

Guidelines for computing summary statistics for data-sets containing non-detects

C. Huston and E. Juarez-Colunga
Department of Statistics and Actuarial Science, Simon Fraser University
chuston@sfu.ca and ejuarezc@sfu.ca

Written for the Bulkley Valley Research Center
with assistance from
the B.C. Ministry of Environment

January 19, 2009

Contents

1	Introduction	11
2	Overview of Statistical Methods	13
2.1	General guidelines	14
2.2	Survival analysis methods in environmental data	16
3	Detection Limit and Quantitation Limit	18
3.1	Computing the detection limit	19
3.2	Setting the quantitation limit	20
3.3	Options for the censoring limit	21
4	Data Storage and Software	23
4.1	Negative numbers	23
4.2	Interval endpoints	24
4.3	Indicator Variables	24
4.4	Software	25
5	Plotting Methods	27
5.1	Box plots	28
5.1.1	Constructing Boxplots	28
5.1.2	Interpreting boxplots	28
5.1.3	Box plots using JMP	30

5.1.4	Box Plots using R	32
5.1.5	Summary of boxplot	34
5.2	Probability plotting	35
5.2.1	Probability plotting using JMP	35
5.2.2	Lognormal probability plotting using R	38
5.2.3	Probability plotting in R if the data are not lognormal	40
6	Calculating Statistics	41
6.1	Summarizing nondetects	42
6.1.1	Summarizing nondetects using JMP	42
6.1.2	Summarizing nondetects using R	42
6.2	Mean and median	44
6.3	Substitution	45
6.3.1	Substitution example	45
6.3.2	Assumptions in substitution	47
6.3.3	Consequences and risks of substitution	48
6.4	Nonparametric methods: Kaplan-Meier	48
6.4.1	Statistics computed based on Kaplan-Meier	49
6.4.2	Assumptions	50
6.4.3	Risks	50
6.5	Parametric methods	51
6.5.1	The lognormal distribution	51
6.5.2	Assumptions	53
6.5.3	Risks	53
6.5.4	Summary statistics using JMP	53
6.5.5	Summary statistics using R	56
6.5.6	Notes on the lognormal method.	59

6.6	Robust Regression on Order Statistics (ROS)	59
6.6.1	Assumptions	59
6.6.2	Risks	60
6.6.3	Notes on ROS	60
6.6.4	ROS using R	60
6.6.5	Computing confidence interval estimates using the bootstrap method	62
7	Introduction to the Second Half: Comparing, correlating, regressing, and other assorted topics	66
8	Brief Review of Some Key Concepts	68
8.1	Detection, Quantitation, and Censoring	68
8.2	Data Transformations	68
8.3	Statistical Concepts	69
8.3.1	Measures of Centre	69
8.3.2	Measures of Spread	70
8.3.3	Communicating Statistical Estimates	70
9	Comparing Centers of Two Independent Populations	72
9.0.1	The Data Set	73
9.0.2	Preliminary Data Inference: Graphing	74
9.1	What Not To Do: T-Tests and Substitution	76
9.2	What To Do: Parametric Methods	78
9.3	What To Do: Non-Parametric Methods	83
9.3.1	Non-parametric tests when there is only one detection limit	84
9.3.2	Non-parametric tests when there are multiple detection limits	85
10	Comparisons Between Paired Observations	87
10.0.1	Paired Data Example	88

10.0.2 Preliminary Inference: Graphing and Data Transformation	89
10.1 Parametric Tests for Paired Data	92
10.2 Non-Parametric Testing For Paired Data	95
10.3 Comparing Data to a Standard	97
11 Comparisons Between Multiple Groups	98
11.0.1 Example Data	99
11.0.2 Preliminary Data Inference: Graphing	99
11.1 What Not To Do	101
11.2 Parametric Methods for Multi-Group Data	101
11.2.1 Confidence Intervals	102
11.2.2 Changing The Reference Category	103
11.2.3 Dangers of Multiple Comparisons	104
11.2.4 Model Assessment	104
11.2.5 Interpretation When Data Are Log Transformed	105
11.3 Non-Parametric Methods	105
11.3.1 Performing The Test	106
11.3.2 Followup Testing	106
12 Trends: Correlation and Regression	108
12.0.1 Conceptual Framework	108
12.0.2 Example Data	110
12.0.3 Preliminary Data Inference: Graphing	111
12.1 Maximum Likelihood Estimation	112
12.2 Non-Parametric Approaches	117
12.3 Comparing Maximum Likelihood and Non-Parametric Results: Some Cautions!! . .	118
13 Further Topics in Regression: Seasonal Trend Analysis	121

13.1	Seasonal Kendall Test	121
13.2	Multiple Regression With Censoring	122
13.2.1	Example Data	123
13.2.2	Model Fitting	124
13.2.3	Model Interpretation	125
13.2.4	Testing, Confidence Intervals, and Model Checking	127
14	Things to do when censored observations make up more than 50% of the data	128
14.1	Tests Including Covariate Information	129
14.1.1	Example Data	130
14.1.2	Performing and interpreting analysis	130
14.1.3	Goodness of fit testing	132
A	Data Sets	134
A.1	Savona data	134
A.2	Retena data	134
A.3	Arsenic data	134
B	Transferring Data from Excel to JMP	137
B.1	When ‘<’ and the observations are in separate columns	137
B.2	When ‘<’ and the observations are in the same column	139
C	Transferring Data from Excel to R	141
C.1	Reading the formatted data in R	141
C.2	Formatting the data in R	141
D	Starting with R	143
E	Bootstrap	152

F	Probability plotting for distributions other than the lognormal	153
G	Kaplan-Meier	154
G.1	Computation of the Kaplan-Meier estimator	154
G.2	Kaplan Meier estimation using JMP	155
G.3	Kaplan Meier estimation using R	157
G.4	Computing confidence interval estimates using the B-C inverted sign method	159
G.4.1	B-C inverted sign method	159
G.4.2	Using the B-C inverted sign method in computing interval estimates	160
H	How to install and use an R library when you don't have administrator privileges	161
I	R code to run function DiffCI	163
J	R function for differences in paired observations	165
K	Code for MLE test of Paired Differences	167
L	Non-parametric paired data test functions	169
M	Toboggan Creek Sampling Region	172
N	Vanadium Data Set	173
O	Simulated Seasonal Water Quality Data	175

List of Tables

2.1	Guidelines to determine which method to use to estimate summary statistics	14
4.1	Values of <0.001, <0.002 and 0.003 represented using a “negative” number.	23
4.2	Values of <0.001, <0.002 and 0.003 represented using interval endpoints.	24
4.3	Values of <0.001, <0.002 and 0.003 represented using a censored indicator variable. A 1 indicates a censored observation, and 0 an observed value.	25
4.4	Values of <0.001, <0.002 and 0.003 represented using TRUE and FALSE. A TRUE indicates a censored observation.	25
5.1	Concentrations of orthophosphate measured in the Thompson River at Savona. A ‘0’ or ‘FALSE’ denotes an observed value, and a ‘1’ or ‘TRUE’ denotes a censored value.	27
6.1	Summary statistics for an arsenic sample taken from Oahu after substituting the censored observations by 0, 1/2 CL, and CL.	46
9.1	Concentrations of copper and zinc in the San Joaquin Valley	74
9.2	Toy example for Gehan non parametric test	86
10.1	Groundwater concentrations of atrazine in June and September	88
10.2	Groundwater concentrations of atrazine in June and September, paired data format	89
10.3	Sample of the output returned by the makePaired(..) function	90
10.4	Table including sign of difference evaluations for the atrazine data	95
10.5	Groundwater concentrations of atrazine in June and September	97
11.1	Vanadium concentrations ($\mu\text{g/L}$) at different sampling locations along Toboggan Creek	99

12.1	Iron (Fe) concentrations ($\mu\text{g}/\text{L}$)for different years in the Brazos River	111
13.1	Seasonal water quality data	123
14.1	TCE Concentrations ($\mu\text{g}/\text{L}$)along with potential explanatory variables	130
A.1	Concentrations of orthophosphate in the Thompson River at Savona. A ‘1’ denotes a censored observation, and a ‘0’ an observed value.	135
A.2	Concentrations of arsenic measured in streamwaters at Oahu, Hawaii. A ‘0’ denotes an observed value, and a ‘1’ denotes a censored value; at the value presented.	136
G.1	Example computations of the Kaplan-Meier estimator for the savona data.	155
G.2	Table of summary statistics using Kaplan-Meier for the savona data.	157

List of Figures

2.1	Diagram of the methods for estimating summary statistics with data containing non-detects. ¹ For MLE-Lognormal method, see Section 6.5; ² for ROS method, see Section 6.6; ³ and for Substitution, see Section 6.3. p_{censor} denotes the percentage of censoring and n_{sample} the sample size.	17
3.1	Illustration of censoring types. The dashed line indicates the censoring region for an observation in the three graphs. CL denotes the censoring limit(s).	19
3.2	Illustrative plot where the DL is fixed at 3.14 sd from the putative zero concentration.	20
3.3	Illustrative plot where Reporting limits determined from a sample of size 7. The DL is fixed at 3.14 sd from zero, and the QL is fixed at 10 sd from zero.	21
5.1	Boxplot showing the features of a boxplot.	29
5.2	Boxplot for savona data.	33
5.3	Probability plot for the savona data.	38
5.4	Probability plot for lognormal distribution with savona data. The vertical axis is given in log scale.	39
6.1	Histogram of data skewed to the right.	44
6.2	Boxplots of arsenic concentrations at streamwaters in Oahu, Hawaii after substituting the censoring limits for 0, 1/2 CL and CL.	46
6.3	Boxplot of arsenic concentration at streamwaters in Oahu, Hawaii.	47
6.4	Survivor curve of the Kaplan-Meier estimator for the concentration of orthophosphate in savona data. The data are transformed by subtracting each observed value from 0.012.	50
6.5	Illustration of quantiles based on the Kaplan-Meier estimate of the survivor curve for the savona data.	50

6.6	Histograms of a lognormal and normal distributions. The natural logarithm of log-normal data gives a normal distribution.	52
6.7	Probability plot for savona data, assuming a lognormal distribution.	57
9.1	Side by side boxplots for zinc data	75
9.2	Side by side boxplots for log zinc Data	75
9.3	Illustration of the difference between handling censored data with substitution vs. accounting for it directly in estimation	79
9.4	Normal probability plot of residuals from cenmle fit of the CuZn data	83
10.1	Atrazine difference boxplot	91
10.2	Log-atrazine difference boxplot	92
10.3	Normal Probability Plot of Residuals From Parametric Atrazine Analysis	94
11.1	Side by side boxplots for vanadium data	100
11.2	Normal Probability Plot For Vanadium Residuals	105
12.1	Lines with negative and positive slopes	109
12.2	Representation of a regression line	110
12.3	Censored scatterplot of the Brazos River Iron concentration data	112
12.4	Normal probability plot of residuals from censored regression	115
12.5	Residual plot from censored regression	115
12.6	Residual plot from censored lognormal regression	116
12.7	Comparison of trend lines estimated by maximum likelihood and non-parametric methods	119
13.1	Censored scatter and line plot of data showing a high flow/low flow seasonal trend	124
13.2	Plot of data showing a high flow/low flow points and censored regression lines for each flow season	126
G.1	Cumulative distribution function based on the Kaplan-Meier estimate for the savona data.	158

Chapter 1

Introduction

As part of its responsibilities, the BC Ministry of Environment monitors water quality in the province's streams, rivers, and lakes. Often, it is necessary to compile statistics involving concentrations of contaminants or other compounds.

Quite often the instruments used cannot measure concentrations below certain values. These observations are called *non-detects* or *less thans*. However, non-detects pose a difficulty when it is necessary to compute statistical measurements such as the mean, the median, and the standard deviation for a data set. The way non-detects are handled can affect the quality of any statistics generated.

Non-detects, or censored data are found in many fields such as medicine, engineering, biology, and environmetrics. In such fields, it is often the case that the measurements of interest are below some threshold. Dealing with non-detects is a significant issue and statistical tools using *survival* or *reliability* methods have been developed.

Basically, there are three approaches for treating data containing censored values: 1. substitution, which gives poor results and therefore, is not recommended in the literature; 2. maximum likelihood estimation, which requires an assumption of some distributional form; and 3. and nonparametric methods which assess the shape of the data based on observed percentiles rather than a strict distributional form.

This document provides guidance on how to record censor data, and on when and how to use certain analysis methods when the percentage of censored observations is less than 50%. The methods presented in this document are: 1. substitution; 2. Kaplan-Meier, as part of nonparametric methods; 3. lognormal model based on maximum likelihood estimation; 4. and robust regression on order statistics, which is a semiparametric method.

Statistical software suitable for survival or reliability analysis is available for dealing with censored data. This software has been widely used in medical and engineering environments. In this document, methods are illustrated with both R and JMP software packages, when possible. JMP often requires some intermediate steps to obtain summary statistics with most of the methods described in this document. R, with the NADA package is usually straightforward. The package NADA was developed specifically for computing statistics with non-detects in environmental data based on

Helsel (2005b).

The data used to illustrate the methods described for computing summary statistics for non-detects are either simulated or based on information acquired from the B.C. Ministry of Environment.

This document is strongly based on the book *Nondetects And Data Analysis* written by Dennis R. Helsel in 2005 (Helsel, 2005b).

Chapter 2

Overview of Statistical Methods

Often water quality data contains some values known to be below a certain threshold. These numbers below the threshold are called left censored data.

The purpose of taking a sample is usually to describe characteristics of a population. Statistics computed on a sample estimate characteristics of the population. However, when censored data is involved, how the computation should be conducted is not straightforward.

There are four major approaches for obtaining descriptive statistics with censored data in the environmental field: substitution, maximum likelihood, nonparametric methods, and semi-parametric methods.

The substitution approach consists of substituting the non-detects with a single value. Common choices used to replace nondetect values are 0, 1/2 the limit of detection, or the limit itself. Frequently, analysts attempt to be conservative in their estimates by substituting the censoring limit in order to protect against the worst case scenario. However, any other analysis involving results based on the substitution will be wrong.

Although substitution is an easy method, it has no theoretical basis, has been shown to give poor results in simulation studies.

One set of statistical methods that are commonly used to estimate parameters of interest, such as average concentration, are maximum likelihood methods. Maximum likelihood, or parametric methods assume that the data will be similar to a certain known shape (think of a bell curve). Based both on our shape assumption, and on information contained in the data we can estimate parameters of interest. This is a straightforward method of obtaining estimates that will be discussed in greater detail in upcoming chapters.

In order to assume that the data follow a specific distribution, it is necessary to have some evidence of this fact. In some cases, however, approximating a distribution is questionable. In this case, another approach should be taken. Nonparametric are likely to be more suitable.

Nonparametric methods do not assume that the data follow any known form. They estimate an empirical function to compute the summary statistics of the data.

An alternative approach to parametric and nonparametric methods is the robust regression on order statistics (ROS) approach, which is called a semiparametric method. Robust ROS methods are based on a combination of ideas from both parametric and non-parametric methods.

2.1 General guidelines

It is difficult to give general guidelines on when to apply certain statistical methods since the appropriateness not only depends on the sample size and the percentage of censoring, but also on the validity of the assumptions made about the data. Figure 2.1 and Table 2.1 give guidelines for the method to use according to the percentage of censoring and the sample size. Only cases in which less than half the data are censored are considered. Once a method is chosen based on the amount of censoring and the sample size, the assumptions must be checked in order to assure the validity of the results.

Table 2.1: Guidelines to determine which method to use to estimate summary statistics

% censoring (p)	Sample size (n)	Method
$p < 15$		Substitution (Section 6.3)
$0 < p < 50$	Small ($n < 50$)	Kaplan-Meier (Section 6.4)
$0 < p < 50$	Small ($n < 50$)	Robust ROS (Section 6.6)
$0 < p < 50$	Large ($n > 50$)	MLE-Lognormal (Section 6.5)

Even though substitution is not generally recommended in the literature, when the percentage of censoring is less than 15% (any sample size) few authors suggest the use of this method. More appropriately, one of the other three methods in Table 2.1 should be adopted depending on the sample size and data attributes.

Which method to use depends on the sample size and the amount of censoring. When the censoring percentage is less than 50% and the sample size is small, either Kaplan-Meier or robust Regression on Order Statistics (ROS) works reasonably well. With a larger sample size, but a similar censoring percentage, a parametric lognormal maximum likelihood method generally works better.

Between 30 and 50 observations are generally considered as the cutpoint between small and large sample methods. The more conservative limit is 50. A conservative limit is appropriate when outliers are present or the distributional assumption is questionable. The 50 observations limit can be relaxed if the data meets all the assumptions and has no outliers.

When dealing with small sample sizes, the choice between Kaplan-Meier and ROS depends on whether the median or the mean will be used as a measure of centre. In water quality data, the mean computed using the Kaplan-Meier method when left censoring is present is biased upward because some of the smallest observations are censored. In the presence of censoring it would be better to use the median for estimates based on the Kaplan-Meier method. For a small sample size, the median is the best choice of a measure of center based on Kaplan-Meier methods. An equally effective alternative is the mean calculated using the Robust ROS method.

As previously stated, assumptions should be checked in all cases before applying any method to compute summary statistics.

2.2 Survival analysis methods in environmental data

Survival analysis software is often used to analyze data with censored observations. Survival analysis can be performed using most statistical software packages including R, JMP, SAS, etc. Survival analysis originated for use in health care applications to measure quantities of interest, such as the median survival time for new health care treatments. Despite its origins in health care, survival analysis techniques are now used in many contexts where censored observations exist. This makes it appropriate for use in water quality data in environmental monitoring.

The data found in the environmental field are usually regarded as left censored because values are known to be *below* a certain threshold. However, survival analysis software is commonly available for right censored data only. As a result, an intermediate step has to be taken and left censored data transformed.

Once the data are transformed, all the statistics of interest can be computed using standard survival analysis statistical software. The results of the statistics can then be back transformed to the original units for interpretation. Examples of these analysis techniques will be given in upcoming chapters.

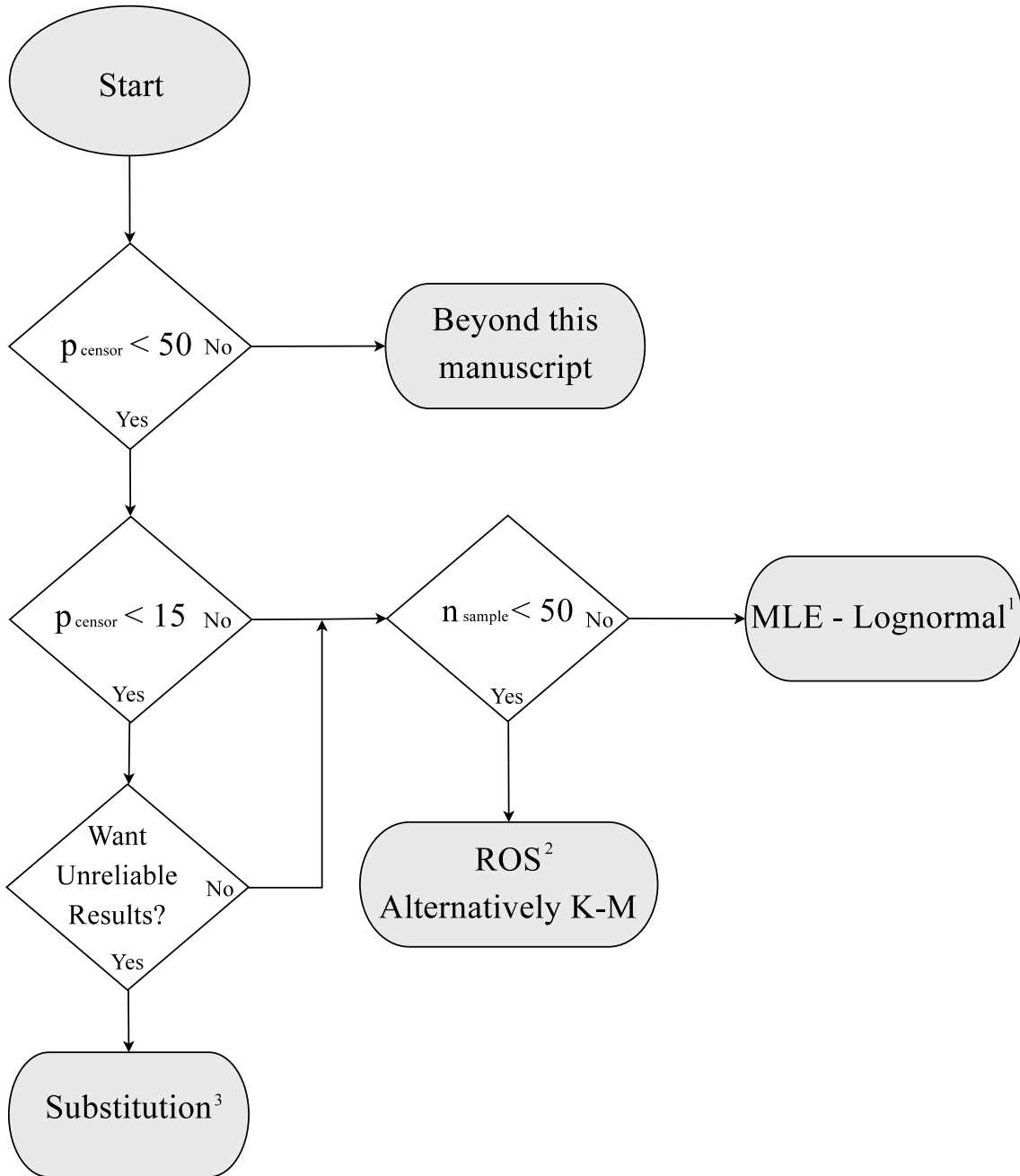


Fig. 2.1: Diagram of the methods for estimating summary statistics with data containing non-detects. ¹For MLE-Lognormal method, see Section 6.5; ²for ROS method, see Section 6.6; ³and for Substitution, see Section 6.3. p_{censor} denotes the percentage of censoring and n_{sample} the sample size.

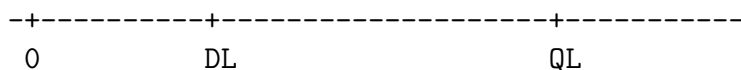
Chapter 3

Detection Limit and Quantitation Limit

When components are measured chemically, there are two quantities involved with reporting. One is the detection limit and the other is the quantitation limit.

The detection limit (DL) is a concentration benchmark in water quality data. By definition, water samples containing solutes greater than or equal to the DL are chemically distinguishable from samples containing solutes (blanks). Although values above the DL are considered to have non-zero concentrations, measurements near the DL are considered unreliable. The quantitation limit (QL) is a point at which solute concentration values can begin being reported with a high degree of confidence.

Values between zero and the detection limit cannot be distinguished, and values greater than the quantitation limit are known to be measured precisely. However, quantities between the detection limit and the quantitation limit do not have the same precision as quantities above the QL, nor are they considered equal to zero.



The above definitions of DL and QL draw on their meanings from laboratory science and chemistry. In statistics though, these types of reporting restrictions are called *censoring limits* (CL). Regardless of whether the DL or QL is chosen as the minimum margin of the data, for the remainder of this paper we will refer to both as a censoring limit. Some additional information on censoring limits (CL), and the resulting censored data is presented below.

Censoring provides partial information, i.e. data values are known to lie within a certain range. For example, in an experiment involving parasites in water, the time when each parasite dies is recorded, and the experiment ends in 15 days. At the end of 15 days, there are still 5 parasites (out of 40) alive. The value or time at which they die is not known since the experiment ends before their death. What is known is that they die after 15 days. This type of censoring is called “right censoring”. Left censoring is defined similarly, and is the most common type of censoring seen in water quality data. When observations or measurements are known to be between any two values,

the type of censoring is called “interval”. Refer to Figure 3.1 for a graphic representation of the types of censoring.

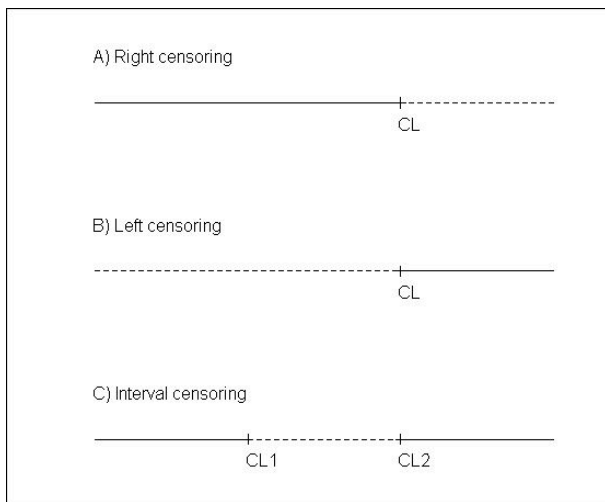


Fig. 3.1: Illustration of censoring types. The dashed line indicates the censoring region for an observation in the three graphs. CL denotes the censoring limit(s).

Statistical techniques are available to deal with censored data. The appropriate technique to use depends on the type of censoring (right, left, or interval) involved.

The type of censoring involved in any data set should be clearly identified and understood in order to choose the best statistical tool. If the quantities below the DL are considered to be censored (left censored), the observations between the DL and the QL are regarded as having the same precision as observations above the QL. On the other hand, if all quantities below the QL are considered censored, some loss of information occurs, since values are known between the DL and QL. A third possibility is to consider values between the two points as “interval” censored.

3.1 Computing the detection limit

The most common method used to determine the DL assumes that the variance in the measurement process is constant at low concentrations.

Measurements are typically assumed to follow a normal distribution around the unknown true value. For example, values below the detection limit are assumed to follow a normal curve centered around zero ¹. To estimate the standard deviation, measurements are taken at a low concentration and their standard deviation is used as a substitute.

When using a normal distribution to describe the error in measured concentrations, there is a rule of thumb that gives an idea of the distribution of the data. This rule is called the 68-95-99 rule.

¹Clearly negative numbers are not possible in measurement in concentrations, but due to measurement error we make the zero centered assumption.

This rule states that about 68% of the data are within the mean ± 1 sd; about 95% of the data are within the mean ± 2 sd; and about 99% of the data are within the mean ± 3 sd (sd denotes the sample standard deviation). Based on this rule, the detection limit can be sensibly fixed at 2 or 3 standard deviations from zero. Theoretically, this gives a false positive rate below 1% (see Figure 3.2).

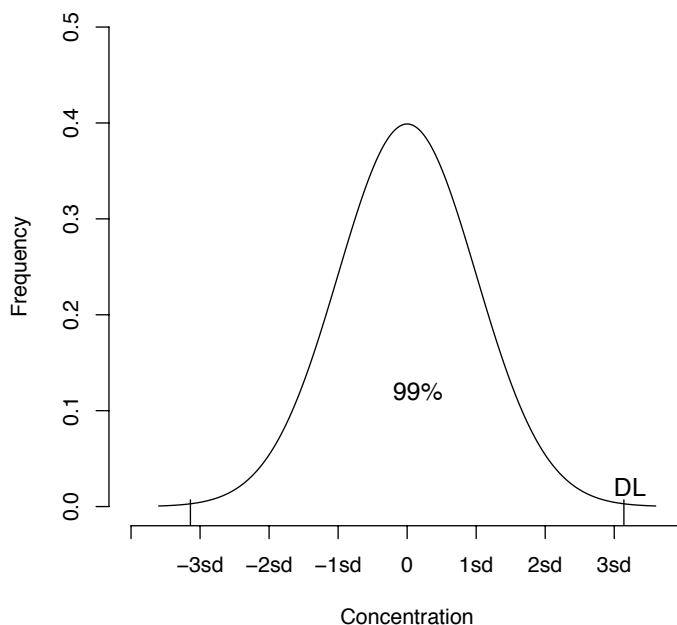


Fig. 3.2: Illustrative plot where the DL is fixed at 3.14 sd from the putative zero concentration.

Values measured below the detection limit that are truly above the detection limit are called false negatives. One role of the quantitation limit (QL), discussed in 3.2, is to protect against false negatives.

3.2 Setting the quantitation limit

The idea of a quantitation limit arises from the need to have a limit above which single measurements can be reported reliably. This also protects against false negatives.

The QL is typically defined as 10 times the standard deviation. The value of 10 is chosen because it is a number considerably larger than the detection limit.

To protect against false negatives, the QL is taken as 10 sd above 0. As seen in Figure 3.3, curve 1 and curve 2 have very little appreciable overlap indicating that the percentage of false negatives should be negligible. The QL is set far away from the DL. This ensures that we are nearly 100% certain that all values measured above the QL, even accounting for error, are above the DL. Nothing truly above the QL will be measured as a false negative.

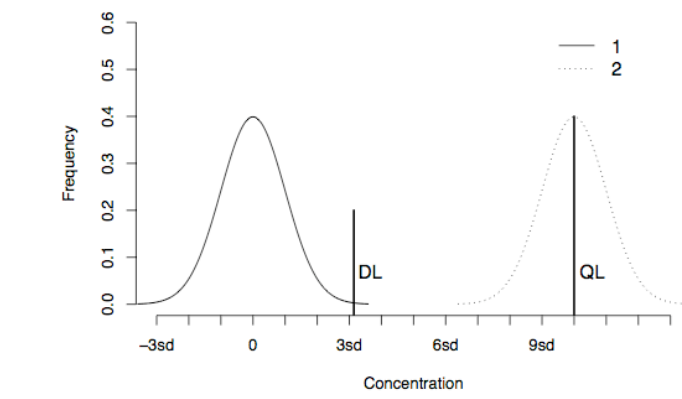


Fig. 3.3: Illustrative plot where Reporting limits determined from a sample of size 7. The DL is fixed at 3.14 sd from zero, and the QL is fixed at 10 sd from zero.

Nevertheless, there is an obvious problem in dealing with values between the two limits. These values are not as reliable as observations above the QL, but they are not small enough to be safely considered zero. Section 3.3 introduces how to handle these data using censoring techniques.

3.3 Options for the censoring limit

Some options on how to manage the information available from data including non-detects are presented in this section. The choice of method to use should be made after consultation with the laboratory scientist to ensure that their reporting methods are clearly understood. It is important to have an understanding of the relative precision of the data between the DL and the QL. As mentioned in the introduction of this chapter and Figures 3.1 there are three recognized censoring options:

1. to treat the quantitation limit as the censoring limit. Values below the QL will be censored. Some loss of information will occur, because measurements above and below the DL are treated identically even though they are different.
2. to treat the detection limit as the censoring limit. Only values below the DL will be considered censored. The assumption is that values between the DL and the QL have the same precision as those above the QL.
3. to use interval censoring methods. Values below the DL are considered as intervals of the form 0 to the DL, and values between the DL and the QL are considered as intervals from the DL to the QL. Refer again to Figure 3.1 for a visual interpretation of each type of censoring.

All three censoring methods will give estimators for percentiles and means that are unbiased. Although all three methods will measure the center of the data in roughly the correct place, they will

have different precision. Helsel (2005a) showed in simulation studies that the mean computed with method 2 has smaller variance than it does with method 1. Survival analysis software is commonly used for analyses based on these approaches.

All 3 censoring methods assume that the variability of concentration measurements is a constant. Because variability can be a function of concentration, this is not always a reasonable assumption. Methods that allow for non-constant measurement variability are sometimes necessary. One method is to treat measurement variance at low concentrations as a function of concentration. For example, if the measurement variance increases with concentration, then the measurement variance at the detection limit can be predicted using information collected at higher concentrations.

Under these assumptions, Gibbons and Coleman (2001) presented an algorithm for calculating censoring limits (DL and QL) when the measurement variance is a function of the concentration. Iteratively Weighted Regression methods achieve stable estimates of the DL and the QL based on multiple computer calculations. This method is also useful because measurements with greater certainty are given more importance in the calculations.

Chapter 4

Data Storage and Software

One of the most straightforward ways to simplify the analysis of censored data is to store the data in a manner that is compatible with available statistical software. There are three common ways of storing censored data:

1. negative numbers,
2. usage of interval endpoints,
3. an indicator variable.

A detailed discussion of the strengths and weaknesses of these methods comes in the following sections. One advantage of all these methods is that each avoids the use of alphabetic characters in numeric fields. Letters do not allow numerical computations!

A description of the available software for censored data analysis is given in Section 4.4.

4.1 Negative numbers

In this format, data less than the censor limit (CL) are represented by negative numbers. For example, Table 4.1 shows three concentration values where there is one known observation and two observations with reporting limits.

Table 4.1: Values of <0.001 , <0.002 and 0.003 represented using a “negative” number.

<hr/> <hr/> -0.001
-0.002
0.003
<hr/>

The censored values are represented with their corresponding negative number. The observed value, 0.003, is represented unchanged because it is not censored.

The use of negative numbers to represent the censored data is efficient in terms of storage space. Unfortunately, it has an important disadvantage - negative values, indicating censoring, can be mistakenly interpreted as truly negative numbers by unwary users. The mean computed using these negative numbers would be completely wrong! A second disadvantage to using negative values for censoring is that negative numbers are unable to allow for right and interval censoring ie. it is limited to left censored values only.

4.2 Interval endpoints

The most commonly used method to represent censored data is the interval endpoints method.

Two columns are needed to represent data in the interval endpoints format. The columns are used to represent the upper and lower bound of the measurement values. When the observations are known and not censored, the values in both columns are the same. When the observations are left censored, the first value is zero, and the second value is the censoring limit (CL). For example, the known observation, 0.003, is represented with two equal points, 0.003. The censored observations are denoted with a lower limit of 0 and the censor limit as the upper value. These cases are shown in 4.2.

Table 4.2: Values of <0.001, <0.002 and 0.003 represented using interval endpoints.

Start	End
0	0.001
0	0.002
0.003	0.003

This method of representing censored data is probably the easiest, the most flexible, and the least confusing. It is unambiguous because the upper and lower bounds are clearly stated. Additionally, it can incorporate different censoring intervals for each observation. Finally, there are no deceptive negative numbers that can sneak into calculations.

The main disadvantage of interval endpoints occurs when the data are not recognized as being censored. Summary statistics calculated on either of the two endpoints give wrong answers.

4.3 Indicator Variables

The indicator variable method is used to represent left or right censored data. In this method we define an indicator variable that identifies when an observation is censored. Two fields/columns are required to represent each observation. The first number can represent one of two values: The

measured number for a detected observation, or the censoring limit (CL) for a non-detect value. The second column is a set of numbers acting as indicators identifying whether a given observation is detected or censored. Typically, a 0 is used to indicate one state, and 1 the other. Another common indicator is to use the logical values TRUE or FALSE rather than 0 or 1. Tables 4.3 and 4.4 show the same three values using both 0/1 and TRUE/FALSE codings. It is always important to remember which code corresponds to which state!

Table 4.3: Values of <0.001 , <0.002 and 0.003 represented using a censored indicator variable. A 1 indicates a censored observation, and 0 an observed value.

Value	Indicator
0.001	1
0.002	1
0.003	0

Table 4.4: Values of <0.001 , <0.002 and 0.003 represented using TRUE and FALSE. A TRUE indicates a censored observation.

Value	Indicator
0.001	TRUE
0.002	TRUE
0.003	FALSE

The indicator variable and interval endpoints methods will be used in this document to illustrate examples and computations with data containing non-detects. Using negative numbers is not a recommended method and will not be further discussed in this document.

4.4 Software

Any statistical software that has the ability to perform survival analysis can be used to compute estimates of parameters for data containing censored observations. The three software packages employed in this manuscript are Excel, JMP, and R. The advantages and disadvantages of each software package are discussed below.

Excel is useful for spreadsheet purposes and conducting fast and easy computations, but it lacks survival analysis tools. Nevertheless, Excel is friendly to use when manipulating and formatting data, which can then be exported to other software packages.

JMP is also a fairly common commercial package which has survival analysis capabilities. Unfortunately, JMP lacks some of the more advanced features necessary for the analysis of water quality data. For instance, its functions are only designed for the analysis of right censored data (recall that water quality data are left censored).

R is a statistical software package that allows for both ‘direct’ analysis of water quality data, and for the programming of new analysis routines as needed. Because R is a programming language designed for statistical applications, it can seem foreign when one starts using it. On the other hand, as a user becomes more familiar with it, R provides extensive flexibility in terms of manipulating the data and performing the desired analyses.

The software to use in any statistical analysis for water quality data depends on the personal preferences of the user, and on the constraints posed by the structure of the data itself. If a user is already familiar with a specific software package, and if this package can perform the requisite analysis, then this is the appropriate software to use. Unfortunately, if an analyst is familiar with a package (like Excel), that does not provide the necessary statistical tools, then time must be invested to learn more advanced software like JMP or R. Time spent on learning how to use new software to conduct analyses is time saved in the long term. It will enhance ease of interpretation and good decision making based on the data.

Chapter 5

Plotting Methods

Plotting gives an idea of the behavior and shape of the data. We need to determine the appropriate type of graph for the data. This is decided based on the purpose of the analysis conducted. Methods for plotting data including non-detects are described in this chapter. The graphs described are the box plot and the probability plot.

The box plot gives an idea of the distribution of the data. It shows the data in terms of quartiles. It displays the range of the data, from minimum to maximum, and shows whether outliers are present or not.

The probability plot is used to assess the conformation of the data to a distribution. For instance, the presumption of data being normally distributed can be evaluated based on a probability plot.

Graphical procedures will be illustrated with an example involving concentrations of orthophosphate in the Thompson River at Savona on 32 different dates. The full data set is available in Appendix A.1, and is introduced in Table 5.1. This data will also be used to demonstrate statistical methods in Chapter 6.

Table 5.1: Concentrations of orthophosphate measured in the Thompson River at Savona. A ‘0’ or ‘FALSE’ denotes an observed value, and a ‘1’ or ‘TRUE’ denotes a censored value.

orthophos	censor	cen
0.001	1	TRUE
0.002	0	FALSE
0.002	0	FALSE
0.002	0	FALSE
0.002	0	FALSE
0.001	1	TRUE
⋮	⋮	

Notice that Table 5.1 contains two columns indicating the censoring status of each observation. These two columns are necessary for the data to be compatible with both JMP and R since they

use different conventions. JMP uses the numeric variable `cen` and R uses the logical variable `cen`.

5.1 Box plots

Box plots show the distribution (shape, typical values, spread and outliers) of data. Basically a box plot for censored data is the same as one for uncensored data. The exception being that there is a line at the maximum censoring limit, where below the line the behavior of the data is unknown. Notice that when there are multiple censoring limits, boxplots are no longer useful and other methods should be substituted.

5.1.1 Constructing Boxplots

Boxplots illustrate several key features of the data, specifically regarding the centre, spread, and shape of the data. Boxplots are constructed based on a 5 number summary of the data consisting of the minimum, 25th percentile (1st quartile) , median (50th percentile, 2nd quartile), 75th percentile (3rd quartile), and maximum (100th percentile). Some boxplots also identify the presence of outliers.

