Functions, operators, and arguments

Jeffrey R. Stevens

2022-01-21

Data types

- `>`, `>=`, `<`, `<=`, `==`, `!=`, `%in%`: logical operators that output `TRUE` or `FALSE`
- `typeof()`, `class()`, `str()`: outputs object type, class, and structure
- `as.numeric()`, `as.character()`, `as.factor()`: coerces (converts) object to numeric, character, factor
- `[]`: index elements in vector, matrix, data frame, tibble
- `$`: index column by name in data frame, tibble, list
- `seq()`, `rep()`: creates sequences and repetitions of numbers
- `length()`: outputs length of vector
- `dim()`, `nrow()`, `ncol()`: outputs dimensions, number of rows, number of columns of matrices, data frames, tibbles
- `names()`: outputs (and can assign) column names
- `head()`, `tail()`, `dplyr::glimpse()`: outputs compressed views of data frames, tibbles
- `c()`, `matrix()`, `data.frame()`, `tibble::tibble()`: creates vectors, matrices, data frames, tibbles

Importing data

- `here::here()`: starts path at project directory
- `read.csv()`, `write.csv()`, `readr::read_csv()`, `readr::write_csv()`: imports and writes CSV files
- `readxl::read_excel()`: imports Excel files
- `skimr::skim()`: outputs overview of data
- `dataReporter::makeCodebook()`: creates codebook of data

Cleaning columns (`{dplyr}`)

- `select()`: selects subset of columns from data frame, tibble
- `everything()`, `contains()`, `starts_with()`: helper functions for `select()`
- `relocate()`, `rename()`: moves and renames columns in data frame, tibble
- `mutate()`, `transmute()`: applies function to change existing column or create new column
- `across()`: applies function across multiple columns inside `mutate()`
- `rowwise()`: applies function to each row
- `%>%`: pipe operator that transfers output to the next command
- `pull()`: creates a vector from a data frame/tibble column

Wrangling rows (`{dplyr}`)

- `filter()`: filters subset of rows from data frame, tibble
- `is.na()`: checks whether object is `NA` and outputs logical
- `arrange()`, `desc()`: sorts rows by column variable, in descending order
- `group_by()`: groups data by column levels
- `summarise()`: applies function over whole column or group

Tidy data (`{tidyr}`)

- `pivot_longer()`, `pivot_wider()`: reshapes data to be longer or wider
- `separate()`, `unite()`: separates or combines column data with separator
- `complete()`, `expand()`, `nesting()`: finds all unique combinations of levels

Merging data (`{dplyr}` and `{tidyr}`)

- `bind_rows()`, `bind_cols()`: binds rows or columns to data frame
- `inner_join()`, `left_join()`, `right_join()`, `outer_join()`: mutating joins that merge data frames
- `semi_join()`, `anti_join()`: filtering joins that filter data frame based on another data frame

Strings (`{stringr}`)

- `str_length()`: finds the number of characters in a string
- `str_trunc()`, `str_pad()`: removes or adds characters to strings
- `str_trim()`, `str_squish()`: removes whitespace from strings
- `str_c()`: combine character vectors into single string
- `str_sub()`: extracts parts of strings based on character position
- `str_to_lower()`, `str_to_upper()`: converts all letters to lowercase or uppercase
- `str_to_title()`, `str_to_sentence()`: converts strings to title or sentence case
- `str_detect()`, `str_subset()`, `str_extract()`: detects, subsets, and extracts strings
- `str_replace()`, `str_replace_all()`: replaces patterns with strings
- `str_split()`: splits strings based on separators
- `str_glue()`, `str_glue_data()`: combines strings with R output

Factors (`{forcats}`)

- `base::levels()`: prints factor levels
- `fct_inorder()`, `fct_rev()`: orders levels by order in data or in reverse of current order
- `fct_relevel()`: manually reorders levels
- `fct_reorder()`: orders levels based on another variable
- `fct_recode()`: recodes level with new value
- `fct_collapse()`: recodes multiple levels into single new value
- `fct_lump()`: lumps infrequent levels into level "Other"

