

Package ‘DisImpact’

January 20, 2025

Title Calculates Disproportionate Impact When Binary Success Data are Disaggregated by Subgroups

Version 0.0.21

Description Implements methods for calculating disproportionate impact: the percentage point gap, proportionality index, and the 80% index. California Community Colleges Chancellor's Office (2017). Percentage Point Gap Method. <<https://www.cccco.edu/-/media/CCCCO-Website/About-Us/Divisions/Digital-Innovation-and-Infrastructure/Research/Files/PercentagePointGapMethod2017.ashx>>. California Community Colleges Chancellor's Office (2014). Guidelines for Measuring Disproportionate Impact in Equity Plans. <<https://www.cccco.edu/-/media/CCCCO-Website/Files/DII/guidelines-for-measuring-disproportionate-impact-in-equity-plans-tfa-ada.pdf>>.

Depends R (>= 3.4.0)

Imports dplyr (>= 0.8.5), rlang, tidyselect, purrr, tidyr, parallel, fst, DBI, duckdb (>= 0.5.0), glue, stringr, data.table (>= 1.14.2), collapse, sets

License GPL-3

URL <https://github.com/vinhdizzo/DisImpact>

BugReports <https://github.com/vinhdizzo/DisImpact/issues>

Encoding UTF-8

LazyData true

RoxygenNote 7.1.2

Suggests knitr, rmarkdown, markdown, prettydoc, ggplot2, forcats, scales, tinytest

VignetteBuilder knitr

NeedsCompilation no

Author Vinh Nguyen [aut, cre]

Maintainer Vinh Nguyen <nguyenvq714@gmail.com>

Repository CRAN

Date/Publication 2022-10-10 18:00:02 UTC

Contents

di_80_index	2
di_calc_dt	4
di_calc_sql	6
di_iterate	8
di_iterate_dt	11
di_iterate_on_long	14
di_iterate_sql	16
di_ppg	19
di_ppg_iterate	22
di_prop_index	24
ppg_moe	25
ssm_cohort	26
student_equity	27
surround_quote_if_needed	28
Index	30

di_80_index	<i>Calculate disproportionate impact per the 80% index</i>
-------------	------------------------------------------------------------

Description

Calculate disproportionate impact per the 80% index method.

Usage

```
di_80_index(
  success,
  group,
  cohort,
  weight,
  data,
  di_80_index_cutoff = 0.8,
  reference_group = "hpg",
  check_valid_reference = TRUE
)
```

Arguments

success	A vector of success indicators (1/0 or TRUE/FALSE) or an unquoted reference (name) to a column in data if it is specified. It could also be a vector of counts, in which case weight should also be specified (group size).
group	A vector of group names of the same length as success or an unquoted reference (name) to a column in data if it is specified.

cohort	(Optional) A vector of cohort names of the same length as success or an unquoted reference (name) to a column in data if it is specified. disproportionate impact is calculated for every group within each cohort. When cohort is not specified, then the analysis assumes a single cohort.
weight	(Optional) A vector of case weights of the same length as success or an unquoted reference (name) to a column in data if it is specified. If success consists of counts instead of success indicators (1/0), then weight should also be specified to indicate the group size.
data	(Optional) A data frame containing the variables of interest. If data is specified, then success, group, and cohort will be searched within it.
di_80_index_cutoff	A numeric value between 0 and 1 that is used to determine disproportionate impact if the index comparing the success rate of the current group to the reference group falls below this threshold; defaults to 0.80.
reference_group	The reference group value in group that each group should be compared to in order to determine disproportionate impact. By default (= 'hpg'), the group with the highest success rate is used as reference. The user could also specify a value of 'overall' to use the overall rate as the reference for comparison, or 'all but current' to use the combined success rate of all other groups excluding the current group for each comparison.
check_valid_reference	Check whether reference_group is a valid value; defaults to TRUE. This argument exists to be used in di_iterate as when iterating DI calculations, there may be some scenarios where a specified reference group does not contain any students.

Details

This function determines disproportionate impact based on the 80% index method, as described in [this](#) reference from the California Community Colleges Chancellor's Office. It assumes that a higher rate is good ("success"). For rates that are deemed negative (eg, rate of drop-outs, high is bad), then consider looking at the converse of the non-success (eg, non drop-outs, high is good) instead in order to leverage this function properly.

Value

A data frame consisting of:

- cohort (if used),
- group,
- n (sample size),
- success (number of successes for the cohort-group),
- pct (proportion of successes for the cohort-group),
- reference_group (the reference group used to compare and determine disproportionate impact),

- reference (the reference rate used for comparison, corresponding to reference_group),
- di_80_index (ratio of pct to the reference),
- di_indicator (1 if di_80_index < di_80_index_cutoff),
- success_needed_not_di (the number of additional successes needed in order to no longer be considered disproportionately impacted as compared to the reference), and
- success_needed_full_parity (the number of additional successes needed in order to achieve full parity with the reference).

References

California Community Colleges Chancellor's Office (2014). [Guidelines for Measuring Disproportionate Impact in Equity Plans](#).

Examples

```
library(dplyr)
data(student_equity)
di_80_index(success=Transfer, group=Ethnicity, data=student_equity) %>%
  as.data.frame
```

di_calc_dt	<i>Calculates disproportionate impact using multiple methods for data stored in a data.table object.</i>
------------	--------------------------------------------------------------------------------------------------------------------------

Description

Calculate disproportionate impact via the percentage point gap (PPG), proportionality index, and 80% index methods for data stored in a [data.table](#) object. This is the workhorse function leveraged by the [di_iterate_dt](#) function.

