Bootstrap Confidence Intervals

Nathaniel E. Helwig

Assistant Professor of Psychology and Statistics
University of Minnesota (Twin Cities)

Updated 04-Jan-2017
Copyright © 2017 by Nathaniel E. Helwig
Outline of Notes

1) Confidence Intervals
   - Overview of CIs
   - Interpreting CIs
   - What is a good CI?

2) Basic Bootstrap CIs
   - \( t \) with bootstrap SE
   - Percentile intervals
   - Examples

3) Better Bootstrap CIs:
   - Expanded percentile
   - Bootstrap \( t \) tables
   - Bias-corrected & accelerated
   - Examples (revisited)

For a thorough treatment see:
Confidence Intervals
A symmetric 100(1 − \(\alpha\))% confidence interval (CI) has the form:

\[ \hat{\theta} \pm t_{\alpha/2} \hat{\sigma} \]

where \(\hat{\theta}\) is our estimate of \(\theta\), \(\hat{\sigma}\) is the standard error of \(\hat{\theta}\), and \(t_{\alpha/2}\) is the critical value of the test statistic, i.e., \(P(t \leq t_{\alpha/2}) = \alpha/2\).

- Assumes that distribution of test statistic is symmetric around zero
- As \(n \to \infty\) we often have \(\hat{\theta} \sim N(\theta, \hat{\sigma}^2)\), so that \(t_{\alpha/2} = z_{\alpha/2}\).

More generally we can write a 100(1 − \(\alpha\))% CI as

\[ [\hat{\theta} - t_{1-\alpha/2} \hat{\sigma}, \hat{\theta} - t_{\alpha/2} \hat{\sigma}] \]

where \(P(t \leq t_{1-\alpha/2}) = 1 - \alpha/2\) and \(P(t \leq t_{\alpha/2}) = \alpha/2\).
Visualization of 90% Gaussian Confidence Intervals

Figure 12.1: An *Introduction to the Bootstrap* (Efron & Tibshirani, 1993)
Proper Interpretation of CIs

Unfortunately, (frequentist) confidence intervals don’t have the interpretation that one might expect (or hope) for...

- Incorrect interpretations of CIs are prevalent in scientific papers

Interpreting a 99% Confidence Interval:

- **Correct**: through repeated samples, e.g., 99 out of 100 confidence intervals would be expected to contain true $\theta$ with $\alpha = .01$
- **Wrong**: through one sample; e.g., there is a 99% chance the confidence interval around my $\hat{\theta}$ contains the true $\theta$ (with $\alpha = .01$)
Proper Interpretation of CIs: Example

```r
> set.seed(1)
> n = 100
> B = 10^4
> X = replicate(B, rnorm(n))
> xbar = apply(X, 2, mean)
> xsd = apply(X, 2, sd)
> cilo = xbar - qt(.95, df=n-1)*(xsd/sqrt(n))
> ciup = xbar - qt(.05, df=n-1)*(xsd/sqrt(n))
> ci90 = (0>=cilo & 0<=ciup)
> mean(ci90)
[1] 0.902
> summary(ci90)
    Mode FALSE   TRUE   NA's
logical   980  9020      0
```
Two properties we can use to describe a confidence interval:

- **length** = $\hat{\theta}_{up} - \hat{\theta}_{lo}$
- **shape** = $\frac{\hat{\theta}_{up} - \hat{\theta}}{\hat{\theta} - \hat{\theta}_{lo}}$

Note that . . .

- **Length**: describes the overall size of the CI
- **Shape**: describes the asymmetry of the CI

$\text{shape} > 1$ indicates a greater distance between $\hat{\theta}_{up}$ and $\hat{\theta}$ than between $\hat{\theta}_{lo}$ and $\hat{\theta}$
Defining a Good Confidence Interval

What is a “good” bootstrap confidence interval?

- If an exact CI can be formed (e.g., sample mean), bootstrap CI should closely match exact CI
- If an exact CI cannot be formed (e.g., sample median), bootstrap CI should give accurate coverage probabilities

Note that a narrower CI is not necessarily a better CI. Length and shape are only important if the coverage probabilities are accurate.

