Case Study 2 AKSTA Statistical Computing

Hanna Kienast Iulia Mihaela Enache Kateryna Ponomarenko

2024-04-30

Task

a

Load in R the following data sets which you can find in TUWEL. For each data set, ensure that missing values are read in properly, that column names are unambiguous. Each data set should contain at the end only two columns: country and the variable.

```
library(readr)
```

```
## Warning: package 'readr' was built under R version 4.3.2
```

```
library(tidyr)
```

```
## Warning: package 'tidyr' was built under R version 4.3.2
```

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(readxl)
```

```
## Warning: package 'readxl' was built under R version 4.3.3
```

```
library(data.table)
```

```
## Warning: package 'data.table' was built under R version 4.3.2
```

```
##
## Attaching package: 'data.table'

## The following objects are masked from 'package:dplyr':
##
##     between, first, last
```

```r
require(stringr)
```

```
## Loading required package: stringr
```

```r
median_age_data <- read.fwf("rawdata_343.txt", widths = c(8,66,4,18), skip=2)
names(median_age_data) <- c("Rank", "Country", "Median_Age")
median_age_data <- median_age_data[, c("Country", "Median_Age")]
head(median_age_data)
```

```
##                                                               Country Median_Age
## 1 Monaco                                                                    55.4
## 2 Japan                                                                     48.6
## 3 Saint Pierre and Miquelon                                                 48.5
## 4 Germany                                                                   47.8
## 5 Italy                                                                     46.5
## 6 Andorra                                                                   46.2
```

```r
migration_data <- read.fwf("rawdata_347.txt", widths = c(8,64,6,18), skip=2)
names(migration_data) <- c("Rank", "Country", "Migration_rate", "Date")
migration_data <- migration_data[, c("Country", "Migration_rate")]
head(migration_data)
```

```
##                                                             Country
## 1 Syria
## 2 British Virgin Islands
## 3 Luxembourg
## 4 Cayman Islands
## 5 Singapore
## 6 Anguilla
##   Migration_rate
## 1           27.1
## 2           15.5
## 3           13.3
## 4           13.0
## 5           11.8
## 6           11.1
```

```r
unemployment_data <- read.csv("rawdata_373.csv")
names(unemployment_data) <- c("Country", "Youth_unemployment_rate")
head(unemployment_data)
```

```
##                                                               Country
## 1 French Polynesia
## 2 Kosovo
## 3 South Africa
## 4 Libya
## 5 Eswatini
## 6 Saint Lucia
##   Youth_unemployment_rate
## 1                    56.7
## 2                    55.4
## 3                    53.4
## 4                    48.7
## 5                    47.1
## 6                    46.2
```

b

Merge the data sets containing raw data using dplyr function on the unique keys. Keep the union of all observations in the tables. What key are you using for merging? Return the dimension of the merged data set. We use the column “country” for merging. We encountered an issue where the column for migration rate was always NaN so we had to trim the whitespaces.

```r
median_age_data$Country <- trimws(median_age_data$Country)
migration_data$Country <- trimws(migration_data$Country)
unemployment_data$Country <- trimws(unemployment_data$Country)

joined <- left_join(median_age_data, unemployment_data, by = join_by(Country))
joined_data <- left_join(joined, migration_data, by = join_by(Country == Country))
head(joined_data)
```

```
##                      Country Median_Age Youth_unemployment_rate Migration_rate
## 1                     Monaco       55.4                    26.6            8.3
## 2                      Japan       48.6                     3.6            0.0
## 3 Saint Pierre and Miquelon       48.5                      NA           -7.7
## 4                    Germany       47.8                     6.2            1.5
## 5                      Italy       46.5                    32.2            3.2
## 6                    Andorra       46.2                      NA            0.0
```

```r
dim(joined_data)
```

```
## [1] 227   4
```

c

You will acquire more country level information such as the classification of the country based on income. Such an information can be found at https://datahelpdesk.worldbank.org/knowledgebase/articles/906519. From there extract the classification for 2020 into low/lower-middle/upper-middle/high income countries

