



ENVIRONMENTAL DATA ANALYTICS: M9 – SPATIAL ANALYSIS

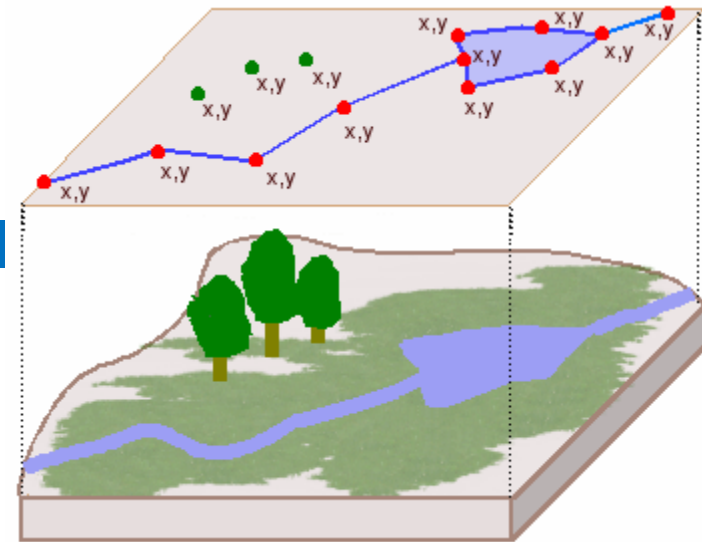
Fall 2023

Nicholas School of the Environment - Duke University

M9.1 – Spatial Data

- Importance of Spatial Analysis in EDA
- Representing Spatial Data in R
- Simple Features & Spatial Dataframes
- Creating geometries
- Coordinate Reference Systems, Transformations, EPSG codes
- GeoJSON, Shapefiles, ...

Simple Features...



```
## Simple feature collection with 100 features and 6 fields
## geometry type: MULTIPOLYGON
## dimension: XY
## bbox: xmin: -84.32385 ymin: 33.88199 xmax: -75.45698 ymax: 36.58965
## epsg (SRID): 4267
## proj4string: +proj=longlat +datum=NAD27 +no_defs
## precision: double (default; no precision model)
```

First 3 features:

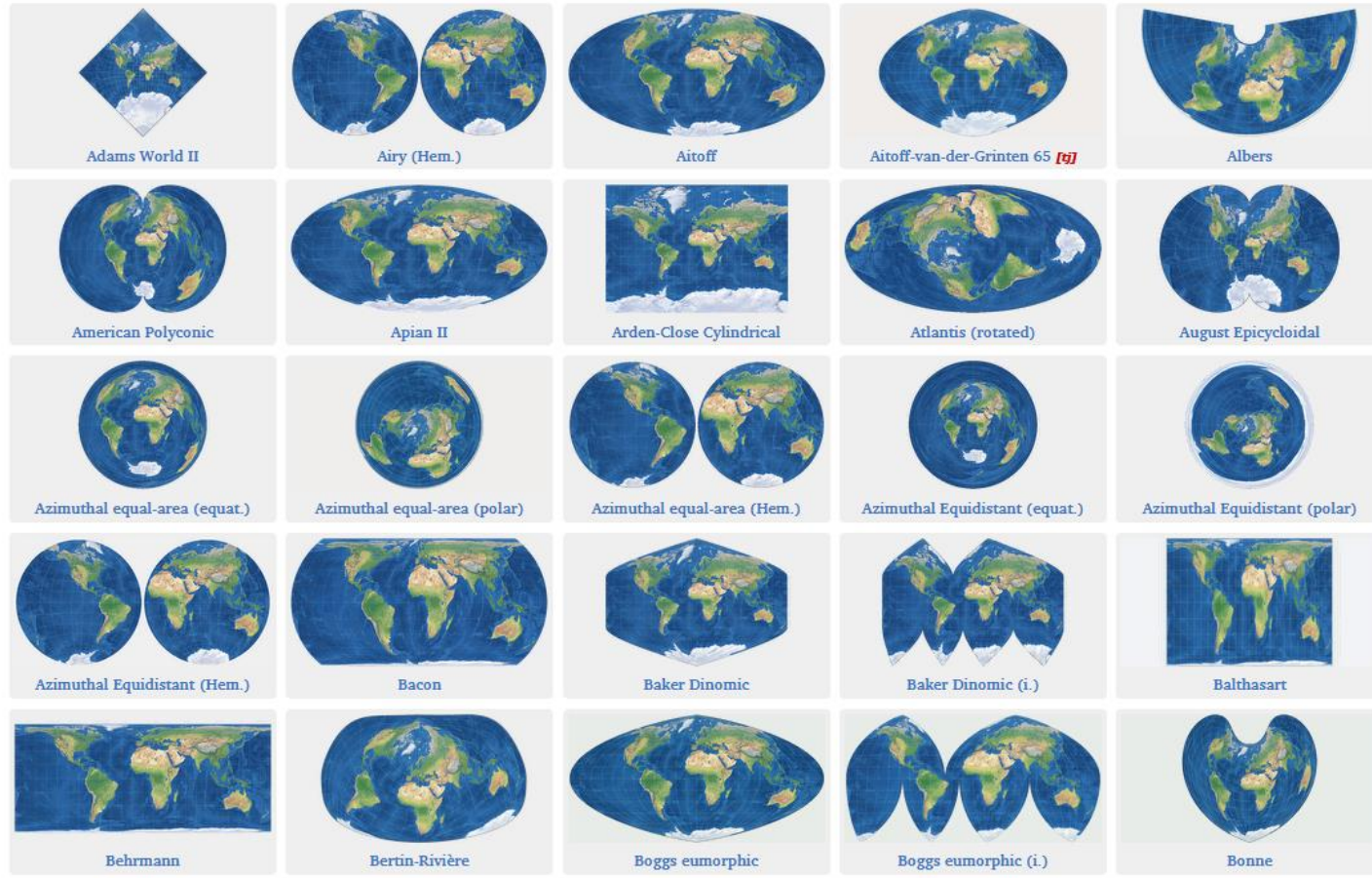
##	BIR74	SID74	NWBIR74	BIR79	SID79	NWBIR79	geom
## 1	1091	1	10	1364	0	19	MULTIPOLYGON(((-81.47275543...
## 2	487	0	10	542	3	12	MULTIPOLYGON(((-81.23989105...
## 3	3188	5	208	3616	6	260	MULTIPOLYGON(((-80.45634460...

Simple feature

Simple feature geometry list-column (sfc)

Simple feature geometry (sfg)

Coordinate Reference Systems



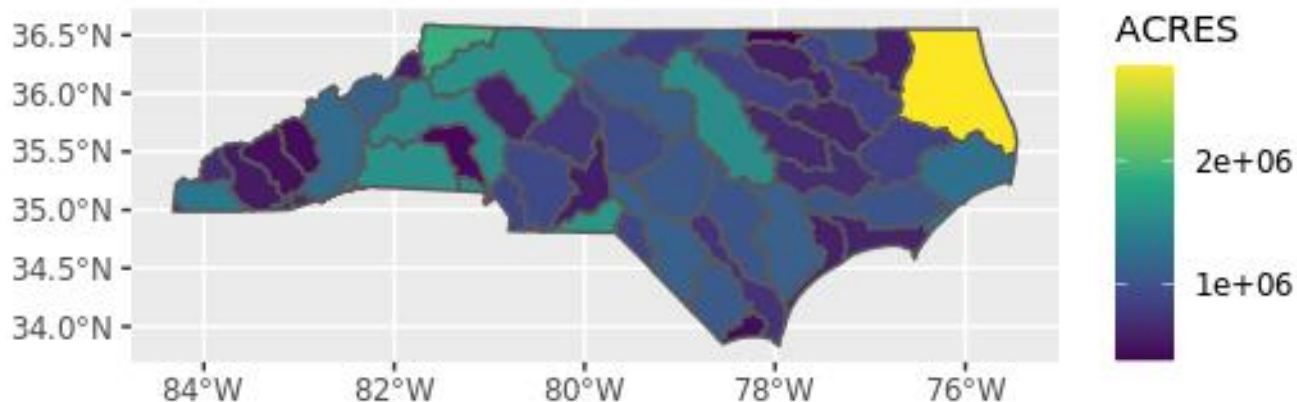
<https://www.nceas.ucsb.edu/sites/default/files/2020-04/OverviewCoordinateReferenceSystems.pdf>

