Bootstrap Inference

Bootstrap Inference

Except for the classical normal linear model, the way to make inferences has traditionally been to rely on asymptotic theory.

Bootstrap Inference

Except for the classical normal linear model, the way to make inferences has traditionally been to rely on asymptotic theory.

But this not does not always work well. It sometimes works dreadfully! **Bootstrap methods** very often perform better.

Bootstrap Inference

Except for the classical normal linear model, the way to make inferences has traditionally been to rely on asymptotic theory.

But this not does not always work well. It sometimes works dreadfully! **Bootstrap methods** very often perform better.

- All bootstrap methods involve generating a large number (B) of simulated samples, called **bootstrap samples**.

Bootstrap Inference

Except for the classical normal linear model, the way to make inferences has traditionally been to rely on asymptotic theory.

But this not does not always work well. It sometimes works dreadfully! **Bootstrap methods** very often perform better.

- All bootstrap methods involve generating a large number (B) of simulated samples, called **bootstrap samples**.
- The model is then estimated using every bootstrap sample, and functions of the B bootstrap estimates are used for inference.

Bootstrap Inference

Except for the classical normal linear model, the way to make inferences has traditionally been to rely on asymptotic theory.

But this not does not always work well. It sometimes works dreadfully! **Bootstrap methods** very often perform better.

- All bootstrap methods involve generating a large number (B) of simulated samples, called **bootstrap samples**.
- The model is then estimated using every bootstrap sample, and functions of the B bootstrap estimates are used for inference.

The term **bootstrap**, which was introduced by Efron (1979), is taken from the phrase "to pull oneself up by one's own bootstraps."

Bootstrap Inference

Except for the classical normal linear model, the way to make inferences has traditionally been to rely on asymptotic theory.

But this not does not always work well. It sometimes works dreadfully! **Bootstrap methods** very often perform better.

- All bootstrap methods involve generating a large number (B) of simulated samples, called **bootstrap samples**.
- The model is then estimated using every bootstrap sample, and functions of the B bootstrap estimates are used for inference.

The term **bootstrap**, which was introduced by Efron (1979), is taken from the phrase “to pull oneself up by one’s own bootstraps.”

Some authors refer to **the bootstrap**, but it is not a single procedure.

Bootstrap Inference

Except for the classical normal linear model, the way to make inferences has traditionally been to rely on asymptotic theory.

But this not does not always work well. It sometimes works dreadfully! **Bootstrap methods** very often perform better.

- All bootstrap methods involve generating a large number (B) of simulated samples, called **bootstrap samples**.
- The model is then estimated using every bootstrap sample, and functions of the B bootstrap estimates are used for inference.

The term **bootstrap**, which was introduced by Efron (1979), is taken from the phrase "to pull oneself up by one's own bootstraps."

Some authors refer to **the bootstrap**, but it is not a single procedure.

Bootstrap samples can be generated in many different ways, and there are many procedures for making inferences from bootstrap estimates.

Resampling

Resampling

The first method for generating bootstrap samples was to **resample** the data (Efron, 1979). This assumes that they are i.i.d.

Resampling

The first method for generating bootstrap samples was to **resample** the data (Efron, 1979). This assumes that they are i.i.d.

Suppose we have a sample of N observations x_i in a vector $\boldsymbol{x}$. Formally, each bootstrap sample is a draw from the EDF of the x_i:

$$x_i^* \sim \mathrm{EDF}(\boldsymbol{x}). \tag{1}$$

Resampling

The first method for generating bootstrap samples was to **resample** the data (Efron, 1979). This assumes that they are i.i.d.

Suppose we have a sample of N observations x_i in a vector $\boldsymbol{x}$. Formally, each bootstrap sample is a draw from the EDF of the x_i:

$$x_i^* \sim \text{EDF}(\boldsymbol{x}). \tag{1}$$

Recall that the EDF assigns probability $1/N$ to each observation.

Resampling

The first method for generating bootstrap samples was to **resample** the data (Efron, 1979). This assumes that they are i.i.d.

Suppose we have a sample of N observations x_i in a vector $\boldsymbol{x}$. Formally, each bootstrap sample is a draw from the EDF of the x_i:

$$x_i^* \sim \text{EDF}(\boldsymbol{x}). \tag{1}$$

Recall that the EDF assigns probability $1/N$ to each observation.

Metaphorically speaking, we throw all the x_i into a hat and then randomly pull them out one at a time, with replacement. This is a very simple **bootstrap DGP**.

Resampling

The first method for generating bootstrap samples was to **resample** the data (Efron, 1979). This assumes that they are i.i.d.

Suppose we have a sample of N observations x_i in a vector $\boldsymbol{x}$. Formally, each bootstrap sample is a draw from the EDF of the x_i:

$$x_i^* \sim \text{EDF}(\boldsymbol{x}). \tag{1}$$

Recall that the EDF assigns probability $1/N$ to each observation.

Metaphorically speaking, we throw all the x_i into a hat and then randomly pull them out one at a time, with replacement. This is a very simple **bootstrap DGP**.

Each bootstrap sample contains some of the x_i exactly once, some of them more than once, and some of them not at all.

Resampling

The first method for generating bootstrap samples was to **resample** the data (Efron, 1979). This assumes that they are i.i.d.

Suppose we have a sample of N observations x_i in a vector $\boldsymbol{x}$. Formally, each bootstrap sample is a draw from the EDF of the x_i:

$$x_i^* \sim \text{EDF}(\boldsymbol{x}). \tag{1}$$

Recall that the EDF assigns probability $1/N$ to each observation.

Metaphorically speaking, we throw all the x_i into a hat and then randomly pull them out one at a time, with replacement. This is a very simple **bootstrap DGP**.

Each bootstrap sample contains some of the x_i exactly once, some of them more than once, and some of them not at all.

The probability that a bootstrap sample omits x_i is quite large.

A given x_i does not appear in a bootstrap sample with probability

$$\Pr(\#x_i = 0) = \left(\frac{N-1}{N}\right)^N. \tag{2}$$

A given x_i does not appear in a bootstrap sample with probability

$$\Pr(\#x_i = 0) = \left(\frac{N-1}{N}\right)^N. \tag{2}$$

A first-order Taylor expansion to the log of (2), which converges to a constant as $N \to \infty$, yields

$$\log\big(\Pr(\#x_i = 0)\big) = N \log(1 - 1/N) \cong N(-1/N) = -1. \tag{3}$$

A given x_i does not appear in a bootstrap sample with probability

$$\Pr(\#x_i = 0) = \left(\frac{N-1}{N}\right)^N. \tag{2}$$

A first-order Taylor expansion to the log of (2), which converges to a constant as $N \to \infty$, yields

$$\log\big(\Pr(\#x_i = 0)\big) = N \log(1 - 1/N) \cong N(-1/N) = -1. \tag{3}$$

Therefore,

$$\Pr(\#x_i = 0) \cong \exp(-1) = 1/(2.71828) = 0.36788. \tag{4}$$

A given x_i does not appear in a bootstrap sample with probability

$$\Pr(\#x_i = 0) = \left(\frac{N-1}{N}\right)^N. \tag{2}$$

A first-order Taylor expansion to the log of (2), which converges to a constant as $N \to \infty$, yields

$$\log\big(\Pr(\#x_i = 0)\big) = N \log(1 - 1/N) \cong N(-1/N) = -1. \tag{3}$$

Therefore,

$$\Pr(\#x_i = 0) \cong \exp(-1) = 1/(2.71828) = 0.36788. \tag{4}$$

Any x_i will be missing from almost 37% of the bootstrap samples.

A given x_i does not appear in a bootstrap sample with probability

$$\Pr(\#x_i = 0) = \left(\frac{N-1}{N}\right)^N. \tag{2}$$

A first-order Taylor expansion to the log of (2), which converges to a constant as $N \to \infty$, yields

$$\log\big(\Pr(\#x_i = 0)\big) = N \log(1 - 1/N) \cong N(-1/N) = -1. \tag{3}$$

Therefore,

$$\Pr(\#x_i = 0) \cong \exp(-1) = 1/(2.71828) = 0.36788. \tag{4}$$

Any x_i will be missing from almost 37% of the bootstrap samples. Oddly, it will also appear exactly once with the same probability!

A given x_i does not appear in a bootstrap sample with probability

$$\Pr(\#x_i = 0) = \left(\frac{N-1}{N}\right)^N. \tag{2}$$

A first-order Taylor expansion to the log of (2), which converges to a constant as $N \to \infty$, yields

$$\log\big(\Pr(\#x_i = 0)\big) = N \log(1 - 1/N) \cong N(-1/N) = -1. \tag{3}$$

Therefore,

$$\Pr(\#x_i = 0) \cong \exp(-1) = 1/(2.71828) = 0.36788. \tag{4}$$

Any x_i will be missing from almost 37% of the bootstrap samples. Oddly, it will also appear exactly once with the same probability!

Resampling necessarily involves replacement. Without it, every bootstrap sample would just be the actual sample reordered.

The total number of possible bootstrap samples is N^N. Each occurs with probability N^{-N}.

The total number of possible bootstrap samples is N^N. Each occurs with probability N^{-N}.

The number of distinct samples is

$$ {}_{2N-1}C_N = \frac{(2N-1)!}{N!(N-1)!}. \quad (5) $$

The total number of possible bootstrap samples is N^N. Each occurs with probability N^{-N}.

The number of distinct samples is

$$ {}_{2N-1}C_N = \frac{(2N-1)!}{N!(N-1)!}. \tag{5} $$

This is usually a very big number, but not nearly as big as N^N.

The total number of possible bootstrap samples is N^N. Each occurs with probability N^{-N}.

The number of distinct samples is

$$ {}_{2N-1}C_N = \frac{(2N-1)!}{N!(N-1)!}. \tag{5} $$

This is usually a very big number, but not nearly as big as N^N.

