Turkey Migration Map

rboys

22 December 2018

1- Libraries Used

```
library(tidyverse)
library(reshape2)
library(lubridate)
library(ggplot2)
library(scales)
library(sp)
library(gridExtra)
library(igraph)
library(ggraph)
library(tidygraph)
library(readxl)
```

2- Source Files For The Analysis

We are going to use TUIK data to map domestic migration in Turkey. Original excel file can be found in this link.

2.1- Description Of The Data Set

Data set we obtained from the TUIK website contains 4 years of population movement within Turkey. Original excel file consists of 329 rows and 86 columns of data. Each row represents the destination cities which people migrated to and on each column we can obtain the distribution of migrated population by the place of birth. There are 81 distinct Turkish cities as a destination and 83 distinct columns as a place of birth with the addition for people who were born abroad or unknown locations.

3- Objectives

- Primary goal of this analysis is to find & address the specific patterns that shape the migration dynamics in Turkey.
- First we will analyze the data how the migration figures evolved in four years and then visualize the common patterns observed along this period.
- Finally we will try to link additional demographic statistics from TUIK and show that what would be the population density across cities if people could migrate to place where they were born instead of where they migrated actually.

4- Abstract

Human migration is the movement of people from one province to another with the intentions of settling down, permanently or temporarily in a new location. The movement might be often over long distances and from one country to another, but internal migration is also possible; indeed, this is the dominant form globally. If migration takes place within the borders of a country, then it is called internal migration; if it crosses a country's border, it is called external or international migration. Migration can occur owing to social, economic, political, cultural, and ethnic reasons. Internal migration began with modernization in agriculture and industrialization activities after the Second World War in Turkey. Migration first occured from village to city, then from small- and medium-sized cities to large cities. In the 1990s, a new form of

migration emerged: from cities to villages. Internal migration has caused social changes in both city and village settlements. Along with these changes, a number of problems emerged, especially in those of cities, and these problems exist till date.This study analyzed internal migration addresses from a city to another city in Turkey between 2014 and 2017.

5- Data Cleaning & Pre-processing

We will go through the steps for reading, reshaping, organizing and finally saving a RDS file in order to have a clean and multi purpose data set.

5.1- Reading The Data

We will start by reading raw data from the excel file and get rid of some unnecessary lines containing footnotes.

```
tmp1<-tempfile(fileext=".xls")
download.file("https://github.com/MEF-BDA503/gpj18-r_boys/blob/master/source_files/domestic_migration_in
maindata<- read_excel(tmp1,skip=3) %>%
  slice(-c(329:333))
```

First we check the structure and head of data in order to get an insight about our dataset.

```
head(maindata)
```

```
## # A tibble: 6 x 86
##   Yil   Alan  Toplam Adana `Adiyaman` Afyonkarahisar `Agri` Amasya Ankara
##   <chr> <chr>  <dbl> <dbl>      <dbl>          <dbl>  <dbl>  <dbl>  <dbl>
## 1 2017  Topl~ 2.68e6 78422      34453          28264  40965  18889 121902
## 2 2017  Adana 4.95e4 21992        785             82    184     61    827
## 3 2017  Adiy~ 1.80e4   460       9448             21     57     12    208
## 4 2017  Afyo~ 2.15e4   271        100           7301    143     68    943
## 5 2017  Agri  1.51e4   183        115            112   6201     63    427
## 6 2017  Amas~ 1.43e4   135         72             34     95   5079    565
## # ... with 77 more variables: Antalya <dbl>, Artvin <dbl>, `Aydin` <dbl>,
## #   `Balikesir` <dbl>, Bilecik <dbl>, Bingöl <dbl>, Bitlis <dbl>,
## #   Bolu <dbl>, Burdur <dbl>, Bursa <dbl>, Çanakkale <dbl>,
## #   `Çankiri` <dbl>, Çorum <dbl>, Denizli <dbl>, `Diyarbakir` <dbl>,
## #   Edirne <dbl>, `Elazig` <dbl>, Erzincan <dbl>, Erzurum <dbl>,
## #   `Eskisehir` <dbl>, Gaziantep <dbl>, Giresun <dbl>, `Gümüshane` <dbl>,
## #   Hakkari <dbl>, Hatay <dbl>, Isparta <dbl>, Mersin <dbl>,
## #   `Istanbul` <dbl>, `Izmir` <dbl>, Kars <dbl>, Kastamonu <dbl>,
## #   Kayseri <dbl>, `Kirklareli` <dbl>, `Kirsehir` <dbl>, Kocaeli <dbl>,
## #   Konya <dbl>, Kütahya <dbl>, Malatya <dbl>, Manisa <dbl>,
## #   `Kahramanmaras` <dbl>, Mardin <dbl>, `Mugla` <dbl>, `Mus` <dbl>,
## #   `Nevsehir` <dbl>, `Nigde` <dbl>, Ordu <dbl>, Rize <dbl>,
## #   Sakarya <dbl>, Samsun <dbl>, Siirt <dbl>, Sinop <dbl>, Sivas <dbl>,
## #   `Tekirdag` <dbl>, Tokat <dbl>, Trabzon <dbl>, Tunceli <dbl>,
## #   `Sanliurfa` <dbl>, `Usak` <dbl>, Van <dbl>, Yozgat <dbl>,
## #   Zonguldak <dbl>, Aksaray <dbl>, Bayburt <dbl>, Karaman <dbl>,
## #   `Kirikkale` <dbl>, Batman <dbl>, `Sirnak` <dbl>, `Bartin` <dbl>,
## #   Ardahan <dbl>, `Igdir` <dbl>, Yalova <dbl>, Karabük <dbl>,
## #   Kilis <dbl>, Osmaniye <dbl>, Düzce <dbl>, `Yurt disi\nAbroad` <dbl>,
## #   `Bilinmeyen\nUnknown` <dbl>
```

5.2- Reshaping The Data

Since the file we have is structured in a horizontal way it is better to transform it to a vertical format by using melt function in reshape2 library.

```r
# First three columns should remain as it is while rest melted
melted_data <- melt(maindata, id.vars = c("Yil", "Alan", "Toplam"))
# This will be the main structure
head(melted_data)
```

```
##    Yil           Alan  Toplam variable value
## 1 2017   Toplam-Total 2684820    Adana 78422
## 2 2017          Adana   49509    Adana 21992
## 3 2017       Adiyaman   18040    Adana   460
## 4 2017 Afyonkarahisar   21453    Adana   271
## 5 2017           Agri   15088    Adana   183
## 6 2017         Amasya   14326    Adana   135
```

