

DOING INVERSE REGRESSION ESTIMATION VIA SAS/IML®

E. C. Drago and T. G. Filloon

Abstract

Inverse regression (also known as linear calibration) is used to determine when/where a linear regression line ($\pm$ SE) crosses a given horizontal line. We often need to quantify the potency of drug compounds using their dose response curves. An example of this would be to predict a compound's ED50 – what dose level a compound must be to reduce a negative reaction (such as pain, swelling, etc.) below a specific limit (e.g., 50% reduction from a control compound).

The point estimate of an ED50 is simple to calculate, but the standard error of the ED50 is usually needed also. Using IML, we have programmed the algebraic equations (inverting the confidence interval for the mean regression line) from Snedecor and Cochran to obtain standard error limits for our ED50 point estimate. One can visually see the estimate when plotting the data using PROC GPLOT, but SAS/IML® code is needed to define the ED50 $\pm$ SE exactly.

Introduction/Examples

Regression analysis is used in statistics to determine the relationship between variables; namely, Y and X. Usually the main purpose of regression analysis is to estimate the functional relationship between Y (a response variable) and X (a variable which is controlled), thus predicting Y from X. As a result, this is the type of regression mentioned in most textbooks.

However, inverse regression (predicting X from Y), although seldom mentioned in textbooks, is also important. Specifically, if one wants to determine what value of X_0 is needed to achieve a certain mean response (Y_0), then inverse regression is needed. While one can eyeball this value from a graph (see Fig. 1), the precise value is often required. In addition to the precise value, one would also like standard error limits around that value to judge the precision of the results.

In Figure 1, an analgesic assay was run on three different levels of a test compound. One wants to be able to determine what dose is needed to achieve a 50% reduction in pain (ED50). The 50% reduction line was calculated as 50% of a control group's mean response to pain. Then, using simple linear regression, the point at which the response curve crossed the 50% line was determined to be the X_0 point (the dose level needed to achieve this response).

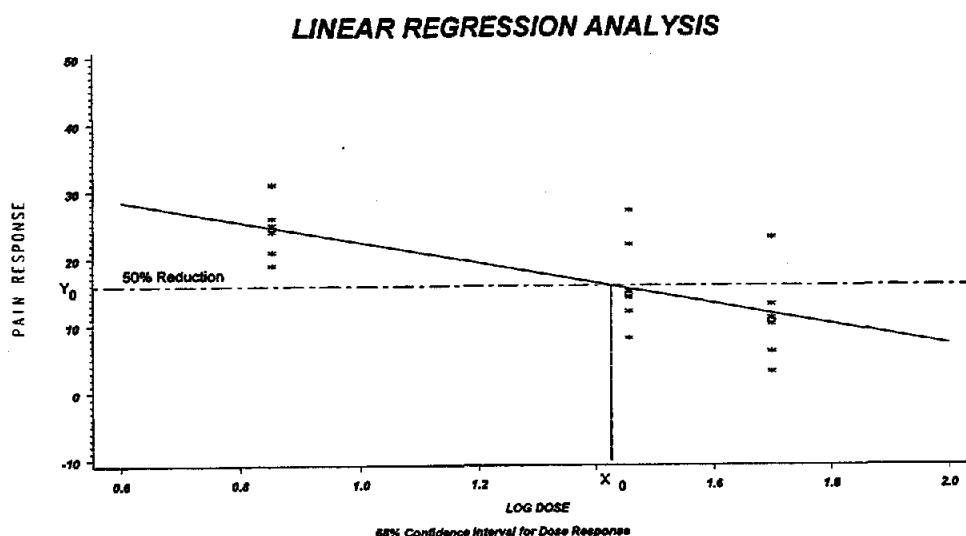
The estimation of X_0 is defined as:

$$Y = \bar{Y} + \beta(X - \bar{X}) + \varepsilon$$

$$\hat{X}_0 = \bar{X} + (Y_0 - \bar{Y}) / b_1$$

where b_1 is the estimated slope for the regression line.

FIGURE 1.



