USER'S GUIDE FOR the STATA AND FORTRAN CODES

The files in this folder allow one to replicate the empirical results reported in the paper "On the Persistence of Income Shocks over the Life Cycle: Evidence, Theory, and Implications" by Fatih Karahan and Serdar Ozkan in RED (2012). In addition to the instructions here, each Stata and Fortran program contains comments that explain the purpose of each piece of code.

1 Estimation of the Various Income Processes in the Paper

Following the steps below allows one to replicate the estimation of the income process (both the age-dependent and age-invariant specifications) on the benchmark sample. We also explain how to do the estimation using different samples (for example, using hourly wage data, using different sample selection criteria, etc.) as well as how to use the code to estimate other income processes reported in the Appendix (for example, an income process where the transitory component is modeled as MA(1), one in which we take an external estimate for measurement error, an age-dependent income process with some age-invariant parameters, etc.).

Step 1: Sample Selection

To see the codes referred to in this section, please refer to the "Data" folder. The Stata code "cleaning_age_all.do" takes as input the Stata data files named "ready_newdata.dta" and "mergedpsid.dta", and creates 6 text files containing the data which are used an input to the Fortran code that does the estimation. These files are "idno.txt", "year.txt", "expr.txt", "residual.txt", "wageresidual.txt", and "sample_size.txt" and they contain the ids, ages, residual earnings, residual hourly wages of individuals, and number of observations in the sample, respectively.

Details of the Stata Code: There are two input data files.

1. The Stata dataset "ready_newdata.dta" is provided by Fatih Guvenen and is the same data file used in Guvenen (RED, 2009). It contains variables from the Panel Study of Income Dynamics (PSID) downloaded from PSID's website. Individuals in the SEO (poverty sample) and the Latino subsample are excluded when downloading the data. No other sample selection has been done.

2. The file "mergedpsid.dta" is downloaded from PSID's website and contains information on the employment status of individuals. This information is only used for the results in Appendix B5.

3. The file "cleaning_age_all.do" merges the two Stata datasets, cleans the data, selects the sample based on several options that we describe below, runs the first-stage regressions, and obtains the residuals. By using these options, you will be able to generate the different samples that we use in the paper. The options and their values to obtain the benchmark sample are as follows:

- Tinit: first wave of the PSID in the sample (benchmark: 68)
- Tlast: last wave of the PSID in the sample (benchmark: 97)
- ageinit: youngest individual in the sample (benchmark: 24)
- agelast: oldest individual in the sample (benchmark: 60)
- minyrs: minimum years of observation an individual should have in order to be included in the sample (benchmark: 3)

- fulltime: Setting this to 1 excludes part-time workers. Set this to 0 to include them. (benchmark: 0)
- cleanage: Setting this to 1 means that we use age in the first-stage regressions as opposed to potential experience. (benchmark: 1)
- consecutive: If set to 1, the code requires individuals in the sample to have at least "minyrs" years of *consecutive* observations. Set this to 0 to include people with "minyrs" of, not necessarily consecutive, observations. (benchmark: 0)
- cleanempst: Setting this to 1 includes only people in the labor force (i.e. employed, laid off, or unemployed) and excludes students, disabled, etc. (benchmark: 0)
- colonly: Setting this to 1 includes only those with a college degree. (benchmark: 0)
- hsonly: Setting this to 1 excludes those with a college degree. (benchmark: 0)

Step 2: Obtaining Point Estimates

To find the files we refer to below, please see the "Estimation" folder. We describe below how to estimate the various income processes in the paper. Note that these programs are written in Fortran and make use of the IMSL libraries.

Obtaining Point Estimates The main estimation can be done using codes in one of the three folders provided. The Fortran files in the "NonPar" folder are used to estimate the nonparametric specification in the paper. The code in the "Poly3" folder estimates the cubic specification. Finally, the code in the "3BIN" folder estimates the specification with three age intervals (see Section 2.4). All of these programs share a similar structure. To run the estimation, follow the steps below:

1. Include the text files generated by Stata under the same directory as the project.
2. You will need to specify an initial guess for the estimation. Either use the file "output_theta0.txt" (provided) or modify the values to start the estimation from a different initial guess.
3. Set the options using the parameters in the "PARAMETERS.F90" file. See below for details.
4. Compile and run the program (the main file is "Estimate.f90").
5. The point estimates will be recorded in "output_theta.txt".

Options for the Estimation The parameters that we describe below are contained in the "PARAMETERS.F90".

