ragis\_data\_wrangle.R

Tom Brailey

2019-12-03

# Regional Autonomy GIS data wrangling  
  
# Setup  
rm(list=ls())  
  
library(magrittr)  
library(ggplot2)  
  
# Load data  
epr <- rio::import(paste0(here::here(), "/data/EPR-2018.1.1.csv"))  
geo\_epr <- rio::import(paste0(here::here(), "/data/GeoEPR-2018.1.1.csv"))  
geom <- sf::st\_read(paste0(here::here(), "/data/GeoEPR-2018.1.1/GeoEPR.shp")) %>%  
 dplyr::select(group, geometry)

## Reading layer `GeoEPR' from data source `C:\Users\Tom Brailey\Dropbox\github\_private\RegionalAutonomyGIS\data\GeoEPR-2018.1.1\GeoEPR.shp' using driver `ESRI Shapefile'  
## replacing null geometries with empty geometries  
## Simple feature collection with 1470 features and 10 fields (with 134 geometries empty)  
## geometry type: GEOMETRY  
## dimension: XY  
## bbox: xmin: -180 ymin: -55.31195 xmax: 180 ymax: 76.99887  
## epsg (SRID): 4326  
## proj4string: +proj=longlat +ellps=WGS84 +no\_defs

mali\_conflict <- rio::import(paste0(here::here(), "/data/1997-01-01-2003-12-31-Mali.csv")) %>%  
 dplyr::mutate(time = ifelse(year < 2000, 0, 1))  
write.csv(mali\_conflict, paste0(here::here(), "/data/mali\_conflict.csv"))  
  
health <- rio::import(paste0(here::here(), "/data/health.xlsx")) %>%  
 dplyr::mutate(`DATE OUVERTURE` = stringr::str\_extract(`DATE OUVERTURE`, "^.{4}"))  
write.csv(health, paste0(here::here(), "/data/health.csv"))  
  
arc\_conf <- dplyr::as\_data\_frame(rio::import(paste0(here::here(), "/data/final\_conflict.xlsx")))

## Warning: `as\_data\_frame()` is deprecated, use `as\_tibble()` (but mind the new semantics).  
## This warning is displayed once per session.

xtable::xtable(arc\_conf)

## % latex table generated in R 3.5.3 by xtable 1.8-4 package  
## % Tue Dec 03 15:45:03 2019  
## \begin{table}[ht]  
## \centering  
## \begin{tabular}{rlrrr}  
## \hline  
## & Autonomy & Total Number of Conflicts & Conflict in Autonomous Region & Conflict in AR as Percentage of Total Conflict \\   
## \hline  
## 1 & Pre-Autonomy & 89.00 & 57.00 & 64.04 \\   
## 2 & Post-Autonomy & 36.00 & 3.00 & 8.33 \\   
## \hline  
## \end{tabular}  
## \end{table}

# Subset data (Explore Variables)  
group\_t <- epr %>%  
 dplyr::filter(reg\_aut == "TRUE") %>%  
 dplyr::distinct(group) %>%  
 dplyr::pull()  
  
group\_f <- epr %>%  
 dplyr::filter(reg\_aut == "FALSE") %>%  
 dplyr::distinct(group) %>%  
 dplyr::pull()  
  
group\_t <- dplyr::as\_tibble(group\_t)

## Warning: Calling `as\_tibble()` on a vector is discouraged, because the behavior is likely to change in the future. Use `tibble::enframe(name = NULL)` instead.  
## This warning is displayed once per session.

group\_f <- dplyr::as\_tibble(group\_f)  
  
group\_aut <- group\_t %>%  
 dplyr::filter(value %in% unique(group\_f$value))  
  
epr\_sub <- epr %>%  
 dplyr::filter(group %in% unique(group\_aut$value))  
  
geo\_epr\_sub <- geo\_epr %>%  
 dplyr::filter(group %in% unique(group\_aut$value)) %>%  
 dplyr::select(-from, -to)  
  
epr\_join <- dplyr::left\_join(epr\_sub, geo\_epr\_sub)

## Joining, by = c("gwid", "statename", "group", "groupid", "gwgroupid", "umbrella")

unique(epr\_join$statename)

