Comparison of the Kaplan-Meier, Maximum Likelihood, and ROS Estimators for Left-Censored Data Using Simulation Studies

Samuel Young Annan
Piaomu Liu
Yuan Zhang

December 6, 2009

[1]

[1]

Abstract

This Paper seeks to compare Parametric (MLE), Semi-Parametric (ROS) and non-parametric (K-M) estimators for left censored data which are common occurrences in environmental Science. The comparison is done using simulation studies.

1 Introduction

Analyzing left-censored data has significant application to environmental sciences and relevant industries where data from research may be incomplete due to limitation of tools, methodologies for measurements or inability to observe data. Left-censored data are similar to survival data in that data are not known beyond a certain limit. In environmental sciences and chemistry, the numeric value is called the "detection limit" which is chosen based on the techniques or equipments used in the study. Available statistical methods that are used to analyze left-censored data include the Kaplan-Meier estimator, Maximum Likelihood estimator, and the "Robust Regression on Ordered Statistics (ROS)" method. These three methods are non-parametric, parametric and semi-parametric, respectively. The "NADA" package in R provides the most updated written functions that include all of the three statistical methods to analyze left-censored data.

In our study, we intend to investigate the strengths of the three methods and thus, find a method recommendation for analyzing left-censored data using simulation. To compare the methods we calculate the biases and variances of estimators they produce for different levels of censoring and sample sizes of simulated data. By varying sample sizes and detection limit, we will find the "elastic limits" of the various methods and get a better idea of how these variables affect the effectiveness of the methods to calculate statistics. We will run 1000 simulations for different pairs of sample sizes of 25, 35, 60 and 80 and censoring levels of 15%, 25%, 60% and 80%.

We simulate left-censored data under the assumption that the underlying distribution is exponential. Other types of distributions that are usually associated with left-censored data are normal, lognormal and gamma. Since all of the distributions belong to the exponential family, we use the exponential distribution and expect to see similar conclusions about the performance of the different methods.

2 Description of Methods

2.1 Kaplan-Meier

The nonparametric Kaplan-Meier (K-M) method has always been considered as a standard method for estimating summary statistics of censored survival data. It is however overlooked in some other settings where censored data occurs. One such setting is in the environmental sciences; the field from which this study is concerned with. K-M is sensitive to sample size, and level of censoring. Let S(t) be the probability that an item from a given population will have a lifetime exceeding time t. For a sample from this population of size N let the observed times until death of n sample members be

$$t_1 \leq t_2 \leq t_3 \leq \cdots \leq t_N.$$

Corresponding to each t_i, n_i is the number "at risk" just prior to time t_i and d_i is the number of deaths at time t_i.

Then KM estimator is a product of the form

$$\hat{S}(t) = \prod_{t_i < t} \frac{n_i - d_i}{n_i}. \tag{1}$$

When there is no censoring, n_i is just the number of survivors just prior to time t_i. With censoring, n_i is the number of survivors less the number of censored data. It is only those surviving cases that are still being observed (have not yet been censored) that are "at risk" of an (observed) death.

The mean of K-M estimator is evaluated as the area under the K-M survival curve. The median is the value that corresponds to the 50th percentile on the K-M survival curve. And we should note that when more than 50% of the data are censored, the median cannot be estimated using K-M. We can use a method which assumes some kind of model for the data distribution. Two possible methods for doing this include the MLE (parametric) and ROS method (semi-parametric). Fig.1 on the next page is an example of the Kaplan-Meier survival curve.

2.2 Maximum Likelihood Estimator

The parametric Maximum Likelihood Estimator (MLE) assumes a distribution that will closely fit the observed data. The MLE computes the mean and standard deviation for the assumed distribution using the observed detected values, and the observed proportions of data below one or more censoring thresholds. Some literature on this subject indicate that for data set of at least 50 observations, and where either the percent censoring is reasonable (to the extent that the distributional shape can be evaluated) or the distribution can be assumed from knowledge outside the data set, MLE methods are the method of choice because its efficiency (pg 13). For data sets with less than 25 to 50 observations, MLE has been shown to perform poorly.For censored data, the likelihood function L is given by

$$L = \prod p(x_i)^{\delta_i} * F(x_i)^{1-\delta_i} \tag{2}$$

$$p(x) = \frac{\exp\left(-\frac{(x-\frac{\mu}{\sigma})^2}{2}\right)}{\sigma * \sqrt{2\pi}} \tag{3}$$

$$F(x) = \varphi\left(\frac{x-\mu}{\sigma}\right) \tag{4}$$

$$\varphi(y) = \left(\frac{1}{\sqrt{2\pi}}\right) * \int_0^y \exp\left(-\frac{\mu^2}{2}\right)\, d\mu \tag{5}$$