Usage

```
di_calc_dt(
  dt,
  success_var,
  group_var,
  cohort_var = "",
  weight_var = NULL,
  ppg_reference_group = "overall",
  min_moe = 0.03,
  use_prop_in_moe = FALSE,
  prop_sub_0 = 0.5,
  prop_sub_1 = 0.5,
  di_prop_index_cutoff = 0.8,
  di_80_index_cutoff = 0.8,
  di_80_index_reference_group = "hpg",
  filter_subset = ""
)
```

Arguments

dt	A data frame of class data.table . If the object is not a data table, one could surround the object with as.data.table .
success_var	A character value specifying the success variable name.
group_var	A character value specifying the group (disaggregation) variable name.
cohort_var	(Optional) A character value specifying the cohort variable. If not specified, then a single cohort is assumed (defaults to an empty string, '').
weight_var	(Optional) A character variable specifying the weight variable if the input data set is summarized (i.e., the the success variables specified in <code>success_vars</code> contain count of successes). Weight here corresponds to the denominator when calculating the success rate. Defaults to NULL for an input data set where each row describes an individual.
ppg_reference_group	Either 'overall', 'hpg', 'all but current', or a character value specifying a group from <code>group_var</code> to be used as the reference group for comparison using percentage point gap method.
min_moe	The minimum margin of error to be used in the PPG calculation; see di_ppg .
use_prop_in_moe	(TRUE or FALSE) Whether the estimated proportions should be used in the margin of error calculation by the PPG; see di_ppg .
prop_sub_0	Default is 0.50; see di_ppg .
prop_sub_1	Default is 0.50; see di_ppg .
di_prop_index_cutoff	Threshold used for determining disproportionate impact using the proportionality index; see di_prop_index ; defaults to 0.80.
di_80_index_cutoff	Threshold used for determining disproportionate impact using the 80% index; see di_80_index ; defaults to 0.80.
di_80_index_reference_group	Either 'overall', 'hpg', 'all but current', or a character value specifying a group from <code>group_var</code> to be used as the reference group for comparison using 80% index.
filter_subset	A character value such as "Ethnicity == 'White' & Gender == 'M'" used in the <code>i</code> argument (filtering rows via <code>dt[i, j, by]</code>) to filter data in <code>dt</code> . The character value is parsed using <code>eval(parse(text=filter_subset))</code> . Defaults to '' for no filtering.

Value

A [data.table](#) object with summarized results.

di_calc_sql	<i>Generate SQL code that calculates disproportionate impact using multiple methods for a specified table.</i>
-------------	----------------------------------------------------------------------------------------------------------------

Description

Generate SQL code that calculates disproportionate impact via the percentage point gap (PPG), proportionality index, and 80% index methods for a specified table name, success variable, group variable, and cohort variable. This is the workhorse function leveraged by the [di_iterate_sql](#) function.

Usage

```
di_calc_sql(
  db_table_name,
  success_var,
  group_var,
  cohort_var = "",
  weight_var = 1,
  ppg_reference_group = "overall",
  min_moe = 0.03,
  use_prop_in_moe = FALSE,
  prop_sub_0 = 0.5,
  prop_sub_1 = 0.5,
  di_prop_index_cutoff = 0.8,
  di_80_index_cutoff = 0.8,
  di_80_index_reference_group = "hpg",
  before_with_statement = "",
  after_with_statement = "",
  end_of_select_statement = "",
  where_statement = "",
  select_statement_add = ""
)
```

Arguments

db_table_name	A character value specifying a database table name.
success_var	A character value specifying the success variable name.
group_var	A character value specifying the group (disaggregation) variable name.
cohort_var	(Optional) A character value specifying the cohort variable. If not specified, then a single cohort is assumed (defaults to an empty string, '').
weight_var	(Optional) A character variable specifying the weight variable if the input data set is summarized (i.e., the the success variables specified in success_vars contain count of successes). Weight here corresponds to the denominator when calculating the success rate. Defaults to a numeric 1 which treats each row as an individual.

ppg_reference_group	Either 'overall', 'hpg', 'all but current', or a character value specifying a group from group_var to be used as the reference group for comparison using the percentage point gap method.
min_moe	The minimum margin of error to be used in the PPG calculation; see di_ppg .
use_prop_in_moe	(TRUE or FALSE) Whether the estimated proportions should be used in the margin of error calculation by the PPG; see di_ppg .
prop_sub_0	Default is 0.50; see di_ppg .
prop_sub_1	Default is 0.50; see di_ppg .
di_prop_index_cutoff	Threshold used for determining disproportionate impact using the proportionality index; see di_prop_index ; defaults to 0.80.
di_80_index_cutoff	Threshold used for determining disproportionate impact using the 80% index; see di_80_index ; defaults to 0.80.
di_80_index_reference_group	Either 'overall', 'hpg', 'all but current', or a character value specifying a group from group_var to be used as the reference group for comparison using 80% index.
before_with_statement	Character value to be added to the SQL query to allow for modification. Defaults to '' (empty string).
after_with_statement	Character value to be added to the SQL query to allow for modification. Defaults to '' (empty string).
end_of_select_statement	Character value to be added to the SQL query to allow for modification. Defaults to '' (empty string).
where_statement	Character value to be added to the SQL query to allow for modification. Defaults to '' (empty string).
select_statement_add	Character value to be added to the SQL query to allow for modification. Defaults to '' (empty string).

Value

A character value (SQL query) that could be executed on a database.

di_iterate	<i>Iteratively calculate disproportionate impact using multiple method for many variables.</i>
------------	------------------------------------------------------------------------------------------------

Description

Iteratively calculate disproportionate impact via the percentage point gap (PPG), proportionality index, and 80% index methods for many success variables, disaggregation variables, and scenarios.

Usage

```
di_iterate(
  data,
  success_vars,
  group_vars,
  cohort_vars = NULL,
  scenario_repeat_by_vars = NULL,
  exclude_scenario_df = NULL,
  weight_var = NULL,
  include_non_disagg_results = TRUE,
  ppg_reference_groups = "overall",
  min_moe = 0.03,
  use_prop_in_moe = FALSE,
  prop_sub_0 = 0.5,
  prop_sub_1 = 0.5,
  di_prop_index_cutoff = 0.8,
  di_80_index_cutoff = 0.8,
  di_80_index_reference_groups = "hpg",
  check_valid_reference = TRUE,
  parallel = FALSE,
  parallel_n_cores = parallel::detectCores(),
  parallel_split_to_disk = FALSE
)
```

Arguments

data	A data frame for which to iterate DI calculations for a set of variables.
success_vars	A character vector of success variable names to iterate across.
group_vars	A character vector of group (disaggregation) variable names to iterate across.
cohort_vars	(Optional) A character vector of the same length as success_vars to indicate the cohort variable to be used for each variable specified in success_vars. A vector of length 1 could be specified, in which case the same cohort variable is used for each success variable. If not specified, then a single cohort is assumed for all success variables.