Here is how resampling from a sample $x_i, i = 1, \ldots, N$, works:

The total number of possible bootstrap samples is N^N. Each occurs with probability N^{-N}.

The number of distinct samples is

$$ {}_{2N-1}C_N = \frac{(2N-1)!}{N!(N-1)!}. \quad (5) $$

This is usually a very big number, but not nearly as big as N^N.

Here is how resampling from a sample $x_i, i = 1, \ldots, N$, works:

1. Divide the interval $[0, 1]$ into N subintervals of length $1/N$, and number them from 1 to N.

The total number of possible bootstrap samples is N^N. Each occurs with probability N^{-N}.

The number of distinct samples is

$$ {}_{2N-1}C_N = \frac{(2N-1)!}{N!(N-1)!}. \tag{5} $$

This is usually a very big number, but not nearly as big as N^N.

Here is how resampling from a sample $x_i, i = 1, \ldots, N$, works:

1. Divide the interval $[0, 1]$ into N subintervals of length $1/N$, and number them from 1 to N.
2. Draw a random number η from the $\mathrm{U}(0, 1)$ distribution.

The total number of possible bootstrap samples is N^N. Each occurs with probability N^{-N}.

The number of distinct samples is

$$ {}_{2N-1}C_N = \frac{(2N-1)!}{N!(N-1)!}. \tag{5} $$

This is usually a very big number, but not nearly as big as N^N.

Here is how resampling from a sample $x_i, i = 1, \ldots, N$, works:

1. Divide the interval $[0, 1]$ into N subintervals of length $1/N$, and number them from 1 to N.
2. Draw a random number η from the $\mathrm{U}(0, 1)$ distribution.
3. When η falls into the l^{th} subinterval, put x_l into the bootstrap sample that is being created.

The total number of possible bootstrap samples is N^N. Each occurs with probability N^{-N}.

The number of distinct samples is

$$ {}_{2N-1}C_N = \frac{(2N-1)!}{N!(N-1)!}. \tag{5} $$

This is usually a very big number, but not nearly as big as N^N.

Here is how resampling from a sample $x_i, i = 1, \ldots, N$, works:

1. Divide the interval $[0, 1]$ into N subintervals of length $1/N$, and number them from 1 to N.
2. Draw a random number η from the $\mathrm{U}(0, 1)$ distribution.
3. When η falls into the l^{th} subinterval, put x_l into the bootstrap sample that is being created.
4. Repeat steps 2 and 3 above N times to generate a single bootstrap sample of N observations.

Suppose that $N = 10$, and the ten observations y_i are

$$6.45,\ 1.28,\ -3.48,\ 2.44,\ -5.17,\ -1.67,\ -2.03,\ 3.58,\ 0.74,\ -2.14.$$

Suppose that $N = 10$, and the ten observations y_i are

$$6.45,\ 1.28,\ -3.48,\ 2.44,\ -5.17,\ -1.67,\ -2.03,\ 3.58,\ 0.74,\ -2.14.$$

Now suppose that, when forming one of the bootstrap samples, the ten drawings from the $\mathrm{U}(0,1)$ distribution happen to be

$$0.631,\ 0.277,\ 0.745,\ 0.202,\ 0.914,\ 0.136,\ 0.851,\ 0.878,\ 0.120,\ 0.259.$$

Suppose that $N = 10$, and the ten observations y_i are

$$6.45,\ 1.28,\ -3.48,\ 2.44,\ -5.17,\ -1.67,\ -2.03,\ 3.58,\ 0.74,\ -2.14.$$

Now suppose that, when forming one of the bootstrap samples, the ten drawings from the $\mathrm{U}(0,1)$ distribution happen to be

$$0.631,\ 0.277,\ 0.745,\ 0.202,\ 0.914,\ 0.136,\ 0.851,\ 0.878,\ 0.120,\ 0.259.$$

This implies that the ten index values are

$$7,\ 3,\ 8,\ 3,\ 10,\ 2,\ 9,\ 9,\ 2,\ 3.$$

Suppose that $N = 10$, and the ten observations y_i are

$$6.45,\ 1.28,\ -3.48,\ 2.44,\ -5.17,\ -1.67,\ -2.03,\ 3.58,\ 0.74,\ -2.14.$$

Now suppose that, when forming one of the bootstrap samples, the ten drawings from the $\mathrm{U}(0,1)$ distribution happen to be

$$0.631,\ 0.277,\ 0.745,\ 0.202,\ 0.914,\ 0.136,\ 0.851,\ 0.878,\ 0.120,\ 0.259.$$

This implies that the ten index values are

$$7,\ 3,\ 8,\ 3,\ 10,\ 2,\ 9,\ 9,\ 2,\ 3.$$

Therefore, this bootstrap sample consists of

$$-2.03,\ -3.48,\ 3.58,\ -3.48,\ -2.14,\ 1.28,\ 0.74,\ 0.74,\ 1.28,\ -3.48.$$

Suppose that $N = 10$, and the ten observations y_i are

$$6.45,\ 1.28,\ -3.48,\ 2.44,\ -5.17,\ -1.67,\ -2.03,\ 3.58,\ 0.74,\ -2.14.$$

Now suppose that, when forming one of the bootstrap samples, the ten drawings from the $\mathrm{U}(0, 1)$ distribution happen to be

$$0.631,\ 0.277,\ 0.745,\ 0.202,\ 0.914,\ 0.136,\ 0.851,\ 0.878,\ 0.120,\ 0.259.$$

This implies that the ten index values are

$$7,\ 3,\ 8,\ 3,\ 10,\ 2,\ 9,\ 9,\ 2,\ 3.$$

Therefore, this bootstrap sample consists of

$$-2.03,\ -3.48,\ 3.58,\ -3.48,\ -2.14,\ 1.28,\ 0.74,\ 0.74,\ 1.28,\ -3.48.$$

Some of the observationss appear just once in this particular sample, but numbers 2, 3, and 9 appear more than once, and numbers 1, 4, 5, and 6 do not appear at all.

Random Number Generators

Random Number Generators

A **random number generator**, or **RNG**, is a program for generating a sequence of pseudo-random numbers, say η_j for $j = 1, \ldots, J$.

Random Number Generators

A **random number generator**, or **RNG**, is a program for generating a sequence of pseudo-random numbers, say η_j for $j = 1, \ldots, J$.

Most RNGs generate the η_j from the $\mathrm{U}(0, 1)$ distribution. They can then be transformed into drawings from other distributions.

Random Number Generators

A **random number generator**, or **RNG**, is a program for generating a sequence of pseudo-random numbers, say η_j for $j = 1, \ldots, J$.

Most RNGs generate the η_j from the $\mathrm{U}(0, 1)$ distribution. They can then be transformed into drawings from other distributions.

- An RNG starts with a (large) positive integer z_0 called the **seed**, or maybe a vector of seeds, to determine the sequence of η_j.

Random Number Generators

A **random number generator**, or **RNG**, is a program for generating a sequence of pseudo-random numbers, say η_j for $j = 1, \ldots, J$.

Most RNGs generate the η_j from the $\mathrm{U}(0, 1)$ distribution. They can then be transformed into drawings from other distributions.

- An RNG starts with a (large) positive integer z_0 called the **seed**, or maybe a vector of seeds, to determine the sequence of η_j.
- Most packages pick a seed based on the system clock if one is not provided, so that different sequences are generated each time.

Random Number Generators

A **random number generator**, or **RNG**, is a program for generating a sequence of pseudo-random numbers, say η_j for $j = 1, \ldots, J$.

Most RNGs generate the η_j from the $\mathrm{U}(0, 1)$ distribution. They can then be transformed into drawings from other distributions.

- An RNG starts with a (large) positive integer z_0 called the **seed**, or maybe a vector of seeds, to determine the sequence of η_j.
- Most packages pick a seed based on the system clock if one is not provided, so that different sequences are generated each time.
- To use the same sequence of random numbers more than once, we must give the RNG the same seed z_0 every time it is called.

Random Number Generators

A **random number generator**, or **RNG**, is a program for generating a sequence of pseudo-random numbers, say η_j for $j = 1, \ldots, J$.

Most RNGs generate the η_j from the $\mathrm{U}(0, 1)$ distribution. They can then be transformed into drawings from other distributions.

- An RNG starts with a (large) positive integer z_0 called the **seed**, or maybe a vector of seeds, to determine the sequence of η_j.
- Most packages pick a seed based on the system clock if one is not provided, so that different sequences are generated each time.
- To use the same sequence of random numbers more than once, we must give the RNG the same seed z_0 every time it is called.

We can obtain standard normal random numbers by using the fact that, if η is distributed as $\mathrm{U}(0, 1)$, then $\Phi^{-1}(\eta)$ is distributed as $\mathrm{N}(0, 1)$. However, much faster methods are available.

Regression Models

Regression Models

Even for models as simple as the linear regression model, there are many ways to specify a bootstrap DGP.

Regression Models

Even for models as simple as the linear regression model, there are many ways to specify a bootstrap DGP.

For a regression model, the oldest approach is the **pairs bootstrap**.

Regression Models

Even for models as simple as the linear regression model, there are many ways to specify a bootstrap DGP.

For a regression model, the oldest approach is the **pairs bootstrap**.

We resample the data, keeping the dependent and independent variables together in pairs.

Regression Models

Even for models as simple as the linear regression model, there are many ways to specify a bootstrap DGP.

For a regression model, the oldest approach is the **pairs bootstrap**.

We resample the data, keeping the dependent and independent variables together in pairs.

Each row $[y_i \;\; \boldsymbol{X}_i]$ is implicitly assumed to be an independent random drawing from an unknown multivariate distribution.

Regression Models

Even for models as simple as the linear regression model, there are many ways to specify a bootstrap DGP.

