ELSEVIER

Computers & Geosciences 31 (2005) 1241–1248

COMPUTERS & GEOSCIENCES

www.elsevier.com/locate/cageo

Statistical analysis of water-quality data containing multiple detection limits: S-language software for regression on order statistics ☆

Lopaka Lee*, Dennis Helsel

US Geological Survey, Denver Federal Center, MS 973, Denver, CO 80225, USA

Received 20 August 2004; received in revised form 15 March 2005; accepted 17 March 2005

Abstract

Trace contaminants in water, including metals and organics, often are measured at sufficiently low concentrations to be reported only as values below the instrument detection limit. Interpretation of these "less thans" is complicated when multiple detection limits occur. Statistical methods for multiply censored, or multiple-detection limit, datasets have been developed for medical and industrial statistics, and can be employed to estimate summary statistics or model the distributions of trace-level environmental data.

We describe S-language-based software tools that perform robust linear regression on order statistics (ROS). The ROS method has been evaluated as one of the most reliable procedures for developing summary statistics of multiply censored data. It is applicable to any dataset that has 0 to 80% of its values censored. These tools are a part of a software library, or add-on package, for the R environment for statistical computing. This library can be used to generate ROS models and associated summary statistics, plot modeled distributions, and predict exceedance probabilities of water-quality standards.

Keywords: Geochemistry; S; Censored data

1. Introduction

Water-quality data sets commonly contain analytical values that at the time of determination, were lower than limits deemed reliable enough to report as numerical values. These observations are reported as seminumerical values that contain qualifiers indicating that the analyte is below the limits of reliability for accurate quantification. Typically these values are expressed as "nondetects" or "less thans" such as <0.5. Data sets containing values produced in this manner are referred to as censored data sets, where the qualified value is known as the censoring limit.

Furthermore, it is common to have water-quality data sets that contain more than one censoring limit. This occurs when the data set may have been generated over a time when the analyzing laboratory has changed levels of detection as instruments have gained accuracy, or laboratory protocols have established new limits. Data containing multiple detection limits are called multiply censored data sets.

Several data-analysis procedures are available for multiply censored data. These procedures can be divided

☆ Code available from Comprehensive R Archive Network at http://cran.r-project.org.

*Corresponding author. Tel.: +1 30 32365529; fax: +1 30 32363200.

E-mail address: rclee@usgs.gov (L. Lee).

0098-3004/$ - see front matter © 2005 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cageo.2005.03.012

into three classes (Helsel, 2005): (1) Simple-Substitution Methods, (2) Parametric Methods, and (3) Nonparametric Methods.

Simple substitution methods, where arbitrary quantitative values are substituted for each censoring limit, have been shown by investigators to be the least precise method (Helsel, 1990). Different disciplines have different traditions for the "best" substitution values (Sanford et al., 1993). One-half, seven-tenths, and one over the square-root of two times the detection limit are among the most common substitution values used. However, any single value between zero and the detection limit is arguably as good as another. Thus, simple substitution, particularly when used on multiply censored data, may introduce a signal that is not present in the data, or obscure a signal that is present. Simple substitution produces biased estimates of summary statistics that are dependent on the value being substituted.

Parametric methods require sufficient data to validate the use of a specific distributional model—a requirement that is difficult to meet with small multiply censored datasets. Additionally, these methods assume a single distributional model that may, in fact, change for different grouping factors within a sample population. The most common classes of assumed distributions with environmental data are skewed distributions such as the lognormal and gamma. Less frequently, square-root transformations or the normal distribution are assumed when data are less skewed than usually found for trace-element concentrations (Helsel and Cohn, 1988).

Nonparametric distributional modeling, such as methods based on Kaplan–Meier statistics, do not require the assumption of a specific distribution to estimate summary statistics for multiply censored datasets.

In between parametric and nonparametric is a "robust" semiparametric method developed by Helsel and Cohn (1988). This method is an implementation of what is generally referred to as a regression on order statistics or ROS.

This communication describes S-language-based software tools that perform robust linear ROS. These tools can be used to generate summary statistics, plot modeled distributions, and predict or estimate modeled values based on the modeled distributions. The tools are part of a software library called NADA for R. The library is named after *Nondetects and Data Analysis*: *Statistics for Censored Environmental Data* (Helsel, 2005) and is a add-on package for the R environment for statistical computing (R Development Core Team, 2003).