scenario_repeat_by_vars	<p>(Optional) A character vector of variables to repeat DI calculations for across all combination of these variables. For example, the following variables could be specified:</p> <ul style="list-style-type: none"> • Ed Goal: Degree/Transfer, Short-term Career, Non-credit • First time college student: Yes, No • Full-time status: Yes, No <p>Each combination of these variables (eg, full time, first time college students with an ed goal of degree/transfer as one combination) would constitute an iteration / sample for which to calculate disproportionate impact for outcomes listed in <code>success_vars</code> and for the disaggregation variables listed in <code>group_vars</code>. The overall rate of success for full time, first time college students with an ed goal of degree/transfer would just include these students and not others. Each variable specified is also collapsed to an '- All' group so that the combinations also reflect all students of a particular category. The total number of combinations for the previous example would be (+1 representing the all category): $(3 + 1) \times (2 + 1) \times (2 + 1) = 36$.</p>
exclude_scenario_df	<p>(Optional) A data frame with variables that match <code>scenario_repeat_by_vars</code> for specifying the combinations to exclude from DI calculations. Following the example specified above, one could choose to exclude part-time non-credit students from consideration.</p>
weight_var	<p>(Optional) A character variable specifying the weight variable if the input data set is summarized (i.e., the the success variables specified in <code>success_vars</code> contain count of successes). Weight here corresponds to the denominator when calculating the success rate. Defaults to NULL for an input data set where each row describes each individual.</p>
include_non_disagg_results	<p>A logical variable specifying whether or not the non-disaggregated results should be returned; defaults to TRUE. When TRUE, a new variable '- None' is added to the data set with a single data value '- All', and this variable is added <code>group_vars</code> as a disaggregation/group variable. The user would want these results returned to review non-disaggregated results.</p>
ppg_reference_groups	<p>Either 'overall', 'hpg', 'all but current', or a character vector of the same length as <code>group_vars</code> that indicates the reference group value for each group variable in <code>group_vars</code> when determining disproportionate impact using the percentage point gap method.</p>
min_moe	<p>The minimum margin of error to be used in the PPG calculation, passed to di_ppg.</p>
use_prop_in_moe	<p>Whether the estimated proportions should be used in the margin of error calculation by the PPG, passed to di_ppg.</p>
prop_sub_0	<p>passed to di_ppg; defaults to 0.50.</p>
prop_sub_1	<p>passed to di_ppg; defaults to 0.50.</p>

<code>di_prop_index_cutoff</code>	Threshold used for determining disproportionate impact using the proportionality index; passed to <code>di_prop_index</code> ; defaults to 0.80.
<code>di_80_index_cutoff</code>	Threshold used for determining disproportionate impact using the 80% index; passed to <code>di_80_index</code> ; defaults to 0.80.
<code>di_80_index_reference_groups</code>	Either 'overall', 'hpg', 'all but current', or a character vector of the same length as <code>group_vars</code> that indicates the reference group value for each group variable in <code>group_vars</code> when determining disproportionate impact using the 80% index.
<code>check_valid_reference</code>	Check whether <code>ppg_reference_groups</code> and <code>di_80_index_reference_groups</code> contain valid values; defaults to TRUE.
<code>parallel</code>	If TRUE, then perform calculations in parallel based on the scenarios specified by <code>scenario_repeat_by_vars</code> . Defaults to FALSE. Parallel execution is based on the <code>parallel</code> package included in base R, using <code>parLapply</code> on Windows and <code>mclapply</code> on POSIX-based systems (Linux/Mac).
<code>parallel_n_cores</code>	The number of CPU cores to use if <code>parallel=TRUE</code> . Defaults to the maximum number CPU cores on the system.
<code>parallel_split_to_disk</code>	If TRUE and <code>parallel=TRUE</code> , then create intermediate data sets for each scenario generated by <code>scenario_repeat_by_vars</code> , write them to disk, and import the required data set when necessary for each scenario executing in parallel. This feature is useful when the data set specified by <code>data</code> is very large and parallel execution is desired for speed in order to reduce the likelihood of consuming all the system's memory and crashing. Note that there is an overhead I/O cost on speed when this feature is used. Defaults to FALSE.

Details

Iteratively calculate disproportionate impact via the percentage point gap (PPG), proportionality index, and 80% index methods for all combinations of `success_vars`, `group_vars`, and `cohort_vars`, for each combination of subgroups specified by `scenario_repeat_by_vars`.

Value

A summarized data set (data frame) consisting of:

- `success_variable` (elements of `success_vars`),
- `disaggregation` (elements of `group_vars`),
- `cohort` (values corresponding to the variables specified in `cohort_vars`,
- `di_indicator_ppg` (1 if there is disproportionate impact per the percentage point gap method, 0 otherwise),
- `di_indicator_prop_index` (1 if there is disproportionate impact per the proportionality index, 0 otherwise),

- di_indicator_80_index (1 if there is disproportionate impact per the 80% index, 0 otherwise), and
- other relevant fields returned from [di_ppg](#), [di_prop_index](#), and [di_80_index](#).

Examples

```
library(dplyr)
data(student_equity)
# Multiple group variables
di_iterate(data=student_equity, success_vars=c('Transfer')
, group_vars=c('Ethnicity', 'Gender'), cohort_vars=c('Cohort')
, ppg_reference_groups='overall')
```

di_iterate_dt	<i>Iteratively calculate disproportionate impact using multiple method for many variables, using data.table and collapse.</i>
---------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------

Description

Iteratively calculate disproportionate impact via the percentage point gap (PPG), proportionality index, and 80% index methods for many success variables, disaggregation variables, and scenarios, using [data.table](#) and [collapse](#).

Usage

```
di_iterate_dt(
  dt,
  success_vars,
  group_vars,
  cohort_vars = NULL,
  scenario_repeat_by_vars = NULL,
  exclude_scenario_df = NULL,
  weight_var = NULL,
  include_non_disagg_results = TRUE,
  ppg_reference_groups = "overall",
  min_moe = 0.03,
  use_prop_in_moe = FALSE,
  prop_sub_0 = 0.5,
  prop_sub_1 = 0.5,
  di_prop_index_cutoff = 0.8,
  di_80_index_cutoff = 0.8,
  di_80_index_reference_groups = "hpg",
  check_valid_reference = TRUE,
  parallel = FALSE,
  parallel_n_cores = parallel::detectCores()/2
)
```

