Interim progress - Geog 701

Robert Chastain

24 March, 2021

What is your progress to date on getting the needed data / layers for your project? Provide a brief narrative.

I planned on using proprietary date from my employer for my Geo-AVM (Automated Valuation Model) project, unfortunately they were unwilling to share the data. As per your suggestion I was able to source the housing data directly from Washoe County through their "Washoe GIS Open Data". This database contains most of the housing related data I needed (although their data dictionary leaves something to be desired).

I still need to source the zoning data, topographic index and noise data (also as per your suggestion). I've considered tax and demographic data, however I don't want to offer any sort of appearance of violation of the Fair Housing Act so I've decided to leave that data out.

What difficulties (if any) have you run into in what you had initially planned?

I initially tried to source this data through the API provided by the Washoe County website, however I encountered complications around the formatting of the data. To save myself a headache I ended up simply downloading the data manually. I'd like to revisit this approach to data collection if I have time to improve reproducibility of my project.

Another significant source of frustration came from converting my data from a Mercator projection to a latitude/longitude projection. This seems to be user error; after hours of banging my head against my keyboard I was able to solve this problem using the code provided in class:

```
## Subset the Washoe county parcel data to a smaller region for visualization
reno_shp <- washoe_shp[washoe_shp@data$REGION == 'GOLDN', ]

## Set the correct CRS
crs4326 =  CRS("+init=epsg:4326")

## Transform the data
reno_4326 <- spTransform(reno_shp, crs4326)
```

I originally was using `CRS("+init=epsg:26911")` and the transformation placed my map in India.

Aside from the user error, I'm encountering some issues rasterizing my data, however I'm confident I'll be able to get this problem sorted out soon.

What changes to your project have you made in response to input in D1 and/or question 2 above?

The largest change in my project stems from the loss of my primary data source; because I don't have access to cleaned and sanitized data I'm now doing a lot more data munging than I'd anticipated. This may end up benefiting my project. I had originally planned to use only the location and value for each house, but the data from the county is significantly more detailed including characteristics of each home. Given this, I now plan to have a more detailed set of training data.

What environmental data will you likely be using, and why are you choosing these layers?

For my project I plan to use flood maps, a topographic index, and a noise map (as suggested). I'm considering other environmental variables such as rain or temperature, however I'm not entirely sure that these features are actually informative of housing prices at the small scale I'm working with. It does seem like these variables would introduce a lot of noise (for example, there is significantly more rain in the western part of Reno, but I doubt the extra rain is a driver of housing prices).

What obstacles might you still face in completing your intended project?

From a coding perspective I anticipate having trouble generating the raster stack given my data comes from several disparate sources. Aside from that, I'm slightly worried about computational time associated with this project, however I should be able to multi-thread a lot of the work load. If that still proves insufficient I'm comfortable using the computational resources made available by the university (Okapi in particular).

Provide some plots/graphics of your progress to date.

I'd like to present my project as an interactive R dashboard so I've been using leaflet for most of my visualizations. I may revert to ggplot2, plotly, or base graphics.

A code snippet used to generate the following graphics:

```
## Required libraries
library(rgdal)
library(raster)
library(sp)
library(sf)
library(leaflet)

## Load the shape file for parcels in Washoe county
washoe_file = '_data/washoe_shp.RData'
if (file.exists(washoe_file)) {
  load(washoe_file)
} else {
  washoe_shp <- rgdal::readOGR(dsn = '_data/washoe_county', layer = 'Parcels')
  save(washoe_shp, file = washoe_file)
}

## Remove NAs from 'REGION' in Washoe county data so that the spatial data frame can be subset by regio
washoe_shp <- washoe_shp[!is.na(washoe_shp@data$REGION), ]

## Subset the data to only the "Golden Valley" region for easier visualization
reno_shp <- washoe_shp[washoe_shp@data$REGION == 'GOLDN', ]

## Set appropriate projection so that the polygons show up in the right spots on the map tiles
crs4326 =  CRS("+init=epsg:4326")
reno_4326 <- spTransform(reno_shp, crs4326)

## Get the bounding box for the smaller region; this will be used to center the map on the desired regi
center <- bbox(reno_4326)

## Create a raster from the shape file
reno_raster <- raster(reno_4326)
reno_rooms <- rasterize(x = reno_4326, y = reno_raster, field = 'BEDROOMS')
color_pal <- colorBin("Blues", values(reno_rooms))
```

The first graphic contains the parcel polygons for selected region:

```
leaflet() %>%
  setView(lng = mean(center[1,]), lat = mean(center[2,]), zoom = 14) %>%
  addRasterImage(reno_rooms, opacity = 0.5) %>%
  addPolygons(data = reno_4326, weight = 2, opacity = 0.2) %>%
  addTiles()
```

```
## PhantomJS not found. You can install it with webshot::install_phantomjs(). If it is installed, pleas
```

The second graphic contains the parcel polygons with the bedroom count raster layer:

```
leaflet() %>%
  setView(lng = mean(center[1,]), lat = mean(center[2,]), zoom = 14) %>%
  addRasterImage(reno_rooms, opacity = 0.5) %>%
  addPolygons(data = reno_4326, weight = 2, opacity = 0.2) %>%
  addLegend(pal = color_pal, values = values(reno_rooms), title = 'Bedroom Count') %>%
  addTiles()
```

As you can see, this part is not going well.