Survey on Practices and Challenges for Reproducibility of Empirical Research

Field	Question	Answer
Introduction		
note1	This survey aims to assess and learn from the practices commonly used by Principal Investigators in the top development economics research institutions to advance in the process of improving the reproducibility of their work, potential constraints that they may face, and the tools used to make this process easier. If needed, please only delegate the response of the survey to the Principal Investigator or Research Manager more knowledgeable about the data analysis process in your projects in the last two years. We thank you for not forwarding this link to other respondents, as we are tracking response rates over our target sample.	
note2	Please answer all the questions to the best of your knowledge. Approximate answers are good.	
respondent *(required)*	Respondent *Select one.*	1 I am the targeted respondent of the survey (received the original email) 2 I am answering this survey by delegation
education *(required)*	Highest education completed *Question relevant when: ${respondent} =2*	1 Undergraduate 2 Master's 3 Ph.D.
background		
years *(required)*	For how many years have you been conducting empirical work, simulations or experimental work?	0 I do not do this type of work 1 1 2 2 3 3 4 4 5 5 6 6 7 7 8 8 9 9 10 10 11 11 12 12 13 13 14 14 15 15 16 16 17 17 18 18 19 19 20 20 21 21 22 22 23 23 24 24 25 25 26 26 27 27 28 28 29 29 30 30 31 More than 30
coauthors *(required)*	How many co-authors, on average, are in your currently active projects involving data analysis? *Please approximate and select the option closest to the mode.* *Question relevant when: not(selected(${years} ,0))*	0 0 1 1 2 2 3 3 4 4 5 5 6 6 7 More than 6

ras_all *(required)*	Approximately how many research assistants (RAs) are currently working in all of your projects involving data analysis? *Question relevant when: not(selected(${years} ,0))*	0 0 1 1 2 2 3 3 4 4 5 5 6 6 7 7 8 8 9 9 10 10 11 11-20 12 More than 20
ras_pp *(required)*	What is the average number of RAs that conduct data analysis PER PROJECT? *Please approximate and select the option closest to the mode.* *Question relevant when: not(selected(${years} ,0))*	0 0 1 1 2 2 3 3 4 4 5 5 6 6 7 More than 6
projects *(required)*	How many of your projects have been in the data analysis and/or journal submission phase in the last 2 years? *Question relevant when: not(selected(${years} ,0))*	0 0 1 1 2 2 3 3 4 4 5 5 6 6 7 7 8 8 9 9 10 10 11 More than 10
prepared *(required)*	How prepared do you think you and your teams are in complying with the new policies on data and code availability such as the ones implemented last July by the AEA to improve reproducibility and transparency? *AEA's Data and Code Availability Policy (<u><font color="blue">https://www.aeaweb.org/journals/policies/data-code</u>) The spirit of the new policy is to verify that the process of empirical work, simulations or experimental work–from raw data to reported results–is replicable, which requires to write and document code that is replicable and present it at submission. Select between 1 and 5, with 1 being very unprepared and 5 very prepared.* *Question relevant when: not(selected(${years} ,0))*	1 1 2 2 3 3 4 4 5 5

current

Group relevant when: not(selected(${years} ,0)) and not(selected(${projects} ,0))

note3	Please answer the following questions considering your projects that have been in the data analysis/submission phase in the last two years.	
involvement *(required)*	How involved are you in the data coding process?	1 I direct the RAs/Research manager but they write all the code. I never review the codes directly. 2 I direct the RAs/Research manager but they write all the code. I sometimes review the codes. 3 I direct the RAs/Research manager but they write all the code. I often review the codes. 4 I do most of the coding. 0 Other
involvement_other *(required)*	If other, please specify. *Question relevant when: ${involvement} =0*	
tasks *(required)*	How do you/your team commonly manage the tasks for data analysis and coding? *This is the means used, for instance, to record how the task is defined, whose responsibility is the task, the timeline for the task, or what are the final outputs of the task. Mark all that apply.*	1 Email 2 Google docs 3 Asana 4 Slack