Arguments

dt	A data frame of class <code>data.table</code> . If the object is not a data table, one could surround the object with <code>as.data.table</code> .
success_vars	A character vector of success variable names to iterate across.
group_vars	A character vector of group (disaggregation) variable names to iterate across.
cohort_vars	(Optional) A character vector of the same length as <code>success_vars</code> to indicate the cohort variable to be used for each variable specified in <code>success_vars</code> . A vector of length 1 could be specified, in which case the same cohort variable is used for each success variable. If not specified, then a single cohort is assumed for all success variables (defaults to NULL).
scenario_repeat_by_vars	(Optional) A character vector of variables to repeat DI calculations for across all combination of these variables. For example, the following variables could be specified: <ul style="list-style-type: none"> • Ed Goal: Degree/Transfer, Shot-term Career, Non-credit • First time college student: Yes, No • Full-time status: Yes, No Each combination of these variables (eg, full time, first time college students with an ed goal of degree/transfer as one combination) would constitute an iteration / sample for which to calculate disproportionate impact for outcomes listed in <code>success_vars</code> and for the disaggregation variables listed in <code>group_vars</code> . The overall rate of success for full time, first time college students with an ed goal of degree/transfer would just include these students and not others. Each variable specified is also collapsed to an '- All' group so that the combinations also reflect all students of a particular category. The total number of combinations for the previous example would be (+1 representing the all category): $(3 + 1) \times (2 + 1) \times (2 + 1) = 36$.
exclude_scenario_df	(Optional) A data frame with variables that match <code>scenario_repeat_by_vars</code> for specifying the combinations to exclude from DI calculations. Following the example specified above, one could choose to exclude part-time non-credit students from consideration.
weight_var	(Optional) A character variable specifying the weight variable if the input data set is summarized (i.e., the the success variables specified in <code>success_vars</code> contain count of successes). Weight here corresponds to the denominator when calculating the success rate. Defaults to NULL for an input data set where each row describes an individual.
include_non_disagg_results	A logical variable specifying whether or not the non-disaggregated results should be returned; defaults to TRUE. When TRUE, a new variable <code>~ None</code> is added to the data set with a single data value '- All', and this variable is added to <code>group_vars</code> as a disaggregation/group variable. The user would want these results returned to review non-disaggregated results.
ppg_reference_groups	Either 'overall', 'hpg', 'all but current', or a character vector of the same length as <code>group_vars</code> that indicates the reference group value for each group

	variable in <code>group_vars</code> when determining disproportionate impact using the percentage point gap method.
<code>min_moe</code>	The minimum margin of error to be used in the PPG calculation; see di_ppg .
<code>use_prop_in_moe</code>	(TRUE or FALSE) Whether the estimated proportions should be used in the margin of error calculation by the PPG; see di_ppg .
<code>prop_sub_0</code>	Default is 0.50; see di_ppg .
<code>prop_sub_1</code>	Default is 0.50; see di_ppg .
<code>di_prop_index_cutoff</code>	Threshold used for determining disproportionate impact using the proportionality index; see di_prop_index ; defaults to 0.80.
<code>di_80_index_cutoff</code>	Threshold used for determining disproportionate impact using the 80% index; see di_80_index ; defaults to 0.80.
<code>di_80_index_reference_groups</code>	Either 'overall', 'hpg', 'all but current', or a character vector of the same length as <code>group_vars</code> that indicates the reference group value for each group variable in <code>group_vars</code> when determining disproportionate impact using the 80% index.
<code>check_valid_reference</code>	(TRUE or FALSE) Check whether <code>ppg_reference_groups</code> and <code>di_80_index_reference_groups</code> contain valid values; defaults to TRUE.
<code>parallel</code>	If TRUE, then perform calculations in parallel. Defaults to FALSE. Parallel execution is based on the <code>parallel</code> package included in base R, using parLapply on Windows and mclapply on POSIX-based systems (Linux/Mac).
<code>parallel_n_cores</code>	The number of CPU cores to use if <code>parallel=TRUE</code> . Defaults to half of the maximum number of CPU cores on the system.

Details

Iteratively calculate disproportionate impact via the percentage point gap (PPG), proportionality index, and 80% index methods for all combinations of `success_vars`, `group_vars`, and `cohort_vars`, for each combination of subgroups specified by `scenario_repeat_by_vars`, using [data.table](#) and [collapse](#).

Value

A summarized data set of class [data.table](#), with variables as described in [di_iterate](#).

di_iterate_on_long	<i>Iteratively calculate disproportionate impact using multiple methods for a long and summarized data set</i>
--------------------	----------------------------------------------------------------------------------------------------------------

Description

Calculate disproportionate impact via the percentage point gap (PPG), proportionality index, and 80% index methods for a "long" and summarized data set with many success variables and disaggregation variables, where the success counts and disaggregation groups are stored in a single column or variable for each.

Usage

```
di_iterate_on_long(
  data,
  num_var,
  denom_var,
  disagg_var_col,
  group_var_col,
  disagg_var_col_2 = NULL,
  group_var_col_2 = NULL,
  cohort_var_col = NULL,
  summarize_by_vars = NULL,
  custom_reference_group_flag_var = NULL,
  ...
)
```

Arguments

data	A data frame for which to iterate DI calculations for a set of variables.
num_var	A variable name (character value) from data where the variable stores success counts (the numerator in success rates). Success rates are calculated by aggregating num_var and denom_var for each unique combination of values in disagg_var_col, group_var_col, disagg_var_col_2, group_var_col_2, cohort_var_col, and summarize_by_vars. If such combinations are unique (single row), then rows are not collapsed.
denom_var	A variable name (character value) from data where the variable stores the group size (the denominator in success rates).
disagg_var_col	A variable name (character value) from data where the variable stores the different disaggregation scenarios. The disaggregation variable could include such values as 'Ethnicity', 'Age Group', and 'Foster Youth', corresponding to three disaggregation scenarios.
group_var_col	A variable name (character value) from data where the variable stores the group name for each group within a level of disaggregation specified in disagg_var_col. For example, the group names could include 'Asian', 'White', 'Black', 'Latinx', 'Native American', and 'Other' for a disaggregation on ethnicity; 'Under 18',

'18-21', '22-25', and '25+' for an age group disaggregation; and 'Yes' and 'No' for a foster youth status disaggregation.

disagg_var_col_2

(Optional) A variable name (character value) from data where the variable stores an optional second disaggregation variable, which allows for the intersectionality of variables listed in `disagg_var_col` and `disagg_var_col_2`. The second disaggregation variable could describe something not in `disagg_var_col_2`, such as 'Gender', which would require all groups described in `group_var_col` to be broken out by gender.

group_var_col_2

(Optional) A variable name (character value) from data where the variable stores the group name for each group within a second level of disaggregation specified in `disagg_var_col_2`. For example, the group names could include 'Male', 'Female', 'Non-binary', and 'Unknown' if 'Gender' is a value in the variable `disagg_var_col_2`.

cohort_var_col

(Optional) A variable name (character value) from data where the variable stores the cohort label for the data described in each row.

summarize_by_vars

(Optional) A character vector of variable names in data for which `num_var` and `denom_var` are used for aggregation to calculate success rates for the disproportionate impact (DI) analysis set up by `disagg_var_col`, `group_var_col`, `disagg_var_col_2`, and `group_var_col_2`. For example, `summarize_by_vars=c('Outcome')` could specify a single variable/column that describes the outcome or metric in `num_var`, where the outcome values might include 'Completion of Transfer-Level Math', 'Completion of Transfer-Level English', 'Transfer', 'Associate Degree'.

custom_reference_group_flag_var

(Optional) A variable name (character value) from data where the variable flags the row or group that should be used as the reference group (1 if row is a reference group, 0 otherwise) for comparison in the percentage point gap method and the 80% index method. When this argument is used, then the `ppg_reference_groups` and `di_80_index_reference_groups` arguments should not be specified.

