Computational Methods in Econometrics: Assignment 2

The purpose of this assignment is to investigate the accuracy of the discussed bootstrap methods regarding their level. We will start with an example that we discussed during the lectures, the t-test under heteroskedasticity. Remember the setup as follows. We have a linear regression model

$$y = X\beta + \varepsilon, \qquad \mathbb{E}(\varepsilon) = 0, \qquad \mathbb{V}\mathrm{ar}(\varepsilon) = \Sigma,$$

where Σ is a diagonal matrix with possible different variance for every $1 \leq t \leq n$. To test H_0: $\mathring{\beta}_2 = 0$ versus H_1: $\mathring{\beta}_2 \neq 0$ with a t-statistic, we first need to obtain standard OLS estimates $\hat{\beta}_{OLS} = \mathring{\beta} + (X'X)^{-1}X'\varepsilon$ and residual $\hat{\varepsilon}_{OLS} = y - X\hat{\beta}_{OLS}$ and then use a heteroskedasticity-consistent covariance matrix estimator such as

$$\widehat{\mathrm{Var}}(\hat{\beta}_{OLS}) = (X'X)^{-1}X'\hat{\Sigma}X(X'X)^{-1}.$$

where $\hat{\Sigma}$ is a diagonal matrix with entry $\hat{\varepsilon}^2_{t,OLS}$ as its t'th diagonal element. Let $\hat{\beta}_{2,OLS}$ be the second element of $\hat{\beta}_{OLS}$ and let s_2^2 be the second diagonal element of $\widehat{\mathrm{Var}}(\hat{\beta}_{OLS})$, then the t-statistic

$$T_n(\vec{Y}) = \frac{\hat{\beta}_{2,OLS}}{\sqrt{s_2^2}}$$

can be shown to have a distribution that asymptotically converges to a standard normal. However, the estimator for the variance s_2^2 has been shown to be possibly seriously biased for smaller values of n.

Exercise 1.

a. Load the matrix X from the csv file called "Regressors.txt" and the matrix Y from the csv file called "Observables.txt". Each column of Y represents a single vector of observables y that has been generated using X with parameter $\mathring{\beta} = (1, 0, 1)'$ and heteroskedastic innovations.

b. Perform the theoretical t-test described above for each column of Y at level $\alpha = 0.05$ and report the average number of times that the test rejects. This should be far from the hoped for level 0.05.

c. To improve the test we explore bootstrap methods. Let $\hat{\beta}$ be the OLS estimator for β that satisfies the null hypothesis and define the residuals $\hat{\varepsilon} = y - X\hat{\beta}$. A residual based bootstrap method provides us with a simulated sequence $\varepsilon_1^*, \ldots, \varepsilon_n^*$ from which we calculate $y^* = X\hat{\beta} + \varepsilon^*$ and $t^* = T_n(\vec{y}^*)$. Perform for each column of Y each of the bootstrap methods that follow for $B = 99$ times and use the Monte-Carlo testing method to approximate the p-value and decide whether you reject or not. Then count the number of times you rejected over each column and divide by the total number of columns to get an approximation of the level of each method.

(a) Nonparametric residual bootstrap: Sample $\varepsilon_1^*, \ldots, \varepsilon_n^*$ from $E_n(\hat{\varepsilon}_1, \ldots, \hat{\varepsilon}_n)$, why don't we have to recenter? Hint: take a look at the matrix X.

(b) Wild bootstrap: Sample $\varepsilon_1^*, \ldots, \varepsilon_n^*$ as $\varepsilon_t^* = \hat{\varepsilon}_t v_t^*$, where $V_t^* \sim IID(0,1)$. Try both the standard normal distribution and the two-point distribution for V_t.

d. Do the same thing for the pairs bootstrap. Perform for each column of Y the pairs bootstrap for $B = 99$ times and use the Monte-Carlo testing method to approximate the p-value and decide whether you reject or not. Then count the number of times you rejected over each column and divide by the total number of columns to get an approximation of the level of each method.

e. Report and shortly discuss your results.