		0 Other
tasks_other *(required)*	If other, please specify. *Question relevant when: selected(${tasks} ,0)*	
current > current1 *Group relevant when: ${tasks_sum} =1*		
note4	What percentage of the tasks is managed through each channel/system? *Please approximate the percentages based on your best recollection. For example, Email (50%); Asana (50%).*	
tasks_p_email *(required)*	Email *Question relevant when: selected(${tasks} ,1)*	0 0% 5 5% 10 10% 15 15% 20 20% 25 25% 30 30% 35 35% 40 40% 45 45% 50 50% 55 55% 60 60% 65 65% 70 70% 75 75% 80 80% 85 85% 90 90% 95 95% 100 100%
tasks_p_google *(required)*	Google docs *Question relevant when: selected(${tasks} ,2)*	0 0% 5 5% 10 10% 15 15% 20 20% 25 25% 30 30% 35 35% 40 40% 45 45% 50 50% 55 55% 60 60% 65 65% 70 70% 75 75% 80 80% 85 85% 90 90% 95 95% 100 100%
tasks_p_asana *(required)*	Asana *Question relevant when: selected(${tasks} ,3)*	0 0% 5 5% 10 10% 15 15% 20 20% 25 25% 30 30% 35 35% 40 40% 45 45% 50 50% 55 55%

		60	60%
		65	65%
		70	70%
		75	75%
		80	80%
		85	85%
		90	90%
		95	95%
		100	100%
tasks_p_slack *(required)*	Slack *Question relevant when: selected(${tasks} ,4)*	0	0%
		5	5%
		10	10%
		15	15%
		20	20%
		25	25%
		30	30%
		35	35%
		40	40%
		45	45%
		50	50%
		55	55%
		60	60%
		65	65%
		70	70%
		75	75%
		80	80%
		85	85%
		90	90%
		95	95%
		100	100%
tasks_p_other *(required)*	[tasks_other] *Question relevant when: selected(${tasks} ,0)*	0	0%
		5	5%
		10	10%
		15	15%
		20	20%
		25	25%
		30	30%
		35	35%
		40	40%
		45	45%
		50	50%
		55	55%
		60	60%
		65	65%
		70	70%
		75	75%
		80	80%
		85	85%
		90	90%
		95	95%
		100	100%
tasks_v *(required)*	You responses do not add up to 100%. Are your previous answers correct? Please go back to change the answers if necessary. *Question relevant when: ${tasks_c} !=100 and ${tasks_sum} =1* *Response constrained to: .=1*	1	Yes
		0	No
current > current2			
versions *(required)*	What strategies/tools do you use to track different versions of your data, code or output files? *Mark all that apply.*	1	Version identifier names (e.g., “_v1”, “_v2”)
		2	Dates and/or initials to identify creation or

			last modification details (e.g., clean_17Aug2019_GB.do)
		3	Version-control software (e.g., Github, BitBucket)
		0	Other
note5	**Version-control software:** Systems designed by software engineers to manage changes to files (e.g., documents or programs) in a logical and user-friendly manner, facilitating collaborative work, with the ability to easily recall previous versions later and track changes by every contributor, helping prevent conflicts. Examples of version-control software include Git/Github and Subversion.		
versions_other *(required)*	If other, please specify. *Question relevant when: selected(${versions} ,0)*		
current > current3 *Group relevant when: ${versions_sum} =1*			
note6	What percentage of the tracking is used through each system? *Please approximate the percentages based on your best recollection. For example, Version identifiers (50%); Dates (50%).*		
versions_p_id	Version identifier (e.g., “_v1”, “_v2”) *Question relevant when: selected(${versions} ,1)*	0	0%
		5	5%
		10	10%
		15	15%
		20	20%
		25	25%
		30	30%
		35	35%
		40	40%
		45	45%
		50	50%
		55	55%
		60	60%
		65	65%
		70	70%
		75	75%
		80	80%
		85	85%
		90	90%
		95	95%
		100	100%
versions_p_date	Dates and/or initials to identify creation or last modification details (e.g., clean_17Aug2019_GB.do) *Question relevant when: selected(${versions} ,2)*	0	0%
		5	5%
		10	10%
		15	15%
		20	20%
		25	25%
		30	30%
		35	35%
		40	40%
		45	45%
		50	50%
		55	55%
		60	60%
		65	65%
		70	70%
		75	75%
		80	80%
		85	85%
		90	90%
		95	95%
		100	100%
versions_p_software	Version-control software (e.g., Github, BitBucket) *Question relevant when: selected(${versions} ,3)*	0	0%
		5	5%
		10	10%
		15	15%

