Center for Demography and Ecology

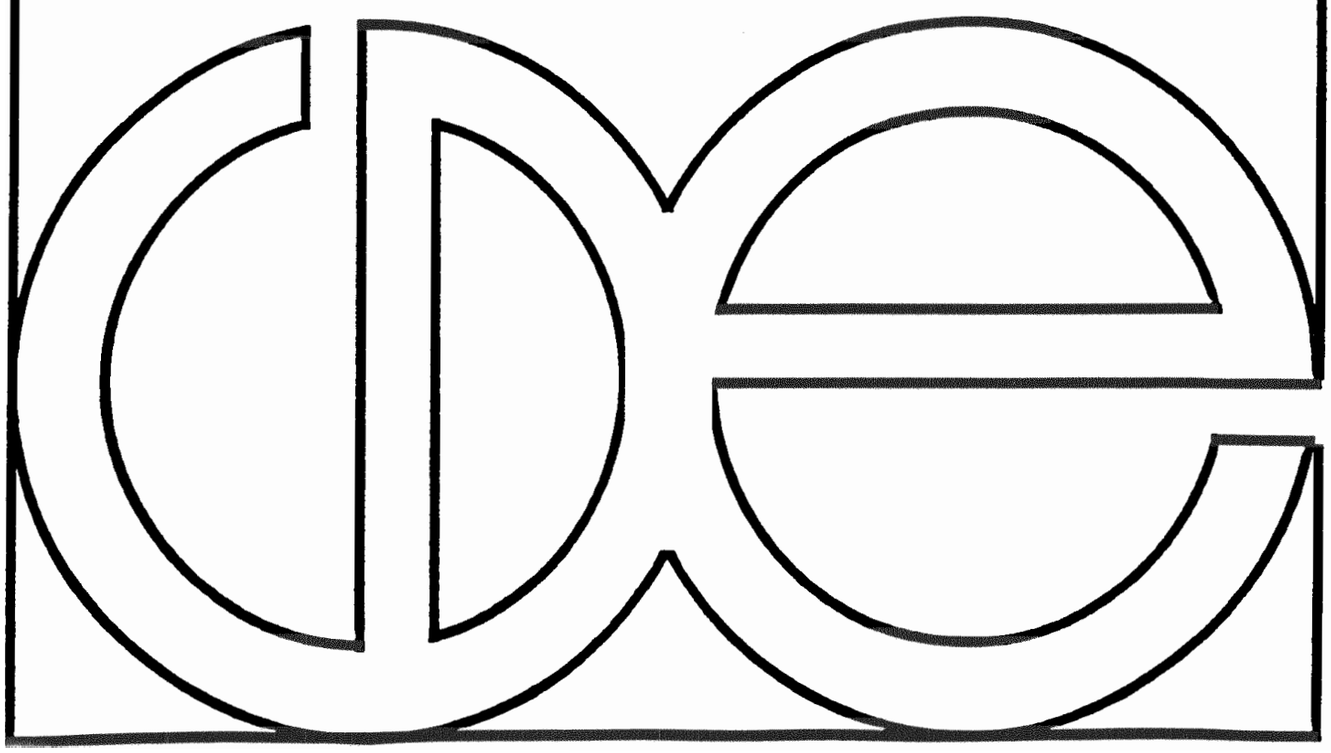
University of Wisconsin-Madison

A Comparative Evaluation of Selected Statistical Software for Computing a Logistic Regression

Nancy McDermott

Cynthia White

CDE Working Paper 94-27



A Comparative Evaluation of Selected Statistical Software for Computing a Logistic Regression

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 a logistic regression. The purpose of this comparison was to see how the newly acquired STATA software package compared with more commonly used packages at the University of Wisconsin Social Science Computing Cooperative. 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), LIMDEP (version 6.0), and S-PLUS (version 3.1).

The results from the logistic regression analysis runs on a small data set are first presented for each of the six software packages. A Hosmer-Lemeshow Goodness-of-Fit test is then presented for the software packages which have this feature.

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

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 containing 4165 observations is used to compare performance (on UNIX) among the four packages. Finally, some recommendations based on features and performance are also presented.

1.1 Data

The first data set analyzed was taken from Agresti (1990, pp. 122-123) and was derived from a small study on senility. A sample of elderly people were given a psychiatric examination to determine whether symptoms of senility were present. This information was used to construct the binary response variable, SYM: 1=symptoms present, 0=no symptoms present. One explanatory variable was included in the model which was the score on a subtest of the Wechsler Adult Intelligence Scale and was called WAIS.

The second example analyzes a larger data set from the 1980 World Fertility Survey in the Cote d'Ivoire. The data represent responses to interviews of a stratified random sample of women ages 15-50 in the Cote d'Ivoire. A total of 5764 women were interviewed. However, for this analysis, only 4165 married, self-reporting fecund women were included in the analysis.

The dependent variable, WANTYES, was a woman's response when asked whether or not she wanted more children. It was coded 0 if the woman wanted more children and 1 if the woman did not want more children. The independent variables were AGE, number of children ever born (NUMLIV), percent of children who died (PERMORT), EDUCATION, RELIGION, and urbanization (CITY). RELIGION and CITY were categorical variables with three levels each.

