Statistical Analysis of Water-Quality Data Containing Multiple Detection Limits II: S-language Software for Nonparametric Distribution Modeling and Hypothesis Testing [1]

Lopaka Lee [2] and Dennis Helsel

U.S. Geological Survey, Denver Federal Center, MS 973 Denver CO 80225

Abstract

Analysis of low concentrations of trace contaminants in environmental media often results in left-censored data that are below some limit of analytical precision. Interpretation of values becomes complicated when there are multiple detection limits in the data – perhaps as a result of changing analytical precision over time.

Parametric and semi-parametric methods, such as maximum likelihood estimation and robust regression on order statistics, can be employed to model distributions of multiply censored data and provide estimates of summary statistics. However, these methods are based on assumptions about the underlying distribution of data. Nonparametric methods provide an alternative that does not require such assumptions.

A standard nonparametric method for estimating summary statistics of multiply-censored data is the Kaplan-Meier (K-M) method. This method has seen widespread usage in the medical sciences within a general framework termed "survival analysis" where it is employed with right-censored time-to-failure data. However, K-M methods are equally valid for the left-censored data common in the geosciences.

Our S-language software provides an analytical framework based on K-M methods that is tailored to the needs of the earth and environmental sciences community. This includes routines for the generation of empirical cumulative distribution functions, prediction or exceedance probabilities, and related confidence limits computation. Additionally, our software contains K-M based routines for nonparametric hypothesis testing among an unlimited number of grouping variables.

A primary characteristic of K-M methods is that they do not perform extrapolation and interpolation. Thus, these routines cannot be used to model statistics beyond the observed data range or when linear interpolation is desired. For such applications, the aforementioned parametric and semi-parametric methods must be used.

Preprint submitted to Elsevier Science *4 August 2005*

1 Introduction

It is common for water-quality data sets to contain analytical values that, at the time of determination, were lower than limits deemed reliable enough to report as numerical values. Data sets with this characteristic are referred to as "censored" data sets. If multiple censoring thresholds are present, perhaps as a result of changing instrument resolution, then the data are called "multiply censored".

Several data-analysis procedures are available for censored and multiply-censored data. These procedures can be divided into three classes (Helsel, 2005): 1) Simple-Substitution Methods, 2) Parametric Methods, and 3) Nonparametric Methods.

Simple substitution, where arbitrary quantitative values are substituted for each censoring limit, produces biased estimates of summary statistics that are dependent on the value being substituted. Thus, substitution is not a defensible statistical procedure (Helsel, 1990).

Parametric methods require sufficient data to validate the use of a specific distributional model – a requirement that is difficult to meet with smaller multiply-censored datasets.

Helsel and Cohn (1988) and Shumway et al. (2002) describe a semi-parametric method that is an implementation of regression on order statistics called Robust ROS. This method has been evaluated as one of the better-performing methods for estimating summary statistics and modeling distributions of multiply censored data.

In a prior communication (Lee and Helsel, in press) we described our S-language software for Robust ROS analysis of multiply-censored datasets. Robust ROS is a powerful and useful method. However, since ROS is a procedure based on parametric-linear regression, resultant models and related statistics are only valid if the assumptions of parametric linear regression are fulfilled. This includes the assumption that the response variable is a linear function of the explanatory variable and that the error variance of the model is constant. Since the statistical distribution of geochemical data is typically skewed, these assumptions are usually addressed by transforming the data prior to analysis. Transformation can usually fulfill these requirements, but not always.

In contrast to parametric methods, nonparametric methods do not require

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

[2] Corresponding author email: rclee@usgs.gov

the assumption of a specific distribution to estimate summary statistics for multiply-censored datasets. The standard nonparametric method for estimating summary statistics of censored data is the Kaplan-Meier (K-M) method. It has seen widespread usage in the medical sciences within a more general framework termed "survival analysis".

In survival analysis, the interest is time-to-death, or time-to-failure. In this context, censored data result when the life of a subject exceeds the study length. This results in censored data that are right-censored (expressed as "greater-than" values).