The central box in a boxplot spans from the 1st to 3rd quartiles, with a line somewhere in the centre indicating the location of the median. To either side of the main box, *whiskers* extend. The whisker at the bottom extends from the minimum up to the 1st quartile. Conversely, the whisker at the top extends from the 3rd quartile to the maximum.

Outliers are usually defined as values 1.5 times the length of the box extending past either the 1st or 3rd quartile. See Figure 5.1 for a demonstration of the characteristics of a boxplot.

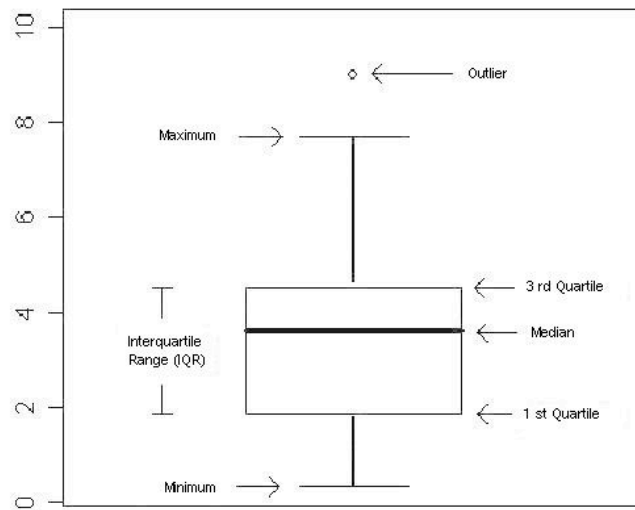
5.1.2 Interpreting boxplots

In addition to picturing the location, spread, and shape of a data set, boxplots are also valuable instruments for detecting outliers and comparing data from different groups.

Location/Centre

Important information about any data set involves identifying where the middle of the data set is located. Boxplots locate the centre of the data using the median. If a data set is symmetric (as in a bell curve), the mean and the median will fall in approximately the same place. In a skewed/nonsymmetric data set, the two values will be different. When data is skewed, the median is often chosen as a more ‘robust’ and better measure of centre. The median is not affected by extreme observations in either the upper or lower tails of the distribution.

Fig. 5.1: Boxplot showing the features of a boxplot.



Spread

Spread is a notion used to discuss the variability of observations in a data set. When the mean is used as the measure of centre, it is typically reported with a standard deviation; when a median is used instead, it is often reported along with the interquartile range as a substitute measure of spread. The interquartile range is obvious in a boxplot. The interquartile range ($Q3 - Q1$) is the distance between the top and bottom of the box; the range of the data is the distance between the top and bottom whiskers.

Looking at the box and whiskers of a boxplot can help identify unusual features in the data. For example, extremely long whiskers could indicate a long-tailed distribution. Conversely, short whiskers tend to indicate a short tailed distribution. Both of these are violations of normality in data, and could affect the coverage probability of confidence intervals in future analyses.

Skewness

Any boxplot that is not symmetric is considered skewed. If the data extends further to the left (smaller observations), it is called left skewed. Conversely, if the 'tail' extends further to the right, a boxplot is described as right skewed. Skewness also indicates violations in a normality assumption.

Outliers

On a boxplot, possible outliers are generally identified as an asterisk extending beyond the whiskers on the plot. There are a variety of reasons why outliers can occur. All of these reasons should be considered prior to further analysis because outliers can drastically change analytic results.

One of the most common sources of outliers is human error in either taking or recording a measurement. If this is the case, the error should be corrected if possible, or the observation discarded.

It is also possible that the experimental unit that was measured belongs to a different population than a majority of the individuals measured. For example, if a majority of water measurements were made in lakes, an observation taken from a nearby river might be an outlier because it belongs to the population of rivers, not lakes. If the outlier is from such a separate population, it can be excluded from future analysis.

The last scenario occurs when the outlier is legitimate data. In this case, the outlier is providing information about the true variability of the population. If this is the case, the outlier *should* be included in subsequent analyses.

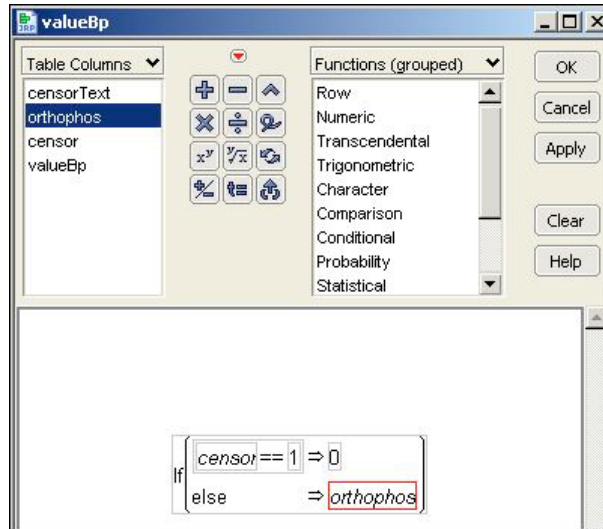
Comparison of groups

When evaluating multiple boxplots it is valid to compare and contrast plots from different groups on all of the criteria mentioned above. Are the means of the groups similar? Is one group skewed, or are all groups symmetric? Are the spreads of the different groups similar? etc.?

5.1.3 Box plots using JMP

In JMP, censor data must be modified so that it is suitable for presentation as a boxplot. The steps to modify censor data are given below.

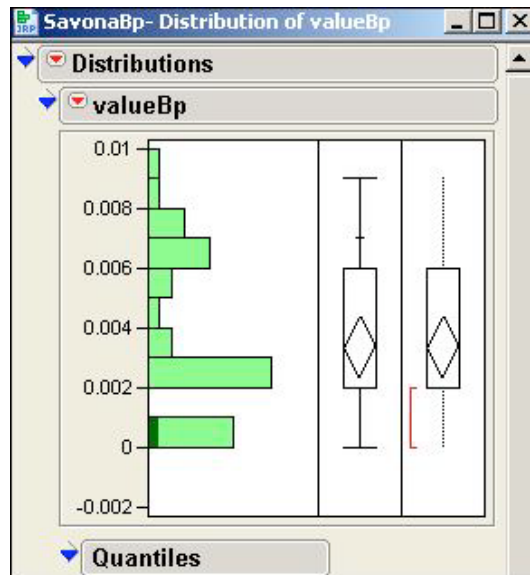
1. Create a new variable that substitutes the censored observations with a single value less than the censoring limit. For example, use zero to indicate the censored values in `savona` data. See the dialog and instructions below to see how to create variables in JMP.



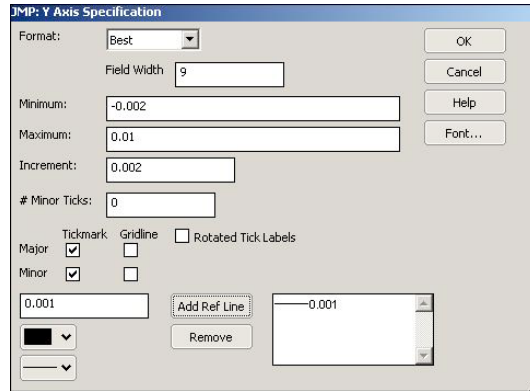
To create a new variable, double click at the top of the spreadsheet to make a new column. Click again to open a window describing the properties of the data column. The new column in this example is named `valueBp`.

For the data used in the JMP example, the concentration variable is called `orthophos`, the censoring indicator is called `sensor`.

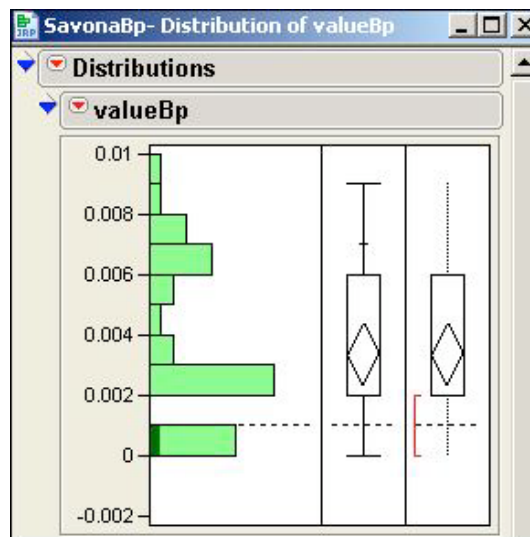
2. Create a standard box plot using `valueBp`; Open the standard box plot window; then ask for a *quantile box plot*.



3. Draw a line at the censoring limit, 0.001; Right click on the axis and specify in the resultant window that a reference line at 0.001 be added; click *OK*.



The resulting graph has a line at the censoring limit (CL), 0.001, indicating that below this line the behavior of the data is unknown. Anything below the CL should not be interpreted.



In the boxplot, many features of the data can be observed. Remember that due to censoring, the minimum value of the data is unknown. The 1st quartile (25%) and the median are both located at 0.002 indicating the data is not completely symmetric. The 3rd quartile (75%) is at a concentration of 0.006 giving a total interquartile range (IQR) of 0.004. There are no apparent outliers.

The exact values of the quartiles can be obtained from the JMP table attached to the box plot and histogram. As mentioned above, the censoring limit is placed at 0.001 and nothing below this number is known. All of the percentiles above 0.001 are computed correctly.

5.1.4 Box Plots using R

A boxplot similar to the one made using JMP in 5.1.3 can also be made using the `savona` data in R. To construct a boxplot using R, follow the steps below.

1. Read the data into R as illustrated in Appendix C.

2. Load the NADA library using the command *library* as follows

```
library(NADA)
```

3. Construct the box plot using the command *cenboxplot* as illustrated below

```
with(savona,cenboxplot(orthophos,cen,main='Boxplot',ylab='Concentration'))
```

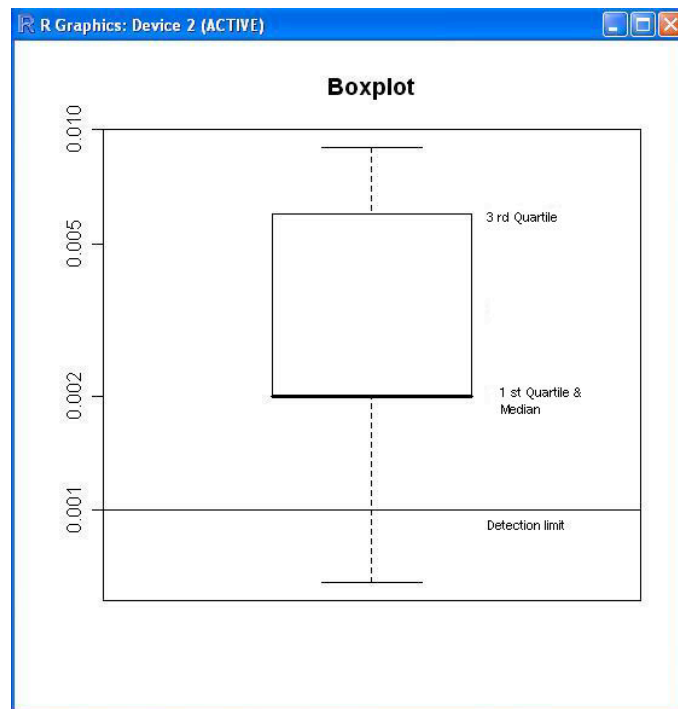
or

```
cenboxplot(savona$orthophos,savona$cen,main='Boxplot',ylab='Concentration')
```

Where *savona* is the data set name, *orthophos* is the concentration measurement, and *cen* is the censoring indicator variable.

In the commands for constructing the box plot, the *main* and *ylab* commands specify the title and the vertical label axis for the plot.

Fig. 5.2: Boxplot for *savona* data.



The graph shows that the censoring limit (identified as the detection limit in the plot) is fixed at 0.001.

5.1.5 Summary of boxplot

The proportion of censored data is represented by how much the data are below the line for the censoring limit. In Figure 5.2, it is observed that less than 25% of the data are below the CL because the 25th percentile is above the censoring limit.

All the detected values are represented correctly, but the distribution below the censoring limit is unknown, and should not be represented in the same way as the data above the limit.

Notice that in this example the lines for the 1st and 2nd (median) quartiles are coincident; thus, the 1st quartile and the median have the same value of 0.002. All observations between the 25th and 50th percentile have a value of 0.002.

In Figure 5.2 the quartiles have been labeled for information purposes, but in general the output does not include this information.

5.2 Probability plotting

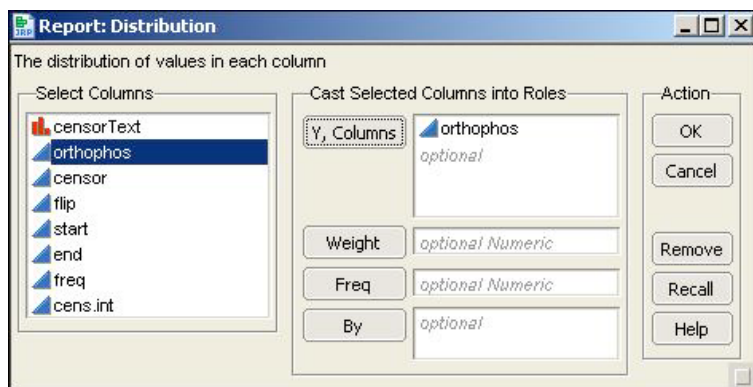
Probability plots are a method to check distributional assumptions. They give a visual check of the conformity of the data to some specific distribution. For example, it is possible to see if the data can be described by a lognormal distribution. If the probability plot shows the points falling roughly on a straight line, then the assumed distribution is appropriate for the data.

5.2.1 Probability plotting using JMP

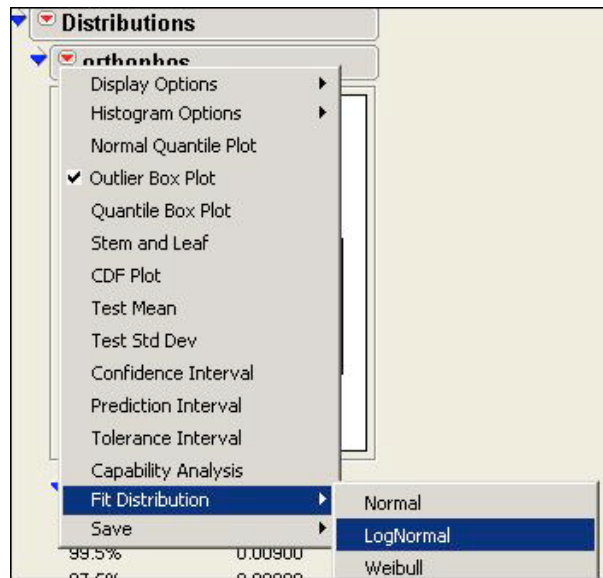
A probability plot for any distribution can be built based on the standard probability plots function in JMP. However, caution should be taken with interpreting the fitted line. The line fitted by JMP is the line corresponding to data with no censored observations. As a result, the ‘line’ that the data should be compared to is not properly displayed by JMP. The true line must be characterized in the mind of the analyst. R does not have this constraint, and can be programmed to correctly plot censored data directly.

The following steps show how to construct a probability plot using a lognormal distribution for the *savona* data. Probability plots can be built for distributions other than the lognormal in a similar way. The instructions for constructing a lognormal probability plot in JMP are shown below.

1. Choose the variable containing the censored and uncensored observations. In the case of *savona* data this variable is *orthophos*. Then, request the distribution report as shown in the picture below.



2. Next, under the red triangle for the variable `orthophos` in the output, select *Fit Distribution* and then *Fit Lognormal*.

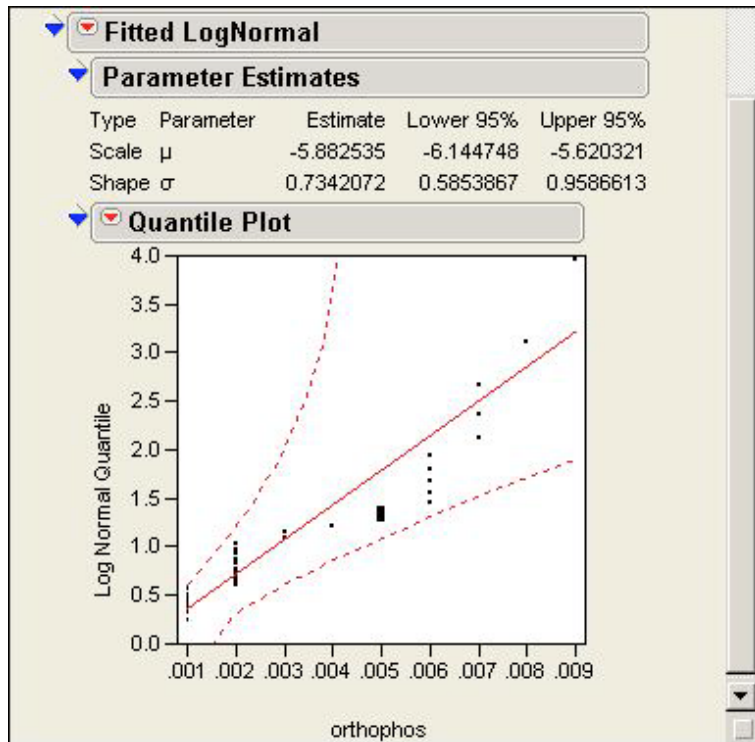
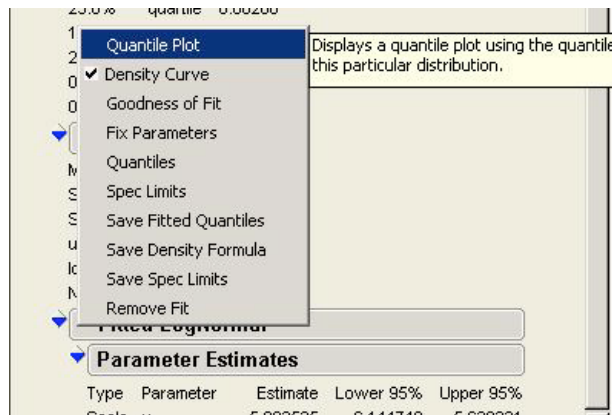


3. In the output, a new section will now be reported, called *Fitted LogNormal*. This is shown in the picture below.

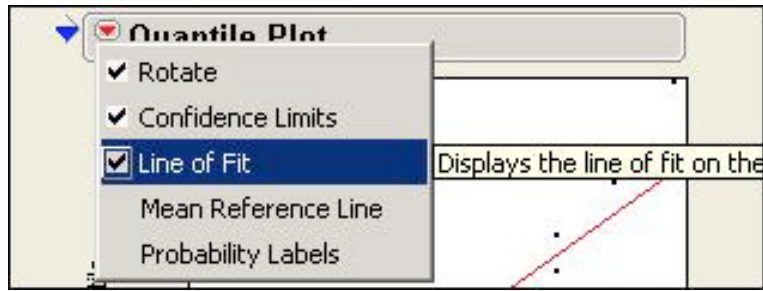
The screenshot shows the Minitab output window. At the top, summary statistics are displayed: 'upper 95% Mean' is 0.0044857, 'lower 95% Mean' is 0.0027018, and 'N' is 32. Below this, a section titled 'Fitted LogNormal' is expanded to show 'Parameter Estimates'. A table provides the following data:

Type	Parameter	Estimate	Lower 95%	Upper 95%
Scale	μ	-5.882535	-6.144748	-5.620321
Shape	σ	0.7342072	0.5853867	0.9586613

- Subsequently, under the menu *Fitted Lognormal*, select the option *Quantile Plot*. The resulting graph is the probability plot shown below.



- Go to the menu *Quantile Plot* and deselect the option *Line of Fit*.
Because the line is incorrect for censored data, this step is used to remove it from the probability plot to prevent confusion.



6. The correct probability plot is shown in the picture 5.3.

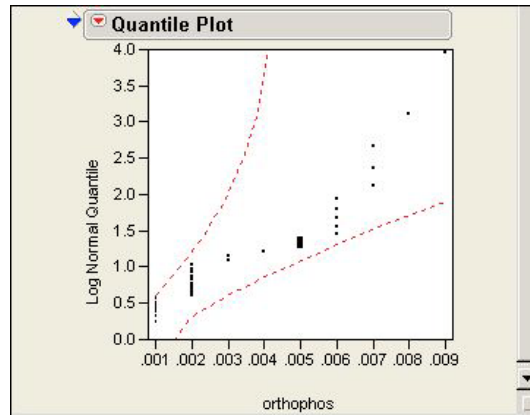


Fig. 5.3: Probability plot for the `savona` data.

This probability plot should be interpreted with caution because of the censored values. Due to their presence the default line provided by JMP is incorrect and there is no way to amend this in JMP. The line we are looking for should only consider points above the censored values. If the uncensored points fall roughly in a straight line, the underlying distributional assumption is reasonable. In Figure 5.3, it is observed that the assumption of lognormal distribution for the data is acceptable.

Probability plots for distributions other than lognormal can be constructed in JMP in a way similar to that described. Comparisons to these other distributions are conducted by choosing a different distribution in the menu *Fit Distribution*. Other distributions commonly used in probability plotting are the normal distribution, the gamma distribution, and the exponential distribution.

5.2.2 Lognormal probability plotting using R

In R, only a lognormal distribution can be checked directly for data with censored observations. In the event that you want to test distributions other than the lognormal, R has limitations similar to JMP. The analyst will have to visualize a straight line for the data, ignoring points below the CL. R only automatically handles censoring in the case of the lognormal distribution.

Again using the concentrations of orthophosphate in the Thompson River at Savona, we can decide if a lognormal distribution is an appropriate fit for these data based on a probability plot shown in Figure 5.4.

To create this plot, the steps below should be followed.

1. Fit the regression on statistics (ROS) model to the data. The model fitted by default with command `ros` is lognormal.

```
savros = with(savona, ros(obs, cen))
```

Where `savona` is the data set name, `obs` is the concentration of orthophosphate, and `cen` is the censoring indicator variable.

2. Plot the lognormal fitted model using the command

```
plot(savros)
```

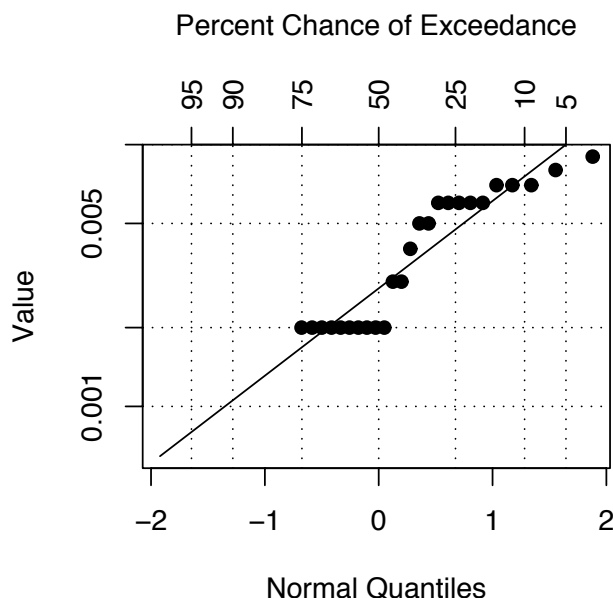


Fig. 5.4: Probability plot for lognormal distribution with `savona` data. The vertical axis is given in log scale.

For Figure 5.4, the vertical axis records the concentration values on the log scale. Each value represents the logarithm of the concentration. As stated before, the lognormal distribution is a distribution such that the logarithm of the data values have a normal distribution. On the horizontal axis, the expected normal percentiles are reported.

If points in this probability plot fall in an approximately straight line, the assumption that the data has a lognormal distribution is reasonable. By contrast, if the points in the probability plot are far from falling in a line, the assumption of a lognormal distribution may be questionable.

In Figure 5.4 the points fall close to a straight line. This indicates that the lognormality assumption is reasonable. Thus, methods assuming the lognormal distribution can still be used.

5.2.3 Probability plotting in R if the data are not lognormal

If the probability plot based on the lognormal method shown in Section 5.2 shows serious deviations from a straight line (and lognormality) a number of actions can be considered. In the absence of lognormality, statistical techniques based on a lognormal assumption *should not* be used; if used, the results should be interpreted with extreme caution. A better alternative is to choose an analysis method that relies on different assumptions (such as non-parametric or semi-parametric methods).

Despite this, if you should want to construct p-p plots based on a distribution other than the lognormal in R, please see Appendix F.

Chapter 6

Calculating Statistics

With any data set it is important to look at some of the simple features of the data, so that the behavior of subsequent analyses are better understood. For example, it is important to know the percentage of nondetects in the data when we are trying to select the appropriate method for computing statistics.

There are basically three methods for obtaining statistics for nondetects: substitution, maximum likelihood, and nonparametric methods.

Although substitution is an easy method, it has no theoretical basis, and in simulation studies has been shown to give poor results.

Maximum likelihood, or parametric methods, consist of fitting a distribution to the data by estimating the parameters that define the distribution. The estimation consists of finding the most likely values of the parameters that could have generated the data. Estimated parameters, such as a mean, or a standard deviation can then be used to make inferences about the population of interest. Censored values are taken into account in maximum likelihood estimation.

Maximum likelihood methods are useful when the sample size is large; some authors recommend at least 30 observations and others 50. Determining an appropriate sample size within this range can depend on how well the data fits model assumptions (eg. normality). The better model assumptions are met, the smaller the sample size that is necessary for reliable estimates.

Nonparametric methods are used when the sample size is not sufficient to use maximum likelihood methods, or the distributional assumptions are questionable.

When the assumptions for parametric methods seem to be reasonable, these methods are desirable. However, when the data do not seem to fit any distribution, nonparametric methods are suggested.

Conditions for the appropriate use of each method and how to use them are given in Sections 6.4, 6.5, and 6.6.

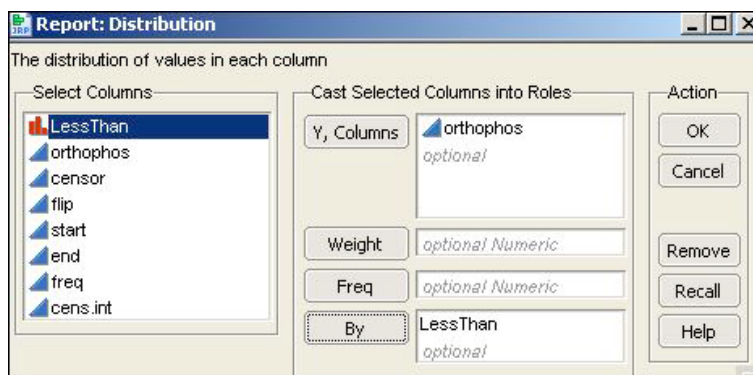
6.1 Summarizing nondetects

Simple data summaries can give an idea of the distribution of the data. One basic analysis is to count the number of censored and uncensored observations. This summary of the censored and uncensored data can be performed in both JMP and R. Use of both of these programs is described in the following two sections.

6.1.1 Summarizing nondetects using JMP

Using JMP, censored and uncensored observations can be summarized using the following steps as guidelines. The `savona` data is used as an illustrative example.

1. Go to *Analyze*→*Distribution*; specify the variable concentration, `orthophos`, as shown in the picture below. The variable `LessThan` defines the two groups of interest, censored and uncensored data.



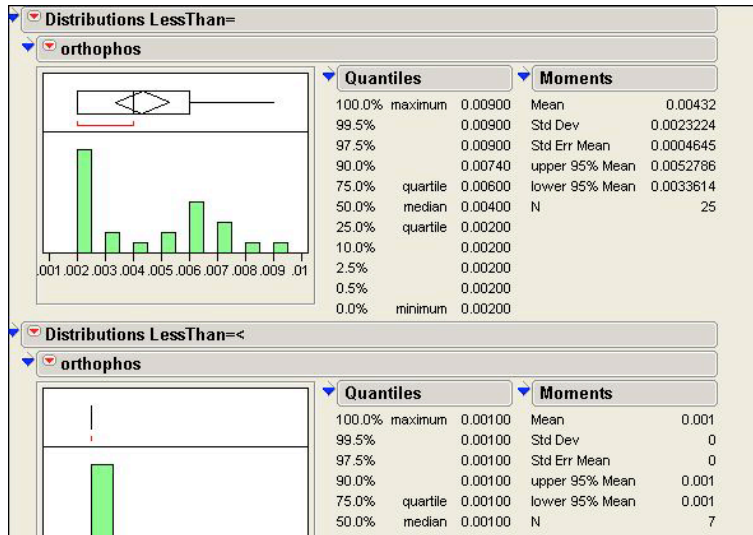
Note that variable `LessThan` should be specified as a nominal variable in order to obtain the correct estimates. The resulting output will be as shown in the picture below.

From this output we can see that there are 7 censored observations and 25 uncensored values in the `savona` data.

6.1.2 Summarizing nondetects using R

The `NADA` package in R has a command called `censummary` for summarizing censored data. The exact percentage of censoring values and other quantities of interest can be computed with `censummary` as follows.

```
> censummary(savona$obs, savona$cen)
all:
      n  n.cen pct.cen      min      max
32.000  7.000  21.875  0.001  0.009
```



limits:

```

limit n uncen pexceed
1 0.001 7 25 0.78125

```

How to interpret the output in the all part is as follows.

- `n` is the total number of observations (detects and nondetects);
- `n.cen` is the number of nondetect/censored values;
- `pct.cen` is the percentage of censored/censored observations; and
- `min` and `max` are the minimum and the maximum values of all data.

In the `limits` part, the output can be explained such that:

- `limit` is the censoring limit,
- `n` is the number of nondetect values,
- `uncen` is the number of detect observations, and
- `pexceed` is the percentage of values exceeding this censoring limit.

For the data on concentrations of orthophosphate in the `savona` sample, there are 32 observations and 7 of them are censored. This yields a 21.875% censoring percentage. The minimum observation is also the censoring limit, 0.001; the maximum value is 0.009.

6.2 Mean and median

The mean, median and mode are measures of central tendency, and fall in the same place if the distribution of the data is symmetric. However, if the distribution of the data is skewed, the three values will no longer be identical. For instance, if the data is right skewed, the mean becomes larger than the median and no longer represents a “typical” value in the data. In these cases, it might be more sensible to report the median, which is a more resistant measure of centre. The mean and the standard deviation are highly influenced by extreme values (outliers) and skewed data. The median is more robust and remains relatively unchanged in the presence of skew and outliers. For example, Figure 6.1 presents a data set skewed to the right (long right tail).

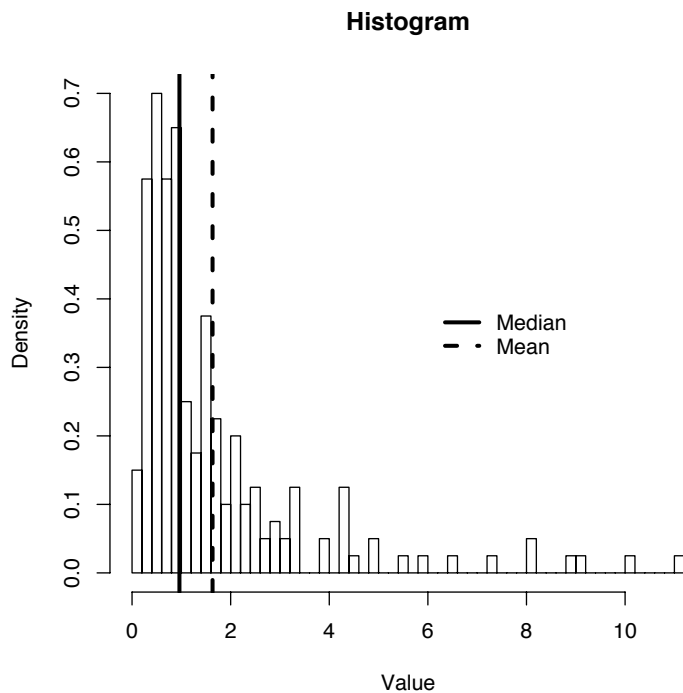


Fig. 6.1: Histogram of data skewed to the right.

In Figure 6.1, the sample mean is 1.63 and the sample median is 0.96. In the histogram of the data it can be observed that the median is a more “typical” value than the mean. Because there are some large values in the data, the mean is pulled to the right. A similar phenomenon occurs when there are outliers. Extreme observations can pull the sample mean in their direction. This situation is also discussed in Section 6.3 where unusual observations are a result of substitutions.

Because it is not as influenced by extreme observations and skew, the median is a more stable statistic to measure the centre of data than the mean. As long as the percentage of nondetects is less than 50%, the sample median will be an unbiased estimate of the population median. In contrast the mean will be greatly affected if nondetected cases are dropped or substituted. In particular, if zero or the censoring limit are used as a substitution, the resulting means can lead to two different conclusions, as is shown in Section 6.3. To conclude, the median is a more robust measure of centre when censoring is present.

6.3 Substitution

It is computationally simple to just substitute a single value or a set of values for the nondetects, and then apply standard statistical techniques to get the quantities of interest from the data. Unfortunately, it has been shown in the last two decades that the substitution method for dealing with nondetects is inappropriate. In many cases misleading conclusions can be drawn based on data where this approach has been taken, since the choice of the value substituted is completely arbitrary, and statistical results will be biased depending on the value chosen.

Some authors have unsuccessfully tried to prove the merits of substitution, at least under certain conditions. Although the desired results for the mean were achieved when the distribution was uniform, this is not a common situation with environmental data.

All the studies in the literature show the poor performance of substitution and the absence of a theoretical basis for its use. Despite this, the substitution method is still recommended by some organizations such as the United States Environmental Protection Agency (USEPA). USEPA, in its “Guidance for Data Quality Assessment”, recommends substitution when the percentage of censoring is less than 15%. This assertion has been criticized by Helsel (2006).

Helsel (2006) suggests that substitution should only be used when approximate outcomes are required since the quality of the results is dubious. Substituting fabricated numbers for nondetect values biases the mean and, more seriously, the variance.

The variance is a measure of spread of the data about the mean. If an arbitrary value is substituted for the nondetects, there is no variability in the replaced values. Therefore a false estimate of variance is generated.

Substitution commonly uses three numbers: zero, the censoring limit (CL), or 1/2 the censoring limit. Choosing a value of 1/2 the censoring limit assumes that this value is the mean of the unobserved data. As stated earlier, this is only possible when the distribution of the concentrations is uniform. Since the choice of both the DL and QL are based on a normal assumption, having a simultaneous uniform assumption is impossible!!

In the case of substitution with the CL or zero the variability of the data is artificially changed. Because p-values and confidence intervals are inherently linked to the variance estimate, these substitutions make it statistically difficult to detect true environmental trends.

The variance is incorrectly altered by substituting for non-detects. Subsequently, all the tests and intervals based on the variance estimate are influenced too. Comparisons between groups and estimates of regression trends will be unreliable. The results based on substitution can lead to completely inaccurate conclusions.

6.3.1 Substitution example

To illustrate the deleterious effects of simple substitution for censored data, an example from Helsel(2005b) is shown. Twenty-four measurements of arsenic concentration ($\mu\text{g/L}$) taken from

streams near Oahu, Hawaii comprise the data set. There are three censoring limits: 0.9, 1, and 2. The number of censored observations is 13, corresponding to 54% of the data (See Section 6.1.2). Table A.2 contains the raw observations.

As stated earlier, three values are commonly used to substitute for nondetect - zero, the CL, and 1/2 the CL. Boxplots of the data under the three substitution approaches are illustrated in Figure 6.2. Figure 6.3 shows the correct boxplot for the data.

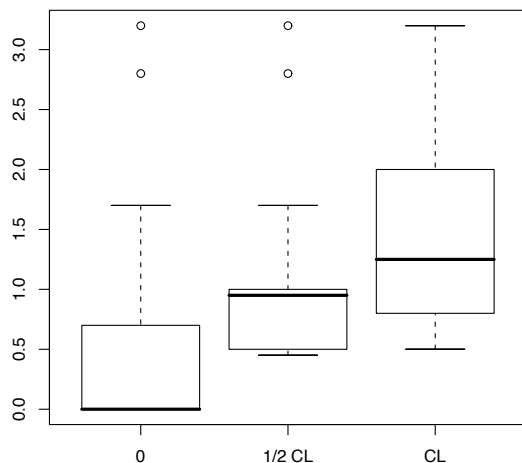


Fig. 6.2: Boxplots of arsenic concentrations at streamwaters in Oahu, Hawaii after substituting the censoring limits for 0, 1/2 CL and CL.

In addition to making boxplots look vastly different, substitution also vastly changes many of our parameter estimates. Table 6.1 contains a variety of summary statistics after the different substitution choices have been implemented.

Table 6.1: Summary statistics for an arsenic sample taken from Oahu after substituting the censored observations by 0, 1/2 CL, and CL.

Value Substituted	Mean	Standard deviation	25 th	Median	75 th
Substitution with 0	.567	0.895	0.000	0.000	0.700
Substitution with 1/2 CL	1.002	0.699	0.500	0.950	1.000
Substitution with CL	1.438	0.761	0.850	1.250	2.000

Notice that the results from different substitution methods diverge considerably. It is clear that the estimates of the mean are deeply affected by the choice of substituted value. Compounding the problem, the standard deviations vary from 0.699 to 0.895, a difference of nearly 0.2, or a ratio of nearly 1.3 times changed. Such multiplicative differences in variance estimates lead to equivalent multiplicative changes in our ability to achieve statistically significant p-values.

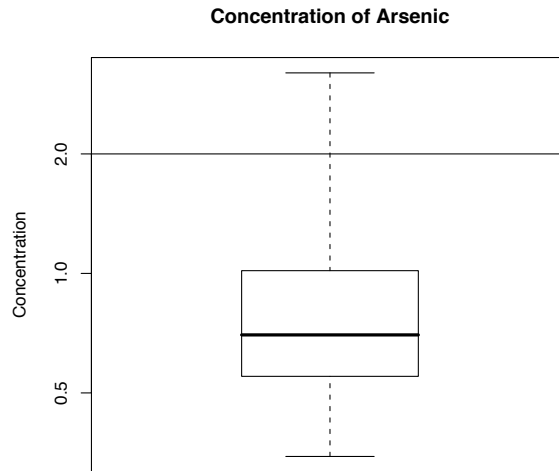


Fig. 6.3: Boxplot of arsenic concentration at streamwaters in Oahu, Hawaii.

As Helsel(2005b) points out, if legal requirements are to maintain arsenic levels below $1 \mu\text{g}/\text{L}$, the criteria would be assessed as met using substitution with 0, and would not be met with substitution using the CL.

6.3.2 Assumptions in substitution

This section presents the assumptions involved with substitution and discusses why these assumptions are not valid.

- The substitution of nondetects by arbitrary numbers does not affect the mean “significantly” when the percentage of censoring is less than 15%.