...

(Optional) Other arguments such as `ppg_reference_groups`, `min_moe`, `use_prop_in_moe`, `prop_sub_0`, `prop_sub_1`, `di_prop_index_cutoff`, `di_80_index_cutoff`, `di_80_index_reference`, and `check_valid_reference` from [di_iterate](#).

Details

Iteratively calculate disproportionate impact via the percentage point gap (PPG), proportionality index, and 80% index methods for all combinations of `success_vars`, `group_vars`, and `cohort_vars`, for each combination of subgroups specified by `scenario_repeat_by_vars`.

Value

A summarized data set (data frame) consisting of:

- variables specified by `summarize_by_vars`, `disagg_var_col`, `group_var_col`, `disagg_var_col_2`, and `group_var_col_2`,

- di_indicator_ppg (1 if there is disproportionate impact per the percentage point gap method, 0 otherwise),
- di_indicator_prop_index (1 if there is disproportionate impact per the proportionality index, 0 otherwise),
- di_indicator_80_index (1 if there is disproportionate impact per the 80% index, 0 otherwise), and
- other relevant fields returned from [di_ppg](#), [di_prop_index](#), and [di_80_index](#).

Examples

```
library(dplyr)
data(ssm_cohort)
di_iterate_on_long(data=ssm_cohort %>% filter(missingFlag==0) # remove missing data
  , num_var='value', denom_var='denom'
  , disagg_var_col='disagg1', group_var_col='subgroup1'
  , cohort_var_col='academicYear', summarize_by_vars=c('categoryLabel')
  , ppg_reference_groups='all but current' # PPG-1
  , di_80_index_reference_groups='all but current')
```

di_iterate_sql	<i>Iteratively calculate disproportionate impact using multiple methods for many variables, using SQL.</i>
----------------	------------------------------------------------------------------------------------------------------------

Description

Iteratively calculate disproportionate impact via the percentage point gap (PPG), proportionality index, and 80% index methods for many success variables, disaggregation variables, and scenarios, using SQL (for data stored in a database or in a parquet data file).

Usage

```
di_iterate_sql(
  db_conn,
  db_table_name,
  success_vars,
  group_vars,
  cohort_vars = NULL,
  scenario_repeat_by_vars = NULL,
  exclude_scenario_df = NULL,
  weight_var = NULL,
  include_non_disagg_results = TRUE,
  ppg_reference_groups = "overall",
  min_moe = 0.03,
  use_prop_in_moe = FALSE,
  prop_sub_0 = 0.5,
  prop_sub_1 = 0.5,
  di_prop_index_cutoff = 0.8,
```



```

di_80_index_cutoff = 0.8,
di_80_index_reference_groups = "hpg",
check_valid_reference = TRUE,
parallel = FALSE,
parallel_n_cores = parallel::detectCores()/2,
mssql_flag = FALSE,
return_what = "data",
staging_table = paste0("DisImpact_Staging_", paste0(sample(1:9, size = 5, replace =
  TRUE), collapse = "")),
drop_staging_table = TRUE
)

```

Arguments

- db_conn** A database connection object, returned by [dbConnect](#).
- db_table_name** A character value specifying a database table name.
- success_vars** A character vector of success variable names to iterate across.
- group_vars** A character vector of group (disaggregation) variable names to iterate across.
- cohort_vars** (Optional) A character vector of the same length as `success_vars` to indicate the cohort variable to be used for each variable specified in `success_vars`. A vector of length 1 could be specified, in which case the same cohort variable is used for each success variable. If not specified, then a single cohort is assumed for all success variables (defaults to NULL).
- scenario_repeat_by_vars** (Optional) A character vector of variables to repeat DI calculations for across all combination of these variables. For example, the following variables could be specified:
- Ed Goal: Degree/Transfer, Shot-term Career, Non-credit
 - First time college student: Yes, No
 - Full-time status: Yes, No
- Each combination of these variables (eg, full time, first time college students with an ed goal of degree/transfer as one combination) would constitute an iteration / sample for which to calculate disproportionate impact for outcomes listed in `success_vars` and for the disaggregation variables listed in `group_vars`. The overall rate of success for full time, first time college students with an ed goal of degree/transfer would just include these students and not others. Each variable specified is also collapsed to an 'All' group so that the combinations also reflect all students of a particular category. The total number of combinations for the previous example would be (+1 representing the all category): $(3 + 1) \times (2 + 1) \times (2 + 1) = 36$.
- exclude_scenario_df** (Optional) A data frame with variables that match `scenario_repeat_by_vars` for specifying the combinations to exclude from DI calculations. Following the example specified above, one could choose to exclude part-time non-credit students from consideration.

weight_var	(Optional) A character variable specifying the weight variable if the input data set is summarized (i.e., the the success variables specified in success_vars contain count of successes). Weight here corresponds to the denominator when calculating the success rate. Defaults to NULL for an input data set where each row describes an individual.
include_non_disagg_results	A logical variable specifying whether or not the non-disaggregated results should be returned; defaults to TRUE. When TRUE, a new variable <code>~ None</code> is added to the data set with a single data value <code>'- All'</code> , and this variable is added to group_vars as a disaggregation/group variable. The user would want these results returned to review non-disaggregated results.
ppg_reference_groups	Either <code>'overall'</code> , <code>'hpg'</code> , <code>'all but current'</code> , or a character vector of the same length as group_vars that indicates the reference group value for each group variable in group_vars when determining disproportionate impact using the percentage point gap method.
min_moe	The minimum margin of error to be used in the PPG calculation; see di_ppg .
use_prop_in_moe	(TRUE or FALSE) Whether the estimated proportions should be used in the margin of error calculation by the PPG; see di_ppg .
prop_sub_0	Default is 0.50; see di_ppg .
prop_sub_1	Default is 0.50; see di_ppg .
di_prop_index_cutoff	Threshold used for determining disproportionate impact using the proportionality index; see di_prop_index ; defaults to 0.80.
di_80_index_cutoff	Threshold used for determining disproportionate impact using the 80% index; see di_80_index ; defaults to 0.80.
di_80_index_reference_groups	Either <code>'overall'</code> , <code>'hpg'</code> , <code>'all but current'</code> , or a character vector of the same length as group_vars that indicates the reference group value for each group variable in group_vars when determining disproportionate impact using the 80% index.
check_valid_reference	(TRUE or FALSE) Check whether ppg_reference_groups and di_80_index_reference_groups contain valid values; defaults to TRUE.
parallel	If TRUE, then perform calculations in parallel. The parallel feature is only supported when db_table_name is a path to a parquet file (<code>"/path/to/data.parquet"</code>) and that db_conn is a connection to a duckdb database (e.g., <code>dbConnect(duckdb(), dbdir=':memory:')</code>). Defaults to FALSE.
parallel_n_cores	The number of CPU cores to use if parallel=TRUE. Defaults to half of the maximum number of CPU cores on the system.
mssql_flag	User-specified logical flag (TRUE or FALSE) that indicates if the MS SQL Server variant of the SQL language should be used.