Although survival analysis is rarely concerned with left-censored data, the methods used, particularly the Kaplan-Meier method, are equally valid for the left-censored data that are common in the earth and environmental sciences. Unfortunately, K-M based methods have largely been overlooked in the geosciences.

This communication describes S-language based software tools that generate empirical cumulative distribution functions (ECDFs) using nonparametric Kaplan-Meier methods. These tools can be used to generate summary statistics, plot modeled distributions, and predict or estimate modeled values based on the modeled distributions. Additionally, these tools provide methods for nonparametric hypothesis testing based on rank-sum methods.

The tools are part of a software library called NADA for R. The library name is taken from *Nondetects and Data Analysis: Statistics for Censored Environmental Data* (Helsel, 2005) and is an add-on package for the R environment for statistical computing (R Development Core Team, 2005).

2 Software Implementation

The functions detailed in this communication build upon functionality provided in the `survival` package of R. The `survival` package is distributed as a standard part of the R environment. However, `survival` routines are incapable of processing left-censored data. To address this issue, we use a formalism suggested by Helsel (2005). In this approach left-censored data are rescaled, or "flipped", to right-censored data by subtracting the observations from a large constant value. These rescaled data are then processed in existing survival-analysis routines and the resultant outputs are flipped back to the original scale when appropriate. Since this rescaling does not change the internal proportionality of the data there is no loss of information, or introduced bias. Our software automatically performs all of these rescalings for the user.

Our software further enhances routines in the `survival` package by providing methods for summary, plotting, query, prediction, and hypothesis testing. These routines provide interfaces and output information that are familiar to the geoscience community and is also consistent with other methods available in the NADA for R package.

Our software is written entirely in the S-language, a computer language designed for data analysis and graphics (Becker et al., 1988; Chambers, 1998). R and S-Plus are the two widely available systems capable of running S-language software. We have chosen to use R as our primary development target for our software. Thus, our exposition and discussion of our software are specific to its use in R. Currently, the routines have not been ported to S-Plus.

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 mono-spaced 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 (`...`) is used to designate that the section has been cut short for the sake of brevity.

2.1 Model Construction

The NADA library functions for constructing and manipulating empirical cumulative distribution functions 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 U.S. 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(Arsenic)
> Arsenic
      As    AsCen Aquifer
1  0.090     TRUE       A
2  0.090     TRUE       B
3  0.090     TRUE       A
4  0.101    FALSE       B
5  0.136    FALSE       B
   ...
>
```

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

Typically, analytical data that are received from a laboratory or downloaded from a database system are 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 on-line help by typing `?splitQual`.

The `cenfit` function computes an empirical cumulative distribution function (ECDF) for censored data using the Kaplan-Meier method. 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.

```
> attach(Arsenic)
> AsECDF = cenfit(obs=As, censored=AsCen)
> AsECDF
       n    n.cen   median     mean  se(mean)
  50.000   23.000    0.638    3.427    1.173
```

The default textual summary of the model includes the total number of observations, the number of censored observations (n.cen), the computed median, mean, and standard error of the mean.

The generic `summary` function provides more detailed information on ECDF objects. This includes each observation (`obs`) with associated values for the number of observations at risk of exceeding the given value (`n.risk`), the number of uncensored observations or "uncensored events" at that value (`n.event`), the computed probability or percentile (`prob`), standard error (`std.err`), and upper and lower model confidence limits (in this case 95%).

```
> summary(AsECDF)
      obs n.risk n.event   prob std.err 0.95LCL 0.95UCL
1   0.090      3       0 0.1600 0.51113 0.05876  0.4357
2   0.101      4       1 0.1600 0.51113 0.05876  0.4357
3   0.136      5       1 0.2133 0.42180 0.09333  0.4876
4   0.340      6       1 0.2667 0.35765 0.13229  0.5375
5   0.457      7       1 0.3200 0.30754 0.17513  0.5847
```

...

2.2 Model Plotting and Evaluation

The generic function `plot` is used to graphically display ECDF objects.

