

Probit Analysis of Correlated Data: Multiple Observations Over
Time at One Concentration

James E. Throne
USDA, ARS
U. S. Grain Marketing Research Laboratory
1515 College Ave.
Manhattan, KS 66502

David K. Weaver
USDA, ARS, IABBRL
1700 SW 23rd Drive
Gainesville, FL 32608

Victor Chew
USDA, ARS
412 Rolfs Hall
University of Florida
Gainesville, FL 32611

James E. Baker
USDA, ARS
U. S. Grain Marketing Research Laboratory
1515 College Ave.
Manhattan, KS 66502

PROBIT ANALYSIS is used to analyze data from bioassay experiments (Finney 1964). Typically in entomology, samples of insects are exposed to various concentrations of an insecticide to determine the concentration that will kill 50% of the insects within a given time span (e.g., Cilek & Knapp 1993). Effects of time on percentage kill at one dose (serial time-mortality data) may be of interest when: 1) material is limiting, as might occur when testing insecticides on field strains where few insects are available, or when testing an experimental pesticide that is available in limited quantities; or 2) when speed of kill is important, as might occur with a pest that lays all of its eggs within a few days (like a short-lived stored-product insect), or in quarantine treatments. Standard probit analysis techniques are not applicable to serial time-mortality data because observations made on the same group of organisms at different times are correlated.

Lampkin & Ogawa (1975) presented a method for calculating the slope and intercept of serial time-mortality data. Reports on results of probit-type analyses should include the standard errors of the slope, intercept, and lethal time (LT) or lethal dose (LD) values, and a test for goodness of fit (Anonymous 1992). Any tests comparing slopes, intercepts, or LT values should include confidence limits on the estimated statistics. Preisler & Robertson (1989) developed a method for analyzing bioassay data when response by the same group of organisms was determined at several times and at several concentrations of the insecticide (time-dose-mortality data). We present a method for analyzing correlated serial time-mortality data and a computer program that can be used to implement the method quickly and easily. The program allows the option of using the complementary log-log, logit, or probit transformation of proportion insects killed, and

allows the choice of using a logarithmic transformation of time. All statistics required for complete reporting of probit-type analyses are provided by the program, and methods for testing for equality of slopes and variances and for calculating relative potency of insecticides using information provided by the program are presented.

Methods

Notation. The following is a list of symbols that will be used. Matrices and vectors will be indicated by bold-faced upper and lower case letters, respectively. Elements in those matrices and vectors will be denoted by lower case letters. Vectors will be column vectors, unless transposed by means of a prime.

Φ = the derivative of p^* with respect to z

C = $m \times m$ matrix containing 0's and 1's that is used as a multiplier to accumulate the proportion insects dead at each observation time

b = regression coefficients (b_1 = intercept, b_2 = slope)

d_{cont} = cumulative number of insects that are dead at each observation time in the control

d_{trt} = cumulative number of insects that are dead at each observation time in the treatment

\hat{d}_r = estimated cumulative number of insects that are dead at each observation time based on the regression line

k_{cont} = number of insects that died in the control during a time interval

k_{obs} = number of insects killed in the treatment during a time interval corrected for control mortality

k_{ome} = observed minus estimated number of insects killed in the treatment during a time interval corrected for control mortality

k_{trt} = number of insects killed in the treatment during a

time interval

k_r = estimated number of insects killed in the treatment
during a time interval based on the regression line

m = number of times observations were made

n_{cont} = number of insects in the control

n_{trt} = number of insects treated

$n_{trtcorr}$ = number of insects treated corrected for number that
died in the control

p^* = cumulative proportion of insects that died during each
time interval in the treatment, corrected for control
mortality

p_{cont} = proportion of insects that died during each time
interval in the control

p_{corr} = p_{trt} corrected for control mortality

p_{trt} = proportion of insects that died during each time
interval in the treatment

st = Student's t

t = times at which observations were made

v_{11} = variance of $b(1)$

v_{12} = covariance of $b(1)$ and $b(2)$

v_{22} = variance of $b(2)$

X = the "design" matrix, the first column of which
consists of 1's and the second row contains the times
at which observations were made

z = the probit, logit, or complementary log-log
transformation of the cumulative proportion dead (p^*)
at each observation time

Probit Analysis. Probit analysis for correlated data differs
from standard probit analysis because in addition to their
variances, the covariances of the probits also must be estimated
to account for correlation between observations. The usual data

obtained in a serial time-mortality experiment are the times at which observations were made (t), the cumulative number of insects that are dead at each observation time in the treatment (d_{trt}) and in the control (d_{cont}), the number of insects treated (n_{trt}) and in the control (n_{cont}), and the number of times observations were made (m). From these data, the number of insects that are killed in the treatment (k_{trt}) or control (k_{cont}) during a time interval are calculated as:

$$k(i) = d(i) - d(i-1) , \quad (1)$$

where $i = 1, 2, \dots, m$. The proportion of insects that die during each time interval in the treatment (p_{trt}) or control (p_{cont}) is calculated as:

$$p(i) = \frac{k(i)}{n} . \quad (2)$$

The proportion of insects killed during each time interval in the treatments is corrected for control mortality as (Abbott 1925):

$$p_{corr}(i) = \frac{p_{trt}(i) - p_{cont}(i)}{1 - p_{cont}(i)} , \text{ if } > 0 \quad (3)$$

$$= \frac{0.5}{n_{trt} + 1} , \text{ if } \leq 0 \quad (4)$$

because the probit and logit transforms of $p(i)$ are undefined if $p(i) = 0$. The correction in (4) is suggested by McCullagh and Nelder (1989).

The cumulative proportion treated insects dead at each time interval, corrected for control mortality, is:

$$\mathbf{p}^* = \mathbf{C} \mathbf{p}_{corr} , \quad (5)$$

where \mathbf{C} is an $m \times m$ matrix of 0's and 1's that is used simply as a multiplier to accumulate the proportion insects dead at each observation time. For an experiment with three observation times, \mathbf{C} is:

$$\begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix}. \quad (6).$$

Each insect is a multinomial trial that can result in one of m or $(m + 1)$ mutually exclusive and exhaustive outcomes, depending on whether all \mathbf{n}_{trt} insects have or have not died at the last (m -th) observation time, respectively. The variance of $p_{trt}(i)$ is:

$$\text{var}(p_{trt}(i)) = \frac{p_{trt}(i) (1 - p_{trt}(i))}{n_{trt} - 1} \quad (7)$$

and the covariances of $p_{trt}(i)$ and $p_{trt}(j)$ are:

$$\text{cov}(p_{trt}(i), p_{trt}(j)) = \frac{-p_{trt}(i) p_{trt}(j)}{n_{trt} - 1} , \quad i \neq j \quad (8)$$

and $j = 1, 2, \dots, m$.