1.1. Previous work

Helsel and Cohn (1988) produced a computer program written in Fortran77 that implements the same robust ROS methodology described in this communication. The code also produces estimates of summary statistics using a maximum likelihood estimation (MLE) method. Although the statistical methodology of the code is sound, it has severe operational limitations. These include: (1) It is difficult to evaluate the quality of the resultant ROS model—the output is terse and consists only of the resultant summary statistics. (2) It is impossible to reuse, or extend, the ROS model in additional calculations such as prediction, or hypothesis tests. (3) There are fixed constraints on the input-data length that can be analyzed—the original code allows up to 1000 data points. This can only be extended by modifying and recompiling the code.

Our S-language software provides solutions to the limitations of the Fortran77 code by leveraging the features of S-language run-time environments. The improvements include: (1) methods, or routines, for the numerical and graphical inspection of resultant models, (2) resultant models may be easily deconstructed or extended for use in other computational routines, (3) there are no limits on input data length (beyond those imposed by the computer operating system and hardware).

2. Statistical methodology

The robust ROS method employed in this study was originally called the "MR" method by Helsel and Cohn (1988). It is a probability plotting and regression procedure that models censored distributions using a linear regression model of observed concentrations vs. their normal quantiles (or "order statistics"). The method has been evaluated as one of the most reliable procedures for developing summary statistics of multiply censored data (Shumway et al., 2002).

2.1. Assumptions and limitations

The ROS method assumes that all censoring thresholds are "left censored", i.e., all censored values are "less thans". It is applicable to any dataset containing 0 to 80% of its values censored. As noted by Helsel and Cohn (1988) and Helsel (2005), statistics derived from ROS models of populations having 80% or more censored values are very tenuous.

For data whose highest detection limit is below the 50th percentile, the median will equal the sample median computed by standard software without special consideration for censored values. The primary advantages of using ROS are realized when 50% to 80% of data are below the highest detection limit, or when estimates of the mean and standard deviation are required. Unlike the median, the mean and standard deviation cannot be estimated without some accommodation for censoring.

Additional assumptions are those inherent to linear regression. This includes the assumptions that the response variable (concentration) is a linear function of the explanatory variable (the normal quantiles) and that the error variance of the model is constant. Since the statistical distribution of water-quality data is typically skewed, these assumptions are usually addressed by transforming the data prior to analysis. Since most water-quality data with multiple censoring limits are lognormally distributed, the default behavior of our routines is to perform a log-normal transformation to input data prior to computation. However, this feature can be entirely suppressed or the user may provide an alternative set of transformation functions.

2.2. Computational methods

The robust ROS method implemented in the software can be summarized in the following algorithmic steps:

Computation of plotting positions for both censored and uncensored data: The plotting positions of censored and uncensored observations are computed using a formula first described by Hirsch and Stedinger (1987), and later reformulated by Helsel and Cohn (1988).

Plotting positions of both censored and uncensored data are computed using the exceedance probability, E_j, of each censoring limit. E_j is the probability of exceeding the jth censoring limit. It is defined as

$$E_j = E_{j+1} + (A_j/[A_j + B_j])(1 - E_{j+1}),$$

where A_j is the total number of uncensored observations in the range $[j, j+1)$ and B_j is the total number of observations, censored and uncensored, less than or equal to the jth censoring limit.

For a given uncensored observation, a Weibull-type plotting position p can be calculated by considering the exceedance probability of the censoring limit below the observation E_j, the exceedance probability of the censoring limit above the observation E_{j+1}, and the observation's rank among all the values within the j and $j+1$ censoring limit. In general, the Weibull-type plotting positions for uncensored observations are

$$p(i) = (1 - E_j) + (E_j - E_{j+1})r_i/(A_j + 1),$$

where r_i is the rank of the ith observation among the observations in the range $(j, j+1]$ (Hirsch and Stedinger, 1987).

Similarly, the Weibull-type plotting positions for censored observations are given by

$$p(i) = (1 - E_j)r_i/(C_j + 1),$$

where C_j is the total number of censored values in the range $(j, j+1]$.

