# CEPHaStat 3.2. Analysis of censored data

## 1 Introduction

The CEPHaStat functions have been assembled to make available R functions developed for the CEPHaS project. These include some basic summary statistics and geostatistical methods introduced at the Third CEPHaS Network Meeting in Lusaka, July 2019. In addition there is now a suite of functions for fitting and comparing water release curves, these are explained in more detail in a separate document *Soil Physics Functions in CEPHaStat*.

In this document we show a simple example of one function in the latest version of CEPHaStat (3.2). This is a function to estimate the mean of a variable (normal, or normal after transformation to natural logs), which is censored. A variable is said to be left-censored if all data smaller than some known lower censor value are not measurable and so are replaced with that censor value. A detection limit in analytical chemistry is such a value (although sometimes the censored values are replaced with half the detection limit, such values should be edited to the known lower bound before analysis).

## 2 Example

Most censored data are left-censored (values smaller than a lower bound are replaced with the bound). Some data are left-truncated (which means that we do not know about data smaller than a lower limit, imagine a mesh to capture aggregates, where smaller aggregates simply drop through), which require different treatment. The example here is a right-censored data set.

The thickness of the soil cover over underlying solid rock has been measured at sample sites. This is done with a graduated auger. However, the auger is of fixed length, so if it is inserted to depth 99cm in the soil all we can do is record a depth of 99 cm meaning 99cm or more. These censored values are recorded, along with smaller values, in the file `Thickness.dat`. We want to estimate the mean thickness of the soil, but what do we do with the 99cm values. If we just assume they are correct or, worse, ignore them, then we shall underestimate the mean thickness. The solution, given the known censor value, is to fit a censored distribution to the data by maximum likelihood. In this code we fit a normal distribution, possibly after

transforming the data to natural logarithms.

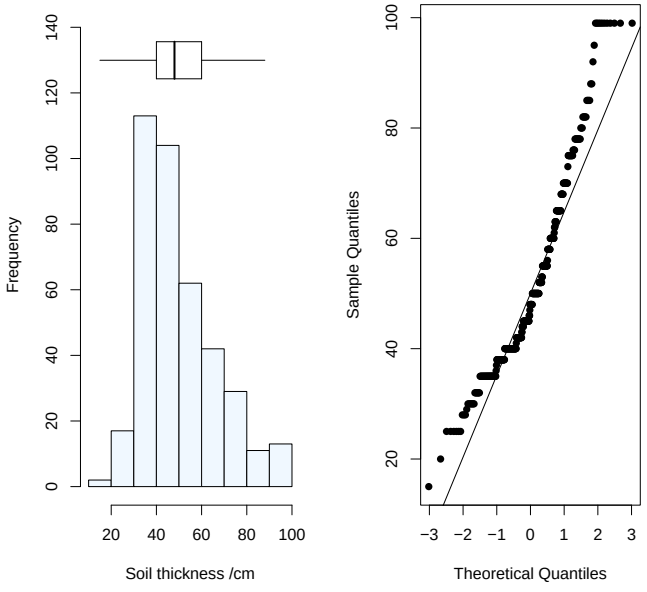
The first step is to set up the `CEPHaStat_3.2` functions with the command: `source("CEPHaStat_3.1.R")`

This will work if the source file is in the working directory. Next we read in the data, and extract the variable of interest. We use the `CEPHaStat` functions `summa` and `summaplot` to produce summary statistics and a plot with a histogram of the data, superimposed box-plot and QQ plots.

```
data.df<-read.table("Thickness.dat",header=F)
y<-as.vector(data.df$V1)
summa(y)
summaplot(y,"Soil thickness /cm")
```

This generates the output below.

```
         Mean Median Quartile.1 Quartile.3 Variance       SD  Skewness
[1,] 51.46056     48         40         60 277.9583 16.67208 0.9244942
     Octile skewness  Kurtosis No. outliers
[1,]            0.35 0.4314808            0
```

