Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 13 February 2023 doi:10.20944/preprints202302.0213.v1

Disclaimer/Publisher's Note: The statements, opinions, and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions, or products referred to in the content.

dbs: Stata command to compute double bootstrap confidence intervals [Working Paper / February 2023]

Felix Bittmann
Leibniz Institute for Educational Trajectories &
University of Bamberg
Bamberg, Germany
felix.bittmann@lifbi.de

Abstract. Bootstrapping is a flexible, powerful and well-established statistical approach to quantify the uncertainty of virtually any point estimate. While multiple versions of bootstrap confidence intervals are already available in Stata, `dbs` implements the double (iterated) bootstrap. Instead of relying on parametric assumptions such as the non-parametric resampling bootstrap confidence interval does, it is more flexible and derives critical values directly from that data. To do so, multiple methods are available (analytic approach, double resampling, jackknife estimation). In a comparative simulation study it is empirically demonstrated that the strengths of the double bootstrap are particularly evident for small samples ($n < 100$) when heteroscedasticity is present. While all other approaches result in undercoverage, only the double bootstrap reaches the target coverage level and hence avoids incorrect statistical conclusions. The computational burden is not even necessarily larger than for other bootstrap approaches.

Keywords: st0001, bootstrapping, confidence intervals, dbs, resampling, variance estimation, heteroscedasticity

1 The logic of bootstrapping

Quantifying uncertainty around point estimates is one of the central goals in statistics. Whenever not entire populations can be studied but only samples are available, the question arises whether a finding can be generalised to the group of individuals or cases that were not part of the study. Standard errors are a crucial tool to achieve this as they allow the computation of p-values or derived statistics like confidence intervals (CIs) to estimate the variability of point estimates. For many applications, analytic approaches and formulas are available to compute these standard errors. Yet, there are certain downsides. First, these computations can be computational challenging, second, they rely on certain statistical assumptions (parametric methods) and third, sometimes they are not available at all.

Bootstrapping is a conceptually simple yet powerful alternative approach to analytic methods to estimate standard errors and derived statistics. They are highly flexible and come in many different versions to fit virtually any situation. Bootstrapped confidence intervals have been established for a long time (DiCiccio and Efron 1996). Especially

© 2023 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 13 February 2023 doi:10.20944/preprints202302.0213.v1

since the American Statistical Association recommends abandoning classical p-values as they are unidimensional and often lead to binary conclusions (Wasserstein et al. 2019), confidence intervals are a more sophisticated option as they enable quantifying range of probable outcomes. To understand the logic behind the approach, recall what a standard error is, a statistic to quantify the variability of a measurement. Usually, researchers collect a single sample from the population and compute the statistic of interest. Imagine if one could not only collect a single sample but many more, independently of each other. By computing the statistic of interest for each sample, one would receive a large number of point estimates, which taken together form the so called sampling distribution of this very statistic and easily allow the estimation of its variability. Unfortunately, collecting that many samples is usually not feasible due to restrictions of time and funding.

Enter the bootstrap. The idea is to treat the available sample as the population and sample randomly and repeatedly with replacement from this sample, which is easily achieved using modern hardware. By doing so one can generate many resamples, which taken together form the bootstrap sampling distribution. Computing the standard deviation of this distribution gives the standard error for the statistic of interest, which can subsequently be used to compute p-values or confidence intervals (Efron and Tibshirani 1994). What sounds simple has been applied successfully in all areas of science in the last decades and is easily implemented in most modern programming languages. The advantages are clear: the approach can be applied to virtually any statistic and no new algorithms have to be derived. It is conceptually simple to understand and easy to implement. Overall, as will be shown below, the approach often outperforms parametric approaches in terms of accuracy and leads to better conclusions. One especially promising candidate, which has not been available in Stata so far, is the double or iterated bootstrap, which is the main topic of this article.

The following article first presents the algorithms and theoretical expectations of normal-based and double bootstrap confidence intervals and then introduces `dbs`, a Stata command to compute the double bootstrap confidence intervals for virtually any statistic. Finally, two simulation studies are conducted to investigate how the double bootstrap performs in commonplace data scenarios.

2 Normal based and double bootstrap confidence intervals

First we start with the general bootstrap algorithm and then see how the double bootstrap differs. The algorithm for the normal-based bootstrap approach is as follows:[1]

1. Compute the statistic of interest for the available data as this is the point estimate ($\hat{\theta}$, theta hat). Now draw J random samples with replacement independently from the original sample. The sample size of each resample is equal to the original sample size.

1. Other bootstrap approaches like the percentile, BC and BCa work slightly differently. These algorithms will not be explained but still be tested further below.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 13 February 2023 doi:10.20944/preprints202302.0213.v1

2. Compute the statistic of interest of each resample ($\hat{\theta}^{*j}$), where j is the index from 1 to J. The asterisk after the theta indicates that this statistic is from a bootstrap resample. Store these statistics and compute the standard deviation of the collection, which gives the bootstrap standard error $SE_{Boot}(\hat{\theta})$.

3. Choose your error level α and compute the critical value using the inverse cumulative standard normal distribution Φ^{-1}: $tcrit_{lower} = \Phi^{-1}(0.5 \cdot \alpha)$ and $tcrit_{upper} = \Phi^{-1}(1-(0.5 \cdot \alpha))$. For example, if α is set to 0.05 (thus generating a 95%-CI), the critical values are approximately ± 1.96.

4. Compute the CI as follows: $(\hat{\theta} - tcrit_{lower} SE_{Boot}(\hat{\theta}); \hat{\theta} + tcrit_{upper} SE_{Boot}(\hat{\theta}))$.

The basic approach is easy to understand and implement. However, this simplicity comes with a few assumptions, even if bootstrapping is usually referred to as a non-parametric technique. Apparently, the critical limits that are used to find the respective ends of the interval are based on the inverse standard normal distribution (in infinite samples) or the t-distribution (as soon as the sample size is large enough differences between the two become marginal). This makes sense as long as the actual bootstrap distribution is indeed normal.[2] The normal-based approach gives precise results as long as the statistic of interest and the standard deviation are independent.[3] However, as soon as the original data is positively skewed, $\hat{\theta}$ and s are positively correlated and the t statistic does not follow a t distribution, which obstructs inference. Note that this problem is sometimes only amended when samples become extremely large (Hesterberg 2015, p.56). If this occurs, the t values used for the critical values of the endpoints of the CI will not give accurate results, leading to wrong conclusions.