This assertion is rarely, if ever, correct. Additionally, without omniscience it is impossible to determine when substitution is appropriate, or even which substitute value to use. True values of a sample below the CL are *never* known, so using a mean obtained by substitution is a risky choice for inference.

- The variability among the nondetects is very small. In fact, by substituting with a single number, variability is reduced to 0 among those values!

A variety of values are usually collected in a random sample; it is rare to see identical values in such a random sample. In using substitution we have artificially removed this natural variability. There is no basis for presuming that values below the CL are less variable than values above the CL.

6.3.3 Consequences and risks of substitution

Additional risks associated with substitution include the following.

- Substitution biases the mean, and more seriously, the standard deviation. Any statistics that rely on these quantities will also be biased. Most p-values and all confidence intervals are calculated based on point estimates and standard deviations. This means that inferences based on these quantities are generally incorrect if the substitution method is used. In real life we do not know the truth, so there are no available criteria to select numbers for substitution.
- The choice of substitution number is arbitrary, and statistical results change depending on this choice. One repercussion of this is that replication of results by different individuals and scientific bodies is not guaranteed, even when based on the same sample! Replication is a cornerstone of the scientific method, and there are techniques other than substitution that can ensure that replicability is achieved.
- The substituted values depend on the conditions which determine the censoring limit, such as laboratory precision or sample matrix interferences.

6.4 Nonparametric methods: Kaplan-Meier

The Kaplan-Meier (KM) estimator is a traditional method for calculating summary statistics based on censored data in survival analysis. The KM method can provide useful estimates when sample sizes are small and the percentage of censored observations is between 0% and 50%. These are the same conditions under which Regression on Order Statistics (ROS) is used. ROS methods will generally provide superior results. This discussion of the KM estimators is included because KM methods are the historic standard, and are still often seen in the literature; in the context of modern computing power though, ROS is the more strongly recommended analysis method.

KM methods are available in all commercial statistical packages that offer survival analysis methods. This includes JMP and R. Most software packages assume that the data is right censored, and left censored data can be transformed to meet this requirement. Results can then be back-transformed for interpretation in original units. An example of how to do this will be given in Appendix G.1

The Kaplan-Meier estimator is an estimate of the *survival curve*, which is the complement of the cumulative distribution function (cdf)¹. In medical or engineering data, the survival curve is computed for data on the survival time of individuals or parts, respectively. Consequently, the KM estimate at time t is an estimate of the probability of an individual surviving *past* time t . The cumulative distribution in this case would estimate the probability of survival *up to* time t .

The Kaplan-Meier estimates the survival curve by accounting for the censored observations at each time point, or, in our case, at each concentration. The KM method is a nonparametric estimator since it does not assume a distributional shape for the data.

¹The survival curve is equal to 1- cdf

Using the Kaplan-Meier estimator, it is possible to estimate the mean and the standard deviation of the population. However, the estimate of the standard deviation is only an approximation, and when even one observation is below the censor limit, the estimate of the mean is biased.

In water quality data, left censored data are present since some values in the sample are known to be below the CL. In this case, the smallest observation (which is censored) is *below* the CL. Consequently, the mean estimated for water quality data using the KM method will be biased upward, and is not an ideal estimate.

Because the KM estimator has problems with bias when censoring occurs at the end points of the data, we strongly recommend using the methods presented in Sections 6.5 and 6.6 to compute mean and standard deviation. However, notice that the Kaplan-Meier always estimates the percentiles correctly.

6.4.1 Statistics computed based on Kaplan-Meier

The statistics computed using the Kaplan-Meier are the percentiles of the survival distribution, the mean, and the standard deviation.

The transformed value of the mean is obtained by summing the area under the KM survival curve. The survival curve for the transformed *savona* data is shown in the Figure 6.4.

By adding up all of the observations, x , and dividing by the number of values present in the sample, n , we obtain the mean

$$\mu = \sum \frac{x}{n},$$

When there are several observations of equal value, the equation can be stated as

$$\mu = \sum \frac{f_i}{n} x_i,$$

where f_i is the number of observations at each of the i unique values of x , and $\frac{f_i}{n}$ is the proportion of the data set at that value.

The mean obtained through Kaplan-Meier is biased if the censoring is present. Since the exact value of the extreme observations are unknown, the rectangle generated in the computation of the mean has an unknown base and we cannot calculate an exact sum for the mean.

Because end observations are usually censored in environmental data; the estimated mean using the Kaplan-Meier method will be biased. Additionally if the distribution is skewed, the mean and the standard deviation will be estimated poorly. The standard deviation is usually estimated in conjunction with the mean in statistical software. Although means and standard deviations are commonly reported in environmental data, medians and interquartile ranges are more desirable statistics.

At the end of this analysis we want to interpret the data in the original units. Imagine we pick the value of 0.006 in Figure 6.5. In this figure the value of 0.006 corresponds to a probability of 0.75 on the vertical axis. This is the third quartile on the original scale².

²This is because the survival probability equals the cumulative distribution function on the original scale.

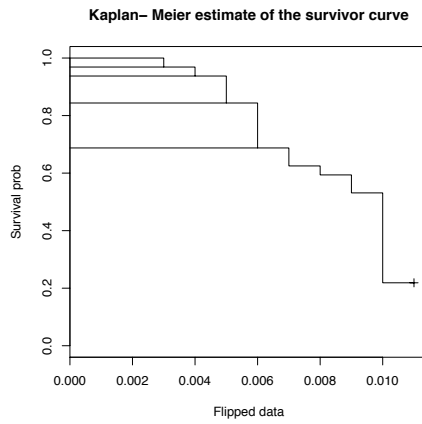


Fig. 6.4: Survivor curve of the Kaplan-Meier estimator for the concentration of orthophosphate in savona data. The data are transformed by subtracting each observed value from 0.012.

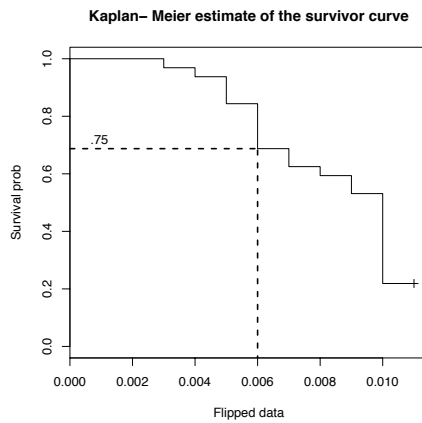


Fig. 6.5: Illustration of quantiles based on the Kaplan-Meier estimate of the survivor curve for the savona data.

6.4.2 Assumptions

- Kaplan-Meier does not assume any distributional shape for the data.
- The percentage of censoring should be less than 50% in order to be able to compute the median and percentiles above it.

6.4.3 Risks

- When censoring is present, as is usual in environmental data, the estimator for the mean is biased.

- Although this method can be used when the sample size is large, if the data seem to have a distributional shape, it is better to use parametric methods. Parametric methods give more precise estimates.

For more information on how to calculate Kaplan Meier curves and their associate statistics refer to Appendix G.1.

6.5 Parametric methods

Another approach to estimating parameters with data containing censored values are maximum likelihood, or parametric methods. These methods are called ‘parametric’ since they assume a parametric form for the distribution of data, e.g. a normal distribution.

Maximum likelihood methods assume that a known distribution will closely fit the shape of the data. Subsequently, the parameters which match the shape of the distribution as closely as possible to the data are found (maximum likelihood estimation).

A lognormal distribution is very commonly used in environmental data. Therefore it is used here to illustrate the parametric procedures. The ideas outlined here could also be applied to any other parametric distribution.

6.5.1 The lognormal distribution

A lognormal distribution is a distribution whose logarithm is normally distributed³. The lognormal distribution can be skewed to the left or the right such that the natural logarithm of the values in a sample follow a normal distribution. This distribution is often used when a lower limit of zero is combined with large observed values.

In Figure 6.6 (a) there is an example of lognormal distribution showing that the shape of the lognormal is usually skewed with large observed values that might seem to be outliers. When the logarithm of the values is graphed in a histogram, the shape of the distribution looks symmetrical like a normal distribution as shown in Figure 6.6 (b). Through this transformation the influence of the largest values is controlled.

³<http://en.wikipedia.org/wiki/Lognormal>

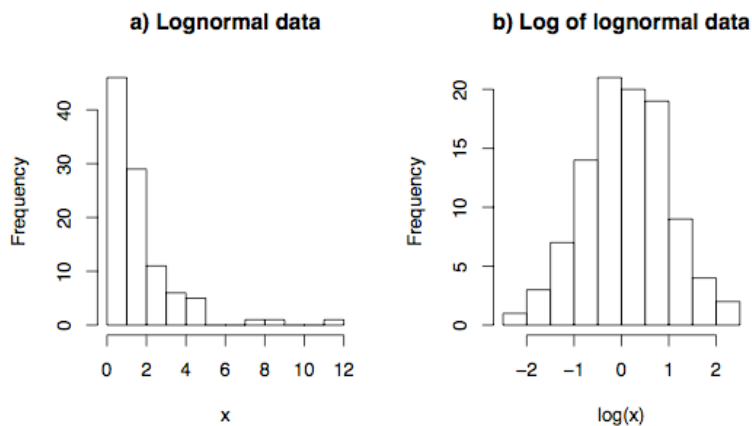


Fig. 6.6: Histograms of a lognormal and normal distributions. The natural logarithm of lognormal data gives a normal distribution.

The lognormal distribution has two parameters: μ , which denotes the centre, and σ^2 , which denotes the variance about the centre on the log-scale and helps to determine the shape of the distribution. When estimating the parameters of the lognormal distribution, $\hat{\mu}^4$ and $\hat{\sigma}$ are obtained. These are estimates based on logarithm of the data, and cannot be interpreted on the original scale. Thus, a transformation is needed in order to get estimates of the mean and the standard deviation on the original scale.

Notice that $\hat{\mu}$ is an estimate of the mean of the log data; a naive transformation would be to take the antilog of this quantity, but this transformation of the mean returns the median on the original scale. Recall that the log transformation should center the data and make it symmetric, and the mean and median in a symmetric distribution are the same. The center in a symmetric distribution is the 50th percentile. It follows that when we back transformed, the 50th percentile stays the same, and therefore the log of the mean becomes the median on the original scale.

The estimators for the mean and the standard deviation on the original lognormal scale are:

$$\hat{\mu}_{originalscale} = \exp(\hat{\mu} + \hat{\sigma}^2/2), \quad (6.1)$$

$$\hat{\sigma}_{originalscale}^2 = \hat{\mu}_{originalscale}^2 \times \exp(\hat{\sigma}^2 - 1), \quad (6.2)$$

and the formula for estimating the quantiles in the original scale is

$$\hat{p}_k = \exp(\hat{\mu} + z_k \hat{\sigma}). \quad (6.3)$$

Here the subscript *originalscale* denotes the estimates for the original units of measure. Then $\hat{\mu}$ and $\hat{\sigma}$ are the estimates obtained for the centre and standard deviation parameters on the log scale, $\hat{\mu}_{originalscale}$ and $\hat{\sigma}_{originalscale}$ are the estimates for the mean and standard deviation on the original

⁴The hat symbol denotes that these quantities are estimates based on a sample.

scale. The quantile for lognormal data is \hat{p}_k , where k denotes the quantile of interest, and z is the quantile for a standard normal distribution. For example, if the percentile of interest is the median, then $z_{.50} = 0$, and $\hat{p}_{.50}$ is $\exp(\hat{\mu})$.

The application of the formulas 6.1, 6.2, and 6.3 will be illustrated in Section 6.5.4.

Summary statistics for parametric methods can be computed with most survival analysis software packages, in particular JMP and R. This is illustrated in Section 6.5.4 and Section 6.5.5.

6.5.2 Assumptions

- The data should have a shape similar to a lognormal distribution. Refer to Figure 6.6 a.
- A sample size of at least 30 to 50 observations is needed. Fifty is more conservative for good estimation. A sample size of 30 should only be used if the data has been assessed graphically and observed to closely follow a lognormal distribution based on these graphs.
- The percentage of censoring should be less than 50%.

6.5.3 Risks

- When the sample size is small, the parametric method will give uncertain results. For smaller sample sizes, or distributions that differ seriously from a lognormal, ROS (or maybe KM) methods should be used.

6.5.4 Summary statistics using JMP

For parametric methods, JMP can handle left censored data, so it is straightforward to get the summary statistics based on the maximum likelihood estimates. To calculate summary statistics in JMP based on lognormal distribution follow the steps below.

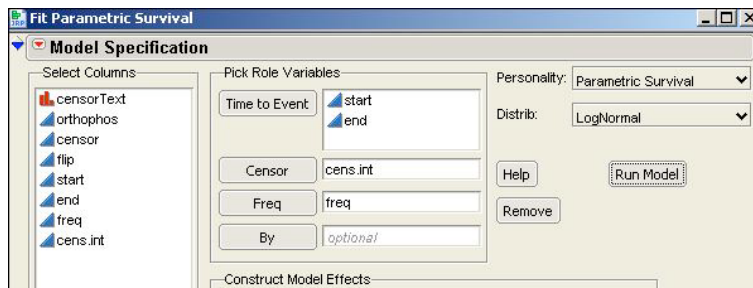
1. Check that the shape of the distribution is similar to the shape of the lognormal distribution. This can be done with a probability plot as shown in Section 5.2. The probability plot for `savona` data was shown in the Figure 5.3.
2. In the spreadsheet, create two columns to represent the observations as intervals. The first column shows the interval starting point, the second its endpoint. (Please see Section 4.2 on data storage.) If an observation is censored, the starting column should show a zero. Otherwise, the observed value should be recorded. For both censored and uncensored observations, the second column will be the observed value; for the `savona` data the column is called `orthophosphate`.
3. Create a new column listing the frequency of each observed value. Because the data are continuous, most observations will be uniquely occurring i.e. the frequency will be 1.

4. Create a new variable to indicate that interval censoring is being used. In JMP, the value “2” is used when the observations are not censored, and “0” when they are. Note that our choice of the numbers 0 and 2 is because we need to signal to JMP to use left censoring routines.

The graphic below shows the variables necessary to fit the lognormal model (ignoring flip). The column/variable names correspond closely to their content. For instance, the variable `cens.int` is an indicator of the censoring status.

censor Text	orthop hos	censor	flip	start	end	freq	cens.int
<	0.001	1	0.011	0	0.001	1	0
	0.002	0	0.01	0.002	0.002	1	2
	0.002	0	0.01	0.002	0.002	1	2
	0.002	0	0.01	0.002	0.002	1	2
	0.002	0	0.01	0.002	0.002	1	2
<	0.001	1	0.011	0	0.001	1	0
	0.002	0	0.01	0.002	0.002	1	2
	0.003	0	0.009	0.003	0.003	1	2
	0.002	0	0.01	0.002	0.002	1	2
<	0.001	1	0.011	0	0.001	1	0

5. Fit the parametric model by going to *Analyze -> Survival and Reliability -> Fit Parametric Survival*. Specify in this window the columns corresponding to the startpoints (`start`), the endpoints (`end`), the censoring indicator variable (`cens.int`), the frequency variable (`freq`), and the distribution to be fitted. This is shown in the picture below.



6. Observe, the output of the fitted maximum likelihood estimates is shown below.

The output from JMP gives estimates of centre of the data and its variability. These are reported as the intercept and σ , respectively. On the logarithmic scale, the estimate of the mean ($\hat{\mu}$) is -5.99 and the estimate for the standard deviation ($\hat{\sigma}$) is 0.91 .

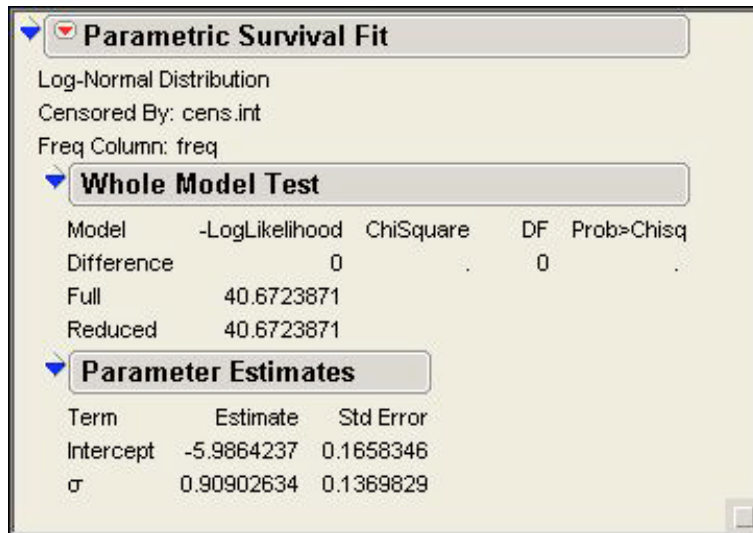
To interpret these values on the original scale, we use the formulas from 6.5.1 to backtransform the estimates above. Based on these formulas, the mean and standard deviation on the original scale are as shown below.

$$\hat{\mu}_{originalscale} = \exp(-5.99 + (0.91)^2/2) = 0.00379$$

$$\hat{\sigma}_{originalscale} = \sqrt{0.00379^2 \times \exp((0.91)^2 - 1)} = 0.0043.$$

Similarly, some sample percentiles on the original scale can be calculated as shown below.

$$p_{.25} = \exp(-5.99 - 0.67 \times 0.91) = 0.0014$$

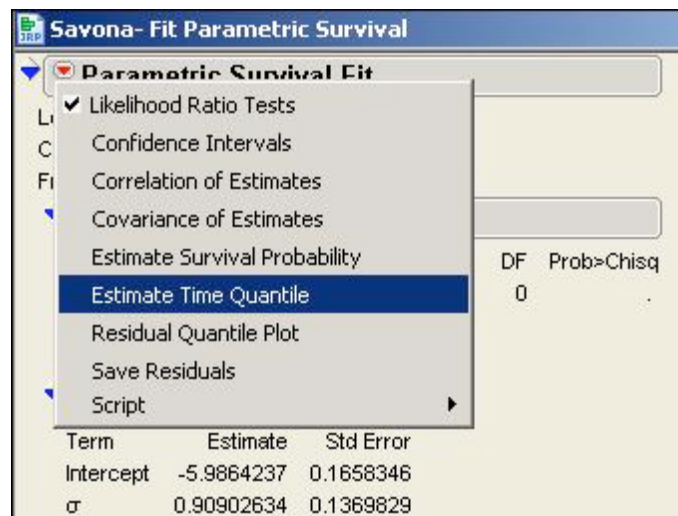


$$\text{Median} = \exp(-5.99 + 0 \times 0.91) = 0.0025$$

$$p_{.75} = \exp(-5.99 + 0.67 \times 0.91) = 0.0046.$$

Notice that the median and the 1st quartile are not identical when using this method. In the boxplots drawn in Chapter 5, these two quantities were identical. Boxplots are non-parametric visualizations of the data, whereas here we are assuming an underlying distributional form. This leads to moderately different estimates.

- We can also obtain percentile estimates in JMP. On the output from above, select *Estimate Time Quantile*.



The complement of the quantiles of interest are shown below in the *Dialog to Estimate Quantiles* window.

Survival Prob	Alpha
0.9	0.0500
0.75	
0.5	
0.25	
0.1	

The 0.75 probability of survival time corresponds to the 25th percentile in our data, and the 0.25 survival corresponds to the 75th percentile. Note that survival time is defined as the probability of an event being *larger* than a point of interest, t . For our purposes, we are interested in concentration that are *below* a certain threshold, and so we take the complement.

Prob Survival	Time	Lower 95%	Upper 95%
0.9	0.0007838	0.0004732	0.0012981
0.75	0.001361	0.0009186	0.0020163
0.5	0.0025126	0.0018154	0.0034777
0.25	0.0046388	0.0032695	0.0065816
0.1	0.008055	0.0051883	0.0125058

Using this method, 90% of the concentrations are greater than 0.00078, so the value 0.00078 represents the 10th percentile.

6.5.5 Summary statistics using R

R works similarly to JMP, but R requires fewer steps. The package *NADA*, used in the environmental field, has implemented these steps internally and written a routine that can be applied directly to get the summary statistics. To estimate the summary statistics using the *NADA* package use the steps below.

1. Read in the data file as shown in Appendix B, or Appendix D.
2. Make sure that the column that indicates the censoring value says “TRUE” for censored observations and “FALSE” for observed values.
3. Fit the model, specifying the data name, the function to fit (in this case *cenmle*), the column containing observed censored values, the column indicating the censoring status, and the distribution that is being fitted. This is done using the commands:

```
fit = with(data,cenmle(values,censored),dist='lognormal'),
```

So using our `savona` data, we would write

```
savmle =with(savona,cenmle(obs,cen),dist='lognormal')
```

4. The next step is to visually test that the lognormal distribution approximates the data. This can be done using a probability plot.

To do this, fit a model with the `ros` command, and then plot the resulting object.

```
savros=with(savona,ros(obs,cen)) plot(savros)
```

The probability is shown in Figure 6.7.

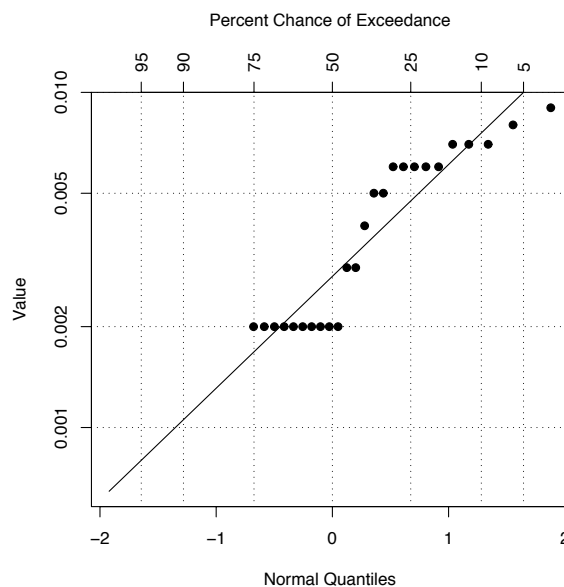


Fig. 6.7: Probability plot for `savona` data, assuming a lognormal distribution.

While not strictly following the line, the probability plot in Figure 6.7 suggests that the assumption of lognormal distribution might be reasonable enough for the data. We can see that the data fall roughly close to the theoretical line. Because parametric methods are large sample methods, they tend to be robust to slight violations in their assumptions.

5. Once we have assessed adequacy of the fit, the parameters estimated for the lognormal distribution can be obtained using the command `summary`

```
summary(savmle)
```

In this summary, the parameter estimates for μ and σ are reported as `Intercept` and `Scale`, respectively. For instance, the estimate for μ is -5.9864 . These parameter estimates are equal to the ones obtained using JMP.

```

> summary(savmle)
              Value      Std. Error      z          p
(Intercept) -5.9864      0.166      -36.099      2.37e-285
Log(scale)  -0.0954      0.151      -0.633      5.27e-01

Scale= 0.909

Log Normal distribution Loglik(model)= 99.2  Loglik(intercept
only)= 99.2 Loglik-r: 0 Number of Newton-Raphson Iterations: 6 n =
32

```

6. Finally, the mean, the standard deviation and the quantile estimates are obtained by typing the commands *mean*, *sd*, and *quantile* as is shown here for the *savona* data.

```

mean(savmle)

sd(savmle)

quantile(savmle)

> mean(savmle)
      mean      se  0.95LCL 0.95UCL
0.003798 0.165835 0.002616 0.005514

> sd(savmle)
0.004305

> quantile(savmle, conf.int=TRUE)
quantile  value  0.95LCL 0.95UCL
  0.05 0.000563 0.000314 0.001009
  0.10 0.000784 0.000473 0.001298
  0.25 0.001361 0.000919 0.002016
  0.50 0.002513 0.001815 0.003478
  0.75 0.004639 0.003270 0.006582
  0.90 0.008055 0.005188 0.012506
  0.95 0.011207 0.006721 0.018688

```

The results for the summary statistics match with the ones obtained using JMP. The mean is 0.00379 and the standard deviation is 0.0043.

Notice that using JMP and R, we also obtain the interval estimates for the quantiles (percentiles). In R, including these intervals is requested using the instruction `conf.int=TRUE`.

6.5.6 Notes on the lognormal method.

- Note that using this method all quantiles can be estimated. This is not the case with the KM method where just the ones above the largest censoring limit can be estimated.
- Parametric methods are good to use when the data seem to follow a familiar distributional shape, and when the sample size is large. These methods usually give more precise results than the KM method (See required sample sizes in Section 2.1).
- The maximum likelihood estimator can handle multiple censoring limits.
- The illustrated procedure is general for any single censoring limit used; this limit can be selected as either the detection or the quantitation limit.

6.6 Robust Regression on Order Statistics (ROS)

Regression on Order Statistics (ROS) is a second method that assumes the data follow a lognormal distribution. It performs a regression on the data assuming lognormal quantiles. The line created predicts unknown observations. Summary statistics can be computed based on the predicted observations and on the non-censored observations.

The idea behind ROS is that if the data follow a lognormal distribution (or some other known distribution), then a probability plot of the log of the ordered data versus the quantiles of a standardized normal should give a straight line. Thus, the mean and standard deviation for the log of the data can be obtained. The mean is estimated using the intercept and the standard deviation using the slope of the line. Subsequently, unknown values below the censoring limit can be extrapolated (using the estimated parameters). Observations for all potential values are known, and all the summary statistics can be estimated.

Transformation bias is a concern any time the log of data is used. To help correct this, quantities from a normal distribution are first transformed to lognormal quantities. The summary statistics are then computed on the scale of the original data (Helsel, 2005c).

ROS is only implemented in R, and is unavailable for JMP.

6.6.1 Assumptions

- Almost any sample size is sufficient for ROS. Sample size does not need to be bigger than 30. ROS works fine for small data sets, as well as large ones.
- There is no limit on the percentage of censoring that can occur. Indeed the censoring percentage can be up to 80% (although results should be interpreted carefully in this situation!).
- This method is resistant to non-normality in the data. Even in the presence of skewed data sets, we can achieve meaningful inference.

6.6.2 Risks

- Like other methods, ROS estimates are biased because of the log transformation. Fortunately, it has been shown in simulation studies that the bias is small using this method (Helsel, 2005c).

6.6.3 Notes on ROS

- ROS can deal with multiple detection limits.
- It is robust because it uses the sample data as much as possible.
- It is good for small data sets, where MLE-lognormal does not perform well.
- The predicted observations should not be used as if they were the unknown individual observations. They are useful for computing some statistics, but they should not be interpreted as individual observations.

6.6.4 ROS using R

The steps to follow in order to get the summary statistics based on ROS estimation are as follows.

1. Read the data file into R. (See Appendix C or Appendix D.)
2. Load package *NADA*.

```
library(NADA)
```

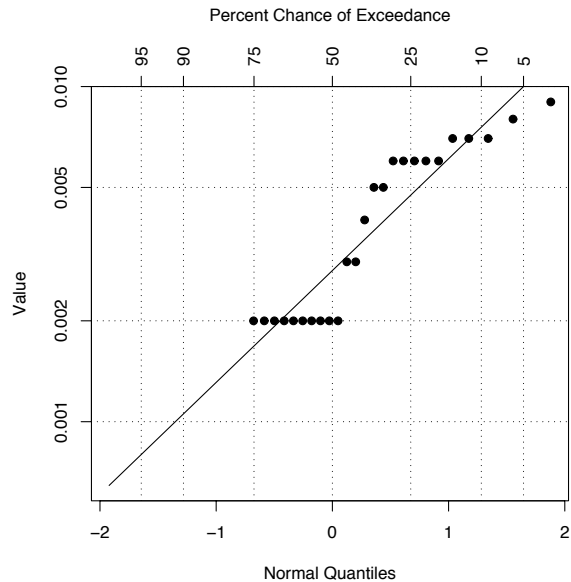
3. Fit the model using the command *ros*:

```
savros = with(savona, ros(obs, cen)).
```

4. To make a probability plot in order to assess lognormality, use:

```
plot(savros).
```

Note that in this probability plot, all points are plotted except for the censored observations. Even the censored points can be plotted with the command



```
plot(savros,plot.censored=TRUE).
```

The probability plot is shown above. Notice that the vertical axis denotes the logarithms of the orthophosphate concentrations.

Even though the points do not fall exactly on a straight line, summary statistics based on ROS can be computed and compared with results from other methods. Recall that ROS is resistant to non-normality even for smaller sample sizes.

5. Compute the summary statistics using commands *mean*, *sd* and *quantile*.

```
mean(savros)    sd(savros)    quantile(savros)

> mean(savros)
[1] 0.003610389

> sd(savros)
[1] 0.00245915

> quantile(savros) #
      5%      10%      25%      50%      75%
0.0009046877 0.0011097892 0.0020000000 0.0020000000 0.0060000000
      90%      95%
0.0070000000 0.0074500000
```

The estimate of the mean is similar to the estimate obtained based on the MLE method. The standard deviation using ROS was only half that found using MLE estimates. Notice that ROS does not give confidence intervals for the parameters estimated.

Confidence interval for estimates based on ROS method should be computed using the bootstrap method discussed in Section 6.6.5 and Appendix E. The standard error gives a good measure of

uncertainty for the mean only when the sample size is large. The standard error for the percentiles, however, should still be estimated using the bootstrap method.

Because ROS uses information from the sample at much as possible, the estimates for the quantiles do not change as long as they are above the censoring limit.

The command `censtats` in R can fit all three of methods (Kaplan Meier, MLE, and ROS) along with summary results. The commands to do this are as follows.

```
with(savona, censtats(obs, cen))

> with(savona, censtats(obs, cen))
      n   n.cen pct.cen
32.000   7.000  21.875
      median      mean      sd
K-M 0.002000 0.003813 0.002274
ROS 0.002000 0.003610 0.002459
MLE 0.002513 0.003798 0.004305
```

The mean estimated by the three methods is similar. The median based on MLE is moderately different from the other two methods. The standard deviation based on MLE is also different from the other two methods. Perhaps this is because the sample size is too small for an MLE approximation to work well.

6.6.5 Computing confidence interval estimates using the bootstrap method

Because confidence intervals provide more information than point estimates alone, it is necessary to compute the confidence intervals for the mean and the percentiles obtained through ROS method. The recommended method to compute confidence intervals for the mean and percentiles is through bootstrap estimation.

Bootstrap

The bootstrap method is very popular for computing standard errors in situations where computing the standard errors directly is complex.

There are numerous variations of the bootstrap method. The one shown in Helsel (2005b) is a bootstrap with replacement. Bootstrap with replacement consists of taking random sub-samples of the main data set. Individual observations from the main sample may be included more than once in a sub-sample; this is what sampling *with replacement* means. Statistics of interest are calculated for each sub-sample, and the distributions of statistics from the sub-samples are used to define confidence intervals for statistics in the main sample.

While interesting, a thorough understanding of how to conduct a bootstrap is not necessary to use its results. For the curious, a sample bootstrap algorithm has been included in Appendix E based on the method found in (Helsel, 2005b).

Using the bootstrap method to compute interval estimates

In R software there is already a routine that computes the bootstrap estimates for censored data. This routine will be used to perform the analysis. To compute confidence intervals for the mean obtained through ROS estimation follow the next steps.

1. Put the data in a form appropriate to bootstrap estimation. It is necessary to make sure that the data is in the required format to apply the ROS procedure in R. The data should contain two columns; one column that has the observed values, and the other one has the censoring indicator. In case this step was not done previously, the data can be formatted using the command above.

```
data = data.frame(savona$obs,savona$cen)

names(data)=c('obs','cen')
```

The command `names` assigns labels to the data columns.

2. Write a routine (function) that computes the statistic of interest using ROS estimation. For instance, a function for the mean is illustrated here.

```
mean.sav <- function(dat){
  mean(ros(dat$obs,dat$cen))
}
```

3. Run the function from step 2 to compute the statistic of interest. Select the text in the editor window and click *CTRL + R* to run the function from step 2.
4. Apply the `censboot` command to the data, specifying the the function written in step 2, `mean.sav`. Indicate the number of samples that will be used with bootstrap, in this case 300.

```
res = censboot(data,mean.sav,R=300)
```

The value `R` denotes the number of samples that will be used for the bootstrap estimate. As stated in this example `R=300` is used, but in general it is recommended to select `R` between 1000 and 10000.

5. Type the word `res` to see in the command window the statistics of the bootstrap estimate. The results for `savona` data are below.

```
> res
```

```
CASE RESAMPLING BOOTSTRAP FOR CENSORED DATA
```

```
Call: censboot(data = data, statistic = mean.sav, R = 300)
```

```
Bootstrap Statistics :
```

```
      original      bias      std. error  
t1* 0.003610389 -5.851149e-07 0.0004274721
```

The estimate for the mean of orthophosphate concentration is given in the `Bootstrap Statistics` section. The mean is the value corresponding to the value of `original`, in the output above. In this case, the bootstrap estimate of the mean is 0.003610389.

6. Compute the confidence interval of the statistics of interest, in this case, the mean. The extreme points of the confidence interval correspond to the 2.5th and 97.5th percentiles of the values obtained for the 300 bootstrap samples. Values other than 0.025 and 0.975 can also be chosen.

```
quantile(res$t,.025) quantile(res$t,.975)
```

The results are shown in the R console.

```
> quantile(res$t,.025)  
      2.5%  
0.002795867  
> quantile(res$t,.975)  
      97.5%  
0.004501423
```

7. Similarly, confidence intervals for other statistics can be computed using the bootstrap method. The commands below show the function that can be written for using the bootstrap method for quantile estimation. This function is specifically being used to find confidence intervals for the 75th percentile.

```
quan.sav <- function(dat,p=.75){  
  quantile(ros(dat$obs,dat$cen),p)  
}
```

The word `dat` indicates the data set to be used in the routine. The `p` indicates the percentile required. Note that this probability can be easily changed and other percentiles of interest can be computed.

8. Analogous to the mean estimation using the bootstrap method, creating the `quan.sav` function and using the commands below give the confidence interval for the 75th percentile.

```
res.quan = censboot(data,quan.sav,R=300) quantile(res.quan$t,.025)
quantile(res.quan$t,.975)
```

The 95% confidence interval for the 75th percentile has extreme values 0.00325 and 0.007, as shown below from the R output.

```
> quantile(res.quan$t,.025)
 2.5%
0.00325
> quantile(res.quan$t,.975)
97.5%
0.007
```

Chapter 7

Introduction to the Second Half: Comparing, correlating, regressing, and other assorted topics

This chapter introduces the second half of a guidance document written for the BC Ministry of Environment to provide a set of consistent and statistically and scientifically defensible guidelines for the analysis of water quality data. Both halves are intended to help improve data analysis procedures when evaluating data for the presence and significance of contaminants (or other compounds) in water quality data. In particular, the focus is on how to summarize and analyze data where some of the observations fall below analytical detection limits. Secondly the document provides guidelines regarding which software packages are appropriate for various analyses, and how to use these packages.

In the first half of the document the main focus was to communicate plotting methods for data with non-detects, and how to calculate measures of centre (mean, median) and measures of spread (standard deviations, interquartile ranges). Some basic definitions of terms are also included, as well as a discussion of data storage and transfer, and a brief introduction to some software packages and their potential uses.

Generally speaking, material that was discussed in the first half of the guidance document will not be repeated in the interests of brevity. It is intended that users of this document will be at least somewhat familiar with topics in Chapters 1 through 6 .

The second half of the document, being introduced here, is intended to extend into a broader range of data analysis. Specific topics to be addressed include: how to compare parameters such as the mean between two groups; extending this to comparisons among 3 or more groups; assessing trends (eg. through time) using correlation; and regression type methods. Included as special topics are what to do when there are small sample sizes; some approaches to try when multiple detection limits are present; and what to do when the proportion of censored observations is above 50% and the above methods are unreliable.

In addition to providing the ideas behind and formulae for various analysis methods, information on software will also be presented, along with examples. Unlike the previous chapters, there will

be little, if any focus on the Excel and JMP software packages. This is for a variety of reasons, but mainly because these packages are intended for very general and/or commonly used statistical analyses.

Instead, the R program will be used almost exclusively for data analysis. R is free, open source, highly respected and accepted in the statistical community, and already has procedures written to address problems in water and air quality research. Despite all of the positive statements I have made about R, at this juncture I sense that some of my readers are beginning to panic! For those of you having this feeling, please take 5 deep breaths, and read on.

As the author of this document, I am well aware that having to perform statistics is rarely greeted with cheer and good feeling. Quite the opposite. Extra anxiety and uncertainty might also be present at the suggestion of learning a new, and at least initially, unfamiliar software program. Realize that far from having an intent to cause uncertainty, statistics is meant to be a way of thinking that assists in quantifying and addressing uncertainty to facilitate decision making.

This document should be viewed as an opportunity to learn and enhance skills rather than as a burden. My goal is to write a document that is relevant, accessible, useful, and (dare I say it?) enjoyable. With some luck, and a small amount of work, it is possible to gain the skills necessary to make data interpretation a more meaningful activity.

Chapter 8

Brief Review of Some Key Concepts

This section is intended to provide a brief review of some of the conventions that were used in earlier chapters. For more information on these topics, refer to the previous document. A small amount of background information is also included on some key statistical ideas.

8.1 Detection, Quantitation, and Censoring

In water quality literature, both *detection limits* (DL), and *quantitation limits* (QL) are familiar concepts. In water quality, the DL is a lower bound on our ability to detect non-zero concentration values in a given lab setting. Measurements near the DL are considered unreliable. To compensate for this lack of reliability near the DL, the idea of a QL is introduced. The QL is a concentration value at which we start to have more confidence in concentration estimates based on the chemical analysis method used.

From a statistical perspective, both the DL and the QL are called *censoring limits*(CL). In this document it is assumed that an analyst will have chosen either the DL or the QL as their lower bound on concentration measurements. Regardless of which is chosen, in this document we will always refer to the chosen limit as the CL to prevent confusion.

8.2 Data Transformations

When people think of statistical distributions, what often comes to mind is a normal distribution which is also known as a bell curve. While there are many examples of variables that closely follow such a bell shaped distribution (height, weight), it is inevitable that not every variable will have this shape.

Common departures from this shape include data that are either skewed to the left, or skewed to the right. In water quality data, it is most common for data to be skewed to the right. This is because there is a lower limit for concentration at zero, but no defined upper limit. This means

that while distributions of concentrations may have their main body at low values, they can extend out to much larger values creating a large ‘tail’ to the right.

Because there are many well known methods for analysis of normally shaped data, it is often convenient to *transform* skewed data in some way so that the distribution of the data is closer to normal. The most common of these transformations is the logarithmic, or log, transformation. Conventional statistical analyses can then be performed on the transformed data, and then the results can be easily back-transformed for interpretation on the original scale on which the data originally occurred.