			20	20%
			25	25%
			30	30%
			35	35%
			40	40%
			45	45%
			50	50%
			55	55%
			60	60%
			65	65%
			70	70%
			75	75%
			80	80%
			85	85%
			90	90%
			95	95%
			100	100%
versions_p_other	[versions_other] *Question relevant when: selected(${versions} ,0)*		0	0%
			5	5%
			10	10%
			15	15%
			20	20%
			25	25%
			30	30%
			35	35%
			40	40%
			45	45%
			50	50%
			55	55%
			60	60%
			65	65%
			70	70%
			75	75%
			80	80%
			85	85%
			90	90%
			95	95%
			100	100%
versions_v *(required)*	You responses do not add up to 100%. Are your previous answers correct? Please go back to change the answers if necessary. *Question relevant when: ${versions_c} !=100 and and /data_policy_wb_dime/current/versions_sum =1* *Response constrained to: .=1*		1	Yes
			0	No
softwares *(required)*	What type of version-control software do you use? *Question relevant when: selected(${versions} ,3)*		1	Git/Github
			2	BitBucket
			3	SVK/Subversion
			0	Other
softwares_other *(required)*	If other, please specify. *Question relevant when: selected(${softwares} ,0)*			
directories *(required)*	How is the BASIC structure of the directories defined or modified?		1	Varies by project
			2	Fixed template for all projects
			0	Other
directories_other *(required)*	If other, please specify. *Question relevant when: selected(${directories} ,0)*			
directories_create *(required)*	How is the fixed structure of the directories created? *Question relevant when: selected(${directories} ,2)*		1	Manually
			2	Copied from a template
			3	Automatically created through a code
			0	Other

directories_create_other *(required)*	If other, please specify. *Question relevant when: selected(${directories_create} ,0)*		
protocols *(required)*	How are protocols and documentation for codes defined?	1	Varies by project
		2	Fixed template for all projects
		0	Other
protocols_other *(required)*	If other, please specify. *Question relevant when: selected(${protocols} ,0)*		
abstraction	Is there a process within your team for improving code and coding practices of RAs through better use of abstraction? *Abstraction is turning the specific instances of something into a general-purpose tool (Gentzkow and Shapiro, 2014). It helps efficiency and clarity reducing redundancy and errors. Examples include developing user-written commands, programs or loops for repetitive operations, or automating the setup for any team member to open codes and run them, as well as produce outputs (such as using Latex).* *Response constrained to: (not(selected(.,0) and count-selected(.)>=2)) and (not(selected(.,4) and count-selected(.)>=2))*	1	Yes, through internal training/review
		2	Yes, through external training/review
		3	Peer review/collaboration encouraged but no systematic process in place
		0	No
		4	Do not know
code_review *(required)*	Have you performed code review internally within your team (for your last 2 projects)?	0	No
		1	Yes
		2	It varied by team
		3	Do not know
code_review_phase *(required)*	When in the process was the internal code review performed (for your last 2 projects)? *Mark all that apply.* *Question relevant when: selected(${code_review} ,1) or selected(${code_review} ,2)*	1	Before working paper publication
		2	Before journal submission
		3	At specific milestones
		4	Scheduled at fixed intervals during data analysis
		5	As per PI/Research manager request
		0	Other
code_review_phase_other *(required)*	If other, please specify. *Question relevant when: selected(${code_review_phase} ,0)*		
code_review_ext *(required)*	Have you hired or accessed independent code reviewers in your organization or externally (for your last 2 projects)?	0	No
		1	Yes
		2	It varied by team
		3	Do not know
code_review_ext_phase *(required)*	When in the process was the independent code review performed (for your last 2 projects)? *Mark all that apply.* *Question relevant when: selected(${code_review_ext} ,1) or selected(${code_review_ext} ,2)*	1	Before working paper publication
		2	Before journal submission
		3	At specific milestones
		4	Scheduled at fixed intervals during data analysis
		5	As per PI/Research manager request
		0	Other
code_review_ext_phase_other *(required)*	If other, please specify. *Question relevant when: selected(${code_review_ext_phase} ,0)*		
training *Group relevant when: not(selected(${years} ,0))*			
training > training1			
trainings *(required)*	Have you ever taken a training on (or self-taught using guides or other external resources) any of the following software or code management tools? *Mark all that apply.* *Response constrained to: (not(selected(.,0) and count-selected(.)>=2))*	1	Git/GitHub or any other version-control software
		2	Pre-established folder structure and master code files (including default settings, prerequisites for programs, macros setup and standardization)
		3	Code automation and abstraction (e.g., automatically generated and formatted tables in Latex, general-purpose functions for code simplification)
		4	Coding practices, including protocols for cleaning data, constructing variables, naming variables, and coding style (such as <u><font color="blue">https://style.tidyverse.org</u>)
		0	None of the above
note7	**Version-control software:** Systems designed by software engineers to manage changes to files (e.g., documents or programs) in a logical and user-friendly manner, facilitating collaborative work, with the		

