A Stata 17 implementation of the local autonomy ratio: Calling Python

2021 US STATA CONFERENCE,
August 5th

Morales-Castillo, J.S
Sáez-Lozano, J.L
Granada University

SUMMARY

1. Introduction: The Local Autonomy Ratio
2. How Python Can Help
3. The Code: Python and Stata
4. Applications
5. Do-file execution

SUMMARY

Introduction: The Local Autonomy Ratio

- Countries whose public sector is decentralized: +Efficiency in public services
- Level of decentralization approximated through the **Local Income Ratio**
  - **but** it has been proven that this covariate is **endogenous**
  - **and** due to the **unobservable heterogeneity**

Introduction: The Local Autonomy Ratio

- In **Chile**, the participation of the **Municipal Compensation Fund (FCM)**, in the total revenues of the municipality, is the best **indicator of decentralization (**Letelier and Sáez-Lozano, 2020)

- In 2011, Martínez-Vazquez, Vulovic, and Liu paper**:**
  - an **instrument variable (**V_{st} **)** shall be used to test the robustness of the empirical model
  - to control for **possible measurement errors** of the FCM
  - and detect **endogeneity biases**

Morales-Castillo, J.S
Sáez-Lozano, J.L

Introduction: The Local Autonomy Ratio

- Martínez-Vázquez et al. (2011) and Sanogo (2019) define the **instrumental variable** of municipality *s* in year *t*, V_{st}:

- $$V_{st} = \frac{1}{\sum_{m-1}^{M} \frac{1}{d_m}} \sum_{m-1}^{M} \frac{1}{d_m} FCM_{mt}\ ,\ s \neq m \mid d_m \leq \bar{d} \quad (1)$$

- where, d_m is distance between municipality *s* and *m*; FCM_{mt} is FCM of the municipality *m* in year *t; M* is the total number of municipalities; d_m is the distance between municipality *m* and the other municipalities in the country; $\bar{d}$ is the threshold distance.

Introduction: The Local Autonomy Ratio

- Letelier and Sáez (2020) **modified V_{st}** to apply it to Chile, including, also to the "distance" restriction, the population size limit:

$$V_{st} = \frac{1}{\sum_{m-1}^{M} \frac{1}{d_m}} \sum_{m-1}^{M} \frac{1}{d_m} FCM_{mt\,,\ s \neq m \mid d_m \leq \bar{d} \ and \ Pob_{st} \geq \overline{Pob}} \qquad (2)$$

  - where, Pob_{st} is the population of municipality *s* in period *t;* and $\overline{Pob}$ is the threshold population.

Morales-Castillo, J.S
Sáez-Lozano, J.L

SUMMARY

1. Introduction: The Local Autonomy Ratio
2. How Python can help
3. The code: Python and Stata
4. Applications
5. Do-file execution

Why python?

- To compute the instrumental variable "v" for the **Improved Local Autonomy Ratio** we face:

1. Large dataset (a matrix with distances between every single pair of municipalities in a given country)
   a) *e.g.: In Chile there are 343 municipalities, we have to deal with a 343x343 matrix*
2. Necessity to update the dataset (Possibility to use API's for data scrapping)
3. Apply different restrictions
   a) minimum distance between towns
   b) or a minimum population per town

Morales-Castillo, J.S
Sáez-Lozano, J.L

Why python?

4. Automatize its calculation to facilitate its application

a) Allow non-python users to take advantage of Stata's power

b) Attract python users to use the best Stata features vs python (Econometrics)

5. Explore Stata 17's newest python features (writing Stata code in Jupyter Lab)

Morales-Castillo, J.S
Sáez-Lozano, J.L

SUMMARY

The code: Calculating the Improved Local Autonomy Ratio (Vst)

- The structure:

1. The data: Excel and structure of data
2. Stata: Ask the user about the dataset to use and the restrictions to apply
3. Python: read the data, apply restrictions, calculate V for each municipality.
4. Stata: Use the SFI module to load data back into Stata's Data Editor
5. Stata: apply econometrics models to reveal possible correlations.

Morales-Castillo, J.S
Sáez-Lozano, J.L

The code: Calculating the Improved Local Autonomy Ratio (Vij)

1. Stata: Prompt the user about the dataset to use (Excel file's path) and the restrictions to apply

*Note that dataset has to include: 1)Distance matrix, 2)Decentralization Index (FCM), 3) Population for every municipality

** Python's input("*question?*") function doesn't work in Stata

```
display "Introduce file's name (no commas needed) " _request(file_stata)
display "Introduce number of columns with no distances" _request(number_stata)
display "Introduce name of the column with the FCM to study" _request(distance_stata)
display "Introduce name of the column with the population to study" _request(distance_stata)
display "Introduce maximum distance between cities" _request(distance_stata)
display "Introduce maximum number of population per town: " _request(population_stata)
```


Morales-Castillo, J.S
Sáez-Lozano, J.L

The code: Calculating the Improved Local Autonomy Ratio (Vij)

2. A) Python: import modules, read data, set distance matrix and checks

```
*We invoke necessary modules and packages
import pandas as pd
import numpy as np
import Dataframe as df
*We read the data and check its shape
file_python= "$file_stata”
data = pd.read_csv(file_python)
print(data.head())
*We set the distance matrix *We need to split the columns with values of variable FCM or Population from distance's value by writting down the number of columns that does not contain distances
number_python= "$number_stata”
n = int(number_python)
distances = data.iloc[:len(data.columns)-1,1:len(data.columns)-n]
*We can check matrix's dimension with:
print(distances.shape)
```


