Plotting Censored Data

Dave Lorenz

July 27, 2015

These examples demonstrate some of the plotting capabilities in the `smwrQW` package. The examples in this vignette use data from the `NADA` package, so that similarities and differences between the assumptions made in Helsel (2012) and the assumptions made in `smwrQW` can be highlighted—as Helsel (2012) states in section 5.5.1 "Know the procedures used by your software." The examples use the `as.lcens` function to convert data to a valid format for `smwrQW` functions; the class "lcens" is most appropriate for these data as they are only left-censored and have only the value and an indicator of censoring. The functions demonstrated in these examples will also accept data of class "qw." The R code following this paragraph gets the data and converts the columns named "June" and "Sept" to class "lcens."

```
> # Load the smwrQW package
> library(smwrQW)
> # And get the data
> data(Atra, package="NADA")
> data(AtraAlt, package="NADA")
> # Convert the data
> Atra <- transform(Atra, June=as.lcens(June, censor.codes=JuneCen),
+   Sept=as.lcens(Sept, censor.codes=SeptCen))
> AtraAlt <- transform(AtraAlt, June=as.lcens(June, censor.codes=JuneCen),
+   Sept=as.lcens(Sept, censor.codes=SeptCen))
```

The examples in this vignette present few options used in the graphics functions. More complete variations can be seen in the vignettes for the `smwrGraphs` package. **NOTE:** to use any of the high-level graphics functions, you must first call a function to set up the graphics environment such as `setPage` or `setPDF`, but these are not desmonstrated here due to the graphics set up function required for the vignette, `setSweave`.

1 Box Plots

The `boxPlot` function has four types of box plots: truncated, simple, Tukey, and extended. The truncated box plot is the default type with truncation at the 10 and 90 percentiles. The other types can be created with a simple revision to the `Box` argument. The simple box plot extends the whiskers to the minimum and maximum values. For the Tukey box plot, `type="tukey"`, the whiskers are extended to the observed value that is within 1.5 times the interquartile range above or below the upper or lower quartile and observed values outside of that range are plotted as individual symbols. The extended box plot is a variation on the simple box plot where all observed values outside of the upper and lower percentiles are plotted as individual symbols. For the default 10 and 90 percentile limits, no more than 10 percent of the total number of observed value will be individually plotted. Note that the y-axis range is set by the range of the data and not the range of the box plot. The four types of box plots are discussed further in the `BoxPlots` vignette in the `smwrGraphs` package.

In addition to the types of box plots, there are also two styles for censored data, censored and estimated. These are controlled by the `censorstyle` component of the `Box` argument. The censored style option in `boxPlot` does not incorporate information from censored values; it does not set single reporting limit values to any arbitrary value less than the reporting level or use flipped Kaplan-Meier for multiple reporting limits, instead no statistics are computed for any value less than the maximum reporting level. This feature protects against arbitrary computation of the quartiles and prevents the incorrect computation of the interquartile range. Therefore, the Tukey type box plot cannot be selected when the style is censored. The estimated style uses regression on order statistics (ROS; Helsel 2012) to impute values for censored data. If the `yaxis.log` argument to `boxPlot` is `TRUE`, then ROS is performed on the log-transformed values, otherwise ROS is performed on the raw values. Any of the four type of box plots can be selected when the style is estimated.