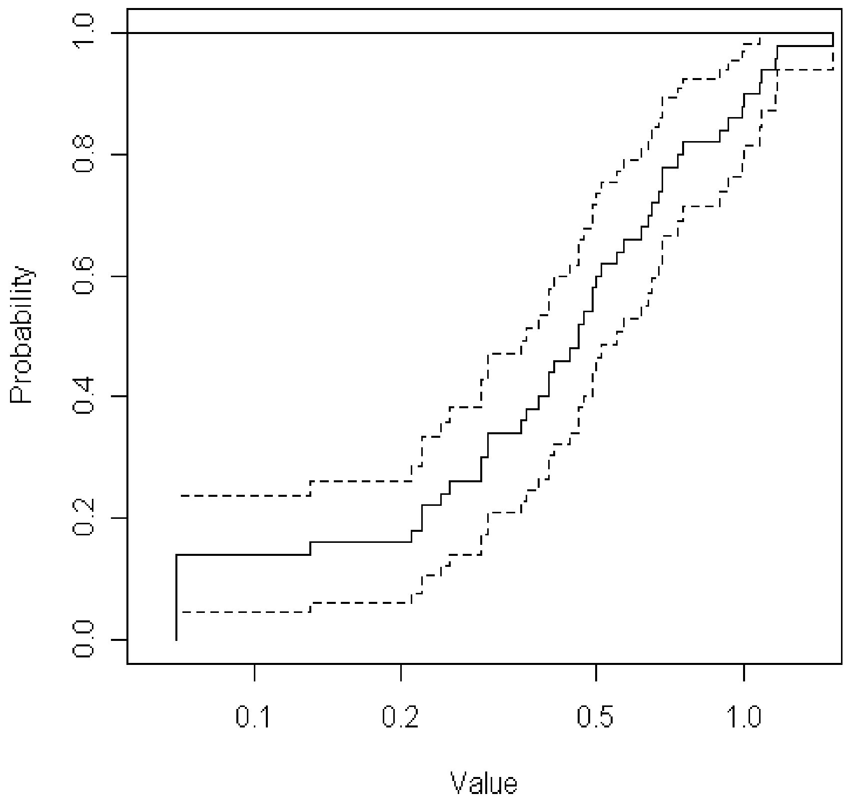

Fig.1: Graph of typical Kaplan-Meier survival function

2.3 ROS: Regression on Order Statistcs

ROS calculates summary statistics with a regression equation on a probability plot, and is called "regression on order statistics". We used a robust approach to ROS. Unobserved values are estimated from a regression equation obtained by using observed data. The regression equation is obtained by fitting observed values to the probability plot, and the explanatory variable in the regression is the normal scores of observed values. Hence, the ROS uses exponentiated (if y is in log units) predicted values of unobserved data as well as observed data to compute summary statistics. Specific formulae are presented below: Calculate the probability of exceeding the jth detection limit:

$$pe_i = pe_{j+1} + \frac{A_j}{A_j + B_j}\left[1 - pe_{j+1}\right] \tag{6}$$

A_j = the number of observations detected between the jth and (j+1)th detection limits, and B_j= the number of observations, censored and uncensored below the jth detection limit. When j = the highest detection limit, pe_{j+1}= 0, and $A_j + B_j = n$. The number of nondetects below the jth detection limit is defined as C_j:

$$C_j = B_j - B_{j-1} - A_{j-1} \tag{7}$$

Calculating the plotting position:

- For observed values

  for $i = 1$ to A_j

$$pd_i = (1 - pe_j) + \left(\frac{i}{A_j + 1}\right) * (pe_j - pe_{j+1}) \tag{8}$$

- For censored observations
  for $i = 1$ to C_j

$$pc_i = \left(\frac{i}{C_j + 1}\right) * (1 - pe_j) \tag{9}$$

  for $i = 1$ to A_j

$$pd_i = (1 - pe_j) + \left(\frac{i}{A_j + 1}\right) * (pe_j - pe_{j+1}) \tag{10}$$

  for $i = 1$ to C_j

$$pc_i = \left(\frac{i}{C_j + 1}\right) * (1 - pe_j) \tag{11}$$

  The regression equation for predicting unobserved data:

  Predicted log-value = $\beta + \alpha *$ normal scores of plotting positions

3 Methodology

We simulate left censored data on which the various estimation methods will be applied. The NADA package us to do the various estimations in R. The detection limit is, for a censoring level of ‘q’, the q-th percentile of the exponential distribution. The censored data will be made up of values that are the maximum of the observed values and the detection limit. The status of each data point will be defined as ‘censored’ if the observed value is less than the detection limit and ‘uncensored’ otherwise. For each run of the simulation we generate 1000 samples of size ‘n’ with censoring level ‘q’ and calculate biases as well as variances of means and medians given by each method. The same statistics will be computed for the uncensored simulation data using the definitions of sample mean and median which gives us the best estimate we could possibly obtain from simulation. This is useful to compare the estimated values from the methods with the directly calculated values from the simulated samples. We also know the theoretical mean and median from the assumed distribution. In this case our parameter has value 0.2. We expect the mean from the simulated samples to be close to 5 and the median to be close to 3.466. In order to verify if the actual simulated data has a censoring level close to the stated level, the proportion of censored data is also calculated to get an idea of the level of accuracy. The results from the simulation study are summarized in tables 1 and 2 below.