It should be noted that in statistics when data is referred to as log-transformed, we always mean the natural logarithm (\log_e , \ln). The natural logarithm, \ln , has many useful mathematical properties that make it easy to compute estimates and back-transform estimates onto the original scale. Unless specifically indicated, a log transformation never refers to \log_{10} .

If log-transformed data looks like it follows a normal distribution, we often say that the data on the original (skewed) scale follow a lognormal distribution.

8.3 Statistical Concepts

Common summary measures of distributions are measures of centre, and measures of spread. This section will discuss both of these measures, and also recommends how to effectively communicate the information they represent.

8.3.1 Measures of Centre

Where the main body of data collected about a variable falls is often a question of key importance. Knowing the location of the center of the distribution is a way of expressing what is common, “normal”, and representative of observations in that data set. Statistics that talk about where the center of a distribution is located are called *measures of center*, *measures of location*, or *measures of central tendency*. Parameters used to conceptualize these values are often called *location parameters*.

The most well known measure of central tendency is the mean, or average. Other well known measures of central tendency are the median, and the mode. When a distribution is symmetric, as with the normal distribution, the mean, median, and mode, will all be located at the same point.

Although we often think of a measure of central tendency as a single point, it can be representative of more complex information. For example, a regression line is also a measure of central tendency. A regression line represents where the centre of a distribution falls while conditional the the values of one or more explanatory variable. It is important to remember that both the mean, and other measures representing trends can both be considered measures of centre.

8.3.2 Measures of Spread

Often under-emphasized in the reporting of data are measures of spread, dispersion, or variability. While measures of central tendency focus on *where* the centre of a distribution is located, measures of spread are concerned with representing *how close* the distribution of data falls in relation to the center. Common measures of spread, or variability, include standard deviations, variances, ranges, and quartiles or percentiles. Each of these measures of spread is reported around a specific location parameter, either the mean or the median¹.

Notice that when presented in isolation, measures of spread provide no information about the location of the centre of a distribution. They only provide information about a variable's dispersal around a location. As a result, parameters used to conceptualize dispersion are often called scale parameters.

When describing distributions, a mean is generally associated with a standard deviation. Means and standard deviations are commonly used when the distribution being described is symmetric.

Analogously, when medians are being used as a measure of center, they are generally reported within a 5 number summary, or with an interquartile range. Generally speaking, medians are a more robust measure of centre than means when the distribution is skewed, or there are unusual observations that might influence the mean. Considering that censored observations can be viewed as unusual, many of the methods that will be discussed in upcoming sections will be based on medians and quartiles rather than means and standard deviations.

8.3.3 Communicating Statistical Estimates

There is a truly awful joke often told about (but not enjoyed by) statisticians. The joke is as follows:

A statistician is a person who stands in a bucket of ice water, sticks their head in an oven, and says "On average, I feel fine!"

In addition to its questionable value in terms of humour, the above statement is also questionable in terms of understanding how statistics should be reported.

Clearly a person with their feet in ice, and head in a hot oven is not fine. Although the average body temperature of a person in this condition might be the standard 37°C, it is the range of temperatures that would be deadly. The joke, by focusing on the *average* temperature of the statistician, accuses the statistician of failing to understand the *variability* in the temperatures they are experiencing.

The message to be taken from the statistician's dilemma is that reporting on measures of centre without reporting on associated measures of spread causes an erroneous decision because where the centre of a distribution is located is only relevant when interpreted in context with its variability measure.

¹Because using modes as a measure of location is not that common, we won't be discussing them further

In all cases, a measure of centre should be reported with an appropriate measure of spread!!

Most statistical tests are calculated based on both a measure of centre, and an estimate of the variability in the distribution of data around that location. As a result, it has become relatively common practice in applied literature to report only the p-value of a statistical test. While not technically wrong, p-values do not convey much information about the variability of the location parameter being estimated.

I recommend on reporting parameter estimates as confidence intervals. Confidence intervals contain information about the location of estimate, and a range of plausible values for the true parameter based on the observed data. Confidence intervals also implicitly include the result of any related statistical test².

Although statistical tests will be discussed, this document will also focus on reporting results as confidence intervals.

²If a confidence interval contains zero (or an alternative null value), it implies support for a null hypothesis of no effect, and that the p-value of an associated statistical test is not significant.

Chapter 9

Comparing Centers of Two Independent Populations

Moving beyond simple summary statistics, one of the pioneering ideas in statistics is in comparing parameters between two populations. Populations are made of independent experimental units, meaning that units in one population are not included in the second population .

Hypotheses about parameters from two groups can take the form of either one-sided or two-sided tests. More on both forms of testing is discussed below.

In a one-sided test, we have a specific question about whether parameter from one group is larger than another. One-sided tests are particularly common in water quality literature. We are often testing whether an area that has potentially been exposed to a contaminant has average measured contaminant levels higher than a control site. In formulating null and alternative hypotheses for testing, a one-sided test will often take a form similar to that shown below.

$$\begin{aligned}H_0 &: \theta_{control} \geq \theta_{contaminated} \\H_A &: \theta_{control} < \theta_{contaminated}\end{aligned}\tag{9.1}$$

In the hypotheses shown above, θ represents the parameter of interest. In our applications θ can refer to either the population mean, or the population median.

In a two-sided test, we are simply interested in determining whether parameters from two groups are different from each other. There is no prior interest that one group should be higher than the other. For example, this might be used to test whether baseline amounts of a dissolved mineral are similar in two different areas. When formulating a hypothesis, a two-sided test will take the form shown below.

$$\begin{aligned}H_0 &: \theta_{population1} = \theta_{population2} \\H_A &: \theta_{population1} \neq \theta_{population2}\end{aligned}\tag{9.2}$$

Above we have only discussed statistical tests. Such tests would involve reporting only a test statistic

or p-value, indicating whether evidence is stronger for H_A or for failing to reject H_0 . Even when the question of interest takes the form of a test similar to above, I strongly recommend including some type of confidence interval about the estimate differences between the populations whenever possible. Such intervals still contain information about the outcome of a hypothesis test, and also about our uncertainty in the parameter.

In the Sections 9.1-9.3 below, I discuss some specific statistical tests that can be used to compare two groups when left censored data are present. The discussion will include tests that should not be used (9.1), as well as tests that should be used when observations include those below a CL (9.2,9.3). These acceptable methods will include discussion of techniques for both large(9.2) and small (9.3) sample sizes. Also included will be examples showing how to perform the tests using the R software package.

9.0.1 The Data Set

The data we will be using to demonstrate analysis of a two independent sample design was originally published in a paper by Millard and Deverel (1988). The data were collected from the San Joaquin Valley of California. The question of interest is whether or not typical zinc concentrations in groundwater are different in two geomorphic zones: *alluvial fan*; and *basin trough*. Because we have no prior belief of which geomorphic zone might have higher Zinc concentrations, a two-sided test is appropriate.

The data consists of a variety of copper and zinc concentrations taken from groundwater sampled in the two geomorphic basins. The data are freely available with the download of the NADA package in R. The data have two censor limits at 3 and 10 $\mu\text{g}/\text{L}$. The detection limit changed due to changes in analysis methods over time.

Loading the Data in R

Appendix D discusses how to install R, and download the NADA library. For information on how to install an R library when you don't have administrator privileges, refer to appendix H.

Once NADA has been loaded into R, you can access the San Joaquin data using the following command in the R console: `data(CuZn)`. Once this command has been executed, you can view the data by typing `CuZn` in the R console. A few lines of the data are presented in Table 9.1

Table 9.1: Concentrations of copper and zinc in the San Joaquin Valley

Cu	Cucen	Zn	ZnCen	Zone
1	TRUE	10	TRUE	AlluvialFan
1	TRUE	9	FALSE	AlluvialFan
3	FALSE	NA	NA	AlluvialFan
3	FALSE	5	FALSE	AlluvialFan
5	FALSE	18	FALSE	AlluvialFan
⋮	⋮			

9.0.2 Preliminary Data Inference: Graphing

Before any analysis of the data, it is almost always useful to create some preliminary plots that give an idea of the shape of the data and any unusual features. Boxplots are an excellent method to do this, and to compare the data across two or more groups. Notice though that in the complete data set there are some NA, or unavailable observations for zinc. Prior to making boxplots, the data need to be cleaned of these missing values. To clean the NA observations, and create a side by side boxplot of the zinc concentrations for the alluvial fan and basin trough regions execute either of the following sets of commands in R:

```
#Removes row 3, containing NA
CuZn=CuZn[-3,]
#creates the censored boxplot
with(CuZn,cenboxplot(Zn,ZnCen,group=Zone,
main="Censored Boxplot of San Joaquin Zinc Concentrations",
ylab="Zinc Concentration",range=1,log=FALSE))

#removes row 3, containing NA
CuZn=CuZn[-3,]
#alternative method to create a censored boxplot
cenboxplot(CuZn$Zn,CuZn$ZnCen,group=CuZn$Zone,
main="Censored Boxplot of San Joaquin Zinc Concentrations",
ylab="Zinc Concentration",range=1,log=FALSE)
```

Either command set will result in the creation of the boxplot shown below in Figure 9.1.

A key feature of this boxplot is that it is extremely difficult to observe any features on the boxplot! This is due to the presence of extreme outliers and right skewness. To make the features of the data easier to see, it is desirable to view the boxplot on the log-transformed scale. To create side by side boxplots based on the same data but on the log scale, use the same commands as shown above, but substitute the argument `log=TRUE` for the argument `log=FALSE`. Performing this change will result in the boxplots being displayed on the log scale, as can be seen in Figure 9.2.

Fig. 9.1: Side by side boxplots for zinc data

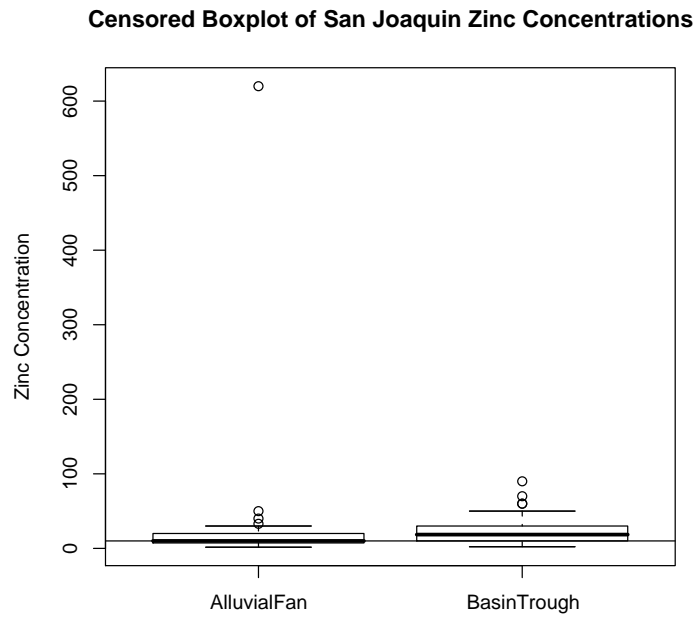
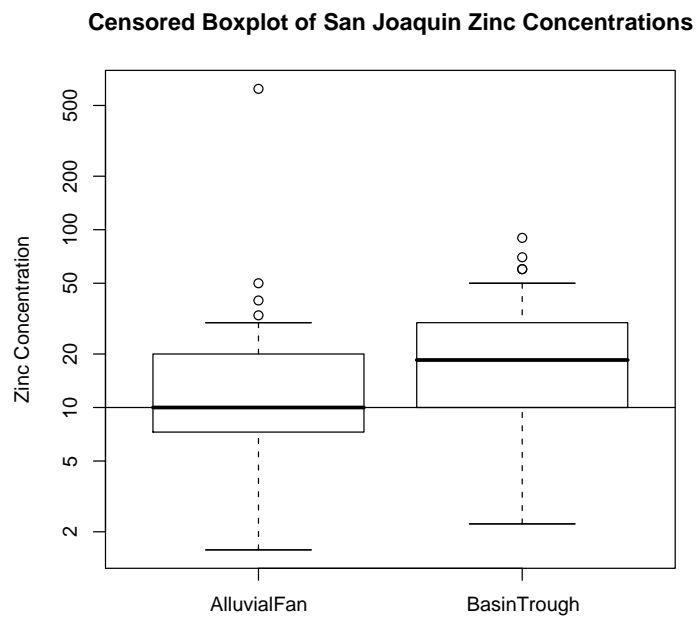


Fig. 9.2: Side by side boxplots for log zinc Data



Key features of the data are much clearer on this log-transformed boxplot. Recall from Chapter 6 that when using the `cenboxplot(.)` command, a horizontal line is drawn at the upper limit for censoring. *Features below this line should not be interpreted.*

Notice that more of the observations from the alluvial fan zone are at or below the CL than for the basin trough zone. This can be determined by the difference in the relative placement of the medians of the two groups. By definition, censoring in water quality occurs at low concentration values, so the presence of more censored observations in the alluvial fan zone indicates a larger proportion of data at small concentrations.

Also notice that the median of the basin trough data occurs at a higher concentration value than for the alluvial fan zone. It seems plausible that typical zinc concentrations are not the same in groundwater samples from the two regions. Zinc concentrations tend to be lower in the alluvial fan zone than in the basin trough zone.

9.1 What Not To Do: T-Tests and Substitution

A common method for comparing the population means of two independent groups is the independent sample t-test. This method has assumptions of approximately normal distributions, equal variances, and independent observations. Because observations below the CL will affect both the normality and variability of the underlying population data, t-tests are not an appropriate analysis method for data sets that include censor values.

To emphasize this point, we will examine the zinc concentration data discussed in 9.0.1 where the zinc concentration differences between the two geomorphic groups were apparent based on boxplots. T-tests will not give reliable estimates of the difference between group means, regardless of what value is substituted for the non-detects. This data set will be re-analyzed in 9.2 and 9.3 where more correct analysis methods will be demonstrated.

To perform a t-test on the zinc concentration data where substitution has been used, observations falling below the CL must be replaced by a substituted value. The most common values used for substitution are 0, 1/2 the CL, and the CL. For purposes of demonstration, I will substitute censor values with 1/2 the CL. Additionally, I will perform the t-test using the log-transformed data (with the transformation occurring after substitution) because this is a transformation that improves the normalcy of the data.

T-tests can be easily computed in R, and demonstration code will be given below. When performing statistical test, I often recommend *assigning a name* to the test that is being computed by R so that the results can be easily referred to on future occasions. Demonstration code of how to perform a two-sample t-test on the zinc concentration data is shown below. The test below is named `Zn.t.test`, and this name was assigned by writing the name, and then an = sign before the command for the test.

```
Zn.t.test=t.test(log(CuZn$Zn)~CuZn$Zone,alternative="two.sided",conf.level=0.95)
```

or you could use

```
Zn.t.test=with(CuZn, t.test(log(Zn)~Zone,alternative="two.sided",conf.level=0.95))
```

Notice that I have not discussed how to create substituted values in R, because I don't want you to know how to do that! R can also be used to perform standard statistical analyses, so the t-test code will work on data sets that do not have values below the CL.

There are some options that are also worth discussing when using the `t.test(..)` command in R. One of them is the argument `alternative`, which takes the form `"two.sided"` in the R code above. This is referring to the form of the alternative hypothesis, and choices are `"two.sided"`, `"less"`, and `"greater"`. The `alternative` argument, in conjunction with the `conf.level` argument specify the direction of the alternative hypothesis, as well as the level of significance that the hypothesis is to be tested at.

In addition to the significance test, when the `t.test` command is executed a variety of other information is stored, including a confidence interval on the difference in group means. The `alternative`, and `conf.level` arguments also specify the confidence level and form of the confidence interval that created.

For the curious, a list of other arguments that can be supplied to the `t.test` command can be found by typing `?t.test` in the R console which will bring up a window of information on t-tests. Placing a `?` in front of any command in R will provide the help files for that command. An alternative option when the exact syntax of a command is not known is to type `help.search("t test")` to bring up a list of commands related to t-tests.

After fitting the t-test model in R, it is desirable to look at a summary of results from the test. To do this in R, simply type the name of the test, `Zn.t.test`, in the R console. Executing this command will result in the `Zn.t.test` object being shown as is seen below.

```
> Zn.t.test

Welch Two Sample t-test

data:  log(Zn) by Zone
t = -1.6155, df = 98.955, p-value = 0.1094
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.58624500  0.06004236
sample estimates:
mean in group AlluvialFan mean in group BasinTrough
          2.444042          2.707144
```

From this output, we find a value $t = -1.6155$ which corresponds to a p-value of 0.1094 based on 99 degrees of freedom (df). Because a p-value of 0.05 is generally chosen as a cutoff for statistical

significance, the results of this analysis imply that there is little evidence against a null hypothesis of no difference in zinc concentrations between the alluvial fan and basin trough zones. After our review of the boxplots, which appeared to indicate differences between typical zinc concentrations between the zones, this should come as a somewhat surprising result (and make us suspicious about t-tests)!

Recall that at the beginning of this section, we discussed how substitution will affect the underlying assumptions of regarding both the variability and normality of the data. With two major assumptions violated, results from a t-test are unreliable. It turns out that in calculating t -tests using all of the possible substitutions (0,1/2CL,CL), none will indicate significant differences in the log zinc concentrations between the two geomorphic zones. If you substitute 0, you cannot take the log transformation of the data, so the test has to be performed on the original skew scale. On the log scale, both 1/2CL (as shown above) and substitution with the CL itself (try it on your own) result in non-significant t-tests. Non-significant differences between the means is inconsistent with the results we would expect based on the boxplots seen in Figure 9.2

In the section below, 9.2, we will discuss a large sample parametric method for comparing two independent samples that have left censoring. This is a more appropriate parametric method than a t-test as it does incorporate information from values falling below the censor limit.

9.2 What To Do: Parametric Methods

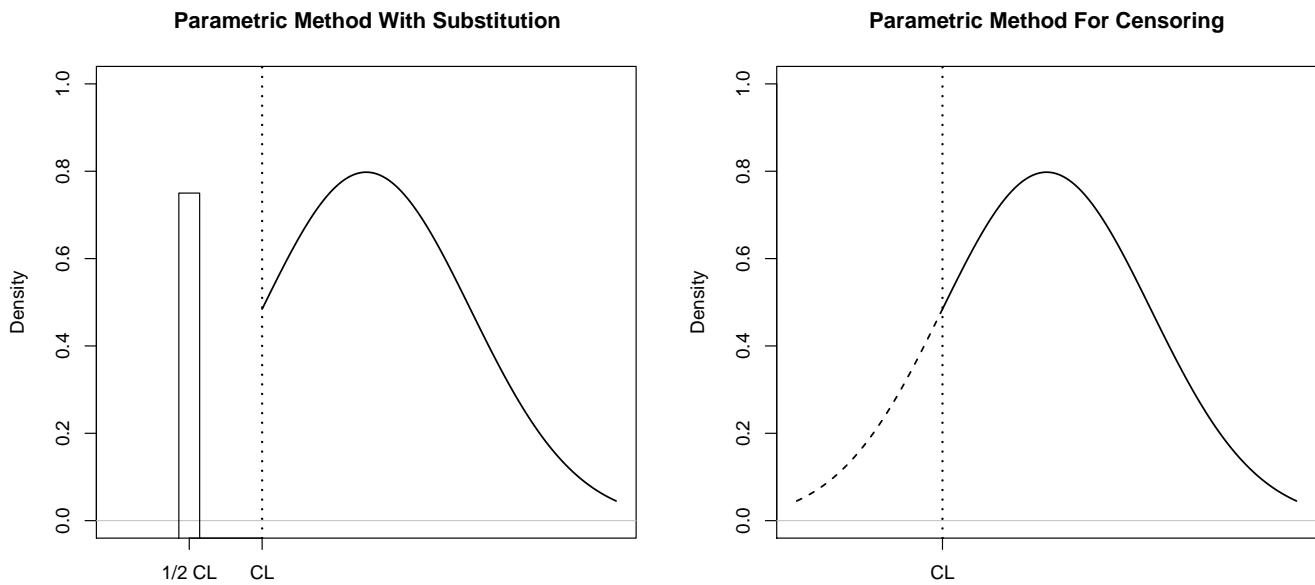
Practical Background on Parametric Methods for Comparing 2 Populations When Censored Observations are present

In Section 9.1 we discussed the use of t-tests to compare the population centers of two independent groups. I also mentioned that they are inappropriate for application on censored data. T-tests are a form of parametric method because we make the assumption that our data follow a specific parametric form: the normal distribution. Estimates of means and standard errors used in the t-test are chosen based on what is mathematically likely given the data and the distribution we are assuming, similar to *maximum likelihood estimates*(MLEs).

In this section, we also discussing likelihood methods that are based on parametric distributions. Here we are specifically defining an underlying distributional form to incorporate censor values into our parameter estimates, rather than arbitrarily substituting values. The conceptual difference in the treatment of the data is illustrated below in Figure 9.3 where data below the CL is imputed as following the same distribution data above the CL. This allows for more sensible estimates of center and variability. Note that when censor values are present, MLE methods are *large sample* methods, and should be used when samples sizes are larger than 30-50.

When using MLE methods which incorporate information from censoring, the assumptions and cautions are similar to those methods which are not specifically intended for censor data. These parametric methods assume that the data approximately follow a certain specified distribution (eg. normal or lognormal), and in the case of comparing two or more groups, that their population standard deviations are equal.

Fig. 9.3: Illustration of the difference between handling censored data with substitution vs. accounting for it directly in estimation



If an inappropriate distribution is chosen, or if there are outliers present, it can affect the test results.

As with t-tests, this implies that model assumption testing has to be performed on the data. This can include normal probability plots of the residuals from the analysis, as well as an examination of boxplots to assess variability.

R Code and Interpretation of a 2 Sample MLE Test

To demonstrate a parametric method for censored data, we will again use the San Joaquin valley data that was presented in Section 9.0.1. From the boxplots that were generated in Section 9.0.2 it is apparent that it is more appropriate to assume a lognormal distribution for the data. Therefore, when illustrating parametric methods below, MLE estimates and tests will be calculated based on a lognormal distribution.

In R, parametrically comparing two groups which contain censored data is performed using the command/function `cenmle(.)` which is part of the NADA library. The full set of commands used to perform a comparison between population means between groups is shown below where the MLE test results are named `fitcenmle`. The `cenmle(.)` function was also demonstrated in Chapter 6.5 but in this case we have added an argument `group=Zone` to indicate that we wish to compare typical concentrations between two groups using likelihood methods.


```
fitcenmle=cenmle(obs=CuZn$Zn,censored=CuZn$ZnCen,group=Zinc$Zone,dist="lognormal")
```

or

```
fitcenmle=with(CuZn,cenmle(obs=Zn,censored=ZnCen,group=Zone,dist="lognormal"))
```

When using the `cenmle(.)` function in R, the argument `dist=` can only be specified as "lognormal" or "gaussian" (normal). If the data appears to follow any distributional shape other than these two, and contains censoring, it is recommended that one of the non-parametric methods shown in Section 9.3 be chosen for analysis instead.

Using the commands listed above in R on the CuZn data set results in the R output seen below.

```
>fitcenmle
```

	Value	Std. Error	z	p
(Intercept)	2.466	0.1078	22.88	6.65e-116
CuZn\$ZoneBasinTrough	0.257	0.1613	1.60	1.10e-01
Log(scale)	-0.171	0.0734	-2.33	1.99e-02

```
Scale= 0.843
```

```
Log Normal distribution
```

```
Loglik(model)= -407.3   Loglik(intercept only)= -408.6
```

```
Loglik-r: 0.1467494
```

```
Chisq= 2.55 on 1 degrees of freedom, p= 0.11
```

```
Number of Newton-Raphson Iterations: 3
```

```
n = 117
```

In this output, the estimated difference in the means of the lognormal zinc data is the value 0.257 found next to the text `CuZn$ZoneBasinTrough`. This difference in the means of the log distributions is associated with the p-value 1.10×10^{-01} , or 0.110. Like the t-test, this parametric test indicates no evidence of difference in population centers between the groups. This somewhat unexpected finding will be discussed during the upcoming section 9.2 on model assumption testing.

It should also be noted that the `cenmle(.)` function performs a test for a 2-sided hypothesis. For an appropriate one-sided hypothesis it will be necessary to either divide the p-value in half, or calculate $1-(p\text{-value}/2)$ depending on the direction of the test.

Recall that when we take logarithms of the data, our hypothesis tests on the original scale are phrased in terms of the median instead of the mean. In this case, the hypothesis is about the ratio of the medians on the original scale, and is formally written below. For more information on

the relationship between means on the log scale, and medians on the original scale please refer to Chapter 6.5

$$H_0 : \frac{\text{median}(\text{basintrough})}{\text{median}(\text{alluvialfan})} = 1$$

$$H_A : \frac{\text{median}(\text{basintrough})}{\text{median}(\text{alluvialfan})} \neq 1$$

Based on our estimated difference on the log scale of 0.257, we would calculate the ratio of the medians to be $e^{0.257} = 1.293$. In written terms, this implies that the median zinc concentration in the basin trough zone is approximately 1.3 times greater than the median concentration in the alluvial fan zone.

Confidence Intervals

At the time of this writing, the function `cenmle` does not provide confidence intervals for the estimated mean difference between groups. The R output does provide a standard error estimate though, making it possible to construct confidence intervals. To construct a confidence interval manually, simply multiply the standard error¹ provided by R to an appropriate critical value from a Z table as outlined below. This value can then be added and subtracted from the estimated mean difference to find an appropriate interval.

$$\text{difference} \pm SE * Z_{\alpha/2} \tag{9.3}$$

Alternatively, I have written a function in R called `DiffCI(..)` which will take a `cenmle` object (such as `fitcenmle`) and calculate an appropriate confidence interval for the difference in population means between groups. To do this will require the user to load the script for the function. The code for the function and the details on how to load it in R can be found in Appendix I.

Using this approach we simply type the following commands into the R console

```
DiffCI(cenobj=fitcenmle,conf.level=0.95,ngroups=2)
```

in order to obtain a 95% confidence interval on the mean difference between groups on the log scale. This code provides the following output:

```
lower      upper
-0.05865096  0.57358015
```

On the original scale, this corresponds to a 95% confidence interval of $(e^{-0.059}, e^{0.573}) \Rightarrow (0.94, 1.77)$ on the ratio of the median zinc concentrations between zones. Notice that the interval contains the value 1, showing no evidence against the null hypothesis of no difference between the ratio of the medians between the zones.

¹Recall that a standard error is a measure of the sampling variability in the mean, but is not a measure of the population variability.

Post-Hoc Model Assumption Testing

As discussed in section 9.2, when using parametric methods, it is necessary to assess the model assumptions to evaluate the appropriateness of the model. In this case, this requires an assessment of whether or not the variances from the two groups are approximately equal, and whether either the normal or lognormal assumption seem appropriate. Boxplots can be used to check for equal variances between the groups, and can also be examined for gross violations of the normality assumptions (recall that after we take the log of our data, it should be approximately normally distributed). For a more in depth look at the normality of the data, probability plots can be used.

Boxplots can be drawn as was shown in Section 9.0.2, and for this data are the same as the ones found in Figure 9.2. Looking at these boxplots, there is no evidence that the equal variance assumption is violated. Due to the presence of outliers, it is difficult to determine whether a normal data assumption is appropriate, so we need to look at a normal probability plot (introduced in Chapter 5.2) of residuals from the model.

We can construct a probability plot for the residuals from our model using the R-code below.

```
res=residuals(fitcenmle)
forresplot=with(Zinc,ros(res,ZnCen,forwardT=NULL))

plot(forresplot)
```

The first command in the sequence, `res=residuals(fitcenmle)` creates an R object called `res` that contains the residuals from our `cenmle` model. The second line of code, creating the object `forresplot` creates an `ros` object so that we can create a censored probability plot. In the example code above, we have added one extra argument to the `ros` function, `forwardT=NULL`. This command tells R that no forward transformation of the data is necessary, because we log-transformed it ourselves in the arguments to the `cenmle` function. If this argument is not included, R will automatically log-transform the data. The final function, `plot(...)`, creates the desired probability plot.

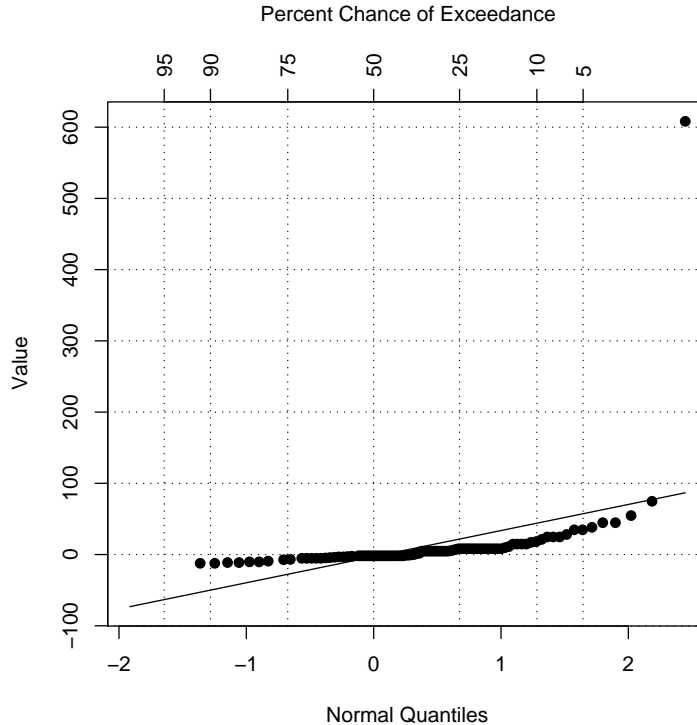
Executing the code above in the R console results in the graph shown below in Figure 9.4 being drawn.

In Figure 9.4, notice that although a majority of the data fall fairly close to the normal line², there is one outlier at a concentration of near 600 that falls extremely far from the line. This outlier causes the data to violate the normality assumption, and causes the model to fit poorly. This point is likely the reason that even though we are accounting for censoring using the MLE method, we still cannot find a significant difference in the mean zinc concentrations between the two geomorphic zones.

We will see in Section 9.3 below that non-parametric methods can account for both censoring and the presence of outliers in the data when looking for a difference between the two groups.

²The normal line is the line we would expect points to fall on if the data came from a perfectly normal distribution

Fig. 9.4: Normal probability plot of residuals from cenmle fit of the CuZn data



9.3 What To Do: Non-Parametric Methods

Parametric methods are large sample methods, sensitive to the choice of distribution, and sensitive to outliers. Conversely, non-parametric methods can be used with large or small sample sizes, make fewer assumptions about the distribution of the data, and are more robust to the influence of outliers. The disadvantage of non-parametric methods is that they are not as powerful at detecting differences between group centers when there is a large sample size, no outliers, and a clear distributional choice.

Assumptions for non-parametric test include that the variability of the different groups should be similar, and that the shape of the distributions for the different groups should be similar. For example, right skewed data should not be compared to left skewed data using non-parametric tests. On the other hand, if both sets of observations indicate similar skew, results from a non-parametric test should be valid.

To assess differences between distributions from different groups, non-parametric methods essentially combine all of the data, and then rank it. The data is then split back into two groups, and the relative rankings between the two groups are assessed. If one group has enough values with higher rankings, then it implies that the percentiles of that group tend to be higher, and that there might be a difference between groups. Censor values are incorporated by assigning all observations below the censor limit the same ranking.

9.3.1 Non-parametric tests when there is only one detection limit

When only one detection limit is present, all values below the detection limit share the same tied rank. This rank is lower than that for the smallest detected observation. In this way, percentiles of two groups can be compared, and the test incorporates information on censored values. The test using this method is commonly called the *Mann-Whitney* test, or the *Wilcoxon rank sum*.

Because data transformations such as the log transformation maintain the ordering of data, it is not necessary to transform the data prior to applying a score test.

In our example data set comparing zinc concentrations in the San Joaquin valley, there are two CLs: one at 3 $\mu\text{g/L}$, and one at 10 $\mu\text{g/L}$. Because we are discussing non-parametric methods where only one CL is present, we will use the zinc data set, but treat all observations $< 10 \mu\text{g/L}$ as censored at 10 $\mu\text{g/L}$. Some information is lost in this step, but the power of the test may still be sufficient to detect a difference in medians between groups.

To perform a Mann Whitney test on the zinc data in R, the commands shown below are used:

```
Zn.rank.test=wilcox.test(CuZn$Subs~CuZn$Zone,alternative='two.sided',conf.int=TRUE,
conf.level=0.95)
```

or

```
Zn.rank.test=with(CuZn,wilcox.test(Subs~Zone,alternative="two.sided",conf.int=TRUE,
conf.level=0.95))
```

Note that the variable `Subs` is referring to a data column where all values less than 10 have been treated as censored at 10. The arguments available for the `wilcox.test` command are very similar to those for the `t.test` command, and they may be specified in the same way as for the t-test in section 9.1 to achieve the desired confidence intervals and test direction.

Using the R commands above results in the following output from R.

```
> Zn.rank.test
```

```
Wilcoxon rank sum test with continuity correction
```

```
data: Zinc$Subs by Zinc$Zone
W = 1255.5, p-value = 0.01846
alternative hypothesis: true location shift is not equal to 0
95 percent confidence interval:
 -1.000001e+01 -4.133280e-05
sample estimates:
difference in location
 -4.999977
```

Even with the loss of information that occurs by censoring all of the values below 10 $\mu\text{g/L}$, this test still calculates a p-value of 0.018 which indicates that there is evidence of a difference between group medians at the 5% level. The estimated difference between the two groups on the original scale is $-4.99\mu\text{g/L}$ with an associated 95% confidence interval of the difference being between values of $(-10, 0)$.

Because this is a non-parametric test, little assumption testing is required to validate the model. In this case we can conclude that the variances are similar, and that the distributions have similar shapes by examining the boxplots in Figure 9.2

9.3.2 Non-parametric tests when there are multiple detection limits

In the above example, we artificially censored all values below 10 $\mu\text{g/L}$. Although there was evidence of a difference between the group centres (medians) in this example, this may not always be the case. When multiple detection limits are present, it is appropriate to use a score test that incorporates information from multiple detection limits.

To illustrate how this type of score test works, I will discuss below how to calculate a Gehan test statistic (Gehan, 1965). The score test implemented in NADA, called the *Peto Prentice*, or *generalized Wilcoxon* test represents a slight variation to the Gehan test which improves its robustness (Peto and Peto (1972), Prentice (1978), and Prentice and Marek (1979)).

In this test, all possible comparisons are made between observations from different groups, determining whether observations are greater than, less than, or tied to each other. The evaluation of the differences, U_{ij} 's are calculated as shown in equation 9.4. The test statistic then becomes $U = \sum_{i=1}^n \sum_{j=1}^m U_{ij}$ where m and n are the number of observations in each group.

$$U_{ij} = \begin{cases} -1 & \text{when } x_i > y_j, \\ +1 & \text{when } x_i < y_j, \\ 0 & \text{for } x_i = y_j \text{ or} \\ & \text{for indeterminate comparisons (<10 vs. 5)}. \end{cases} \quad (9.4)$$

A toy example showing how the U_{ij} 's are calculated is shown below in Table 9.2

As stated above, one advantage to this test is that it allows for multiple left censoring limits to be present. As with the Mann-Whitney test, it can compare population medians even when samples have different numbers of observations.

To implement the generalized Wilcoxon test in R on the zinc concentration data, the command `cendiff` is used as shown below to create an object called `Zn.Wilcox`.

```
Zn.Wilcox=cendiff(obs=CuZn$Zn,censored=CuZn$ZnCen,groups=CuZn$Zone,rho=1)
```

Table 9.2: Toy example for Gehan non parametric test

Group1	81.3	4.9	4.6	3.5	3.4	3	2.9	0.6	0.6	<0.2
Group2										
1.4	1	1	1	1	1	1	1	-1	-1	-1
0.8	1	1	1	1	1	1	1	-1	-1	-1
0.7	1	1	1	1	1	1	1	-1	-1	-1
<6	1	1	1	1	1	1	1	1	1	0
0.4	1	1	1	1	1	1	1	1	1	-1
0.4	1	1	1	1	1	1	1	1	1	-1
0.4	1	1	1	1	1	1	1	1	1	-1
<0.4	1	1	1	1	1	1	1	1	1	0
<0.3	1	1	1	1	1	1	1	1	1	0

or

```
Zn.Wilcox=with(CuZn, cendiff(obs=Zn,censored=ZnCen,groups=Zone,rho=1))
```

Notice the argument `rho=1`. This is what specifies a generalized Wilcoxon test be used to calculate the differences between groups. Implementing this method results in the following output from R.

```
> Zn.Wilcox
```

```

              N Observed Expected (0-E)^2/E (0-E)^2/V
Zinc$Zone=AlluvialFan 67      31.9      38.7      1.20      5.18
Zinc$Zone=BasinTrough 50      30.1      23.3      1.98      5.18
```

```
Chisq= 5.2 on 1 degrees of freedom, p= 0.0228
```

Running this test in R results in a p-value of 0.0228, similar to that found for the Mann-Whitney test. This implies that the information in the data that can be attributed to the arrangement of observations below 10 is relatively not as important as the information found in the number of observations below and above 10.

Notice that using this generalized Wilcoxon method in R does not result in the creation of a confidence interval for the median difference between groups. For an approximate confidence interval, the interval obtained using the Mann-Whitney test can be used. Theoretically, this interval is slightly wider than would be found based on the Wilcoxon test, but will still provide information on the variability of the data.

As with the Mann-Whitney test in Section 9.3.1, few assumptions have been made on this model, and our main concerns of equal variances and similar distributions have already been discussed.

Chapter 10

Comparisons Between Paired Observations

T-tests can occur in the two independent sample case, as was discussed in Chapter 9. T-tests are often also used to examine the differences in paired data. Paired data occurs when there is some dependency or matching occurring due to observations being measured on the same experimental unit. The typical example of paired data involves a discussion of identical twins, one twin who receives a treatment, and one twin who does not. By defining the mean difference as the difference between *twins* as opposed to the difference between independent treatment *groups*, we are able to control for factors known to influence the variability of the data. For example, in the case of twins we will have successfully controlled for variability in the data caused by genetic differences between people.

A weakness of the twins example is that with twins, it is completely obvious what the pairing characteristic is. Such pairing characteristics are not always as obvious as being genetically identical. In water quality, if measurements are taken upstream and downstream on the same day, the measurements could be considered paired due to being matched on day. If measurements were taken on different days this would no longer be true. Using day as the pairing factor would reduce the influence of day to day variability on the measurements. Another example might occur if we sampled water in the same locations, but at different times of the year. In this situation observations are matched based on location, and we are able to detect a seasonal effect while controlling for location to location variability. Paired tests can also be used to compare data to see if it falls above or below a standard on average.

One key feature of data that is paired is that the number of observations in each group is equal due to matching. There are the same number of observations before and after, upriver and downriver, etc. Although there are some specialized paired designs where the number of observations differs between groups, they will not be discussed here.

