

Guidelines for Computational Reproducibility in Economics

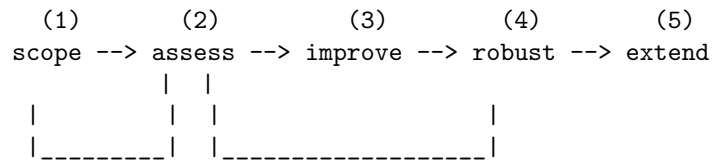
ACRE Team

2020-02-14

Contents

	5
1 Introduction	7
1.1 Stages of the exercise	8
1.2 Recording the results of the exercise	9
2 Scoping	11
2.1 Steps	11
2.2 Sample strategies for scoping	14
3 Assessment	15
3.1 Describe inputs	15
3.2 Connect each output to all its inputs	18
3.3 Assign a reproducibility score	19
4 Improvements	25
4.1 Types of output-level improvements	25
4.2 Types of paper-level improvements	29
5 Checking for Robustness	31
5.1 Identifying Analytical Choices	31
5.2 Choose and justify alternative values for analytical choices	33
5.3 Test the robustness of results	33
6 Concluding the reproduction	35

7 Tips for Communication	37
7.1 For students or researchers conducting a reproduction	37
7.2 For original authors contacted by reproducers	38
7.3 Responding to inquiries	38
8 Code of Conduct	39
9 Contributions	41
9.1 Ask for feedback on guidelines	41
9.2 List of Contributors	41
10 Reproduction Diagrams	43
10.1 Different Scenarios	43
10.2 Additional resources	44
11 Additional resources	47
11.1 Some summaries	47
11.2 Links	47
[ADD ONE PAGER BETWEEN COVER AND INTRO]	



Scoping	Assessment	Improvement	Robustness	Extension
Select paper	Describe inputs	Output-level imp:		
Check ACRE	Reproduction diagrams	+ Raw data		
Search for rep. mat.	Reproduction score			
Read and summ.				
Declare scope:				
Output(s)				
Depth				

Chapter 1

Introduction

Since in 2019, the American Economic Association began requiring all the materials for reproduction of any new publication. In addition to the requirements, several specific recommendations were produced to facilitate the authors' compliance. This change in policy and guidance should radically improve the computational reproducibility of all new published research. However all the published research before 2019 did not have neither the requirements or the guidance to develop proper reproduction materials.

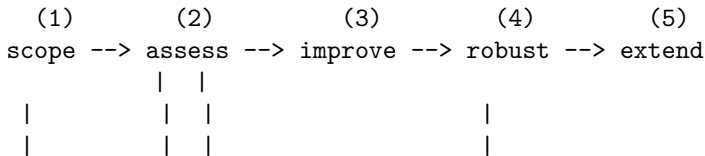
As a result, several studies point towards rates of computational reproducibility in economics that range from alarmingly low (ADD Gertler, Chang Li) to just low (ADD Kingi et al.). The project of Advancing Computational Reproducibility in Economics (ACRE) has the goals of improving, and having a much clearer picture of, computational reproducibility in economics.

Replication, or the process by which a study's hypotheses and findings are re-examined using different data or different methods (or both) (King (1995)) is an essential part of the scientific process that allows science to be "self-correcting." Computational reproducibility, or the ability to reproduce the results, tables, and other figures using the available data, code, and materials, is a precondition for replication. Computational reproducibility is assessed through the process of reproduction. At the center of this process is the reproducer (you!), a party not involved in the production of the original paper. Reproductions sometimes involve the original author (whom we refer to as "the author") in cases where additional guidance and materials are needed to execute the process.

This exercise is designed for reproductions performed in Economics graduate courses or undergraduate theses, with the goal of providing common terminology and standards for conducting reproductions. The goal of reproduction, in general, is to assess and improve the computational reproducibility of published research in a way that facilitates future replication, extension, and collaboration.

This reproduction exercise is divided into four to five stages:

1. **Scoping**, where the reproducer defines the scope of the exercise by declaring a paper and specific output(s) on which they will focus;
2. **Assessment**, where the reproducer reviews and describes in detail the available reproduction materials (or the “reproduction package”) and assess current levels of computational reproducibility;
3. **Improvement**, where the reproducer modifies the content and/or the organization of the reproduction materials to improve its reproducibility;
4. **Robustness checks**, where the reproducer assesses the quality of selected analytical choices from the paper; and
5. **Extension** (if applicable), where the reproducer may “extend” the current paper by including new methodologies or data. This step brings the reproduction exercise a step closer to replication.



