



Program SURVIV User's Manual Version 1.2 -- March 1986

Gary C. White¹

Environmental Science Group, MS-K495

Los Alamos National Laboratory

Los Alamos, NM 87545

(505)667-2914

1. Present Address: Department of Fishery and Wildlife Biology, Colorado State University, Fort Collins, CO 80523 (303)491-6678.

[Recent modifications\(10/99\)](#) by

Jim Hines

Patuxent Wildlife Research Center

Laurel, Md. 20708

Program SURVIV (a stand alone program) is used to estimate survival rates from user specified cell probability functions. The program reads the user specified algebraic expressions and constructs a FORTRAN 77 subroutine that calculates the log-likelihood function. Numerical optimization is then used to estimate the unknown parameters. Currently the program is available for CDC-NOS, VAX-VMS, UNIX, PC-DOS, and MS-DOS operating systems. The NOS version is written in single precision, while the VMS, UNIX, and DOS versions are written in double precision.

The details of the syntax for each of the procedures in Program SURVIV are shown in Table 1. As the reader works through the examples illustrating these procedures, he should occasionally refer back to Table 1 to understand the information presented. Table 1 is a summary of each procedure and does not provide the detailed explanations provided in the examples.

Data Analysis

Three general tasks are required to estimate survival rates with the above procedure. First, the algebraic expressions must be read and the FORTRAN 77 subroutine constructed -- PROC MODEL performs this task. Second, the likelihood function must be numerically maximized by varying the

parameter estimates for one or more sets of constraints on the parameters -- PROC ESTIMATE performs this task. Finally comparisons of parameter estimates by hypothesis tests must be made to identify the model appropriate for the observed data, performed by PROC TEST.

Table 1. Summary of command syntax for each of the procedures and their modifiers in Program SURVIV. Modifiers enclosed in brackets indicate optional input. Program SURVIV allows comments enclosed between /* and */ symbols, and statements are delineated with semicolons.

PROC TITLE Any title information to be printed at top of each page;

PROC MODEL NPAR = # of S(I)'s in model;

INLINE [DECLARE] FORTRAN declaration;

COHORT = # of animals in cohort{FORTRAN STATEMENTS};
 # observed: algebraic expression for cell probability;
 .
 .
 .
 repeated for up to maximum number of cells per cohort;

COHORT = repeated for up to maximum number of cohorts;

LABELS;

S(1) = identifying label;

.
 .
 .

S(NPAR) = identifying label;

PROC ESTIMATE [NSIG = # of significant digits to find for
 estimates (default is 5)]
 [MAXFN = maximum number of function evaluations
 (default is 1000)]
 [NAME = name of submodel]
 [TRANSFORM = NONE or ABS (default is SIN
 transform)]
 [NOVAR (Do not print variance-covariance matrix,
 default is to print matrix)];

INITIAL;

S(1) = value;

.
 .
 .

S(NPAR) = value;

or ALL = value;

or RETAIN = modname /* Obtain initial estimates from
 output of previous model named in "modname" */

CONSTRAINTS /*Constraints of the form */;

S(1) = S(2) /* S(1) is the same as S(2) */ ;

S(1) < S(2) /* S(1) is less than S(2) */ ;

S(1) > S(2) /* S(1) is greater than S(2) */;

S(1) = value /* Fix parameter to constant */;

PROC TEST /* No modifiers */;

PROC BROWNIE [NSIG = # of significant digits to find for
 estimates (default is 5)]
 [MAXFN = maximum number of function evaluations
 (default is 1000)]
 [TRANSFORM = NONE or ABS (default is SIN
 transform)]
 [NOVAR (Do not print variance-covariance matrix,
 default is to print this matrix)];

```

\* Input to BROWNIE is a free format version of the
input described by Brownie et al. (1978:155-158)
for programs ESTIMATE and BROWNIE, with the
additional requirement that each card image
must have a semicolon as the last non-blank
character. */;

\* Currently BROWNIE does not handle data of types
2 or 3 */;

PROC BMDPAR /* No modifiers */;

    INITIAL /* As in PROC ESTIMATE */;

    CONSTRAINTS /* As in PROC ESTIMATE */;

PROC IDENTIFIABLE [ADD = Constant to add for numerical derivative
                  (Default is 0.05)]
[MULTIPLY = Constant to multiply by for numerical derivative
            (Default is 1.05)];

    INITIAL /* As in PROC ESTIMATE */;

    CONSTRAINTS /* As in PROC ESTIMATE */;

PROC SIMULATE [NSIM = # of replications of each simulation
              (default is 5)]
              [DETAIL (print details of each simulation,
                    with default of only simulation summaries)]
              [SEED = random number seed (default is 7654321)];
              [PARFILE = file name for simulated data output for
                later analysis (default is no data output)]
              [ERROR TEST (do not include simulations in data
                summaries with warnings about convergence --
                default to include questionable convergence
                simulations)];

    INITIAL /* As in PROC ESTIMATE */;

    CONSTRAINTS /* As in PROC ESTIMATE */;

PROC SAMPLE SIZE /* Notation follows Brownie et al. 1978:190-193
                 */;
S-BAR = values of S /* Define average adult survival rates.
                 */;
S'-BAR = values of S' /* Define average young survival
                 rates. */;
F-BAR = values of f /* Define average adult recovery rate.
                 */;
CV(S) = values of CV of S /* Define coefficients of
                 variation for values of S */;
CV(S') = values of CV of S' /* Define coefficient of
                 variation for values of S' */;
K = number of years /* Define number of years to
                 conduct banding effort. */;

PROC STOP /* Stops execution. */;

```

I will illustrate these procedures first with a simple example taken from Brownie et al. (1978), page 21. Ducks were banded for 3 years, and band recoveries summarized for 5 years. The input to run this model and the resulting output are shown in Fig. 1.

All input instructions read by Program SURVIV are echoed in the output with the identifying INPUT --- to delineate them. Thus on page 1 of the output in Fig. 1, PROC TITLE is called with a title for the run. Statements are separated by semicolons. Next PROC MODEL is called, with the number of parameters to be estimated set to 7. Additional comments are specified to clarify the input. PROC MODEL processes the observed data and the associated cell probability statements. Thus the next line of input is COHORT = 1603 with a comment to identify this cohort as the number of ducks banded in 1964. The COHORT card specifies the number of animals that are subject to mortality in the mortality categories that follow it. Therefore the next input statement specifies that 127 duck bands were returned, and the expected cell probability is S(3). That is, the observed multinomial cell value is 127, and the expected cell value is $1603 \cdot S(3)$, but the number of animals at risk of mortality is known

from the COHORT statement, so only the expected cell probability is given. The colon (:) separates the observed number of recoveries and the expected cell probability.