When dealing with paired data, hypothesis tests take a form identical to that discussed in Chapter 9, so will not be discussed again here. The difference is in the form of the test. In chapter 9, we calculated means or medians based on the whole group, and then calculated a difference. In this test we take the difference between each pair of observations, and then perform tests on differences between measurements.

At this point in the document, it should be clear that it is never appropriate to use substitution to account for non-detect values. Substitution will always affect results in unpredictable ways, and this is not scientific, or acceptable in scientific reporting. Although it is possible to find examples of how substitution and statistical methods that don't account for censoring will result give incorrect conclusions, this document is meant to focus on what *to do* in order to perform good inference. To this end, this chapter and those following will focus exclusively on correct analysis methods for data including non-detects.

10.0.1 Paired Data Example

The example data set for this section focuses on the effect of herbicide application on groundwater quality. Measurements were taken on 24 groundwater wells in an agricultural area during the spring (June). Atrazine, a commercial herbicide used under many product names, was applied during the growing season. At the end of the growing season, in September, water samples were taken on the *same* 24 wells to examine whether typical atrazine concentrations in the water had increased due to atrazine application at the surface. Notice that we are interested in a potential *increase* in atrazine concentrations, meaning that this analysis is associated with a one-sided alternative hypothesis as shown below. In this hypothesis we are once again using θ to represent our parameter of interest, be it the population mean or median. Our data are paired on location because we are taking multiple measurements on the same wells.

$$\begin{aligned} H_0 &: \theta_{June} \geq \theta_{September} \\ H_A &: \theta_{June} < \theta_{September} \end{aligned} \tag{10.1}$$

To view the data set in R, make sure that the NADA library has been loaded, and then type `data(Atrazine)` in the R console window. If you type `Atrazine` in the R console, you should see a data set that looks something like Table 10.1 below.

Table 10.1: Groundwater concentrations of atrazine in June and September

Atra	Month	AtraCen
0.38	June	FALSE
0.04	June	FALSE
0.01	June	TRUE
⋮	⋮	⋮
0.03	Sept	FALSE
88.36	Sept	FALSE

Notice that the format of this table does not make it clear that observations in June and September are paired. This could easily be mistaken for two sample data. To put the data in a more intuitive format given the paired nature of the observations, the following R code can be used to manipulate the observations.

```

#first separate the data into June and September
#components of equal length
June=Atrazine[Atrazine$Month=="June",]
September=Atrazine[Atrazine$Month=="Sept",]

#create a variable that numbers each pair
#of observations
obs=rep(1:NROW(June),1)

#bind the numbering to the June and
#September observations
June=cbind(June,obs)
September=cbind(September,obs)

#merge the data back together such that paired
#observations are placed in the same row
pairedata=merge(June,September,by="obs")

```

Executing these commands in R will result in a data set in a format similar to table 10.2, where observations taken on the same well in different months occur in the same row. A `.x` after a variable name indicates it comes from the June data, and a `.y` indicates information pertaining to the September data. R automatically introduces this convention when data are merged.

Table 10.2: Groundwater concentrations of atrazine in June and September, paired data format

obs	Atra.x	Month.x	AtraCen.x	Atra.y	Month.y	AtraCen.y
1	0.38	June	FALSE	2.66	Sept	FALSE
2	0.04	June	FALSE	0.63	Sept	FALSE
3	0.01	June	TRUE	0.59	Sept	FALSE
4	0.03	June	FALSE	0.05	Sept	FALSE
5	0.03	June	FALSE	0.84	Sept	FALSE
⋮	⋮	⋮				

In upcoming sections it is assumed that when the atrazine data is being analyzed it is in the format of table 10.2, or the `pairedata` data set that was created by the earlier R commands.

10.0.2 Preliminary Inference: Graphing and Data Transformation

As with most statistical problems, there is valuable information to be gained from graphing the data. In this case we are interested in the *differences* between pairs of observations. Consequently the natural quantity to graph is the difference between September and June atrazine concentrations. Notice that when we take differences of pairs where one or both of the observations is censored,

the difference will fall between an interval of values. In statistical terms, the data will be *interval censored*. I have written a function in R called `makePaired(..)` that will calculate the differences between paired data sets when left censoring can occur on the individual observations. This function can be found in Appendix J. It can be installed for use in R using the same method as for the function `DiffCI(..)` in appendix I.

Once this function has been loaded into R, it can be used to calculate the differences between observations in data sets where left censoring occurs using the following command in R. Recall that we created the `pairedata` data set using the R commands shown for data manipulation in section 10.0.1.

```
Atra.diff=makePaired(pairedata)
```

The paired data set that is created by the above command is returned as two columns. Recall that subtracting observations where left censoring is present results in an interval range of possible differences. For example, if our June observation is censored at $0.01 \mu\text{g/L}$, and our September observation is $0.59 \mu\text{g/L}$, then the value of the difference would range from $(0.58, 0.59) \mu\text{g/L}$. The first column in the `Atra.diff` data set is the upper bound of this interval (`diffU`), and the second column of this data is the lower bound of this interval (`diffL`). In the case of observations where there is no censoring, the upper and the lower bound will be identical. A sample of how this paired data set might look is shown in table 10.3 below. The differences in `Atra.diff` are in the same order as the pairings in the `pairedata` data set.

Table 10.3: Sample of the output returned by the `makePaired(..)` function

diffU	diffL
⋮	⋮
0.81	0.81
0.53	0.53
0.00	0.00
0.01	-0.01
0.01	-0.01
⋮	⋮

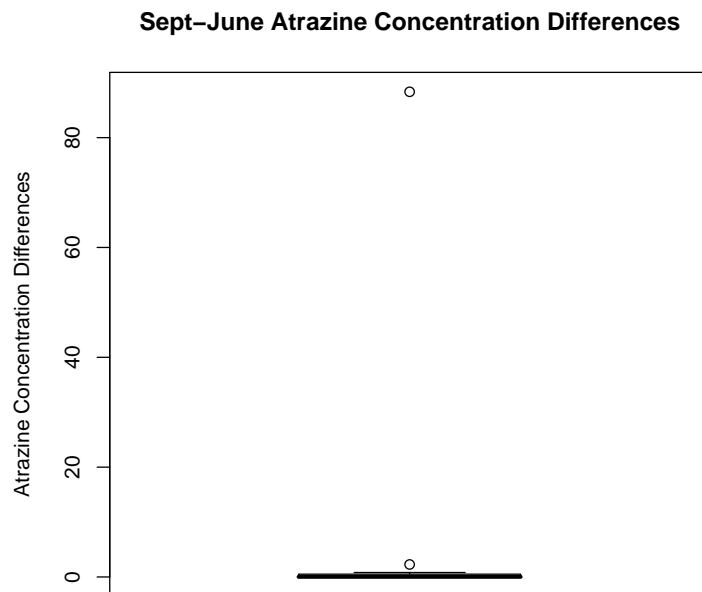
One consequence of calculating the differences between the original concentrations is that the difference data illustrated in table 10.3 are now interval censored. Although looking at a graphical representation of the data will still be useful, the interval censored nature of the data will limit our ability to interpret plots. Despite this, we can still make general observations about any skewness in the data set, and have a relative idea of where the center of the data falls.

Boxplots for this data need to be drawn on one difference column or the other. We will draw a boxplot based on only one of the columns. Either column results in similar information being shown in the plot, so we will select the `diffU` column. To draw the boxplot in R, simply use the following command to create the boxplot seen in Figure 10.1.

```
boxplot(x=Atra.diff$diffU, main="Sept-June Atrazine Concentration Differences")
```

Notice that we use the `boxplot(...)` command rather than the `cenboxplot(...)` command because `cenboxplot(...)` is only for use on left censor data sets, and will not incorporate interval censor data.

Fig. 10.1: Atrazine difference boxplot



It is evident from the boxplot in Figure 10.1 that there are some extreme outliers in the data, and the data is right skewed. Although the presence of skewness and outliers is not a problem for non-parametric methods, a transformation of the data seems desirable in order to improve model assumptions for parametric methods. The instructions on how to log transform the concentration data, and recalculate the differences on the log-scale are shown below.

```
#transforms the concentration values from June onto the log scale  
pairedata$Atra.x=log(pairedata$Atra.x)
```

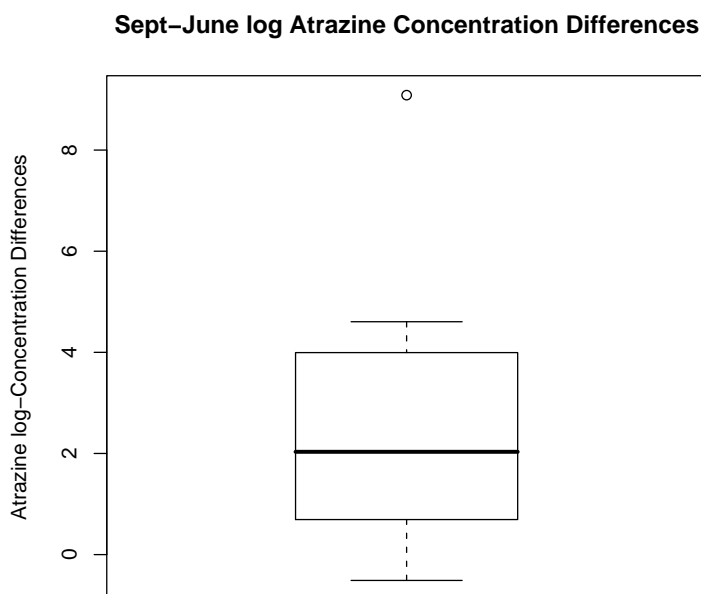
```
#transforms the concentration values from Sept. onto the log scale  
pairedata$Atra.y=log(pairedata$Atra.y)
```

```
#re-calculates the differences, but on the log scale
```

```
Atra.diff=makePaired(pairdata)
```

After log transforming the data, and recalculating the Sept-June differences on the log scale, we can use the difference data to create the boxplot shown in Figure 10.2. This boxplot was created using the same R boxplot commands as found above. Notice how log transforming the data has made the distribution much more symmetric and appropriate for parametric testing using a normal assumption.

Fig. 10.2: Log-atrazine difference boxplot



10.1 Parametric Tests for Paired Data

Once we have taken the difference calculation, performing a parametric/MLE test of whether the average difference is statistically significant is fairly straightforward. Practically speaking, the commands to perform this in R are quite similar to those found for estimation using parametric methods in the first guidance document. The biggest change here is that we need to accommodate the methods to interval censored data, and calculate a p-value to assess whether the is for or against the null hypothesis of no difference¹.

To conduct a parametric test with our data, we use the R commands below.

¹Because we are now using log-transformed data, recall that our null and alternative hypotheses are about the ratio of the medians of the two groups on the original scale

```
paired.mle=survreg(Surv(time=Atra.diff$diffU,time2=Atra.diff$diffL,type="interval2")~1,
dist="gaussian")
```

Notice that we have selected the option `dist="gaussian"`. We have already log transformed the data to a more normal shape. Additionally, this option should not be used to naively evaluate the data on the log scale as a short cut, because it would calculate the differences incorrectly.

Using the command `summary(paired.mle)` in the R console shows us the following output from our MLE estimation.

```
> summary(paired.mle)
```

Call:

```
survreg(formula = Surv(time = Atra.diff$diffU, time2 = Atra.diff$diffL,
type = "interval2") ~ 1, dist = "gaussian")
```

	Value	Std. Error	z	p
(Intercept)	1.294	0.337	3.84	0.000123
Log(scale)	0.349	0.175	1.99	0.046773

Scale= 1.42

Gaussian distribution

Loglik(model)= -29 Loglik(intercept only)= -29

Number of Newton-Raphson Iterations: 6

n= 24

The quantity of interest in this model is the `(Intercept)` with an estimated value of 1.294. This estimate represents the mean difference on the log scale between September and June atrazine concentrations. On the original scale, this corresponds to a ratio between the medians of $e^{1.294} = 3.65$, indicating that the median concentration is 3.65 times higher in September than in June. The p-value associated with this test is two sided, so to make it appropriate for our one-sided test we need to divide by two, giving us a p-value of 0.0000615.

As with an earlier example in Section 9.2, this summary of the analysis does not include a confidence interval. One can be calculated and reported manually, as was also shown in 9.2. Alternatively, I have written a function in R (`PairDiffCI(.)`) that returns the estimated difference, a two sided p-value, and a confidence interval. The code that needs to be loaded into R for this function to work can be found in Appendix K. A demonstration on the use of this function, and the information that it returns is shown below.

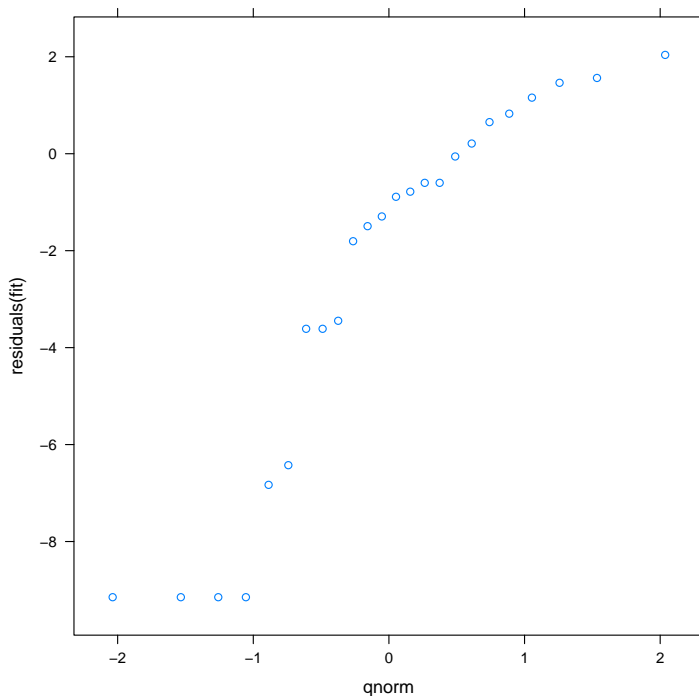
```
>PairDiffCI(obj=paired.mle,conf.level=0.95)
      lower      estimate      upper      p.value
0.6334261229 1.2936472269 1.9538683308 0.0001228425
```

The confidence interval about the mean that is returned, $(0.633, 1.953)$, clearly does not contain zero. This suggests that atrazine concentrations in groundwater have increased between June and September.

Before ending the analysis with those conclusions, it is always valuable to perform some model assumption testing. The R commands below can be used to make a normal probability plot of the residuals as is shown in Figure 10.3.

```
library(lattice)
assumptest=residuals(paired.mle)
qqmath(x=assumptest,distribution=qnorm)
```

Fig. 10.3: Normal Probability Plot of Residuals From Parametric Atrazine Analysis



Drawing an imaginary normal line through the points in Figure 10.3 shows that the fit is not horrible, but not perfect. There is some evidence that the data is a bit right skewed, which could make the results from the MLE estimation more significant than they ought to be in the positive direction. A good check on the conclusions from this MLE test is to see if they concur with the conclusions we would draw from a non-parametric paired test.

If they differ, and there is evidence of any assumption violations, then the results from the non-parametric test should be used. The disadvantage of the non-parametric test in this case is that a confidence interval estimate is not available for it.

10.2 Non-Parametric Testing For Paired Data

Non-parametric tests for paired data are very similar in spirit to non-parametric tests used when comparing two groups as was discussed in Section 9.3. When using data with paired observations, non-parametric tests are commonly called *sign* tests. This is because the test is evaluating the sign of the differences. Differences can be positive, negative, or tied (two left censored values would be considered tied). If there is no effect on average, the distribution of differences should be centered around zero. This means that there should be an approximately equal number of positive and negative differences in the sample. Some example sign of difference calculations are shown below in Table 10.4. This table is based on a portion of the atrazine concentration values.

Table 10.4: Table including sign of difference evaluations for the atrazine data

June	Sept.	Upper Diff,	Lower Diff.	Sign of Difference
⋮	⋮	⋮	⋮	⋮
0.03	0.84	0.81	0.81	+
0.05	0.58	0.53	0.53	+
0.02	0.02	0.00	0.00	0
<0.01	<0.01	0.01	-0.01	0
<0.01	<0.01	0.01	-0.01	0
⋮	⋮	⋮	⋮	⋮

While the simplest versions of the sign test do not take into account tied values, in the method I am demonstrating, tied values are incorporated into the test according to a method outlined by Fong et al. (2003). This method is called a *modified sign test*.

Although the modified sign test is not available in commercial statistical software packages, I have written R functions to compute summaries of the difference signs, and the exact p-value based on the Fong test. The code for these functions is shown in Appendix L. How to use these functions is demonstrated on the atrazine data as shown below.

First, to calculate the number of positive, negative, and tied differences, we can use the function `signs(...)`. This function requires the output that is returned from the `makePaired(...)` function that was discussed earlier. In previous example, we named this output data set `Atra.diff`.

```
Atra.signs=signs(pairs=Atra.diff)
```

Using the command in R results in the following output.

```
> Atra.signs
[[1]]
```


	diffU	diffL	sign
1	2.28000009	2.28000009	1
2	0.59000000	0.59000000	1
3	0.58999997	0.57999997	1
4	0.02000000	0.02000000	1
5	0.80999997	0.80999997	1
6	0.52999998	0.52999998	1
7	0.00000000	0.00000000	0
8	0.01000000	-0.01000000	0
9	0.01000000	-0.01000000	0
10	0.01000000	-0.01000000	0
11	-0.02000000	-0.02000000	-1
12	0.22000000	0.22000000	1
13	0.02000000	0.01000000	1
14	0.01000000	-0.01000000	0
15	0.50000000	0.49000000	1
16	0.03000000	0.02000000	1
17	0.07000000	0.07000000	1
18	0.03000000	0.03000000	1
19	0.01000000	0.01000000	1
20	-0.01000000	-0.02000000	-1
21	0.05000000	0.05000000	1
22	0.22000000	0.22000000	1
23	-0.02000000	-0.02000000	-1
24	88.36000061	88.35000061	1

```
[[2]]
[1] 16
```

```
[[3]]
[1] 3
```

```
[[4]]
[1] 5
```

The signs of each difference are located next to the interval censored differences. Here, 1 indicates a positive difference, 0 indicates a tie, and -1 indicates a negative difference. Below this are 3 numbers representing the number of positive differences(16), negative differences(3), and tied differences(5) respectively. This output, named `Atra.signs` can then be passed to the `calcP` function. This function will calculate the modified sign test p-value and return it.

```
Non.Par.paired=calcP(Atra.signs)
```

Executing this command returns the following p-value.

```
> Non.Par.paired
[1] 0.0895632
```

The `calcP(.)` function produces a two-sided p-value on a test with a null hypothesis of no difference. For our test, we are interested in a one sided p-value, so the actual p-value for the hypothesis being evaluated is $0.0895632/2 = 0.0447816$. This p-value is less than 0.05, indicating that there is evidence against null hypothesis, and conclude that there is a difference in typical atrazine concentrations based on the paired June and September groundwater measurements. This conclusion agrees with the results we found in Section 10.1, where we had an observed p-value of $p = 0.0000615$ (compared to the current 0.0448). Despite this agreement, the p-values from the two tests are several orders of magnitude different. This is most likely due to the fact that the data had large positive outliers making the magnitude of the difference between groups seem larger, and artificially lowering the reported p-value for the parametric method. The sign test is a more robust test in the presence of outliers, and suggests slightly less evidence against the null hypothesis of equality in the mean difference, as demonstrated by its larger p-value.

10.3 Comparing Data to a Standard

An important activity in many monitoring programs is evaluating whether concentrations of contaminants are exceeding safety standards. Testing data with censored values against a standard is straightforward. In the atrazine data we had two sets of observations, one set from June, and one set from September. When testing against a standard, there will only be one set of measured observations. Instead of a second set of observations, we can merely insert a column containing the standard to be tested against. If we imagine that we are testing the June atrazine data to a standard of $0.25 \mu\text{g/L}$, the data would look similar to that demonstrated in Table 10.5

Table 10.5: Groundwater concentrations of atrazine in June and September

obs	Atra.x	Month.x	AtraCen.x	Standard	Spacing	AtraCen.standard
1	0.38	June	FALSE	0.25	space	FALSE
2	0.04	June	FALSE	0.25	space	FALSE
3	0.01	June	TRUE	0.25	space	FALSE
4	0.03	June	FALSE	0.25	space	FALSE
5	0.03	June	FALSE	0.25	space	FALSE
⋮	⋮	⋮				

In table 10.5 the number and ordering of the columns has been kept the same as was is table 10.2. To achieve this, we introduced a ‘spacing’ column. This is because to test data against a standard, we simply use paired comparison methods as outlined in sections 10.1-10.2. For the functions created to do this to work, the data must have the same number of columns in the same order as when conducting a regular paired comparison test.

Chapter 11

Comparisons Between Multiple Groups

In Chapters 9 and 10, emphasis was placed on discussing hypothesis tests and confidence intervals; explaining and demonstrating parametric methods; explaining and demonstrating non-parametric methods; and discussing how interpretation changes when log transformations are used on the data. These topics all extend very directly into methods for comparing parameters from more than two groups. Comparing parameters from two groups and comparing parameters from multiple groups are conceptually similar activities, and the software techniques are also similar.

Because of the similarities in approaches, this chapter, 11, will be short and straightforward showing the extensions of two group analysis to multiple group analysis.

When comparing multiple groups it is common to employ a two stage approach to hypothesis testing. Initially, a general test is run which determines whether or not the centers of any of the populations are significantly different based on the observations. If this initial test indicates that there are differences between the any two population centres, subsequent analyses are often run to determine which groups are different from one another.

The general form of a null and alternative hypothesis formulation for evaluating whether there are differences between the parameters in k different groups is shown below.

$$\begin{aligned}H_0 &: \theta_1 = \theta_2 = \theta_3 \dots = \theta_k \\H_A &: \theta_i \neq \theta_j, i \neq j \text{ for at least one } i, j\end{aligned}$$

This test indicates whether at least one group is significantly different from another. Subsequent tests can evaluate which group(s) is(are) different, and can take various forms depending on the research questions of interest. Generally speaking, it is good if the researcher has some intuition regarding which groups centers might be different rather than engaging in blanket testing of all possible group differences. More discussion on contrast tests between groups will be discussed below in Sections 11.2 and 11.3

11.0.1 Example Data

Prior to approving applications for development in watersheds, it can be useful to collect baseline information about underlying pollutant concentrations in an area. In this example, data were collected at five different sampling points along Toboggan Creek near Smithers, British Columbia. A map of the sampling region can be found in Appendix M. A sample of the data can be seen in Table 11.1, and the complete data set used for analysis can be found in text form in Appendix N. The data were collected from five different sampling regions, with between eight and nine samples taken at each location during a single year.

This data has one censor limit located at $0.2 \mu\text{g/L}$, and 45% of the observations are censored.

Table 11.1: Vanadium concentrations ($\mu\text{g/L}$) at different sampling locations along Toboggan Creek

Vanadium	VanCen	Location	LocCode
0.2	TRUE	Railway	E271363
0.2	TRUE	UpperTCreek	E245607
0.2	TRUE	Hatchery	E245367
\vdots	\vdots	\vdots	\vdots
0.2	FALSE	Hatchery	E245367
0.3	FALSE	Hwy16	E245370

Solutes other than Vanadium were measured for concentration. Vanadium concentrations were chosen for this analysis because its proportion of censoring ($< 50\%$) make it suitable for analysis using the methods being described.

Because we are interested in establishing baseline analyte¹ concentrations, we have no concern that one sampling location might have higher or lower analyte levels than another. At the same time, it is of interest to learn whether some locations typically have underlying analyte levels that are higher or lower than others. In the event of development in the region, such variability by location would have to be incorporated in any monitoring program.

11.0.2 Preliminary Data Inference: Graphing

As in Chapters 9 and 10, graphing data is an important part of exploratory data analysis. The commands in R to create a censored boxplot when multiple groups are present are the same as for when two groups are present. This means that boxplots showing measured vanadium concentrations can be constructed using either of the following commands in R.

```
cenboxplot(obs=Vanadium$Vanadium,cen=Vanadium$VanCen,group=Vanadium$Location,log=FALSE,
range=1,main="Vanadium Concentration at 5 Sampling Locations",
ylab="Vanadium Concentration",xlab="Location")
```

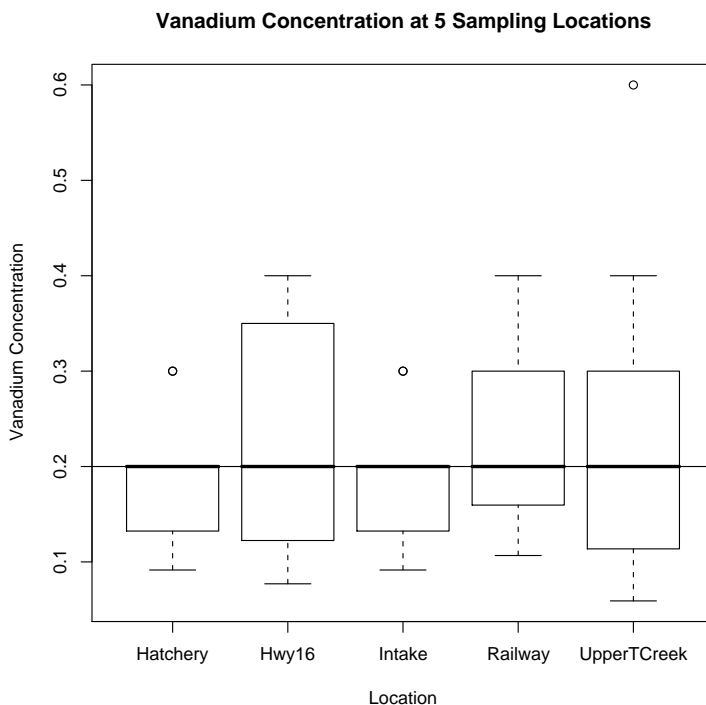
¹An analyte is defined to be a chemical or other substance in water that is the subject of analysis

or

```
with(Vanadium, cenboxplot(obs=Vanadium,cen=VanCen,group=Location,log=FALSE,range=1,  
main="Vanadium Concentration at 5 Sampling Locations",ylab="Vanadium Concentration",  
xlab="Location"))
```

Using either of these two commands in R results in the boxplots shown in Figure 11.1.

Fig. 11.1: Side by side boxplots for vanadium data



Examining the boxplots, there is no obvious evidence of skewness in the data that would require a log-transformation of the data. The Hatchery and Intake locations have more than 50% of their observations at or below the CL. This similarity is not surprising considering that these two locations are relatively nearer to one another than the other locations. The remaining groups have at least 50% of their observations at or above the CL.

From the boxplots, I have no strong intuition that any of the locations have vanadium concentrations that are typically higher or lower than one another because the median line occurs at the same concentration in the data for all sampling locations. This is as we expected because we are attempting to establish a baseline, and there are no known vanadium sources in the vicinity.

Despite this, some caution should be used when interpreting both the boxplots in Figure 11.1, and subsequent analysis of the vanadium data. There are only 8 to 9 observations in each group. Unless the concentration differences between the group centers are large, the sample size is likely

insufficient to detect them. Conversely, if too many sample observations are taken, it will always be possible to find evidence of a difference between groups, even if the size of the difference in concentrations is not environmentally relevant.

11.1 What Not To Do

ANalysis Of VAriance (ANOVA) is the usual method for comparing whether there are differences between group means in the world of non-censored normally distributed data. ANOVA tests are an extension of two sample t-tests from Chapter 9, with the same model assumptions of equal variances between populations, normally distributed data, and independent observations in different groups.

Substituting values for observations below the CL will cause problems similar to those seen when substitution was used in the two sample case. The reasons for this are similar. Substitution artificially affects both mean and variance estimates, which causes violations in our assumptions that data are normally distributed and that variances are equal between groups. How substitution will influence test results is unpredictable. It cannot be emphasized too much that the substitution solution to censor data should always be avoided.

11.2 Parametric Methods for Multi-Group Data

The parametric maximum likelihood methods discussed here have assumptions similar to those for ANOVA tests. We are assuming an underlying distributional form for the data, such as normal or lognormal. We also have assumptions of equal variances between different groups, and independent observations in different groups. The important difference is that the parametric methods discussed in this section directly account for the presence of censored values in the data set.

Prior to performing a test using parametric methods, it is useful to assess assumptions as much as possible. Looking at the boxplots in figure 11.1, there is no obvious difference in the size of the boxes, indicating that the equal variances assumption is reasonable. Additionally, there is no obvious skewness in the data that would indicate the need for a transformation of the data.

After this preliminary assessment of model assumptions, a parametric test for the differences in vanadium concentrations at different locations can be conducted using the following commands in R.

```
van.mle.fit=cenmle(obs=Vanadium$Vanadium,censored=Vanadium$VanCen,  
groups=Vanadium$Location,dist="gaussian",conf.int=0.95)
```

Executing this code in R results in the following output. The reasons that only four locations are explicitly included in the output will be explained in greater detail in Chapter 11.2.2.

```
> van.mle.fit
```

	Value	Std. Error	z	p
(Intercept)	1.88e-01	0.0375	5.00e+00	5.80e-07
Vanadium\$LocationHwy16	4.99e-02	0.0542	9.20e-01	3.58e-01
Vanadium\$LocationIntake	2.57e-17	0.0530	4.84e-16	1.00e+00
Vanadium\$LocationRailway	4.54e-02	0.0525	8.65e-01	3.87e-01
Vanadium\$LocationUpperTCreek	7.38e-02	0.0527	1.40e+00	1.62e-01
Log(scale)	-2.24e+00	0.1199	-1.87e+01	4.95e-78

```
Scale= 0.106
```

```
Gaussian distribution
```

```
Loglik(model)= 4.5   Loglik(intercept only)= 3.1
```

```
Loglik-r: 0.2535912
```

```
Chisq= 2.92 on 4 degrees of freedom, p= 0.57
```

```
Number of Newton-Raphson Iterations: 3
```

```
n = 44
```

Firstly, this output indicates that there is no evidence of differences between the average measurements at the five locations. The overall test for the significance of the model is presented as a χ^2 test on four degrees of freedom, **Chisq= 2.92 on 4 degrees of freedom, p= 0.57**. Technically at this point testing is finished. There is no evidence of differences among the group centers, so it is not necessary to conduct further analyses to determine which groups might be different from each other. Despite this, for educational purposes we will discuss how to further interpret this type of model and create confidence intervals for the differences between groups.

The (**intercept**) row of output has a value of 0.188, which represents the mean of the Hatchery group. This mean has a small p-value, which should *not* be taken to mean that the Hatchery group is significantly different from the other groups. This p-value is simply saying that the mean of the Hatchery group is not zero, which would be true of all the groups.

11.2.1 Confidence Intervals

Confidence intervals on the differences in means among groups are constructed in the same way as in chapter 9.2. A function was written for R, and can be found in Appendix I. As before, this function is called **DiffCI**, and takes a **cenmle** object and creates confidence intervals for it. We need to inform the function of the number of groups that are to be tested, but otherwise the code remains the same. To take our **cenmle** object, which we named **van.mle.fit**, we use the following commands in R.

```
DiffCI(cenobj=van.mle.fit,conf.level=0.95,ngroups=5)
```

Executing these commands results in the following confidence intervals on the difference between the mean of the Hatchery group compared to the other groups being reported. These intervals are

the difference between the identified groups and the reference group(Hatchery).

	lower	upper
coefficients.Vanadium\$Location2	-0.05637355	0.1561403
coefficients.Vanadium\$Location3	-0.10389760	0.1038976
coefficients.Vanadium\$Location4	-0.05751870	0.1483099
coefficients.Vanadium\$Location5	-0.02956178	0.1771684

11.2.2 Changing The Reference Category

In multiple comparison tests such as the one above, R chooses one group to be the reference group. R makes this selection alphabetically, and the Hatchery group is alphabetically first. The important p-values testing for no differences in means between the reference and other groups are the four located below the (Intercept) row. These represent the differences between these categories and the reference group. Notice that none of the p-values associated with these differences are significant, which is what we would expect based on our original conclusion that there was no evidence of differences between the group centers. This was also seen in the boxplots.

In R, it is possible to select other reference groups, and other methods of comparison between the groups. The method of comparison above is called a *treatment* contrast, meaning that it compares means between the reference group compared to the other groups. I recommend this method due to its ease of interpretation. R sometimes defaults to other non-intuitive comparison methods, so it can be useful to explicitly set the comparison method to be used. The functions to change the reference group, and also to set the comparison method, are shown below.

To ensure that R is using treatment contrasts, we use the line of code below. Notice that we specify the contrasts on the grouping variable in our data set, in this case `Location`. Also notice that we specify 5 as the argument for `contr.treatment` indicating that we have 5 different groups that need to be compared.

```
contrasts(Vanadium$Location)=contr.treatment(5)
```

If we wish to change the reference group from Hatchery to something else, we use the R code shown below.

```
Vanadium$Location=ordered(Vanadium$Location,c("Intake", "Hatchery", "Hwy16",  
"Railway", "UpperTCreek"))
```

In the code above, we have changed the reference location to `Intake`, and the rest of the comparisons will be made in the order that the other groups are specified.

11.2.3 Dangers of Multiple Comparisons

When conducting statistical tests, we often specify α , the type 1 error rate. If we find a p-value lower than α (often 0.05), we conclude that there is evidence against the null hypothesis. Another way to interpret this α value, is that if were to conduct 20 different tests, we would expect at least one of them to show up as significant just by chance, even if there were no real differences in the population means. When performing multiple tests (or reporting lots and lots of confidence intervals), it is expected to find some that are significant just by chance.

We are not interested in results that are due purely to chance! We are interested in finding meaningful differences.

To protect against the inflated type 1 error rate that can occur when when we are making multiple comparisons, it common to introduce a correction to our selection of α , such that α is the type 1 error rate for the entire family of tests being conducted, rather than each individual test.

One of the most common corrections is known as a Bonferroni correction. This correction simply states that if we want a type 1 error rate of α across the entire body of our tests, then each individual test/interval should be conducted at a value of $1 - \alpha/p$. Here, p represents the number of individual tests being conducted.

Because this type of correction greatly increases the width of intervals, I strongly suggest that the experimenter outline which tests are important prior to conducting an analysis. Blanket testing of all possible differences will make it difficult to find any that are genuinely significant. This approach should be avoided.

11.2.4 Model Assessment

It was mentioned earlier that model assumptions for parametric methods include: data following a parametric distribution (normal, lognormal); equal variances between the different groups; and independent observations in the different groups.

The boxplots in section 11.0.2 do not indicate any obvious violation of the equal variances assumption. This is based on the observation that the boxes are all similar in size.

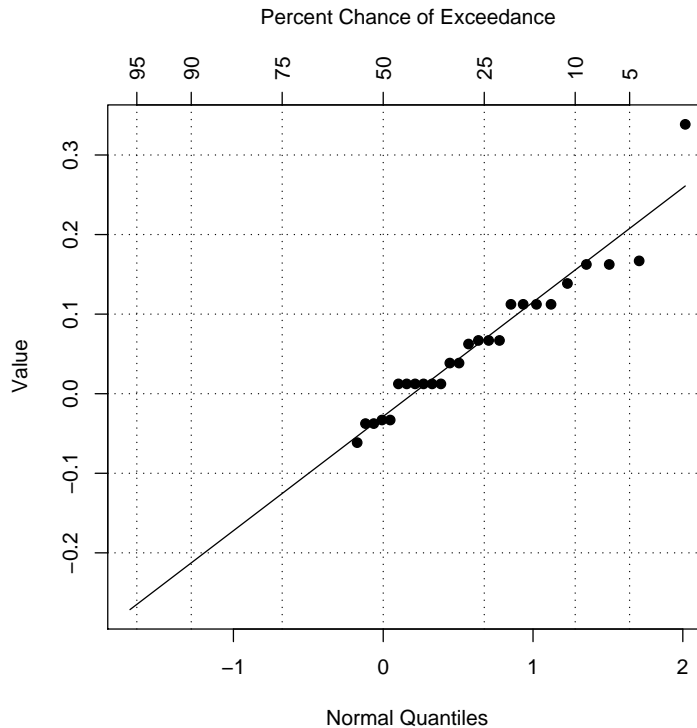
One of the best ways to assess the normality assumption is with a normal probability plot of the residuals. This type of plot has been demonstrated in earlier chapters, and will be demonstrated again here using the vanadium data.

To create a normal probability plot of the residuals from the vanadium data, the following commands are used in R.

```
van.mle.resid=residuals(van.mle.fit)
group.mle.ros=with(Vanadium,ros(obs=van.mle.resid,censored=VanCen,forwardT=NULL))
plot(group.mle.ros)
```

Executing this code results in the normal probability plot shown in Figure 11.2 being created.

Fig. 11.2: Normal Probability Plot For Vanadium Residuals



This plot shows no severe violations of the normality assumption. The data appear to follow the normal line relatively closely.

11.2.5 Interpretation When Data Are Log Transformed

Although we did not log transform the concentration values in the vanadium data, it will sometimes be necessary to do so. As occurred when comparing two groups, this results in a hypothesis of whether the ratio of the population medians among the different groups is one.

Confidence intervals of the differences between groups will be indicating whether the ratio of the group medians as opposed to the reference groups is one.

For more information on how to interpret confidence intervals and hypothesis tests when a log transformation has been performed please refer to Chapter 9

11.3 Non-Parametric Methods

I am also discussing a non-parametric method for comparisons among group centers. The method being discussed here is the Wilcoxon Score Test, a multivariate extension of the Peto and Prentice variation of the Wilcoxon test from Chapter 9 for differences between two groups.

Assumptions for this non-parametric test are much more relaxed than those for parametric methods. We don't need to specify a distributional form, we merely need to ensure that the variances in the different groups are equal, and that the shapes of the distributions are similar. For example, we would not want to compare strongly left skewed data to strongly right skewed data.

If the assumptions above hold true, the Wilcoxon score test will control for the presence of skewness or outliers in the data better than the parametric methods of Section 11.2. It is suitable for data where there are multiple left censoring locations.

11.3.1 Performing The Test

Using the same vanadium data that was introduced in Section 11.0.1 we can perform a Wilcoxon score test. Recall from the boxplots in Figure 11.1 that the shapes of the distributions for different groups seem similar, and the variances are also reasonably similar. This means that the assumptions for performing the Wilcoxon test have been satisfied.

To conduct the Wilcoxon score test in R, the following commands are implemented in the console window:

```
van.nonparfit=cendiff(obs=Vanadium$Vanadium,censored=Vanadium$VanCen,
groups=Vanadium$Location).
```

Executing these commands will results in the following output in R.

