Package 'esaps'

October 13, 2022

Type Package

Title Indicators of Electoral Systems and Party Systems

Version 0.2.2

Description It allows structuring electoral data of different size and structure
to calculate various indicators frequently used in the studies of electoral systems and party systems.
Indicators of electoral volatility, electoral disproportionality, party nationalization and the effective number of parties are included.

License GPL-2

URL https://github.com/Nicolas-Schmidt/esaps

Encoding UTF-8

Imports dplyr, tidyr, plyr (>= 1.8.4), readODS (>= 1.6.4), readxl (>= 1.0.0)

RoxygenNote 7.1.1

NeedsCompilation no

Author Nicolas Schmidt [aut, cre] (<https://orcid.org/0000-0001-5083-5792>)

Maintainer Nicolas Schmidt <nschmidt@cienciassociales.edu.uy>

Repository CRAN

Date/Publication 2021-09-13 18:30:05 UTC

R topics documented:

esaps-package *Indicators of Electoral Systems and Party Systems*

Description

It allows structuring electoral data of different size and structure to calculate various indicators frequently used in the studies of electoral systems and party systems:

- Electoral Volatility:
  - Pedersen (1979) <doi:10.1111/j.1475-6765.1979.tb01267.x>
  - Powell and Tucker (2014) <doi:10.1017/S0007123412000531>
  - Lago and Torcal (2019) <doi:10.1177/1354068818795191>
- Effective Number of Parties:
  - Laakso and Taagepera (1979) <doi:10.1177/001041407901200101>
- Electoral disproportionality:
  - Rae (1971, ISBN:9780300015171)
  - Loosemore and Hanby (1971) <doi:10.1017/S000712340000925X>
  - Lijphart (1986, ISBN:0875860648, 0875860745)
  - Lijphart (1994) <doi:10.1093/acprof:oso/9780198273479.001.0001>
  - Cox and Shugart (1991) <doi:10.1016/0261-3794(91)90025-N>
  - Gallagher (1991) <doi:10.1016/0261-3794(91)90004-C>
- Party System Nationalization:
  - Mainwaring and Jones (2003) <doi:10.1177/1354068830092002>
  - Chhibber and Kollman (2004, ISBN:9781400826377)

Author(s)

Nicolas Schmidt <nschmidt@cienciassociales.edu.uy>

See Also

Useful link: https://nicolas-schmidt.github.io/esaps/index.html

convert_esaps *Converts data in table form to tidy_data*

Description

Convert data in table format to tidy_data to use in the indicators of the tidy_data.

Usage

```
convert_esaps(
  path = NULL,
  dataset = NULL,
  file.name = NULL,
  nSheets = 1,
  election.name,
  unit.name,
  M.name = NULL,
  votes_nac.name = NULL,
  seats = FALSE,
  allSheet = FALSE
)
```

Arguments

path	Character vector containing one or more path names.
dataset	Electoral results by party. It can be a data.frame or a list.
file.name	Name of the data file with extension.
nSheets	Number of countries (number of sheets).' 'Country' is a generic unit of analysis (province, department, etc.)
election.name	Name of the variable that contains elections.
unit.name	Name of the variable that contains the unit. 'unit' is a generic unit of analysis (province, department, etc.)
M.name	Name of the variable that contains the district magnitude (M+1). It is for the calculation of endogenous and exogenous electoral volatility (Torcal and Lago, 2015).
votes_nac.name	la la la
seats	By default it is FALSE. If it is TRUE, it indicates that, in addition to electoral data per party, there is allocation data per seat. In this case, one column must be loaded for each party with the electoral result and another with the number of seats it obtained. The structure must be: party_1, party_2, ..., party_n, seats_party_1, seats_party_2, ..., seats_party_n.
allSheet	By default it is FALSE. Load all the sheets that are in the files selected in file.name. This argument takes precedence over nSheets.

