Case Study 2

AKSTA Statistical Computing

Fani Sentinella-Jerbić

04.04.2023

1.

Obtaining country information:

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.2.2
```

```
##
## 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
```

```
countries <- read.csv("country-codes_csv.csv") %>%
  select(official_name_en,
         ISO3166.1.Alpha.3,
         ISO3166.1.Alpha.2,
         Developed...Developing.Countries,
         Region.Name,
         Sub.region.Name)
head(countries)
```

```
##   official_name_en ISO3166.1.Alpha.3 ISO3166.1.Alpha.2
## 1                                TWN                TW
## 2      Afghanistan               AFG                AF
## 3          Albania               ALB                AL
## 4          Algeria               DZA                DZ
## 5   American Samoa               ASM                AS
## 6          Andorra               AND                AD
##   Developed...Developing.Countries Region.Name Sub.region.Name
## 1
## 2                       Developing        Asia   Southern Asia
## 3                        Developed      Europe Southern Europe
## 4                       Developing      Africa Northern Africa
## 5                       Developing     Oceania       Polynesia
## 6                        Developed      Europe Southern Europe
```

2.

Loading the csv file:

```
yur <- read.csv("rawdata_373.csv") %>%
        rename(country=country_name)

head(yur)
```

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

Loading the txt file:

```
age <- read.fwf(
  file="rawdata_343.txt",
  skip=2,
  widths=c(8, 66, 4))

age <- age %>%
  select(V2, V3) %>%
  rename(country=V2, median_age=V3)

head(age)
```

```
##                                                               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
```

From both files I removed the trailing spaces which would otherwise cause problems in merging.

```
trim <- function(x) sub("\\s+$", "", x)
yur$country <- trim(yur$country)
age$country <- trim(age$country)
```

3.

Joining the datasets with full join on key country to keep all observations:

```
joined <- full_join(yur, age, by = "country")
head(joined)
```

```
##            country youth_unempl_rate median_age
## 1 French Polynesia              56.7       33.3
## 2           Kosovo              55.4       30.5
## 3     South Africa              53.4       28.0
## 4            Libya              48.7       25.8
## 5         Eswatini              47.1       23.7
## 6      Saint Lucia              46.2       36.9
```

4.

For the sake of inspecting the problem of using country names as the key, I'm performing a full join:

```
df_vars <- joined %>% full_join(countries,by=c('country'='official_name_en'))
df_vars %>%
  arrange(country)%>%
  select(country) %>%
  head(20)
```

```
##                country
## 1
## 2          Afghanistan
## 3        Åland Islands
## 4              Albania
## 5              Algeria
## 6       American Samoa
## 7              Andorra
## 8               Angola
## 9             Anguilla
## 10          Antarctica
## 11 Antigua and Barbuda
## 12           Argentina
## 13             Armenia
## 14               Aruba
## 15           Australia
## 16             Austria
## 17          Azerbaijan
## 18             Bahamas
## 19        Bahamas, The
## 20             Bahrain
```

We can see some countries didn't get matched. For example one dataframe contained "Bahamas" and the other contained "Bahamas, the". They should be considered one observation but can't be based on these country names. This is why country codes should be used.

```
library("readxl")
match <- read_excel("CIA_factbook_matching_table_iso.xlsx")
head(match)
```

```
## # A tibble: 6 x 3
##   Country        `ISO 3166 2` `ISO 3166 3`
##   <chr>          <chr>        <chr>
## 1 Afghanistan    AF           AFG
## 2 Albania        AL           ALB
## 3 Algeria        DZ           DZA
## 4 American Samoa AS           ASM
## 5 Andorra        AD           AND
## 6 Angola         AO           AGO
```

```
df_vars <- joined %>%
  left_join(match,by=c('country'='Country'))

df_vars <- df_vars %>%
    left_join(countries, by=c('ISO 3166 3'='ISO3166.1.Alpha.3')) %>%
    select(!c('ISO 3166 3', 'ISO 3166 2', 'ISO3166.1.Alpha.2'))

head(df_vars)
```

```
##            country youth_unempl_rate median_age official_name_en
## 1 French Polynesia              56.7       33.3 French Polynesia
## 2           Kosovo              55.4       30.5             <NA>
## 3     South Africa              53.4       28.0     South Africa
## 4            Libya              48.7       25.8            Libya
## 5         Eswatini              47.1       23.7             <NA>
## 6      Saint Lucia              46.2       36.9      Saint Lucia
##   Developed...Developing.Countries Region.Name                 Sub.region.Name
## 1                       Developing     Oceania                       Polynesia
## 2                             <NA>        <NA>                            <NA>
## 3                       Developing      Africa              Sub-Saharan Africa
## 4                       Developing      Africa                 Northern Africa
## 5                             <NA>        <NA>                            <NA>
## 6                       Developing    Americas Latin America and the Caribbean
```

5.

Most special cases are countries which couldn't be matched even with the provided codes.

