

Geographic Data Visualization Using Open-Source Data & R

Christopher N. Lawrence

Texas A&M International University - Laredo, Texas

Why Use R?

- Do all of your analysis in the same environment.
- Easy reuse of data from statistical models.
- No need to learn or procure specialized GIS software.
- It's **Free!** (No cost to install, no licensing restrictions.)

Package Options

There are several different add-on packages for R in **CRAN** that meet somewhat different needs.

maps (+ mapdata, mapproj, gmaps)

maptools

The **classInt** and **RColorBrewer** packages are also helpful for selecting color themes for maps.

Simple Maps with **maps**

The **maps** package includes a map of the world, maps of several countries, and county-level maps of U.S. states. The **mapdata** package includes a map of China and a high-resolution world map with national boundaries.

These maps are not projected by default; common projections are included in the **mapproj** package.

maps demonstration

```
# Map showing the % of registered voters in each southern state
# who were black 40 years pre-VRA and 18 years post-VRA

# import the correct map data
library(maps)
library(mapproj)

# this CSV file has lines like:
# "State",1960,1984
# "South Carolina",11,28
# "Mississippi",4,26
black.voter.pct <- read.csv('black-pct-registered.csv')

# create a list that matches up the data
state.to.map <- match.map(map("state", black.voter.pct$State, plot=F),
                          black.voter.pct$State)

# create a single PDF file that includes both maps.
# for inclusion in another document, create separate PDFs for each map with onefile=F.
pdf("black-pct-registered-maps.pdf", height=6, width=8)
palette(gray(40:0/40)) # create a grayscale palette
x <- black.voter.pct$X1960[state.to.map]

# note we use an Albers equal-area projection to make the map look sensible.
# the parameters are the two "true" latitudes that will be the correct length
map("state", black.voter.pct$State, fill=T, col=x, proj='albers', param=c(30,36))
title("Percentage of blacks among registered voters, 1960")

x <- black.voter.pct$X1984[state.to.map]
map("state", black.voter.pct$State, fill=T, col=x, proj='albers', param=c(30,36))
title("Percentage of blacks among registered voters, 1984")
dev.off()
```

Mapping with shapefiles using **maptools**

If R doesn't already include the boundaries you need, **maptools** is the answer. The **maptools** package supports most common formats, including ArcGIS **shapefiles** (.shp) which are the most common these days.

Where to get shapefiles

The U.S. Census Bureau now publishes shapefiles of the United States and its territories for:

- Voting tabulation districts (voting precincts).
- Cities and counties.
- State legislative districts.
- U.S. House districts.

Other sources include GeoBase for Canada and U.S. state GIS repositories.

maptools example

```
# load the right packages
library(maptools)
library(RColorBrewer)

# Orleans Parish VTDs
sppoly <- readShapePoly("tgr22071vot00", IDvar="VOTE00")

# My data is in a separate file with the imaginative name "dat"
load("governor.Rdata")
# Remap the Secretary of State's precinct numbering to the Census Bureau's format
dat$p2 <- gsub('0?([0-9]+) 0?([0-9]+[A-Z]?)', '\\1-\\2', dat$Precinct)

# Match up rows
o <- match(sppoly$VOTE00, dat$p2)
mdat <- dat[o,]
row.names(mdat) <- sppoly$VOTE00
newpoly <- spCbind(sppoly, mdat)

# Set the projection – here we can be simple (note this is really in the sp package)
# see http://www.remotesensing.org/geotiff/proj\_list/
proj4string(newpoly) <- CRS("+proj=longlat +datum=NAD83")

# Now draw away.
pdf(file="votegraphs.pdf", onefile=T, width=7.5, height=9)

greys <- brewer.pal(8, "Blues")
# Here we actually plot four variables together; you could only plot one and make a simpler map!
spplot(newpoly, c("jindalpct", "blackpct", "w.turnout", "b.turnout"),
  names.attr=c("Jindal vote", "% of voters black",
  "White turnout %", "Black turnout %"),
  main="Orleans Parish Vote, 2007 Gubernatorial Election",
  col.regions=greys, cuts=length(greys)-1,
  xlim=c(-90.15,-89.90), ylim=c(29.9,30.1), as.table=T)
dev.off()
```


Challenges

Matching your data to the names used in the GIS data is critical.