The variances of the corrected \mathbf{p} 's, \mathbf{p}_{corr} , are derived from the first-order Taylor series expansion as:

$$\begin{aligned}
\text{var}(p_{corr}(i)) &= \left(\frac{1}{1 - p_{cont}(i)} \right)^2 \frac{p_{trt}(i) (1 - p_{trt}(i))}{n_{trt} - 1} \\
&+ \left(\frac{1 - p_{trt}(i)}{(1 - p_{cont}(i))^2} \right)^2 \frac{p_{cont}(i) (1 - p_{cont}(i))}{n_{cont} - 1}
\end{aligned} \tag{9}$$

and the covariances of $p_{corr}(i)$ and $p_{corr}(j)$ are:

$$\begin{aligned}
\text{cov}(p_{corr}(i), p_{corr}(j)) &= \left[\frac{1}{1 - p_{cont}(i)} \frac{1}{1 - p_{cont}(j)} \right] \\
&\quad - \frac{p_{trt}(i) p_{trt}(j)}{n_{trt} - 1} \\
&+ \frac{-(1 - p_{trt}(i))}{(1 - p_{cont}(i))^2} \frac{-(1 - p_{trt}(j))}{(1 - p_{cont}(j))^2} \frac{-p_{cont}(i) p_{cont}(j)}{n_{cont} - 1}, \quad i \neq j
\end{aligned} \tag{10}$$

When there is no control mortality, the formulas for the variances and covariances of \mathbf{p}_{corr} reduce to the formulas for the variances and covariances of \mathbf{p}_{trt} . From the covariance propagation theorem, the estimate of the covariance matrix for \mathbf{p}^* is:

$$\text{cov}(\mathbf{p}^*) = \mathbf{C} \text{cov}(\mathbf{p}_{corr}) \mathbf{C}' . \tag{11}$$

If using a probit transformation, the probit (\mathbf{z}) of the cumulative proportion dead (\mathbf{p}^*) at each observation time is that

z -value of the standardized normal random variable such that the area under the probability density curve to the left of $z(i)$ is $p^*(i)$, or mathematically (Beyer 1987):

$$p^*(i) = \int_{-\infty}^{z(i)} \frac{e^{-\frac{u^2}{2}}}{\sqrt{2\pi}} du. \quad (12)$$

The derivative of $p^*(i)$ with respect to $z(i)$ is the probability density of a standardized random variable, $\phi(z(i))$, which is calculated as (Beyer 1987):

$$\phi(z(i)) = \frac{e^{-\frac{z(i)^2}{2}}}{\sqrt{2\pi}}. \quad (13)$$

$\phi(z(i))$ is the height of the standardized normal probability density curve at $z = z(i)$. From first-order Taylor series expansion, the variance of $z(i)$ is:

$$\text{var}(z(i)) = \frac{\text{var}(p^*(i))}{\phi(z(i))^2} \quad (14)$$

and the covariance of $z(i)$ and $z(j)$ is:

$$\text{cov}(z(i), z(j)) = \frac{\text{cov}(p^*(i), p^*(j))}{\phi(z(i)) \phi(z(j))}, \quad i \neq j. \quad (15)$$

If using the logit transformation:

$$z(i) = \log_e \left[\frac{p^*(i)}{1 - p^*(i)} \right]. \quad (16)$$

The variance of $z(i)$ is:

$$\text{var}(z(i)) = \frac{1}{n_{\text{trt}} p^*(i) (1 - p^*(i))} \quad (17)$$

and the covariance of $z(i)$ and $z(j)$ is:

$$\text{cov}(z(i), z(j)) = \frac{1}{p^*(i) (1 - p^*(i))} \frac{1}{p^*(j) (1 - p^*(j))} \text{cov}(p^*(i), p^*(j)). \quad (18)$$

If using the complementary log-log transformation,

$$z(i) = \log_e \left[-\log_e(1 - p^*(i)) \right]. \quad (19)$$

The variance of $z(i)$ is:

$$\text{var}(z(i)) = \frac{p^*(i)}{n_{\text{trt}} (1 - p^*(i)) [\log_e(1 - p^*(i))]^2} \quad (20)$$

and the covariance of $z(i)$ and $z(j)$ is:

$$\text{cov}(z(i), z(j)) = \frac{1}{1 - p^*(i)} \frac{1}{1 - p^*(j)} \frac{1}{-\log_e[1 - p^*(i)]}$$

$$\frac{1}{-\log_e[1 - p^*(j)]} \text{cov}(p^*(i), p^*(j)). \quad (21)$$

Having estimated \mathbf{z} and the covariance matrix of \mathbf{z} , generalized least squares methods (Neter et al. 1990) can be used to estimate parameters for the regression of \mathbf{z} on \mathbf{t} . An \mathbf{X} , or "design", matrix is formulated with the first column consisting of 1's and the second column containing the times at which observations were made. For example, for a design at which observations were made at 1, 2, and 3 h:

$$\mathbf{X} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \\ 1 & 3 \end{bmatrix}. \quad (22)$$

\mathbf{b} , where b_1 is the intercept and b_2 is the slope, is calculated as:

$$\mathbf{b} = [\mathbf{X}' \text{cov}(\mathbf{z})^{-1} \mathbf{X}]^{-1} [\mathbf{X}' \text{cov}(\mathbf{z})^{-1} \mathbf{z}']. \quad (23)$$

The covariance matrix of \mathbf{b} is:

$$\text{cov}(\mathbf{b}) = [\mathbf{X}' \text{cov}(\mathbf{z})^{-1} \mathbf{X}]^{-1}. \quad (24)$$

The covariance matrix of \mathbf{b} contains the variances of the slope and intercept, and the covariance of the slope and the intercept.

This method provides initial estimates for the slope and intercept, because the true value of $\text{cov}(\mathbf{z})$ is unknown and its estimate is used in (23) and (24). The final values of the slope and intercept are determined iteratively. In further iterations,

new $\mathbf{z}(i)$'s are calculated from the estimates of the slope and intercept as:

$$\mathbf{z}(i) = \mathbf{b}(1) + \mathbf{b}(2) \mathbf{t}(i). \quad (25)$$

If using the probit transformation, new $p^*(i)$'s are calculated from the $\mathbf{z}(i)$'s using (12). New $p(i)$'s are calculated as:

$$p(i) = p^*(i) - p^*(i-1), \quad (26)$$

with $p(1) = p^*(1)$. The covariance matrix of the new $p(i)$'s is then calculated using (7) and (8), the covariance matrix of the new $p^*(i)$'s is estimated as:

$$\text{cov}(\mathbf{p}^*) = \mathbf{C} \text{cov}(\mathbf{p}) \mathbf{C}', \quad (27)$$

new $\mathbf{z}(i)$'s are calculated using (13), a new covariance matrix for \mathbf{z} is calculated using (14) and (15), a new \mathbf{b} is calculated using (23), and a new covariance matrix for \mathbf{b} is calculated using (24). The new $b(1)$ and $b(2)$ are then compared with the $b(1)$ and $b(2)$ from the previous iteration to determine whether the differences are smaller than some predetermined convergence criterion. If the differences are not smaller, further iterations are run until convergence is achieved.