```
df_vars[is.na(df_vars$official_name_en), ] %>%
  arrange(country) %>%
  select(country)
```

```
##                                           country
## 1                                     Cabo Verde
## 2                                        Curacao
## 3                                        Czechia
## 4                                       Eswatini
## 5                                       Guernsey
## 6                                    Isle of Man
## 7                                         Jersey
## 8                                         Kosovo
## 9                                     Montenegro
## 10                              Saint Barthelemy
## 11 Saint Helena, Ascension, and Tristan da Cunha
## 12                                  Saint Martin
## 13                                        Serbia
## 14                                  Sint Maarten
## 15                                   South Sudan
## 16                                   Timor-Leste
```

I split these in 3 categories:

1. Don't have a valid country code in the provided file:

- Isle of Man
- Guernsey
- Jersey
- Saint Barthelemy

2. Don't have an entry in the provided file:

- Curacao
- Eswatini
- Kosovo
- Montenegro
- Serbia
- Saint Helena, Ascension, and Tristan da Cunha
- Saint Martin
- Sint Maarten
- South Sudan

3. Have wrong country names:

- Cabo Verde
- Czechia
- Timor-Leste

Another special case is Taiwan which does achieve a match but has empty strings instead of useful data:

```
df_vars %>%
  filter(country=="Taiwan")
```

```
##   country youth_unempl_rate median_age official_name_en
## 1  Taiwan                NA       42.3
##   Developed...Developing.Countries Region.Name Sub.region.Name
## 1
```

From these, the third category can be easily fixed, whereas other would require finding data from other sources or something similar. Because of this, I think it would be better to drop them. Especially in the controversial case of Kosovo, Serbia and Montenegro. For Taiwan, I am replacing empty strings with NA to mark it as missing values.

Fixing the special cases:

```
joined["country"][joined["country"] == "Czechia"] <- "Czech Republic"
joined["country"][joined["country"] == "Cabo Verde"] <- "Cape Verde"
joined["country"][joined["country"] == "Timor-Leste"] <- "East Timor"
```

```
df_vars <- joined %>%
    left_join(match,by=c('country'='Country')) %>%
    left_join(countries, by=c('ISO 3166 3'='ISO3166.1.Alpha.3')) %>%
    select(!c('ISO3166.1.Alpha.2','official_name_en')) %>%
    filter(!(country %in% c('Curacao','Eswatini','Kosovo','Montenegro','Serbia',
                      'Saint Helena, Ascension, and Tristan da Cunha',
                      'Saint Martin','Sint Maarten','South Sudan',
                      'Isle of Man', 'Guernsey', 'Jersey', 'Saint Barthelemy'))) %>%
    na_if('')
```

```
df_vars[!complete.cases(df_vars), ] %>%
  arrange(country) %>%
  head()
```

```
##                    country youth_unempl_rate median_age ISO 3166 2 ISO 3166 3
## 1           American Samoa                NA       27.2         AS        ASM
## 2                  Andorra                NA       46.2         AD        AND
## 3                 Anguilla                NA       35.7         AI        AIA
## 4      Antigua and Barbuda                NA       32.7         AG        ATG
## 5                    Aruba                NA       39.9         AW        ABW
## 6   British Virgin Islands                NA       37.2         VG        VGB
##   Developed...Developing.Countries Region.Name                 Sub.region.Name
## 1                       Developing     Oceania                       Polynesia
## 2                        Developed      Europe                 Southern Europe
## 3                       Developing    Americas Latin America and the Caribbean
## 4                       Developing    Americas Latin America and the Caribbean
## 5                       Developing    Americas Latin America and the Caribbean
## 6                       Developing    Americas Latin America and the Caribbean
```

Now the only missing values left (other than Taiwan) are in youth_unempl_rate, which we can leave and address accordingly later if needed.

Lastly, for the sake of simplicity I'm changing the variable names to something simpler:

```
df_vars <- df_vars %>%
  rename(dev=Developed...Developing.Countries,
         region=Region.Name,
         subregion=Sub.region.Name)
```

6.

```
df_vars %>% count(dev)
```

```
##          dev   n
## 1  Developed  53
## 2 Developing 160
## 3       <NA>   1
```

7.

```
df_vars %>%
  count(region)
```

```
##     region  n
## 1   Africa 52
## 2 Americas 46
## 3     Asia 51
## 4   Europe 43
## 5  Oceania 21
## 6     <NA>  1
```

8.

