Data Management and Curation

Juan Bonilla

2018-03-01

SELECTING, PROCESSING AND CLEANING OF THE DATASET

The purpose of this exercise is to study bird sightings and a relationship with human population in America. is there a relationship between countries population and reported number of bird sightings? I will start the study with the **Hypothesis** that countries with higher population have a higher number of bird sightings. It seems obvious that if there are more bird watchers, there must be more bird sightings reports. If the hypothesis is not true, I will examine if population density is related to the number of bird sightings.

Using data collected from two different sources and different methods for data wrangling with R, I will test these hypotheses to prove them true or wrong.

The motivation for this project aligns with the goals of ebirds.org which seeks to keep track of bird species, find more birds, explore latest sightings and contributing data to science and conservation. Throughout this exercise, I will do different calculations such as population density per country and the total number of bird sightings per species and per country. The results of these operations will be visualized using the ggplot library available in R and the spatial data will be visualized using google fusion tables and its embedded map options.

Resources:

- R Markdown for document design
- R Studio for programming.
- R libraries for visualization (ggplot, RColorBrewer, formattable, knitr)
- R libraries for data wrangling (tidyverse, dplyr, xml2, httr, rvest, curl)
- R libraries for data collection (geonames, rebird)

STEP 1 Collecting and cleaning data:

- Data was collected by making requests to two different APIs: geonames and rebird (ebird for R).
- The GeoNames geographical database covers all countries and contains over eleven million placenames that are available for download free of charge. More information can be found at: http://www.geonames.org
- eBird is the world's largest biodiversity-related citizen science project, with more than 100 million bird sightings contributed each year by eBirders around the world.

LOADING THE NECESSARY LIBRARIES

```
suppressWarnings(suppressMessages(easypackages::libraries("ggplot2", "xml2",
"tidyverse", "geonames", "rebird", "httr", "rplos", "curl", "rvest", "stringr", "knitr",
"kableExtra", "formattable", "kableExtra", "RColorBrewer")))
```

To start, I need to request general information about countries to the *geonames API* using the following code:

geon<- GET("http://api.geonames.org/countryInfo?username=") *this part of the code is hiden to protect my user ID and password.*

The API returns the results of the query in XML format:

```
xml_country<- read_xml(content(geon, as = "text"))
```

To access the text inside the XML document I need to create a chunk of code that goes inside the lists takes the text and create a vector with the names of the nodes that I need. After, extracting the text form those nodes, I create a data frame with the list and assigning names to the columns.

```
countryinfo<- c("//countryCode", "//countryName", "//continentName",  "//population", "//areaInSqKm" )
output<- list()
for (i in seq_along(countryinfo)) {
  output[[i]]<- (xml_text(xml_find_all(xml_country,countryinfo[[i]])))}
dfcountries<- data.frame(output)
names(dfcountries)<- str_trim(str_replace(countryinfo, "//", ""))
dfcountries %>% head(10) %>%  kable("latex") %>%
  kable_styling(full_width = F) %>%
  column_spec(1:2, bold = T, border_right = T) %>%
  column_spec(2, width = "10em")
```

countryCode	countryName	continentName	population	areaInSqKm
AD	Andorra	Europe	84000	468.0
AE	United Arab Emirates	Asia	4975593	82880.0
AF	Afghanistan	Asia	29121286	647500.0
AG	Antigua and Barbuda	North America	86754	443.0
AI	Anguilla	North America	13254	102.0
AL	Albania	Europe	2986952	28748.0
AM	Armenia	Asia	2968000	29800.0
AO	Angola	Africa	13068161	1246700.0
AQ	Antarctica	Antarctica	0	1.4E7
AR	Argentina	South America	41343201	2766890.0

- Here we can see that I have 250 observations (that how R calls records) with 5 variables (that's how R calls attributes). Here I can also see the format of each variable: *factors*. *Factor* is a data type for text that uses levels to group categories. For Example, the variable continent has 250 observations but 7 levels because there are only 7 continents. One of the problems that I see here is that population and areaInSqKm are numbers so it does not make sense that they are in *factor* format.