2. Logistic Regression in STATA

To do logistic regression in STATA, you may use either the LOGIT or LOGISTIC command. The STATA manual states that LOGISTIC is generally preferred to LOGIT for the following reasons:

1. LOGISTIC presents the estimates in terms of odds ratios rather than coefficients.

2. LOGISTIC displays confidence intervals for the odds ratios.

3. After LOGISTIC, you can test the fit, display various summary statistics, graph the ROC curve, and examine residuals and influence statistics.

The LOGISTIC command was used for the analysis that follows.

Following is the STATA code for reading in the data from the senility study and generating crosstabulations of the two variables:

```
. use elder.dta
. label var wais "WAIS score"
. label var sym "Symptoms of Senility"
. sort wais
. tab wais sym
```

WAIS score	Symptoms of Senility 0	1	Total
4	1	1	2
5	0	1	1
6	1	1	2
7	1	2	3
8	0	2	2
9	4	2	6
10	5	1	6
11	5	1	6
12	2	0	2
13	5	1	6
14	5	2	7
15	3	0	3
16	4	0	4
17	1	0	1
18	1	0	1
19	1	0	1
20	1	0	1
Total	40	14	54

The TAB (or TABULATE) command provides the crosstabulation of the WAIS scores by the presence or absence of symptoms.

Use the LOGITISTIC command to perform a logistic regression of SYM on WAIS:

```
. logistic sym wais

Logit Estimates                                   Number of obs =      54
                                                  chi2(1)       =   10.79
                                                  Prob > chi2   =  0.0010
Log Likelihood =  -25.50869                       Pseudo R2     =  0.1746

------------------------------------------------------------------------------
     sym | Odds Ratio   Std. Err.       z     P>|z|      [95% Conf. Interval]
---------+--------------------------------------------------------------------
    wais |     .72359   .0824746    -2.838   0.005       .5787257     .904716
------------------------------------------------------------------------------
```

The overall chi-squared statistic (labelled `chi2(1)`) evaluates the null hypothesis that all coefficients in the model, except the constant, equal zero. The equation is

$X^2 = -2\ (\ln L(\text{i}) - \ln L(\text{f}))$

where ln L(i) is the initial (model with constant only) log likelihood and ln L(f) is the log likelihood for the final iteration. Note that the chi-squared statistic does not have an approximate chi-squared distribution when applied to logistic models like the one in this example having a continuous covariate, unless there are many observations at each observed level of the covariate.

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

The numbers in the "Odds Ratio" column of LOGISTIC's output are amounts by which the odds favoring Y=1 are multiplied, per 1-unit increase in the explanatory variables. STATA also displays 95% confidence intervals for the conditional odds ratio. SAS was the only other statistical package which included this output.

You can get the underlying coefficients for the odds ratios by typing LOGIT without arguments after the LOGISTIC command:

```
. logit

Logit Estimates                                   Number of obs =      54
                                                  chi2(1)       =   10.79
                                                  Prob > chi2   =  0.0010
Log Likelihood =  -25.50869                       Pseudo R2     =  0.1746

------------------------------------------------------------------------------
     sym |      Coef.   Std. Err.       z     P>|z|      [95% Conf. Interval]
---------+--------------------------------------------------------------------
    wais |  -.3235304   .1139798    -2.838   0.005      -.5469266    -.1001342
   _cons |   2.404043   1.191835     2.017   0.044       .0680896     4.739997
------------------------------------------------------------------------------
```

STATA performs a Z test for the parameter estimates. Some packages examined performed Wald's chi-square statistic instead.

2.1 Testing Goodness-of-Fit

To test Goodness-of-Fit when the explanatory variables are continuous, you usually first need to construct categories for the continuous variables and then apply the Goodness-of-Fit tests to these groups. Otherwise, the Goodness-of-Fit statistic does not have an approximate chi-squared distribution when applied to logistic models having a continuous covariate, unless there are many observations at each observed level of the covariate.

Hosmer and Lemeshow (1989) presented ways of forming the categories for the continuous variables for constructing a Goodness-of-Fit statistic. Observations are sorted in increasing order of their estimated probability of having an event outcome. The observations are then divided into nearly equal sized groups.

To request the Hosmer-Lemeshow test for Goodness-of-Fit, first fit the logistic regression and then use the LFIT command to carry out the test. The GROUP() option is used to specify the number of groups you want constructed. Eleven groups were specified in the example below so that the output would be comparable to that of SAS which is presented in section 3.1.

```
. lfit, group(11)

Logistic estimates for sym, goodness-of-fit test
(Table collapsed on percentiles of estimated probabilities)

          no. of observations =          54
                no. of groups =          11
      Hosmer-Lemeshow chi2(9) =        8.42
                        P>chi2 =      0.4925
```

STATA automatically groups the data in 11 groups after ordering on the predicted probabilities. These groups are then used to calculate the Hosmer-Lemeshow chi-square. In this example, you fail to reject the null hypothesis that there is lack of fit for the model. If you leave the GROUP option off the LFIT command, STATA computes the usual Goodness-of-Fit test, not the Hosmer-Lemeshow test for goodness-of fit.

