Estimating statistical uncertainties with "hijacked" bootstrap

May 31, 2022

The frequently encountered problem, after the experiment is over, is how to report the statistical uncertainty for the measured observables of interest. This is particularly difficult for the compound observables. For instance, if the directly measured observables are x and y, but we are interested in the quantity $z \equiv x^2 + e^{-y} + \cos xy$, it is very difficult, if not impossible, to perform the standard error propagation and obtain the statistical uncertainty for desired observable z, from the statistical uncertainties of σ_x, σ_y and their covariance $\mathrm{Cov}(x, y)$, which can be obtained directly, because only x and y are measured directly. This sort of problem in general can be resolved with the *bootstrap* technique.

There is a long story about how to apply the bootstrap technique in the data analysis (see for instance `https://en.wikipedia.org/wiki/Bootstrapping_(statistics)`). To make this long story short, we have endorsed in ALICE the simplified ("hijacked") version of this technique. That is the main subject of this exercise.

The simplified version of bootstrap technique consists of the following steps:

1. You have to divide the initial sample in 10 subsamples, so that each subsample contains roughly the same statistics;

2. For each subsample, you need perform an independent measurement and get for your observable of interest the results μ_0, μ_1, ... , μ_9, respectively;

3. You then compute an unbiased variance from these 10 subsamples, i.e.

$$\mathrm{Var} = \frac{1}{10 - 1} \sum_i^{10} (\mu_i - \mu)^2 \,, \tag{1}$$

   where the result for the starting large sample of your observable of interest is denoted by μ in the above formula.

4. The final result for your observable of interest and its statistical uncertainty are:

$$\mu \pm \sqrt{\frac{\mathrm{Var}}{10}} \,. \tag{2}$$

If the statistics permits, one can divide the initial sample in $N > 10$ subsamples, but the final results do not change much (it's easy to implement the above steps for arbitrary N, however, the procedure makes sense as long as it is possible to ensure that each subsample contains roughly the same statistics.). On the other hand, for $N < 10$ subsamples the bootstrap technique is not stable and is therefore not recommended.

In practice, therefore, the statistical errors are reported in either of the two ways:

1. For simple observables, which can be measured directly, whatever you get from ROOT for its statistical error (e.g. from histogram or profile), is perfectly fine;

2. For compound observables, the standard error propagation and calculation of covariance matrices are typically not feasible. Instead, the boostrap technique is used as described above. Note that here you only need to develop some clever algorithm that will divide your initial dataset in 10 subsets of approx. equal size, but this is typically straightforward.

Question 1: *Consider again the Bessel-Gaussian p.d.f., now parameterized as*

$$f(v) \equiv \frac{v}{b^2} e^{-\frac{v^2+a^2}{2b^2}} I_0\left(\frac{va}{b^2}\right).$$

(3)

Use in this exercise that $a^2 + 2b^2 = 0.0025$ and that $a = b/10$ (it can be shown analytically that $\langle v^2 \rangle \equiv \mu_{v^2} = a^2 + 2b^2$, so these suggested numbers correspond to the realistic values of elliptic flow in reality, i.e. $v \approx 0.05$). Sample 10M times v from the above distribution, and fill the histogram (or profile) with v^2. What is the statistical error of v^2 obtained directly from histogram (or profile)? What is the statistical error of v^2 obtained from the bootstrap technique using 10 subsamples explained at the beginning? Are the two independently obtained results for statistical error consistent?

Question 2: *Repeat the bootstrap part of the previous examples for $N = 20$ and $N = 100$ subsamples. Are all 3 results for the statistical error obtained via bootstrap consistent?*

Once validated in a simple Toy Monte Carlo study like this, bootstrap technique can be used readily also for the compound observables, for which the statistical error cannot be obtained directly from the native ROOT classes.