If using the logit transformation, the iterative procedure differs from that for the probit transformation as: new $p^*(i)$'s are calculated as:

$$p^*(i) = \frac{e^{\mathbf{z}(i)}}{1 + e^{\mathbf{z}(i)}} \quad (28)$$

and the covariance matrix of \mathbf{z} is estimated using (17) and (18).

If using the complementary log-log transformation, new $p^*(i)$'s are calculated as:

$$p^*(i) = 1 - e^{-e^{z(i)}} \quad (30)$$

and the covariance matrix of \mathbf{z} is estimated using (20) and (21).

Once convergence is achieved, the χ^2 goodness of fit test is used to determine how well the regression line fits the observed data. New $z(i)$'s are calculated using the final estimates of $b(1)$ and $b(2)$ and (25), new $p^*(i)$'s are calculated [using (12) for probits, (28) for logits, or (29) for the complementary log-log transformation], n_{trt} is corrected for control mortality as:

$$n_{trtcorr} = n_{trt} \left(1 - \frac{p(m)}{n_{cont}} \right), \quad (30)$$

the cumulative number that would be observed dead at each observation time based on the regression line is calculated as:

$$d_r(i) = p^*(i) n_{trtcorr}, \quad (31)$$

and the number that died during each time interval based on the regression line is calculated as:

$$k_r(i) = d_r(i) - d_r(i-1) \quad (32)$$

with $k(1) = d(1)$. The corrected number of insects observed dead during each time interval in the experiment is calculated as:

$$k_{obs}(i) = p_{corr}(i) n_{trtcorr}. \quad (33)$$

Thus, the observed minus predicted number of insects killed during

each time interval is:

$$k_{ome}(i) = k_{obs}(i) - k_r(i) . \quad (34)$$

The covariance matrix for k_{ome} is calculated by multiplying the covariance matrix for \mathbf{p} from the last iteration by n_{ttcorr}^2 . This is because the predicted number dead at each time interval is theoretical and has no variance, and the observed number dead at each time interval is n_{ttcorr} times $p_{corr}(i)$. The variance of a constant (n_{ttcorr}) times a variable (p_{corr}) is the constant squared times the variance of the variable. From standard multivariate normal distribution theory, χ^2 is calculated as:

$$\chi^2 = \mathbf{k}_{ome} \text{ cov}(\mathbf{k}_{ome})^{-1} \mathbf{k}_{ome} , \quad (35)$$

with $m - 2$ degrees of freedom. A significant χ^2 may indicate that the data are heterogeneous and should be corrected using a heterogeneity factor or that an alternative transformation would be more appropriate for the data (Finney 1964). Although the program automatically calculates a heterogeneity factor to compensate for significant χ^2 , the user should examine the graph of observed versus calculated $z(i)$'s to ensure that the selected transformation is appropriate for the data.

An LT value is then obtained by calculating \mathbf{z} [using (12) for probits, (16) for logits, or (19) for the complementary log-log transformation] of the proportion kill for which an LT value is desired, and calculating the LT value as:

$$LT = \frac{z - b(1)}{b(2)} . \quad (36)$$

If the χ^2 is significant, all variances used to calculate lethal

times and confidence limits on lethal times must be multiplied by the heterogeneity factor, which is χ^2 divided by $m - 2$ (Finney 1964). Fieller's theorem (Finney 1964) is then used to calculate confidence limits on lethal times. If the χ^2 is not significant, a standard z of 1.96 is used to calculate confidence limits. If the χ^2 is significant, a Student's t (st) with $m - 2$ df is used to calculate confidence limits. Letting v_{11} equal the variance of $b(1)$, v_{22} equal the variance of $b(2)$, and v_{12} equal the covariance of $b(1)$ and $b(2)$, then:

$$x_1 = L T + \frac{v_{12}}{v_{22}} , \quad (37)$$

$$x_2 = v_{11} + (2 L T v_{12}) + (L T^2 v_{22}) , \quad (38)$$

$$x_3 = v_{11} - \frac{v_{12}^2}{v_{22}} , \quad (39)$$

$$g = \frac{st^2 v_{22}}{(b(2))^2} , \quad (40)$$

$$y_1 = L T + \frac{g x_1}{1 - g} , \quad (41)$$

and

$$y_2 = \frac{st \sqrt{x_2 - (g x_3)}}{b(2) (1 - g)} . \quad (42)$$

The lower confidence limit for LT is then $y1 - y2$, and the upper confidence limit is $y1 + y2$. It is possible for the quantity within the square root in equation (42) to become negative, in which case the confidence limits are undefined.

The computer program. The probit analysis was programmed in Mathematica® language (Wolfram Research, Inc., Champaign, IL). The program is commented to facilitate understanding the structure, and the variable names used generally correspond to those used in the above description of the analysis.

The program is modified by the user in any text editor to include the data and the name of the output file. The program is then saved, brought into a Mathematica® session, and executed. Output is written to a file.

The user should provide a name for the output file (line 17 in program listing in Appendix I). Input to the program (lines 44-51) are the number of observations (notimes), the cumulative number of insects dead in the control (contdead) and treatment (expdead) at each observation time, the actual times that observations were made (times), and the number of insects tested in the control (ncont) and treatment (nexp). A logarithmic transformation of time may be executed by setting tranx to "yes" (line 50). The choice of making a complementary log-log, logit, or probit transformation on proportion insects killed is made on line 51.

The program automatically modifies the input data set. Observations at the beginning of the test when no insects were dead are deleted, except for the last zero observation. Observations when all insects are dead are deleted, due to the multinomial design. Observations when no insects died during a time interval are deleted, except as already discussed for those at the beginning of the test.

The program writes a graph of observed and calculated probits (or logits or complementary log-log transforms) to the screen. The output file (Appendix II) contains the input data for verification (lines 4-12); the slope and intercept of the regression line with their standard errors and covariance (lines 14-17); the observed minus expected number of insects dying in each time interval (lines 20-21); the χ^2 value, the probability of a greater χ^2 value ($p \leq 0.05$ is significant), and the degrees of freedom for the χ^2 (lines 22-24); the heterogeneity factor if χ^2 is significant (χ^2 is not significant in the example in Appendix II); the t or z used for calculating confidence limits (line 27); and the LT values with confidence limits corresponding to proportion kills of 0.1 to 0.9 by 0.1, 0.91 to 0.99 by 0.01, and 0.991 to 0.999 by 0.001 (lines 31-57). Any negative confidence limits or LT values are set to 0. Undefined confidence limits are shown as "undefined". If a transformation on time is used, an additional table will be printed showing LT values and confidence limits converted from transformed values back to real time.

Versions of Mathematica® are available for Macintosh and IBM-compatible computers. The program shown in Appendix I took 10 sec to run on a 120 Mhz Pentium computer using the OS/2 version of Mathematica®.