```r
income_data <- read_excel("OGHIST.xlsx", sheet="Country Analytical History", col_names = TRUE)
```

```
## New names:
## * ‘‘ -> ‘...1‘
## * `` -> `...3`
## * `` -> `...4`
## * `` -> `...5`
## * `` -> `...6`
## * `` -> `...7`
## * `` -> `...8`
## * `` -> `...9`
## * `` -> `...10`
## * `` -> `...11`
## * `` -> `...12`
## * `` -> `...13`
## * `` -> `...14`
## * `` -> `...15`
## * `` -> `...16`
## * `` -> `...17`
## * `` -> `...18`
## * `` -> `...19`
## * `` -> `...20`
## * `` -> `...21`
## * `` -> `...22`
## * `` -> `...23`
## * `` -> `...24`
## * `` -> `...25`
## * `` -> `...26`
## * `` -> `...27`
## * `` -> `...28`
## * `` -> `...29`
## * `` -> `...30`
## * `` -> `...31`
## * `` -> `...32`
## * `` -> `...33`
## * `` -> `...34`
## * `` -> `...35`
## * `` -> `...36`
## * `` -> `...37`
## * `` -> `...38`
```

```r
income_data <- income_data[, c("World Bank Analytical Classifications", "...36")]
income_data <- income_data[11:nrow(income_data), ]
names(income_data) <- c("Country", "Income_class")
head(income_data)
```

```
## # A tibble: 6 x 2
##   Country        Income_class
##   <chr>          <chr>
## 1 Afghanistan    L
## 2 Albania        UM
## 3 Algeria        LM
## 4 American Samoa UM
## 5 Andorra        H
## 6 Angola         LM
```

d

Merge this information to the data set in b. 1. What are the common variables? Can you merge using them? Why or why not?

We can merge the dataframes on the column Country.

```
data <- left_join(joined_data, income_data, by = join_by(Country == Country))
head(data)
```

```
##                      Country Median_Age Youth_unemployment_rate Migration_rate
## 1                     Monaco       55.4                    26.6            8.3
## 2                      Japan       48.6                     3.6            0.0
## 3 Saint Pierre and Miquelon       48.5                      NA           -7.7
## 4                    Germany       47.8                     6.2            1.5
## 5                      Italy       46.5                    32.2            3.2
## 6                    Andorra       46.2                      NA            0.0
##   Income_class
## 1            H
## 2            H
## 3         <NA>
## 4            H
## 5            H
## 6            H
```

2. A reliable merging for countries are ISO codes as they are standardized across data sources. Download the mapping of ISO codes to countries from https://www.cia.gov/the-world-factbook/references/country data-codes/ and load it into R.

```
country_code <- fread("Country Data Codes.csv", sep = ",", header = TRUE, fill=TRUE, stringsAsFactors =
names(country_code)[names(country_code) == 'ISO 3166'] <- 'ISO'
names(country_code)[names(country_code) == 'Name'] <- 'Country'

country_code <- country_code %>%
  mutate(
    Country = ifelse(
      str_detect(GENC, "[a-z]"),
      paste(Country, GENC, sep = ", "),
      Country
    ),
    GENC = ifelse(
      str_detect(GENC, "[a-z]"),
      NA,
      GENC
    )
  )

country_code <- country_code[, c("Country", "ISO")]

country_code <- country_code %>%
  mutate(across(everything(), ~ str_replace_all(., '"', '')))

head(country_code)
```

```
##           Country         ISO
## 1:    Afghanistan AF|AFG|004
## 2:       Akrotiri           -
## 3:        Albania AL|ALB|008
## 4:        Algeria DZ|DZA|012
## 5: American Samoa AS|ASM|016
## 6:        Andorra AD|AND|020
```

3. Merge the data sets using the ISO codes.

