

# Read-me file for replication package to: “Save, Spend or Give? A Model of Housing, Family Insurance, and Savings in Old Age”

Review of Economic Studies  
Daniel Barczyk, Sean Fahle & Matthias Kredler  
June 2022

## 1 Stata code for empirical work

### 1.1 Overview

Results generated by Stata appear in Sections 2, 5, and 6 of the article as well as in Appendices A, G, H, I, and K. All files related to the Stata analyses can be found in the **Stata** directory of the replication package. Within this directory, all do-files can be found in the **Code** directory, all data inputs in the **Inputs** directory, and all outputs in the **Output** directory, which includes sub-directories for data outputs (**.dta** files in the **data** directory), tables (**.tex** files in the **tabs** directory), figures (**.eps** files in the **figs** directory) and log files (**.txt.** files in the **logs** directory). The **MATLAB Utility** directory contains **.m** files that are used to convert MATLAB output to Stata inputs.

### 1.2 Data citation and data availability statement

TO BE COMPLETED ON PUBLICATION OF THIS REPLICATION PACKAGE. For now, we have included the raw HRS, RAND files etc. as inputs so that the REStud Data Editor can replicate our results easily. However, the HRS website tells us we should not publish the HRS/RAND data on Zenodo once the article is published. Our plan is to replace the raw data that are property of the HRS/RAND by instructions for the user on how to request and download them. We also want to make some intermediate files that we create publicly available in repository that the HRS has created for precisely this purpose so that replicators can spare these tedious initial steps if they so wish.

### 1.3 Preparing MATLAB results for Stata analyses

In order to read the model results into Stata, the MATLAB results must be exported to a Stata-readable format. The main MATLAB results are stored in the **res** struct. This struct is saved as part of a matrix file (e.g., **resBaseline.mat**). The user can export the results in the **res** struct to text files using the MATLAB script **PrepareResultsForStata.m**. The user need only edit the variables in the script containing the file path of results matrix (**ResultsFile**) and two destination file paths for the exported data: one for cross-sectional data (**CrossXOutFile**) and another for panel data (**PanelOutFile**). The script executes another function (**WriteResultsToTxt.m**) that generates the text files.

The MATLAB data are required by Stata in order to produce Figures 8, 10, 11, and 12 in the article. Figures 8, 10, 11(a), and 12 use data from the baseline model results. Figure 11(b) uses data from the no-child counterfactual results. Once created, the text files should be placed in the directory **Inputs\_from.Model** in the **Inputs** directory. The names of the text files are hardcoded in the Stata do-files that process these inputs: **Model\_Get\_Baseline\_Sample.do** and **Model\_Get\_NoChild\_Sample.do**.

### 1.4 Steps required to reproduce the Stata output used in the article

To reproduce the Stata data inputs and analyses in the article, execute the following steps.

1. Verify that all of the file and directory paths in **GetDirNames.do** are correct. The paths in the replication package are all defined relative to the location of the do-files. Unless the directory structure changes, no edits to these paths should be necessary.

2. To generate the data sets used in the analyses, run `_Make_Datasets.do`. A small number of tables and graphs are also produced in this step.
3. To produce the tables and figures, and statistics cited in the text (the subset of these that are generated by Stata), run `_Make_Tables_and_Figures.do`.

For the graphs to render exactly as in the article, the Stata scheme file `scheme-hack.scheme` must be included in the do-file directory. To determine which particular Stata programs and output files are associated with each table and figure in the article, see Tables 1, 2, 3, and 4 in this document. For statistics that are cited in the text of the article or appendices but which are not associated with any particular table or figure, see Table 5.

**Note:** Not all tables and figures that utilize Stata-generated data inputs are produced by Stata. The portions of Tables 3, 4(a), 4(c), 6, 7, and 8 in the text and Tables A1, A2(a), A2(b), and G3 in the appendix that pertain to the data (as opposed to the model) are assembled into `.tex` files manually from statistics generated within Stata. Figures 7 and 9 in the text are produced by MATLAB after manually copying Stata results into the MATLAB code. In each of these cases, all of the necessary statistics appear in `.txt` files located in the `logs` sub-directory within the `Output` sub-directory of the `Stata` directory.