To avoid this potential pitfall, the double bootstrap offers an alternative approach. Instead of referring to the standard normal distribution (or t-distribution), it would be desirable if one could generate the correct distribution for this specific variable from the actual data. The general idea to achieve this is to studentize each bootstrap resample (see step 2 in the algorithm below). This is done so that the data are transformed to have a mean of zero and a standard deviation of one. By doing so, the statistic of interest in the given resample is compared to the factual statistic in the sample and the distance is standardized by the standard error of the resample. This allows the computation of the factual bootstrap sampling distribution as a t-distribution. The main obstacle in this process is that the standard error of each bootstrap resample must be known as this is the denominator of the equation. There are two options: Either the standard error can be derived analytically, which is possible as long as there are algorithms available to compute this (for example, for the standard error of the mean). However, if no such algorithm exists (or is not feasible to compute), other means are necessary, for example the jackknife or simply a second round of bootstrapping. By doing so, the current resample is then used as the 'population' for a second round for random resamples and the same process is applied to compute a standard error. The

2. For a basic and visual explanation of what sampling distributions are and why they are of special interest for bootstrapping, refer to Hesterberg (2015).

3. To be precise: the t-statistic $t = \frac{\hat{\theta}-\theta}{s/\sqrt{n}}$ only has a t-distribution if this holds.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 13 February 2023 doi:10.20944/preprints202302.0213.v1

advantage of this procedure is that it is general and can be applied to any statistic. The downside is the additional computational burden. It follows the precise algorithm for the double bootstrap:

1. Draw J random samples with replacement independently from the original sample. The sample size of each resample is equal to the original sample size.

2. Compute the statistic of interest of each resample ($\hat{\theta}^{*j}$), where j is the index from 1 to J. Studentize each resample in relation to the original data as follows: $t^{*j} = \frac{\hat{\theta}^{*j} - \hat{\theta}}{SE(\hat{\theta}^{*j})}$. By doing so, a t-value is generated for each bootstrap resample. The numerator of the equation is the difference between the bootstrapped theta and the factual theta. In the denominator we need the standard error of the bootstrapped theta to standardize the difference of the numerator.

3. Therefore, compute the standard error analytically. If this is not feasible, a second round of bootstrapping is started for every resample. Thus, to generate the standard error, K random and independent resamples are drawn from the "outer" bootstrap resample and the same principle is applied. By doing so, the standard error is available for any statistic of interest. Theoretically, any other approach to compute standard errors can be used here, for example the jackknife (Efron and Tibshirani 1994, p.141-152).

4. This process is carried out for all outer resamples. After that, the bootstrap t-distribution is generated, which is used instead of the standard normal distribution. Choose your error level α and compute the critical value using the respective percentiles P of the t-distribution: $tcrit_{lower} = P(0.5 \cdot \alpha)$ and $tcrit_{upper} = P(1 - (0.5 \cdot \alpha))$.

5. Compute the CI as follows: $(\hat{\theta} - tcrit_{upper} SE(\hat{\theta}); \hat{\theta} - tcrit_{lower} SE(\hat{\theta}))$.[4].

While this general idea is quite old and has already been introduced together with the basics of bootstrapping (Efron and Tibshirani 1994), the usage was usually not feasible due to the additional computational burden. Since for every outer bootstrap resample (that is, resampling from the original data), an inner round of bootstrapping is necessary (thus resampling from the bootstrap resample), the computational costs increases K-fold. While the number of inner resamples is usually much lower than for the outer ones (values between 100 and 500 seem adequate for most applications), the computational work is still immense. Nevertheless, the ever growing computational capacities render the method achievable for most applications on modern systems. Further below we will test empirically how many inner resamples are necessary to compute precise results.

There are two important properties of double bootstrap confidence intervals. First, the intervals are second-order accurate, meaning that the difference between the nominal

4. Note that the *lower* critical value is utilized to compute the *upper* bound of the interval and vice versa. The reason for this is basic algebra, a detailed explanation is given in (Hesterberg 2015, p.57)

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 13 February 2023 doi:10.20944/preprints202302.0213.v1

coverage and the factual coverage approaches zero at a rate of n^{-1} (where n is the sample size). This is a clear advantage over the non-parametric normal-based bootstrap confidence intervals, which are only first-order[5] accurate (McCullough and Vinod 1998; Chernick 2011; Martin 1990). The second important property is that in contrast to some other bootstrap approaches like the percentile, double bootstrap confidence intervals are not transformation invariant. For example, this means that it does make a difference whether to obtain confidence intervals for *logits* or *odds ratios*. In this respect the double bootstrap behaves like the non-parametric normal-based approach. Users should be aware of this. In the following section, a command is introduced that implements the described algorithm in Stata.

3 A command for double bootstrap confidence intervals

In this section the command `dbs` is introduced and available options are explained.[6]

3.1 Syntax and options

The usage of `dbs` is similar to the standard `bootstrap` prefix.

`dbs` *exp_list* `[,` *options*`]` `:` *command*

Here, *exp_list* is a list of return values produced by *command*. *Command* is any command that follows standard Stata syntax. The options are as follows:

`reps`(integer) specifies the number of outer resamples to draw. The default is 50. More is always better and for precise results, 10,000 or more resamples are advised (Hesterberg 2015).

`repsinner`(integer) specifies the number of inner resamples to draw. The default is 20. This is the number of resamples drawn for every outer resample. The total number of bootstrap samples is thus reps * repsinner, so be aware of the additional computational burden. The number of inner resamples is usually much lower than for outer resamples and values between 100 and 500 are probably fine. Overwritten if *analytic* or *jackknife* is specified.

`analytic`(string) specifies analytic standard errors for the computation of the t-values. This is possible if the command used provides analytic standard errors. The order of the analytic standard errors must be identical to the order of expressions in *exp_list*. The provision of analytic standard errors increases the overall computation speed manifold and often gives highly accurate results. Keep in mind that analytic standard errors usually come with more (parametric) assumptions, so this is a potential point of failure.

5. For them the rate is slower as it is only $n^{-1/2}$, meaning that larger samples are required for the same level of accuracy.
6. To install the most up-to date version of the command, see https://github.com/fbittmann/dbs

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 13 February 2023 doi:10.20944/preprints202302.0213.v1

`jackknife` specifies the usage of jackknife standard errors for the computation of the t-values. Use either *repsinner*, *analytic* or *jackknife*.

`level`**(integer)** specifies the confidence level, as a percentage, for confidence intervals. The default is **level(95)** which produces 95% confidence intervals.

`seed`**(integer)** sets the random-number seed.

`strata`**(varlist)** specifies the variables that identify strata. If this option is specified, bootstrap samples are taken independently within each stratum.

`cluster`**(varlist)** specifies the variables that identify resampling clusters. If this option is specified, the sample drawn during each replication is a bootstrap sample of clusters.