The next statement continues to describe the fate of the 1603 ducks banded in 1964. Forty-four bands were recovered during the year 1965, with the expected cell probability being $S(1)*S(4)$. The cell probability statements must be FORTRAN 77 readable, as these algebraic statements are used to construct the likelihood subroutine. The specification of observed recoveries and expected cell probabilities continues for the 1964 cohort until 1968. At this time, $17 + 44 + 37 + 40 + 17 = 265$ bands have been returned. Program SURVIV knows that the sum of the observed cells should add up to the number of animals specified on the proceeding cohort card. Because this is not the case, an additional cell is constructed with observed value equal to $1603 - 265 = 1338$ ducks never seen again, and with expected cell probability equal to 1.0 minus the sum of the previous 5 cells. The sum of all the cell probabilities must be 1.0. Thus if a duck was never seen again (i.e., the band is not reported), then it falls into the last cell.

```

Program SURVIV - Survival Rate Estimation with User Specified Cell Probability Functions 12-JAN-84 19:43:42 Page
01
Version 1.1(VMS) Aug.,
1983
-----
INPUT --- PROC TITLE Banding Handbook Example from Page 21 of Brownie et al.(1978);
      CPU time in seconds for last procedure was      0.05
INPUT --- PROC MODEL NPAR=7 /* Banding Handbook Page 21 */ ;
INPUT --- COHORT = 1603 /* BANDED 1964*/ ;
INPUT --- 127:S(3) /* BANDS RECOVERED IN 1964 */ ;
INPUT --- 44:S(1)*S(4) /* BANDS RECOVERED IN 1965 */ ;
INPUT --- 37:S(1)*S(2)*S(5) /* BANDS RECOVERED IN 1966 */ ;
INPUT --- 40:S(1)*S(2)*S(6) /* BANDS RECOVERED IN 1967 */ ;
INPUT --- 17:S(1)*S(2)*S(7) /* BANDS RECOVERED IN 1968 */ ;
INPUT --- COHORT = 1595 /* BANDED 1965*/ ;
INPUT --- 62:S(4) /* Bands recovered in 1965 */ ;
INPUT --- 76:S(2)*S(5) /* Bands recovered in 1966 */ ;
INPUT --- 44:S(2)*S(6) /* Bands recovered in 1967 */ ;
INPUT --- 28:S(2)*S(7) /* Bands recovered in 1968 */ ;
INPUT --- COHORT = 1157 /* BANDED 1966*/ ;
INPUT --- 82:S(5) /* BANDS RECOVERED IN 1966 */ ;
INPUT --- 61:S(6) /* BANDS RECOVERED IN 1967 */ ;
INPUT --- 24:S(7) /* BANDS RECOVERED IN 1968 */ ;
INPUT --- LABELS;
INPUT --- S(1)=Survival for 1964;
INPUT --- S(2)=Survival for 1965;
INPUT --- S(3)=RECOVERY FOR 1964;
INPUT --- S(4)=RECOVERY FOR 1965;
INPUT --- S(5)=RECOVERY FOR 1966;
INPUT --- S(6)=Survival for 1966 and recovery 1967;
INPUT --- S(7)=Survival for 1966-67 and recovery 1968;
      CPU time in seconds for last procedure was      1.48

```

Fig. 1. Output from Program SURVIV for a banding analysis model from Brownie et al. (1978:21).

```

Program SURVIV - Survival Rate Estimation with User Specified Cell Probability Functions 12-JAN-84 19:43:42 Page
02
Banding Handbook Example from page 21 of Brownie et al. (1978)
1983
-----
Version 1.1 (VMS) Aug.,
INPUT --- PROC ESTIMATE NSIG=5 MAXFN=750 NAME=SV&RC_SAME /* COMBINATION OF SURVIVAL AND RECOVERY RATES
INPUT --- CONSTANT BY TIME*/;
INPUT --- INITIAL;
INPUT --- ALL=0.05;
INPUT --- CONSTRAINTS;
INPUT --- S(1)=S(2);
INPUT --- S(3)=S(4);
INPUT --- S(3)=S(5);
      Number of parameters in model = 7
      Number of parameters set equal = 3

```

Number of parameters fixed = 0

Number of parameters estimated = 4

I	Parameter	S(I)	Lower Bound	Upper Bound	Label
1	1	0.050	0.000	1.000	Survival for 1964
2	1	0.050	0.000	1.000	Survival for 1965
3	2	0.050	0.000	1.000	RECOVERY FOR 1964
4	2	0.050	0.000	1.000	RECOVERY FOR 1965
5	2	0.050	0.000	1.000	RECOVERY FOR 1966
6	3	0.050	0.000	1.000	Survival for 1966 and recovery 1967
7	4	0.050	0.000	1.000	Survival for 1966-67 and recovery 1968

Final function value 2648.6431 (Error Return = 0)

Number of significant digits 7

Number of function evaluations 133

I	Parameter	S(I)	Standard Error	95% Confidence Interval	
				Lower	Upper
1	1	0.64058338	0.34957855E-01	0.57206599	0.70910078
2	1	0.64058338	0.34957855E-01	0.57206599	0.70910078
3	2	0.60617526E-01	0.32176157E-02	0.54310999E-01	0.66924053E-01
4	2	0.60617526E-01	0.32176157E-02	0.54310999E-01	0.66924053E-01
5	2	0.60617526E-01	0.32176157E-02	0.54310999E-01	0.66924053E-01
6	3	0.51111879E-01	0.47218499E-02	0.41857053E-01	0.60366704E-01
7	4	0.24322205E-01	0.30891896E-02	0.18267393E-01	0.30377016E-01

Program SURVIV - Survival Rate Estimation with User Specified Cell Probability Functions 12-JAN-84 19:43:42 Page 03
 Banding Handbook Example from Page 21 of Brownie et al.(1978) Version 1.1(VMS) Aug., 1983

Variance-Covariance matrix of estimates on diagonal and below,
 Correlation matrix of estimates above diagonal.

	1	2	3	4	5	6	7
1	0.12220516E-02	1.0000000	-0.50834998	-0.50834998	-0.50834998	-0.47184568	-0.34320111
2	0.12220516E-02	0.12220516E-02	-0.50834998	-0.50834998	-0.50834998	-0.47184568	-0.34320111
3	-0.57179685E-04	-0.57179685E-04	0.10353051E-04	1.0000000	1.0000000	0.19792254	0.14396070
4	-0.57179685E-04	-0.57179685E-04	0.10353051E-04	0.10353051E-04	1.0000000	0.19792254	0.14396070
5	-0.57179685E-04	-0.57179685E-04	0.10353051E-04	0.10353051E-04	0.10353051E-04	0.19792254	0.14396070
6	-0.77885558E-04	-0.77885558E-04	0.30070566E-05	0.30070566E-05	0.30070566E-05	0.22295866E-04	0.13951651
7	-0.37062783E-04	-0.37062783E-04	0.14309442E-05	0.14309442E-05	0.14309442E-05	0.20350840E-05	0.95430924E-05

Cohort	Cell	Observed	Expected	Chi-square	Note
1	1	127	97.170	9.158	0 < P < 1
1	2	44	62.245	5.348	0 < P < 1
1	3	37	39.873	0.207	0 < P < 1
1	4	40	33.621	1.210	0 < P < 1
1	5	17	15.999	0.063	0 < P < 1
1	6	1338	1354.092	0.191	0 < P < 1
2	1	62	96.685	12.443	0 < P < 1
2	2	76	61.935	3.194	0 < P < 1
2	3	44	52.223	1.295	0 < P < 1
2	4	28	24.851	0.399	0 < P < 1
2	5	1385	1359.307	0.486	0 < P < 1
3	1	82	70.134	2.007	0 < P < 1
3	2	61	59.136	0.059	0 < P < 1
3	3	24	28.141	0.609	0 < P < 1
3	4	990	999.588	0.092	0 < P < 1

Total (Degrees of freedom = 8) 36.761
 Pr(Larger Chi-square) = 0.0000
 With pooling, Degrees of freedom = 8 Chi-square = 36.761
 Pr(Larger Chi-square) = 0.0000

Log-likelihood = -52.874994

CPU time in seconds for last procedure was 1.86

Program SURVIV - Survival Rate Estimation with User Specified Cell Probability Functions 12-JAN-84 19:43:42 Page 04
 Banding Handbook Example from Page 21 of Brownie et al.(1978) Version 1.1(VMS) Aug., 1983

```
INPUT --- PROC ESTIMATE NSIG=5 MAXFN=750 NAME=GENERAL /*General model*/;
INPUT --- INITIAL;
INPUT --- RETAIN=SV&RC_SAME ;
```

Number of parameters in model = 7

Number of parameters set equal = 0

Number of parameters fixed = 0

Number of parameters estimated = 7

I	Parameter	S(I)	Lower Bound	Upper Bound	Label
1	1	0.641	0.000	1.000	Survival for 1964
2	2	0.641	0.000	1.000	Survival for 1965
3	3	0.061	0.000	1.000	RECOVERY FOR 1964
4	4	0.061	0.000	1.000	RECOVERY FOR 1965
5	5	0.061	0.000	1.000	RECOVERY FOR 1966
6	6	0.051	0.000	1.000	Survival for 1966 and recovery 1967
7	7	0.024	0.000	1.000	Survival for 1966-67 and recovery 1968