Censoring level	Sample size	Bias of the mean				Bias of the median			
		Simulated sample	KM	ROS	MLE	Simulated sample	KM	ROS	MLE
0.15	80	0.04352555	0.118948	0.090244	0.611581	0.020385	-0.04314	0.020385	-0.39398
	60	0.01825956	0.096818	0.06492	0.593373	0.077242	-0.00971	0.077242	-0.40562
	35	-0.00976954	0.079197	0.035074	0.566708	0.07749	0.07749	0.07749	-0.39437
	25	0.007603281	0.106485	0.051077	0.607625	0.118766	0.118766	0.118766	-0.37407
0.25	80	-0.01023404	0.198649	0.085939	0.391705	0.035265	-0.02637	0.035265	-0.2834
	60	-0.01321944	0.203647	0.082382	0.398177	0.028767	-0.05423	0.028767	-0.2877
	35	0.000725052	0.240941	0.091826	0.431727	0.052917	NA	0.053722	-0.27301
	25	0.005196799	0.267574	0.09567	0.461761	0.066219	NA	0.067614	-0.26939
0.5	80	-0.01267204	1.017127	0.263381	0.311009	0.009126	NA	0.173855	0.078281
	60	-0.00644805	1.046416	0.268602	0.326154	0.020316	NA	0.190828	0.093336
	35	0.02161498	1.12765	0.284693	0.374557	0.102568	NA	0.25471	0.136096
	25	0.02784621	1.193839	0.262004	0.385578	0.101785	NA	0.206278	0.133262
0.8	80	-0.00540826	4.319457	0.693626	0.74264	0.008792	NA	0.869537	0.914195
	60	0.03098471	4.423823	0.706137	0.797189	0.071652	NA	0.886196	0.991675
	35	-0.002761	4.766759	0.741728	0.81552	0.080919	NA	0.921654	1.022509
	25	N.A	N.A	N.A	N.A	N.A	N.A	N.A	N.A

Table1: Bias of the mean or median for different censoring level and sample size

Censoring level	Sample size	Variance of the mean				Variance of the median			
		Simulated sample	KM	ROS	MLE	Simulated sample	KM	ROS	MLE
0.15	80	0.272684	0.267754	0.272462	0.391551	0.229516	0.225803	0.229516	0.152
	60	0.404776	0.394351	0.402707	0.614019	0.41623	0.41109	0.41623	0.205763
	35	0.748142	0.727375	0.742451	1.10121	0.745567	0.745567	0.745567	0.391229
	25	1.020904	1.003718	1.022511	1.617684	1.086053	1.086053	1.086053	0.500759
0.25	80	0.354705	0.334682	0.35545	0.479042	0.322828	0.321164	0.322828	0.172148
	60	0.419803	0.392543	0.41715	0.561902	0.417925	0.409372	0.417925	0.215938
	35	0.678919	0.643951	0.680851	0.954442	0.720446	NA	0.717526	0.360336
	25	0.967907	0.906756	0.966376	1.36333	0.983524	NA	0.978131	0.506677
0.5	80	0.293511	0.217678	0.317824	0.332695	0.315826	NA	0.236933	0.205015
	60	0.424414	0.318604	0.43261	0.47151	0.379728	NA	0.288327	0.263784
	35	0.755739	0.592384	0.808218	0.850812	0.806969	NA	0.706813	0.511067
	25	0.95808	0.779739	1.035281	1.097327	0.933569	NA	0.943476	0.625179
0.8	80	0.296524	0.200061	0.735755	0.450155	0.297984	NA	1.030087	0.626801
	60	0.441378	0.315531	1.040165	0.688904	0.460489	NA	1.469383	0.947061
	35	0.73405	1.168076	2.073139	1.112623	0.716128	NA	2.93816	1.530752
	25	N.A	N.A	N.A	N.A	N.A	N.A	N.A	N.A

Table2: Variance of the mean or median for different censoring level and sample size

References

[1]

