DATA 607 Project 2

Samuel I Kigamba

October 06, 2019

Contents

DATA 607 Project 2.

The goal of this assignment is to give you practice in preparing different datasets for downstream analysis work. Your task is to: (1) Choose any three of the "wide" datasets identified in the Week 5 Discussion items. (You may use your own dataset; please don't use my Sample Post dataset, since that was used in your Week 6 assignment!) For each of the three chosen datasets: ??? Create a .CSV file (or optionally, a MySQL database!) that includes all of the information included in the dataset. You're encouraged to use a "wide" structure similar to how the information appears in the discussion item, so that you can practice tidying and transformations as described below. ??? Read the information from your .CSV file into R, and use tidyr and dplyr as needed to tidy and transform your data. [Most of your grade will be based on this step!] ??? Perform the analysis requested in the discussion item. ??? Your code should be in an R Markdown file, posted to rpubs.com, and should include narrative descriptions of your data cleanup work, analysis, and conclusions. (2) Please include in your homework submission, for each of the three chosen datasets: ??? The URL to the .Rmd file in your GitHub repository, and ??? The URL for your rpubs.com web page.

set working directory and Install all the relevant packages and load their respective libraries into R.

Male migrants

Load the following libraries

library(stringr)

library(tidyr)

library(dplyr)

library(tidyverse)

library(tibble)

library(caret)

library(readr)

Upload the data into Github

This will ensure that everyone with access to the github repository can easily audit or retest the data. This ensures ease of accessibility and testing by a wide audience. Follow this link to see uploaded Male migrants .csv file (https://raw.githubusercontent.com/igukusamuel/DATA-607-Project-2/master/UN_MigrantStockMale_2019.csv)

```
male_migrants <- read_csv("https://raw.githubusercontent.com/igukusamuel/DATA-607-Project-2/master/UN_Mi
head(male_migrants)
```

```
## # A tibble: 6 x 530
##   X1    X2    X3    X4    X5      X6    X7    X8    X9    X10   X11   X12
##   <chr> <chr> <chr> <chr> <chr>   <chr> <chr> <chr> <chr> <chr> <chr> <chr>
## 1 <NA>  <NA>  <NA>  <NA>  <NA>    <NA>  <NA>  <NA>  <NA>  <NA>  <NA>  <NA>
## 2 <NA>  <NA>  <NA>  <NA>  <NA>    <NA>  <NA>  <NA>  <NA>  <NA>  <NA>  <NA>
## 3 <NA>  <NA>  <NA>  <NA>  <NA>    <NA>  <NA>  <NA>  <NA>  <NA>  <NA>  <NA>
## 4 <NA>  <NA>  <NA>  <NA>  United~ <NA>  <NA>  <NA>  <NA>  <NA>  <NA>  <NA>
## 5 <NA>  <NA>  <NA>  <NA>  Popula~ <NA>  <NA>  <NA>  <NA>  <NA>  <NA>  <NA>
## 6 <NA>  <NA>  <NA>  <NA>  Depart~ <NA>  <NA>  <NA>  <NA>  <NA>  <NA>  <NA>
## # ... with 518 more variables: X13 <chr>, X14 <chr>, X15 <chr>, X16 <chr>,
## #   X17 <chr>, X18 <chr>, X19 <chr>, X20 <chr>, X21 <chr>, X22 <chr>,
## #   X23 <chr>, X24 <chr>, X25 <chr>, X26 <chr>, X27 <chr>, X28 <chr>,
## #   X29 <chr>, X30 <chr>, X31 <chr>, X32 <chr>, X33 <chr>, X34 <chr>,
## #   X35 <chr>, X36 <chr>, X37 <chr>, X38 <chr>, X39 <chr>, X40 <chr>,
## #   X41 <chr>, X42 <chr>, X43 <chr>, X44 <chr>, X45 <chr>, X46 <chr>,
## #   X47 <chr>, X48 <chr>, X49 <chr>, X50 <chr>, X51 <chr>, X52 <chr>,
## #   X53 <chr>, X54 <chr>, X55 <chr>, X56 <chr>, X57 <chr>, X58 <chr>,
## #   X59 <chr>, X60 <chr>, X61 <chr>, X62 <chr>, X63 <chr>, X64 <chr>,
## #   X65 <chr>, X66 <chr>, X67 <chr>, X68 <chr>, X69 <chr>, X70 <chr>,
## #   X71 <chr>, X72 <chr>, X73 <chr>, X74 <chr>, X75 <chr>, X76 <chr>,
## #   X77 <chr>, X78 <chr>, X79 <chr>, X80 <chr>, X81 <chr>, X82 <chr>,
## #   X83 <chr>, X84 <chr>, X85 <chr>, X86 <chr>, X87 <chr>, X88 <chr>,
## #   X89 <chr>, X90 <chr>, X91 <chr>, X92 <chr>, X93 <chr>, X94 <chr>,
## #   X95 <chr>, X96 <chr>, X97 <chr>, X98 <chr>, X99 <chr>, X100 <chr>,
## #   X101 <chr>, X102 <chr>, X103 <chr>, X104 <chr>, X105 <chr>,
## #   X106 <chr>, X107 <chr>, X108 <chr>, X109 <chr>, X110 <chr>,
## #   X111 <chr>, X112 <chr>, ...
```

```
#view(head(male_migrants, 20)) # vIew data frame structure and see how many rows to skip.
```

Skip first 15 rows

As part of data cleanup, skip the first 15 rows that include source information not relevant to out analysis.

```
male_migrants <- read_csv("https://raw.githubusercontent.com/igukusamuel/DATA-607-Project-2/master/UN_M

head(male_migrants) #Print out first few rows to confirm that the data have been loaded correctly.
```

```
## # A tibble: 6 x 530
##      X1     X2 X3    X4       X5 X6    Total `Other South` `Other North`
##   <dbl>  <dbl> <chr> <chr> <dbl> <chr> <chr> <chr>         <chr>
## 1  1990 1.99e6 WORLD <NA>    900 <NA>  77,6~ 3,412,163     1,159,981
## 2  1990 1.99e6 UN d~ <NA>     NA <NA>  ..    ..            ..
## 3  1990 1.99e6 More~ b       901 <NA>  40,4~ 1,809,849     507,312
## 4  1990 1.99e6 Less~ c       902 <NA>  37,2~ 1,602,314     652,669
## 5  1990 1.99e6 Leas~ d       941 <NA>  5,55~ 244,501       135,262
## 6  1990 1.99e6 Less~ <NA>    934 <NA>  31,6~ 1,357,813     517,407
## # ... with 521 more variables: Afghanistan <chr>, Albania <chr>,
## #   Algeria <chr>, `American Samoa` <chr>, Andorra <chr>, Angola <chr>,
## #   Anguilla <chr>, `Antigua and Barbuda` <chr>, Argentina <chr>,
## #   Armenia <chr>, Aruba <chr>, Australia <chr>, Austria <chr>,
## #   Azerbaijan <chr>, Bahamas <chr>, Bahrain <chr>, Bangladesh <chr>,
## #   Barbados <chr>, Belarus <chr>, Belgium <chr>, Belize <chr>,
## #   Benin <chr>, Bermuda <chr>, Bhutan <chr>, `Bolivia (Plurinational
## #   State of)` <chr>, `Bonaire, Sint Eustatius and Saba` <chr>, `Bosnia
## #   and Herzegovina` <chr>, Botswana <chr>, Brazil <chr>, `British Virgin
## #   Islands` <chr>, `Brunei Darussalam` <chr>, Bulgaria <chr>, `Burkina
## #   Faso` <chr>, Burundi <chr>, `Cabo Verde` <chr>, Cambodia <chr>,
## #   Cameroon <chr>, Canada <chr>, `Cayman Islands` <chr>, `Central African
## #   Republic` <chr>, Chad <chr>, `Channel Islands` <chr>, Chile <chr>,
## #   China <chr>, `China, Hong Kong SAR` <chr>, `China, Macao SAR` <chr>,
## #   Colombia <chr>, Comoros <chr>, Congo <chr>, `Cook Islands` <chr>,
## #   `Costa Rica` <chr>, `CÃ´te d'Ivoire` <chr>, Croatia <chr>, Cuba <chr>,
## #   CuraÃ§ao <chr>, Cyprus <chr>, Czechia <chr>, `Dem. People's Republic of
## #   Korea` <chr>, `Democratic Republic of the Congo` <chr>, Denmark <chr>,
## #   Djibouti <chr>, Dominica <chr>, `Dominican Republic` <chr>,
## #   Ecuador <chr>, Egypt <chr>, `El Salvador` <chr>, `Equatorial
## #   Guinea` <chr>, Eritrea <chr>, Estonia <chr>, Eswatini <chr>,
## #   Ethiopia <chr>, `Falkland Islands (Malvinas)` <chr>, `Faroe
## #   Islands` <chr>, Fiji <chr>, Finland <chr>, France <chr>, `French
## #   Guiana` <chr>, `French Polynesia` <chr>, Gabon <chr>, Gambia <chr>,
## #   Georgia <chr>, Germany <chr>, Ghana <chr>, Gibraltar <chr>,
## #   Greece <chr>, Greenland <chr>, Grenada <chr>, Guadeloupe <chr>,
## #   Guam <chr>, Guatemala <chr>, Guinea <chr>, `Guinea-Bissau` <chr>,
## #   Guyana <chr>, Haiti <chr>, `Holy See` <chr>, Honduras <chr>,
## #   Hungary <chr>, Iceland <chr>, India <chr>, Indonesia <chr>, ...
```