`idcluster`**(newvar)** creates a new variable containing a unique identifier for each resampled cluster. This option requires that cluster() also be specified.

`saving`**(filename)** creates a Stata data file (**.dta** file) consisting of (for each statistic in exp_list) a variable containing the replicates.

`dots`**(integer)** displays dots every # replications. By default, one dot character is displayed each ten successful replications. When dots(0) is specified, no dots are displayed. No dots are displayed when multiple threads are specified.

`graph` displays diagnostic quantile-quantile plots for the generated t-values for each statistic of interest. If the t-values deviate from a normal distribution the double bootstrap will produce more accurate results than the normal-based bootstrap CIs. Of special interest are the tails of the distribution - even small deviations in these regions legitimate the use of the double bootstrap approach. The Shapiro-Francia test is a numerical test for normality; a small p-value indicates non-normality.

`nowarn` suppresses the display of a warning message when *command* does not set **e(sample)**.

`parallel`**(integer)** allows the usage of multiple threads to speed up computation. This function makes use of the Stata program `parallel` (Vega Yon and Quistorff 2019). This package must be installed if more than one thread is supposed to be used. For details refer to package documentation. If more threads than actually available threads are specified, the computer might crash! If you want to run parallel with user-written programs, these cannot be defined "on the fly" (in the same do file). Save the program in an .ado file and place it in the correct folder (for example, if the program is called `xcom`, then in "/ado/plus/x/". Restart Stata afterwards. Otherwise `parallel` cannot find the command.

The command stores relevant results and statistics in scalars and matrices for further processing if so desired.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 13 February 2023 doi:10.20944/preprints202302.0213.v1

4 Example

In this section the usage of `dbs` is introduced to give a few examples of how it can be applied. We start with a minimal example and bootstrap the arithmetic mean and of *price*. For comparison, the results using regular `bootstrap` are also provided.

```
. version 16.1
. sysuse auto, clear
(1978 Automobile Data)
. dbs r(mean), reps(2500) repsinner(300) graph seed(123) dots(0) nowarn: ///
> summarize price, meanonly
Bootstrap results                                       Number of obs = 74
                                                                Reps = 2500
                                                      Reps (inner) = 300
command:  summarize price, meanonly
    _bs_1: r(mean)
```

	Observed Coef.	Bootstrap Std. Err.	Bias	Shapiro-Francia	[95% Conf.	Interval]
_bs_1	6165.2568	337.8183	12.525	0.000	5548.8943	6947.5050

```
. bs r(mean), reps(2500) seed(123) dots(0) nowarn: ///
> summarize price, meanonly
Bootstrap results                               Number of obs     =         74
                                                Replications      =      2,500
      command:  summarize price, meanonly
        _bs_1:  r(mean)
```

	Observed Coef.	Bootstrap Std. Err.	z	P>|z|	Normal-based [95% Conf.	Interval]
_bs_1	6165.257	334.7477	18.42	0.000	5509.163	6821.35

The usage of `dbs` is very similar to the non-parametric bootstrap prefix and the results are structured likewise. The main information is the standard error, the bias (as a rule of thumb, the ratio bias to standard error should be smaller than 0.25) and of course the limits of the confidence interval. The Shapiro-Francia statistic tests numerically whether the bootstrap t-value distribution differs from the normal one (Mbah and Paothong 2015). If this value is small, the null hypothesis has to be rejected and we must assume that it is non-normal. If this is the case, we know that a non-parametric bootstrap confidence interval would give biased results since the critical values deviate from the factual distribution. However, note that the Shapiro-Francia test is often misleading when the number of replications is small (< 150). As we see in the given example, the p-value is very small, so the double bootstrap should give more precise results than the non-parametric bootstrap. We can also see this with the created graph, see (Figure 1). The deviations at the tails of the distribution appear to be minor but are of special interest since it is usually the tails that affect the critical values. Consequently, even minor deviations from the straight line can affect the critical values of the confidence intervals massively (Hesterberg 2015).

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 13 February 2023 doi:10.20944/preprints202302.0213.v1

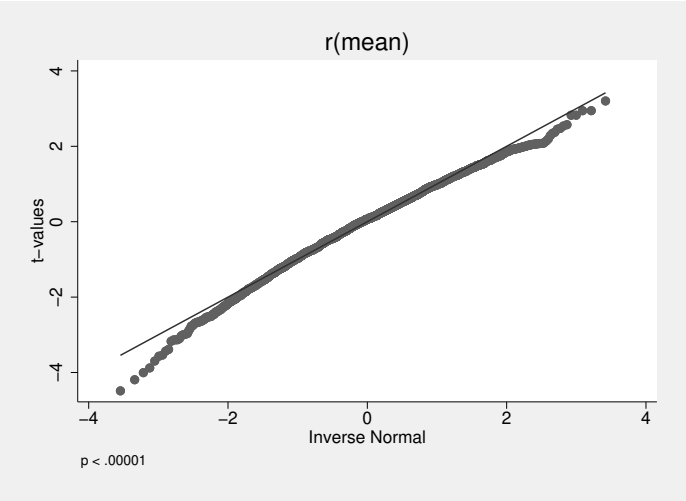

Figure 1: Quantile-quantile plot for the generated bootstrap t-distribution. The apparent deviations in the tails of the distribution signify that the normal-based bootstrap CIs might be biased since the normality assumption is violated.

When we compare the CIs, we note two things: First, the DBS CI is about 6.2% larger than the BS CI. As the graph indicates, we have good arguments to believe that the BS interval might be too short due to the apparent deviations from the normal distribution. Second, in comparison to the BS interval, which is always symmetric, this does not hold for the DBS interval. The reason for this is that the target variable *price* is highly skewed. In the simulations below we will compare the quality of bootstrap CIs in more detail.