Different compound's dose response curves can be compared using the ED50 calculations $\pm$ their standard errors – much better than just looking at a plot of the data.

Using SAS/IML[®], we were able to do both – perform linear calibration (estimation of X from Y) and add 95% confidence limits (Figure 2) to the estimation (per Snedecor & Cochran). The calculation of the 95% confidence limit is

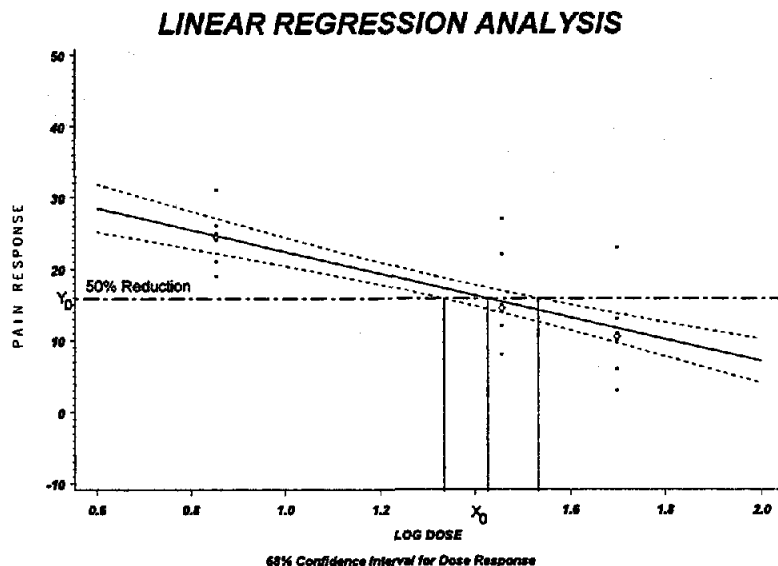
$$x = \frac{\hat{x} \pm (ts_{y,x} / b_1) \sqrt{1 / n(1 - c^2) + \hat{x}^2 / \sum x^2}}{1 - c^2}$$

where

$$c^2 = t^2 s_b^2 / b_1^2 = (1 / \sum x^2) \left(\frac{ts_{y,x}}{b_1} \right)^2$$

Note that in Snedecor and Cochran: $x = X - \bar{X}$ and $\hat{x} = \hat{X} - \bar{X}$.

FIGURE 2.

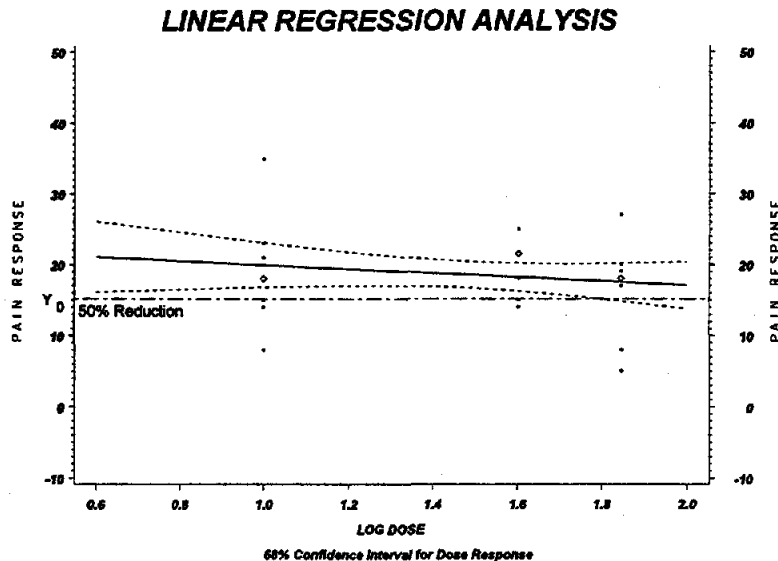


The results from the inverse regression analysis in mg/kg (milligrams/kilogram) are:

	ED50	STD. ERROR LIMITS
Log Dose	1.43	(1.33 - 1.54)
Actual Dose	26.7	(21.5 - 34.3)

One general warning about this procedure is that if the slope (b_1) is not sufficiently steep (i.e., t-statistic < 1), the denominator becomes negative, and finite confidence limits cannot be determined. In those cases, we report the ED50 (if available) with a disclaimer. Following is an example of that situation.