- Graduate research:	5%	10%	5%	10%	70%
- Graduate course:	10%	25%	20%	40%	5%
- Undergrad. thesis:	10%	30%	40%	20%	0%

Figure 1 depicts the suggested levels of effort for each stage of the exercise depending on the context in which you are performing this exercise. This process need not be chronologically linear, e.g. you can realize that the scope of their reproduction is too ambitious and then switch to a less intensive reproduction. Later in the exercise, you can also begin testing different specifications for robustness while also assessing the current level of reproducibility.

1.2 Recording the results of the exercise

You will be asked to record the results of your reproduction as you make progress through different stages. After the scoping stage, you will be asked to complete a survey that records paper of choice and specific outputs to reproduce. As part of this step you will also be asked to write a brief (max two page) summary of the paper of choice. In the assessment stage, you should inspect all the reproduction materials (raw data, analysis data and code), draw diagrams that connect the target output (to reproduce) with all its inputs, and score the level of reproducibility. All this information will be recorded in a standardized spreadsheet. At the improvement stage, you can record any specific improvements report any potential increase in the reported level of reproducibility in a second short survey. Finally, in the robustness stage you will record any analytical choice and possible variations of it.

Chapter 2

Scoping

In this first stage, you are assigned or choose a paper to reproduce. This section describes the different steps required to successfully define the scope of a reproduction exercise. **Use Survey 1 to record your work as part of this stage.**

2.1 Steps

Before you invest any time in reading/summarizing the paper:

2.1.1 Check records on the ACRE platform

Check the ACRE database for previous assessments of the same paper. If there is no previous assessment, create the first entry. If there are previous entries, check if the paper-status is “open to reproductions”, and create a new entry¹.

2.1.2 Verify that reproduction materials exist {verify-rep-mat}

Verify that the paper has reproduction materials (consult the AEA unofficial Data and Code Guidance to determine whether you have everything before you start). If yes, continue. If no, verify in our records if authors have been requested these materials before. If nobody has submitted a similar request before, contact the authors using the language suggested here, and make sure

¹If the paper appears as “closed to reproductions” it means that ACRE has documented interactions between previous reproducers and original authors, showing that it is not possible to access any further required materials.

to CC (acre@berkeley.edu). If the authors have been contacted before and the information is still missing, submit a request for closing the paper for further reproductions and choose another paper.

2.1.3 Read and summarize the paper

Read the paper and write a two-page summary: focus on what is the main question the paper answers, its methodology and data sources.

2.1.4 Declare scope of the exercise

Declare if assessment will be about all or main output, and if it aims to start from raw data, or analytic data. See definitions of these concepts below.

2.1.4.1 Reproduction margins

2.1.4.1.1 Extensive margin: outputs to reproduced

- **Raw data:** A data set is considered to be raw if it corresponds to the unmodified file that was obtained by the authors from the sources cited in the paper. The only possible modification that can be made to raw data, without changing its category to processed data, is that of deleting personally identifiable information.
- **Processed data:** a raw data set that has gone through any transformation should be defined as processed data. There are two kinds of processed data:
 - *Intermediate data:* a processed dataset is defined as intermediate if it is not directly used as final input for an analysis in the final paper (including appendix). Intermediate data should not contain direct identifiers
 - *Analytic data:* data will be defined as analytic if it will be used as the last input in the workflow, to produce a statistic displayed in the final paper (including the appendix).

2.1.4.1.1.1 Types of output:

- **Tables:** Describe how to cite record location.
- **Figures:** ADD
- **In-line:** ADD

2.1.4.1.2 Intensive margin: depth of reproduction in terms of original data

- **Raw data:** A data set is considered to be raw, if it corresponds to the a unmodified file that was obtained by the authors from the sources cited in the paper. The only possible modification that can be made to raw data, without changing its category to processed data, is that of deleting personally identifiable information.
- **Processed data:** a raw data set that has gone through any transformation should be defined as processed data. Processed data can be separated into intermediate data and analytic data.
 - *Intermediate data:* a processed dataset is defined as intermediate if it is not directly used as final input for an analysis in the final paper (including appendix). Intermediate data should not contain direct identifiers.
 - *Analytic data:* data will be defined as analytic if it will be used as the last input in the workflow, to produce a statistic displayed in the final paper (including the appendix).

