Center for Demography and Ecology

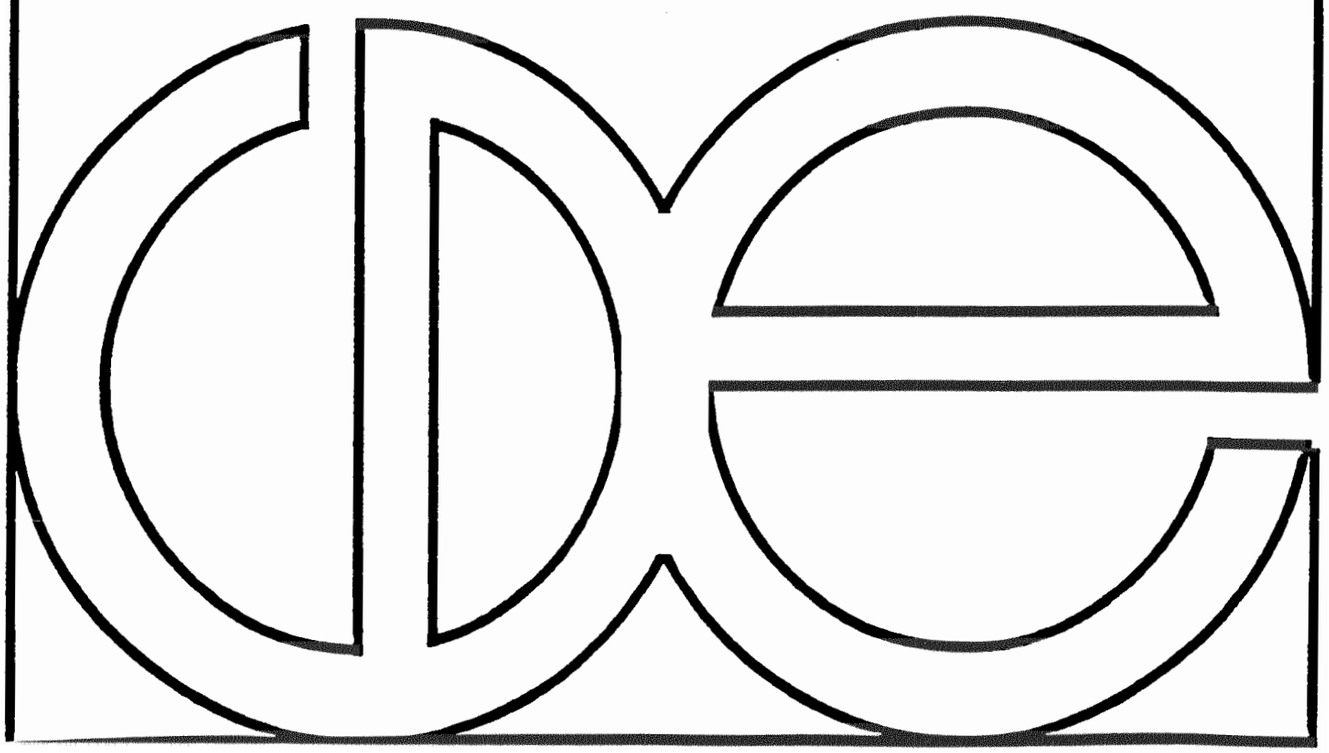
University of Wisconsin-Madison

A Comparative Evaluation of Selected Statistical Software for Computing Loglinear Models

Nancy McDermott

Cynthia White

CDE Working Paper 94-28



A Comparative Evaluation of Selected Statistical Software for Computing Loglinear Models

Nancy McDermott

Cynthia White

Social Science Computing Cooperative
University of Wisconsin - Madison

December 1994

The Center for Demography & Ecology receives core support for Population Research from the Institute for Child Health and Human Development (P30 HD05876).

Table of Contents

1. Introduction

This paper is a comparative evaluation of statistical software for computing loglinear models. The following statistical packages were included in the evaluation: STATA (version 3.1), SAS (version 6.09), SPSS (version 4.0), GLIM (version 3.77) and LIMDEP (version 6.0).

The results from the loglinear analysis runs are first presented for each of the five software packages on a 2×2×2 contingency table. Important and unique features of the analyses are noted. In general, the output is not interpreted, except where clarification is needed or where a statistic provided may not be one commonly used. Following the output, comparisons of package features are provided.

Next, a larger data set consisting of a 36×2×2×4×2 contingency table is used to compare performance (on UNIX) among the five packages. For each package, an attempt is made to fit a sequence of models, beginning with the saturated model and continuing through models of decreased complexity, until a model can be fit without running out of memory. The largest model the package can compute is reported. Then, a comparison of times between each of the packages is made for two models, one simple and one more complex.

Finally, recommendations of the appropriate package to use in certain situations are provided.

The results for each software package are presented in the following order: STATA, SAS, SPSS, GLIM, and LIMDEP. The ordering, although not random, is not meant to reflect any ranking of preference. SAS and SPSS output is presented before output for GLIM, and LIMDEP because they are the packages mostly commonly used and thus the ones readers will most likely want to compare against STATA.

1.1 Data

The first data set analyzed is a 2×2×2 contingency table from Agresti (1990, pp. 171-177). The data represent a study of the effects of racial characteristics on whether individuals convicted of homicide receive the death penalty. The variables are death penalty verdict (P) - yes or no, race of defendant (D) - white or black, race of victim (V) - white or black, and the cell count (COUNT). There are 326 subjects.

