

**CONSTITUENCY-LEVEL
ELECTIONS ARCHIVE (CLEA):
EFFECTIVE NUMBER OF PARTIES
AND PARTY NATIONALIZATION
DATA**

Codebook

Version: September 8, 2026 (20260908)

TABLE OF CONTENTS

INTRODUCTION	3
Effective Number of Parties and Party Nationalization Project Description	3
CLEA Project Description	3
Funding Support.....	4
Bibliographic Citation	4
CLEA Credits	4
Data Procedures.....	5
Calculation of Variables in ENP Dataset	5
DOCUMENTATION OF VARIABLES	7
Variable List.....	7
National Dataset	7
Party Dataset.....	7
Constituency Dataset	8
Variable Descriptions.....	9
REFERENCES	23
APPENDIX: CODE TO REPLICATE NATIONALIZATION CALCULATIONS	24
Stata Code (Lower Chamber).....	24
Gini Index for Parties	24
Party Nationalization Measures - Party Level Dataset.....	25
Party Nationalization Measures - National Level Dataset.....	29
Party Nationalization Measures - Constituency Level Dataset	38
Stata Code (Upper Chamber)	41
Data Prep	41
Gini Index for Parties	41
Party Nationalization Measures - Party Level Dataset.....	43
Party Nationalization Measures - National Level Dataset.....	45
Party Nationalization Measures - Constituency Level Dataset	51

INTRODUCTION

Effective Number of Parties and Party Nationalization Project Description

The Effective Number of Parties and Party Nationalization datasets uses the Constituency-Level Elections Archive (CLEA) data (described below) to generate the effective number of parties as well as several measures of party nationalization that are described in the political science literature. We present these data in three separate datasets that represent different levels of aggregation: the national level, the party level, and the constituency level. The Effective Number of Parties and Party Nationalization data are provided separately for the lower chamber and upper chamber. This codebook applies to both sets of data as they contain the same variable list and descriptions. Files associated with the lower chamber data begin with ‘*clea_lc_enp*’ and files associated with the upper chamber data begin with ‘*clea_uc_enp*’.

At the national level, we present three measures of the effective number of parties (Laakso and Taagepera, 1979) and nine measures of party system nationalization, or the degree to which the set of political parties in a given country-year have electoral success at the national level (as opposed to more localized or regionalized success).

At the party level, we provide four measures that indicate the level of nationalization for a particular party in a given country-year.

Finally, the constituency level dataset includes the national measures for each constituency in a given country-year, the effective number of parties at the constituency level, and a local inflation measure.

Details about the interpretation and application of the measures are available in the references provided for each variable. Summaries of the measures are provided in this codebook and the appendix provides full computer code for calculations in *Stata* and *R* software.

CLEA Project Description

The central aim of the Constituency-Level Elections Archive (CLEA) project is to produce a repository of detailed results – i.e. votes received by each candidate/party, total votes cast, number of eligible voters – at the constituency level for the lower chamber and upper chamber legislative elections that have been conducted around the world. Our motivation is to preserve and consolidate these valuable data in one comprehensive reference resource that is publicly available at no cost. This public good is expected to be of use to a range of audiences for purposes of research, education, policy-making, and evaluation.

Funding Support

We gratefully acknowledge the American National Science Foundation (www.nsf.gov) and the Taiwan Foundation for Democracy for their funding of the project, and we thank the Institute of Political Science at the University of St. Gallen and the Department of Political Science at the University of Zurich for research assistance. The Center for Political Studies (CPS) at the University of Michigan has contributed financially and in kind to the CLEA project.

Bibliographic Citation

APA (7th edition)

Kollman, K., Hicken, A., Caramani, D., Backer, D., Lublin, D., and Kim, W. (2019). *Constituency-Level Elections Archive: Party Nationalization Dataset*. [Data set and codebook]. Ann Arbor, MI: Center for Political Studies, University of Michigan [producer and distributor]. Retrieved from <http://www.electiondataarchive.org>.

MLA (9th edition)

Kollman, Ken, Allen Hicken, Daniele Caramani, David Backer, David Lublin, and Wooseok Kim. *Constituency-Level Elections Archive: Party Nationalization Dataset*. Ann Arbor, MI: Center for Political Studies, University of Michigan [producer and distributor], 2023. Web. 15 February 2023. <http://www.electiondataarchive.org>

Chicago (17th edition)

Kollman, Ken, Allen Hicken, Daniele Caramani, David Backer, David Lublin, and Wooseok Kim. 2020. *Constituency-Level Elections Archive: Party Nationalization Dataset*. Produced and distributed by Ann Arbor, MI: Center for Political Studies, University of Michigan, <http://www.electiondataarchive.org>.

These materials are based on work supported by the National Science Foundation (under grants SES-0617101, SES-1021915, and SES-1528262). Any opinions, findings and conclusions or recommendations expressed in these materials are those of the author(s) and do not necessarily reflect the views of the funding organizations.

CLEA Credits

The co-directors of the CLEA project are Ken Kollman of the University of Michigan, Allen Hicken of the University of Michigan, Daniele Caramani of the University of Zurich, David Backer of the University of Maryland, and David Lublin of American University. The project manager is Julia Lippman of the Center for Political Studies, University of Michigan.

Data in this and prior releases have been contributed by David Lublin, Jan Teorell, Jose Manuel Magallanes, Nir Atmor, David Backer, Daniele Caramani, Adam Carr, Cengiz Erisen, Federico Ferrara, Brian Gaines, Judy Geist, Allen Hicken, Ken Kollman, Arend Lijphart, Scott Morgenstern, Jairo Nicolau, Daniel Posner, Jae-Jae Spoon, Marcelo Leiras, Steven Reed, Ching-

hsing Yu, Matt Singer, Heather Stoll, Jack Vowles, Sarah Shair-Rosenfield, Daniel Bochsler, Joel Selway, Francesca Jensenius, Gilles Verniers, Yen-Pin Su, Binod Paudel, and Anthony Sayers, as well as ICPSR, The Project on Political Transformation and the Electoral Process in Post-Communist Europe program at the University of Essex, The Trivedi Centre for Political Data at Ashoka University, and Tindak Malaysia.

Research assistance was provided by Megan Bilbao, Sam Braden, Ryan Comrie, Sarah Fernandes, Diogo Ferrari, Luke Gaines, Samantha Goldstein, Jacob Granzow, Lauren Jacobs, Sara Jex, Abby Jodloski, Kirill Kalinin, Michael Keller, Sarah Kent, Ethan Miles, Erica Mirabitor, Yioryos Nardis, Sandra Nwogu, Simon Page, Rebecca Shipan, Anna Torell, Morgan Vanderleest, Fabricio Vasselai, and Tyler Watt.

Data Procedures

In order to construct this dataset, we have used the CLEA data to recreate several measures described in the party nationalization literature. We have attempted to construct the measures in accordance with the procedures described by the originating authors. Should readers have questions about the interpretation of a variable, the originating work (cited in the description of each variable in this codebook) should be considered the authoritative source. Additionally, although great care has been taken to prepare the data and codebook prior to release, we would appreciate receiving your comments, feedback and notifications of any remaining errors by email to: thea-project@umich.edu

Calculation of Variables in ENP Dataset

Variety in electoral systems across countries presents challenges on how to treat party vote totals, constituency and national vote totals, and proportions. The measures in these datasets require calculations of votes cast, votes counted for parties, and votes sometimes aggregated into larger units. These calculations are used in formulae for measures of nationalization. The challenges mean that we need to make decisions that may not seem obvious to all users of the data. As an example, in Germany's mixed system, should the first and second votes be pooled within each constituency to give a party vote for that constituency? In fact, we do not pool; instead, the upper tiers are treated as separate constituencies. We typically do this for countries with upper tiers. This is a choice (perhaps arbitrary), but we are following two principles, one theoretical and one practical.

First, as a general rule in CLEA we try as best we can to organize the data in the main CLEA dataset around how voters encounter their options on ballots. If, for instance, candidate names are given on the ballot and that's how voters choose, then candidates' votes are given in the database. If only parties are listed on ballots as options, then parties are the units with votes in the database. This principle cannot always be followed and we are at the mercy of how data are released by governments. But when faced with decisions about how to organize data, we follow the principle focusing on voters' options on ballots. Second, the manner in which data are organized in the main CLEA dataset determines how the calculations for nationalization are conducted for these datasets here. Thus, in our example on Germany, since the upper tiers are considered separate constituencies in the large elections database, they are treated that way for

calculating nationalization measures. The computer code at the end of this codebook shows, for instance, that to calculate party vote totals in each constituency, we use the party codes as they are assigned for the main CLEA dataset. Refer to the computer code in the appendix to this codebook and to the main CLEA dataset for assistance with questions about the treatment of specific countries. And as always, feel free to give comments, feedback and notifications of any remaining errors by email to: thea-project@umich.edu

DOCUMENTATION OF VARIABLES

Variable List

National Dataset

ID	Election Identifier
CTR_N	Country Name
CTR	Country Code
YR	Election Year
MN	Election Month
CST_TOT	Total Number of Constituencies
NVVI	National Valid Vote Indicator
ENP_NAT	Effective Number of Parties (National)
ENP_AVG	Effective Number of Parties (Average)
ENP_WGHT	Effective Number of Parties (Weighted Average)
INFLATION1	Cox Inflation Score
INFLATION2	Moenius and Kasuya Inflation Score
INFLATION3	Moenius and Kasuya Weighted Inflation Score
INFLATION4	Kasuya and Moenius Inflation and Dispersion Score
PSNS	Party System Nationalization Score
PSNS_S	Standardized Party System Nationalization Score
PSNS_W	Weighted Party System Nationalization Score
PSNS_SW	Standardized and Weighted Party System Nationalization Score
LOCAL_E	Local Entrant Measure

Party Dataset

ID	Election Identifier
CTR_N	Country Name
CTR	Country Code
YR	Election Year
MN	Election Month
CST_TOT	Total Number of Constituencies
PTY_N	Party Name
PTY	Party Code
PNS	Party Nationalization Score
PNS_S	Standardized Party Nationalization Score
PNS_W	Weighted Party Nationalization Score
PNS_SW	Standardized and Weighted Party Nationalization Score

Constituency Dataset

ID	Election Identifier
CTR_N	Country Name
CTR	Country Code
YR	Election Year
MN	Election Month
CST_N	Constituency Name
CST	Constituency Code
CST_TOT	Total Number of Constituencies
NVVI	National Valid Vote Indicator
CVVI	Constituency Valid Vote Indicator
ENP_CST	Effective Number of Parties (Constituency)
ENP_NAT	Effective Number of Parties (National)
ENP_AVG	Effective Number of Parties (Average)
ENP_WGHT	Effective Number of Parties (Weighted Average)
INFLATION1	Cox Inflation Score
INFLATION2	Moenius and Kasuya Inflation Score
INFLATION3	Moenius and Kasuya Weighted Inflation Score
INFLATION4	Kasuya and Moenius Inflation and Dispersion Score
INFLATION5	Moenius and Kasuya Local Inflation Score
PSNS	Party System Nationalization Score
PSNS_S	Standardized Party System Nationalization Score
PSNS_W	Weighted Party System Nationalization Score
PSNS_SW	Standardized and Weighted Party System Nationalization Score
LOCAL_E	Local Entrant Measure

Variable Descriptions

ID ***Election Identifier***

This variable uniquely identifies each election in the archive.

NOTE: Early U.S. elections to the House of Representatives were scheduled differently across states and there were no federal laws or regulations requiring specific timing other than that they occur every two years. U.S. House of Representatives elections prior to 1880 are assigned an election identifier of ‘-999’ in the CLEA Lower Chamber data.