```
df_vars %>%
  group_by(region) %>%
  count(dev)
```

```
## # A tibble: 9 x 3
## # Groups:   region [6]
##   region   dev            n
##   <chr>    <chr>      <int>
## 1 Africa   Developing    52
## 2 Americas Developed      5
## 3 Americas Developing    41
## 4 Asia     Developed      3
## 5 Asia     Developing    48
## 6 Europe   Developed     43
## 7 Oceania  Developed      2
## 8 Oceania  Developing    19
## 9 <NA>     <NA>           1
```

9.

```
df_vars %>%
  filter(!(country=='Taiwan')) %>%
  group_by(dev) %>%
  summarise(avgMedAge=mean(median_age),
            stdMedAge=sd(median_age),
            avgYUR=mean(youth_unempl_rate, na.rm=TRUE),
            stdYUR=sd(youth_unempl_rate, na.rm=TRUE))
```

```
## # A tibble: 2 x 5
##   dev        avgMedAge stdMedAge avgYUR stdYUR
##   <chr>          <dbl>     <dbl>  <dbl>  <dbl>
## 1 Developed       41.9      4.15   16.2   9.78
## 2 Developing      27.6      7.17   18.0  12.4
```

The output is expected. In developed countries the average median age is larger, and it's standard deviation is lower. I would guess large differences in wealth in developing countries could be the cause for the large standard deviation. The youth unemployment rate is also larger in developed countries, however I would guess in developing countries most of labour is done "under the table", untracked and thus resulting in such stats.

10.

```
df_vars %>%
  filter(!(country=='Taiwan')) %>%
  group_by(region, dev) %>%
  summarise(avgMedAge=mean(median_age),
            stdMedAge=sd(median_age),
            avgYUR=mean(youth_unempl_rate, na.rm=TRUE),
            stdYUR=sd(youth_unempl_rate, na.rm=TRUE))
```

```
## `summarise()` has grouped output by 'region'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 8 x 6
## # Groups:   region [5]
##   region   dev        avgMedAge stdMedAge avgYUR stdYUR
##   <chr>    <chr>          <dbl>     <dbl>  <dbl>  <dbl>
## 1 Africa   Developing      21.1     4.93    18.8 14.2
## 2 Americas Developed       41.3     5.34    16.3 11.3
## 3 Americas Developing      32.7     5.29    16.9  9.28
## 4 Asia     Developed       39.0     9.15    10.3  8.73
## 5 Asia     Developing      30.0     6.39    16.9 11.3
## 6 Europe   Developed       42.4     3.58    16.9 10.0
## 7 Oceania  Developed       37.4     0.212   11.6  0.212
## 8 Oceania  Developing      28.5     4.50    21.3 15.5
```

11.

I create temporary columns for means of groups and then create the new columns based on these.

```
df_vars <- df_vars  %>%
  group_by(region)%>%

  mutate(avg_median_age=mean(median_age),
         avg_yu=mean(youth_unempl_rate, na.rm=TRUE)) %>%

  mutate(above_average_median_age=ifelse(median_age > avg_median_age, "yes", "no"),
         above_average_yu =ifelse(youth_unempl_rate > avg_yu, "yes", "no")) %>%

  ungroup() %>%
  select(-c(avg_median_age,
            avg_yu))

head(df_vars)
```

```
## # A tibble: 6 x 10
##   country    youth~1 media~2 ISO 3~3 ISO 3~4 dev   region subre~5 above~6 above~7
##   <chr>        <dbl>   <dbl> <chr>   <chr>   <chr> <chr>  <chr>   <chr>   <chr>
## 1 French P~     56.7    33.3 PF      PYF     Deve~ Ocean~ Polyne~ yes     yes
## 2 South Af~     53.4    28   ZA      ZAF     Deve~ Africa Sub-Sa~ yes     yes
## 3 Libya         48.7    25.8 LY      LBY     Deve~ Africa Northe~ yes     yes
## 4 Saint Lu~     46.2    36.9 LC      LCA     Deve~ Ameri~ Latin ~ yes     yes
## 5 Macedonia     45.4    39   MK      MKD     Deve~ Europe Southe~ no      yes
## 6 Gaza Str~     42.2    18   PS      PSE     Deve~ Asia   Wester~ no      yes
## # ... with abbreviated variable names 1: youth_unempl_rate, 2: median_age,
## #   3: `ISO 3166 2`, 4: `ISO 3166 3`, 5: subregion,
## #   6: above_average_median_age, 7: above_average_yu
```

12.

```
write.table(df_vars, "final_dataset.csv", sep=';', na='.', row.names=FALSE)
```