For a regression model, the oldest approach is the **pairs bootstrap**.

We resample the data, keeping the dependent and independent variables together in pairs.

Each row $[y_i \;\; \boldsymbol{X}_i]$ is implicitly assumed to be an independent random drawing from an unknown multivariate distribution.

For the linear regression model, this amounts to forming the matrix

$$\boldsymbol{Z} \equiv [\boldsymbol{y} \quad \boldsymbol{X}], \tag{6}$$

with typical row $\boldsymbol{Z}_i = [y_i \;\; \boldsymbol{X}_i]$, and then resampling the rows of $\boldsymbol{Z}$.

Regression Models

Even for models as simple as the linear regression model, there are many ways to specify a bootstrap DGP.

For a regression model, the oldest approach is the **pairs bootstrap**.

We resample the data, keeping the dependent and independent variables together in pairs.

Each row $[y_i \;\; \boldsymbol{X}_i]$ is implicitly assumed to be an independent random drawing from an unknown multivariate distribution.

For the linear regression model, this amounts to forming the matrix

$$\boldsymbol{Z} \equiv [\boldsymbol{y} \quad \boldsymbol{X}], \tag{6}$$

with typical row $\boldsymbol{Z}_i = [y_i \;\; \boldsymbol{X}_i]$, and then resampling the rows of $\boldsymbol{Z}$.

Every observation of every bootstrap sample is simply $\boldsymbol{Z}_j^*$, for $j \in \{1, \ldots, N\}$, a randomly chosen row of the matrix $\boldsymbol{Z}$.

If $\boldsymbol{y}^{*b}$ and $\boldsymbol{X}^{*b}$ are the data for the b^{th} bootstrap sample, then

$$\hat{\boldsymbol{\beta}}^{*b} = \left((\boldsymbol{X}^{*b})^\top \boldsymbol{X}^{*b}\right)^{-1} (\boldsymbol{X}^{*b})^\top \boldsymbol{y}^{*b}. \tag{7}$$

If $\boldsymbol{y}^{*b}$ and $\boldsymbol{X}^{*b}$ are the data for the b^{th} bootstrap sample, then

$$\hat{\boldsymbol{\beta}}^{*b} = \left((\boldsymbol{X}^{*b})^\top \boldsymbol{X}^{*b}\right)^{-1} (\boldsymbol{X}^{*b})^\top \boldsymbol{y}^{*b}. \tag{7}$$

The pairs bootstrap can be used for any model where the data are believed to be independent across observations.

If $\boldsymbol{y}^{*b}$ and $\boldsymbol{X}^{*b}$ are the data for the b^{th} bootstrap sample, then

$$\hat{\boldsymbol{\beta}}^{*b} = \big((\boldsymbol{X}^{*b})^\top \boldsymbol{X}^{*b}\big)^{-1} (\boldsymbol{X}^{*b})^\top \boldsymbol{y}^{*b}. \tag{7}$$

The pairs bootstrap can be used for any model where the data are believed to be independent across observations.

This assumption would not be reasonable for data with any sort of serial dependence or clustered disturbances.

If $\boldsymbol{y}^{*b}$ and $\boldsymbol{X}^{*b}$ are the data for the b^{th} bootstrap sample, then

$$\hat{\boldsymbol{\beta}}^{*b} = \big((\boldsymbol{X}^{*b})^\top \boldsymbol{X}^{*b}\big)^{-1} (\boldsymbol{X}^{*b})^\top \boldsymbol{y}^{*b}. \tag{7}$$

The pairs bootstrap can be used for any model where the data are believed to be independent across observations.

This assumption would not be reasonable for data with any sort of serial dependence or clustered disturbances.

- It is natural to use pairs bootstrap with cross-section data, but it can also be used with some models that use time-series data.

If $\boldsymbol{y}^{*b}$ and $\boldsymbol{X}^{*b}$ are the data for the b^{th} bootstrap sample, then

$$\hat{\boldsymbol{\beta}}^{*b} = \left((\boldsymbol{X}^{*b})^\top \boldsymbol{X}^{*b}\right)^{-1} (\boldsymbol{X}^{*b})^\top \boldsymbol{y}^{*b}. \tag{7}$$

The pairs bootstrap can be used for any model where the data are believed to be independent across observations.

This assumption would not be reasonable for data with any sort of serial dependence or clustered disturbances.

- It is natural to use pairs bootstrap with cross-section data, but it can also be used with some models that use time-series data.
- When regressors include lagged dependent variables, they are treated in the same way as any other column of $\boldsymbol{X}$.

If $\boldsymbol{y}^{*b}$ and $\boldsymbol{X}^{*b}$ are the data for the b^{th} bootstrap sample, then

$$\hat{\boldsymbol{\beta}}^{*b} = \left((\boldsymbol{X}^{*b})^\top \boldsymbol{X}^{*b}\right)^{-1} (\boldsymbol{X}^{*b})^\top \boldsymbol{y}^{*b}. \tag{7}$$

The pairs bootstrap can be used for any model where the data are believed to be independent across observations.

This assumption would not be reasonable for data with any sort of serial dependence or clustered disturbances.

- It is natural to use pairs bootstrap with cross-section data, but it can also be used with some models that use time-series data.
- When regressors include lagged dependent variables, they are treated in the same way as any other column of $\boldsymbol{X}$.

Pairs bootstrap does not require that disturbances be homoskedastic. Disturbances do not explicitly appear in the bootstrap DGP at all.

If $\boldsymbol{y}^{*b}$ and $\boldsymbol{X}^{*b}$ are the data for the b^{th} bootstrap sample, then

$$\hat{\boldsymbol{\beta}}^{*b} = \left((\boldsymbol{X}^{*b})^{\top}\boldsymbol{X}^{*b}\right)^{-1}(\boldsymbol{X}^{*b})^{\top}\boldsymbol{y}^{*b}. \tag{7}$$

The pairs bootstrap can be used for any model where the data are believed to be independent across observations.

This assumption would not be reasonable for data with any sort of serial dependence or clustered disturbances.

- It is natural to use pairs bootstrap with cross-section data, but it can also be used with some models that use time-series data.
- When regressors include lagged dependent variables, they are treated in the same way as any other column of $\boldsymbol{X}$.

Pairs bootstrap does not require that disturbances be homoskedastic. Disturbances do not explicitly appear in the bootstrap DGP at all.

The **pairs cluster bootstrap** resamples by cluster instead of by observation. The **pigeonhole bootstrap** resamples by cluster in two clustering dimensions.

In general, it is desirable to make the bootstrap DGP as close as possible to the (unknown) true DGP.

In general, it is desirable to make the bootstrap DGP as close as possible to the (unknown) true DGP.

In this respect, the pairs bootstrap has two big weaknesses:

In general, it is desirable to make the bootstrap DGP as close as possible to the (unknown) true DGP.

In this respect, the pairs bootstrap has two big weaknesses:

1. It does not condition on the actual $\boldsymbol{X}$ matrix. Instead, each pairs bootstrap sample has a different $\boldsymbol{X}^*$ matrix.

In general, it is desirable to make the bootstrap DGP as close as possible to the (unknown) true DGP.

In this respect, the pairs bootstrap has two big weaknesses:

1. It does not condition on the actual $\boldsymbol{X}$ matrix. Instead, each pairs bootstrap sample has a different $\boldsymbol{X}^*$ matrix.
2. It cannot impose the assumption that the null hypothesis is true.

In general, it is desirable to make the bootstrap DGP as close as possible to the (unknown) true DGP.

In this respect, the pairs bootstrap has two big weaknesses:

1. It does not condition on the actual $\boldsymbol{X}$ matrix. Instead, each pairs bootstrap sample has a different $\boldsymbol{X}^*$ matrix.
2. It cannot impose the assumption that the null hypothesis is true.

Point 1 is important whenever the distribution of whatever we are bootstrapping depends strongly on $\boldsymbol{X}$.

In general, it is desirable to make the bootstrap DGP as close as possible to the (unknown) true DGP.

In this respect, the pairs bootstrap has two big weaknesses:

1. It does not condition on the actual $\boldsymbol{X}$ matrix. Instead, each pairs bootstrap sample has a different $\boldsymbol{X}^*$ matrix.
2. It cannot impose the assumption that the null hypothesis is true.

Point 1 is important whenever the distribution of whatever we are bootstrapping depends strongly on $\boldsymbol{X}$.

Point 2 implies that bootstrap test statistics must be calculated for a *different* null hypothesis than the actual test statistic.

In general, it is desirable to make the bootstrap DGP as close as possible to the (unknown) true DGP.

In this respect, the pairs bootstrap has two big weaknesses:

1. It does not condition on the actual $\boldsymbol{X}$ matrix. Instead, each pairs bootstrap sample has a different $\boldsymbol{X}^*$ matrix.
2. It cannot impose the assumption that the null hypothesis is true.

Point 1 is important whenever the distribution of whatever we are bootstrapping depends strongly on $\boldsymbol{X}$.

Point 2 implies that bootstrap test statistics must be calculated for a *different* null hypothesis than the actual test statistic.

Pairs bootstrap rarely performs as well as the best available bootstrap method for any particular case, but it often performs acceptably.

In general, it is desirable to make the bootstrap DGP as close as possible to the (unknown) true DGP.

In this respect, the pairs bootstrap has two big weaknesses:

1. It does not condition on the actual $\boldsymbol{X}$ matrix. Instead, each pairs bootstrap sample has a different $\boldsymbol{X}^*$ matrix.
2. It cannot impose the assumption that the null hypothesis is true.

Point 1 is important whenever the distribution of whatever we are bootstrapping depends strongly on $\boldsymbol{X}$.

Point 2 implies that bootstrap test statistics must be calculated for a *different* null hypothesis than the actual test statistic.