Forming the linear regression model: A linear regression of the uncensored observations vs. the normal quantiles of the uncensored plotting positions is formed. The normal quantiles of the plotting positions are the "order statistics" of the ROS method.

Estimation of the censored concentrations: The censored concentrations are modeled using the parameters of the linear regression and normal quantiles of the censored data. These modeled censored observations are only used corporately, along with the uncensored observations, to model the distribution of the sample population. Individually, they are not considered the values that would have existed in the absence of censoring.

Computation of summary statistics: The observed uncensored values are combined with modeled censored values to corporately estimate summary statistics of the entire population. By combining the uncensored values with modeled censored values, this method avoids transformation bias (Helsel and Cohn, 1988).

3. S-language implementation

Our software implementation of the robust ROS method is written entirely in the S-language, a computer language designed for data analysis and graphics (Becker et al., 1988; Chambers, 1998).

There are currently two widely available software systems that possess the ability to run S-language software: S-Plus, a proprietary statistical computing environment developed by the Insightful Corporation, and "R", an open source computing environment of the S language developed by the R Development Core Team (2003). Although both of these software systems contain S-language interpreters, there are notable differences in the S-language constructs available in each system.

We have chosen to use R as our primary development target for our software. Thus, our exposition and discussion of our software is specific to its use in R. Currently, the routines will not run on S-Plus.

Our software is a part of a library, or package for the R environment called *NADA for R*. The library name is taken from Helsel (2005) and implements other methods detailed in the reference.

This communication does not provide documentation of every function within the NADA for R package. Additional information on functions in the package is available through the on-line help system in R.

Examples of the usage of each function, and a discussion of options and output is provided below. Throughout the discussion, S-language constructs and output are set in monospaced font `like this`. The R command-line prompt is shown as: > (the greater-than symbol). Where the output is lengthy or is implied from a previous example, ellipsis (…) are used to designate that the section has been cut short for the sake of brevity.

3.1. Model construction

The NADA library functions for constructing and manipulating ROS models are listed in Table 1. For the following examples, we use a dataset of dissolved arsenic concentrations in groundwater. These data are a subset from the US Geological Survey National Water Quality Assessment (NAWQA) Data Warehouse (Williamson and Booth, 2004). The data are distributed as a part of the NADA module and can be loaded using the `data()` function after the NADA library has been attached to the working environment.

```
> library(NADA)
> data(NADA.As)
> ls()
[1] "As"
> As
```

```
     obs       censored
1    0.090     TRUE
2    0.090     TRUE
3    0.090     TRUE
4    0.101     FALSE
5    0.136     FALSE
6    0.340     FALSE
7    0.457     FALSE
8    0.514     FALSE
9    0.629     FALSE
...
>
```

The arsenic dataset is structured in a S-language data frame which is a table, or spread-sheet like structure. The "obs" column is a numeric vector which contains all the observed arsenic concentration values, both censored and uncensored. The "censored" column is a logical vector containing TRUE or FALSE where the concentrations in "obs" are censored (are a "less than") or uncensored, respectively.

Table 1
NADA for R library functions for the creation and manipulation of ROS models. Detailed information on individual functions is available through the on-line help system

Function name	Purpose
`ros()`	Construct ROS models
`summary()`	Verbose summary of ROS model
`plot()`	Produces a Q–Q norm plot of a ROS model
`as.data.frame()`	Converts ROS model to a data frame
`quantile()`	Returns quantile estimates of an ROS model
`mean()`	Returns the mean of the modeled ROS data
`median()`	Returns the median of the modeled ROS data
`sd()`	Returns the standard deviation of the modeled ROS data
`predict()`	Predict normal quantiles of a ROS model

Typically, analytical data that is received from a laboratory or downloaded from a database system is not in the above format. It is common for the censoring qualifiers, or symbols, to be concatenated with numeric values in "less than" strings such as `<0.5`. The NADA library contains the function, `splitQual()` that can separate the character-qualifier symbols from numeric symbols in these strings and form separate value and qualifier vectors, or columns. Detailed information on this function is available through online help by typing `?splitQual`.

