Generalized Lee Bounds

Guide to Code

1 Overview

The code for the paper "Generalized Lee Bounds" is contained in the zipped file "GenLeeBounds.zip". Unzipping this file provides this guide as well as several files and sub-directories. Specifically, the sub-directory "R" contains the files that construct basic generalized bound and sharp bound. The sub-directory "Raw_Data" is empty, it needs to be populated by the user with JobCorps Public Use Files from Schochet et al. (2008). The sub-directories, namely, "Logs", "Figures", "Tables" and "Derived_Data" are populated at run time. These files produce the tables and figures appearing in the paper and in the Online Appendix, as summarized on the next page.

2 JobCorps Public Use Files

We do not include original files from Schochet et al. (2008) because of GitHub size restrictions. In these notes and associated files, we provide computer code that allows a user to replicate our results given access to these data. Specifically, the R code assumes that the sub-directory Raw_Data contains the four files: *baseline.sas7bdat*, *empl_tl.sas7bdat*, *mileston.sas7bdat*, *key_vars.sas7bdat*. These files were downloaded from the AER website, replication code of Schochet et al. (2008). Covariate descriptions are given in Appendix L of Schochet et al. (2008).

Table 1: Overview of Files Creating Figures and Tables

Table/Figure	**File(s) Creating It**	**File(s) Being Created**
Figures 1 and 2	STEP2_TEST_MONOTONICITY/*.R	Figures/Figure1.png, Figure2.png
Tables 1 and 2	STEP3_ESTIMATE_BOUNDS/Table*.R	Tables/STEP3_Bounds/Vira/csv/Table*.csv
Figure 3	STEP4_FIGURE3/Figure3.R, print_Figure3.R	Figures/Figure3.png

3 Overview of R Files

Table 2: R Files in R folder

R Function	Input	Output	Description
auxiliary.R			
prepare_leedata	raw Lee data (panel for all weeks), week number	data vector `leedata` for given week	logs of hourly earnings, missing values by zero, attaches weights, assembles `leedata` $W = (D,X,S,SY,\text{weight},\Pr(D=1 \mid X),\Pr(D=0 \mid X))$
main_bb	data, name of function, number of repetitions	bootstrapped estimates	
compute_confidence_region	main estimate, bootstrapped draws, and significance level α	$CR_{1-\alpha}$ in (5.13)	estimates cov matrix from bootstrapped estimates, takes cholesky sqrtm, returns CR
leebounds.R			
basic_lee_bound	`leedata`	β_U^{basic} in (2.4)	basic Lee bound under (Assumption 1)
discrete_bound	`leedata`, indicator of treatment helps	bound in (2.8)	discrete Lee bound of Algorithm 1
first_stage_functions.R			
estimate_selection	`leedata`, formula `selection~ treat*(covnames)`, method (glm or rlassologit),	`glm.fit` object (coefs γ^d in (5.1))	`glm.fit` object
predict_selection	`glm.fit` object, `leedata`	$s(0,x)$ and $s(1,x)$ in (3.1)	s.hat=fitted values for $s(0,x)$ and $s(1,x)$
estimate_quantile	`leedata`, formula `outcome~ treat*(covnames)`, quantreg:rq method (default or lasso), quantile grid size	$Q^1(u,x)$ and $Q^0(u,x)$ on Page 6 for $u \in (0.1,0.2,\dots,0.9)$ if grid=0.1	data frame of fitted values for $Q^1(u,x)$ and $Q^0(u,x)$
impute_quantile_level	$p_0(x)$ and quantile estimates	$Q(p_0(x),x)$	plugs $p_0(x)$ into $Q^1(u,x)$ to get $Q(p_0(x),x)$
estimate_borderline_wage	$p_0(x)$ and quantile estimates	data frame `y.hat=(y.p0.hat,y.1.p0.hat)`	sorted quantile estimates + trimmed at min/max wage encoding $Q(p_0(x),x)$, $Q(1-p_0(x),x)$
first_stage_wrapper	`leedata`	`leedata + s.hat+ y.hat`: fitted values for Algorithm 2	