For large areas (states, countries, or the world), you need to choose an appropriate projection.

Unprojected maps look “funny” even to the untrained viewer.

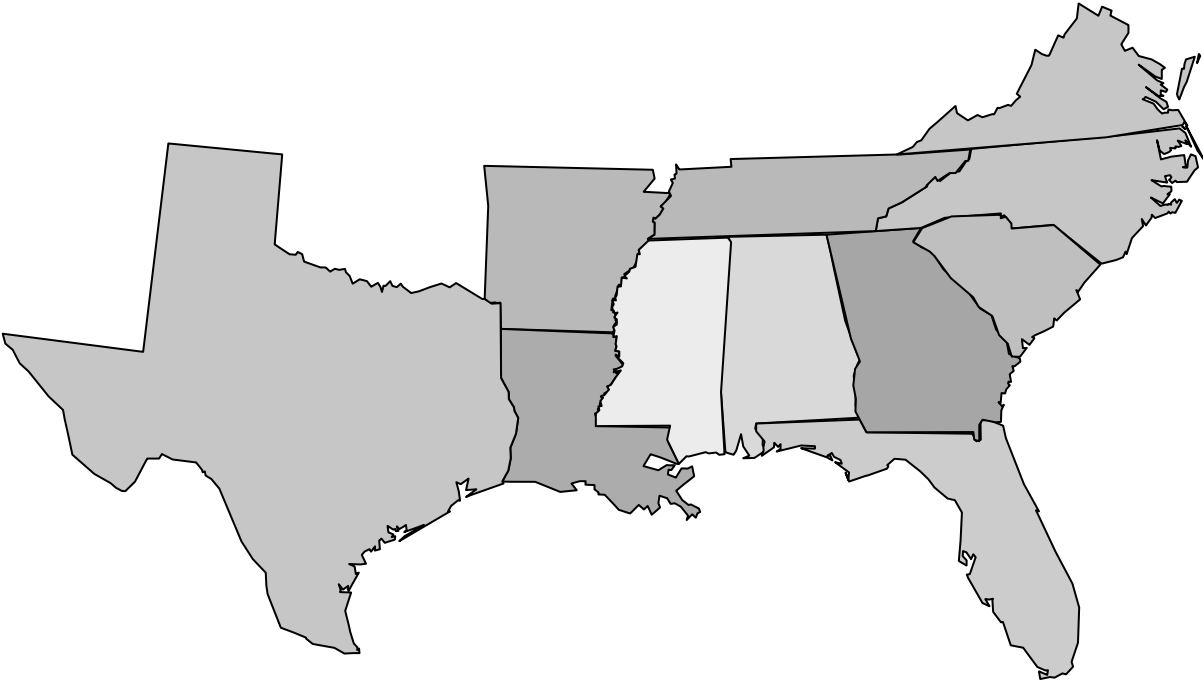
Selecting a good set of colors or gray scales for printed or projected work can also be challenging; the **RColorBrewer** package is designed to help.

