Analyzing Right - Censored Data with MLE Techniques

Dustin Dickerson

Department of Mathematical Sciences
Montana State University

May 7, 2010

A writing project submitted in partial fulfillment of the requirements for the degree

Master of Science in Statistics
APPROVAL

of a writing project submitted by

Dustin Dickerson

This writing project has been read by the writing project advisor and has been found to be satisfactory regarding content, English usage, format, citations, bibliographic style, and consistency, and is ready for submission to the Statistics Faculty.

Date  Dr. John Borkowski
       Writing Project Advisor

Date  Mark C. Greenwood
       Writing Project Coordinator
Introduction

Almost every major company invests millions of dollars in product reliability each year. This research is used to evaluate risks and liabilities, establish warranties, evaluate replacement policies, assess design changes, and compare different vendors, materials, manufacturing practices, etc. Often, these findings are the result of the analysis of survival data from a relatively small number of units. In an ideal situation, the quality engineer would have complete data from each individual unit, that is each unit would fail in the desired way within the study time. The engineer would then be able to input the data into a statistical package and have it fit a variety of possible distributions (Exponential, Weibull, LogNormal, etc.) using standard maximum likelihood methods. Then he or she could evaluate the output and determine which distribution offers the best fit. However, the process is rarely this simple.

Censoring

Censoring occurs when the exact failure time of a certain item is unknown. There are two main types of censoring:

1. Right Censoring. When a unit’s failure time is only known to exceed some value, it is said to be right censored. For example, reliability experiments only last for a finite amount of time and if a product has not failed by the end of the study time, it is right censored since its actual failure time is only known to be greater than the study time. Right censoring is the most common form of censoring, and is usually the result of limited resources or competing failure modes.

2. Left Censoring. In some situations, one knows only whether a unit failed after it was inspected once, revealing for instance a cracked covering or leaking hose. The unit may have failed in a engineering sense at one time but may not have been noticed until further deterioration caused an inspection. In this case, one only knows that the failure occurred sometime prior to the inspection.

There are numerous combinations and special cases of left and right censoring for different situations. For example, in interval censoring, items are censored from the left and the right. The exact failure time is still unknown, but the researcher knows that it is greater than one time and less than another. My report focuses exclusively on right, singly censored data. The presence of right censored data complicates survival analysis, but it does not make it impossible.

Maximum Likelihood Estimation with Censored Data

Traditional MLE procedures estimate parameter values by using calculus to determine what values make the observed data most probable. This is achieved by
differentiating the likelihood function (1) and finding the critical values that correspond to a maximum.

\[ L(\theta, X) = \prod_{i=1}^{n} f(x_i; \theta) \]  \hspace{1cm} (1)

Here, \( f \) represents the probability density function of a random variable, \( x_i \), representing failure times and \( \theta \) represents the parameter(s) associated with that distribution. However, in many reliability experiments, the probability distribution is unknown and computer software is used to compare the maximum likelihood estimates of several distributions.

This procedure is further complicated when dealing with right censored data because not all of the failure times are known. This requires a modification of the likelihood function taking into account the censoring:

\[ L(\theta, X) = \prod_{i=1}^{n} f(x_i; \theta)^{\delta_i} [1 - F(x_i; \theta)]^{1-\delta_i} \]

\[ \delta_i = \begin{cases} 
1 & \text{if } x_i \text{ is censored} \\
0 & \text{if } x_i \text{ is not censored} 
\end{cases} \]

**An Example Using Software**

Calculating the maximum likelihood estimator is only half the battle; the quality engineer must still decide which family of distributions produces the best estimate. This translates into doing multiple MLE calculations and then comparing goodness-of-fit statistics. Thankfully, statistical software such as SAS, R, and Minitab can do these calculations in a matter of seconds and the engineer can concentrate on interpreting the output. Let’s take a look at an example:

**Example.** Consider the censored data resulting from a reliability experiment on a small appliance component (Nelson, 1983). What is recorded below is the number of cycles each unit completed before it failed. Values marked with a + sign represent censored values.