The example graphs below illustrate the censored and estimated styles. Note that the y-axis is on a log-scale because most water-quality data are assumed to be log-normally distributed, so the default for the `yaxis.log` argument for censored data is `TRUE`. The red bar in the third graph indicates that the ROS method computed at least one imputed value that was greater than the reporting limit.

```
> # setSweave is a specialized function that sets up the graphics page for
> # Sweave scripts. It should be replaced by a call to setPage or setPDF
> # in a regular script.
> setSweave("graph01", 6 ,6)
> # Set layout for 4 graphs
> AA.lo <- setLayout(num.cols=2, num.rows=2)
> # Set up and graph the data
> AA.gr <- setGraph(1, AA.lo)
> with(Atra, boxPlot(June, Sept, Box=list(type="simple")))
> addTitle("Censored Box Plot, Single Reporting Level", Bold=FALSE)
> AA.gr <- setGraph(2, AA.lo)
> with(AtraAlt, boxPlot(June, Sept, Box=list(type="simple")))
> addTitle("Censored Box Plot, Multiple Reporting Level", Bold=FALSE)
> AA.gr <- setGraph(3, AA.lo)
> with(Atra, boxPlot(June, Sept,
+   Box=list(type="simple", censorstyle="estimated")))
> addTitle("Estimated Box Plot, Single Reporting Level", Bold=FALSE)
> AA.gr <- setGraph(4, AA.lo)
```

```
> with(AtraAlt, boxPlot(June, Sept,
+   Box=list(type="simple", censorstyle="estimated")))
> addTitle("Estimated Box Plot, Multiple Reporting Level", Bold=FALSE)
> # Required call to close PDF output graphics
> graphics.off()
```

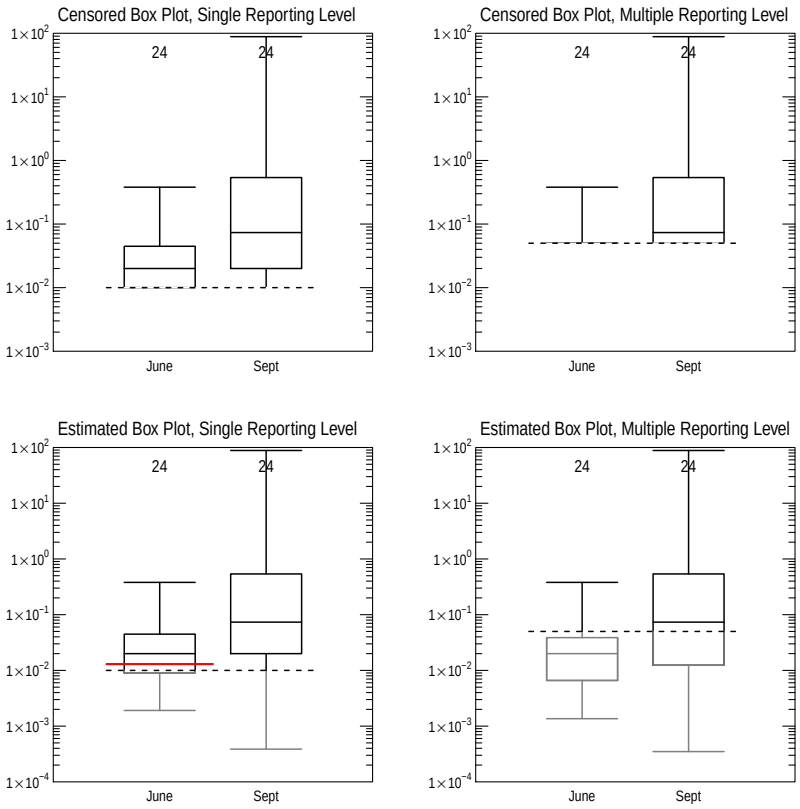

Figure 1. Censored and estimated style box plots.

2 Empirical Distribution Function

The empirical distribution function describes the cumulative distribution given an empirical measure of the observations. The empirical measure is the step function that increases by 1/n for each of the sorted observations. This graph is often used to explore or describe the observations without describing them in the context of any distribution, inferences by interpolations through various plotting positions, or making inferences about the population from which the sample was taken. For multiple reporting limits, the flipped Kaplan-Meier method is used to estimate the distribution function.