CTR ***Country Code***

Country codes developed by the UN: <https://unstats.un.org/unsd/methodology/m49/>

004 Afghanistan	070 Bosnia and Herzegovina
248 Åland Islands	072 Botswana
008 Albania	076 Brazil
012 Algeria	092 British Virgin Islands
020 Andorra	100 Bulgaria
024 Angola	854 Burkina Faso
660 Anguilla	108 Burundi
028 Antigua and Barbuda	116 Cambodia
032 Argentina	120 Cameroon
051 Armenia	124 Canada
533 Aruba	132 Cape Verde
036 Australia	136 Cayman Islands
040 Austria	152 Chile
031 Azerbaijan	170 Colombia
044 Bahamas	174 Comoros
048 Bahrain	184 Cook Islands
050 Bangladesh	188 Costa Rica
052 Barbados	384 Côte d'Ivoire
112 Belarus	191 Croatia
056 Belgium	531 Curaçao
084 Belize	196 Cyprus
204 Benin	203 Czech Republic
060 Bermuda	208 Denmark
064 Bhutan	262 Djibouti
068 Bolivia	212 Dominica

214 Dominican Republic	404 Kenya
218 Ecuador	410 Korea
222 El Salvador	414 Kuwait
226 Equatorial Guinea	417 Kyrgyzstan
233 Estonia	428 Latvia
748 ESwatini	422 Lebanon
231 Ethiopia	426 Lesotho
234 Faroe Islands	430 Liberia
242 Fiji	434 Libya
246 Finland	438 Liechtenstein
250 France	440 Lithuania
266 Gabon	442 Luxembourg
270 Gambia	446 Macau
268 Georgia	450 Madagascar
276 Germany	454 Malawi
288 Ghana	458 Malaysia
292 Gibraltar	462 Maldives
300 Greece	470 Malta
304 Greenland	584 Marshall Islands
308 Grenada	480 Mauritius
320 Guatemala	484 Mexico
324 Guinea	583 Micronesia
624 Guinea-Bissau	492 Monaco
328 Guyana	498 Moldova
340 Honduras	496 Mongolia
344 Hong Kong	499 Montenegro
348 Hungary	500 Montserrat
352 Iceland	504 Morocco
356 India	508 Mozambique
360 Indonesia	104 Myanmar
364 Iran	516 Namibia
368 Iraq	520 Nauru
372 Ireland	524 Nepal
833 Isle of Man	528 Netherlands
376 Israel	554 New Zealand
380 Italy	558 Nicaragua
388 Jamaica	562 Niger
392 Japan	566 Nigeria
400 Jordan	807 North Macedonia
398 Kazakhstan	578 Norway

586 Pakistan	144 Sri Lanka
585 Palau	670 St. Vincent and the Grenadines
275 Palestine	740 Suriname
591 Panama	752 Sweden
598 Papua New Guinea	756 Switzerland
600 Paraguay	834 Tanzania
604 Peru	764 Thailand
608 Philippines	626 Timor-Leste
616 Poland	768 Togo
620 Portugal	776 Tonga
630 Puerto Rico	780 Trinidad and Tobago
642 Romania	788 Tunisia
643 Russian Federation	792 Turkey
646 Rwanda	796 Turks and Caicos Islands
654 Saint Helena	798 Tuvalu
659 Saint Kitts and Nevis	800 Uganda
662 Saint Lucia	804 Ukraine
674 San Marino	784 United Arab Emirates
882 Samoa	826 United Kingdom
686 Senegal	840 United States
688 Serbia	858 Uruguay
690 Seychelles	548 Vanuatu
694 Sierra Leone	862 Venezuela
702 Singapore	894 Zambia
534 Sint Maarten	716 Zimbabwe
703 Slovakia	1001 Taiwan ***
705 Slovenia	1002 Kosovo ***
090 Solomon Islands	1003 Somaliland ***
710 South Africa	
724 Spain	

*** Taiwan (the Republic of China), Kosovo, and Somaliland do not have a U.N. Standard Country Code. This code is assigned by CLEA.

YR ***Election Year***

Year of election.

MN ***Election Month***

Month of election, if available.

Note: To conduct calculations for elections spanning more than one month, the *mn* variable is set to “-990. Missing Data (information not available /category not applicable)”.

- 01. January
- 02. February
- 03. March
- 04. April
- 05. May
- 06. June
- 07. July
- 08. August
- 09. September
- 10. October
- 11. November
- 12. December

CST_N ***Constituency Name***

Name of geographical area that a particular elected representative or group of elected representatives represents.

CST ***Constituency Code***

A unique numeric code assigned to each constituency in each election in a country. In general, all constituencies in a country are sorted alphabetically, according to their names, and then assigned a constituency code. This code assignment is repeated in each election in the country. Thus, the same code may or may not belong to the same constituency across elections, depending upon whether redistricting occurs between elections. In the event of special districts for minority populations (e.g., the Maori districts in New Zealand prior to the electoral reform in 1996) or semi-autonomous regions (e.g., Greenland for Danish parliamentary elections) these districts receive the first numeric code following the last alphabetically sorted geographical district.

In a case where a country uses a multi-tier or mixed electoral system, the CLEA dataset uses the following coding scheme:

001-900. Lower-tier electoral districts (in multi-tier PR) or electoral districts where a majoritarian formula is used (in a mixed electoral system)

901-999. Upper-tier electoral districts (in multi-tier PR) or electoral districts where PR is used (in a mixed electoral system)

If a country uses a single-tier system, only constituency codes for lower-tier electoral districts are used.

CST_TOT***Total Number of Constituencies***

The total number of constituencies in a given election.

PTY_N***Party Name***

Name of a party or electoral alliance. If possible, the official name in the original language is used. If this name is not available, the transliterated or English-translated party name is used. For more information, refer to *Appendix II: Party Codes*.

In some cases where the original data sources we collected have small parties grouped under an “*Others*” category, “*others*” is used. In several countries, special kinds of party groupings are used in reported election results. For instance, categories such as “*miscellaneous right-wing*” and “*regionalists and separatists*” are used in France. For those special categories, their names are used for this variable and unique numeric codes are assigned to each such category (see “PTY” below for more information about these categories and also refer to *Appendix II: Party Codes*).

Independent candidates are handled in two different ways when election results are reported. For much of the data we have collected, all independent candidates are grouped under a single category. In such cases, “*Independents*” is used. However, when each and every individual independent candidate is identified and his/her votes received are reported separately in the election returns, “*Independent*” is used instead. Different numeric codes are assigned with these different methods. See “PTY” below for more information about numeric codes and Appendix II for the labels for independent candidates.

In a few countries, there are independent candidates who are affiliated with a party, but cannot officially stand under its label. As a result, they may be labeled in a manner that reflects both their independent status and their party affiliation (e.g., “*Independent Labour*” or “*Independent Greens*”). In the CLEA dataset, we treat these candidates as independents in the coding, but keep their label under the PTY_N.

For a full list of political parties, see *Appendix II: Party Codes*.

PTY***Party Code***

A unique numeric code is assigned to each party that runs a candidate in any given election. In general, political parties in a country are sorted alphabetically according to PTY_N and then assigned party codes. Parties have been matched across the lower chamber and upper chamber election results. Therefore, a party in a given country that runs in both chambers receives the same party code. To differentiate political parties and

the aforementioned special and residual categories (see *PTY_N* above), the CLEA dataset uses the following coding scheme:

0001-3999. Political parties

- 3996. None of these candidates (in some countries, voters have the option to express disapproval for all the candidates on the ballot)
- 3997. Elected (for several early elections in Iceland and Sweden, the results for political parties are not available)
- 3998. No against for uncontested (in Denmark)
- 3999. Unknown

4000. “Others” (i.e., more than two small parties are grouped)

- 4001-. Special kinds of ‘others’ (see Appendix II for more information)
- 4998. Write-in

5001-5999. Electoral coalitions or alliances between political parties

6000. “Independents” (i.e., more than two independent candidates are grouped)

- 6001-. Independent 1, Independent 2, and so on (i.e., a single unaffiliated candidate), including special kinds of ‘independents’.

For a full list of political parties and their codes in each country, see *Appendix II: Party Codes*.

Each party and electoral alliance is assigned a unique numeric code that remains consistent across elections. If a party changes its name, merges with other parties or splits into separate parties, a new numeric code is given to the party that emerges as a result of such changes.

Party codes for ‘other’ and ‘independent’ are assigned to parties or unaffiliated candidates in each election. This code assignment is repeated in each election in the country. Thus, the same code does not belong to the same minor party or independent candidate across elections.

NOTE: In India, there were more than 4,000 independent candidates in several general elections. In such cases, many independent candidates are assigned a five-digit party code.

NVVI

National Valid Vote Indicator

An indicator variable that takes on a value of “1” when the national effective number of parties measures (ENP_{NAT} , ENP_{AVG} , and ENP_{WGHT}) rely on the total sum of party votes (rather than the raw number of valid votes), and “0” otherwise. The CLEA data on the number of valid votes cast in a constituency do not always sum to the total number of party votes in the constituency. This is problematic in the calculation of the effective

number of parties; we substitute summed party votes for valid votes as indicated by this variable.

CVVI ***Constituency Valid Vote Indicator***

An indicator variable that takes on a value of “1” when the constituency effective number of parties measure (ENP_{CST}) relies on the total sum of party votes (rather than the raw number of valid votes), and “0” otherwise. The CLEA data on the number of valid votes cast in a constituency do not always sum to the total number of party votes in the constituency. This is problematic in the calculation of the effective number of parties; we substitute summed party votes for valid votes as indicated by this variable.

ENP_CST ***Effective Number of Parties (Constituency)***

The effective number of (electoral) parties in a country’s party system at the constituency level for the specified election year. This is calculated at the constituency level following Laakso and Taagepera’s (1979) specification:

$$ENP_{cst} = \frac{1}{\sum_1^n p_i^2}$$

Here n represents the number of parties in a district and p_i represents the party’s share of the constituency vote. Values coded as “-990” indicate missing data.

For more information on the interpretation and calculation of this variable, see: Laakso, M., & Taagepera, R. (1979). “Effective Number of Parties: A Measure with Application to West Europe.” *Comparative Political Studies* 12(1): 3-27.

ENP_NAT ***Effective Number of Parties (National)***

The effective number of (electoral) parties in a country’s party system at the national level for the specified election year. This is calculated at the national level following Laakso and Taagepera’s (1979) specification:

$$ENP_{nat} = \frac{1}{\sum_1^n p_i^2}$$

Here n represents the number of parties in a country and p_i represents the party’s share of the national vote. Values coded as “-990” indicate missing data. Due to the large number of independent candidates, we do not provide these data for Korea in 1950 or the United States before 1834.

For more information on the interpretation and calculation of this variable, see: Laakso, M., & Taagepera, R. (1979). “Effective Number of Parties: A Measure with Application to West Europe.” *Comparative Political Studies* 12(1): 3-27.

ENP_AVG***Effective Number of Parties (Average)***

The average effective number of (electoral) parties in a country's party system at the national level for the specified election year. The number of parties is calculated at the constituency level, following Laakso and Taagepera's (1979) specification, and then averaged at the national level.

$$ENP_{avg} = \frac{\sum_1^d ENEP_{cst}}{d}$$

Here d represents the number of districts in a country. Values coded as "-990" indicate missing data. Due to the large number of independent candidates, we do not provide these data for Korea in 1950 or the United States before 1834.

For more information on the interpretation and calculation of this variable, see: Laakso, M., & Taagepera, R. (1979). "Effective Number of Parties: A Measure with Application to West Europe." *Comparative Political Studies* 12(1): 3-27.

ENP_WGHT *Effective Number of Parties (Weighted)*

A weighted average of the effective number of (electoral) parties in a country's party system at the national level for the specified election year. The number of parties is calculated at the constituency level, following Laakso and Taagepera's (1979) specification, and then averaged at the national level, weighting each constituency according to its share of voters out of the national total (Moenius & Kasuya, 2004). Formally, this is:

$$ENP_{wght} = \sum_1^d ENP_{cst} \frac{vot_{cst}}{vot_{nat}}$$

Here, d represents the number of districts in the country, vot_{cst} indicates the number of votes cast in the constituency and vot_{nat} indicates the number of votes cast in the country. Values coded as "-990" indicate missing data. Due to the large number of independent candidates, we do not provide these data for Korea in 1950 or the United States before 1834.

For more information on the interpretation and calculation of this variable, see: Laakso, M., & Taagepera, R. (1979). "Effective Number of Parties: A Measure with Application to West Europe." *Comparative Political Studies* 12(1): 3-27.

INFLATION1 *Cox Inflation Score*

A measure of party linkage across a country's electoral constituencies that builds on the economic principle of inflation. Here inflation refers to the discrepancy that occurs in

party linkage as parties are aggregated from the constituency-level to the national-level party system. This measure was developed by Cox (1999).

$$inflation1 = \frac{ENP_{nat} - ENP_{avg}}{ENP_{nat}}$$

Values coded as “-990” indicate missing data.

For more information on the interpretation and calculation of this variable, see: Cox, G. (1999). “Electoral Rules and Electoral Coordination.” *Annual Review of Political Science* 2(1): 145-161.