```
country_code$Country <- trimws(country_code$Country)
country_code$ISO <- trimws(country_code$ISO)

data <- left_join(data, country_code, by = join_by(Country == Country))
names(data)[names(data) == 'ISO.y'] <- 'ISO'
data$ISO.x <- NULL
head(data)
```

```
##                      Country Median_Age Youth_unemployment_rate Migration_rate
## 1                     Monaco       55.4                    26.6            8.3
## 2                      Japan       48.6                     3.6            0.0
## 3 Saint Pierre and Miquelon       48.5                      NA           -7.7
## 4                    Germany       47.8                     6.2            1.5
## 5                      Italy       46.5                    32.2            3.2
## 6                    Andorra       46.2                      NA            0.0
##   Income_class        ISO
## 1            H MC|MCO|492
## 2            H JP|JPN|392
## 3         <NA> PM|SPM|666
## 4            H DE|DEU|276
## 5            H IT|ITA|380
## 6            H AD|AND|020
```

e

Introduce into the data set information on continent for each country and subcontinent (region). You should find a way to gather this data. You can find an appropriate online resource, download the data and merge the information with the existing data set. Name the merged data set df_vars.

We found this dataset on kaggle: https://www.kaggle.com/datasets/andradaolteanu/country-mapping-iso-continent-region?resource=download However, despite ISO being standardized, we could not find a dataset where the ISO layout was the same as the one provided from the previous link. For making the merging work, we had to merge 3 columns to create a similar column to ISO from the previous dataframe.

```
continents <- read.csv("continents2.csv")

continents <- continents %>%
  mutate(
    country.code = str_pad(country.code, width = 3, pad = "0"),

    ISO = paste(alpha.2, alpha.3, country.code, sep = "|")
    )
```

```
names(continents)[names(continents) == 'name'] <- 'Country'
continents <- continents[, c("Country", "region", "sub.region", "ISO")]
head(continents)
```

```
##          Country  region      sub.region         ISO
## 1    Afghanistan    Asia   Southern Asia AF|AFG|004
## 2  Åland Islands  Europe Northern Europe AX|ALA|248
## 3        Albania  Europe Southern Europe AL|ALB|008
## 4        Algeria  Africa Northern Africa DZ|DZA|012
## 5 American Samoa Oceania       Polynesia AS|ASM|016
## 6        Andorra  Europe Southern Europe AD|AND|020
```

```
df_vars <- left_join(data, continents, by = join_by(Country == Country))
names(df_vars)[names(df_vars) == 'ISO.x'] <- 'ISO'
df_vars$ISO.y <- NULL
dim(df_vars)
```

```
## [1] 227   8
```

```
head(df_vars)
```

```
##                     Country Median_Age Youth_unemployment_rate Migration_rate
## 1                    Monaco       55.4                    26.6            8.3
## 2                     Japan       48.6                     3.6            0.0
## 3 Saint Pierre and Miquelon       48.5                      NA           -7.7
## 4                   Germany       47.8                     6.2            1.5
## 5                     Italy       46.5                    32.2            3.2
## 6                   Andorra       46.2                      NA            0.0
##   Income_class        ISO   region       sub.region
## 1            H MC|MCO|492   Europe   Western Europe
## 2            H JP|JPN|392     Asia     Eastern Asia
## 3         <NA> PM|SPM|666 Americas Northern America
## 4            H DE|DEU|276   Europe   Western Europe
## 5            H IT|ITA|380   Europe  Southern Europe
## 6            H AD|AND|020   Europe  Southern Europe
```

f

Discuss on the tidyness of the data set df_vars. What are the observational units, what are the variables? What can be considered fixed vs measured variables? Tidy the data if needed.

The obtained data set df_vars is almost tidy, while it almost fully satisfies the requirements to the data to be considered tidy: our data set is the collection of quantitative and qualitative values and they are organized in a way, that each value belongs to an observation and a variable. Our data is organized in a way that each row represents an observational unit (a country) and each column is a variable. All our variables have the same unit and measure the same attribute.

In our case the fixed variables are Country (char), region (char), sub.region (char), ISO (char), Income_class (char). The left variables Median_Age (double), Migration_rate (double) and Youth_unemployment_rate (double) are measured variables in our case. Typically, the fixed variables are put in the beginning of the data set, while measured are put after them. So we will perform this small change to have our data perfectly tidy.

Also, since in the following tasks the region will be pointed to as "continent", we decided to rename the name of this variable.