<table>
<thead>
<tr>
<th>Cycles to Failure</th>
</tr>
</thead>
<tbody>
<tr>
<td>45+ 281+ 508+ 608+ 1164+</td>
</tr>
<tr>
<td>47   311  508+ 630  1164+</td>
</tr>
<tr>
<td>73   417+ 608  670  1164+</td>
</tr>
<tr>
<td>136+ 485+ 508+ 670  1164+</td>
</tr>
<tr>
<td>136+ 485+ 508+ 731+ 1198+</td>
</tr>
<tr>
<td>136+ 490  608  838  1198</td>
</tr>
<tr>
<td>136+ 569+ 508+ 964  1300+</td>
</tr>
<tr>
<td>136+ 571+ 508+ 964  1300+</td>
</tr>
<tr>
<td>145  571  508+ 1164+ 1300+</td>
</tr>
<tr>
<td>190+ 575  608  1164+</td>
</tr>
<tr>
<td>190+ 608+ 508+ 1164+</td>
</tr>
</tbody>
</table>
Analysis in SAS

When reading the data into SAS, instead of using plus signs, 1’s were used for censored values and 0’s were used for uncensored values (this is the default). SAS has two procedures, LIFETEST and RELIABILITY to calculate survival statistics. PROC LIFETEST is used to construct the empirical survival (i.e. reliability) function, using the Kaplin-Meyer Method. A plot of the survival function for the fan data is given below:

Both PROC LIFETEST and PROC RELIABILITY can generate probability plots, however, the plots from PROC RELIABILITY are easier to read and interpret. RELIABILITY can also output summary statistics for specific fitted distributions. For instance, the output below and on the next page shows what SAS would calculate if the engineer decided to fit an exponential distribution to the appliance component data.
Analysis in R

This same data was analyzed in R using the survival package. One key difference at the very beginning, is that unlike SAS, R defaults to using a 0 for representing censored values. Using the Surv and survfit functions, the Kaplin-Meyer plot on the next page can be created. Notice that the default with R is to include 95% confidence bands (dashed lines) and tabular output can be obtained by doing summary on the survfit object (one could also include confidence bands in SAS, though it is not the default).
Fitting different distributions to the data is a little more complicated in R but not impossible. Through the use of the `survreg` function, one can output an exponential fit nearly identical to SAS. Unfortunately, R does not output the quantile information automatically. However, these calculations can easily be written into a function.

Call:
```
survreg(formula = Surv(fans$time, event = fans$censor) ~ 1, weights = fans$freq,
        dist = "exponential")
```

Percent Estimate Lower 95% Upper 95%
--- ------- ------- ------- -------
 0.1 2.160330 1.323475 3.526342
 0.2 4.322824 2.648276 7.056216
 0.5 10.823331 6.630658 17.667099
 1.0 21.701188 13.294720 35.423201
 2.0 43.622696 26.724415 71.206034

Exponential distribution
Loglik(model) = -138.8  Loglik(intercept only) = -138.8
Number of Newton-Raphson Iterations: 5
n = 26
5.0  110.755046  67.851464  180.787258
10.0  227.499693  139.372317  371.351440
20.0  481.822713  295.177311  786.487030
30.0  770.150373  471.814446  1257.128945
40.0  1103.000228  675.727053  1800.445163
50.0  1496.678050  916.904477  2443.051856
60.0  1978.500763  1212.081788  3229.538886
70.0  2599.678278  1592.631529  4243.497019
80.0  3475.178812  2128.986265  5672.590743
90.0  4971.856862  3045.890742  8123.316363
95.0  6468.534912  3959.792319  10558.694456
99.0  9943.713724  6091.781482  16231.285199
99.9 14915.570586  9137.672225  24346.927798

Notice that the parameter estimate labeled “Intercept” is 7.68. By default, R is fitting an extreme value distribution. Thus, to obtain the exponential parameter that SAS outputs, the researcher must take $\frac{1}{e^{\text{Intercept}}}$ = 2159.24. This is then the value that was used to obtain the percentile estimates, not 7.68.

Analysis in Minitab

Unlike SAS and R, Minitab is less of a traditional programming language and more of a point-and-click spreadsheet interface roughly similar to Microsoft Excel. Once the data was inserted into Minitab’s spreadsheet, analysis was done using the Reliability/Survival menu under the Stat tab (see below). This menu then provided the necessary options to perform MLE calculations on the censored data.

