607-Tidyverse

Jose Mawyin

12/9/2019

607 - Using Tidyverse Packages on NYPD Arrests Data

This programming sample "vignette" will showcase how three different Tidyverse package can be used to import, manipulate and graph the following dataset:

"List of every arrest in NYC going back to 2006 through the end of the previous calendar year. This is a breakdown of every arrest effected in NYC by the NYPD going back to 2006 through the end of the previous calendar year. This data is manually extracted every quarter and reviewed by the Office of Management Analysis and Planning before being posted on the NYPD website. Each record represents an arrest effected in NYC by the NYPD and includes information about the type of crime, the location and time of enforcement. In addition, information related to suspect demographics is also included. This data can be used by the public to explore the nature of police enforcement activity. Please refer to the attached data footnotes for additional information about this dataset."

https://catalog.data.gov/dataset/nypd-arrests-data-historic

Using readr

The readr package can be used to import our dataset and the output will be a Tibble wich is a dataframe with extra properties suitable for data analysis. Below we import our dataset and "readr" outputs some column properties that may be of interest.

```
NY.Arrest <- read_csv("/Users/josemawyin/Downloads/NYPD_Arrests.csv")
```

```
## Parsed with column specification:
## cols(
##   ARREST_KEY = col_double(),
##   ARREST_DATE = col_character(),
##   PD_CD = col_double(),
##   PD_DESC = col_character(),
##   KY_CD = col_double(),
##   OFNS_DESC = col_character(),
##   LAW_CODE = col_character(),
##   LAW_CAT_CD = col_character(),
##   ARREST_BORO = col_character(),
##   ARREST_PRECINCT = col_double(),
##   JURISDICTION_CODE = col_double(),
##   AGE_GROUP = col_character(),
##   PERP_SEX = col_character(),
##   PERP_RACE = col_character(),
##   X_COORD_CD = col_double(),
##   Y_COORD_CD = col_double(),
##   Latitude = col_double(),
##   Longitude = col_double()
## )
```

The lenght and number of columns dimensions of of our initial dataframe is 4798339, 18. It is always useful to take a peek into imported data even if just to see that the import process did not make a mess of things.

```
glimpse(NY.Arrest)
```

```
## Observations: 4,798,339
## Variables: 18
## $ ARREST_KEY        <dbl> 173130602, 173114463, 173113513, 173113423, ...
## $ ARREST_DATE       <chr> "12/31/2017", "12/31/2017", "12/31/2017", "1...
## $ PD_CD             <dbl> 566, 478, 849, 101, 101, 397, 101, 511, 101,...
## $ PD_DESC           <chr> "MARIJUANA, POSSESSION", "THEFT OF SERVICES,...
## $ KY_CD             <dbl> 678, 343, 677, 344, 344, 105, 344, 235, 344,...
## $ OFNS_DESC         <chr> "MISCELLANEOUS PENAL LAW", "OTHER OFFENSES R...
## $ LAW_CODE          <chr> "PL 2210500", "PL 1651503", "LOC000000V", "P...
## $ LAW_CAT_CD        <chr> "V", "M", "V", "M", "M", "F", "M", "M", "M",...
## $ ARREST_BORO       <chr> "Q", "Q", "K", "M", "M", "K", "M", "M", "M",...
## $ ARREST_PRECINCT   <dbl> 105, 114, 73, 18, 18, 73, 9, 25, 23, 17, 83,...
## $ JURISDICTION_CODE <dbl> 0, 0, 1, 0, 0, 0, 0, 0, 2, 0, 0, 0, 0, 0, 0,...
## $ AGE_GROUP         <chr> "25-44", "25-44", "18-24", "25-44", "45-64",...
## $ PERP_SEX          <chr> "M", "M", "M", "M", "M", "M", "F", "M", "M",...
## $ PERP_RACE         <chr> "BLACK", "ASIAN / PACIFIC ISLANDER", "BLACK"...
## $ X_COORD_CD        <dbl> 1063056, 1009113, 1010719, 987831, 987073, 1...
## $ Y_COORD_CD        <dbl> 207463, 219613, 186857, 217446, 216078, 1885...
## $ Latitude          <dbl> 40.73577, 40.76944, 40.67952, 40.76352, 40.7...
## $ Longitude         <dbl> -73.71564, -73.91024, -73.90457, -73.98707, ...
```

Using dplyr and stringr

We will use the filter() function from dplyr and the function str_detect() from stringr to only select the arrest of perpetrators in the Borough of Queens, using "Dangerous Weapons", male and in the year of 2017. We can use the "&" pipeline to sequentially apply all these filters.

```
Queens.Dangerous.Male.2017.Arrests <- filter(NY.Arrest, NY.Arrest$ARREST_BORO ==
    "Q" & NY.Arrest$OFNS_DESC == "DANGEROUS WEAPONS" & NY.Arrest$PERP_SEX ==
    "M" & str_detect(NY.Arrest$ARREST_DATE, "2017"))
```

The lenght and number of columns dimensions of our filtered dataframe is 2014, 18.