Next we analyze the performance of the bootstrap in a time series context. Suppose that we have an autoregressive model of order two:

$$y_t = \phi_1 y_{t-1} + \phi_2 y_{t-2} + \varepsilon_t, \qquad \varepsilon \sim \mathrm{IID}(0, \sigma^2).$$

Then the model is nonstationary if it contains a unit root $\mathring{\phi}_1 + \mathring{\phi}_2 = 1$. If this is the case, then estimators do not converge as we expect them to and thus standard analysis results become invalid. A standard test to check H_0: $\mathring{\phi}_1 + \mathring{\phi}_2 - 1 = 0$ versus H_1: $\mathring{\phi}_1 + \mathring{\phi}_2 - 1 < 0$ is the Dickey-Fuller test, where we rewrite the AR(2) as

$$\begin{aligned}\Delta y_t &= (\phi_1 + \phi_2 - 1) y_{t-1} + \phi_2 \Delta y_{t-1} + \varepsilon_t \\ &= \beta y_{t-1} + \phi_2 \Delta y_{t-1} + \varepsilon_t\end{aligned}$$

and test H_0: $\mathring{\beta} = 0$ versus H_1: $\mathring{\beta} \neq 0$ using the t-statistic

$$T_n(\vec{Y}) = \frac{\hat{\beta}_{OLS}}{\sqrt{\widehat{\mathrm{Var}}(\hat{\beta}_{OLS})}}.$$

The finite sample distribution of this test statistic is unknown and its limit distribution is not standard normal. Instead, this statistic converges to the so called Dickey-Fuller distribution.

Exercise 2.

a. Load the matrix Y_{het} from the csv file called "Timeseries_het.txt". Each column of Y_{het} represents a single sample path $y_1, \ldots, y_n$ that has been generated with parameters $(\mathring{\phi}_1, \mathring{\phi}_2) = (0.75, 0.25)'$ and *heteroskedastic* innovations.

b. Perform the theoretical Dickey-Fuller test described above for each column of Y at level $\alpha = 0.05$ and report the average number of times that the test rejects. Note that the innovations contain heteroskedasticity and thus that you should use the heteroskedasticity-consistent covariance matrix estimator. The theoretical rejection region is unknown but you can use the approximation $R_T = (-\infty, -1.95)$.

c. To improve the testing procedure we again explore bootstrap methods. Let $\hat{\phi}_2$ be the OLS estimator for ϕ_2 that satisfies the null hypothesis and define the residuals $\hat{\varepsilon}_t = \Delta y_t - \hat{\phi}_2 \Delta y_{t-1}$. A residual based bootstrap method provides us with a simulated sequence $\varepsilon_1^*, \ldots, \varepsilon_n^*$ from which we recursively calculate $y_t^* = (1-\hat{\phi}_2)y_{t-1}^* + \hat{\phi}_2 y_{t-2}^* + \varepsilon_t^*$ and $t^* = T_n(\vec{y}^*)$. Perform for each column of Y each of the bootstrap methods that follow for $B = 99$ times and use the Monte-Carlo testing method to approximate the p-value and decide whether you reject or not. Then count the number of times you rejected over each column and divide by the total number of columns to get an approximation of the level of each method.

   (a) Nonparametric residual bootstrap: Sample $\varepsilon_1^*, \ldots, \varepsilon_n^*$ from a recentered $E_n(\hat{\varepsilon}_1, \ldots, \hat{\varepsilon}_n)$. Why do we have to recenter here?

   (b) Wild bootstrap: Sample $\varepsilon_1^*, \ldots, \varepsilon_n^*$ as $\varepsilon_t^* = \hat{\varepsilon}_t v_t^*$, where $V_t^* \sim IID(0,1)$. Try both the standard normal distribution and the two-point distribution for V_t.

   (c) Block bootstrap: Sample $\varepsilon_1^*, \ldots, \varepsilon_n^*$ according to the block bootstrap. Experimenting with block length is encouraged, you can also just use block length $\ell = 20$.

   (d) Sieve bootstrap: Sample $\varepsilon_1^*, \ldots, \varepsilon_n^*$ using an autoregressive approximation according to the sieve bootstrap. Experimenting with number of lags is encouraged, you can also just use $p = 5$ lags.

d. Load the matrix Y_{dep} from the csv file called "Timeseries_dep.txt". Each column of Y_{dep} represents a single sample path $y_1, \ldots, y_n$ that has been generated with parameters $(\mathring{\phi}_1, \mathring{\phi}_2) = (0.75, 0.25)'$ and *dependent* innovations.

e. Redo steps b. and c. for the matrix Y_{dep}.

f. Report, compare and shortly discuss your results.

As a team, handing in the assignment will have to include a report and the written code. The report should contain:

- Pseudo code explaining what you have exactly done in your code.
- Explanations what could have gone wrong if unexpected results are obtained.

The code can be written in Matlab, R or Python and should contain :

- One file called "mainGROUPNUMBER" (for example if you are group 4 it is called "main4") that prints all the results for the complete assignment in the same order and in an easy to interpret manner. Take for example the output of the Matlab code on canvas called "Exponential".
- Comments in places where formulas are not instantly clear from a first read.

The deadline is Friday the 18th of October at 23:59.

HAVE FUN! :)