5.3- Organizing & Preprocessing Data

There are some additional information that we will not use for our analysis in the current data frame. Thus, we will remove them and also do some renaming and formatting.

```r
clean_data <- melted_data %>%
# We would not need total lines that can be calculated easily if necessary
  filter(Alan != "Toplam-Total") %>%
  select(Yil, Alan, variable, value) %>%
  arrange(Yil, Alan, variable, value)
# In order to use same language we assign new names to the columns
colnames(clean_data) <- c("Year", "Destination", "Birth_Place", "People")
clean_data$Year <- as.integer(clean_data$Year)
suppressWarnings(clean_data$`People` <- as.integer(clean_data$`People`))
clean_data$Birth_Place <- as.character(clean_data$Birth_Place)
clean_data$Destination <- enc2native(clean_data$Destination)
clean_data$Birth_Place <- enc2native(clean_data$Birth_Place)
# This should be the representation of our data set
head(clean_data)
```

```
##   Year Destination    Birth_Place People
## 1 2014       Adana          Adana  20638
## 2 2014       Adana       Adiyaman   1009
## 3 2014       Adana Afyonkarahisar     84
## 4 2014       Adana           Agri    184
## 5 2014       Adana         Amasya     63
## 6 2014       Adana         Ankara    868
```

Year column is currently in integer format which can be seen above. However, in order to do more analysis on ggplot2 or other packages it is better to transform it to date format with lubridate.

```r
clean_data$Year <- ymd(sprintf("%d-01-01",clean_data$Year))
```

5.4- Translation from TR to EN

As the last part of data cleaning steps, we will translate characters from TR to EN by using below function

```r
mgsub <- function(pattern, replacement, x, ...) {
  n = length(pattern)
  if (n != length(replacement)) {
```

```
    stop("pattern and replacement do not have the same length.")
  }
  result = x
  for (i in 1:n) {
    result <- gsub(pattern[i],replacement[i],result)
  }
  return(result)
}

tr_to_en <- function(datafile){
  turkish_letters <- c("Ã ","S","G","I","Ã ","Ã ","Ã§","s","g","i","Ã¼","Ã¶")
  english_letters <- c("C","S","G","I","U","O","c","s","g","i","u","o")
  datafile <- mgsub(turkish_letters,english_letters,datafile)
  return(datafile)
}

clean_data$Birth_Place <- tr_to_en(clean_data$Birth_Place)
clean_data$Destination <- tr_to_en(clean_data$Destination)
head(clean_data)
```

```
##         Year Destination    Birth_Place People
## 1 2014-01-01       Adana          Adana  20638
## 2 2014-01-01       Adana       Adiyaman   1009
## 3 2014-01-01       Adana Afyonkarahisar     84
## 4 2014-01-01       Adana           Agri    184
## 5 2014-01-01       Adana         Amasya     63
## 6 2014-01-01       Adana         Ankara    868
```

5.5- Saving as RDS format

RDS file allows us to work with the same data later on without replicating above steps.

```
saveRDS(clean_data, file = "D:/Data Analytics Esentials/project/clean_migration.rds")
```

6- Exploratory Data Analysis (as of 2017)

We first list top 6 cities which preferred as a "Province of residence" and top 6 cities that migrated population have as a "place of birth". So we can compare the relation between each other.

Top 6 Cities Preffered for Migration

```
clean_data %>%
  filter(Year == "2017-01-01") %>%
  group_by(Destination) %>%
  summarize(sumofpeople = sum(People)) %>%
  top_n(n = 6, wt = sumofpeople) %>%
  arrange(desc(sumofpeople)) %>%
  head(n=6)
```

```
## # A tibble: 6 x 2
##   Destination sumofpeople
##   <chr>             <int>
## 1 Istanbul         416587
## 2 Ankara           188100
## 3 Izmir            127394
## 4 Kocaeli           87796
```

```
## 5 Antalya           87232
## 6 Bursa             86119
```

Top 6 Cities That Migrated People Have as a “Place of Birth”

```
clean_data %>%
  filter(Year == "2017-01-01") %>%
  group_by(Birth_Place) %>%
  summarize(sumofpeople = sum(People)) %>%
  top_n(n = 6, wt = sumofpeople) %>%
  arrange(desc(sumofpeople)) %>%
  head(n=6)
```

```
## # A tibble: 6 x 2
##   Birth_Place sumofpeople
##   <chr>             <int>
## 1 Istanbul         259999
## 2 Ankara           121902
## 3 Adana             78422
## 4 Izmir             74483
## 5 Sanliurfa         66673
## 6 Diyarbakir        65475
```

As you can notice from the two tables above, ranking of top 6 cities differ except Istanbul, Ankara and Izmir. We can make deduction that *Kocaeli, Antalya and Bursa* have higher possibility to be considered as a destination place to migrate. People who were born in *Adana, Sanliurfa and Diyarbakir* prefer to migrate to other cities rather than living in their hometowns. The reasons may be *economic* such as job opportunities, density of industrial activity, compulsory state service also it could be *social* such as terrorism or *environmental* such as climate.

We inferred that three biggest cities of Turkey; Istanbul, Ankara and Izmir have the same major role as migrated cities and people who migrated from. But what would be the real impact of the migration on population? If migration to these cities is higher than migrated from, then we should expect an increase in the population or vice versa.

```
top3to <- clean_data %>%
  filter(Destination %in% c("Istanbul", "Ankara", "Izmir")) %>%
  group_by(Year, Destination) %>%
  summarize(Migrated_to = sum(People))
colnames(top3to) <- c("Year", "City", "Migrated_to")

top3fr <- clean_data %>%
  filter(Birth_Place %in% c("Istanbul", "Ankara", "Izmir")) %>%
  group_by(Year, Birth_Place) %>%
  summarize(Migrated_from = sum(People))
colnames(top3fr) <- c("Year", "City", "Migrated_from")

top3 <- inner_join(top3to, top3fr, by =c("Year", "City")) %>%
  mutate(Impact = Migrated_to - Migrated_from)
head(top3,12)
```

```
## # A tibble: 12 x 5
## # Groups:   Year [4]
##    Year       City     Migrated_to Migrated_from Impact
##    <date>     <chr>          <int>         <int>  <int>
##  1 2014-01-01 Ankara        203621        123590  80031
##  2 2014-01-01 Istanbul      438998        227860 211138
```

```
##   3 2014-01-01 Izmir         124439         71524  52915
##   4 2015-01-01 Ankara        204048        122252  81796
##   5 2015-01-01 Istanbul      453407        239472 213935
##   6 2015-01-01 Izmir         126238         74313  51925
##   7 2016-01-01 Ankara        177166        119985  57181
##   8 2016-01-01 Istanbul      369582        249689 119893
##   9 2016-01-01 Izmir         122668         73603  49065
##  10 2017-01-01 Ankara        188100        121902  66198
##  11 2017-01-01 Istanbul      416587        259999 156588
##  12 2017-01-01 Izmir         127394         74483  52911
```