3. Logistic Regression in SAS

SAS has four procedures which can carry out logistic regression: LOGISTIC, GENMOD, CATMOD, and NLIN. The LOGISTIC procedure is by far the most straightforward to use for logistic regression and is what will be used in this discussion.

Following is the SAS code for reading in the data from the senility study and generating crosstabulations of the two variables:

```
data one;
   infile "logit.dat";
   input wais sym;
   label wais="WAIS score";
   label sym="Symptoms of Senility";
run;

proc freq;
   tables wais*sym / nocol norow nocum nopercent;
run;
```

The following SAS code requests the logistic regression of SYM on WAIS:

```
proc logistic descending;
   model sym=wais / risklimits;
run;
```

Unlike other statistical packages, by default SAS models the probability that the event equals zero (SYM=0). To change this to model the probability that the event equals 1 (SYM=1) as in other packages, specify the DESCENDING option on the PROC statement. If you do not add the DESCENDING option, the sign of your parameter estimates will be the opposite in sign of what you would normally get from other statistical packages.

The RISKLIMITS option on the MODEL statement requests confidence intervals for the conditional odds ratio. 95% confidence intervals are computed by default. Only one other statistical package (STATA) provided this output.

The output follows:

```
The LOGISTIC Procedure

Data Set: WORK.ONE
Response Variable: SYM       Symptoms of Senility
Response Levels: 2
Number of Observations: 54
Link Function: Logit
```

Response Profile

Ordered Value	SYM	Count
1	1	14
2	0	40

Criteria for Assessing Model Fit

Criterion	Intercept Only	Intercept and Covariates	Chi-Square for Covariates
AIC	63.806	55.017	.
SC	65.795	58.995	.
-2 LOG L	61.806	51.017	10.789 with 1 DF (p=0.0010)
Score	.	.	9.795 with 1 DF (p=0.0017)

Analysis of Maximum Likelihood Estimates

Variable	DF	Parameter Estimate	Standard Error	Wald Chi-Square	Pr > Chi-Square	Standardized Estimate
INTERCPT	1	2.4040	1.1918	4.0687	0.0437	.
WAIS	1	-0.3235	0.1140	8.0570	0.0045	-0.661626

Conditional Odds Ratio and 95% Confidence Limits

Variable	Odds Ratio	Lower	Upper	Variable Label
INTERCPT	11.068	1.070	114.434	Intercept
WAIS	0.724	0.579	0.905	WAIS score

Association of Predicted Probabilities and Observed Responses

Concordant	= 74.8%	Somers' D	= 0.563
Discordant	= 18.6%	Gamma	= 0.602
Tied	= 6.6%	Tau-a	= 0.220
(560 pairs)		c	= 0.781

The `Criteria for Accessing Model Fit` table gives the various criteria based on the likelihood for fitting a model with the intercept only and for fitting a model with the intercept and explanatory variables. `AIC` is the Akaike Information Criterion and `SC` is Schwartz Criterion. The third column of the table gives the chi-square statistics and p-values for the -2*LL* statistic and for the Score statistic. These test the joint effect of the explanatory variables included in the model. The Score criterion is always missing for the models identified by the first two columns of the table. Note also that the first two rows of the chi-square column are always missing since the AIC and SC criteria are not meaningful in accessing the contribution of covariates.

The Association between Predicted Probabilities and Observed Responses table is a table of measures which includes a breakdown of the number of pairs with different responses and four rank correlation indexes: Somers' D, Goodman-Kruskal Gamma, c, and Kendall's Tau-a.

3.1 Testing Goodness-of-Fit

The LOGISTIC procedure computes the Hosmer-Lemeshow Goodness-of-Fit test. This test was described in Section 2.1.

Just add the LACKFIT option on the MODEL statement to request the Hosmer-Lemeshow Goodness-of-Fit Test:

```
proc logistic descending;
   model sym=wais / lackfit;
run;
```

The output from LACKFIT follows:

```
            Hosmer and Lemeshow Goodness-of-Fit Test

                          SYM = 1                SYM = 0
                    --------------------   --------------------
 Group       Total     Observed    Expected   Observed    Expected
     1           5            0        0.17          5        4.83
     2           5            0        0.34          5        4.66
     3           5            2        0.51          3        4.49
     4           5            1        0.60          4        4.40
     5           5            0        0.75          5        4.25
     6           5            1        1.14          4        3.86
     7           5            1        1.39          4        3.61
     8           5            2        1.66          3        3.34
     9           5            1        1.96          4        3.04
    10           5            4        2.67          1        2.33
    11           4            2        2.80          2        1.20
Goodness-of-Fit Statistic = 9.8651 with 9 DF (p=0.3615)
```

Unlike STATA, SAS does not allow you to specify the number of groups to construct for the Hosmer-Lemeshow Goodness-of-Fit test. SAS uses approximately 10 groups when constructing the test.

