

Global Repository of Income Dynamics

Instructions for the STATA Code

Serdar Ozkan* Sergio Salgado†

December, 2023

1 General Directions

This document discusses the codes used to generate the sample and the statistics for the core section of the Global Repository of Income Dynamics (GRID). The code packet—available in GitHub at <https://github.com/salga010/QE-MasterCode>—contains eight main do-files that execute the initialization of the parameters, the sample creation, main statistics, and save the files to upload to the GRID website. The packet also contains one auxiliary file used for statistics (myprogs.do). The codes are written in Stata 16, designed to produce the statistics listed in the GRID Guidelines document, save the results in CSV files, and collect the necessary information that need to be submitted to the GRID team. The basics steps to run these codes are the following (with additional details in section 2).

1. Create in your local machine the following sub-folders (all in lower cases) under the same folder (i.e., the main folder):
 - do
 - dta
 - log
 - out

Next, download the provided do files in folder /do and copy the country-specific raw data file in folder /dta. The log files will be saved in the /log folder, and the results will be saved under /out.

2. **0_Initialize.do.** Open this do file and assign country-specific parameters such as the starting and ending years of the sample, the name and location of the raw dataset, the country's CPI, etc., for which further instructions are given in Section 2. Please, when you submit the results back to the GRID team, include the 0_Initialize.do as it contains important information used in the website.

*St. Louis Fed, University of Toronto; serdar.ozkan@utoronto.ca; www.serdarozkan.me

†The Wharton School-University of Pennsylvania; ssalgado@wharton.upenn.edu

3. **1_Gen_Base_Sample.do.** This do file renames the variables, does very basic sample selection (described in the GRID Guidelines), creates new variables (e.g., log, residual, growth rate of earnings), and generates the master_sample.dta, which is a wide-form dataset which will be used in the rest of the do files. The main output of this do file (master_sample.dta) is saved in the folder /dta and contains, among others, the following variables:
 - (a) personid: id of the individual used throughout the do files
 - (b) male: indicator variable equal to 1 if male and 0 if female
 - (c) yob: year of birth of the individual
 - (d) yod: year of death of the individual
 - (e) educ: indicator variable with education categories
 - (f) labor: real labor earnings in levels
 - (g) logearn: real labor earnings in log levels
 - (h) permearn: permanent income defined as $P_{it-1} = \frac{\sum_{s=t-3}^{t-1} y_{i,s}}{3}$ as explained in the Guidelines
 - (i) permearnalt: alternative measure of permanent income. See the Guidelines, section “Key statistics 4: Mobility” for additional details
 - (j) reearn: residual log earnings
 - (k) reearn1F: 1-year forward residualized log earnings change, g_{it}
 - (l) reearn5F: 5-year forward residualized log earnings change, g_{it}^5
4. **2_DescriptiveStats.do.** This do file generates a folder under /out, whose name consists of the date the program is run and “Descriptive_Stat”. This folder will contain a set of .csv files with the statistics for the section “Key statistics 1: Descriptive statistics” described in the Guidelines of the project.
5. **3_Inequality.do.** This do file generates the moments described under the section “Key statistics 2: Inequality and Concentration” in the Guidelines document. These moments will be saved under the corresponding folder under /out.
6. **4_Volatility.do.** This do file generates a set of .csv files with the statistics for the section “Key statistics 3: Volatility and Higher-Order Moments”.
7. **5_Mobility.do.** This do file generates a set of .csv files with the statistics for the section “Key statistics 4: Mobility”.
8. **6_Insheeting_datasets.do.** Open this do file Specify the directory of the main folder in your local machine, the directories where the different results are saved (Inequality, Mobility, etc.) and where the results for GRID will be saved. This file should be used *after* all other files have finished.

In the next section, we provide some additional details for the 0_Initialize.do and 6_Insheeting_datasets.do files. All programs are heavily commented and we have made the best of our efforts to make them run

bug-free. If you find any problem, please let us know on the Slack channel, so we can update the code and share the information with the rest of the teams. Furthermore, there may be situations where some changes on the code will be necessary due to the idiosyncratic features of each country's dataset or you may need further instructions. To facilitate smooth communication and collaboration between us and the teams, we have created a *Slack* channel.¹ Send Salgado your email so he can add you to the workplace. Please make sure that at least one of the members of your team joins this messaging group for further communication.