2.1.4.1.2.1 How to classify a data set?

To classify a data set as **raw** you can use some of the following strategies:

- Check if the data is stored in a folder labeled as “raw”
- Check if the file name has the word “raw” in some part of it
- Verify that the data set is not the output of any code script in the reproduction materials
- Verify that the same file can be independently obtained from the data source cited in the paper

To classify a data set as **analytic** you can use some of the following strategies:

- Check if the data is stored in a folder denominated as “analytic” or “analysis”
- Check if the file name has the word “analytic” or “analysis” in some part of it
- Verify that the data set is the last input required to produce some of the output (formatted or unformatted) of the paper

2.2 Sample strategies for scoping

JOEL, PLEASE ADD.

2.2.1 Strategy 1:

- First main estimates from paper.
- Then main 2 main tables and/or two main figures.
- Once done with improvements stage of these outcomes, then go back to rest of the paper.

2.2.2 Strategy 2:

- get everything to run up to analytic data, then go to improvements. Once completed go back for raw data (or in parallel?)

Chapter 3

Assessment

The goal of this stage is to provide a standardized assessment of computational reproducibility. All the details required in this section are designed to leave a record of most of the learning process behind a reproduction. Such a record facilitates incremental improvements, allowing new reproducers to pick up where others left.

First, you will be asked to provide a detailed description of all the reproduction materials. Second, you requested to connect the outputs chosen to reproduce with all of their corresponding inputs. With all this elements in place, you can then score the level of reproducibility of a specific output, and report on paper-level dimensions of reproducibility.

Use Survey 2 to record your work as part of this step.

Tip: We recommend that you first focus on one specific output (e.g. “Table 1”). After completing the assessment to this first output, you will have a much easier time translating those improvements to other outputs.

ADD: how a reproduction package should look like

3.1 Describe inputs

This section explains how to list all the input materials found or referred to in the reproduction package. First, identify all data sources and connect them with raw data files (when available). Second, locate and provide a description a brief description of the analytic files. In the last step, locate, inspect and describe all the computer code used in the paper.

In addition to the concepts of raw and processed data, defined already, this section use the following concepts:

- **Cleaning code:** a script can be classified as primarily data cleaning if most of its content is dedicated to actions such as: deleting variables or observations, merging data sets, removing outliers, and reshaping the structure of the data (from long to wide or vice versa).
- **Analysis code:** a script can be classified as primarily analysis code if most of its content is dedicated to actions such as: running regressions, running hypothesis tests, computing standard errors, and imputing missing values.

– ADD DEFINITION FOR REPRODUCTION PACKAGE.

3.1.1 Data sources and raw data

In the paper to reproduce, find references to all the data sources used in the analysis. A data source is usually described in a narrative form, for example the body of the paper can have text like “...for earnings in 2018 we use the Current Population Survey...”– in this example, the data source is “Current Population Survey 2018”. It is mentioned for the first time on page 1 of the appendix, so its location should be recorded as “A1”. Do this for all the data sources mentioned in the paper.

Next, look at the reproduction package and map the *data sources* mentioned in the paper to the *data files* in the available materials. In addition to looking at all the existing data files, it is recommended to review the first lines of all code files (especially cleaning code), looking for lines that call the data sets. By inspecting this scripts you might be able to understand how each different data sources is used, and possibly identify any missing files from the reproduction package.

Record all the information of this section into a assessment tool with the following structure:

Raw data information:

data_source	page	data_files
"Current Population Survey 2018"	A1	cepr_march_2018.dta
"DHS 2010 - 2013"	4	nicaraguaDHS_2010.csv; boliviaDHS_2010.csv; nicaraguaDHS_2011.csv; nicaraguaDHS_2012.csv; boliviaDHS_2012.csv; nicaraguaDHS_2013.csv; boliviaDHS_2013.csv
"2017 SAT scores"	4	Not available

-----	-----	-----	-----
...
-----	-----	-----	-----

3.1.2 Describe analytic data sets

List all the analytic files that you find in the reproduction materials and identify its location relative to the master reproduction folder¹. Record all this information in a assessment tool.

On an initial review, it will probably be hard to provide a one-line description of each analytic data set. But as you progress through the exercise, return to the spreadsheet and add a one-line description of the main content in each file (for example: `all_waves.csv` has the simple description `data for region-level analysis`).

The resulting report will have the following structure:

Analysis data information:

-----	-----	-----
analysis_data	location	description
-----	-----	-----
final_data.csv	/analysis/fig1/	data for figure1
-----	-----	-----
all_waves.csv	/final_data/v1_april/	data for region-level analysis
-----	-----	-----
...
-----	-----	-----

3.1.3 Describe code scripts

List all the code files that you find in the reproduction materials and identify its location relative to the master reproduction folder. Review the beginning and end of each code file and identify the inputs required to successfully run the file. Example of inputs are data sets or other code scripts that are typically found at the beginning of the script (e.g.: `load`, `read`, `source`, `run`, `do`). Examples of outputs are other data sets, or plain text files that are typically at the end of a script (e.g.: `save`, `write`, `export`). Record all this information in the assessment tool.