Pairs bootstrap rarely performs as well as the best available bootstrap method for any particular case, but it often performs acceptably.

It is most attractive for nonlinear models, where methods specifically adapted to the linear regression model are not available.

The Residual Bootstrap

The Residual Bootstrap

We can resample from (transformed) residuals. Let $\acute{\boldsymbol{u}}$ have typical element $\acute{u}_i = (N/(N-k))^{1/2}\hat{u}_i$. Then the **unrestricted residual bootstrap** DGP is

$$y_i^* = \boldsymbol{X}_i\hat{\boldsymbol{\beta}} + u_i^*, \quad u_i^* \sim \mathrm{EDF}(\acute{\boldsymbol{u}}). \tag{8}$$

The Residual Bootstrap

We can resample from (transformed) residuals. Let $\acute{\boldsymbol{u}}$ have typical element $\acute{u}_i = (N/(N-k))^{1/2}\hat{u}_i$. Then the **unrestricted residual bootstrap** DGP is

$$y_i^* = \boldsymbol{X}_i\hat{\boldsymbol{\beta}} + u_i^*, \quad u_i^* \sim \text{EDF}(\acute{\boldsymbol{u}}). \tag{8}$$

This bootstrap DGP does not impose any restrictions on $\boldsymbol{\beta}$. But when we are testing restrictions, it is often good to impose them.

The Residual Bootstrap

We can resample from (transformed) residuals. Let $\boldsymbol{\acute{u}}$ have typical element $\acute{u}_i = (N/(N-k))^{1/2}\hat{u}_i$. Then the **unrestricted residual bootstrap** DGP is

$$y_i^* = \boldsymbol{X}_i\hat{\boldsymbol{\beta}} + u_i^*, \quad u_i^* \sim \text{EDF}(\boldsymbol{\acute{u}}). \tag{8}$$

This bootstrap DGP does not impose any restrictions on $\boldsymbol{\beta}$. But when we are testing restrictions, it is often good to impose them.

Suppose that $\tilde{\boldsymbol{\beta}}$ and $\tilde{\boldsymbol{u}}$ denote restricted estimates and restricted residuals. Then the **restricted residual bootstrap** DGP is

$$y_i^* = \boldsymbol{X}_i\tilde{\boldsymbol{\beta}} + u_i^*, \quad u_i^* \sim \text{EDF}(\boldsymbol{\grave{u}}). \tag{9}$$

The Residual Bootstrap

We can resample from (transformed) residuals. Let $\acute{\boldsymbol{u}}$ have typical element $\acute{u}_i = (N/(N-k))^{1/2}\hat{u}_i$. Then the **unrestricted residual bootstrap** DGP is

$$y_i^* = \boldsymbol{X}_i\hat{\boldsymbol{\beta}} + u_i^*, \quad u_i^* \sim \text{EDF}(\acute{\boldsymbol{u}}). \tag{8}$$

This bootstrap DGP does not impose any restrictions on $\boldsymbol{\beta}$. But when we are testing restrictions, it is often good to impose them.

Suppose that $\tilde{\boldsymbol{\beta}}$ and $\tilde{\boldsymbol{u}}$ denote restricted estimates and restricted residuals. Then the **restricted residual bootstrap** DGP is

$$y_i^* = \boldsymbol{X}_i\tilde{\boldsymbol{\beta}} + u_i^*, \quad u_i^* \sim \text{EDF}(\grave{\boldsymbol{u}}). \tag{9}$$

Here $\grave{\boldsymbol{u}}$ has typical element $\grave{u}_i = (N/(N-k_1))^{1/2}\tilde{u}_i$ when there are $k_2 = k - k_1$ restrictions.

The factors $(N/(N-k))^{1/2}$ and $(N/(N-k_1))^{1/2}$ in $\acute{\boldsymbol{u}}$ and $\grave{\boldsymbol{u}}$ ensure that the bootstrap disturbances have the correct expectation.

The factors $(N/(N-k))^{1/2}$ and $(N/(N-k_1))^{1/2}$ in $\acute{\boldsymbol{u}}$ and $\grave{\boldsymbol{u}}$ ensure that the bootstrap disturbances have the correct expectation.

- Other transformations of the $\hat{u}_i$ and $\tilde{u}_i$ can also be used.

The factors $(N/(N-k))^{1/2}$ and $(N/(N-k_1))^{1/2}$ in $\acute{\boldsymbol{u}}$ and $\grave{\boldsymbol{u}}$ ensure that the bootstrap disturbances have the correct expectation.

- Other transformations of the $\hat{u}_i$ and $\tilde{u}_i$ can also be used.
- Unlike the pairs bootstrap, which is fully **nonparametric**, the residual bootstrap is **semiparametric**.

The factors $(N/(N-k))^{1/2}$ and $(N/(N-k_1))^{1/2}$ in $\acute{\boldsymbol{u}}$ and $\grave{\boldsymbol{u}}$ ensure that the bootstrap disturbances have the correct expectation.

- Other transformations of the $\hat{u}_i$ and $\tilde{u}_i$ can also be used.
- Unlike the pairs bootstrap, which is fully **nonparametric**, the residual bootstrap is **semiparametric**.
- The form of the regression function is treated as known, but the distribution of the u_i is treated as unknown.

The factors $(N/(N-k))^{1/2}$ and $(N/(N-k_1))^{1/2}$ in $\acute{\boldsymbol{u}}$ and $\grave{\boldsymbol{u}}$ ensure that the bootstrap disturbances have the correct expectation.

- Other transformations of the $\hat{u}_i$ and $\tilde{u}_i$ can also be used.
- Unlike the pairs bootstrap, which is fully **nonparametric**, the residual bootstrap is **semiparametric**.
- The form of the regression function is treated as known, but the distribution of the u_i is treated as unknown.
- However, every one of the u_i is assumed to have the same distribution. This rules out heteroskedasticity.

The factors $(N/(N-k))^{1/2}$ and $(N/(N-k_1))^{1/2}$ in $\acute{\boldsymbol{u}}$ and $\grave{\boldsymbol{u}}$ ensure that the bootstrap disturbances have the correct expectation.

- Other transformations of the $\hat{u}_i$ and $\tilde{u}_i$ can also be used.
- Unlike the pairs bootstrap, which is fully **nonparametric**, the residual bootstrap is **semiparametric**.
- The form of the regression function is treated as known, but the distribution of the u_i is treated as unknown.
- However, every one of the u_i is assumed to have the same distribution. This rules out heteroskedasticity.

Like the $s^2(\boldsymbol{X}^\top\boldsymbol{X})^{-1}$ covariance matrix estimator, the residual bootstrap is now considered to be too restrictive for use with cross-section data.

The factors $(N/(N-k))^{1/2}$ and $(N/(N-k_1))^{1/2}$ in $\acute{\boldsymbol{u}}$ and $\grave{\boldsymbol{u}}$ ensure that the bootstrap disturbances have the correct expectation.

- Other transformations of the $\hat{u}_i$ and $\tilde{u}_i$ can also be used.
- Unlike the pairs bootstrap, which is fully **nonparametric**, the residual bootstrap is **semiparametric**.
- The form of the regression function is treated as known, but the distribution of the u_i is treated as unknown.
- However, every one of the u_i is assumed to have the same distribution. This rules out heteroskedasticity.

Like the $s^2(\boldsymbol{X}^\top\boldsymbol{X})^{-1}$ covariance matrix estimator, the residual bootstrap is now considered to be too restrictive for use with cross-section data.

HCCMEs have largely replaced the former, and the **wild bootstrap** (next slide) has largely replaced the latter.

The factors $(N/(N-k))^{1/2}$ and $(N/(N-k_1))^{1/2}$ in $\acute{\boldsymbol{u}}$ and $\grave{\boldsymbol{u}}$ ensure that the bootstrap disturbances have the correct expectation.

- Other transformations of the $\hat{u}_i$ and $\tilde{u}_i$ can also be used.
- Unlike the pairs bootstrap, which is fully **nonparametric**, the residual bootstrap is **semiparametric**.
- The form of the regression function is treated as known, but the distribution of the u_i is treated as unknown.
- However, every one of the u_i is assumed to have the same distribution. This rules out heteroskedasticity.

Like the $s^2(\boldsymbol{X}^\top\boldsymbol{X})^{-1}$ covariance matrix estimator, the residual bootstrap is now considered to be too restrictive for use with cross-section data.

HCCMEs have largely replaced the former, and the **wild bootstrap** (next slide) has largely replaced the latter.

When there is assumed to be intra-cluster correlation, it is common to combine a CRVE with the **wild cluster bootstrap**.

The Wild Bootstrap

The Wild Bootstrap

The wild bootstrap conditions each value y_i^* not only on $\boldsymbol{X}_i$ but also on the residual for observation i, which is either $\tilde{u}_i$ or $\hat{u}_i$.

The Wild Bootstrap

The wild bootstrap conditions each value y_i^* not only on $\boldsymbol{X}_i$ but also on the residual for observation i, which is either $\tilde{u}_i$ or $\hat{u}_i$.

For a linear regression model with heteroskedastic disturbances, the wild bootstrap DGP is

$$y_i^* = \boldsymbol{X}_i\ddot{\boldsymbol{\beta}} + u_i^*, \quad u_i^* = v_i^*\ddot{u}_i. \tag{10}$$

The Wild Bootstrap

The wild bootstrap conditions each value y_i^* not only on $\boldsymbol{X}_i$ but also on the residual for observation i, which is either $\tilde{u}_i$ or $\hat{u}_i$.

