Data Wrangling with tidyr

Open RStudio and create a new project under your Module 5 folder and call it **Mod5Assignment1.** For this assignment, you will be creating an R Markdown document that will include topics previously covered as well as the use of tidyr to create tidy data that can be used to generate various visuals. Once completed, all you need to do is submit the word document that is created.

Create the R Markdown Document

1.) In RStudio, select *File -> New File -> Text File*. This will create a blank text file in the same area that scripts were created in previous assignments (upper left panel). Save this file to your project as **Mod6Assign1Answer.rmd** (it is important to save with the .rmd extension as this saves the text file as an R Markdown file).

2.) Create a Header 1 with the title: **Module 5 - Assignment 1**

3.) Create a Header 2 with the title: **Last Name, First Name** (replace with your name)

4.) Create a Header 3 with the title: **Data Wrangling**

5.) Click on the dropdown arrow next to the Knit icon Knit ▾ at the top of the R Markdown Pane in RStudio and select Knit to Word.

6.) Notice that you now have a document in your files for the project named **Mod5Assign1Answer.docx**. This is the file you will be uploading later to Canvas.

7.) For this assignment, you will need to download the **UN_migrant.xlsx** file from Canvas. Save this in the same folder that you created the project in.

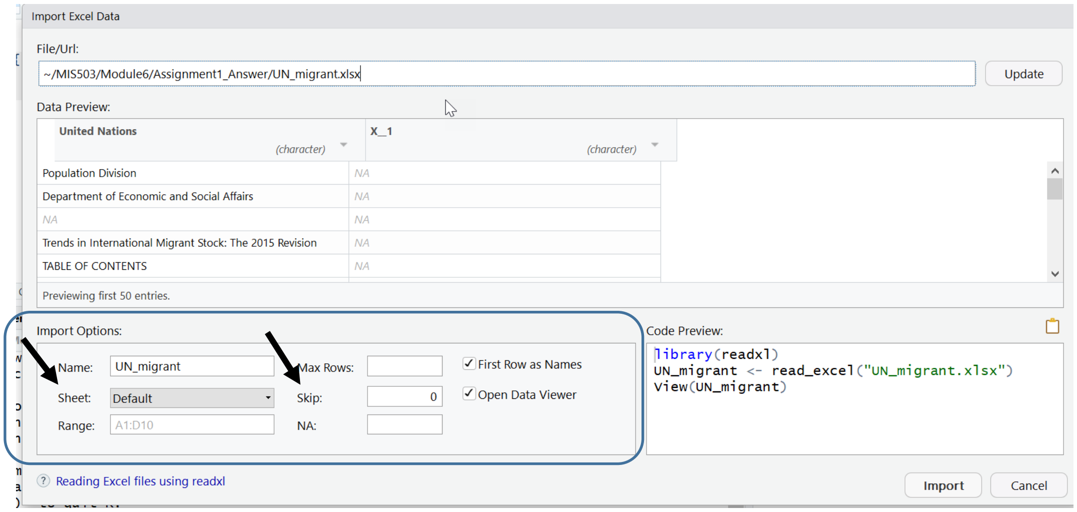
Part 1: Importing the dataset using readxl

8.) Within R Studio, create a new chunk of R code in your R Markdown document. Include the code to load the *tidyverse* as well as the *readxl* package since we will be importing an excel file.

9.) Next, you will want to copy the following code to import the file:

```
UN_migrant <- read_excel("UN_migrant.xlsx",
            sheet = "Table 6", col_types = c("numeric",
            "text", "text", "numeric", "text",
            "numeric", "numeric", "numeric",
            "numeric", "numeric", "numeric","text",
            "text","text","text","text","text","text",
            "text","text","text","text"), skip = 15)
```

10.) You may notice that this is different from the csv files we have been importing up to this point. Specifically, we are naming the Excel sheet to use and as well as skipping the first 15 lines. Below is how to change these if you were to click on the Excel file and select Import in the Import Dataset…



File/Url: ~/MIS503/Module6/Assignment1_Answer/UN_migrant.xlsx

Data Preview:

United Nations (character)	X_1 (character)
Population Division	NA
Department of Economic and Social Affairs	NA
NA	NA
Trends in International Migrant Stock: The 2015 Revision	NA
TABLE OF CONTENTS	NA

Previewing first 50 entries.

Import Options:
Name: UN_migrant
Sheet: Default
Range: A1:D10
Max Rows:
Skip: 0
NA:
☑ First Row as Names
☑ Open Data Viewer

Code Preview:

```
library(readxl)
UN_migrant <- read_excel("UN_migrant.xlsx")
View(UN_migrant)
```

Part 2 – Cleaning Data with dplyr

11.) Now that we have imported the dataset, let's start to clean up the newly created tibble. Examining the tibble in the Global Environment panel, we can see that there are 265 observations and 22 variables. This is showing migration statistics across the world.