Agresti considered several models for these data but favored the following model:

$$\log(m_{ijk}) = \mu + \lambda_i^{d} + \lambda_j^{v} + \lambda_k^{p} + \lambda_{ij}^{dv} + \lambda_{jk}^{vp}$$

where "d" is defendant's race, "v" is victim's race, and "p" is whether or not the individual received the death penalty. This model was fit using each of the five packages. Parameter estimates along with residuals and predicted values were also requested.

For simplicity, models will be referred to using a notation that lists only the highest order effects for each variable. Using this notation, the above model will be referred to as (VP, DV) for the rest of this discussion.

The second example used for performance comparisons analyzes a 36×2×2×4×2 contingency table based on the 5% PUMS. The variables include 36 occupation/industry categories (O), nativity (N) with two categories, sex (S) with two categories, ethnicity (E) with four categories, and time (T) with two categories (1980 or 1990). The table has a total of 1152 cells.

2. Loglinear Models in STATA

STATA has two commands for carrying out a loglinear analysis: LOGLIN and POISSON. LOGLIN is a user contributed program which is available as an "ado" file. LOGLIN is distributed with a subscription to the Stata Technical Bulletin (STB). The SSC Co-op currently subscribes to STB and has all the user contributed procedures including LOGLIN. Both LOGLIN and POISSON estimate a Poisson maximum-likelihood regression for the number of occurrences of an event. The main difference between the two commands is in the generation of indicator variables. Indicator variables are not generated automatically with the POISSON command as they are in the LOGLIN command. However, with the LOGLIN command, you are restricted to four effects. Results from both LOGLIN and POISSON are presented below.

2.1 Analysis using the LOGLIN Command

The following commands were used to read in the data and fit the model (VP, DV) with the LOGLIN command:

```
infile d v p count using dp.dat
loglin count defend victim penalty, fit(victim penalty, defend victim) resid
```

Different variable names had to be assigned than were used with the other packages because LOGLIN would not work with one-letter variable names. Thus, DEFEND is equivalent to D, VICTIM is equivalent to V, and PENALTY is equivalent to P as used in the other packages.

STATA automatically generates the indicator variables associated with the variables specified. FIT(VICTIM PENALTY, DEFEND VICTIM) specifies that LOGLIN should fit a model with the two-way interactions between VICTIM and PENALTY and DEFEND and VICTIM. The main effects DEFEND, VICTIM, and PENALTY are fit automatically.

The RESID command is optional and instructs STATA to produce the predicted cell values, residuals, and standardized residuals. The output from the above commands is shown below:

```
Variable defend = A
Variable victim = B
Variable penalty = C
Margins fit: victim penalty, defend victim
Note: Regression-like constraints are assumed.  The first level of each
variable (and all interactions with it) will be dropped from estimation.

Iteration 0: Log Likelihood = -18.941284
Iteration 1: Log Likelihood = -18.782349
Iteration 2: Log Likelihood = -18.781738
```

```
Poisson regression                                Number of obs     =          8
Goodness-of-fit chi2(2)      =      1.882         Model chi2(5)     =    394.033
Prob > chi2                  =     0.3903         Prob > chi2       =     0.0000
Log Likelihood               =    -18.782         Pseudo R2         =     0.9130

------------------------------------------------------------------------------
       count |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
          A2 |   -.8741451   .1499851    -5.828   0.000    -1.16811   -.5801797
        AB22 |    3.311649   .3785702     8.748   0.000    2.569666    4.053633
          B2 |   -3.782016   .5514818    -6.858   0.000     -4.8629   -2.701132
        BC22 |    1.057941   .4635394     2.282   0.022.    .1494208    1.966462
          C2 |    1.813738   .1968962     9.212   0.000    1.427829    2.199648
       _cons |    3.052501   .1878376    16.251   0.000    2.684346    3.420656
------------------------------------------------------------------------------
```

STATA reports the model goodness-of-fit chi-squared value, 1.882, along with its degrees of freedom, 2 . It also reports the probability that a chi-square with the right number of degrees of freedom would be greater than the goodness-of-fit chi-square. Large values of the probability value indicate that the model is a good fit. STATA also reports a chi-square value for the overall model and its corresponding probability value.

The output labelled `Pseudo R2` is equal to 1- ln L(f)/ln L(i). However, pseudo R^2 statistics lack the straightforward explained-variance interpretation of true R^2 in ordinary least squares regression.

The log likelihood value reported by STATA is different than that reported by SAS even though all the other output was identical within round off error. STATA uses the following equation for computing the log likelihood:

$$L = \sum_j -\ln(Y_j!) - \mu_j + Y_j \ln(\mu_j)$$

where Y_j is the cell count and μ_j is the predicted cell count for the jth observation. SAS drops terms involving the binomial coefficients from this equation which accounts for the difference in log likelihoods reported for the two packages.

Next, STATA produces parameter estimates or coefficients. These should be interpreted as deviations from an (omitted) baseline level. This is the default for STATA unless you specify ANOVA at the end of the command line. Anova-like constraints produce parameter estimates that are deviations from the grand mean. STATA also produces the standard error of the estimate, the standardized estimate, z, and the probability that a standard normal score would be greater than the standardized estimate. Finally, STATA gives a 95% confidence interval for the estimated coefficient.