The example graphs below illustrates the empirical distribution function graph for the June atrazine data using an untransformed and log axis.

```
> setSweave("graph02", 6 ,6)
> # Set layout for 2 graphs
> AA.lo <- setLayout(num.rows=2)
> # Create the graphs
> setGraph(1, AA.lo)
> with(Atra, ecdfPlot(June, xtitle="June Atrazine"))
> setGraph(2, AA.lo)
> with(Atra, ecdfPlot(June, xtitle="June Atrazine", xaxis.log=FALSE))
> graphics.off()
```

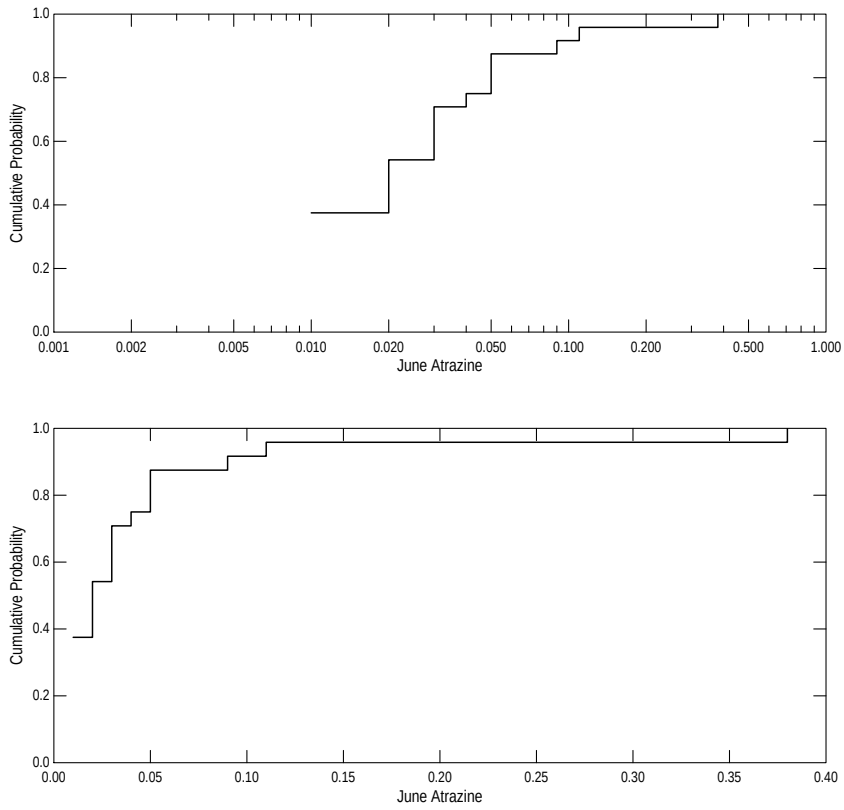

Figure 2. The empirical distribution function graph.

3 Probability Plot

The probability plot graphs the observed data against the normal or other specified distribution, often as a check on the assumed distribution. The default symbolization scheme for `probPlot` is to plot the complete data as solid circles at their specified plotting position and the censored data as open circles, distributed over the range appropriate for that censoring level. The default axis setting for `probPlot` is to log-transform the y-axis and use the normal distribution for the x-axis, producing the equivalent of a log-normal distribution. The default value for a in the plotting position equation, the `alpha` argument in `probPlot` is 0.4, the Cunnane plotting position that was selected by Helsel and Hirsch (2002) as having good general plotting properties.

This example demonstrates how to create a probability plot with specific distributional parameters. Two graphs are created one using "log ROS" and the other using "log MLE." Note that the difference in the computations between "log ROS" and "log MLE" is that the latter adjusts the distributional parameters so that quantiles associated with the values less than 0.01 are less than 0.01, as indicated by the probability value of the line equal to 0.01!

```
> setSweave("graph03", 6 ,6)
> # Set layout for 2 graphs
> AA.lo <- setLayout(num.rows=2)
> # Create the graphs
> setGraph(1, AA.lo)
> # Compute the statistics using "log ROS"
> AA.st <-  with(Atra, censStats(June, method="log ROS"))
> print(AA.st)

     mean std. dev.
  0.04313   0.07657
Statistics for the log transforms:
     mean std. dev.
    -3.88     1.208

> with(Atra, probPlot(June, ytitle="June Atrazine", ylabels=5,
+   mean=AA.st$meanlog, sd=AA.st$sdlog))
> addTitle("Statistics using log ROS", Bold=FALSE)
> setGraph(2, AA.lo)
> # Compute the statistics using "log MLE"
> AA.st <-  with(Atra, censStats(June, method="log MLE"))
> print(AA.st)

     mean std. dev.
  0.04471    0.1054
Statistics for the log transforms:
     mean std. dev.
   -4.047     1.371

> with(Atra, probPlot(June, ytitle="June Atrazine",  ylabels=5,
+   mean=AA.st$meanlog, sd=AA.st$sdlog))
> addTitle("Statistics using log MLE", Bold=FALSE)
> graphics.off()
```