Filter for N/As in column X6

Careful review of the data shows that column named X6 only includes data for rows related to countries and N/A's for rows relating to regions and regional totals. Thus filtering out all N/As in column X6 will leave us with country data only, which is the basis of out analysis. We first view all the N/As under column X6 to confirm none of them relate to country information.

```
colX6 <- filter(male_migrants, is.na(X6))

x <- length(colX6)
x
```

```
## [1] 530
```

```
head(colX6)
```

```
## # A tibble: 6 x 530
##      X1     X2 X3    X4       X5 X6    Total `Other South` `Other North`
##   <dbl>  <dbl> <chr> <chr> <dbl> <chr> <chr> <chr>         <chr>
## 1  1990 1.99e6 WORLD <NA>    900 <NA>  77,6~ 3,412,163     1,159,981
## 2  1990 1.99e6 UN d~ <NA>     NA <NA>  ..    ..            ..
## 3  1990 1.99e6 More~ b       901 <NA>  40,4~ 1,809,849     507,312
## 4  1990 1.99e6 Less~ c       902 <NA>  37,2~ 1,602,314     652,669
## 5  1990 1.99e6 Leas~ d       941 <NA>  5,55~ 244,501       135,262
## 6  1990 1.99e6 Less~ <NA>    934 <NA>  31,6~ 1,357,813     517,407
## # ... with 521 more variables: Afghanistan <chr>, Albania <chr>,
## #   Algeria <chr>, `American Samoa` <chr>, Andorra <chr>, Angola <chr>,
## #   Anguilla <chr>, `Antigua and Barbuda` <chr>, Argentina <chr>,
## #   Armenia <chr>, Aruba <chr>, Australia <chr>, Austria <chr>,
## #   Azerbaijan <chr>, Bahamas <chr>, Bahrain <chr>, Bangladesh <chr>,
## #   Barbados <chr>, Belarus <chr>, Belgium <chr>, Belize <chr>,
## #   Benin <chr>, Bermuda <chr>, Bhutan <chr>, `Bolivia (Plurinational
## #   State of)` <chr>, `Bonaire, Sint Eustatius and Saba` <chr>, `Bosnia
## #   and Herzegovina` <chr>, Botswana <chr>, Brazil <chr>, `British Virgin
## #   Islands` <chr>, `Brunei Darussalam` <chr>, Bulgaria <chr>, `Burkina
## #   Faso` <chr>, Burundi <chr>, `Cabo Verde` <chr>, Cambodia <chr>,
## #   Cameroon <chr>, Canada <chr>, `Cayman Islands` <chr>, `Central African
## #   Republic` <chr>, Chad <chr>, `Channel Islands` <chr>, Chile <chr>,
## #   China <chr>, `China, Hong Kong SAR` <chr>, `China, Macao SAR` <chr>,
## #   Colombia <chr>, Comoros <chr>, Congo <chr>, `Cook Islands` <chr>,
## #   `Costa Rica` <chr>, `CÃ´te d'Ivoire` <chr>, Croatia <chr>, Cuba <chr>,
## #   CuraÃ§ao <chr>, Cyprus <chr>, Czechia <chr>, `Dem. People's Republic of
## #   Korea` <chr>, `Democratic Republic of the Congo` <chr>, Denmark <chr>,
## #   Djibouti <chr>, Dominica <chr>, `Dominican Republic` <chr>,
## #   Ecuador <chr>, Egypt <chr>, `El Salvador` <chr>, `Equatorial
## #   Guinea` <chr>, Eritrea <chr>, Estonia <chr>, Eswatini <chr>,
## #   Ethiopia <chr>, `Falkland Islands (Malvinas)` <chr>, `Faroe
## #   Islands` <chr>, Fiji <chr>, Finland <chr>, France <chr>, `French
## #   Guiana` <chr>, `French Polynesia` <chr>, Gabon <chr>, Gambia <chr>,
## #   Georgia <chr>, Germany <chr>, Ghana <chr>, Gibraltar <chr>,
## #   Greece <chr>, Greenland <chr>, Grenada <chr>, Guadeloupe <chr>,
## #   Guam <chr>, Guatemala <chr>, Guinea <chr>, `Guinea-Bissau` <chr>,
## #   Guyana <chr>, Haiti <chr>, `Holy See` <chr>, Honduras <chr>,
## #   Hungary <chr>, Iceland <chr>, India <chr>, Indonesia <chr>, ...
```

Exclude N/As in column X6

We then exclude all N/A's in column X6 and print out the first 6 rows using the head() function.

```
male_migrants_by_country <- filter(male_migrants, !is.na(X6))

head(male_migrants_by_country)
```

```
## # A tibble: 6 x 530
##       X1      X2 X3    X4        X5 X6    Total `Other South` `Other North`
##    <dbl>   <dbl> <chr> <chr> <dbl> <chr> <chr> <chr>         <chr>
## 1  1990 1.99e6 Buru~ <NA>    108 B R   163,~ 24,837        4,383
## 2  1990 1.99e6 Como~ <NA>    174 B     6,717 432           342
## 3  1990 1.99e6 Djib~ <NA>    262 B R   64,2~ 3,056         1,018
## 4  1990 1.99e6 Erit~ <NA>    232 I     6,228 390           179
## 5  1990 1.99e6 Ethi~ <NA>    231 B R   607,~ 11,603        3,868
## 6  1990 1.99e6 Kenya <NA>    404 B R   161,~ 37,825        18,905
## # ... with 521 more variables: Afghanistan <chr>, Albania <chr>,
## #   Algeria <chr>, `American Samoa` <chr>, Andorra <chr>, Angola <chr>,
## #   Anguilla <chr>, `Antigua and Barbuda` <chr>, Argentina <chr>,
## #   Armenia <chr>, Aruba <chr>, Australia <chr>, Austria <chr>,
## #   Azerbaijan <chr>, Bahamas <chr>, Bahrain <chr>, Bangladesh <chr>,
## #   Barbados <chr>, Belarus <chr>, Belgium <chr>, Belize <chr>,
## #   Benin <chr>, Bermuda <chr>, Bhutan <chr>, `Bolivia (Plurinational
## #   State of)` <chr>, `Bonaire, Sint Eustatius and Saba` <chr>, `Bosnia
## #   and Herzegovina` <chr>, Botswana <chr>, Brazil <chr>, `British Virgin
## #   Islands` <chr>, `Brunei Darussalam` <chr>, Bulgaria <chr>, `Burkina
## #   Faso` <chr>, Burundi <chr>, `Cabo Verde` <chr>, Cambodia <chr>,
## #   Cameroon <chr>, Canada <chr>, `Cayman Islands` <chr>, `Central African
## #   Republic` <chr>, Chad <chr>, `Channel Islands` <chr>, Chile <chr>,
## #   China <chr>, `China, Hong Kong SAR` <chr>, `China, Macao SAR` <chr>,
## #   Colombia <chr>, Comoros <chr>, Congo <chr>, `Cook Islands` <chr>,
## #   `Costa Rica` <chr>, `CÃ´te d'Ivoire` <chr>, Croatia <chr>, Cuba <chr>,
## #   CuraÃ§ao <chr>, Cyprus <chr>, Czechia <chr>, `Dem. People's Republic of
## #   Korea` <chr>, `Democratic Republic of the Congo` <chr>, Denmark <chr>,
## #   Djibouti <chr>, Dominica <chr>, `Dominican Republic` <chr>,
## #   Ecuador <chr>, Egypt <chr>, `El Salvador` <chr>, `Equatorial
## #   Guinea` <chr>, Eritrea <chr>, Estonia <chr>, Eswatini <chr>,
## #   Ethiopia <chr>, `Falkland Islands (Malvinas)` <chr>, `Faroe
## #   Islands` <chr>, Fiji <chr>, Finland <chr>, France <chr>, `French
## #   Guiana` <chr>, `French Polynesia` <chr>, Gabon <chr>, Gambia <chr>,
## #   Georgia <chr>, Germany <chr>, Ghana <chr>, Gibraltar <chr>,
## #   Greece <chr>, Greenland <chr>, Grenada <chr>, Guadeloupe <chr>,
## #   Guam <chr>, Guatemala <chr>, Guinea <chr>, `Guinea-Bissau` <chr>,
## #   Guyana <chr>, Haiti <chr>, `Holy See` <chr>, Honduras <chr>,
## #   Hungary <chr>, Iceland <chr>, India <chr>, Indonesia <chr>, ...
```

Rename column X1 and X3

From the above print out, there is need to rename column X1 and X3 as year and country_to respectively.

```
male_migrants_by_country <- male_migrants_by_country %>%
        rename(
                year = X1,
                country_to = X3
        )
head(male_migrants_by_country)
```

```
## # A tibble: 6 x 530
##    year     X2 country_to X4       X5 X6    Total `Other South`
##   <dbl>  <dbl> <chr>      <chr> <dbl> <chr> <chr> <chr>
## 1  1990 1.99e6 Burundi    <NA>    108 B R   163,~ 24,837
## 2  1990 1.99e6 Comoros    <NA>    174 B     6,717 432
## 3  1990 1.99e6 Djibouti   <NA>    262 B R   64,2~ 3,056
## 4  1990 1.99e6 Eritrea    <NA>    232 I     6,228 390
## 5  1990 1.99e6 Ethiopia   <NA>    231 B R   607,~ 11,603
## 6  1990 1.99e6 Kenya      <NA>    404 B R   161,~ 37,825
## # ... with 522 more variables: `Other North` <chr>, Afghanistan <chr>,
## #   Albania <chr>, Algeria <chr>, `American Samoa` <chr>, Andorra <chr>,
## #   Angola <chr>, Anguilla <chr>, `Antigua and Barbuda` <chr>,
## #   Argentina <chr>, Armenia <chr>, Aruba <chr>, Australia <chr>,
## #   Austria <chr>, Azerbaijan <chr>, Bahamas <chr>, Bahrain <chr>,
## #   Bangladesh <chr>, Barbados <chr>, Belarus <chr>, Belgium <chr>,
## #   Belize <chr>, Benin <chr>, Bermuda <chr>, Bhutan <chr>, `Bolivia
## #   (Plurinational State of)` <chr>, `Bonaire, Sint Eustatius and
## #   Saba` <chr>, `Bosnia and Herzegovina` <chr>, Botswana <chr>,
## #   Brazil <chr>, `British Virgin Islands` <chr>, `Brunei
## #   Darussalam` <chr>, Bulgaria <chr>, `Burkina Faso` <chr>,
## #   Burundi <chr>, `Cabo Verde` <chr>, Cambodia <chr>, Cameroon <chr>,
## #   Canada <chr>, `Cayman Islands` <chr>, `Central African
## #   Republic` <chr>, Chad <chr>, `Channel Islands` <chr>, Chile <chr>,
## #   China <chr>, `China, Hong Kong SAR` <chr>, `China, Macao SAR` <chr>,
## #   Colombia <chr>, Comoros <chr>, Congo <chr>, `Cook Islands` <chr>,
## #   `Costa Rica` <chr>, `CÃ´te d'Ivoire` <chr>, Croatia <chr>, Cuba <chr>,
## #   CuraÃ§ao <chr>, Cyprus <chr>, Czechia <chr>, `Dem. People's Republic of
## #   Korea` <chr>, `Democratic Republic of the Congo` <chr>, Denmark <chr>,
## #   Djibouti <chr>, Dominica <chr>, `Dominican Republic` <chr>,
## #   Ecuador <chr>, Egypt <chr>, `El Salvador` <chr>, `Equatorial
## #   Guinea` <chr>, Eritrea <chr>, Estonia <chr>, Eswatini <chr>,
## #   Ethiopia <chr>, `Falkland Islands (Malvinas)` <chr>, `Faroe
## #   Islands` <chr>, Fiji <chr>, Finland <chr>, France <chr>, `French
## #   Guiana` <chr>, `French Polynesia` <chr>, Gabon <chr>, Gambia <chr>,
## #   Georgia <chr>, Germany <chr>, Ghana <chr>, Gibraltar <chr>,
## #   Greece <chr>, Greenland <chr>, Grenada <chr>, Guadeloupe <chr>,
## #   Guam <chr>, Guatemala <chr>, Guinea <chr>, `Guinea-Bissau` <chr>,
## #   Guyana <chr>, Haiti <chr>, `Holy See` <chr>, Honduras <chr>,
## #   Hungary <chr>, Iceland <chr>, India <chr>, ...
```