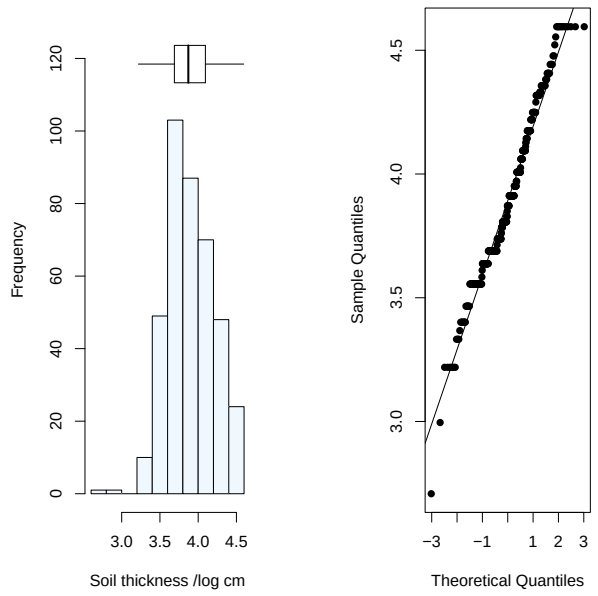



The effect of the censoring is clear on the QQ plot and histogram. Note that the octile skewness exceeds the value of 0.2 which, by a rule of thumb, is the threshold at which we would transform our data. Also the skewness approaches 1.

Examining the histogram, and assuming that the final bin contains censored values which would exceed 99cm, it is clear that there is something of an upper tail to the original distribution. For this reason we consider a transformation to natural logarithms. The transformation, and exploration of the transformed data are below:

```
log.y<-log(y)
summa(log.y)
summaplot(log.y)
```

```
       Mean   Median Quartile.1 Quartile.3   Variance        SD  Skewness
[1,] 3.8918 3.871201   3.688879   4.094345 0.09693999 0.3113519 0.1427507
     Octile skewness   Kurtosis No. outliers
[1,]        0.171142 0.01356065            0
```




Note that the skewness and octiles skewness are much reduced by transformation. Also the QQ plot conforms much more closely to the normal line. For this reason we set the censoring value to the natural log of 99cm

```
censored.value<-log(99)
```

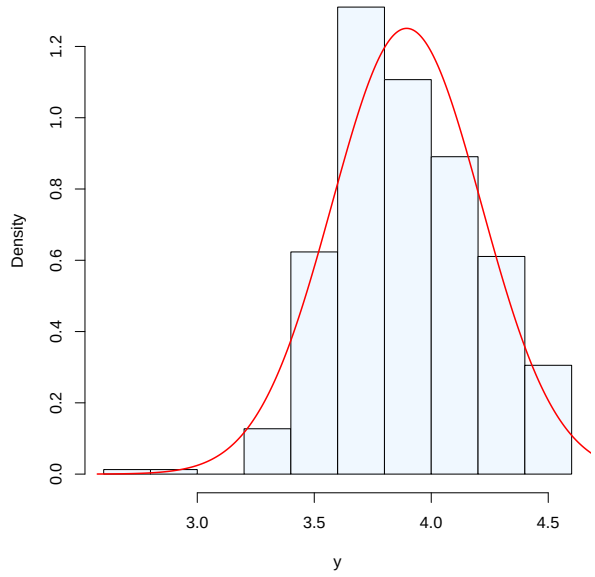
We next call the function `mean_censor`. This has four arguments. The first one is the vector containing the data to which we want to fit the model. This is `log.y` here. The second is the censoring value, `censored.value` here, as defined in

the step above. The third argument is optional, but necessary here. By default the function assumes that the data are left-censored at the censoring value, but here it is right-censored (larger depths are set to 99cm). To tell R this we include the argument `side.left=F`. The fourth argument is also optional, but must be set in this case. By specifying `log.t=T` tell R that the data are on a log-scale. This makes it do some back-transformations after fitting the censored distribution.

The function call and outputs are as below:

```
op<-mean_censor(log.y,censored.value,side.left=F,log.t=T)
print(op)
```

```
     log_likelihood Number_censored    Mean        SD    SE_mean Lower_95%
[1,]      -121.7005              11 3.89491 0.3189742 0.01611454  3.863228
     Upper_95% Mean_uncensored_only Back_mean Back_median Back_Lower_95%
[1,]  3.926591             3.871547  57.65037    49.15161       47.61881
     Back_Upper_95%
[1,]       50.73375
```




The first two outputs are the maximized likelihood for the fitted model, and the number of censored values. Next are presented the estimated mean and standard

deviation of the censored normal distribution fitted to the observations. After these are given the standard error of the mean, and its lower and upper confidence bounds (95%). The next value is the mean of the uncensored data only, the back-transformed mean (original units), the median-unbiased backtransform, and its lower and upper confidence bounds (95%).

As well as this set of outputs the function plots a histogram of the data (density scale) with the probability density function for the fitted normal distribution superimposed.