return_what	A character value specifying the return value for the function call. For 'data', the function will return a long data frame with the disproportionate calculations and relevant statistics, after the calculations are performed on the SQL database engine. For 'SQL', a list object of individual queries will be returned for the user to execute elsewhere. Defaults to 'data'.
staging_table	A character value indicating the name of the staging or results table in the database for storing the disproportionate impact calculations.
drop_staging_table	TRUE/FALSE A logical flag indicating whether or not the staging table specified in staging_table should be dropped in the database after the results are returned to R; defaults to TRUE.

Details

Iteratively calculate disproportionate impact via the percentage point gap (PPG), proportionality index, and 80% index methods for all combinations of success_vars, group_vars, and cohort_vars, for each combination of subgroups specified by scenario_repeat_by_vars, using SQL (calculations done on the database engine or duckdb for parquet files).

Value

When return_what='data' (default), a long data frame is returned (see the return value for [di_iterate](#)). When return_what='SQL' (default), a list object where each element is a query (character value) is returned.

di_ppg	<i>Calculate disproportionate impact per the percentage point gap (PPG) method.</i>
--------	-------------------------------------------------------------------------------------

Description

Calculate disproportionate impact per the percentage point gap (PPG) method.

Usage

```
di_ppg(
  success,
  group,
  cohort,
  weight,
  reference = c("overall", "hpg", "all but current", unique(group)),
  data,
  min_moe = 0.03,
  use_prop_in_moe = FALSE,
  prop_sub_0 = 0.5,
  prop_sub_1 = 0.5,
  check_valid_reference = TRUE
)
```

Arguments

success	A vector of success indicators (1/0 or TRUE/FALSE) or an unquoted reference (name) to a column in data if it is specified. It could also be a vector of counts, in which case weight (group size) should also be specified.
group	A vector of group names of the same length as success or an unquoted reference (name) to a column in data if it is specified.
cohort	(Optional) A vector of cohort names of the same length as success or an unquoted reference (name) to a column in data if it is specified. Disproportionate impact is calculated for every group within each cohort. When cohort is not specified, then the analysis assumes a single cohort.
weight	(Optional) A vector of case weights of the same length as success or an unquoted reference (name) to a column in data if it is specified. If success consists of counts instead of success indicators (1/0), then weight should also be specified to indicate the group size.
reference	<p>Either 'overall' (default), 'hpg' (highest performing group), 'all but current' (success rate of everyone excluding the comparison group; also known as 'ppg minus 1'), a value from group (specifying a reference group), a single proportion (eg, 0.50), or a vector of proportions (one for each cohort). Reference is used as a point of comparison for disproportionate impact for each group. When cohort is specified:</p> <ul style="list-style-type: none"> • 'overall' will use the overall success rate of each cohort group as the reference; • 'hpg' will use the highest performing group in each cohort as reference; • 'all but current' will use the calculated success rate of each cohort group excluding the comparison group • the success rate of the specified reference group from group in each cohort will be used; • the specified proportion will be used for all cohorts; • the specified vector of proportions will refer to the reference point for each cohort in alphabetical order (so the number of proportions should equal to the number of unique cohorts).
data	(Optional) A data frame containing the variables of interest. If data is specified, then success, group, and cohort will be searched within it.
min_moe	The minimum margin of error (MOE) to be used in the calculation of disproportionate impact and is passed to ppg_moe . Defaults to 0.03.
use_prop_in_moe	A logical value indicating whether or not the MOE formula should use the observed success rates (TRUE). Defaults to FALSE, which uses 0.50 as the proportion in the MOE formula. If TRUE, the success rates are passed to the proportion argument of ppg_moe .
prop_sub_0	For cases where proportion is 0, substitute with prop_sub_0 (defaults to 0.5) to account for the zero MOE. This is relevant only when use_prop_in_moe=TRUE.
prop_sub_1	For cases where proportion is 1, substitute with prop_sub_1 (defaults to 0.5) to account for the zero MOE. This is relevant only when use_prop_in_moe=TRUE.

check_valid_reference

Check whether reference is a valid value; defaults to TRUE. This argument exists to be used in `di_iterate` as when iterating DI calculations, there may be some scenarios where a specified reference group does not contain any students.

Details

This function determines disproportionate impact based on the percentage point gap (PPG) method, as described in [this](#) reference from the California Community Colleges Chancellor's Office. It assumes that a higher rate is good ("success"). For rates that are deemed negative (eg, rate of drop-outs, high is bad), then consider looking at the converse of the non-success (eg, non drop-outs, high is good) instead in order to leverage this function properly. Note that the margin of error (MOE) is calculated using $1.96 * \sqrt{0.25^2/n}$, with a `min_moe` used as the minimum by default.

Value

A data frame consisting of:

- cohort (if used),
- group,
- n (sample size),
- success (number of successes for the cohort-group),
- pct (proportion of successes for the cohort-group),
- reference_group (reference group used in DI calculation),
- reference (reference value used in DI calculation),
- moe (margin of error),
- pct_lo (lower 95% confidence limit for pct),
- pct_hi (upper 95% confidence limit for pct),
- di_indicator (1 if there is disproportionate impact, ie, when `pct_hi <= reference`),
- success_needed_not_di (the number of additional successes needed in order to no longer be considered disproportionately impacted as compared to the reference), and
- success_needed_full_parity (the number of additional successes needed in order to achieve full parity with the reference).

References

California Community Colleges Chancellor's Office (2017). [Percentage Point Gap Method](#).