```
df_vars <- df_vars %>%
  select(ISO, Country, region, sub.region, Income_class, Median_Age, Youth_unemployment_rate, Migration

names(df_vars)[names(df_vars) == "region"] <- "continent"

head(df_vars)
```

```
##          ISO                  Country continent       sub.region Income_class
## 1 MC|MCO|492                   Monaco    Europe   Western Europe            H
## 2 JP|JPN|392                    Japan      Asia     Eastern Asia            H
## 3 PM|SPM|666 Saint Pierre and Miquelon  Americas Northern America         <NA>
## 4 DE|DEU|276                  Germany    Europe   Western Europe            H
## 5 IT|ITA|380                    Italy    Europe  Southern Europe            H
## 6 AD|AND|020                  Andorra    Europe  Southern Europe            H
##   Median_Age Youth_unemployment_rate Migration_rate
## 1       55.4                    26.6            8.3
## 2       48.6                     3.6            0.0
## 3       48.5                      NA           -7.7
## 4       47.8                     6.2            1.5
## 5       46.5                    32.2            3.2
## 6       46.2                      NA            0.0
```

g

Make a frequency table for the status variable in the merged data set. Briefly comment on the results.

```
income_status_frequency <- df_vars %>%
  count(Income_class, name = "frequency") %>%
  mutate(
    percentage = 100 * (frequency / sum(frequency))
  )

print(income_status_frequency)
```

```
##   Income_class frequency percentage
## 1            H        66   29.07489
## 2            L        23   10.13216
## 3           LM        45   19.82379
## 4           UM        50   22.02643
## 5         <NA>        43   18.94273
```

From the generated table we can conclude that the high income countries are represented at most, while low income countries at least. The representation of lower-middle and upper-middle income countries are on the same level. Also, a lot of observations (about 19%) are missing the value for the variable Income_class in our data set.

h

What is the distribution of income status in the different continents? Compute the absolute frequencies as well as the relative frequency of status within each continent. Briefly comment on the results.

```r
income_distribution_within_continent <- df_vars %>%
  group_by(continent, Income_class) %>%
  summarise(frequency = n(), .groups = 'drop')

income_distribution_within_continent <- income_distribution_within_continent %>%
  group_by(continent) %>%
  mutate(relative_frequency = frequency / sum(frequency))

print(income_distribution_within_continent)
```

```
## # A tibble: 26 x 4
## # Groups:   continent [6]
##    continent Income_class frequency relative_frequency
##    <chr>     <chr>            <int>              <dbl>
##  1 Africa    H                    1             0.0204
##  2 Africa    L                   20             0.408
##  3 Africa    LM                  19             0.388
##  4 Africa    UM                   7             0.143
##  5 Africa    <NA>                 2             0.0408
##  6 Americas  H                   13             0.302
##  7 Americas  LM                   6             0.140
##  8 Americas  UM                  17             0.395
##  9 Americas  <NA>                 7             0.163
## 10 Asia      H                   10             0.222
## # i 16 more rows
```

We can conclude that the low and lower-income classes are prevailing in Africa, while the biggest group in Europe and Oceania is high income class. In Asia the lower-middle income class is the biggest one followed by the upper-middle income class. In Americas the prevailing income class is upper-middle and in our data set there is no low income class countries represented. Also, it should be noted that each continent has some countries with no income class specification available (Africa ≈ 4%, Americas ≈ 16%, Asia ≈ 18%, Europe ≈ 13% and Oceania 10%). In addition, some countries have values for variable Income_class, while no value for variable continent.

i

From h. identify the countries which are the only ones in their respective group. Explain in few words the output.

