Readme for the "stata" folder for: "Automation and the Future of Work: Assessing the Role of Labor Flexibility"

Michele Fornino* Andrea Manera†

January 24, 2021

This Readme illustrates how we obtained the target statistics of interest described in the Calibration appendix of our paper. Following the steps outlined below will produce the files needed as an input for the calibration exercise, specifically for the function "setParametersGE". The code was run using STATA 16 SE on a Macintosh machine running OS 11.1.

IMPORTANT: This replication kit assumes that the folder "dataRaw" contains the file "Compustatquery0726.dta", which contains proprietary Compustat data that cannot be shared. The reader is encouraged to get in touch with the authors to explore possible solutions to this issue, which may include running a new query as detailed in the data Appendix B of the main text, or sharing our data if possible. The reader can skip the particular step of computing the raw residuals from Compustat by commenting line 23 in the file runGenerateAllData.do.

1 Folder Structure and Master File

The main "Calibration" folder contains three subfolders which have all the datasets and codes needed to reproduce the data used for our calibration:

- "dataRaw" contains the data from the sources detailed in the appendix "Data Sources";
- "bin" contains the Stata do-files needed to build our statistics. The **master file** is "runGenerateAllData.do" which executes all the other chunks of code contained in the subfolder;
- "out" collects the output STATA datasets and the files "OUStatistics.csv", "HamiltonEstimationSample.csv" and "RawResiduals.csv", which the user should move into the folder "matlab/data" to run the Matlab replication file "Run.m" .

An important note is in order. To correctly run the codes, users should modify the global variable "$masterpath" in the master file "runGenerateAllData.do". The "masterpath" folder should contain the subfolders listed above in order for the code to run correctly. All files can also be run as stand-alone by uncommenting the preamble and setting the "masterpath" as needed. A more detailed description of the subfolders and files follows.

2 "dataRaw"

See the next section for a description of the input files for the Stata do-files. Here we just mention the source of the robot purchase price to mean wages of production-line employees, p_R/w, used in our benchmark calibration.

*MIT. Email: mfornino@mit.edu
†MIT. Email: manera@mit.edu

The series on robot prices comes from the first figure in Korus (2019) ("Industrial Robot Cost Decline"). The data on mean wages of production-line employees in the years 1995, 1996, 2004, 2005, 2007, 2010, 2014, 2017 comes from the OES, and is contained in the files "national_1997_dl.xls" (we use 1997 OES code 80000 for both 1995 and 1996 since the OES classification does not allow a clear mapping in 1995 and there is no data for 1996); and in the respective subfolders of "BLSmeanwages". In each subfolder, there is a "xsls" file named according to the following pattern "national_MYYYY_dl". Upon opening, the file will show a highlighted line containing, in order, the wage in 2015 terms of all production occupations in manufacturing (column U), the price of robots in 2015 dollars from Korus (column V), and their ratio (column W). The wage in 2015 dollars is obtained using the series in "CPIAUCSL.csv" as a deflator.

3 "bin"

This folder contains the do-files needed to build our data. They are generously commented. The inputs, outputs and main steps of each files are as follows:

- "runGenerateAllData" executes all of the following files in the order specified below.
- "runThetaProd" computes the shares of production-line employees on value-added in the various IFR sectors. Inputs:
  - "CPIAUCSL.csv", the CPI series for all urban consumers from FRED;
  - "sic5811.dta", containing SIC-level data from the NBER-CES dataset;
  - "CrossWalkSICIFR.dta" is the crosswalk between SIC codes and IFR codes kindly provided by Daron Acemoglu and Pascual Restrepo, who used it in Acemoglu and Restrepo (2020).

  Outputs:
  - "ThetaProdTsIfr.dta" containing the full time series of labor shares (not used in our Matlab code);
  - "ThetaProdIFR.dta" containing the average share of production-line employees for each IFR sector before 1980, to use as a value for the DRS parameter θ.

  Summary of the procedure: the value added data from the NBER-CES is matched to the IFR code using the crosswalk. The parameter θ is obtained as the average of the trend component of the share of production-line employees on value added before 1980.
- "runThetaProd_all" computes the shares of production-line employees on value-added for the whole of manufacturing. The procedure and inputs are the same as before, except that now sectors are lumped together, regardless of which IFR sector they belong to. Output:
  - "ThetaProdIFR_all.dta" containing the average share of production-line employees for the manufacturing sector before 1980, to use as a value for the DRS parameter θ in the one-sector calibration exercise.
- "runProdLineEmp" computes the number of production-line employees in the various IFR sectors in 1989. This data is needed for the file "OUStatistics.csv" Inputs:
  - "BLSProdLineEmp/mf89d3.csv", containing the number of employees by OES occupation code in 3-digit SIC sectors in 1989;
  - "CrossWalkSICIFR.dta" is the crosswalk between SIC codes and IFR codes kindly provided by Daron Acemoglu and Pascual Restrepo, who used it in Acemoglu and Restrepo (2020);

– "dataRaw/BLSProdLineEmp.dta", intermediate output of the code containing the production line in the BLS matched to the IFR sectors.

Outputs:

– "ProdLineEmpIFR.dta" containing the unmber of employees in each IFR sector.

Summary of the procedure: the OES data is filtered to include only production-line employees ("Production occupations"), matched to IFR sectors and collapsed.