Examples

```
library(dplyr)
data(student_equity)
# Vector
di_ppg(success=student_equity$Transfer
        , group=student_equity$Ethnicity) %>% as.data.frame
# Tidy and column reference
di_ppg(success=Transfer, group=Ethnicity, data=student_equity) %>%
```

```

  as.data.frame
# Cohort
di_ppg(success=Transfer, group=Ethnicity, cohort=Cohort
, data=student_equity) %>%
  as.data.frame
# With custom reference (single)
di_ppg(success=Transfer, group=Ethnicity, reference=0.54
, data=student_equity) %>%
  as.data.frame
# With custom reference (multiple)
di_ppg(success=Transfer, group=Ethnicity, cohort=Cohort
, reference=c(0.5, 0.55), data=student_equity) %>%
  as.data.frame
# min_moe
di_ppg(success=Transfer, group=Ethnicity, data=student_equity
, min_moe=0.02) %>%
  as.data.frame
# use_prop_in_moe
di_ppg(success=Transfer, group=Ethnicity, data=student_equity
, min_moe=0.02
, use_prop_in_moe=TRUE) %>%
  as.data.frame

```

di_ppg_iterate	<i>Iteratively calculate disproportionate impact via the percentage point gap (PPG) method for many variables.</i>
----------------	--------------------------------------------------------------------------------------------------------------------

Description

Iteratively calculate disproportionate impact via the percentage point gap (PPG) method for many disaggregation variables.

Usage

```

di_ppg_iterate(
  data,
  success_vars,
  group_vars,
  cohort_vars,
  reference_groups,
  repeat_by_vars = NULL,
  weight_var = NULL,
  min_moe = 0.03,
  use_prop_in_moe = FALSE,
  prop_sub_0 = 0.5,
  prop_sub_1 = 0.5
)

```

Arguments

data	A data frame for which to iterate DI calculation for a set of variables.
success_vars	A character vector of success variable names to iterate across.
group_vars	A character vector of group (disaggregation) variable names to iterate across.
cohort_vars	A character vector of cohort variable names to iterate across.
reference_groups	Either 'overall', 'hpg', or a character vector of the same length as 'group_vars' that indicates the reference group value for each group variable in 'group_vars'.
repeat_by_vars	A character vector of variables to repeat DI calculations for across all combination of these variables, including '- All' as a group for each variable. The reference rate used for DI comparison differs for every combination of the variables listed here.
weight_var	A character scalar specifying the weight variable if the input data set is summarized (ie, the the success variables specified in 'success_vars' contain count of successes). Weight here corresponds to the denominator when calculating the success rate. Defaults to 'NULL' for an input data set where each row describes each individual.
min_moe	The minimum margin of error to be used in the PPG calculation, passed to 'di_ppg'.
use_prop_in_moe	Whether the estimated proportions should be used in the margin of error calculation by the PPG, passed to 'di_ppg'.
prop_sub_0	Passed to 'di_ppg'.
prop_sub_1	Passed to 'di_ppg'.

Details

Iteratively calculate disproportionate impact via the percentage point gap (PPG) method for all combinations of 'success_vars', 'group_vars', and 'cohort_vars', for each combination of subgroups specified by 'repeat_by_vars'.

Value

A data frame with all relevant returned fields from 'di_ppg' plus 'success_variable' (elements of 'success_vars'), 'disaggregation' (elements of 'group_vars'), and 'reference_group' (elements of 'reference_groups').

Examples

```
library(dplyr)
data(student_equity)
# Multiple group variables
di_ppg_iterate(data=student_equity, success_vars=c('Transfer')
, group_vars=c('Ethnicity', 'Gender'), cohort_vars=c('Cohort')
, reference_groups='overall')
```

di_prop_index	<i>Calculate disproportionate impact per the proportionality index (PI) method.</i>
---------------	-------------------------------------------------------------------------------------

Description

Calculate disproportionate impact per the proportionality index (PI) method.

Usage

```
di_prop_index(success, group, cohort, weight, data, di_prop_index_cutoff = 0.8)
```

Arguments

success	A vector of success indicators (1/0 or TRUE/FALSE) or an unquoted reference (name) to a column in data if it is specified. It could also be a vector of counts, in which case weight should also be specified (group size).
group	A vector of group names of the same length as success or an unquoted reference (name) to a column in data if it is specified.
cohort	(Optional) A vector of cohort names of the same length as success or an unquoted reference (name) to a column in data if it is specified. disproportionate impact is calculated for every group within each cohort. When cohort is not specified, then the analysis assumes a single cohort.
weight	(Optional) A vector of case weights of the same length as success or an unquoted reference (name) to a column in data if it is specified. If success consists of counts instead of success indicators (1/0), then weight should also be specified to indicate the group size.
data	(Optional) A data frame containing the variables of interest. If data is specified, then success, group, and cohort will be searched within it.
di_prop_index_cutoff	A numeric value between 0 and 1 that is used to determine disproportionate impact if the proportionality index falls below this threshold; defaults to 0.80.

Details

This function determines disproportionate impact based on the proportionality index (PI) method, as described in [this](#) reference from the California Community Colleges Chancellor's Office. It assumes that a higher rate is good ("success"). For rates that are deemed negative (eg, rate of drop-outs, high is bad), then consider looking at the converse of the non-success (eg, non drop-outs, high is good) instead in order to leverage this function properly.

Value

A data frame consisting of:

- cohort (if used),

- group,
- n (sample size),
- success (number of successes for the cohort-group),
- pct_success (proportion of successes attributed to the group within the cohort),
- pct_group (proportion of sample attributed to the group within the cohort),
- di_prop_index (ratio of pct_success to pct_group),
- di_indicator (1 if di_prop_index < di_prop_index_cutoff), and
- success_needed_not_di (the number of additional successes needed in order to no longer be considered disproportionately impacted as compared to the reference), and
- success_needed_full_parity (the number of additional successes needed in order to achieve full parity with the reference).

When $di_prop_index < 1$, then there are signs of disproportionate impact.

References

California Community Colleges Chancellor's Office (2014). [Guidelines for Measuring Disproportionate Impact in Equity Plans](#).

Examples

```
library(dplyr)
data(student_equity)
di_prop_index(success=Transfer, group=Ethnicity, data=student_equity) %>%
  as.data.frame
```

ppg_moe

Margin of error for the PPG

Description

Calculate the margin of error (MOE) for the percentage point gap (PPG) method.