2 The Initialize Do file

The 0_Initialize.do defines the variable names, time span, and vectors used throughout the codes and allows each team to select some options that best suit their dataset. It also defined whether the team wants to run all of just some of the sub do files. Given its importance, here we discuss several key details (more comments can be found in the do-file).

Lines 5 to 18 of 0_Initialize.do define general variables that must be followed by the teams to generate the core statistics. Hence, no change is required in this section. These definitions ensure that the sample used for the core section of the paper is comparable across countries.

Lines 20 to 100 require the input of each team. Please read in detail.

1. **maindir.** Define what is the main dir. This is the folder where you have created the other sub folders, saved the do files, and so on.
2. **Unix vs. Windows.** Define whether the machine you are running your codes is Unix/Mac (unix=1) or Windows (unix=0).²
3. **Wide vs. Long Format.** Define whether the raw sample is in wide form (wide =1) or long form (wide=0). If it is in long form, the 1_Gen_Base_Sample.do file will convert it to wide form (one row per individual) when creating the dataset master_sample.dta. The rest of the codes are designed to work with this .dta.
 - (a) By long format, we mean a dataset in which each observation (row) is an individual-year pair. In other words, workers' observations are stacked, there is one column that defines the unit of time (year) and one column for each variable defining the value of each variable within the year (one column for earnings, one for education, etc.).
 - (b) By wide format, we mean a dataset in which each observation (row) is an individual and different columns define different observations for the same individual. In other words, workers' observations are side-by-side, and there is one column per year defining each variable (one column is the earnings in 2000, a second column is the earnings in 2001, and so on).

¹For more information on this app, visit <https://slack.com/features>.

²Although STATA run on Windows machines corrects the folder separators, just to be on the safe side, we specify whether the separator is "/" or "\", which will then be used to locate the sub-folders.

4. **Missing values for labor income.** If there are genuine missing values for labor income please set global `{miss_earn}` to 1 (lines 33 to 36). If it is set to zero (the default), the code will convert all missing earnings observations to zero. This is particularly important if your raw dataset is in long form and there are no observations for zero labor income in a given year.
5. **What codes you want to run.** Besides `1_Gen_Base_Sample.do`, the rest of the codes are modular and can be ran after the sample has been created. You can change what codes you want to run between lines 43 to 48. As a suggestion, you can turn all the globals to 0 with the exception of `1_Gen_Base_Sample.do` as it is the most time-consuming code. After that is completed, turn on the rest of the globals.
6. **Variable Names.** Specify the names of the variables in your data set between lines 41 and 48. These variables are the minimum set necessary to generate all the statistics in the Guidelines, hence, each team must make sure the raw data contains these variables. The `1_Gen_Base_Sample.do` file then will rename these variable to our choices in the `master_sample.dta`. This helps to simplify the code in the rest of the do-files.
7. **Variable Types.** The do files are written under certain assumptions about the type of variables available in each dataset. We did not attempt to change the format of the variables, hence, each team must make sure that the raw data contains the correct format (i.e. education must be a numerical categorical integer variable, gender must be binary, etc.). Here we describe in detail the variables used in the analysis
 - (a) `{personid_var}`: Numerical categorical variable. Teams must make sure an individual id appears only one time per year in the sample.
 - (b) `{male_var}`: Numerical categorical variables which is equal to 1 if the individual is male, 0 if female.
 - (c) `{yob_var}`: Numerical categorical variable that defines the year of birth of an individual. Teams must make sure this is not missing or changes across different observations of the same individual (if the raw data is in long form). Individuals with missing age will be dropped from the sample.
 - (d) `{yod_var}`: Numerical categorical variable that defines the year of death of an individual. Teams must make sure this does not change across different observations of the same individual. Individuals with missing `{yod_var}` will be treated as they where still alive by the end of the sample.
 - (e) `{educ_var}`: Numerical categorical variable that defines the education group of an individual. This can change across different observations of an individual. There is no restriction on the number of categories this might contain. The GID Guidelines, however, defines certain commonly use groups.³

³Some datasets do not contain information on education. Instead of modifying the code, we would recommend to create a place holder of such variable which has at least two distinct values. This will not change any of the results but will allow the code to run smoothly.