```r
unique_countries_in__their_group <- df_vars %>%
  group_by(continent, Income_class) %>%
  filter(n() == 1) %>%
  ungroup()

print(unique_countries_in__their_group)
```

```
## # A tibble: 3 x 8
##   ISO        Country     continent sub.region         Income_class Median_Age
##   <chr>      <chr>       <chr>     <chr>              <chr>             <dbl>
## 1 UA|UKR|804 Ukraine     Europe    Eastern Europe     LM                 41.2
## 2 SC|SYC|690 Seychelles  Africa    Sub-Saharan Africa H                  36.8
## 3 AF|AFG|004 Afghanistan Asia      Southern Asia      L                  19.5
## # i 2 more variables: Youth_unemployment_rate <dbl>, Migration_rate <dbl>
```

We can see that Ukraine is the only lower-middle income country in Europe, while the only high income country in Africa is Seychelles and the only low income country in Asia is Afghanistan.

j

For each continent count the number of sub-regions in the data set. How granular are the subcontinents that you employ in the analysis?

```
sub_region_counts_for_continents <- df_vars %>%
  group_by(continent) %>%
  summarise(number_of_sub_regions = n_distinct(sub.region), .groups = 'drop')

print(sub_region_counts_for_continents)
```

```
## # A tibble: 6 x 2
##   continent number_of_sub_regions
##   <chr>                     <int>
## 1 Africa                        2
## 2 Americas                      2
## 3 Asia                          5
## 4 Europe                        4
## 5 Oceania                       4
## 6 <NA>                          1
```

It is seen that the Asia has the highest granularity, followed by Europe and Oceania, while Africa and Americas are represented only by two sub-regions in our data set.

k

Look at the frequency distribution of income status in the subregions of North- and South-Americas. Comment on the results.

```
income_distribution_americas <- df_vars %>%
  filter(continent %in% "Americas") %>%
  group_by(sub.region, Income_class) %>%
  summarise(frequency = n(), .groups = 'drop') %>%
  group_by(sub.region) %>%
  mutate(relative_frequency = frequency / sum(frequency))

print(income_distribution_americas)
```

```
## # A tibble: 6 x 4
## # Groups:   sub.region [2]
##   sub.region                      Income_class frequency relative_frequency
##   <chr>                           <chr>            <int>              <dbl>
## 1 Latin America and the Caribbean H                    9              0.237
## 2 Latin America and the Caribbean LM                   6              0.158
## 3 Latin America and the Caribbean UM                  17              0.447
## 4 Latin America and the Caribbean <NA>                 6              0.158
## 5 Northern America                H                    4              0.8
## 6 Northern America                <NA>                 1              0.2
```

We obtained that in the Northern America all countries are high income and also we have one country with no specified income status. In the Southern America prevailing number of countries are upper-middle class income countries, around 24% of countries are high income.

l.

Dig deeper into the low-middle income countries of the Americas. Which ones are they? Are they primarily small island states in the Caribbean? Comment.

```
low_middle_income_americas <- df_vars %>%
  filter(continent %in% "Americas" &
         Income_class == "LM")

print(low_middle_income_americas)
```

```
##           ISO     Country continent                      sub.region Income_class
## 1 SV|SLV|222 El Salvador  Americas Latin America and the Caribbean           LM
## 2 NI|NIC|558   Nicaragua  Americas Latin America and the Caribbean           LM
## 3 BO|BOL|068     Bolivia  Americas Latin America and the Caribbean           LM
## 4 HN|HND|340    Honduras  Americas Latin America and the Caribbean           LM
## 5 HT|HTI|332       Haiti  Americas Latin America and the Caribbean           LM
## 6 BZ|BLZ|084      Belize  Americas Latin America and the Caribbean           LM
##   Median_Age Youth_unemployment_rate Migration_rate
## 1       27.7                     9.6           -4.8
## 2       27.3                     8.5           -2.4
## 3       25.3                     6.9           -0.3
## 4       24.4                    10.7           -1.4
## 5       24.1                      NA           -1.9
## 6       23.9                    15.3           -1.0
```

No, these countries are not island states (except Haiti), however they all except Bolivia have the coastal line. Also, they all are quite different in terms of the size.

m

Create a table of average values for median age, youth unemployment rate and net migration rate separated into income status. Make sure that in the output, the ordering of the income classes is proper (i.e., L, LM, UM, H or the other way around). Briefly comment the results.

We will calculate it by ignoring the NA values. Also, we will exclude from the final data frame the values for the observations with no income class specified