## 1.5 Detail on the specific operations performed by each Stata file

### 1.5.1 Assembling the data sets

The file `_Make_Datasets.do` performs the following tasks (note that the order in which these tasks are executed is important):

1. Creates a data file (`CPIdeflator.dta`) with CPI deflators for each wave which are required in many of the following steps. This is done by executing the following file:

- `GetCPIdeflator.do`

2. Creates a data extract for long-term care helpers (`helpers_with_imputations.dta`). Note that this step must be done *before* creating the decedent and core samples in steps 3 and 4 in this section. The extract is created by executing the following file:

- `HelperData_Run_All_Files.do`.

This file, in turn, executes the following files (in order):

- `HelperData_Get_Exit_ADLs.do`
- `HelperData_Get_Helper_Counts.do`
- `HelperData_Get_CoreData.do`
- `HelperData_Get_Add_Core_Waves.do`
- `HelperData_Get_ExitData.do`
- `HelperData_CombineData.do`
- `HelperData_ImputeHours.do`

3. Creates the single decedent sample (`decedent_sample_single_final.dta`). This is done by executing the following three files (in order):

- `Decedents_GetSample.do`
- `Decedents_Sample_Supplement.do`
- `Decedents_ProcessData.do`

In addition to preparing the final data extract for single decedents, `Decedents_ProcessData.do` also creates tables and figures for the appendix. To identify the specific tables and figures, refer to the Tables 3 and 4 in this document.

4. Creates the core interview sample (`core_sample.dta`). This is done by executing the following file:
  - `CoreIWs_GetSample.do`
5. Creates a dataset with data on inter-generational transfers (`transfers_sample.dta`). This task must be done *after* generating the decedent sample in step 3 because it requires the bequest data produced in that step. The task is performed by executing the following file:
  - `Transfers_GetSample.do`
6. Creates data extracts from the model output for the baseline and no-child counterfactual by running the following files (order matters insofar as, in each pair of files, the file ending in `_Sample` must run before the file ending in `_Extracts`):
  - `Model_Get_Baseline_Sample.do`
  - `Model_Get_Baseline_Trajectories_Extracts.do`
  - `Model_Get_NoChild_Sample.do`
  - `Model_Get_NoChild_Trajectories_Extracts.do`

The extracts containing the model output for the baseline and no-child counterfactual are `baseline-merged.dta` and `nochild-merged.dta`, respectively. Four additional `.dta` files with names beginning with `model_traj_data` are also created. These files contain end-of-life wealth trajectories for certain groups of decedents.

### 1.5.2 Performing the analyses

Running the file `_Make_Tables_and_Figures.do` executes the following operations:

1. Generates the results pertaining to the decedent sample by running the following files (order does not matter):
  - `Decedents_Analysis_Regressions.do`
  - `Decedents_Analysis_Tables.do`
  - `Decedents_Analysis_Trajectories.do`
2. Generates the results pertaining to the core interview sample by running the following files (order does not matter):
  - `CoreIWs_Analysis_HomeLiquidation.do`
  - `CoreIWs_Analysis_CalibrationTargets.do`
  - `CoreIWs_Analysis_Regressions.do`
  - `CoreIWs_Analysis_Tables.do`
3. Generates the results pertaining inter-generational transfers by running the file:
  - `Transfers_Analysis.do`
4. Constructs the histogram of dollars transferred by age using the model results by running the file:
  - `Model_Analysis_Transfers_Histogram.do`
5. Produces the comparison of wealth trajectories between the Health and Retirement Study and the Survey of Consumer Finances in Appendix H by running the files:
  - `SCF_HRS_Trajectories_Age50.do`
  - `SCF_HRS_Trajectories_Age65.do`

## 1.6 Hardware, software, and runtime

Stata analyses were performed using Stata/SE 14.2 for Mac (64-bit Intel) Revision 29 Jan 2018 on a Mac-BookPro Model 15.2 with 2.7 GHz Quad-Core Intel Core i7 processor. Assembly of the datasets takes about 7 minutes. Performing the analyses takes about 5 minutes.