Value

data.frame

Author(s)

Nicolas Schmidt <nschmidt@cienciassociales.edu.uy>

Examples

```
votes <- list(data.frame(country = rep("ARG", 3),
                         year = c(1995, 2000, 2005),
                         party_A = c(40,10,20),
                         party_B = c(35,20,40),
                         party_C = c(25,70,40)),
              data.frame(country = rep("URY", 4),
                         year = c(1995, 2000, 2005, 2010),
                         party_A = c(30,30,20,20),
                         party_B = c(30,50,40, 30),
                         party_C = c(30,10,30, 25),
                         party_D = c(10,10,10,25)),
              data.frame(country = rep("BRA", 2),
                         year = c(1993, 1998),
                         party_A = c(30, 55),
                         party_B = c(70, 45)))

votes <- convert_esaps(dataset = votes, unit.name = "country", election.name = "year")


votes2 <- data.frame(year = c(2000, 2005),
                     country = "URY",
                     votes_party1 = c(20, 30),
                     votes_party2 = c(30, 35),
                     votes_party3 = c(40, 25),
                     votes_party4 = c(10, 10),
                     seats_party1 = c(25, 35),
                     seats_party2 = c(20, 30),
                     seats_party3 = c(40, 30),
                     seats_party4 = c(15, 5))

votes <- convert_esaps(dataset = votes2, unit.name = "country", election.name = "year",
                       seats = TRUE)


## Not run:
v1 <- convert_esaps(path = getwd(),
                    file.name = c("electionBRA.xlsx", "electionARG.xlsx"),
                    election.name = "elec",
                    unit.name = "district",
                    allSheet = TRUE)

v2 <-convert_esaps(path = getwd(),
                    file.name = c("ARG.ods", "URY.ods", "BRA.ods"),
```

```
                    nCountry = c(2, 3, 1),
                    election.name = "elec",
                    unit.name = "province")

v3 <- convert_esaps(path = here::here(),
                    file.name = list.files(pattern = "*.xlsx"),
                    election.name = "year",
                    unit.name = "country",
                    M.name = "magnitude",
                    seats = TRUE,
                    allSheet = TRUE)

## End(Not run)
```

`dispro` *Electoral Disproportionality*

Description

Electoral Disproportionality: Rae (1971), Loosemore and Hanby (1971), Lijphart (1986), Lijphart (1994), Gallagher (1991) and Cox and Shugart (1991).

Usage

```
dispro(tidy_data, method, scale = 1)
```

Arguments

`tidy_data`	data.frame that contains the following variables with these names: • `election`: year of election. • `unit`: the unit of analysis (province, department ...) • `party`: name of the political parties that obtain votes. • `votes`: votes obtained by each party. • `seats`: . If the data is not structured in this way you can order it with: `convert_esaps`.
`method`	Method to calculate electoral volatility: • `method = "Rae"` or `method = 1`. • `method = "Loosemore and Hanby"` or `method = 2`. • `method = "Lijphart_1"` or `method = 3`. • `method = "Lijphart_2"` or `method = 4`. • `method = "Gallagher"` or `method = 5`. • `method = "Cox and Shugart"` or `method = 6`.
`scale`	By default it is `100`, the indices will vary between 0 and 100. If `scale = 1` the variation will be between 0 and 1.

Value

data.frame.

Author(s)

Nicolas Schmidt <nschmidt@cienciassociales.edu.uy>

Examples

```
votes <- data.frame(election = rep(c(2000, 2005), each = 4),
                    unit  = rep(c("ARG", "URY"), each = 4),
                    party = c("party_A", "party_B","party_C","party_D"),
                    votes = c(20, 30, 40, 10, 30, 35, 25, 10),
                    seats = c(25, 20, 40, 15, 35, 30, 30, 5)
                    )

dispro(votes, 1:6, 1)
dispro(votes, 3)
dispro(tidy_data = votes, method = 3:5)
dispro(tidy_data = votes, method = c(1,3,6))
dispro(tidy_data = votes, method = c("Rae", "Gallagher"))
```

---

enp *Effective Number of Parties*

Description

The Effective Number of Parties (ENP) is an index developed by Laakso and Taagepera (1979) that allows to count the relevant parties in a party system. The formula consists on dividing one over the sum of the squares of the proportions (votes or seats) that the parties obtain in an electoral instance.

Usage

```
enp(tidy_data, enp_seats = FALSE, summary = FALSE)
```

Arguments

`tidy_data`	data.frame that contains the following variables with these names: • `election`: year of election. • `unit`: the unit of analysis (province, department ...) • `party`: name of the political parties that obtain votes. • `votes`: votes obtained by each party. • `seats`: . If the data is not structured in this way you can order it with: `convert_esaps`.
`enp_seats`	enp_seats = TRUE allows us to calculate jointly the effective number of electoral parties and in the congress.
`summary`	Summary of the data by unit, by default it is `FALSE`.

Value

if `summary = FALSE`, return data.frame.

if `summary = TRUE`, return a list with two data.frame.