Data cleaning requires not only removing unwanted characters but checking the format of the data. Population and areaInSqKm were imported as *factors* because these variables have characters mixed with numbers in some fields. That's a problem because I need them as *numeric* to do calculations and arrange them. To fix these issue, I identify the conflicting data field, filter them out (if they are not needed) or change them to *character* format. In the code chunk below, I selected countries in America (North and South), mutated the factor type fields into character and then into *numeric* (did not work directly from factor to numeric). The highlighted columns are in *numeric* type.

```
dfcountry<- dfcountries %>%
  filter(continentName %in% c("South America", "North America")) %>% droplevels() %>%
  mutate(pop= as.numeric(as.character(population)), area=as.numeric(as.character(areaInSqKm)))
dfcountry  %>%
  select(-population, -areaInSqKm) %>%  arrange(desc(pop)) %>% head(15) %>%
  kable( "latex") %>%
  kable_styling(full_width = F) %>%
  column_spec(4:5, width = "10em", background = "green")
```

countryCode	countryName	continentName	pop	area
US	United States	North America	310232863	9629091
BR	Brazil	South America	201103330	8511965
MX	Mexico	North America	112468855	1972550
CO	Colombia	South America	47790000	1138910
AR	Argentina	South America	41343201	2766890
CA	Canada	North America	33679000	9984670
PE	Peru	South America	29907003	1285220
VE	Venezuela	South America	27223228	912050
CL	Chile	South America	16746491	756950
EC	Ecuador	South America	14790608	283560
GT	Guatemala	North America	13550440	108890
CU	Cuba	North America	11423000	110860
BO	Bolivia	South America	9947418	1098580
DO	Dominican Republic	North America	9823821	48730
HT	Haiti	North America	9648924	27750

