DATA and STATISTICAL CODE:

Gender, Confidence, and the Mismeasure of Intelligence, Competitiveness and Literacy

by

Glenn W. Harrison, Don Ross and J. Todd Swarthout

January 2025

The data and statistical code are contained in three directories, referring to the three sets of experiments in the paper:

- intelligence
- gender and competitiveness
- literacy

Each can be run separately, and has a similar structure. The code for the **intelligence** directory is explained in more detail, since similar procedures are used in the other directories.

All statistical analysis code has been written in *Stata*, and uses version 18.0. Most of the debugging used version 17, so we would expect that it runs under earlier versions than 18. At the beginning of each of the command DO files, there are lists of user-written programs that are likely to be accessed. These are commented out, so as not to slow the code down every time it is run, but they may need to be un-commented once to make sure the programs are in place. For example, in the Main.do file in the intelligence directory, we have this code:

```
* package install
*capture: ssc install vcemway
*capture: ssc install norm
*capture: ssc install corrtable

* changing, new code, so be sure to get the latest version
*net install qmodel, replace from(http://www.imm.ki.se/biostatistics/stata)
*ssc install mylabels
```

Intelligence

The *Stata* command file **Main.do** runs all of the analyses. It is driven by a series of globals near the top, which determine blocks of code to be run or not. All have been turned on here, except the global $useRECOVERED that determines if the analyses use the reported beliefs or the recovered beliefs. Since we need to explain below how to recover beliefs, we initially have $useRECOVERED turned off. Once the beliefs have been recovered, as explained below, this global can be set to "y" to repeat the analyses with recovered beliefs (at that point some of the initial data-reading blocks could be skipped, but they are fast anyway). Here are the globals:

```
* globals
global doRiskLink "y"
global doBaseline "y"
global doData "y"

global doRECOVERY "y"
global getRECOVERED "y"
global useRECOVERED "n"

global doGignac "y"
global doArthurDay "y"

* some slower graphs and estimates, might skip
global doGraphs "y"
global doEstimates "y"
```

The **doRiskLink** code refers to the linking of data across sessions, requiring us to have common subject ID variables: specifically, the risk preference task was run on one day, and the Raven task on a later day, with the subjects linked. The **doBaseline** code refers to reading in the data from the Baseline "paper and pen" experiment. The **doData** code refers to the main Raven data that we collected from 2019 and 2022.

The **doRECOVERY** code generates the files needed for the Raven token allocation reports to be saved in a format that we can then process to recover them as beliefs (explained below). The **getRECOVERY** code reads these recovered beliefs in, if they have been generated, and saves a data file with the beliefs. The **useRECOVERY** global tells later code whether to use the beliefs or to use the reports. So the logic is that you can generate and get the recovered beliefs, and then decide whether they are to be used or not for the analysis.

The **doGignac** and **doArtherDay** code refers to reading in the raw data generously supplied by those authors, referenced in the paper, and to be compared to our own "paper and pen" Baseline data.

The **doGraphs** and **doEstimates** code refers to the generation of detailed graphs and statistical estimates.

The belief recovery code is in a sub-directory called **Recover Beliefs – Raven**. It contains the following files:

- **belief_recovery.do** contains the code used to recover beliefs, using methods documented in Harrison, Monroe and Ulm [2022]. Do not tamper with this code: it is called by another DO file listed below.
- **Beliefs on Raven from 2019 and 2022.do** is indeed the program that reads in the data saved by running the doRECOVERY block in **Main.do**, mentioned above, prepares the data for recovery, and then recovers it. It generates sub-directories **recovery_room** and **recovered**, which will be filled with a long list of files, as explained in the appendix of Harrison, Monroe and Ulm [2022]. This program takes about 3.7 hours on a laptop, most of it in setting up the belief files to be processed: the BHM for each subject and each question needs to be merged with the token-allocations for that subject and question, and saved.
- **bhm_rdu2019_posterior.dta** is a *Stata* data file that contained the posterior estimates of the

RDU model of risk preferences for each subject from the 2019 experiments, using the methods and software explained in Appendix A of Harrison, Gao and Tchernis [2023]. This file is read in and used by **Beliefs on Raven from 2019 and 2022.do**.