It is also possible to generate CIs for results from regression-type commands using Stata's way to directly access the coefficients. As `regress` also provides standard errors for the coefficients, we can use them to speed up the computation. To find them, use *matrix list r(table)* after running `regress` and provide them in the option *analytic*. It is important that the order of the standard errors corresponds to the order of the expressions when calling `dbs`. For more examples with bootstrapping in Stata, especially when clustered or longitudinal data are used, see Bittmann (2021).

```
. dbs _b[weight] _b[_cons], reps(2500) analytic(r(table)[2,1] r(table)[2,2]) ///
> seed(123) dots(0) nowarn: ///
> regress mpg weight

Bootstrap results                                       Number of obs = 74
                                                                Reps = 2500
                                                        Reps (inner) = 0

command:  regress mpg weight
analytic standard error(s) provided (shown in brackets)
    _bs_1: _b[weight] [r(table)[2,1]]
    _bs_2: _b[_cons] [r(table)[2,2]]
```

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 13 February 2023 doi:10.20944/preprints202302.0213.v1

	Observed Coef.	Bootstrap Std. Err.	Bias	Shapiro-Francia	[95% Conf.	Interval]
_bs_1	-0.0060	0.0006	-0.000	0.000	-0.0074	-0.0048
_bs_2	39.4403	1.9754	0.052	0.000	34.8309	44.7620

Some additional practical advice: Whenever possible use analytic standard errors for the inner bootstrap loop since this increases computational speed manifold in comparison to the other options and gives precise results. Luckily, many of the most widely used statistical techniques, like regression models or even matching, provide analytic standard errors. However, if this is not the case, the user can either use an inner bootstrap loop or the jackknife. The jackknife has the advantage that computational times only depend on the size of the sample and no specification has to be set for the number of inner resamples. As will be demonstrated further below, 300 or more inner resamples might be necessary to obtain precise results. If the user wants to rely on any of the two resampling options despite the availability of analytic standard errors, it makes sense to have them not computed at all (use options like *nostderr* for multilevel models or *nose* for `margins`).

5 Simulation studies

In the following section two simulations are presented. The first compares various methods to generate CIs to each other to see which one performs best (regarding coverage and width). By doing so, researchers can gauge how the double bootstrap performs compared to various other approaches. The second simulation tests how the performance of the double bootstrap behaves when changing the number of inner bootstrap resamples to take. This is of interest so the reader can estimate how many resamples should be taken to obtain precise results. All simulations are conducted in Stata 16.1.

5.1 Simulation 1: Comparing CI algorithms

Currently there are already four different types of bootstrap CIs implemented in Stata: normal based, percentile, bias-corrected (BC), and bias-corrected and accelerated (BCa).[7] Naturally the question arises which is the "best" choice. In the literature there is no consensus of which approach to use and different CIs perform differently, depending on the data and type of statistic of interest. To test this empirically, a simulation study is conducted. In a simulation, the researcher generates data so the "true" outcome is known and therefore can be used to gauge which method is the best to recover this ground truth. For the following simulation, an OLS regression was chosen for demonstration since it is a very popular method in many fields and hence reflects a "typical"

7. The three latter ones do not rely on the computed standard error and the critical values taken from the cumulative standard normal distribution but take the actual distribution of the generated bootstrap sampling distribution into account. BC and BCa also implement some correction factors if this distribution differs strongly from the original sample.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 13 February 2023 doi:10.20944/preprints202302.0213.v1

scenario. It also represents a kind of ideal case for normal based approaches because regression coefficients, whether estimated via OLS or maximum-likelihood, are (asymptotically) normal. In total, two variables are introduced, which are systematically varied to test how the performance of bootstrap CIs is affected by them. The first is the sample size with four levels (15, 30, 70, 200). 15 represents a very small sample where the central limit theorem usually does not apply. The question arises how bootstrapping performs when parametric assumptions do not hold. 30 is the usually minimally accepted size where the theorem should be valid, meaning that the normal distribution of estimates should hold. 70 and 200 represent an average and "comfortable" sample size for regressions with few explanatory variables.

The second variable is the skewness of the dependent variable. It is known that OLS regression methods give the best results when the dependent variable is approximately normally distributed. However, this is often violated due to the nature of the data. Many variables do not follow such a distribution, for example wage data. Researchers then either apply a transformation (e.g., log transformation) or simply ignore the non-normal distribution. We test empirically how such a deviation can affect the inference. Lastly, heteroscedasticity is introduced, which is a common problem in applied statistics and can bias standard errors (albeit not the point estimates, so only the inference is affected). This means that the variance of the outcome variable is no longer independent of the explanatory variable (the variance of the disturbance term is no longer constant). One common remedy to combat heteroscedasticity in OLS is to apply robust ("Hubert-White") standard errors, which is, however, no panacea (King and Roberts 2015).

For the simulation, the true regression model looks like this:

$$y = x_1 + x_2 + e \tag{1}$$

With $x_1 \sim \mathcal{N}(0, 1)$. In the scenarios without skewness, x_2 follows the same distribution as x_1. However, when skewed data is to be generated, the condition is $x_2 \sim \mathcal{B}(5, 1.5)$ (beta distribution). To increase the influence of this skewed, variable the term x_2 is also multiplied by 25 in this scenario. In the baseline model without heteroscedasticity, the disturbance term e is also normally distributed ($e \sim \mathcal{N}(0, 1)$). When heteroscedasticity is specified, e is dependent on x_1 ($e \sim \mathcal{N}(0, 1)e^{0.6x_1}$). Summarized, there are four scenarios (no skewness and no heteroscedasticity; skewness and no heteroscedasticity; no skewness and heteroscedasticity, skewness and heteroscedasticity) and four sample sizes, which gives a total of 16 simulations to conduct.

There are two main statistics that are to be evaluated using the simulations. The first is the coverage, which is the probability that the true effect is included in the computed confidence interval. Usually, researchers set the alpha-level error (for example, 5%) which corresponds to a 95% confidence interval. This is the probability to incorrectly reject the null hypothesis. The main assumption is that on average, the coverage should reflect this alpha level. However, it is well known that when certain statistical assumptions are violated, the coverage can be off. For example, if the coverage is smaller than the target level (that is, below 95% in this example), undercoverage is present and

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 13 February 2023 doi:10.20944/preprints202302.0213.v1

it gets "easier" to reject the null hypothesis and more false-positive findings occur. If there is overcoverage, the CI is "conservative". Both errors are problematic in science since they lead to wrong conclusions. Optimally, the coverage should closely reflect the target level set by the researchers. The second statistic to observe is the width of the CI. This is simply the difference of the upper limit and the lower limit. Ideally, the width of the CI should be as small as possible while still having the correct coverage. The broader the width, the less precise the results are. In the simulation, the results are evaluated for x_1. As the true coefficient is 1, it can be tested whether the computed CIs include this value or not. Potentially, the coverage is the more relevant statistic since here wrong conclusions can arise. An incorrect width, in contrast, lowers the precision of the findings and makes it more difficult to pinpoint the "true" effect.

Five bootstrap CIs are compared: normal, percentile, BC, BCa, and the double bootstrap. The first four are available in Stata per default, the double is implemented using `dbs`. As a benchmark, the regular OLS CIs and the robust version are reported as well. Per scenario, 5000 simulations are computed. The bootstrap relies on 1500 random resamples per computation.[8] The double bootstrap uses the standard errors provided by OLS to increase precision and computational speed. The results for the coverage are shown in the following four figures, including 95% confidence intervals to visualize the Monte-Carlo error. Detailed numerical results, including the width of the CIs, are printed in the appendix (Table 1).