```
table_of_average_values <- df_vars %>%
  filter(!is.na(Income_class)) %>%
  mutate(Income_class = factor(Income_class, levels = c("L", "LM", "UM", "H"))) %>%
  group_by(Income_class) %>%
  summarise(
    Avg_Median_Age = mean(Median_Age, na.rm = TRUE),
    Avg_Youth_Unemployment_Rate = mean(Youth_unemployment_rate, na.rm = TRUE),
    Avg_Migration_Rate = mean(Migration_rate, na.rm = TRUE),
    .groups = 'drop'
  )

print(table_of_average_values)
```

```
## # A tibble: 4 x 4
##   Income_class Avg_Median_Age Avg_Youth_Unemployment_Rate Avg_Migration_Rate
##   <fct>                 <dbl>                       <dbl>              <dbl>
## 1 L                      18.4                        13.2              -1.74
## 2 LM                     24.9                        16.4              -1.50
## 3 UM                     31.5                        21.8              -4.48
## 4 H                      39.3                        16.7               2.18
```

We can trace the growth of the average median age of the population with the drop of the status of country‘s income. In addition, we observe that the highest average unemployment youth rate is experienced by the upper-middle status income countries. The high and lower-middle income countries have almost the same average unemployment youth rate and the indicator for low income status countries is the smallest. Average migration rate in countries of all statuses except high income status is negative, being the most negative in countries with upper-middle income status.

n

Look also at the standard deviation instead of the mean in m. Do you gain additional insights? Briefly comment the results.

```
table_of_sd_values <- df_vars %>%
  filter(!is.na(Income_class)) %>%
  mutate(Income_class = factor(Income_class, levels = c("L", "LM", "UM", "H"))) %>%
  group_by(Income_class) %>%
  summarise(
    Sd_Median_Age = sd(Median_Age, na.rm = TRUE),
    Sd_Youth_Unemployment_Rate = sd(Youth_unemployment_rate, na.rm = TRUE),
    Sd_Migration_Rate = sd(Migration_rate, na.rm = TRUE),
    .groups = 'drop'
  )

print(table_of_sd_values)
```

```
## # A tibble: 4 x 4
##   Income_class Sd_Median_Age Sd_Youth_Unemployment_Rate Sd_Migration_Rate
##   <fct>                  <dbl>                       <dbl>              <dbl>
## 1 L                       1.74                        12.4                2.60
## 2 LM                      5.08                        11.3                2.21
## 3 UM                      6.34                        13.3               13.2
## 4 H                       5.66                        10.8                5.99
```

The standard deviation results do provide some additional insights. For the median age, we can see that the sd are relatively low across all income classes, meaning less variability, therefore similar age distributions. The sd of youth unemployment rate is higher compared to the median age. This indicates a bigger variability in the rates, suggesting that employment conditions vary more among countries. For the migration rate, the sd is lower for L, LM and H, and is the highest in UM class. This might show how the countries with UM income have more diverse migration patters.

O

Repeat the analysis in m. for each income status and continent combination. Discuss the results.

```
table_of_average_values_continent <- df_vars %>%
  filter(!is.na(Income_class)) %>%
  mutate(Income_class = factor(Income_class, levels = c("L", "LM", "UM", "H"))) %>%
  group_by(Income_class, continent) %>%
  summarise(
    Avg_Median_Age = mean(Median_Age, na.rm = TRUE),
    Avg_Youth_Unemployment_Rate = mean(Youth_unemployment_rate, na.rm = TRUE),
    Avg_Migration_Rate = mean(Migration_rate, na.rm = TRUE),
    .groups = 'drop'
  )