M9.2 – Spatial Analysis

- Reading spatial data into R `st_read()`
- Attribute joins `left_join(), inner_join(), ...`
- Spatial aggregation `group_by() & summarize()`
- Coordinate system transformations `st_transform()`
- Intersecting data `st_filter(...,.pred)`
- Clipping data `st_intersection()`
- Spatial Selection `st_intersects()`

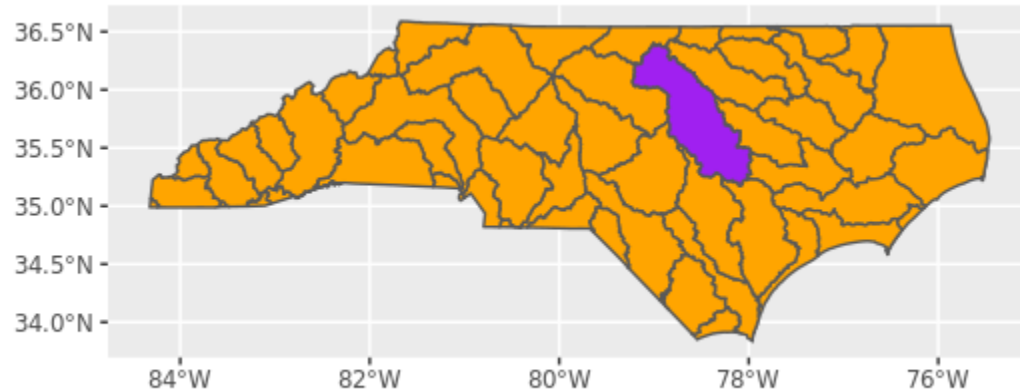
Exercise 2.2.1

- Read in the NC HUC-8 shapefile & filter for :
`./Data/Spatial/NCHUC8.shp`
- View features, colored by ACRES



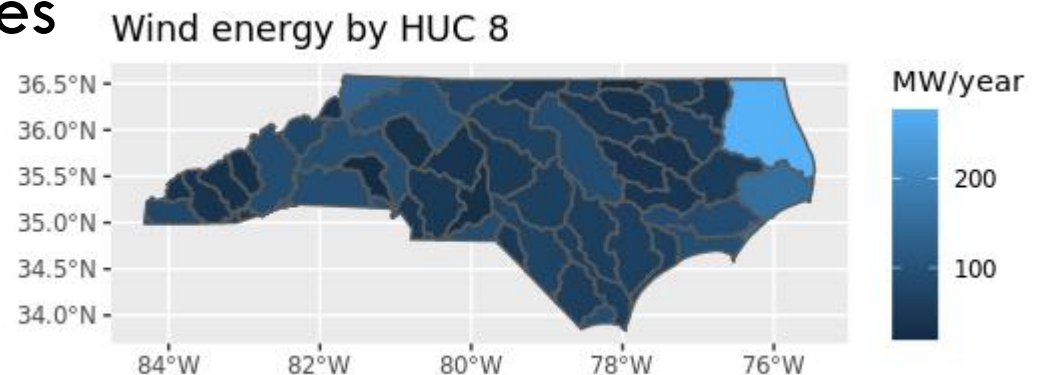
Exercise 2.2.2 - Challenge

- Read in the NC HUC-8 shapefile & *filter for SUBBASIN is "Upper Neuse"*
- View all HUCs in orange, Upper Neuse in purple