2.2 Analysis using the POISSON Command

The POISSON command does not generate the indicator variables automatically so the TABULATE command is used below to generate indicator variables for the main effects and then the GENERATE command is used to compute the interation variables between the main

effects. GENERATE was used because TABULATE can not generate indicator variables for interaction effects.

```
quietly tabulate d, gen(ddum)
quietly tabulate v, gen(vdum)
quietly tabulate p, gen(pdum)
generate vp=vdum2*pdum2
generate dv=ddum2*vdum2
```

Note that DDUM1, VDUM1, and PDUM1 were not included in the model so that it would not be overdetermined. Following is the POISSON command used to perform the loglinear analysis and the results:

```
. poisson count ddum2 vdum2 pdum2 vp dv
Iteration 0: Log Likelihood = -18.941284
Iteration 1: Log Likelihood = -18.782349
Iteration 2: Log Likelihood = -18.781738

Poisson regression                              Number of obs     =          8
Goodness-of-fit chi2(2)      =      1.882       Model chi2(5)     =    394.033
Prob > chi2                  =     0.3903       Prob > chi2       =     0.0000
Log Likelihood               =    -18.782       Pseudo R2         =     0.9130

------------------------------------------------------------------------------
       count |      Coef.   Std. Err.       z     P>|z|       [95% Conf. Interval]
-------------+----------------------------------------------------------------
       ddum2 |   -.8741451   .1499851    -5.828   0.000      -1.16811   -.5801797
       vdum2 |   -3.782016   .5514818    -6.858   0.000       -4.8629   -2.701132
       pdum2 |    1.813738   .1968962     9.212   0.000      1.427829    2.199648
          vp |    1.057941   .4635394     2.282   0.022      .1494208    1.966462
          dv |    3.311649   .3785702     8.748   0.000      2.569666    4.053633
       _cons |    3.052501   .1878376    16.251   0.000      2.684346    3.420656
------------------------------------------------------------------------------
```

The output is identical to that of the LOGLIN command discussed above.

The residuals are produced as part of the output or they can also be produced by using the LIST command after either LOGLIN or POISSON:

```
. list cellhat resid stdres

     cellhat    resid   stdres
  1.  21.168   -2.168   -0.471
  2. 129.832    2.168    0.190
  3.   0.482   -0.482   -0.694
  4.   8.518    0.482    0.165
  5.   8.832    2.168    0.730
  6.  54.168   -2.168   -0.295
  7.   5.518    0.482    0.205
  8.  97.482   -0.482   -0.049
```

In this case, CELLHAT is the estimated expected cell frequencies. Residuals are calculated as actual cell count minus the estimated expected cell count. Standardized residuals are calculated as residual divided by the square root of the estimated expected cell count.

3. Loglinear Models in SAS

SAS has two procedures which carry out a loglinear analysis: CATMOD and GENMOD. The CATMOD procedure is an old SAS procedure which fits linear models to functions of response frequencies. It can be very cumbersome to use. The GENMOD procedure was recently introduced in version 6.09 of SAS and fits generalized linear models as defined by Nelder and Wedderburn (1972). It is much easier to use for a loglinear analysis than the CATMOD procedure and is what will be used in this discussion.

Following is the SAS code for reading in the death penalty data and printing it out:

```
proc format;
   value race 1="white" 2="black";
   value death 1="yes" 2="no";
run;

data dp;
   infile "dp.dat";
   input d v p count;
   format d v race.;
   format p death.;
run;

proc print; run;
```

PROC FORMAT was used to assign value labels to each of the variables. The results from PROC PRINT follow:

OBS	D	V	P	COUNT
1	white	white	yes	19
2	white	white	no	132
3	white	black	yes	0
4	white	black	no	9
5	black	white	yes	11
6	black	white	no	52
7	black	black	yes	6
8	black	black	no	97

You analyze this 2×2×2 contingency table by means of a generalized linear model with a log link function. The error distribution appropriate to the counts is Poisson. You specify the link function and error distribution in SAS's GENMOD procedure with the DIST=POISSON and LINK=LOG options on the MODEL statement. The following SAS code fits the model (VP, DV) as was discussed in the previous section:

```
proc genmod;
   class d v p;
   model count=d v p v*p d*v
      / dist=poisson link=log obstats type1 type3;
run;
```

DRACE, VRACE, and PENALTY are specified as CLASS variables so that PROC GENMOD automatically generates the indicator variables associated with these variables. The output follows:

Model Information

Description	Value
Data Set	WORK.DP
Distribution	POISSON
Link Function	LOG
Dependent Variable	COUNT
Observations Used	8

Class Level Information

Class	Levels	Values
D	2	black white
V	2	black white
P	2	no yes

Criteria For Assessing Goodness Of Fit

Criterion	DF	Value	/DF
Deviance	2	1.8819	0.9409
Scaled Deviance	2	1.8819	0.9409
Pearson Chi-Square	2	1.4313	0.7157
Scaled Pearson X2	2	1.4313	0.7157
Log Likelihood	.	1079.6473	.

Analysis Of Parameter Estimates

Parameter			DF	Estimate	Std Err	ChiSquare	Pr>Chi
INTERCEPT			1	3.0525	0.1878	264.0868	0.0000
D	black		1	-0.8741	0.1500	33.9681	0.0000
D	white		0	0.0000	0.0000	.	.
V	black		1	-3.7820	0.5515	47.0310	0.0000
V	white		0	0.0000	0.0000	.	.
P	no		1	1.8137	0.1969	84.8544	0.0000
P	yes		0	0.0000	0.0000	.	.
V*P	black	no	1	1.0579	0.4635	5.2089	0.0225
V*P	black	yes	0	0.0000	0.0000	.	.
V*P	white	no	0	0.0000	0.0000	.	.
V*P	white	yes	0	0.0000	0.0000	.	.
D*V	black	black	1	3.3116	0.3786	76.5237	0.0000
D*V	black	white	0	0.0000	0.0000	.	.
D*V	white	black	0	0.0000	0.0000	.	.
D*V	white	white	0	0.0000	0.0000	.	.
SCALE			0	1.0000	0.0000	.	.

NOTE: The scale parameter was held fixed.