Final function value 2632.5922 (Error Return = 0)

Number of significant digits 7

Number of function evaluations 190

I	Parameter	S(I)	Standard Error	95% Confidence Interval	
				Lower	Upper
1	1	0.65386329	0.67817609E-01	0.52094078	0.78678580
2	2	0.63432469	0.65052567E-01	0.50682166	0.76182772
3	3	0.79226450E-01	0.67459754E-02	0.66004339E-01	0.92448562E-01
4	4	0.40103773E-01	0.41474639E-02	0.31974743E-01	0.48232802E-01
5	5	0.68816791E-01	0.60804365E-02	0.56899135E-01	0.80734446E-01
6	6	0.51171460E-01	0.50073002E-02	0.41357151E-01	0.60985768E-01
7	7	0.24350557E-01	0.31908973E-02	0.18096398E-01	0.30604715E-01

Variance-Covariance matrix of estimates on diagonal and below,
Correlation matrix of estimates above diagonal.

	1	2	3	4	5	6	7
1							
2							
3							
4							
5							
6							
7							

Program SURVIV - Survival Rate Estimation with User Specified Cell Probability Functions 12-JAN-84 19:43:42 Page 05
Banding Handbook Example from Page 21 of Brownie et al.(1978) Version 1.1(VMS) Aug., 1983

1	0.45992280E-02-0.38874199	-0.70637676E-01-0.38549334	0.16916143E-13	0.72043515E-13	0.86797202E-13	
	-0.17150167E-02	0.42318365E-02-0.25947279E-13	0.11893140	-0.56544739	-0.51057168	-0.38126647
3	-0.32316449E-04-0.11386782E-16	0.45508185E-04	0.53900554E-13-0.13003798E-13	0.16998136E-13	0.21718938E-13	
4	-0.10842813E-03	0.32088069E-04	0.15080669E-17	0.17201457E-04-0.36206355E-13-0.68325729E-13	0.84330358E-13	
5	0.69755521E-17-0.22366158E-03-0.53339598E-18-0.91306601E-18	0.36971708E-04	0.30982260	0.23135825		
6	0.24464762E-16-0.16631246E-03	0.57418214E-18-0.14189612E-17	0.94330357E-05	0.25073055E-04	0.20890532	
7	0.18782830E-16-0.79141792E-04	0.46751567E-18-0.11160391E-17	0.44888239E-05	0.33378434E-05	0.10181826E-04	

Cohort	Cell	Observed	Expected	Chi-square	Note
1	1	127	127.000	0.000	0 < P < 1
1	2	44	42.034	0.092	0 < P < 1
1	3	37	45.754	1.675	0 < P < 1
1	4	40	34.022	1.050	0 < P < 1
1	5	17	16.190	0.041	0 < P < 1
1	6	1338	1338.000	0.000	0 < P < 1
2	1	62	63.966	0.060	0 < P < 1
2	2	76	69.625	0.584	0 < P < 1
2	3	44	51.773	1.167	0 < P < 1
2	4	28	24.637	0.459	0 < P < 1
2	5	1385	1385.000	0.000	0 < P < 1
3	1	82	79.621	0.071	0 < P < 1
3	2	61	59.205	0.054	0 < P < 1
3	3	24	28.174	0.618	0 < P < 1
3	4	990	990.000	0.000	0 < P < 1

Total (Degrees of freedom = 5) 5.872
Pr(Larger Chi-square) = 0.3189
With pooling, Degrees of freedom = 5 Chi-square = 5.872
Pr(Larger Chi-square) = 0.3189

Log-likelihood = -36.824115

CPU time in seconds for last procedure was 2.40

Program SURVIV - Survival Rate Estimation with User Specified Cell Probability Functions 12-JAN-84 19:43:42 Page 06
Banding Handbook Example from Page 21 of Brownie et al.(1978) Version 1.1(VMS) Aug., 1983

INPUT --- PROC ESTIMATE NSIG=5 MAXFN=750 NAME=SURVL_SAME /* 1964 SURVIVAL EQUALS 1965 SURVIVAL*/;
 INPUT --- INITIAL;
 INPUT --- RETAIN=GENERAL ;
 INPUT --- CONSTRAINTS;
 INPUT --- S(2)=S(1);

Number of parameters in model = 7
 Number of parameters set equal = 1
 Number of parameters fixed = 0
 Number of parameters estimated = 6

I	Parameter	S(I)	Lower Bound	Upper Bound	Label
1	1	0.654	0.000	1.000	Survival for 1964
2	1	0.654	0.000	1.000	Survival for 1965
3	2	0.079	0.000	1.000	RECOVERY FOR 1964
4	3	0.040	0.000	1.000	RECOVERY FOR 1965
5	4	0.069	0.000	1.000	RECOVERY FOR 1966
6	5	0.051	0.000	1.000	Survival for 1966 and recovery 1967
7	6	0.024	0.000	1.000	Survival for 1966-67 and recovery 1968

Final function value 2632.6078 (Error Return = 0)

Number of significant digits 10

Number of function evaluations 168

I	Parameter	S(I)	Standard Error	95% Confidence Interval	
				Lower	Upper
1	1	0.64393595	0.36818149E-01	0.57177238	0.71609952
2	1	0.64393595	0.36818149E-01	0.57177238	0.71609952
3	2	0.79277320E-01	0.67417990E-02	0.66063394E-01	0.92491246E-01
4	3	0.40325278E-01	0.39726728E-02	0.32538839E-01	0.48111717E-01
5	4	0.68453790E-01	0.57011518E-02	0.57279533E-01	0.79628048E-01
6	5	0.50901536E-01	0.47490351E-02	0.41593428E-01	0.60209645E-01
7	6	0.24222110E-01	0.30925154E-02	0.18160780E-01	0.30283441E-01

Variance-Covariance matrix of estimates on diagonal and below,
 Correlation matrix of estimates above diagonal.

Program SURVIV - Survival Rate Estimation with User Specified Cell Probability Functions 12-JAN-84 19:43:42 Page 07
 Banding Handbook Example from Page 21 of Brownie et al.(1978) Version 1.1(VMS) Aug., 1983

	1	2	3	4	5	6	7
1	0.13555761E-02	1.0000000	-0.63570483E-01	-0.25020257	-0.54624638	-0.48761740	-0.35633115
	0.13555761E-02	0.13555761E-02	-0.63570483E-01	-0.25020257	-0.54624638	-0.48761740	-0.35633115
3	-0.15779501E-04	-0.15779501E-04	0.45451854E-04	-0.13455097E-01	0.15415179E-01	0.13760657E-01	0.10055734E-01
4	-0.36596243E-04	-0.36596243E-04	-0.36036734E-06	0.15782129E-04	0.11355109	0.10136358	0.74072419E-01
5	-0.11466031E-03	-0.11466031E-03	0.59249810E-06	0.25717972E-05	0.32503132E-04	0.23272711	0.17006759
6	-0.85260233E-04	-0.85260233E-04	0.44057551E-06	0.19123620E-05	0.63010794E-05	0.22553334E-04	0.15181413
7	-0.40572111E-04	-0.40572111E-04	0.20965317E-06	0.91002055E-06	0.29984447E-05	0.22296127E-05	0.95636516E-05

