Bootstrap Methods: A Review*

by S. N. Lahiri

Department of Statistics, Iowa State University

June 2, 2008

*. Presentation by Yihui Xie; School of Statistics, Renmin University of China, Beijing 100872; http://www.yihui.name; This document has been written using the GNU TeXmacs text editor (see `www.texmacs.org`).

Table of contents

1. Introduction

→ Inception: Efron (1979)

→ Wide applications: Hall (1992), Efron and Tibshirani (1993), Shao and Tu (1995), Davison and Hinkley (1997), Lahiri (2003a), ...

→ Basic idea

- How to measure the accuracy of an estimate $\hat{\theta}_n$? It's usually difficult because of the unknown sampling distribution

- Bootstrap: we don't need assumptions on the data generating mechanism; just resample with replacement to reproduce the estimators

- From $\theta \to \hat{\theta}_n \to \hat{\theta}_n^*$ (with corresponding concepts such as the variance or α-quantile)

- → The quality of the bootstrap approximation depends on the estimator $\hat{P}_n$ of the joint distribution P_n

- → Data structure: i.i.d or dependence

- → The indespensable role of the computer: highly involved with computation because of it's difficult to derive the closed form analytical expressions for the bootstrap estimators

- → Routine: first produce a large number of independent copies of $\hat{\theta}_n^*$, then we get the empirical distribution, which is a Monte-Carlo approximation to the true bootstrap distribution

2. Bootstrap for i.i.d Data

→ Sometimes bootstrap outperforms classical approaches

→ But a blind application of the bootstrap gives a wrong answer too

2.1 Performance of the Bootstrap

Let $T_n = t_n(\boldsymbol{X}_n; \theta)$ be a random variable of interest, and a common example is $T_n = \sqrt{n}(\bar{X}_n - \mu)/\sigma$, where $\bar{X}_n$ is the sample mean. The bootstrap version of T_n based on a resample of size m is

$$T^*_{m,n} = t_m(X^*_1, \ldots, X^*_m; \hat{\theta}_n) = \sqrt{m}(\bar{X}^*_m - \bar{X}_n)/s_n. \tag{1}$$

When $m = n$, it can be shown that the bootstrap approximation is asymptotically consistent, and the rate is $o(n^{-1/2})$ under some conditions[1], which is better than the rate of the classical normal approximation $O(n^{-1/2})$.

Additional moment conditions may yield even more precise results.

The Problem with Delete-1 Jackknife

The variance estimator of the sample quantile is inconsistent!

---

1. This $o(n^{-1/2})$ is called second order correctness.

2.2 Superiority of Bootstrap May Not Always Hold

→ For lattice random variables, the bootstrap looses its second order correctness.

→ When the variance is infinite, the certered and scaled sample mean will converge to a random limit. However, after modifying the resample size n to a smaller number, say, m, the inconsistency can be overcome! (see Figure 1)