12.) Create a new Header 3 with the title: **Part 2 – Cleaning Data with dplyr**

13.) Create a new chunk of R code. We need to change some of the variable to represent the data correctly. Add the code to the new chunk to rename the following columns within the **UN_migrant** in the tibble (remember you will have to start with *UN_migrant* <- so it will overwrite the column name in the tibble):

- Rename …2 to Country

- Rename …4 to Country_Code
- Rename …5 to Type
- Rename 1990…6 to 1990
- Rename 1995…7 to 1995
- Rename 2000…8 to 2000
- Rename 2005…9 to 2005
- Rename 2010…10 to 2010
- Rename 2015…11 to 2015

Note: To rename the year columns above, you will need to use quotes similar to the code:

UN_migrant <- rename(UN_migrant, "1990" = "1990...6")

14.) Using dplyr, we want to create a new dataset that will only use some of the columns within the tibble. Write the code to create a new tibble called **Migration** that will include the following variables:

- Country
- Country_Code
- Type
- "1990"
- "1995"
- "2000"
- "2005"
- "2010"
- "2015"

NOTE: to select the year columns above, you will need to include them in quotes when writing the code. If you don't, R will start looking for a numeric value in the function, not a column/variable.

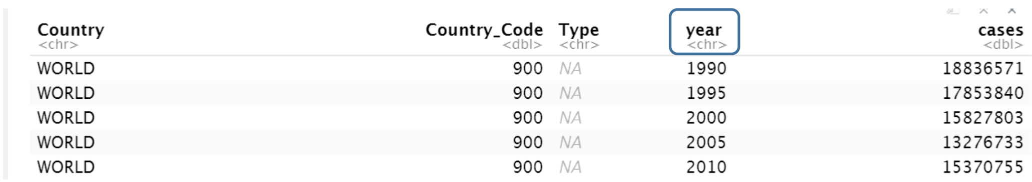
15.) You should have a new tibble called Migration that has 265 observations and 9 variables.

Part 3 – Creating tidy data using tidyr

16.) Now that we have the data imported and have the subset we want to work with, we need to check to make sure the data is tidy. Remember, to be tidy data each row must represent a different observation but right now, rows actually represent multiple observations across a number of years.

17.) Create a Header 3 with the title: **Part 3 – Creating tidy data using tidyr**

18.) Create a new chunk of R code. Using tidyr, use pivoting to create tidy data from the year columns. You will need to create a new tibble called **Migration2** and use pivoting to create a new column (or name) being year and the value being cases (see the online chapter on tidyr in R for Data Science as a reference, if needed). Include the head command in your code so you can view the first 6 rows. Once this has been done, your Migration2 tibble should look similar to the one below.



Country <chr>	Country_Code <dbl>	Type <chr>	year <chr>	cases <dbl>
WORLD	900	NA	1990	18836571
WORLD	900	NA	1995	17853840
WORLD	900	NA	2000	15827803
WORLD	900	NA	2005	13276733
WORLD	900	NA	2010	15370755

19.) Notice that we now have two columns (year and cases) but also take note that year is showing up as characters not numbers. Within the R Markdown document, write the code to change this variable from character to numeric (hint: this was done in one of the videos for this module).

Part 4 – Visualizing your data

20.) From this data, we would like to know about migration in different regions and countries around the world.

21.) Create a Header 3 with the title: **Part 4 – Research Questions**

22.) Let's create 2 subsets of data to understand migration trends around the world. For these tibbles, you will be creating a subset of just a few variables. Here is an article that will help on filtering with multiple values (https://blog.exploratory.io/filter-data-with-dplyr-76cf5f1a258e). In a new chunk of code, create the two subsets below:

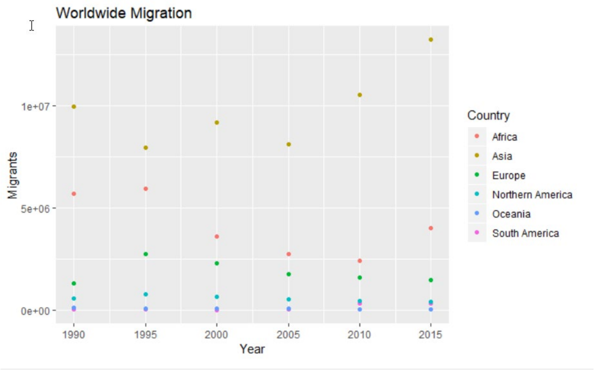
a. **RegionalMigration** – this subset should include all the variables from **Migration2** for the following regions in the Country column: Africa, Asia, Europe, Oceania, Northern America and South America.
b. **Americas -** this subset should include all the variables from **Migration2** for the following regions in the Country column: Central America, South America and Northern America a

23.) Next, within your R Markdown document create a new Header 3 titled: **Worldwide Migration based on Regions** and write out the following questions in the document:

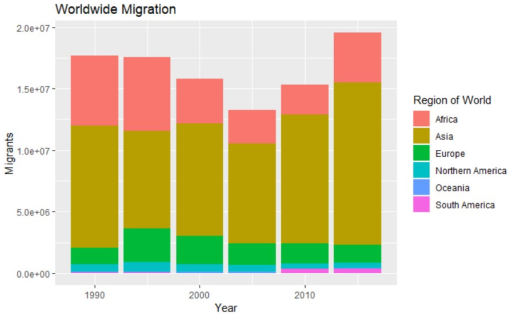
a. Which region in the world had the highest number of migrants in the year 2005?
b. Over the years, which region consistently has the most migrants every 5 year span? Which has the second most?
c. What region has seen the fewest migrants over the years?
d. Which plot was most useful in answering these questions and why?