Cohort	Cell	Observed	Expected	Chi-square	Note
1	1	127	127.082	0.000	0 < P < 1
1	2	44	41.625	0.136	0 < P < 1
1	3	37	45.501	1.588	0 < P < 1
1	4	40	33.834	1.124	0 < P < 1
1	5	17	16.100	0.050	0 < P < 1
1	6	1338	1338.859	0.001	0 < P < 1
2	1	62	64.319	0.084	0 < P < 1
2	2	76	70.307	0.461	0 < P < 1
2	3	44	52.280	1.311	0 < P < 1
2	4	28	24.878	0.392	0 < P < 1
2	5	1385	1383.216	0.002	0 < P < 1
3	1	82	79.201	0.099	0 < P < 1
3	2	61	58.893	0.075	0 < P < 1
3	3	24	28.025	0.578	0 < P < 1
3	4	990	990.881	0.001	0 < P < 1

Total (Degrees of freedom = 6) 5.901
 Pr(Larger Chi-square) = 0.4343
 With pooling, Degrees of freedom = 6 Chi-square = 5.901
 Pr(Larger Chi-square) = 0.4343

Log-likelihood = -36.839708

CPU time in seconds for last procedure was 2.91

```

INPUT --- PROC ESTIMATE NSIG=5 MAXFN=750 NAME=REC_SAME /*CONSTANT RECOVERY RATES*/;
INPUT --- INITIAL;
INPUT --- RETAIN=GENERAL ;
INPUT --- CONSTRAINTS;
INPUT --- S(3)=S(4);
INPUT --- S(3)=S(5);
    
```

Number of parameters in model = 7
 Number of parameters set equal = 2
 Number of parameters fixed = 0
 Number of parameters estimated = 5

I	Parameter	S(I)	Lower Bound	Upper Bound	Label
1	1	0.654	0.000	1.000	Survival for 1964
2	2	0.634	0.000	1.000	Survival for 1965
3	3	0.079	0.000	1.000	RECOVERY FOR 1964
4	3	0.079	0.000	1.000	RECOVERY FOR 1965
5	3	0.079	0.000	1.000	RECOVERY FOR 1966
6	4	0.051	0.000	1.000	Survival for 1966 and recovery 1967
7	5	0.024	0.000	1.000	Survival for 1966-67 and recovery 1968

Final function value 2647.1385 (Error Return = 0)

Number of significant digits 7

Number of function evaluations 131

I	Parameter	S(I)	Standard Error	95% Confidence Interval	
				Lower	Upper
1	1	0.56107126	0.53357772E-01	0.45649002	0.66565249
2	2	0.72176072	0.63359611E-01	0.59757588	0.84594556
3	3	0.60639881E-01	0.32256549E-02	0.54317597E-01	0.66962164E-01
4	3	0.60639881E-01	0.32256549E-02	0.54317597E-01	0.66962164E-01
5	3	0.60639881E-01	0.32256549E-02	0.54317597E-01	0.66962164E-01
6	4	0.48934496E-01	0.47092640E-02	0.39704339E-01	0.58164654E-01
7	5	0.23286071E-01	0.30247206E-02	0.17357618E-01	0.29214523E-01

Variance-Covariance matrix of estimates on diagonal and below,
 Correlation matrix of estimates above diagonal.

	1	2	3	4	5	6	7
1	0.28470518E-02	0.26501651	-0.29049126	-0.29049126	-0.29049126	-0.57322427E-01	-0.42469136E-01
2	-0.89594864E-03	0.40144403E-02	0.32749735	-0.32749735	-0.32749735	-0.49397615	-0.36597788
3	-0.49997542E-04	0.66932676E-04	0.10404849E-04	1.0000000	1.0000000	0.19732926	0.14619764
4	-0.49997542E-04	0.66932676E-04	0.10404849E-04	0.10404849E-04	1.0000000	0.19732926	0.14619764
5	-0.49997542E-04	0.66932676E-04	0.10404849E-04	0.10404849E-04	0.10404849E-04	0.19732926	0.14619764
6	-0.14403741E-04	0.14739119E-03	0.29975223E-05	0.29975223E-05	0.29975223E-05	0.22177167E-04	0.18809952
7	-0.68541938E-05	0.70137875E-04	0.14264072E-05	0.14264072E-05	0.14264072E-05	0.26793286E-05	0.91489349E-05

Cohort	Cell	Observed	Expected	Chi-square	Note
1	1	127	97.206	9.132	0 < P < 1
1	2	44	54.539	2.037	0 < P < 1
1	3	37	39.364	0.142	0 < P < 1
1	4	40	31.766	2.134	0 < P < 1
1	5	17	15.116	0.235	0 < P < 1
1	6	1338	1365.009	0.534	0 < P < 1
2	1	62	96.721	12.464	0 < P < 1
2	2	76	69.809	0.549	0 < P < 1
2	3	44	56.334	2.700	0 < P < 1
2	4	28	26.807	0.053	0 < P < 1
2	5	1385	1345.329	1.170	0 < P < 1
3	1	82	70.160	1.998	0 < P < 1
3	2	61	56.617	0.339	0 < P < 1
3	3	24	26.942	0.321	0 < P < 1
3	4	990	1003.280	0.176	0 < P < 1

Total (Degrees of freedom = 7) 33.985
 Pr(Larger Chi-square) = 0.0000
 With pooling, Degrees of freedom = 7 Chi-square = 33.985
 Pr(Larger Chi-square) = 0.0000

Log-likelihood = -51.370480

CPU time in seconds for last procedure was 1.44

INPUT --- PROC TEST;

Submodel	Name	Log-likelihood	NDF	G-O-F
1	SV&RC_SAME	-52.874994	8	0.0000
2	GENERAL	-36.824115	5	0.3189
3	SURVL_SAME	-36.839708	6	0.4343
4	REC_SAME	-51.370480	7	0.0000

GENERAL VS. SV&RC_SAME Chi-square = 32.102 D.F. = 3 Pr(Larger Chi-square) = 0.0000

SURVL_SAME VS. SV&RC_SAME Chi-square = 32.071 D.F. = 2 Pr(Larger Chi-square) = 0.0000

REC_SAME VS. SV&RC_SAME Chi-square = 3.009 D.F. = 1 Pr(Larger Chi-square) = 0.0828

GENERAL VS. SURVL_SAME Chi-square = 0.031 D.F. = 1 Pr(Larger Chi-square) = 0.8598

GENERAL VS. REC_SAME Chi-square = 29.093 D.F. = 2 Pr(Larger Chi-square) = 0.0000

SURVL_SAME VS. REC_SAME Chi-square = 29.062 D.F. = 1 Pr(Larger Chi-square) = 0.0000

** WARNING ** Sequence of models reinitialized to zero.

CPU time in seconds for last procedure was 0.08

INPUT --- PROC STOP /* End of this run. */ ;

CPU time in minutes for this job was 0.17

E X E C U T I O N S U C C E S S F U L

PROC MODEL continues to process COHORT statements and their associated cell probability statements for the ducks banded in 1965 and 1966. Note that some of the parameters to be estimated occur in more than one cohort. This is reasonable because the survival rate of a duck banded in 1964 and one banded in 1965 should both be the same in 1966, assuming that the leg band is not changing the survival rate.

The LABELS statement is used to document the biological meaning of the parameters. Thus S(1) is shown to be the finite survival rate for the year 1964. S(3) is the recovery rate for 1964. These parameters correspond to S_1 and f_1 in Brownie et al. (1978:21). S(6) and S(7) are products of survival and recovery rates which are not identifiable. These "nuisance" parameters must be in the model to estimate S(1) through S(5), but are not of much value biologically.

When PROC MODEL reaches the end of its input (signified by the PROC ESTIMATE statement), it constructs the FORTRAN 77 subroutine and writes it to a disk file. Program SURVIV then exits, the FORTRAN 77 compiler is called to compile the subroutine, the loader is called to place the binary code into the core image, and Program SURVIV is started up again, only this time, the compile run option is not set on the execute line so that Program SURVIV assumes that the version of the likelihood subroutine in the core image corresponds to the input to PROC MODEL. Thus PROC MODEL does not stop execution when the PROC ESTIMATE statement is reached, but passes control to PROC ESTIMATE.