Other Advice

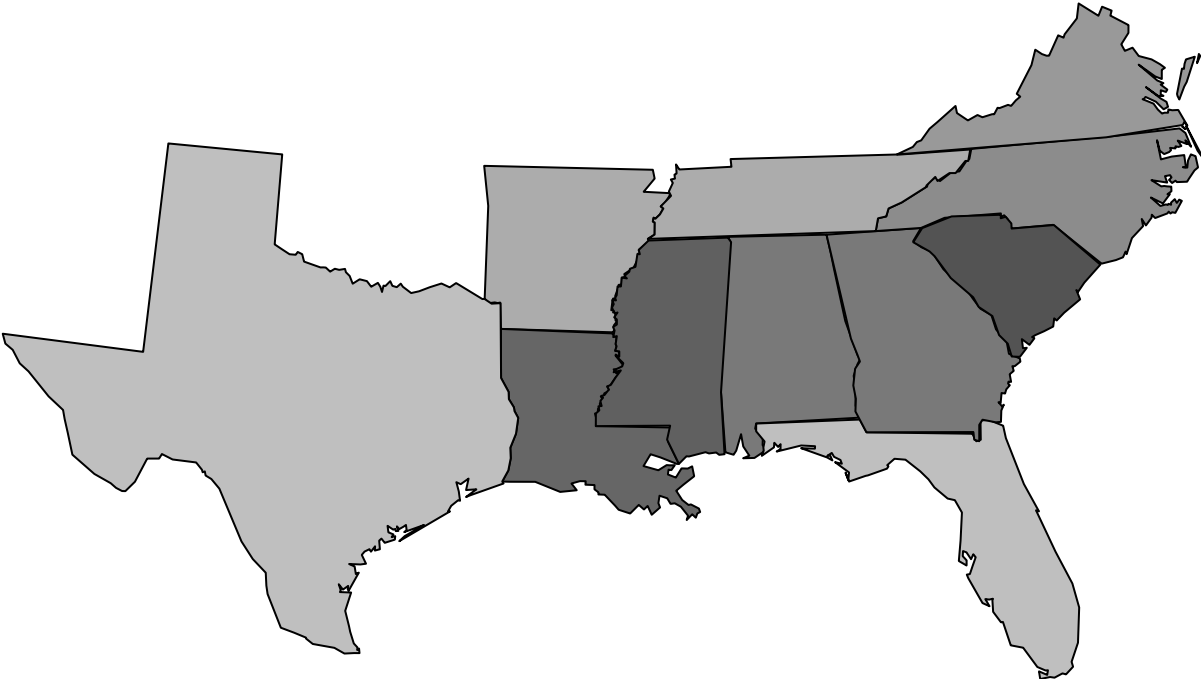
In both of the examples above, I produced PDF files rather than writing to the screen. You will usually get better output for use in other documents by producing PDF (useful for xelatex/pdflatex) or SVG files rather than saving from an R graphics window.

As maps are vector format, writing to a GIF or PNG file is not recommended unless you are creating a map for the web.

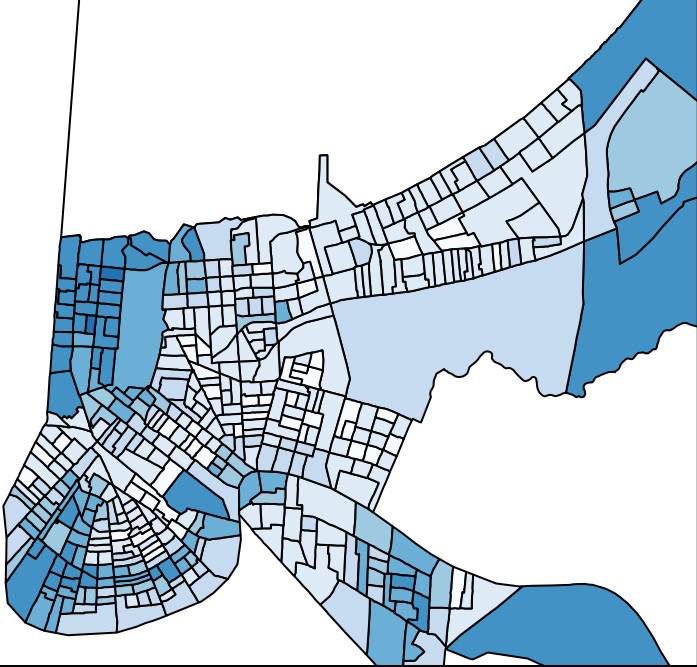
Percentage of blacks among registered voters, 1960