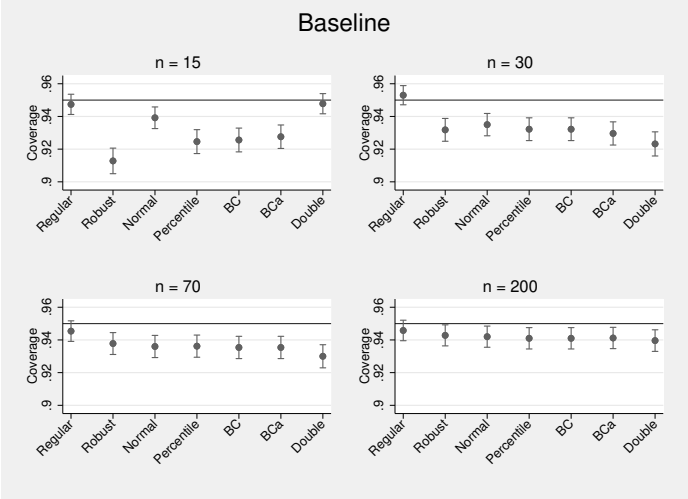

Figure 2: Coverage by type of CI, baseline model. 95% confidence intervals depicted to visualize Monte-Carlo error. 5000 simulations, 1500 bootstrap resamples.

In the baseline model (Figure 2), the regular OLS standard error is the only method that always performs well, independent of the sample size. The target coverage of 95% is always reached. All bootstrap methods lead to undercoverage, while the error declines

8. Parts of the study were repeated using 5000 bootstrap resamples with very similar results.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 13 February 2023 doi:10.20944/preprints202302.0213.v1

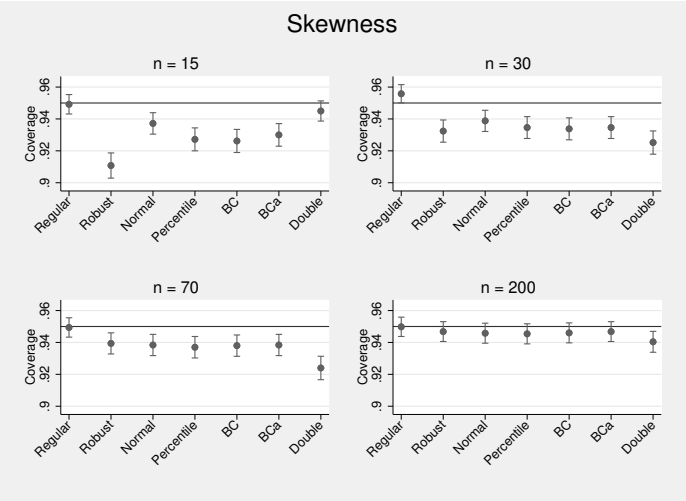

Figure 3: Coverage by type of CI, skewness present. 95% confidence intervals depicted to visualize Monte-Carlo error. 5000 simulations, 1500 bootstrap resamples.

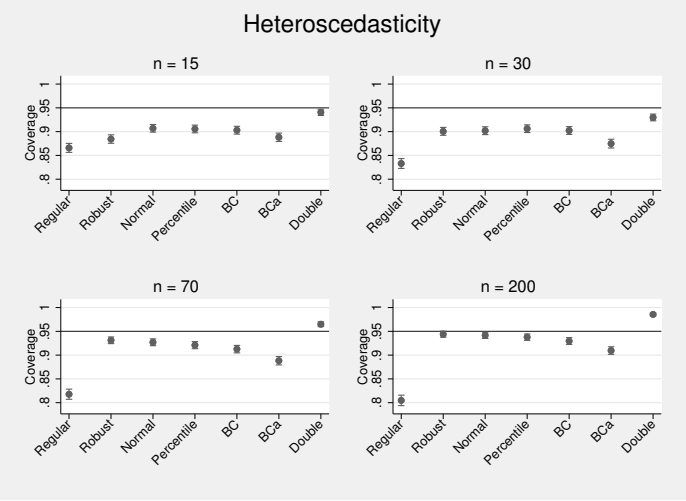

Figure 4: Coverage by type of CI, heteroscedasticity present. 95% confidence intervals depicted to visualize Monte-Carlo error. 5000 simulations, 1500 bootstrap resamples.

with growing sample sizes. To be crystal clear, when the data available are normally distributed without any heteroscedasticity present, researchers are advised to rely on the OLS standard error since the width of the CIs is also the shortest for this method.

The second scenario (Figure 3) is a skewed dependent variable. The conclusions are

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 13 February 2023 doi:10.20944/preprints202302.0213.v1

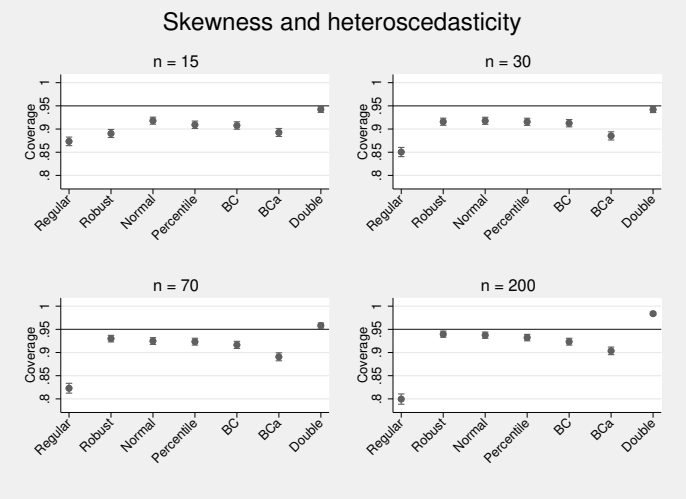

Figure 5: Coverage by type of CI, skewness and heteroscedasticity present. 95% confidence intervals depicted to visualize Monte-Carlo error. 5000 simulations, 1500 bootstrap resamples.

very similar to the normally distributed one; the regular OLS standard error delivers a fantastic performance and bootstrapping is not beneficial in this scenario either.

The third scenario introduces heteroscedasticity (Figure 4). The first thing to notice is that the overall coverages are suddenly much too low. For the first two scenarios virtually all coverages were above 92%, so rather close to the target. This is now very different as coverages as low as only 80% are reported, which means huge errors are present. Here, the OLS standard approach performs abysmally. Even the "robust" standard errors come with rather large errors, even though they are specifically designed to combat heteroscedasticity. This should, again, alert researchers that using this kind of standard error is not a general solution that resolves any issue of heteroscedasticity. Interestingly, bootstrapping in general is also not a simple solution as even "advanced" and "corrected" approaches such as BC and BCa perform often even worse than the robust standard errors. Finally, the double bootstrap performs very well as the coverage is close to the target value, especially in smaller samples (70 or fewer observations). When we look at the appendix, we notice that the other CIs are usually too short, leading to undercoverage. The double bootstrap CI, which is always the largest, delivers the correct width. In the larger sample it tends to slightly overcoverage.