Testing for equality of slopes and intercepts. Bioassay data are often collected for two sexes of a species, for two separate species, or on the effects of two insecticides on one species, etc. The user might want to compare the two regression lines for equality of slopes and intercepts. This test is easily calculated using output from the program (Snedecor & Cochran 1967).

If the χ^2 for each sex (or other class variable) is not significant, the test statistic is calculated as:

$$\frac{\text{parameter}(1) - \text{parameter}(2)}{\sqrt{\text{var}(\text{parameter}(1)) + \text{var}(\text{parameter}(2))}} \quad (43).$$

The parameter in (43) is either the slope or the intercept from the output file, and the variance of the parameter in (43) is the square of the standard error of the parameter from the output file. If the test statistic exceeds the z for the desired probability level, then the parameter tested for the two regression lines is different.

If the χ^2 for either sex (or other class variable) is significant, the test statistic is calculated as above. However, the test statistic is compared to a modified Student's t calculated as:

$$t' = \frac{w_1 t_1 + w_2 t_2}{w_1 + w_2}, \quad (44)$$

where t_1 = Student's t with $n_1 - 2$ df, t_2 = Student's t with $n_2 - 2$ df, w_1 = the variance of parameter 1, and w_2 = the variance of parameter 2. A program called SLOPE was developed for comparing slopes and intercepts. The program can be used with output from any probit program.

Calculating relative potency. If θ_1 and θ_2 are times required to kill the same proportion of insects with two insecticides, then the ratio θ_1/θ_2 is the potency of insecticide 1 relative to insecticide 2 at a given mortality level (Robertson & Preisler 1992). If the slopes for the regression lines for the two insecticides are parallel, then relative potency will be the same

for any mortality level. If the slopes are not parallel, then relative potency must be calculated for each mortality level for which a comparison is desired.

Thetas corresponding to a given mortality level can be obtained from the output from the program. For the example in Appendix II, the theta corresponding to 0.90 mortality is 3.8397. The variance of θ_1/θ_2 is:

$$\text{var} \left(\frac{\theta_1}{\theta_2} \right) = \left(\frac{1}{\theta_2} \right)^2 \text{var}(\theta_1) - \left[\left(\frac{\theta_1}{\theta_2^2} \right)^2 \text{var}(\theta_2) \right]. \quad (45)$$

The variance of theta is:

$$\text{var}(\theta) = \frac{1}{\beta^2} [\text{var}(\alpha) + 2\theta \text{cov}(\alpha, \beta) + \theta^2 \text{var}(\beta)]. \quad (46)$$

The variances of alpha and beta, and the covariance of alpha and beta are obtained from the program output. Approximate 95% confidence limits for θ_1/θ_2 are:

$$\frac{\theta_1}{\theta_2} \pm 1.96 \sqrt{\text{var} \left(\frac{\theta_1}{\theta_2} \right)}. \quad (47)$$

If a logarithmic transformation on time is used, the relative potency (T_1/T_2) on the log scale becomes $10^{T_1}/10^{T_2}$ or $10^{(T_1 - T_2)}$ in the original (untransformed) units (where $T = \theta$). If:

$$\sigma = \sqrt{\text{var}(\theta_1 - \theta_2)} = \sqrt{\text{var}(\theta_1) + \text{var}(\theta_2)}, \quad (48)$$

then the approximate 95% confidence limits for the relative potency are:

$$10^{(\theta_1 - \theta_2 - 1.96\sigma)}, 10^{(\theta_1 - \theta_2 + 1.96\sigma)}, \quad (49)$$

A program called RELPOT is included on the diskette for calculating relative potency. The program can be used with output from any probit program.

References Cited

- Abbott, W. S.** 1925. A method of computing the effectiveness of an insecticide. J. Econ. Entomol. 18: 265-267.
- Anonymous.** 1992. Publishing with ESA. Entomological Society of America, Lanham, MD.
- Baker, J.E., D.K. Weaver, J.E. Throne, and J.L. Zettler.** Resistance to protectant insecticides in two field strains of the parasitoid *Bracon hebetor* (Hymenoptera: Braconidae). J. Econ. Entomol. 88: 512-519.
- Beyer, W. H.** 1987. CRC standard mathematical tables, 28th edition. CRC Press, Inc., Boca Raton, FL.
- Cilek, J. E. & F. W. Knapp.** 1993. Enhanced diazinon susceptibility in pyrethroid-resistant horn flies (Diptera: Muscidae): potential for insecticide resistance management. J. Econ. Entomol. 86: 1303-1307.
- Finney, D. J.** 1964. Probit analysis: a statistical treatment of the sigmoid response curve. Cambridge Univ. Press, London, England.
- Lampkin, H. & J. Ogawa.** 1975. Estimation of distribution parameters in time mortality trials: an example of time mortality analysis. Can. J. Statistics 4C: 65-93.
- McCullagh, P. & J. A. Nelder.** 1989. Generalized linear models, second edition. Chapman & Hall, New York, NY.
- Neter, J., W. Wasserman & M. H. Kutner.** 1990. Applied linear statistical models: regression, analysis of variance, and experimental designs, third edition. Irwin, Homewood, IL.
- Preisler, H. K. & J. L. Robertson.** 1989. Analysis of time-dose-mortality data. J. Econ. Entomol. 82: 1534-1542.
- Robertson, J. L. & H. K. Preisler.** 1992. Pesticide bioassays with arthropods. CRC Press, Boca Raton, FL.

Snedecor, G. W. & W. G. Cochran. 1967. Statistical methods,
sixth edition. Iowa State Univ. Press, Ames, IA.

APPENDIX I - PROGRAM LISTING

```

1      (* PROBIT - This program is used to analyze serial time-mortality data
2      from bioassay studies. Follow the instructions below for modifying the
3      program to include your data. I would suggest that you save the
4      original version of the program and make changes to a copy of the
5      program. *)

6      (* This program was developed by:

7          Jim Throne
8          USDA, ARS, USGMRL
9          1515 College Avenue
10         Manhattan, KS 66502
11         (913) 776-7624
12         throne@crunch.usgmrl.ksu.edu

13         Please notify me of problems or suggestions.  5 Oct 95 *)

14     Off[]

15     (* You may change the name of the output file by changing the name in
16     quotation marks on the next line. *)

17     out=OpenWrite["outprob",FormatType->OutputForm,PageWidth->76]

18     (* You may type in headings for the output file between the
19     quotation marks on the next program lines. Total line length
20     should not exceed 72 characters. *)

21     WriteString[out,"probit transformation"]
22     WriteString[out," \n"]
23     WriteString[out,"Anisopteromalus calandrae"]
24     WriteString[out," \n"]
25     WriteString[out,"malathion data"]
26     WriteString[out," \n\n"]

27     (* The eight lines after the following descriptions are where you
28     enter your data.

29         notimes    = number of time intervals for which you will enter data
30         contdead   = the observed cumulative number of insects dead in the
31                     control at each time interval, enclosed in brackets
32         expdead    = the observed cumulative number of insects dead in the
33                     treatment at each time interval, enclosed in brackets
34         times      = the times at which observations were made, enclosed in
35                     brackets
36         ncont      = the initial number of insects in the control
37         nexpt      = the initial number of insects in the treatment
38         tranx      = decision to use log10(time) in analysis (yes or no in
39                     quotation marks)
40         tranyp     = decision to use "probit", "logit", or "gompertz"
41                     transformation on y. Enclose choice in quotation marks.
42                     Note that gompertz is being used to represent the
43                     complementary log-log transformation *)

44     notimes=5
45     contdead={0,0,0,0,0}
46     expdead={2,6,20,23,25}
47     times={1,2,3,4,5}
48     ncont=25