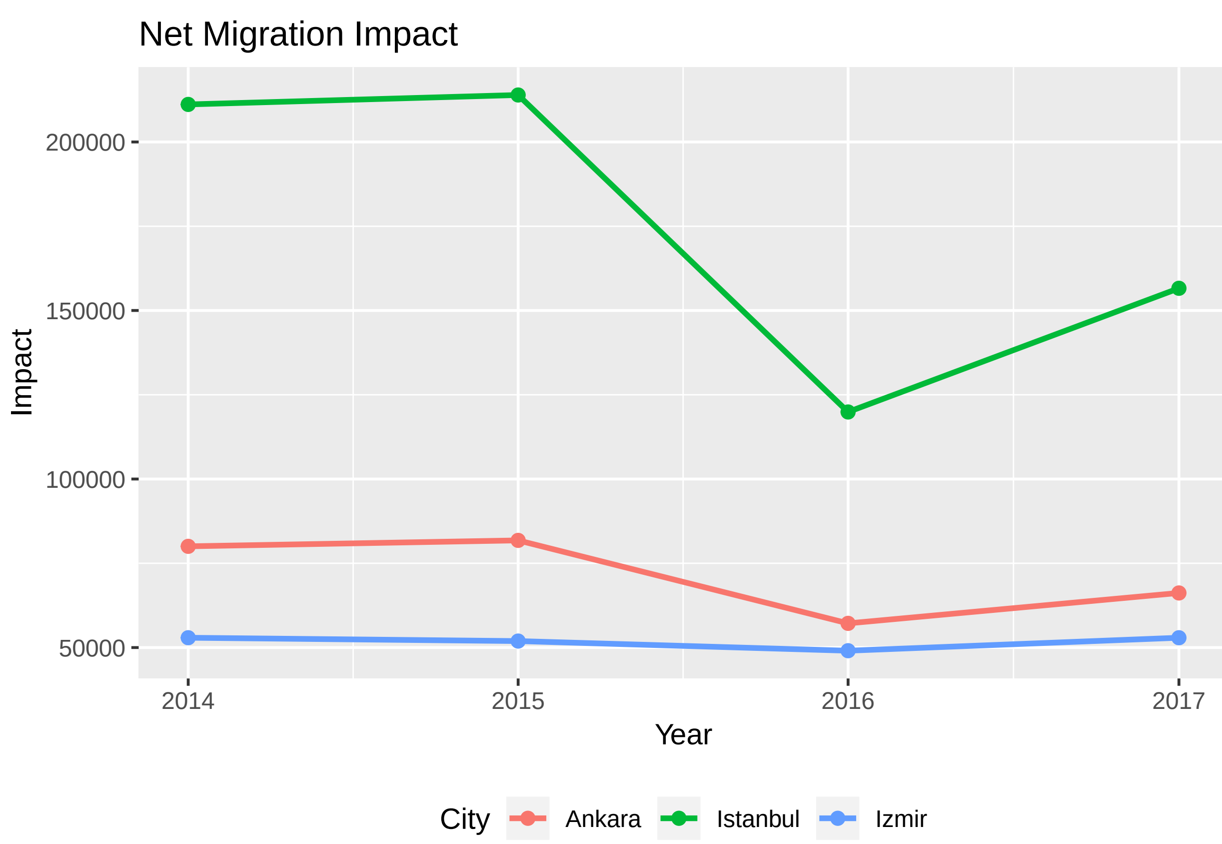
7- Findings

7.1- Three largest cities population growth

Below graph represents net annual impact of migration on three biggest Turkish cities. As shown, during the period between 2014 and 2017 net migration impact resulted in increasing population in these cities.

2015 was the peak of the population growth induced by migration for all three cities and 2016 was the lowest.