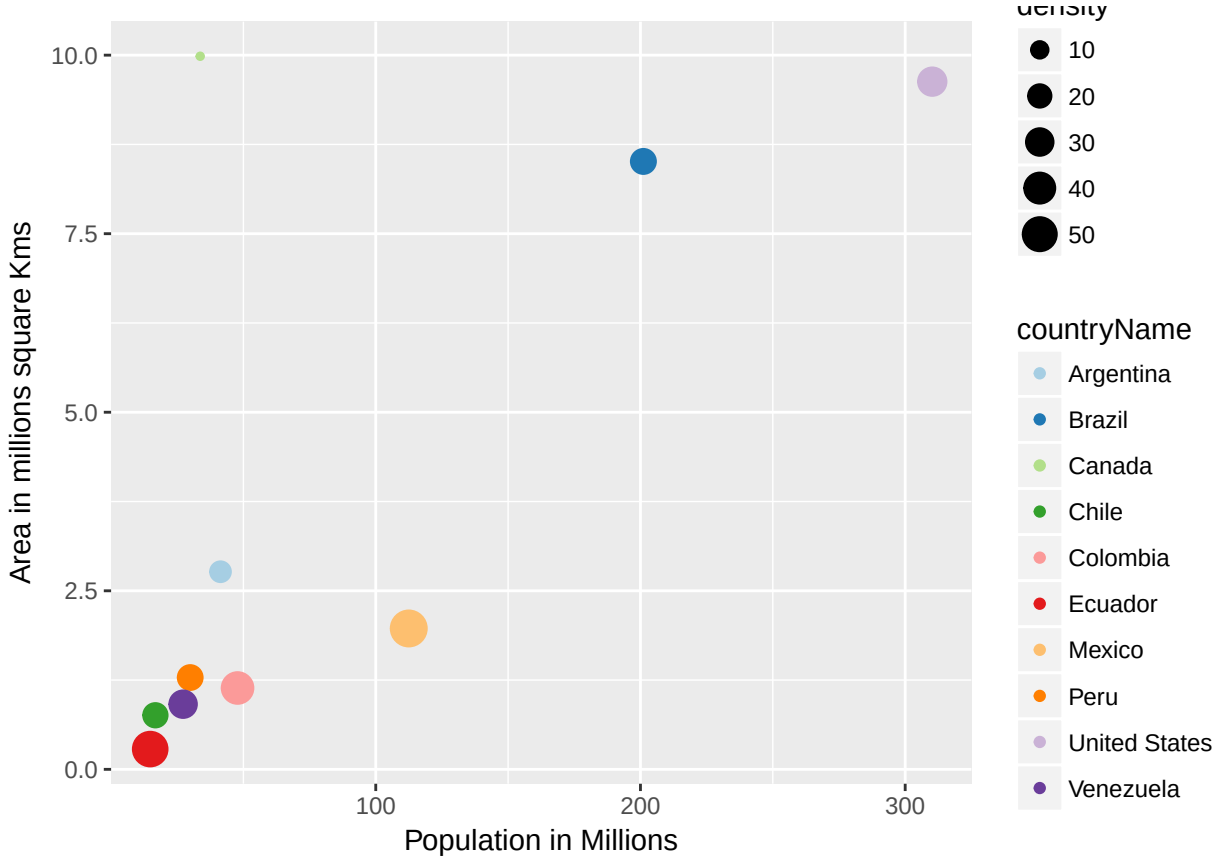
Now that I have the information of all the 55 countries in America, I need to calculate the population density of each country which is equivalent to dividing the population by area. I will find the top 10 countries with the higher population and visualize the density distribution.

```
dfcountries2<- dfcountry%>% select(countryCode, countryName, pop, area) %>%
  droplevels() %>% arrange(desc(pop, area)) %>% head(10) %>%
  mutate(density= pop/area)
dfcountries2%>% kable( "latex") %>%
  kable_styling(full_width = F) %>%
  column_spec(5, background = "green")
```

countryCode	countryName	pop	area	density
US	United States	310232863	9629091	32.218292
BR	Brazil	201103330	8511965	23.625958
MX	Mexico	112468855	1972550	57.016986
CO	Colombia	47790000	1138910	41.961173
AR	Argentina	41343201	2766890	14.942120
CA	Canada	33679000	9984670	3.373071
PE	Peru	29907003	1285220	23.269948
VE	Venezuela	27223228	912050	29.848394
CL	Chile	16746491	756950	22.123642
EC	Ecuador	14790608	283560	52.160418

To understand the data better, I will create a chart to see data from a different perspective. Here we get some see that: the United States has the highest population, Canada is the biggest country and has the lowest population density. On the other hand, Mexico and Ecuador are the most densely populated countries. Argentina has a low density since the area in squareKm is extensive (more than double than Colombia) and the population not very large (lower than Colombia).

```
dfcountries2 %>% ggplot(aes(x= pop/10^6, y= area/10^6))+
  geom_point(aes(colour= countryName, size= density))+
  labs(x= "Population in Millions", y= "Area in millions square Kms")+
  scale_colour_manual(values=brewer.pal(n=10, "Paired"))
```



Based on this early results and following the initial hypothesis, we would expect to see the highest number of bird sightings from the United States and Brazil. To test that, I will get the bird sighting data from ebirds API.