```

```

49      nex=25
50      tranx="no"
51      trany="probit"

52      (* The user should not make any further changes to the program. *)

53      WriteString[out,"Input data: \n\n"]
54      WriteString[out,"Number of time intervals = "]
55      Write[out,notimes]
56      WriteString[out,"Number dead in controls = "]
57      Write[out,contdead]
58      WriteString[out,"Number dead in treatments = "]
59      Write[out,expdead]
60      WriteString[out,"Time intervals = "]
61      Write[out,times]
62      WriteString[out,"Number of controls = "]
63      Write[out,ncont]
64      WriteString[out,"Number treated = "]
65      Write[out,nex]
66      WriteString[out,"Logarithmic transformation of time = "]
67      Write[out,tranx]
68      WriteString[out,"Transformation = "]
69      Write[out,trany]
70      WriteString[out," \n\n"]

71      (* Delete data for time periods when no mortality occurred or in
72      which all insects were dead. Apply log10 transformation to time. *)

73      If[tranx=="yes",times=Log[10,times] //N]
74      input=Table[0,{i,notimes},{j,3}]
75      input[[1,1]]=times[[1]]
76      input[[1,2]]=contdead[[1]]
77      input[[1,3]]=expdead[[1]]
78      inc=1
79      Do[If[expdead[[i]]==expdead[[i-1]],Goto[elim],Goto[add]];
80      Label[add];
81      If[expdead[[i]]==nex,Goto[elim],inc=inc+1];
82      input[[inc,1]]=times[[i]];
83      input[[inc,2]]=contdead[[i]];
84      input[[inc,3]]=expdead[[i]];
85      Goto[end1];
86      Label[elim];
87      If[expdead[[i]]==0,input[[1,1]]=times[[i]]];
88      If[expdead[[i]]==0,input[[1,3]]=contdead[[i]]];
89      Label[end1];
90      ,{i,2,notimes}]

91      (* Set up vectors containing the time intervals (t), the number dead
92      at each time interval in the control (dcont), the number dead at
93      each time interval in the treatment (dexp), and the number of time
94      intervals that had valid data (m) *)

95      t=Table[input[[i,1]],{i,inc}]
96      dcont=Table[input[[i,2]],{i,inc}]
97      dexp=Table[input[[i,3]],{i,inc}]
98      m=inc
99      pdead=dcont[[m]]/ncont

100      (* Calculate the number of insects that die during each time
101      interval (kexp or kcont), the proportion that die during each time
102      interval (pexp or pcont), and correct for 0 dead. *)

```

```

103 kexp=Table[If[i==1,dexp[[1]],dexp[[i]]-dexp[[i-1]]],{i,m}]
104 pexpinit=kexp/nexp //N
105 pexpcorr=Table[If[pexpinit[[i]]==0,0.5/(nexp+1),pexpinit[[i]]],{i,m}]
106 kcont=Table[If[i==1,dcont[[1]],dcont[[i]]-dcont[[i-1]]],{i,m}]
107 pcont=Table[kcont[[i]]/ncont,{i,m}]

108 (* Calculate the covariance matrix of p. *)

109 covp=Table[If[i==j,(pexpcorr[[i]]*(1-pexpcorr[[i]]))/
110 (nexp-1),-(pexpcorr[[i]]*pexpcorr[[j]])/(nexp-1)],{i,m},{j,m}]

111 (* Apply Abbott's formula to correct for control mortality.
112 Calculate covariance matrix for corrected p's. *)

113 pexp2=Table[(pexpinit[[i]]-pcont[[i]])/(1-pcont[[i]]),{i,m}]
114 pexp=Table[If[pexp2[[i]]<=0,0.5/(nexp+1),pexp2[[i]]],{i,m}]

115 covpcorr=Table[If[i==j,((1/(1-pcont[[i]]))^2)*((1-pexp[[i]])*
116 pexp[[i]])/(nexp-1))+((1-pexp[[i]])/((1-pcont[[i]]^2)^2)*
117 ((1-pcont[[i]])*pcont[[i]])/(ncont-1)),(1/(1-pcont[[i]]))^2*
118 (1/(1-pcont[[j]]))^2*(-(pexp[[i]]*pexp[[j]])/(nexp-1))+
119 (-1-pexp[[i]])/((1-pcont[[i]]^2))*(-(1-pexp[[j]])/
120 ((1-pcont[[j]]^2))*(-(pcont[[i]]*pcont[[j]])/(ncont-1))],
121 {i,m},{j,m}]

122 (* Calculate the c and x matrices. *)

123 c=Table[If[j<=i,1,0],{i,m},{j,m}]
124 x=Table[If[j==1,1,t[[i]]],{i,m},{j,2}]

125 (* Calculate the cumulative proportion killed during each interval
126 (pstarobs) and the covariance matrix of pstarobs. *)

127 pstarobs=c.pexp
128 covpstar=c.covpcorr.Transpose[c]

129 (* If using probit transform, calculate the probits (z) of the
130 cumulative proportion killed during each interval, the probability
131 density of the probits (phiz), and the covariance matrix of the
132 probits (covz). If using logits calculate the logits (z) and the
133 covariance matrix of the logits (covz). If using Gompertz
134 transformation, calculate the transforms (z) and the covariance
135 matrix of the transforms. *)

136 Needs["Statistics`ContinuousDistributions`"]
137 Do[If[trany != "probit",Goto[other],ndist=NormalDistribution[0,1]];
138 y=Map[Quantile[ndist,#]&,pstarobs];
139 z=Table[y[[i]],{i,m}];
140 phiz=(E^(-(z^2)/2))/Sqrt[2*Pi] //N;
141 covz=Table[If[i==j,1/(nexp*pstarobs[[i]]*phiz[[i]]^2,
142 covpstar[[i,j]]/(phiz[[i]]*phiz[[j]])],{i,m},{j,m}];
143 Goto[endtrans];
144 Label[other];
145 If[trany != "logit",Goto[gomp],z=Table[Log[pstarobs[[i]]/
146 (1-pstarobs[[i]])],{i,m}];
147 covz=Table[If[i==j,1/(nexp*pstarobs[[i]]*(1-pstarobs[[i]])),
148 (1/(pstarobs[[i]]*(1-pstarobs[[i]])))*(1/(pstarobs[[j]]*
149 (1-pstarobs[[j]])))*covpstar[[i,j]]],{i,m},{j,m}];
150 Goto[endtrans];
151 Label[gomp];
152 z=Table[Log[-Log[1-pstarobs[[i]]]],{i,m}];
153 covz=Table[If[i==j,pstarobs[[i]]/(nexp*(1-pstarobs[[i]]))*