Using ggplot2

Let's plot the geographic distribution of the arrested perpetrators in NY. First, let's get a geographic outline of NY from a "shapefile".

```
counties <- readOGR("/Users/josemawyin/Downloads/nybb_19d/nybb.shp",
    layer = "nybb")
```

```
## OGR data source with driver: ESRI Shapefile
## Source: "/Users/josemawyin/Downloads/nybb_19d/nybb.shp", layer: "nybb"
## with 5 features
## It has 4 fields
```

The following 3 chunks convert the Latitude and Longitude information from the arrest dataframe into a format that ggplot can recognize and use for plotting.

```
proj4string(counties)
```

```
## [1] "+proj=lcc +lat_1=41.03333333333333 +lat_2=40.66666666666666 +lat_0=40.16666666666666 +lon_0=-74
```

```
class(Queens.Dangerous.Male.2017.Arrests)
```

```
## [1] "spec_tbl_df" "tbl_df"      "tbl"         "data.frame"
```

```
coordinates(Queens.Dangerous.Male.2017.Arrests) <- ~Longitude +
    Latitude
class(Queens.Dangerous.Male.2017.Arrests)
```

```
## [1] "SpatialPointsDataFrame"
## attr(,"package")
## [1] "sp"
```

```
proj4string(Queens.Dangerous.Male.2017.Arrests)
```

```
## [1] NA
```

```
proj4string(Queens.Dangerous.Male.2017.Arrests) <- CRS("+proj=longlat +datum=NAD83")
mapdata <- spTransform(Queens.Dangerous.Male.2017.Arrests, CRS(proj4string(counties)))
identical(proj4string(mapdata), proj4string(counties))
```

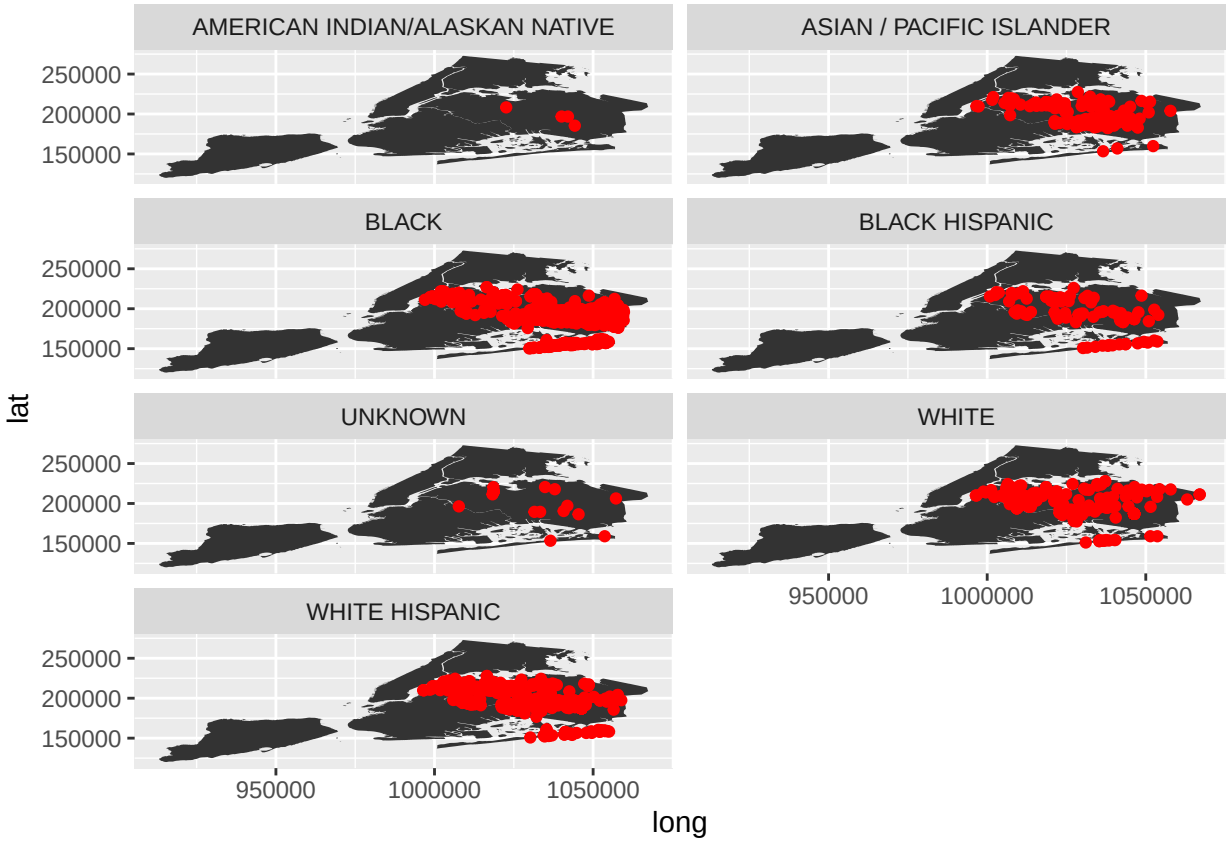
```
## [1] TRUE
```

Finally, let's map the geographic distribution of the perpetrators splitting the maps in facets containing the reported race.

```
mapdata <- data.frame(mapdata)
names(mapdata)[names(mapdata) == "Longitude"] <- "x"
names(mapdata)[names(mapdata) == "Latitude"] <- "y"
map <- ggplot() + geom_polygon(data = counties, aes(x = long,
    y = lat, group = group)) + geom_point(data = mapdata, aes(x = x,
    y = y), color = "red")
```

```
## Regions defined for each Polygons
```

```
map + facet_wrap(PERP_RACE ~ ., ncol = 2)
```



Final Comments

> We have seen through these "vignette" how tightly together Tidyverse packages work to import, manipulate and display data. The packages used in this excercise extend the built-in capabilities of R, streamline user workflow and help to get things done quickly.

Useful Links

Mapping in R using the ggplot2 package http://zevross.com/blog/2014/07/16/mapping-in-r-using-the-ggplot2-package/

Political and Administrative Districts - Download and Metadata for New York City https://www1.nyc.gov/site/planning/data-maps/open-data/districts-download-metadata.page