STATA reported a Hosmer-Lemeshow Goodness-of-Fit test of 8.42 for this example while SAS reported 9.865. The reason for the difference is unclear because both packages constructed 11 groups after ordering on the predicted probabilities. The difference in test statistics *may* be attributed to the way observations are assigned to the groups. SAS uses the following scheme. Let N be the total number of subjects. Let M be the target number of subjects for each group given by

$$M = [0.1 \times N + .5]$$

The first group consists of the first M observations, the second group consists of the next M observations and so on. Subjects that correspond to the first observation are put in the first group. Suppose there are n_1 subjects for the first observation and n_2 subjects for the second observation. Subjects for the second observation are also put into the first group if

$n_1 < M$ and $n_1 + [0.5 \text{ x } n_2] <= M$

Otherwise, they are placed in the second group.

The Hosmer-Lemeshow Goodness-of-Fit statistic in SAS is obtained by calculating the Person chi-square statistic from the 2 x g table of observed and expected frequencies, where g is the number of groups. The Hosmer-Lemeshow statistic is then compared to a chi-square distribution with (g -2) degrees of freedom.

4. Logistic Regression in SPSS

SPSS has four commands which can carry out logistic regression: LOGISTIC REGRESSION, LOGLINEAR, HILOGLIN, and NLR. The LOGISTIC REGRESSION command is by far the most straightforward to use for logistic regression and is what will be used in this discussion.

Following is the SPSS code for reading in the data from the senility study and generating crosstabulations of the two variables:

```
data list file=logit.dat free / wais sym
variable labels wais "WAIS score"
                sym "Symptoms of Senility"

crosstabs variables=wais(4,20) sym(0,1)
   / tables=wais by sym
```

The following SPSS code requests the logistic regression of SYM on WAIS:

```
logistic regression sym with wais
```

The output follows:

```
      Total number of cases:      54 (Unweighted)
      Number of selected cases:   54
      Number of unselected cases: 0

      Number of selected cases:                  54
      Number rejected because of missing data:   0
      Number of cases included in the analysis: 54


Dependent Variable Encoding:

Original        Internal
Value           Value
     .00        0
    1.00        1

Dependent Variable..   SYM        Symptoms of Senility

Beginning Block Number  0.  Initial Log Likelihood Function

-2 Log Likelihood    61.806315

* Constant is included in the model.


Beginning Block Number  1.  Method: Enter

Variable(s) Entered on Step Number
1..        WAIS       WAIS score

Estimation terminated at iteration number 4 because
Log Likelihood decreased by less than .01 percent.
```

```
                    Chi-Square     df Significance
 -2 Log Likelihood      51.017     52         .5125
 Model Chi-Square       10.789      1         .0010
 Improvement            10.789      1         .0010
 Goodness-of-Fit        51.603     52         .4895


Classification Table for SYM

                     Predicted
                     .00     1.00      Percent Correct
                       0  |     1
Observed               +-------+-------+
   .00        0        |   37  |     3 |    92.50%
                       +-------+-------+
   1.00       1        |    9  |     5 |    35.71%
                       +-------+-------+
                              Overall     77.78%


---------------------- Variables in the Equation -----------------------

Variable            B        S.E.      Wald     df       Sig        R    Exp(B)

WAIS           -.3235       .1140    8.0569      1     .0045  -.3130     .7236
Constant       2.4040      1.1918    4.0686      1     .0437
```

The output labelled `Model Chi-Square` is the difference between -2*LL* for the model with only a constant and -2*LL* for the current model. Thus, the `Model Chi-Square` tests the null hypothesis that the coefficients for all the terms in the model except the constant, are zero. The output labelled `Improvement` is the change in -2*LL* between successive steps of building a model. Since all the variables in this example (only one) were entered in a single step, this statistic is identical to the one labelled `Model Chi-Square`.

The output labelled `R` in the `Variables in the Equation` table is a measure of the partial correlation between the dependent variable and each of the independent variables. A positive value indicates that as the variable increases in value so does the likelihood of the event occurring. If R is negative, the opposite is true. Small values of R indicate that the variable has a small partial contribution to the model.

4.1 Testing Goodness-of-Fit

The LOGISTIC REGRESSION command computes a Goodness-of-Fit test by default as is seen in the output above. However, use caution interpreting this statistic for this example because there are very few observations at each observed level of the covariate. Hence, the Goodness-of-Fit statistic does not have an approximate chi-squared distribution when applied to logistic models like the one in this example. A test similar to the Hosmer-Lemeshow Goodness-of-Fit tests computed by SAS and STATA would be much more appropriate for the data in this example. SPSS does not compute this test, however.

5. Logistic Regression in GLIM

GLIM fits generalized linear models, as defined by Nelder and Wedderburn (1972), which include logistic regression. You need to specify a binomial distribution function with a logit link function. The following commands read in the senility data set:

```
$units 54$
$data wais sym$
$dinput 7$
```

The following commands are used to perform a logistic regression in GLIM with SYM as the dependent variable and WAIS as the independent variable.

```
$yvar sym$
$calc n=1$
$error binomial n$
$fit wais$
$link g$
$display e$
```

The YVAR command specifies the dependent variable. You must set n equal to 1 with the CALC command because there is only one measurement on each person. The ERROR specification is binomial with the total number of observations on each person equal to 1. The LINK G command specifies that a logit link function will be used for the fit. The FIT command fits a logistic model with WAIS as the independent variable.