```



```

154      ((Log[1-pstarobs[[i]]]^2)), (1/(1-pstarobs[[i]]))*
155      (1/-Log[1-pstarobs[[i]]])*(1/(1-pstarobs[[j]]))*
156      (1/-Log[1-pstarobs[[j]]])*covpstar[[i,j]],{i,m},{j,m}];
157      Label[endtrans];,{i2,1}]

158      (* Calculate the (X'X)-1 (binit1) and X'Y (binit2) matrices.
159      Calculate the slope and intercept (the matrix newb) as binit1 times
160      binit2. *)

161      binit1=Inverse[Transpose[x].Inverse[covz].x]
162      binit2=(Transpose[x].Inverse[covz].z)
163      newb=binit1.binit2
164      Needs["Graphics`Graphics`"]
165      g1=ListPlot[Table[{t[[i]],z[[i]]},{i,m}],DisplayFunction
166      ->Identity]

167      (* Begin iterations to converge on best values of slope and
168      intercept. Calculate new probits (or logits or Gompertz's
169      transforms) using slope and intercept (y). Calculate new cumulative
170      proportion dead at each interval (pstar) from probits (or logits or
171      Gompertz's transforms). Calculate new proportion dying during each
172      interval (p) from pstar. Calculate covariance matrix of p (covp)
173      and covariance matrix of pstar (covpstar). Calculate new
174      probability density of probits (phiz), if using probit
175      transformation, and covariance matrix of the probits (or logits or
176      Gompertz's transform) (covz). Obtain new estimates of slope and
177      intercept (newb), and keep iterating until convergence criterion
178      met. *)

179      Do[
180      b=newb;
181      y=b[[1]]+(b[[2]]*t);
182      z=Table[y[[i]],{i,m}];
183      If[trany != "probit",Goto[other2],pstar=Map[CDF[ndist,#]&, z]];
184      p=Table[If[i==1,pstar[[1]],pstar[[i]]-pstar[[i-1]]],{i,m}];
185      covp=Table[If[i==j,(p[[i]]*(1-p[[i]]))/(nexp-1),
186      -(p[[i]]*p[[j]])/(nexp-1)],{i,m},{j,m}];
187      phiz=(E^(-(z^2)/2))/Sqrt[2*Pi] //N;
188      covpstar=c.covp.Transpose[c];
189      covz=Table[If[i==j,covpstar[[i,i]]/phiz[[i]]^2,
190      covpstar[[i,j]]/(phiz[[i]]*phiz[[j]])],{i,m},{j,m}];
191      Goto[endtrans2];
192      Label[other2];
193      If[trany != "logit",Goto[gomp2],pstar=Exp[z]/(1+Exp[z])];
194      p=Table[If[i==1,pstar[[1]],pstar[[i]]-pstar[[i-1]]],{i,m}];
195      covp=Table[If[i==j,(p[[i]]*(1-p[[i]]))/(nexp-1),
196      -(p[[i]]*p[[j]])/(nexp-1)],{i,m},{j,m}];
197      covpstar=c.covp.Transpose[c];
198      covz=Table[If[i==j,1/(nexp*pstar[[i]]*(1-pstar[[i]])),
199      (1/(pstar[[i]]*(1-pstar[[i]])))*(1/(pstar[[j]]*
200      (1-pstar[[j]])))*covpstar[[i,j]],{i,m},{j,m}];
201      Goto[endtrans2];
202      Label[gomp2];
203      pstar=1-Exp[-Exp[z]];
204      p=Table[If[i==1,pstar[[1]],pstar[[i]]-pstar[[i-1]]],{i,m}];
205      covp=Table[If[i==j,(p[[i]]*(1-p[[i]]))/(nexp-1),
206      -(p[[i]]*p[[j]])/(nexp-1)],{i,m},{j,m}];
207      covpstar=c.covp.Transpose[c];
208      covz=Table[If[i==j,pstar[[i]]/(nexp*(1-pstar[[i]]))*
209      ((Log[1-pstar[[i]]]^2)), (1/(1-pstar[[i]]))*
210      (1/-Log[1-pstar[[i]]])*(1/(1-pstar[[j]]))*
211      (1/-Log[1-pstar[[j]]])*covpstar[[i,j]],{i,m},{j,m}];

```

```

212 Label[endtrans2];
213 binit1=Inverse[Transpose[x].Inverse[covz].x];
214 binit2=(Transpose[x].Inverse[covz].z);
215 newb=binit1.binit2;
216 If[Abs[newb[[1]]-b[[1]]]<.00000001 &&
217     Abs[newb[[2]]-b[[2]]]<.00000001,Break[]];,{i,50}}

218 (* Calculate the standard errors of the slope (seslope) and intercept
219 (seint) from the covariance matrix of b (binit1). *)

220 seint=Sqrt[binit1[[1,1]]]
221 seslope=Sqrt[binit1[[2,2]]]

222 (* Write output about slope and intercept to file. *)

223 WriteString[out,"Intercept = "]
224 Write[out,newb[[1]]]
225 WriteString[out,"\nSE Intercept = "]
226 Write[out,seint]
227 WriteString[out,"\nSlope = "]
228 Write[out,newb[[2]]]
229 WriteString[out,"\nSE Slope = "]
230 Write[out,seslope]
231 WriteString[out,"\nCovariance of Slope & Intercept = "]
232 Write[out,binit1[[1,2]]]
233 WriteString[out," \n"]

234 (* Calculate Chi-square to test for goodness of fit of regression
235 line to observed data. Probit, logit, or Gompertz's transform
236 (zpred) is calculated from final values of slope and intercept. A
237 new p (ppred) is calculated from the probit (or logit or Gompertz's
238 transform). The cumulative number dead at each time interval (dpred) is
239 calculated from the p and ncorr (the number of insects tested minus
240 the number that died in the control). The noncumulative number
241 dying during each time interval (kpred) is calculated from dpred.
242 The observed number dead at each time interval (kpred) is calculated
243 from the cumulative proportion dead at each time interval (pexp)
244 times the initial number of treated insects (ncorr). The observed
245 minus expected number of insects dead is calculated as dobs - dexp.
246 The covariance matrix for the observed minus expected number dead
247 (covobsme) is calculated as ncorr squared times the covariance
248 matrix of p (covp). The Chi-square value (chisq) is then
249 calculated, and the probability of a larger Chi-square (prchisq)
250 with n-2 degrees of freedom (dfchisq) is determined. *)

251 zpred=newb[[1]]+(newb[[2]]*t)
252 Do[
253 If[trany != "probit",Goto[other3],ppred=Map[CDF[ndist,#]&,zpred]];
254 Goto[endtrans3];
255 Label[other3];
256 If[trany != "logit",Goto[gomp3],ppred=Exp[zpred]/(1+Exp[zpred])];
257 Goto[endtrans3];
258 Label[gomp3];
259 ppred=1-Exp[-Exp[zpred]];
260 Label[endtrans3];
261 ,{i3,1}]
262 ncorr=nexp*(1-pdeadc)
263 dpred=ppred*ncorr
264 kpred=Table[If[i==1,dpred[[1]],dpred[[i]]-dpred[[i-1]]],{i,m}]
265 kobs=pexp*ncorr
266 obsmexp=kobs-kpred
267 covobsme=(ncorr^2)*covp

