Data Wrangling: Exercises

Erasmus Q-Intelligence B.V.

Use the `fligths` data from the `nycflights13`-package. The data set contains information on flights from airports in New York in 2013.

Exercise 1. `filter()`

(a) Find all flights without any delay on departure (`dep_delay`) and with at least 2 hours delay on arrival (`arr_delay`)

(b) Find all fligths of carriers United (UA), American (AA) and Delta (DL)

Exercise 2. `arrange()`

(a) Sort the `flights` dataset in such a way that flights with the highest delay at arrival (`arr_delay`) come first.

(b) Sort the `flights` dataset in such a way that flights with `tailnum` equal to NA come first.

Exercise 3. `select()`

(a) Select all columns that end with `_time` or with `_delay`.

Exercise 4. `mutate()`

(a) The `dep_time` column is formatted like a digital clock, where 213 actually means 2:13. Use `mutate()` to create a column `dep_time_mins` which gives the total number of minutes since midnight.

Exercise 5. `summarise()` and `group_by()`

(a) Calculate the number of canceled flights by month, the total number of flights per month and the percentage of canceled flights per month. Sort the new `data.frame` such that months with the highest percentage of canceled flights come first. Canceled fligths either have no `dep_time` or do not have `arr_time`.

(b) Calculate the minimum, maximum, mean, median and standard deviation of `air_time` by destination. Sort the new `data.frame` such that destinations with the highest mean `air_time` comes first.

Exercise 6. `group_by()`, `filter()` and `mutate()`

a Add a column `dep_delay_lag` to `fligths`, with `dep_delay` of the previous fligth from that airport (see `?lag` to find out how to store the value of row x-1 in row x).

b Filter the observations such that you only have those where neither `dep_delay`, nor `dep_delay_lag` is missing.

c Find the average of `dep_delay`, separate for each value of `dep_delay_lag` and for each `origin` using `group_by`. Name this variable `dep_delay_avg`.

d Visualize the relation (using a scatter plot) between `dep_delay_lag` and `dep_delay_avg`, where the color of the point depends on `origin`.

Exercise 7. combined
Consider the `forsale.Rds`-dataset from Canvas. It contains information about houses for sale in Rotterdam area during the recent past.

a Load the data from Canvas.

b Make a `data.frame` of the first 10 observations, with only the variables `postcode`, `city`, `suburb`, `asking_price` and `living_area`.

c Add a variable to the original `data.frame` `forsale` that contains the price per square metre. Call this column `price_m2`.

d Calculate the number of houses in this dataset in each city, as well as the minimum, average and maximum asking price. Which city is the cheapest, on average?

e Calculate the ratio between each house's price per square metre and the mean price per square metre in that house's 4-digit postcode. Call this new variable `price_premium`.

f Create a subset of the `forsale` dataset from (e), which contains only houses that satisfy the following criteria: it is within the city of Rotterdam, it has at least 3 bedrooms and 2 bathrooms and costs no more than 400.000 euro.

g Sort the new `data.frame` such that those with the highest `price_premium` occur first. What does a high `price_premium` mean?

Exercise 8. `across()`

a Load the penguins-data from `penguins.Rds`, available on Canvas. The data is on penguin species in Palmer Archipelago.

b Get the number of distinct categories for all variables, using functions `across()` and `n_distinct()`.

c Get the number of distinct categories, only for `species`, `island` and `sex`.

d Get the number of missing values, only for the variables of type `factor` (categorical variables).

e Get the number of missing values for all variables with `length` in their name.

f **Bonus** Transform all variables where `mm` is in the name to be relative to the species-specific mean of that variable (divide by the species-specific mean). Hence, you will need to use `group_by()`, `mutate()` and `across()`.

Exercise 9. `join`

(a) Add location (`lat` and `lon`) from the `airport` data set (available from the `nycflights13`-package) to the `flights-data.frame`. Make sure you only add these two columns. Which `key` should be used?

(b) **Bonus** Make the following `data.frame`: Full name of carrier (`name`), number of active planes (`nr_active_planes`) and average build year (`mean_build_year`). Add a 4th column with the number of planes of which the build year is unkown (`unknown_planes`). Think carefully: which `data.frames` from the `nycflights13`-package do you need, how do you merge them, which columns do you need?

Exercise 10. `tidyr`

(a) Load the data `pivot_example.Rds` from Canvas. The data is on the distribution of male and female in the US navy.

(b) Gather the information in the columns on the amount of persons in one column, named `gender_marital`, where the value is stored in the column `amount`.

Exercise 11. `tidyr`

(a) Load the data on NBA players and points scored (fictive) from Canvas.

(b) Gather the columns `day1points` and `day2points` into a new column `day` with the values in the new column `points`.