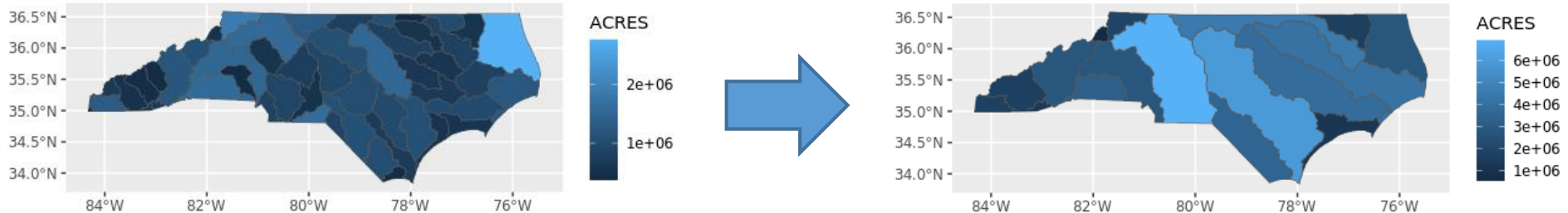
Exercise 3.1.1

- Read an online CSV file into a dataframe\
https://raw.githubusercontent.com/ENV859/EnviroAtlasData/main/Wind_Energy.csv
 - ▣ Set `HUC12` column to be a factor (*colClasses*)
 - ▣ Compute `HUC8` from `HUC12` (*substr*)
 - ▣ Group on `HUC8`
 - ▣ Compute sum of AvgWindEnergy for each HUC8
- Join to HUC8 features



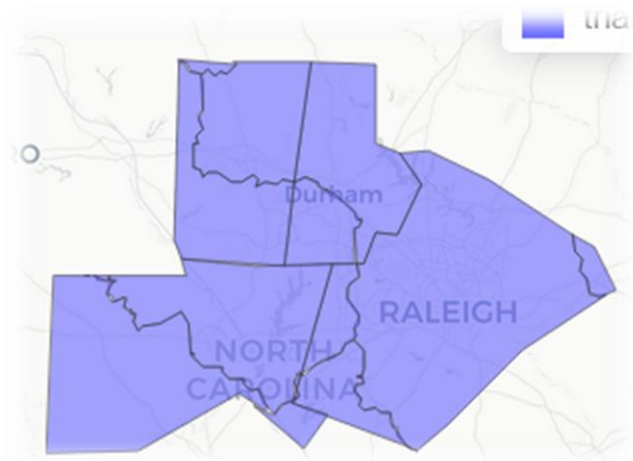
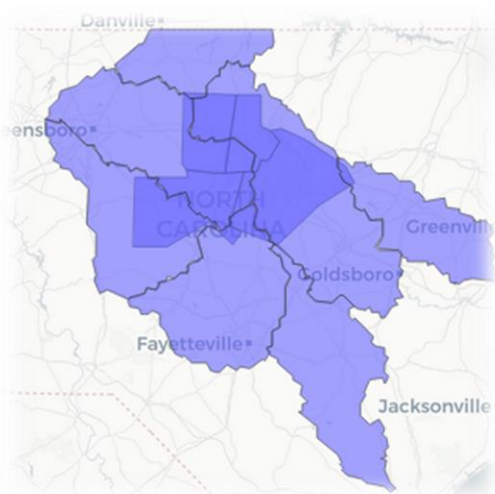
Exercise 3.1.2

- Aggregate HUC 8 to HUC 6 (“DWQ Sub-basin”)
 - Compute aggregate area in acres and square miles



Exercise 3.3.1

- Select Triangle counties from all counties:
Chatham, Durham, Orange, and Wake
- Select HUC8s that intersect the Triangle counties
- Intersect (clip) the HUC8 areas falling w/in Triangle



UPDATE 3.3

```
369 #Select intersecting counties using the `filter()` command
370 neuse_intersect_2 <- counties_sf_utm %>%
371   filter(st_intersects(x = ., y = neuse_sf, sparse = FALSE))
```



```
369 #Select intersecting counties using the `filter()` command
370 neuse_intersect_2 <- counties_sf_utm %>%
371   st_filter(neuse_sf)
```

```
neuse_intersect_3 <- counties_sf_utm[
  st_is_within_distance(
    x=counties_sf_utm,
    y=neuse_sf,
    dist=10000,
    sparse=F),]
```

*Creates a non-sparse matrix
which can be used as a filter to
subset records in the counties
sf object*

A NOTE ON KNITTING...