INFLATION2 *Moenius and Kasuya Inflation Score*

A measure of party linkage across a country’s electoral constituencies that builds on the economic principle of inflation. Here inflation refers to the discrepancy that occurs in party linkage as parties are aggregated from the constituency-level to the national-level party system. This measure corresponds with Moenius and Kasuya’s (2004) variable *I*, and is distinct from Inflation 1 because it has a different denominator.

$$inflation2 = \frac{ENP_{nat} - ENP_{avg}}{ENP_{avg}}$$

Values coded as “-990” indicate missing data.

For more information on the interpretation and calculation of this variable, see: Moenius, J., & Kasuya, Y. (2004). “Measuring Party Linkage Across Districts.” *Party Politics* 10(5): 543-564.

INFLATION3 *Moenius and Kasuya Weighted Inflation Score*

A measure of party linkage across a country’s electoral constituencies that builds on the economic principle of inflation. Here inflation refers to the discrepancy that occurs in party linkage as parties are aggregated from the constituency-level to the national-level party system. This measure corresponds with Moenius and Kasuya’s (2004) weighted inflation variable *I_w*, which accounts for variation in district size.

$$inflation3 = \frac{ENP_{nat} - ENP_{wght}}{ENP_{wght}}$$

Values coded as “-990” indicate missing data.

For more information on the interpretation and calculation of this variable, see: Moenius, J., & Kasuya, Y. (2004). “Measuring Party Linkage Across Districts.” *Party Politics* 10(5): 543-564.

INFLATION4***Kasuya and Moenius Inflation and Dispersion Score***

A measure of party linkage across a country's electoral constituencies that builds on the economic principle of inflation. Here inflation refers to the discrepancy that occurs in party linkage as parties are aggregated from the constituency-level to the national-level party system. This measure also account for a dispersion, which "refers to the extent to which the contribution of each district's party system to the size of the national-level party system varies across districts" (Kasuya & Moenius, 2008). The formula is:

$$inflation4 = inflation3^{\alpha} D^{1-\alpha}$$

where *inflation3* captures the inflation dimension and *D* captures the dispersion dimension. The formula for dispersion is:

$$D = CV(inflation5)^{\gamma} k(inflation5)^{1-\gamma}$$

Here, *CV* (the coefficient of variation) and *k* (the district's kurtosis) both refer to measures of the local inflation measure (*inflation5*---see next variable). Note that the alpha and gamma parameters are user-defined. Gamma weights the districts according to their relationship to the aggregate party system inflation rate and alpha is the weight assigned to inflation versus dispersion. For the purposes of this analysis (and following Kasuya and Moenius's basic recommendation) inflation and dispersion are equally weighted (alpha = .5), as are the districts (gamma = 0.5). Values coded as "-990" indicate missing data.

For more information on the interpretation and calculation of this variable, see: Kasuya, Y., & Moenius, J. (2008). "The Nationalization of Party Systems: Conceptual Issues and Alternative District-focused Measures." *Electoral Studies* 27(1): 126-135.

INFLATION5 *Moenius and Kasuya Local Inflation Score*

A measure of party linkage that measures the difference between the party system at the local constituency level and the national level. This measure is similar to Moenius and Kasuya's (2004) weighted inflation variable *I_i*:

$$inflation5 = \frac{ENP_{nat} - ENP_{cst}}{ENP_{cst}}$$

Values coded as "-990" indicate missing data.

For more information on the interpretation and calculation of this variable, see: Moenius, J., & Kasuya, Y. (2004). "Measuring Party Linkage Across Districts." *Party Politics* 10(5): 543-564

PNS **Party Nationalization Score**

A measure of the nationalization of a party, based on the Gini coefficient of inequality (G_i) in vote shares across constituencies (Jones & Mainwaring, 2003). To calculate, we take the inverse of the Gini coefficient:

$$PNS = 1 - G_i$$

The Stata program *INEQDEC0* (Jenkins, 1999) was used to create the Gini coefficient. We do not calculate the Party Nationalization Score for parties that received less than a five percent share of the national vote in an election. Values coded as “-990” indicate missing data.

For more information on the interpretation and calculation of this variable, see: Jones, M. P., & Mainwaring, S. (2003). “The Nationalization of Parties and Party Systems.” *Party Politics* 9(2): 139-166.

PNS_S **Standardized Party Nationalization Score**

A measure of the nationalization of a party, based on the Gini coefficient of inequality in vote shares across constituencies (Jones & Mainwaring, 2003). For this measure, the Party Nationalization Score (*PNS*) is standardized to account for variation in the number of territorial units across countries (Bochsler, 2006). The formula is:

$$PNS_s = (PNS)^{\frac{1}{\log(d)}}$$

where d is the number of districts in a country. The Stata program *INEQDEC0* (Jenkins, 1999) was used to create the Gini coefficient. We do not calculate the Standardized Party Nationalization Score for parties that received less than a five percent share of the national vote in an election. Values coded as “-990” indicate missing data.

For more information on the interpretation and calculation of this variable, see: Bochslers, D. (2006). “The Nationalization of Political Parties: A Triangle Model, Applied on the Central and Eastern European Countries.” *CEU Political Science Journal* 1(4): 6-32.

PNS_W **Weighted Party Nationalization Score**

A measure of the nationalization of a party, based on the Gini coefficient of inequality in vote shares across constituencies (Bochsler, 2010). The formula is:

$$PNS_w = 2 * \frac{\sum_1^d (vot_{cst} * (\sum_1^i pty_j - \frac{pty_i}{2}))}{\sum_1^d vot_{cst} * \sum_1^d pty_i}$$

The number of votes (absolute) in the constituency is vot_{cst} and pty_i is the number of votes for party i (absolute) in the constituency. Here pty_j is the cumulative number of votes for

party i in the election, with districts sorted (increasing) according to the party's vote share in the constituency. Values coded as "-990" indicate missing data.

For more information on the interpretation and calculation of this variable, see: Bochslers, D. (2010). "Measuring Party Nationalisation: A New Gini-based Indicator that Corrects for the Number of Units." *Electoral Studies* 29(1): 155-168.

PNS_SW ***Standardized and Weighted Party Nationalization Score***

A measure of the nationalization of a party that standardizes for the number of territorial units in a country and also weights for the size of the territorial units (Bochsler, 2010). This measure builds upon the weighted party nationalization score (*PNS_W*) by adding an additional correction for the unequal sizes of units across countries. The formula is:

$$PNS_{sw} = (PNS_w)^{\frac{1}{\log(E)}}$$

where, the variable E is a constant calculated at the national level as follows:

$$E = \frac{(\sum_1^d vot_{cst})^2}{\sum_1^d vot_{cst}^2}$$

where vot_{cst} is the raw number of votes cast in constituency i . Values coded as "-990" indicate missing data.

For more information on the interpretation and calculation of this variable, see: Bochslers, D. (2010). "Measuring Party Nationalisation: A New Gini-based Indicator that Corrects for the Number of Units." *Electoral Studies* 29(1): 155-168.

PSNS ***Party System Nationalization Score***

A summary expression of the level of nationalization of a party system (Jones & Mainwaring, 2003). It is calculated according to the following formula:

$$PSNS = \sum_1^n PNS_i * p_i$$

where p_i is the party's share of the national vote. Parties that received less than a five percent share of the vote at the national level are excluded from this analysis. Values coded as "-990" indicate missing data.

For more information on the interpretation and calculation of this variable, see: Jones, M. P., & Mainwaring, S. (2003). "The Nationalization of Parties and Party Systems." *Party Politics* 9(2): 139-166.

PSNS_S Standardized Party System Nationalization Score

A summary expression of the level of nationalization of a party system that is standardized to account for variation in the number of territorial units across countries (Bochsler, 2006).

$$PSNS_S = \sum_1^n PNS_{S,i} * p_i$$

where p_i is the party's share of the national vote. Parties that received less than a five percent share of the vote at the national level are excluded from this analysis. Values coded as "-990" indicate missing data.

For more information on the interpretation and calculation of this variable, see: Bochsler, D. (2006). "The Nationalization of Political Parties: A Triangle Model, Applied on the Central and Eastern European Countries." *CEU Political Science Journal* 1(4): 6-32.

PSNS_W Weighted Party System Nationalization Score

A summary expression of the level of nationalization of a party system that is weighted to account for variation in the size of the territorial units across countries (Bochsler, 2010).

$$PSNS_W = \sum_1^n PNS_{W,i} * p_i$$

where p_i is the party's share of the national vote. Values coded as "-990" indicate missing data.

For more information on the interpretation and calculation of this variable, see: Bochsler, D. (2010). "Measuring Party Nationalisation: A New Gini-based Indicator that Corrects for the Number of Units." *Electoral Studies* 29(1): 155-168.

PSNS_SW Standardized and Weighted Party System Nationalization Score

A summary expression of the level of the nationalization of a party system that standardizes for the number of territorial units and also weights for the size of the territorial units (Bochsler, 2010).

$$PSNS_{SW} = \sum_1^n PNS_{SW,i} * p_i$$

where p_i is the party's share of the national vote. Values coded as "-990" indicate missing data.

For more information on the interpretation and calculation of this variable, see: Bochslers, D. (2010). "Measuring Party Nationalisation: A New Gini-based Indicator that Corrects for the Number of Units." *Electoral Studies* 29(1): 155-168.

LOCAL_E***Local Entrant Measure***

A measure of party system nationalization based on parties' entry decisions (Lago & Montero, 2010). This measure accounts for the differential size of territorial units, the variance in party supply across districts, and party votes shares. It varies between 0 and 1. The formula for this measure is:

$$local_E = \sum_1^n v_i * s_i$$

Here, v_i is the proportion of votes obtained by party i at the national level and s_i is the proportion of seats of the national total to be allocated in those districts where party i entered the race. Values coded as "-990" indicate missing data.

For more information on the interpretation and calculation of this variable, see: Lago, I., & Montero, J. R. (2010). *The Nationalisation of Party Systems Revisited: A New Measure Based on Parties' Entry Decisions, Electoral Results, and District Magnitude*. Paper presented at the Annual Meeting of the Canadian Political Science Association, Concordia University, Montreal, PQ.

REFERENCES

- Bochsler, D. (2006). The Nationalization of Political Parties: A Triangle Model, Applied on the Central and Eastern European Countries. *CEU Political Science Journal*, 1(4), 6 - 32.
- Bochsler, D. (2010). Measuring Party Nationalisation: A New Gini-based Indicator that Corrects for the Number of Units. *Electoral Studies*, 29(1), 155-168.
- Cox, G. (1999). Electoral Rules and Electoral Coordination. *Annual Review of Political Science*, 2(1), 145-161.
- Jenkins, S. P. (1999). INEQDEC0: Stata Module to Calculate Inequality Indices with Decomposition by Subgroup: Boston College Department of Economics.
- Jones, M. P., & Mainwaring, S. (2003). The Nationalization of Parties and Party Systems. *Party Politics*, 9(2), 139-166.
- Kasuya, Y., & Moenius, J. (2008). The Nationalization of Party Systems: Conceptual Issues and Alternative District-focused Measures. *Electoral Studies*, 27(1), 126-135.
- Laakso, M., & Taagepera, R. (1979). Effective Number of Parties: A Measure with Application to West Europe. *Comparative political studies*, 12(1), 3-27.
- Lago, I., & Montero, J. R. (2010). *The Nationalisation of Party Systems Revisited: A New Measure Based on Parties' Entry Decisions, Electoral Results, and District Magnitude*. Paper presented at the Annual Meeting of the Canadian Political Science Association, Concordia University, Montreal, PQ.
- Moenius, J., & Kasuya, Y. (2004). Measuring Party Linkage Across Districts. *Party Politics*, 10(5), 543-564.

APPENDIX: CODE TO REPLICATE NATIONALIZATION CALCULATIONS

Set the working directory to the folder that contains the raw CLEA data, and save the code in the folder before running it. The code below uses CLEA versions “clea_lc_20251015.csv” and “clea_uc_20261001.csv” for the lower and upper chamber, respectively. Please replace the lines that load the raw CLEA data files to use different versions.

Stata Code (Lower Chamber)

Gini Index for Parties

/*This file creates the Gini index for CLEA data, indicating the inequality in vote shares received by parties in districts. This information is used for calculating Jones and Mainwaring's (2003) Party Nationalization Score (PNS).*/

*To create the Gini index, we use the command INEQDEC0. The code below installs the command.