```
> plot(AsECDF)
```

Figure 1 shows ECDFs produced using the K-M method. The default x-axis of the plots is lognormal; however, this can be changed using options to the `plot` function. Furthermore, the R function `par` can be used to extensively customize the plot.

The ECDFs produced using the K-M method are discrete-interval step functions. These step functions are estimates of a cumulative distribution for the data. The K-M method approximates this distribution using the following formula:

$$F = \prod_{j=1}^{k} \frac{b_j - d_j}{b_j}$$

The method ranks detected observations from small to large, accounting for the number of censored data in between each detected observation, and placing each non-detect at its detection limit prior to ranking. The number b equals the number of observations, both detected and censored, at and below each detected concentration. The number of detected observations at that concentration is d (d is greater than 1 for tied values). The *incremental* exceedance probability $\frac{b_j - d_j}{b_j}$ is the probability of exceeding the next highest detected concentration, given the number of data at and below that concentration. The probability of an observation is the product of the $j = 1$ to k incremental probabilities to that point.

The result of this approach is a discrete percentile estimate for every observation. A step function is formed by plotting each observation-percentile pair and using constant-interpolation between points. Every step (or jump) in the plot is at a position where a new observation occurs.

It is important to realize that since the resultant ECDF is a step function, it is by definition discontinuous, and therefore incapable of linear interpolation or extrapolation. Thus, when the percentile of interest lies outside of the range of observations, or when a linearly-interpolated percentile estimate is desired, the K-M method is not appropriate. In such cases, a method that assumes some sort of model for the data distribution must be employed. Two possible methods for doing so include fully-parametric Maximum Likelihood

Estimation (MLE) methods and the aforementioned Robust-ROS method.

2.3 Model Query and Prediction

The software also provides the ability to use ECDF objects as the basis for simple queries and univariate-predictive modeling.

Generic methods for querying ECDF objects include `median`, `mean`, and `sd`. For example:

```
> median(AsECDF)
[1] 0.638
> mean(AsECDF)
  rmean se(rmean)
  3.427     1.173
> sd(AsECDF)
[1] 8.295
```

The mean is generally considered less useful than the median when working with censored geochemical data. Such data are usually so skewed that the mean is not a typical value. In addition, when the lowest observation in the dataset is censored, the K-M estimate of the mean will be biased high.

Estimates of the standard deviation are of less interest than the mean. The variance and standard deviation are not resistant to skewness and outliers, and so provide a poor measure of the variability of the data when those data are strongly skewed.

Quantile statistics, such as the median and inter-quartile range (IQR, or Q75-Q25) provide more robust measures of central tendency and variability. Percentile estimates generated using the K-M method contain no bias.

For quantile estimates, the `quantile` function returns the observation associated with a particular quantile value. For our example data set, the 25th percentile occurs at approximately 0.14 μg/L:

```
> quantile(AsECDF, 0.25)
  25%
  0.136
```

Note that for percentile estimates, the value is the minimum observation (x-value) on the ECDF that is intersected by the line drawn at probability (y-value) of interest. Thus, when the percentile of interest happens to be on a horizontal portion of a step, the associated observation is the *minimum* value

along that continuum.

Similarly, the `predict` function provides a method to predict the probability of any observation.

```
> predict(AsECDF, 10)
[1] 0.92
```

Similar to the `quantile` function above, `predict` returns the minimum quantile (y-value) on the ECDF that is intersected by the line drawn at the queried observation (x-value). Thus, when the observation of interest happens to be on the vertical portion of a step, the associated quantile is the *minimum* value along that continuum.

The `pexceed` function is a convenience function that returns the probability of exceedance for an observed value. This is simply one minus the probability of an observation. This function is useful in cases were the exceedance probability of an unobserved value is of interest. For example, the exceedance probability of a water-quality standard or criterion at 10 $\mu g/L$ is approximately 8 percent:

```
> pexceed(AsECDF, 10)
[1] 0.08
```

Figure 1 shows vertical and horizontal lines at an observation of 10 $\mu g/L$ and the associated probability and percent chance of exceedance.

2.4 Confidence Interval Estimates

An important option to the `cenfit` function is the specification of the desired confidence limits to associate with the ECDF. This is specified using the `conf.int` option of `cenfit`. This option takes a decimal fraction specifying the desired confidence limits of the ECDF. The default confidence limit is 0.95. Note that the confidence limits of an ECDF can not be updated once it is constructed. Manipulations, such as query and prediction, will utilize this confidence interval.

The dotted step functions on Figure 1 show the confidence limits of our constructed model. Confidence limits are shown only when a single ECDF is plotted and may be suppressed using the true/false `conf.int` option to `plot`.

The `median`, `quantile`, `predict` and `pexceed` functions can provide estimates of confidence intervals around any computed percentile. Confidence intervals may be presented using the logical `conf.int` option to these functions. The following example shows that we can have a 95% certainty that a water-quality

criterion of 10 $\mu g/L$ has at most a 15 percent chance of exceedance in our data set.

```
> pexceed(AsECDF, 10, conf.int=TRUE)
  obs pexceed 0.95LCL  0.95UCL
1  10    0.08  0.1522 0.001644
```

Confidence intervals are based on standard error estimates computed using Greenwood's formula – a widely-used method of computing standard errors in survival analysis (Collett, 2003).

2.5 Factoring ECDFs by Groups

The function `cenfit` also accepts a third term "`groups`" which is a vector of factors that can be used to break the observations into different groups, or treatments. Grouping observations provides a means to construct multiple ECDFs that can simultaneously be plotted and queried in functions. Grouping factors can be any number of discriminating labels such as sampling location, methods, or analytical instruments.

In our Arsenic data, the third column contains a vector of factors named "`Aquifer`" which describes two hypothetical geohydrologic sources for the data. If we specify `Aquifer` as a grouping variable to `cenfit`, the result contains separate ECDFs for each aquifer group.

```
> AsECDF2 = cenfit(As, AsCen, Aquifer)
> AsECDF2
           n  n.cen median mean se(mean)
aquifer=A 18     10  0.774 1.99    0.965
aquifer=B 32     19  0.788 4.24    1.734
```

Individual ECDFs can be obtained by indexing the output object as in the following: `AsECDF2[1]`, `AsECDF2[2]`.

All of the plotting, query, and prediction functions discussed above also work with grouped ECDFs. Thus, we can plot the grouped models simply by using the `plot` function as in `plot(AsECDF2)` or interactively overlay plots using indexing. For example, Figure 2 was produced using the following sequential commands:

```
> plot(AsECDF2[1], conf.int=FALSE, lty="solid")
> lines(AsECDF2[2], lty="dashed")
> legend(locator(), c("Aquifer A", "Aquifer B"), lty=c("solid", "dashed"))
```

Once multiple ECDFs are produced and plotted, the natural progression in analysis is to determine if significant differences exist between ECDFs.

The `cendiff` function can test for differences between two or many groups of data. This function operates identically to the `cenfit` function. However, the `groups` vector is mandatory and serves to factor the observations apart for hypothesis testing. For example, to test if there is a significant difference between the arsenic concentrations of the two aquifers shown in Figure 2:

```
> cendiff(As, AsCen, Aquifer)
           N Observed Expected (O-E)^2/E (O-E)^2/V
aquifer=A 18      5.5     7.34     0.461     0.994
aquifer=B 32     13.3    11.48     0.295     0.994

