Nonlinear Regression and Logistic Regression for Binary Dependent Data

MacDonald R. Phillips
PhillipsM@gao.gov

March 2001

The routines in this folder solve nonlinear regression problems using the Gauss_Newton Method with Step-Halving and logistic regression problems for binary dependent data using the probit, normit, or complementary log-log link functions.

NOTE: These programs are offered "as is." I make no claim that they are entirely bug free, although I believe they are. If you encounter any problems with the programs, please send me an email so I can correct them.

NOTE: These programs require the use of the Statistics with List Editor Flash application.

My aim is to teach you how to use these programs, not to teach you statistics. Thus, when I mention the ANOVA table or logit link function, I assume you already know what they are and/or when they are used, or are learning about them either in a class or on you own.

Fitting data to an arbitrary function is more of an art than a science. Convergence to a solution can be very sensitive to the initial starting values, i.e., guesses. And, there may be more than one solution or local minimum around the starting values. If you get error messages such as singular matrix, this may mean that there is no solution or you need to choose a different set of starting values.

The routines are in a group file, nonlin.89g. Use your TI-Graph Link program to ungroup them and then transfer them to a new folder, nonlin, on your calculator. There is a menu program; this needs to be run in order to use the regression routines. The custom menu sets up three pull-down menus: Nonlinear, Logistic, and Tools.

Both regression routines use a data matrix created with the Data/Matrix Editor. The data matrix consists of the variables in any order. The first row of the matrix must be the variable names; I recommend one-letter names. In any case, just make sure they do not conflict with any of the variable names in the nonlin folder or TI reserved names. (There are no variables with one-letter names in the folder.) When performing regression there is no need to use all of the variables in a dataset; this means you can do many different regressions on a dataset without having to enter a new data matrix each time. It also means you can do "model building" and test the significance of adding variables to a regression equation; this ability is one of the routines.

Give the data matrix a name and save it. Open the Tools menu and select AddDS. You will be prompted to enter the name of the data matrix. When you run NonLin() or Logit() you will be asked to select a dataset from the list of datasets created with AddDS. AddDS also archives the dataset.

Nonlinear Regression

Most regressions are linear regressions or can be transformed into linear regressions. Some, however, cannot be transformed. For instance, an exponential equation of the form

$$y_i = b_0 \times \exp(b_1 \times (year_i - 1790)) + b_2$$

may model the growth of the U.S. population by decade from 1790, but it cannot be transformed into a linear regression problem. (If the b_2 variable was not there, it could be transformed into a linear regression problem.) A nonlinear regression program is needed. Press F1, Nonlinear, and select the first menu item, NonLin(), and then press enter, once or twice as needed. This sets up a data input form.

The first item is “Select dataset.” “pop” is the offered choice. Press the right arrow key to display the other datasets. “pop,” of course, has the U.S. population figures, in millions, from 1790 to 1990, by decade. Since the first example uses this dataset, press the down arrow key to go to the next line. Here you enter the dependent variable; for this example it is “p.” The next line is used to enter the regression equation; the equation there is the one displayed above. The next line is used to enter the independent variable(s) as a list. In this case there is only one; it is {y}. The next line is used to enter the parameters as a list; for the above equation they are {b0, b1, b2}. The last line is used to enter the initial values or guesses for the parameters, again as a list. The values there are {20, .03,10}.

After the data is input, press enter to begin computing the regression. The program keeps you informed as to what is going on. It first sets up the necessary matrices, etc., needed to compute the regression. After that, each iteration is displayed along with the current sum-of-squared-errors. At the end, a message will be displayed indicating whether or not the routine converged to an answer. (As will be seen below, the convergence criteria can be changed.)

The other options under F1 (Nonlinear) display the output of the regression. Option 2, FeqN, displays the fitted equation. In this case it is

$$2.50746E^{-7}(1.01055)^{y} - 39.2905$$

It is unfortunate that the calculator simplifies the answer instead of leaving it in the form of an exponential equation.

Option 3 under the F1 menu, OutN, displays a matrix of the parameters, their values, standard errors, t values and probability(t). For this regression the output is

"Parm"	"Value"	"STD"	"$t(18)$"	"$\Pr ob(t)$"
$b0$	36.054	4.11342	8.76498	$6.51895E^{-8}$
$b1$	.010494	.000506	20.7386	$5.1456E^{-14}$
$b2$	-39.2905	5.88023	-6.6818	.000003

(The 18 in “t(18) is the degrees of freedom of the t statistics.)

Option 4 under the F1 menu, Iter, displays a matrix of the iterations the program went through to reach the estimated values of the parameters. The iteration number, or subiteration number, parameter values, and sum-of-square errors are displayed for each iteration.

Option 5 under the F1 menu, ANOVA, displays the analysis of variance matrix.

"Source"	"DF"	"SS"	"MS"	""F"	"Prob(F)"
"Reg"	2.	122823.	61411.4	3331.81	$7.47334E^{-24}$
"Error"	18.	331.773	18.4318	""	""
"Total"	20.	123154.	""	""	""

Options 6,7, and 8 under the F1 menu display the R square, adjusted R square, and standard error of the regression statistics. For this problem they are: .997306, .997007, and 4.29323.