Morales-Castillo, J.S
Sáez-Lozano, J.L

The code: Calculating the Improved Local Autonomy Ratio (Vij)

2. B) Python: Applying restrictions and calculating inverse

```
*We apply the chosen maximum population restriction
population_1 = "$population_stata" #This is a variable already defined in Stata in the second line of code
population = int(population_1)
populationcolumn_python = "$populationcolumn_stata"
distances.loc[data[populationcolumn_python]<=population]
 *We apply the maximum distance restriction
y_stata = "$distance_stata"
y = int(y_stata)
matrix_minor_y= (np.where(distances<=y, distances,0))
print(matrix_minor_y)
*Inverse
inverse = 1/matrix_minor_y
from numpy import inf
inverse[inverse==inf]=0 #Correcting mistake for 1/0 (equal to "if error")
print('INVERSE')
print(inverse)
```

The code: Calculating the Improved Local Autonomy Ratio (Vij)

2. C) Python: We compute the formula

```
*We get COLUMN C =
FCM_python = “FCM_stata”
product_fcm = np.matmul(inverse,data[FCM_python])
print(pd.Series(product_fcm))
print(product_fcm.shape) #We use pd.Series so results are shown in a single column
*We get column D =
sum_inverses = np.sum(inverse,0)
print('Sum inverses per municipality')
print(pd.Series(sum_inverses))
print(sum_inverses.shape)
inverse_sum_inverses = 1/sume_inverses
print(pd.Series(inverse_sum_inverses))
*We finnaly get Vst
v = inverse_sum_inverses*product_fcm
v_df = pd.DataFrame(v)
v_df.columns = ["v value"]
print(v_df)
```

The code: Calculating the Improved Local Autonomy Ratio (Vst)

```
*We clean the results, removing municipalities that did not satisfy our restrictions
results = pd.Series(v), index = data['MUNICIPALITY']).dropna()
print(results)
results = pd.DataFrame(results)
results.columns = ["v value"]
print(results)
```

The code: Calculating the Improved Local Autonomy Ratio (Vst)

4) Now we use SFI to load the data back in Stata and save results

```
from sfi import Data
Data.setObsTotal(len(results))
Data.addVarStr("MUNICIPALITY",len(data['MUNICIPALITY']))
Data.addVarDouble("v_value")

Data.store("MUNICIPALITY", None, data['MUNICIPALITY'],None)
Data.store("v_value",None, v ,None)
end
*Rename v_value for FCM_2011
rename v_value_ v_value_2011
*Generate variables year and id
gen year = 2011
gen id = _n
*Save v_value_2011
save /Users/juanmorales/Desktop/Stata/v_value_2011.dta , replace
```

INDEX

APPLICATIONS

- The **instrumental variable V_{st}** is a **covariate of the happiness model** that Letelier and Saez-Lozano (2020) estimated for the case of Chile, in the years 2011 and 2013.

- Since happiness is a latent variable, the **level of satisfaction** with individual life, as a **proxy** variable for happiness. The authors specified a multilevel model, in which one of the covariates was V_{st}.

- They used the Newton-Raphson algorithm to maximize the likelihood function, which is done according to the adaptive quadrature procedure proposed by Rabe-Hesketh et. to the. (2005). The main conclusion that this research provides **evidence in support of the hypothesis that decentralization (V_{st}) increases the level of happiness of the Chilean population.**

Morales-Castillo, J.S
Sáez-Lozano, J.L

INDEX

REFERENCES

Letelier Saavedra, L; and Sáez-Lozano, J.L. (2020). *Fiscal Decentralization and Life Satisfaction in Chile*, Georgia: International Center for Public Policy, Andrew Young School of Policy Studies, Georgia State University. Working Paper Series, at AYSPS, GSU. Working Paper 20-07

Martinez-Vasquez, J., Vulovic, V., & Liu, Y. (2011). *Direct versus Indirect Taxation: Trends, Theory, and Economic Significance*, Georgia State University. International Studies Program, 09–11: 37–92.

Rabe-Hesketh, S., Skrondal, A. and Pickles, A. (2005). *Maximum likelihood estimation of limited and discrete dependent variable models with nested random effects*. Journal of Econometrics, 128(2): 301-323.

Sanogo T (2019) *Does fiscal decentralization enhance citizens access to public services and reduce poverty? Evidence from Côte d'Ivoire municipalities in a conflict setting*. World Development 113: 204-221.

Morales-Castillo, J.S
Sáez-Lozano, J.L

Bonus: A hand for creating an ado?

- Attempts to create an ado from this do file has been unsuccessfull
- When using Stata's program function, inability to connect Stata's variable with Python's variable
  - E.g.: file_python= "$file_stata"
- How could we compile all of these code into a program?

Thanks for your attention!

Special appreciations to:
Stata
Statalist community
Bishop, Nathan. Educational Services Coordinator at StataCorp
Escobar, Modesto. professor at Universidad de Salamanca

Contact:
Morales-Castillo, J.S: juanoptim@gmail.com
Sáez-Lozano, J.L: josaez@ugr.es