*ssc inst ineqdec0, replace

*Create Gini using CLEA data

use "clea_lc_20251015.dta", clear

replace yr=2004 if id==1792

replace mn=10 if id==1792

replace yr=1984 if id==1422

replace mn=12 if id==1422

replace yr=1991 if id==1424

replace mn=5 if id==1424

replace mn=4 if id==1425

replace ctr_n="Macedonia" if ctr_n=="North Macedonia"

replace yr=1970 if id==2308

replace mn=12 if id==2308

replace mn=2 if id==2079

replace mn=7 if id==2081

replace mn=4 if id==2326

replace mn=2 if id==2331

*Eliminate single constituency elections

bysort id ctr yr mn (cst): gen single_constituency = (cst[1] == cst[_N])

bysort id ctr yr mn (cst): gen single_constituency2 = cst[1] == 1 & cst[_N] == 1

drop if single_constituency==1 | single_constituency2==1

save "CLEA.no.SC.dta", replace

*Move the data from candidate-cst level to party-cst level

collapse (first) ctr_n vv1 pv1 if vv1 > 0 & pv1 > 0, by (id ctr yr mn pty cst)

*Drop observations where the party wins less than five percent of the national party vote

egen nat_vv1 = sum(pv1), by(id ctr yr mn)

egen pty_nat_vv1 = sum(pv1), by(id ctr yr mn pty)

```
gen nat_pvs = pty_nat_vv1/ nat_vv1
drop if nat_pvs < .05
drop nat_vv1 pty_nat_vv1 nat_pvs
save "CLEA.list.dta", replace
*Create a shell dataset that includes all the parties/constituencies
drop pty ctr_n vv1 pv1
duplicates drop
save "CLEA.unique.constituencies.dta", replace
use "CLEA.list.dta", clear
drop cst ctr_n vv1 pv1
duplicates drop
joinby id ctr yr mn using "CLEA.unique.constituencies.dta", unmatched(both)
drop _merge
joinby id ctr yr mn cst pty using "CLEA.no.SC.dta", unmatched(master)
drop _merge release rg sub cst_n mag pty_n can pev1 vot1 ivv1 to1 cv1 cvs1 cvs1 pvs1
save "gini_lc.dta", replace
*Create party vote share (constituency) variable
gen pvs = pv1/vv1
replace pvs = 0 if pvs ==.
*gen gini placeholder
gen gini = .
*Now we use the INEQDECO command to create the Gini index using the CLEA data
egen group = group(id ctr yr mn pty) if gini == .
save, replace
su group, meanonly
forval i = 1/`r(max)' {
  ineqdec0 pvs if group == `i'
  replace gini = r(gini) if group == `i'
}
save "gini_lc.dta", replace
**cleaning**
drop ctr_n vv1 pv1 pvs
sort id ctr yr mn cst
save "gini_lc.dta", replace
```

Party Nationalization Measures - Party Level Dataset

```
*Create Party System Nationalization Score (PSNS) and Standardized Party Nationalization Score (PSNS_s)
use "clea_lc_20251015.dta", clear
replace yr=2004 if id==1792
replace mn=10 if id==1792
replace yr=1984 if id==1422
replace mn=12 if id==1422
replace yr=1991 if id==1424
```

```
replace mn=5 if id==1424
replace mn=4 if id==1425
replace ctr_n="Macedonia" if ctr_n=="North Macedonia"
replace yr=1970 if id==2308
replace mn=12 if id==2308
replace mn=2 if id==2079
replace mn=7 if id==2081
replace mn=4 if id==2326
replace mn=2 if id==2331
keep id ctr ctr_n yr mn pty_n pty
sort id ctr yr mn pty
duplicates drop
save "party.level_lc.dta", replace
use "clea_lc_20251015.dta", clear
replace yr=2004 if id==1792
replace mn=10 if id==1792
replace yr=1984 if id==1422
replace mn=12 if id==1422
replace yr=1991 if id==1424
replace mn=5 if id==1424
replace mn=4 if id==1425
replace ctr_n="Macedonia" if ctr_n=="North Macedonia"
replace yr=1970 if id==2308
replace mn=12 if id==2308
replace mn=2 if id==2079
replace mn=7 if id==2081
replace mn=4 if id==2326
replace mn=2 if id==2331
*Move data from candidate-cst level to pty-cst level
collapse (first) ctr_n vv1 pv1 if vv1 >0 & pv1 > 0, by (id ctr yr mn pty cst)
*Call in the gini data
joinby id ctr yr mn pty cst using "gini_lc.dta", unmatched(using)
drop _merge
*Gen PNS
gen PNS = 1 - gini
*Standardized PNS (PNS_s)
by id ctr yr mn cst, sort: gen districts = _n == 1
egen n_districts = sum(districts), by(id ctr yr mn)
gen power = 1/(log(n_districts))
gen PNS_s = PNS^power
drop districts n_districts power
keep id ctr yr mn pty PNS PNS_s
duplicates drop
joinby id ctr yr mn pty using "party.level_lc.dta", unmatched(using)
```

```
drop _merge
save "party.level_lc.dta", replace
*Create Bochsler's PNS_w and PNS_sw (2010)
use "clea_lc_20251015.dta", clear
replace yr=2004 if id==1792
replace mn=10 if id==1792
replace yr=1984 if id==1422
replace mn=12 if id==1422
replace yr=1991 if id==1424
replace mn=5 if id==1424
replace mn=4 if id==1425
replace ctr_n="Macedonia" if ctr_n=="North Macedonia"
replace yr=1970 if id==2308
replace mn=12 if id==2308
replace mn=2 if id==2079
replace mn=7 if id==2081
replace mn=4 if id==2326
replace mn=2 if id==2331
collapse (first) pv1 vv1 ,by (id ctr yr mn pty cst)
*Create the denominator
drop if pv1 < 0
drop if vv1 < 0
egen nat_vv1 = sum(pv1), by(id ctr yr mn)
egen pty_vv1 = sum(pv1), by(id ctr yr mn pty)
gen denominator = nat_vv1 * pty_vv1
*Create the numerator
rename vv1 cst_vv1
gen vote_share = pv1/cst_vv1
sort id ctr yr mn pty vote_share
by id ctr yr mn pty: gen p_j =sum(pv1)
gen inside = cst_vv1 * (p_j - (pv1/2))
egen numerator = sum(inside), by(id ctr yr mn pty)
gen PNS_w = (2 * numerator)/ denominator
*FIX PNS_w for countries where sum(pv1) != vv1 & -990 values lead to inaccuracies
egen cst_vv1_new = sum(pv1), by(id ctr yr mn cst)
gen vote_share_new = pv1/cst_vv1_new
sort id ctr yr mn pty vote_share_new
by id ctr yr mn pty: gen p_j_new =sum(pv1)
gen inside_new = cst_vv1_new * (p_j_new - (pv1/2))
egen numerator_new = sum(inside_new), by(id ctr yr mn pty)
gen PNS_w_new = (2 * numerator_new)/ denominator
gen alt_vv1 = 1 if PNS_w >= 1
replace PNS_w = PNS_w_new if PNS_w >= 1
replace PNS_w = -990 if pty == 3999 | pty >= 6000
```

*Setting single constituencies to missing (single constituencies cause inaccurate PNS_w scores)

bysort id ctr yr mn (cst): gen single_constituency = (cst[1] == cst[_N])

bysort id ctr yr mn (cst): gen single_constituency2 = cst[1] == 1 & cst[_N] == 1

replace PNS_w = -990 if single_constituency==1 | single_constituency2==1

drop single_constituency single_constituency2

*Also PNS_w is messed up for one cst in US in 1959, so correct

replace PNS_w = -990 if ctr == 840 & yr == 1959 & cst == 435

replace PNS_w = -990 if PNS_w == .

replace alt_vv1 = 0 if PNS_w == -990

keep id ctr yr mn cst pty cst_vv1 nat_vv1 PNS_w alt_vv1

*Create the PSNS_sw measure from the PNS_w measure

gen top = nat_vv1 * nat_vv1

gen square = cst_vv1 * cst_vv1

gen helper = .

by id ctr yr mn cst, sort: gen pid = _n

replace helper = square if pid == 1

egen bottom = sum(helper), by(id ctr yr mn)

gen power_E = top/ bottom

gen PNS_sw = (PNS_w)^(1/(log10(power_E)))

replace PNS_sw = -990 if PNS_w == -990

replace PNS_sw = -990 if PNS_sw > 1 | PNS_sw == .

keep id ctr yr mn pty PNS_w PNS_sw

duplicates drop

sort id ctr yr mn pty

joinby id ctr yr mn pty using "party.level_1c.dta", unmatched (using)

drop _merge

label var ctr_n "Country name"

label var yr "Election year"

label var mn "Election month"

label var pty_n "Party name"

label var pty "Party code"

label var ctr "CLEA country code"

label var PNS "Jones & Mainwaring Party Nationalization Score"

label var PNS_s "Bochsler standardized Party Nationalization Score"

label var PNS_w "Bochsler weighted Party Nationalization Score"

label var PNS_sw "Bochsler standarized & weighted Party Nationalization Score"

*Cleaning

replace PNS = -990 if PNS == .

replace PNS_s = -990 if PNS_s == .

replace PNS_w = -990 if PNS_w == .

replace PNS_sw = -990 if PNS_sw == .

compress

replace PNS = -990 if PNS == 1

replace PNS = -990 if pty >= 3999 & pty < 5000

```
replace PNS_s = -990 if pty >= 3999 & pty < 5000
replace PNS_w = -990 if pty >= 3999 & pty < 5000
replace PNS_sw = -990 if pty >= 3999 & pty < 5000
replace pty = 3999 if pty == -990
replace PNS = -990 if pty >= 3996 & pty < 5000
replace PNS_s = -990 if pty >= 3996 & pty < 5000
replace PNS_w = -990 if pty >= 3996 & pty < 5000
replace PNS_sw = -990 if pty >= 3996 & pty < 5000
order ctr_n id ctr yr mn pty_n pty PNS PNS_s PNS_w PNS_sw, first
sort ctr_n id ctr yr mn pty_n pty PNS PNS_s PNS_w PNS_sw
*drop all duplicates
duplicates drop id ctr yr mn pty, force
save "party.level_lc.dta", replace
```