LR Statistics For Type 1 Analysis

Source	Deviance	DF	ChiSquare	Pr>Chi
INTERCEPT	395.9153	0	.	.
D	395.8049	1	0.1104	0.7396
V	363.3485	1	32.4564	0.0000
P	137.9294	1	225.4192	0.0000
V*P	131.6796	1	6.2497	0.0124
D*V	1.8819	1	129.7977	0.0000

LR Statistics For Type 3 Analysis

Source	DF	ChiSquare	Pr>Chi
D	1	22.2276	0.0000
V	1	48.5684	0.0000
P	1	222.6827	0.0000
V*P	1	6.2497	0.0124
D*V	1	129.7977	0.0000

Observation Statistics

COUNT	Pred	Xbeta	Std	HessWgt	Lower
19	21.1682244	3.05250121	0.18783757	21.1682244	14.6486229
132	129.831776	4.86623958	0.08593254	129.831776	109.707188
0	0.48214751	-0.7295052	0.5185052	0.48214751	0.17451289
9	8.51787306	2.14216667	0.33409018	8.51787306	4.42540562
11	8.83177572	2.1783561	0.2110295	8.83177572	5.84008801
52	54.1682243	3.99209447	0.12897636	54.1682243	42.0687611
6	5.51789997	1.70799735	0.40920107	5.51789997	2.47435063
97	97.48214	4.57966918	0.10106497	97.48214	79.9647151

Observation Statistics

Upper	Resraw	Reschi	Resdev
30.5894775	-2.1682244	-0.4712615	-0.4796708
153.648	2.1682243	0.19028901	0.18976302
1.33208628	-0.4821475	-0.6943684	-0.9819852
16.3949178	0.48212694	0.16519459	0.16367185
13.3560081	2.16822428	0.72959221	0.70243387
69.7476333	-2.1682243	-0.2945994	-0.2965983
12.3051356	0.48210003	0.20523444	0.20234969
118.83701	-0.48214	-0.0488327	-0.048873

The `Criteria For Assessing Goodness Of Fit` table contains the log likelihood value along with two statistics that are helpful in assessing the goodness of fit of a given model, the scaled deviance and Pearson's chi-square statistic. The scaled versions of these statistics are identical for this example because no dispersion parameter was specified. The values for the scaled deviance and Pearson chi-square statistics indicate that the model fits the data well.

The log likelihood value reported by SAS is different than that reported by STATA even though all the other output was identical within round off error. SAS uses the following equation for computing the log likelihood:

$$L = \sum_i [r_i \log(p_i) + (n_i - r_i)\log(1 - p_i)]$$

where r_i is the number of events, n_i is the number of trials, and p_i is the predicted mean for the ith observation. Terms involving the binomial coefficients have been dropped from the equation (which is ok since these terms do not affect parameter estimates or standard errors). STATA, on the other hand, does not drop these terms which accounts for the difference in log likelihoods reported for the two packages.

Next, the parameter estimates are given in the output. By default, SAS omits the last level of each variable so that the model is not overdetermined. Parameter estimates of zero with zero degrees of freedom in the table reflect this. The parameter denoted SCALE in the table is the scale variable and is held constant at one because no dispersion parameter was specified. The chi-square value given in the table is the Wald chi-squared statistic and tests the null hypothesis that the coefficient is zero. GENMOD has several options for including confidence intervals for the parameter estimates but none were specified in this example.

The `LR Statistics For Type 1 Analysis` table is optional output that was requested with the TYPE1 option on the MODEL statement. A Type 1 analysis consists of fitting a sequence of models, beginning with a simple model with only an intercept term, and continuing through a model of specified complexity, fitting one additional effect on each step. Likelihood ratio statistics are computed between successive models. These statistics can be used in a test of hypothesis of the significance of each additional term fit. The results depend on the order in which the terms of the model are fitted. The terms are fitted in the order in which they are specified in the MODEL statement.

The `LR Statistics For Type 3 Analysis` table is optional output that was requested with the TYPE3 option on the MODEL statement. A Type 3 analysis is similar to a Type III sums of squares used in analysis of variance, except that likelihood ratios are used instead of sums of squares. In a Type 3 analysis, each effect being tested is adjusted for all other effects in the model. For example, the tests for the main effects VRACE and PENALTY are adjusted for the interaction VRACE*PENALTY. The results do not depend on the order in which the terms of the model are fitted. A Type 3 analysis can consume considerable computation time because a constrained model is fitted for each effect.

The OBSTATS option which was specified on the MODEL statement resulted in the table labelled `Observational Statistics`. For each observation in the data set, the following is printed:

- the predicted value of the mean, denoted by Pred
- the value of the linear predictor, denoted by Xbeta
- the estimated standard error of the predictor, denoted by Std

- the value of the weight in the Hessian matrix at the final iteration, denoted by HessWgt
- approximate lower and upper endpoints for a 95% confidence interval for the predicted value of the mean, denoted by Lower and Upper
- raw residual, denoted by Resraw
- Pearson residual, denoted by Reschi
- deviance residual, denoted by Resdev.

4. Loglinear Models in SPSS

You can analyze loglinear models in SPSS using either the HILOGLINEAR or the LOGLINEAR procedure. Both of these procedures have slightly different features and the features you need will determine which procedure you use. HILOGLINEAR is well suited for hierarchical log-linear models in which models are nested one within the other. It may be best to use HILOGLINEAR when the intent is to select the best possible model. The design statement syntax of HILOGLINEAR is somewhat less complicated than the design syntax of LOGLINEAR. In HILOGLINEAR, lower order interaction terms will automatically be included in the design if the highest order interaction term is specified in the design statement. One drawback of HILOGLINEAR is that it will only produce parameter estimates for the saturated model. LOGLINEAR will produce estimates for all models, but its design statements are not as easy to use. However, LOGLINEAR is well suited for less standard designs, including those where there are structural zeros.