- **bhm_rdu2022_posterior.dta** contained the posterior estimates for subjects from the 2022 experiments. This file is read in and used by **Beliefs on Raven from 2019 and 2022.do**.
- **Evaluate Recovered Beliefs on Raven.do** does post-processing of the recovered beliefs, and prepares a detailed file of results as well as a summary file, which we use. This program takes about 3.1 hours on a laptop.
- **link_raven_2019.dta** provides data to link the same 2019 subjects across tasks.
- **link_raven_2022.dta** provides data to link the same 2022 subjects across tasks.
- **raven_beliefs_recovery.dta** is the file generated by the doRECOVERY block in **Main.do**, mentioned above.
- **raven_summary.dta** is the final, processed files with the recovered beliefs data. This will be read in by the getRECOVERY block in **Main.do**, mentioned above.

We include the **raven_summary.dta** file in the distribution, so this belief recovery step does not need to be run in order to replicate our results.

Gender and Competitiveness

The *Stata* command file **GSU Raven Data -- Gender and Competitiveness 2024.do** runs all of the analyses. It is driven by a series of globals near the top, which determine blocks of code to be run or not. All have been turned on here:

```
* task need to be run first, then restart with "n"
global getRisk "y"

* use the recovered beliefs rather than reports
global useRECOVERED "y"

* tasks
global doData "y"
global doRaven "y"
```

The **getRisk** code reads in the EUT and RDU risk preferences, in the form of their Bayesian posterior distributions. The **useRECOVERED** code again tells the code to use recovered beliefs or reported beliefs. The **doData** code reads in the raw data from these experiments from the sub-directory **raw data**, and the **doRaven** code undertakes the statistical analysis of the data.

A sub-directory **Niederle and Vesterlund QJE** contains raw data files for the discussion of their experiment in Appendix C.

A sub-directory **Recover Beliefs** undertakes the same type of recovery of beliefs as undertaken in the **intelligence** directory. It consists of these files:

- **belief_recovery.do**, as discussed earlier, and not to be edited.

- **bhm_eut2022_posterior.dta** is the posterior for the EUT risk preferences of the subjects, estimated using the procedures of Gao, Harrison and Tchernis [2023].
- **bhm_rdu2022_posterior.dta** is the posterior for the RDU risk preferences of the subjects, estimated using the procedures of Gao, Harrison and Tchernis [2023].
- **Recover Beliefs on GSU Gender and Competitiveness.do** processes the data from the experiment, generated from the main directory described above.
- **Evaluate Recovered Beliefs on GSU Gender and Competitiveness.do** is the post-processing code, to be run after the beliefs have been recovered.
- **GC_beliefs.dta** is the raw data for belief recovery generated by the program described earlier, and saved into this sub-directory.
- **gc_eut_summary.dta** is the final, processed file with the recovered beliefs data using EUT risk preferences. This will be read in by the useRECOVERED block described above.
- **gc_rdu_summary.dta** is the final, processed file with the recovered beliefs data using RDU risk preferences. This will be read in by the useRECOVERED block described above.

Literacy

The *Stata* command file **GSU Risk and Beliefs Data on Core Literacy.do** collates the raw beliefs data and risk preference estimates into files that can then be processed to recover the beliefs. It should be run first.

The *Stata* command file **Evaluate Recovered Beliefs on Core Literacy.do** then runs all of the analyses. It is driven by a series of globals near the top, which determine blocks of code to be run or not:

```
* tasks
global doIND "y"
global doPOOL "y"
global doREPORT "y"

* use reports rather than recovered beliefs (for Raven)
global useREPORTS "n"

* do risk display or not
global doRiskDisplay "n"
```

The **doIND** code generates the data files for belief recovery, recovers them, and then processes them at the level of the individual subject. The **doPOOL** code collates the individual recovered beliefs and does some basic analysis of them. The **doREPORT** code generates the analyses and displays of the data for this paper. The **useREPORTS** global tells this code whether to use the token allocation reports or the recovered beliefs. When set to "n" as here, it therefore uses the recovered beliefs. The **doRiskDisplay** code generates displays of the RDU risk preferences of each subject, and is quite tedious to run and read.

The files in this directory are as follows:

- **belief_recovery.do**, as discussed earlier, and not to be edited.
- **bhm_static_bayes_posterior.dta** is the posterior for the RDU risk preferences of the

subjects, estimated using the procedures of Gao, Harrison and Tchernis [2023].

- **core_literacy.dta** is the raw literacy data on beliefs for the two questions considered here, in the form of token allocations of reports. It is used to generate the files for belief recovery.
- **GSU Risk and Beliefs Data on Core Literacy.do** and **Evaluate Recovered Beliefs on Core Literacy.do**, described above.

This process will take 6 to 7 hours on modern computers. We do save interim data files that allow one to bypass the time-consuming steps, but the Harvard DataVerse has some cap on the size of archived files that we then violate.