PROC ESTIMATE performs the numerical optimization of the likelihood function. Several parameters are specified on the statement. NSIG = 5 tells PROC ESTIMATE to estimate the parameters to 5 significant digits. MAXFN = 750 says that only 750 evaluations of the likelihood function are to be performed before an error is printed and optimization stopped. NAME=GENERAL gives this model a name (up to 10 characters) to identify it later in the PROC TEST output. All the parameters for PROC ESTIMATE have defaults, including NAME=SUBMODELii, where ii is the model number of this PROC ESTIMATE execution.

The INITIAL statement tells PROC ESTIMATE to initialize the parameter values in preparation for optimization. ALL = 0.05 sets all the values to 0.05, i.e., S(1) = 0.05, S(2) = 0.25, ..., S(7) = 0.05. S(1) = 0.5 and S(2) = 0.7 resets the values of S(1) and S(2).

Following the input to PROC ESTIMATE is the summary of the initial conditions for the upcoming optimization. The initial value of each parameter, the lower bound, the upper bound, and the label are summarized in a table. After the table, the results of the optimization procedure are printed. No errors are reported, the number of significant digits in the estimates are 6, and 211 function cells were required. The next table summarizes the parameter values, standard errors and 95%

confidence intervals. Then the variance-covariance and correlation matrix is printed for the parameter values. Finally, a table of the observed versus expected cell probabilities is printed, and the contribution of each cell to the overall chi-square goodness-of-fit statistic. If the model does not fit the data, this chi-square test will reject the null hypothesis. The NOTE column provides another method to check that the model was correctly optimized. The message $0 < P < 1$ indicates that the cell probability was in the interval $[0 - 1]$. Error messages are printed if the cell probability is not in the $[0 - 1]$ interval.

The next statement is another run of PROC ESTIMATE, this time to estimate the model parameters with equal survival in 1964 and 1965, and equal recovery rates in 1964, 1965, and 1966. These equalities are set with the CONSTRAINTS statement. $S(1) = S(2)$ sets survival rates for 1964 and 1965 equal. $S(3) = S(4)$ and $S(3) = S(5)$ sets 1964, 1965, and 1966 recovery rates equal.

These constraints are reflected in the summary of initial conditions, because only 4 parameters are to be estimated. The column labeled PARAMETER in the summary table on page 004 of the output (Fig. 1) indicates the relationship between $S(1)$ and the parameter index. Thus $S(1)$ and $S(2)$ are both parameter 1, and $S(3)$, $S(4)$, and $S(5)$ are all parameter 2.

In the summary of the estimation results, the values of $S(1)$ and $S(2)$ are identical, and likewise for $S(3)$, $S(4)$, and $S(5)$. Similarly, the variance-covariance matrix and the correlation matrix reflects that some of the parameters are constrained to be equal.

Two additional runs of PROC ESTIMATE are made in the example output. In the first, the 1964 and 1965 survival rates are not constrained, but the 1964, 1965, and 1966 recovery rates are all constrained equal. Thus a series of 4 models has been generated. The most general model allows every identifiable survival and recovery rate to be estimated. The other three models force various constraints on the estimates.

PROC TEST (page 010 of the output in Fig. 1) helps decide which of the four models is appropriate for the observed data. Each model is summarized in a table giving the log-likelihood function value, the degrees of freedom of the model, and the chi-square goodness-of-fit significance probability. In the example, only the GENERAL and SURVL_SAME models fit the observed data. Both the models with recoveries constrained equal for 1964, 1965, and 1966 reject the goodness-of-fit null hypothesis.

The next 6 lines provide the likelihood ratio test results of all pairs of models that do not have equal degrees of freedom. Thus the first test suggests that the GENERAL model with 7 parameters does not fit the data much better than constraining survival rates by year to be equal, because the significance level of the chi-square statistic is not less than 0.05.

However, tests of GENERAL versus either of the models with recovery rates constant by year strongly reject the null hypothesis, further suggesting that constant recovery rates do not fit the observed data.

Thus the biologists would conclude from this analysis that constant recovery rates are not valid for the observed data, but constant survival rates may be.

PROC ESTIMATE Fails

The chances that the numerical optimization procedure will not converge increase with the number of parameters, with parameter estimates against their boundaries, and with model complexity. Generally the failure of the procedure to converge will be obvious: the number of significant digits is less than what was requested, an error message is printed, and/or some parameters never changed from their initial value.

Generally when PROC ESTIMATE does not converge properly, a warning about the possible lack of parameter identifiability is printed. When this warning is received, the user should verify that each of the parameters can actually be estimated with the

model and data input to SURVIV. I have not found a suitable numerical approach to determine parameter identifiability other than the error messages printed from PROC ESTIMATE. Occasionally, the numerical optimization procedure in PROC ESTIMATE will converge, but the variance-covariance matrix of the parameter estimates is not positive-definite. Lack of positive-definiteness of the variance-covariance matrix is a strong indication that the problem lacks identifiability.

Determination of parameter identifiability with numerical methods is difficult because of the limited numerical precision of any calculations performed. That is, a large problem (many parameters to be estimated) will appear to lack identifiability, even though the parameters are actually identifiable. I have found no suitable method to distinguish numerical problems in a large or poorly conditioned problem from the case where the parameters really are not identifiable. Thus the user must carefully check the model input to SURVIV to verify that parameters are identifiable.

When PROC ESTIMATE does not produce reasonable answers, the user can try several things to correct the situation. The first is change the initial parameter estimates to values closer to the true values, i.e., make better initial guesses. Use the observed data to obtain these guesses rather than just setting the initial value to 0.5. Also, run PROC ESTIMATE with a constrained model to obtain better estimates, and use the INITIAL; RETAIN=_____ feature of PROC ESTIMATE to input the parameter estimates from this reduced model for the more complex model.

Second, PROC ESTIMATE allows an absolute value transformation that may improve convergence of the algorithm when one or more parameters are at their upper or lower boundaries. For example, if $S(1) < 0.75$ is specified as a constraint, and the optimization procedure wants to make $S(1) > 0.75$, a convergence problem may result. By specifying TRANSFORM = ABS on the PROC ESTIMATE statement, the difficulties with constrained parameters may disappear.

Finally, another procedure is available in SURVIV to generate input files for the BMDPAR program of the BMDP statistical package (Dixon 1981). Two files are generated for input to BMDPAR: BMDFUN (NOS), BMDFUN.FOR (VMS), or bmdfun.f (UNIX) contains the FORTRAN source code for the BMDPAR FUN subroutine; and BMDINP (NOS), BMDINP.DAT (VMS) or bmdinp.d (UNIX) contains the input to BMDPAR. PROC BMDPAR utilizes the same INITIAL and CONSTRAINT statements as PROC ESTIMATE and passes this information on to the BMDPAR program. One advantage of using the BMDPAR program to perform the optimization is the greater flexibility to specify inequality constraints. Once the BMDPAR input file has been generated, the user can modify it to provide constraints not handled by SURVIV. Thus the user can modify the file of instructions to the BMDP-BMDPAR program to handle constraints such as survival $< (1 - \text{recovery})$. Verification of results from SURVIV with BMDPAR is also encouraged to guard against the possibility of a local maximum of the likelihood function being accepted as the global maximum.

Data Simulation and Sample Sizes

The preceding example illustrates the basic data analysis methods of Program SURVIV. Two additional procedures help design survival estimation studies.

PROC SIMULATE performs Monte Carlo simulations for the model defined in the preceding PROC MODEL statements. An example utilizing a simple biotelemetry experiment is shown in Fig. 2 and is also used in White (1983). The question to be answered in this simulation is whether 40 collars in year 1 and 50 collars in year 2 are enough effort to detect changes in survival rates of 0.1. Thus the first PROC ESTIMATE simulates 4 survival rates: 0.6, 0.7, 0.5, and 0.6. The second PROC ESTIMATE uses the same data, but constrains $S(1) = S(2)$ and $S(3) = S(4)$. Thus the PROC TEST procedure then tests for constant survival for years 1 and 2 and years 3 and 4.