## 1.7 User-written Stata functions

The Stata do-files use several user-written Stata functions. Examples include `renvars`, `carryforward`, `fsum`, and `mipolate`. If the user encounters an error such as `command ... is not recognized`, the likely explanation is that the user has not yet downloaded one or more of these functions. To resolve the problem, use the `search` or `net search` commands in Stata with the name of the problematic command (e.g., `search renvars`) to identify the Stata package containing the function. The function can then be installed either by clicking the links that come up in the search or by typing `ssc install` followed by the package name.

## 2 Matlab code for illustrative model

### 2.1 Overview

The Matlab codes that solve the illustrative models (Section 3 and Appendix B in the paper) can be found in the folder `MatlabIllustrModel` of the replication package. There are two main scripts that generate the relevant figures and tables. Outputs are given in the `Output` directory, which includes sub-directories for tables (`.tex` files in the `tabs` directory) and figures (one `.eps` file and one `.fig` file for each figure in the `figs` directory).

- `ToyModelCommDevice.m`: Solves the illustrative model (=toy model) from Section 3 (An illustrative model: Housing as a commitment device). Running this script under the default parameters produces the following tables and figures, depending on the option `plotLRplanner`:
  - `plotLRplanner=false`: FIGURE 6 (Payoffs(=Hamiltonians) from  $dt$ -allocations at  $a_t^p = 1$  in basic illustrative model)
  - `plotLRplanner=true`: FIGURE B.1 (Planner's solutions in illustrative model) and TABLE B.1 (Allocations in basic illustrative model)
- `ToyModelCommDeviceLTC.m`: Solves the illustrative model with long-term care (LTC) described in Appendix B.2 (Illustrative model with LTC). This code adds LTC to `ToyModelCommDevice.m`. Running this script under the default parameters produces the following tables and figures, depending on the option `plotLRplanner`:
  - `plotLRplanner=false`: FIGURE B.2 (Payoffs from  $dt$ -allocations involving IC ( $a_t^p = 1$ , illustrative model with LTC))
  - `plotLRplanner=true`: FIGURE B.3 (Planner's solution in illustrative model with LTC)

Each script also contains several functions at its end, which are called only by the respective script. The Matlab files provide extensive further comments.

### 2.2 Hardware, software and runtime

To obtain the results reported in the paper, we used Matlab (R2021b, 9.11.0.1873467, 64bit maci64) using the Mac OS Big Sur (version 11.6.2) operating system on an iMac Pro (2017) with a 3.2 GHz 8-Core Intel Xeon W. Solving each model takes about 1 second. No additional toolboxes are required; the code is compatible with previous Matlab versions (according to Matlab's code compatibility report as of June 2022).

## 3 Matlab code for quantitative model (Sections 4-7)

### 3.1 Overview

The Matlab codes used for the solution of the quantitative model (see Sections 4-7 in paper) are provided in the sub-directory `MatlabQuantModel`. The Matlab files contain extensive comments on what we are doing at each step, so we only give a brief overview here. The *main program* to start with is `SSG_MotherShip.m`, in which the parameters of the model (structure `par`) and the options for the algorithm (structure `opt`) are set. This main program calls the function `SSG_SolveModel.m`, which solves the model for a given parameter vector, generates an artificial panel of model households, and returns the results in the structure `res`. Once the model is solved, the main program calls the function `SSG_GetMoments.m` (calculates moments based on the artificial panel generated by the model and displays many results in tables and graphs). The programs in the sub-folder `HelperFct` are helper functions called by the main programs. Outputs are given in the sub-folder `Output`, which includes sub-directories for tables (`.tex` files in the `tabs` directory) and figures (one `.eps` file and one `.fig` file for each figure in the `figs` directory). Below, we list in capitals (e.g. "TABLE 3") outputs that were generated in Matlab and in plain letters (e.g. "Figure 8") outputs that are ultimately generated in Stata (using a Matlab input).