- "runCompustatResidIFR" generates the productivity residuals variables contained in "rawCompustatResidIFR.dta" Inputs:

  – "Compustatquery0726.dta", containing the WRDS query detailed in the section "Data Sources" of the "Calibration" appendix;

  – "FIXEDINVDEF.csv", the fixed investment deflator from FRED;

  – "CrossWalkSICIFR.dta" is the crosswalk between SIC codes and IFR codes kindly provided by Daron Acemoglu and Pascual Restrepo, who used it in Acemoglu and Restrepo (2020);

  – "apr_measures_ifr19.dta", containing the robot penetration measures from Acemoglu and Restrepo (2020).

  Outputs:

  – "out/dataCleanedCompustat.dta" , intermediate output containing the real net investment in property, plant and equipment;

  – "out/rawCompustatResidIFR.dta", containing the raw TFP residuals for each firm.

  Summary of the procedure: the series "ppegt" is used to obtain the first value of the capital stock registered for each firm, deflated by the fixed investment deflator. The series "ppent" (net investment in property, plant and equipment" is then interpolated to fill in missing years. A series for real net investment is then constructed using the changes in the interpolated "ppent" series deflated by the fixed investment deflator from FRED. These are then added to the initial value of capital to obtain the value for the capital stock in each year. We then compute the residuals we need by regressing log-sales on log-employment, log-capital stock (both interacted with the IFR sector), sector-by-year fixed effects and firm fixed effects.

- "outsheetResiduals.do"" produces the .csv file contained the residuals obtained from the previous step. Input:

  – "out/rawCompustatResidIFR.dta", containing the raw TFP residuals for each firm.

  Output:

  – "out/RawResiduals.csv" containing the series of raw Compustat TFP residuals, employed for the calibration with non-stationary shocks.

- "runHamiltonResiduals.do" makes the raw Compustat TFP series stationary, by applying the filter proposed by Hamilton (2018). Inputs:

  – "out/rawCompustatResidIFR.dta"

  Output:

– "out/HamiltonCompustatResidIFR.dta," containing the filtered TFP residuals, as well as p-values for the Augmented Dickey-Fuller (ADF) test for each firm, which are used to select the estimation sample in the following step.

We regress raw TFP residuals for each firm on their first five lags, and use the fitted residuals from this regression as estimates for the stationary part of the TFP process. We conduct ADF tests on the resulting series for each firm separately.

- "runOUEstimatorsIFR.do" computes the MLE estimates for the parameters of the exponential Ornstein-Uhlenbeck process. Inputs:

  – "out/HamiltonCompustatResidIFR.dta."

  Outputs:

  – "out/HamiltonOUBetaIFR.dta," containing the MLE estimates for the intermediate parameters β_1, β_2 needed for the formula of the EOU parameters (Tang and Chen, 2009);

  – "out/HamiltonXWalkIFRCodeIndustry.dta", an intermediate crosswalk between IFR Codes and industry names, needed to add the latter to the output file;

  – "out/HamiltonEstimationSample.csv", containing the observations used to compute the MLE estimators, used in the Matlab code to portray the TFP kernel densities reported in Figure B.1;

  – "out/HamiltonOUEstimatorsIFR.dta", containing the MLE estimates for the parameters of the exponential Ornstein-Uhlenbeck for each IFR sector.

  Summary of the procedure: we follow the formulas in Tang and Chen (2009), which we estimate on the subsample of firms for which we can reject the null hypothesis of the ADF test at 10% significance. We further restrict the sample to firms with more than 20 residual degrees of freedom in the ADF regression, to ensure sufficient power for the test.

- "runOUEstimatorsIFR_allsectors" computes the MLE estimates for the parameters of the exponential Ornstein-Uhlenbeck process, lumping all sectors together, needed for the one-sector calibration. Inputs:

  – "out/HamiltonCompustatResidIFR.dta."

  Outputs:

  – "out/HamiltonOUBetaIFR_total.dta," containing the MLE estimates for the intermediate parameters β_1, β_2 needed for the formula of the EOU parameters (Tang and Chen, 2009);

  – "out/HamiltonXWalkIFRCodeIndustry.dta", an intermediate crosswalk between IFR Codes and industry names, needed to add the latter to the output file;

  – "out/HamiltonOUEstimatorsIFR_total.dta", containing the MLE estimates for the parameters of the exponential Ornstein-Uhlenbeck considering manufacturing as a single sector.

  Summary of the procedure: same as above, but estimators are not computed for each sector separately.

- "runMergeAll" merges all the .dta files together with the share of manufacturing value added for each IFR sector. Inputs:

  – "VA_DATA14.csv" containing the shares of manufacturing value added of each IFR sector in 2014. The match of GDP-by-sector reported by the BEA can be read from the formulas in the Sheet "Manufacturing14" of the Excel file "VALUEADDEDBEA.xls". "VA_DATA14.csv" is a copy of the last Sheet in the file.

– The output files mentioned in the previous steps.

Outputs:

– "OUStatistics.csv", the input for our main Matlab calibration exercise.

References

Daron Acemoglu and Pascual Restrepo. Robots and Jobs: Evidence from US Labor Markets. *Journal of Political Economy*, 128(6):2188–2244, 2020.

James D Hamilton. Why You Should Never Use the Hodrick-Prescott Filter. *Review of Economics and Statistics*, 100(5):831–843, 2018.

Sam Korus. Industrial Robot Cost Declines Should Trigger Tipping Points in Demand. https://ark-invest.com/research/industrial-robot-cost-declines, April 2019.

Cheng Yong Tang and Song Xi Chen. Parameter Estimation and Bias Correction for Diffusion Processes. *Journal of Econometrics*, 149(1):65–81, April 2009.