```

```

268      chisq=obsmexp.Inverse[covobsme].obsmexp
269      chidist=ChiSquareDistribution[m-2]
270      prchisq=2*PDF[chidist,chisq]
271      dfchisq=m-2

272      (* If the Chi-square is significant, correct all variances used to
273      calculate confidence limits on LT levels by multiplying each variance
274      by h. *)

275      h=If[prchisq<.05,chisq/(m-1),1]

276      (* Chi-square information is output to file. *)

277      WriteString[out,"Obs - Exp number of insects dying during each time
278      interval = \n"]
279      Write[out,obsmexp]
280      WriteString[out,"\nChiSquare = "]
281      Write[out,chisq]
282      WriteString[out,"\nProbability of a greater ChiSquare = "]
283      Write[out,FortranForm[prchisq]]
284      WriteString[out,"\nDegrees of freedom for ChiSquare = "]
285      Write[out,dfchisq]
286      WriteString[out," \n"]
287      If[prchisq>0.05,Null,WriteString[out,"Heterogeneity factor = "];
288      Write[out,h]]
289      WriteString[out," \n"]
290      g2=Plot[Table[newb[[1]]+(newb[[2]]*i)],{i,input[[1,1]],t[[m]]},
291      DisplayFunction->Identity]

292      Do[
293      If[trany != "probit",Goto[other4]];
294      If[tranx == "yes",Goto[logyes1],Show[g1,g2,Frame->True,PlotLabel->
295      "Observed and Calculated Probits vs Time \n          - Hit any key when
296      done -",DisplayFunction->$DisplayFunction,AxesOrigin->{0,0}]];
297      Goto[endtrans4];
298      Label[logyes1];
299      Show[g1,g2,Frame->True,PlotLabel->
300      "Observed and Calculated Probits vs Log Time \n          - Hit any key
301      when done -",DisplayFunction->$DisplayFunction,AxesOrigin->{0,0}]];
302      Goto[endtrans4];
303      Label[other4];
304      If[trany != "logit",Goto[gomp4]];
305      If[tranx == "yes",Goto[logyes2],Show[g1,g2,Frame->True,PlotLabel->
306      "Observed and Calculated Logits vs Time \n          - Hit any key when
307      done -",DisplayFunction->$DisplayFunction,AxesOrigin->{0,0}]];
308      Goto[endtrans4];
309      Label[logyes2];
310      Show[g1,g2,Frame->True,PlotLabel->
311      "Observed and Calculated Logits vs Log Time \n          - Hit any key
312      when done -",DisplayFunction->$DisplayFunction,AxesOrigin->{0,0}]];
313      Goto[endtrans4];
314      Label[gomp4];
315      If[tranx == "yes",Goto[logyes3],Show[g1,g2,Frame->True,PlotLabel->
316      "Observed and Calculated CLL transforms vs Time \n          - Hit any key
317      when done -", DisplayFunction->$DisplayFunction,AxesOrigin->{0,0}]];
318      Goto[endtrans4];
319      Label[logyes3];
320      Show[g1,g2,Frame->True,PlotLabel->
321      "Observed and Calculated CLL transforms vs Log Time \n          - Hit any
322      key when done -", DisplayFunction->$DisplayFunction,AxesOrigin->{0,0}]];
323      Label[endtrans4];
324      ,{i4,1}]];

```

```

325      (* Use Fieller's theorem to calculate confidence limits on LT levels.
326      *)

327      tdist=StudentTDistribution[m-2]
328      t2=If[prchisq>.05,1.96,Quantile[tdist,.975]]
329      If[prchisq>.05,WriteString[out,"z for calculating confidence limits =
330      "],WriteString[out,"t for calculating confidence limits = "]]
331      Write[out,t2]
332      WriteString[out," \n"]

333      a=newb[[1]]
334      b=newb[[2]]
335      v11=binit1[[1,1]]*h
336      v22=binit1[[2,2]]*h
337      v12=binit1[[1,2]]*h

338      outprob1=Table[i,{i,.1,.9,.1}]
339      outprob2=Table[i,{i,.91,.99,.01}]
340      outprob3=Table[i,{i,.991,.999,.001}]

341      Do[
342      If[trany != "probit",Goto[other5],lt1=Map[Quantile[ndist,#]&,
343      outprob1]];
344      lt2=Map[Quantile[ndist,#]&,outprob2];
345      lt3=Map[Quantile[ndist,#]&,outprob3];
346      Goto[endtrans5];
347      Label[other5];
348      If[trany != "logit",Goto[gomp5],lt1=Table[Log[outprob1[[i]]/(
349      (1-outprob1[[i]])],{i,9}]];
350      lt2=Table[Log[outprob2[[i]]/(1-outprob2[[i]])],{i,9}];
351      lt3=Table[Log[outprob3[[i]]/(1-outprob3[[i]])],{i,9}];
352      Goto[endtrans5];
353      Label[gomp5];
354      lt1=Table[Log[-Log[1-outprob1[[i]]]],{i,9}];
355      lt2=Table[Log[-Log[1-outprob2[[i]]]],{i,9}];
356      lt3=Table[Log[-Log[1-outprob3[[i]]]],{i,9}];
357      Label[endtrans5];
358      ,{i5,1}]

359      m1=(lt1-a)/b
360      m2=(lt2-a)/b
361      m3=(lt3-a)/b
362      x11=m1+(v12/v22)
363      x12=m2+(v12/v22)
364      x13=m3+(v12/v22)
365      x21=v11+2*m1*v12+(m1^2)*v22
366      x22=v11+2*m2*v12+(m2^2)*v22
367      x23=v11+2*m3*v12+(m3^2)*v22
368      x3=v11-((v12^2)/v22)
369      g=(t2^2)*v22/(b^2)
370      y11=m1+(g*x11/(1-g))
371      y12=m2+(g*x12/(1-g))
372      y13=m3+(g*x13/(1-g))
373      y21=t2*Sqrt[x21-(g*x3)]/(b*(1-g))
374      y22=t2*Sqrt[x22-(g*x3)]/(b*(1-g))
375      y23=t2*Sqrt[x23-(g*x3)]/(b*(1-g))
376      l1=y11-y21
377      l2=y12-y22
378      l3=y13-y23
379      u1=y11+y21
380      u2=y12+y22
381      u3=y13+y23

