ISyE 6404 (EP-2): K-M Estimation, Kernel Regression and Spline

Yuan Gao, Kevin Lee, Akshay Govindaraj
Yijun (Emma) Wan, Peter Williams, Ruixuan Zhang
ygao390, kylee20, ywan40, agovindaraj6, pwilliams60, rzhang438 | @gatech.edu

2018-10-22

Contents

1. K-M Estimation (25%):

Locate a data set with right-censoring (in Type-I Censoring) in the field of your interest, e.g., eCommerce, medical study, drug development, supply-chain/logistics operations, for applying the K-M Estimator to estimate the survival function with pointwise confidence intervals.

For this exercise, we located a dataset, that consists of measures on 69 different patients who received a heart transplant, taken from the first edition of the text *The Statistical Analysis of Failure Time Data* by Kalbfleisch and Prentice, Appendix I (230-232), published originally in 1980, which can also be found via the following link on the Carnegie Mellon statistics site: http://lib.stat.cmu.edu/datasets/stanford, and has three columns with the following measures:

- Age: Age of patient in years at the time of heart transplant in years
- Status: Survival status (1=dead, 0=alive)
- Days: Survival time in days after transplant (in days)

Table 1: Preview: Heart Transplant Data

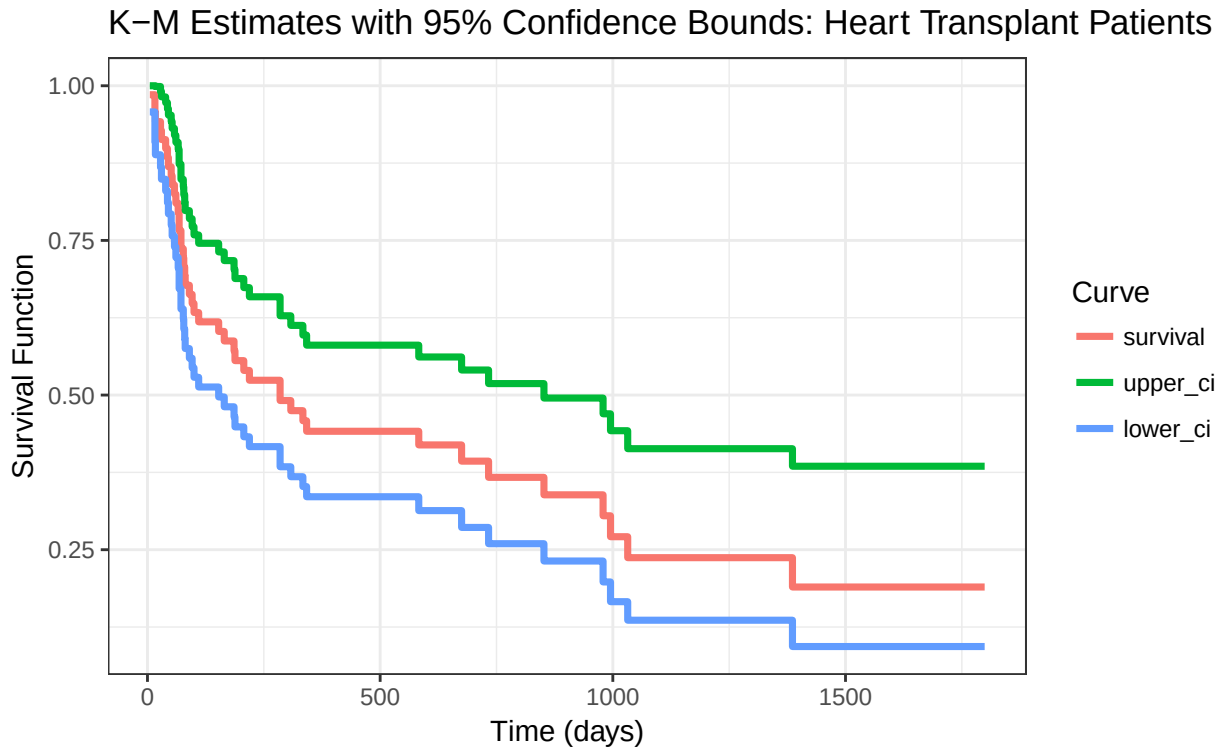
Age	Status	Days
41	1	5
40	1	16
35	0	39
50	1	53
45	1	68
26	0	180

This dataset is right censored, because as shown above, we don't exactly how long patients who currently have a (status=0) will survive, or survived.

Utilizing this dataset, and the *survival* package in R, we generate the Kaplan-Meier estimates, and visualize the survival function, i.e. $S_{KM}(x_{i:n}) = 1 - F_{KM}(x_{i:n})$, with confidence intervals below. Note that the R code utilized is also shown:

R Code

```
km_fit <- survfit(Surv(time = Days, event = Status) ~ 1, data = hdat,
                  type = 'kaplan-meier') #only one group of patients
km_dat <- data.frame(time=km_fit$time,
                     survival=km_fit$surv,
                     upper_ci = km_fit$upper,
                     lower_ci = km_fit$lower) #model results
melted_km_dat <- melt(km_dat, id.vars = c('time')) #transform for viz
ggplot(aes(x=time,y=value, color = variable ,group = variable), data=melted_km_dat) +
  geom_step(size = 1.25) + theme_bw() + xlab('Time (days)') +
  ylab('Survival Function') +
  ggtitle('K-M Estimates with 95% Confidence Bounds: Heart Transplant Patients') +
  labs(color='Curve')
```



Visual analysis of the plot here, indicates that the probability of survival for 500 days, after a heart transplant is approximately 42%, with a 95% confidence range of approximately 30-55%, among patients in this data, when this data was collected decades ago. We hope that survival rates have increased significantly since this study was conducted.

2. Kernel and Related Regression with One Explanatory Variable (40%):

Locate a data set suitable for nonparametric regression (usually has nonlinear y-x relationship) in the field of your interest, e.g., eCommerce, medical study, drug development, supply-chain/logistics operations. Apply all of the procedures below:

1) *Kernel Regression,*

2) *Local Polynomial Regression,*

3) *LOESS,*

4) *Smoothing Spline, to the y-x data-fit.*
   - Compare fits from the four methods.

3. Cross-Validation With the "Leave-One-Out" Procedure (10%):

Compare the above four methods with a leave-one-out cross-validation procedure.

4. Resampling Procedures: Bootstrap and Jackknife (25%):

1) *Select an input x_0 in the [min(x-data), max(x-data)].*

2) *Use all four regression models built in Task #2 to make point-predictions of Y at x_0.*

3) *Use both bootstrap ($B = 1000$) and jackknife resampling procedures to find a 90% pointwise confidence interval (CI) for the point-prediction. If the resampled distribution of the point-prediction is symmetric, use 5% in each tail to find the CI-bounds. If the distribution is not symmetric, use the HPD-interval idea to find the CI-bound. Compare the results from four regression methods.*