As you gain understanding of each code script, you will likely find more inputs and outputs – we encourage you update the assessment tool. Once finished with the reproduction exercise, classify each code file as analysis or cleaning type.

¹relative location takes the form `/folder_in_rep_materials/sub_folder/file.txt`, in contrast to absolute location that has the form `username/documents/projects/repros/folder_in_rep_materials/sub_folder/file.txt`

This subjective assessment should be made base on the students' interpretation of the main role of each script.

Record all this information in a assessment tool

file_name	location	inputs	outputs
output_table1.do	/code/analysis/	analysis_data01.csv	output1_part1.txt
data_cleaning02.R	/code/cleaninig/	admin_01raw.csv	analysis_data02.csv
...

3.2 Connect each output to all its inputs

Draw diagrams from output to raw data sources. For more examples of diagrams connecting final output to initial raw data, see [here](#).

```

table 1
  [code] formatting_table1.R
  output1_part1.txt
  |   [code] output_table1.do
  |       [data] analysis_data01.csv
  |           [code] data_cleaning01.R
  |               [data] survey_01raw.csv
  output1_part2.txt
  |   [code] output_table2.do
  |       [data] analysis_data02.csv
  |           [code] data_cleaning02.R
  |               [data] admin_01raw.csv

```

This diagram can be represented in data format by specifying how each component depends to its inputs. For example:

Data representation of diagram behind Table 1.

ouput	order	component	depends_on	inpt_type
-------	-------	-----------	------------	-----------

table1 1	table1	formatting_table1.R	code	
----- -----	-----	-----	-----	
table1 2	formatting_table1.R	output1_part2.txt	output	
----- -----	-----	-----	-----	
table1 3	formatting_table1.R	output1_part1.txt	output	
----- -----	-----	-----	-----	
table1 4	output_table1.do	analysis_data01.csv	data	
----- -----	-----	-----	-----	
... 	
----- -----	-----	-----	-----	

Record all this information in the assessment tool. In case of any difficulty translating the diagram into a spreadsheet, students can draw it with pen and paper, take a picture and upload the picture in the assessment survey.

3.3 Assign a reproducibility score

Once all the possible inputs have been identified, and there is a clear understanding of the connection between the outputs and inputs, it is possible to assess the output-specific level of reproducibility.

The following concepts will be used in this section:

- **Computationally Reproducible from Analytic data (CRA):** the output can be reproduced with minimal effort starting from the analytic data sets.
- **Computationally Reproducible from Raw data (CRR):** the output can be reproduced with minimal effort from the raw data sets.
- **Minimal effort:** spending five minutes or less in getting the code running. This five minutes do not include the computing time.

3.3.1 Levels of Computational Reproducibility for a Specific Output

We can now outline different levels of computational reproducibility. Each level is defined on the basis of data and code availability, possible improvements, and whether or not it achieves some type of reproducibility. In the next chapter we describe each possible improvement with more detail.

MAKE EXPLICIT UNDERLYING VALUATIONS:

- L1: The worst is to have no access to any data or code.

- L2: Having only code is better than nothing, but worst than any other combination
- L3: Having analytic data but not code is better than having code and no data.
- L4 and L5: Conditional on having all analytic material. Reproducible is better than not
- L6: Having raw and analytic data is better than having code but no data.
- L7: Given that there is raw and analysis data, having cleaning code only does not add much. having analysis code does add (to lvl 7 and lvl 8 if CRA)
- L9: Having all materials that but not being able to reproduce is better than all previous cases. Achieving CRA (lvl 10) and CRR (lvl 11) are the best levels according to this criteria.

ADD ONE QUESTION AT THE END OF SURVEY TWO TO ASK REPRODUCER IF THEY AGREE WITH THIS ORDER, AND IF THEY WOULD ADD/DELETE/MODIFY ANY SPECIFIC LEVEL

The assessment is made at the output level. Hence a paper can be highly reproducible for its main results, but suffer of low reproducibility for other outputs. The assessment is on a 10-level scale, where 0 represents that, under the current circumstances, reproducers cannot access any reproduction materials and 10 represents access to all the materials and the target outcome can be reproduced starting from raw data.

Level 1 (L1): There are no data or code available of any type. Possible improvements include adding: raw data (+AD), analysis data (+RD), cleanig code and analysis code (+CC, +AC).