### 3.2 How to replicate figures and tables

The following tables and figures (and others that are not in the paper) are displayed when calculating the baseline model (set `mode = 'baseline'` in `SSG_MotherShip.m` and run `SSG_GetMoments.m` subsequently):

- **Section 5:**

TABLE 3 (Calibration)

- **Section 6:**

Tables: TABLE 4 (Net worth distribution at the start of retirement), TABLE 5 (Net worth distributions by homeownership at the start of retirement), TABLE 6 (LTC arrangements), TABLE 7 (Bequest distribution), and TABLE 8 (Bequest distribution by asset class)

Figures: FIGURE 7 (Net-worth trajectories: All), Figure 8 (Net-worth trajectories: Own vs. Rent), FIGURE 9 (Homeownership in retirement), and Figure 10 (Net-worth trajectories: Community vs. nursing home)

- Note: Figures 8 and 10 in the paper version are produced in STATA using the data generated in Matlab together with the empirical data. Figure 12 (Timing of inter-generational transfers) does not appear when running the Matlab code; it is only generated using STATA.

- **Section 7:**

TABLE 10 (Effects of owning at age 65 on expected future outcomes), TABLE 11 (Behavioral effects of home-owning), and TABLE 12 (Value of housing technology, measure by dynasty wealth equivalent variation (WEV) at age 65)

- **Appendix D:**

TABLE D.1 (Earnings process fit)

- **Appendix F:** TABLE F.1 (Bequest distribution by asset class ( $\xi = 0.90$ )), TABLE F.2 (Net worth distribution at the start of retirement ( $\xi = 0.90$ )), TABLE F.3 (Net worth distributions by homeownership at the start of retirement ( $\xi = 0.90$ )), TABLE F.4 (Value of housing technology, measure by dynasty wealth equivalent variation (WEV) at age 65 ( $\xi = 0.90$ )), and TABLE F.5 (Counterfactual exercises for bargaining power).

Note: Tables F.1-F.4 are calculated in baseline mode when setting  $\xi = 0.90$  by hand.

Note: Table F.5 (Counterfactual exercises for bargaining power) is calculated in baseline mode when setting `bargainMode = NAME OF BARGAINING PROTOCOL` and `bargainModeRent = NAME OF BARGAINING PROTOCOL` where the name is ‘parent’ or ‘kid’ in `SSG_MotherShip.m`.

To calculate figures and table entries of counterfactual scenarios run `SSG_MotherShip.m` when setting `mode = ‘counterf’` and `CountF1 = SCENARIO NAME` (the name is given and described in the program, e.g. ‘noKid’ for the no child economy) and run `SSG_GetMoments.m` subsequently. Repeat this sequentially for each counterfactual scenario that is included in a table:

- **Section 6:** Figure 11 (Net-worth trajectories: Parents vs. childless). To generate the model-based portion of this figure, baseline results from `SSG_GetMoments` must first be saved as `momBaseline.mat`. Then set `CountF1 = noKid` and run the code.

Note: Figure 11 in the paper version is produced in STATA using the data generated in Matlab together with the empirical data.

- **Section 7:** TABLE 12 (Value of housing technology, measure by dynasty wealth equivalent variation (WEV) at age 65) and TABLE 13 (Counterfactual bequest distributions)
- **Appendix E:** TABLE E.1 (Wealth distributions ages 65 to 69) and TABLE E.2 (Bequest distributions)

FIGURE E.1 (Counterfactuals: homeownership rate by age) is generated by function `PlotCtflOwnRate` contained in the folder `HelperFct` after the relevant counterfactual scenarios are calculated and the structure resulting from `SSG_GetMoments.m` (e.g. `momBaseline.mat`) is saved.