Different bootstrap CI methods have different coverage accuracies.
First and Second Order Accurate

“Big Oh” notation: \( f(x) = O(g(x)) \) is read as “\( f(x) \) is big-oh of \( g(x) \)”

- \( f(x) = O(g(x)) \) as \( x \to \infty \) if and only if \( |f(x)| \leq h|g(x)| \) for all \( x \geq x_0 \) and some \( h > 0 \)
- For sufficiently large \( x \), the magnitude of \( f(x) \) is at most \( h \) times the magnitude of \( g(x) \)

A confidence interval is **first-order accurate** if the non-coverage probability on each side differs from the nominal value by \( O(n^{-1/2}) \).

- \( P(\theta < \hat{\theta}_{lo}) = \alpha + h_{lo}/\sqrt{n} \) and \( P(\theta > \hat{\theta}_{up}) = \alpha + h_{up}/\sqrt{n} \)

A confidence interval is **second-order accurate** if the non-coverage probability on each side differs from the nominal value by \( O(n^{-1}) \).

- \( P(\theta < \hat{\theta}_{lo}) = \alpha + h_{lo}/n \) and \( P(\theta > \hat{\theta}_{up}) = \alpha + h_{up}/n \)
Basic Bootstrap CIs
Basic Bootstrap CIs

$t$ with Bootstrap Standard Error

**t Confidence Interval with Bootstrap Standard Error**

Uses the classic CI formula with the bootstrap SE estimate:

\[
\text{Classic SE : } \hat{\theta} \pm t_{\alpha/2} \hat{\sigma}_{\hat{\theta}} \\
\text{Bootstrap SE : } \hat{\theta} \pm t_{\alpha/2} \hat{\sigma}_B
\]

No real benefit over the class $t$ interval using $\hat{\sigma}_{\hat{\theta}}$.

This CI procedure is only first-order accurate.
Properties of $t$ CI with Bootstrap SE

**Pros:**
- Simple to form and easy to understand
- Can be applied to situations where $\sigma_{\hat{\theta}}$ is difficult to derive

**Cons:**
- Tends to be too narrow for small $n$ because $\hat{\sigma}_B$ is too narrow.
- Comparable to using the MLE $\hat{\sigma} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (x_i - \bar{x})^2}$ instead of the unbiased estimate $s = \sqrt{\frac{1}{n-1} \sum_{i=1}^{n} (x_i - \bar{x})^2}$
- Can perform poorly if distribution is highly skewed
Another intuitive approach is to use the $100\alpha$-th and $100(1 - \alpha)$-th percentiles of bootstrap distribution of $\hat{\theta}$.

For example, if we have $B = 10,000$ bootstrap replications of $\hat{\theta}$

$$\hat{\theta}^{*}(1) \leq \hat{\theta}^{*}(2) \leq \cdots \leq \hat{\theta}^{*}(B)$$

we would define the 90% confidence interval using

$$[\hat{\theta}^{*}(500), \hat{\theta}^{*}(9500)] = [\hat{\theta}_{lo}, \hat{\theta}_{up}]$$

This CI procedure is only first-order accurate.
Properties of Bootstrap Percentile CIs

Pros:
- Simple to form and easy to understand
- Range preserving and transformation invariant
- Advantage over $t$ CI with bootstrap SE when data are skewed

Cons:
- Tends to be too narrow for small $n$ (worse than $t$ w/ bootstrap SE)
- Comparable to using $z_{\alpha/2}\hat{\sigma}/\sqrt{n}$ instead of $t_{\alpha/2} s/\sqrt{n}$
- Does partial skewness correction, which adds random variability
Example 1: Sample Mean CI

```r
> dev.new(width=5,height=5,noRStudioGD=TRUE)
> set.seed(1)
> n = 50
> x = rnorm(n,mean=1)
> bsamp = bootsamp(x)
> bse = bootse(bsamp,mean)
> mean(x)
[1] 1.100448
> c(mean(x)-qt(0.975,df=n-1)*sd(x)/sqrt(n),
+ mean(x)-qt(0.025,df=n-1) *sd(x)/sqrt(n))
[1] 0.8641687 1.3367278
> ci = quantile(bse$theta,c(0.025,0.975))
> ci
 2.5%    97.5%
0.8707902 1.3187369
> hist(bse$theta)
> lines(rep(ci[1],2),c(0,1500),lty=2)
> lines(rep(ci[2],2),c(0,1500),lty=2)
```