A new ROS model may be constructed by calling the `ros()` function. This function takes two mandatory arguments, a numeric vector of observations "`(obs)`" and a logical vector "`censored`" indicating TRUE or FALSE, where the corresponding numeric vector elements are censored or not censored respectively.

```
> AsModel = ros(obs = As$obs,
censored = As$censored)
> AsModel
```

```
Multiply-Censored ROS Model

             N: 50
      Censored: 23
    % Censored: 46

      Mean: 3.41
    StdDev: 8.381

         Quantiles:
5%       10%      25%      50%
0.0353   0.0593   0.1740   0.6335

75%      90%      95%
2.7460   6.1214   16.9438

Use summary() to view the linear regression
model
```

The default textual summary of the model includes an indication of the percentage of censored values, the resultant mean, standard deviation and quantiles of the modeled population.

By default, `ros()` performs a log-normal transformation to the data prior to forming the linear regression model. The reverse transformation (exp) is applied after

the modeled censored values have been predicted. This default is because a lognormal transformation is commonly the best transformation to normalize error variance in multiply censored water-quality data. *However*, the `ros()` function allows the user to supply any desired transformation function, or none at all. This is accomplished using the optional arguments to the `ros` function "`forwardT`" and "`reverseT`". These arguments name the forward and reverse transformation functions that should be applied to the data (ie., "log" and "exp" for a lognormal transform). If either or both of these arguments are set to `NULL`, no transformation is performed. Thus, the following example would model would perform the same analysis as above, without logtransforming the data:

```
> ros(obs = As$obs,
censored = As$censored,
forwardT = NULL)Multiply-Censored ROS
ModelN: 50Censored: 23% Censored: 46Mean:
0.08904StdDev: 11.09
```

```
Quanti-
les:5%10%25%50%75%90%95%−14.97−12.30-
4.870.572.846.1216.94Use summary() to
view the linear regression modelNote
```

that the resultant model predicts negative values. The ROS routine does not prevent the user from formulating such models; we feel it is best to leave issues of interpretation to the user.

3.2. Model plotting and evaluation

For a graphical display of constructed models, the generic `plot()` may be used to display a Q–Q plot of the ROS model and observations.

```
> plot(AsModel)
```

The output plot shows the observed, uncensored values, and the linear regression model as a solid line (Fig. 1). The plot also shows the normal quantiles of the data expressed as a percent chance of exceedance.

As an option to the `plot` function, the user can plot the modeled censored observations as well (Fig. 2).

```
> plot(AsModel, plot.censored = TRUE)
```

Although this feature is useful to help visualize how the ROS method works, we stress that the modeled censored observations are only used collectively to model the distribution of the uncensored population. *Modeled censored values should not be used on an individual basis to represent the values that would have been detected in the absence of censoring.*

Any of the modeled values shown in Fig. 2 could be associated with any censored observation having the same detection limit. None of these modeled values is necessarily the same as values that would have been measured had the lab instrument possessed greater resolution.

The regression models produced by `ros()` may be summarized using the generic R function `summary()`

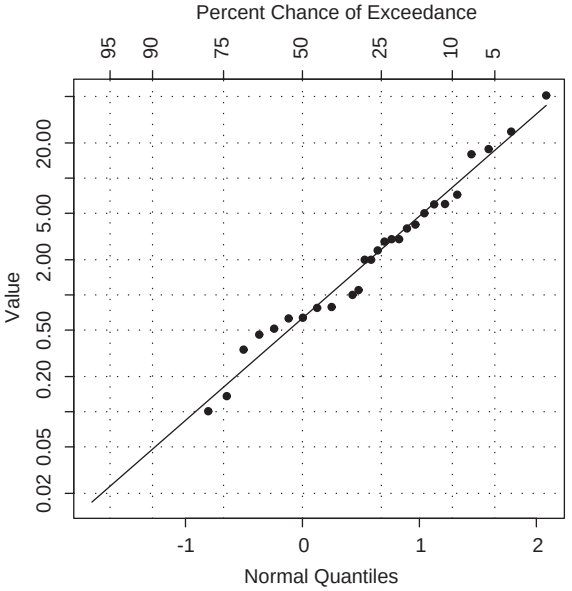

Fig. 1. Normal Q–Q plot for a ROS model.