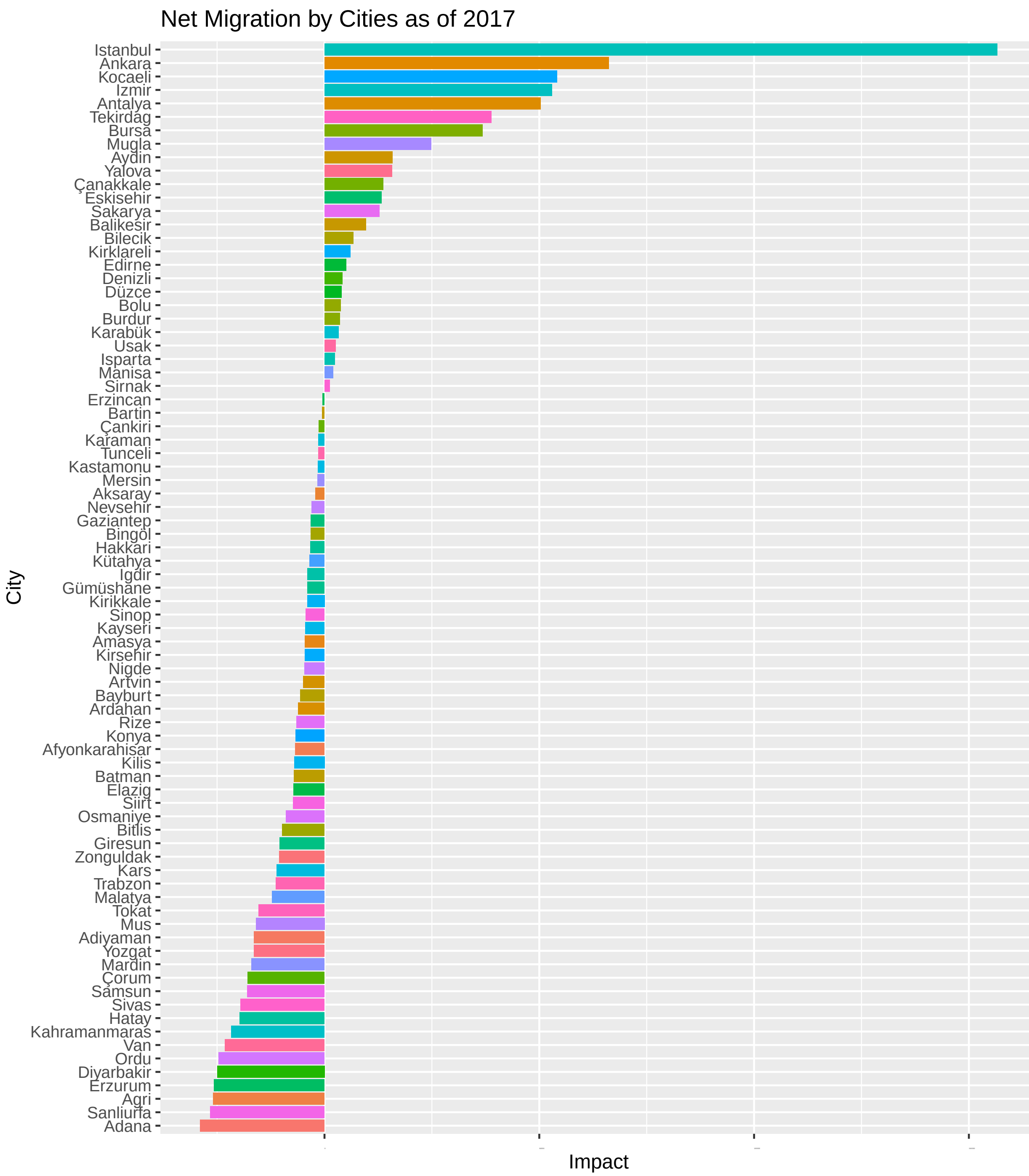
```r
ggplot(data = top3, aes(x = Year, y = Impact, color = City)) + geom_line(size = 1) +
  geom_point(size = 2) + ggtitle("Net Migration Impact") + theme(legend.position="bottom")
```



7.2- Net Migration Impact as of 2017 by All Cities

As shown below, 26 cities out of 81 have a position of net receiver. On the other hand remaining 55 cities' population declined due to migration. In this perspective we can assert that population concentrated on 32% of the Turkish cities in 2017 due to migration. Also it is visible in the graph that cities with net receiver status tend to concentrate on western and more industrialized part of the country. Without any other information we can assume majority of the migration in 2017 can be explained by economical motives.

```r
# increasing factor
inc <- clean_data %>%
  group_by(Year, Destination) %>%
  summarize(Migrated_to = sum(People))
colnames(inc) <- c("Year", "City", "Migrated_to")
# decrasing factor
dec <- clean_data %>%
  group_by(Year, Birth_Place) %>%
  summarize(Migrated_from = sum(People))
colnames(dec) <- c("Year", "City", "Migrated_from")
# net off
nets <- inner_join(inc, dec, by =c("Year", "City")) %>%
  mutate_if(is.numeric,funs(ifelse(is.na(.),0,.))) %>%
  mutate(Impact = Migrated_to - Migrated_from) %>%
  mutate(Pct = percent(Impact / sum(Migrated_to))) %>%
  arrange(desc(Impact))
# plot as of 2017
nets %>%
  filter(Year == "2017-01-01") %>%
  ggplot(aes(x=reorder(City, Impact), y=Impact, fill=City)) +
  geom_bar(stat = "identity") +
  coord_flip() +
  labs(x = "City", y = "Impact") +
  ggtitle("Net Migration by Cities as of 2017") +
  theme(legend.position = "none", axis.text.x = element_text(angle = 0.0, vjust = 0.0,
                                                              hjust = 0.0, size = 1))
```



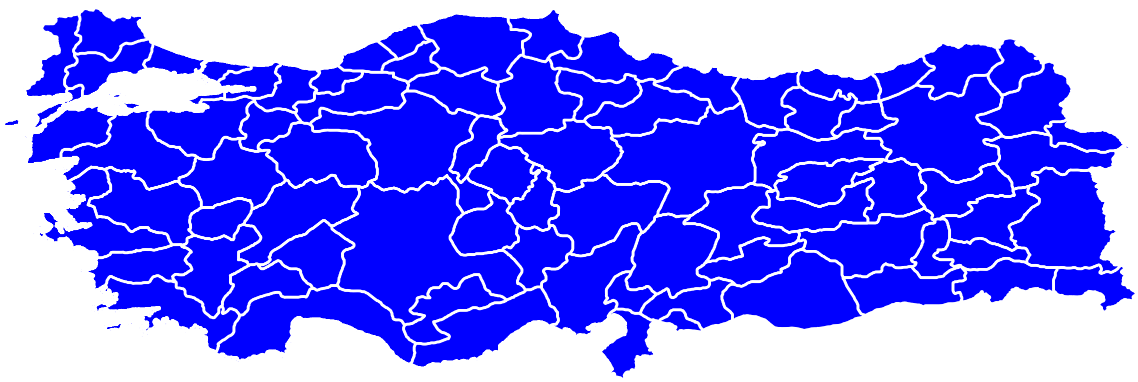
8- Geographical Distrubituon of Migration to Three Biggest Cities

We are going to visualize migration to three biggest cities on Turkey map.