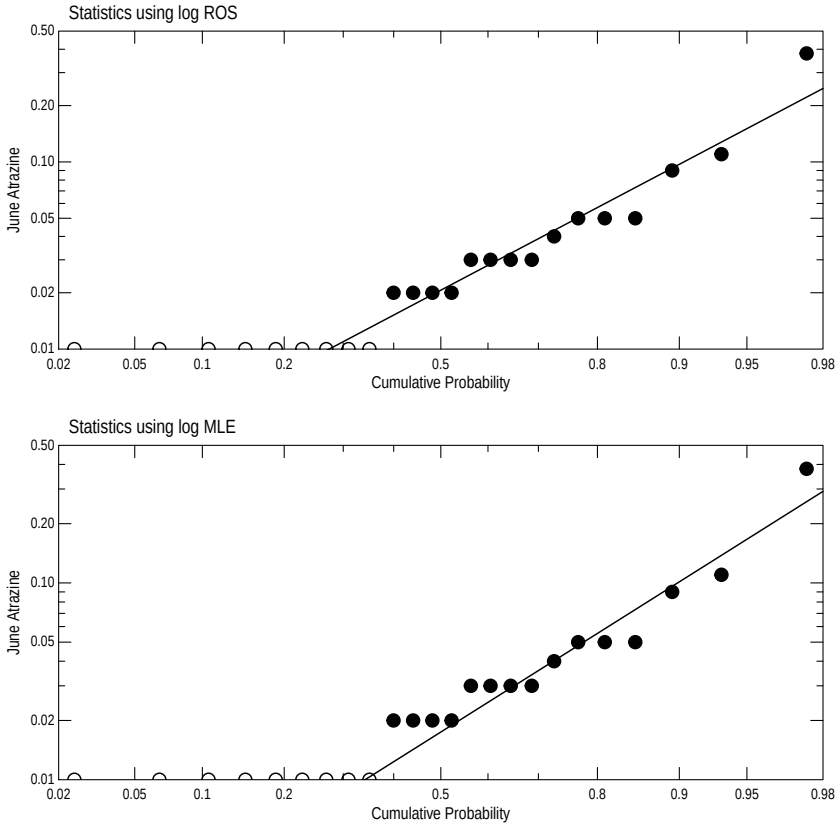

Figure 3. Example probability plots.

4 The Q-Q and Q-normal Plots

These plots compare samples from one populations to samples from another population, the Q-Q plot or to the normal distribution, the Q-normal plot. Both are created using the `qqPlot` function. The difference between the Q-normal plot and the probability plot is that the x-axis is normal quantiles rather than probabilities and the Q-normal plot uses only the standard normal distribution.

The first example creates the Q-Q plot. It compares the atrazine data for June with the altered atrazine data for June. The Q-Q plot generated by `qqPlot` uses "log ROS" or "ROS," depending on whether the axis is log-transformed or not to create a complete picture of the distribution. The gray line in figure 4 is the 1:1 line, the black line is the best-fit line.

```
> setSweave("graph04", 6 ,6)
> # The Q-Q plot
> qqPlot(Atra$June, AtraAlt$June,
+   xtitle="Atrazine", ytitle="Altered Atrazine",
+   xaxis.log=TRUE, yaxis.log=TRUE)
> graphics.off()
```