FIGURE 3.

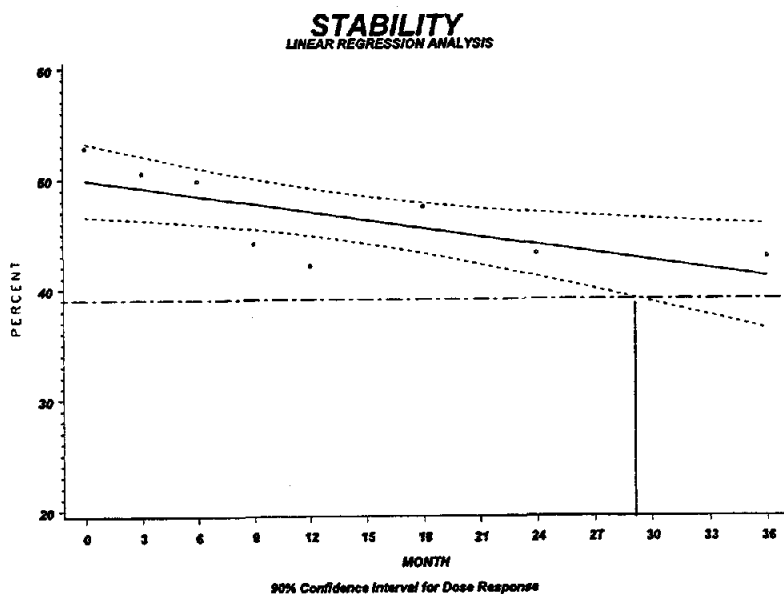


In this case, the resulting ED50 (Log Dose = 2.66; Actual Dose = 457.2) is not accurate since the standard error limits can not be calculated saying that the flatness of the dose-response curve does not allow reliable estimation of an ED50 value.

Additional Application

Another use for the Inverse Regression program is for stability calculations. In this example expiration dates are determined as the month at which the mean of six capsules is 39% dissolved. In this case alpha will be set to .95 and only the lower standard error limit will be used for determining the X_0 value.

FIGURE 4.



In this case the resulting shelf life estimate occurs at approximately 29 months, which is the time at which the lower confidence limit for the mean hits 39%.

In summary, inverse regression is a common estimation problem in a variety of settings and SAS IML code is available for doing such estimations simply.

Macro Listing

Following is a copy of the macro:

```
*****
*THIS MACRO CALCULATES ED50'S OR SHELF LIFE ESTIMATES BY INVERSE
*REGRESSION. IT ALSO CALCULATES THE STANDARD ERROR LIMITS FOR EACH
*ED50 CALCULATION
*****
* DSET = THE INPUT DATASET;
* XRESP= THE X VARIABLE (DOSE, TIME, ETC.);
* YRESP= THE RESPONSE VARIABLE;
* YZERO= THE POINT OF INTEREST (I.E., 50% OF CONTROL);
```

* ALPHA = ALPHA REQUIRED;

%MACRO RUNIML(DSET=,XRESP=,YRESP=,YZERO=,ALPH=);

PROC IML;

USE &DSET ;

READ ALL VAR{&XRESP} INTO X;

READ ALL VAR{&YRESP} INTO Y;

READ ALL VAR{&YZERO} INTO Z;

*DEFINE THE NUMBER OF OBSERVATIONS (N) AND THE NUMBER OF PARAMETERS (P)
AS THE NUMBER OF ROWS (Y) AND COLUMNS OF X. ADD A COLUMN OF ONES, FOR
THE INTERCEPT VARIABLE, TO THE X MATRIX. ;

ALPHA=&ALPH;

N=NROW(Y);

P=NCOL(X);

X=J(N,1,1)||X;