In order to do that we need to go to this web page to download Turkey political map. It is only for non commercial use. First we need to select Turkey and then select R (SpatialPolygonsDataFrame) with level 1 information which only includes cities (Level 2 would also include towns).

```
tmp2<-tempfile(fileext=".rds")
download.file("https://github.com/MEF-BDA503/gpj18-r_boys/blob/master/source_files/gadm36_TUR_1_sp.rds?
```

```r
TRmap <- readRDS(tmp2)
# We need to do one more convertion in order to match two data set
TRmap@data$NAME_1 <- tr_to_en(TRmap@data$NAME_1)
TRmap@data$NAME_1 <- gsub("K. Maras", "Kahramanmaras",TRmap@data$NAME_1)
TRmap@data$NAME_1 <- gsub("Kinkkale","Kirikkale",TRmap@data$NAME_1)
TRmap@data$NAME_1 <- gsub("Zinguldak", "Zonguldak", TRmap@data$NAME_1)
TRmap@data$NAME_1 <- gsub("Afyon","Afyonkarahisar", TRmap@data$NAME_1)
TRmap@data$NAME_1 <- enc2native(gsub("Bartin","Bartin", TRmap@data$NAME_1))
TRmap@data$NAME_1 <- enc2native(gsub("Elazig","Elazig", TRmap@data$NAME_1))
TRmap@data$NAME_1 <- enc2native(gsub("Igdir","Igdir", TRmap@data$NAME_1))
# fortify will allow us to define cities
TRcity <- fortify(TRmap)
ggplot(TRcity) +
  geom_polygon(aes(x = long, y = lat, group = group), color = "white", fill = "blue") +
  theme_void() +
  coord_fixed()
```



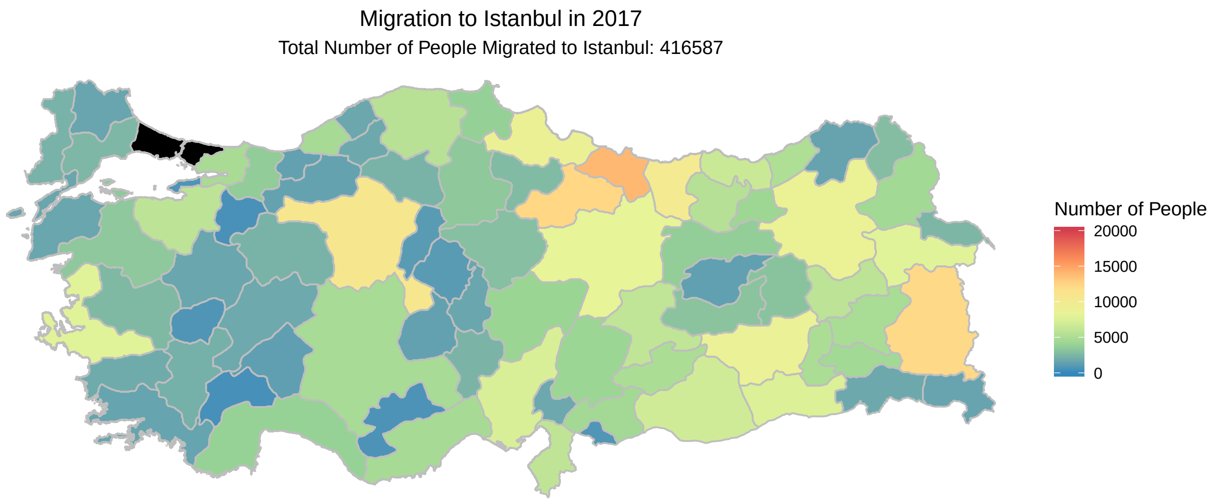
8.1- Geographical Distribution of Migration to Istanbul and Ankara as of 2017

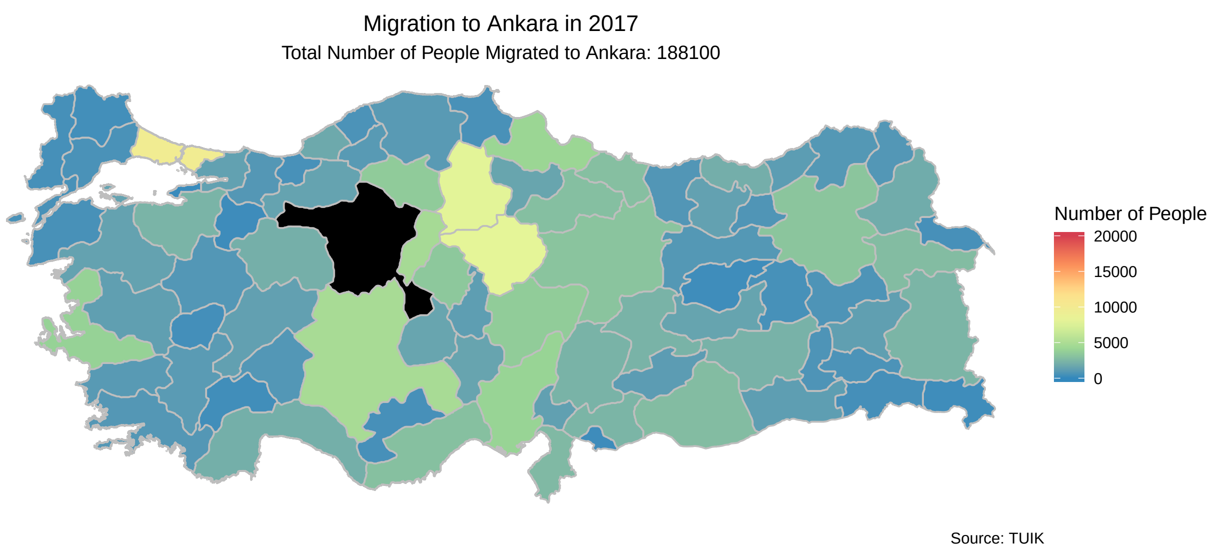
Below two maps illustrates source cities of migration to Istanbul and Ankara as of 2017.

```r
#Istanbul 2017
ist17 <- clean_data %>%
  filter(Year == "2017-01-01", Destination == "Istanbul")
mig_ist <- data_frame(id = rownames(TRmap@data), Birth_Place = TRmap@data$NAME_1) %>%
  left_join(ist17, by = "Birth_Place")
```

```
mig_ist_map <- left_join(TRcity, mig_ist, by = "id")
#Ankara 2017
ank17 <- clean_data %>%
  filter(Year == "2017-01-01", Destination == "Ankara")
mig_ank <- data_frame(id = rownames(TRmap@data), Birth_Place = TRmap@data$NAME_1) %>%
  left_join(ank17, by = "Birth_Place")
mig_ank_map <- left_join(TRcity, mig_ank, by = "id")

grid.arrange(ggplot(mig_ist_map) +
  geom_polygon( aes(x = long, y = lat, group = group, fill = People), color = "grey") +
  coord_map() + theme_void() +
  labs(title = "Migration to Istanbul in 2017", subtitle = paste0("Total Number of People Migrated to I
  scale_fill_distiller(name = "Number of People", palette = "Spectral", limits = c(0,20000), na.value =
  geom_polygon( aes(x = long, y = lat, group = group, fill = People), color = "grey") +
  coord_map() + theme_void() +
  labs(title = "Migration to Ankara in 2017", subtitle = paste0("Total Number of People Migrated to Ank
  scale_fill_distiller(name = "Number of People", palette = "Spectral", limits = c(0,20000), na.value =
```





As seen above migration to Istanbul is more diversified compared to migration to Ankara. Ankara mostly choosen by people in close proximity on the other hand Istanbul attracks people all over Turkey but mostly rural areas of Black Sea region and Eastern Anatolia. These result are confirming the results about economical motivation of the migration.

