Introduction

This directory contains data sets and MATLAB functions used in the paper Stochastic Monotonicity in Intergenerational Mobility Tables by V. Dardanoni, M. Fiorini, A. Forcina. The functions performs the following tasks

- computation of unconditional likelihood ratios and unconditional testing,
- parametric conditional maximum likelihood estimation and testing,
- non-parametric conditional testing,

Unconditional estimation and testing

The set of 6x6 social mobility tables for 149 settings (country x year) are contained in the file `Treiman data.pdf` with the 36 frequencies for each table organized by row in a single vector (social classes are in descending order). To compute the T01, T12 statistics for a chosen table, just copy and paste the 36 frequencies into row 3 of `luncon.m`, run and type `[t01 t12]`. The collection of the T12 statistics for the whole set have already been computed and are contained in `unconstrLR.m`; by running this function the values are read, the p-value computed and plotted against the likelihood ratio and the 5% and 1% limits marked with a dotted line.

Conditional estimation

The `lanciasc.m` function reads the individual data, selects all available covariates and responses, defines the marginal model described in the paper and fits the model under the monotonicity constraints. To compute the ML estimates under independence, just remove the comment on line 42. The value of the log-likelihood is contained in the scalar LL and the T_{12} statistics may be computed by LR=-2*(LLmonotonicity-LLunconstrained).

Monotonicity or independence within the first two or last two rows may be fitted by removing the comments on the corresponding sets of lines clearly marked from 39 to 53.

The maximization is performed with the `newmult.m` function; possible alternatives, in case of difficulties in convergence, are `fiscor.m` which is based on a Fischer'scoring algorithm and `macsil.m` which is the most stable but slowest.

Similar calculations for wage mobility are obtained by running the `lanciawg.m` function

Parameter estimates and covariates

The `model.m` defines the model and the order with which parameter estimates are arranged. In the present version, the first 8 parameter are, respectively, the two logits for the father's marginal, the two logits for the son's marginal and the four odds rations. The remaining parameters are regression coefficients for the covariates in the three components of the model. The father's marginal has only the age as covariate; the son's marginal has the full set of 11 covariates for each of the two logits

The names of variables and other details on the dataset are contained in `data structure.rtf`.

Conditional testing

Once the T_{01} T_{12} statistics have been computed, the weights for the chi-bar distribution may be obtained by the command `w = west(F,C,D,n);`, where F is the information matrix estimated under the independence model, C is the null matrix defined in `lanciasc.m`, D is the matrix of inequality constraints and n is the number of replications, possibly not less than 1000. Once the vector of weights has been computed, the p-values for testing monotonicity against independence and unconstrained model against monotonicity may be computed as `p=jocbar(t01,t12,w)`, this gives a 2 x 2 table where the p-value against independence is the total of the second row and the p-value against monotonicity is the total of the second column.
A similar procedure may be used to test partial monotonicity.

Non-parametric unconditional test

The function `aggrega.m` performs the following operations: (i) discretize covariates, (ii) construct 3x3 social mobility tables conditional on all possible configurations of discretized covariates, (iii) detects tables with 0 patterns which prevent ML estimation under monotonicity, (iv) searches for similar table into which to merge tables with too many 0s, (v) computes the t01, t12 statistics for each conditional table. In the end the total for the two tests across all available tables is computed and the overall information matrix F and overall matrix of constraints D are constructed. Probability weights for the overall test may be computed as `w = west(F,C,D,n);`, where F; D; C are the output of `aggrega.m`

Trouble shooting

If you have questions or experience problems, please e-mail to `forcina@stat.unipg.it` to contant `antonio.forcina` on Skype.