Level 1.5 (L1.5): There are only code scripts available, but no data of any type are available. Possible improvements include adding: raw data (+AD), analysis data (+RD).

Level 2 (L2): There are analytic data available, but no raw data or any type of code. Possible improvements include: adding raw data (+RD) and adding analysis code (+AC).

Level 3 (L3): Both analytic data sets and analysis code are available. However, the code does not run or produces different results than those of the paper (not CRA). Possible improvements include obtaining raw data (+RD) or debugging the analysis code (DAC).

Level 4 (L4): Both analytic data sets and analysis code are available, and they produce the same output as in the paper (yes CRA). The reproducibility package can still be improved by obtaining the original raw data sets, or by documenting the steps required to obtain those files.

Level 5 (L5): All data, analytic and raw, are available. However, some or all the codes for cleaning and analysis are missing. Steps for improvement include adding analysis code (+AC) and/or cleaning code (+CC).

Level 6 (L6): All data and analysis code are available. However, the code does not run or produces different results than those of the paper (not CRA). Possible improvements include adding the missing cleaning code (+CD) or debugging the analysis code (DAC).

Level 7 (L7): All data and analysis code are available, and they produce the same output as in the paper (yes CRA). The reproducibility package can still be improved by adding the missing cleaning code (+CD).

Level 8 (L8): All materials (raw and analysis data, and cleaning and analysis code) are available. However, the code does not run or produces different results than those of the paper (not CRR and not CRA). Possible improvements include debugging the cleaning code (DCC) or debugging the analysis code (DAC).

Level 9 (L9): All materials (raw and analysis data, and cleaning and analysis code) are available, and the analysis code produces the same output as in the paper (yes CRA). However the cleaning code does not run or produces different results than those of the paper (not CRR). Possible improvements include debugging the cleaning code (DCC).

Level 10 (L10): All materials are available and produce the same results as in the paper with minimal effort, starting from the analytic data sets (yes CRA) and from the raw data (yes CRR).

The following figure summarizes the different levels of computational reproducibility (for any given output). For each level, there will be possible improvements that have been done already (-), that can be done to move up one level of reproducibility () or that are out of reach given the current level of reproducibility (x).

Figure 1: Levels of Computational Reproducibility and Possible Improvements

		Possible improvements					
		Level	+Analysis Data	+Raw Data	+Analysis Code(AC)	+Cleaning Code(CC)	Debug AC
What data are available?							
None	L1						x
Analytic data only. Code?							
No code or cleaning code only.....	L2	-					x
Analysis code only. Is it CRA?							
No.....	L3	-		-			x
Yes.....	L4	-		-		-	
Raw & Analytic data. Code?							

None	L5	-	-		
Analysis code only. CRA?					
No.....	L6	-	-	-	
Yes.....	L7	-	-	-	
A. and cleaning code. Is it CRR?					
No. CRA?					
No.....	L8	-	-	-	-
Yes.....	L9	-	-	-	-
Yes.....	L10	-	-	-	-

Choose the appropriate level of computational reproducibility and record it using the following format.

output_name	level	additional_explanation	other_info
table 1	4		...
table 2	7		...
figure 1	5		...
...

Record all this information in the assessment tool. You will be asked to provide this information in the assessment and improvement survey.

3.3.2 Reproducibility dimensions at the paper level

In addition to an output-specific assessment of computational reproducibility, there are several practices that facilitate the overall computational reproducibility of the paper. These practices are described in detail in the Improvement chapter. In this section of assessment it is only required that you verify that the original reproduction package made use of any of the following:

- version control
- dynamic document
- translate to open source software
- file organization

- computing capsule (e.g. CodeOcean, Binder, etc.)

Congratulations! You have now completed the Assessment stage. You just provided a concrete building block of knowledge to better understand the state of reproducibility in Economics.

Please continue to the next section where you can help to improve it!

Chapter 4

Improvements

After completing the assessment of the current reproducibility packages, it is possible to propose ways to increase the reproducibility of the paper. Creating improvements provides an opportunity to gain a more in-depth knowledge of the paper, including its methods, findings, and its overall contribution. In addition to this individual benefits, each contribution will be assessed by the ACRE community and can potentially be used by students and researchers around the world as an improved version of the reproducibility package for the paper.

As in the Assessment section, we recommend that you first focus on one specific output (e.g. “Table 1”). After completing the improvements to this first output, you will have a much easier time translating those improvements to other outputs.