Histogram of `bse$theta`

Nathaniel E. Helwig (U of Minnesota)
Example 2: Sample Median CI

```r
> dev.new(width=5,height=5,noRStudioGD=TRUE)
> set.seed(1)
> n = 50
> x = rnorm(n,mean=1)
> bsamp = bootsamp(x)
> bse = bootse(bsamp,median)
> median(x)
[1] 1.129104
> ci = quantile(bse$theta,c(0.025,0.975))
> ci
     2.5%    97.5%
0.8972123 1.4874291
> hist(bse$theta)
> lines(rep(ci[1],2),c(0,4000),lty=2)
> lines(rep(ci[2],2),c(0,4000),lty=2)
```

Histogram of bse$theta

Nathaniel E. Helwig (U of Minnesota)
Example 3: Sample Variance CI

```r
> dev.new(width=5,height=5,noRStudioGD=TRUE)
> set.seed(1)
> n = 50
> x = rnorm(n,sd=2)
> bsamp = bootsamp(x)
> bse = bootse(bsamp,var)
> var(x)
[1] 2.764863
> c((n-1)*var(x)/qchisq(0.975,df=n-1),
+ (n-1)*var(x)/qchisq(0.025,df=n-1))
[1] 1.929274 4.293414
> ci = quantile(bse$theta,c(0.025,0.975))
> ci
 2.5% 97.5%
1.632342 3.948860
> hist(bse$theta)
> lines(rep(ci[1],2),c(0,2500),lty=2)
> lines(rep(ci[2],2),c(0,2500),lty=2)
```
Transformation Respecting Property of Percentile CIs

> set.seed(1)
> x = rnorm(50, mean=1)
> bsamp = bootsamp(x)
> bse = bootse(bsamp, function(x) exp(mean(x)))
> exp(mean(x))
[1] 3.005513
> mean(x)
[1] 1.100448
> ci = quantile(bse$theta, c(0.025, 0.975))
> ci
  2.5%   97.5%
2.388798 3.738696
> bse = bootse(bsamp, mean)
> quantile(bse$theta, c(0.025, 0.975))
  2.5%   97.5%
0.8707902 1.3187369
> log(ci)
  2.5%   97.5%
0.8707902 1.3187369
Better Bootstrap CIs
Can interpret $t$ interval as multiplying the length of a normal interval

$$\bar{x} \pm z_{\alpha/2} \hat{\sigma}/\sqrt{n}$$

by a factor $a_{\alpha,n} = (t_{\alpha/2}/z_{\alpha/2})(s/\hat{\sigma})$ where

- $s = \left\{ \frac{1}{n-1} \sum_{i=1}^{n} (x_i - \bar{x})^2 \right\}^{1/2}$
- $\hat{\sigma} = \left\{ \frac{1}{n} \sum_{i=1}^{n} (x_i - \bar{x})^2 \right\}^{1/2}$

Percentile CIs comparable to using $z_{\alpha/2} \hat{\sigma}/\sqrt{n}$ instead of $t_{\alpha/2} s/\sqrt{n}$, so we can use an adjustment to correct for narrowness bias.