Party Nationalization Measures - National Level Dataset

```
use "clea_lc_20251015.dta", clear
replace yr=2004 if id==1792
replace mn=10 if id==1792
replace yr=1984 if id==1422
replace mn=12 if id==1422
replace yr=1991 if id==1424
replace mn=5 if id==1424
replace mn=4 if id==1425
replace ctr_n="Macedonia" if ctr_n=="North Macedonia"
replace yr=1970 if id==2308
replace mn=12 if id==2308
replace mn=2 if id==2079
replace mn=7 if id==2081
replace mn=4 if id==2326
replace mn=2 if id==2331
keep ctr_n id ctr yr mn
duplicates drop
save "national.level_lc.dta", replace
*Create ENEP national
use "clea_lc_20251015.dta", clear
replace yr=2004 if id==1792
replace mn=10 if id==1792
replace yr=1984 if id==1422
replace mn=12 if id==1422
replace yr=1991 if id==1424
replace mn=5 if id==1424
replace mn=4 if id==1425
replace ctr_n="Macedonia" if ctr_n=="North Macedonia"
replace yr=1970 if id==2308
```

```
replace mn=12 if id==2308
replace mn=2 if id==2079
replace mn=7 if id==2081
replace mn=4 if id==2326
replace mn=2 if id==2331
*Get rid of candidate level by moving data to pty-cst level
collapse (first) vv1 pv1 if vv1 >0 & pv1 > 0, by (id ctr yr mn pty cst)
sort id ctr yr mn cst pty
*Move to pty-ctr level
collapse(sum) pv1, by (id ctr yr mn pty)
egen nat_vv1 = sum(pv1), by(id ctr yr mn)
gen party_prop_nat2 = (pv1/nat_vv1) * (pv1/nat_vv1)
egen denom = sum(party_prop_nat2), by(id ctr yr mn)
gen ENEP_nat = 1/denom
drop pty pv1 nat_vv1 party_prop_nat2 denom
duplicates drop
replace ENEP_nat = . if yr < 1834 & ctr == 840
replace ENEP_nat = . if yr == 1950 & ctr == 410
joinby id ctr yr mn using "national.level_lc.dta", unmatched (using)
drop _merge
save "national.level_lc.dta", replace
clear
*Create ENEP national average & ENEP national weighted average
use "clea_lc_20251015.dta", clear
replace yr=2004 if id==1792
replace mn=10 if id==1792
replace yr=1984 if id==1422
replace mn=12 if id==1422
replace yr=1991 if id==1424
replace mn=5 if id==1424
replace mn=4 if id==1425
replace ctr_n="Macedonia" if ctr_n=="North Macedonia"
replace yr=1970 if id==2308
replace mn=12 if id==2308
replace mn=2 if id==2079
replace mn=7 if id==2081
replace mn=4 if id==2326
replace mn=2 if id==2331
keep id ctr yr mn cst pty vv1 pv1
duplicates drop
collapse (sum) pv1 if pv1 >=0 & vv1 > 0, by (id ctr yr mn pty cst)
egen new_vv1 = sum(pv1), by (id ctr yr mn cst)
duplicates drop
save "national.level.pv1.dta", replace
```

```
use "clea_lc_20251015.dta", clear
replace yr=2004 if id==1792
replace mn=10 if id==1792
replace yr=1984 if id==1422
replace mn=12 if id==1422
replace yr=1991 if id==1424
replace mn=5 if id==1424
replace mn=4 if id==1425
replace ctr_n="Macedonia" if ctr_n=="North Macedonia"
replace yr=1970 if id==2308
replace mn=12 if id==2308
replace mn=2 if id==2079
replace mn=7 if id==2081
replace mn=4 if id==2326
replace mn=2 if id==2331
keep id ctr yr mn cst pty vv1 pv1
duplicates drop
collapse (first) vv1 if pv1 >=0 & vv1 > 0, by (id ctr yr mn pty cst)
joinby id ctr yr mn pty cst using "national.level.pv1.dta", unmatched(using)
drop _merge
gen indicator = 1 if new_vv1 != vv1
gen share2 = (pv1/new_vv1) * (pv1/new_vv1)
egen denom = sum(share2), by (id ctr yr mn cst)
gen ENEP_cst = 1/denom
drop pv1 vv1 share2 denom pty
duplicates drop
egen nat_vv1 = sum(new_vv1), by(id ctr yr mn)
gen cst_wght = new_vv1/ nat_vv1
gen weighted = cst_wght * ENEP_cst
egen ENEP_wght = sum(weighted), by(id ctr yr mn)
egen ENEP_avg = mean(ENEP_cst), by(id ctr yr mn)
drop new_vv1 ENEP_cst cst nat_vv1 cst_wght weighted
duplicates drop
collapse (first) ENEP_avg ENEP_wght (max) indicator, by (id ctr yr mn)
joinby id ctr yr mn using "national.level_lc.dta", unmatched(using)
drop _merge
order ctr ctr_n yr mn ENEP_nat ENEP_avg ENEP_wght, first
replace ENEP_nat = . if yr < 1834 & ctr == 840
replace ENEP_nat = . if yr == 1950 & ctr == 410
replace ENEP_avg = . if yr < 1834 & ctr == 840
replace ENEP_avg = . if yr == 1950 & ctr == 410
replace ENEP_wght = . if yr < 1834 & ctr == 840
replace ENEP_wght = . if yr == 1950 & ctr == 410
duplicates drop
```

```

save "national.level_lc.dta", replace
*Create Cox score (Cox)
gen Cox = (ENEP_nat - ENEP_avg)/ ENEP_nat
*Create Moenius & Kasuya Inflation score (MK_I)
gen MK_I = ( ENEP_nat - ENEP_avg)/ ENEP_avg
*Create Moenius & Kasuya Weighted Inflation score (MK_I_w)
gen MK_I_w = ( ENEP_nat - ENEP_wght)/ ENEP_wght
duplicates drop
save "national.level_lc.dta", replace
*Create composite party system nationalization measure (MK_N)
*Note: first define the values of the scalars: gamma, alpha, and beta
/*According to Kasuya and Moenius, these scalars are arbitrary and depend upon the research question at
hand.
To give equal weight to the inflation and dispersion measures, define alpha = 0.5, gamma = 0.5, and beta =
0.25.*
scalar alpha = .5
scalar beta = .25
scalar gamma = .5
use "clea_lc_20251015.dta", clear
replace yr=2004 if id==1792
replace mn=10 if id==1792
replace yr=1984 if id==1422
replace mn=12 if id==1422
replace yr=1991 if id==1424
replace mn=5 if id==1424
replace mn=4 if id==1425
replace ctr_n="Macedonia" if ctr_n=="North Macedonia"
replace yr=1970 if id==2308
replace mn=12 if id==2308
replace mn=2 if id==2079
replace mn=7 if id==2081
replace mn=4 if id==2326
replace mn=2 if id==2331
collapse (first) pv1 vv1 if pv1 >=0 & vv1 > 0, by (id ctr yr mn pty cst)
egen new_vv1 = sum(pv1), by (id ctr yr mn cst)
gen share2 = (pv1/new_vv1) * (pv1/new_vv1)
egen denom = sum(share2), by (id ctr yr mn cst)
gen ENEP_cst = 1/denom
*Create I_i
joinby id ctr yr mn using "national.level_lc.dta", unmatched (using)
drop _merge
duplicates drop
gen I_i = (( ENEP_nat- ENEP_cst)/ ENEP_cst) * 100
*Create W_tilde

```

```

egen n_districts = max(cst), by(id ctr yr mn)
egen nat_vote = sum(new_vv1), by(id ctr yr mn)
gen cst_vote_proportion = new_vv1/nat_vote
gen product = ENEP_cst * cst_vote_proportion
egen sum_cst = sum (product), by (id ctr yr mn)
gen denominator = n_districts * sum
gen W_tilde = ENEP_cst/ denominator
drop denominator
*Create I_w
gen I_w = ((ENEP_nat - sum_cst)/sum_cst)*100
*Create coefficient of variation
gen numerator = (I_i - I_w)^2 * W_tilde
egen sum_numerator = sum(numerator), by(id ctr yr mn)
gen coeff_var_I_i = sqrt(sum_numerator)/ I_w
*Create kurtosis measure
gen numerator2 = ((I_i - I_w)^4) * W_tilde
egen sum_numerator2 = sum(numerator2), by (id ctr yr mn)
gen denominator2 = ((I_i - I_w)^2) * W_tilde
egen sum_denominator2 = sum(denominator2), by (id ctr yr mn)
gen sq_sum_denominator2 = (sum_denominator2)^2
gen kurtosis_I_i = sum_numerator2/sq_sum_denominator2
*Create dispersion measure
gen D = (coeff_var_I_i)^gamma * (kurtosis_I_i)^(1 - gamma)
*Create composite measure (Inflation and Dispersion)
gen MK_N = ((I_w)^alpha) * (D^(1 - alpha))
gen MK_N_two = ((I_w)^alpha) * ((coeff_var_I_i)^beta) * ((kurtosis_I_i)^(1 - alpha - beta))
collapse (first) MK_N MK_N_two, by (id ctr yr mn)
joinby id ctr yr mn using "national.level_lc.dta", unmatched(both)
drop _merge
save "national.level_lc.dta", replace
*Create Party System Nationalization Score (PSNS) measures
use "clea_lc_20251015.dta", clear
replace yr=2004 if id==1792
replace mn=10 if id==1792
replace yr=1984 if id==1422
replace mn=12 if id==1422
replace yr=1991 if id==1424
replace mn=5 if id==1424
replace mn=4 if id==1425
replace ctr_n="Macedonia" if ctr_n=="North Macedonia"
replace yr=1970 if id==2308
replace mn=12 if id==2308
replace mn=2 if id==2079
replace mn=7 if id==2081

```

```
replace mn=4 if id==2326
replace mn=2 if id==2331
*Move data from candidate-cst level to pty-cst level
collapse (first) ctr_n vv1 pv1 if vv1 >0 & pv1 > 0, by (id ctr yr mn pty cst)
egen nat_vv1 = sum(pv1), by(id ctr yr mn)
egen pty_vv1 = sum(pv1), by(id ctr yr mn pty)
keep id ctr yr mn pty pty_vv1 nat_vv1
duplicates drop
joinby id ctr yr mn pty using "party.level_lc.dta", unmatched(using)
order ctr ctr_n yr mn pty pty_n
drop _merge
*Gen Party System Nationalization Score (PSNS)
replace PNS = . if PNS == -990
replace PNS_s = . if PNS_s == -990
replace PNS_w = . if PNS_w == -990
replace PNS_sw = . if PNS_sw == -990
gen weight = pty_vv1/nat_vv1
egen PSNS = sum(PNS * weight), by(id ctr yr mn)
egen PSNS_s = sum(PNS_s * weight), by(id ctr yr mn)
egen PSNS_w = sum(PNS_w * weight), by(id ctr yr mn)
egen PSNS_sw = sum(PNS_sw * weight), by(id ctr yr mn)
keep id ctr ctr_n yr mn PSNS PSNS_s PSNS_w PSNS_sw
duplicates drop
replace PSNS = -990 if PSNS == 0
replace PSNS_s = -990 if PSNS_s == 0
replace PSNS_w = -990 if PSNS_w == 0
replace PSNS_sw = -990 if PSNS_sw == 0
replace PSNS = -990 if PSNS > 1
replace PSNS_s = -990 if PSNS_s > 1
replace PSNS_sw = -990 if PSNS_sw > 1
replace PSNS_w = -990 if PSNS_w > 1
joinby id ctr yr mn using "national.level_lc.dta", unmatched (both)
drop _merge
save "national.level_lc.dta", replace
*Create Local Entrant Measure E
use "clea_lc_20251015.dta", clear
replace yr=2004 if id==1792
replace mn=10 if id==1792
replace yr=1984 if id==1422
replace mn=12 if id==1422
replace yr=1991 if id==1424
replace mn=5 if id==1424
replace mn=4 if id==1425
replace ctr_n="Macedonia" if ctr_n=="North Macedonia"
```

```
replace yr=1970 if id==2308
replace mn=12 if id==2308
replace mn=2 if id==2079
replace mn=7 if id==2081
replace mn=4 if id==2326
replace mn=2 if id==2331
collapse (first) ctr_n vv1 pv1 (sum)seat cv1 if seat >= 0 & vv1 >0 & pv1 > 0, by (id ctr yr mn pty cst)
egen nat_vote = sum(pv1) if pv1>= 0, by (id ctr yr mn)
egen seat_cst = sum(seat), by (id ctr yr mn cst)
collapse (first) ctr_n vv1 pv1 nat_vote seat_cst (sum) cv1 seat if vv1 >0 & pv1 > 0 & seat >= 0, by (id ctr yr mn pty cst)
sort id ctr yr mn pty
gen party_vote_proportion = pv1/ nat_vote
egen seat_total = sum(seat), by (id ctr yr mn)
egen seat_contest = sum(seat_cst), by (id ctr yr mn pty)
gen seat_proportion = seat_contest/seat_total
gen local_E = party_vote_proportion * seat_proportion
collapse (sum) local_E, by(id ctr yr mn)
joinby id ctr yr mn using "national.level_lc.dta", unmatched(both)
drop _merge
drop MK_N_two
label var ctr_n "Country name"
label var ctr "CLEA country code"
label var yr "Election year"
label var mn "Election month"
label var ENEP_nat "Effective Number of Electoral Parties nationally"
label var ENEP_avg "Average Effective Number of Electoral Parties nationally"
label var ENEP_wght "Weighted average of Effective Number of Electoral Parties nationally"
label var Cox "Cox inflation score"
label var MK_I "Moenius & Kasuya inflation score"
label var MK_I_w "Moenius & Kasuya weighted inflation score"
label var MK_N "Moenius & Kasuya composite measure"
label var PSNS "Jones & Mainwaring Party System Nationalization Score"
label var PSNS_s "Bochsler standardized Party System Nationalization Score"
label var PSNS_w "Bochsler weighted Party System Nationalization Score"
label var PSNS_sw "Bochsler standarized & weighted Party System Nationalization Score"
label var local_E "Local entrant measure"
label var indicator "National Valid Vote Indicator"
compress
save "national.level_lc.dta", replace
*Cleaning
rename Cox inflation1
rename MK_I inflation2
rename MK_I_w inflation3
```

```

rename MK_N inflation4
rename indicator nvvi
replace nvvi = 0 if nvvi ==.
replace ENEP_nat = -990 if ENEP_nat ==.
replace ENEP_avg = -990 if ENEP_avg ==.
replace ENEP_wght = -990 if ENEP_wght ==.
replace inflation1 = -990 if inflation1 ==.
replace inflation2 = -990 if inflation2 ==.
replace inflation3 = -990 if inflation3 ==.
replace inflation4 = -990 if inflation4 ==.
replace PSNS = -990 if PSNS ==.
replace PSNS_s = -990 if PSNS_s ==.
replace PSNS_w = -990 if PSNS_w ==.
replace PSNS_sw = -990 if PSNS_sw ==.
replace local_E = -990 if local_E ==.
rename ENEP_nat ENP_nat
rename ENEP_avg ENP_avg
rename ENEP_wght ENP_wght
replace ENP_nat = -990 if ctr == 144 & yr == 1947 | ctr == 144 & yr == 1952 | ctr == 144 & yr == 1956 |
ctr == 144 & yr == 1960 | ctr == 144 & yr == 1965 | ctr == 144 & yr == 1970 | ctr == 144 & yr == 1977
replace ENP_avg = -990 if ctr == 144 & yr == 1947 | ctr == 144 & yr == 1952 | ctr == 144 & yr == 1956 |
ctr == 144 & yr == 1960 | ctr == 144 & yr == 1965 | ctr == 144 & yr == 1970 | ctr == 144 & yr == 1977
replace ENP_wght = -990 if ctr == 144 & yr == 1947 | ctr == 144 & yr == 1952 | ctr == 144 & yr == 1956
| ctr == 144 & yr == 1960 | ctr == 144 & yr == 1965 | ctr == 144 & yr == 1970 | ctr == 144 & yr == 1977
replace inflation1 = -990 if inflation1 == 0
replace inflation2 = -990 if inflation2 == 0
replace inflation3 = -990 if inflation3 == 0
replace inflation4 = -990 if inflation4 == 0
order ctr_n id ctr yr mn nvvi ENP_nat ENP_avg ENP_wght inflation1 inflation2 inflation3 inflation4 PSNS
PSNS_s PSNS_w PSNS_sw local_E, first
sort ctr_n id ctr yr mn nvvi ENP_nat ENP_avg ENP_wght inflation1 inflation2 inflation3 inflation4 PSNS
PSNS_s PSNS_w PSNS_sw local_E
*drop all duplicates
duplicates drop id ctr yr mn, force
save "national.level_lc.dta", replace

***Code below cleans the national level data by excluding the following observations (comment out the
lines below to include them)

// Drop elections in which independents or non-identifiable candidates won more than 40% of the vote
use "clea_lc_20251015.dta", clear
replace yr=2004 if id==1792
replace mn=10 if id==1792
replace yr=1984 if id==1422
replace mn=12 if id==1422