Finally, there is a separate script `WEVillustration.m` that creates the purely illustrative figures that explain the concept of wealth equivalent variation in Appendix C.4. No input from the model solution is used here. The figures created are FIGURE C.1 (WEV: regular case) AND FIGURE C.2 (WEV: corner solution)

### 3.3 Hardware, software and runtime

To obtain the results reported in the paper, we used Matlab (R2021b, 9.11.0.1873467, 64bit maci64) using the Mac OS Big Sur (version 11.6.2) operating system on an iMac Pro (2017) with a 3.2 GHz 8-Core Intel Xeon W. Solving the baseline model under this configuration takes about 17 minutes. No additional toolboxes are required; the code is compatible with previous Matlab versions (according to Matlab’s code compatibility report as of June 2022).

Table 1: Tables in the text with data inputs generated by Stata

Table	Source file	Created by
Table 1	TAB1.tex	CoreIWs_Analysis_Tables.do
Table 2 (a)	TAB2a.tex	CoreIWs_Analysis_Tables.do
Table 2 (b)	TAB2b.tex	Decedents_Analysis_Tables.do
Table 3	CALIBRATION_TARGETS_AND_TAB6.txt	CoreIWs_Analysis_CalibrationTargets.do
Table 4 (a)	TAB4a.txt	CoreIWs_Analysis_Tables.do
Table 4 (c)	TAB4c.txt	CoreIWs_Analysis_Tables.do
Table 5 (a)	TAB5a.tex	CoreIWs_Analysis_Tables.do
Table 6	CALIBRATION_TARGETS_AND_TAB6.txt	CoreIWs_Analysis_CalibrationTargets.do
Table 7	TAB7_AND_TAB8.txt	Decedents_Analysis_Tables.do
Table 8	TAB7_AND_TAB8.txt	Decedents_Analysis_Tables.do
Table 9	TAB9.tex	Decedents_Analysis_Regressions.do

Table 2: Figures in the text with data inputs generated by Stata

Figure	Source file	Created by
Figure 1 (a)	FIG1a.eps	Decedents_Analysis_Trajectories.do
Figure 1 (b)	FIG1b.eps	Decedents_Analysis_Trajectories.do
Figure 2 (a)	FIG2a.eps	Decedents_Analysis_Trajectories.do
Figure 2 (b)	FIG2b.eps	Decedents_Analysis_Trajectories.do
Figure 3	FIG3.eps	CoreIWs_Analysis_HomeLiquidation.do
Figure 4 (a)	FIG4a.eps	Decedents_Analysis_Trajectories.do
Figure 4 (b)	FIG4b.eps	Decedents_Analysis_Trajectories.do
Figure 5	FIG5_and_FIG12a.eps	Transfers_Analysis.do
Figure 7	FIG7_DATA.txt Data in text file manually entered into MATLAB to produce Figure 7	Decedents_Analysis_Trajectories.do
Figure 8 (a)	FIG8a.eps	Decedents_Analysis_Trajectories.do
Figure 8 (b)	FIG8b.eps	Decedents_Analysis_Trajectories.do
Figure 9 (left)	FIG9_DATA_LEFT_PANEL.txt	CoreIWs_Analysis_Tables.do
Figure 9 (right)	FIG9_DATA_RIGHT_PANELS.txt Data in text files manually entered into MATLAB to produce Figure 9	CoreIWs_Analysis_Tables.do
Figure 10 (a)	FIG10a.eps	Decedents_Analysis_Trajectories.do
Figure 10 (b)	FIG10b.eps	Decedents_Analysis_Trajectories.do
Figure 11 (a)	FIG11a.eps	Decedents_Analysis_Trajectories.do
Figure 11 (b)	FIG11b.eps	Decedents_Analysis_Trajectories.do
Figure 12 (a)	FIG5_and_FIG12a.eps	Transfers_Analysis.do
Figure 12 (b)	FIG12b.eps	Model_Analysis_Transfers_Histogram.do