- `list[[1]]` Indicator
- `list[[2]]` Summary
  - `min` variable 'election'
  - `max` variable 'election'
  - number of elections
  - `mean` indicator

Author(s)

Nicolas Schmidt <nschmidt@cienciassociales.edu.uy>

Examples

```
votes <- data.frame(election = rep(c(2000, 2005), each = 4),
                    unit  = rep(c("ARG", "URY"), each = 4),
                    party = c("party_A", "party_B","party_C","party_D"),
                    votes = c(20, 20, 50, 10, 30, 35, 25, 10),
                    seats = c(25, 25, 40, 10, 30, 30, 30, 10)
                    )

enp(votes)
enp(votes, enp_seats = TRUE)
enp(votes, summary = TRUE)
```

`evolat` *Electoral volatility calculation*

Description

Electoral volatility calculation: Pedersen (1979), Powell and Tucker (2014) and Torcal and Lago (2015).

Usage

```
evolat(
  tidy_data,
  method,
  threshold = 2,
  summary = FALSE,
  digits = 2,
  scale = 100
)
```

Arguments

tidy_data	data.frame that contains the following variables with these names: • election: year of election. • unit: the unit of analysis (province, department ...) • party: name of the political parties that obtain votes. • votes: votes obtained by each party. • M: magnitude of the district. Only if the method = 3. It refers to the M + 1 rule (only 'M' must be loaded). If the data is not structured in this way you can order it with: convert_esaps.
method	Method to calculate electoral volatility: • method = "Pedersen" or method = 1. • method = "Powell and Tucker" or method = 2. • method = "Torcal and Lago" or method = 3.
threshold	Minimum threshold for 'Type A' electoral volatility calculation (Powell and Tucker, 2014). By default is 2%.
summary	Summary of data by unit, by default it is FALSE.
digits	integer indicating the number of decimal places to be used.
scale	By default it is 100, the indices will vary between 0 and 100. If scale = 1 the variation will be between 0 and 1.

Value

if summary = FALSE, return data.frame.

if summary = TRUE, return a list with two data.frame.

- list[[1]] Indicator
- list[[2]] Summary by 'unit'
  - min variable 'election'
  - max variable 'election'
  - number of elections
  - mean indicator
  - standard deviation indicator

Author(s)

Nicolas Schmidt <nschmidt@cienciassociales.edu.uy>

Examples

```
votes <- data.frame(election = rep(c(1995, 2000, 2005, 2010),4),
                    unit = "ARG",
                    party = rep(c("party_A","party_B","party_C","party_D"), each = 4),
                    votes = c(30,30,20,20,30,50,40,30,30,10,30,25,10,10,10,25))
evolat(votes, 1)
evolat(tidy_data = votes, method = 1, summary = TRUE)
```

psns *Party System Nationalization Score*

Description

Party System Nationalization Score Mainwaring and Jones (2003) and Chhibber and Kollman (2004)

Usage

```
psns(tidy_data, method, pns = FALSE, scale = 100)
```

Arguments

`tidy_data`	data.frame that contains the following variables with these names: • `election`: year of election. • `unit`: the unit of analysis (province, department ...) • `party`: name of the political parties that obtain votes. • `votes`: votes obtained by each party. • `votes_nac`: votes at national level for each party. If the data is not structured in this way you can order it with: `convert_esaps`.
`method`	Method to calculate Party System Nationalization Score: • `method = "Mainwaring and Jones"` or `method = 1`. • `method = "Chhibber and Kollman"` or `method = 2`.
`pns`	by default it is `FALSE`. If `TRUE`, the Party Nationalization Score is calculated. In method, you must indicate: `method = 1`.
`scale`	By default it is `100`, the indices will vary between 0 and 100. If `scale = 1` the variation will be between 0 and 1.

Value

if `pns = FALSE`, return data.frame.

if `pns = TRUE`, return a list with two data.frame.

- `list[[1]]` PSNS: Party System Nationalization Score
- `list[[2]]` PNS: Party Nationalization Score

Author(s)

Nicolas Schmidt <nschmidt@cienciassociales.edu.uy>

Examples

```
votes <- data.frame(election = rep(2000,4),
                    unit  = rep(c("District_1", "District_2"), each = 2),
                    party = rep(c("party_A", "party_B"), 2),
                    votes = c(0.60,0.40, 0.30, 0.70),
                    votes_nac = rep(c(0.55,0.45),2)
                   )
psns(tidy_data = votes, method = 1)
psns(tidy_data = votes, method = 1, pns = TRUE)
```

Index