*COMPUTE THE NECESSARY MATRICES;

XPX=T(X)*X;

XPY=T(X)*Y;

YPY=T(Y)*Y;

YPX=T(Y)*X;

*COMPUTE THE INVERSE OF XPX AND THE VECTOR OF COEFFICIENT ESTIMATES BHAT;

XPXINV=INV(XPX);

BHAT=XPXINV*XPY;

*COMPUTE SSE, THE RESIDUAL SUM OF SQUARES, AND MSE, THE RESIDUAL
MEAN SQUARE (VARIANCE ESTIMATE).;

SSE=YPY-T(BHAT)*XPY;

DFE=N-P-1;

MSE=SSE / DFE;

ROOTMSE=SQRT(MSE); *SIGHAT;

PRINT "INVERSE REGRESSION STATISTICS";

PRINT " ";

PRINT XPXINV BHAT SSE DFE MSE ROOTMSE;

Y0 = Z[1];

XBAR=X[:,2];

YBAR=Y[:,1];

SLOPE=BHAT[2];

X0=XBAR + (Y0-YBAR)/SLOPE; *ED50 VALUE ON LOG SCALE;

SMX0=X0-XBAR;

SSX=T(X[:,2]-XBAR)*(X[:,2]-XBAR);

TSTAT=TINV(ALPHA,DFE); *TWO-SIDED TEST;

C2=(TSTAT*TSTAT) * (ROOTMSE*ROOTMSE) / (SLOPE*SLOPE) / SSX;

PRINT TSTAT C2 ROOTMSE SLOPE N SSX SMX0 Y0;

RCL=(TSTAT*ROOTMSE) / SLOPE * SQRT((1-C2)/N + (SMX0*SMX0)/SSX);

```

CL=ABS(RCL);
IF C2 >= 1 THEN DO;
  DOSEX0=10##X0;  *CONVERTS LOG DOSE TO DOSE;
  PRINT "SLOPE IS NOT SIGNIFICANT; I.E. TOO FLAT TO CALCULATE STANDARD ERRORS";
  PRINT " ";
  PRINT "IF LOG DOSES WERE ENTERED ACTUAL CALIBRATED LOG DOSE IS ",X0;
  PRINT " ";
  PRINT " AND ACTUAL CALIBRATED DOSE IS ",DOSEX0;
  PRINT " ";
  PRINT "IF OTHER DATA WERE ENTERED, THEN THE CALIBRATION RESULT IS ",X0;
END;

ELSE IF C2 < 1 THEN DO;
  UCL=(SMX0+CL)/(1-C2) + XBAR;
  LCL=(SMX0-CL)/(1-C2) + XBAR;
  DOSEUCL=10##UCL;
  DOSEX0=10##X0;
  DOSELCL=10##LCL;
PRINT ALPHA Y0 SLOPE SMX0 SSX TSTAT C2 CL;
PRINT " ";
RESET NONAME;
PRINT "IF LOG DOSES WERE ENTERED THEN:";
PRINT " ";
PRINT " ACTUAL CALIBRATED LOG DOSE (with s.e. limits) IS: ";
PRINT " " X0 "(" LCL ", " UCL ").";
PRINT " ";
PRINT " ACTUAL CALIBRATED DOSE (with s.e. limits) IS: ";
PRINT " " DOSEX0 "(" DOSELCL ", " DOSEUCL ").";
PRINT " ";
PRINT " ";
PRINT "IF OTHER DATA WERE ENTERED THEN:";
PRINT " THE LOWER CONFIDENCE LIMIT AT REQUESTED LEVEL IS: " LCL;
PRINT " THE UPPER CONFIDENCE LIMIT AT REQUESTED LEVEL IS: " UCL;
END;

QUIT;

%MEND RUNIML;

```

SAS is a registered trademark or trademark of SAS Institute Inc. in the USA and other countries.
 ® indicates USA registration.

Reference

Snedecor, G. W. and Cochran, W. G. Statistical Methods. 8th ed. Iowa State University Press, 1989.