print(table_of_average_values_continent)
```

```
## # A tibble: 20 x 5
##    Income_class continent Avg_Median_Age Avg_Youth_Unemployment_Rate
##    <fct>        <chr>              <dbl>                       <dbl>
##  1 L            Africa              18.2                        12.9
##  2 L            Asia                19.5                        17.6
##  3 L            <NA>                19.9                        13.1
##  4 LM           Africa              22.1                        20.8
##  5 LM           Americas            25.4                        10.2
##  6 LM           Asia                27.5                        13.1
##  7 LM           Europe              41.2                        17.9
##  8 LM           Oceania             24.4                        13.5
##  9 UM           Africa              25.6                        39.3
## 10 UM           Americas            30.9                        15.3
## 11 UM           Asia                31.9                        19.4
## 12 UM           Europe              40.3                        19.7
## 13 UM           Oceania             26.3                        15.7
## 14 UM           <NA>                36.9                        44.6
## 15 H            Africa              36.8                        11.6
## 16 H            Americas            38.3                        19.1
## 17 H            Asia                34.4                        11.1
## 18 H            Europe              43.2                        15.7
## 19 H            Oceania             33                          25.7
## 20 H            <NA>                35                          25.8
## # i 1 more variable: Avg_Migration_Rate <dbl>
```

From the results we can observe the socioeconomic differences across the continents. We see that lower income countries have lower median ages, while higher income ones have older populations. Also, unemployment rates vary widely across regions, some having significantly higher rates than others. For L class, we have similar rates; for LM and UM, Africa has the highest rate; for H, Oceania has the highest rate. Furthermore, migration patters also differ. We have regions experimenting negative migration rates, while others having positive ones. L class has only negative rates; for LM, Europe has positive rate;for UM Africa has the positive one, while Oceania followed by Asia have the lowest rates; for H, we see only Oceania with negative rate. Lastly, we have for each L, UM, H classes, an observation with NA values for the continent variable. This is is important to note as it might affect the accuracy of the results.

p

Identify countries which are doing well in terms of both youth unemployment and net migration rate (in the top 25% of their respective continent in terms of net migration rate and in the bottom 25% of their respective continent in terms of youth unemployment).

```r
top_countries <- df_vars %>%
  group_by(continent) %>%
  mutate(
    top_migration = quantile(Migration_rate, 0.75, na.rm = TRUE),
    bottom_unempoyment = quantile(Youth_unemployment_rate, 0.25, na.rm = TRUE)
  ) %>%
  filter(Migration_rate >= top_migration & Youth_unemployment_rate <= bottom_unempoyment) %>%
  select(Country, continent, Migration_rate, Youth_unemployment_rate)


top_countries
```

```
## # A tibble: 17 x 4
## # Groups:   continent [6]
##    Country              continent Migration_rate Youth_unemployment_rate
##    <chr>                <chr>              <dbl>                   <dbl>
##  1 Czechia              <NA>                 2.3                     6.7
##  2 Switzerland          Europe               4.6                     7.9
##  3 Malta                Europe               6.6                     9.1
##  4 Macau                <NA>                 3.3                     5.3
##  5 Norway               Europe               4                       9.7
##  6 United States        Americas             3                       8.6
##  7 United Arab Emirates Asia                 7.6                     6.9
##  8 Palau                Oceania              0.9                     5.6
##  9 Qatar                Asia                 6.5                     0.4
## 10 Bahrain              Asia                10.6                     5.3
## 11 Kazakhstan           Asia                 0.4                     3.8
## 12 Israel               Asia                 2.1                     7.2
## 13 Papua New Guinea     Oceania              0                       3.6
## 14 Madagascar           Africa               0                       1
## 15 Togo                 Africa               0                       3.9
## 16 Guinea               Africa               0                       1
## 17 Benin                Africa               0.3                     5.6
```

The results show diverse regions. Countries like Qatar, Switzerland, United Arab Emirates etc. present favorable results for both rates. Meanwhile, Africa countries, which have lower unemployment rates, have also lower migration rates.

r

Export the final data set to a csv with ";" separator and "." as a symbol for missing values; no rownames should be included in the csv. Upload the .csv to TUWEL together with your .Rmd and .html (or .pdf).

```
write.table(df_vars, file = "final_data.csv", sep = ";", na = ".", row.names = FALSE, col.names = TRUE)
```