View all columns

The above printout shows a number of irrelevant columns that are not necessary for our analysis. Lets print out the entire column names and delete the unnecessary ones to have a cleaner data set.

```r
column_names <- colnames(male_migrants_by_country)
#column_names # umcomment to view entire list of column names
head(column_names)
```

```
## [1] "year"       "X2"         "country_to" "X4"         "X5"
## [6] "X6"
```

Exclude irrelevant columns

The above print out reveals that we do not need all column names that start with "X", "Total" or "Other". We delete these columns using the srtarts_with function.

```r
male_migrants_by_country <- male_migrants_by_country %>%
        select(-starts_with("X"), -starts_with("Other"), -starts_with("Total"))

head(male_migrants_by_country)
```

```
## # A tibble: 6 x 234
##    year country_to Afghanistan Albania Algeria `American Samoa` Andorra
##   <dbl> <chr>      <chr>       <chr>   <chr>   <chr>            <chr>
## 1  1990 Burundi    <NA>        <NA>    <NA>    <NA>             <NA>
## 2  1990 Comoros    <NA>        <NA>    <NA>    <NA>             <NA>
## 3  1990 Djibouti   <NA>        <NA>    <NA>    <NA>             <NA>
## 4  1990 Eritrea    <NA>        <NA>    <NA>    <NA>             <NA>
## 5  1990 Ethiopia   <NA>        <NA>    <NA>    <NA>             <NA>
## 6  1990 Kenya      <NA>        <NA>    <NA>    <NA>             <NA>
## # ... with 227 more variables: Angola <chr>, Anguilla <chr>, `Antigua and
## #   Barbuda` <chr>, Argentina <chr>, Armenia <chr>, Aruba <chr>,
## #   Australia <chr>, Austria <chr>, Azerbaijan <chr>, Bahamas <chr>,
## #   Bahrain <chr>, Bangladesh <chr>, Barbados <chr>, Belarus <chr>,
## #   Belgium <chr>, Belize <chr>, Benin <chr>, Bermuda <chr>, Bhutan <chr>,
## #   `Bolivia (Plurinational State of)` <chr>, `Bonaire, Sint Eustatius and
## #   Saba` <chr>, `Bosnia and Herzegovina` <chr>, Botswana <chr>,
## #   Brazil <chr>, `British Virgin Islands` <chr>, `Brunei
## #   Darussalam` <chr>, Bulgaria <chr>, `Burkina Faso` <chr>,
## #   Burundi <chr>, `Cabo Verde` <chr>, Cambodia <chr>, Cameroon <chr>,
## #   Canada <chr>, `Cayman Islands` <chr>, `Central African
## #   Republic` <chr>, Chad <chr>, `Channel Islands` <chr>, Chile <chr>,
## #   China <chr>, `China, Hong Kong SAR` <chr>, `China, Macao SAR` <chr>,
## #   Colombia <chr>, Comoros <chr>, Congo <chr>, `Cook Islands` <chr>,
## #   `Costa Rica` <chr>, `CÃ´te d'Ivoire` <chr>, Croatia <chr>, Cuba <chr>,
## #   CuraÃ§ao <chr>, Cyprus <chr>, Czechia <chr>, `Dem. People's Republic of
## #   Korea` <chr>, `Democratic Republic of the Congo` <chr>, Denmark <chr>,
## #   Djibouti <chr>, Dominica <chr>, `Dominican Republic` <chr>,
## #   Ecuador <chr>, Egypt <chr>, `El Salvador` <chr>, `Equatorial
## #   Guinea` <chr>, Eritrea <chr>, Estonia <chr>, Eswatini <chr>,
## #   Ethiopia <chr>, `Falkland Islands (Malvinas)` <chr>, `Faroe
## #   Islands` <chr>, Fiji <chr>, Finland <chr>, France <chr>, `French
## #   Guiana` <chr>, `French Polynesia` <chr>, Gabon <chr>, Gambia <chr>,
## #   Georgia <chr>, Germany <chr>, Ghana <chr>, Gibraltar <chr>,
## #   Greece <chr>, Greenland <chr>, Grenada <chr>, Guadeloupe <chr>,
## #   Guam <chr>, Guatemala <chr>, Guinea <chr>, `Guinea-Bissau` <chr>,
## #   Guyana <chr>, Haiti <chr>, `Holy See` <chr>, Honduras <chr>,
## #   Hungary <chr>, Iceland <chr>, India <chr>, Indonesia <chr>, `Iran
## #   (Islamic Republic of)` <chr>, Iraq <chr>, Ireland <chr>, `Isle of
## #   Man` <chr>, Israel <chr>, ...
```

View dimentions of resulting data frame

We use dim() function to have an idea of how many rows and columns we have for our analysis.

```
dim(male_migrants_by_country)
```

```
## [1] 1624  234
```

Confrim column names.

This is what we need for our analysis.

```
column_names_clean <- colnames(male_migrants_by_country)
#column_names_clean # uncomment to view entire list of cleaned up column names
head(column_names_clean)
```

```
## [1] "year"           "country_to"     "Afghanistan"    "Albania"
## [5] "Algeria"        "American Samoa"
```

View number of columns

Get the length of the column names to be used in the next line of code.

```
y <- length(colnames(male_migrants_by_country))

y
```

```
## [1] 234
```

Gather relevant columns

Let us use gather() function to gather all columns with country names from the 3rd column spanning the entire length of the columns into a single column and exclude any and all N/As to obtain clean data.

```
no_of_migrants_per_country <- gather(male_migrants_by_country, "country_from", "no_of_migrants", 3:y, na

head(no_of_migrants_per_country)
```

```
## # A tibble: 6 x 4
##    year country_to   country_from no_of_migrants
##   <dbl> <chr>        <chr>        <chr>
## 1  1990 Namibia      Afghanistan  26
## 2  1990 South Africa Afghanistan  37
## 3  1990 Egypt        Afghanistan  194
## 4  1990 Libya        Afghanistan  556
## 5  1990 Azerbaijan   Afghanistan  175
## 6  1990 Bahrain      Afghanistan  154
```

Conversion of chr to dbl

convert the no_of_migrants data column from characters to doubles for statistical analysis. This we will do using the parse_number() function. Print out using head() function the first 6 rows and confirm this conversion.

```
no_of_migrants_per_country$no_of_migrants <- parse_number(no_of_migrants_per_country$no_of_migrants)

clean_male_data <- no_of_migrants_per_country

head(clean_male_data)
```

```
## # A tibble: 6 x 4
##    year country_to   country_from no_of_migrants
##   <dbl> <chr>        <chr>                 <dbl>
## 1  1990 Namibia      Afghanistan              26
## 2  1990 South Africa Afghanistan              37
## 3  1990 Egypt        Afghanistan             194
## 4  1990 Libya        Afghanistan             556
## 5  1990 Azerbaijan   Afghanistan             175
## 6  1990 Bahrain      Afghanistan             154
```

Down stream analysis

Ordering of data

Ordering data by country with largest inflow of male migrants

```
by_country_to <- clean_male_data %>%
        group_by(year, country_from, country_to) %>%
        summarise(total_male_migrants = sum(no_of_migrants)) %>%
        arrange(desc(total_male_migrants))
head(by_country_to)
```

```
## # A tibble: 6 x 4
## # Groups:   year, country_from [6]
##    year country_from country_to               total_male_migrants
##   <dbl> <chr>        <chr>                                  <dbl>
## 1  2010 Mexico       United States of America             6554739
## 2  2015 Mexico       United States of America             6230901
## 3  2019 Mexico       United States of America             6138480
## 4  2005 Mexico       United States of America             5782166
## 5  2000 Mexico       United States of America             5104175
## 6  1995 Mexico       United States of America             3692951
```

Ordering the data by the total no of male migrants since 1995 to 2019.

```
total_migrants_since_1995 <- clean_male_data %>%
        group_by(country_from, country_to) %>%
        summarise(total_male_migrants = sum(no_of_migrants)) %>%
        arrange(desc(total_male_migrants))
head(total_migrants_since_1995)
```

```
## # A tibble: 6 x 3
## # Groups:   country_from [6]
##   country_from       country_to                 total_male_migrants
##   <chr>              <chr>                                    <dbl>
## 1 Mexico             United States of America              35819262
## 2 Bangladesh         India                                 13403551
## 3 Ukraine            Russian Federation                    10995103
## 4 Russian Federation Ukraine                               10963679
## 5 India              United Arab Emirates                   9860368
## 6 Afghanistan        Iran (Islamic Republic of)             9155798
```

Ordering the data by the countries sending out the least number of migrants

```
least_no_migrants_from <- clean_male_data %>%
        group_by(country_from) %>%
        summarise(total_migrants_since_1995 = sum(no_of_migrants)) %>%
        arrange(total_migrants_since_1995)
head(least_no_migrants_from)
```

```
## # A tibble: 6 x 2
##   country_from                total_migrants_since_1995
##   <chr>                                           <dbl>
## 1 Holy See                                          394
## 2 Saint Pierre and Miquelon                        3224
## 3 Falkland Islands (Malvinas)                      3302
## 4 Cayman Islands                                   4076
## 5 Nauru                                            6480
## 6 Tokelau                                          7105
```

Ordering the data by the countries receiving the largest number of imigrants since 1995.

```
largest_no_migrants_to <- clean_male_data %>%
        group_by(country_to) %>%
        summarise(total_migrants_since_1995 = sum(no_of_migrants)) %>%
        arrange(desc(total_migrants_since_1995))
head(largest_no_migrants_to)
```

```
## # A tibble: 6 x 2
##   country_to               total_migrants_since_1995
##   <chr>                                        <dbl>
## 1 United States of America                 123217813
## 2 Russian Federation                        40297028
## 3 Saudi Arabia                              35877383
## 4 Germany                                   32321606
## 5 France                                    23819337
## 6 United Arab Emirates                      23496867
```

Ordering the data by the countries receiving the least number of imigrants since 1995.

```
least_no_migrants_to <- clean_male_data %>%
        group_by(country_to) %>%
        summarise(total_migrants_since_1995 = sum(no_of_migrants)) %>%
        arrange(total_migrants_since_1995)

head(least_no_migrants_to)
```

```
## # A tibble: 6 x 2
##   country_to                total_migrants_since_1995
##   <chr>                                         <dbl>
## 1 Tuvalu                                          513
## 2 Saint Helena                                    867
## 3 Tokelau                                        1109
## 4 Niue                                           1781
## 5 Saint Pierre and Miquelon                      3833
## 6 Tonga                                          3888
```

Conclusion:

The top 5 countries receiving the largest mumber of male migrants are USA, Rusia Federation, Saudi Arabia, GErmany and France The top 5 countries receiving the least mumber of male migrants are Tivalu, Saint Helena, Tokelau, Niue and Saint Pierre and Miqueton

Female migrants

The second section will involve replicating the code above to analyse the immigration data on women migrants. This will serve as a confirmation of the replicability of the code to similar data.