4.1 Analysis using the HILOGLIN Command

The following set of commands were used to read in the data and compute the (VP,DV) model using HILOGLINEAR:

```
data list file='dp.dat' free/ d v p count
variable labels d "Defendant's Race"
variable labels v "Victim's Race"
variable labels p 'Death Penalty'
weight by count
set width 80
hiloglinear d(1,2) v(1,2) p(1,2)
 /design=v*p d*v
```

The statement WEIGHT BY COUNT tells HILOGLINEAR to use the counts from the table as weights in the loglinear procedure. On the HILOGLINEAR command statement, you must specify the levels of each categorical variable that will be used in the model. SPSS automatically generates the indicator variables associated with these variables. Lastly, you must specify a DESIGN subcommand. In this case, the subcommand /DESIGN=V*P D*V specifies that HILOGLINEAR should fit a model with the two-way interactions between V and P and D and V. The main effects D, V, and P will be fit automatically.

The output from the model is as follows:

```
* * * * * * * H I E R A R C H I C A L   L O G   L I N E A R * * * * * * * *

DATA   Information

         8 unweighted cases accepted.
         0 cases rejected because of out-of-range factor values.
         0 cases rejected because of missing data.
       326 weighted cases will be used in the analysis.


FACTOR Information

   Factor  Level  Label
   D           2  Defendant's Race
   V           2  Victim's Race
   P           2  Death Penalty

- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -


DESIGN 1 has generating class

    V*P
    D*V


 Observed, Expected Frequencies and Residuals.

Factor          Code            OBS count  EXP count  Residual  Std Resid
D                      1
V                      1
P                      1           19.0       21.2     -2.17       -.47
P                      2          132.0      129.8      2.17        .19
V                      2
P                      1             .0         .5      -.48       -.69
P                      2            9.0        8.5       .48        .17
D                      2
V                      1
P                      1           11.0        8.8      2.17        .73
P                      2           52.0       54.2     -2.17       -.29
V                      2
P                      1            6.0        5.5       .48        .21
P                      2           97.0       97.5      -.48       -.05


Goodness-of-fit test statistics

    Likelihood ratio chi square =     1.88190    DF = 2  P =  .390
              Pearson chi square =     1.43134    DF = 2  P =  .489
```

Note that although HILOGLINEAR does not produce parameter estimates for the model (because it only prints parameter estimates for saturated models), it does give observed and expected counts as well as residuals and standardized residuals. Finally, the procedure reports the results from the likelihood ratio chi-square test.

4.1 Analysis using the LOGLINEAR Command

The following commands were used to produce an analysis of the model (VP,DV). There are several differences between this command syntax and that for HILOGLINEAR. First, it is possible to get parameter estimates for the model so a /PRINT=ESTIM subcommand has been included. Also, it is not enough to specify the two-way interactions on the design statement. One-way factors must also be included on the design statement.

```
weight by count
loglinear d(1,2) v(1,2) p(1,2)
 /print=estim
 /design=d p v v by p d by v
```

The output from the LOGLINEAR command is shown below:

```
* * * * * * * * * L O G   L I N E A R   A N A L Y S I S * * * * * * * * * *

DATA   Information

          8 unweighted cases accepted.
          0 cases rejected because of out-of-range factor values.
          0 cases rejected because of missing data.
        326 weighted cases will be used in the analysis.


FACTOR Information

   Factor  Level  Label
   D           2
   V           2
   P           2


DESIGN Information

   1 Design/Model will be processed.


* * * * * * * * * L O G   L I N E A R   A N A L Y S I S * * * * * * * * * *

 Correspondence Between Effects and Columns of Design/Model 1

  Starting  Ending
   Column   Column   Effect Name

      1        1     D
      2        2     P
      3        3     V
      4        4     V BY P
      5        5     D BY V
```

Observed, Expected Frequencies and Residuals

Factor	Code	OBS count	EXP count	Residual	Adj Resid
D	1				
V	1				
P	1	19.00	21.17	-2.168	-.937
P	2	132.00	129.83	2.168	.937
V	2				
P	1	.00	.48	-.482	-.744
P	2	9.00	8.52	.482	.744
D	2				
V	1				
P	1	11.00	8.83	2.168	.937
P	2	52.00	54.17	-2.168	-.937
V	2				
P	1	6.00	5.52	.482	.744
P	2	97.00	97.48	-.482	-.744

Goodness-of-Fit test statistics

Likelihood Ratio Chi Square = 1.88190 DF = 2 P = .390
Pearson Chi Square = 1.43134 DF = 2 P = .489

Estimates for Parameters

D

Parameter	Coeff.	Std. Err.	Z-Value	Lower 95 CI	Upper 95 CI
1	-.3908398251	.09464	-4.12964	-.57634	-.20534

P

Parameter	Coeff.	Std. Err.	Z-Value	Lower 95 CI	Upper 95 CI
2	-1.171354500	.11588	-10.10792	-1.39849	-.94422

V

Parameter	Coeff.	Std. Err.	Z-Value	Lower 95 CI	Upper 95 CI
3	.7986103213	.13779	5.79584	.52854	1.06868

V BY P

Parameter	Coeff.	Std. Err.	Z-Value	Lower 95 CI	Upper 95 CI
4	.2644853120	.11588	2.28231	.03735	.49162

D BY V

Parameter	Coeff.	Std. Err.	Z-Value	Lower 95 CI	Upper 95 CI
5	.8279123803	.09464	8.74778	.64241	1.01341

The `Adj Resid` column in the `Observed, Expected Frequencies and Residuals` table contains the adjusted residuals which are formed by dividing each standardized residual by an estimate of its standard error. Standardized residuals shown in the output for LOGLINEAR are formed by dividing each residual by the square root of the expected count. Thus, adjusted residuals are created by dividing through the residuals by the square root of the expected count and the estimated standard error of the residual.