 Chisq= 1 on 1 degrees of freedom, p= 0.319
```

The reported p-value (0.319 in the above output) is always for a two-sided test. In this case, the null hypothesis is that there is no difference in arsenic distributions between the two aquifers. Here we can safely conclude that there is not a significant difference between the two ECDFs shown in Figure 2 as the reported p-value is larger than 0.05.

The `cendiff` function uses nonparametric score tests that are extensions to censored data of Wilcoxon-style tests (rank-sum and Kruskal-Wallis) for uncensored data. Scores are modified ranks, based on the Kaplan-Meier percentiles for all detected observations. The null hypothesis is that the distributions of data in every group are identical (their ECDFs are the same). The alternative hypothesis is that at least one group is different. Scores are computed using a weight function; the family of tests derived using different weights is called the G-rho family of tests (Harrington and Fleming, 1982). The two most common weights result in the log-rank test, and the Generalized Wilcoxon or Peto-Peto test. The `cendiff` function can produce either.

The default test used by the `cendiff` function is the Peto-Peto test. This form of the test is more powerful than the log-rank test, and is therefore more likely to detect true differences when data come from a lognormal distribution (Lee, 1992). The Peto-Peto test "gives more weight to early failures", meaning that it is sensitive to differences in the higher values of left-censored data sets (Lee, 1992). Because many environmental data sets are approximately lognormal, and the upper portions of groups are where detected differences often occur, the Peto-Peto test is judged to be the most appropriate and is set as the default.

Both `cenfit` and `cendiff` accept S-language formulas as input. S-language formulas are a syntax that allows the succinct expression of complex statistical models. These models are expressed in a syntax like: `response ~ explanatory` where the tilde symbol (`~`) reads "is modeled as a function of". Within the context of K-M methods, the response is the censored observations, and the explanatory variables provide means of grouping and/or stratifying the observations.

The `Cen` function is provided for the purpose of creating censored-response objects in formulas. Its usage is analogous to the `Surv` function in the `survival` package. However, unlike `Surv`, the `Cen` function provides the necessary framework to process left, right, or interval censored data.

In simple cases, the formula interface merely provides an alternative method of inputting data into routines. For example, the group factored models described above could be expressed using formulas:

```
> cenfit(Cen(As, AsCen) ~ Aquifer)
           N Observed Expected (O-E)^2/E (O-E)^2/V
aquifer=A 18      5.5     7.34     0.461     0.994
aquifer=B 32     13.3    11.48     0.295     0.994