For a linear regression model with heteroskedastic disturbances, the wild bootstrap DGP is

$$y_i^* = \boldsymbol{X}_i \ddot{\boldsymbol{\beta}} + u_i^*, \quad u_i^* = v_i^* \ddot{u}_i. \tag{10}$$

Here $\ddot{\boldsymbol{\beta}}$ denotes either $\tilde{\boldsymbol{\beta}}$ or $\hat{\boldsymbol{\beta}}$, $\ddot{u}_i$ denotes either $\hat{u}_i$, $\tilde{u}_i$, or a transformed version of one of them, and v_i^* is an **auxiliary random variable** with mean 0 and variance 1.

The Wild Bootstrap

The wild bootstrap conditions each value y_i^* not only on $\boldsymbol{X}_i$ but also on the residual for observation i, which is either $\tilde{u}_i$ or $\hat{u}_i$.

For a linear regression model with heteroskedastic disturbances, the wild bootstrap DGP is

$$y_i^* = \boldsymbol{X}_i \ddot{\boldsymbol{\beta}} + u_i^*, \quad u_i^* = v_i^* \ddot{u}_i. \tag{10}$$

Here $\ddot{\boldsymbol{\beta}}$ denotes either $\tilde{\boldsymbol{\beta}}$ or $\hat{\boldsymbol{\beta}}$, $\ddot{u}_i$ denotes either $\hat{u}_i$, $\tilde{u}_i$, or a transformed version of one of them, and v_i^* is an **auxiliary random variable** with mean 0 and variance 1.

Because v_i^* has variance 1, $\text{Var}(u_i^*) = \text{Var}(\ddot{u}_i)$.

The Wild Bootstrap

The wild bootstrap conditions each value y_i^* not only on $\boldsymbol{X}_i$ but also on the residual for observation i, which is either $\tilde{u}_i$ or $\hat{u}_i$.

For a linear regression model with heteroskedastic disturbances, the wild bootstrap DGP is

$$y_i^* = \boldsymbol{X}_i \ddot{\boldsymbol{\beta}} + u_i^*, \quad u_i^* = v_i^* \ddot{u}_i. \tag{10}$$

Here $\ddot{\boldsymbol{\beta}}$ denotes either $\tilde{\boldsymbol{\beta}}$ or $\hat{\boldsymbol{\beta}}$, $\ddot{u}_i$ denotes either $\hat{u}_i$, $\tilde{u}_i$, or a transformed version of one of them, and v_i^* is an **auxiliary random variable** with mean 0 and variance 1.

Because v_i^* has variance 1, $\text{Var}(u_i^*) = \text{Var}(\ddot{u}_i)$.

Thus, on average, u_i^* will be large for observations with large residuals and small for ones with small residuals.

The Wild Bootstrap

The wild bootstrap conditions each value y_i^* not only on $\boldsymbol{X}_i$ but also on the residual for observation i, which is either $\tilde{u}_i$ or $\hat{u}_i$.

For a linear regression model with heteroskedastic disturbances, the wild bootstrap DGP is

$$y_i^* = \boldsymbol{X}_i \ddot{\boldsymbol{\beta}} + u_i^*, \quad u_i^* = v_i^* \ddot{u}_i. \tag{10}$$

Here $\ddot{\boldsymbol{\beta}}$ denotes either $\tilde{\boldsymbol{\beta}}$ or $\hat{\boldsymbol{\beta}}$, $\ddot{u}_i$ denotes either $\hat{u}_i$, $\tilde{u}_i$, or a transformed version of one of them, and v_i^* is an **auxiliary random variable** with mean 0 and variance 1.

Because v_i^* has variance 1, $\mathrm{Var}(u_i^*) = \mathrm{Var}(\ddot{u}_i)$.

Thus, on average, u_i^* will be large for observations with large residuals and small for ones with small residuals.

The wild bootstrap is a form of **multiplier bootstrap**.

We saw when discussing heteroskedasticity-robust inference that

$$\hat{u}_i^2 \overset{a}{=} \omega_i^2 + v_i, \tag{11}$$

where ω_i^2 is the variance of the i^{th} disturbance.

We saw when discussing heteroskedasticity-robust inference that

$$\hat{u}_i^2 \stackrel{a}{=} \omega_i^2 + v_i, \tag{11}$$

where ω_i^2 is the variance of the i^{th} disturbance.

This suggests that $\hat{u}_i^2$ (and $\ddot{u}_i^2$) can be used to estimate ω_i^2. Of course, $\hat{u}_i^2$ is a very noisy estimator, but that does not matter asymptotically.

We saw when discussing heteroskedasticity-robust inference that

$$\hat{u}_i^2 \overset{a}{=} \omega_i^2 + v_i, \tag{11}$$

where ω_i^2 is the variance of the i^{th} disturbance.

This suggests that $\hat{u}_i^2$ (and $\ddot{u}_i^2$) can be used to estimate ω_i^2. Of course, $\hat{u}_i^2$ is a very noisy estimator, but that does not matter asymptotically.

Ideally, v_i^* should have mean 0, variance 1, and all higher moments equal to 1. If so, u_i^* has same moments as $\ddot{u}_i$.

We saw when discussing heteroskedasticity-robust inference that

$$\hat{u}_i^2 \overset{a}{=} \omega_i^2 + v_i, \tag{11}$$

where ω_i^2 is the variance of the i^{th} disturbance.

This suggests that $\hat{u}_i^2$ (and $\ddot{u}_i^2$) can be used to estimate ω_i^2. Of course, $\hat{u}_i^2$ is a very noisy estimator, but that does not matter asymptotically.

Ideally, v_i^* should have mean 0, variance 1, and all higher moments equal to 1. If so, u_i^* has same moments as $\ddot{u}_i$.

Unfortunately, there exists no distribution with these properties!

We saw when discussing heteroskedasticity-robust inference that

$$\hat{u}_i^2 \overset{a}{=} \omega_i^2 + v_i, \tag{11}$$

where ω_i^2 is the variance of the i^{th} disturbance.

This suggests that $\hat{u}_i^2$ (and $\ddot{u}_i^2$) can be used to estimate ω_i^2. Of course, $\hat{u}_i^2$ is a very noisy estimator, but that does not matter asymptotically.

Ideally, v_i^* should have mean 0, variance 1, and all higher moments equal to 1. If so, u_i^* has same moments as $\ddot{u}_i$.

Unfortunately, there exists no distribution with these properties!

The best choice usually seems to be the **Rademacher distribution**. It takes on just two values, 1 and -1, each with equal probability.

We saw when discussing heteroskedasticity-robust inference that

$$\hat{u}_i^2 \overset{a}{=} \omega_i^2 + v_i, \qquad (11)$$

where ω_i^2 is the variance of the i^{th} disturbance.

This suggests that $\hat{u}_i^2$ (and $\ddot{u}_i^2$) can be used to estimate ω_i^2. Of course, $\hat{u}_i^2$ is a very noisy estimator, but that does not matter asymptotically.

Ideally, v_i^* should have mean 0, variance 1, and all higher moments equal to 1. If so, u_i^* has same moments as $\ddot{u}_i$.

Unfortunately, there exists no distribution with these properties!

The best choice usually seems to be the **Rademacher distribution**. It takes on just two values, 1 and -1, each with equal probability.

- The third and fourth moments of the Rademacher distribution are 0 and 1, respectively.

We saw when discussing heteroskedasticity-robust inference that

$$\hat{u}_i^2 \overset{a}{=} \omega_i^2 + v_i, \tag{11}$$

where ω_i^2 is the variance of the i^{th} disturbance.

This suggests that $\hat{u}_i^2$ (and $\ddot{u}_i^2$) can be used to estimate ω_i^2. Of course, $\hat{u}_i^2$ is a very noisy estimator, but that does not matter asymptotically.

Ideally, v_i^* should have mean 0, variance 1, and all higher moments equal to 1. If so, u_i^* has same moments as $\ddot{u}_i$.

Unfortunately, there exists no distribution with these properties!

The best choice usually seems to be the **Rademacher distribution**. It takes on just two values, 1 and -1, each with equal probability.

- The third and fourth moments of the Rademacher distribution are 0 and 1, respectively.
- Because the third moment is 0, the u_i^* must be symmetric, which seems like a serious restriction.

Mammen (1993) suggested another two-point distribution that has a third moment of 1, but its fourth moment is 2. It is

$$v_i^* = \begin{cases} -(\sqrt{5}-1)/2 & \text{with prob. } (\sqrt{5}+1)/(2\sqrt{5}), \\ (\sqrt{5}+1)/2 & \text{with prob. } (\sqrt{5}-1)/(2\sqrt{5}). \end{cases} \tag{12}$$

Mammen (1993) suggested another two-point distribution that has a third moment of 1, but its fourth moment is 2. It is

$$v_i^* = \begin{cases} -(\sqrt{5}-1)/2 & \text{with prob. } (\sqrt{5}+1)/(2\sqrt{5}), \\ (\sqrt{5}+1)/2 & \text{with prob. } (\sqrt{5}-1)/(2\sqrt{5}). \end{cases} \tag{12}$$

Rademacher seems to outperform Mammen, even when the u_i are asymmetric; see Davidson and Flachaire (2008) and Djogbenou, MacKinnon, and Nielsen (2019).

Mammen (1993) suggested another two-point distribution that has a third moment of 1, but its fourth moment is 2. It is

$$v_i^* = \begin{cases} -(\sqrt{5}-1)/2 & \text{with prob. } (\sqrt{5}+1)/(2\sqrt{5}), \\ (\sqrt{5}+1)/2 & \text{with prob. } (\sqrt{5}-1)/(2\sqrt{5}). \end{cases} \tag{12}$$

Rademacher seems to outperform Mammen, even when the u_i are asymmetric; see Davidson and Flachaire (2008) and Djogbenou, MacKinnon, and Nielsen (2019).

- The $\mathrm{N}(0,1)$ distribution has first three moments 0, 1, 0, but its fourth moment is 3, which is much too large.