	ability to easily recall previous versions later and track changes by every contributor, helping prevent conflicts. Examples of version-control software include Git/Github and Subversion. **Code automation and abstraction:** Automation is key for writing good code. It allows a higher abstraction level–turning specific instances of something into a general-purpose tool. It helps efficiency and clarity reducing redundancy and errors. Examples include developing user-written commands, programs or loops for repetitive operations, or automating the setup for any team member to open codes and run them, as well as produce outputs (such as using Latex). **Pre-established folder structure and master program files:** A folder structure can help organize data and code so that it will not cause problems as it evolves. For example, DIME's automated folder command iefolder in the package ietoolkit available through SSC sets up a pre-established folder structure based on best practices. A good practice also includes having a master script (master do-file or Rscript for example) which runs all other files or scripts and also serves as a map to navigate the data folder.	
training > training2		
trainings_ra *(required)*	To the best of your knowledge, have any of your RAs/research managers ever taken a training on (or self-taught using guides or other external resources) any of the following software or code management tools? *Mark all that apply.* *Response constrained to: (not(selected(.,0) and count-selected(.)>=2)) and (not(selected(.,5) and count-selected(.)>=2))*	1 Git/GitHub or any other version-control software 2 Pre-established folder structure and master code files (including default settings, prerequisites for programs, macros setup and standardization) 3 Code automation and abstraction (e.g., automatically generated and formatted tables in Latex, general-purpose functions for code simplification) 4 Coding practices, including protocols for cleaning data, constructing variables, naming variables, and coding style (such as <u><font color="blue">https://style.tidyverse.org</u>) 0 None of the above 5 Do not know
note8	**Version-control software:** Systems designed by software engineers to manage changes to files (e.g., documents or programs) in a logical and user-friendly manner, facilitating collaborative work, with the ability to easily recall previous versions later. Examples of version-control software include Git/Github and Subversion. **Code automation and abstraction:** Automation is key for writing good code. It allows a higher abstraction level–turning specific instances of something into a general-purpose tool. It helps efficiency and clarity reducing redundancy and errors (Gentzkow and Shapiro, 2014). Examples include developing user-written commands, programs or loops for repetitive operations, or making automatic the setup for any team member to open codes and run them, as well as produce outputs (such as using Latex). **Pre-established folder structure and master program files:** A folder structure can help organize data and code so that it will not cause problems as it evolves. For example, DIME's automated folder command iefolder in the package ietoolkit available through SSC sets up a pre-established folder structure based on best practices. A good practice also includes having a master script (master do-file or Rscript for example) which runs all other files or scripts and also serves as a map to navigate the data folder.	
training > training3 *Group relevant when: selected(${trainings_ra} ,1) or selected(${trainings_ra} ,2) or selected(${trainings_ra} ,3) or selected(${trainings_ra} ,4)*		
note9	Approximately what proportion of your RAs/research managers have ever taken a training on (or self-taught using guides or other external resources) any of the following software or code management tools? Select between 1 and 5, with 1 being very few and 5 all team members.	
trainings_git_ra	Git/GitHub or any other version-control software *Question relevant when: selected(${trainings_ra} ,1)*	1 1 2 2 3 3 4 4 5 5 0 Do not know
trainings_structure_ra	Pre-established folder structure and master code files (including default settings, prerequisites for programs, macros setup and standardization)	1 1

