README for replication of Fairness gaps for earnings tax design

Erwin Ooghe, Erik Schokkaert, Hannes Serruys[1]

OVERVIEW

The estimation of the conditional logit and the simulation of the economy has all been written in Python, with exception of the wage estimation, which happens via an R script that is called from within the python code. The budget set of the individuals is simulated via EUROMOD.

In order to run the estimation + welfare simulations one needs:

- To have a python distribution installed (e.g. via Anaconda)
- The EUROMOD software installed[2]
- R software is installed
- Access to the micro data of both EU-SILC for Belgium year 2017 (income year 2016) and have access to the EUROMOD input data. [3] [4]

In order to run the estimation, one goes to the folder *pythoncode_cleaned*, where one, after installing the necessary software sets the necessary paths via *settings.py*. Although the code should work for any Python interpreter, we recommend using Spyder and its IPython interpreter to run the code.

[1] Any questions w.r.t the code can be directed to this author via hannes.serruys@gmail.com.
[2] See https://euromod-web.jrc.ec.europa.eu/download-euromod.
[3] For access to the EU-SILC micro-data, please visit https://ec.europa.eu/eurostat/web/microdata/access.
[4] For access to the EUROMOD input data see https://euromod-web.jrc.ec.europa.eu/download-euromod/data-request-form?version=all.

ESTIMATION

In order to run the estimation of the file one needs to run the file *estimation.py*. This file contains:

- the definition of the micro-econometric models [5]
- Functions that construct the input data passed to EUROMOD
- Cleaning of the data + selection of the subsample
- Construction of the utility functions, choice probability functions and log-likelihood specification
- The estimation of the model itself.

The utility function specification is a sympy [6] expression, which can be manipulated in order to test multiple utility specifications. It is passed as the ut_abstr object

The following tables can be generated by running this file:

- Table 1
- Table 2
- Table 3
- Table 4

Additionaly, using the *figure_grosswage.py file,* one obtains also Figure 1D.

The parameter estimates of Table 9 and Table 10 can be obtained by the following lines of code:

- *mod_log_r2.solution.to_tex(table9.tex,"SOME/OUTPUT/PATH")*
- *mod_ruro3_1. solution.to_tex(table10.tex,"SOME/OUTPUT/PATH")*

This should be run after running *estimation.py*.

WELFARE ANALYSIS

In order to perform the welfare analysis one first needs to run estimation.py such that we obtain estimated utility functions that then are passed to the initialiser methods of our Economy and EconomyRURO classes, which are both specified in the *Economy2.py* file.
Both classes represent economies, one where there is no rationing w.r.t job opportunities and one where there is rationing. Here we replicate every individual in our subsample 809 times and we generate preferences using the micro-econometric model estimated in the first step. Unobserved preference heterogeneity gets introduced by drawing random error terms out of the EV-1 distribution. For the RURO economy, the opportunity set of the individual is modelled explicitly as stochastic.

The two main files for running the welfare analysis are:

- *welfare_analysis_ruro.py* (welfare analysis with rationing),
- *welfare_analysis_voluntary.py* (welfare analysis without rationing).

These two files allow us to generate the following tables and figures:

- Table 5

[5] EuroModLSModel and RURO_simpe3 are the ones used by the final model, but some other iterations with other loglikelihood function specifications are included in the code

[6] Sympy is a powerful Python library for symbolic computation, which allows us to specify the utility function and derive the loglikelihood expressions automatically.

- Table 6
- Figure 4
- Figure 5

Table 7 can be generated via the *get_revenue* method of the Economy and EconomyRURO classes, and via the function *print_eb_table from* the welfare_analysis_ruro and welfare_analysis_voluntary.

In order to generate the figures 6 and 7 the following scripts need to be run (which are contained in the Figures/ruro and Figures/voluntary folders respectively:

- *equity_gain_ruro.py & equity_gain_voluntary.py* for figure 6
- *welfare_gain_ruro.py* & *welfare_gain_ruro.py* for figure 7.

Note that before running these scripts one need to have run the respective welfare_analysisf files, which initialise an Economy or EconomyRURO object. [7]

[7] These Economy and EconomyRURO objects itself have State objects which represent the uncertainty that we have w.r.t the preferences of the individual. In an earlier version of the paper an economy had multiple states and we would aggregate over this uncertainty. We removed this however out of the scope of our paper, but this is still present in the code. For the sake of the welfare analysis we just chose the first state.