- Don’t want to apply correction by multiplying both sides of interval by $a_{\alpha,n}$ because this would not be transformation invariant
- Instead, apply correction to quantiles of bootstrap distribution
If the bootstrap distribution is approximately normal, then

\[ \hat{F}^{-1}(\alpha/2) \approx \hat{\theta} + z_{\alpha/2} \hat{\sigma} / \sqrt{n} \]

and we want to find a modified quantile value \( \alpha' \) such that

\[ \hat{F}^{-1}(\alpha'/2) \approx \hat{\theta} + z_{\alpha'/2} \hat{\sigma} / \sqrt{n} \]
\[ = \hat{\theta} + t_{\alpha/2} s / \sqrt{n} \]

This implies that

\[ z_{\alpha'/2} = \sqrt{n/(n-1)} t_{\alpha/2} \]

so the modified quantile is

\[ \alpha'/2 = \Phi(\sqrt{n/(n-1)} t_{\alpha/2}) \]
Properties of Expanded Percentile CIs

Pros:
- Simple to form and easy to understand
- Range preserving and transformation invariant
- Corrects for narrowness bias of percentile CIs

Cons:
- Does partial skewness correction, which adds random variability
- No correction for bias, and doesn’t fully correct for skewness
- Only first-order accurate
Example 1: Sample Mean CI (revisited)

```r
> set.seed(1)
> n = 50
> x = rnorm(n, mean=1)
> bsamp = bootsamp(x)
> bse = bootse(bsamp, mean)
> mean(x)
[1] 1.100448
> c(mean(x) - qnorm(0.975) * sd(x) * sqrt((n-1)/n)/sqrt(n),
+ mean(x) - qnorm(0.025) * sd(x) * sqrt((n-1)/n)/sqrt(n))
[1] 0.8723180 1.3285790
> quantile(bse$theta, c(0.025, 0.975))
   2.5%     97.5%
0.8707902 1.3187369
> alphaD2 = pnorm(sqrt(n/(n-1)) * qt(.025, df=n-1))
> alphaD2
[1] 0.02117941
> c(mean(x) - qt(0.975, df=n-1) * sd(x)/sqrt(n),
+ mean(x) - qt(0.025, df=n-1) * sd(x)/sqrt(n))
[1] 0.8641687 1.3367278
> quantile(bse$theta, c(alphaD2, 1-alphaD2))
 2.117941% 97.88206%
0.862373 1.326751
```
Example 2: Sample Median CI (revisited)

```r
> set.seed(1)
> n = 50
> x = rnorm(n, mean=1)
> bsamp = bootsamp(x)
> bse = bootse(bsamp, median)
> median(x)
[1] 1.129104
> quantile(bse$theta, c(0.025, 0.975))
   2.5%   97.5%
0.8972123 1.4874291
> alphaD2 = pnorm(sqrt(n/(n-1))*qt(.025, df=n-1))
> alphaD2
[1] 0.02117941
> quantile(bse$theta, c(alphaD2, 1-alphaD2))
2.117941% 97.88206%
 0.892433 1.487429
```
Example 3: Sample Variance CI (revisited)

```r
> set.seed(1)
> n = 50
> x = rnorm(n, sd=2)
> bsamp = bootsamp(x)
> bse = bootse(bsamp, var)
> var(x)
[1] 2.764863
> quantile(bse$theta, c(0.025, 0.975))
  2.5%     97.5%  
1.632342 3.948860
> alphaD2 = pnorm(sqrt(n/(n-1)) * qt(.025, df=n-1))
> alphaD2
[1] 0.02117941
> quantile(bse$theta, c(alphaD2, 1-alphaD2))
  2.117941% 97.88206%
1.600105 3.997069
```
Given $B$ bootstrap samples $x_1^*, \ldots, x_B^*$, we compute

$$t_b^* = \frac{\hat{\theta}_b^* - \hat{\theta}}{\hat{\sigma}_b}$$

where $\hat{\sigma}_b$ is standard error for $b$-th bootstrap sample.

Note that $\hat{\sigma}_b$ may not have a closed form solution:

- If $\hat{\theta}$ is sample mean, then $\hat{\sigma}_b = \left\{ \sum_{i=1}^n (x_{i(b)}^* - \bar{x}_b^*)^2 / n^2 \right\}^{1/2}$
- For other statistics, need bootstrap SE for each bootstrap sample
Given $t^*_b$ for $b \in \{1, \ldots, B\}$, define the $\alpha$-th quantile $q_\alpha$ as

$$\frac{\# \{ t^*_b \leq q_\alpha \}}{B} = \alpha$$

and note that we have

$$1 - \alpha = P(q_{\alpha/2} < t^*_b < q_{1-\alpha/2})$$
$$\approx P(q_{\alpha/2} < t < q_{1-\alpha/2})$$
$$= P(q_{\alpha/2}\hat{\sigma}_B < \hat{\theta} - \theta < q_{1-\alpha/2}\hat{\sigma}_B)$$
$$= P(\hat{\theta} - q_{\alpha/2}\hat{\sigma}_B > \theta > \hat{\theta} - q_{1-\alpha/2}\hat{\sigma}_B)$$