```

```
replace yr=1991 if id==1424
replace mn=5 if id==1424
replace mn=4 if id==1425
replace ctr_n="Macedonia" if ctr_n=="North Macedonia"
replace yr=1970 if id==2308
replace mn=12 if id==2308
replace mn=2 if id==2079
replace mn=7 if id==2081
replace mn=4 if id==2326
replace mn=2 if id==2331

preserve

replace pty=9999 if pty==-990
keep if pty>=6000
replace pv1=. if pv1<0
replace cv1=. if cv1<0
replace cv1=pv1 if cv1==.
egen totindv1=sum(cv1), by(id ctr_n yr mn)
keep id ctr_n yr mn totindv1
duplicates drop
save "indv.dta", replace

restore

keep id ctr_n yr mn cst vv1
replace vv1=. if vv1<0
duplicates drop
egen totv1=sum(vv1), by(id ctr_n yr mn)
drop cst vv1
duplicates drop

merge 1:1 id ctr_n yr mn using indv.dta
drop if ctr_n=="United States"
drop if id==.
replace totindv1=0 if totindv1==.
gen indvs=totindv1/totv1
gen drop_id=0
replace drop_id=1 if indvs>=0.4
keep if drop_id==1
drop if id==1622 // Calculations off due to many single constituencies
keep id ctr_n yr mn drop_id
save "indv.dta", replace

use "national.level_lc.dta"
```

```
merge 1:1 id ctr_n yr mn using indiv.dta
drop if drop_id==1
drop drop_id _merge

// Drop US elections that have less than 200 constituencies and non-regular US elections.
gen odd_yr=yr/2
drop if ctr_n=="United States" & yr<1876
drop if ctr_n=="United States" & mn!=11
drop if ctr_n=="United States" & odd_yr!=floor(odd_yr)
drop odd_yr

// Drop observations for which no nationalization scores can be calculated
drop if ENP_nat== -990

sort ctr_n yr mn id
save "national.level_lc.dta", replace
```

Party Nationalization Measures - Constituency Level Dataset

/*Note: this part of the file uses CLEA data to create several alternate measures of party nationalization identified in the literature */

*Create frame dataset

```
use "clea_lc_20251015.dta", clear
replace yr=2004 if id==1792
replace mn=10 if id==1792
replace yr=1984 if id==1422
replace mn=12 if id==1422
replace yr=1991 if id==1424
replace mn=5 if id==1424
replace mn=4 if id==1425
replace ctr_n="Macedonia" if ctr_n=="North Macedonia"
replace yr=1970 if id==2308
replace mn=12 if id==2308
replace mn=2 if id==2079
replace mn=7 if id==2081
replace mn=4 if id==2326
replace mn=2 if id==2331
keep ctr_n id ctr yr mn cst_n cst
duplicates drop
save "cst.level_lc.dta", replace
```

*Create ENP constituency

```
use "clea_lc_20251015.dta", clear
replace yr=2004 if id==1792
replace mn=10 if id==1792
replace yr=1984 if id==1422
```

```

replace mn=12 if id==1422
replace yr=1991 if id==1424
replace mn=5 if id==1424
replace mn=4 if id==1425
replace ctr_n="Macedonia" if ctr_n=="North Macedonia"
replace yr=1970 if id==2308
replace mn=12 if id==2308
replace mn=2 if id==2079
replace mn=7 if id==2081
replace mn=4 if id==2326
replace mn=2 if id==2331
collapse (first) pv1 vv1 if pv1 >=0 & vv1 > 0, by (id ctr yr mn pty cst cst_n)
egen new_vv1 = sum(pv1), by (id ctr yr mn cst)
gen share2 = (pv1/new_vv1) * (pv1/new_vv1)
egen denom = sum(share2), by (id ctr yr mn cst)
gen ENP_cst = 1/denom
gen cvvi = 1 if vv1 != new_vv1
drop share2 denom pv1 vv1 new_vv1 pty
duplicates drop
joinby id ctr yr mn using "national.level_lc.dta", unmatched (using)
drop _merge
save "cst.level_lc.dta", replace
*Add national level data
joinby id ctr yr mn using "national.level_lc.dta", unmatched (master)
drop _merge
save "cst.level_lc.dta", replace
compress
gen inflation5 = (ENP_nat - ENP_cst)/ENP_cst if ENP_nat != -990 & ENP_cst != -990
replace inflation5 = -990 if inflation5 ==.
duplicates drop
order ctr_n id ctr yr mn cst_n cst nvvi cvvi ENP_cst ENP_nat ENP_avg ENP_wght inflation1 inflation2
inflation3 inflation4 inflation5 PSNS PSNS_s PSNS_w PSNS_sw local_E
sort ctr_n id ctr yr mn cst_n cst nvvi cvvi ENP_cst ENP_nat ENP_avg ENP_wght inflation1 inflation2
inflation3 inflation4 inflation5 PSNS PSNS_s PSNS_w PSNS_sw local_E
label var ctr_n "Country Name"
label var ENP_cst "Effective Number of Electoral Parties at the constituency level"
label var inflation5 "Moenius & Kasuya local inflation measure"
label var cvvi "Constituency Valid Vote Indicator"
replace ENP_cst = -990 if ENP_cst ==.
replace cvvi = 0 if cvvi ==.
replace ENP_nat = -990 if ENP_nat ==.
replace ENP_avg = -990 if ENP_avg ==.
replace ENP_wght = -990 if ENP_wght ==.
replace inflation1 = -990 if inflation1 ==.

```

```
replace inflation2 = -990 if inflation2 ==.  
replace inflation3 = -990 if inflation3 ==.  
replace inflation4 = -990 if inflation4 ==.  
replace inflation5 = -990 if inflation5 ==.  
replace PSNS = -990 if PSNS ==.  
replace PSNS_s = -990 if PSNS_s ==.  
replace PSNS_w = -990 if PSNS_w ==.  
replace PSNS_sw = -990 if PSNS_sw ==.  
replace local_E = -990 if local_E ==.  
replace ENP_cst = -990 if yr < 1834 & ctr == 840  
*remove all duplicates  
duplicates drop ctr yr cst mn, force  
save "cst.level_lc.dta", replace  
  
** Country name corrections  
use "CLEA.list.dta", clear  
replace ctr_n="North Macedonia" if ctr_n=="Macedonia" & yr>=2020  
save "CLEA.list.dta", replace  
  
use "CLEA.no.SC.dta", clear  
replace ctr_n="North Macedonia" if ctr_n=="Macedonia" & yr>=2020  
save "CLEA.no.SC.dta", replace  
  
use "cst.level_lc.dta", clear  
replace ctr_n="North Macedonia" if ctr_n=="Macedonia" & yr>=2020  
save "cst.level_lc.dta", replace  
  
use "national.level_lc.dta", clear  
replace ctr_n="North Macedonia" if ctr_n=="Macedonia" & yr>=2020  
save "national.level_lc.dta", replace  
  
use "party.level_lc.dta", clear  
replace ctr_n="North Macedonia" if ctr_n=="Macedonia" & yr>=2020  
save "party.level_lc.dta", replace
```

Stata Code (Upper Chamber)

Data Prep

```
import delimited "clea_uc_19.csv", clear

**Merge elections that span multiple months**
replace mn=10 if id==2 // Argentina 2003

replace yr=2007 if id==9 // Bhutan 2007
replace mn=12 if id==9 // Bhutan 2007

replace mn=11 if id==48 // United States 1916
replace mn=11 if id==49 // United States 1918
replace mn=11 if id==50 // United States 1920
replace mn=11 if id==54 // United States 1928
replace mn=11 if id==55 // United States 1930
replace mn=11 if id==57 // United States 1934
replace mn=11 if id==61 // United States 1942
replace mn=11 if id==63 // United States 1946
replace mn=11 if id==64 // United States 1948
replace mn=11 if id==66 // United States 1952
replace mn=11 if id==67 // United States 1954
replace yr=1958 if id==69 // United States 1958
replace mn=11 if id==69 // United States 1958
replace mn=11 if id==79 // United States 1978
replace mn=11 if id==80 // United States 1980
replace mn=11 if id==82 // United States 1984
replace mn=11 if id==83 // United States 1986
replace mn=11 if id==85 // United States 1990
replace mn=11 if id==86 // United States 1992
replace mn=11 if id==88 // United States 1996

replace mn=8 if id==238 // Haiti 2015

**Drop Czechia 2025 by-election
drop if id==236

**Save
save "clea_uc_19_calc.dta", replace
```

Gini Index for Parties

/*This file creates the Gini index for CLEA data, indicating the inequality in vote shares received by parties in districts. This information is used for calculating Jones and Mainwaring's (2003) Party Nationalization Score (PNS).*/