4.1 Types of output-level improvements

4.1.1 Add missing raw data files or meta-data (+RD)

It is common that reproductions packages do not include all the original raw datasets. To obtain any missing raw data, or information about them, follow these steps:

- 1 - Identify a specific missing file. During the assessment stage, you identified all data sources from the paper and appendices (column **data_source** in this standardized spreadsheet). However, some data sources (as delivered to the original investigators) might be missing one or more data files. You can sometimes find the specific name of those files by looking at the beginning of the cleaning code scripts. If you find the name of the file, record it in the **data_file** field of the same spreadsheet

as above. If not, recorded as to “Some (or all) of the files used in the paper corresponding to the data source X”.

- 2 - Verify whether this file(s) can be easily obtained from the web.
 - 2.1 - If yes: obtain the missing files and add them to the reproducibility package. Make sure to obtain permission to repost this data. See tips for communication for a template email.
 - 2.2 - If no: proceed to step 3.
- 3 - Verify the ACRE database for previous attempts to contact the authors regarding this paper.
- 4 - Contact the original authors and kindly request the original materials. Be mindful of the authors’ time, and remember that the paper you are trying to reproduce was possibly published at a time where standards for computational reproducibility were different. See tips for communication for sample language on how to approach the authors.
- 5 - If the data sets are not available due to confidentiality or proprietary issues, the researcher conducting the reproduction can still improve the reproduction package by providing detailed instructions, including contact information and possible costs, for future researchers to follow.

In additions to the efforts to obtain raw data, you can also contribute by obtaining missing analysis data.

4.1.2 Add missing analysis data files (+AD)

Analysis data can be missing for two reasons: (i) raw data exists, but the procedures to transform it into analysis data are not fully reproducible, or (ii) some or all raw data is missing and some of all the analysis data is not included in the original reproduction package. To obtain any missing analysis data, follow these steps:

- 1 - Identify the specific name of the missing data set. Typically this information can be found in some of the analysis code that calls such data in order to perform an analysis (eg `analysis_data_03.csv`).
- 2 - Verify that such data cannot be obtained by running the data cleaning code over the raw data.

- 3 - Verify the ACRE database for previous attempts to contact the authors on this topic.
- 4 - Contact the authors and request the specific data set.

4.1.3 Add missing analysis code (+AC)

Analysis code can be added when there are analytic data files, but some or all the methodological steps are missing from the code. In this case, follow these steps:

- 1 - Identify the specific line/paragraph in the paper that describes the analytic step that is missing from the code (eg “we impute missing values to...,” or “we estimate this regression using a bandwidth of ...”).
- 2 - Identify the code file and the approximate line in the script where the analysis can be carried out. If no relevant code file is found, identify the location of the missing file relative to the steps in the reproduction diagram.
- 3 - Verify the ACRE database for previous attempts to contact the authors on this issue.
- 4 - Contact the authors and request the specific code files.
- 5 - If no response from the authors, researchers reproducing the paper are encouraged to attempt to recreate the analysis based on their interpretation of the paper, and filling in any missing piece by making explicit assumptions.

4.1.4 Add missing data cleaning code (+CC)

Data cleaning (processing) code can be added when there are certain steps missing in the creation/re-coding of variables, merging, subsetting of the data sets, and other steps related to data cleaning and processing. Researchers conducting the reproduction should follow the same steps (1-5) as when adding missing analysis code.

4.1.5 Debug analysis code (DAC)

Whenever any code is available in the reproduction package, reproducers should be able to debug those scripts. There are four types of debugging that you can add as part of the Improvements step:

- *Code cleaning*: instructions are simplified (e.g. by wrapping repetitive steps in a function or a loop) or redundant code is removed (eg. old code that was commented out), while keeping the original output intact.
- *Performance improvement*: instructions are replaced by other that perform the same tasks but take less time (eg. choosing one numerical optimization algorithm over another, but obtaining the same results).
- *Environment set up*: the code is modified to include correct paths to files, specific versions of software, and instructions to install missing packages or libraries.
- *Correcting errors*: a coding error will occur when a section in the code of the reproduction package executes a procedure that is in direct contradiction with the intended procedure expressed in the documentation (paper or comments of the code). For example, an error happens if the paper specifies that the analysis is performed on the population of males, but the code restricts the analysis to females only. Please follow the ACRE procedure to report coding errors.

4.1.6 Debug cleaning code (DCC)

Same as for analysis code, only separate for reporting purposes.