You will notice that the parameter estimates are different from those of the other three packages. This is because SPSS parameterizes the model differently. By default, SPSS computes the parameter estimates by constructing contrasts of the deviation from the overall effect. All the other packages examined computes the parameter estimates by constructing contrasts for each level of a factor to the last level. The CONTRAST subcommand in SPSS's LOGLINEAR command can be used to specify other types of contrasts including that used by other packages, but only for models that do not contain interaction effects.

5. Loglinear Models in GLIM

GLIM fits generalized linear models, as defined by Nelder and Wedderburn (1972). To analyze contingency tables by means of a generalized linear model with GLIM, you specify a log link function and a Poisson error distribution.

The following set of commands were used to read in the data and compute the (VP,DV) model using GLIM:

```
$units 8
$factor d 2 v 2 p 2
$data d v p count
$dinput 7
$yvar count
$error pois
$fit d + v + p + d.v + v.p $
$display e
$look %X2
```

The FACTORS directive identifies the explanatory variables and instructs GLIM to generate the indicator variables associated with these variables. The YVAR directive specifies the dependent variable containing the cell counts. The ERROR specification is Poisson. No LINK directive was specified because the log link is used by default with the Poisson error. The FIT directive fits the loglinear model (VP,DV).

The DISPLAY directive instructs GLIM to display the parameter estimates. Other options for the DISPLAY directive include R, a listing of the Y variate, fitted values and residuals, D, the scaled deviance and degrees of freedom, V, the covariances of the parameter estimates, and C, the correlations of the parameter estimates.

The LOOK directive is used to display the Pearson's chi-square statistic which is helpful is assessing the goodness-of-fit of a given model.

The output follows:

```
$fit d + v + p + d.v + v.p $

scaled deviance = 1.8819 at cycle  3
           d.f. = 2
$display e
         estimate        s.e.     parameter
    1       3.053      0.1878     1
    2     -0.8741      0.1500     D(2)
    3      -3.782      0.5515     V(2)
    4       1.814      0.1969     P(2)
    5       3.312      0.3786     D(2).V(2)
    6       1.058      0.4635     V(2).P(2)
    scale parameter taken as  1.000

 $look %X2
   1.431
```

A Pearson's chi-square statistic of 1.431 with two degrees of freedom indicates that the model fits the data well.

As with SAS, the scale parameter is held constant at one because no dispersion parameter was specified.

6. Loglinear Models in LIMDEP

LIMDEP uses a Poisson regression model to fit loglinear models. As is true with the POISSON command in STATA, the POISSON command in LIMDEP will not generate indicator variables for you. You must create them manually with the CREATE command.

Following is the LIMDEP code for reading in the data and generating the indicator variables needed to fit the model:

```
read; nvar=4; nobs=8; file=dp.dat;
names=d,v,p,count$
create; ddum=d-1;
        vdum=v-1;
        pdum=p-1;
        dv=ddum*vdum;
        vp=vdum*pdum $
```

Following is the POISSON command used to perform the loglinear analysis and the results:

```
poisson; lhs=count; rhs=one,ddum,vdum,pdum,dv,vp $

 MODEL COMMAND: POISSON; LHS=COUNT; RHS=ONE,DDUM,VDUM,PDUM,DV,VP $
 Poisson Regression
 Method=NEWTON; Maximum iterations = 50


 Iterations: Method=D/F/P    Maximum interations  50
 Convergence criteria:       Gradient= 0.100D-03    F= 0.100D-03    b=.100D-04

********************************************************************************
                         Method=D/F/P ; Maximum interations 50
   Convergence criteria: Gradient=   0.1000000E-03
                         Function  =   0.1000000E-03
                         Parameters=   0.1000000E-04
   Starting values:       0.          0.          0.          0.          0.
                          0.

 Iteration:   1 Fn=  -18.82358
 PARAM    3.06    -0.871     -3.65      1.80      3.25     0.981
 GRADNT -0.983    -0.267    -0.557    -0.691E-01-0.229E-11-0.211E-11

 Iteration:   2 Fn=  -18.78180
 PARAM    3.05    -0.874     -3.78      1.81      3.31      1.05
 GRADNT -0.398E-01-0.209E-01-0.375E-01-0.140E-01-0.199E-01-0.138E-01

 Iteration:   3 Fn=  -18.78174
 PARAM    3.05    -0.874     -3.78      1.81      3.31      1.06
 GRADNT -0.607E-04-0.400E-04-0.606E-04-0.131E-04-0.400E-04-0.131E-04
 ** Gradient has converged.
 ** Function has converged.
 ** B-vector has converged.

********************************************************************************
```

```
Poisson Regression
Log-likelihood=    -18.78174
Restricted Log-L=  -215.7984          LR Statistic=    394.0
CHI- squared =  1.4313
G  - squared =  1.8818
N(0,1) used for significance levels.
Variable  Coefficient  Std. Error   t-ratio Prob:t:>x   Mean of X  Sd.Dv.of X
---------------------------------------------------------------------------
Constant    3.0525        0.1878       16.251  0.00000
DDUM      -0.87415        0.1500       -5.828  0.00000  0.50000       0.53452
VDUM       -3.7820        0.5515       -6.858  0.00000  0.50000       0.53452
PDUM        1.8137        0.1969        9.212  0.00000  0.50000       0.53452
DV          3.3116        0.3786        8.748  0.00000  0.25000       0.46291
VP          1.0579        0.4635        2.282  0.02247  0.25000       0.46291
```

The output begins with the iteration history for the maximum likelihood estimates. The log-likelihood is the same one reported by STATA. (Refer to section 2.1 for a discussion of the differences between STATA and SAS in how each computes the log-likelihood.) The output labelled `Restricted Log-L` is the log-likelihood function for a model with only a constant. The output labelled `LR Statistic` is the chi-square value for the fit of the overall model. The `Chi-Squared` tests the null hypothesis that the coefficients for all the terms in the model except the constant are zero. The statistic labelled `G-squared` is equivalent to the statistic labelled `Deviance` by SAS and is a test of goodness-of-fit.