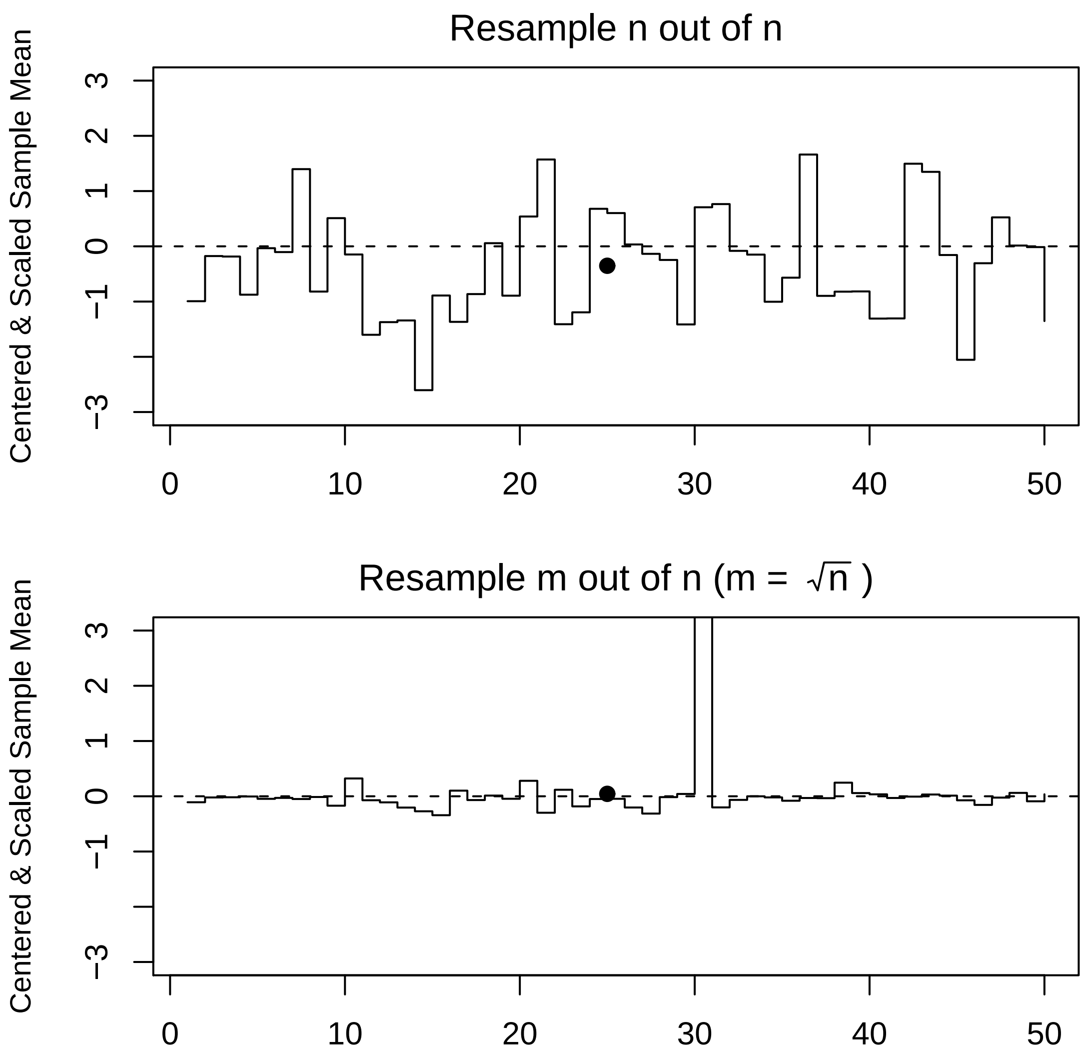

Figure 1. Application of "m out of n" bootstrap in the case of infinite variance (a stable distribution of order $\alpha \in (0, 2)$, here $\alpha = 0.8$), contrasted by the usual "n out of n" bootstrap. Actually this figure is only a snapshot.

R Code for Figure 1:

```
> if (require(fBasics)) {
    par(mfrow = c(2, 1), mar = c(3, 4, 2, 1))
    x = rstable(n = 1000, alpha = 0.8, beta = 0)
    n1 = length(x)
    n2 = round(sqrt(length(x)))
    d1 = replicate(50, sqrt(n1) * (mean(sample(x, size = n1,
        rep = T)) - mean(x))/sd(x))
    d2 = replicate(50, sqrt(n2) * (mean(sample(x, size = n2,
        rep = T)) - mean(x))/sd(x))
    while (T) {
        plot(d1, type = "s", ylim = c(-3, 3), ylab = "Centered & Scaled Sample Mean",
            cex.lab = 0.9)
```

```
        mtext("Resample n out of n", 3, 0.5, cex = 1.2)
        abline(h = 0, lty = 2)
        points(25, mean(d1), pch = 19)
        d1 = c(d1[-1], sqrt(n1) * (mean(sample(x, size = n1,
            rep = T)) - mean(x))/sd(x))
        plot(d2, type = "s", ylim = c(-3, 3), ylab = "Centered & Scaled Sample Mean",
            cex.lab = 0.9)
        mtext(expression("Resample m out of n (m = " ~ sqrt(n) ~
            ")"), 3, 0.5, cex = 1.2)
        abline(h = 0, lty = 2)
        points(25, mean(d2), pch = 19)
        d2 = c(d2[-1], sqrt(n2) * (mean(sample(x, size = n2,
            rep = T)) - mean(x))/sd(x))
        Sys.sleep(0.3)
    }
}
```

The code above will create two animated pictures, if the package fBasics has already been installed (because we have to generate random numbers following stable distribution using the function rstable()).

3. Model Based Bootstrap

In practical applications, the i.i.d case is too primitive, thus various extensions are explored.

i. multiple linear regression model

ii. stationary autoregressive process

3.1 Multiple linear regression model

Suppose the data is generated by

$$Y_i = x_i^{'}\beta + \epsilon_i, \quad i = 1, ..., n. \tag{2}$$

How to approximate the sampling distribution of $T_n = A_n(\hat{\beta}_n - \beta)$?