4.1.7 Reporting results

Track all the different types of improvements implemented and record in this standardized spreadsheet of the assessment tool with the following structure:

Level-specific quality improvements: add data/code, debug code.

output_name	imprv	description_of_added_files	lvl
table 1	+AD	ADD EXAMPLES	5
table 1	+RD	ADD EXAMPLES	5
table 1	DCC	ADD EXAMPLES	5
figure 1	+CC		6
figure 1	DAC		6
inline 1	DAC		8
...

4.2 Types of paper-level improvements

In addition to the different levels of computational reproducibility described in the previous sections, there are at least six additional that you can implement to improve the overall reproducibility of a paper. Such additional improvements can be applied across levels (including level 10).

- 1 - Set up the replication package using version control software (Git).
- 2 - Improve documentation: add extensive comments to the code.
- 3 - Integrate documentation with code: adapt the paper into a literate programming environment (eg: using Jupyter notebooks, RMarkdown, Stata Dynamic Doc).
- 4 - Re-write the code from proprietary statistical software (eg Stata, Matlab) into an open-source statistical software (eg R, Python, Julia).
- 5 - Re-organize the reproduction materials into a set of folders and sub-folders that follow standardized best practices, and add a master script that executes all the code in order and with no further modifications. See AEA's reproduction template.
- 6 - Set up a computing capsule that executes all the reproduction in the browser without the need to install any software. See for examples Binder and Code Ocean.

4.2.1 Reporting improvements

Reproducers will be asked to provide this information in the assessment and improvement survey.

Chapter 5

Checking for Robustness

[UNDER CONSTRUCTION]

- Identify all possible analytical choices: original and repeated ones.
- Identify type of choice.
- Identify choice value.
- Suggest choice alternative and justify (one line)

5.1 Identifying Analytical Choices

As part of the requirements to demonstrate comprehension of the paper and the code researchers conducting the reproduction will be asked to record all the analytical choices identified during the code review process. This is done in two steps: first adding comment lines into the code files where an analytic choice are found, and second, compiling those analytic choices into a standardized data set.

In your copy of the replication code, add the comment `"# ANALYTICAL CHOICE OF TYPE ____ . RECORDED FOR THE FIRST TIME [HERE or IN "FILE_NAME-LINE_NUMBER"]"` above each analytical choice detected in the code. Possible types of analytical choices include (but are not limited to):

- Analytical choices in data cleaning code:
 - Variable definition
 - Data sub-setting

- Data reshaping (merge, append, long/gather, wide/spread)
- Others (specify as “processing - other”)
- Analytical choices in analysis code:
 - Regression function (link function)
 - Key parameters (tuning, tolerance parameters, etc.)
 - Controls
 - Adjustment of standard errors
 - Choice of weights
 - Treatment of missing values
 - Imputations
 - Other (specify as “methods - other”)

Once finished, transcribe all the information on analytical choices into a data set. For the `source` field type “original” whenever the analytical choice is identified for the first time, and `file_name-line number` every time that the same analytical choice is applied subsequently (for example if a analytic choice is identified for the first time in line 103 and for a second in line 122 their respective values for the `source` field should be `original` and `code_01.do-L103` respectively).

The resulting data base should have the following structure:

file_name	line_number	choice_type	choice_value	Source
code_01.do	73	data	males	original
		subsetting		
code_01.do	122	variable	income = wages +	“code_01.do-
		definition	capital gains	L103”
code_05.R	143	controls	age, income,	original
			education	
...

5.2 Choose and justify alternative values for analytical choices

5.3 Test the robustness of results

Test the robustness of results to alternative (sensible) specifications

- Identify sensible alternatives to analytical choices.
- Sample from sensible analytical choices and re-run: report how much do results change as fraction of standard deviations.
- Jackknife the preferred estimate.
- Use ML to select among covariates...

Chapter 6

Concluding the reproduction

[UNDER CONSTRUCTION]

Walk the students on checking that they have completed all the steps and where can they see their output.

6.0.1 Final products

- One-page introduction describing why you chose this paper
- Two-page summary of paper
- 2 Completed surveys:
 - i - General information about the paper and specific information about output to reproduce.
 - ii - Assessment of how (computationally) reproducible is the paper; description of improvements to its reproducibility; record of all the analytical choices identified in the exercise.
- ACRE report card with all the improvements that were created by the researcher reproducing the paper. The list of improvements will be made public and original authors will receive a copy of the report card. The option of anonymity will be provided to the researchers reproducing the paper.
- New Readme file (autogenerated).
- Data with all analytical choices identified.

- ?? Narrated description of improvements to original CR of the paper, assessment of robustness of results. Lessons from the exercise and possible future extensions.