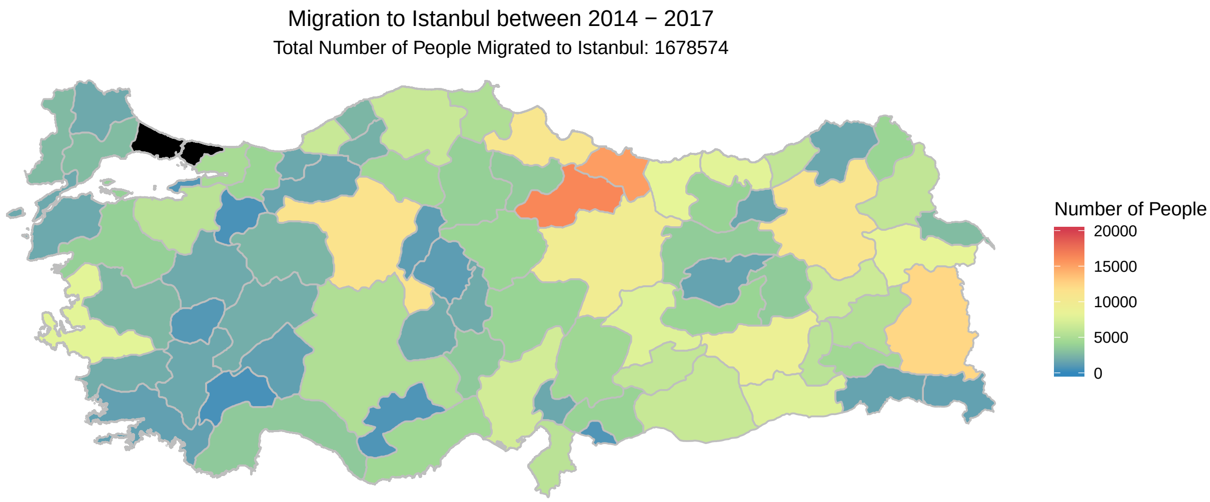
8.2- Geographical Distribution of Cumulative Migration to Istanbul and Ankara Between 2014 - 2017

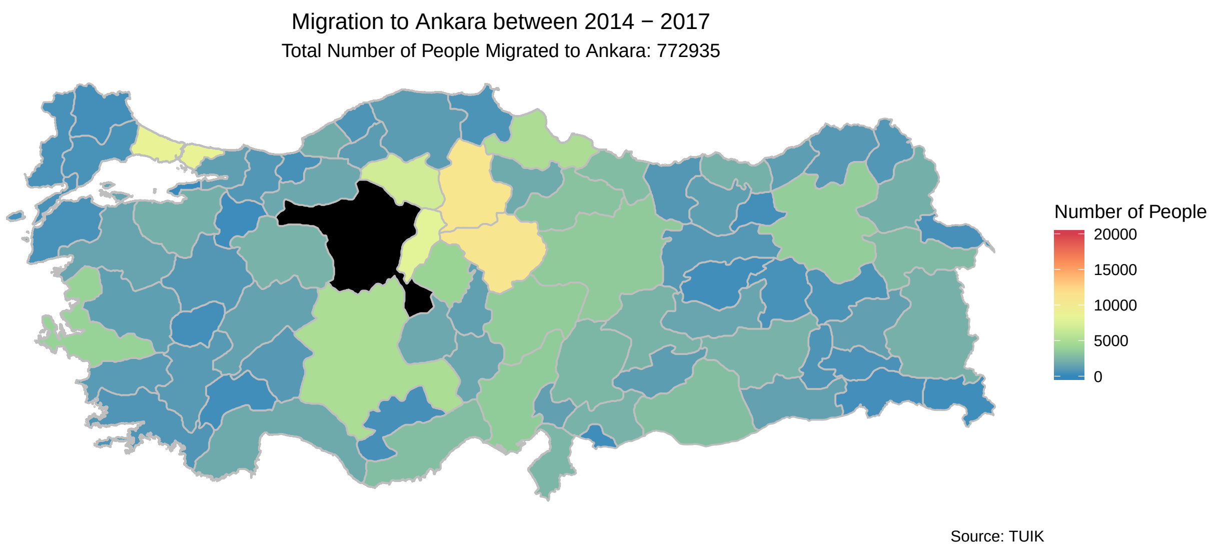
Geographical distribution for the 4 years period between 2014- 2017 for Istanbul and Ankara shown below. Results are pretty similar to distribution realized in 2017 only impacts which we can conclude motives are consistent for the last 4 years.

```
#Istanbul cum
istcum <- clean_data %>%
  filter(Destination == "Istanbul")
```

```
mig_ist_cum <- data_frame(id = rownames(TRmap@data), Birth_Place = TRmap@data$NAME_1) %>%
  left_join(istcum, by = "Birth_Place")
mig_ist_cum_map <- left_join(TRcity, mig_ist_cum, by = "id")
#Ankara cum
ankcum <- clean_data %>%
  filter(Destination == "Ankara")
mig_ank_cum <- data_frame(id = rownames(TRmap@data), Birth_Place = TRmap@data$NAME_1) %>%
  left_join(ankcum, by = "Birth_Place")
mig_ank_cum_map <- left_join(TRcity, mig_ank_cum, by = "id")

grid.arrange(ggplot(mig_ist_cum_map) +
  geom_polygon( aes(x = long, y = lat, group = group, fill = People), color = "grey") +
  coord_map() + theme_void() +
  labs(title = "Migration to Istanbul between 2014 - 2017", subtitle = paste0("Total Number of People Mi
  scale_fill_distiller(name = "Number of People", palette = "Spectral", limits = c(0,20000), na.value =
  geom_polygon( aes(x = long, y = lat, group = group, fill = People), color = "grey") +
  coord_map() + theme_void() +
  labs(title = "Migration to Ankara between 2014 - 2017", subtitle = paste0("Total Number of People Mig
  scale_fill_distiller(name = "Number of People", palette = "Spectral", limits = c(0,20000), na.value =
```