The fourth and final scenario combines heteroscedasticity and skewness, however, the overall results are very similar to the third one. As soon as heteroscedasticity is present, most approaches perform badly, especially when the sample sizes are smaller. Again, especially when sample sizes are smaller, the performance of the double bootstrap is close to ideal.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 13 February 2023 doi:10.20944/preprints202302.0213.v1

5.2 Simulation 2: Number of inner resamples to take

As has been discussed above when introducing the algorithm for the double bootstrap, researchers have to specify the number of inner resamples to take when no analytic standard errors are available by the command used (alternatively, they can also use the jackknife). The question is how many inner resamples are required for precise results. Based on the literature, which is rather old, it is difficult to give a general recommendation (Lee and Young 1999; Booth and Hall 1994). However, we can conduct a simulation to gauge how the precision of the computed CI varies with the number of inner resamples.

In this second simulation we will generate a normally distributed outcome variable with a mean of 0 ($y \sim \mathcal{N}(0, 1)$), the sample size is set to 500. We are interested in bootstrapping the 95% CI around the mean. The outer resamples are set to 1500, the inner resamples are set to 20, 40, 80, 160, 320 and 640. For comparison, analytic and jackknife SEs are also provided. A total of 1000 simulations are conducted. For the evaluation of the results, the produced t-values for each simulation are saved so they can be compared later. As we can predict from normal theory, for a 95% CI, the target values should be approximately ±1.96. Since the target variable is normally distributed and the sample size is rather large, we know that this is the true value. To save space, the results are evaluated for the upper bound only, which is the percentile 97.5 of the generated t-distributions (since the outcome is normally distributed and hence symmetric, all differences to the lower bound must be due to Monte Carlo error). The results are visualized in Figure 6.

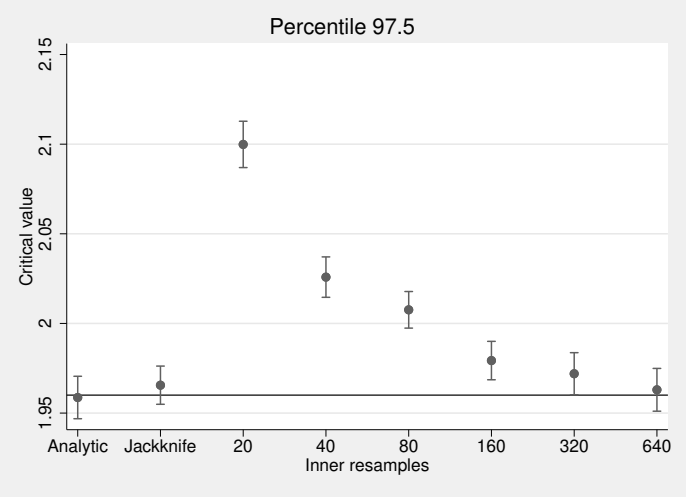

Figure 6: Critical values for percentile 97.5 of generated t-distributions when the number of inner resamples is varied. 1000 simulations, 95% confidence bands included to visualize the Monte Carlo error. Note the (partial) logarithmic scaling of the x-axis.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 13 February 2023 doi:10.20944/preprints202302.0213.v1

The analytic value serves as a benchmark; it is extremely close to the target value. The jackknife works also very well as the deviations to the target value are minimal. However, when the number of inner resamples is low, the deviations are large. Note that these deviations can result in massively incorrect CIs since the critical values are *multiplied* with the bootstrap standard errors. As the critical values are larger than the target value, this means that CIs will be too wide, resulting in overcoverage. As the number of resamples increases, the target value is approached. In this specific example, at least 320 inner resamples are required to reduce the error to an acceptable value while it is up to the researcher to find the compromise between error and computational burden. Note that this is a very "benign" example as the target variable is normally distributed and the sample size is rather large. In other scenarios even more inner resamples might be necessary. This also shows that using analytic standard errors is clearly preferable whenever they are available.

These results open up a final and quite relevant question: If no analytic standard errors are available, which method should be used? Apparently, the jackknife performs very well and gives results that are highly similar to the analytic approach. This frees the researcher from thinking about the number of inner resamples to take. Apparently, at least 300 are required for a similar precision. But what about the runtime? For the jackknife the runtime depends strongly on the size of the sample as the number of times the command has to be computed is equal to the sample size. For the inner bootstrap approach this is different as this aspect only depends on the number of inner resamples to take. How is computational time affected by this? To explore this, a simple simulation is conducted where the same task (CIs for the sample mean) is estimated using various approaches (jackknife, inner resampling with 100, 300 and 600 resamples) and sample sizes (ranging from 50 to 2000). The number of outer resamples is set to 80. 20 simulations are conducted under the usage of four vCPUs. The quadratic fit is visualised in Figure 7.

If one considers 300 inner resamples the lower limit for accurate results, we notice that the jackknife is faster up to a sample size of about 1600, afterwards the double resampling is quicker. This simple example highlights that researchers might want to investigate these questions in more detail for their own studies if time is critical. Note how the slopes are not linear in general. As we have seen above, the double bootstrap performs especially well in small samples ($n < 100$). In this region the jackknife is clearly preferable.

6 Discussion and conclusion

As researchers usually rely on samples as rarely entire populations can be studied, inference is a crucial aspect in applied statistics. Quantifying the uncertainty around point estimates is of greatest relevance, and confidence intervals are one popular and reliable option to achieve this. While bootstrapping in general is highly interesting due to its flexibility, the availability of various different techniques can be confusing. This article has outlined the logic of the bootstrap and how it can fail due its parametric

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 13 February 2023 doi:10.20944/preprints202302.0213.v1

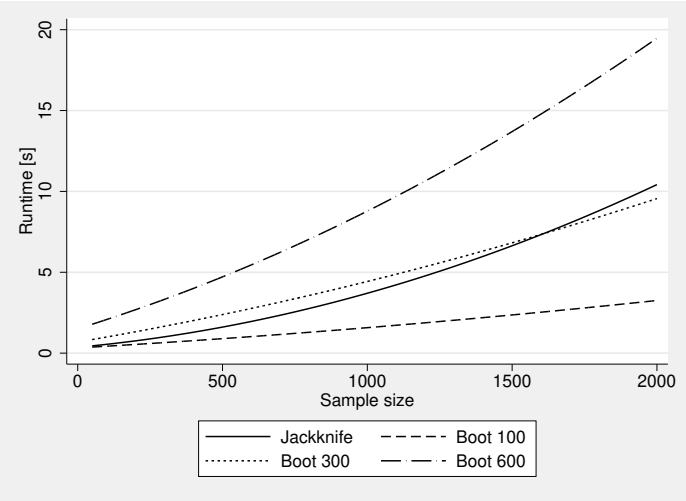

Figure 7: Aggregated times to compute the DBS CI for the sample mean, depending on the sample size. Compared are the jackknife approach and double resampling with 100, 300, and 600 inner resamples.

