Estimation Algorithms

May 30, 2021

This document describes two methods for estimating the total size of a population given a collection of samples taken with replacement. The first is described in Cuthbert, Michael Scott. 2009. "Tipping the Iceberg: Missing Italian Polyphony from the Age of Schism," *Musica Disciplina* 54: 39–74. The second is described in Boneh, Shahar, Arnon Boneh, and R. J. Caron. 1998. "Estimating the Prediction Function and the Number of Unseen Species in Sampling with Replacement," *Journal of the American Statistical Association* 93: 372–79. This discussion adopts the following notation.

- N is the true population size.
- y is the number of samples taken.
- x_k is the size of the k^{th} sample.
- n is the number of distinct entities observed across all samples.
- n_k is the number of entities observed k times across all samples, including n_0, which is the number of unobserved entities in the population.
- $p = \frac{n}{N}$ is the proportion of the population observed.

Cuthbert

Cuthbert's method of estimation is a two-stage process that relies on probabilistic reasoning. For the first stage, we momentarily assume that the samples are independent, random, and select from the entire population. Under this assumption, the probability that a given entity will appear in the

k^{th} sample is $\frac{x_k}{N}$, and the probability that it will not appear in any sample is the product of the probabilities that it does not appear in each sample: $\mathbb{P}(unobserved) = \prod_{k=1}^{y}(1 - \frac{x_k}{N}) = \frac{\prod_{k=1}^{y} N - x_k}{N^y}$. Moreover, since under this assumption each of the N entities in the population has an identical likelihood of not appearing in any sample, the expected number of unobserved entities is $E(n_0) = N \cdot \mathbb{P}(unobvserved) = \frac{\prod_{k=1}^{y} N - x_k}{N^{y-1}}$. And since, by definition, $n_0 = N - n$, an estimate for N can be generated by solving for it in the following equation (all parameters other than N are known).

$$\frac{\prod_{k=1}^{y} N - x_k}{N^{y-1}} - (N - n) = 0 \tag{1}$$

This equation is challenging to solve analytically, since the left-hand side of this expression is largely a ratio of polynomials in N of approximate degree y. However, since the value of N is assumed to be a positive integer greater than n but less than some reasonably large upper bound, and since the left-hand side of the equation is a decreasing function of N, an approximate solution to this equation can be found relatively expediently via recursive binary search (this is the strategy implemented in `iceberg.estimate`). This approximate solution is our "initial estimate" of N, $\hat{N}_0$.

In the second stage of Cuthbert's method, we cross-validate the initial estimate of N to test and correct for the assumption that the samples are independent and random. Note that neither the independence nor the randomness of the samples themselves are really being tested here. Rather, we are testing the degree to which the distribution of entities among the samples, *as a whole*, approximates the distribution that would be expected of truly random samples. The process involves first simulating a population of $\hat{N}_0$ entities that includes every observed entity along with $\hat{N}_0 - n$ "dummy" entities, each representing an unobserved entity. We then randomly choose a number of the original samples to serve as a "validation set," and note how many entities out of the n originally observed would not have been observed if those samples had not been collected—call this n_{lost}. Then construct a "simulated set" of samples by iterating through the validation set and taking truly random samples of identical size from the simulated population (all originally observed entities plus the new dummy entities), and count how many "new" entities are in the simulated set, from the perspective of the corpus of known samples *not* in the validation set—call this $\hat{n}_{lost}$. The

"error factor" associated with this cross-validation experiment, ε_i, is then either the extent by which the true number exceeds the simulated number relative to the smaller quantity, or 1 if the simulated number is greater than the true number: $\varepsilon_i = 1 + \max\left(\frac{n_{lost} - \hat{n}_{lost}}{\hat{n}_{lost}}, 0\right)$. Restricting $\varepsilon_i \geq 1$ ensures that cross-validation only increases our ultimate estimates of N (or equivalently, that it only decreases our estimates of p). The "corrected estimate" for cross-validation experiment i, $\hat{N}_i$, is then:

$$\hat{N}_i = n + \varepsilon_i \left(\hat{N}_0 - n\right) \tag{2}$$

The resultant distribution of $\{\hat{N}_i\}$ across a sufficiently large number of cross-validation experiments then indicate something about the stability of $\hat{N}$ for a given population—which is to say, the sensitivity of this estimate to non-randomness in the samples. More specifically, while the definition of the error factors ensures that most (if not all) distributions of $\{\hat{N}_i\}$ will exhibit some leftward skew, the relative severity of this skew can still indicate whether the estimate is comparatively stable or unstable.

Boneh, Boneh, and Caron (BBC)

The second method, proposed by BBC, is completely different in its motivation and execution. the authors begin by considering a multinomial distribution, describing the outcome of sampling from N objects with replacement and with probabilities $p_1, \ldots, p_N$. They then observe that this is related in the limit to a scenario in which there are N independent Poisson processes with parameters $\lambda_1, \ldots, \lambda_N$. The relation is fairly transparent: if we track these Poisson processes in the interval $[0, 1]$ and count how many of them occur once, how many occur twice, etc., then this is identical to generating values for $\{n_1, n_2, \ldots n_m\}$, where m is the maximum number of times that any individual Poisson process is detected.

In order to use this information to estimate the total number of Poisson processes, N, it is useful to define the auxiliary function $D(t)$ to be the number of processes detected in the interval $(1, t+1]$ that were not first detected in the interval $[0, 1]$, and the function $\Psi(t) = E(D(t))$, which BBC call "the prediction function." $\Psi(t)$ has several attractive mathematical properties, including that it has infinite order alternating copositivity (that is, its k^{th} derivative takes positive values on the positive half-line for all odd k and negative values for all even k) and that it is bounded, which together mean that

it has an asymptotic limit as t increases. Computing this limit is tantamount to generating an estimate for n_0.

BBC also show that this limit may be estimated with a relatively simple two-part process. First, calculate a biased estimate, $\hat{\Psi}(\infty)$, using a simple sum of exponentials:

$$\hat{\Psi}(\infty) = \sum_{k=1}^{m} n_k e^{-k} \tag{3}$$

An unbiased estimate, $\hat{n}_0$, can then be obtained by numerically solving the equation:

$$\hat{n}_0 \left(1 - e^{\frac{-n_1}{\hat{n}_0}}\right) = \hat{\Psi}(\infty) \tag{4}$$

BBC also give some details for how this equation may be efficiently solved via numerical methods; `iceberg.estimate` utilizes their algorithm.