Mammen (1993) suggested another two-point distribution that has a third moment of 1, but its fourth moment is 2. It is

$$v_i^* = \begin{cases} -(\sqrt{5}-1)/2 & \text{with prob. } (\sqrt{5}+1)/(2\sqrt{5}), \\ (\sqrt{5}+1)/2 & \text{with prob. } (\sqrt{5}-1)/(2\sqrt{5}). \end{cases} \tag{12}$$

Rademacher seems to outperform Mammen, even when the u_i are asymmetric; see Davidson and Flachaire (2008) and Djogbenou, MacKinnon, and Nielsen (2019).

- The $\mathrm{N}(0, 1)$ distribution has first three moments 0, 1, 0, but its fourth moment is 3, which is much too large.
- A better choice is probably the uniform $\mathrm{U}(-\sqrt{3}, \sqrt{3})$ distribution, which has third moment 0 and fourth moment 1.8.

Mammen (1993) suggested another two-point distribution that has a third moment of 1, but its fourth moment is 2. It is

$$v_i^* = \begin{cases} -(\sqrt{5}-1)/2 & \text{with prob. } (\sqrt{5}+1)/(2\sqrt{5}), \\ (\sqrt{5}+1)/2 & \text{with prob. } (\sqrt{5}-1)/(2\sqrt{5}). \end{cases} \quad (12)$$

Rademacher seems to outperform Mammen, even when the u_i are asymmetric; see Davidson and Flachaire (2008) and Djogbenou, MacKinnon, and Nielsen (2019).

- The $\mathrm{N}(0,1)$ distribution has first three moments 0, 1, 0, but its fourth moment is 3, which is much too large.
- A better choice is probably the uniform $\mathrm{U}(-\sqrt{3}, \sqrt{3})$ distribution, which has third moment 0 and fourth moment 1.8.

It is common to replace $\ddot{u}_i$ in the bootstrap DGP (10) by $\psi(\ddot{u}_i)$, where $\psi(\cdot)$ is a monotonically increasing transformation.

Mammen (1993) suggested another two-point distribution that has a third moment of 1, but its fourth moment is 2. It is

$$v_i^* = \begin{cases} -(\sqrt{5}-1)/2 & \text{with prob. } (\sqrt{5}+1)/(2\sqrt{5}), \\ (\sqrt{5}+1)/2 & \text{with prob. } (\sqrt{5}-1)/(2\sqrt{5}). \end{cases} \tag{12}$$

Rademacher seems to outperform Mammen, even when the u_i are asymmetric; see Davidson and Flachaire (2008) and Djogbenou, MacKinnon, and Nielsen (2019).

- The $\mathrm{N}(0,1)$ distribution has first three moments 0, 1, 0, but its fourth moment is 3, which is much too large.
- A better choice is probably the uniform $\mathrm{U}(-\sqrt{3}, \sqrt{3})$ distribution, which has third moment 0 and fourth moment 1.8.

It is common to replace $\ddot{u}_i$ in the bootstrap DGP (10) by $\psi(\ddot{u}_i)$, where $\psi(\cdot)$ is a monotonically increasing transformation.

- $\psi(\hat{u}_i) = \hat{u}_i/(1-h_i)$ corresponds to HC_3 (the jackknife)
- $\psi(\hat{u}_i) = \hat{u}_i/(1-h_i)^{1/2}$ corresponds to HC_2

Bootstrap Standard Errors

Bootstrap Standard Errors

Having generated B vectors $\boldsymbol{y}^{*b}$, what do we do with them?

Bootstrap Standard Errors

Having generated B vectors $\boldsymbol{y}^{*b}$, what do we do with them?

The easiest, and oldest, method of bootstrap inference uses the bootstrap samples to compute **bootstrap standard errors**.

Bootstrap Standard Errors

Having generated B vectors $\boldsymbol{y}^{*b}$, what do we do with them?

The easiest, and oldest, method of bootstrap inference uses the bootstrap samples to compute **bootstrap standard errors**.

The procedure for obtaining a bootstrap standard error for $\hat{\theta}$, an estimate of the parameter θ, is very simple:

Bootstrap Standard Errors

Having generated B vectors $\boldsymbol{y}^{*b}$, what do we do with them?

The easiest, and oldest, method of bootstrap inference uses the bootstrap samples to compute **bootstrap standard errors**.

The procedure for obtaining a bootstrap standard error for $\hat{\theta}$, an estimate of the parameter θ, is very simple:

1. Specify a bootstrap DGP that does not impose a restriction on θ. Use it to generate B bootstrap samples, $\boldsymbol{y}^{*b}$, for $b = 1, \ldots, B$.

Bootstrap Standard Errors

Having generated B vectors $\boldsymbol{y}^{*b}$, what do we do with them?

The easiest, and oldest, method of bootstrap inference uses the bootstrap samples to compute **bootstrap standard errors**.

The procedure for obtaining a bootstrap standard error for $\hat{\theta}$, an estimate of the parameter θ, is very simple:

1. Specify a bootstrap DGP that does not impose a restriction on θ. Use it to generate B bootstrap samples, $\boldsymbol{y}^{*b}$, for $b = 1, \ldots, B$.
2. For each $\boldsymbol{y}^{*b}$, compute an estimate $\hat{\theta}^*_b$.

Bootstrap Standard Errors

Having generated B vectors $\boldsymbol{y}^{*b}$, what do we do with them?

The easiest, and oldest, method of bootstrap inference uses the bootstrap samples to compute **bootstrap standard errors**.

The procedure for obtaining a bootstrap standard error for $\hat{\theta}$, an estimate of the parameter θ, is very simple:

1. Specify a bootstrap DGP that does not impose a restriction on θ. Use it to generate B bootstrap samples, $\boldsymbol{y}^{*b}$, for $b = 1, \ldots, B$.
2. For each $\boldsymbol{y}^{*b}$, compute an estimate $\hat{\theta}^*_b$.
3. Calculate $\bar{\theta}^*$, the sample mean of the $\hat{\theta}^*_b$, and their sample variance

$$\widehat{\mathrm{Var}}^*(\hat{\theta}^*_b) = \frac{1}{B-1}\sum_{b=1}^{B}(\hat{\theta}^*_b - \bar{\theta}^*)^2. \tag{13}$$

Bootstrap Standard Errors

Having generated B vectors $\boldsymbol{y}^{*b}$, what do we do with them?

The easiest, and oldest, method of bootstrap inference uses the bootstrap samples to compute **bootstrap standard errors**.

The procedure for obtaining a bootstrap standard error for $\hat{\theta}$, an estimate of the parameter θ, is very simple:

1. Specify a bootstrap DGP that does not impose a restriction on θ. Use it to generate B bootstrap samples, $\boldsymbol{y}^{*b}$, for $b = 1, \ldots, B$.
2. For each $\boldsymbol{y}^{*b}$, compute an estimate $\hat{\theta}^*_b$.
3. Calculate $\bar{\theta}^*$, the sample mean of the $\hat{\theta}^*_b$, and their sample variance

$$\widehat{\text{Var}}^*(\hat{\theta}^*_b) = \frac{1}{B-1}\sum_{b=1}^{B}(\hat{\theta}^*_b - \bar{\theta}^*)^2. \tag{13}$$

Then $\text{se}^*(\hat{\theta})$ is simply the square root of $\widehat{\text{Var}}^*(\hat{\theta}^*_b)$.

This method provides an alternative to using asymptotic standard errors for nonlinear transformations of parameters.

This method provides an alternative to using asymptotic standard errors for nonlinear transformations of parameters.

Suppose we wish to calculate the covariance matrix of the vector $\hat{\gamma} = \boldsymbol{g}(\hat{\boldsymbol{\theta}})$, where $\boldsymbol{g}(\cdot)$ is a possibly nonlinear transformation.

This method provides an alternative to using asymptotic standard errors for nonlinear transformations of parameters.

Suppose we wish to calculate the covariance matrix of the vector $\hat{\gamma} = \boldsymbol{g}(\hat{\boldsymbol{\theta}})$, where $\boldsymbol{g}(\cdot)$ is a possibly nonlinear transformation.

1. Specify an unrestricted bootstrap DGP, and use it to generate B bootstrap samples, $\boldsymbol{y}^{*b}$.

This method provides an alternative to using asymptotic standard errors for nonlinear transformations of parameters.

Suppose we wish to calculate the covariance matrix of the vector $\hat{\boldsymbol{\gamma}} = \boldsymbol{g}(\hat{\boldsymbol{\theta}})$, where $\boldsymbol{g}(\cdot)$ is a possibly nonlinear transformation.

1. Specify an unrestricted bootstrap DGP, and use it to generate B bootstrap samples, $\boldsymbol{y}^{*b}$.
2. For each bootstrap sample, compute the vector $\hat{\boldsymbol{\theta}}^{*b}$ in the same way as $\hat{\boldsymbol{\theta}}$ was computed from the original sample $\boldsymbol{y}$. Use it to calculate $\hat{\boldsymbol{\gamma}}^{*b} = \boldsymbol{g}(\hat{\boldsymbol{\theta}}^{*b})$.

This method provides an alternative to using asymptotic standard errors for nonlinear transformations of parameters.

Suppose we wish to calculate the covariance matrix of the vector $\hat{\gamma} = \boldsymbol{g}(\hat{\boldsymbol{\theta}})$, where $\boldsymbol{g}(\cdot)$ is a possibly nonlinear transformation.

