Code Documentation for Inference on Winners

Stata Package

Isaiah Andrews Toru Kitagawa Adam McCloskey

Code author: Noah Siderhurst (noahside@mit.edu)

February 2024

1 Overview

This Stata package applies the methodology from Andrews, Kitagawa, and McCloskey (2023) for feasible inference on winners. We wrote and ran our code using Stata 17 SE on a 6-core Intel i7-8700 processor with 16GB of RAM running on Windows 10 Enterprise.

2 Instructions to Run the Code

1. Open **`winners.do`** and change the **`home_dir`** macro to the folder where your vector of estimates (mu) and covariance matrix (sigma) are saved.

2. Update the macros under the section **`User Inputs`**:

   - **`ndraws`**: The number of draws used for confidence region computation. We do not recommend increasing the number of draws as it is near the maximum number of draws that Stata 17 SE can handle. However, if you are running Stata MP, you may be able to safely increase the number of draws. We recommend using the Matlab or R implementation of the code if you would like to increase the number of draws beyond the default of 10,000.
   - **`alpha`**: The significance level of the test. The default is 0.05.
   - **`beta`**: The level for the projection critical value used in the hybrid methods. The default is 0.005.
   - **`mu_format`**: The format of the vector of estimates (mu). If one variable is used for selection and is the outcome of interest (X=Y), change this local to "x". This is the case for the Jobstart example explored in the paper. If one variable is used for selection (X) and another is the outcome of interest (Y), leave this local as the default "x y".

- **K**: The number of treatment arms for your setting.
- **mu**: The name of the CSV containing the vector of estimates (X and Y). See **Section 3** for more information on the format of this file.
- **sigma**: The name of the CSV containing the covariance matrix of X and Y. See **Section 3** for more information on the format of this file.

3 Input Format

- **mu**: This should be a CSV with a 1 column by 2K row vector of estimates where rows 1 through K correspond to the variable used for selection (X) and rows K+1 through 2K correspond to estimates of the outcome of interest (Y). When X=Y, this vector will only have K rows; there is no need to list the estimates of X twice as the code will automatically set the vector Y equal to the vector of X in this case.
- **sigma**: This should be a CSV with the 2K by 2K covariance matrix of X and Y. When X=Y, this matrix will be K by K; the code will automatically expand the matrix to be 2K by 2K.

4 Ado Implementation

To make the use of our methods as easy as possible, we have also created an ado file that allows the user to run our procedure by typing the command - **winners "home_dir" ndraws alpha beta "mu_format" K "mu" "sigma"** - where the arguments correspond to the macros described in **Section 2.2**. The arguments must be typed in the order they are listed above. If not, the command will fail. Before running this command, make sure to copy **winners.ado** to Stata's ado directory.

5 Code Structure

1. **Preliminaries** cleans the input CSVs and saves all necessary matrices and vectors. The truncation values and critical values are also computed in this step.
2. **Median Unbiased Estimate** computes the median unbiased estimate using a bisection search algorithm. This involves two steps. First, the grid is expanded until the function crosses zero, then the search algorithm is applied to solve the function. The result is saved in the macro **median_unbiased_estimate**.
3. **Equal-Tailed CI Upper** uses the same bisection search algorithm (with a different function) to compute the upper bound of the equal-tailed confidence interval. The result is saved in the macro **equi_ci_upper**.
4. **Equal-Tailed CI Lower** uses the bisection search algorithm to compute the lower bound of the equal-tailed confidence interval. The result is saved in the macro **equi_ci_lower**.

5. **Hybrid Median Estimate** uses the bisection search algorithm to compute the median estimate in the hybrid setting. The result is saved in the macro **hyb_med_est**.

6. **Hybrid Equal-Tailed CI Upper** uses the bisection search algorithm to compute the upper bound of the equal-tailed confidence interval in the hybrid setting. The result is saved in the macro **hyb_eq_ci_upper**.

7. **Hybrid Equal-Tailed CI Lower** uses the bisection search algorithm to compute the lower bound of the equal-tailed confidence interval in the hybrid setting. The result is saved in the macro **hyb_eq_ci_lower**.

8. **Projection CI** computes the projection confidence interval using the critical value. The results are saved in the macros **proj_ci_upper** and **proj_ci_lower**.

9. **Print Output** displays the median unbiased estimate, the equal-tailed confidence interval, the median hybrid estimate, the equal-tailed hybrid confidence interval, and the projection confidence interval.

6 Folder Structure and Example

The folder **winners_stata** contains three files (not including this readme).

1. **winners.do** is the do-file used to implement our feasible inference procedure in Stata.

2. **mu.csv** contains an example vector of estimates corresponding to the Jobstart example explored in the paper. This example CSV only contains estimates of X, so if you would like to replicate the results from the paper, you will need to modify the **mu_format** macro (see **Section 1.3**).

3. **sigma_xy.csv** contains an example covariance matrix from the Jobstart example explored in the paper.