Program SURVIV - Survival Rate Estimation with User Specified Cell Probability Functions 11-JAN-84 19:05:08 Page 01

Version 1.1(VMS) Aug.,

1983

INPUT --- PROC TITLE Small Biotelemetry Example for Simulation in User's Manual;

CPU time in seconds for last procedure was 0.04

INPUT --- PROC MODEL NPAR=4 /* Simple Biotelemetry Example */;

INPUT --- COHORT = 40 /* Number of animals collared in year 1*/;

INPUT --- 0:(1.-S(1));

INPUT --- 0:S(1)*(1.-S(2));

INPUT --- 0:S(1)*S(2)*(1.-S(3));

INPUT --- 40:S(1)*S(2)*S(3);

```

INPUT --- COHORT = 50 /* Number of animals collared in year 2*/;
INPUT --- 0:(1.-S(2));
INPUT --- 0:S(2)*(1.-S(3));
INPUT --- 0:S(2)*S(3)*(1.-S(4));
INPUT --- 50:S(2)*S(3)*S(4);

INPUT --- LABELS;
INPUT --- S(1)=Survival rate in year 1;
INPUT --- S(2)=Survival rate in year 2;
INPUT --- S(3)=Survival rate in year 3;
INPUT --- S(4)=Survival rate in year 4;

```

CPU time in seconds for last procedure was 1.04

Fig. 2. Example of estimating survival rates from biotelemetry data when the data are simulated with PROC SIMULATE.

Program SURVIV - Survival Rate Estimation with User Specified Cell Probability Functions 11-JAN-84 19:05:08 Page 02
Small Biotelemetry Example for Simulation in User's Manual Version 1.1(VMS) Aug., 1983

```

INPUT --- PROC SIMULATE NSIM=200 SEED=4567555 PARFILE=SIMDATA ERROR TEST ;
INPUT --- INITIAL;
INPUT --- S(1)=0.6;
INPUT --- S(2)=0.7;
INPUT --- S(3)=0.5;
INPUT --- S(4)=0.6;

```

Number of parameters in model = 4

Number of parameters set equal = 0

Number of parameters fixed = 0

Number of parameters estimated = 4

I	Parameter	S(I)	Lower Bound	Upper Bound	Label
1	1	0.600	0.000	1.000	Survival rate in year 1
2	2	0.700	0.000	1.000	Survival rate in year 2
3	3	0.500	0.000	1.000	Survival rate in year 3
4	4	0.600	0.000	1.000	Survival rate in year 4

```

INPUT --- PROC ESTIMATE NAME=GENERAL /* All parameters individually estimated */ ;
INPUT --- INITIAL;
INPUT --- S(1)=0.6;
INPUT --- S(2)=0.7;
INPUT --- S(3)=0.5;
INPUT --- S(4)=0.6;

```

Number of parameters in model = 4

Number of parameters set equal = 0

Number of parameters fixed = 0

Number of parameters estimated = 4

I	Parameter	S(I)	Lower Bound	Upper Bound	Label
1	1	0.600	0.000	1.000	Survival rate in year 1
2	2	0.700	0.000	1.000	Survival rate in year 2
3	3	0.500	0.000	1.000	Survival rate in year 3
4	4	0.600	0.000	1.000	Survival rate in year 4

Program SURVIV - Survival Rate Estimation with User Specified Cell Probability Functions 11-JAN-84 19:05:08 Page 03
Small Biotelemetry Example for Simulation in User's Manual Version 1.1(VMS) Aug., 1983

```

INPUT --- PROC ESTIMATE NAME=CONSTRAIN /* Sets of 2 parameters constrained. */ ;
INPUT --- INITIAL;
INPUT --- S(1)=0.6;
INPUT --- S(2)=0.7;
INPUT --- S(3)=0.5;
INPUT --- S(4)=0.6;

INPUT --- CONSTRAINTS;
INPUT --- S(1)=S(2);
INPUT --- S(3)=S(4);

```

Number of parameters in model = 4

Number of parameters set equal = 2

Number of parameters fixed = 0

Number of parameters estimated = 2

I	Parameter	S(I)	Lower Bound	Upper Bound	Label
1	1	0.600	0.000	1.000	Survival rate in year 1
2	1	0.600	0.000	1.000	Survival rate in year 2
3	2	0.500	0.000	1.000	Survival rate in year 3
4	2	0.500	0.000	1.000	Survival rate in year 4

Parameter estimates are written to the file SIMDATA with the order being Replicate Number, Model Number, and the parameters (S(I),SE(S(I)),I=1,NPAR). The format used is '(8F10.6)', which generates (NPAR*2+2)/8+1 = 2 'CARDS' per model.

*** ERROR MESSAGE IER=130 FROM ROUTINE OPTMIZ

** WARNING ** Rounding errors became dominant before parameters estimated to NSIG digits for Model CONSTRAIN

** WARNING ** Check to be sure the parameters are identifiable, but the problem may just be ill-conditioned.

*** ERROR MESSAGE IER=130 FROM ROUTINE OPTMIZ

** WARNING ** Rounding errors became dominant before parameters estimated to NSIG digits for Model CONSTRAIN

** WARNING ** Check to be sure the parameters are identifiable, but the problem may just be ill-conditioned.

Program SURVIV - Survival Rate Estimation with User Specified Cell Probability Functions 11-JAN-84 19:05:08 Page 04
 Small Biotelemetry Example for Simulation in User's Manual Version 1.1(VMS) Aug., 1983

S I M U L A T I O N R E S U L T S
 MODEL GENERAL NSIM = 200 Converged = 200

I	MEAN	SE	95% C. I.		Label
	S(I)-Hat	S(I)-Hat	Lower	Upper	
1	0.60625000	0.54008596E-02	0.59566432	0.61683568	Survival rate in year 1
2	0.69703761	0.39232190E-02	0.68934810	0.70472712	Survival rate in year 2
3	0.50514662	0.47237913E-02	0.49588799	0.51440525	Survival rate in year 3
4	0.59308799	0.84027701E-02	0.57661856	0.60955741	Survival rate in year 4

I	MEAN	SE	95% C. I.		Label
	SE(S(I)-Hat)	SE(S(I)-Hat)	Lower	Upper	
1	0.76255140E-01	0.19810598E-03	0.75866852E-01	0.76643427E-01	Survival rate in year 1
2	0.52897333E-01	0.20781961E-03	0.52490007E-01	0.53304660E-01	Survival rate in year 2
3	0.69079840E-01	0.22577465E-03	0.68637322E-01	0.69522358E-01	Survival rate in year 3
4	0.11440702	0.80088804E-03	0.11283728	0.11597676	Survival rate in year 4

Results of Goodness of Fit Tests
 <0.01 <0.05 <0.10 <0.50

3 16 28 99

S I M U L A T I O N R E S U L T S
 MODEL CONSTRAIN NSIM = 200 Converged = 197

I	MEAN	SE	95% C. I.		Label
	S(I)-Hat	S(I)-Hat	Lower	Upper	
1	0.66448527	0.31448248E-02	0.65832141	0.67064912	Survival rate in year 1
2	0.66448527	0.31448248E-02	0.65832141	0.67064912	Survival rate in year 2
3	0.52493834	0.41833879E-02	0.51673890	0.53313778	Survival rate in year 3
4	0.52493834	0.41833879E-02	0.51673890	0.53313778	Survival rate in year 4

I	MEAN	SE	95% C. I.		Label
	SE(S(I)-Hat)	SE(S(I)-Hat)	Lower	Upper	
1	0.43532547E-01	0.12374531E-03	0.43290007E-01	0.43775088E-01	Survival rate in year 1
2	0.43532547E-01	0.12374531E-03	0.43290007E-01	0.43775088E-01	Survival rate in year 2
3	0.60142105E-01	0.20428022E-03	0.59741716E-01	0.60542494E-01	Survival rate in year 3
4	0.60142105E-01	0.20428022E-03	0.59741716E-01	0.60542494E-01	Survival rate in year 4

