Funnels

Feb 18th, 2017

Eric Lehman, Alice Zhao, April Wang, Lin Chen, Francisco Calderon

1 Relationship Between λ and Survival Times

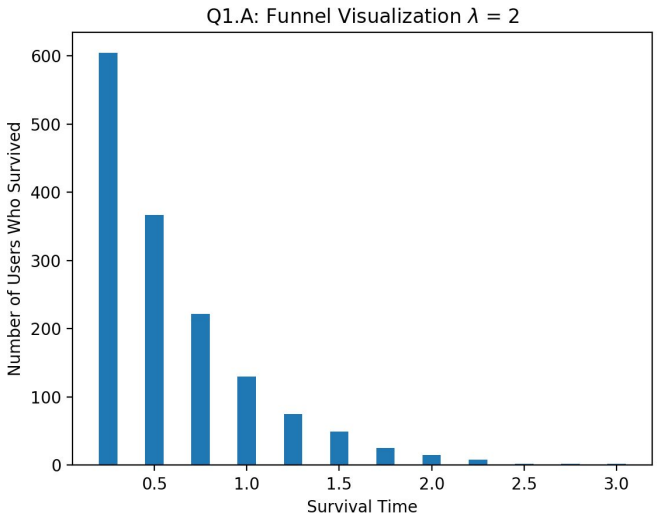
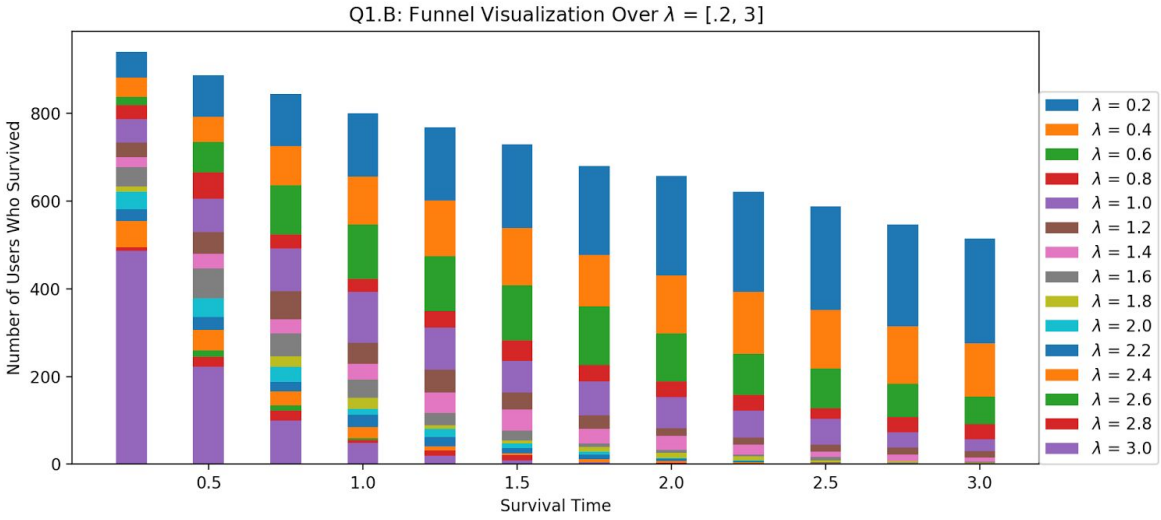


Figure 1 shows an example of a Funnel visualization where the number of users who made it past each survival time decreases at an exponential rate. It can be shown that a relationship between simulated users and parameter that governs the exponential distribution exists, as shown in Figure 2. The number of users who reach higher survival times decreases as λ increases.



