GEOG 491/891: Special Topics - Spatial Analysis in R

Week 10.01: Rasters

Dr. Bitterman

Today's schedule

- Open discussion
- Raster basics

Anything to discuss? Questions?

Remaining topics

- Week 10: Rasters (Friday: update presentations)
- Week 11: Making maps (Intro lab 4)
- Week 12: Interactive mapping (Intro lab 5)
- Week 13: Applications
- Week 14: Thanksgiving week
- Week 15: Applications
- Week 16: Project presentations

Raster data structure review

1. What's a raster dataset?
2. How are they used?
3. What are the key properties of a raster?
4. Any other concerns (e.g., toppolgy) that you're aware of?

Today's setup

```
library(tidyverse)
library(raster)
library(tmap)
```

What do you notice when you load these packages?

Let's read a raster image

```r
myras <- raster::raster("./data/ts_2016.1007_1013.L4.LCHMP3.CIcyano.MAXIMUM_7day.tif")
```

Plot it

```r
plot(myras)
```

Basic properties

```
# properties
myras

raster::extent(myras)
raster::nbands(myras)
```

What's the data structure of a raster?

Getting the value by index

```
#[index]
myras[1]

myras[31225]
```

Or by row, column

```
#[row, column]
myras[600, 175]
```

Two questions:

1. How is "single indexing" different than row, column indexing?
2. For row, column indexing, what other information is useful/required to know what you're doing?

Frequency of values

```
raster::freq(myras)
```

what's the output?

Let's use it to make a quick histogram

Who wants to break down this code function-by-function, parameter-by-parmeter?

```
# quick histogram
myras %>% raster::freq() %>% data.frame() %>%
  ggplot(., aes(x = value, y = count)) +
  geom_bar(stat = "identity")
```

Was the plot useful? Why/why not?

Let's try again

```r
# filter out the 252 (no data) values
myras %>% raster::freq() %>% data.frame() %>%
  dplyr::filter(value < 252) %>%
  ggplot(., aes(x = value, y = count)) +
  geom_bar(stat = "identity")
```

Better?

Another way to "get" cell values

```r
# get all the values
myras %>% raster::values()
```

Raster aggregation

Why might we want to change the resolution of a raster?

Break it down

```r
raster::aggregate(myras, 2, fun = max)
```

What happened?

More obvious comparisions

```r
raster::aggregate(myras, 2, fun = max) %>% plot()
raster::aggregate(myras, 5, fun = max) %>% plot()
```

Different functions --> different results

```r
raster::aggregate(myras, 5, fun = max) %>% plot()
raster::aggregate(myras, 5, fun = mean) %>% plot()
```

Data conversions

Turning cells into points

```
myras %>% raster::rasterToPoints()
```

What's the data structure returned?

We can also vectorize/polygon-ize

Break it down again - what do we expect the output to be?

```
# vectorize
poly1 <- rasterToPolygons(myras, dissolve = T)

tmap_mode("view")
tm_shape(poly1) + tm_polygons()
```

What was the result?

For this week

- Wedneday: Raster math
- Friday: in-class update presentations… upload them to Canvas by the due date
- Readings posted on Canvas
- Practice, practice, practice
- Work on your projects