After specifying which column represented the indicator for censoring (Minitab uses the same designation as SAS 1 for censored and 0 for uncensored), selecting MLE methods, and requesting probability plots and distribution estimates for the output, the following charts and graphs were generated.
Characteristics of Distribution

<table>
<thead>
<tr>
<th></th>
<th>Standard</th>
<th>95.0% Normal CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean (MTTF)</td>
<td>2159.25</td>
<td>539.812</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1322.83 3524.55</td>
</tr>
<tr>
<td>Standard Deviation</td>
<td>2159.25</td>
<td>539.812</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1322.83 3524.55</td>
</tr>
</tbody>
</table>

Table of Percentiles

<table>
<thead>
<tr>
<th>Percent</th>
<th>Percentile</th>
<th>Standard</th>
<th>95.0% Normal CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>21.7012</td>
<td>5.42530</td>
<td>13.2948 35.4229</td>
</tr>
<tr>
<td>2</td>
<td>43.6227</td>
<td>10.9057</td>
<td>26.7247 71.2054</td>
</tr>
<tr>
<td>3</td>
<td>65.7690</td>
<td>16.4423</td>
<td>40.2922 107.355</td>
</tr>
<tr>
<td>4</td>
<td>88.1449</td>
<td>22.0362</td>
<td>54.0004 143.879</td>
</tr>
<tr>
<td>5</td>
<td>110.755</td>
<td>27.6888</td>
<td>67.8521 180.786</td>
</tr>
<tr>
<td>6</td>
<td>133.604</td>
<td>33.4011</td>
<td>81.8504 218.083</td>
</tr>
<tr>
<td>7</td>
<td>156.698</td>
<td>39.1746</td>
<td>95.9984 255.779</td>
</tr>
<tr>
<td>8</td>
<td>180.042</td>
<td>45.0104</td>
<td>110.299 293.882</td>
</tr>
<tr>
<td>9</td>
<td>203.640</td>
<td>50.9101</td>
<td>124.757 332.402</td>
</tr>
<tr>
<td>10</td>
<td>227.500</td>
<td>56.8749</td>
<td>139.374 371.348</td>
</tr>
<tr>
<td>20</td>
<td>481.823</td>
<td>120.456</td>
<td>295.180 786.480</td>
</tr>
<tr>
<td>30</td>
<td>770.150</td>
<td>192.538</td>
<td>471.819 1257.12</td>
</tr>
<tr>
<td>40</td>
<td>1103.00</td>
<td>275.750</td>
<td>675.733 1800.43</td>
</tr>
<tr>
<td>50</td>
<td>1496.68</td>
<td>374.170</td>
<td>916.913 2443.03</td>
</tr>
<tr>
<td>60</td>
<td>1978.50</td>
<td>494.625</td>
<td>1212.09 3229.51</td>
</tr>
<tr>
<td>70</td>
<td>2599.68</td>
<td>649.920</td>
<td>1592.65 4243.46</td>
</tr>
<tr>
<td>80</td>
<td>3475.18</td>
<td>868.795</td>
<td>2129.01 5672.54</td>
</tr>
<tr>
<td>90</td>
<td>4971.86</td>
<td>1242.96</td>
<td>3045.92 8115.57</td>
</tr>
</tbody>
</table>
91 5199.36 1299.84 3185.29 8486.92
92 5453.68 1363.42 3341.10 8902.05
93 5742.01 1435.50 3517.74 9372.69
94 6074.86 1518.71 3721.65 9916.00
95 6468.53 1617.13 3962.83 10558.6
96 6950.36 1737.59 4258.01 11345.1
97 7571.54 1892.88 4638.56 12359.0
98 8447.04 2111.76 5174.92 13788.1
99 9943.71 2485.93 6091.84 16231.1

**Conclusion**

SAS, R, and Minitab are all capable of conducting survival analysis with right censored data. From the user’s point of view, Minitab was the easiest to use simply because of its point-and-click environment. (I did not experiment with R commander or other R plug-ins). Whichever package the researcher decides to use, they must be aware of some of the differences among the packages. Some of the differences I notices are listed below.

- Each package has a limited number of distributions available to fit to the data. SAS has the most distributions available (9), while R and Minitab have 7 and 6 available, respectively.

- Minitab’s options allow the user to specify the value of the indicator that represents a censored value. SAS defaults to having a 1 represented censored value and R defaults to having a 0 represent a censored value.

- At least in the case of the exponential distribution, R fits a form of the extreme value distribution which requires a transformation of $\frac{1}{e^{\theta}}$ in order to get the results on the same scale as SAS and Minitab.