- (f) `#{labor_var}`: Numerical variable that defines the labor earnings of an individual. This variable might contain missing values. Recall that you also need to choose whether the missing observations are set to 0 by setting global `#{miss_earn}` to 1 or 0 in line 36.
 - (g) `#{year_var}`: Numerical variable that defines the year of the observation if the raw data is in long form.
 - (h) `#{educ_typ}`: some countries have education as a string variable. Use this global to define what is the format.⁴
 - (i) `#{iso}`: global used to define what is the name of the csv file that need to be sent to the GRID team. Please use ISO codes (e.g. for Norway use NOR, for Spain use ESP, and so on).
 - (j) `#{minnumberobs}`: due to disclosure requirements, some countries require a minimum number of observations used in any statistics. This global defines that number are replaces by a missing any statistics that was calculated using a sub sample that does not pass this minimum value.
8. **First and last year.** Specify the first and last year of the sample for which the statistics will be calculated. The sample is assumed to have no gaps in between (all years between `#{yrfirst}` and `#{yrlast}` are available). If that is not the case and your sample contains gaps, please contact us.⁵
 9. **Density estimation.** Global `#{kyear}` defines for which years the densities will be calculated. By default, the code calculated the densities in years ending with 0 and 5 (i.e., 1995, 2000, 2005, etc.). In case you want to calculate densities every year, change `#{kyear} = 1`.
 10. **Quantile estimates.** Quantile estimates are mainly used in the 5_Mobility.do do file. See the code for additional details. The global `#{nquantiles}` defines how many quantiles will be used to divide the distribution of permanent income. The default is 40, as suggested in the Guidelines. The global `#{nquantilesalt}` does the same for the quantiles of the distribution of alternative permanent income. The global `#{nquantilestran}` defines the number of quantiles used in the transition matrices.
 11. **Heterogeneity groups.** The global `#{hetgroup}` specifies what heterogeneous characteristics are considered. By default the code follows the GID Guidelines, calculating the statistics by gender, education, age, and the cross groups. Additional levels of heterogeneity can be easily incorporated as long as the corresponding variables are passed to the sample.⁶ Contact us in case you have issues incorporating additional degrees of heterogeneity.
 12. **CPI, min income, and exchange rate.** The matrices `cpimat`, `rmininc`, and `exrate` contain the CPI, the min income threshold, and the exchange rate (nominal) that is used throughout the code. These need to be imputed from `#{yrfirst}` to `#{yrlast}` *without* gaps. All nominal variables must be deflated by 2018 prices. Hence, set the global `#{cpi2018}` equal to the corresponding value the CPI in 2018 for your country.

⁴If you country does not have education variable, you can create in the raw data a variable that is called `educ` and it is equal to a particular number (say “`gen educ = 1`”). In that way, you do not need to change anything in the codes.

⁵The code is flexible enough to deal with samples with gaps but will require some small changes. In the case your sample has gaps, please contact us.

⁶Check the code `myprogs.do` for details.

The rest of the code re defines the minimum income threshold as suggested in the Guidelines and several lists of years that will be used by other do-files for different calculations. Please do not change them. If you think you have a good reason to change them please Salgado/Ozkan on Slack.

3 Submitting your Results

After the codes 1 to 5 are finished and the data has been saved, you will need to run 6_Insheeting_datasets.do. This code requires a few definitions similar to those described in 0_Initialize.do. Importantly, you need to define the folder where the final results will be saved (line 37) and the location of the results generated by the rest of the do-files (lines 40 to 44). You will need to send back to the GRID team the files generated saved in the folder `${datafran}` and the Initialize do-file.