Finally, the DISPLAY directive instructs GLIM to display the results of the model fit. In this case, DISPLAY E instructs GLIM to display the parameter estimates and their standard errors, including extrinsically aliased parameters. 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 results of the above commands are shown below:

```
$fit wais$
scaled deviance = 51.017 at cycle  4
           d.f. = 52

$display e$
         estimate        s.e.     parameter
    1       2.404       1.186     1
    2     -0.3235      0.1132     WAIS
    scale parameter taken as  1.000
```

The output labelled scaled deviance is -2 times the log likelihood.

5.1 Testing Goodness-of-Fit

GLIM does not provide a test similar to the Hosmer-Lemeshow Goodness-of-Fit tests computed by SAS and STATA.

6. Logistic Regression in LIMDEP

The easiest way to run a simple logistic regression in LIMDEP is with the LOGIT command. The LOGIT command carries out both binomial and multinomial logit models.

Following is the LIMDEP code for reading in the data from the senility study and generating crosstabulations of the two variables:

```
read; nvar=2; nobs=54; file=logit.dat;
names=wais,sym $
crosstab; lhs=wais; rhs=sym $
```

The following LIMDEP code requests the logistic regression of SYM on WAIS:

```
logit; lhs=sym; rhs=one,wais $
```

The output follows:

```
 Multinomial Logit Model
 2 Outcomes: SYM=0     SYM=1
 Coefficients for SYM=0     set to zero.
 Least squares starting values:
 Dep. Var. is binary: SYM=1
 N(0,1) used for significance levels.


Variable  Coefficient  Std. Error   t-ratio Prob:t:>x   Mean of X  Std.Dv.of X
---------------------------------------------------------------------------
Constant  0.84712       0.1817        4.662  0.00000
WAIS     -0.50791E-01   0.1496E-01
95  0.00069   11.574         3.7093

Iterations: Method=NEWTON   Maximum iterations  25
Convergence criteria:      Gradient= 0.100D-03    F= 0.100D-03    b= 0.100D-04

***************************************************************************


                        Method=NEWTON; Maximum iterations 25
   Convergence criteria: Gradient=     0.1000000E-03
                         Function  =   0.1000000E-03
                         Parameters=   0.1000000E-04
   Starting values:   0.8471     -0.5079E-01

 ==> NEWTON ITERATIONS

 Iteration:   1 Fn=   39.60204
 PARAM   0.847    -0.508E-01
 GRADNT   16.5      218.


 Iteration:   2 Fn=   26.33419
 PARAM    1.36    -0.202
 GRADNT   1.92      31.4
```

```
 Iteration:   3 Fn=   25.54230
 PARAM    2.19    -0.298
 GRADNT  0.357      5.67

 Iteration:   4 Fn=   25.50878
 PARAM    2.39    -0.322
 GRADNT  0.188E-01 0.293

 Iteration:   5 Fn=   25.50869
 PARAM    2.40    -0.324
 GRADNT  0.562E-04 0.870E-03

 ** Gradient has converged.
 ** Function has converged.
 ************************************************************************

Maximum Likelihood Estimates

Log-Likelihood.............      -25.509
Restricted (Slopes=0) Log-L.     -30.903
Chi-Squared ( 1)...........       10.789
Significance Level.........     0.10211E-02
N(0,1) used for significance levels.


Variable  Coefficient  Std. Error   t-ratio Prob:t:>x   Mean of X  Std.Dv.of X
---------------------------------------------------------------------------
Constant   2.4040        1.192        2.017  0.04369
WAIS     -0.32353       0.1140       -2.838  0.00453   11.574        3.7093

Frequencies of actual & predicted outcomes
Predicted outcome has maximum probability.


            Predicted
Actual       0     1        TOTAL
  0         37     3           40
  1          9     5           14

Total       46     8           54
```

The LOGIT output begins with results from a least squares estimation which LIMDEP uses to obtain starting values. The iteration history for the maximum likelihood estimates appears next. The output labelled `Restricted Log-L` is the log-likelihood function for a model with only a constant. The `Chi-Squared` tests the null hypothesis that the coefficients for all the terms in the model except the constants are zero.

6.1 Testing Goodness-of-Fit

LIMDEP does not provide a test similar to the Hosmer-Lemeshow Goodness-of-Fit tests computed by SAS and STATA.

7. Logistic Regression in S-Plus

You can perform logistic regression in S-Plus using either the function GLIM() or the more general GLM(). This handout will describe both functions.

The general form of the function GLIM() is as follows:

```
glim(x,y,n, error="gaussian",link="identity",wt=vector of weights)
```

X is a vector or matrix of explanatory variables. Each column of a matrix should represent a matrix and each row should represent an observation.