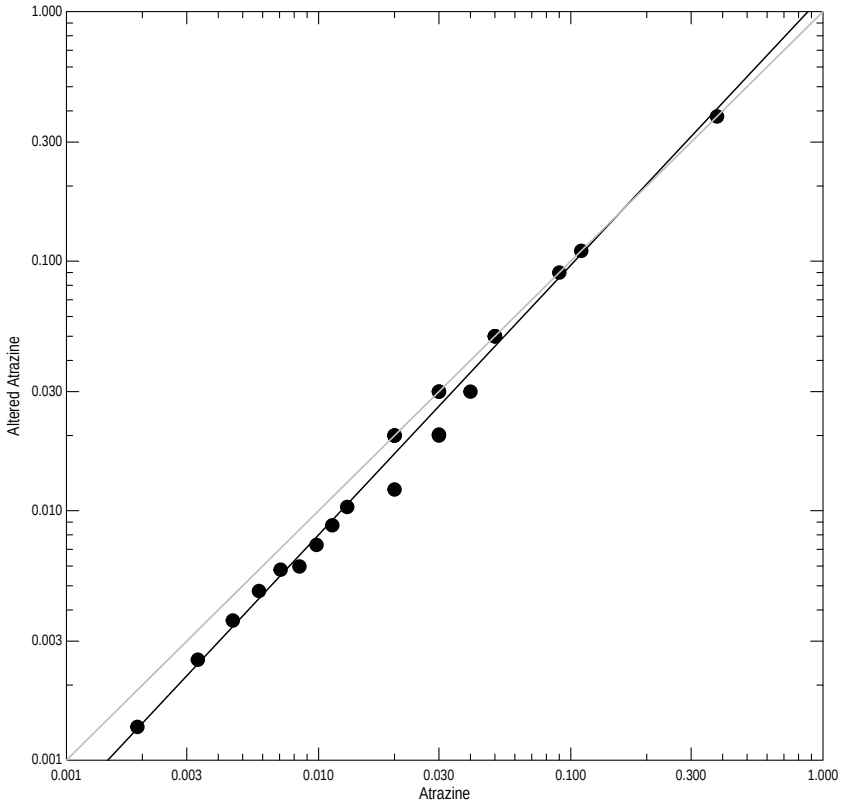

Figure 4. The Q-Q plot.

The second example shows the Q-normal plot for the June Atrazine data. The Q-normal plot created by the `qqPlot` function has added features that illustrate the imputed values from the "log ROS" or "ROS" method, depending on whether the y-axis is log-transformed or not. As with the probability plot, the complete values are solid circles and censored values a re open circles, the q-normal plot includes the imputed values from the censored values projected onto the line computed by "log ROS" or "ROS" depending on whether `yaxis.log` is set to `TRUE` or `FALSE`.

```
> setSweave("graph05", 6 ,6)
> # The Q-normal plot
> qqPlot(Atra$June, ytitle="Atrazine", yaxis.log=TRUE)
> graphics.off()
```