STEP 2 Collecting and cleaning data from rebirds

This procedure to query data from ebirds API is slightly different than the one I used before. Ebirds has a R library with a number of R functions to query data depending on different variables. Here, I will use region (ebirdregion function) since I am interested in the places (countries) where the sighting occurred. In the code chunk below, I called the ebirdregion function and I fed it with the list of countries that I am interested in.

```
birds_countries<- list()
for (i in seq_along(dfcountries2$countryCode)) {
  birds_countries[[i]]<- ebirdregion(region = (as.character(dfcountries2$countryCode[[i]])),
                                     regtype = "country")}
names(birds_countries)<- dfcountries2$countryCode
```

The result of the query is a data frame with 940 observations and 12 variables. These variables include the scientific and common name of bird species, the number bird per observation, whether the observation is valid or reviewed, the region of the sighting, the country and the coordinates.

```
names(birds_countries$US)
```

```
##  [1] "lng"             "locName"         "howMany"
##  [4] "sciName"         "obsValid"        "locationPrivate"
##  [7] "obsDt"           "obsReviewed"     "comName"
## [10] "lat"             "locID"           "locId"
```

The variable howMany refers to the number of birds in each sighting (default ebird name). The format of this variable is integer but it has NA values (Not Applicable). This is a problem because in R every operation that involves a NA will result in NA as result. I need to calculate the total number of observation per country and species so to avoid NA results I need replace NA values for 0s. Notice here that when I bound all the list into a data frame, R created an index with the name of the list (initials of the country) and a numeric value.

```
df <- do.call("rbind", lapply(birds_countries, data.frame))
df$howMany[is.na(df$howMany)] <- 0
df<-df %>% select(c(-locId, -locID, -obsDt, -locationPrivate ))
df %>% select(-lng, -locName, -obsValid, -obsReviewed, -lat) %>%
  head(5) %>% kable( "latex") %>%
  kable_styling(full_width = F) %>%
  column_spec(1, background = "yellow")
```

	howMany	sciName	comName
US.1	1	Anhinga anhinga	Anhinga
US.2	1	Setophaga dominica	Yellow-throated Warbler
US.3	1	Setophaga petechia	Yellow Warbler
US.4	1	Fulica americana	American Coot
US.5	1	Vireo griseus	White-eyed Vireo

That is nice but I do need the initials of the countries in the index so I will turn it into a column and use *regular expressions* to remove the numeric values.

** regular expressions (regex or regexp for short) is a special text string for identifying a pattern and modify it.

```
df2<-df %>% mutate(countryCode= rownames(df))
 ## keeps the first group (first two characters)
df2$countryCode<- str_replace_all(df2$countryCode,"(^..)(.+)", "\\1") %>%
  as.factor()
df2 %>% select(countryCode, howMany, sciName, comName) %>%
  head(5) %>% kable( "latex") %>%
  kable_styling(full_width = F) %>%
  column_spec(1, background = "yellow")
```

countryCode	howMany	sciName	comName
US	1	Anhinga anhinga	Anhinga
US	1	Setophaga dominica	Yellow-throated Warbler
US	1	Setophaga petechia	Yellow Warbler
US	1	Fulica americana	American Coot
US	1	Vireo griseus	White-eyed Vireo

Here, I'm grouping the birds per name and calculated the 10 species most commonly seen in all six countries by common name and scientific name.

```
df2  %>% group_by(comName, sciName) %>%
  summarise(mostseen= sum(howMany)) %>%
  arrange(desc(mostseen)) %>% head(10) %>%kable( "latex") %>%
  kable_styling("striped", full_width = F) %>%
  row_spec(1, bold = T, color = "white", background = "green")
```

comName	sciName	mostseen
Dickcissel	**Spiza americana**	**20008**
Snow Goose	Anser caerulescens	5005
Sooty Shearwater	Ardenna grisea	1637
Calidris sp.	Calidris sp.	1513
Semipalmated Sandpiper	Calidris pusilla	1266
Kelp Gull	Larus dominicanus	1063
California Gull	Larus californicus	1051
peep sp.	Calidris sp. (peep sp.)	614
Roseate Spoonbill	Platalea ajaja	551
Cattle Egret	Bubulcus ibis	542