Y is a vector containing the response variable.

Optional arguments include:

N required when error is "BINOMIAL", but ignored otherwise. N is the vector of denominators for the proportion (and Y is the number of "successes" out of N trials).

ERROR a character string specifying the error structure of the model. Possible values are GAUSSIAN, BINOMIAL, POISSON, GAMMA, and INVERSE GAUSSIAN.

LINK a character string specifying the link function. Possible values are IDENTITY, LOG, LOGIT, PROBIT, SQRT, INVERSE, and LOGLOG.

WT a vector of case weights.

To see the results of the generalized linear regression, use the function GLIM.PRINT().

The function GLM() produces a somewhat different type of output than the function GLIM() and the results from GLM() can be used to generate fitted values. To use the function GLM(), you must specify your model in the format Y~X1 + X2 + X3. The general format for the function is as follows:

```
glm(formula, family = binomial, weights)
```

FORMULA is a formula expression of the form response~predictors

FAMILY is a family object - a list of functions and expressions for defining the link and variance functions. Families supported are GAUSSIAN, BINOMIAL, POISSON, GAMMA, INVERSE.GAUSSIAN and QUASI. Family functions can take arguments, as in BINOMIAL(LINK=PROBIT).

Following is the S-Plus code for reading in the data from the senility study. X is the WAIS score, Y is the senility indicator, and N is a vector of 1's, since there is only one case for each outcome of Y.

```
> x
[1]  9 13  6  8 10  4 14  8 11  7  9  7  5 14 13 16 10 12 11 14 15 18  7 16  9
[26] 9 11 13 15 13 10 11  6 17 14 19  9 11 14 10 16 10 16 14 13 13  9 15 10 11
[51] 12  4 14 20
> y
[1]1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[39] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
> n
[1]1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
[39] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
```

The following command fits a logistic regression model with the function GLIM() and stores the results as glim.elder.

```
> glim.elder_glim(x,y,n,error="binomial",link="logit")
```

In this case, ERROR="BINOMIAL" and LINK="LOGIT" because the functional form is the logistic model. To see the results of the logistic regression, use GLIM.PRINT().

```
> glim.print(glim.elder)

             coef se(coef)     z      p  Deviance df change     p
Intercept  2.4040 1.1916    2.02 0.0436     61.81 53
       X1 -0.3235 0.1139   -2.84 0.0045     51.02 52 10.8   0.001
```

The deviance for the model with the WAIS score, X1, is 51.02 while the deviance for the model without is 61.81. The difference in the deviances is equal to 10.8. When compared to a Chi-Square distribution with d.f. 1, the deviance is significant at the .001 level. Thus, the WAIS score does account for variation in senility scores.

To calculate the odds ratio, y/(1-y), simply calculate exp(B). This represents the amount by which the odds favoring Y=1 are multiplied, per 1-unit increase in this X variable.

```
> exp(-.3235)
 .723612
```

This indicates that those with higher WAIS scores were less likely to be diagnosed as senile.

You can perform the same logistic regression using GLM():

```
> wais.glm_glm(y~x,weights=n,family=binomial(link=logit))
```

To see output from this model, use the command SUMMARY():

```
> summary(wais.glm)
```

```
 Call: glm(formula = y ~ x, family = binomial(link = logit), weights = n)
Deviance Residuals:

       Min          1Q      Median          3Q       Max
 -1.670194 -0.7401801 -0.4749403 0.5200231 2.115723

Coefficients:

                Value Std. Error   t value
(Intercept)  2.404007  1.1897106  2.020666
          x -0.323526  0.1136966 -2.845520

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 61.80632 on 53 degrees of freedom

Residual Deviance: 51.01738 on 52 degrees of freedom

Number of Fisher Scoring Iterations: 4

Correlation of Coefficients:
  (Intercept)
x -0.9567246
```

The null deviance is the deviance from fitting the model with only the constant, while the residual deviance is the deviance from fitting a model with X1 included. The difference between the two is 61.80 - 51.02 = 10.78. When compared against a Chi-Squared distribution with 1 d.f., this value is significant at the p=.001 level.

To calculate p-values for the Chi-Square distribution, use the PCHISQ() command. PCHISQ takes the following parameters: value, d.f. The function returns the cumulative probability for the stated value, so to obtain a p-value, you need to subtract the given probability from 1.

```
> 1-pchisq(10.78,1)
[1] 0.001026027
```

The predicted values for this model are stored under the name FITTED.VALUES and are part of the objects returned by the function GLM(). To see them, type the following:

```
> fit_wais.glm$fitted.values
> round(fit,3)
 [1] 0.376 0.142 0.614 0.454 0.303 0.752 0.107 0.454 0.240 0.535 0.376 0.535
[13] 0.687 0.107 0.142 0.059 0.303 0.186 0.240 0.107 0.080 0.032 0.535 0.059
[25] 0.376 0.376 0.240 0.142 0.080 0.142 0.303 0.240 0.614 0.043 0.107 0.023
[37] 0.376 0.240 0.107 0.303 0.059 0.303 0.059 0.107 0.142 0.142 0.376 0.080
[49] 0.303 0.240 0.186 0.752 0.107 0.017
```

To graph the fitted values against the values of X, the WAIS scores, use the PLOT() command.

```
> plot(x,fit,main=c("Fitted values against WAIS scores"))
```

The subcommand MAIN specifies the main title.