1. Specify an unrestricted bootstrap DGP, and use it to generate B bootstrap samples, $\boldsymbol{y}^{*b}$.
2. For each bootstrap sample, compute the vector $\hat{\boldsymbol{\theta}}^{*b}$ in the same way as $\hat{\boldsymbol{\theta}}$ was computed from the original sample $\boldsymbol{y}$. Use it to calculate $\hat{\gamma}^{*b} = \boldsymbol{g}(\hat{\boldsymbol{\theta}}^{*b})$.
3. Compute the vector $\bar{\gamma}^*$, which is the mean of the $\hat{\gamma}^{*b}$ vectors. Then calculate the estimated bootstrap covariance matrix as

$$\widehat{\mathrm{Var}}^*(\hat{\gamma}) = \frac{1}{B-1}\sum_{b=1}^{B}(\hat{\gamma}^{*b} - \bar{\gamma}^*)(\hat{\gamma}^{*b} - \bar{\gamma}^*)^\top. \quad (14)$$

This method provides an alternative to using asymptotic standard errors for nonlinear transformations of parameters.

Suppose we wish to calculate the covariance matrix of the vector $\hat{\gamma} = \boldsymbol{g}(\hat{\boldsymbol{\theta}})$, where $\boldsymbol{g}(\cdot)$ is a possibly nonlinear transformation.

1. Specify an unrestricted bootstrap DGP, and use it to generate B bootstrap samples, $\boldsymbol{y}^{*b}$.
2. For each bootstrap sample, compute the vector $\hat{\boldsymbol{\theta}}^{*b}$ in the same way as $\hat{\boldsymbol{\theta}}$ was computed from the original sample $\boldsymbol{y}$. Use it to calculate $\hat{\gamma}^{*b} = \boldsymbol{g}(\hat{\boldsymbol{\theta}}^{*b})$.
3. Compute the vector $\bar{\gamma}^*$, which is the mean of the $\hat{\gamma}^{*b}$ vectors. Then calculate the estimated bootstrap covariance matrix as

$$\widehat{\mathrm{Var}}^*(\hat{\gamma}) = \frac{1}{B-1}\sum_{b=1}^{B}(\hat{\gamma}^{*b} - \bar{\gamma}^*)(\hat{\gamma}^{*b} - \bar{\gamma}^*)^\top. \tag{14}$$

Of course, this also works if $\gamma(\boldsymbol{\theta}) = \boldsymbol{\theta}$. So we can easily obtain $\widehat{\mathrm{Var}}^*(\hat{\boldsymbol{\theta}})$.

Given $\text{se}^*(\hat{\theta})$, we can perform (approximate) t tests and compute conventional-looking confidence intervals.

Given $\mathrm{se}^*(\hat{\theta})$, we can perform (approximate) t tests and compute conventional-looking confidence intervals.

An approximate t statistic is

$$t^*(\theta_0) = \frac{\hat{\theta} - \theta_0}{\mathrm{se}^*(\hat{\theta})}. \tag{15}$$

Given $\text{se}^*(\hat{\theta})$, we can perform (approximate) t tests and compute conventional-looking confidence intervals.

An approximate t statistic is

$$t^*(\theta_0) = \frac{\hat{\theta} - \theta_0}{\text{se}^*(\hat{\theta})}. \tag{15}$$

We can pretend that this follows the $\text{N}(0, 1)$ or $t(N - k)$ distributions.

Given $\text{se}^*(\hat{\theta})$, we can perform (approximate) t tests and compute conventional-looking confidence intervals.

An approximate t statistic is

$$t^*(\theta_0) = \frac{\hat{\theta} - \theta_0}{\text{se}^*(\hat{\theta})}. \tag{15}$$

We can pretend that this follows the $\text{N}(0, 1)$ or $t(N - k)$ distributions.

This test should work well if:

Given $\mathrm{se}^*(\hat{\theta})$, we can perform (approximate) t tests and compute conventional-looking confidence intervals.

An approximate t statistic is

$$t^*(\theta_0) = \frac{\hat{\theta} - \theta_0}{\mathrm{se}^*(\hat{\theta})}. \quad (15)$$

We can pretend that this follows the $\mathrm{N}(0, 1)$ or $t(N - k)$ distributions.

This test should work well if:

- $\hat{\theta} - \theta_0$ is approximately normally distributed with mean 0;

Given $\mathrm{se}^*(\hat{\theta})$, we can perform (approximate) t tests and compute conventional-looking confidence intervals.

An approximate t statistic is

$$t^*(\theta_0) = \frac{\hat{\theta} - \theta_0}{\mathrm{se}^*(\hat{\theta})}. \tag{15}$$

We can pretend that this follows the $\mathrm{N}(0,1)$ or $t(N-k)$ distributions.

This test should work well if:

- $\hat{\theta} - \theta_0$ is approximately normally distributed with mean 0;
- The variance of $\hat{\theta} - \theta_0$ is approximately equal to $\widehat{\mathrm{Var}}^*(\hat{\theta}^*_b)$.

Given $\text{se}^*(\hat{\theta})$, we can perform (approximate) t tests and compute conventional-looking confidence intervals.

An approximate t statistic is

$$t^*(\theta_0) = \frac{\hat{\theta} - \theta_0}{\text{se}^*(\hat{\theta})}. \tag{15}$$

We can pretend that this follows the $\text{N}(0, 1)$ or $t(N - k)$ distributions.

This test should work well if:

- $\hat{\theta} - \theta_0$ is approximately normally distributed with mean 0;
- The variance of $\hat{\theta} - \theta_0$ is approximately equal to $\widehat{\text{Var}}^*(\hat{\theta}^*_b)$.

In general, there is no theoretical reason to expect inference based on (15) to be more or less accurate than asymptotic inference.

Given $\text{se}^*(\hat{\theta})$, we can perform (approximate) t tests and compute conventional-looking confidence intervals.

An approximate t statistic is

$$t^*(\theta_0) = \frac{\hat{\theta} - \theta_0}{\text{se}^*(\hat{\theta})}. \tag{15}$$

We can pretend that this follows the $\text{N}(0, 1)$ or $t(N - k)$ distributions.

This test should work well if:

- $\hat{\theta} - \theta_0$ is approximately normally distributed with mean 0;
- The variance of $\hat{\theta} - \theta_0$ is approximately equal to $\widehat{\text{Var}}^*(\hat{\theta}^*_b)$.

In general, there is no theoretical reason to expect inference based on (15) to be more or less accurate than asymptotic inference.

It depends on whether $\text{se}^*(\hat{\theta})$ is more or less accurate, and more or less independent of $\hat{\theta}$, than an asymptotic standard error.

Once we have a bootstrap standard error $\text{se}^*(\hat{\theta})$, we can easily form bootstrap confidence intervals.

Once we have a bootstrap standard error $\text{se}^*(\hat{\theta})$, we can easily form bootstrap confidence intervals.

A conventional bootstrap interval at level α is

$$\left[\hat{\theta} - \text{se}^*(\hat{\theta})\, z_{1-\alpha/2}, \quad \hat{\theta} + \text{se}^*(\hat{\theta})\, z_{1-\alpha/2}\right], \tag{16}$$

where $z_{1-\alpha/2}$ denotes the $1 - \alpha/2$ quantile of the standard normal distribution. When $\alpha = .05$, $z_{1-\alpha/2} = 1.96$.

Once we have a bootstrap standard error $\text{se}^*(\hat{\theta})$, we can easily form bootstrap confidence intervals.

A conventional bootstrap interval at level α is

$$\left[\hat{\theta} - \text{se}^*(\hat{\theta})\, z_{1-\alpha/2}, \quad \hat{\theta} + \text{se}^*(\hat{\theta})\, z_{1-\alpha/2}\right], \tag{16}$$

where $z_{1-\alpha/2}$ denotes the $1-\alpha/2$ quantile of the standard normal distribution. When $\alpha = .05$, $z_{1-\alpha/2} = 1.96$.

- We could also use a critical value from the $t(N-k)$ distribution, which would be more conservative.

Once we have a bootstrap standard error $\mathrm{se}^*(\hat{\theta})$, we can easily form bootstrap confidence intervals.

A conventional bootstrap interval at level α is

$$\left[\hat{\theta} - \mathrm{se}^*(\hat{\theta})\, z_{1-\alpha/2}, \quad \hat{\theta} + \mathrm{se}^*(\hat{\theta})\, z_{1-\alpha/2}\right], \tag{16}$$

where $z_{1-\alpha/2}$ denotes the $1 - \alpha/2$ quantile of the standard normal distribution. When $\alpha = .05$, $z_{1-\alpha/2} = 1.96$.

- We could also use a critical value from the $t(N - k)$ distribution, which would be more conservative.

The interval (16) may not be very accurate if the distribution of $\hat{\theta} - \theta_0$ is not well approximated by the normal distribution with mean zero.

Once we have a bootstrap standard error $\mathrm{se}^*(\hat{\theta})$, we can easily form bootstrap confidence intervals.

A conventional bootstrap interval at level α is

$$\left[\hat{\theta} - \mathrm{se}^*(\hat{\theta})\, z_{1-\alpha/2}, \quad \hat{\theta} + \mathrm{se}^*(\hat{\theta})\, z_{1-\alpha/2}\right], \tag{16}$$

where $z_{1-\alpha/2}$ denotes the $1-\alpha/2$ quantile of the standard normal distribution. When $\alpha = .05$, $z_{1-\alpha/2} = 1.96$.

- We could also use a critical value from the $t(N-k)$ distribution, which would be more conservative.

The interval (16) may not be very accurate if the distribution of $\hat{\theta} - \theta_0$ is not well approximated by the normal distribution with mean zero.

If $\hat{\theta}$ is biased, or its distribution is asymmetric or has thick tails, using the $1-\alpha/2$ quantile of the $\mathrm{N}(0, 1)$ distribution to obtain the limits of the interval may cause it to undercover, perhaps severely.

Using bootstrap standard errors makes sense if asymptotic standard errors are not available or may be seriously unreliable. It is good to compute both to see how well they agree.

Using bootstrap standard errors makes sense if asymptotic standard errors are not available or may be seriously unreliable. It is good to compute both to see how well they agree.