1. Compute the residuals $e_i = Y_i - x_i^{'}\hat{\beta}_n$, and center them as $\tilde{e}_i = (e_i - \bar{e}_n)$, $i = 1, ..., n$
2. Generate bootstrap error variabels ϵ_1^*, ..., ϵ_n^* by resampling with replacement from $\{\tilde{e}_i \colon i = 1, ..., n\}$
3. Define
$$Y_i^* = x_i^{'}\hat{\beta}_n + \epsilon_i^*, \quad i = 1, ..., n.$$
4. Get bootstrap version coefficients β_n^*, and consequently $T_n^* = A_n(\beta_n^* - \hat{\beta}_n)$

3.2 Stationary autoregressive process

Consider an AR(1) model:

$$X_t = \beta_1 X_{t-1} + \epsilon_t,\ \ t = 0, \pm 1, \pm 2, \ldots \tag{3}$$

where $\beta_1 \in (-1, 1)$. Let $\hat{\beta}_1$ denote the LSE of the above model.

1. Compute the residuals $e_t = X_t - \hat{\beta}_1 X_{t-1}$, $t = 2, \ldots, n$
2. Center them as in Section 3.1
3. Generate $\epsilon_t^*, t = 2, \ldots, n$
4. Get a sequence of bootstrap observations $X_1^*, \ldots, X_n^*$ as

$$X_t^* = \hat{\beta}_1 X_{t-1}^* + \epsilon_t^*,\ \ t = 2, \ldots, n. \tag{4}$$

Notice: for unstable AR processes, the AR-bootstrap fails with the usual choice of the resample size $m = n$, but works if $m = o(n)$ as $n \to \infty$

4. Block Bootstrap

→ One of the common drawbacks of model based bootstrap is that it's very sensitive to model misspecification! e.g. unstable time series

→ So we turn to a new approach to bootstrapping time series in the absence of a model, i.e. block bootstrap

→ An obvious advantage: the dependence structure is preserved because we resample blocks instead of single elements!

→ A critical question: how to get an optimal block size ℓ?

- explicit formulas for MSE-optimal block size: hard derivation and extension
- Hall *et al.* (1995) method: use subsampling method to construct an estimator of the MSE as a function of the block size and then minimize
- Jackknife-after-bootstrap method