	N	Observed	Expected	(O-E) ² /E	(O-E) ² /V
Vanadium\$Location=Hatchery	9	3.75	4.82	0.237	0.480
Vanadium\$Location=Hwy16	8	4.16	3.52	0.115	0.213
Vanadium\$Location=Intake	9	3.75	4.82	0.237	0.480
Vanadium\$Location=Railway	9	4.95	4.09	0.182	0.344
Vanadium\$Location=UpperTCreek	9	4.41	3.77	0.107	0.200

Chisq= 1.4 on 4 degrees of freedom, p= 0.851

This test has a χ^2 form as can be seen from the output. The overall p-value for the test is 0.851, confirming the results that we found using the parametric test. There there is no evidence of differences between group centres. This output does not include any multiple testing information, so if the p-value were significant, subsequent tests have to be coded manually.

11.3.2 Followup Testing

As was discussed in chapter 11.2.3, multiple testing requires some correction on the α level of significance to give the correct overall level of significance for the tests.

To compare whether two groups are different in the multiple group setting, we simply create a new data set containing only those two groups, and then perform the same test as was used in Section 11.3.1. For demonstration purposes, I will show a test comparing the Railway and Intake centres.

The R code below first shows how to make a new data set that contains only information from the Railway and Intake groups, and then shows the commands used to compare these groups.

```
Vanadium2G=Vanadium[Vanadium$Location=="Railway" | Vanadium$Location=="Intake",]  
van.nonparfit2G=with(Vanadium2G,cendiff(obs=Vanadium,censored=VanCen,groups=Location))
```

Executing this code results in the following χ^2 test on whether the medians of the two groups are different.

	N	Observed	Expected	(O-E) ² /E	(O-E) ² /V
Location=Intake	9	3.89	4.94	0.225	0.78
Location=Railway	9	5.17	4.11	0.271	0.78

Chisq= 0.8 on 1 degrees of freedom, p= 0.377

The overall test in Section 11.3.1 indicated that there was no evidence of differences between the medians of any of the groups. This subsequent test on two groups supports that conclusion. The p-value of 0.377 indicates that there is no significant difference between the means of the Railway and Intake locations.

Unfortunately, confidence intervals are not commonly calculated or reported with these non-parametric tests.

Chapter 12

Trends: Correlation and Regression

Up until this point we have been discussing how to analyze analyte¹ concentration as a function of categorical or grouping variables such as locations. Another useful way to analyze trends in data is when analyte concentration is measured as a function of a *continuous* variable. For example, we might want to consider whether concentration is increasing or decreasing through time. Alternately, we might want to know at what rate analyte concentrations decrease as we sample further and further away from a pollution point source.

Describing these types of relationships statistically is done through the use of correlation coefficients and regression equations, both of which will be discussed below.

12.0.1 Conceptual Framework

Correlation

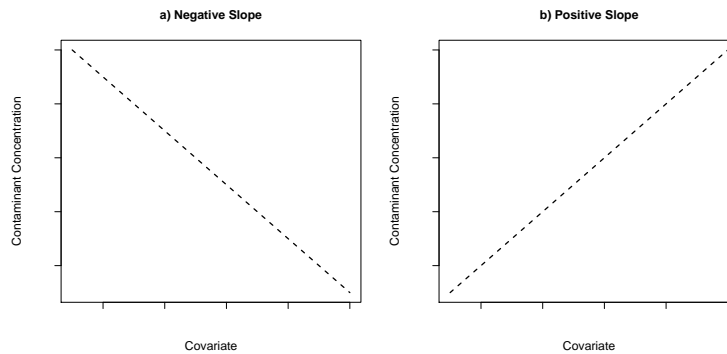
In the parametric, non-censored context, the correlation is a measure of the linear association between two variables. The sample correlation coefficient is usually denoted as r , and its range falls between -1 and 1 . Correlation values of -1 or 1 imply a perfect linear relationship between variables. This means that all sample points would fall exactly on a straight line. Values close to -1 or 1 imply a very strong linear relationship between variables, while values approaching closer to zero imply much weaker linear relationships between two variables.

Correlation values that are negative imply that the slope of the line associating the two variables is also negative. This means that as the covariate values (for example, time) increase, analyte concentration values would be decreasing. A line with a negative slope is demonstrated in Figure 12.1 a).

The converse situations is occurring when correlation values are positive. In this case, the slope of the line is also positive, implying that analyte concentrations are increasing as covariate values are increasing. A line with a positive slope, and is demonstrated in Figure 12.1 b).

¹An analyte is defined to be a chemical or other substance in water that is the subject of analysis

Fig. 12.1: Lines with negative and positive slopes



Regression

Where correlation is used to describe the strength of a linear relationship, regression equations are used to provide more specific information about the relationship between two variables. Regression equations are a very useful way to summarize trends, and are also sometimes used to predict new points for values of x that weren't specifically sampled. The mean estimated regression line is often generalized to the form shown below.

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x \quad (12.1)$$

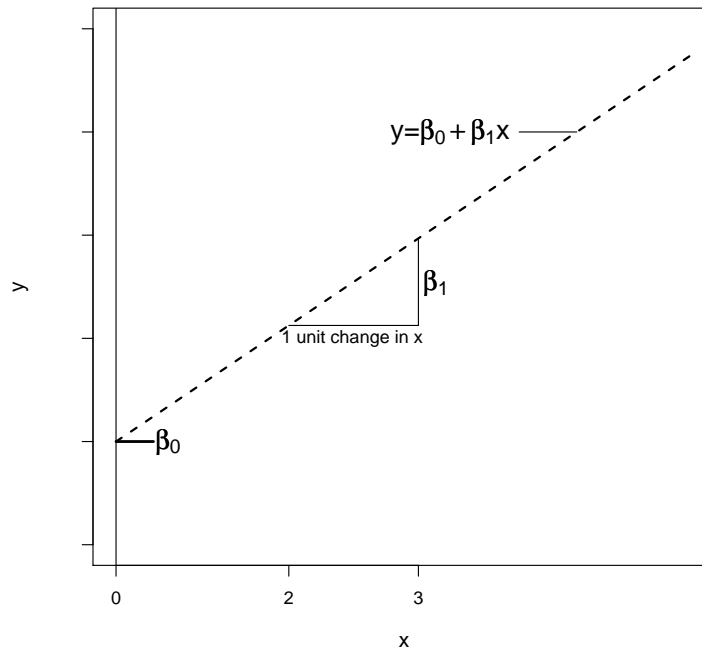
In this equation, \hat{y} represents our quantity of interest. For water quality data this will often be analyte concentration. On the right hand side of the equation, $\hat{\beta}_0$ is the y-intercept, which is the analyte concentration which is estimated to occur when the continuous covariate (x) is zero. The next greek letter, $\hat{\beta}$ represents the estimated slope of the regression line. This slope represents the estimated average change in y for a one unit change in x , the continuous covariate. The sign of the slope on the regression line (positive or negative) will always match the sign of the correlation coefficient.

A regression equation is represented graphically in Figure 12.2.

Hypothesis Testing

Statistical testing regarding whether there is a trend can equivalently be evaluated by testing whether the slope of the regression line is zero, or whether the population correlation coefficient is zero. If there is a significant difference from zero for these values it implies that there is a detectable trend. Formally written hypothesis tests for trend can be expressed as is shown in the equations below. The first set of hypotheses are in terms of the correlation coefficient, and the second set is written in terms of the slope.

Fig. 12.2: Representation of a regression line



$$H_0 : \rho = 0$$

$$H_A : \rho \neq 0$$

$$H_0 : \beta_1 = 0$$

$$H_A : \beta_1 \neq 0$$

12.0.2 Example Data

Correlation and regression type methods for censored data will be demonstrated using data collected on dissolved iron concentrations in the Brazos River in Texas. The data consists of summer iron concentrations taken over a period of several years. In this data the covariate is time, and we are interested in seeing if there is a linear trend in dissolved iron concentrations through time.

The Brazos River iron data have two different CLs, one at $3 \mu\text{g/L}$, and one at $10 \mu\text{g/L}$. The data set can be seen below in Table 12.1. This data is also available as part of the NADA package in R. After loading the NADA library, the data set can be accessed through the R command `data(DFe)`.

Table 12.1: Iron (Fe) concentrations ($\mu\text{g}/\text{L}$) for different years in the Brazos River

Year	YearCen	Summer	SummerCen
1977	FALSE	20	FALSE
1978	FALSE	10	TRUE
1979	FALSE	10	TRUE
1980	FALSE	10	TRUE
1981	FALSE	10	TRUE
1982	FALSE	7	FALSE
1983	FALSE	3	FALSE
1984	FALSE	3	TRUE
1985	FALSE	3	TRUE

12.0.3 Preliminary Data Inference: Graphing

Prior to conducting data analysis, it is generally advisable to do some preliminary inspection of the data in the form of graphing. This can identify potential problems in the data such as the presence of outliers, or the need to transform the data. In the context of regression and correlation this is also true.

Simple model assumptions can also be assessed, particularly in determining whether any trend in the data is in fact linear. If there is a curved relationship between analyte concentration and a covariate, it might be advisable to consider log transforming analyte concentration. More complicated patterns might suggest that the data is non-linear and should be analyzed using a different method.

The traditional method to graph data when both the \mathbf{x} and \mathbf{y} axis variables are continuous is using a scatterplot. In such a plot, each observation has a unique (\mathbf{x}, \mathbf{y}) coordinate which is then plotted as a point. In censor data, the \mathbf{y} variable, analyte concentration, is sometimes censored. To plot such a censored value, the range of possible \mathbf{y} -values is represented as a line rather than a point.

To construct a censored boxplot in R using the Brazos river iron concentration data the following commands may be used.

```
cenxyplot(x=DFe$Year,xcen=0,y=DFe$Summer,ycen=DFe$SummerCen,log="")
```

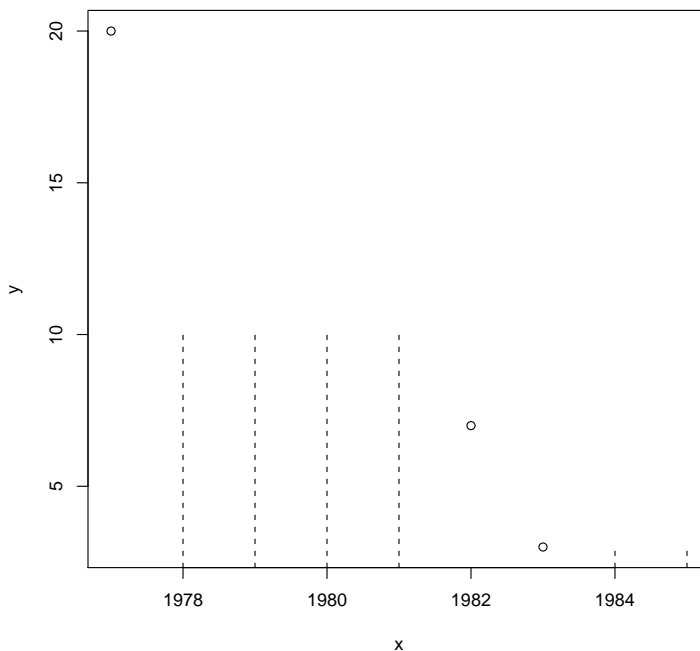
or

```
with(DFe, cenxyplot(x=Year,xcen=0,y=Summer,ycen=SummerCen,log=""))
```

Notice the argument `log=""`. In this case, the value `""` implies that we want to see the data on the original scale. If we wished to see the data on the log scale we would enter the argument `log="y"`. Also notice the argument `xcen`. This implies that the \mathbf{x} variable is censored too, which is not the case in water quality data, so I have set it to a value of 0 (no censoring) for all values of \mathbf{x} to prevent confusion.

The R code shown above results in the censored scatterplot seen in Figure 12.3. Although censoring does make it more difficult to see trends, it seems possible from this plot that there is a negative relationship between time and iron concentrations. As time increases, iron concentrations appear to be decreasing. Based on the plot, such is arguably linear (different people will assess this differently), so there is no apparent need to transform the data (if you disagree with my linearity assessment, don't worry, we transform the data later).

Fig. 12.3: Censored scatterplot of the Brazos River Iron concentration data



12.1 Maximum Likelihood Estimation

Maximum likelihood estimation in the presence of censored data is very similar to the estimation that occurs when conducting a standard linear regression. The difference is that the likelihood that is computed when censored values are present explicitly accounts for the values below the CL. Assumptions for correlation and regression type maximum likelihood estimators include: the presence of a linear trend in the data; observations being approximately normally distributed about the estimated trend line; variances being approximately equal in magnitude at all points along the trend line; and independent observations.

Linearity can be assessed through the censored scatterplot illustrated in Figure 12.3 above, and also through an analysis of the residuals from the model. A residual plot should show no obvious patterns if the data are linear.

The normality assumption can be assessed both by constructing a normal probability plot of the

residuals, and also based on a residual plot. In a residual plot, points should be approximately evenly spaced above and below 0, with the majority of points between -2 and 2 standard deviations from the center.

The equal variances assumption is more difficult to assess in the presence of censor data. The distance of the observed points from the regression line should be similar at all points along the line. A residual plot from the regression can be used to assess the variability, but should be interpreted with care because of the presence of censoring.

To estimate maximum likelihood correlation and regression coefficients when censored observations are present, the R code shown below can be used. The method is demonstrated on the Brazos River iron concentration measurements.

```
mle.reg=cenreg(Cen(obs=DFe$Summer,censored=DFe$SummerCen)~DFe$Year,dist="gaussian")
```

or

```
mle.reg=with(DFe,cenreg(Cen(obs=Summer,censored=SummerCen)~Year,dist="gaussian")
```

As with other methods we have seen, the `dist` argument can be set to either `gaussian` or `lognormal` depending on the shape of the data.

Using this code creates the following censored regression information.

```
mle.reg
      Value Std. Error      z      p
(Intercept) 3426.07    859.278  3.99 6.69e-05
DFe$Year     -1.73     0.434 -3.98 6.90e-05
Log(scale)    1.13     0.315  3.60 3.15e-04

Scale= 3.11

Gaussian distribution
Loglik(model)= -13.2  Loglik(intercept only)= -16.9
Loglik-r: 0.747004

Chisq= 7.35 on 1 degrees of freedom, p= 0.0067
Number of Newton-Raphson Iterations: 5
n = 9
```

Here the estimate of the slope for the year effect is -1.73 , indicating that for every year increase, iron concentrations are estimated to decrease by $1.73 \mu\text{g/L}$. This slope indicates that the relationship between time and iron concentration is negatively correlated. Notice though that the correlation coefficient, `Loglik-r` has an estimated value of 0.747004 , which is not negative. Because of how the correlation is calculated in the presence of censor data, the correct sign for the correlation coefficient

needs to be taken from the slope of the regression line. This means that the estimated value of the correlation between the two variables is actually -0.747004 .

An overall test of whether a line helps to explain the variability in the data is presented at the end of the output, and has a p-value of 0.0067, indicating that the model is highly significant at explaining the variability of the data. This means that using a line to describe the data explains more variability than not using a line. The actual test of significance for our linear coefficient, year, corresponds to a p-value of 0.000069. The data provides evidence that time is a useful variable in describing Brazos River iron concentrations.

A confidence interval on the slope of the regression can also be calculated using the `DiffCI(.)` function discussed in Chapters 9 and 11, and found in Appendix I. To obtain the confidence interval of the slope, the `DiffCI` function must first be loaded into R, and then the following commands can be executed.

```
slope.CI=DiffCI(cenobj=mle.reg,ngroups=2)
```

Notice the argument `ngroups=2`. A value of 2 is correct because we are estimating two parameters in our model. The first is β_0 , the y-intercept, and the second is β_1 , the slope. Executing the code gives the output shown below.

```
                lower      upper
coefficients.DFe$Year -2.576001 -0.8760004
```

This confidence interval does not include zero, once again confirming that there is evidence of a negative relationship between the variables.

Finally, the last step in evaluating this data is performing model assumption checks. To create a normal probability plot of the residuals to evaluate the normality assumption, the following commands should be executed in R.

```
mle.reg.resid=residuals(mle.reg)
qqnorm(mle.reg.resid)
```

Executing the above commands results in the normal probability plot shown in Figure 12.4 being created.

The points in this plot approximately follow a straight line, indicating that the normality assumption is not violated.

To further evaluate the normality assumption, and additionally the equal variances assumption, a residual plot can be created using the `plot` command, `plot(mle.reg.resid)`. Executing this command in the R console gives the residual plot shown in Figure 12.5.

Although the presence of censored values makes this somewhat difficult to interpret, it appears that the size of the residuals is decreasing through time, a violation of the equal variances assumption.

Fig. 12.4: Normal probability plot of residuals from censored regression

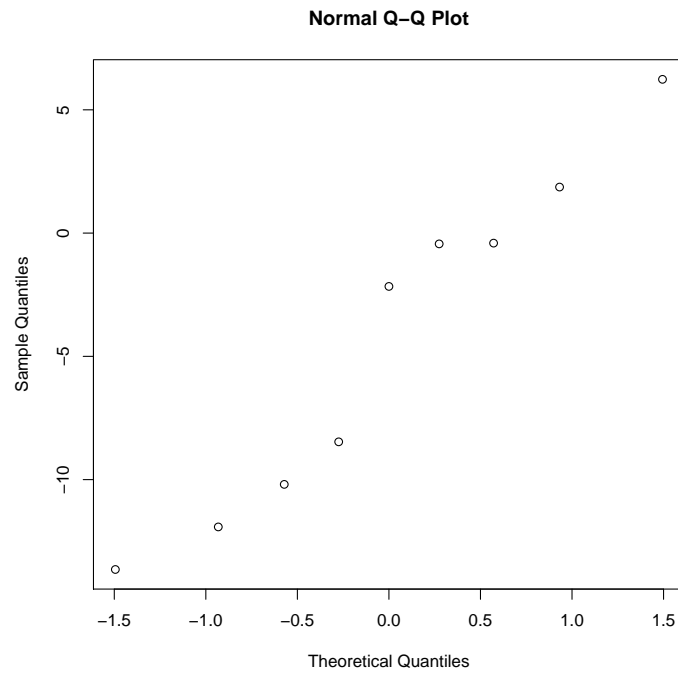
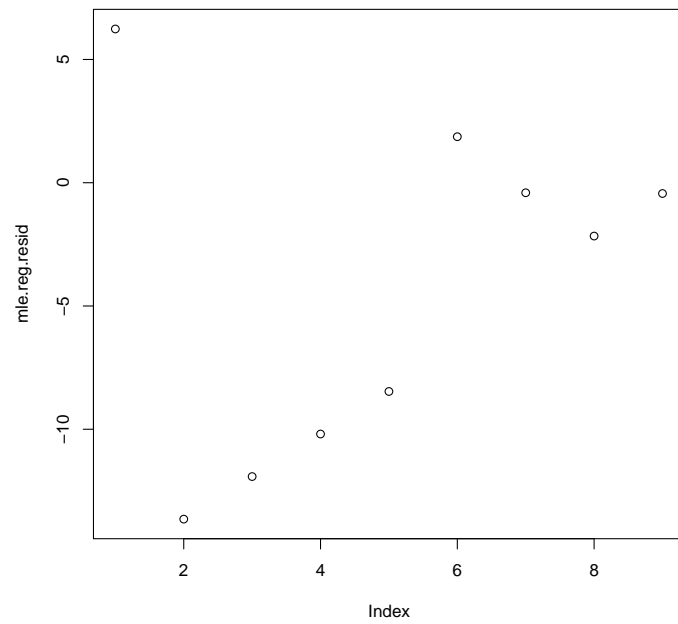
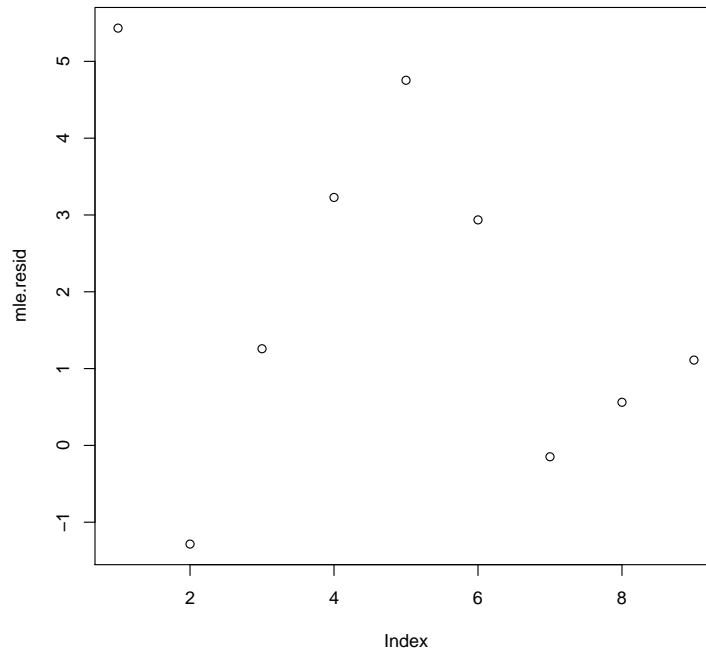


Fig. 12.5: Residual plot from censored regression



To correct for the problem, the analysis is repeated on the lognormal scale. This analysis results in the residual plot seen in Figure 12.6. The variability of the residuals about the line is stabilized by the log transformation, indicating that it is more appropriate to analyze the data on this scale.

Fig. 12.6: Residual plot from censored lognormal regression



12.2 Non-Parametric Approaches

Non-parametric measures of association tend to evaluate the monotonic association between two variables. This means that such methods are evaluating whether values of the response tend to increase as values of the explanatory variable increase (or vice versa). These non-parametric measures do not quantify how big the increase or decrease is, merely whether there is an increase or decrease. This means that non-parametric methods should be useful at evaluating whether there is an increasing or decreasing trend in the data, regardless of whether or not it is linear. A further discussion of the usefulness of non-parametric methods compared to maximum likelihood methods will be included in Section 12.3. Before using the non-parametric methods of association discussed below, the user should read section 12.3 to understand the weaknesses and risks of the method.

One of the most cited non-parametric measures of association between variables in water quality papers was proposed by Kendall (1955). This measure of association is known as Kendall's tau. Similar to a correlation coefficient, Kendall's tau falls between -1 and 1 , where values close to 1 indicate a strong positive association, and values close to -1 indicate a strong negative association. Values of tau near 0 indicate little or no association.

Related to Kendall's tau are non-parametric measures for slope. One of the most popular of these is known as the Theil-Sen slope estimate, Theil (1950); Sen (1968). Essentially this slope is estimated as the median of all possible slopes between pairs of data, given that some of those slopes will occur based on interval censored values (meaning the slope can fall within a range). The p-value for the Theil-Sen slope is the same as the p-value testing for evidence of monotonic association based on Kendall's tau.

The final component of our line is the y-intercept. A method of calculating the y-intercept in the case of non-parametric regression was proposed by Dietz (1987, 1989), and is the estimation method used in the R software. This estimated intercept is intended to minimize the error in the residuals about the line.

To conduct a non-parametric correlation and regression analysis of the Brazos River iron data, the following commands can be executed in R.

```
Fe.Kendall=cenken(y=DFe$Summer, ycen=DFe$SummerCen, x=DFe$Year, xcen=DFe$YearCen)
or
Fe.Kendall=with(DFe, cenken(y=Summer, ycen=SummerCen, x=Year, xcen=YearCen))
```

Executing the above commands will give the following results in R.

```
> Fe.Kendall
slope
[1] -2.572113

intercept
[1] 5103.5
```

```
tau
[1] -0.3611111
```

```
p
[1] 0.1315868
```

In these results Kendall's tau is estimated to be -0.3611 , with an associated p-value of 0.1315868 , which would seem to indicate little evidence against the null hypothesis of no association between variables. However, this method does not detect a significant trend between iron concentrations and time.

These results are very different than what was found when analyzing the iron data using censored maximum likelihood methods. The reasons for this difference will be explained in Section 12.3. For the record, I believe that the results given by the censored maximum likelihood method are superior in this case, and should be used in preference to the non-parametric alternative shown above. Reasons for this will be discussed in Chapter 12.3

Realistically, using this data, conclusions from either the parametric or non-parametric methods should be interpreted carefully. The Brazos River data were used in both the parametric and non-parametric demonstrations. This data set is quite small, with only nine observations. Additionally, it has a very high rate of censoring - greater than 50%. Consequently neither method is truly appropriate for this data set.

12.3 Comparing Maximum Likelihood and Non-Parametric Results: Some Cautions!!

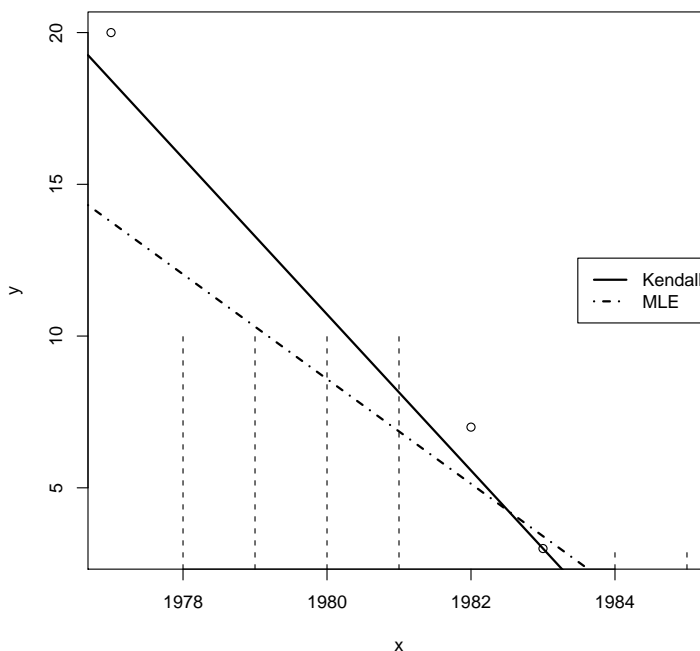
In previous chapters, I have often described non-parametric methods as *robust* compared to parametric ones. This meant that when extreme outliers were present, or the distribution of points was highly unusual, non-parametric methods were recommended. In less extreme situations, non-parametric methods performed similarly or slightly worse than maximum likelihood methods. In estimating regression type statistics, I no longer believe available non-parametric methods to be the more desirable option in *any* data situation. This will be demonstrated and explained below.

Kendall's tau was originally intended for data where there are no tied values in either the x or y variables. When ties occur, they are essentially ignored in calculating the tau coefficient. Unfortunately, values below the same CL are considered ties, and do not contribute information as they should to the calculation of tau. Notice that for the iron data or maximum likelihood measure of correlation was -0.747 , and tau was estimated at only -0.3611 . This could in large part be due to the down-weighting of the censor values in the tau calculation, because the scatterplot makes a negative correlation between the variables intuitively plausible.

There is a corrected version of Kendall's tau for use when tied observations are present. This corrected estimator is called Kendall's tau-b. Unfortunately, other regression estimates, particularly the Theil-Sen estimator for slope, are based on Kendall's tau, not tau b. As a result, I feel these

estimators are not good because they do not take enough information from the censored values. I create a visual argument for this in Figure 12.7

Fig. 12.7: Comparison of trend lines estimated by maximum likelihood and non-parametric methods



In Figure 12.7, notice how much steeper the slope of the line based on Kendall's tau is compared to that estimated by the censored maximum likelihood methods. The tau line nearly passes through every observation that is not censored, at the expense of information present in the censored values. This is a particularly obvious problem near the beginning of the data where there is a very high concentration observation for 1977. The entire non-parametric line is *leveraged* higher to try and reach this point. It is not advisable to have the magnitude of an estimate rely so heavily on a single observation.

To contrast this, the line estimated using maximum likelihood methods is doing a much better job at incorporating information from the censored values. The line is not as leveraged to the high concentration value in 1977. In this type of analysis, the maximum likelihood method is performing better than the corresponding non-parametric method. For linear data, parametric methods will either create estimates that are similar to non-parametric methods, or create estimates that are more sensible than non-parametric methods. As a result, when calculating regression type estimators I STRONGLY recommend the use of the maximum likelihood estimates that were presented in 12.1.

As a small aside, the R code that was used to create the graph in Figure 12.7 is shown below for the reader's interest.

```
cenxplot(x=DFe$Year,xcen=0,y=DFe$Summer,ycen=DFe$SummerCen,log="")
abline(mle.reg,lty=4,lwd=2)
```



```
lines(Fe.Kendall,lwd=2)
legend(x="right",legend=c("Kendall","MLE"),lty=c(1,4),lwd=2)
```

Chapter 13

Further Topics in Regression: Seasonal Trend Analysis

In Chapter 12 we considered a simple linear regression example where there was a decreasing trend in concentration over time. In water quality, trends can sometimes be more difficult to detect due to the influence of confounding variables. For example, flow volume in a river can be higher or lower at different times in the year, with higher flow volumes being associated with lower concentrations due to dilution. Such seasonal variability can mask an overall increasing or decreasing trend in concentrations. Statistical methods used to analyze this type of seasonal data must first account for confounding information, such as seasonal variability. With this variability properly accounted for, a valid estimate of increasing or decreasing analyte¹ concentrations is then possible.

In water quality analysis, a very popular test is the Seasonal Kendall test for trend as proposed by Hirsch et al.(1982), and Hirsch and Slack (1984). Seasonal effects can also be detected using maximum likelihood multiple regression type methods. For information purposes, both methods will be discussed in Sections 13.1 and 13.2 below. For data with observations below the CL, the maximum likelihood approach is recommended over the Seasonal Kendall test.

13.1 Seasonal Kendall Test

The Seasonal Kendall test essentially proposes dividing data into different *seasons*, or other natural groupings. If a water quality sample were taken from a location once a month over a period of years, data could be divided into monthly groups. For each of the twelve months of the year Kendall's tau can be calculated. Assuming the months are independent groupings, the test statistics from each of the individual monthly calculations can be summed and used to provide an overall significance test for increasing or decreasing association.

The logic of this is that months in different years should be similar to one another in terms of confounding variables such as flow rates. If there is an overall increasing (or decreasing trend) over the years, then this is a trend we would expect to see when we analyze any given month.

¹An analyte is defined to be a chemical or other substance in water that is the subject of analysis.

By analyzing months separately, and then combining test statistics at the end, the test intends to control for seasonal variability while still detecting trends of increase or decrease through time.

Despite the intuitive appeal of this type of approach, there are two major problems associated with its implementation. The first problem relates to those discussed in Section 12.3, where the tau estimate does not account for censored values well. The second relates to the assumption in the test that adjacent months are independent from one another.

In their article discussing how to select a method to analyze water quality data, Hirsch et al. (1991) acknowledge that the Seasonal Kendall test performs poorly when there are multiple censored observations present, and that the Theil-Sen estimate of slope is also inaccurate when there is a large percentage of observations below the CL. This is consistent with the concerns about the method that were discussed in Section 12.3 which showed how poorly the Theil-Sen slope fit the data in Figure 12.7.

Where the comment above relates only to the use of the Seasonal Kendall test in the presence of censored data, this second criticism is more general to the method as a whole. The Seasonal Kendall test makes the assumption that groups, be they seasons or months, can be treated independently. This implies that we believe that there is no covariance relationships between observations in May and June of the same year if we are creating groups by month. This is not always a realistic assumption! This issue was addressed in an article by El-Shaarawi and Niculescu (1992). They corrected the Seasonal Kendall test to account for the presence of covariance between groups, and in an example discussed how this improves the ability of the test to detect trends. Unfortunately the covariance terms are large and awkward, and the correction is not commonly used or discussed.

While the Seasonal Kendall test is not appropriate in data containing censoring, it is still possible to account for seasonality when measuring trends through time. Maximum likelihood regression type methods can easily be adapted to address this problem, as will be demonstrated in Section 13.2 below.

13.2 Multiple Regression With Censoring

The type of analysis that is being conducted with the Seasonal Kendall test can actually be implemented through censored maximum likelihood regression type methods. This type of analysis handles censoring correctly, and due to the structure of the estimation, it naturally allows for covariance terms between the different seasons, or groups. This type of analysis also allows for the inclusion of *multiple* continuous and discrete variables as regressors at the model, meaning that their effects on concentration can be modeled simultaneously.

Additionally, this multiple regression type method allows for a broader set of parameters to be estimated, giving more information about the types of trends that are present. For example, using this type of model, we can control for seasonal effects and estimate their magnitude. Models can also be created that allow for the direction of the trends to be different for different seasons. We will be focusing on the prior type of seasonal model in this chapter.

For an overview of the types of methods that can be created using multiple linear regression, it

can be useful to reference a textbook that includes ANCOVA, or multiple linear regression with indicator variables in its outline (same thing, different names). Two good recent references include texts by Montgomery et al. (2001), and Kutner et al. (2005)

Because of its correct handling of censoring, ability to allow for covariance between related seasons (or groups), and the broader range of hypotheses about trends that can be tested, the multiple regression type maximum likelihood approach is recommended for testing seasonal effects where censored observations are present.

13.2.1 Example Data

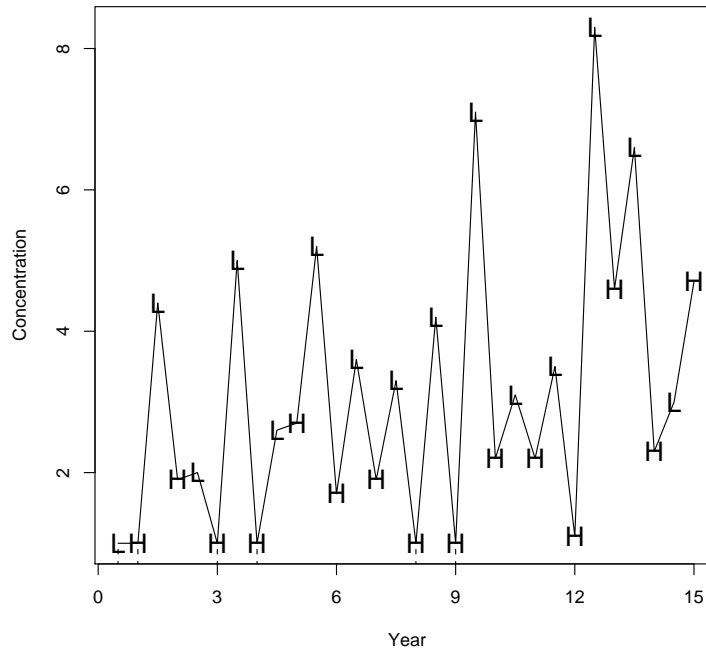
This data was simulated to resemble a water quality data set with one CL at 1 $\mu\text{g}/\text{L}$, and distinct high flow and low flow seasons. Data were measured twice a year for fifteen years. Each year, water quality was measured once during high flow, and once during low flow. When water is at high flow, it is assumed that some analytes will be more dilute, and measured concentration values will be lower. The converse is true during for the same analytes during the low flow measurements. The data overall are assumed to show a trend of increasing concentrations. A small sample of the data generated are shown below in Table 13.1, and the complete data set can be found in Appendix O.

Table 13.1: Seasonal water quality data

season	obsNumber	values	year	cen
lowSeason	1	1.0	1	TRUE
highSeason	2	1.0	1	TRUE
lowSeason	3	4.4	2	FALSE
highSeason	4	1.9	2	FALSE
⋮				
highSeason	30	2.7	15	FALSE

This data has been graphed below, and some key features of the data can be seen in the scatter and line plot that is shown in Figure 13.1. Rather than dots, data points are illustrated using either an H to represent the high water flow season, or L to represent the low water flow season. As designed, during low water flow, analyte concentrations tend to be higher than for the high water flow measurements in the same year. This cyclic rotation between low and high flow, and therefore high and low analyte concentrations has been emphasized by the line that has been drawn between the points. The line bounces up and down as relative analyte concentrations change between our seasons of interest.

Fig. 13.1: Censored scatter and line plot of data showing a high flow/low flow seasonal trend



13.2.2 Model Fitting

To obtain censored regression estimates to the *SeasonalWater* data described in Section 13.2.1, the following commands should be executed in R.

```
contrasts(SeasonalWater$season)=contr.treatment(2)
```

```
Mult.reg.season=cenreg(Cen(obs=SeasonalWater$values,censored=SeasonalWater$cen)  
~SeasonalWater$year+SeasonalWater$season,dist="gaussian")
```

or

```
with(SeasonalWater, cenreg(Cen(obs=values, censored=cen)~year+season,dist="gaussian"))
```

Note that as was discussed in Chapter 11.2.2 on changing reference categories, we have specified treatment contrasts in R for ease of interpretation.

After correctly specifying the seasonal categories, we then calculate censored maximum likelihood estimates for the regression using the `cenreg(.)` command. Notice that the syntax of this command is virtually identical to how it was used in Chapter 12 when creating simple linear regression estimates. The major difference in this command is that we have included more than one covariate using the pseudo equation that occurs after the `~` sign, ie. `~year+season`.

Although the model we are considering has only two covariates, year and season, multiple variables can be included in a model by simply adding them in using the addition sign!

Executing the commands given above on the example data set results in the following output from R.

	Value	Std. Error	z	p
(Intercept)	0.232	0.6107	0.38	7.04e-01
Data\$year	0.204	0.0604	3.37	7.39e-04
Data\$seasonlowSeason	2.303	0.5217	4.41	1.02e-05
Log(scale)	0.353	0.1299	2.72	6.62e-03

Scale= 1.42

Gaussian distribution

Loglik(model)= -53.2 Loglik(intercept only)= -63.9

Loglik-r: 0.7127424

Chisq= 21.28 on 2 degrees of freedom, p= 2.4e-05

Number of Newton-Raphson Iterations: 5

n = 30

How to interpret this output is discussed in the section below, 13.2.3

13.2.3 Model Interpretation

What we have created in the model above is called a parallel lines regression model. We have one line estimating parameters for observations taken at low flow, and one line estimating parameters associated with observations taken at high flow. Both lines are assumed to have a similar trend in terms of slope during the 15 year observation period. Despite my claim of there being two lines, the data is reported as a single linear equation. This equation is reproduced below where \hat{y} represents the estimated contaminant concentration value, and t represents the year at observation, and *lowSeason* is an indicator variable that is 1 when the data is observed during the low season, and zero otherwise (this coding happens automatically when the `contr.treatment()` command is used).

$$\hat{y} = 0.232 + 0.204 * t + 2.303 * lowSeason$$

The indicator *lowSeason* is key to breaking the above equation into two separate lines, one for each flow season. When we want to estimate the line for the high flow season, the *lowSeason* variable will always be zero, resulting in the estimated regression equation shown below.

$$\hat{y}_{high} = 0.232 + 0.203 * t + 2.303 * 0$$

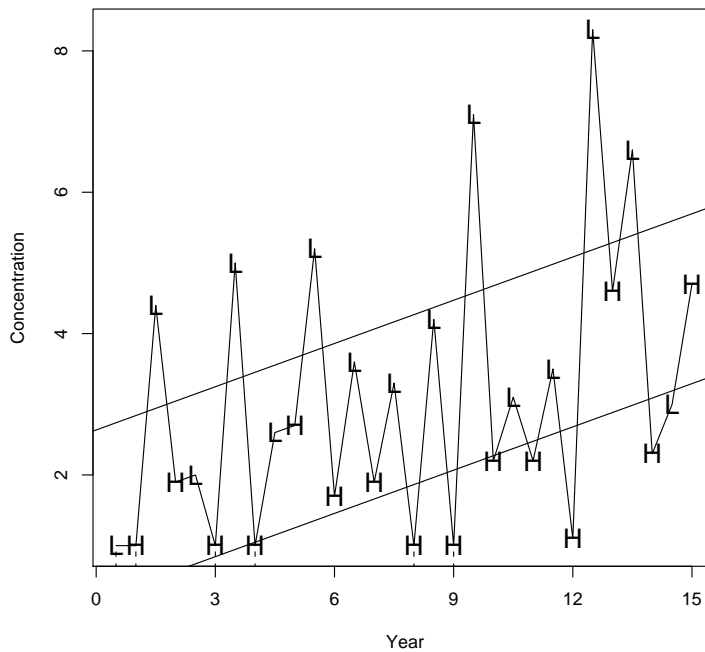
$$= 0.232 + 0.203 * t \tag{13.1}$$

Similarly, when *lowSeason* is one, indicating estimates for the low flow season, we find the following estimated equation for a line.

$$\begin{aligned} \hat{y}_{low} &= 0.232 + 0.203 * t + 2.303 * 1 \\ &= (0.232 + 2.303) + 0.203 * t \\ &= 2.535 + 0.203 * t \end{aligned} \tag{13.2}$$

We now have two separate linear equations, one for each of the flow situations. The estimated coefficient associated with the indicator variable *lowSeason* is essentially a measure of the vertical shift between the two parallel lines. This could also be interpreted as the estimated average difference in concentration due to high flow vs. low flow conditions. This shift between the regression lines is shown visually below in Figure 13.2 where the lines for both high and low flow conditions are plotted on a scatterplot of the observations.