Program SURVIV - Survival Rate Estimation with User Specified Cell Probability Functions 11-JAN-84 19:05:08 Page 05
 Small Biotelemetry Example for Simulation in User's Manual Version 1.1(VMS) Aug., 1983

Results of Goodness of Fit Tests
 <0.01 <0.05 <0.10 <0.50

7 25 41 124

Likelihood Ratio Test Results

<0.01 <0.05 <0.10 <0.50 <1.00

GENERAL VS. CONSTRAIN 9 33 53 132 197

```

CPU time in seconds for last procedure was 167.20
INPUT --- PROC STOP /* Signal end of analysis */ ;
CPU time in minutes for this job was 2.80
EXECUTION SUCCESSFUL

```

The DETAIL specification on PROC SIMULATE causes a detailed printout of each of the simulations. Thus parameter estimates and the chi-square goodness-of-fit table are both printed. If a large number of simulations are performed, the output produced will be very large. Thus DETAIL should only be used with NSIM = 10 or less.

The PARFILE = file name specification on PROC SIMULATE is used to name an output file to receive the estimates and associated standard errors for each of the parameters to be estimated for each of the PROC ESTIMATES. This file might be used as input to a statistical package to perform a more through analysis of the simulations than the summary printed by PROC SIMULATE. The format of this file is documented in the output from PROC SIMULATE.

A random number seed to simulate data with PROC SIMULATE is provided in the SEED = value specification. Generally, the seed should be a 5 or 7 digit odd random integer. Specifying the same seed for two PROC SIMULATEs will generate identical data if the same model is used.

PROC SAMPLE SIZE is used to estimate the number of animals to band based on the methods described in Brownie et al. (1978:190-193). The example shown in Fig. 3 duplicates the example in Brownie et al. (1978:186-193).

```

Program SURVIV - Survival Rate Estimation with User Specified Cell Probability Functions 11-JAN-84 19:12:46 Page
01

```

Version 1.1(VMS) Aug.,

1983

```

-----
INPUT --- PROC TITLE Sample Size Estimation for Banding Study;

```

```

CPU time in seconds for last procedure was 0.04
INPUT --- PROC SAMPLE SIZE /* Sample size's for Brownie et al. (1978:192) example */;

INPUT --- S-BAR=0.6;
INPUT --- S'-BAR=0.5;
INPUT --- F-BAR=0.07;
INPUT --- CV(S)=0.04;
INPUT --- CV(S')=0.04 0.07 0.03;
INPUT --- K=2 10 1;

```

SAMPLE SIZE ESTIMATION FOLLOWING BROWNIE ET AL. 1976:190-193

NOTATION	INTERPRETATION
S-BAR	Mean adult survival rate per year
S'-BAR	Mean young survival rate per year
F-BAR	Mean adult recovery rate per year
CV(S)	Coefficient of variation of mean adult survival rate
CV(S')	Coefficient of variation of mean young survival rate
K	Number of years to conduct banding effort
M	Number of young to band each year
N	Number of adults to band each year

Fig. 3. Example input to Program SURVIV for PROC SAMPLE SIZE and the output produced.

```

Program SURVIV - Survival Rate Estimation with User Specified Cell Probability Functions 11-JAN-84 19:12:46 Page
02

```

Sample Size Estimation for Banding Study

Version 1.1(VMS) Aug.,

1983

```

-----

```

S-BAR	S'-BAR	F-BAR	CV(S)	CV(S')	K	N	M
0.60	0.50	0.07	0.04	0.04	2	22560	27269
0.60	0.50	0.07	0.04	0.04	3	5698	16597
0.60	0.50	0.07	0.04	0.04	4	2554	15982
0.60	0.50	0.07	0.04	0.04	5	1462	23514
0.60	0.50	0.07	0.04	0.04	6	959	736686

0.60	0.50	0.07	0.04	0.04	7	686	-16917
0.60	0.50	0.07	0.04	0.04	8	520	-7295
0.60	0.50	0.07	0.04	0.04	9	411	-4331
0.60	0.50	0.07	0.04	0.04	10	336	-2949
0.60	0.50	0.07	0.04	0.07	2	22560	6395
0.60	0.50	0.07	0.04	0.07	3	5698	2798
0.60	0.50	0.07	0.04	0.07	4	2554	1740
0.60	0.50	0.07	0.04	0.07	5	1462	1257
0.60	0.50	0.07	0.04	0.07	6	959	987
0.60	0.50	0.07	0.04	0.07	7	686	816
0.60	0.50	0.07	0.04	0.07	8	520	700
0.60	0.50	0.07	0.04	0.07	9	411	616
0.60	0.50	0.07	0.04	0.07	10	336	553

```

CPU time in seconds for last procedure was    0.28
- INPUT --- PROC STOP /* End of this run. */ ;

```

```

CPU time in minutes for this job was    0.01

```

```

EXECUTION SUCCESSFUL

```

The first number in each of the specification cards [S-BAR, S'-BAR, F-BAR, CV(S), CV(S'), and K], provides the value for that parameter to be used in the sample size calculation. However, because a range of parameter values are usually required, two additional values can be used to specify a range of values. Thus S-BAR = 0.1 0.5 0.1 tells PROC SAMPLE SIZE to use a range of S from 0.1 to 0.5 in increments of 0.1. Note that specifying ranges for all 5 input parameters can result in many pages of sample sizes to be output.

Additional Procedures

PROC STOP is used to exit the program and is provided to stop execution in a data file when additional models are included in the file, but cannot be executed in this particular run because the wrong PROC MODEL has been loaded into core. PROC STOP allows the stacking of multiple models in an input file, but only running the first one. A text editor can be used to move models from within the input file to the top.

PROC BROWNIE will generate PROC ESTIMATE type output from input to the programs ESTIMATE and BROWNIE described in Brownie et al. (1978:155-158). The fixed format requirements of these 2 programs has been relaxed, so that free format input can be used. Each numerical entry should be separated from its neighbors by a space or comma. To use SURVIV on this input, a semicolon must be put at the end of each card image, i.e., each card of input to the Brownie et al. programs is treated as a statement in SURVIV. PROC BROWNIE generates the appropriate COHORT and cell probability cards for input to PROC MODEL, lists these cards on the output, and then runs PROC MODEL to generate the estimation subroutine. Once the estimation subroutine is loaded, PROC BROWNIE calls PROC ESTIMATE to generate estimates and likelihood function values for the models appropriate to the type of input. Thus PROC BROWNIE can be viewed as a combination of PROC MODEL and PROC ESTIMATE statements. PROC BROWNIE will generate each of the models analyzed in the Brownie et al. programs, and if PROC TEST is used, test results between models can be generated. The big advantage of using SURVIV to analyze these models is that additional hypothesis tests can be performed by using PROC ESTIMATE to analyze models not specified in the Brownie et al. programs. For example, to test if survival is constant for years 1970-75 but different than years 1976-1980, the appropriate constraint cards can be used. The definitions of the S(I)'s must be read from a previous run of PROC BROWNIE unless the user is very familiar with the coding scheme used.

PROC IDENTIFIABLE is somewhat useful in locating which pair of parameters are most highly correlated. The input specifications ADD and MULTIPLY allow the user to determine how the parameter value is changed to calculate the numerical derivatives used in this process. Generally PROC IDENTIFIABLE is a last resort at determining the non-identifiable parameters in a model.