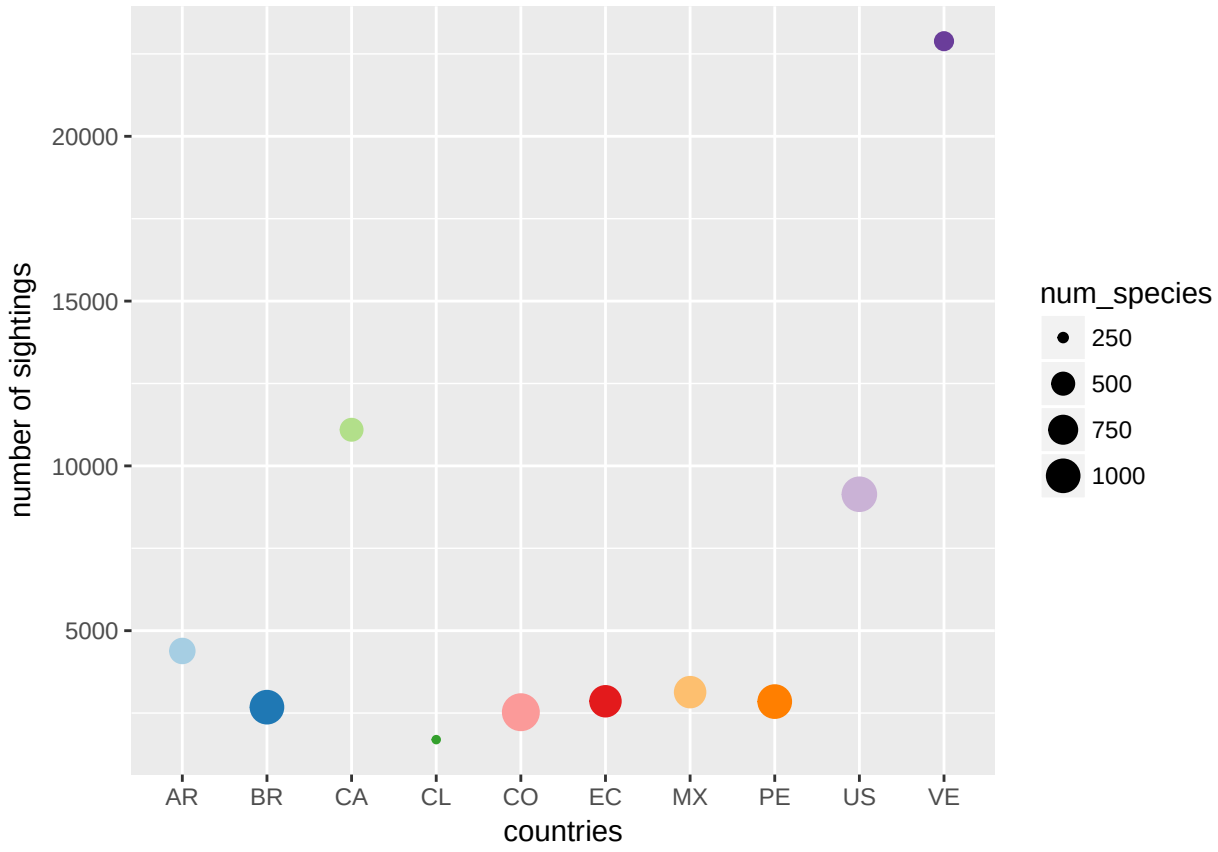
In the code below, I am calculating the overall number of sights (reviewed and not) and the different number of species per country.

```
df3<-df2 %>%  group_by(countryCode) %>%
  summarise(allsights= sum(howMany), num_species= length(sciName))
df3 %>% arrange(desc(allsights)) %>%  kable( "latex") %>%
  kable_styling(bootstrap_options = "striped", full_width = F)
```

countryCode	allsights	num_species
VE	22886	380
CA	11096	493
US	9139	1057
AR	4384	590
MX	3135	876
EC	2858	875
PE	2848	1003
BR	2679	1002
CO	2526	1210
CL	1694	245

