

Day 5 Cheatsheet

Data Cleaning

Major concepts

- **Most important rule of data handling - Always be looking at your data!**
- **NA** - general missing data
- **NaN** - stands for “Not a Number”, happens when you do 0/0.
- **Inf** and **-Inf** - Infinity, happens when you take a positive number (or negative number) by 0.

Functions

| Library/Package | Piece of code | Example of usage | What it does |
|-----------------|------------------------------|---|---|
| Base R | <code>is.na(x)</code> | <code>is.na(x)</code> | checks if <code>x</code> is NA . |
| Base R | <code>is.nan(x)</code> | <code>is.nan(x)</code> | checks if <code>x</code> is NaN . |
| Base R | <code>is.infinite(x)</code> | <code>is.infinite(x)</code> | checks if <code>x</code> is Inf or -Inf . |
| naniar | <code>pct_complete(x)</code> | <code>pct_complete(x)</code> | Reports the percentage of data that is complete in <code>x</code> . |
| naniar | <code>gg_miss_var(x)</code> | <code>gg_miss_var(x)</code> | Reports as a plot the percentage of data that is complete in <code>x</code> . |
| tidyr | <code>drop_na(df)</code> | <code>drop_na(df)</code> | Drops rows of NA from a given data frame/tibble |
| dplyr | <code>case_when()</code> | <code>df <- arrange(df, mpg)</code> | This function allows you to vectorise multiple <code>if_else()</code> statements. If no cases match, NA is returned. |
| dplyr | <code>mutate()</code> | <code>df <- mutate(df, newcol = wt/2.2)</code> | Adds a new column that is a function of existing columns |
| dplyr | <code>separate()</code> | <code>df %>% separate(x, c("A", "B"))</code> | Separate a character column into multiple columns with a regular expression or numeric locations |
| dplyr | <code>unite()</code> | <code>df %>% unite("z", x:y, remove = FALSE)</code> | Unite multiple columns together into one column |
| stringr | <code>str_detect</code> | <code>df %>% filter(str_detect(col_name, "string_pattern"))</code> | Returns logical vector to indicate if string pattern was detected |
| stringr | <code>str_replace</code> | <code>str_replace(vector, "replace_me", "with_me")</code> | Replaces all instances of one specified string with another specified string |

* This format was adapted from the cheatsheet format from AlexsLemonade.