Fig. 13.2: Plot of data showing a high flow/low flow points and censored regression lines for each flow season



13.2.4 Testing, Confidence Intervals, and Model Checking

While hypothesis testing and calculating confidence intervals for coefficients are an important part of an analysis, these exercises are not included here. This is because the R code and statistical techniques are essentially the same as those discussed in Chapter 12, and the reader is referred to the appropriate section of this chapter for guidance in further analysis.

Model checking is similarly very important, and should be considered in any true data analysis. For the sake of brevity the reader is once again referred to Chapter 12 for further information on these topics.

Chapter 14

Things to do when censored observations make up more than 50% of the data

The methods discussed in previous chapters are most effective when the percent of observations below the CL is less than 50%. Although parameter estimates are still calculable when the percent censoring is higher than this, their values become more suspect. Consider the example given in Chapter 12.0.2 on Brazos River iron concentrations. A full 2/3rds of the data are censored. Imagine removing the uncensored data point at a concentration of 20 $\mu\text{g/L}$, and suddenly the downward trend through time seems less plausible. The tests discussed in this chapter are more appropriate to this type of highly censored data, because they model the *probability* of exceeding the CL rather than actual analyte levels.

Although we are specifically discussing methods for use when the percent censoring is high, the methods discussed in this chapter will be valid when the percent censoring is anywhere between 15% – 85%. Because the methods discussed here in Chapter 14 ignore information about estimating analyte concentrations, I am specifically recommending these methods when the percent censoring is higher than 50%.

It may seem that because the information on values below the censor limit is missing that little can be done to excerpt information from that data. This is not true. There are many types of dilution experiments present in the sciences. In some of these, the goal is to dilute a solute until some characteristic is either present (TRUE) or absent (FALSE). Although the concentration is fundamentally unknown in our water quality data, the idea that information in the form of TRUE or FALSE responses can be a valid information source holds true.

Because of the presence of a wide range of experiments that result in these kinds of TRUE/FALSE responses, statistical methods for analyzing this type of data are well developed. Due to the common nature of these types of analyses, covering all possible methods would require a whole separate guidance document! This chapter aims to provide a few basic methods that will allow analyses similar to those performed in previous chapters to be performed when using TRUE/FALSE information about censoring rather than modeling concentration values directly. For a reader who is motivated to have a more in depth understanding of these topics, I refer you to the notes for a recent Statistics 402 class at Simon Fraser University. The website for this course (shown in the bibliography) includes notes, example data, and S-plus code for conducting the analyses. S-plus

and R are similar enough that most of the code can be used as is in R.

In the sections below, I will discuss how to analyze data when covariate information is present. When no covariate information is present, the data summary techniques outlined in Chapters 6 may be used to describe the data. Many of the ideas and techniques are similar to those discussed in Chapters 7-13, only instead of modeling solute concentration directly, we are modeling the probability of exceeding a CL.

14.1 Tests Including Covariate Information

When response variables are discrete (ie. TRUE/FALSE), and covariate information is present, data can be modeled using generalized linear models (GLMs). Specifically, I will be focusing on a type of generalized linear model called a logistic regression model. This type of model is very flexible and will allow us to incorporate discrete, continuous, or a combination of discrete and continuous covariates. The idea behind this type of model is that we are modeling the *probability* that an observation with given covariate values will be above or below the CL.

Because probabilities are values between 0 and 1, and our observations are either TRUE or FALSE (0 or 1), we create a transformation that maps from the 0 to 1 probability scale onto a range of $-\infty$ to $+\infty$. Once the data are on this broader scale, they can be modeled in the same regression type linear model context that has been discussed in Chapters 11, 12, and 13. The calculation used to achieve this transformation is called a link function.

The link function that we will be using in the analyses demonstrated in this chapter is called the *logit* link, which has the form shown below.

$$\log\left(\frac{\pi}{1-\pi}\right)$$

In the above, π represents the proportion of interest, and the quantity $\frac{\pi}{1-\pi}$ is an odds ratio. This link is then equated to a linear equation similar to the one shown below.

$$\log\left(\frac{\pi}{1-\pi}\right) = \beta_0 + \beta_1 * x_1 + \dots \beta_p * x_p$$

In the above equation, x_1 through x_p represent the covariates of interest, such as time or group. Their corresponding β parameters represent the effect of the covariates on the log odds. One advantage to the logit, or log odds link function is its ease of interpretation. To interpret the beta coefficient, one can simply say that “the estimated effect of increasing x_1 by one unit changes the odds of detection by $exp(\beta_1)$ times.” If the variable is categorical, a similar interpretation can be made. An interpretive sentence could be phrased “the estimated odds of detection in group j compared to the reference group change by $exp(\beta_j)$ times.”

14.1.1 Example Data

The data set that will be used to introduce the use of GLMs is called TCEReg, and is available in R after loading the NADA library. Once NADA has been loaded into R, the TCEReg data set can be loaded using the command `data(TCEReg)`. The TCEReg data set contains information on TCE concentrations in the groundwater in Long Island New York. The data has four detection limits at 1, 2, 4, and 5 $\mu\text{g/L}$.

In addition to information on TCE concentrations, there are several explanatory variables (covariates). Some of these covariates are continuous, and some of these covariates are discrete, making this an ideal data set to demonstrate the flexibility of including a variety of variables and variable types in a GLM model. A few lines of the data set are shown in Table 14.1.

Table 14.1: TCE Concentrations ($\mu\text{g/L}$) along with potential explanatory variables

TCECen	TCEConc	LandUse	PopDensity	PctIndLU	Depth	PopAbv1
TRUE	1.0	9	9	10	103	1
TRUE	1.0	8	3	4	142	1
TRUE	1.0	8	3	4	209	1
TRUE	1.0	5	1	3	140	1
TRUE	1.0	5	2	1	218	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮

Altogether there are 247 different observations in the TCEReg data set, and 194 if them are censored. This corresponds to a 78.5% rate of censoring in the data. This high level of censoring makes it more practical to analyze the data using the logit model shown above than other methods for censored data.

14.1.2 Performing and interpreting analysis

For the purpose of demonstrating the use and interpretation of GLM equations, I will be using 2 variables from the TCEReg data set for analysis: one continuous, and one categorical. The continuous variable will be well depth (Depth), and the categorical variable will be land use category (LandUse). There are three different land use categories, commercial, residential, and agricultural. These levels are associated with the levels 5, 8, and 9 found in the TCEReg data.

Before performing the analysis, it is useful to make sure that the data are specified in the correct format. Specifically, the LandUse variable should be specified as a factor variable with treatment contrasts. To check whether the LandUse variable is correctly specified as a factor, we can use the R command `is.factor(TCEReg$LandUse)`. Unfortunately, the output given by R to this query is `FALSE`, meaning that the LandUse variable has not been specified as a grouping variable.

To correctly specify the LandUse variable as a factor with levels that will be compared using treatment contrasts, we execute the following commands in the R console window.

```
TCEReg$LandUse=as.factor(TCEReg$LandUse)
contrasts(TCEReg$LandUse)=contr.treatment(3)
```

After executing this code, if we once again query R regarding whether LandUse is a factor variable, `is.factor(TCEReg$LandUse)`, we will receive the response `TRUE`, indicating that the variable is now being stored correctly.

To create the GLM in R, we use the command given below. Notice that we specify a `logit` link. There are other link functions that can be used, but as stated above I am focusing on the logit link due to its being commonly used and easy to interpret.

```
p.ANCOVA.logit=glm(TCECen~LandUse+Depth, family=binomial(link="logit"),data=TCEReg)
```

Executing the above code results in the following model being created in R.

```
summary(p.ANCOVA.logit)
```

Call:

```
glm(formula = TCECen ~ LandUse + Depth, family = binomial(link = "logit"),
     data = TCEReg)
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-2.6583	0.2409	0.5413	0.6430	1.0553

Coefficients:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	1.576765	0.830497	1.899	0.0576 .
LandUse2	-0.272990	0.797970	-0.342	0.7323
LandUse3	-1.351880	0.805794	-1.678	0.0934 .
Depth	0.003642	0.001887	1.930	0.0536 .

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

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 256.86 on 246 degrees of freedom
Residual deviance: 231.12 on 243 degrees of freedom
AIC: 239.12

Number of Fisher Scoring iterations: 5

Recall that in our model we used both a categorical and a continuous explanatory variable. If we had included only a categorical variable, the model and its interpretation would be analagous to

the ANOVA type models that were discussed in Chapter 11. If we had only used the continuous variable, then our model and its interpretation would be analagous to the simple linear regression models that were discussed in Chapter 12. Because our model is a mixture of both categorical and continuous components, our interpretation of the model is a combination of both these elements.

In our model, we discover that LandUse type 3 (corresponding to factor level 9) has an estimated coefficient of -1.352. This corresponds to an interpretation that odds of censoring being TRUE are estimated to be $\exp(-1.352) = 0.26$ times greater in land use type 3 than land use type 1. Although technically correct, the wording of the previous sentence is somewhat confusing. Finding a negative coefficient value implies that it is actually less likely that censoring is TRUE in land use type 3 than 1. Another way to interpret this in terms of the odds is to take the inverse of the effect, and write a sentence in terms of the odds that an observation is *not* censored (FALSE). In this case we would say that the odds of no censoring are $1/0.26 = 3.87$ times greater in land use category 3 than in land use category 1. Since the odds of non-censored data are higher in land use category 3, it seems likely that this land use category contains a higher number of large and uncensored observations.

We can similarly interpret the effect of a 1 unit increase of our continuous variable, well depth. In this case we would say that for every 1 unit increase in well depth, we have an estimated increase of $\exp(0.0036) = 1.003$ times the odds of observing censoring. This implies that as wells get deeper, the estimated odds of censored values (corresponding to small TCE concentrations) increases. Although this seems like a small effect size, when we consider that the depth of wells ranged from 19 units to 733 units (giving a difference of 714 units), the odds of censoring actually change quite a bit over the range of the data.

An important note should be made about the parameter interpretations described above. When interpreting the parameter/effect sizes of variables in the model, we are discussing the parameter estimates after accounting for other variables in the model. If we were to remove one of the parameters from the model, we would likely discover that the parameter estimates change, because we are no longer accounting for their influence on estimates.

In the model discussed above, neither Depth nor any of the LandUse categories are significant at the 5% level. Despite this, if we run a simple model just including LandUse as a variable, there is a statistically significant difference between land use categories 5 and 9. Similarly, if we run a model including just Depth as a variable, Depth also becomes significant. This is likely due to some multicollinearity in the variables, and a suggested solution is to just choose a simpler model including Depth *or* LandUse rather than both.

14.1.3 Goodness of fit testing

In previous chapters, we tended to evaluate the goodness of our models based on the normality and equal variance assumptions of the underlying distributions. We often used graphical tests to conduct these evaluations. Graphical tests are more complicated to interpret for categorical data, so it is common to evaluate how well a model is fitting in other ways. To evaluate how well a GLM is fitting, we perform goodness of fit tests.

The goodness of fit test which is most commonly reported, and which we will be demonstrating here is called the likelihood ratio goodness of fit test, or the deviance goodness of fit test.

To calculate a deviance statistic we compare the model we have created to what is called the *saturated* model. The idea of a saturated model is that it fits the observed data perfectly, but requires the estimation of one model parameter for each observation! We are comparing how well this complicated model fits relative to our simpler model. If the difference between these two models is small, it implies that our much simpler model fits almost as well as the saturated model. Therefore, our simpler model is useful at explaining the observed data. This means that we are hoping to find a large p-value associated with our test statistic, indicating that there is little difference between the saturated model and our model.

Calculating a the p-value for a deviance test in R takes a few lines of code. First we need to calculate the deviance statistic. Second we need to calculate the appropriate degrees of freedom for the statistic, and finally we need to compute the p-value for the test. The deviance statistic follows a chi-square distribution, and by comparing our statistic to this distribution we are able to calculate the correct p-value. To conduct a deviance test in R, the following commands can be used. Notice that the output of the test, a p-value is shown in the last line. This p-value will naturally appear as output in R after the three lines of code appearing previously are entered.

```
deviance=p.ANCOVA.logit$deviance
deviance.df=summary(p.ANCOVA.logit)$df.residual
1-pchisq(deviance,deviance.df)
[1] 0.6976828
```

From the last line, the p-value from the deviance test for our model is 0.698, which is larger than 0.05. This indicates that we have evidence supporting the idea that our model is relatively as good as the saturated model. The model that we created is useful for explaining the odds of censoring in the TCEReg data set.

Appendix A

Data Sets

A.1 Savona data

This data set was provided by the BC Ministry of Environment. It contains information on orthophosphate concentrations taken at the Thompson River from Savona. This data set contains four variables: **Date**, which is the date of the measurement taken; **Less than** indicates whether a measurement is below the detection limit; **concentration** is the level of orthophosphate observed at each date; and **Censored** also indicates whether an observation is below the detection limit or not. The data can be downloaded from the website

<http://www.sfu.ca/~ejuarezc/nondetects/SavonaData.xls> and it is shown in Table A.1.

A.2 Retena data

This data set was provided by the BC Ministry of Environment. The data set consist of several water quality indicators taken at various dates. The measurements in this data set are taken at Kitimat. This data set is used only in the Appendix to illustrate methods of formatting data. The data can be downloaded from <http://www.sfu.ca/~ejuarezc/nondetects/RetenaData.xls>.

A.3 Arsenic data

The **arsenic** data comes from (Helsel,2005b) and is shown in Table A.2.

Table A.1: Concentrations of orthophosphate in the Thompson River at Savona. A ‘1’ denotes a censored observation, and a ‘0’ an observed value.

Date	Less than	Concentration	Censored
27-Oct-99	<	0.001	1
23-Nov-99		0.002	0
21-Dec-99		0.002	0
18-Jan-00		0.002	0
01-Mar-00		0.002	0
28-Nov-00	<	0.001	1
27-Dec-00		0.002	0
22-Jan-01		0.003	0
20-Feb-01		0.002	0
14-Mar-01	<	0.001	1
14-Nov-01	<	0.001	1
10-Dec-01		0.002	0
16-Jan-02		0.008	0
13-Feb-02	<	0.001	1
12-Mar-02	<	0.001	1
18-Dec-02		0.002	0
21-Jan-03		0.002	0
04-Mar-03		0.006	0
29-Oct-03	<	0.001	1
22-Dec-03		0.002	0
09-Feb-04		0.007	0
23-Mar-04		0.009	0
23-Feb-05		0.006	0
15-Mar-05		0.006	0
12-Apr-05		0.005	0
25-Oct-05		0.003	0
22-Nov-05		0.004	0
20-Dec-05		0.006	0
17-Jan-06		0.007	0
15-Feb-06		0.007	0
08-Mar-06		0.006	0
06-Apr-06		0.005	0

Table A.2: Concentrations of arsenic measured in streamwaters at Oahu, Hawaii. A ‘0’ denotes an observed value, and a ‘1’ denotes a censored value; at the value presented.

Concentration	Censored
0.5	0
0.5	0
0.5	0
0.6	0
0.7	0
0.7	0
0.9	1
0.9	0
1	1
1	1
1	1
1	1
1.5	0
1.7	0
2	1
2	1
2	1
2	1
2	1
2	1
2	1
2	1
2	1
2.8	0
3.2	0
⋮	⋮

Appendix B

Transferring Data from Excel to JMP

In Excel it is common to represent the data below the censoring limit using the symbol “<” . For instance, the expression < 0.002 means that the observation is known to be below 0.002. In order to transfer the data from Excel to JMP to compute summary statistics, it is necessary reformat the data. The format needed includes an indicator variable denoting which observations are censored and which are not.

There are basically two presentations of the data in Excel; the first one is where the symbol “<” is in a separate column preceding the method detection limit indicating whether the observation is censored. The second case where the symbol “<” is in the same column as the observation value; in this case, the column is recognized by Excel as a general, or text format.

In the first case where the symbol “<” is in an isolated column, it is easy to recode this symbol in a new variable that indicates when censoring is present using either 0/1 or TRUE/FALSE. In the second case is necessary to extract this symbol in a new column and recode it as an indicator variable.

How to handle each of these two cases is discussed in Appendix B.1 and B.2.

B.1 When ‘<’ and the observations are in separate columns

Because the symbol “<” is in a separate column, it is straightforward to recode this symbol in a new column that indicates censoring.

In Excel the unformatted will usually look like the picture below.

	A	B	C	D	E	F	G	H	I	J
1	Water Quality Results Columbia River at Waneta (0200559) April/May 2003									
2										
3				Apr-14		Apr-16		Apr-23		May-01
4	Field Dissolv Oxygen	mg/L		9.1				10.2		11.7
5	Field pH	pH units		7.4		7.48		7.96		8.06
6	pH	pH units		7.8		7.9		7.9		7.8
7	Field Conductivity	uS/cm		166		163		165		100
8	Residue Nonfilterable (TSS)	mg/L	<	4	<	4	<	4	<	4
9	Turbidity	NTU		0.53		0.48		0.32		0.27
10	Hardness Total -T	mg/L		69.4		69.6		68.3		68.3
11	Field Temperature	Celsius		6.6		6.3		7.4		8
12	Ammonia Nitrogen (N)	mg/L		0.012	<	0.005		0.018		0.04
13	Aluminum	ug/L		14.9		11.9		12.8		9.3
14	Antimony	ug/L		0.391		0.32		0.27		0.21
15	Arsenic	ug/L		0.4		0.3		0.3		0.3
16	Barium	ug/L		23.5		22.2		21.7		23.8

To reformat the column containing “<” symbols, apply the *IF* command. The *IF* command can be applied to a specific cell, and then extended to the rest of the column. For example, for cell C4, the instruction could be,

$$=IF(C4="<", 1, 0)$$

In this case the censoring is indicated using a 1 for censored observations, and 0 for non-censored ones.

The IF instruction is saying that if in cell C4 there is a symbol <, then define the value for this cell as 1, otherwise define it as 0. The values of censored and observed values remain in the same column.

After recoding all < symbols and sending the symbol columns at the end of the spreadsheet, the data will look like the table below

	A	B	C	E	F	H	I	K	L	N
1	Water Quality Results from Columbia River At Birchbank (0200003) April/May 2003									
2										
3				Apr-14		Apr-16		Apr-23		May-01
4	Field Dissolv Oxygen	mg/L	0	9.4	0	9.2	0	6.6	0	12
5	Field pH	pH units	0	7.38	0	7.54	0	7.98	0	8
6	pH	pH units	0	7.9	0	7.9	0	7.9	0	7.8
7	Field Conductivity	uS/cm	0	165	0	162	0	163	0	99.7
8	Residue Nonfilterable (TSS)	mg/L	1	4	1	4	1	4	1	4
9	Turbidity	NTU	0	0.54	0	0.4	0	0.26	0	0.26
10	Hardness Total -T	mg/L	0	69.1	0	70.9	0	67.7	0	69.9
11	Field Temperature	Celsius	0	6.3	0	6.2	0	7.2	0	8.4
12	Ammonia Nitrogen (N)	mg/L	0	0.008	1	0.005	0	0.012	1	0.005
13	Aluminum	ug/L	0	11.6	0	10.6	0	14.7	0	8.4
14	Antimony	ug/L	0	0.028	0	0.06	0	0.14	0	0.053
15	Arsenic	ug/L	0	0.3	0	0.2	0	0.3	0	0.2
16	Barium	ug/L	0	23.1	0	22.2	0	22.7	0	23.2

In the picture above we can see that “<” symbols have been replaced with 1s.

You can save the reformatted data set using a new name. This file can be directly imported into JMP from Excel.

B.2 When ‘<’ and the observations are in the same column

Many times the censored observations are denoted in Excel with expressions including the symbol < attached to the censored observation values in the same column.

In this case, the data in Excel looks like the picture below

	A	B	C	D	E
1	EOT PAH 2m depth ug/g	HB1.3	HB1.3	HB1.3	HB1.3
2	Acenaphthene	< 0.001	< 0.001	< 0.001	< 0.001
3	Acenaphthylene	< 0.002 (1)	< 0.002 (1)	< 0.002 (1)	< 0.002 (1)
4	Anthracene	< 0.001	< 0.001	< 0.001	< 0.001
5	Benzo(a)anthracene	0.007	0.006	0.006	0.003
6	Benzo(b+j)fluoranthene	0.013	0.019	0.014	0.004
7	Benzo(k)fluoranthene	0.002	0.004	0.004	0.001
8	Benzo(g,h,i)perylene	0.002	0.004	0.003	0.002
9	Benzo(e)pyrene	0.011	0.015	0.011	0.011
10	Benzo(a)pyrene	0.002	0.003	0.002	< 0.001

The symbol < can be extracted using the command *MID*. It can then be recoded using the instruction *IF*, as shown in the example above for cell B2.

$$C2=IF(MID(B2,1,1)("<",1,0)$$

The *MID(Text, Start_num, Num_chars)* command extracts a string character from a string expression. Where *Text* indicates the character to be extracted from, in this example B2; *Start_num* indicates the starting point for the string to be extracted, 1 in this example; and *Num_chars* indicates the length of the string to be extracted, 1 in this case.

The *IF* instruction returns 1 if the character extracted with *MID* command is <, and 0 otherwise.

A new column for the censored and uncensored values also needs to be created. The extraction of the characters from B2 starts at the second character i.e. at the beginning of the number rather at the symbol. On the other hand, the values that are uncensored should remain the same. The instruction for the column containing the censored and uncensored observations would be as follows.

$$D2=IF(C2=1,VALUE(MID(B2,2,6)),B2)$$

If an observation is censored, the instruction reads the text from the second character of length 6 and converts the character to a number because of the instruction *VALUE*. If, by contrary, the observation is uncensored, the value of C2 is 0 and the resulting value for D2 is the same as the one for B2.

The final sheet will look like this picture.

	A	G	H	I	J	K	L	M	N
1	EOT PAH 2m depth ug/g		HB1.3		HB1.3		HB1.3		HB1.3
2	Acenaphthene	1	0.001	1	0.001	1	0.001	1	0.001
3	Acenaphthylene	1	0.002	1	0.002	1	0.002	1	0.002
4	Anthracene	1	0.001	1	0.001	1	0.001	1	0.001
5	Benzo(a)anthracene	0	0.007	0	0.006	0	0.006	0	0.003
6	Benzo(b+j)fluoranthene	0	0.013	0	0.019	0	0.014	0	0.004
7	Benzo(k)fluoranthene	0	0.002	0	0.004	0	0.004	0	0.001
8	Benzo(g,h,i)perylene	0	0.002	0	0.004	0	0.003	0	0.002
9	Benzo(e)pyrene	0	0.011	0	0.015	0	0.011	0	0.011
10	Benzo(a)pyrene	0	0.002	0	0.003	0	0.002	1	0.001
11	Dibenz(a,h)anthracene	1	0.001	1	0.001	1	0.001	1	0.001
12	2,6-Dimethylnaphthalene	0	0.002	0	0.001	1	0.001	1	0.001
13	Fluoranthene	0	0.029	0	0.034	0	0.02	0	0.033
14	Fluorene	0	0.001	0	0.001	0	0.001	0	0.002
15	Indeno(1,2,3-cd)perylene	0	0.002	0	0.002	0	0.002	1	0.001

Appendix C

Transferring Data from Excel to R

There are two possibilities for transferring data from Excel to R. The first one is to format the file in Excel as shown in Section B and then just read the data in R. And the second one is to format the file in R.

C.1 Reading the formatted data in R

1. Load the required library *xlsReadWrite*

```
library(xlsReadWrite)
```

2. With the command *read.xls* read the data.

```
data = read.xls("DataName.xls", colNames=TRUE)
```

For example, when reading the *retena* data the instructions would be

```
retena = read.xls("RetenaData.xls", colNames=TRUE)
```

The code `colNames=TRUE` indicates that the columns in the file are named (e.g. *conc*).

For the above code to work, the work directory must be the same as the one where the data is stored. Otherwise the complete address can be specified similar to below.

```
data = read.xls("c:/CONSULTING/RetenaData.xls", colNames=TRUE)
```

C.2 Formatting the data in R

Once the data has been read into R as shown in Section C.1, the command `splitQual` can be used. `splitQual` can be used to separate the symbol < and the values in a column. The command `splitQual` must be applied to each column of values individually; it cannot be applied to the whole data set. For example, for the *retena* data the command would be:

```
conc.vector=splitQual(RetenaData$conc)
```

where `conc` is the name of the column containing the observed values for the concentration.

The command `splitQual` returns a data set with two variables for each vector it is applied to: one for the observed values, which is named `obs`, or in this case `conc.vector$obs`; and the other one for the indicator variable for the censoring, which is named `cen`, or in this case `conc.vector$cen`.

The list of objects, i.e. the column names, that are part of an object or data set can be obtained using the command `attributes`. For instance, for the object `conc.vector`, this command would give the next output.

```
> attributes(conc.vector)
$names [1] "obs" "cen"
```

The censoring indicator variable is coded as TRUE/ FALSE for censored and uncensored observations, respectively.

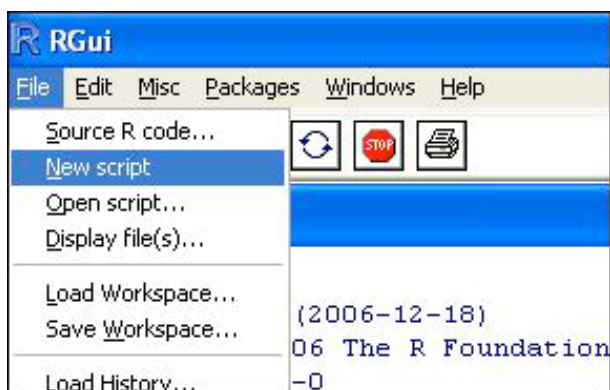
Appendix D

Starting with R

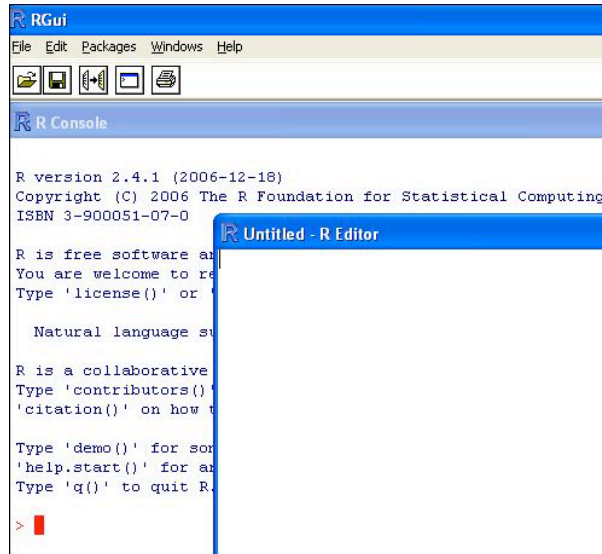
R is a language used to perform statistical computing and graphics. This software allows the user to execute calculations by using the available functions, or by programming new routines. Specifically for environmental data with non-detects there is a package available (NADA: Nondetects and Data Analysis for environmental data) which carries out statistical analysis specific to this kind of research.

Below are some steps to help start running R for the first time.

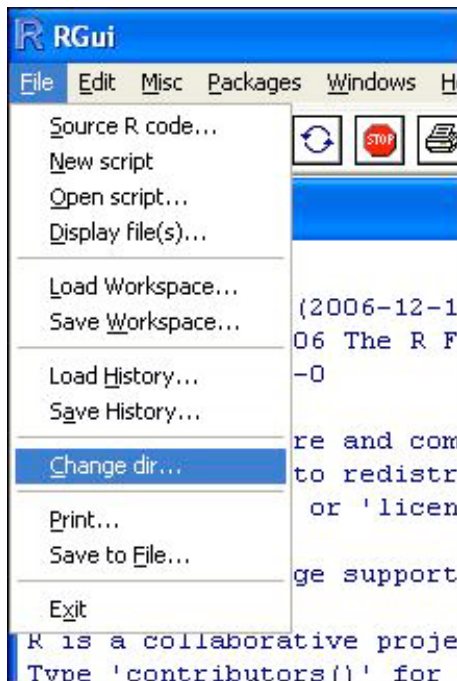
1. **Go to the R website and install the program.** The R software package can be downloaded from the website <http://www.r-project.org/> following the instructions given in R website.
2. **Open R.** To open *R* double click on the R icon or go to *Start->Programs->R*. Then select *File-> New Script*. Work can be saved in this script file. See below.



The full screen will look similar to below.



3. **Specify the directory.** To specify the directory where you want to work and save your files, you can change directories by going from the R console to *File*→*Change Dir*.

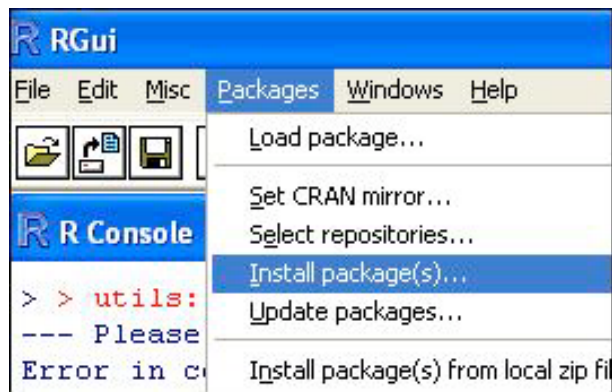


Specify the directory where you want to save the script, as in the example.
Select OK.



4. **Packages.** When we first download R, it includes only very basic statistical functions. To analyze more specialized data, such as we have, it is necessary to download special ‘libraries’ or ‘packages’.

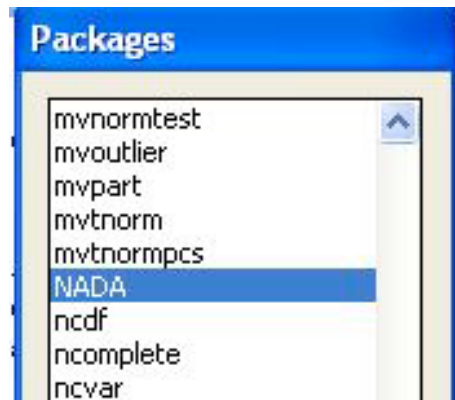
It is not necessary to install these libraries every time R is opened. In this case the package needed is NADA (Nondetects and Data Analysis for environmental data). To load the NADA package, go to *Packages* -> *Install Package(s)*



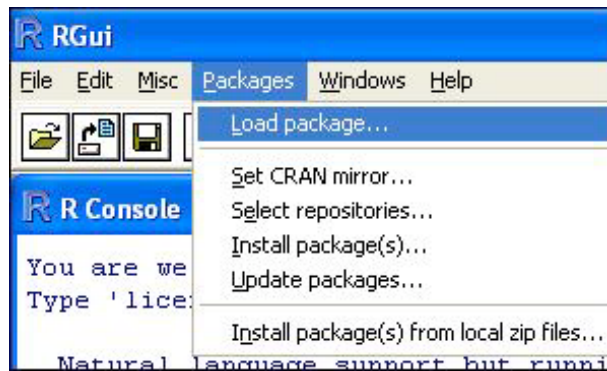
Select your location and click *OK*. See the below for step by step pictures.



Select the package to install, in this case *NADA*, and click *OK*.



5. **Load the NADA library.** This library is needed to perform statistical analysis for censored data. It is necessary to load the library every time *R* is opened even though it only needs to be downloaded once. To load the library, go to *Packages* -> *Load Package*->*OK*.



After selecting *NADA* click *OK*.

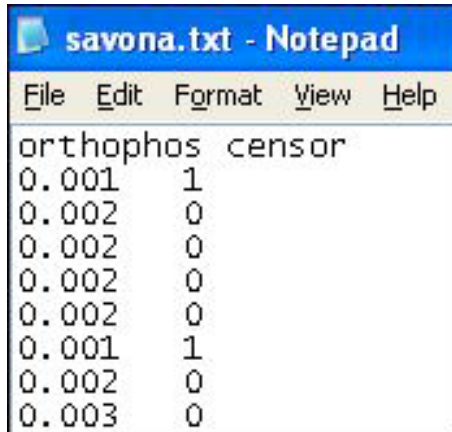


Another way to load NADA package is to write in instruction in the R console

```
library(NADA)
```

6. **Read in the data.** There are two ways of reading the data into R. One is as a text file with the format 'txt'. The other alternative is to read it from an Excel file. If the data is text, use the commands shown below.

```
savona = read.table('savona.txt',header=TRUE)
```



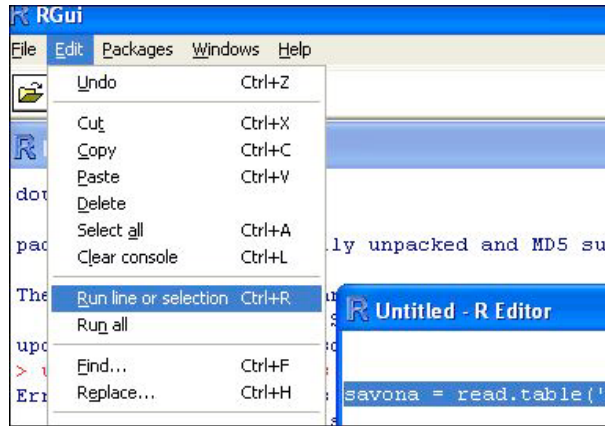
orthophos	censor
0.001	1
0.002	0
0.002	0
0.002	0
0.002	0
0.001	1
0.002	0
0.003	0

The code *Header=TRUE* tells R that the file has names. A '1' or '0' denotes whether an observation is censored or not, respectively. A description of these data is given in the Appendix A.1.

For reading in the data in from Excel files, there is detailed information in Appendix C.1. The instructions would be as follows.

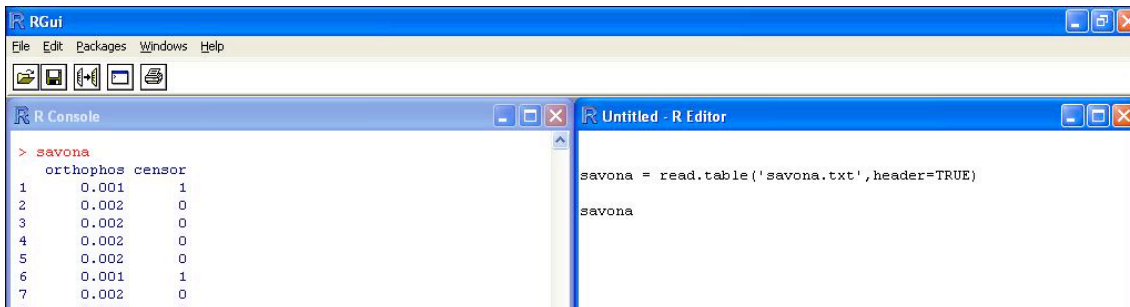
```
library(xlsReadWrite)
savona = readxls('savona.xls',colNames=TRUE)
```

7. **Run the procedures in R.** In the R editor, select the lines that you wish to run in R, or to run everything use the run all command. To do this go to *Edit -> Run all*. Or *Edit -> Run line or selection*, having previously selected the line (s).

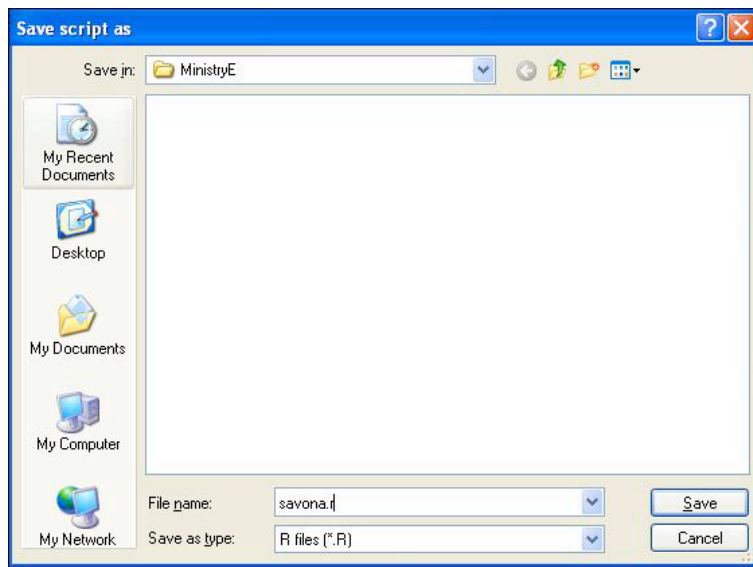
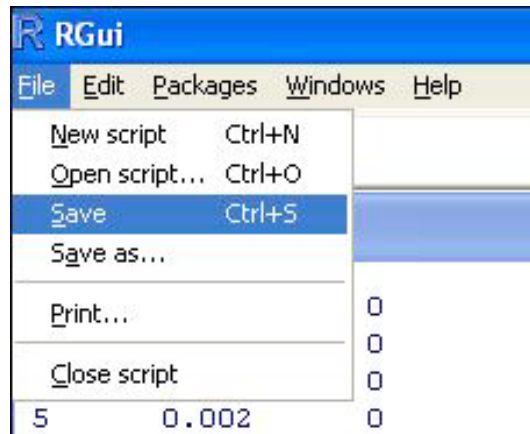


A shortcut for running a selection is to select the command(s) to be run, and then clicking *CTRL+R*.

8. **Visualize the data.** Type the name of the file in the console and *ENTER*, in this case `savona`, or type `savona` in the script file and press *CTRL+R*. Then the data is printed in the console window as is shown in the next step. This is not a spreadsheet of the data, so any changes should be made in Excel and imported into R.
9. **Organize the screen.** To visualize the windows for both the console and the script file, go to *Windows-> Title*.



10. **Save a script file.** After highlighting the 'CURSOR' in the script window, go to *File -> Save*. Write a name for the file in usual way.



11. **Create a logical indicator (of censoring).** The last step of starting with R is to give a format to the data such that the analysis is straightforward. The *NADA* package uses a logical indicator to represent censored data. It takes one line to create a new variable that has the required format.

