Data Wrangling with dplyr and tidyr

Cheat Sheet

Syntax - Helpful conventions for wrangling

dplyr::tbl_df(iris)
Converts data to tbl class. tbl’s are easier to examine than data frames. R displays only the data that fits onscreen:

Source: local data frame [150 x 5]
Sepal.Length Sepal.Width Petal.Length Species
1 5.1 3.5 1.4 setosa
2 4.9 3.0 1.4 versicolor
3 4.7 3.2 1.3 virginica
4 4.6 3.1 1.5 setosa
5 5.0 3.6 1.4 versicolor
...
Variables not shown: Petal.Width (dbl), Species (fct)

dplyr::glimpse(iris)
Information dense summary of tbl data.

utils::View(iris)
View data set in spreadsheet-like display (note capital V).

dplyr::%>%%
"Piping" with %>% makes code more readable, e.g.
iris %>% group_by(Species) %>% summarise(avg = mean(Sepal.Width))

Reshaping Data - Change the layout of a data set

tidyr::gather(cases, "year", "n", 2:4)
Gather columns into rows.
tidyr::spread(pollution, size, amount)
Spread rows into columns.
tidyr::separate(storms, date, c("y", "m", "d"))
Separate one column into several.
tidyr::unite(data, col, ..., sep)
Unite several columns into one.

dplyr::data_frame(a = 1:3, b = 4:6)
Combine vectors into data frame (optimized).
dplyr::arrange(mtcars, mpg)
Order rows by values of a column (low to high).
dplyr::arrange(mtcars, desc(mpg))
Order rows by values of a column (high to low).
dplyr::rename(tb, y = year)
Rename the columns of a data frame.

Subset Observations (Rows)

dplyr::filter(iris, Sepal.Length > 7)
Extract rows that meet logical criteria.
dplyr::distinct(iris)
Remove duplicate rows.
dplyr::sample_frac(iris, 0.5, replace = TRUE)
Randomly select fraction of rows.
dplyr::sample_n(iris, 10, replace = TRUE)
Randomly select n rows.
dplyr::slice(iris, 10:15)
Select rows by position.
dplyr::top_n(storms, 2, date)
Select and order top n entries (by group if grouped data).

Logic in R - ?Comparison, ?base::Logic

Helper functions for select - ?select
select(iris, contains(""))
Select columns whose name contains a character string.
select(iris, starts_with("Length"))
Select columns whose name starts with a character string.
select(iris, everything())
Select every column.
select(iris, matches("t"))
Select columns whose name matches a regular expression.
select(iris, num_range("5", 1:5))
Select columns named x1, x2, x3, x4, x5.
select(iris, one_of(list("Species", "Genus")))
Select columns whose names are in a group of names.
select(iris, starts_with("Sepal"))
Select columns whose name starts with a character string.
select(iris, Sepal.Length:Petal.Width)
Select all columns between Sepal.Length and Petal.Width (inclusive).
select(iris, -Species)
Select all columns except Species.
Summarise Data

- `summarise(iris, avg = mean(Sepal.Length))`
- `summarise_each(iris, funs(mean))`
- `count(iris, Species, wt = Sepal.Length)`
- `summarise_each(iris, funs(mean))`
- `summarise(iris, avg = mean(Sepal.Length))`

Summarise uses **summary functions**, functions that take a vector of values and return a single value, such as:

- `min`: Minimum value in a vector.
- `max`: Maximum value in a vector.
- `mean`: Mean value of a vector.
- `median`: Median value of a vector.
- `sd`: Standard deviation of a vector.
- `IQR`: Interquartile range of a vector.

Make New Variables

- `mutate(iris, sepal = Sepal.Length + Sepal.Width)`
- `mutate_each(iris, funs(min_rank))`

Mutate uses **window functions**, functions that take a vector of values and return another vector of values, such as:

- `cummean`: Cumulative mean
- `cumsum`: Cumulative sum
- `cummax`: Cumulative max
- `cummin`: Cumulative min
- `cumprod`: Cumulative product
- `pmax`: Element-wise max
- `pmin`: Element-wise min

Group Data

- `group_by(iris, Species)`
- `ungroup(iris)`

Group data into rows with the same value of Species.

Combine Data Sets

- `left_join(a, b, by = "x1")`
- `right_join(a, b, by = "x1")`
- `inner_join(a, b, by = "x1")`
- `full_join(a, b, by = "x1")`

Filtering Joins

- `semi_join(a, b, by = "x1")`
- `anti_join(a, b, by = "x1")`

Set Operations

- `intersect(y, z)`
- `union(y, z)`
- `setdiff(y, z)`

Binding

- `bind_rows(y, z)`
- `bind_cols(y, z)`

Caution: matches rows by position.