File Names and Execution Line

Program SURVIV utilizes five files: to read input from, to write output on, to construct the FORTRAN 77 subroutine on, to construct the BMDPAR FUN subroutine on, and to construct the BMDPAR input on. The default file names for VAX/VMS are SURVIN.DAT, SUPVLP.DAT, SUBROUT.FOR, BMDFUN.FOR, and BMDINP.DAT, for CDC/NOS they are SURVIN, SURVLP, SUBROUT, BMDFUN, and BMDINP, and for UNIX they are survin, survlp, subrou.f, bmdfun.f, and bmdinp.d, respectively. For DOS, the default input and output files are the keyboard and monitor, respectively. The FORTRAN 77 routine is constructed on the EST.FOR file. Other output files are the same as the VAX/VMS version of the program.

The input file can be modified on the execute line by specifying I=filename. Likewise, the output from Program SURVIV can be placed on file XYZ by specifying L=XYZ on the execution line. The subroutine output file and the BMDPAR files can be changed only by modifying the program. The DOS version of the program cannot read the execution line. Rather, SURVIV asks for the user to enter the execution line parameters from the keyboard after the code is executed.

Besides the I and L parameters on the execution line, three additional parameters can be specified. COMPILE RUN (for NOS just COMPILE) tells PROC MODEL to write the estimation subroutine to disk and stop execution, that is, the correct model has not yet been replaced in the core image. If COMPILE RUN is not specified, execution proceeds assuming that the correct model is in core. NOECHO causes the input lines to not be printed in the output. This option can save some paper when the output is printed, but is not advised because of the difficulty in locating errors in the input. LINES = # specifies the number of output lines to print per page, with the default LINES = 60.

Installation Guide for Mainframes

Program SURVIV is provided on a 9-track, ASCII, 1600 bpi, unlabeled tape with 3 files. The first file is the current source code for VAX/VMS, the second is the current source code for CDC/NOS, and the third is the input for multiple examples that run correctly, including all of the examples shown in this manual. The three files are 80 character record lengths, blocked at 960 characters per block, giving 12 card images per block. To set up Program SURVIV on your system, dump the appropriate source code and the input data file from the tape. Because Program SURVIV is written in ANSI FORTRAN 77, few problems should be encountered in compiling the source code with an ANSI standard FORTRAN 77 compiler. Versions of the code are also available for the UNIX and LTSS operating systems upon request.

The first step to implement Program SURVIV is to create an object deck (binary) file of the entire program with the FORTRAN 77 compiler. This object file should be saved, because the likelihood function subroutine must be replaced every time a PROC MODEL constructs a new subroutine.

Next a procedure file should be written to execute the program twice for each run, first to generate the model subroutine, and then to execute the program with this subroutine replacing the default (null) model subroutine. The basic steps that this procedure should perform are:

- Execute SURVIV with option COMPILE RUN specified,

- Execute the FORTRAN compiler with the subroutine file generated in the previous step,

- Replace the new object deck of subroutine EST in the SURVIV
object file, and

- Execute SURVIV again with the new EST subroutine.

Examples of this control language procedures for VAX/VMS and CDC/NOS are shown in Figs. 4 and 5, respectively. In both cases, 3 parameters are passed when the procedure is executed: input file name, output file name, and a shortcut flag to indicate that the current EST subroutine in the SURVIV object is correct for the PROC MODEL specifications in the input file. Thus if multiple runs are made with the same model, the overhead of writing a subroutine, compiling it, and replacing it in the SURVIV object are avoided by specifying the shortcut flag.

```

$ ! VAX/VMS Procedure file to execute SURVIV
$ ! with input from P1, output on P2, and P3
$ ! used to shortcut compilation and linking
$ ! of a new MODEL.
$ ON ERROR THEN EXIT
$ DATAFILE := "SURVIN.DAT"
$ IF P1 .NES. "" THEN DATAFILE := 'P1'
$ OUTFILE := "SUPVLP.LIS"
$ IF P2 .NES. "" THEN OUTFILE := 'P2'
$ ! Logical name SYS$DISK is defined as the user's path name
$ RUNSURV := $SYS$DISK:SURVIV.EXE
$ IF P3 .NES. "" THEN GOTO SHORTCUT
$ ! Copy SURVIV object from system
$ COPY [SURVIV]SURVIV.OBJ SURVIV.OBJ
$ LINK SURVIV/NOMAP
$ RUNSURV I='DATAFILE' L='OUTFILE' COMPILE RUN
$ FORTRAN/NOLIST SUBROUT
$ LINK SUBROUT,SURVIV/EXECUTABLE
$ SHORTCUT: CONTINUE
$ RUNSURV I='DATAFILE' L='OUTFILE'
$ EXIT

```

Fig. 4. Control language procedure for VAX/VMS operating system.

```

.PROC,SURVIV,DATAFIL=SURVIN,OUTFILE=SURVLP,SHORTCT=$NO$/YES$.
IFE,$SHORTCT$.NE.$YES$,SHORT.
ATTACH,SURVOBJ=SURVIV/UN=SURVIV.
REWIND(DATAFIL,SUBROBJ,OUTFILE)
RETURN(OUTFILE)
SURVOBJ(I=DATAFIL,L=OUTFILE,COMPILE)
FTN5(I=SUBROUT,L=0,B=SUBROBJ,REW)
LIBEDIT(I=0,L=0,B=SUBROBJ,P=SURVOBJ,N=SURVOB2)
ENDIF(SHORT)
RETURN(OUTFILE)
REWIND(DATAFIL)
SURVOB2(I=DATAFIL,L=OUTFILE)
REVERT. SURVIV SUCCESSFUL
EXIT.
REVERT,ABORT. *** SURVIV ABORTED ***

```

Fig. 5. Cyber Control Language (CCL) procedure for CDC/NOS operating system.

Installation Guide for DOS Microcomputers

SURVIV is distributed on two 5 1/4 inch DSDD floppy diskettes. The first contains the source code with separate files for each subroutine. The second floppy contains the object code generated with the Ryan-McFarland FORTRAN 77 compiler. To run SURVIV, at least 512k of RAM is required, plus a hard disk. For most problems, the size of the SURVIV.EXE file exceeds 360k, thus the hard disk requirement. A batch file (SURVIVE.BAT) is included on the second floppy disk to execute SURVIV, including the compile run to create EST.FOR, executing the Ryan-McFarland compiler to generate EST.OBJ, recreating SURVIV.EXE, and running SURVIV again to perform the numerical estimation phase of the process. The batch file is interactive, telling the user what input to enter. Any DOS editor can be used to create the input file. Example input is included in the SURVIN file on the floppy diskettes. A copy of this user's manual is also included on the floppies as USERMAN.WP, to be printed with WordPerfect.

Setting Larger Program Dimensions

Program SURVIV has been set up with flexible array dimensions through the use of FORTRAN 77 PARAMETER statements. All the critical dimensions that would need to be changed for larger problems are in the INCLUDE file MODEL.C. These PARAMETERS are well documented with comment statements. Values which can be changed are MAXPAR (maximum number of parameters that can be estimated), MAXCHT (maximum number of COHORTS allowed in a model), MXCLAS (maximum number of probability cells within a cohort), MAXMOD (maximum number of models that can be generated with PROC ESTIMATE and then tested with PROC TEST), and MAXCHS (the maximum size of the character string holding the input buffer). DOS does not allow character strings larger than 255 characters, so MAXCHS is set to 255. This limitation means that statements longer than about 120 characters are not allowed. Also, cell probability expressions in PROC MODEL may not exceed 255 characters in the DOS version of SURVIV.

Literature Cited

Brownie, C., D. R. Anderson, K. P. Burnham, and D. S. Robson. 1978. Statistical inference from band recovery data--A handbook. U. S. Fish and Wildlife Service Res. Publ. No. 131, Washington, D. C. 212 pp.

Dixon, W. J.(ed.). 1981. BMDP Statistical Software 1981. Univ. of California Press, Berkeley, Calif. 726 pp.

White, G. C. 1983. Numerical estimation of survival rates from band recovery and biotelemetry data. J. Wildl. Manage. 47(3):716-728.