The assumption that $\hat{\theta}$ is unbiased and approximately normally distributed may be uncomfortably strong.

Using bootstrap standard errors makes sense if asymptotic standard errors are not available or may be seriously unreliable. It is good to compute both to see how well they agree.

The assumption that $\hat{\theta}$ is unbiased and approximately normally distributed may be uncomfortably strong.

More reliable tests and confidence intervals can be obtained if we can compute a bootstrap test statistic, say τ_b^*, for every bootstrap sample.

Using bootstrap standard errors makes sense if asymptotic standard errors are not available or may be seriously unreliable. It is good to compute both to see how well they agree.

The assumption that $\hat{\theta}$ is unbiased and approximately normally distributed may be uncomfortably strong.

More reliable tests and confidence intervals can be obtained if we can compute a bootstrap test statistic, say τ_b^*, for every bootstrap sample.

- This might (or might not) be $(\hat{\theta}_b^* - \theta_0)/s_b^*$, where s_b^* is an asymptotic standard error calculated at the same time as $\hat{\theta}_b^*$.

Using bootstrap standard errors makes sense if asymptotic standard errors are not available or may be seriously unreliable. It is good to compute both to see how well they agree.

The assumption that $\hat{\theta}$ is unbiased and approximately normally distributed may be uncomfortably strong.

More reliable tests and confidence intervals can be obtained if we can compute a bootstrap test statistic, say τ_b^*, for every bootstrap sample.

- This might (or might not) be $(\hat{\theta}_b^* - \theta_0)/s_b^*$, where s_b^* is an asymptotic standard error calculated at the same time as $\hat{\theta}_b^*$.
- We can then use the distribution of the τ_b^* to estimate the distribution of the actual test statistic τ.

Using bootstrap standard errors makes sense if asymptotic standard errors are not available or may be seriously unreliable. It is good to compute both to see how well they agree.

The assumption that $\hat{\theta}$ is unbiased and approximately normally distributed may be uncomfortably strong.

More reliable tests and confidence intervals can be obtained if we can compute a bootstrap test statistic, say τ_b^*, for every bootstrap sample.

- This might (or might not) be $(\hat{\theta}_b^* - \theta_0)/s_b^*$, where s_b^* is an asymptotic standard error calculated at the same time as $\hat{\theta}_b^*$.
- We can then use the distribution of the τ_b^* to estimate the distribution of the actual test statistic τ.

In theory, tests and confidence intervals based on comparing τ with the τ_b^* may provide an **asymptotic refinement**.

Using bootstrap standard errors makes sense if asymptotic standard errors are not available or may be seriously unreliable. It is good to compute both to see how well they agree.

The assumption that $\hat{\theta}$ is unbiased and approximately normally distributed may be uncomfortably strong.

More reliable tests and confidence intervals can be obtained if we can compute a bootstrap test statistic, say τ_b^*, for every bootstrap sample.

- This might (or might not) be $(\hat{\theta}_b^* - \theta_0)/s_b^*$, where s_b^* is an asymptotic standard error calculated at the same time as $\hat{\theta}_b^*$.
- We can then use the distribution of the τ_b^* to estimate the distribution of the actual test statistic τ.

In theory, tests and confidence intervals based on comparing τ with the τ_b^* may provide an **asymptotic refinement**.

If so, mistakes made by a bootstrap test are of lower order in N than mistakes made by the asymptotic test on which it is based.

Bootstrap Quantiles

Bootstrap Quantiles

It does not make sense to compute a bootstrap standard error for an estimator $\hat{\theta}$ that does not have a finite variance.

Bootstrap Quantiles

It does not make sense to compute a bootstrap standard error for an estimator $\hat{\theta}$ that does not have a finite variance.

In such cases, the bootstrap variance (13) will not converge.

Bootstrap Quantiles

It does not make sense to compute a bootstrap standard error for an estimator $\hat{\theta}$ that does not have a finite variance.

In such cases, the bootstrap variance (13) will not converge.

Whenever $\hat{\theta}$ has a finite variance but an infinite fourth moment, the bootstrap variance will probably converge very slowly.

Bootstrap Quantiles

It does not make sense to compute a bootstrap standard error for an estimator $\hat{\theta}$ that does not have a finite variance.

In such cases, the bootstrap variance (13) will not converge.

Whenever $\hat{\theta}$ has a finite variance but an infinite fourth moment, the bootstrap variance will probably converge very slowly.

In such cases, it is much better to estimate bootstrap quantiles than anything based on sample moments of the bootstrap distribution.

Bootstrap Quantiles

It does not make sense to compute a bootstrap standard error for an estimator $\hat{\theta}$ that does not have a finite variance.

In such cases, the bootstrap variance (13) will not converge.

Whenever $\hat{\theta}$ has a finite variance but an infinite fourth moment, the bootstrap variance will probably converge very slowly.

In such cases, it is much better to estimate bootstrap quantiles than anything based on sample moments of the bootstrap distribution.

A simple alternative to the bootstrap standard error is a function of the rescaled **interquartile range**, or **IQR**, which is the difference between the third and first quartiles of the $\hat{\theta}^*_b$.

Bootstrap Quantiles

It does not make sense to compute a bootstrap standard error for an estimator $\hat{\theta}$ that does not have a finite variance.

In such cases, the bootstrap variance (13) will not converge.

Whenever $\hat{\theta}$ has a finite variance but an infinite fourth moment, the bootstrap variance will probably converge very slowly.

In such cases, it is much better to estimate bootstrap quantiles than anything based on sample moments of the bootstrap distribution.

A simple alternative to the bootstrap standard error is a function of the rescaled **interquartile range**, or **IQR**, which is the difference between the third and first quartiles of the $\hat{\theta}_b^*$.

If we sort the $\hat{\theta}_b^*$ from smallest to largest, then the first quartile is approximately number $B/4$, and the third quartile is approximately number $3B/4$, in the sorted list.

If we choose $B = 99$, then they are numbers 25 and 75.

If we choose $B = 99$, then they are numbers 25 and 75.

More generally, whenever $B + 1$ is divisible by 4, they are numbers $(B + 1)/4$ and $(3B + 1)/4$.

If we choose $B = 99$, then they are numbers 25 and 75.

More generally, whenever $B + 1$ is divisible by 4, they are numbers $(B + 1)/4$ and $(3B + 1)/4$.

Because the first and third quartiles of the standard normal distribution are -0.6744898 and 0.6744898, the IQR for it is 1.349.

If we choose $B = 99$, then they are numbers 25 and 75.

More generally, whenever $B + 1$ is divisible by 4, they are numbers $(B + 1)/4$ and $(3B + 1)/4$.

Because the first and third quartiles of the standard normal distribution are -0.6744898 and 0.6744898, the IQR for it is 1.349.

Thus $\widehat{\mathrm{IQR}}/1.349$ is an estimator of $\sigma(\hat{\theta})$. It is not an efficient estimator under normality, but it works far better than $\mathrm{se}^*(\hat{\theta})$ when the bootstrap distribution has thick tails. If efficiency is an issue, make B larger.

If we choose $B = 99$, then they are numbers 25 and 75.

More generally, whenever $B + 1$ is divisible by 4, they are numbers $(B + 1)/4$ and $(3B + 1)/4$.

Because the first and third quartiles of the standard normal distribution are -0.6744898 and 0.6744898, the IQR for it is 1.349.

Thus $\widehat{\text{IQR}}/1.349$ is an estimator of $\sigma(\hat{\theta})$. It is not an efficient estimator under normality, but it works far better than $\text{se}^*(\hat{\theta})$ when the bootstrap distribution has thick tails. If efficiency is an issue, make B larger.

It is often a good idea to calculate $\widehat{\text{IQR}}/1.349$ and compare it with the bootstrap standard error.

If we choose $B = 99$, then they are numbers 25 and 75.

More generally, whenever $B + 1$ is divisible by 4, they are numbers $(B + 1)/4$ and $(3B + 1)/4$.

Because the first and third quartiles of the standard normal distribution are -0.6744898 and 0.6744898, the IQR for it is 1.349.

Thus $\widehat{\text{IQR}}/1.349$ is an estimator of $\sigma(\hat{\theta})$. It is not an efficient estimator under normality, but it works far better than $\text{se}^*(\hat{\theta})$ when the bootstrap distribution has thick tails. If efficiency is an issue, make B larger.

It is often a good idea to calculate $\widehat{\text{IQR}}/1.349$ and compare it with the bootstrap standard error.

If bootstrap standard errors do not seem to converge at rate $1/B$ as B increases, or if there is reason to suspect that $\hat{\theta}$ has thick tails, use $\widehat{\text{IQR}}/1.349$ instead of the bootstrap standard error.

If we choose $B = 99$, then they are numbers 25 and 75.

More generally, whenever $B + 1$ is divisible by 4, they are numbers $(B + 1)/4$ and $(3B + 1)/4$.

Because the first and third quartiles of the standard normal distribution are -0.6744898 and 0.6744898, the IQR for it is 1.349.

Thus $\widehat{\text{IQR}}/1.349$ is an estimator of $\sigma(\hat{\theta})$. It is not an efficient estimator under normality, but it works far better than $\text{se}^*(\hat{\theta})$ when the bootstrap distribution has thick tails. If efficiency is an issue, make B larger.

It is often a good idea to calculate $\widehat{\text{IQR}}/1.349$ and compare it with the bootstrap standard error.

If bootstrap standard errors do not seem to converge at rate $1/B$ as B increases, or if there is reason to suspect that $\hat{\theta}$ has thick tails, use $\widehat{\text{IQR}}/1.349$ instead of the bootstrap standard error.

If desired, we can plot the EDF of the $\hat{\theta}^*_b$, or a smoothed EDF, or compute many quantiles, including extreme ones, to see what the bootstrap distribution looks like.