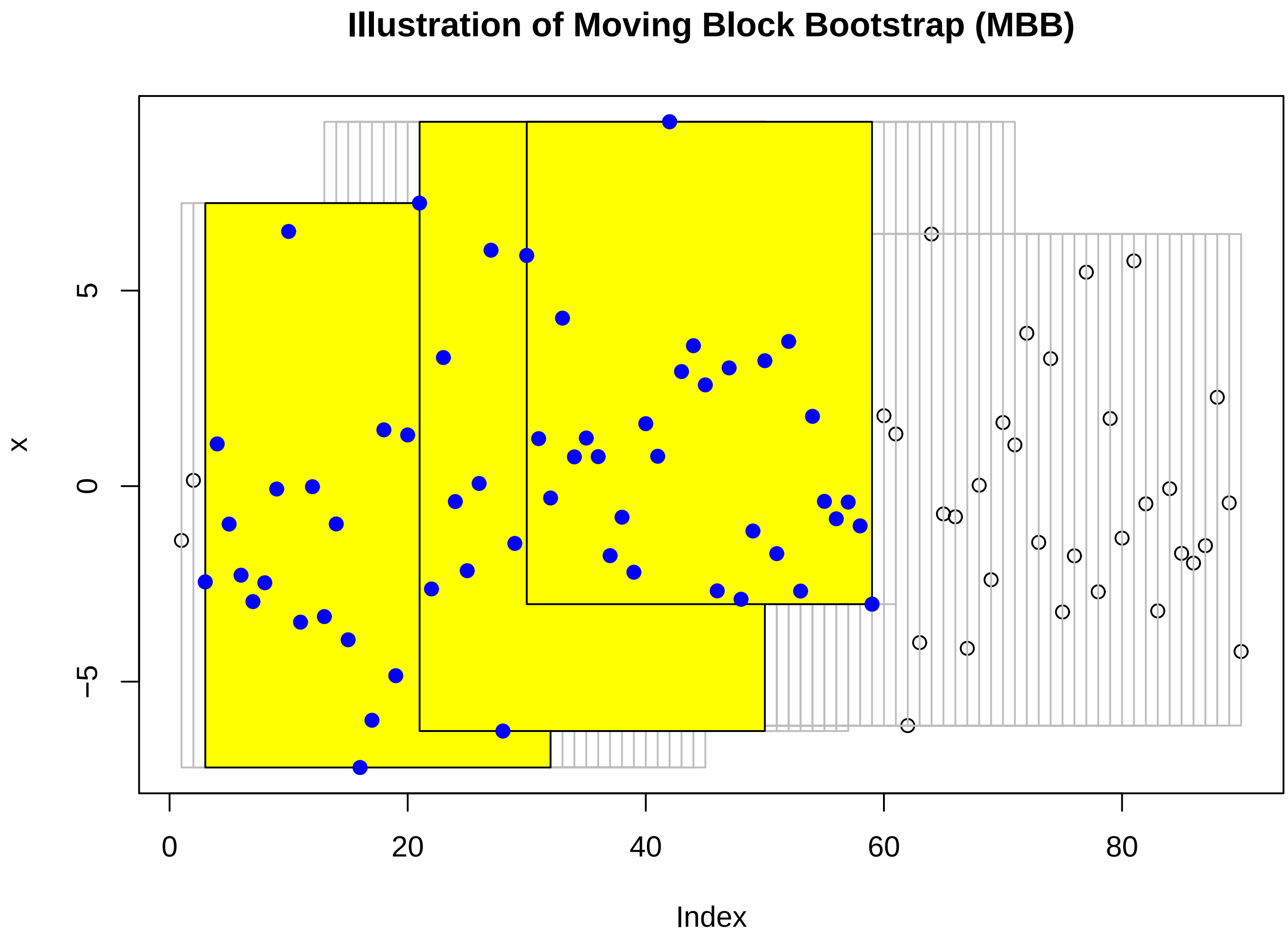

Figure 2. An illustration of "Moving Block Bootstrap"; the blue points in the three yellow rectangles (blocks) are bootstrap samples.

```
> x = 5 * sin(seq(0, pi, length = 90)) + rnorm(90)
> plot(x, main = "Illustration of Moving Block Bootstrap (MBB)")
> for (idx in 1:(length(x) - 30 + 1)) {
    rect(idx, min(x[idx:(idx + 30 - 1)]), idx + 30 - 1, max(x[idx:(idx +
        30 - 1)]), border = "gray")
    Sys.sleep(0.2)
}
> bt = sample(1:(length(x) - 30 + 1), 3, rep = T)
> for (b in bt) {
    rect(b, min(x[b:(b + 30 - 1)]), b + 30 - 1, max(x[b:(b +
        30 - 1)]), col = "yellow")
    points(b:(b + 30 - 1), x[b:(b + 30 - 1)], col = "blue", pch = 19)
}
```

5. Sieve Bootstrap

Construct a sequence of probability distributions $\{\tilde{P}_n\}_{n\geq 1}$ that forms a sieve, i.e. the sequence $\{\tilde{P}_n\}_{n\geq 1}$ is such that for each $n \geq 1$, $\tilde{P}_{n-1}$ is a finer approximation to P than $\tilde{P}_n$ and $\tilde{P}_n$ converges to P in a suitable sense.

6. Transformation Based Bootstrap

Why transformation? – to build an appropriate structure, e.g. independent structure.
Let $\theta \equiv \theta(P)$ be a parameter of interest depending on the joint distribution P of the sequence $\{X_t\}_{t\in\mathbb{Z}}$. We want to approximate the sampling distribution of a statistic $R_n = r_n(\boldsymbol{X}_n; \theta)$.

1. Let $\boldsymbol{Y}_n = h_n(\boldsymbol{X}_n)$ is a transformation of $\boldsymbol{X}_n$ such that the components of $\boldsymbol{Y}_n$ are approximately independent.

2. Suppose R_n can be expressed (or a close approximation) in terms of $\boldsymbol{Y}_n$ as $R_n = r_{1n}(\boldsymbol{Y}_n; \theta)$ for some reasonable function r_{1n}

3. Resample from a suitable sub-collection of $\{\boldsymbol{Y}_i\}$ to generate bootstrap observations $\boldsymbol{Y}_n^*$, then approximate the distribution of R_n by the *transformation-based bootstrap* (TrBB).