**R topics documented:**

- diversity ............................................................. 2
- div_add_median_label ............................................. 3
- div_ci_median .......................................................... 4
- div_conf_colour ...................................................... 4
- div_fake_team .......................................................... 5
- div_gauge_plot .......................................................... 6
- div_parse_paygap ...................................................... 7
- div_paygap ............................................................ 8
- div_plot_paygap_distribution ....................................... 8
- div_round_paygap ...................................................... 9
- print.paygap .......................................................... 10
- summary.paygap ........................................................ 10

---

**diversity**

*Calculate the diversity index*

**Description**

This function calculates the entropy of a system with discrete states.

**Usage**

```r
diversity(x, prior = NULL)
```

**Arguments**

- `x` numeric vector, observed probabilities of the classes
- `prior` numeric vector, the prior probabilities of the classes

**Value**

the entropy or diversity measure

**Examples**

```r
x <- c(0.4, 0.6)
diversity(x)
```
div_add_median_label   *Adds a column with new labels (H)igh and (L) for a given colName (within a given grade and jobID)*

**Description**

This function calculates the entropy of a system with discrete states.

**Usage**

```r
div_add_median_label(
  d,
  colName = "age",
  value1 = "T",
  value2 = "F",
  newColName = "isYoung"
)
```

**Arguments**

- **d**  
  *tibble*, a tibble with team data columns as defined in the documentation (at least the column colName (as set by next parameter), ‘grade’, and ‘jobID’)

- **colName**  
  the name of the columns that contains the factor object to be used as explaining dimension for the paygap (defaults to ‘gender’)

- **value1**  
  character, the label to be used for the first half of observations (the smallest ones)

- **value2**  
  character, the label to be used for the second half of observations (the biggest ones)

- **newColName**  
  the value in new column name that will hold the values value1 and value2

**Value**

dataframe (with columns grade, jobID, salary_selectedValue, salary_others, n_selectedValue, n_others, paygap, confidence), where "confidence" is one of the following: NA = not available (numbers are too low), "" = no bias detectable, "." = there might be some bias, but we’re not sure, "#" = bias detected with some degree of confidence, "##" = quite sure there is bias, "###" = trust us, this is biased.

**Examples**

```r
df <- div_add_median_label(div_fake_team())
colnames(df)
```
div_ci_median

*Function to calculate the confidence interval for the median*

**Description**

Function to calculate the confidence interval for the median

**Usage**

```
div_ci_median(x, conf = 0.95)
```

**Arguments**

- `x` numeric, data from which the median is calculated
- `conf` numeric, the confidence interval as 1 - P(x < x0)

**Value**

`ci` (confidence interval object)

**Examples**

```r
x <- 1:100
div_ci_median(x)
```

---

div_conf_colour

*return a colour code given a number of stars for the confidence level of bias*

**Description**

This function returns a colour (R named colour) based on the confidence level

**Usage**

```
div_conf_colour(x)
```

**Arguments**

- `x` the string associated to the paygap confidence: NA, `.`, `·`, `*`, `***`, `***`

**Value**

string (named colour)

**Examples**

```r
div_conf_colour("*")
```
Description

This function generates a data frame with data for a team (with salaries, gender, FTE, etc). This is a good start to test the package and to experiment what level of bias will be visible in the paygap for example.

Usage

```r
div_fake_team(
  seed = 100,
  N = 200,
  genders = c("F", "M", "O"),
  gender_prob = c(0.4, 0.58, 0.02),
  gender_salaryBias = c(1, 1.1, 1),
  jobIDs = c("sales", "analytics"),
  jobID_prob = c(0.6, 0.4),
  citizenships = c("Polish", "German", "Italian", "Indian", "Other"),
  citizenship_prob = c(0.6, 0.2, 0.1, 0.05, 0.05)
)
```

Arguments

- `seed` numeric, the seed to be used in set.seed()
- `N` numeric, the size of the team to be used (default = 200)
- `genders` character, a vector of the genders to be used
- `gender_prob` numeric, relative probabilities of the different genders to occur (must have the same length as `genders`)
- `gender_salaryBias` numeric, vector with the relative salaries of the different genders (must have the same length as `genders`)
- `jobIDs` character, a vector with the labels of the job categories in the team (they will appear in each grade)
- `jobID_prob` numeric, a vector with the relative sizes of the different jobs in the team (must have the same length as `jobIDs`)
- `citizenships` character, a vector of the citizenships to be generated
- `citizenship_prob` numeric, relative probabilities of the different citizenships to occur (must have the same length as `citizenships`)

Value
dataframe (employees of the random team)
div_gauge_plot

Uses ggplot2 to produce a gauge plot in RAG colour

Description

This function produces one or more gauge plots coloured in red (R), amber (A) or green (G) for a value between 0 and 1.

Usage

```r
div_gauge_plot(df, breaks = c(0, 0.8, 0.95, 1), ncol = NULL, nbrSize = 6)
```

Arguments

- **df**: tibble, a tibble with columns "value" and "label" (value = the values between 0 and 1; label = text to show e.g. paste("group", colnames(t))
- **breaks**: numeric vector with the lower limit, the border between green and amber, the border between amber and red, and the upper limit
- **ncol**: numeric, the number of columns to produce
- **nbrSize**: numeric, the font size for the label

Value

ggplot object

Examples

```r
d <- div_fake_team()
head(d)
diversity(table(d$gender))

div_gauge_plot(d, breaks = c(0, 0.8, 0.95, 1), ncol = NULL, nbrSize = 6)
```
Prepare the paygap matrix to be published in LaTeX

**Description**

This function formats the paygap matrix (created by `div_paygap()`) and prepares it for printing via the function `knitr::kable()`.