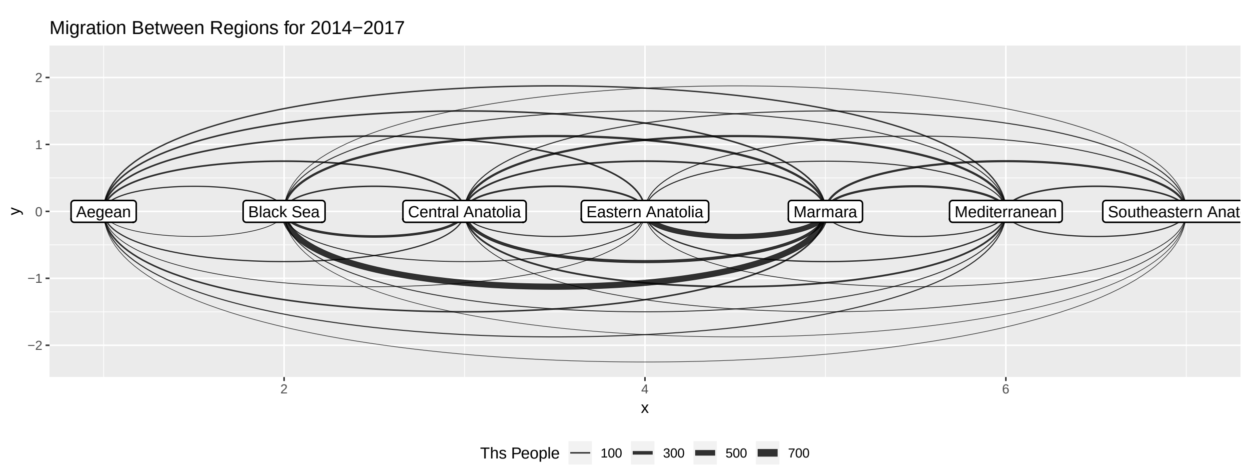
8.3 Regional Impact of Migration

Total inflows and outflows of migration is hard to analyze just in one look for 81 cities of Turkey, essentially we would need to check combination of 81 cities in groups of 2. Instead of this approach, we can analyze on geographical regions of Turkey which is a combination of 7 regions in groups of 2.

```
# File is created manually
tmp3<-tempfile(fileext=".xlsx")
download.file("https://github.com/MEF-BDA503/gpj18-r_boys/blob/master/source_files/region_tr.xlsx?raw=t
reg<-readxl::read_excel(tmp3)
# first region for birth place
breg <- reg %>%
  select(Birth_Place, Region) %>%
  inner_join(., clean_data, by = "Birth_Place")
```

```
# second region for destination
dreg <- inner_join(reg, breg, by = "Destination") %>%
  select(Year, Birth_Place.x, Region.x, Destination, Region.y, People)
colnames(dreg) <- c("Year", "Birth_Place", "Birth_Region", "Destination", "Destination_Region", "People
# migration by regions as of 2017
regions <- dreg %>%
  select(Year, Birth_Region, Destination_Region, People) %>%
  group_by(Birth_Region, Destination_Region) %>%
  summarise(R_People = sum(People))
#graph
nodes <- regions %>%
  distinct(Birth_Region) %>%
  rename(label = Birth_Region) %>%
  rowid_to_column("id")
per_region <- regions %>%
  group_by(Birth_Region, Destination_Region) %>%
  summarise(R_People = round(sum(R_People)/1000,0)) %>%
  arrange(desc(R_People)) %>%
  ungroup()
edges <- per_region %>%
  left_join(nodes, by = c("Birth_Region" = "label")) %>%
  rename(from = id)
edges <- edges %>%
  left_join(nodes, by = c("Destination_Region" = "label")) %>%
  rename(to = id)
edges <- select(edges, from, to, R_People)
regions_igraph <- graph_from_data_frame(d = edges, vertices = nodes, directed = TRUE)
regions_igraph_tidy <- as_tbl_graph(regions_igraph)

ggraph(regions_igraph, layout = "linear") +
  geom_edge_arc(aes(width = R_People), alpha = 0.8) +
  scale_edge_width(range = c(0.2, 3), breaks = c(100,300,500,700,1000)) +
  geom_node_text(aes(label = label)) +
  geom_node_label(aes(label = label), label.size = 0.5) +
  labs(title = "Migration Between Regions for 2014-2017", edge_width = "Ths People") +
  theme(legend.position = "bottom")
```



Above graph represents migration between 7 geographical regions of Turkey. When we are following streams from left to right we need to follow lines under x axis or we need to follow upper lines for streams from right to left.

As presented in the previous analysis concentration of the migration is Marmara region. Most thick lines are Black Sea region to Marmara and Eastern Anatolia to Marmara below x asis. This means most intense migration flows realized from these two region to Marmara.

8.4 Current Breakdown of Population for Istanbul as of 2017

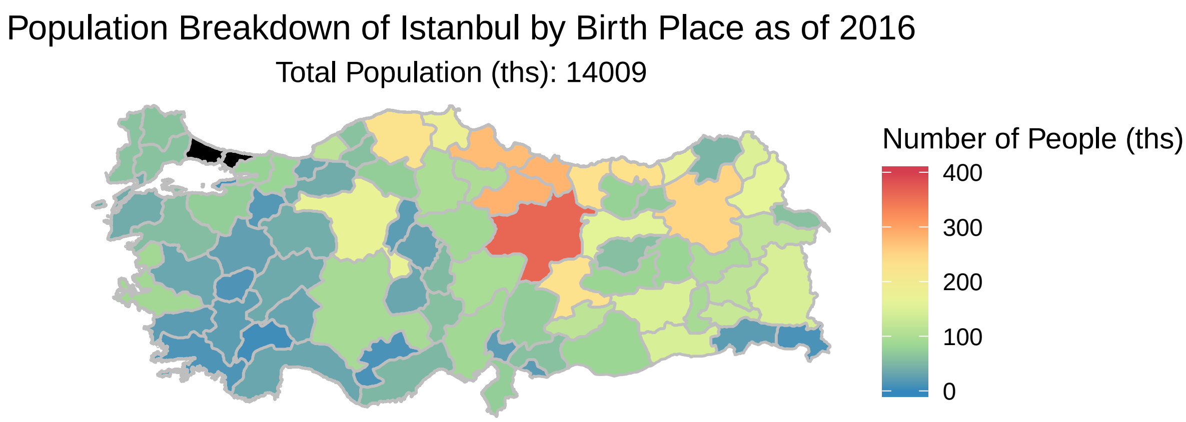
The first map shows the distribution of people living in Istanbul according to their place of birth in 2016, in the second map we see the total number of people who migrated to Istanbul in 2017. According to these two maps we can see that those who are not from Istanbul are calling their relatives to Istanbul or it is easier to migrate to Istanbul for relatives. Especially, families from Black Sea region and Middle Anatolia are have an active role in this migration process.