Option 9 under F1, PrdNonl, computes the predicted values for the mean and individual values of the dependent variable. Enter the dependent values in a list in the same order you entered them for the regression and indicate a confidence interval. The default confidence interval is .95 for a 95 percent confidence interval. The output for the year 2000 is:

"Value"	287.301	""
"Se$\bar{y}$/SeY"	4.19343	6.00139
"CI $\bar{y}$"	278.49	296.111
"CIY"	274.692	299.909

The first line of the matrix gives the value of the equation for the year 2000, 287.2 million people. The second line gives the standard errors of the mean and individual values of the dependent variable, in this case p. The third and fourth lines give the 95 percent confidence interval for the mean and individual values of p for the year 2000.

Option A under the F1 menu, MB(), is for "model building." If you added one or more variables to the previous regression, MB() will compute the F statistic and probability associated with adding the variable(s).

Option B under the F1 menu allows you to change the convergence criteria by setting the maximum number of iterations and subiterations and the criteria for the percentage change in successive sum-of-squares values. The values I have set are 30, 10, and 10^{-8}.

NOTE: NonLin() may be used for linear regression also, that is, where the equation is linear in its parameters. There are no restrictions on the independent variables. They may be any differentiable function. For instance, if x is an independent variable, it may occur in the equation as x^2 or x^5 ,etc., or SIN(x), LN(x), EXP(x), etc. When using NonLin() for linear regression, you may set the initial guesses of the parameters all equal to 1.

Logistic Regression

This program computes the logistic regression for binary dependent data using the logit, probit (normit), or complementary log-log link functions. Binary dependent data is in the form of 0s and 1s, where 1 signifies the occurrence of an event and 0 its nonoccurrence. The output is a regression equation that can be used to predict the probability of an event happening given a set of values for the independent variable(s).

The dataset may have a frequency variable or two variables denoting the results of a binomial experiment. The two variables are the number of successful events out of the total number of trials. If the dataset is from a binomial experiment, there is no dependent variable to enter; the program will create it.

The F2 menu is for computing the displaying the results of a logistic regression. Option 1, Logit(), is for entering the information and computing the regression. It sets up an input form. The first item is to select a dataset. The one presented is "ingot," which will be used in the example. Next, enter a list of the independent variables. The variables for this example are {h,s}. Next, you are prompted to enter the link function; use the logit link function. Next, you will be prompted for a frequency variable. The options are "No" for none, "Yes" for a frequency variable, and "Events/Trials" for a binomial experiment. Select "Events/Trials." The last two options are to change the maximum number of iterations and convergence criteria. Leave them at 30 and E^{-8} for now. Press Enter to continue.

Having selected "Events/Trials", you are now prompted to enter the events and trials variables. For this example they are e and t. Press Enter.

The program will now run and take several minutes to complete. A message will be displayed indicating whether the program was successful in estimating the regression.

(If you had selected “Yes” for a frequency variable, you would have been prompted to enter the name of the frequency variable. Then you would be prompted to enter the name of the dependent binary variable. If you had selected “No” for no frequency or binomial experiment variables, you would have been prompted to enter the name of the dependent binary variable.)

Option 2 under the F2 menu, FeqL, displays the fitted equation. For this example, it is:

.056771*s + .082031*h - 5.55917

Option 3, OutL, displays a matrix of the parameters, their values, standard errors, and the Wald chi-square statistics and probabilities.

"Parm"	"Value"	"StdErr"	"WChi2(1)"	"Prob(W)"
intercept	−5.55917	1.11969	24.6502	$6.87383E^{-7}$
h	.082031	.023734	11.9452	.000548
s	.056771	.331213	.029379	.863906

Option 4, LLRatio, displays the –2 log likelihood ratio that tests the significance of the covariates, that is, of the independent variables taken together. The output is:

"Chi2"	"DF"	"Prob"
11.6428	2	.002963

Chi2 is the chi-square statistic, DF the degrees of freedom, and Prob the probability of obtaining that value by chance.

Option 5, PrdLogt, computes the logit (or probit or complementary log-log) of “p,” where “p” is the probability of the event occurring given a set of values for the independent variables, the value of “p,” and the confidence interval of “p.” If h=7 and s=1, entering PrdLogt({7,1}, .95) produces:

"Value"	−4.92818	.007188
"C.I."	.001662	.030517

Reading across, the value of logit(p) is –4.92818; the value of p is .007188. The 95 percent confidence interval around p is .001662 to .030517.

Tools Menu

The options under the Tools menu are straightforward. Option 1 clears the home history screen.

Option 2, AddDS(), prompts you to add the name of a dataset to the list of datasets; this list is how you tell the programs what dataset to use. It also archives the dataset.

Option 3, DelDS(), deletes a dataset from memory and the dataset list when you select its name from dataset list.

Option 4, GetVars(), displays the variables in a dataset, in case you forgot what they were. Choose the dataset from the list presented.

Note: Once the programs have been run at least once, all programs and functions in the Nonlin folder may be archived. DO NOT archive anything else. The datasets will be archived by the AddDS() program.

I hope you find the programs useful and enjoyable. I know I had fun programming them. I have also programmed these routines for DERIVE 5. If you would like them, just drop me an email.

Any comments, suggestions, frustrations, with the programs? If so, again, just drop me an email.