**Usage**

```r
div_parse_paygap(
  pg,
  label = NULL,
  min_nbr_show = NULL,
  max_length_jobID = 12,
  max_length_colnames = 9
)
```

**Arguments**

- `pg`: paygap object as created by `div::div_paygap()`. This is an S3 object with a specific structure.
- `label`: character, the label to be used in the caption of the `kable` object.
- `min_nbr_show`: numeric, if provided then only groups that have more than `min_nbr_show` employees in both categories (`selectedValue` and `others`) will be shown.
- `max_length_jobID`: numeric, if provided the maximal length of the column `jobID` (in characters).
- `max_length_colnames`: numeric, if provided the maximal length of the column `names` (in characters).

**Value**

`knitr::kable` object (for LaTeX)

**Examples**

```r
d <- div_fake_team()
pq <- div_paygap(d)
div_parse_paygap(pq)
```
### div_paygap

*Function to calculate the paygap as a ratio.*

#### Description

This function calculates the entropy of a system with discrete states.

#### Usage

```r
div_paygap(d, x = "gender", y = "salary", x_ctrl = "F", ctrl_var = "age")
```

#### Arguments

- `d`: tibble, a tibble with columns as defined
- `x`: the name of the columns that contains the factor object to be used as explaining dimension for the paygap (defaults to 'gender')
- `y`: the name of the columns that contains the numeric value to be used to calculate the paygap (could be salary or bonus for example)
- `x_ctrl`: the value in the column defined by `x` that should be isolated (this versus the others), defaults to 'F'
- `ctrl_var`: a control variable to be added (shows median per group for that variable)

#### Value

dataframe (with columns grade, jobID, salary_x_ctrl, salary_others, n_x_ctrl, n_others, paygap, confidence), where "confidence" is one of the following: NA = not available (numbers are too low), "" = no bias detectable, "." = there might be some bias, but we're not sure, "*" = bias detected with some degree of confidence, "**" = quite sure there is bias, "***" = trust us, this is biased.

#### Examples

```r
df <- div_paygap(div_fake_team())
df
```

---

### div_plot_paygap_distribution

*Produce a histogram and normal distribution*

#### Description

Plots a histogram, a normal distribution with the same standard deviation and mean as well as one with a mean centred around 1.
**div_round_paygap**

**Usage**

```r
div_plot_paygap_distribution(x, label = "Gender", mu_unbiased = 1)
```

**Arguments**

- **x**: numeric vector, column of paygap observations
- **label**: character, prefix for the title
- **mu_unbiased**: numeric, the mean of the unbiased distribution (for paygaps this should be 1)

**Value**

ggplot2 object

**Examples**

```r
d <- div_fake_team()
pg <- div_paygap(d)
div_plot_paygap_distribution(pg$data$paygap)
```

---

**div_round_paygap**

*Rounds all numbers in the paygap data-frame*

**Description**

This function all numbers to zero decimals, except the paygap (which is rounded to 2 decimals):

**Usage**

```r
div_round_paygap(x)
```

**Arguments**

- **x**: paygap object (output of div::div_paygap())

**Value**

the paygap data-frame (tibble only, not the whole paygap object)

**Examples**

```r
d <- div_fake_team()
pg <- div_paygap(d)
div_round_paygap(pg)
```
**summary.paygap**

*Title*

**Description**

print the paygap object in the terminal

**Usage**

```r
## S3 method for class 'paygap'
print(x, ...)
```

**Arguments**

- `x` paygap object, as created by the function `div_paygap()`
- `...` arguments passed on to the generic print function: `print(x$data)`

**Value**

text output

**Examples**

```r
library(div)
div_fake_team() %>%
div_paygap %>%
print
```

**summary.paygap**

*Title*

**Description**

Title

**Usage**

```r
## S3 method for class 'paygap'
summary(object, ...)
```

**Arguments**

- `object` paygap S3 object, as created by the function `dif_paygap()`
- `...` passed on to `summary()`
Value

a summary of the paygap object

Examples

```r
library(div)
d <- div_fake_team()
pg <- div_paygap(d)
summary(pg)
```
Index

* (seniority)
  div_add_median_label, 3
  div_paygap, 8
* (type)
  div_add_median_label, 3
  div_paygap, 8
* -
  div_add_median_label, 3
  div_paygap, 8
* and
  div_add_median_label, 3
  div_paygap, 8
* a
  div_add_median_label, 3
  div_paygap, 8
* by
  div_add_median_label, 3
  div_paygap, 8
* calculate
  div_add_median_label, 3
  div_paygap, 8
* controlled
  div_add_median_label, 3
  div_paygap, 8
* dataset
  div_add_median_label, 3
  div_paygap, 8
* distribution
  div_plot_paygap_distribution, 8
* d
  div_add_median_label, 3
  div_paygap, 8
* entropy
  diversity, 2
* equity
  div_plot_paygap_distribution, 8
* for
  div_add_median_label, 3
  div_paygap, 8
  div_add_median_label, 3
  div_paygap, 8
INDEX

summary.paygap. 10