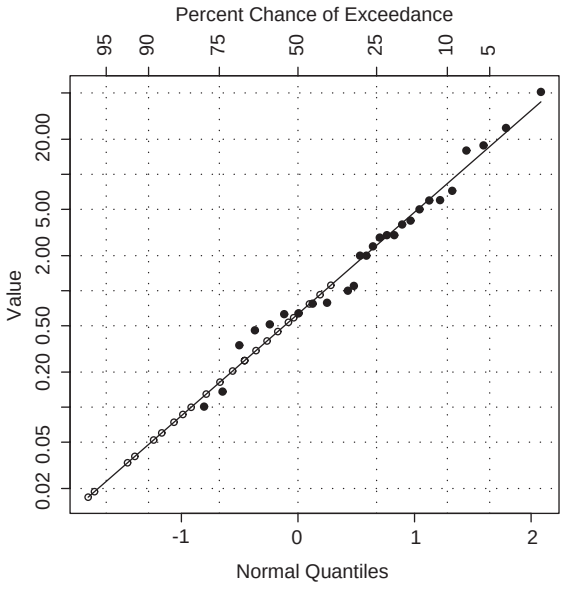

Fig. 2. Normal Q–Q plot for a ROS model. Solid circles are uncensored observations. Open circles are modeled censored values.

which produces a short textual description of the linear regression model. By specifying the `plot = TRUE` option to the `summary()` function, the routine will interactively cycle through four plots which graphically summarize the quality of the linear regression model.

```
> summary(AsModel, plot = TRUE)

Call:
lm(formula = obs.transformed ~ pp.nq)

Residuals:
Min        1Q        Median    3Q       Max
−0.4097   −0.1531   −0.0220   0.1146   0.4212


Coefficients:
          Estimate    Error    t value   Pr(>|t|)
          Std.
(Inter-   −0.4604     0.0560   −8.21     1.4e-08 ***
cept)
pp.nq     2.0132      0.0591   34.07     < 2e-16 ***
---

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.226 on 25 degrees of freedom
Multiple R-Squared: 0.979, Adjusted R-squared: 0.978
F-statistic: 1.16e+03 on 1 and 25 DF, p-value: <2e-16
```

This command provides four plots which can be used to evaluate the quality of the linear model fit: a plot of residuals against fitted values (Fig. 3), a Scale–Location plot of residuals against fitted values (Fig. 4), a Normal Q–Q plot of residuals (Fig. 5), and a plot of Cook's distances for residuals versus observation number (Fig. 6). These plots are standard tools in assessing the quality of a linear model.

It is often useful to deconstruct the ROS model and use the modeled data in other types of computations. Any of the standard functions used to manipulate linear model objects can be used with ROS model objects. This includes `coef()` to extract the linear model coefficients, and `resid()` to extract the linear model residuals. For example, the following returns the coefficients of our arsenic ROS model:

```
> coef(AsModel)
(Intercept)                    pp.nq
−0.46                          2.01
```

ROS models may also be converted to a data frame using the generic function `as.data.frame()`:

```
> as.data.frame(AsModel)
        obs censored        pp modeled
1      0.09     TRUE     0.040    0.019
2      0.09     TRUE     0.081    0.038
3      0.09     TRUE     0.121    0.060
4      0.10     FALSE    0.210    0.101
5      0.14     FALSE    0.259    0.136
6      0.34     FALSE    0.307    0.340
7      0.46     FALSE    0.356    0.457
8      0.51     FALSE    0.404    0.514
9      0.63     FALSE    0.453    0.629
10     0.64     FALSE    0.501    0.638
...
>
```

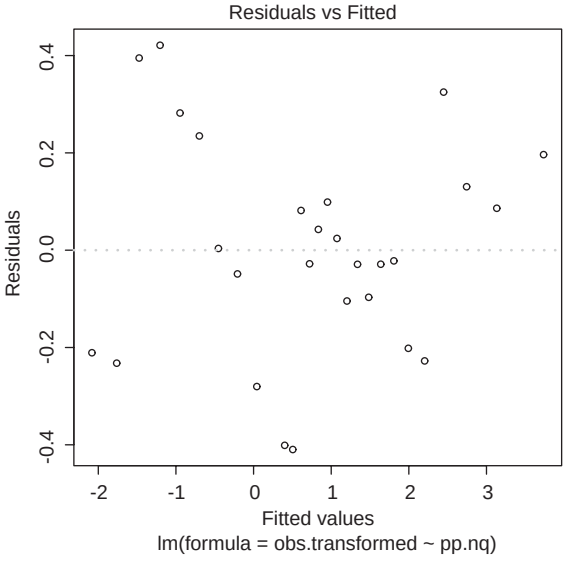

Fig. 3. ROS model residuals plotted against fitted values.