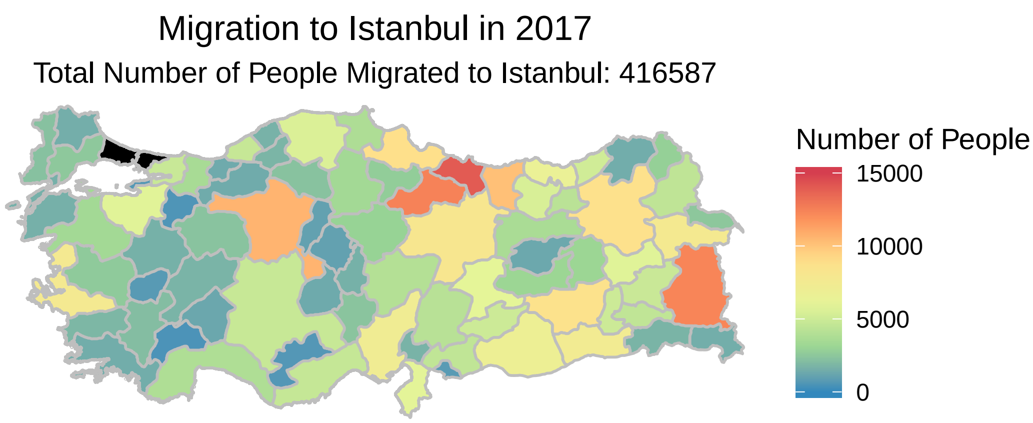
```
tmp4<-tempfile(fileext=".xls")
download.file("https://github.com/MEF-BDA503/gpj18-r_boys/blob/master/source_files/origin_home_town_2016
pop_data<-read_excel(tmp4)
#melt
pop_data <- melt(pop_data)
```

```
## Using X__1, Nüfusa Kayitli Olunan Il as id variables
```

```
colnames(pop_data) <- c("Year", "Province", "Birth_Place", "People")
pop_data <- pop_data %>%
  select(Province, Birth_Place, People)
pop_data$Birth_Place <- as.character(pop_data$Birth_Place)
pop_data$Province <- enc2native(pop_data$Province)
pop_data$Birth_Place <- enc2native(pop_data$Birth_Place)
pop_data$Province <- tr_to_en(pop_data$Province)
pop_data$Birth_Place <- tr_to_en(pop_data$Birth_Place)
# Istanbul
pop_data_ist <- pop_data %>%
  filter(Province == "Istanbul")
pop_data_ist$People <- pop_data_ist$People/1000
pop_data_ist_m <- data_frame(id = rownames(TRmap@data), Birth_Place = TRmap@data$NAME_1) %>%
  left_join(pop_data_ist, by = "Birth_Place")
pop_data_ist_map <- left_join(TRcity, pop_data_ist_m, by = "id")
```

```
grid.arrange(ggplot(pop_data_ist_map) +
  geom_polygon( aes(x = long, y = lat, group = group, fill = People), color = "grey") +
  coord_map() + theme_void() +
  labs(title = "Population Breakdown of Istanbul by Birth Place as of 2016", subtitle = paste0("Total Po
  scale_fill_distiller(name = "Number of People (ths)", palette = "Spectral", limits = c(0,400), na.valu
  geom_polygon( aes(x = long, y = lat, group = group, fill = People), color = "grey") +
  coord_map() + theme_void() +
  labs(title = "Migration to Istanbul in 2017", subtitle = paste0("Total Number of People Migrated to Is
  scale_fill_distiller(name = "Number of People", palette = "Spectral", limits = c(0,15000), na.value =
```





8.5 - Comparison Between Population With and Without Migration

In the first map, we can see the current distribution of population in Turkey. The most crowded city is Istanbul with a population of over 15 million. Ankara and Izmir are following Istanbul.

What would be the distribution of population without migration in Turkey?

The second map gives the answer of this question. The most crowded cities would not change in ranking, Istanbul, Ankara and Izmir, but the distribution of population over the cities would change. The most affected city would be Istanbul. If if the migration effect is disregarded, the population of Istanbul would be 7.8 million.

```
org_pop <- pop_data %>%
  select(Birth_Place, People) %>%
  group_by(Birth_Place) %>%
  summarise(pop = sum(People)/1000)
cur_pop <- pop_data %>%
  select(Province, People) %>%
  group_by(Province) %>%
  summarise(popp = sum(People)/1000)
#join
org_pop_m <- data_frame(id = rownames(TRmap@data), Birth_Place = TRmap@data$NAME_1) %>%
  left_join(org_pop, by = "Birth_Place")
org_pop_map <- left_join(TRcity, org_pop_m, by = "id")
cur_pop_m <- data_frame(id = rownames(TRmap@data), Province = TRmap@data$NAME_1) %>%
  left_join(cur_pop, by = "Province")
cur_pop_map <- left_join(TRcity, cur_pop_m, by = "id")
```

```r
#plot
grid.arrange(ggplot(cur_pop_map) +
  geom_polygon( aes(x = long, y = lat, group = group, fill = popp), color = "grey") +
  coord_map() + theme_void() +
  labs(title = "Population as of 2016", subtitle = paste0("Total Population (ths): ", round(sum(cur_pop$
  scale_fill_distiller(name = "Number of People (ths)", palette = "Spectral", limits = c(0,15000)) + the
  geom_polygon( aes(x = long, y = lat, group = group, fill = pop), color = "grey") +
  coord_map() + theme_void() +
  labs(title = "Population without Migration", subtitle = paste0("Total Population (ths): ", round(sum(c
  scale_fill_distiller(name = "Number of People (ths)", palette = "Spectral", limits = c(0,15000)) + the
```