## [1] "Canada"   
## [2] "Mexico"   
## [3] "Belize"   
## [4] "Guatemala"   
## [5] "Honduras"   
## [6] "El Salvador"   
## [7] "Nicaragua"   
## [8] "Costa Rica"   
## [9] "Panama"   
## [10] "Colombia"   
## [11] "Venezuela"   
## [12] "Guyana (British Guiana)"   
## [13] "Suriname"   
## [14] "Ecuador"   
## [15] "Peru"   
## [16] "Brazil"   
## [17] "Bolivia"   
## [18] "Paraguay"   
## [19] "Chile"   
## [20] "Argentina"   
## [21] "Uruguay"   
## [22] "United Kingdom"   
## [23] "Belgium"   
## [24] "France"   
## [25] "Spain"   
## [26] "Germany"   
## [27] "Germany Democratic Republic"   
## [28] "Poland"   
## [29] "Austria"   
## [30] "Hungary"   
## [31] "Czechoslovakia"   
## [32] "Czech Republic"   
## [33] "Slovakia"   
## [34] "Italy"   
## [35] "Albania"   
## [36] "Serbia"   
## [37] "Montenegro"   
## [38] "Macedonia"   
## [39] "Croatia"   
## [40] "Serbia and Montenegro"   
## [41] "Bosnia and Herzegovina"   
## [42] "Kosovo"   
## [43] "Slovenia"   
## [44] "Greece"   
## [45] "Cyprus"   
## [46] "Bulgaria"   
## [47] "Moldova"   
## [48] "Romania"   
## [49] "Russia"   
## [50] "Estonia"   
## [51] "Latvia"   
## [52] "Lithuania"   
## [53] "Ukraine"   
## [54] "Belarus"   
## [55] "Armenia"   
## [56] "Georgia"   
## [57] "Azerbaijan"   
## [58] "Finland"   
## [59] "Mali"   
## [60] "Niger"   
## [61] "Nigeria"   
## [62] "Central African Republic"   
## [63] "Chad"   
## [64] "Congo"   
## [65] "Congo, DRC"   
## [66] "Uganda"   
## [67] "Tanzania"   
## [68] "Zanzibar"   
## [69] "Ethiopia"   
## [70] "Angola"   
## [71] "Zambia"   
## [72] "Comoros"   
## [73] "Libya (Tripolitania, Cyrenaica, Fezzan)"  
## [74] "Sudan"   
## [75] "South Sudan"   
## [76] "Iran"   
## [77] "Turkey"   
## [78] "Iraq"   
## [79] "Egypt"   
## [80] "Syria"   
## [81] "Lebanon"   
## [82] "Afghanistan"   
## [83] "Turkmenistan"   
## [84] "Tajikistan"   
## [85] "Kyrgyzstan"   
## [86] "Uzbekistan"   
## [87] "Kazakhstan"   
## [88] "China"   
## [89] "Mongolia"   
## [90] "Democratic People's Republic of Korea"   
## [91] "Republic of Korea"   
## [92] "Japan"   
## [93] "India"   
## [94] "Pakistan"   
## [95] "Myanmar"   
## [96] "Sri Lanka"   
## [97] "Thailand"   
## [98] "Laos"   
## [99] "Vietnam"   
## [100] "Republic of Vietnam"   
## [101] "Philippines"   
## [102] "Indonesia"   
## [103] "Papua New Guinea"

af\_sub <- c("Mali", "Nigeria", "Chad", "Congo, DRC", "Comoros", "Sudan", "Zambia", "Libya (Tripolitania, Cyrenaica, Fezzan)",  
 "Angola", "Ethiopia", "Congo", "Central African Republic", "Niger")  
  
epr\_join\_af <- epr\_join %>%  
 dplyr::filter(statename %in% c(af\_sub)) %>%   
 dplyr::group\_by(statename, from, to, group) %>%  
 dplyr::summarise\_all(dplyr::funs(dplyr::first(na.omit(.)))) %>%   
 dplyr::filter(status != "SELF-EXCLUSION" &  
 !is.na(reg\_aut) &  
 type != "Urban") %>%  
 dplyr::group\_by(statename, group) %>%  
 tidyr::complete(group, from = min(from):2019) %>%  
 tidyr::fill(gwid, groupid, gwgroupid, umbrella, size, status, reg\_aut, sqkm, type, the\_geom) %>%  
 dplyr::select(-to, -the\_geom) %>%  
 dplyr::rename(year = from)

## Warning: funs() is soft deprecated as of dplyr 0.8.0  
## Please use a list of either functions or lambdas:   
##   
## # Simple named list:   
## list(mean = mean, median = median)  
##   
## # Auto named with `tibble::lst()`:   
## tibble::lst(mean, median)  
##   
## # Using lambdas  
## list(~ mean(., trim = .2), ~ median(., na.rm = TRUE))  
## This warning is displayed once per session.

# Create final subsetted data for Mali  
final <- sf::st\_sf(dplyr::left\_join(epr\_join\_af, geom)) %>%  
 dplyr::filter(statename %in% c("Mali"))

## Joining, by = "group"

## Warning: Column `group` joining character vector and factor, coercing into  
## character vector

write.csv(final, paste0(here::here(), "/data/epr\_af.csv"))  
sf::st\_write(final, paste0(here::here(), "/data/epr\_af.shp"), delete\_dsn = TRUE)

## Deleting source `C:/Users/Tom Brailey/Dropbox/github\_private/RegionalAutonomyGIS/data/epr\_af.shp' using driver `ESRI Shapefile'  
## Writing layer `epr\_af' to data source `C:/Users/Tom Brailey/Dropbox/github\_private/RegionalAutonomyGIS/data/epr\_af.shp' using driver `ESRI Shapefile'  
## Writing 240 features with 12 fields and geometry type Unknown (any).

#rm(epr, epr\_join, epr\_sub, geo\_epr, geo\_epr\_sub, group\_aut, group\_f, group\_t, af\_sub)