Usage

```
ppg_moe(n, proportion, min_moe = 0.03, prop_sub_0 = 0.5, prop_sub_1 = 0.5)
```

Arguments

- | | |
|------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| n | Sample size for the group of interest. |
| proportion | (Optional) The proportion of successes for the group of interest. If specified, then the proportion is used in the MOE formula. Otherwise, a default proportion of 0.50 is used (conservative and yields the maximum MOE). |
| min_moe | The minimum MOE returned even if the sample size is large. Defaults to 0.03. This equates to a minimum threshold gap for declaring disproportionate impact. |

prop_sub_0	For cases where ‘proportion’ is 0, substitute with prop_sub_0 (defaults to 0.5) to account for the zero MOE.
prop_sub_1	For cases where ‘proportion’ is 1, substitute with prop_sub_1 (defaults to 0.5) to account for the zero MOE.

Value

The margin of error for the PPG given the specified sample size.

References

California Community Colleges Chancellor’s Office (2017). [Percentage Point Gap Method](#).

Examples

```
ppg_moe(n=800)
ppg_moe(n=c(200, 800, 1000, 2000))
ppg_moe(n=800, proportion=0.20)
ppg_moe(n=800, proportion=0.20, min_moe=0)
ppg_moe(n=c(200, 800, 1000, 2000), min_moe=0.01)
```

ssm_cohort

Long summarized disaggregated data set

Description

Sample data downloaded from the California Community College’s Chancellor’s Office Student Success Metrics dashboard.

Usage

```
data(ssm_cohort)
```

Format

A data frame with summarized data:

value Success count (numerator).

denom Group size (denominator).

categoryLabel Metric or outcome.

academicYear Academic year for given data.

disagg1 Different levels of disaggregation.

subgroup1 Groups corresponding to each disaggregation in disagg1.

disagg2 Second level of disaggregation: ‘None’ or ‘Gender’.

subgroup2 Groups corresponding to each disaggregation in disagg2.

cohort Not actually a cohort, but the time-window for the outcome in categoryLabel.

localeName College name.
metricID ID for current metric.
title Title of visualization.
categoryID ID for categoryLabel.
perc value / denom.
dataType All are 'Percent'.
missingFlag 1 if missing.
ferpaFlag 1 if FERPA-suppressed.
X20 Ignore.
description Ignore.
source Ignore.

Examples

```
data(ssm_cohort)
```

student_equity	<i>Fake data on student equity</i>
----------------	------------------------------------

Description

Data randomly generated to illustrate the use of the package.

Usage

```
data(student_equity)
```

Format

A data frame with 20,000 rows:

Ethnicity ethnicity (one of: Asian, Black, Hispanic, Multi-Ethnicity, Native American, White).

Gender gender (one of: Male, Female, Other).

Cohort year student first enrolled in any credit course at the institution (one of: 2017, 2018).

Transfer 1 or 0 indicating whether or not a student transferred within 2 years of first enrollment (Cohort).

Cohort_Math year student first enrolled in a math course at the institution; could be NA if the student have not attempted math.

Math 1 or 0 indicating whether or not a student completed transfer-level math within 1 year of their first math attempt (Cohort_Math); could be NA if the student have not attempted math.

Cohort_English year student first enrolled in a math course at the institution; could be NA if the student have not attempted math.

English 1 or 0 indicating whether or not a student completed transfer-level English within 1 year of their first math attempt (Cohort_English); could be NA if the student have not attempted English.

Ed_Goal student's educational goal (one of: Deg/Transfer, Other).

College_Status student's educational status (one of: First-time College, Other).

Student_ID student's unique identifier.

EthnicityFlag_Asian 1 (yes) or 0 (no) indicating whether or not a student self-identifies as Asian.

EthnicityFlag_Black 1 (yes) or 0 (no) indicating whether or not a student self-identifies as Black.

EthnicityFlag_Hispanic 1 (yes) or 0 (no) indicating whether or not a student self-identifies as Hispanic.

EthnicityFlag_NativeAmerican 1 (yes) or 0 (no) indicating whether or not a student self-identifies as Native American.

EthnicityFlag_PacificIslander 1 (yes) or 0 (no) indicating whether or not a student self-identifies as Pacific Islander.

EthnicityFlag_White 1 (yes) or 0 (no) indicating whether or not a student self-identifies as White.

EthnicityFlag_Caribbean 1 (yes) or 0 (no) indicating whether or not a student self-identifies as Caribbean.

EthnicityFlag_EastAsian 1 (yes) or 0 (no) indicating whether or not a student self-identifies as East Asian.

EthnicityFlag_SouthEastAsian 1 (yes) or 0 (no) indicating whether or not a student self-identifies as Southeast Asian.

EthnicityFlag_SouthWestAsianNorthAfrican 1 (yes) or 0 (no) indicating whether or not a student self-identifies as Southwest Asian / North African (SWANA).

EthnicityFlag_AANAPI 1 (yes) or 0 (no) indicating whether or not a student self-identifies as Asian-American or Native American Pacific Islander (AANAPI).

EthnicityFlag_Unknown 1 (yes) or 0 (no) indicating whether or not a student self-identifies as Unknown.

EthnicityFlag_TwoorMoreRaces 1 (yes) or 0 (no) indicating whether or not a student self-identifies as two or more races.

Examples

```
data(student_equity)
```

```
surround_quote_if_needed
```

Helper function: Surround character values with double quotes if not present.

Description

Function used internally by [di_calc_sql](#) and [di_iterate_sql](#) to surround variable names by double quotes in SQL queries in order to support non-alphanumeric characters in variable names.

Usage

`surround_quote_if_needed(value)`

Arguments

`value` A character vector.

Value

A character vector with double quotes surrounding `value` if the first and last characters of `value` aren't yet double quotes. For `value` that is already surrounded by double quotes, nothing is changed.

Index

* datasets

ssm_cohort, 26

student_equity, 27

as.data.table, 5, 12

collapse, 11, 13

data.table, 4, 5, 11–13

dbConnect, 17

di_80_index, 2, 5, 7, 10, 11, 13, 16, 18

di_calc_dt, 4

di_calc_sql, 6, 28

di_iterate, 3, 8, 13, 15, 19, 21

di_iterate_dt, 4, 11

di_iterate_on_long, 14

di_iterate_sql, 6, 16, 28

di_ppg, 5, 7, 9, 11, 13, 16, 18, 19

di_ppg_iterate, 22

di_prop_index, 5, 7, 10, 11, 13, 16, 18, 24

duckdb, 18

mclapply, 10, 13

parLapply, 10, 13

ppg_moe, 20, 25

ssm_cohort, 26

student_equity, 27

surround_quote_if_needed, 28