Peru has less species than the Unite States but has almost double of sightings. Also, we can see here that Chile reports the lowest number of species but still reports more sightings than Canada. This is understandable since many birds had migrated south before winter started. It is less clear why the number is so low in Venezuela which is a tropical country. Other tropical countries such as Colombia and Ecuador have a high diversity of birds but do not report as many sightings.

```
df3%>% ggplot(aes(x= countryCode, y= allsights, colour= countryCode ))+
  geom_point(aes(size= num_species))+labs(x= "countries", y= "number of sightings")+
  guides(colour= FALSE)+ scale_colour_manual(values=brewer.pal(n=10, "Paired"))
```



We can see that Argentina has a high number of sights and significantly more reviewed sights than all the other countries. This could suggest that there is more interest in this country for bird watching and studying.

In terms of biodiversity, Colombia and Brazil have more bird species which makes sense because they are tropical countries.

```
df2 %>% group_by(countryCode, obsReviewed) %>%
  summarise(sights= sum(howMany), species= length(sciName)) %>%
  kable( format = "latex") %>%
  kable_styling("striped", full_width = F) %>%
  row_spec(0, color = "white", bold= TRUE, background= "green" ) %>%
  row_spec(c(2,4, 6, 8, 10, 12, 14, 16, 18, 20), italic= TRUE, bold= TRUE, background= "yellow" )
```

countryCode	obsReviewed	sights	species
AR	FALSE	4182	551
AR	***TRUE***	***202***	***39***
BR	FALSE	2672	999
BR	***TRUE***	***7***	***3***
CA	FALSE	9117	416
CA	***TRUE***	***1979***	***77***
CL	FALSE	1689	242
CL	***TRUE***	***5***	***3***
CO	FALSE	2515	1205
CO	***TRUE***	***11***	***5***
EC	FALSE	2086	843
EC	***TRUE***	***772***	***32***
MX	FALSE	3100	872
MX	***TRUE***	***35***	***4***
PE	FALSE	2760	991
PE	***TRUE***	***88***	***12***
US	FALSE	8628	930
US	***TRUE***	***511***	***127***
VE	FALSE	2751	375
VE	***TRUE***	***20135***	***5***

STEP 3 MERGING BOTH DATASETS

Joining countries and bird dataframes to examine the relationship between sights and population.

```
both<-left_join(dfcountries2, df3, by= "countryCode")
```

```
## Warning: Column `countryCode` joining factors with different levels,
## coercing to character vector
```