Dates (`{lubridate}`)

- `today()`, `now()`: print today's date or time
- `mdy()`, `dmy()`, `ymd()`: convert various date formats to YYYY-MM-DD
- `mdy_hm()`, `mdy_hms()`: converts various date/time formates to YYYY-MM-DD HH:MM:SS
- `year()`, `month()`, `day()`, `wday()`: extracts year, month, day, or weekday from date

Functions

- `function()`: creates new function
- `if()`, `else()`: creates conditional operations in functions
- `ifelse()`: creates conditional operations over vectors
- `for()`: creates iterations over sequences or vectors

Grammar of graphics ({ggplot2})

- `ggplot()`: creates a ggplot
- `+`: pipe operator for ggplots
- `aes()`: defines aesthetic properties of plot
- `color`, `fill`, `shape`, `size` arguments: properties for geometric objects
- `ggsave()`: saves ggplot to file

Visualizing distributions ({ggplot2})

- `geom_histogram()`: plots histograme
- `geom_density()`: plots density plot
- `geom_boxplot()`: plots boxplot
- `geom_violin()`: plots violin plot
- `stat_summary()`: plots summaries of data (e.g., means ± standard error)

Visualizing amounts and proportions ({ggplot2})

- `count()`: calculates counts of data by variables
- `geom_bar()`: plots bar plot with raw data
- `geom_col()`: plots bar plot with counts
- `position` argument: controls whether data are stacked, dodged, jittered, nudged
- `geom_point()`: plots scatterplots
- `coord_flip()`: flips x and y coordinates
- `coord_polar()`: converts to polar coordinates
- `geom_linerange()`: plots point and error bar

Visualizing x-y data ({ggplot2})

- `geom_abline()`: plots line with slope and intercept
- `base::pairs()`: plots correlation plots
- `GGally::ggpairs()`: plots correlation plots
- `geom_tile()`: plots tile plot
- `ggcorrplot::ggcorrplot()`: plots correlation heatmaps
- `geom_line()`: plots line plot
- `geom_area()`: plots area under curve or line plot
- `geom_smooth()`: plots fitted lines and curves
- `geom_rug()`: plots rug plot

Color ({ggplot2})

- `scale_color_brewer()`, `scale_fill_brewer()`: uses existing qualitative colors scales for color and fill
- `scale_color_manual()`, `scale_fill_manual()`: sets manual colors for color and fill
- `scale_color_gradient()`, `scale_fill_gradient()`: sets sequential color gradient for color and fill
- `scale_color_distiller()`, `scale_fill_distiller()`: sets diverging color scale for color and fill

Finessing plots ({ggplot2})

- `geom_jitter()`: plots jittered scatterplot

- `ggbeeswarm::geom_beeswarm()`: plots beeswarm plot
- `scale_x_discrete()`, `scale_y_discrete()`: adjusts discrete scale properties (e.g., limits, ticks)
- `scale_x_continous()`, `scale_y_continuous()`: adjusts continuous scale properties (e.g., limits, ticks)
- `lims()`, `xlim()`, `ylim()`: adjusts axis limits
- `facet_wrap()`, `facet_grid()`: creates facets based on discrete variables

Adorning plots (`{ggplot2}`)

- `labs()`, `xlab()`, `ylab()`: replaces axis labels
- `annotate(geom = c("text", "segment", "rect"))`: annotates plot with text, segments, rectangles, etc.
- `geom_text()`: plots text as aesthetic property
- `geom_hline()`, `geom_vline()`: plots horizontal and vertical reference lines
- `stat_ellipse()`: plots ellipse around data

Tables

- `knitr::kable()`: creates table from data frame
- `kableExtra::kable_styling()`: styles table
- `kableExtra::pack_rows()`, `kableExtra::add_header_above()`: adds grouping variables to rows or columns
- `kableExtra::footnote()`: adds table note
- `kableExtra::landscape()`: rotates table to landscape orientation
- `papaja::apa_table()`: formats data frame to APA style table
- `papaja::apa_print()`: formats statistics to APA style