7.1 Testing Goodness-of-Fit

S-PLUS does not provide a test similar to the Hosmer-Lemeshow Goodness-of-Fit tests computed by SAS and STATA. It is fairly straightforward, however, to construct a similar test by using programming statements provided by S-PLUS to construct categories for the continuous variable. The procedure followed for constructing the test was what was recommended by Dr. Robert Mare in his SOC 952 class, spring 1994. The authors are grateful to Vickie Chang, Mare's reader, for providing the example used for this portion of the analysis.

In this case, you can group the WAIS variable into three categories: 4-9, 10-13, and 14-20. The midpoints of these categories are 6.5, 11.5, and 17. You can compare a linear logistic model with only the midpoints as the dependent variable to a saturated model with the three categories.

To create a categorical variable, use the CUT() command. This breaks the data into three categories: 4-9, 10-13, and 14-20.

```
> x.cat_cut(x,breaks=c(0,9,13,20))
> x.cat
 [1] 1 2 1 1 2 1 3 1 2 1 1 1 1 3 2 3 2 2 2 3 3 3 1 3 1 1 2 2 3 2 2 2 1 3 3 3 1
2 [39] 3 2 3 2 3 3 2 2 1 3 2 2 2 1 3 3
attr(, "levels"):
[1] " 0+ thru 9"  "9+ thru 13"  "13+ thru 20"
```

Then, use the categories created by the CUT command to create dummy variables for the logistic regression. The following commands create a vector of 0's and then replace those whenever X.CAT is equal to 1.

```
> x1_rep(0,54)
> x1[x.cat==1]_1
```

You can do the same for the second and third categories.

```
> x2_rep(0,54)
> x2[x.cat==2]_1
> x3_rep(0,54)
> x3[x.cat==3]_1
```

Finally, create a matrix by combining the three dummy variables.

```
> x.matrix_cbind(x1,x2,x3)
```

Note, however, that you will need to drop the first variable when you use the X matrix in the logistic regression.

```
> eldercat.glim_glim(x.matrix[,-1],y,n,
+ error="binomial",link="logit")
```

```
> glim.print(eldercat.glim)
             coef se(coef)      z      p  Deviance df change      p
Intercept  0.2513 0.5039    0.50 0.6180     61.81 53
       X1 -1.9859 0.8038   -2.47 0.0135
       X2 -2.3307 0.9005   -2.59 0.0096     51.40 51 10.4   0.0055
```

You can compare this model to a model with a linear component. First, substitute the midpoints of the intervals for the category.

```
x.int_x.cat
x.int[x.cat==1]_6.5
x.int[x.cat==2]_11.5
x.int[x.cat==3]_17

>elderlin.glim_glim(x.int,y,n,error="binomial",link="logit")

> glim.print(elderlin.glim)
             coef se(coef)      z      p  Deviance df change      p
Intercept  1.6772 0.98594   1.70 0.0889     61.81 53
       X1 -0.2514 0.09287  -2.71 0.0068     52.76 52 9      0.0026
```

To assess Goodness-of-Fit, compare the deviance (the change in likelihood for the full model from a model with just the data) for the categorical model with the deviance from the linear logit model. 52.76 - 51.40 is equal to 1.36. The probability that a Chi-Square with 1 degree of freedom is greater than or equal to 1.36 is .24. This means that you cannot reject the null hypothesis that all the coefficients for the independent variables in the first model are zero. Thus, the linear logit model is an adequate model.

8. Feature Comparisons

The table on the next page summarizes the features of the logistic regression analyses for each of the software packages examined. The key for the table is as follows:

- ✓ software package has feature
- X software package has feature but requires extra coding
- – software package does not have feature
- W wrong or inappropriate value which user may be tempted to use

Only one "W" was assigned. That was for the Goodness-of-Fit Test provided on SPSS's output. The test provided is inappropriate for models like the one in this example having a continuous covariate, unless there are many observations at each observed level of the covariate. Otherwise, the chi-squared statistic does not have an approximate chi-squared distribution.