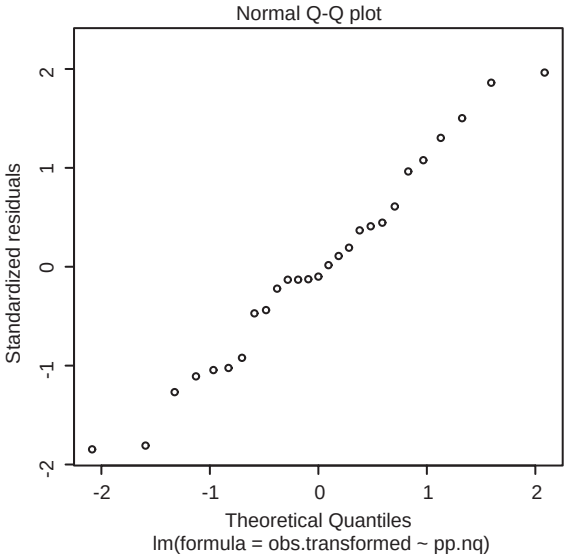

Fig. 4. Scale–Location plot of ROS model residuals plotted against fitted values.

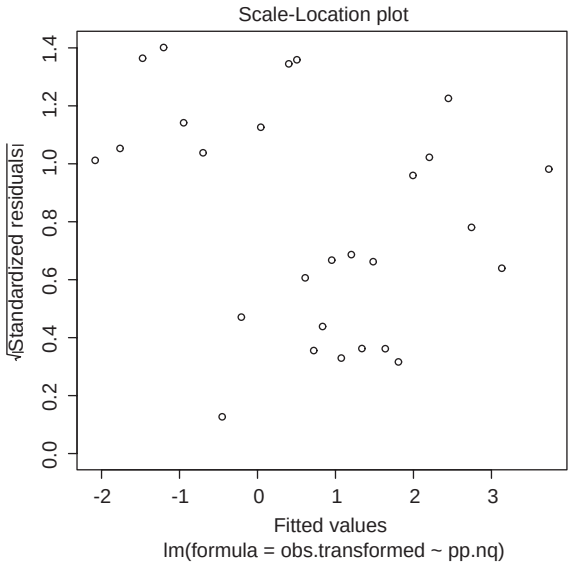

Fig. 5. Normal Q–Q plot for ROS model residuals.

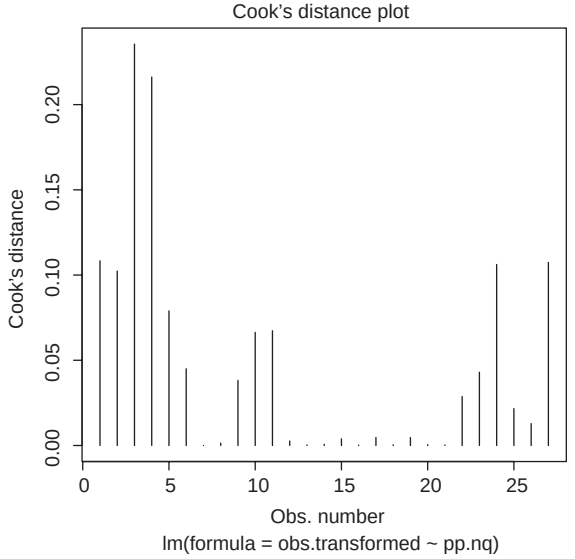

Fig. 6. Cook's distances for ROS model residuals plotted against observation number.

The returned data frame contains the original sorted observations (obs), the associated indication of censoring (censored), the calculated plotting positions (pp), and the modeled data (modeled). As discussed above, the modeled data consists of the original observations where values are uncensored, and data predicted from the linear model where values are censored. Note that this conversion discards all of the linear model information contained in the original output object.

3.3. Model query and prediction

The software also provides the ability to use ROS output objects as the basis for simple queries and basic predictive modeling.