Form the “bootstrap-t” interval: $[\hat{\theta} - q_{1-\alpha/2}\hat{\sigma}_B, \hat{\theta} - q_{\alpha/2}\hat{\sigma}_B]$
Properties of Bootstrap $t$-Table CIs

Pros:
- Simple idea with intuitive procedure
- Works well for location parameters
- Second-order accurate

Cons:
- Not range preserving or transformation invariant
- Formation of $\hat{\sigma}_b$ requires \textit{iterated bootstrap}
- Doesn’t work as well for correlation/association measures
Example 1: Sample Mean CI (revisited)

```r
> set.seed(1)
> n = 50
> x = rnorm(n, mean=1)
> bsamp = bootsamp(x)
> bse = bootse(bsamp, mean)
> bsampSE = apply(bsamp, 2, sd) * sqrt((n-1)/n) / sqrt(n)
> theta = mean(x)
> Z = (bse$theta - theta) / bsampSE
> cval = quantile(Z, probs=c(0.025, 0.975))

# bootstrap t-table:
> c(theta - cval[2]*bse$se, theta - cval[1]*bse$se)
   97.5%     2.5%
0.8538653 1.3189295

# percentile:
> quantile(bse$theta, c(0.025, 0.975))
  2.5%     97.5%
0.8707902 1.3187369
```
Example 2: Sample Median CI (revisited)

```r
> set.seed(1)
> n = 50
> x = rnorm(n, mean=1)
> bsamp = bootsamp(x)
> bse = bootse(bsamp, median)
> theta = median(x)
> bsampSE = rep(0, ncol(bsamp))
> for(k in 1:ncol(bsamp)) {
+    bsampSE[k] = bootse(bootsamp(bsamp[,k], nsamp=2000), median)$se
+  }
> Z = (bse$theta - theta) / bsampSE
> cval = quantile(Z, probs=c(0.025, 0.975))

# bootstrap t-table
> c(theta - cval[2]*bse$se, theta - cval[1]*bse$se)
   97.5%  2.5%
0.6226602 1.5705624

# percentile
> quantile(bse$theta, c(0.025, 0.975))
   2.5%   97.5%
0.8972123 1.4874291
```
Example 3: Sample Variance CI (revisited)

```r
> set.seed(1)
> n = 50
> x = rnorm(n, sd=2)
> bsamp = bootsamp(x)
> bse = bootse(bsamp, var)
> theta = var(x)
> bsampSE = rep(0, ncol(bsamp))
> for(k in 1:ncol(bsamp)){
+ bsampSE[k] = bootse(bootsamp(bsamp[,k], nsamp=2000), var)$se
+ }
> Z = (bse$theta - theta) / bsampSE
> cval = quantile(Z, probs=c(0.025, 0.975))

# bootstrap t-table
> c(theta - cval[2]*bse$se, theta - cval[1]*bse$se)
   97.5%   2.5%
1.796957 4.971694

# percentile
> quantile(bse$theta, c(0.025, 0.975))
   2.5%   97.5%
1.632342 3.948860
```
Better Bootstrap CIs

Bias Corrected and Accelerated ($BC_a$) Bootstrap CIs

$BC_a$ Bootstrap CIs

$BC_a$ intervals use percentiles of bootstrap distribution, but they do not necessarily use the $100\alpha$-th and $100(1 - \alpha)$-th percentiles.