Table 3: R Files in R folder

second_stage_functions.R			
trimmed_moments	`leedata`	$m_U(W,\xi)$ in (3.15)	trimmed moments
orthogonal_correction	`leedata` (including s.hat and y.hat), indicator of treatment helps	N-vector of $\mathrm{cor}_U^{\mathrm{hurt}}(W,\xi)$	bias correction terms: $\mathrm{cor}_U^{\mathrm{hurt}}(W,\xi)$, $\mathrm{cor}_U^{\mathrm{help}}(W,\xi)$, $\mathrm{cor}_L^{\mathrm{hurt}}(W,\xi)$, $\mathrm{cor}_L^{\mathrm{help}}(W,\xi)$ in (4.9)-(4.11)
numerator_2ndstage	`leedata` (including s.hat and y.hat)	N-vector of $g_U^{\mathrm{help}}(W,\xi)$ and $g_U^{\mathrm{hurt}}(W,\xi)$ in (4.7)	$g_U^{\mathrm{help}}(W,\xi)$ and $g_U^{\mathrm{hurt}}(W,\xi)$ in (4.7) if "ortho == True" and $m_U^{\mathrm{help}}(W,\xi)$ and $m_U^{\mathrm{hurt}}(W,\xi)$ in (2.10) if option "ortho == False
always_takers_share	`leedata` (including s.hat)	$\pi_{\mathrm{AT}} = \mathbb{E}\min(s(0,X),s(1,X))$	uses (3.9) if option "ortho_d == False"; $g_D(W,s)$ in (4.16) if option "ortho_d == True"
second_stage_wrapper	`leedata` (including s.hat and y.hat)	output of Algorithm 1	estimates $g_U^{\mathrm{help}}(W,\xi)$ and $g_U^{\mathrm{hurt}}(W,\xi)$, combines into $g_U(W,\xi)$ as in (5.8), divides by π_{AT}, returns
ortho_leebounds	`leedata`	output of Algorithm 1	(1) first-stage-wrapper (2) second-stage-wrapper
basic_generalized_bound	`leedata`	$\bar{\beta}_U$ in (6.8)	basic generalized bound (Assumption 2(b))

Table 4: Helper files for Steps 2, 3 and 4

R Function	**Input**	**Output**	**Description**
/STEP2_TEST_MONOTONICITY/utils_for_test.R			
compute_tstat_by_group	`leedata`	t statistics by groups	t statistics
test_wrapper	week, group	maximum of t statistics	t statistics

Table 5: Overview of Code (Step 1 and 2)

Filename	**Input Files**	**Output File**	**Description**
		Step 1: Prepare Data	
prepare_basic_data.R	/Raw_Data/ key_vars.sas7bdat, mileston.sas7bdat, baseline.sas7bdat, empl_tl.sas7bdat	/Derived_Data/dataLee2009.csv	merge input files by MPRID, recodes some covariates to match Lee(2009) and saves the result into data/dataLee2009.csv
prepare_covariates.R	baseline.sas7bdat, /Derived_Data/dataLee2009.csv	/Derived_Data/dataLee2009covariates .feather	identifies all categorical variables and recodes them as binary indicators. Output is not loaded at GitHub due to memory constraint.
prepare_covariates_short.R	baseline.sas7bdat, /Derived_Data/dataLee2009.csv	/Derived_Data/dataLee2009covariates _short.feather	Drops welfare, school, training history. Output is not loaded at GitHub due to memory constraint.

Table 6: Overview of Code (Step 2)

Filename	**Input Files**	**Output File**	**Description**
		Step 2: Test Monotonicity	
Figures12.R	/Derived_Data/dataLee2009.csv, dataLee2009covariates.feather	/Tables/STEP2_Monotonicity/Vira/ Figure1.csv, Figure2.csv, Frechet-Bounds.csv,	data for Figures 1 and 2
test_monotonicity.R	/Derived_Data/dataLee2009.csv, dataLee2009covariates.feather	/Tables/STEP2_Monotonicity/Vira/ test_result.csv	1. Conduct a monotonicity test for each week using a week-specific test statistic (7.4) and p-value; 2. See results in Table C.10
print_Figures12.R	/Tables/STEP2_Monotonicity/ Vira/Figure1.csv, Figure2.csv	/Figures/Figure1ṗng, Figure2ṗng	Print Figure 1,2

Table 7: Overview of Code (Step 3)

Filename	Input Files	Output File	Description
		Step 3: Estimate Bounds	
Table1.R	dataLee2009.csv	csv files under Tables/STEP3_Bounds/	(1) estimate always-takers share, basic Lee bounds (2.4) and discrete Lee bounds (2.8) and 95%-CR under Assumption 1
Tab_Lee.R	dataLee2009.csv, dataLee2009covariates.feather	csv files under Tables/STEP3_Bounds/	estimates generalized bound (3.18) using Algorithm 2 with first-stage fitted values based on penalized estimates for logistic and quantile regression under Assumption 2(b)
Tab_Full.R	dataLee2009.csv, dataLee2009covariates_short.feather	csv files under Tables/STEP3_Bounds/	estimates generalized bound (3.18) using Algorithm 2 with first-stage fitted values based on penalized estimates for logistic and quantile regression under Assumption 2(b)
Tab_Lasso.R	dataLee2009.csv, dataLee2009covariates.feather	csv files under Tables/STEP3_Bounds/	estimates generalized bound (3.18) using Algorithm 2 with first-stage fitted values based on estimates for logistic and quantile regression under Assumption 2(b) using Lasso-selected covariates
Table2.R			assembles Table2.txt based on Tab_*.R above

References

Schochet, P. Z., Burghardt, J., and McConnell, S. (2008). Does job corps work? impact findings from the national job corps study. *American Economic Review*, 98(1):1864–1886.