Follow this link to see uploaded female migrants .csv file (https://raw.githubusercontent.com/igukusamuel/DATA-607-Project-2/master/UN_MigrantStockFemale_2019.csv)

```
female_migrants <- read_csv("https://raw.githubusercontent.com/igukusamuel/DATA-607-Project-2/master/UN_

#view(head(female_migrants, 20)) # uncomment to view data frame structure and see how many rows to skip
```

Skip first 15 rows

As part of data cleanup, skip the first 15 rows that include source information not relevant to out analysis.

```
female_migrants <- read_csv("https://raw.githubusercontent.com/igukusamuel/DATA-607-Project-2/master/UN_

head(female_migrants) #Print out first few rows to confirm that the data have been loaded correctly.
```

```
## # A tibble: 6 x 530
##      X1     X2 X3    X4       X5 X6    Total `Other South` `Other North`
##   <dbl>  <dbl> <chr> <chr> <dbl> <chr> <chr> <chr>         <chr>
## 1  1990 1.99e6 WORLD <NA>    900 <NA>  75,3~ 3,136,363     1,206,819
## 2  1990 1.99e6 UN d~ <NA>     NA <NA>  ..    ..            ..
## 3  1990 1.99e6 More~ b       901 <NA>  42,3~ 1,575,254     569,867
## 4  1990 1.99e6 Less~ c       902 <NA>  33,0~ 1,561,109     636,952
## 5  1990 1.99e6 Leas~ d       941 <NA>  5,50~ 238,252       104,494
## 6  1990 1.99e6 Less~ <NA>    934 <NA>  27,4~ 1,322,857     532,458
## # ... with 521 more variables: Afghanistan <chr>, Albania <chr>,
## #   Algeria <chr>, `American Samoa` <chr>, Andorra <chr>, Angola <chr>,
## #   Anguilla <chr>, `Antigua and Barbuda` <chr>, Argentina <chr>,
## #   Armenia <chr>, Aruba <chr>, Australia <chr>, Austria <chr>,
## #   Azerbaijan <chr>, Bahamas <chr>, Bahrain <chr>, Bangladesh <chr>,
## #   Barbados <chr>, Belarus <chr>, Belgium <chr>, Belize <chr>,
## #   Benin <chr>, Bermuda <chr>, Bhutan <chr>, `Bolivia (Plurinational
## #   State of)` <chr>, `Bonaire, Sint Eustatius and Saba` <chr>, `Bosnia
## #   and Herzegovina` <chr>, Botswana <chr>, Brazil <chr>, `British Virgin
## #   Islands` <chr>, `Brunei Darussalam` <chr>, Bulgaria <chr>, `Burkina
## #   Faso` <chr>, Burundi <chr>, `Cabo Verde` <chr>, Cambodia <chr>,
## #   Cameroon <chr>, Canada <chr>, `Cayman Islands` <chr>, `Central African
## #   Republic` <chr>, Chad <chr>, `Channel Islands` <chr>, Chile <chr>,
## #   China <chr>, `China, Hong Kong SAR` <chr>, `China, Macao SAR` <chr>,
## #   Colombia <chr>, Comoros <chr>, Congo <chr>, `Cook Islands` <chr>,
## #   `Costa Rica` <chr>, `CÃ´te d'Ivoire` <chr>, Croatia <chr>, Cuba <chr>,
## #   CuraÃ§ao <chr>, Cyprus <chr>, Czechia <chr>, `Dem. People's Republic of
## #   Korea` <chr>, `Democratic Republic of the Congo` <chr>, Denmark <chr>,
## #   Djibouti <chr>, Dominica <chr>, `Dominican Republic` <chr>,
## #   Ecuador <chr>, Egypt <chr>, `El Salvador` <chr>, `Equatorial
## #   Guinea` <chr>, Eritrea <chr>, Estonia <chr>, Eswatini <chr>,
## #   Ethiopia <chr>, `Falkland Islands (Malvinas)` <chr>, `Faroe
## #   Islands` <chr>, Fiji <chr>, Finland <chr>, France <chr>, `French
## #   Guiana` <chr>, `French Polynesia` <chr>, Gabon <chr>, Gambia <chr>,
## #   Georgia <chr>, Germany <chr>, Ghana <chr>, Gibraltar <chr>,
## #   Greece <chr>, Greenland <chr>, Grenada <chr>, Guadeloupe <chr>,
## #   Guam <chr>, Guatemala <chr>, Guinea <chr>, `Guinea-Bissau` <chr>,
## #   Guyana <chr>, Haiti <chr>, `Holy See` <chr>, Honduras <chr>,
## #   Hungary <chr>, Iceland <chr>, India <chr>, Indonesia <chr>, ...
```

Filter for N/As in column X6

Careful review of the data shows that column named X6 only includes data for rows related to countries and N/A's for rows relating to regions and regional totals. Thus filtering out all N/As in column X6 will leave us with country data only, which is the basis of out analysis. We first view all the N/As under column X6 to confirm none of them relate to country information.

```r
colX6 <- filter(female_migrants, is.na(X6))

a <- length(colX6)
a
```

```
## [1] 530
```

```r
head(colX6)
```

```
## # A tibble: 6 x 530
##      X1     X2 X3    X4       X5 X6    Total `Other South` `Other North`
##   <dbl>  <dbl> <chr> <chr> <dbl> <chr> <chr> <chr>         <chr>
## 1  1990 1.99e6 WORLD <NA>    900 <NA>  75,3~ 3,136,363     1,206,819
## 2  1990 1.99e6 UN d~ <NA>     NA <NA>  ..    ..            ..
## 3  1990 1.99e6 More~ b       901 <NA>  42,3~ 1,575,254     569,867
## 4  1990 1.99e6 Less~ c       902 <NA>  33,0~ 1,561,109     636,952
## 5  1990 1.99e6 Leas~ d       941 <NA>  5,50~ 238,252       104,494
## 6  1990 1.99e6 Less~ <NA>    934 <NA>  27,4~ 1,322,857     532,458
## # ... with 521 more variables: Afghanistan <chr>, Albania <chr>,
## #   Algeria <chr>, `American Samoa` <chr>, Andorra <chr>, Angola <chr>,
## #   Anguilla <chr>, `Antigua and Barbuda` <chr>, Argentina <chr>,
## #   Armenia <chr>, Aruba <chr>, Australia <chr>, Austria <chr>,
## #   Azerbaijan <chr>, Bahamas <chr>, Bahrain <chr>, Bangladesh <chr>,
## #   Barbados <chr>, Belarus <chr>, Belgium <chr>, Belize <chr>,
## #   Benin <chr>, Bermuda <chr>, Bhutan <chr>, `Bolivia (Plurinational
## #   State of)` <chr>, `Bonaire, Sint Eustatius and Saba` <chr>, `Bosnia
## #   and Herzegovina` <chr>, Botswana <chr>, Brazil <chr>, `British Virgin
## #   Islands` <chr>, `Brunei Darussalam` <chr>, Bulgaria <chr>, `Burkina
## #   Faso` <chr>, Burundi <chr>, `Cabo Verde` <chr>, Cambodia <chr>,
## #   Cameroon <chr>, Canada <chr>, `Cayman Islands` <chr>, `Central African
## #   Republic` <chr>, Chad <chr>, `Channel Islands` <chr>, Chile <chr>,
## #   China <chr>, `China, Hong Kong SAR` <chr>, `China, Macao SAR` <chr>,
## #   Colombia <chr>, Comoros <chr>, Congo <chr>, `Cook Islands` <chr>,
## #   `Costa Rica` <chr>, `CÃ´te d'Ivoire` <chr>, Croatia <chr>, Cuba <chr>,
## #   CuraÃ§ao <chr>, Cyprus <chr>, Czechia <chr>, `Dem. People's Republic of
## #   Korea` <chr>, `Democratic Republic of the Congo` <chr>, Denmark <chr>,
## #   Djibouti <chr>, Dominica <chr>, `Dominican Republic` <chr>,
## #   Ecuador <chr>, Egypt <chr>, `El Salvador` <chr>, `Equatorial
## #   Guinea` <chr>, Eritrea <chr>, Estonia <chr>, Eswatini <chr>,
## #   Ethiopia <chr>, `Falkland Islands (Malvinas)` <chr>, `Faroe
## #   Islands` <chr>, Fiji <chr>, Finland <chr>, France <chr>, `French
## #   Guiana` <chr>, `French Polynesia` <chr>, Gabon <chr>, Gambia <chr>,
## #   Georgia <chr>, Germany <chr>, Ghana <chr>, Gibraltar <chr>,
## #   Greece <chr>, Greenland <chr>, Grenada <chr>, Guadeloupe <chr>,
## #   Guam <chr>, Guatemala <chr>, Guinea <chr>, `Guinea-Bissau` <chr>,
## #   Guyana <chr>, Haiti <chr>, `Holy See` <chr>, Honduras <chr>,
## #   Hungary <chr>, Iceland <chr>, India <chr>, Indonesia <chr>, ...
```

Exclude N/As in column X6

We then exclude all N/A's in column X6 and print out the first 6 rows using the head() function.

```
female_migrants_by_country <- filter(female_migrants, !is.na(X6))

head(female_migrants_by_country)
```

```
## # A tibble: 6 x 530
##      X1     X2 X3    X4       X5 X6    Total `Other South` `Other North`
##   <dbl>  <dbl> <chr> <chr> <dbl> <chr> <chr> <chr>         <chr>
## 1  1990 1.99e6 Buru~ <NA>    108 B R   169,~ 25,839        4,560
## 2  1990 1.99e6 Como~ <NA>    174 B     7,362 415           330
## 3  1990 1.99e6 Djib~ <NA>    262 B R   57,9~ 2,428         809
## 4  1990 1.99e6 Erit~ <NA>    232 I     5,620 347           166
## 5  1990 1.99e6 Ethi~ <NA>    231 B R   548,~ 10,472        3,490
## 6  1990 1.99e6 Kenya <NA>    404 B R   136,~ 28,123        16,506
## # ... with 521 more variables: Afghanistan <chr>, Albania <chr>,
## #   Algeria <chr>, `American Samoa` <chr>, Andorra <chr>, Angola <chr>,
## #   Anguilla <chr>, `Antigua and Barbuda` <chr>, Argentina <chr>,
## #   Armenia <chr>, Aruba <chr>, Australia <chr>, Austria <chr>,
## #   Azerbaijan <chr>, Bahamas <chr>, Bahrain <chr>, Bangladesh <chr>,
## #   Barbados <chr>, Belarus <chr>, Belgium <chr>, Belize <chr>,
## #   Benin <chr>, Bermuda <chr>, Bhutan <chr>, `Bolivia (Plurinational
## #   State of)` <chr>, `Bonaire, Sint Eustatius and Saba` <chr>, `Bosnia
## #   and Herzegovina` <chr>, Botswana <chr>, Brazil <chr>, `British Virgin
## #   Islands` <chr>, `Brunei Darussalam` <chr>, Bulgaria <chr>, `Burkina
## #   Faso` <chr>, Burundi <chr>, `Cabo Verde` <chr>, Cambodia <chr>,
## #   Cameroon <chr>, Canada <chr>, `Cayman Islands` <chr>, `Central African
## #   Republic` <chr>, Chad <chr>, `Channel Islands` <chr>, Chile <chr>,
## #   China <chr>, `China, Hong Kong SAR` <chr>, `China, Macao SAR` <chr>,
## #   Colombia <chr>, Comoros <chr>, Congo <chr>, `Cook Islands` <chr>,
## #   `Costa Rica` <chr>, `CÃ´te d'Ivoire` <chr>, Croatia <chr>, Cuba <chr>,
## #   CuraÃ§ao <chr>, Cyprus <chr>, Czechia <chr>, `Dem. People's Republic of
## #   Korea` <chr>, `Democratic Republic of the Congo` <chr>, Denmark <chr>,
## #   Djibouti <chr>, Dominica <chr>, `Dominican Republic` <chr>,
## #   Ecuador <chr>, Egypt <chr>, `El Salvador` <chr>, `Equatorial
## #   Guinea` <chr>, Eritrea <chr>, Estonia <chr>, Eswatini <chr>,
## #   Ethiopia <chr>, `Falkland Islands (Malvinas)` <chr>, `Faroe
## #   Islands` <chr>, Fiji <chr>, Finland <chr>, France <chr>, `French
## #   Guiana` <chr>, `French Polynesia` <chr>, Gabon <chr>, Gambia <chr>,
## #   Georgia <chr>, Germany <chr>, Ghana <chr>, Gibraltar <chr>,
## #   Greece <chr>, Greenland <chr>, Grenada <chr>, Guadeloupe <chr>,
## #   Guam <chr>, Guatemala <chr>, Guinea <chr>, `Guinea-Bissau` <chr>,
## #   Guyana <chr>, Haiti <chr>, `Holy See` <chr>, Honduras <chr>,
## #   Hungary <chr>, Iceland <chr>, India <chr>, Indonesia <chr>, ...
```