 Chisq= 1 on 1 degrees of freedom, p= 0.319
```

3 Conclusions

Our S-language software provides easy to use, extensible functions for generating empirical cumulative distribution functions (ECDF) and related statistics using nonparametric Kaplan-Meier methods. Additionally, our software provides functions that perform nonparametric hypothesis testing of two or more ECDFs allowing tests for statistical significance between various grouping factors within a dataset. All of these functions are tailored for the left-censored data that are common within the geosciences.

These nonparametric methods are applicable when users do not want to make assumptions about the underlying distribution of data. But, these methods cannot predict beyond the observed range of data or linearly interpolate values between observations. In such cases, methods such as MLE or Robust ROS are more appropriate.

This software is part of a developing project and will include enhancements as our work continues. Future enhancements will include maximum-likelihood estimation methods, related regression methods, and plotting utilities for censored data.

4 Acknowledgments

David Kinniburgh of the British Geological Survey and Scott Charlton of the U.S. Geological Survey provided thorough internal reviews of the manuscript and code.

5 Appendix A – Obtaining and Installing the Software

The functions described in this communication are part of a software library, or package, for the R statistical computing environment called NADA for R.

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 for a variant of the S-language developed by the R Development Core Team (2005). 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 made R our primary development target. Thus, 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, 2005). 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 non-Free 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 U.S. Geological Survey also maintains a larger, more extensive S-Plus package for water-resource statistics described by Slack and Lorenz (2003) (see http://water.usgs.gov/software/statistics.html). At the time of publication this package does not contain the code described in this communication.

References

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

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