assumptions. The double bootstrap, in contrast, avoids these additional assumptions and only relies on the data. Simulation studies have tried to gauge the performance of various analytic and bootstrap approaches to estimate CIs in very common scenarios researchers face every day. Skewed and heteroscedastic samples are the reality as data is often not "benign". While researchers are usually aware of these obstacles, common remedies, such as robust standard errors, often only perform little better than the standard approach. Clearly, the strength of the double bootstrap is in small ($n < 100$) and heteroscedastic samples. It is the only approach that does not result in huge undercoverage in contrast to all other tested methods. It should be pointed out that undercoverage is a serious flaw as it means that it gets "easier" to reject the null hypothesis, leading to false-positive findings. Depending on the research question, this can have dire consequences. Since researchers can easily test for heteroscedasticity in their sample, they have the tools to choose the right approach for their data.

It should also be made clear that the double bootstrap is not necessarily better than other approaches. In the most benign scenarios where neither heteroscedasticity nor skewness is present, the normal OLS standard errors outperform all other methods in terms of computational speed, coverage and CI length. In samples where heteroscedasticity is not present the double bootstrap is not better than other bootstrap approaches. In larger samples with heteroscedasticity, the double bootstrap appears to lead to slight overcoverage. It is up to the researcher to decide whether over- or undercoverage results in the more problematic error to make. Furthermore, it should be highlighted that other approaches are available in Stata, such as the wild bootstrap (Roodman et al. 2019).

Finally, the limitations of the current study should be addressed. Clearly, the OLS

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 13 February 2023 doi:10.20944/preprints202302.0213.v1

regressions used in the simulation study are quite popular, yet these results should not be overgeneralized to all other methods. Since the number of potential outcomes is extremely large, no general statements should be made and researchers are advised to conduct similar simulation studies to gauge the quality of the different approaches for their methods. Potentially, research also might want to increase the number of bootstrap resamples to 10,000 or even more for a higher precision. Furthermore, users should keep in mind that using analytic standard errors for the inner resamples improves speed manifold but does come with more (parametric) assumptions as the analytic computation of the standard errors often has assumptions on the data. If the produced standard errors are highly off the generated t-values and cutoff values might be affected. If this is the case, users should rather rely on the bootstrap or jackknife option. Be aware of this potential point of failure.

7 Acknowledgments

Florian Scholze supported the project and conducted programming tests. Daniel Klein read an early version of the draft and gave valuable feedback. Linda Ruppert proofread the final text. Tim Hesterberg answered various technical questions. Thank you so much for your help!

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 13 February 2023 doi:10.20944/preprints202302.0213.v1

8 References

Bittmann, F. 2021. *Bootstrapping - An Integrated Approach with Python and Stata*. De Gruyter Oldenbourg.

Booth, J. G., and P. Hall. 1994. Monte Carlo approximation and the iterated bootstrap. *Biometrika* 81(2): 331–340.

Chernick, M. R. 2011. *Bootstrap methods: A guide for practitioners and researchers*. Vol. 619. John Wiley & Sons.

DiCiccio, T. J., and B. Efron. 1996. Bootstrap Confidence Intervals. *Statistical Science* 11: 189–228.

Efron, B., and R. J. Tibshirani. 1994. *An introduction to the bootstrap*. CRC press.

Hesterberg, T. C. 2015. What teachers should know about the bootstrap: Resampling in the undergraduate statistics curriculum. *The American Statistician* 69(4): 371–386.

King, G., and M. E. Roberts. 2015. How robust standard errors expose methodological problems they do not fix, and what to do about it. *Political Analysis* 23(2): 159–179.

Lee, S. M., and G. A. Young. 1999. The effect of Monte Carlo approximation on coverage error of double-bootstrap confidence intervals. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 61(2): 353–366.

Martin, M. A. 1990. On bootstrap iteration for coverage correction in confidence intervals. *Journal of the American Statistical Association* 85(412): 1105–1118.

Mbah, A. K., and A. Paothong. 2015. Shapiro–Francia test compared to other normality test using expected p-value. *Journal of Statistical Computation and Simulation* 85(15): 3002–3016.

McCullough, B., and H. Vinod. 1998. Implementing the double bootstrap. *Computational Economics* 12(1): 79–95.

Roodman, D., M. Ø. Nielsen, J. G. MacKinnon, and M. D. Webb. 2019. Fast and wild: Bootstrap inference in Stata using boottest. *The Stata Journal* 19(1): 4–60.

Vega Yon, G. G., and B. Quistorff. 2019. parallel: A command for parallel computing. *The Stata Journal* 19(3): 667–684.

Wasserstein, R. L., A. L. Schirm, and N. A. Lazar. 2019. Moving to a world beyond "p¡ 0.05". *The American Statistician* 73(sup1): 1–19.

About the author

Felix Bittmann is a research associate at the Leibniz Institute for Educational Trajectories and a doctoral candidate at the University of Bamberg, Germany. He is a sociologist with interests in social inequality, the role of education in the life course, quantitative methods and philosophy of science.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 13 February 2023 doi:10.20944/preprints202302.0213.v1

9 Appendix

Table 1: Simulation results (study 1)