Rename column X1 and X3

From the above print out, there is need to rename column X1 and X3 as year and country_to respectively.

```
female_migrants_by_country <- female_migrants_by_country %>%
        rename(
                year = X1,
                country_to = X3
        )
head(female_migrants_by_country)
```

```
## # A tibble: 6 x 530
##    year     X2 country_to X4       X5 X6    Total `Other South`
##   <dbl>  <dbl> <chr>      <chr> <dbl> <chr> <chr> <chr>
## 1  1990 1.99e6 Burundi    <NA>    108 B R   169,~ 25,839
## 2  1990 1.99e6 Comoros    <NA>    174 B     7,362 415
## 3  1990 1.99e6 Djibouti   <NA>    262 B R   57,9~ 2,428
## 4  1990 1.99e6 Eritrea    <NA>    232 I     5,620 347
## 5  1990 1.99e6 Ethiopia   <NA>    231 B R   548,~ 10,472
## 6  1990 1.99e6 Kenya      <NA>    404 B R   136,~ 28,123
## # ... with 522 more variables: `Other North` <chr>, Afghanistan <chr>,
## #   Albania <chr>, Algeria <chr>, `American Samoa` <chr>, Andorra <chr>,
## #   Angola <chr>, Anguilla <chr>, `Antigua and Barbuda` <chr>,
## #   Argentina <chr>, Armenia <chr>, Aruba <chr>, Australia <chr>,
## #   Austria <chr>, Azerbaijan <chr>, Bahamas <chr>, Bahrain <chr>,
## #   Bangladesh <chr>, Barbados <chr>, Belarus <chr>, Belgium <chr>,
## #   Belize <chr>, Benin <chr>, Bermuda <chr>, Bhutan <chr>, `Bolivia
## #   (Plurinational State of)` <chr>, `Bonaire, Sint Eustatius and
## #   Saba` <chr>, `Bosnia and Herzegovina` <chr>, Botswana <chr>,
## #   Brazil <chr>, `British Virgin Islands` <chr>, `Brunei
## #   Darussalam` <chr>, Bulgaria <chr>, `Burkina Faso` <chr>,
## #   Burundi <chr>, `Cabo Verde` <chr>, Cambodia <chr>, Cameroon <chr>,
## #   Canada <chr>, `Cayman Islands` <chr>, `Central African
## #   Republic` <chr>, Chad <chr>, `Channel Islands` <chr>, Chile <chr>,
## #   China <chr>, `China, Hong Kong SAR` <chr>, `China, Macao SAR` <chr>,
## #   Colombia <chr>, Comoros <chr>, Congo <chr>, `Cook Islands` <chr>,
## #   `Costa Rica` <chr>, `CÃ´te d'Ivoire` <chr>, Croatia <chr>, Cuba <chr>,
## #   CuraÃ§ao <chr>, Cyprus <chr>, Czechia <chr>, `Dem. People's Republic of
## #   Korea` <chr>, `Democratic Republic of the Congo` <chr>, Denmark <chr>,
## #   Djibouti <chr>, Dominica <chr>, `Dominican Republic` <chr>,
## #   Ecuador <chr>, Egypt <chr>, `El Salvador` <chr>, `Equatorial
## #   Guinea` <chr>, Eritrea <chr>, Estonia <chr>, Eswatini <chr>,
## #   Ethiopia <chr>, `Falkland Islands (Malvinas)` <chr>, `Faroe
## #   Islands` <chr>, Fiji <chr>, Finland <chr>, France <chr>, `French
## #   Guiana` <chr>, `French Polynesia` <chr>, Gabon <chr>, Gambia <chr>,
## #   Georgia <chr>, Germany <chr>, Ghana <chr>, Gibraltar <chr>,
## #   Greece <chr>, Greenland <chr>, Grenada <chr>, Guadeloupe <chr>,
## #   Guam <chr>, Guatemala <chr>, Guinea <chr>, `Guinea-Bissau` <chr>,
## #   Guyana <chr>, Haiti <chr>, `Holy See` <chr>, Honduras <chr>,
## #   Hungary <chr>, Iceland <chr>, India <chr>, ...
```

View all columns

The above printout shows a number of irrelevant columns that are not necessary for our analysis. Lets print out the entire column names and delete the unnecessary ones to have a cleaner data set.

```
female_col_names <- colnames(female_migrants_by_country)
#female_col_names # uncomment to view entire list of column names
head(female_col_names)
```

```
## [1] "year"       "X2"         "country_to" "X4"         "X5"
## [6] "X6"
```

Exclude irrelevant columns

The above print out reveals that we do not need all column names that start with "X", "Total" or "Other". We delete these columns using the srtarts_with function.

```
female_migrants_by_country <- female_migrants_by_country %>%
        select(-starts_with("X"), -starts_with("Other"), -starts_with("Total"))

head(female_migrants_by_country)
```

```
## # A tibble: 6 x 234
##    year country_to Afghanistan Albania Algeria `American Samoa` Andorra
##   <dbl> <chr>      <chr>       <chr>   <chr>   <chr>            <chr>
## 1  1990 Burundi    <NA>        <NA>    <NA>    <NA>             <NA>
## 2  1990 Comoros    <NA>        <NA>    <NA>    <NA>             <NA>
## 3  1990 Djibouti   <NA>        <NA>    <NA>    <NA>             <NA>
## 4  1990 Eritrea    <NA>        <NA>    <NA>    <NA>             <NA>
## 5  1990 Ethiopia   <NA>        <NA>    <NA>    <NA>             <NA>
## 6  1990 Kenya      <NA>        <NA>    <NA>    <NA>             <NA>
## # ... with 227 more variables: Angola <chr>, Anguilla <chr>, `Antigua and
## #   Barbuda` <chr>, Argentina <chr>, Armenia <chr>, Aruba <chr>,
## #   Australia <chr>, Austria <chr>, Azerbaijan <chr>, Bahamas <chr>,
## #   Bahrain <chr>, Bangladesh <chr>, Barbados <chr>, Belarus <chr>,
## #   Belgium <chr>, Belize <chr>, Benin <chr>, Bermuda <chr>, Bhutan <chr>,
## #   `Bolivia (Plurinational State of)` <chr>, `Bonaire, Sint Eustatius and
## #   Saba` <chr>, `Bosnia and Herzegovina` <chr>, Botswana <chr>,
## #   Brazil <chr>, `British Virgin Islands` <chr>, `Brunei
## #   Darussalam` <chr>, Bulgaria <chr>, `Burkina Faso` <chr>,
## #   Burundi <chr>, `Cabo Verde` <chr>, Cambodia <chr>, Cameroon <chr>,
## #   Canada <chr>, `Cayman Islands` <chr>, `Central African
## #   Republic` <chr>, Chad <chr>, `Channel Islands` <chr>, Chile <chr>,
## #   China <chr>, `China, Hong Kong SAR` <chr>, `China, Macao SAR` <chr>,
## #   Colombia <chr>, Comoros <chr>, Congo <chr>, `Cook Islands` <chr>,
## #   `Costa Rica` <chr>, `CÃ´te d'Ivoire` <chr>, Croatia <chr>, Cuba <chr>,
## #   CuraÃ§ao <chr>, Cyprus <chr>, Czechia <chr>, `Dem. People's Republic of
## #   Korea` <chr>, `Democratic Republic of the Congo` <chr>, Denmark <chr>,
## #   Djibouti <chr>, Dominica <chr>, `Dominican Republic` <chr>,
## #   Ecuador <chr>, Egypt <chr>, `El Salvador` <chr>, `Equatorial
## #   Guinea` <chr>, Eritrea <chr>, Estonia <chr>, Eswatini <chr>,
## #   Ethiopia <chr>, `Falkland Islands (Malvinas)` <chr>, `Faroe
## #   Islands` <chr>, Fiji <chr>, Finland <chr>, France <chr>, `French
## #   Guiana` <chr>, `French Polynesia` <chr>, Gabon <chr>, Gambia <chr>,
## #   Georgia <chr>, Germany <chr>, Ghana <chr>, Gibraltar <chr>,
## #   Greece <chr>, Greenland <chr>, Grenada <chr>, Guadeloupe <chr>,
## #   Guam <chr>, Guatemala <chr>, Guinea <chr>, `Guinea-Bissau` <chr>,
## #   Guyana <chr>, Haiti <chr>, `Holy See` <chr>, Honduras <chr>,
## #   Hungary <chr>, Iceland <chr>, India <chr>, Indonesia <chr>, `Iran
## #   (Islamic Republic of)` <chr>, Iraq <chr>, Ireland <chr>, `Isle of
## #   Man` <chr>, Israel <chr>, ...
```

View dimentions of resulting data frame

We use dim() function to have an idea of how many rows and columns we have for our analysis.

```r
dim(female_migrants_by_country)
```

```
## [1] 1624  234
```

Confrim column names.