Table 3: Tables in the appendices with data inputs generated by Stata

Table	Source file	Created by
Table A.1	TABA1_SAMPLE_COUNTS_1.txt TABA1_SAMPLE_COUNTS_2.txt	CoreIWs_GetSample.do CoreIWs_Analysis_Regressions.do
Table A.2 (a)	TABA2a_SAMPLE_COUNTS_1.txt	Decedents_GetSample.do
Table A.2 (a)	TABA2a_SAMPLE_COUNTS_2.txt	Decedents_ProcessData.do
Table A.2 (b)	TABA2b_SAMPLE_COUNTS.txt	Decedents_Analysis_Trajectories.do
Table A.3	TABA3.tex	CoreIWs_Analysis_Tables.do
Table A.4	TABA4.tex	Decedents_Analysis_Tables.do
Table A.5	TABA5.tex	Decedents_Analysis_Tables.do
Table G.1	TABG1.tex	Decedents_ProcessData.do
Table G.2	TABG2.tex	Decedents_ProcessData.do
Table G.3	IVT_BEQUEST_RATIOS.txt	Transfers_Analysis.do
Table K.1	TABK1.tex	CoreIWs_Analysis_Regressions.do
Table K.2	TABK2.tex	CoreIWs_Analysis_Regressions.do
Table K.3 (a)	TABK3a.tex	CoreIWs_Analysis_Regressions.do
Table K.3 (b)	TABK3b.tex	CoreIWs_Analysis_Regressions.do
Table K.3 (c)	TABK3c.tex	CoreIWs_Analysis_Regressions.do
Table K.4	TABK4.tex	Decedents_Analysis_Regressions.do
Table K.5	TABK5.tex	Decedents_Analysis_Regressions.do
Table K.6	TABK6.tex	Decedents_Analysis_Regressions.do
Table K.7	TABK7.tex	Decedents_Analysis_Regressions.do
Table K.8	TABK8.tex	Decedents_Analysis_Regressions.do
Table K.9	TABK9.tex	Decedents_Analysis_Regressions.do
Table K.10	TABK10.tex	Decedents_Analysis_Regressions.do

Table 4: Figures in the appendices with data inputs generated by Stata

Figure	Source file	Created by
Figure G.1 (a)	FIGG1a.eps	Transfers_Analysis.do
Figure G.1 (b)	FIGG1b.eps	Transfers_Analysis.do
Figure G.1 (c)	FIGG1c.eps	Transfers_Analysis.do
Figure G.1 (d)	FIGG1d.eps	Transfers_Analysis.do
Figure H.1 (a)	FIGH1a.eps	Decedents_Analysis_Trajectories.do
Figure H.1 (b)	FIGH1b.eps	Decedents_Analysis_Trajectories.do
Figure H.1 (c)	FIGH1c.eps	Decedents_Analysis_Trajectories.do
Figure H.2 (a)	FIGH2a.eps	SCF_HRS_Trajectories_Age50.do
Figure H.2 (b)	FIGH2b.eps	SCF_HRS_Trajectories_Age65.do
Figure I.1 (a)	FIGI1a.eps	Decedents_ProcessData.do
Figure I.1 (b)	FIGI1b.eps	Decedents_ProcessData.do
Figure I.1 (c)	FIGI1c.eps	Decedents_ProcessData.do
Figure K.1 (a)	FIGK1a.eps	Decedents_Analysis_Trajectories.do
Figure K.1 (b)	FIGK1b.eps	Decedents_Analysis_Trajectories.do
Figure K.1 (c)	FIGK1c.eps	Decedents_Analysis_Trajectories.do
Figure K.1 (d)	FIGK1d.eps	Decedents_Analysis_Trajectories.do
Figure K.1 (e)	FIGK1e.eps	Decedents_Analysis_Trajectories.do
Figure K.1 (f)	FIGK1f.eps	Decedents_Analysis_Trajectories.do

Table 5: Statistics generated by Stata that are referenced in the text and appendices

Section	Source file	Created by
Section 2.2	HOME_LIQUIDATION_STATISTICS.txt	CoreIWs_Analysis_HomeLiquidation.do
Section 2.3	IVT_BEQUEST_RATIOS.txt	Transfers_Analysis.do
Section 6.3	AGE_AVERAGE_DOLLAR_GIVEN	Transfers_Analysis.do
Appendix H	APPENDIX_H_STATISTICS.txt	Decedents_Analysis_Trajectories.do
Appendix I	POWER_LAW_RESULTS.txt	Decedents_ProcessData.do