- Depend on acceleration parameter $\hat{a}$
- Depend on bias-correction factor $\hat{z}_0$

$BC_a$ intervals have the form:  

$$[\hat{\theta}^*_1, \hat{\theta}^*_2] = [\hat{\theta}_{lo}, \hat{\theta}_{up}]$$

- $\alpha_1 = \Phi \left( \hat{z}_0 + \frac{\hat{z}_0 + z(\alpha)}{1 - \hat{a}(\hat{z}_0 + z(\alpha))} \right)$
- $\alpha_2 = \Phi \left( \hat{z}_0 + \frac{\hat{z}_0 + z(1 - \alpha)}{1 - \hat{a}(\hat{z}_0 + z(1 - \alpha))} \right)$
- $\Phi(\cdot)$ is the cdf of standard normal ($\text{pnorm}$)
- $z(\alpha)$ is the $100\alpha$-th percentile of standard normal
Estimating Bias-Correction Factor

Note that if \( \hat{a} = \hat{z}_0 = 0 \), then...

- \( \alpha_1 = \Phi(z(\alpha)) = \alpha \)
- \( \alpha_2 = \Phi(z(1-\alpha)) = 1 - \alpha \)

and the \( BC_a \) interval is the same as the percentile interval.

The bias-correction factor is estimated as

\[
\hat{z}_0 = \Phi^{-1} \left( \frac{\# \{ \hat{\theta}^*_b < \hat{\theta} \}}{B} \right)
\]

where \( \Phi^{-1}(\cdot) \) is the inverse cdf of standard normal (\texttt{qnorm})

- \( \hat{z}_0 \) measures median bias of \( \hat{\theta}^* \), i.e., difference between median(\( \hat{\theta}^*_b \)) and \( \hat{\theta} \)
Estimating Acceleration Factor

The acceleration factor can be calculated using a jackknife approach:

- Reminder: \( \hat{\theta}_i = s(x_i) \) is estimate of \( \theta \) holding out \( x_i \)
- Reminder: \( \hat{\theta}_a = \frac{1}{n} \sum_{i=1}^{n} \hat{\theta}_i \) is the average of jackknife estimates

The acceleration factor can be expressed as

\[
\hat{a} = \frac{\sum_{i=1}^{n} (\hat{\theta}_a - \hat{\theta}_i)^3}{6\{\sum_{i=1}^{n} (\hat{\theta}_a - \hat{\theta}_i)^2\}^{3/2}}
\]

which estimates the rate of change of the standard error of \( \hat{\theta} \) with respect to the true parameter \( \theta \).

- Assuming that \( \hat{\theta} \sim N(\theta, \sigma^2_{\hat{\theta}}) \) assumes that \( \sigma_{\hat{\theta}} \) is the same for all \( \theta \)
- \( \hat{a} \) corrects for this (possibly unrealistic) assumption
Properties of Bias-Corrected and Accelerated CIs

Pros:
- Range preserving and transformation invariant
- Works well for a variety parameters
- Second-order accurate

Cons:
- Requires estimation of acceleration and bias-correction
- Less intuitive than other methods
We could easily write our own $BC_a$ confidence interval function using the bootstrap and jackknife functions we’ve already created.

- We can calculate $\hat{z}_0$ using output of `bootse`
- We can calculate $\hat{a}$ using output of `jackse`

But there is already the `bcanon` function (in `bootstrap` package).

- Takes in $x$, `nboot`, and `theta` as necessary inputs
- Outputs `confpoint`, `z0`, `acc`, and `u` (jackknife influence)
Our BCa confidence interval function would look something like...

```r
bcafun <- function(x, nboot, theta, ..., alpha = 0.05) {
  theta.hat = theta(x)
  nx = length(x)
  bse = bootse(bootsamp(x, nboot), theta, ...)
  jse = jackse(jacksamp(x), theta, ...)
  z0 = qnorm(sum(bse$theta < theta.hat)/nboot)
  atop = sum((mean(jse$theta) - jse$theta)^3)
  abot = 6*((jse$se^2)*nx/(nx-1))^(3/2)
  ahat = atop/abot
  alpha1 = pnorm(z0 + (z0 + qnorm(alpha))/(1-ahat*(z0 + qnorm(alpha))))
  alpha2 = pnorm(z0 + (z0 + qnorm(1-alpha))/(1-ahat*(z0 + qnorm(1-alpha))))
  confpoint = quantile(bse$theta, probs = c(alpha1, alpha2))
  list(confpoint = confpoint, z0 = z0, acc = ahat, u = (jse$theta - theta.hat),
       theta = bse$theta, se = bse$se)
}
```

