Bootstrap Methods in R

Ella Kaye Xenia Miscouridou

October 20, 2015

Abstract

We present a variety of bootstrap methods in R.

1 The Bootstrap

The bootstrap [1], is a statistical tool which gives measures of accuracy of estimators, and can therefore be used draw inference about the parameters of the sampling distribution. It is a powerful and widely applicable tool.

Suppose we observe a sample of n iid realisations $x_1, \ldots, x_n \sim P$, for some probability measure P. The goal is to obtain an estimate $\theta_n = \theta_n(x_1, ..., x_n)$ and compute an assessment ξ of the quality of θ_n (for example, θ_n could be the median, and ξ the standard error). To obtain a bootstrap estimate, we procede as follows:

1. Repeatedly (B times) sample n points with replacement from the original dataset.
2. Compute θ_n^* on each of the B resamples.
3. Compute ξ^* based on these B realizations of θ_n^* as our estimate of ξ of θ_n

We have developed a function, bootstrap, which automates the above procedure. It takes as inputs a data set (which can be a vector, matrix or dataframe), a function, FUN, which is used to obtain the statistic of interest, and the number, B, of bootstrap resamples required. We now show how to use this function to replicate the example on page 37-38 of CITE EFRON AND GONG 1983, where we are interested in obtaining the bootstrap estimate of the standard error of the Pearson correlation coefficient:

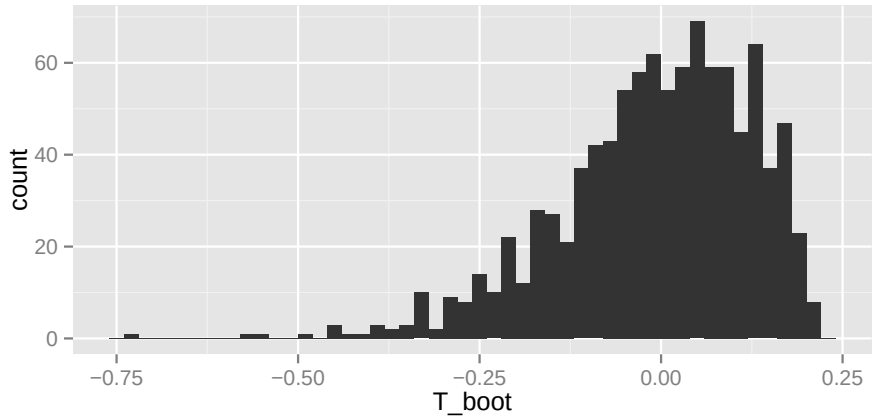
```
law_school <- data.frame(
  LSAT=c(576,635,558,578,666,580,555,
         661,651,605,653,575,545,572,594),
  GPA=c(3.39,3.30,2.81,3.03,3.44,3.07,3.00,
        3.43,3.36,3.13,3.12,2.74,2.76,2.88,2.96))
```

```r
# computes correlation coefficient of data stored in a data frame
cor_df <- function(bivar_data) {
  cor(bivar_data[,1], bivar_data[,2])
}

set.seed(1)
bootstrap_law <- bootstrap(law_school, cor_df, B=1000)
bootstrap_law$se

## [1] 0.1334911

library(ggplot2)
bootstrap_law_df <- data.frame(T_boot = bootstrap_law$T_boot - cor_df(law_school))
qplot(T_boot, data=bootstrap_law_df, geom="histogram", binwidth=0.02)
```



2 Bayesian Bootstrap

Rubin introduced the Bayesian bootstrap (BB) as the natural Bayesian analogue of the bootstrap. Each BB replication generates a posterior probability for each x_i, which is centered at $1/n$, but has variability. To obtain a BB replication, first generate a set of weights by drawing $(n-1)$ uniform(0,1) random variates, $u_1, \ldots, u_(n-1)$, ordering them and calculating the gaps $g_t = u_{(t)} - u_{(t-1)}, t = 1, \ldots, n$, where $u_{(0)} = 0$ and $u_{(n)} = 1$. These gaps, $g = (g_1, \ldots, g_n)$ form the weights to attach to the data values in that replication. Considering all the BB replications gives the BB distribution of X, and thus of any parameter of this distibution.

Our function BB gives a BB distribution. It takes as its inputs a dataset (as a vector, matrix or dataframe), a function for calculating the statistic $\hat{\theta}$ (this function must take two arguments - one for data and one for weights), and B,

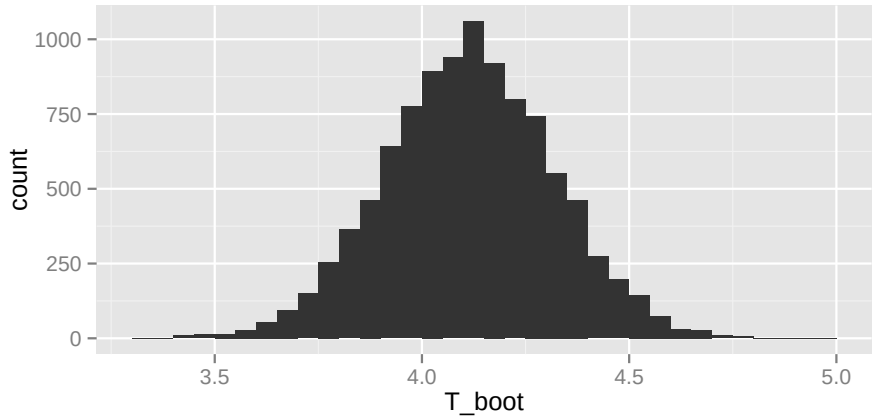
the number of replications required. It returns a list, the first item of which is the standard error of the replicates. The second item is a dataframe with $\theta_1^*, \ldots, \theta_B^*$.