The print out below is a confrimation of the column names. This is what we need for our analysis.

```r
clean_female_col_name <- colnames(female_migrants_by_country)
#clean_female_col_name # uncomment to view entire list of clean column names
clean_female_col_name
```

```
##   [1] "year"
##   [2] "country_to"
##   [3] "Afghanistan"
##   [4] "Albania"
##   [5] "Algeria"
##   [6] "American Samoa"
##   [7] "Andorra"
##   [8] "Angola"
##   [9] "Anguilla"
##  [10] "Antigua and Barbuda"
##  [11] "Argentina"
##  [12] "Armenia"
##  [13] "Aruba"
##  [14] "Australia"
##  [15] "Austria"
##  [16] "Azerbaijan"
##  [17] "Bahamas"
##  [18] "Bahrain"
##  [19] "Bangladesh"
##  [20] "Barbados"
##  [21] "Belarus"
##  [22] "Belgium"
##  [23] "Belize"
##  [24] "Benin"
##  [25] "Bermuda"
##  [26] "Bhutan"
##  [27] "Bolivia (Plurinational State of)"
##  [28] "Bonaire, Sint Eustatius and Saba"
##  [29] "Bosnia and Herzegovina"
##  [30] "Botswana"
##  [31] "Brazil"
##  [32] "British Virgin Islands"
##  [33] "Brunei Darussalam"
##  [34] "Bulgaria"
##  [35] "Burkina Faso"
##  [36] "Burundi"
##  [37] "Cabo Verde"
##  [38] "Cambodia"
##  [39] "Cameroon"
##  [40] "Canada"
##  [41] "Cayman Islands"
##  [42] "Central African Republic"
##  [43] "Chad"
##  [44] "Channel Islands"
##  [45] "Chile"
##  [46] "China"
##  [47] "China, Hong Kong SAR"
##  [48] "China, Macao SAR"
##  [49] "Colombia"
##  [50] "Comoros"
##  [51] "Congo"
##  [52] "Cook Islands"
##  [53] "Costa Rica"
##  [54] "CÃ´te d'Ivoire"
##  [55] "Croatia"
##  [56] "Cuba"
##  [57] "CuraÃ§ao"
##  [58] "Cyprus"
##  [59] "Czechia"
##  [60] "Dem. People's Republic of Korea"
##  [61] "Democratic Republic of the Congo"
##  [62] "Denmark"
##  [63] "Djibouti"
##  [64] "Dominica"
##  [65] "Dominican Republic"
##  [66] "Ecuador"
##  [67] "Egypt"
##  [68] "El Salvador"
##  [69] "Equatorial Guinea"
##  [70] "Eritrea"
##  [71] "Estonia"
##  [72] "Eswatini"
##  [73] "Ethiopia"
##  [74] "Falkland Islands (Malvinas)"
##  [75] "Faroe Islands"
##  [76] "Fiji"
##  [77] "Finland"
##  [78] "France"
##  [79] "French Guiana"
##  [80] "French Polynesia"
##  [81] "Gabon"
##  [82] "Gambia"
##  [83] "Georgia"
##  [84] "Germany"
##  [85] "Ghana"
##  [86] "Gibraltar"
##  [87] "Greece"
##  [88] "Greenland"
##  [89] "Grenada"
##  [90] "Guadeloupe"
##  [91] "Guam"
##  [92] "Guatemala"
##  [93] "Guinea"
##  [94] "Guinea-Bissau"
##  [95] "Guyana"
##  [96] "Haiti"
##  [97] "Holy See"
##  [98] "Honduras"
##  [99] "Hungary"
## [100] "Iceland"
## [101] "India"
## [102] "Indonesia"
## [103] "Iran (Islamic Republic of)"
## [104] "Iraq"
## [105] "Ireland"
## [106] "Isle of Man"
## [107] "Israel"
## [108] "Italy"
## [109] "Jamaica"
## [110] "Japan"
## [111] "Jordan"
## [112] "Kazakhstan"
## [113] "Kenya"
## [114] "Kiribati"
## [115] "Kuwait"
## [116] "Kyrgyzstan"
## [117] "Lao People's Democratic Republic"
## [118] "Latvia"
## [119] "Lebanon"
## [120] "Lesotho"
## [121] "Liberia"
## [122] "Libya"
## [123] "Liechtenstein"
## [124] "Lithuania"
## [125] "Luxembourg"
## [126] "Madagascar"
## [127] "Malawi"
## [128] "Malaysia"
## [129] "Maldives"
## [130] "Mali"
## [131] "Malta"
## [132] "Marshall Islands"
## [133] "Martinique"
## [134] "Mauritania"
## [135] "Mauritius"
## [136] "Mayotte"
## [137] "Mexico"
## [138] "Micronesia (Fed. States of)"
## [139] "Monaco"
## [140] "Mongolia"
## [141] "Montenegro"
## [142] "Montserrat"
## [143] "Morocco"
## [144] "Mozambique"
## [145] "Myanmar"
## [146] "Namibia"
## [147] "Nauru"
## [148] "Nepal"
## [149] "Netherlands"
## [150] "New Caledonia"
## [151] "New Zealand"
## [152] "Nicaragua"
## [153] "Niger"
## [154] "Nigeria"
## [155] "Niue"
## [156] "North Macedonia"
## [157] "Northern Mariana Islands"
## [158] "Norway"
## [159] "Oman"
## [160] "Pakistan"
## [161] "Palau"
## [162] "Panama"
## [163] "Papua New Guinea"
## [164] "Paraguay"
## [165] "Peru"
## [166] "Philippines"
## [167] "Poland"
## [168] "Portugal"
## [169] "Puerto Rico"
## [170] "Qatar"
## [171] "Republic of Korea"
## [172] "Republic of Moldova"
## [173] "RÃ©union"
## [174] "Romania"
## [175] "Russian Federation"
## [176] "Rwanda"
## [177] "Saint Helena"
## [178] "Saint Kitts and Nevis"
## [179] "Saint Lucia"
## [180] "Saint Pierre and Miquelon"
## [181] "Saint Vincent and the Grenadines"
## [182] "Samoa"
## [183] "San Marino"
## [184] "Sao Tome and Principe"
## [185] "Saudi Arabia"
## [186] "Senegal"
## [187] "Serbia"
## [188] "Seychelles"
## [189] "Sierra Leone"
## [190] "Singapore"
## [191] "Sint Maarten (Dutch part)"
## [192] "Slovakia"
## [193] "Slovenia"
## [194] "Solomon Islands"
## [195] "Somalia"
## [196] "South Africa"
## [197] "South Sudan"
## [198] "Spain"
## [199] "Sri Lanka"
## [200] "State of Palestine"
## [201] "Sudan"
## [202] "Suriname"
## [203] "Sweden"
## [204] "Switzerland"
## [205] "Syrian Arab Republic"
## [206] "Tajikistan"
## [207] "Thailand"
## [208] "Timor-Leste"
## [209] "Togo"
## [210] "Tokelau"
## [211] "Tonga"
## [212] "Trinidad and Tobago"
## [213] "Tunisia"
## [214] "Turkey"
## [215] "Turkmenistan"
## [216] "Turks and Caicos Islands"
## [217] "Tuvalu"
## [218] "Uganda"
## [219] "Ukraine"
## [220] "United Arab Emirates"
## [221] "United Kingdom"
## [222] "United Republic of Tanzania"
## [223] "United States of America"
## [224] "United States Virgin Islands"
## [225] "Uruguay"
## [226] "Uzbekistan"
## [227] "Vanuatu"
## [228] "Venezuela (Bolivarian Republic of)"
## [229] "Viet Nam"
## [230] "Wallis and Futuna Islands"
## [231] "Western Sahara"
## [232] "Yemen"
## [233] "Zambia"
## [234] "Zimbabwe"
```

View number of columns

Get the length of the column names to be used in the next line of code.

```
y <- length(colnames(female_migrants_by_country))

y
```

```
## [1] 234
```

Gather relevant columns

Let us use gather() function to gather all columns with country names from the 3rd column spanning the entire length of the columns into a single column and exclude any and all N/As to obtain clean data.

```
no_of_female_migrants_per_country <- gather(female_migrants_by_country, "country_from", "no_of_female_mi

head(no_of_female_migrants_per_country)
```

```
## # A tibble: 6 x 4
##    year country_to   country_from no_of_female_migrants
##   <dbl> <chr>        <chr>        <chr>
## 1  1990 Namibia      Afghanistan  38
## 2  1990 South Africa Afghanistan  22
## 3  1990 Egypt        Afghanistan  43
## 4  1990 Libya        Afghanistan  121
## 5  1990 Azerbaijan   Afghanistan  79
## 6  1990 Bahrain      Afghanistan  61
```

Conversion of chr to dbl

convert the no_of_migrants data column from characters to doubles for statistical analysis. This we will do using the parse_number() function. Print out using head() function the first 6 rows and confirm this conversion.

```
no_of_female_migrants_per_country$no_of_female_migrants <- parse_number(no_of_female_migrants_per_count

clean_female_data <- no_of_female_migrants_per_country

head(clean_female_data)
```

```
## # A tibble: 6 x 4
##    year country_to   country_from no_of_female_migrants
##   <dbl> <chr>        <chr>                        <dbl>
## 1  1990 Namibia      Afghanistan                     38
## 2  1990 South Africa Afghanistan                     22
## 3  1990 Egypt        Afghanistan                     43
## 4  1990 Libya        Afghanistan                    121
## 5  1990 Azerbaijan   Afghanistan                     79
## 6  1990 Bahrain      Afghanistan                     61
```

Down stream analysis

Ordering of data

Ordering data by country with largest inflow of male migrants

```
female_by_country_to <- clean_female_data %>%
        group_by(year, country_from, country_to) %>%
        summarise(total_female_migrants = sum(no_of_female_migrants)) %>%
        arrange(desc(total_female_migrants))
head(female_by_country_to)
```

```
## # A tibble: 6 x 4
## # Groups:   year, country_from [6]
##    year country_from country_to               total_female_migrants
##   <dbl> <chr>        <chr>                                    <dbl>
## 1  2010 Mexico       United States of America               5613923
## 2  2015 Mexico       United States of America               5412397
## 3  2019 Mexico       United States of America               5351204
## 4  2005 Mexico       United States of America               4828898
## 5  2000 Mexico       United States of America               4306354
## 6  1995 Mexico       United States of America               3134994
```

Ordering the data by the total no of male migrants since 1995 to 2019.

```
total_female_migrants_since_1995 <- clean_female_data %>%
        group_by(country_from, country_to) %>%
        summarise(total_female_migrants = sum(no_of_female_migrants)) %>%
        arrange(desc(total_female_migrants))
head(total_female_migrants_since_1995)
```

```
## # A tibble: 6 x 3
## # Groups:   country_from [5]
##   country_from       country_to               total_female_migrants
##   <chr>              <chr>                                    <dbl>
## 1 Mexico             United States of America              30629934
## 2 Russian Federation Ukraine                               15428450
## 3 Ukraine            Russian Federation                    12145118
## 4 Bangladesh         India                                 12110209
## 5 Kazakhstan         Russian Federation                     9321252
## 6 Russian Federation Kazakhstan                             8457532
```