2 Maximum Likelihood Estimation

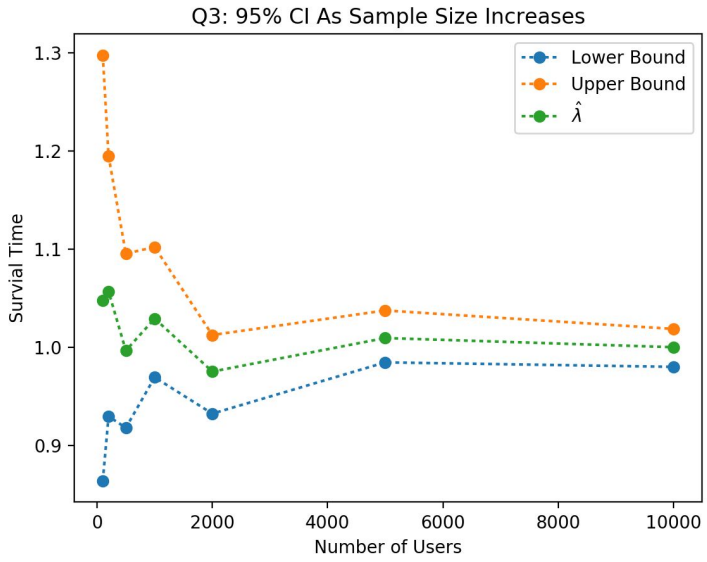
A. Bias

An unbiased estimator has an expected error of zero when compared to the population parameter it is trying to estimate. Bias is defined as:

$$\text{Bias} = \theta - E(\hat{\theta})$$

D. Confidence Interval Relationship

As shown in Figure 3, the confidence interval for the λ estimate and the λ estimate itself begin to converge as the number of users increases. The confidence intervals and estimates show noise at the beginning but it dissipates as number of users increases.



3 Modifying MLE

A. Traditional MLE Does Not Alway Work

Given that event tracking data is censored, usage data does not actually contain the times users actually quit using an app or quit an onboarding process. Additionally, metrics like average time a user quits an onboarding process are difficult to define. This is motivation for estimating breakpoints that assist in estimating user's survival times that minimizes the bias.

B. Different Kind of MLE

$$log(L) = log(\prod_{i=1}^{m_0} F(BP_1|\lambda) + log(\prod_{i=m_0+1}^{m_0+m_1} F(BP_{U_i+1}|\lambda) - F(BP_{U_i}|\lambda)) + log(\prod_{i=m_o+m_1+1}^{n} 1 - F(BP_t|\lambda)) \quad (A.1)$$

$$= m_0 * log(F(BP_1|\lambda) + log(\prod_{i=m_0+1}^{m_0+m_1} F(BP_{U_i+1}|\lambda) - F(BP_{U_i}|\lambda)) + (n - m_o - m_1) * log(1 - F(BP_t|\lambda)) \quad (A.2)$$

$$= m_0 * log(F(BP_1|\lambda) + log(\prod_{i=m_0+1}^{m_0+m_1} F(BP_{U_i+1}|\lambda) - F(BP_{U_i}|\lambda)) + m_2 * log(1 - F(BP_t|\lambda)) \quad (A.3)$$

$$= m_0 * log(1 - e^{-\lambda BP_1}) + m_2 * log(e^{-\lambda BP_t}) + \sum_{i=m_0+1}^{m_0+m_1} log(F(BP_{U_i+1}|\lambda) - F(BP_{U_i}|\lambda)) \quad (A.4)$$

Where:

Note, $F(x) = 1 - e^{\lambda x}$ *as is defined*
U_i *= the number of steps that each user completes*
b breakpoints that occur in time $BP_1, BP_2 .. BP_t$
m_0 *= the number of users who never send the first event* ($U_i = 0$)
m_1 *= the number of users who send any number of events but the last one* (U_i *between* 1 *and* $b - 1$)
m_2 *= the number of the users who send all events* ($U_i = b$)

4 Estimating Bias

A. Average Bias

Bias is estimated as the difference between $\hat{\lambda}$ from a given simulated sample and the $\hat{\lambda}$ that maximizes the log-likelihood equation A.4. Bias can be show to be directly affected by the placement of breakpoints. Specifically, the placement of the second breakpoint can increase or decrease bias. Shown in Table 1, the first set of breakpoints show positive bias, and as the second breakpoint moves further, negative bias results from the simulation. The optimal placement for the second breakpoint is between 0.75 and 3.0.

First Break Point	Second Break Point	Average Difference
0.25	0.75	0.399
0.25	3.0	-0.412
0.25	10.0	-1.932

B. Breakpoint Design

Breakpoints should be designed with minimizing absolute bias in mind. This can be done by adjusting the length between breakpoints, as shown in Table 1.