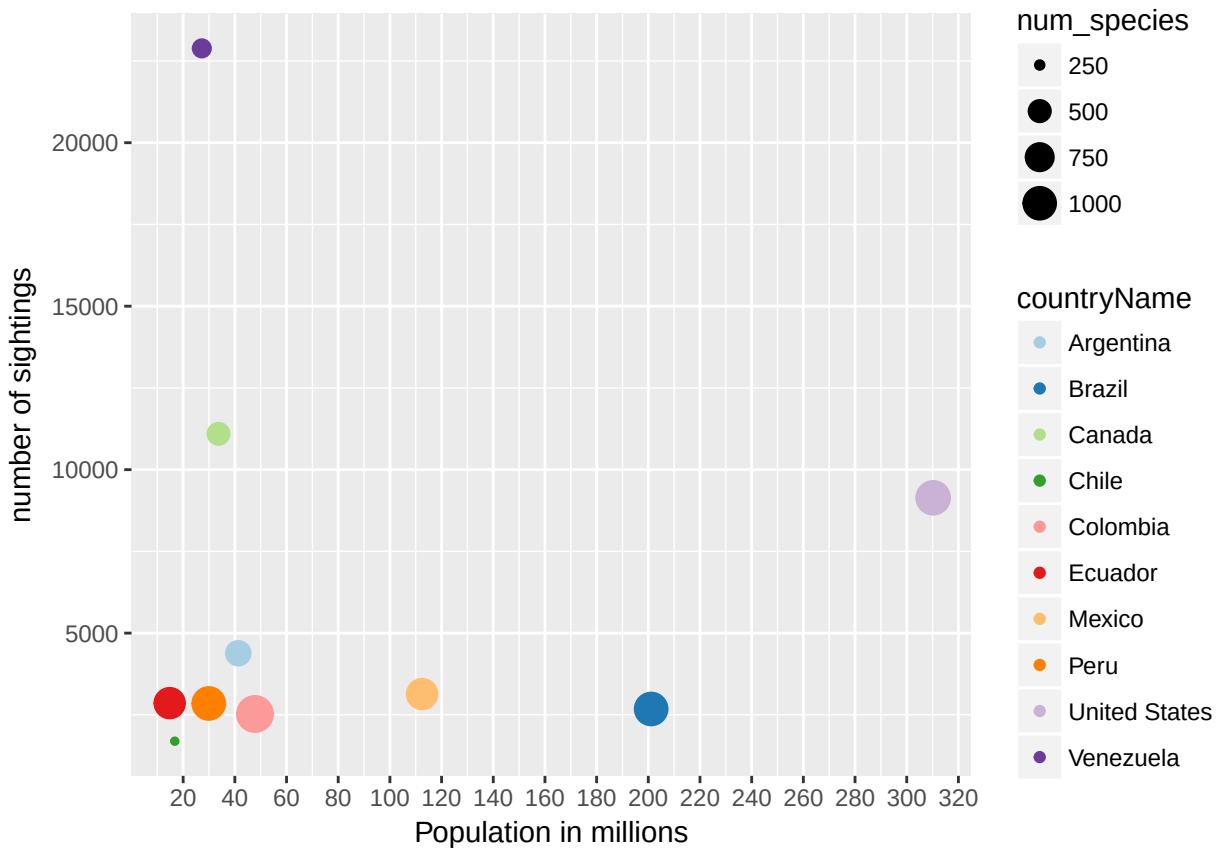
```
both %>% arrange(desc(density)) %>% kable( "latex") %>%
  kable_styling("striped", full_width = F) %>%
  column_spec(5, bold = T) %>%
  row_spec(1, bold = T, color = "white", background = "#D7261E")
```

countryCode	countryName	pop	area	**density**	allsights	num_species
MX	**Mexico**	**112468855**	**1972550**	**57.016986**	**3135**	**876**
EC	Ecuador	14790608	283560	**52.160418**	2858	875
CO	Colombia	47790000	1138910	**41.961173**	2526	1210
US	United States	310232863	9629091	**32.218292**	9139	1057
VE	Venezuela	27223228	912050	**29.848394**	22886	380
BR	Brazil	201103330	8511965	**23.625958**	2679	1002
PE	Peru	29907003	1285220	**23.269948**	2848	1003
CL	Chile	16746491	756950	**22.123642**	1694	245
AR	Argentina	41343201	2766890	**14.942120**	4384	590
CA	Canada	33679000	9984670	**3.373071**	11096	493

- Here we can refute the initial hypothesis. Colombia, Argentina and Peru have smaller population than Brazil but they report more sightings. Peru has a low population but a very high number of sightings.