Size	Hetero	Skew	Type	Coverage	SD(Coverage)	CI Width	SD (CI Width)
15	0	0	Regular	0.947	0.223	1.258	0.385
15	0	0	Robust	0.913	0.282	1.149	0.437
15	0	0	Normal	0.939	0.239	1.254	0.451
15	0	0	Percentile	0.925	0.264	1.258	0.447
15	0	0	BC	0.926	0.262	1.257	0.445
15	0	0	BCa	0.928	0.259	1.267	0.454
15	0	0	Double	0.948	0.222	1.561	0.803
30	0	0	Regular	0.953	0.212	0.791	0.158
30	0	0	Robust	0.932	0.252	0.757	0.196
30	0	0	Normal	0.935	0.247	0.764	0.187
30	0	0	Percentile	0.932	0.251	0.769	0.186
30	0	0	BC	0.932	0.251	0.768	0.185
30	0	0	BCa	0.93	0.256	0.771	0.186
30	0	0	Double	0.923	0.266	0.818	0.299
70	0	0	Regular	0.945	0.227	0.488	0.061
70	0	0	Robust	0.938	0.242	0.479	0.082
70	0	0	Normal	0.936	0.245	0.478	0.079
70	0	0	Percentile	0.936	0.244	0.479	0.079
70	0	0	BC	0.935	0.246	0.479	0.079
70	0	0	BCa	0.935	0.246	0.479	0.079
70	0	0	Double	0.93	0.255	0.488	0.125
200	0	0	Regular	0.946	0.226	0.281	0.02
200	0	0	Robust	0.943	0.232	0.279	0.027
200	0	0	Normal	0.942	0.234	0.279	0.027
200	0	0	Percentile	0.941	0.236	0.279	0.028
200	0	0	BC	0.941	0.236	0.279	0.028
200	0	0	BCa	0.941	0.235	0.279	0.028
200	0	0	Double	0.94	0.238	0.281	0.044
15	0	1	Regular	0.949	0.22	1.25	0.378
15	0	1	Robust	0.911	0.285	1.145	0.435
15	0	1	Normal	0.937	0.243	1.246	0.435
15	0	1	Percentile	0.927	0.26	1.25	0.435
15	0	1	BC	0.926	0.261	1.248	0.433
15	0	1	BCa	0.93	0.255	1.261	0.443
15	0	1	Double	0.945	0.228	1.563	0.804
30	0	1	Regular	0.956	0.206	0.79	0.153
30	0	1	Robust	0.932	0.251	0.753	0.194
30	0	1	Normal	0.939	0.24	0.762	0.185
30	0	1	Percentile	0.935	0.247	0.767	0.183
30	0	1	BC	0.934	0.249	0.766	0.184
30	0	1	BCa	0.935	0.247	0.769	0.186
30	0	1	Double	0.925	0.263	0.814	0.299
70	0	1	Regular	0.949	0.219	0.489	0.06
70	0	1	Robust	0.939	0.239	0.477	0.079
70	0	1	Normal	0.938	0.24	0.476	0.077
70	0	1	Percentile	0.937	0.243	0.478	0.077
70	0	1	BC	0.938	0.241	0.477	0.077
70	0	1	BCa	0.938	0.24	0.478	0.078
70	0	1	Double	0.924	0.265	0.485	0.121
200	0	1	Regular	0.95	0.218	0.28	0.02
200	0	1	Robust	0.947	0.224	0.278	0.028
200	0	1	Normal	0.946	0.226	0.278	0.028
200	0	1	Percentile	0.945	0.227	0.278	0.028
200	0	1	BC	0.946	0.226	0.278	0.028
200	0	1	BCa	0.947	0.224	0.278	0.028
200	0	1	Double	0.94	0.237	0.28	0.044

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 13 February 2023 doi:10.20944/preprints202302.0213.v1

Size	Hetero	Skew	Type	Coverage	SD(Coverage)	CI Width	SD (CI Width)
15	1	0	Regular	0.866	0.341	1.636	0.579
15	1	0	Robust	0.884	0.32	1.768	0.915
15	1	0	Normal	0.907	0.29	1.83	0.861
15	1	0	Percentile	0.906	0.292	1.785	0.776
15	1	0	BC	0.903	0.296	1.785	0.773
15	1	0	BCa	0.888	0.315	1.832	0.816
15	1	0	Double	0.941	0.236	3.108	2.918
30	1	0	Regular	0.833	0.373	1.071	0.269
30	1	0	Robust	0.901	0.299	1.327	0.589
30	1	0	Normal	0.902	0.297	1.297	0.543
30	1	0	Percentile	0.906	0.292	1.271	0.493
30	1	0	BC	0.902	0.297	1.276	0.497
30	1	0	BCa	0.875	0.331	1.311	0.535
30	1	0	Double	0.93	0.256	1.991	1.586
70	1	0	Regular	0.818	0.386	0.684	0.122
70	1	0	Robust	0.931	0.253	0.943	0.365
70	1	0	Normal	0.927	0.26	0.92	0.341
70	1	0	Percentile	0.921	0.27	0.91	0.317
70	1	0	BC	0.913	0.282	0.914	0.323
70	1	0	BCa	0.888	0.315	0.941	0.363
70	1	0	Double	0.965	0.184	1.404	1.033
200	1	0	Regular	0.805	0.396	0.398	0.042
200	1	0	Robust	0.944	0.23	0.583	0.157
200	1	0	Normal	0.942	0.235	0.575	0.151
200	1	0	Percentile	0.938	0.242	0.574	0.147
200	1	0	BC	0.93	0.256	0.576	0.151
200	1	0	BCa	0.91	0.287	0.587	0.173
200	1	0	Double	0.986	0.119	0.873	0.438
15	1	1	Regular	0.873	0.333	1.634	0.579
15	1	1	Robust	0.89	0.312	1.779	0.918
15	1	1	Normal	0.918	0.274	1.836	0.857
15	1	1	Percentile	0.909	0.287	1.788	0.775
15	1	1	BC	0.908	0.29	1.791	0.776
15	1	1	BCa	0.893	0.31	1.839	0.819
15	1	1	Double	0.942	0.233	3.147	2.788
30	1	1	Regular	0.85	0.357	1.069	0.266
30	1	1	Robust	0.916	0.278	1.319	0.577
30	1	1	Normal	0.918	0.275	1.29	0.532
30	1	1	Percentile	0.916	0.278	1.266	0.485
30	1	1	BC	0.913	0.282	1.27	0.488
30	1	1	BCa	0.885	0.319	1.305	0.526
30	1	1	Double	0.942	0.234	1.964	1.507
70	1	1	Regular	0.823	0.382	0.68	0.117
70	1	1	Robust	0.93	0.255	0.927	0.338
70	1	1	Normal	0.925	0.264	0.905	0.315
70	1	1	Percentile	0.923	0.266	0.896	0.297
70	1	1	BC	0.916	0.277	0.9	0.303
70	1	1	BCa	0.891	0.312	0.924	0.336
70	1	1	Double	0.958	0.2	1.352	0.899
200	1	1	Regular	0.8	0.4	0.398	0.043
200	1	1	Robust	0.94	0.238	0.584	0.162
200	1	1	Normal	0.937	0.242	0.576	0.156
200	1	1	Percentile	0.932	0.251	0.574	0.15
200	1	1	BC	0.923	0.266	0.577	0.156
200	1	1	BCa	0.904	0.295	0.589	0.179
200	1	1	Double	0.984	0.126	0.879	0.461

Note: 5000 simulations, 1500 bootstrap resamples.