The following code gives the BB distribution when the parameter of interest is the mean. Here the function weighted.mean is in the R base package.

```
X <- rnorm(100, 4, 2)
BB_mean <- BB(X, weighted.mean, B=10000)
BB_mean$se

## [1] 0.2037395

qplot(T_boot, data=BB_mean$replicates, geom="histogram", binwidth=0.05)
```



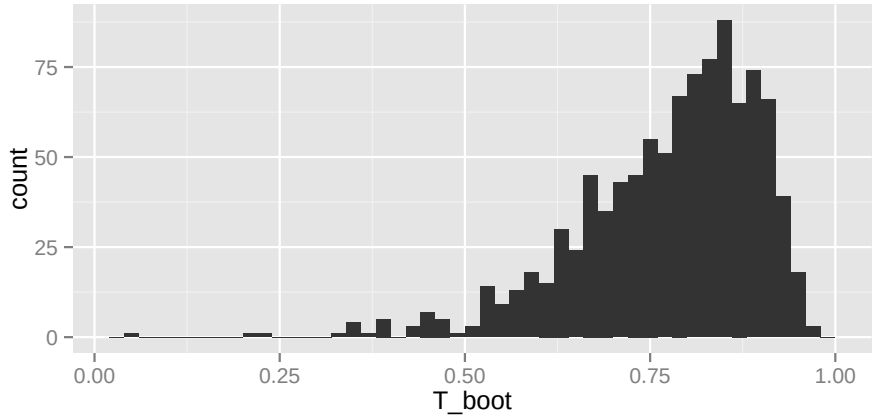
It may be necessary to define your own function for the statistic of interest that can take weights as an argument. By may of example, we show the BB equivalent of the correlation example from the previous section.

```
# takes bivarate data and weights (g) and returns a weighted correlation
weighted.cor <- function(data, g) {
  y <- data[,1]; z <- data[,2]
  wc_num <- sum(g*y*z)-(sum(g*y))*(sum(g*z))
  wc_den <- (sum(g*y^2)-(sum(g*y))^2)*(sum(g*z^2)-(sum(g*z))^2)
  wc_num/sqrt(wc_den)
}

set.seed(1)
BB_law <- BB(law_school, weighted.cor, B=1000)
BB_law$se

## [1] 0.1224361

qplot(T_boot, data=BB_law$replicates, geom="histogram", binwidth=0.02)
```



ISSUES WITH THE BAYESIAN BOOTSTRAP, AND HENCE BOOTSTRAP.

3 Bag of Little Bootstraps

The original bootstrap arose around the time when increases in computing power allowed the development of statistical tools that had previously been too computationally expensive. In recent years, there has been an influx of 'big data', alongside the development of parallel computing architectures. CITE KLEINER ET AL have developed a scalable bootstrap for massive data, known as the Bag of Little Bootstraps (BLB). With massive datasets, the bootstrap's need for recomputation on resamples of the same size as the original dataset is problematic. Rather than obtain bootstrap samples from the whole dataset, the BLB breaks down the process as follows:

1. Repeatedly (s times) subsample $b(n) < n$ points *without replacement* from the original dataset of size n.
2. For each of the s subsamples, do the following:
   (a) Repeatedly (r times) resample n point *with replacement* from the subsample.
   (b) Compute $\hat{\theta}_n^*$ on each resample.
   (c) Compute an estimate of ξ based on these multiple resampled realisations of $\hat{\theta}_n^*$.
3. We now have one estimate of ξ per subsample. Output their average as the final estimate of ξ for $\hat{\theta}_n$.

CITE KLEINER recommends taking $b(n) = n^\gamma$, where $\gamma \in [0.5, 1]$. This procedure dramatically reduces the size of each resample. For example, if n

$= 1$ million and $\gamma = 0.6$, the size of the original dataset is around 1TB, with a bootstrap resample typically occupying approximately 632GB, and a BLB subsample or resample occupying just 4GB.

The function BLB.1d implements the simplest version of BLB. It takes as input a 1-dimensional dataset in a vector and a function with computes the statistic of interest, θ_n. It also takes as arguments s and r, which default to 20 and 100 respectively (KLEINER demonstates are likely as large as they'll need to be to obtain convergence), and γ, which controls the value of b. The function returns ξ, which is set as the standard error.

```
X <- rnorm(5000)
BLB.1d(X, mean, gamma=0.5)

## [1] 0.01390571
```

In their paper, CITE KLEINER conduct a simulation study, and here we replicate their results in the Gaussian case. We generate data from the true underlying distribution, a linear model $Y_i = \tilde{X}_i^\top \mathbb{I}_d + \epsilon_i$, with iid $\tilde{X}_i \sim N(0, 1)$ and $\epsilon_i \sim N(0, 10)$. The estimator $\hat{\theta}_n$ consists of a linear least squares regression with a small L_2 penalty of 10^{-5}.

DISCUSSION OF BLB
Not run in parallel

References

[1] Bradley Efron. "Bootstrap methods: another look at the jackknife". In: *Annals of Statistics* 7 (1979), pp. 1–26.