*To create the Gini index, we use the command INEQDEC0. The code below installs the command.

```
*ssc inst ineqdec0, replace
use "clea_uc_19_calc.dta", clear

*Create Gini using CLEA data
*Eliminate single constituency elections
bysort id ctr yr mn (cst): gen single_constituency = cst[1] == 1 & cst[_N] == 1
drop if single_constituency==1
save "CLEA.no.SC.dta", replace
*Move the data from candidate-cst level to party-cst level
collapse (first) ctr_n vv1 pv1 if vv1 >0 & pv1 > 0, by (id ctr yr mn pty cst)
*Drop observations where the party wins less than five percent of the national party vote
egen nat_vv1 = sum(pv1), by(id ctr yr mn)
egen pty_nat_vv1 = sum(pv1), by(id ctr yr mn pty)
gen nat_pvs = pty_nat_vv1/ nat_vv1
drop if nat_pvs < .05
drop nat_vv1 pty_nat_vv1 nat_pvs
save "CLEA.list.dta", replace
*Create a shell dataset that includes all the parties/constituencies
drop pty ctr_n vv1 pv1
duplicates drop
save "CLEA.unique.constituencies.dta", replace
use "CLEA.list.dta", clear
drop cst ctr_n vv1 pv1
duplicates drop
joinby id ctr yr mn using "CLEA.unique.constituencies.dta", unmatched(both)
drop _merge
joinby id ctr yr mn cst pty using "CLEA.no.SC.dta", unmatched(master)
drop _merge release rg sub cst_n mag pty_n can pev1 vot1 ivv1 to1 cv1 cvs1 cvs1 pvs1
save "gini_uc.dta", replace
*Create party vote share (constituency) variable
gen pvs = pv1/vv1
replace pvs = 0 if pvs == .
*gen gini placeholder
gen gini = .
*Now we use the INEQDECO command to create the Gini index using the CLEA data
egen group = group(id ctr yr mn pty) if gini == .
sleep 1000 // give enough time to finishing saving
save, replace
su group, meanonly
forval i = 1/`r(max)' {
  ineqdec0 pvs if group == `i'
  replace gini = r(gini) if group == `i'
}
save "gini_uc.dta", replace
**cleaning**
drop ctr_n vv1 pv1 pvs
sort id ctr yr mn cst
```

```
save "gini_uc.dta", replace
```

Party Nationalization Measures - Party Level Dataset

```
*Create Party System Nationalization Score (PSNS) and Standardized Party Nationalization Score (PSNS_s)
```

```
use "clea_uc_19_calc.dta", clear
```

```
keep id ctr ctr_n yr mn pty_n pty
```

```
sort id ctr yr mn pty
```

```
duplicates drop
```

```
save "party.level_uc.dta", replace
```

```
use "clea_uc_19_calc.dta", clear
```

```
*Move data from candidate-cst level to pty-cst level
```

```
collapse (first) ctr_n vv1 pv1 if vv1 > 0 & pv1 > 0, by (id ctr yr mn pty cst)
```

```
*Call in the gini data
```

```
joinby id ctr yr mn pty cst using "gini_uc.dta", unmatched(using)
```

```
drop _merge
```

```
*Gen PNS
```

```
gen PNS = 1 - gini
```

```
*Standardized PNS (PNS_s)
```

```
by id ctr yr mn cst, sort: gen districts = _n == 1
```

```
egen n_districts = sum(districts), by(id ctr yr mn)
```

```
gen power = 1/(log(n_districts))
```

```
gen PNS_s = PNS^power
```

```
drop districts n_districts power
```

```
keep id ctr yr mn pty PNS PNS_s
```

```
duplicates drop
```

```
joinby id ctr yr mn pty using "party.level_uc.dta", unmatched(using)
```

```
drop _merge
```

```
save "party.level_uc.dta", replace
```

```
*Create Bochsler's PNS_w and PNS_sw (2010)
```

```
use "clea_uc_19_calc.dta", clear
```

```
collapse (first) pv1 vv1, by (id ctr yr mn pty cst)
```

```
*Create the denominator
```

```
drop if pv1 < 0
```

```
drop if vv1 < 0
```

```
egen nat_vv1 = sum(pv1), by(id ctr yr mn)
```

```
egen pty_vv1 = sum(pv1), by(id ctr yr mn pty)
```

```
gen denominator = nat_vv1 * pty_vv1
```

```
*Create the numerator
```

```
rename vv1 cst_vv1
```

```
gen vote_share = pv1/cst_vv1
```

```
sort id ctr yr mn pty vote_share
```

```
by id ctr yr mn pty: gen p_j = sum(pv1)
```

```

gen inside = cst_vv1 * (p_j - (pv1/2))
egen numerator = sum(inside), by(id ctr yr mn pty)
gen PNS_w = (2 * numerator)/ denominator
*FIX PNS_w for countries where sum(pv1) != vv1 & -990 values lead to inaccuracies
egen cst_vv1_new = sum(pv1), by(id ctr yr mn cst)
gen vote_share_new = pv1/cst_vv1_new
sort id ctr yr mn pty vote_share_new
by id ctr yr mn pty: gen p_j_new =sum(pv1)
gen inside_new = cst_vv1_new * (p_j_new - (pv1/2))
egen numerator_new = sum(inside_new), by(id ctr yr mn pty)
gen PNS_w_new = (2 * numerator_new)/ denominator
gen alt_vv1 = 1 if PNS_w >= 1
replace PNS_w = PNS_w_new if PNS_w >= 1
replace PNS_w = -990 if pty == 3999 | pty >= 6000
*Setting single constituencies to missing (single constituencies cause inaccurate PNS_w scores)
bysort id ctr yr mn (cst): gen single_constituency = cst[1] == 1 & cst[_N] == 1
replace PNS_w = -990 if single_constituency==1
drop single_constituency
*Also PNS_w is messed up for one cst in US in 1959, so correct
replace PNS_w = -990 if ctr == 840 & yr == 1959 & cst ==435
replace PNS_w = -990 if PNS_w ==.
replace alt_vv1 = 0 if PNS_w == -990
keep id ctr yr mn cst pty cst_vv1 nat_vv1 PNS_w alt_vv1
*Create the PSNS_sw measure from the PNS_w measure
gen top = nat_vv1 * nat_vv1
gen square = cst_vv1 * cst_vv1
gen helper = .
by id ctr yr mn cst, sort: gen pid = _n
replace helper = square if pid == 1
egen bottom = sum(helper), by(id ctr yr mn)
gen power_E = top/ bottom
gen PNS_sw = (PNS_w)^(1/(log10(power_E)))
replace PNS_sw = -990 if PNS_w == -990
replace PNS_sw = -990 if PNS_sw >1 | PNS_sw ==.
keep id ctr yr mn pty PNS_w PNS_sw
duplicates drop
sort id ctr yr mn pty
joinby id ctr yr mn pty using "party.level_uc.dta", unmatched (using)
drop _merge
label var ctr_n "Country name"
label var yr "Election year"
label var mn "Election month"
label var pty_n "Party name"
label var pty "Party code"
label var ctr "CLEA country code"
label var PNS "Jones & Mainwaring Party Nationalization Score"

```

```

label var PNS_s "Bochsler standardized Party Nationalization Score"
label var PNS_w "Bochsler weighted Party Nationalization Score"
label var PNS_sw "Bochsler standarized & weighted Party Nationalization Score"
*Cleaning
replace PNS = -990 if PNS ==.
replace PNS_s = -990 if PNS_s ==.
replace PNS_w = -990 if PNS_w ==.
replace PNS_sw = -990 if PNS_sw ==.
compress
replace PNS = -990 if PNS == 1
replace PNS = -990 if pty >= 3999 & pty < 5000
replace PNS_s = -990 if pty >= 3999 & pty < 5000
replace PNS_w = -990 if pty >= 3999 & pty < 5000
replace PNS_sw = -990 if pty >= 3999 & pty < 5000
replace pty = 3999 if pty == -990
replace PNS = -990 if pty >= 3996 & pty < 5000
replace PNS_s = -990 if pty >= 3996 & pty < 5000
replace PNS_w = -990 if pty >= 3996 & pty < 5000
replace PNS_sw = -990 if pty >= 3996 & pty < 5000
order ctr_n id ctr yr mn pty_n pty PNS PNS_s PNS_w PNS_sw, first
sort ctr_n id ctr yr mn pty_n pty PNS PNS_s PNS_w PNS_sw
*drop all duplicates
duplicates drop id ctr yr mn pty, force
save "party.level_uc.dta", replace

```

Party Nationalization Measures - National Level Dataset

```

use "clea_uc_19_calc.dta", clear

keep ctr_n id ctr yr mn
duplicates drop
save "national.level_uc.dta", replace
use "clea_uc_19_calc.dta", clear

*Get rid of candidate level by moving data to pty-cst level
collapse (first) vv1 pv1 if vv1 >0 & pv1 > 0, by (id ctr yr mn pty cst)
sort id ctr yr mn cst pty
*Move to pty-ctr level
collapse(sum) pv1, by (id ctr yr mn pty)
egen nat_vv1 = sum(pv1), by(id ctr yr mn)
gen party_prop_nat2 = (pv1/nat_vv1) * (pv1/nat_vv1)
egen denom = sum(party_prop_nat2), by(id ctr yr mn)
gen ENEP_nat = 1/denom
drop pty pv1 nat_vv1 party_prop_nat2 denom
duplicates drop
replace ENEP_nat = . if yr < 1834 & ctr == 840
replace ENEP_nat = . if yr == 1950 & ctr == 410

```

```
joinby id ctr yr mn using "national.level_uc.dta", unmatched(using)
drop _merge
save "national.level_uc.dta", replace
clear
*Create ENEP national average & ENEP national weighted average
use "clea_uc_19_calc.dta", clear

keep id ctr yr mn cst pty vv1 pv1
duplicates drop
collapse (sum) pv1 if pv1 >=0 & vv1 > 0, by (id ctr yr mn pty cst)
egen new_vv1 = sum(pv1), by (id ctr yr mn cst)
duplicates drop
save "national.level.pv1.dta", replace
use "clea_uc_19_calc.dta", clear
keep id ctr yr mn cst pty vv1 pv1
duplicates drop
collapse (first) vv1 if pv1 >=0 & vv1 > 0, by (id ctr yr mn pty cst)
joinby id ctr yr mn pty cst using "national.level.pv1.dta", unmatched(using)
drop _merge
gen indicator = 1 if new_vv1 != vv1
gen share2 = (pv1/new_vv1) * (pv1/new_vv1)
egen denom = sum(share2), by (id ctr yr mn cst)
gen ENEP_cst = 1/denom
drop pv1 vv1 share2 denom pty
duplicates drop
egen nat_vv1 = sum(new_vv1), by(id ctr yr mn)
gen cst_wght = new_vv1/ nat_vv1
gen weighted = cst_wght * ENEP_cst
egen ENEP_wght = sum(weighted), by(id ctr yr mn)
egen ENEP_avg = mean(ENEP_cst), by(id ctr yr mn)
drop new_vv1 ENEP_cst cst nat_vv1 cst_wght weighted
duplicates drop
collapse (first) ENEP_avg ENEP_wght (max) indicator, by (id ctr yr mn)
joinby id ctr yr mn using "national.level_uc.dta", unmatched(using)
drop _merge
order ctr ctr_n yr mn ENEP_nat ENEP_avg ENEP_wght, first
replace ENEP_nat = . if yr < 1834 & ctr == 840
replace ENEP_nat = . if yr == 1950 & ctr == 410
replace ENEP_avg = . if yr < 1834 & ctr == 840
replace ENEP_avg = . if yr == 1950 & ctr == 410
replace ENEP_wght = . if yr < 1834 & ctr == 840
replace ENEP_wght = . if yr == 1950 & ctr == 410
duplicates drop
save "national.level_uc.dta", replace
*Create Cox score (Cox)
gen Cox = (ENEP_nat - ENEP_avg)/ ENEP_nat
*Create Moenius & Kasuya Inflation score (MK_I)
```

```

gen MK_I = ( ENEP_nat - ENEP_avg)/ ENEP_avg
*Create Moenius & Kasuya Weighted Inflation score (MK_I_w)
gen MK_I_w = ( ENEP_nat - ENEP_wght)/ ENEP_wght
duplicates drop
save "national.level_uc.dta", replace
*Create composite party system nationalization measure (MK_N)
*Note: first define the values of the scalars: gamma, alpha, and beta
/*According to Kasuya and Moenius, these scalars are arbitrary and depend upon the research
question at hand.
To give equal weight to the inflation and dispersion measures, define alpha = 0.5, gamma = 0.5,
and beta = 0.25.*/
scalar alpha = .5
scalar beta = .25
scalar gamma = .5
use "clea_uc_19_calc.dta", clear
collapse (first) pv1 vv1 if pv1 >=0 & vv1 > 0, by (id ctr yr mn pty cst)
egen new_vv1 = sum(pv1), by (id ctr yr mn cst)
gen share2 = (pv1/new_vv1) * (pv1/new_vv1)
egen denom = sum(share2), by (id ctr yr mn cst)
gen ENEP_cst = 1/denom
*Create I_i
joinby id ctr yr mn using "national.level_uc.dta", unmatched (using)
drop _merge
duplicates drop
gen I_i = (( ENEP_nat- ENEP_cst)/ ENEP_cst) * 100
*Create W_tilde
egen n_districts = max(cst), by(id ctr yr mn)
egen nat_vote = sum(new_vv1), by(id ctr yr mn)
gen cst_vote_proportion = new_vv1/nat_vote
gen product = ENEP_cst * cst_vote_proportion
egen sum_cst = sum (product), by (id ctr yr mn)
gen denominator = n_districts * sum
gen W_tilde = ENEP_cst/ denominator
drop denominator
*Create I_w
gen I_w = ((ENEP_nat - sum_cst)/sum_cst)*100
*Create coefficient of variation
gen numerator = (I_i - I_w)^2 * W_tilde
egen sum_numerator = sum(numerator), by(id ctr yr mn)
gen coeff_var_I_i = sqrt(sum_numerator)/ I_w
*Create kurtosis measure
gen numerator2 = ((I_i - I_w)^4) * W_tilde
egen sum_numerator2 = sum(numerator2), by (id ctr yr mn)
gen denominator2 = ((I_i - I_w)^2) * W_tilde
egen sum_denominator2 = sum(denominator2), by (id ctr yr mn)
gen sq_sum_denominator2 = (sum_denominator2)^2