**References**


Appendix

R code

```r
require(survival)  #The survival package has all of the functions I need
fans <- read.csv(file.choose(), head=T)  #Reads in the data
fans$censor <- abs(fans$censor.num -1 )  #I think R counts a 0 as a censored value
#and 1 as a regular failure. SAS does the
#opposite

x <- is.Surv(x)

# This is the code to do a Kaplin-Meyer type plot #
fit <- survfit(Surv(fans$time, fans$censor)~1, weights=fans$freq)
plot(fit, xlab="Cycles to Failure", ylab="Survival Probability",
     main="Empirical Survival Function")
summary(fit)

# Making probability plots to judge the fit of each distribution to the data #
par(mfrow=c(2,2))
require(qAnalyst)  #Need this to make the probability plots
probplot(fans$time, "exponential", confintervals=TRUE, confidence=0.95)
probplot(fans$time, "weibull", confintervals=TRUE, confidence=0.95)
probplot(fans$time, "lognormal", confintervals=TRUE, confidence=0.95)
    #It would be nice to make the red points larger and to figure out the
    #95% bands because I don’t think they match the SAS output

 survreg(Surv(fans$time, event=fans$censor) ~ fans$censor, dis="weibull")
survreg(Surv(fans$time, event=fans$censor) ~ fans$censor, dis="lognormal")
exp.reg <- survreg(Surv(fans$time, event=fans$censor) ~ 1, dis="exponential",
                   weights=fans$freq)
summary(exp.reg)
exp.reg$coeff

percentile <- c(0.1,0.2,0.5,1,2,5,10,20,30,40,50,60,70,80,90,95,99,99.9)
#Vector of percentiles

## Percentile Estimates Function ##
Estimates <- function(percentiles, theta, sd){
estimates <- matrix(data=NA, nrow=18, ncol=4)
estimates[,1] <- percentiles
estimates[,2] <- qexp(percentile/100, 1/exp(theta))
estimates[,3] <- qexp(percentile/100, 1/exp(theta-1.96*sd))
estimates[,4] <- qexp(percentile/100, 1/exp(theta+1.96*sd))
colnames(estimates) <- c("Percent", "Estimate", "Lower 95%", "Upper 95%")
return(estimates)
}
```
SAS Code

*************************************************;
***** Multiply censor data example 1 ************;
***** from Nelson handout (page 13) *************;
*************************************************;
DM'LOG;CLEAR;OUT;CLEAR;';
OPTIONS LS=74 PS=72 NONUMBER NODATE;

Data ex1;
INPUT cycles censor n @@;
LABEL cycles = 'NUMBER OF CYCLES TO FAILURE'; CARDS;
45 1 1 47 0 1 73 0 1 136 1 5 145 0 1
190 1 2 281 1 1 311 0 1 417 1 1 485 1 2
490 0 1 569 1 1 571 1 1 571 0 1 575 0 1
608 1 12 608 0 2 630 0 1 670 0 2 731 1 1
838 0 1 964 0 2 1164 1 7 1198 1 1 1198 0 1
1300 1 3
;
PROC LIFETEST DATA = ex1 PLOTS=(LS, LLS, S) OUTSURV=survive GRAPHICS;
* These inputs above are in a sense making;
* a normal probability or QQ plot for;
* the exponential and Weibull distns;
TITLE F=SWISSBh=.6 'RELIABILITY ANALYSIS: Ex 1';
TIME cycles*censor(1); *value for censored data indicator = 1;
FREQ n;
SYMBOL1 H=1 V=DOT W=2;
NOTE F=SWISSB H=.35 CM MOVE=(18,65)PCT 'LIFETIMES OF'
NOTE F=SWISSB H=.35 CM MOVE=(18,63)PCT 'APPLIANCE COMPONENTS'
NOTE F=SWISSB H=.35 CM MOVE=(18,61)PCT 'PRODUCT-MOMENT METHOD'
PROC PRINT DATA=survive;
RUN;

PROC RELIABILITY DATA=ex1;
DISTRIBUTION EXPONENTIAL;
PROBPLOT cycles*censor(1) / WAXIS=2 WFIT=2 FONT=SWISSB;
FREQ n;
SYMBOL1 H=1.5 V=CIRCLE W=2;
NOTE F=SWISSB H=.43 CM MOVE=(58,24)PCT 'LIFETIMES OF'
NOTE F=SWISSB H=.43 CM MOVE=(58,22)PCT 'APPLIANCE'
NOTE F=SWISSB H=.43 CM MOVE=(58,20)PCT 'COMPONENTS'
TITLE2 F=SWISSB H=0.5 CM 'FITTING AN EXPONENTIAL DISTRIBUTION'
* Title2 is like a sub-title;
RUN;