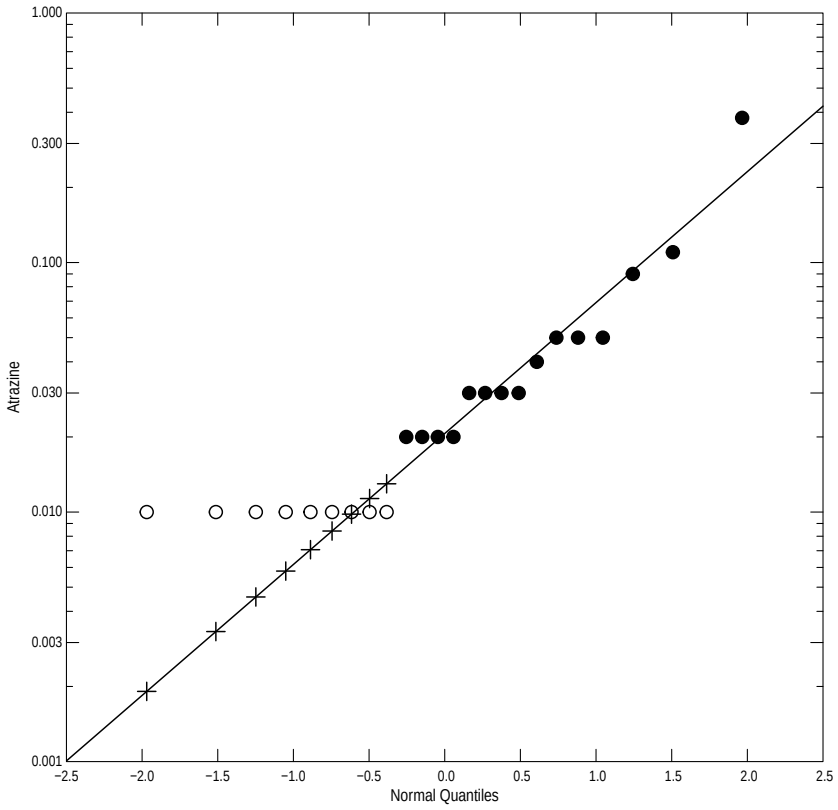

Figure 5. The Q-normal plot.

5 Scatter Plots

Scatter plots show the relations between two variables. The data are plotted as individual points to show any pattern in the relation between the two variables. The current version of the `smwrQW` package has 3 functions that create scatter plots, `xyPlot` for numeric x-axis data and censored y-axis data, `timePlot` for date x-axis data and censored y-axis data, and `dotPlot`, for censored x-axis data and categorical y-axis data. The example graph, fig. 6, demonstrates only the `xyPlot` function and uses the sequential position of the data in the dataset Atra as the x-axis data.

```
> setSweave("graph06", 6 ,6)
> # The x-axis data must be numeric, and add droplines (bar)
> with(Atra, xyPlot(seq(1, nrow(Atra), by=1), June,
+   xtitle="Sequential Number", ytitle="Atrazine in June",
+   Censored=list(bar=TRUE)))
> graphics.off()
```

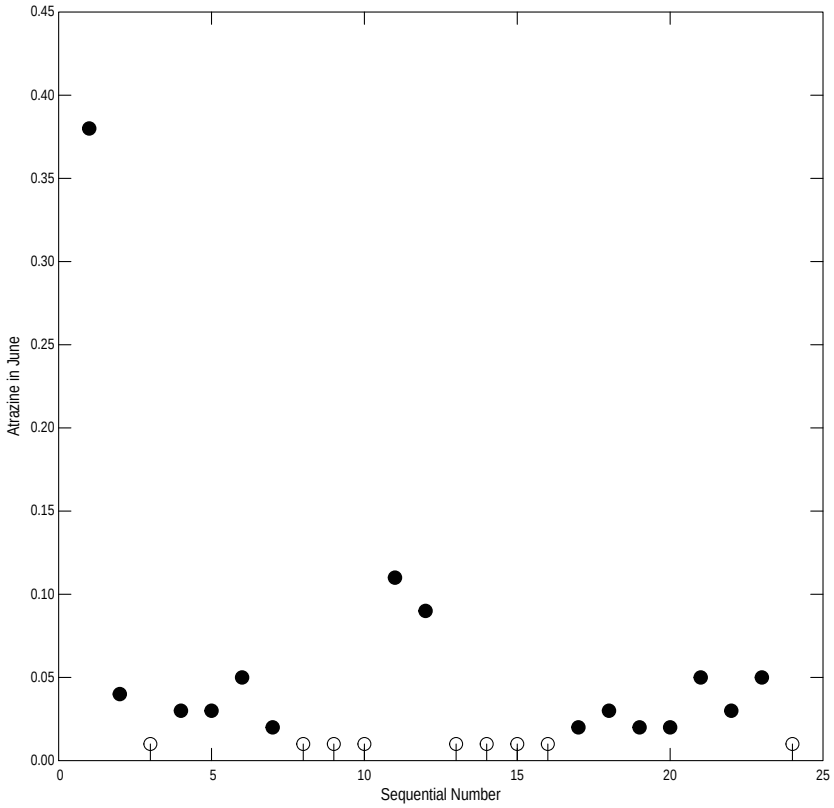

Figure 6. A scatter plot with censored data.

References

[1] Helsel, D.R. 2012, Statistics for Censored Environmental Data Using Minitab and R: New York, Wiley, 324 p.

[2] Helsel, D.R., and Hirsch, R.M., 2002, Statistical methods in water resources: U.S. Geological Survey Techniques of Water-Resources Investigations, book 4, chap. A3, 522 p.