- constant_rho: Set this to 1 if you want to impose a constant persistence profile.
- constant_eta: Set this to 1 if you want to impose a constant profile for the variance of persistent shocks.
- constant_eps: Set this to 1 if you want to impose a constant profile for the variance of transitory shocks.
- bootstrap: Set this to 0 to obtain point estimates only. Otherwise, set it to 1.
- min_obs: This controls the minimum number of observations that should contribute to a moment so that the moment is included in the estimation.
- wage: Set this to 1 if you would like to run the estimation on residual wages as opposed to residual earnings.
- MA_estimate: Set this to 1 to add an MA(1) transitory component to the existing specification.
- rip: Set this to 1 to estimate the age-invariant income process. Note that this option only exists in the "3BIN" folder (age interval specification).

Reading Point Estimates The file "output_theta.txt" contains the point estimates stacked as a vector. The length of the vector varies across different programs (NonPar, Poly3, or 3BIN). The convention is that the first parameter is the variance of fixed effects. This is followed by the parameters related to the persistence profile. In the case of nonparametric estimation, this will be an array of persistence parameters for each age. For cubic estimation, this would contain the 4 parameters of the polynomial. Finally, in the case of age interval specification, this would contain the 3 parameters (persistence for the young, the middle-aged, and the old). Persistence related parameters are followed by parameters related to the variance of persistent shocks. Similar to the persistence parameters, the number of these parameters also varies across different specifications. Variance of persistent shocks are followed by the variance of transitory shocks, which are followed by the time loading factors for persistent and transitory shocks. The table below summarizes the structure of the output file for each specification:

Table 1: Structure of the Output File Across Specifications

Specification	Nonparametric	Cubic	Age Intervals	RIP
Variance of fixed effects	1	1	1	1
Persistence	37	4	3	3
Variance of Persistent Shocks	37	4	3	3
Variance of Transitory Shocks	37	4	3	3
Time Loading Factors (Persistent)	30	30	30	30
Time Loading Factors (Transitory)	30	30	30	30

Step 3: Obtaining Bootstrap Confidence Intervals

To run the bootstrap, you will need to use the Fortran codes provided in the "Bootstrap" folder (under the Estimation folder) as well as the code used to obtain point estimates. Essentially, the program in the "Bootstrap" folder takes the data files generated by Stata as an input, randomly selects a bootstrap sample for each repetition, computes the moments to be used in the estimation, and calls the relevant estimation program (nonparametric, cubic, or the age interval specification), which you choose by using the parameters in the "PARAMETERS.F90" file. Follow the steps below to run the bootstrap:

1. You need to provide an initial guess for each repetition of the bootstrap. Either provide a file with the initial values of your choice, or simply rename the output file from the estimation (output_theta.txt) as "output_theta0.txt" and place it under the folder containing the bootstrap code.

2. In the estimation program, set the "bootstrap" parameter to 1. Compile and link the code and place the resulting executable under the "Bootstrap" folder. The program in the "Bootstrap" folder will repetitively call this executable.

3. Set the options of the bootstrap program and run the code. Parameters are contained in the "PARAMETERS.F90" file (see below for details).

Below are the options for the bootstrap:

- NONPAR, CUBIC, 3BIN: To run a given specification, set the parameter for that specification to 1, and set the remaining parameters to zero. For example, to run the bootstrap for the nonparametric specification, set "NONPAR" to 1, and set "CUBIC" and "3BIN" to zero.
- NRUNS: Sets the number of bootstrap repetitions.
- MIN_OBS: Do not target moments to which less than min_obs people contribute.
- WAGE: Set this to 1 to use wage data.

Output The code creates 3 files containing the output. "theta_est.txt" contains the parameter estimates for all of the bootstrap repetitions. "theta_ci.txt" reports a 95% confidence interval for each parameter, and "theta_std.txt" contains the bootstrap standard errors.

2 Learning Model

The files to be used to replicate the results of Section 3 are contained under the folder "Learning". There are three steps to replicating these results. The Fortran codes in the folder "Model" solve the calibrated learning model of Section 3 and simulate data from it. One of the outputs of this program is the file "wages.csv". Use the Stata file "clean_data.do" (provided in the "Learning" folder) to run the first stage regressions and obtain residuals. Finally, the codes in "Learning_NonPar_Est" take as input these residuals and estimate the nonparametric age-dependent labor income process.

3 Consumption Model

To replicate the results in Section 4, you will need to use the Fortran codes under the "Consumption" folder. Simply set the parameters "income_process" and "tight" as needed and run the program. Set income_process to 0 to use the age-dependent income process (cubic specification). To use the age-invariant process, simply set income_process to 2. Based on this choice, the values of the income process parameters are set in "tauchen.f90". If you would like to change these values, simply edit "tauchen.f90". To compute the economy with natural (tight) borrowing constraints, set tight to 0 (1). The code produces the file "output.txt" as the output. This file contains welfare costs as well as insurance coefficients.