```

```

gen kurtosis_I_i = sum_numerator2/sq_sum_denominator2
*Create dispersion measure
gen D = (coeff_var_I_i)^gamma * (kurtosis_I_i)^(1 - gamma)
*Create composite measure (Inflation and Dispersion)
gen MK_N = ((I_w)^alpha) * (D^(1 - alpha))
gen MK_N_two = ((I_w)^alpha) * ((coeff_var_I_i)^beta) * ((kurtosis_I_i)^(1 - alpha - beta))
collapse (first) MK_N MK_N_two, by (id ctr yr mn)
joinby id ctr yr mn using "national.level_uc.dta", unmatched(both)
drop _merge
save "national.level_uc.dta", replace
*Create Party System Nationalization Score (PSNS) measures
use "clea_uc_19_calc.dta", clear
*Move data from candidate-cst level to pty-cst level
collapse (first) ctr_n vv1 pv1 if vv1 > 0 & pv1 > 0, by (id ctr yr mn pty cst)
egen nat_vv1 = sum(pv1), by(id ctr yr mn)
egen pty_vv1 = sum(pv1), by(id ctr yr mn pty)
keep id ctr yr mn pty pty_vv1 nat_vv1
duplicates drop
joinby id ctr yr mn pty using "party.level_uc.dta", unmatched(using)
order ctr ctr_n yr mn pty pty_n
drop _merge
*Gen Party System Nationalization Score (PSNS)
replace PNS = . if PNS == -990
replace PNS_s = . if PNS_s == -990
replace PNS_w = . if PNS_w == -990
replace PNS_sw = . if PNS_sw == -990
gen weight = pty_vv1/nat_vv1
egen PSNS = sum(PNS * weight), by(id ctr yr mn)
egen PSNS_s = sum(PNS_s * weight), by(id ctr yr mn)
egen PSNS_w = sum(PNS_w * weight), by(id ctr yr mn)
egen PSNS_sw = sum(PNS_sw * weight), by(id ctr yr mn)
keep id ctr ctr_n yr mn PSNS PSNS_s PSNS_w PSNS_sw
duplicates drop
replace PSNS = -990 if PSNS == 0
replace PSNS_s = -990 if PSNS_s == 0
replace PSNS_w = -990 if PSNS_w == 0
replace PSNS_sw = -990 if PSNS_sw == 0
replace PSNS = -990 if PSNS > 1
replace PSNS_s = -990 if PSNS_s > 1
replace PSNS_w = -990 if PSNS_w > 1
replace PSNS_sw = -990 if PSNS_sw > 1
joinby id ctr yr mn using "national.level_uc.dta", unmatched (both)
drop _merge
save "national.level_uc.dta", replace
*Create Local Entrant Measure E
use "clea_uc_19_calc.dta", clear

```

```
collapse (first) ctr_n vv1 pv1 (sum)seat cv1 if seat >= 0 & vv1 >0 & pv1 > 0, by (id ctr yr mn pty
cst)
egen nat_vote = sum(pv1) if pv1>= 0, by (id ctr yr mn)
egen seat_cst = sum(seat), by (id ctr yr mn cst)
collapse (first) ctr_n vv1 pv1 nat_vote seat_cst (sum) cv1 seat if vv1 >0 & pv1 > 0 & seat >= 0,
by (id ctr yr mn pty cst)
sort id ctr yr mn pty
gen party_vote_proportion = pv1/ nat_vote
egen seat_total = sum(seat), by (id ctr yr mn)
egen seat_contest = sum(seat_cst), by (id ctr yr mn pty)
gen seat_proportion = seat_contest/seat_total
gen local_E = party_vote_proportion * seat_proportion
collapse (sum) local_E, by(id ctr yr mn)
joinby id ctr yr mn using "national.level_uc.dta", unmatched(both)
drop _merge
drop MK_N_two
label var ctr_n "Country name"
label var ctr "CLEA country code"
label var yr "Election year"
label var mn "Election month"
label var ENEP_nat "Effective Number of Electoral Parties nationally"
label var ENEP_avg "Average Effective Number of Electoral Parties nationally"
label var ENEP_wght "Weighted average of Effective Number of Electoral Parties nationally"
label var Cox "Cox inflation score"
label var MK_I "Moenius & Kasuya inflation score"
label var MK_I_w "Moenius & Kasuya weighted inflation score"
label var MK_N "Moenius & Kasuya composite measure"
label var PSNS "Jones & Mainwaring Party System Nationalization Score"
label var PSNS_s "Bochsler standardized Party System Nationalization Score"
label var PSNS_w "Bochsler weighted Party System Nationalization Score"
label var PSNS_sw "Bochsler standarized & weighted Party System Nationalization Score"
label var local_E "Local entrant measure"
label var indicator "National Valid Vote Indicator"
compress
save "national.level_uc.dta", replace
*Cleaning
rename Cox inflation1
rename MK_I inflation2
rename MK_I_w inflation3
rename MK_N inflation4
rename indicator nvvi
replace nvvi = 0 if nvvi ==.
replace ENEP_nat = -990 if ENEP_nat ==.
replace ENEP_avg = -990 if ENEP_avg ==.
replace ENEP_wght = -990 if ENEP_wght ==.
replace inflation1 = -990 if inflation1 ==.
```

```
replace inflation2 = -990 if inflation2 == .
replace inflation3 = -990 if inflation3 == .
replace inflation4 = -990 if inflation4 == .
replace PSNS = -990 if PSNS == .
replace PSNS_s = -990 if PSNS_s == .
replace PSNS_w = -990 if PSNS_w == .
replace PSNS_sw = -990 if PSNS_sw == .
replace local_E = -990 if local_E == .
rename ENEP_nat ENP_nat
rename ENEP_avg ENP_avg
rename ENEP_wght ENP_wght
replace inflation1 = -990 if inflation1 == 0
replace inflation2 = -990 if inflation2 == 0
replace inflation3 = -990 if inflation3 == 0
replace inflation4 = -990 if inflation4 == 0
order ctr_n id ctr yr mn nvvi ENP_nat ENP_avg ENP_wght inflation1 inflation2 inflation3
inflation4 PSNS PSNS_s PSNS_w PSNS_sw local_E, first
sort ctr_n id ctr yr mn nvvi ENP_nat ENP_avg ENP_wght inflation1 inflation2 inflation3
inflation4 PSNS PSNS_s PSNS_w PSNS_sw local_E
*drop all duplicates
duplicates drop id ctr yr mn, force
save "national.level_uc.dta", replace

***Code below cleans the national level data by excluding the following observations (comment
out the lines below to include them)

// Drop elections in which independents or non-identifiable candidates won more than 40% of the
vote
use "clea_uc_19_calc.dta", clear

preserve

replace pty=9999 if pty==-990
keep if pty>=6000
replace pv1=. if pv1<0
replace cv1=. if cv1<0
replace cv1=pv1 if cv1==.
egen totindv1=sum(cv1), by(id ctr_n yr mn)
keep id ctr_n yr mn totindv1
duplicates drop
save "indv.dta", replace

restore

keep id ctr_n yr mn cst vv1
duplicates drop
egen totv1=sum(vv1), by(id ctr_n yr mn)
```

```
drop cst vv1
duplicates drop

merge 1:1 id ctr_n yr mn using indiv.dta
drop if ctr_n=="US"
drop if id==.
replace totindv1=0 if totindv1==.
gen indvs=totindv1/totv1
gen drop_id=0
replace drop_id=1 if indvs>=0.4
keep if drop_id==1
drop if id==114 // Calculations off due to missing data
keep id ctr_n yr mn drop_id
save "indv.dta", replace

use "national.level_uc.dta"

merge 1:1 id ctr_n yr mn using indiv.dta
drop if drop_id==1
drop drop_id _merge

// Drop US elections that have less than 200 constituencies and non-regular US elections.
gen odd_yr=yr/2
drop if ctr_n=="US" & yr<1914
drop if ctr_n=="US" & mn!=11
drop if ctr_n=="US" & odd_yr!=floor(odd_yr)
drop odd_yr

// Drop observations for which no nationalization scores can be calculated
drop if ENP_nat==-990

sort ctr_n yr mn id
save "national.level_uc.dta", replace
```

Party Nationalization Measures - Constituency Level Dataset

```
/*Note: this part of the file uses CLEA data to create several alternate measures of party
nationalization identified in the literature */
*Create frame dataset
use "clea_uc_19_calc.dta", clear
keep ctr_n id ctr yr mn cst_n cst
duplicates drop
save "cst.level_uc.dta", replace
*Create ENP constituency
use "clea_uc_19_calc.dta", clear
collapse (first) pv1 vv1 if pv1 >=0 & vv1 > 0, by (id ctr yr mn pty cst cst_n)
egen new_vv1 = sum(pv1), by (id ctr yr mn cst)
gen share2 = (pv1/new_vv1) * (pv1/new_vv1)
```

```
egen denom = sum(share2), by (id ctr yr mn cst)
gen ENP_cst = 1/denom
gen cvvi = 1 if vv1 != new_vv1
drop share2 denom pv1 vv1 new_vv1 pty
duplicates drop
joinby id ctr yr mn using "national.level_uc.dta", unmatched (using)
drop _merge
save "cst.level_uc.dta", replace
*Add national level data
joinby id ctr yr mn using "national.level_uc.dta", unmatched (master)
drop _merge
sleep 1000 // give enough time to finishing saving
save "cst.level_uc.dta", replace
compress
gen inflation5 = (ENP_nat - ENP_cst)/ENP_cst if ENP_nat != -990 & ENP_cst != -990
replace inflation5 = -990 if inflation5 ==.
duplicates drop
order ctr_n id ctr yr mn cst_n cst nvvi cvvi ENP_cst ENP_nat ENP_avg ENP_wght inflation1
inflation2 inflation3 inflation4 inflation5 PSNS PSNS_s PSNS_w PSNS_sw local_E
sort ctr_n id ctr yr mn cst_n cst nvvi cvvi ENP_cst ENP_nat ENP_avg ENP_wght inflation1
inflation2 inflation3 inflation4 inflation5 PSNS PSNS_s PSNS_w PSNS_sw local_E
label var ctr_n "Country Name"
label var ENP_cst "Effective Number of Electoral Parties at the constituency level"
label var inflation5 "Moenius & Kasuya local inflation measure"
label var cvvi "Constituency Valid Vote Indicator"
replace ENP_cst = -990 if ENP_cst ==.
replace cvvi = 0 if cvvi ==.
replace ENP_nat = -990 if ENP_nat ==.
replace ENP_avg = -990 if ENP_avg ==.
replace ENP_wght = -990 if ENP_wght ==.
replace inflation1 = -990 if inflation1 ==.
replace inflation2 = -990 if inflation2 ==.
replace inflation3 = -990 if inflation3 ==.
replace inflation4 = -990 if inflation4 ==.
replace inflation5 = -990 if inflation5 ==.
replace PSNS = -990 if PSNS ==.
replace PSNS_s = -990 if PSNS_s ==.
replace PSNS_w = -990 if PSNS_w ==.
replace PSNS_sw = -990 if PSNS_sw ==.
replace local_E = -990 if local_E ==.
replace ENP_cst = -990 if yr < 1834 & ctr == 840
*remove all duplicates
duplicates drop ctr yr cst mn, force
sleep 2000 // give enough time to finishing saving
save "cst.level_uc.dta", repla
```