Collett, D., 2003. Modeling Survival Data in Medial Research, 2nd Edition. Chapman Hall/CRC, London.

Harrington, D. P., Fleming, T. R., 1982. A class of rank test procedures for censored survival data. Biometrika 69, 553–566.

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. John Wiley and Sons, 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.

Lee, E. T., 1992. Statistical Methods for Survival Data Analysis, 2nd Edition. Wiley, New York.

Lee, L., Helsel, D. R., in press. Statistical analysis of water-quality data containing multiple detection limits: S-language software for regression on order statistics. Computers and Geosciences.

R Development Core Team, 2005. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL `http://www.r-project.org`

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., 2003. USGS library for S-PLUS for Windows – Release 2.1. Open-File Report 03-357, U.S. Geological Survey, 50p.

Williamson, S., Booth, N., May 17 – 20 2004. USGS National Water Quality Data and Maps on the Web. In: Abstracts with programs. National Water Quality Monitoring Conference, p. 65.

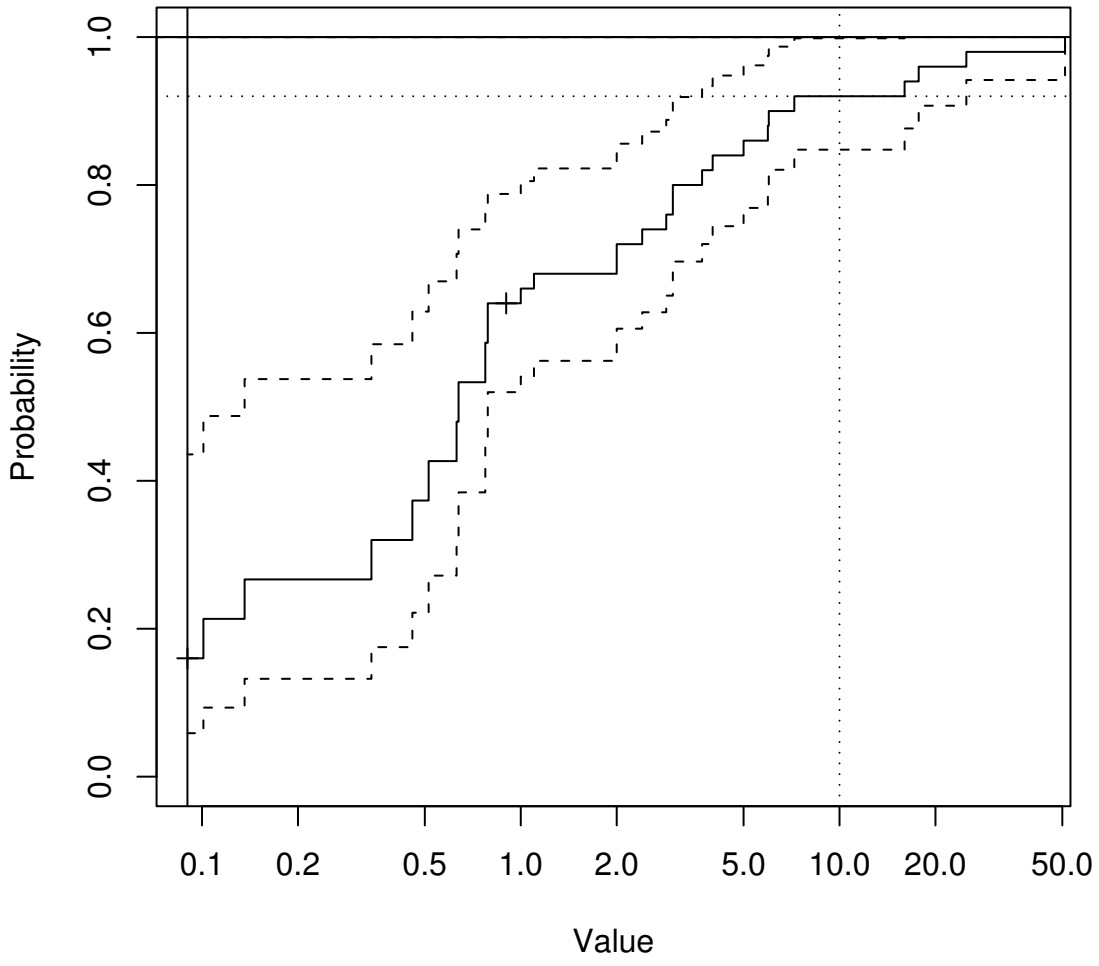

Fig. 1. Empirical Cumulative Distribution Function for multiply-censored data. Dashed (stepped) lines are confidence limits. Dotted vertical and horizontal lines are concentration and associated probability of a hypothetical water-quality criterion at 10 $\mu g/L$.

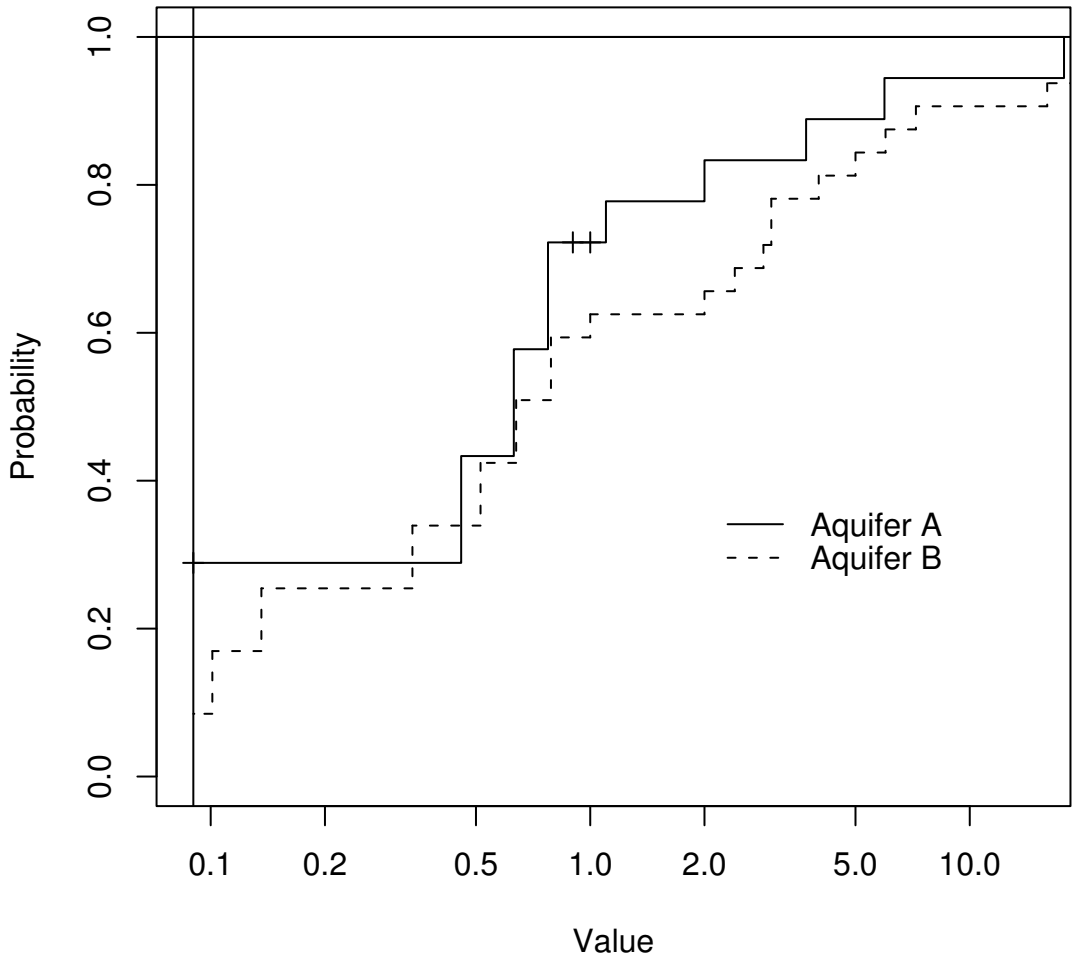

Fig. 2. Empirical Cumulative Distribution Functions for multiply-censored data. Solid and dashed lines are different ECDFs resulting from factoring observations into groups.

Function Name	Purpose
`Cen`	Creates a censored object for use in formulas
`cenfit`	Constructs an ECDF
`cendiff`	Tests for differences between ECDFs
`plot`	Produces a step-function plot of an ECDF
`mean`	Returns the mean of an ECDF
`sd`	Returns the standard deviation of an ECDF
`median`	Returns the median of an ECDF
`quantile`	Returns quantile estimates of an ECDF
`predict`	Predicts the quantiles of a value
`pexceed`	Predicts the exceedance probability of a value

Table 1
NADA for R library functions for the creation and manipulation of empirical distribution functions (ECDF) of multiply-censored data. Detailed information on individual functions is available through the on-line help system.