The generic `quantile()` method can be used to ascertain the concentration associated with a particular quantile value. For our example data set, the 15th percentile occurs at approximately 0.1 μg/L:

```
> quantile(AsModel, 0.15)
  15%
0.0911
```

Additional generic methods that can operate on ROS objects include `mean()`, `sd()`, and `median()`

```
> mean(AsModel)
[1] 3.41
> sd(AsModel)
[1] 8.381
>
```

The generic `predict()` provides a method to predict the observation that would occur at a given normal quantile, given a specific ROS model.

```
> predict(AsModel, 1)
[1] 4.73
>
```

4. Conclusions

Our S-language software enables researchers to perform a robust regression on order statistics method for multiply censored data. These methods are necessary to correctly estimate statistics for multiply censored data such as trace-element analyses of water.

The software provides functions for the numerical and graphical inspection of ROS models that allow users to evaluate the quality of models. The software also includes methods that allow the model output to be easily deconstructed for use in other computations.

We have found this software to be extremely useful in summarizing trace element distributions and used it as our primary computational tool in our work on defining baseline models of trace elements in ground water of the United States (Lee and Helsel, in press).

The software is part of a developing project and will include enhancements as our work continues. Future enhancements will include maximum likelihood, survival analysis, and other methods described in Helsel (2005).

Acknowledgements

David Lorenz and Scott Charlton of the USGS provided thorough internal reviews of this

manuscript and the software code. Additionally, Bob Garrett and an anonymous reviewer greatly improved content.

Appendix A. Obtaining and Installing NADA for R

The functions described in this communication are part of a software library, or package, for the R statistical computing environment called NADA for R. The primary requirement for running the software is a working installation of the R environment. R is free software and can be obtained, used, and modified at no monetary cost (R Development Core Team, 2003). We have made our software available under the same conditions as R itself. Thus, it is possible for entities to use and extend our software even if conditions of law or finance prohibit the use of a nonfree software solution.

Once R is installed and the machine has a functioning Internet network connection, the NADA package may be automatically installed using the following command:

```
>install.packages( ‘ ‘ NADA’’)
```

Alternatively, the package may be manually installed by downloading it from the Comprehensive R Archive Network at http://cran.r-project.org and using the standard package installation methods described at this site and in the R documentation.

The US Geological Survey also maintains a larger, more extensive S-Plus package for water-resource statistics (Slack et al., 2003). At the time of publication, this package does not contain the code described in this communication. However, the S-Plus package does contain an implementation of ROS based on the older Fortran code by Helsel and Cohn (1988) discussed above.

References

Becker, R., Chambers, J., Wilks, A., 1988. The New S Language. Chapman & Hall, New York, 702p.

Chambers, J., 1998. Programming with Data. A Guide to the S Language. Springer, New York, 469p.

Helsel, D.R., 1990. Less than obvious: statistical treatment of data below the detection limit. Environmental Science and Technology 24 (12), 1767–1774.

Helsel, D.R., 2005. Nondetects and Data Analysis. Wiley, New York, 250p.

Helsel, D.R., Cohn, T.A., 1988. Estimation of descriptive statistics for multiply-censored water quality data. Water Resources Research 24 (12), 1997–2004.

Hirsch, R., Stedinger, J., 1987. Plotting positions for historical floods and their precision. Water Resources Research 23 (4), 715–727.

Lee, L., Helsel, D.R., Baseline models of trace elements in major aquifers of the US. Applied Geochemistry. In press.

R Development Core Team, 2003. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria. http://www.r-project.org.

Sanford, R., Pierson, C., Corvelli, R., 1993. An objective method for censored geochemical data. Mathematical Geology 25 (1), 59–80.

Shumway, R.H., Azari, R.S., Kayhanian, M., 2002. Statistical approaches to estimating mean water quality concentrations with detection limits. Environmental Science and Technology 36 (15), 3345–3353.

Slack, J., Lorenz, D., et al., 2003. USGS library for S-PLUS for Windows – Release 2.1. Open-File Report 03-357, US Geological Survey, 50p.

Williamson, S., Booth, N., 2004. USGS national water quality data and maps on the web. In: Abstracts with Programs. National Water Quality Monitoring Conference. 17–20 May, p. 65.