	Question relevant when: selected(${trainings_ra} ,2)	2	2
		3	3
		4	4
		5	5
		0	Do not know
trainings_auto_ra	Code automation and abstraction (e.g., automatically generated and formatted tables in Latex, general-purpose functions for code simplification) *Question relevant when: selected(${trainings_ra} ,3)*	1	1
		2	2
		3	3
		4	4
		5	5
		0	Do not know
trainings_code_ra	Coding practices, including protocols for cleaning data, constructing variables, naming variables, and coding style *Question relevant when: selected(${trainings_ra} ,4)*	1	1
		2	2
		3	3
		4	4
		5	5
		0	Do not know
training > training4			
note10	If you wanted to, how easily could you find trainings or resources within your organization on each of the following software or code management tools? Select between 1 and 5, with 1 being very difficult and 5 very easy.		
trainings_git *(required)*	Git/GitHub or any other version-control software	1	1
		2	2
		3	3
		4	4
		5	5
		0	Do not know
trainings_structure *(required)*	Pre-established folder structure and master code files (including default settings, prerequisites for programs, macros setup and standardization)	1	1
		2	2
		3	3
		4	4
		5	5
		0	Do not know
trainings_auto *(required)*	Code automation and abstraction (e.g., automatically generated and formatted tables in Latex, general-purpose functions for code simplification)	1	1
		2	2
		3	3
		4	4
		5	5
		0	Do not know
trainings_code *(required)*	Coding practices, including protocols for cleaning data, constructing variables, naming variables, and coding style	1	1
		2	2
		3	3
		4	4
		5	5
		0	Do not know
tranings_more *(required)*	Do you think your projects would benefit from more/better use of the following software/tools? *Response constrained to: (not(selected(.,0) and count-selected(.)>=2))*	1	Git/GitHub or any other version-control software
		2	Pre-established folder structure and master code files (including default settings, prerequisites for programs, macros setup and standardization)
		3	Code automation and abstraction (e.g., automatically generated and formatted tables in Latex, general-purpose functions for code simplification)
		4	Coding practices, including protocols for cleaning data, constructing variables, naming variables, and coding style (such as <u><font color="blue">https://style.tidyverse.org</font

				color="blue"></u>)
			5	Internal code review
			6	External code review
			0	None of the above
constraints *(required)*	What do you think are the main constraints to increasing the take-up and usage of these tools within your team? *Question relevant when: not(selected(${years} ,0))*		1	Lack of time to learn how to use these tools/practices
			2	Difficult to know the best options for my projects and teams
			3	Using these tools/practices will make it difficult for me to review the code
			4	Difficult to access training for me or other PIs
			5	Difficult to access training for my RAs
			6	The entry cost of switching to these tools/practices is too high
			7	The benefit of using these tools/practices does not compensates the cost
			0	Other
constraints_other *(required)*	If other, please specify. *Question relevant when: selected(${constraints} ,0)*			

final		
comments	COMMENTS *Please enter any additional comments.* *Question relevant when: not(selected(${years} ,0))*	
note11	We have reached the end of the survey. Thank you very much for your time and contribution! *Question relevant when: not(selected(${years} ,0))*	
note12	We have reached the end of the survey. This survey assesses new data policies for empirical work, simulations, or experimental work only. Thank you very much for your time and contribution! *Question relevant when: selected(${years} ,0)*	