Key: ✓ feature available X feature available but requires extra coding – feature not available W wrong or inappropriate value for feature	STATA	SAS	SPSS	GLIM	LIMDEP	SPLUS
Model Specification						
Automatic Dummy Variable Generation	–	–	✓	✓	–	–
Automatic Model Selection Methods	✓	✓	✓	–	–	–
Model Update (Add or Delete Terms)	–	–	–	✓	–	✓
Intercept Suppression Option	✓	✓	✓	X	✓	✓
Link Functions for Response Probabilities	✓	✓	–	✓	✓	✓
Construct Contrasts for Categorical Independent Variables	✓	–	✓	–	–	–
Maximum Likelihood Estimates						
Standardized Estimate	–	✓	–	–	–	–
Odds Ratios	✓	✓	✓	–	–	–
Wald's Statistic for Parameter Estimate	–	✓	✓	–	–	–
t-statistic or z-statistic for Parameter Estimate	✓	–	–	–	✓	✓
Confidence Intervals for Parameter Estimates	✓	–	–	–	–	–
Confidence Intervals for Odds Ratios	✓	✓	–	–	–	–
Classification Table	✓	✓	✓	✓	✓	–
Regression Diagnostics	✓	✓	✓	✓	✓	✓
Criteria for Assessing Fit						
-2 * Loglikelihood	✓	✓	✓	✓	✓	✓
Goodness-of-Fit Test	✓	✓	W	–	X	–
Akaike Information Criterion	–	✓	–	–	–	–
Schwartz Criterion	–	✓	–	–	–	–
Score Criterion	–	✓	–	–	–	–
Pseudo R-Square	✓	–	–	–	–	–
Rank Correlation between Observed Response and Predicted Probabilities						
Somer's D	–	✓	–	–	–	–
Gamma	–	✓	–	–	–	–
Tau-a	–	✓	–	–	–	–
Correlation Matrix of Parameter Estimates	✓	✓	–	✓	–	✓
Partial Correlation between Response and each Independent Variable	–	–	✓	–	–	–

9. Performance Comparisons

The larger data set from the 1980 World Fertility Survey in the Cote d'Ivoire was used for the performance comparisons. These data contained 4165 observations with six explanatory variables. Only a basic logistic regression was specified for each package. In order to make a fair comparison, the Hosmer-Lemeshow Goodness-of-Fit test was not specified because this test can be computationally intensive and thus inflate the times for the two packages that can compute the test. Also, for SPSS, the /EXTERNAL subcommand on the LOGISTIC command was used to conserve memory.

The UNIX `time` command was used to compare the performances of the statistical packages. Each program for each package (for both the small and large data set) was run 10 times. The average time in seconds spent in execution of the program (not real time) is shown in the table below. All runs were done on one machine (cde2s) because a machine's processing speed would affect the time reported. Times could not be reported for SPSS because the `time` command did not accurately report these for SPSS.

	Time in Seconds Spent in Execution of the Program	
	Senility Data	**Fertility Data**
STATA	0.00 (1)	3.11 (2)
SAS	0.19 (3)	1.50 (1)
SPSS	--	--
GLIM	0.08 (2)	6.09 (3)
LIMDEP	0.45 (4)	31.22 (5)
S-PLUS	1.35 (5)	13.01 (4)

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

10. Recommendations

All the statistical packages gave the same results within round-off error (except for results for the Hosmer-Lemeshow test for Goodness-of-Fit which are discussed in section 3.1). 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.

All of the statistical packages considered provided a procedure for computing a logistic regression with a minimum of fuss. Unless you need a particular option or CPU time is a factor, it may be more convenient just to use the package you are most familiar with. SAS's LOGISTIC procedure provided the most options, especially in the areas of criteria for accessing the fit of the model and rank correlation between the observed response and the predicted probabilities. If you only wanted a Goodness-of-Fit test for accessing fit though, either SAS or STATA would be good choices. Both packages compute the Hosmer-Lemeshow test automatically. Also, SAS and STATA were the only packages that computed odds ratios with confidence intervals. SPSS computed the odds ratio but without confidence intervals. One notable difference between SAS and STATA is the test provided for the maximum likelihood estimates. SAS computes a Wald's statistic and STATA computes a Z-test, so the test you want may affect your decision about which package to use.

SPSS provided most of the options that SAS and STATA did. One nice feature offered by SPSS that only one other package (GLIM) offered was the automatic construction of dummy variables for categorical independent variables. In addition, it provides five different types of contrasts for the categorical variables used for interpreting the coefficients in different ways. Another feature unique to SPSS was the partial correlation between the response and each independent variable.

STATA had the best performance in executing the smaller data set. GLIM followed after that with the next best time. S-PLUS had the worst execution time. You can consider the execution time for the smaller analysis (Senility data) to represent the package's "overhead". For example, SAS is a huge program with lots of overhead and its relatively long execution time for the smaller analysis reflects this. For the larger analysis, however, most of the execution time is spent actually computing the logistic regression. SAS ranked highest here.

STATA's good performance was not unexpected because STATA puts all the data in memory instead of using swap space. Although this method of execution can put a huge drain on a machine's memory when a large job is executing, it usually means the package will execute jobs very quickly. What was unexpected was that SAS actually performed better than STATA for the larger analysis. SAS does make use of swap space instead of putting everything in memory. Times similar to LIMDEP and S-PLUS were what was expected. One possible explanation is that SAS's logistic regression procedure is more efficient than the other packages. SAS or STATA appear to be a good choices when CPU time is a factor.

11. 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.

Hosmer, D.W., Jr. and Lemeshow, S. (1989), *Applied Logistic Regression*, New York: John Wiley and Sons, 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

Statistical Sciences, Inc. (1991), *S-Plus User's Manual Vol 1 and 2*, Seattle, WA

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