```

```

382 Do[If[g*x3<x21[[i]] && (1-g)>0,Goto[Skip1]];
383 l1[[i]]="Undefined";
384 u1[[i]]="Undefined";
385 Label[Skip1},{i,9}]
386 Do[If[g*x3<x22[[i]] && (1-g)>0,Goto[Skip2]];
387 l2[[i]]="Undefined";
388 u2[[i]]="Undefined";
389 Label[Skip2},{i,9}]
390 Do[If[g*x3<x23[[i]] && (1-g)>0,Goto[Skip3]];
391 l3[[i]]="Undefined";
392 u3[[i]]="Undefined";
393 Label[Skip3},{i,9}]
394 Do[If[m1[[i]]<0,m1[[i]]=0},{i,9}]
395 Do[If[m2[[i]]<0,m2[[i]]=0},{i,9}]
396 Do[If[m3[[i]]<0,m3[[i]]=0},{i,9}]
397 Do[If[l1[[i]]<0,l1[[i]]=0},{i,9}]
398 Do[If[l2[[i]]<0,l2[[i]]=0},{i,9}]
399 Do[If[l3[[i]]<0,l3[[i]]=0},{i,9}]
400 Do[If[u1[[i]]<0,u1[[i]]=0},{i,9}]
401 Do[If[u2[[i]]<0,u2[[i]]=0},{i,9}]
402 Do[If[u3[[i]]<0,u3[[i]]=0},{i,9}]

403 (* Output results of Fieller's theorem calculations to file. *)

404 If[tranx=="yes",WriteString[out,"Log10 transform was used; following
405 data output as logarithms\n"]]
406 WriteString[out,"\n Prop. Lower Time Upper\n"]
407 WriteString[out," kill limit limit\n\n"]
408 Do[Write[out,PaddedForm[SequenceForm[outprob1[[i]]," ",l1[[i]],
409 " ",m1[[i]]," ",u1[[i]]],{7,4}]],{i,9}]
410 Do[Write[out,PaddedForm[SequenceForm[outprob2[[i]]," ",l2[[i]],
411 " ",m2[[i]]," ",u2[[i]]],{7,4}]],{i,9}]
412 Do[Write[out,PaddedForm[SequenceForm[outprob3[[i]]," ",l3[[i]],
413 " ",m3[[i]]," ",u3[[i]]],{7,4}]],{i,9}]

414 (* If using log transform, write out LT values in real time. *)

415 Do[
416 If[tranx=="no",Goto[end]];
417 outl1=Table[If[l1[[i]]=="Undefined","Undefined",10^l1[[i]]],{i,9}];
418 outl2=Table[If[l2[[i]]=="Undefined","Undefined",10^l2[[i]]],{i,9}];
419 outl3=Table[If[l3[[i]]=="Undefined","Undefined",10^l3[[i]]],{i,9}];
420 outm1=Table[10^m1[[i]],{i,9}];
421 outm2=Table[10^m2[[i]],{i,9}];
422 outm3=Table[10^m3[[i]],{i,9}];
423 outu1=Table[If[u1[[i]]=="Undefined","Undefined",10^u1[[i]]],{i,9}];
424 outu2=Table[If[u2[[i]]=="Undefined","Undefined",10^u2[[i]]],{i,9}];
425 outu3=Table[If[u3[[i]]=="Undefined","Undefined",10^u3[[i]]],{i,9}];

426 WriteString[out,"\n\nOutput converted to original time units\n\n"];
427 WriteString[out," Prop. Lower Time Upper\n"];
428 WriteString[out," kill limit limit\n\n"];
429 Do[Write[out,PaddedForm[SequenceForm[outprob1[[i2]]," ",outl1[[i2]]
430 " ",outm1[[i2]]," ",outu1[[i2]]],{7,4}]],{i2,9}];
431 Do[Write[out,PaddedForm[SequenceForm[outprob2[[i2]]," ",outl2[[i2]]
432 " ",outm2[[i2]]," ",outu2[[i2]]],{7,4}]],{i2,9}];
433 Do[Write[out,PaddedForm[SequenceForm[outprob3[[i2]]," ",outl3[[i2]]
434 " ",outm3[[i2]]," ",outu3[[i2]]],{7,4}]],{i2,9}];
435 Label[end];
436 ,{i3,1}]

```

438

Close[out]

APPENDIX II - PROGRAM OUTPUT

```

1      probit transformation
2      Anisopteromalus calandrae
3      malathion data

4      Input data:

5      Number of time intervals = 5
6      Number dead in controls = {0, 0, 0, 0, 0}
7      Number dead in treatments = {2, 6, 20, 23, 25}
8      Time intervals = {1, 2, 3, 4, 5}
9      Number of controls = 25
10     Number treated = 25
11     Logarithmic transformation of time = no
12     Transformation = probit
13

14     Intercept = -2.29424
15
16     SE Intercept = 0.449058
17
18     Slope = 0.931281
19
20     SE Slope = 0.159994
21
22     Covariance of Slope & Intercept = -0.0632318
23
24     Obs - Exp number of insects dying during each time interval =
25     {-0.161179, -2.16348, 5.04163, -2.81116}
26
27     ChiSquare = 4.77441
28
29     Probability of a greater ChiSquare = 0.0918861254207547
30
31     Degrees of freedom for ChiSquare = 2
32
33     z for calculating confidence limits = 1.96
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48

```

	Prop. kill	Lower limit	Time	Upper limit
31	0.1000	0.2014	1.0874	1.6197
32	0.2000	0.8546	1.5598	2.0322
33	0.3000	1.3042	1.9004	2.3510
34	0.4000	1.6677	2.1915	2.6440
35	0.5000	1.9861	2.4635	2.9393
36	0.6000	2.2824	2.7356	3.2566
37	0.7000	2.5765	3.0266	3.6191
38	0.8000	2.8962	3.3673	4.0678
39	0.9000	3.3094	3.8397	4.7202
40	0.9100	3.3631	3.9032	4.8099
41	0.9200	3.4210	3.9723	4.9077
42	0.9300	3.4843	4.0482	5.0157
43	0.9400	3.5545	4.1330	5.1369
44	0.9500	3.6340	4.2298	5.2756
45	0.9600	3.7267	4.3434	5.4393
46	0.9700	3.8397	4.4831	5.6414
47	0.9800	3.9886	4.6688	5.9115
48	0.9900	4.2207	4.9615	6.3396

49	0.9910	4.2540	5.0037	6.4015
50	0.9920	4.2905	5.0502	6.4698
51	0.9930	4.3313	5.1021	6.5461
52	0.9940	4.3776	5.1611	6.6328
53	0.9950	4.4311	5.2294	6.7335
54	0.9960	4.4951	5.3113	6.8542
55	0.9970	4.5752	5.4141	7.0060
56	0.9980	4.6840	5.5541	7.2129
57	0.9990	4.8603	5.7818	7.5503

Footnotes

Names of products are included for the benefit of the reader and do not imply endorsement or preferential treatment by USDA.