```
savona$cen =ifelse(savona$censor==0,FALSE,TRUE)
```

Again, the command is executed by highlighting the commands and clicking *CTRL+R*. Now the *savona* file has three variables which can be checked with the command

```
names(savona) .
```

In the console window appears

```
29      0.007      0
30      0.007      0
31      0.006      0
32      0.005      0
> savona$cen =ifelse(savona$ censor==0,FALSE,TRUE)
> names(savona)
[1] "orthophos" "censor"    "cen"
> █
```

Clearly, it might be the case that the data is already in the required format in which case this step can be omitted.

12. **Comments in R.** Comments in the script file are written using the symbol #

```
#----- Reading the data -----
savona = read.table('savona.txt',header=TRUE)

#----- visualizing the data -----
savona

#----- Creating a logical indicator of censoring
savona$cen =ifelse(savona$ censor==0,FALSE,TRUE)

#----- Printing the names of the data -----
names(savona)
```

13. For other questions about R, there is some documentation online at the R website *www.r-project.org*. For someone familiar with programming a good introduction to R is the manual *An introduction to R* available in the R website.

Appendix E

Bootstrap

The bootstrap method is popular for computing standard errors in cases where the formulas for computing the standard errors directly are complex.

There are slight variations of the bootstrap method. The one shown in Helsel (2005b) is bootstrap with replacement. This bootstrap method consists in taking many samples with replacement of the original sample (the same sample size), and computing the statistics over all the different samples. Then a distribution of the statistic is obtained and a confidence interval can be calculated from it.

The bootstrap algorithm is as follows (Helsel, 2005b).

1. From the original set of n observations, take a sample of size n with replacement. This sample will be called the generated sample. Because not all observations are used, the sample will be different than the original.
2. Compute an estimate of the statistics of interest in the generated sample. For instance, the mean.
3. Save the estimate and repeat the process r times with new generated samples of data. A common range for r is between one thousand to ten thousand. These samples allow the user to estimate the distribution of the statistics of interest.
4. Compute the statistics of interest using the estimates from the r samples, for example, the mean. The mean of the means is the bootstrapped estimate.
5. The percentiles of the distribution of the statistics give a confidence interval for the estimate of interest. For instance, for the mean, the 2.5th and 97.5th percentiles give a 95% confidence interval for the mean.

Appendix F

Probability plotting for distributions other than the lognormal

Comparing the `savona` data to distributions other than the lognormal.

1. Load the `lattice` package in R using the command

```
library(lattice)
```

2. Construct a plot using the `qqmath` command.

```
qqmath(x=savona$obs,distribution=qnorm)
```

Within the `qqmath` command, the `distribution=..` argument can be set to any of the following:

- (a) `qnorm`, for comparison to a normal distribution.
- (b) `qunif`, for comparison to a uniform distribution.
- (c) `qf`, for comparison to an F distribution.
- (d) `qchisq`, for comparison to a chi-square distribution.
- (e) `qexp`, for comparison to an exponential distribution.

As with p-p plotting in JMP, using the `qqmath` command does not account for censoring. Values on the y-axis lower than the CL should be ignored when assessing whether points fall approximately on a straight line.

Appendix G

Kaplan-Meier

G.1 Computation of the Kaplan-Meier estimator

The KM is an estimator of the survival curve, S , and it can be represented symbolically as

$$S = P(T > y).$$

This can be interpreted as the probability of an individual surviving past time y .

The following four steps describe the procedure to calculate a Kaplan-Meier estimate; the process is illustrated with the `savona` data example used in Table G.1

1. Create one row for each detected value. Transforming the data will result in the creation of new columns
2. To transform the data from left censored to right censored, choose a constant larger than the maximum of the observed values. For example, in the `savona` data the constant can be $M = 0.012$, which is larger than the maximum value of 0.009.
3. Subtract the observed data from the constant to make the data appear to be right censored i.e. `newOrtho=M-Orthophos`. Note that `M` is our constant, and `Orthophos` is the observed concentration value.
4. For each transformed value, the probability being larger than t given that it is at least t is calculated. This estimated probability of ‘survival’ (p in Table G.1) is the difference between the number of transformed values greater than t and the number of values that are less than t , divided by the number of values greater than t

$$p = (b - d)/b$$

5. The survival probability (S) at each time or detected observation is the product of the previous incremental survival probabilities (p)

$$S_t = \prod_{i < t} p_i$$

Table G.1: Example computations of the Kaplan-Meier estimator for the `savona` data.

Orthophos	newOrtho	# at risk (b)	# detects or failed (d)	$p = (b - d)/b$	S
0.009	0.003	32	1	31/32	0.9688
0.008	0.004	31	1	30/31	0.9375
0.007	0.005	30	3	27/30	0.8438
⋮					

The instructions above are used to to create the last two columns shown in Table G.1.

In the Table G.1, based on the transformed data, the probability of surviving past time 0.003 is 31/32. In words this is saying that there are still 31 subjects larger than 0.003 out of 32, so the probability is 31/32. For calculating the rest of the p 's the logic is similar.

Once the p 's are calculated, the computation of S is straightforward. The values of S are computed as the product of the previous p 's. For example, $S_1 = 31/32 = 0.9688$, and $S_2 = \frac{31}{32} \times \frac{30}{31} = 0.9375$.

Note that in Table G.1 there is no correction for censoring, since the censored observations for the environmental data are found at the end of the list and are not shown in our table. (The first observation is censored.)

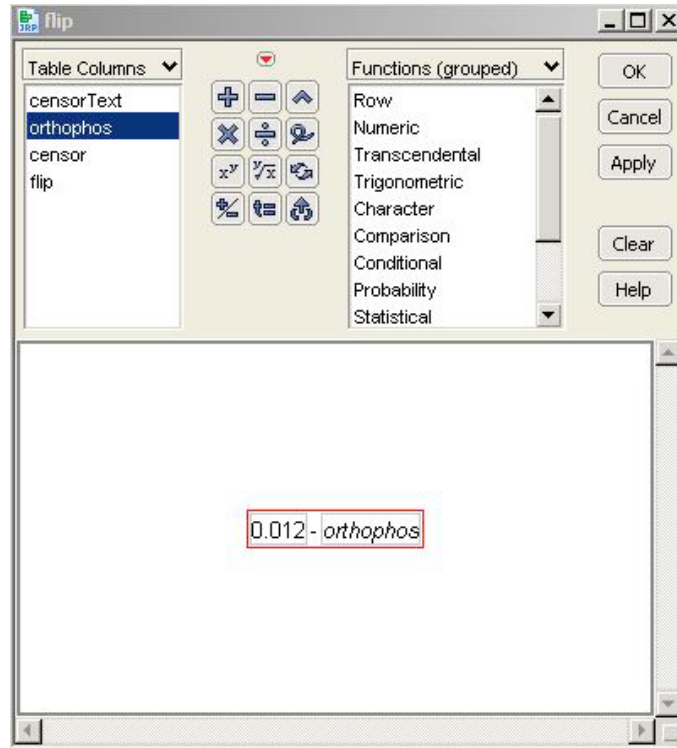
For further information on the computation of KM estimator, please see the Section G.2 and G.3.

G.2 Kaplan Meier estimation using JMP

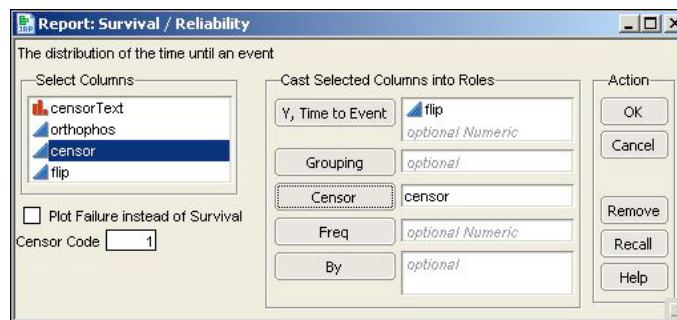
Because the Kaplan-Meier in most statistical packages does not accommodate left censored data, it is necessary to first transform the data, subtracting each observation from a constant greater than the maximum value.

Below are the basic steps to conduct a Kaplan-Meier analysis in JMP.

1. Transform the data, subtracting each value from a constant greater than the maximum value. For example, in this case the maximum value of orthophosphate is 0.009, so a constant of $M = 0.012$ would be enough. Any arbitrary constant larger than 0.009 would also be appropriate. In JMP, this transformation can be performed by going to *Cols -> New Column -> Column properties -> Formula*, and then the window shown below will appear. In this window any formula can be specified. For this case we compute the new variable as `M-orthophos` (orthophos is the name of the variable).



2. Compute the KM estimates. For the `savona` data in the censored column, the observations are denoted as 1 for censored and 0 for observed values. Go to *Analyze* -> *Survival and Reliability* -> *Survival/Reliability*.



Make sure that the censor code in the above window corresponds to the censoring code in the data.

3. Back transform the data, subtracting the estimates of mean and percentiles from the constant M using the formula $M - \text{newOrtho}$.

The output from JMP for the `savona` data is shown below. These are estimates are for the transformed data. It is necessary to back transform the estimates of mean and percentiles, subtracting from the same constant again.

For example for the mean, the estimate obtained is 0.00819 for the transformed data, which gives

Time to event: flip
Censored by censor
Censor Code 1

Summary

Group	N Failed	N Censored	Mean	Std Error
Combined	25	7	0.00819	Biased 0.0004

Quantiles

Group	Median Time	Lower95%	Upper95%	25% Failures	75% Failures
Combined	0.01	0.006	0.01	0.006	0.01

Combined

$0.012 - 0.00819 = 0.00381$ for the mean of the orthophosphate concentration. Notice that even JMP tells us that this estimate of the mean is biased.

Table G.2: Table of summary statistics using Kaplan-Meier for the `savona` data.

Mean	25 th	Median	75 th
0.00381	0.002	0.002	0.006

Although the standard error is reported, this is not reliable in the Kaplan-Meier method.

G.3 Kaplan Meier estimation using R

The `NADA` package in R can be used to compute statistics for data with nondetects. In R it is not necessary to transform the data in order to compute the Kaplan-Meier estimate; R software can accommodate left censored data.

The command `cenfit` in the `NADA` package is used to compute summary statistics based on Kaplan-Meier estimates for data with left censored data as shown the instructions below.

1. Fit the KM model.

```
km = with(savona,cenfit(obs,cen))
```

2. Plot the Kaplan-Meier curve, if necessary.

```
plot(km,main='Kaplan-Meier')
```

Notice that Figure G.1 shows the estimate of the cumulative distribution function of the data, and not the survivor probability function as in JMP. For example, the 3rd quartile is the number such that 75% of the data are below it. This can be confirmed with the output from `quantile` command.

3. Request summary statistics. Use the commands `quantile`, `mean` and `sd` as shown.

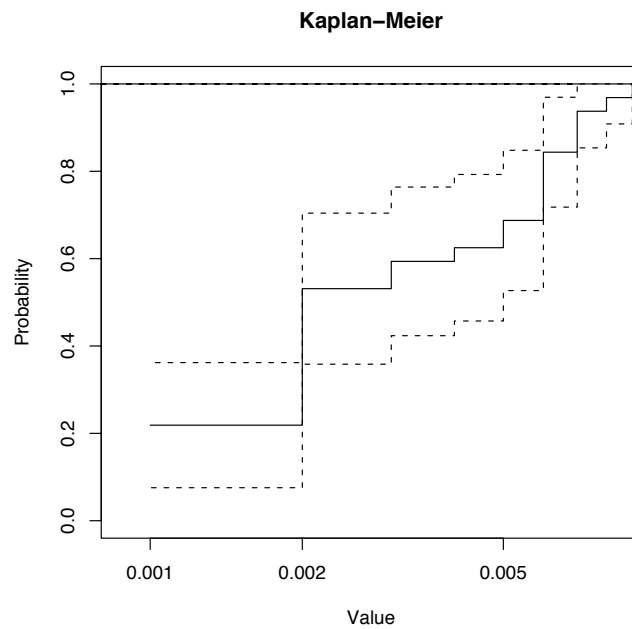


Fig. G.1: Cumulative distribution function based on the Kaplan-Meier estimate for the `savona` data.

```
mean(km) quantile(km)
```

The output from R is

```
> mean(km)
      mean      se    0.95LCL    0.95UCL
0.0038125000 0.0004020149 0.0030245652 0.0046004348
```

The mean estimated based on the Kaplan-Meier method is not necessarily a useful statistics because is often biased.

```
> quantile(km)
 5%  10%  25%  50%  75%  90%  95%
NA   NA 0.002 0.002 0.006 0.007 0.008
```

The first row shows the quantile probability, and the second row gives the quantile estimate for each probability.

Notice that the estimates of the quantiles lack of the corresponding standard error, or a confidence interval. The Section G.4.1 provides the guidelines to conduct the estimation of confidence intervals for the percentiles.

G.4 Computing confidence interval estimates using the B-C inverted sign method

Interval estimates are more advantageous estimates than point estimates since they give a measure of uncertainty about an estimate. It is possible to say that we are 95% confident that the true parameter, e.g. the mean, is within this interval.

G.4.1 B-C inverted sign method

The B-C sign method is used for finding confidence intervals for percentiles based on the Kaplan-Meier method. It uses the survival probability of the observations¹, as well as the standard error associated with each survival probability.

The idea behind B-C inverted sign method is to find confidence intervals for the survival probabilities, assuming that these quantities follow a normal distribution. Then by “inverting” the interval, a confidence interval for the observations can be found, where “inverting” means going from the probability units to the units in the observations. The survival probabilities for each observation and their standard errors are provided in the JMP or R Kaplan-Meier estimation output.

The method is provided in the following steps.

- (a) Fit the Kaplan-Meier model as shown in Section 6.4.
- (b) Set p as the probability of the quantile we are looking for.
- (c) Set the confidence level, α , of the confidence interval we are looking for.
- (d) Using α , find the corresponding theoretical quantiles to use in finding a confidence interval. For instance, if $\alpha = 0.05$, then the theoretical quantiles are -1.96 and 1.96.
- (e) Standardize the probabilities. Define z as the standardized values, where z is

$$z = \frac{\hat{p} - p}{se(\hat{p})}$$

and \hat{p} are the survival probabilities obtained from the JMP or R output; $se(\hat{p})$ are the standard errors for each probability.

Note that the z values follow a normal distribution with mean 0 and standard deviation 1.

- (f) Compare the z values with the theoretical normal quantiles (-1.96 and 1.96). The z values within the range of -1.96 to 1.96 belong to the confidence interval of the standardized probability. Call these numbers potential values.
- (g) Find the extremes of the potential values: the minimum and the maximum.
- (h) Find the corresponding observations that lead to the extreme values. This is the confidence interval for the $100 \times p^{th}$ percentile.

¹Recall that the survival probability is $1 - cdf$, where cdf is the cumulative distribution function.

G.4.2 Using the B-C inverted sign method in computing interval estimates

The following code in R can be used to compute the standard error for each percentile, as it is discussed below.

```
ci.quantiles <- function(alpha=.05,p=.5,km.fit){
  sum.km <- summary(km.fit)
  z.t <- qnorm(1-alpha/2)
  #z values
  z <- (sum.km$prob - p)/sum.km$std.err
  ci <- ifelse(abs(z)<z.t,TRUE,FALSE)
  lower.z <- min(z[ci])
  upper.z <- max(z[ci])
  lower <- min(sum.km$obs[z==lower.z])
  upper <- max(sum.km$obs[z==upper.z])
  data.frame(lower,upper)
}
```

The function `ci.quantiles` needs to have some parameters to compute the confidence intervals: `alpha`, which is the confidence level for the intervals; `p`, is the percentile probability, and `km.fit` is the Kaplan-Meier fit.

Notice that the Kaplan-Meier estimate should be previously computed in order to find confidence intervals for percentiles using the B-C inverted sign method.

For example continuing with the `savona` data, if we are interested in getting confidence intervals for the quartiles of the concentration of orthophosphate we can simply apply the function as follows.

```
ci.quantiles(alpha=.05,p=.75,km.fit=km)
```

The instruction `ci.quantiles(.05,p=.75,km)` is asking for 95% (`alpha=.05`) confidence intervals for the third quartile (`p=.75`) using the Kaplan-Meier estimate.

The result for the confidence interval for the 75th percentile is shown below.

```
> ci.quantiles(alpha=.05,p=.75,km.fit=km)
  lower upper
0.004 0.007
```

Using the B-C inverted sign method to compute a confidence interval for the third quartile, we can observe that the 75th percentile is between 0.004 and 0.007 with 95% confidence.

Note that an interval estimate for a percentile near the censoring limit might give an useless estimate since is too close to the censoring value to be estimated accurately.

As it should be, the results from R and JMP are the same. It is confirmed that the 1st quartile and the mean share the same value of 0.002, and the 3rd quartile is 0.006. Notice that quantiles 0.05 and 0.10 are reported as undefined since their values are below the censoring limit (the percentage of censoring is greater than the probability of the quantiles).

Appendix H

How to install and use an R library when you don't have administrator privileges

To save and install an R library when you don't have administrator privileges on your computer is actually a relatively straightforward and easy thing to do. This means that once R has been installed you can update it with new packages at your own discretion without having to go through an administrator. At the same time, you should probably encourage administrators to download and install updated versions of the core R program about once every 1 or 2 years, but I digress.

Option one is to simply work around your administrators entirely, and install R into a directory you have access to. You can access the R website as described in Appendix D, and then download the *base* installation of the program as an *executable (.exe)* file. When you run through the program setup, the installation will tell you that you aren't an administrator, but then ask if you want to install R into a folder that you do have access to. For example, I have saved R in my `C:\Documents and Settings\chuston\` folder. The installation will then mention that R might be missing a few bits if you install this way, but it will go ahead and try.

Having done this myself, I find that R itself has worked fine so far, but that installation, and downloading and installing packages is slower than if I had administrator privileges.

If you install R in this way, you can download packages exactly as described in Appendix D using the *Install Package(s)* function from the *Packages* menu. No other steps should be necessary.

Option two is also fairly straightforward, and is the technique to use if you have ordered that R be installed on your computer from your systems administrator.

So you want to use R from your Windows account, but you want to use a package that is not a part of the base installation that the administrators have provided. For example, you might want to install the NADA package so that you can analyze some water quality data. Following the instructions from Appendix D, you select *Install Package(s)* from the *Packages* menu, and then select a local mirror and the package you want to install. Unfortunately, you don't have permission to install the package into the global directory for the ministry. This is okay too, because R is smart and asks if you want to install it locally. It even asks you if you want it to create a special directory

for your libraries - something like `\\D0017216\home\chuston\R` libraries. Sounds great, but that is when things can go wrong and you might get the error message

```
Warning in install.packages(NULL, .libPaths()[1], dependencies=NA, type="type"):  
  'lib="c:/PROGR~1/R/R-28~1.1/library"' is not writeable  
Error in install.packages(NULL, .libPaths()[1], dependencies=NA, type="type"):  
  unable to create 'D0017216\home\chuston\R'.
```

Why does this happen, you wonder? I am not completely sure, but I think it has something to do with the fact that your location is actually in your Ministry home directory, and not on your local Windows computer. Subsequently, even if you create this library directory the installation will still fail.

To get around this, we need to create a local space for R libraries, and then tell R to use it, rather than trying to select a folder to install the library in. To do this, follow the steps below.

1. Create a local library directory: The library directory can be anywhere in your space on the local C: drive. For example, you could use the path `C:\Documents and Settings\chuston\R\libs`, which requires me to create the top directory R, and then the subdirectory libs.
2. Define the environment variable R.LIBS: When R starts, it checks to see if you have defined certain variables that determine how R will run. One of these variables, R.LIBS, tells R the location of your local library repository - the folder that you created in step 1. To define this variable, click on My Computer → then find *System Tasks* in the left menu → click on *View system information* → click on *Advanced* → click on *Environment Variables* → find *User variables for* → click on *New*. You should now be able to create a new user variable with the name R.LIBS, and give it a value which is the full path name from step one (in this example `C:\Documents and Settings\chuston\R\libs`)

That should do it! Now you can restart R, and all should work smoothly. To check that the packages are being installed in the correct place, just have a look in your lib folder to make sure they are there.

Appendix I

R code to run function DiffCI

To run this, or any function written for R, the first step is to create a text file that contains all of the commands listed below. By text file, I mean a .txt file, not a .doc file! It is probably easier to work in a text editor that is not Microsoft Word in order to save confusion. Unfortunately, syntax must be exact. All brackets must be closed, and all names must be the same. Typos will either cause the function to run improperly, or not at all.

Once the text is typed in an editor, you can simply highlight the commands for the function, copy them, and paste them into the R console. After hitting return, R should process all the commands in the console window, and correctly store the function. If you type `DiffCI`, with no brackets or arguments, you should see the function as you entered it if everything worked properly.

```
DiffCI=function(cenobj,conf.level=0.95,ngroups=2){
#This function requires as its argument a cenmle object
#it is also possible to request confidence level desired
#and the number of groups that have differences where
#intervals are needed

results=summary(cenobj) #they need to pass the fn a cenmle object
variance=results[8]
v2=unlist(variance)
v3=matrix(data=v2,ncol=ngroups+1,nrow=ngroups+1)

coef=unlist(results[7])
diffs=coef[2:ngroups]

vars1=diag(v3)
ses=sqrt(vars1[2:ngroups])

#remember this is essentially based on a z distribution
critical1=(1-conf.level)/2
critical2=abs(qnorm(critical1))
width=critical2*ses
```

```
#need to get the value of the difference to put in the middle!!  
lls=diffs-width  
uls=diffs+width  
  
interval=cbind(lls,uls)  
colnames(interval)=c("lower","upper")  
return(interval)  
  
}#ends DiffCI
```

Appendix J

R function for differences in paired observations

```
makePaired=function(pairedata){
#data sent to this function should have the form
#from column 1 onward
#observationNumber/ obs1/ Group(Month) / Cens1(TF) / obs2 / Group(Month) / Cens2(TF)
# where in the table with the above columns it is assumed
#that the difference calculation will be obs2-obs1

pairedata$diffL=NA
pairedata$diffU=NA

for(i in 1:NROW(pairedata)){

if(pairedata[i,4]==FALSE & pairedata[i,7]==FALSE){

pairedata$diffL[i]=pairedata[i,5]-pairedata[i,2]
pairedata$diffU[i]=pairedata$diffL[i]

}#ends if for case where neither value is censored

if(pairedata[i,4]==TRUE & pairedata[i,7]==FALSE){

pairedata$diffL[i]=pairedata[i,5]-pairedata[i,2]
pairedata$diffU[i]=pairedata[i,5]-0

}#ends else if where first reading is censored

if(pairedata[i,4]==FALSE & pairedata[i,7]==TRUE){

pairedata$diffL[i]=0-pairedata[i,2]
```

```

pairedata$diffU[i]=pairedata[i,5]-pairedata[i,2]

}#ends case where only the second reading is censored

if(pairedata[i,4]==TRUE & pairedata[i,7]==TRUE){

pairedata$diffL[i]=0-pairedata[i,2]
pairedata$diffU[i]=pairedata[i,5]-0

}#ends if case where both values are censored

}#ends i for

data=cbind(pairedata$diffL,pairedata$diffU)
upper=pairedata$diffU
lower=pairedata$diffL

data=cbind(upper,lower)
colnames(data)=c("diffU","diffL")

return(as.data.frame(data))

}#ends makePaired

```

Appendix K

Code for MLE test of Paired Differences

```
PairDiffCI=function(obj,conf.level=0.95){
#This function requires as its argument a survreg object
#it is also possible to request confidence level desired
#and the number of groups that have differences where
#intervals are needed

results=summary(obj) #they need to pass the fn a cenmle object
estimate=unlist(results[7])
variance=results[8]
v2=unlist(variance)
v3=matrix(data=v2,ncol=2,nrow=2)

se=sqrt(v3[1,1])

#remember this is essentially based on a z distribution
critical1=(1-conf.level)/2
critical2=abs(qnorm(critical1))
width=critical2*se
#need to get the value of the difference to put in the middle!!
ll=estimate-width
ul=estimate+width

#might also want the p-value for the x-square test
#of no difference
values=matrix(unlist(forInt[9]),ncol=4,nrow=2)
p.value=values[1,4]

interval=c(ll,estimate,ul,p.value)
names(interval)=c("lower","estimate","upper","p.value")
return(interval)
```


}#ends PairDiffCI

Appendix L

Non-parametric paired data test functions

Note that the `numerator(...)` and `denominator(...)` functions are used internally by the `signs(...)` and `calcP(...)` functions. This means that they must be loaded into R in order for the `signs(...)` and `calcP(...)` functions to work correctly, even though they are never directly called by the user.

```
numerator=function(N,plus,neg,ntie){

value=max(c(plus,neg))

dfn=vector(mode="numeric",length=N-value+1)
for(i in value:N){
dfn[i-value+1]=dbinom(x=i,size=N,prob=0.5)

}#ends i for

return(dfn)
}#ends numerator

denominator=function(n,tie){

number=(n-tie+1)/2
number=floor(number)

dfn=vector(mode="numeric",length=n-number+1)

for(i in number:n){

dfn[i-number+1]=dbinom(x=i,size=n,prob=0.5)

}#ends i for
```

```
return(dfn)
```

```
}#ends denominator
```

The `signs` function takes the output from the `makePaired(..)` function, and calculates the number of positive differences, negative differences, and tied differences.

```
signs=function(pairs){
```

```
  N=NROW(pairs)
```

```
  pairs$sign=NA
```

```
  ntie=0
```

```
  nplus=0
```

```
  nneg=0
```

```
  for(i in 1:NROW(pairs)){
```

```
    #print(as.logical(pairs[i,1]>0))
```

```
    if(pairs[i,1]>0 & pairs[i,2]>0){
```

```
      pairs$sign[i]=1
```

```
      nplus=nplus+1
```

```
    }#ends if
```

```
    if(pairs[i,1]<0 & pairs[i,2]<0){
```

```
      pairs$sign[i]=-1
```

```
      nneg=nneg+1
```

```
    }#ends if
```

```
    if((pairs[i,1]<0 & pairs[i,2]>0)|(pairs[i,1]>0 & pairs[i,2]<0)|(pairs[i,1]==0 & pairs[i,2]==0))
```

```
      pairs$sign[i]=0
```

```
      ntie=ntie+1
```

```
    }#ends else
```

```
  }#ends i for
```

```
  dataplus=list(pairs,nplus,nneg,ntie)
```

```
  return(dataplus)
```

```
}#ends signs
```

The `calcP(..)` function takes the output from the `signs(..)` function and calculates the two sided p-value for the modified sign test.

```
calcP=function(signlist){
```

```
  #a function that takes the the output from the signs function
```

```
  #and calculates the p-value for the Fong corrected score test
```

```
trial=signlist
nplus=unlist(signlist[2])
nneg=unlist(signlist[3])
ntie=unlist(signlist[4])

N=nplus+nneg+ntie

probn=sum(numerator(N,nplus,nneg,ntie))
probd=sum(denominator(N,ntie))

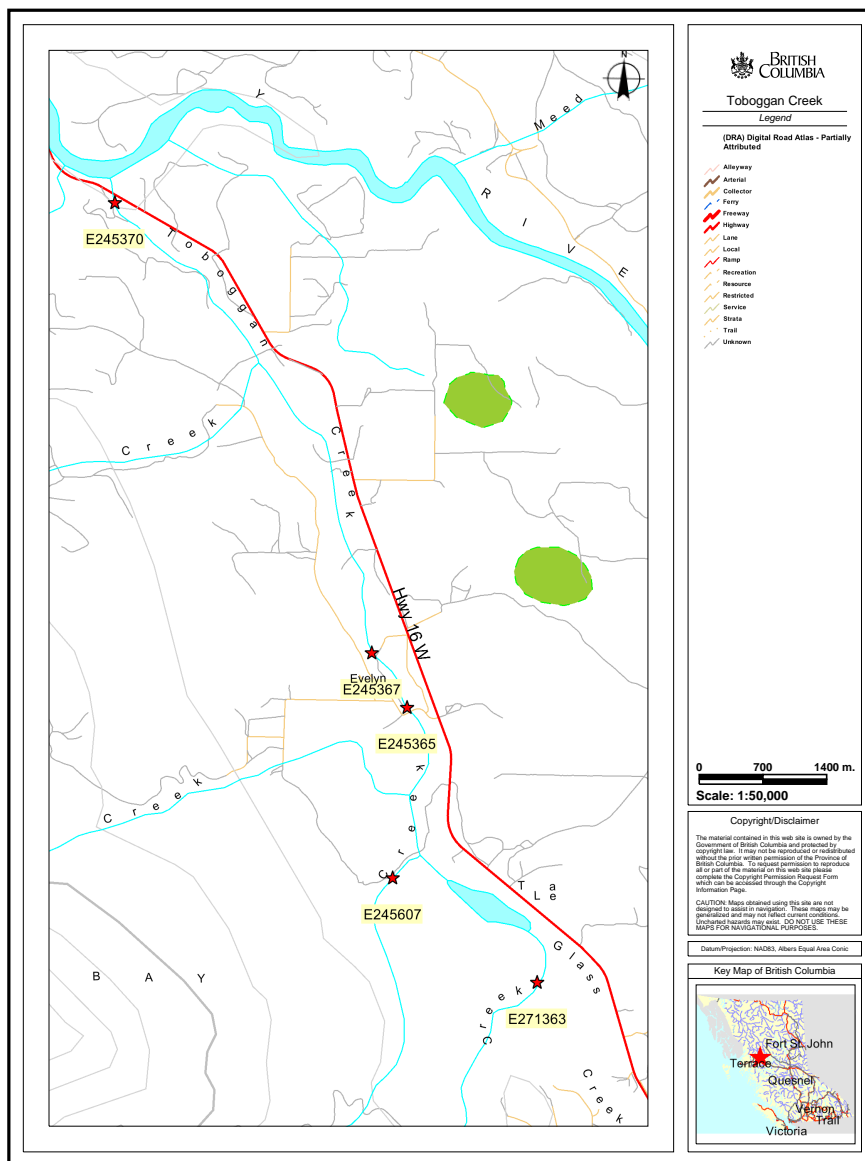
fongP=probn/probd

return(fongP)

}#ends calcP
```

Appendix M

Toboggan Creek Sampling Region



Appendix N

Vanadium Data Set

	Vanadium	VanCen	Location	LocCode
1	0.2	TRUE	Railway	E271363
2	0.2	TRUE	UpperTCreek	E245607
3	0.2	TRUE	Hatchery	E245367
4	0.2	TRUE	Hwy16	E245370
5	0.2	TRUE	Intake	E245365
6	0.4	FALSE	Hwy16	E245370
7	0.3	FALSE	Hatchery	E245367
8	0.2	FALSE	Railway	E271363
9	0.2	FALSE	UpperTCreek	E245607
10	0.2	FALSE	Intake	E245365
11	0.2	FALSE	Intake	E245365
12	0.2	TRUE	Hwy16	E245370
13	0.2	TRUE	Hatchery	E245367
14	0.2	TRUE	Intake	E245365
15	0.2	TRUE	UpperTCreek	E245607
16	0.2	TRUE	Railway	E271363
17	0.3	FALSE	Hatchery	E245367
18	0.3	FALSE	Intake	E245365
19	0.4	FALSE	Railway	E271363
20	0.4	FALSE	UpperTCreek	E245607
21	0.4	FALSE	Hwy16	E245370
22	0.2	FALSE	Railway	E271363
23	0.6	FALSE	UpperTCreek	E245607
24	0.3	FALSE	UpperTCreek	E245607
25	0.2	FALSE	Hatchery	E245367
26	0.2	FALSE	Hwy16	E245370
27	0.2	TRUE	Intake	E245365
28	0.2	TRUE	UpperTCreek	E245607
29	0.2	TRUE	Hatchery	E245367
30	0.2	TRUE	Hatchery	E245367
31	0.2	TRUE	Hwy16	E245370
32	0.2	TRUE	Railway	E271363

33	0.2	TRUE	Intake	E245365
34	0.2	FALSE	Hwy16	E245370
35	0.2	FALSE	Hatchery	E245367
36	0.2	FALSE	Intake	E245365
37	0.3	FALSE	UpperTCreek	E245607
38	0.3	FALSE	Railway	E271363
39	0.3	FALSE	Railway	E271363
40	0.3	FALSE	Railway	E271363
41	0.2	TRUE	UpperTCreek	E245607
42	0.3	FALSE	Intake	E245365
43	0.2	FALSE	Hatchery	E245367
44	0.3	FALSE	Hwy16	E245370

Appendix O

Simulated Seasonal Water Quality Data

	season	observation	values	year	cen
1	lowSeason	1	1.0	1	TRUE
2	highSeason	2	1.0	1	TRUE
3	lowSeason	3	4.4	2	FALSE
4	highSeason	4	1.9	2	FALSE
5	lowSeason	5	2.0	3	FALSE
6	highSeason	6	1.0	3	TRUE
7	lowSeason	7	5.0	4	FALSE
8	highSeason	8	1.0	4	TRUE
9	lowSeason	9	2.6	5	FALSE
10	highSeason	10	2.7	5	FALSE
11	lowSeason	11	5.2	6	FALSE
12	highSeason	12	1.7	6	FALSE
13	lowSeason	13	3.6	7	FALSE
14	highSeason	14	1.9	7	FALSE
15	lowSeason	15	3.3	8	FALSE
16	highSeason	16	1.0	8	TRUE
17	lowSeason	17	4.2	9	FALSE
18	highSeason	18	1.0	9	TRUE
19	lowSeason	19	7.1	10	FALSE
20	highSeason	20	2.2	10	FALSE
21	lowSeason	21	3.1	11	FALSE
22	highSeason	22	2.2	11	FALSE
23	lowSeason	23	3.5	12	FALSE
24	highSeason	24	1.1	12	FALSE
25	lowSeason	25	8.3	13	FALSE
26	highSeason	26	4.6	13	FALSE
27	lowSeason	27	6.6	14	FALSE
28	highSeason	28	2.3	14	FALSE
29	lowSeason	29	3.0	15	FALSE
30	highSeason	30	4.7	15	FALSE

- Altman, R. 2008, Course Information for Stat 402: Introduction to Generalized Linear Models, Simon Fraser University, www.stat.sfu.ca/~raltman/stat402.html, accessed Dec.8, 2008.
- Clark, M. J. R. (1994), "Conflicting perspectives about detection limits and about the censoring of environmental data", *Water Resources bulletin. American Water Resources Association*, 30, 1063-1079.
- Dietz, E.J., 198, A comparison of robust estimation in simple linear regression, *Communications in Statistical Simulation* 16, 1209-1227.
- Dietz, E.J., 1989, Teaching regression in a non-parametric statistics course, *American Statistician* 43, 35-40.
- El-Shaarawi, A.H., Niculescu, S.P. 1992, On Kendall's tau as a test of trend in time series data, *Environmetrics* 3(4), 385-411.
- Fong, D.Y.T., Kwan, C.W., Lam, K.F, Lam, K.S.L. 2003. Use of the sign test for the median in the presence of ties, *American Statistician*. 57, 237-240.
- Gehan, E.A. 1965, A generalized Wilcoxon test for comparing arbitrarily singly censored samples: *Biometrika* 52, 203-223.
- Helsel, D. R. 2006, "Fabricating data: How substituting values for nondetects can ruin results, and what can be done about it", *Chemosphere* 65, 2434-9.
- Helsel, D. R. 2005a, "Insider Censoring: Distorsion of Data with Nondetects", *Human and Ecological Risk Assessment*, 11, 1127-37.
- Helsel, D. R. 2005b, *Nondetects and Data Analysis: Statistics for Censored Environmental Data*, 1st Edition, John Wiley and Sons, New Jersey.
- Hirsh, R.M., Alexander, R.B., Smith, R.A., 1991. Selection of methods for the detection and estimation of trends in water quality, *Water Resources Research* 27(5), 803-813.
- Hirsch, R.M., Slack, J.R. 1984, A Nonparametric trend test for seasonal data with serial dependence, *Water Resources Research*. 20(6), 727-732.
- Hirsch, R.M., Slack, J.R., and Smith, R.A. 1982, Techniques of trend analysis for monthly water quality data, *Water Resources Research*. 18(1), 107-121.

Kendall, M.G. 1955, Rank Correlation Methods, Second Edition. Charles Griffin and Company, London. p. 196.

Kutner, Nachtsheim, Neter, Li. 2005, Applied Linear Statistical Models, 5th Edition, McGraw-Hill Irwin, New York

Lee, L., Helsel, D. R. 2005c, "Statistical analysis of water-quality data containing multiple detection limits: S-language software for regression on order statistics", Computers & Geosciences, 31, 1241-1248.

Millard, S.P., Deverel, S.J. 1988, Nonparametric statistical methods for comparing two sites based on data with multiple nondetect limits: Water Resources Research 24, 2087-2098.

Montgomery, D.C., Peck, E.A., Vining, G.G., 2001, Introduction to linear regression analysis, John Wiley and Sons, New Jersey

Peto, R. and Peto, J. 1972, Asymptotically efficient rank invariant test procedures(with discussion). Journal of the Royal Statistical Society, Series A 135, 185-206.

Prentice, R.L., 1978, Linear rank tests with right-censored data. Biometrika 65, 167-179

Prentice, R.L., Marek, P., 1979, A qualitative discrepancy between censored data rank tests. Biometrics 35, 861-867.

Sen, P.K., 1968, Estimates of the regression coefficient based on Kendall's tau. Journal of the American Statistical Association 63, 1379-1389.

Theil, H. 1950, A rank-invariant method of linear and polynomial regression analysis Nederl. Akad. Wetensch, Proceed, 53, 386-392.

Wikipedia contributors, Log-normal distribution, Wikipedia, The Free Encyclopedia, http://en.wikipedia.org/w/index.php?title=Log-normal_distribution&oldid=139133255 (accessed June 24, 2007).