Replication files: Understanding Markets with Socially Responsible Consumers, Marc Kaufmann, Peter Andre, Botond Kőszegi, The Quarterly Journal of Economics

Data and Code Availability Statement

All **data** are made available.

All **code** is made available.

License

Data and code made available via this replication package are licensed under a Creative Commons Attribution–NonCommercial 4.0 International (CC BY-NC 4.0) license.

Required software and version information

R and RStudio are required. The code was programmed on macOS (13.4) in RStudio 2021.09.02, using R 4.1.2.

The packages `groundhog` and `rjson` need to be installed manually.

In R, the package management package `groundhog` is used. All other required packages will automatically be installed. The package version at date 2021-10-01 will be installed. This ensures replicability. For more information, please see the documentation of the `groundhog` package.

How to start?

1. Start any work session by opening the RStudio project `SRC_replication.Rproj` in RStudio. This will automatically set the correct working directory.
2. Open the main script `code/main.R` and run the code in the `PREPARATION` section. If you use Windows, you need to manually change line 7 (see also comment in line 4). The `PREPARATION` section will install the required packages. It will also load a function named `file_path` which I use to handle file paths throughout the project (for details, see below).
3. Depending on your goals, ...
    - **Replicate the full project.** Execute the main script `code/main.R` . It executes the full analysis and generates the figures and tables of the paper.
    - **Replicate a specific analysis.** Execute the `PREPARATION` and `DATA CLEANING` section of the main script `code/main.R` . Then, execute the script of interest or open it (see section "Folder structure") and execute it manually.
    - **Run new analyses with the data.** Execute the `PREPARATION` and `DATA CLEANING` section of the main script `code/main.R` . Read section "Data" and "Codebook".

Folder structure

- `code/` : Contains all code.
    - `code/calculations/` : Contains code used to derive in-text statistics.
    - `code/cleaning/` : Contains the code of the data management.
    - `code/figures/` : Contains the code that derives the figures.
    - `code/misc/` : Miscellaneous scripts.
    - `code/specs/` : Specifications or definitions that are used throughout the analysis and data management.
    - `code/tables/` : Contains the code that derives regressions and tables.
- `data/` : Contains the raw data (see "Data").
- `out/` : Contains all outputs of the data management and data analysis.
    - `out/data_out/` : Cleaned data, potentially temporary data manipulation output (see "Data").
    - `out/figures/` : Figures.
    - `out/tables/` : Latex tables.

Overview of all analysis scripts

Figures The following list summarizes which figure is created by which script (if not noted otherwise the file ends on `.R`) and how the output is named (if not noted otherwise the file ends on `.pdf`).

- Figure 1: `fig_beliefs`
- Figure 2: `fig_conseq`
- Figure C.1: `fig_explanations`
- Figure C.2: `fig_beliefs_robust` and `fig_explanations_robust`
- Figure C.3: `fig_explanations_conseq`
- Figure C.4: `fig_conseq_robust` and `fig_explanations_conseq_robust`
- Figure C.5: `fig_robustness_studies.R` and `fig_production.pdf`
- Figure C.6: `fig_robustness_studies.R` and `fig_numeric.pdf`
- Figure C.7: `fig_robustness_studies.R` and `fig_beliefs_incentives.pdf`

Tables The following list summarizes which table is created by which script (if not noted otherwise the file ends on `.R`) and how the output is named (if not noted otherwise the file ends on `.tex`).

- Table 1: manually prepared
- Table 2: manually prepared
- Table C.1: `tab_demographics.R` and `tab_demo.tex`
- Table C.2: `reg_attrition`
- Table C.3: `reg_beliefs_hetero`
- Table C.4: `reg_conseq_hetero`
- Table C.5: `manually prepared`
- Table C.6: `manually prepared`
- Table C.7: `tab_demographics_robustness.R` and `tab_demographic_robustness.tex`

Calculations A few additional scripts derive in-text numbers.

- `calc_ancillary_codes.R` : Calculates the frequency of ancillary codes for Appendix C.4.
- `calc_attrition.R` : Calculates attrition statistics for Appendix C.1.
- `calc_beliefs.R` : Calculates statistics for dampening beliefs that are reported in the main text.
- `calc_concern_median.R:` Calculates the median valuations that are reported in the main text.
- `calc_IRR.R` : Calculates the inter-rater reliability that is reported in the main text and Appendix C.4.
- `calc_response_duration.R` : Calculates the response duration that is reported in Appendix C.1.

Data

Raw data: All raw data are in the `data/` folder.

- `coded_data/` : This folder contains the manually coded qualitative text data, before cleaning and cross-validation.
- `consumers_raw.sav` : Main study
- `demographics_benchmark.csv` : Demographic benchmark statistics
- `robustness_beliefs.sav` : Robustness studies

I replaced the IDs of Prolific respondents with randomly generated IDs.

Cleaned data: All cleaned data are in the `out/data_out/` folder. Typically, each file is available as `rds` (an R-internal storage format) and as Stata `dta` file. The variables are not labeled. But a short description of the most important variables is provided in a separate codebook (see Codebook below).

- `robustness_beliefs.dta` : Robustness studies
- `wide.dta` : Main study ("wide" indicates that the data are reported on the respondent level, though we do not use "long" data formats in this project)

All other files are auxiliary or intermediate files which can be ignored.

Codebook

A short description of the most important variables is provided in the `codebook/` folder.

Special features

`path_to` All relative file paths are stored in `misc/specs/file_paths.json`. The function `path_to()` allows to access these paths with a keyword (see the json file). It is used throughout the code.

Regressions and `stargazer` wrapper All regressions are estimated with the `lm` method. Robust standard errors or clustered standard errors are derived to yield the same output as in Stata. Regression tables are created with the `stargazer` package and are further manipulated and polished by adjusting the latex code with the help R text manipulations. Both the estimation and the generation of the latex code take place in wrapper functions (`code/functions/functions_reg`).