

B. Utviding av modellen til å omfatte andre uavhengige variablar enn alder.

For å gi eit eksempel på bruken av programsystemet som eit generelt regresjonsprogram går vi utifra at vi vil utvide modellen fra å være $y = h(x, \underset{\sim}{\theta})$ der h er Hadwigerfunksjonen, x er alder og $\underset{\sim}{\theta}$ parametervektoren (R, S, M, Y) til å omfatte eit lineært element: $y = h(x, \underset{\sim}{\theta}) + A+BZ$ der A og B er nye parametrar vi vil estimere og Z er ein variabel vi les inn fra tape (vi vil nytte logisk eining nr. 15). Vi går da fram slik:

(i) Sett programparametrane KL lik 0 og N = 6.

(ii) Punch opp korta REAL*4 AB(190) og COMMON/DAT10/AB og legg dei inn saman med dei andre COMMON korta i BLOCK DATA subprogrammet.

(iii) Punch opp følgande kort og legg dei bakerst i programdekket. Hugs å endre FORMAT 10 i FSTEST.

```
FUNCTION FFFF (P, A)
IMPLICIT REAL*8(A-H, P-Z)
REAL*4 AB(190)
REAL*8 P(1), A, FFFF
COMMOM/DAT6/ NA, IN, IM
COMMON/DAT9/ NAS, NNS, IKK, C(190)
COMMON/DAT10/ AB
JA = A - NAS + IN-1
FFFF = HAD3(P, A) + P(6)*AB(JA) + P(5)
RETURN
END

SUBROUTINE DFFFF(P, A, HA, DH)
IMPLICIT REAL*8(A-H, P-Z)
REAL*4 AB(190)
REAL*8 P(1), DH(1), A, HA
COMMON/DAT6/ NA, IN, IM
COMMON/DAT9/ NAS, NNS, IKK, C(190)
COMMON/DAT10/ AB
JA = A - NAS + IN-1
CALL DHAD3(P, A, HA, DH)
DH(5) = ÷1.
DH(6) = AB(JA)
RETURN
END
```

```
      SUBROUTINE FSTEST (N, P)
      IMPLICIT REAL*8(A-H, P-Z)
      REAL*4 AB(190), A(190)
      REAL*8 P(1)
      COMMON/DAT/A, NN, KL, K3
      COMMON/DAT6/NA, IN, IM
      COMMON/DAT9/NAS, NNS, IKK, C(190)
      COMMON/DAT10/AB
      READ(15, 10) (AB(I), I = 1, NNS)
      DO 1 I = 1, NNS
 1    AB(I) = AB(I)*1. E-5
      WRITE (8, 11) (((I+NAS), AB(I)), I = 1, NNS)
      NI = 4
      CALL STEST1(NI, P)
      P(5) = -1.
      P(6) = 1.
      WRITE (8, 12) (P(I), I = 5, N)
      RETURN
10    FORMAT (154X, F6.0, 35(//////54X, F6.0))
11    FORMAT (1H ,32H VALUES OF EXPLANATORY
      VARIABLE/150(/9X, I3, 3X, F15.6))
12    FORMAT (1H ,49H START ESTIMATES OF ADDITIONAL PARAMETERS (A,B, .) ,
      6F10.4)
      END
```