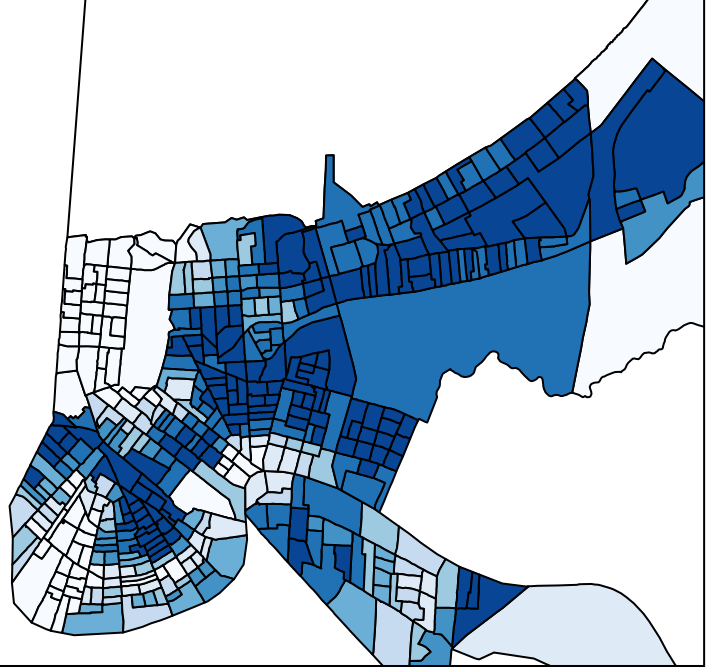
Percentage of blacks among registered voters, 1984



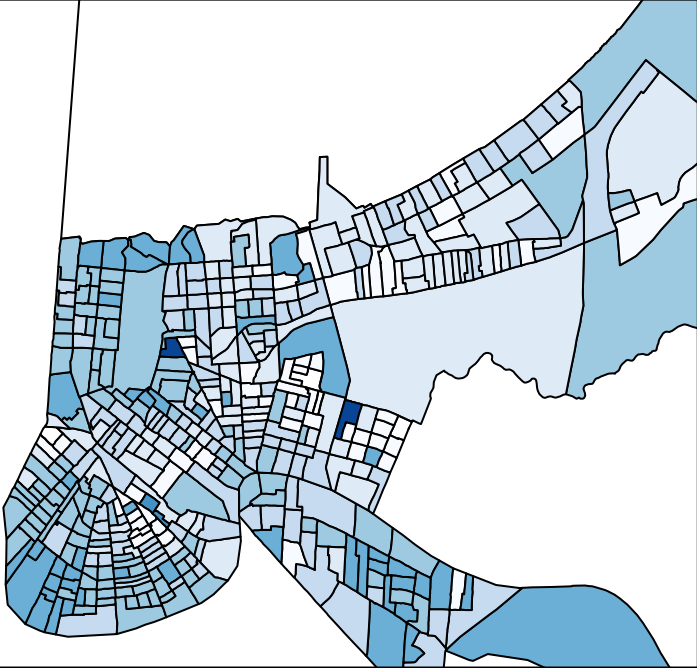
Jindal vote



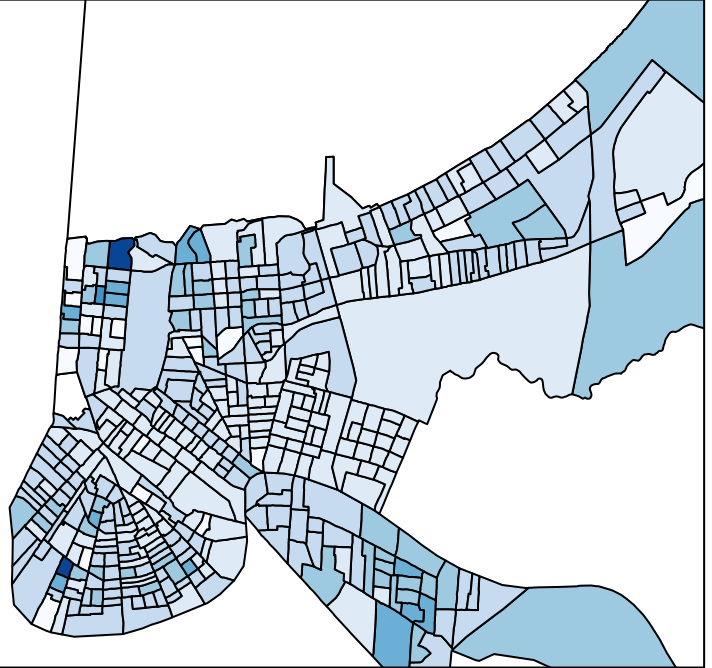
% of voters black



White turnout %



Black turnout %



1.0

0.8

0.6

0.4

0.2

0.0