7. Feature Comparisons

The table below summarizes the features of the loglinear analyses for each of the software packages examined. The columns headed STATAP and STATAL refer to STATA's POISSON and LOGLIN procedures. The columns headed SPSSH and SPSSL refer to SPSS's HILOGLIN and LOGLINEAR procedures, respectively.

Key: ✓ **feature available** – **feature not available**	**STATAP**	**STATAL**	**SAS**	**SPSSH**	**SPSSL**	**GLIM**	**LIMDEP**
Model Specification							
Automatic Dummy Variable Generation	–	✓	✓	✓	✓	✓	–
No Limitation on the Number of Effects that can be Specified	–	–	✓	✓	✓	✓	–
Shortcut Syntax for Specifying Complex Models	–	✓	✓	✓	–	✓	–
Model Update (Add or Delete Terms)	–	–	–	–	–	✓	–
Intercept Suppression Option	–	–	✓	–	–	✓	✓
Automatic Model Selection Methods	✓	–	–	✓	–	–	–
LR Statistics for Adjusted "Sums-of-Squares"	–	–	✓	–	–	–	–
LR Statistics for Type I Analysis or Partial Association Table	–	–	✓	✓	–	–	–
Maximum Likelihood Estimates							
Wald's Statistic for Parameter Estimate		–	✓	–	–	–	–
t-statistic or z-statistic for Parameter Estimate	✓	✓	–	✓	✓	–	✓
Confidence Intervals for Parameter Estimates	✓	✓	✓	–	✓	–	–
Contrasts for Constructing Tests involving Parameter Estimates	✓	✓	✓	–	✓	–	–
Statistics by Observation including Residuals and Predicted Values	✓	✓	✓	✓	✓	✓	✓
Criteria for Assessing Fit							
Loglikelihood	✓	✓	✓	✓	✓	–	✓
Goodness-of-Fit Test -- Deviance	✓	✓	✓	✓	✓	✓	–
Goodness-of-Fit Test -- Pearson's Chi-Square	–	–	✓	✓	✓	✓	✓
Model Chi-Square	✓	✓	–	–	–	–	✓
Correlation/Covariance Matrix of Parameter Estimates	✓	✓	✓	–	✓	✓	✓

8. Performance Comparisons

A 36x2x2x4x2 contingency table based on the 5% PUMS was used for the performance comparisons. The variables include 36 occupation/industry categories (O), nativity (N) with two categories, sex (S) with two categories, ethnicity (E) with four categories, and time (Y) with two categories (1980 or 1990). The table has a total of 1152 cells.

Recall that the POISSON command in STATA and LIMDEP does not construct the indicator variables automatically. Nor do these two packages have a utility that makes it easy to construct the design matrix for a large complicated model. For example, the saturated model for the 36x2x2x4x2 contingency table used in this paper has 1152 columns. To expedite this process, the GLMMOD procedure in SAS 6.09 was used. GLMMOD is designed solely for outputting design matrices; no models are fit. You identify the indicator variables with a CLASS statement and then specify a model with the MODEL statement. Following is the code for the saturated model:

```
proc glmmod outparm=codes outdesign=xy noprint;
   class ethnic sex native occup yr;
   model count=yr|occup|ethnic|sex|native;
run;
```

The columns of the design matrix are then written to a SAS data set called XY and the information regarding the association between model effects and design matrix columns is written to a SAS data set called CODES. The design matrix was then written (via a DATA step) to an ascii file suitable for input to STATA or LIMDEP.

For each package, an attempt was made to fit a sequence of models, beginning with a saturated model and continuing through models of decreased complexity, until a model could be fit without running out of memory. All the packages were run on CDE2S because at the time of this writing, CDE2S had the most memory installed of any of the UNIX machines at the SSC Co-op.

STATA and LIMDEP could not be included in several of these performance comparisons because they have limits the number of variables or effects in a given model. LIMDEP has a limit of 200 variables in a data set and STATA limits the number of effects in a model to 400. The saturated model, the model including all four-way interactions, and the model including all three-way interactions all have over 400 effects and thus could not be included in the comparisons for STATA. There is also a limit of four effects for STATA's LOGLIN command, so it could not be used for any of the comparisons because there were five effects in the example used. STATA's POISSON command had no such restriction though and so it was included in the comparison of the model with all two-way interactions. Even the two-way interaction model has over 200 variables in the design matrix so LIMDEP was not included in any of the performance comparisons.

Only GLIM could fit the saturated model (ONSEY) or the model involving all four-way interactions (ONSE, ONSY, OSEY, ONEY, NSEY). SPSS and SAS required more memory than was available on CDE2S. (These jobs were run at times when no one else was running jobs so that the package could utilize all the memory available on CDE2S. Otherwise, you could run a job and run out of memory simply because other jobs were running, forcing the jobs to share the available memory.)