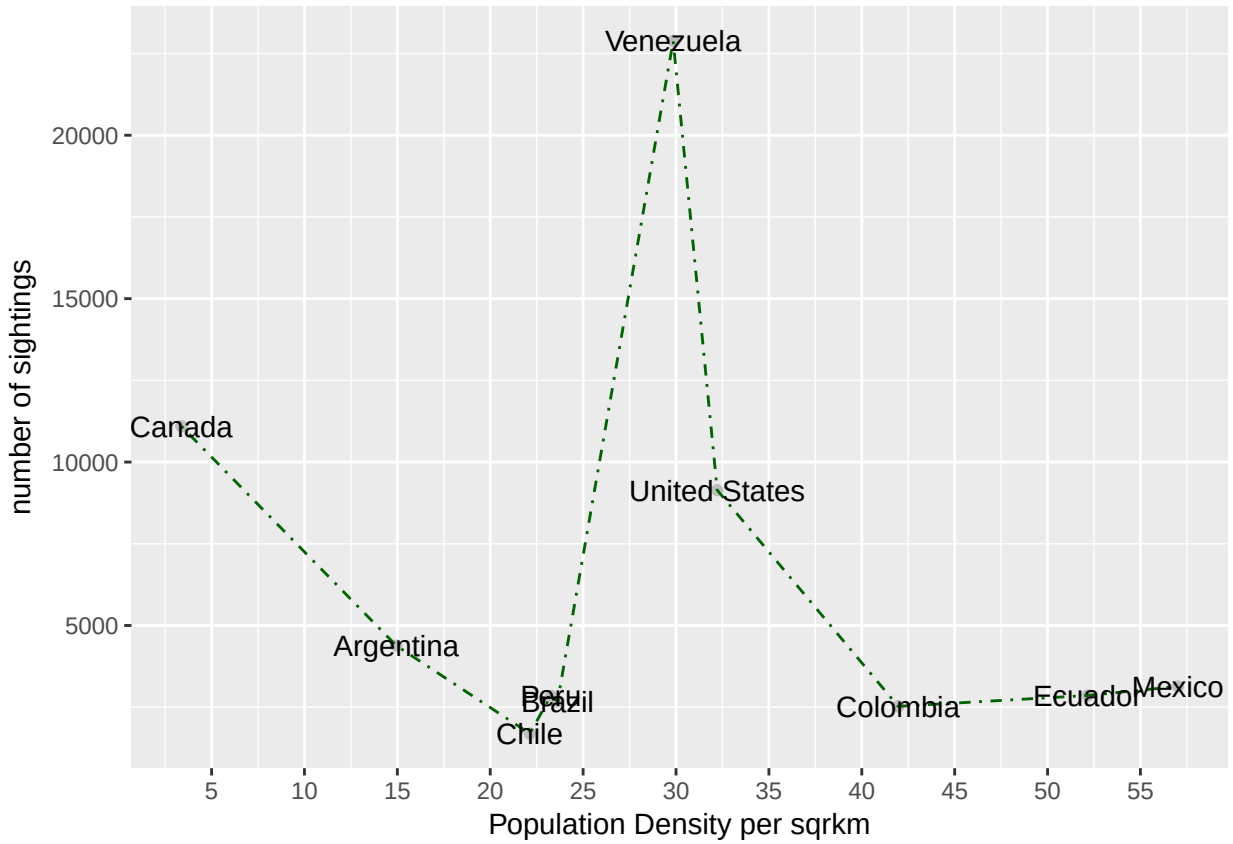
```
ggplot(both, aes(x= pop/10^6, y= allsights, size=num_species ))+
  geom_point(aes(colour= countryName))+
```

```
  labs(x= "Population in millions", y= "number of sightings")+
  scale_x_continuous(breaks = seq(0, 320, by = 20))+
  scale_colour_manual(values=brewer.pal(n=10, "Paired"))
```



The second hypothesis also fails since Mexico, Ecuador and Colombia have a high population density but low number of sightings. Both of the hypotheses were false.

```
ggplot(both, aes(x= density, y= allsights))+
  geom_point(colour= "gray")+
  geom_line(linetype = "dotdash", colour= "darkgreen")+
  geom_text(label= both$countryName)+
  labs(x= "Population Density per sqrkm", y= "number of sightings")+
  scale_x_continuous(breaks = seq(0, 60, by = 5))
```


Venezuela
20000
15000
number of sightings
Canada
10000
United States
5000
Argentina
Peru
Brazil
Chile
Colombia
Ecuador
Mexico
5
10
15
20
25
30
35
40
45
50
55
Population Density per sqrkm