Ordering the data by the countries sending out the least number of migrants

```
least_no_female_migrants_from <- clean_female_data %>%
        group_by(country_from) %>%
        summarise(total_female_migrants_since_1995 = sum(no_of_female_migrants)) %>%
        arrange(total_female_migrants_since_1995)
head(least_no_female_migrants_from)
```

```
## # A tibble: 6 x 2
##   country_from                total_female_migrants_since_1995
##   <chr>                                                  <dbl>
## 1 Holy See                                                 531
## 2 Saint Pierre and Miquelon                               3603
## 3 Falkland Islands (Malvinas)                             4330
## 4 Cayman Islands                                          4793
## 5 Nauru                                                   6775
## 6 Tokelau                                                 7121
```

Ordering the data by the countries receiving the largest number of imigrants since 1995.

```
largest_no_female_migrants_to <- clean_female_data %>%
        group_by(country_to) %>%
        summarise(total_female_migrants_since_1995 = sum(no_of_female_migrants)) %>%
        arrange(desc(total_female_migrants_since_1995))
head(largest_no_female_migrants_to)
```

```
## # A tibble: 6 x 2
##   country_to               total_female_migrants_since_1995
##   <chr>                                               <dbl>
## 1 United States of America                        129578823
## 2 Russian Federation                                41170668
## 3 Germany                                           31169216
## 4 France                                            24701274
## 5 United Kingdom                                    22577016
## 6 Canada                                            22317763
```

Ordering the data by the countries receiving the least number of imigrants since 1995.

```
least_no_female_migrants_to <- clean_female_data %>%
        group_by(country_to) %>%
        summarise(total_female_migrants_since_1995 = sum(no_of_female_migrants)) %>%
        arrange(total_female_migrants_since_1995)

head(least_no_female_migrants_to)
```

```
## # A tibble: 6 x 2
##   country_to                  total_female_migrants_since_1995
##   <chr>                                                  <dbl>
## 1 Saint Helena                                             513
## 2 Tuvalu                                                   547
## 3 Tokelau                                                 1162
## 4 Niue                                                    1490
## 5 Micronesia (Fed. States of)                             3173
## 6 Tonga                                                   3565
```

Conclusion:

The top 5 countries receiving the largest mumber of female migrants are USA, Rusia Federation, Germany, France and United Kingdom. The top 5 countries receiving the least mumber of female migrants are Saint Helena, Tivalu, Tokelau, Niue and Micronesia (Fed. States of).

Migrants by destimation country

Upload the data into Github

This will ensure that everyone with access to the github repository can easily audit or retest the data. This ensures ease of accessibility and testing by a wide audience. Follow this link to see uploaded Male migrants .csv file (https://raw.githubusercontent.com/igukusamuel/DATA-607-Project-2/master/UN_MigrantStockBySexByDestination_2019.csv)

```
migrants <- read_csv("https://raw.githubusercontent.com/igukusamuel/DATA-607-Project-2/master/UN_Migrant
head(migrants)
```

```
## # A tibble: 6 x 26
##   X1    X2    X3    X4    X5      X6    X7    X8    X9    X10   X11   X12
##   <chr> <chr> <chr> <chr> <chr>   <chr> <chr> <chr> <chr> <chr> <chr> <chr>
## 1 <NA>  <NA>  <NA>  <NA>  <NA>    <NA>  <NA>  <NA>  <NA>  <NA>  <NA>  <NA>
## 2 <NA>  <NA>  <NA>  <NA>  <NA>    <NA>  <NA>  <NA>  <NA>  <NA>  <NA>  <NA>
## 3 <NA>  <NA>  <NA>  <NA>  <NA>    <NA>  <NA>  <NA>  <NA>  <NA>  <NA>  <NA>
## 4 <NA>  <NA>  <NA>  <NA>  United~ <NA>  <NA>  <NA>  <NA>  <NA>  <NA>  <NA>
## 5 <NA>  <NA>  <NA>  <NA>  Popula~ <NA>  <NA>  <NA>  <NA>  <NA>  <NA>  <NA>
## 6 <NA>  <NA>  <NA>  <NA>  Depart~ <NA>  <NA>  <NA>  <NA>  <NA>  <NA>  <NA>
## # ... with 14 more variables: X13 <chr>, X14 <chr>, X15 <chr>, X16 <chr>,
## #   X17 <chr>, X18 <chr>, X19 <chr>, X20 <chr>, X21 <chr>, X22 <chr>,
## #   X23 <chr>, X24 <chr>, X25 <chr>, X26 <chr>
```

```
#view(head(male_migrants, 20)) # vIew data frame structure and see how many rows to skip.
```

Skip first 15 rows

As part of data cleanup, skip the first 15 rows that include source information not relevant to out analysis.

```
migrants <- read_csv("https://raw.githubusercontent.com/igukusamuel/DATA-607-Project-2/master/UN_Migrant

head(migrants) #Print out first few rows to confirm that the data have been loaded correctly.
```

```
## # A tibble: 6 x 26
##      X1 X2    X3       X4 X5    `1990` `1995` `2000` `2005` `2010` `2015`
##   <dbl> <chr> <chr> <dbl> <chr> <chr>  <chr>  <chr>  <chr>  <chr>  <chr> 
## 1     1 WORLD <NA>    900 <NA>  153,0~ 161,3~ 173,5~ 191,6~ 220,7~ 248,8~
## 2     2 UN d~ <NA>     NA <NA>  ..     ..     ..     ..     ..     ..    
## 3     3 More~ b       901 <NA>  82,76~ 92,93~ 103,9~ 116,6~ 130,6~ 140,6~
## 4     4 Less~ c       902 <NA>  70,24~ 68,38~ 69,62~ 74,92~ 90,16~ 108,2~
## 5     5 Leas~ d       941 <NA>  11,06~ 11,68~ 10,06~ 9,833~ 10,43~ 13,63~
## 6     6 Less~ <NA>    934 <NA>  59,18~ 56,70~ 59,56~ 65,09~ 79,73~ 94,58~
## # ... with 15 more variables: `2019` <chr>, `1990_1` <chr>,
## #   `1995_1` <chr>, `2000_1` <chr>, `2005_1` <chr>, `2010_1` <chr>,
## #   `2015_1` <chr>, `2019_1` <chr>, `1990_2` <chr>, `1995_2` <chr>,
## #   `2000_2` <chr>, `2005_2` <chr>, `2010_2` <chr>, `2015_2` <chr>,
## #   `2019_2` <chr>
```

Filter for N/As in column X5

Careful review of the data shows that column named X5 only includes data for rows related to countries and N/A's for rows relating to regions and regional totals. Thus filtering out all N/As in column X5 will leave us with country data only, which is the basis of out analysis. We first view all the N/As under column X5 to confirm none of them relate to country information.

```
colX5 <- filter(migrants, is.na(X5))

x <- length(colX5)
x
```

```
## [1] 26
```

```
head(colX5)
```

```
## # A tibble: 6 x 26
##      X1 X2    X3       X4 X5    `1990` `1995` `2000` `2005` `2010` `2015`
##   <dbl> <chr> <chr> <dbl> <chr> <chr>  <chr>  <chr>  <chr>  <chr>  <chr>
## 1     1 WORLD <NA>    900 <NA>  153,0~ 161,3~ 173,5~ 191,6~ 220,7~ 248,8~
## 2     2 UN d~ <NA>     NA <NA>  ..     ..     ..     ..     ..     ..
## 3     3 More~ b       901 <NA>  82,76~ 92,93~ 103,9~ 116,6~ 130,6~ 140,6~
## 4     4 Less~ c       902 <NA>  70,24~ 68,38~ 69,62~ 74,92~ 90,16~ 108,2~
## 5     5 Leas~ d       941 <NA>  11,06~ 11,68~ 10,06~ 9,833~ 10,43~ 13,63~
## 6     6 Less~ <NA>    934 <NA>  59,18~ 56,70~ 59,56~ 65,09~ 79,73~ 94,58~
## # ... with 15 more variables: `2019` <chr>, `1990_1` <chr>,
## #   `1995_1` <chr>, `2000_1` <chr>, `2005_1` <chr>, `2010_1` <chr>,
## #   `2015_1` <chr>, `2019_1` <chr>, `1990_2` <chr>, `1995_2` <chr>,
## #   `2000_2` <chr>, `2005_2` <chr>, `2010_2` <chr>, `2015_2` <chr>,
## #   `2019_2` <chr>
```

Exclude N/As in column X5

We then exclude all N/A's in column X6 and print out the first 6 rows using the head() function.

```
migrants_by_country <- filter(migrants, !is.na(X5))

head(migrants_by_country)
```

```
## # A tibble: 6 x 26
##      X1 X2    X3       X4 X5    `1990` `1995` `2000` `2005` `2010` `2015`
##   <dbl> <chr> <chr> <dbl> <chr> <chr>  <chr>  <chr>  <chr>  <chr>  <chr>
## 1    24 Buru~ <NA>    108 B R   333,1~ 254,8~ 125,6~ 172,8~ 235,2~ 289,8~
## 2    25 Como~ <NA>    174 B     14,079 13,939 13,799 13,209 12,618 12,555
## 3    26 Djib~ <NA>    262 B R   122,2~ 99,774 100,5~ 92,091 101,5~ 112,3~
## 4    27 Erit~ <NA>    232 I     11,848 12,400 12,952 14,314 15,676 15,941
## 5    28 Ethi~ <NA>    231 B R   1,155~ 806,9~ 611,3~ 514,2~ 568,7~ 1,161~
## 6    29 Kenya <NA>    404 B R   298,0~ 618,7~ 707,8~ 773,3~ 954,9~ 1,126~
## # ... with 15 more variables: `2019` <chr>, `1990_1` <chr>,
## #   `1995_1` <chr>, `2000_1` <chr>, `2005_1` <chr>, `2010_1` <chr>,
## #   `2015_1` <chr>, `2019_1` <chr>, `1990_2` <chr>, `1995_2` <chr>,
## #   `2000_2` <chr>, `2005_2` <chr>, `2010_2` <chr>, `2015_2` <chr>,
## #   `2019_2` <chr>
```