Note that this forms a 100\((1 - 2\alpha)\)% confidence interval.
Example 1: Sample Mean (revisited)

```r
> library(bootstrap)
> set.seed(1)
> n = 50
> x = rnorm(n, mean=1)
> c(mean(x) - qt(0.975, df=n-1) * sd(x) / sqrt(n),
+    mean(x) - qt(0.025, df=n-1) * sd(x) / sqrt(n))
[1] 0.8641687 1.3367278
> mybca = bcafun(x, 10000, mean, alpha=0.025)
> quantile(mybca$theta, probs=c(0.025, 0.975))
     2.5%  97.5%
0.8707902 1.3187369
> mybca$conf
 1.933122% 96.8455%
0.8573764 1.3077089
> bca = bcanon(x, 10000, mean, alpha=c(0.025, 0.975))
> bca$conf
    alpha   bca    point
[1,] 0.025 0.8582165
[2,] 0.975 1.3118539
```
Better Bootstrap CIs
Bias Corrected and Accelerated (BCa) Bootstrap CIs

Example 2: Sample Median (revisited)

```r
> library(bootstrap)
> set.seed(1)
> n = 50
> x = rnorm(n,mean=1)
> mybca = bcafun(x,10000,median,alpha=0.025)
> quantile(mybca$theta,c(0.025,0.975))

         2.5%      97.5%
0.8972123  1.4874291
> mybca$conf
1.710689%  96.42576%
  0.892433  1.452685
> bca = bcanon(x,10000,median,alpha=c(0.025,0.975))
> bca$conf
     alpha bca point
  [1,] 0.025 0.8880815
  [2,] 0.975 1.4732532
```
Example 3: Sample Variance (revisited)

```r
> library(bootstrap)
> set.seed(1)
> n = 50
> x = rnorm(n, sd=2)
> mybca = bcafun(x, 10000, var, alpha=0.025)
> quantile(mybca$theta, c(0.025, 0.975))
     2.5%    97.5%
1.632342 3.948860
> mybca$conf
  7.083711% 99.53268%
  1.864701 4.409839
> bca = bcanon(x, 10000, var, alpha=c(0.025, 0.975))
> bca$conf
   alpha  bca  point
[1,] 0.025 1.863058
[2,] 0.975 4.389655
```
Example 1: Sample Mean Results Summary

```r
> tab.mean = rbind(standZ, standT, prcnt.mean, eprcnt.mean, bootT.mean, bca.mean)
> rownames(tab.mean) = c("standZ", "standT", "prcnt", "eprcnt", "bootT", "bca")
> round(tab.mean, 4)
     2.5% 97.5%
standZ 0.8723 1.3286
standT 0.8642 1.3367
prcnt  0.8708 1.3187
eprcnt 0.8624 1.3268
bootT  0.8539 1.3189
bca    0.8574 1.3077
```
Example 2: Sample Median Results Summary

```r
> tab.med = rbind(prcnt.med, eprcnt.med, bootT.med, bca.med)
> rownames(tab.med) = c("prcnt", "eprcnt", "bootT", "bca")
> round(tab.med, 4)
  2.5%   97.5%
prcnt  0.8972  1.4874
eprcnt  0.8924  1.4874
bootT  0.6227  1.5706
bca    0.8924  1.4527
```
Example 3: Sample Variance Results Summary

```r
> tab.var = rbind(standZ.var, prcnt.var, 
+                 eprcnt.var, bootT.var, bca.var)
> rownames(tab.var) = c("standZ", "prcnt", 
+                       "eprcnt", "bootT", "bca")
> round(tab.var, 4)

             2.5%   97.5%
standZ     1.9293  4.2934
prcnt      1.6323  3.9489
eprcnt     1.6001  3.9971
bootT      1.7970  4.9717
bca        1.8647  4.4098
```