Chapter 7

Tips for Communication

[UNDER CONSTRUCTION]

7.1 For students or researchers conducting a reproduction

7.1.1 Contacting the Authors of the Original Study

7.1.1.1 Verify in the ACRE database, if authors have been contacted before

7.1.1.2 Email them

7.1.1.2.1 Template language

draft here

Remember to link template language to documentation for later, to check with team

7.1.1.2.1.1 Requesting raw/analytic data and code

7.1.1.2.1.2 Following up on a non-response requesting additional contact information

- point out the level of reproducibility of project and how to improve

7.1.1.2.1.3 Reporting improvements done to current reproducibility package

7.1.1.2.1.4 Requestion permission to re-post public data in new reproduction packages

Would like to repost your data

You already gave permission to authors XYZ

Just trying to improve this authors reproducibility

Your permission would allow students and researchers do more work in this area.

7.1.1.2.1.5 If response

Follow up with request of steps necessary for any researcher in the future to access this data.

- **Coding errors:** a coding error will occur when a section in the code, of the reproduction package, executes a procedure that is in direct contradiction with the intended procedure expressed in the documentation (paper or comments of the code). For example an error happens if the paper specify that the analysis is perform on the population of males, but the code restricts the analysis to females only. Please follow the ACRE procedure to report coding errors.

7.2 For original authors contacted by reproducers

7.3 Responding to inquiries

Chapter 8

Code of Conduct

[UNDER CONSTRUCTION]

Chapter 9

Contributions

TO DO

9.1 Ask for feedback on guidelines

9.2 List of Contributors

Chapter 10

Reproduction Diagrams

10.1 Different Scenarios

JOEL: Please fill-in.

10.1.1 Complete

```
table 1
  [code] formatting_table1.R
  output1_part1.txt
  |   [code] output_table1.do
  |   [data] analysis_data01.csv
  |   [code] data_cleaning01.R
  |   [data] survey_01raw.csv
  output1_part2.txt
  [code] output_table2.do
  [data] analysis_data02.csv
  [code] data_cleaning02.R
  [data] admin_01raw.csv
```

10.1.2 Raw data and analytic data, but cleaning code is missing.

```
table 1
  [code] formatting_table1.R
  output1_part1.txt
  |   [code] output_table1.do
  |   [data] analysis_data01.csv
```

```

|           [code] MISSING FILE(S)
|           [data] survey_01raw.csv
output1_part2.txt
|           [code] output_table2.do
|           [data] analysis_data02.csv
|           [code] MISSIN FILE(S)
|           [data] admin_01raw.csv

```

10.2 Additional resources

Create a section with short summaries of great resources for comp. repro and invite reader to contribute.

10.2.1 Some summaries

10.2.2 Summary on reproducible workflow (Chapter 11) from Christensen et al. (2019):

- TODO

10.2.3 Links

- Project TIER
- IDB's cheatsheet for transparency, reproducibility and ethics
- Lars Vilhuber LDI's Wiki for Reproducibility. Particularly this section.
- World Bank DIME's Wiki for transparent and reproducible research.
- Dynamic documents in R, Python and Stata
- Git resources:
 - Jenny Bryan's book and video
 - Github learning lab
 - Udacity's intro
 - Git for poets
 - Combining GitHub and Dropbox
 - Atlassian intro to Git

- Software Carpentry tutorial from the command line
- Open Science Framework (OSF)
- R for Stata users

Chapter 11

Additional resources

Create a section with short summaries of great resources for comp. repro and invite reader to contribute.

11.1 Some summaries

11.1.1 Summary on reproducible workflow (Chapter 11) from Christensen et al. (2019):

- TODO

11.2 Links

TODO: Add and classify

- Project TIER
- IDB's cheatsheet for transparency, reproducibility and ethics
- Lars Vilhuber LDI's Wiki for Reproducibility. Particularly this section.
- World Bank DIME's Wiki for transparent and reproducible research.
- Dynamic documents in R, Python and Stata
- Git resources:
 - Jenny Bryan's book and video

- Github learning lab
 - Udacity's intro
 - Git for poets
 - Combining GitHub and Dropbox
 - Atlassian intro to Git
 - Software Carpentry tutorial from the command line
- Open Science Framework (OSF)
- R for Stata users

Bibliography

Christensen, G., Freese, J., and Miguel, E. (2019). *Transparent and reproducible social science research: How to do open science*. University of California Press.

King, G. (1995). Replication, replication. *PS: Political Science and Politics*, 28:444–452. See updates to this paper for how I use this paper as a class assignment now.