Next, a fit was attempted for the model involving three-way interactions. The model (YES, YEN, YEO, YSN, YSO, YNO, ESN, ESO, ENO, SNO) was run for SAS and SPSS. Both packages were able to fit this model without running out of memory.

Two models were used for the CPU comparisons: the model with all two-way interactions (YE, YS, YN, YO, ES, EN, EO, SN, SO, NO) and the model with all three-way interactions (YES, YEN, YEO, YSN, YSO, YNO, ESN, ESO, ENO, SNO). The UNIX `time` command was used to compare the CPU times of the statistical packages. The program for each package was run 10 times for the smaller model and three times for the larger model. The average time in minutes spent in execution of the program (not real time) are shown in the table below. All runs were done on one machine (CDE2S) because a machine's processing speed would affect the time reported.

	Time in Minutes Spent in Execution of the Program	
	Model: (YE, YS, YN, YO, ES, EN, EO, SN, SO, NO)	**Model: (YES, YEN, YEO, YSN, YSO, YNO, ESN, ESO, ENO, SNO)**
STATA	2.50 (3)	Model contains more effects than STATA allows
SAS	0.34 (1)	12.29 (1)
SPSS	--	--
GLIM	1.40 (2)	15.79 (2)

The number in parentheses represents the package's relative rank for performance.

Times could not be reported for SPSS because the `time` command did not accurately report these for SPSS. It is still possible to get a rough idea of how SPSS compared to the other packages if you look at the average results of the real time that elapsed during execution of the program. The real time for the SPSS programs was much longer than for the other programs. For example, the average (over three runs) real time for the larger model for SPSS was 255.24 minutes, 27.70 minutes for GLIM, and 20.96 minutes for SAS.

9. Recommendations

All the statistical packages gave the same results within round-off error. Some packages reported more significant digits than others. However, this does not indicate that the package computed statistics with more accuracy. You should be conservative about the number of significant digits you report from statistical packages; usually the package reports too many digits leading you to believe they are all significant when they are not.

Even though no incorrect results were found with the programs run, keep in mind that no intense testing was done to uncover these types of errors. This is worth mentioning here for two of the packages considered. SAS's GENMOD procedure is still experimental, meaning it has not yet been fully tested by SAS Institute. STATA's LOGLINEAR procedure is a user contributed procedure which means that it also has not undergone rigorous testing. You should encounter no problems with either of these procedures for analyses where the data are not ill-conditioned. But, for example, if you have a large complicated model with a large number of empty or nearly empty cells, caution should be taken.

The complexity and number of effects in the model will probably be the deciding factors in your decision about which package to use. For complex models, the POISSON commands in LIMDEP and STATA are not a good choice because they do not construct the indicator variables for the design matrix automatically. And, even though STATA's LOGLIN command will construct the design matrix for you, it has the four factor restriction.

You will also want to avoid LIMDEP and STATA for models with lots of effects because of their restriction on the number of variables that can be used. LIMDEP has a limit of 200 variables in a data set and STATA limits the number of effects in a model to 400.

GLIM, SAS, and SPSS each provide a procedure for computing a loglinear analysis with a minimum of fuss, no matter how complicated the model. You are restricted only by the amount of memory the package requires to fit the model. By far, GLIM required the least amount of memory to compute a loglinear model. GLIM could fit the saturated model for the 36x2x2x4x2 contingency table whereas the largest model that SAS and SPSS could fit was the model involving all three-way interactions.

If you have a fairly large model, you might also want to avoid SPSS if CPU usage is a concern. It was much slower than the SAS and GLIM. SAS and GLIM performed about equally well in this category.

GLIM was the clear winner when it came to performance, but it also offers the fewest options for enhancing your output. SAS and SPSS's loglinear procedures provided lots of useful output that is not available in GLIM or STATA (refer to the table in section 8).

In summary, for large models, GLIM may be the only package that can fit the model without running out of memory. STATA and LIMDEP, with their restrictions on the number of

effects, are clear losers. If you do not want the hassle of generating your own design matrix, avoid LIMDEP and the POISSON command in STATA. For moderate sized models, SAS provides an easy-to-use procedure with lots of useful options. For smaller models, it may be more convenient to use the package you are most familiar with unless you need a particular option.

10. References

Agresti, A. (1990), *Categorical Data Analysis*, New York: John Wiley and Sons, Inc.

Baker, R.J. and J.A. Nelder (1987), *The GLIM System Release 3.77 Manual - Edition 2*, Numerical Algorithms Group Inc., Downers Grove, IL.

Greene, William H. (1992), *Limdep User's Manual and Reference Guide: Version 6*, Bellport, NY: Econometric Software, Inc.

Nelder, J. A. and R.W.M. Wedderburn (1972), "Generalized Linear Models," *Journal of the Royal Statistical Society A*, 135, 370:384.

Norusis, Marija J. (1990), *Advanced Statistics User's Guide*, Chicago, IL: SPSS Inc.

SAS Institute Inc. (1992), *SAS Technical Report P-229, SAS/STAT Software: Changes and Enhancements, Release 6.07*, Cary, NC: SAS Institute Inc.

SAS Institute Inc. (1989), *SAS/STAT Users Guide, Version 6, Fourth Edition, Volume 2*, Cary, NC: SAS Institute Inc.

SPSS Inc. (1990), *SPSS Reference Guide*, Chicago, IL: SPSS Inc.

Stata Corporation (1993), *Stata Reference Manual: Release 3.1*, College Station, TX

Center for Demography & Ecology
University of Wisconsin
1180 Observatory Drive, Rm. 4412
Madison WI 53706-1393
U.S.A.
608/262/2182
FAX 608/262/8400