The mechanics of `functions.R`

Addison Larson

2/6/2019

Functions

Some of these functions are long and are called in multiple scripts. This file is meant to group all functions together and reduce the amount of code overall.

Override `base` and `stats` function defaults

A time-saver so that it's not required to call `na.rm = TRUE` every time these functions are called in other scripts.

```
min <- function(i, ..., na.rm = TRUE) {
  base::min(i, ..., na.rm = na.rm)
}
mean <- function(i, ..., na.rm = TRUE) {
  base::mean(i, ..., na.rm = na.rm)
}
sd <- function(i, ..., na.rm = TRUE) {
  stats::sd(i, ..., na.rm = na.rm)
}
max <- function(i, ..., na.rm = TRUE) {
  base::max(i, ..., na.rm = na.rm)
}
```

Create custom half-standard deviation breaks

I like to call this IPD's "bespoke classification scheme." For a given vector of numbers `x` and a number of bins `i`, `st_dev_breaks` computes the bin breaks starting at $-0.5 \cdot stdev$ and $0.5 \cdot stdev$. For the purposes of IPD analysis, `i = 5`, and `st_dev_breaks` calculates the minimum, $-1.5 \cdot stdev$, $-0.5 \cdot stdev$, $0.5 \cdot stdev$, $1.5 \cdot stdev$, and maximum values. These values are later used to slice the vector into five bins.

```
st_dev_breaks <- function(x, i, na.rm = TRUE){
  half_st_dev_count <- c(-1 * rev(seq(1, i, by = 2)),
                         seq(1, i, by = 2))
  if((i %% 2) == 1) {
    half_st_dev_breaks <- sapply(half_st_dev_count,
                                 function(i)(0.5 * i * sd(x)) + mean(x))
    half_st_dev_breaks[[1]] <- ifelse(min(x) < half_st_dev_breaks[[1]],
                                      min(x), half_st_dev_breaks[[1]])
    half_st_dev_breaks[[i + 1]] <- ifelse(max(x) > half_st_dev_breaks[[i + 1]],
                                          max(x), half_st_dev_breaks[[i + 1]])
  } else {
    half_st_dev_breaks <- NA
  }
  return(half_st_dev_breaks)
}
```

Move column or vector of columns to last position

The requested schema for IPD data export renames and places all relevant universes in the final columns of the dataset. `move_last` moves a column or vector of columns to the last position in a tibble or data frame.

```
move_last <- function(df, last_col) {
  match(c(setdiff(names(df), last_col), last_col), names(df))
}
```

Summarize data

The requested summary tables of IPD data call for more than `base::summary` exports. `description` tailors the exports from `summarytools::descr` to create summary tables with the requested fields.

```
description <- function(i) {
  summarytools::descr(i, na.rm = TRUE, stats = c("min", "med", "mean", "sd", "max"))
}
```