APPENDIX

– <u>R codes for our simulation</u>

```
>library(NADA)
>sim=function(s,n,q)
>{
>obs<-matrix(rexp(n*s,0.2),nrow=s)
>dl<-qexp(q,0.2)
>cens<-pmax(obs,dl)
>status<-matrix(as.logical(obs<cens),nrow=s)
>themean<-5
>themedian<-log(2)*5
>sample.mean=rep(0,s)
>sample.median=rep(0,s)
>mle.mean=rep(0,s)
>mle.median=rep(0,s)
>km.median<-rep(0,s)
>km.mean<-rep(0,s)
>ros.median<-rep(0,s)
>ros.mean<-rep(0,s)
>perc<-rep(0,s)
>for (i in 1:s) {perc[i]<-sum(status[i,])/n}
>
>for (i in 1:s){mle.median[i]<-median(cenmle(cens[i,],status[i,]))
>               mle.mean[i]<-mean(cenmle(cens[i,],status[i,]))
>}
>
>for (i in 1:s) {km.median[i]<-median(cenfit(cens[i,],status[i,]))
>               km.mean[i]<-mean(cenfit(cens[i,],status[i,]))
>}
>
>for (i in 1:s){
>               ros.median[i]<-median(ros(cens[i,],status[i,]))
>               ros.mean[i]<-mean(ros(obs=cens[i,],censored=status[i,]))
>}
>
>for (i in 1:s){sample.median[i]<-median(obs[i,])
>               sample.mean[i]<-mean(obs[i,])
>}
>
>bias.s.mean<-mean(sample.mean)-themean
```

```
>
>bias.s.median<-mean(sample.median)-themedian
>
>bias.mle<-mean(mle.mean)-themean
>
>mbias.mle<-mean(mle.median)-themedian
>
>bias.km<-mean(km.mean)-themean
>
>mbias.km<-mean(km.median)-themedian
>
>bias.ros<-mean(ros.mean)-themean
>
>mbias.ros<-mean(ros.median)-themedian
>
>bias<-data.frame(sample.bias =bias.s.mean,
>sample.bias.median=bias.s.median,
>bias.mle=bias.mle,bias.km=bias.km,bias.ros=bias.ros,
>med.bias.mle=mbias.mle, med.bias.km=mbias.km,
>med.bias.ros=mbias.ros)
>estimators<-data.frame(s.mean=sample.mean[1:5],s.med=sample.median[1:5],
>
>mle.mean=mle.mean[1:5],mle.med=mle.median[1:5],km.mean=km.mean[1:5],
>
>km.med=km.median[1:5],ros.mean=ros.mean[1:5],ros.med=ros.median[1:5],
>percentage=perc[1:5])
>variance<-data.frame(var.s.mean=var(sample.mean),
>var.s.median=var(sample.median),
>var.km.mean=var(km.mean),var.km.median=var(km.median),
>var.mle.mean=var(mle.mean),
>var.mle.median=var(mle.median), var.ros.mean=var(ros.mean),
>var.ros.median=var(ros.median))
>list(estimators,bias,variance,censoring=sum(status/(n*s)))
>
>}
```

sim(1000,80,0.5)
sim(1000,80,0.25)
sim(1000,80,0.15)
sim(1000,60,0.8)
sim(1000,60,0.5)
sim(1000,60,0.25)
sim(1000,60,0.15)
sim(1000,35,0.8)
sim(1000,35,0.5)

sim(1000,35,0.25)
sim(1000,35,0.15)
sim(1000,25,0.8)
sim(1000,25,0.5)
sim(1000,25,0.25)
sim(1000,25,0.15)

– **Example of R output**

```
> set.seed(3)
> sim(1000, 80, 0.8)
[[1]]
    s.mean    s.med mle.mean  mle.med  km.mean km.med ros.mean  ros.med
1 5.426974 3.686756 5.866621 4.263324 9.396919     NA 5.475861 3.725668
2 3.814150 2.356388 4.977981 3.831457 9.305869     NA 5.054383 3.940800
3 4.999852 3.980329 5.665971 4.617723 9.389451     NA 6.133401 5.249726
4 4.854264 3.429828 5.114879 3.711066 8.933290     NA 5.838407 4.723209
5 4.402331 3.145441 4.360335 2.978963 9.316492     NA 4.616463 3.340823
  percentage
1     0.7875
2     0.8500
3     0.8125
4     0.8375
5     0.8750

[[2]]
   sample.bias sample.bias.median  bias.mle  bias.km  bias.ros med.bias.mle
1 -0.005408258        0.008792226 0.7426402 4.319457 0.6936256    0.9141953
  med.bias.km med.bias.ros
1          NA     0.869537

[[3]]
  var.s.mean var.s.median var.km.mean var.km.median var.mle.mean var.mle.median
1   0.296524    0.2979843   0.2000610            NA    0.4501553      0.6268006
  var.ros.mean var.ros.median
1    0.7357545       1.030087

censoring
[1] 0.7995
```