Rename column X2

From the above print out, there is need to rename column X2 dest_country.

```
migrants_by_country <- migrants_by_country %>%
        rename(
                dest_country = X2
        )
head(migrants_by_country)
```

```
## # A tibble: 6 x 26
##      X1 dest_country X3       X4 X5    `1990` `1995` `2000` `2005` `2010`
##   <dbl> <chr>        <chr> <dbl> <chr> <chr>  <chr>  <chr>  <chr>  <chr> 
## 1    24 Burundi      <NA>    108 B R   333,1~ 254,8~ 125,6~ 172,8~ 235,2~
## 2    25 Comoros      <NA>    174 B     14,079 13,939 13,799 13,209 12,618
## 3    26 Djibouti     <NA>    262 B R   122,2~ 99,774 100,5~ 92,091 101,5~
## 4    27 Eritrea      <NA>    232 I     11,848 12,400 12,952 14,314 15,676
## 5    28 Ethiopia     <NA>    231 B R   1,155~ 806,9~ 611,3~ 514,2~ 568,7~
## 6    29 Kenya        <NA>    404 B R   298,0~ 618,7~ 707,8~ 773,3~ 954,9~
## # ... with 16 more variables: `2015` <chr>, `2019` <chr>, `1990_1` <chr>,
## #   `1995_1` <chr>, `2000_1` <chr>, `2005_1` <chr>, `2010_1` <chr>,
## #   `2015_1` <chr>, `2019_1` <chr>, `1990_2` <chr>, `1995_2` <chr>,
## #   `2000_2` <chr>, `2005_2` <chr>, `2010_2` <chr>, `2015_2` <chr>,
## #   `2019_2` <chr>
```

View all columns

The above printout shows a number of irrelevant columns that are not necessary for our analysis. Lets print out the entire column names and delete the unnecessary ones to have a cleaner data set.

```
column_names <- colnames(migrants_by_country)
#column_names # umcomment to view entire list of column names
head(column_names)
```

```
## [1] "X1"           "dest_country" "X3"           "X4"          
## [5] "X5"           "1990"
```

Exclude irrelevant columns

The above print out reveals that we do not need all column names that start with "X". We delete these columns using the srtarts_with function.

```
migrants_by_country <- migrants_by_country %>%
        select(-starts_with("X"))


migrants_by_country <- migrants_by_country %>%
        select(-c(2:8))

migrants_by_country
```

```
## # A tibble: 232 x 15
##    dest_country `1990_1` `1995_1` `2000_1` `2005_1` `2010_1` `2015_1`
##    <chr>        <chr>    <chr>    <chr>    <chr>    <chr>    <chr>
##  1 Burundi      163,267  124,165  61,094   84,805   115,823  142,790
##  2 Comoros      6,717    6,614    6,511    6,286    6,060    6,071
##  3 Djibouti     64,242   52,476   52,920   51,315   53,295   59,081
##  4 Eritrea      6,228    6,542    6,856    7,729    8,603    8,833
##  5 Ethiopia     607,284  424,117  322,219  269,725  298,069  591,409
##  6 Kenya        161,259  322,189  352,933  400,364  473,093  562,909
##  7 Madagascar   13,348   11,901   13,276   14,744   16,410   18,270
##  8 Malawi       546,520  116,198  111,530  105,931  103,869  110,893
##  9 Mauritius    1,763    3,228    5,705    8,943    13,188   15,832
## 10 Mayotte      8,780    14,679   23,546   31,364   34,500   34,235
## # ... with 222 more rows, and 8 more variables: `2019_1` <chr>,
## #   `1990_2` <chr>, `1995_2` <chr>, `2000_2` <chr>, `2005_2` <chr>,
## #   `2010_2` <chr>, `2015_2` <chr>, `2019_2` <chr>
```

View dimentions of resulting data frame

We use dim() function to have an idea of how many rows and columns we have for our analysis.

```
dim(migrants_by_country)
```

```
## [1] 232  15
```

Confrim column names.

This is what we need for our analysis.

```
column_names_clean <- colnames(migrants_by_country)
#column_names_clean # uncomment to view entire list of cleaned up column names
head(column_names_clean)
```

```
## [1] "dest_country" "1990_1"       "1995_1"       "2000_1"
## [5] "2005_1"       "2010_1"
```

View number of columns

Get the length of the column names to be used in the next line of code.

```
y <- length(colnames(migrants_by_country))

y
```

```
## [1] 15
```

clean up data

Let us use gather() function to gather all columns with years into a single columns and exclude any and all N/As to obtain clean data. Spread the resulting data by year column and rename "1" as male and "2" as female.

```
no_of_migrants_per_country <- mutate(gather(migrants_by_country, "year", "no_of_migrants", 2:y, na.rm =

head(no_of_migrants_per_country)
```

```
## # A tibble: 6 x 3
##   dest_country year   no_of_migrants
##   <chr>        <chr>  <chr>
## 1 Burundi      1990_1 163,267
## 2 Comoros      1990_1 6,717
## 3 Djibouti     1990_1 64,242
## 4 Eritrea      1990_1 6,228
## 5 Ethiopia     1990_1 607,284
## 6 Kenya        1990_1 161,259
```

```
no_of_migrants_per_country <- no_of_migrants_per_country %>%
        separate(year, c("year", "sex"), sep = "_")

no_of_migrants_per_country
```

```
## # A tibble: 3,248 x 4
##    dest_country year  sex   no_of_migrants
##    <chr>        <chr> <chr> <chr>
##  1 Burundi      1990  1     163,267
##  2 Comoros      1990  1     6,717
##  3 Djibouti     1990  1     64,242
##  4 Eritrea      1990  1     6,228
##  5 Ethiopia     1990  1     607,284
##  6 Kenya        1990  1     161,259
##  7 Madagascar   1990  1     13,348
##  8 Malawi       1990  1     546,520
##  9 Mauritius    1990  1     1,763
## 10 Mayotte      1990  1     8,780
## # ... with 3,238 more rows
```

Convert the years column to number format

```
no_of_migrants_per_country$year <- parse_number(no_of_migrants_per_country$year)

no_of_migrants_per_country
```

```
## # A tibble: 3,248 x 4
##    dest_country  year sex   no_of_migrants
##    <chr>        <dbl> <chr> <chr>
##  1 Burundi       1990 1     163,267
##  2 Comoros       1990 1     6,717
##  3 Djibouti      1990 1     64,242
##   4 Eritrea       1990 1     6,228
##   5 Ethiopia      1990 1     607,284
##   6 Kenya         1990 1     161,259
##   7 Madagascar    1990 1     13,348
##   8 Malawi        1990 1     546,520
##   9 Mauritius     1990 1     1,763
##  10 Mayotte       1990 1     8,780
## # ... with 3,238 more rows
```

```
no_of_migrants_per_country <- no_of_migrants_per_country %>%
        spread(sex, no_of_migrants)


names(no_of_migrants_per_country)
```

```
## [1] "dest_country" "year"         "1"             "2"
```

```
no_of_migrants_per_country <- no_of_migrants_per_country %>%
        rename(
                male = "1",
                female = "2"
        )
head(no_of_migrants_per_country)
```

```
## # A tibble: 6 x 4
##   dest_country  year male    female
##   <chr>        <dbl> <chr>   <chr>
## 1 Afghanistan   1990 32,558  25,128
## 2 Afghanistan   1995 39,105  32,417
## 3 Afghanistan   2000 42,848  33,069
## 4 Afghanistan   2005 49,274  38,026
## 5 Afghanistan   2010 57,709  44,537
## 6 Afghanistan   2015 248,212 241,537
```

Conversion of chr to dbl

convert the no_of_migrants data column from characters to doubles for statistical analysis. This we will do using the parse_number() function. Print out using head() function the first 6 rows and confirm this conversion.

```
no_of_migrants_per_country$male <- parse_number(no_of_migrants_per_country$male)
no_of_migrants_per_country$female <- parse_number(no_of_migrants_per_country$female)

clean_migrants_data <- no_of_migrants_per_country

head(clean_migrants_data)
```

```
## # A tibble: 6 x 4
##   dest_country  year   male female
##   <chr>        <dbl>  <dbl>  <dbl>
## 1 Afghanistan   1990  32558  25128
## 2 Afghanistan   1995  39105  32417
## 3 Afghanistan   2000  42848  33069
## 4 Afghanistan   2005  49274  38026
## 5 Afghanistan   2010  57709  44537
## 6 Afghanistan   2015 248212 241537
```

Down stream analysis

Ordering of data

Ordering data by country with largest inflow of migrants

```
by_country <- clean_migrants_data %>%
        group_by(year, dest_country) %>%
        summarise(total_migrants = male + female) %>%
        arrange(desc(total_migrants))
head(by_country)
```

```
## # A tibble: 6 x 3
## # Groups:   year [6]
##    year dest_country             total_migrants
##   <dbl> <chr>                             <dbl>
## 1  2019 United States of America       50661149
## 2  2015 United States of America       48178877
## 3  2010 United States of America       44183643
## 4  2005 United States of America       39258293
## 5  2000 United States of America       34814053
## 6  1995 United States of America       28451053
```

Ordering the data of male migrants by the destination countries by year

```
male_by_country <- clean_migrants_data %>%
        group_by(dest_country, year) %>%
        summarise(male = male) %>%
        arrange(desc(male))
head(male_by_country)
```

```
## # A tibble: 6 x 3
## # Groups:   dest_country [1]
##   dest_country              year     male
##   <chr>                    <dbl>    <dbl>
## 1 United States of America  2019 24488382
## 2 United States of America  2015 23446873
## 3 United States of America  2010 21694169
## 4 United States of America  2005 19614878
## 5 United States of America  2000 17310785
## 6 United States of America  1995 14032159
```

Ordering the data of female migrants by the destination countries by year

```
female_by_country <- clean_migrants_data %>%
        group_by(dest_country, year) %>%
        summarise(female = female) %>%
        arrange(desc(female))
head(female_by_country)
```

```
## # A tibble: 6 x 3
## # Groups:   dest_country [1]
##   dest_country              year   female
##   <chr>                    <dbl>    <dbl>
## 1 United States of America  2019 26172767
## 2 United States of America  2015 24732004
## 3 United States of America  2010 22489474
## 4 United States of America  2005 19643415
## 5 United States of America  2000 17503268
## 6 United States of America  1995 14418894
```

Ordering the data by % of male migrants by the destination countries by year

```
Perc_male_by_country <- clean_migrants_data %>%
        group_by(dest_country, year) %>%
        summarise(perc_male = male/(male + female)) %>%
        arrange(desc(perc_male))
head(Perc_male_by_country)
```

```
## # A tibble: 6 x 3
## # Groups:   dest_country [4]
##   dest_country  year perc_male
##   <chr>        <dbl>     <dbl>
## 1 Maldives      2019     0.877
## 2 Maldives      2015     0.877
## 3 Bhutan        2019     0.849
## 4 Bhutan        2015     0.849
## 5 Qatar         2015     0.839
## 6 Oman          2019     0.836
```

Ordering the data by % female migrants by the destination countries by year

```
Perc_female_by_country <- clean_migrants_data %>%
        group_by(dest_country, year) %>%
        summarise(perc_female = female/(male + female)) %>%
        arrange(desc(perc_female))
head(Perc_female_by_country)
```

```
## # A tibble: 6 x 3
## # Groups:   dest_country [1]
##   dest_country  year perc_female
##   <chr>        <dbl>       <dbl>
## 1 Nepal         1990       0.707
## 2 Nepal         2019       0.697
## 3 Nepal         2015       0.693
## 4 Nepal         1995       0.685
## 5 Nepal         2010       0.672
## 6 Nepal         2000       0.663
```

Conclusion

Maldives received the highest % of male migrants while nepal received the highest % of female migrants