24.) To answer these questions, create the following two plots in your R Markdown document:

a. A scatterplot with the year on the x-axis and cases on the y-axis with country being represented on the plot as a different color (also have the same labels/titles on your scatterplot as below):



b. Instead of using *geom_bar()*, you can create a bar graph of totals (this is because we already have all the cases totaled by region) using the *geom_col()*. The code is similar to the previous scatterplot but as you create your aesthetic statement, you will need to use “fill=” in place of “color=” to place the regions in color on the bar chart. You will also notice I change the title in the legend. This can be added to the original ggplot function statement by including scale_fill_discrete(name=”Region of World”*)*.



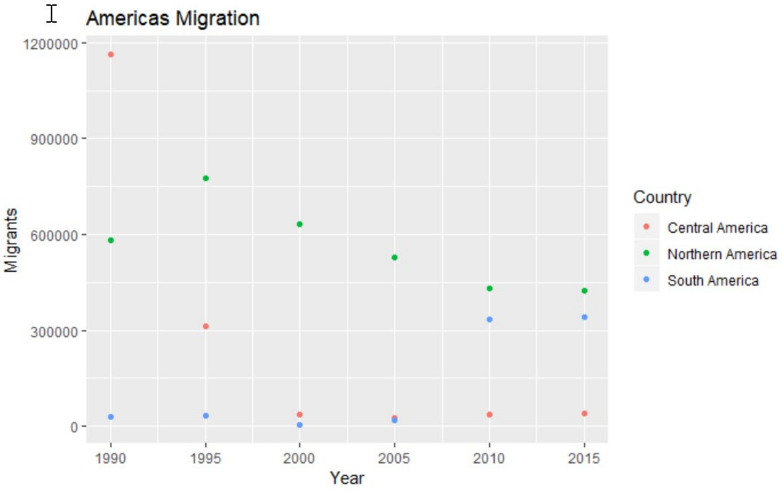
25.)Go back to the previous questions in the document (see step 27) and answer them based on the results from your graphs.

26.)Next, create a new Header 3 with the title: **Americas Migration by Region** and then add the following questions:

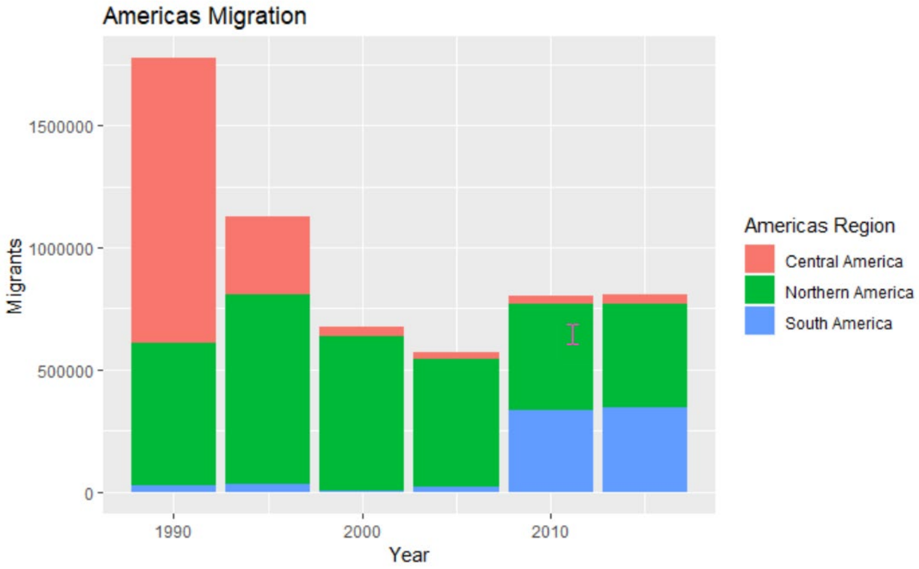
a. In 1990, which region had the largest number of migrants for the Americas?
b. Has this region continued to grow since 1990?
c. What trends do you notice happening in the Americas over the years?
d. Specifically, has Northern America increased or decreased over the years?
e. Which plot was most useful in answering these questions and why?

27.)To answer these questions, create the following two plots in your R Markdown document:

a. Scatterplot similar to the one below using similar syntax as before. Your plot should look like this:



b. Create a bar graph with geom_col( ) using similar syntax as you used in the previous analysis. Your plot should look like this:



28.) Go back to the previous questions in the document (step 30) and answer them based on the results from your graphs.

29.) Finally, knit your R Markdown to Word and upload the document to Canvas