*Maps created with MapView and Leaflet
don't knit well to PDF!*

□ **Solutions:**

- Use ggplot...
- Knit to HTML...

M9.3 – Spatial Data Visualization

- `ggplot() + geomsf()`
- `mapview`
- `leaflet`



Solutions...

2.2.2 Read HUCs shapefile

```
118 ▾ ~~~{r read HUCs shapefile}
119 #Read the shapefile into an sf dataframe named "huc8_sf"
120 huc8_sf <- st_read('./Data/Spatial/NCHUC8.shp')
121
122 #Reveal the columns
123 colnames(huc8_sf)
124
125 #Check the CRS
126 st_crs(huc8_sf)
127
128 #Examine some records
129 head(huc8_sf)
130
131 #View the data as a map, colored by the acreage in each
132 ggplot() +
133   geom_sf(data=huc8_sf, aes(fill=ACRES))+
134   scale_fill_distiller(palette = "YlOrRd")
135
136 mapview(huc8_sf,zcol='ACRES')
137 ▲ ~~~
```

2.2.3 Challenge

```
143 ▾ ```{r Select the Upper Neuse HUC 8}
144 #Read the shapefile into an sf dataframe
145 neuse_sf <- st_read('./Data/Spatial/NCHUC8.shp') %>%
146   filter(SUBBASIN == 'Upper Neuse')
147
148 #Create a map
149 ggplot() +
150   geom_sf(data=huc8_sf,fill='purple',color='gray') +
151   geom_sf(data=neuse_sf,fill='orange')
152
153 mapview(huc8_sf,col.regions='purple') +
154   mapview(neuse_sf,col.regions='orange',alpha=1)
155 ▸ ```
```


3.1 Join wind data to HUC8s

```
249 ~~~{r}
250 #Compute HUC8 wind energy
251 wind_raw <- read.csv('https://raw.githubusercontent.com/ENV859/EnviroAtlasData/main/Wind_Energy.csv',
252   colClasses = c('HUC_12' = 'factor')) %>%
253   mutate(HUC_8 = substr(HUC_12,1,8)) %>%
254   group_by(HUC_8) %>%
255   summarize(SumWindEnergy = sum(AvgWindEnergy))
256
257 #Join to HUC_8 features
258 huc8_sf_join = merge(x = huc8_sf,
259   y = wind_raw,
260   by.x = 'HUC_8',
261   by.y = 'HUC_8')
262
263 #View the outputs
264 ggplot(data=huc8_sf_join) +
265   geom_sf(aes(fill=SumWindEnergy)) +
266   labs(
267     title='Wind energy by HUC 8',
268     fill = 'MW/year'
269   )
270
271 ~~~
```

3.1 Aggregate HUC-8s to HUC-6s

```
291 ~~~{r Aggregate the HUC data on an attribute, saving as huc2_sf}
292 #List the unique values in the DWQ_Basin field
293 unique(huc8_sf$DWQ_Basin)
294
295 #Summarize on DWQ Basin value
296 huc2_sf <- huc8_sf %>%
297   group_by(DWQ_Basin) %>%
298   summarize(
299     ACRES = sum(ACRES),|
300     SQ_MILES = sum(SQ_MILES)
301   )
302
303 #Map the data
304 mapview(huc2_sf,zcol='ACRES')
305
306 ~~~
```

3.3 Select HUC8s in the Triangle

```
385 ▾ ```{r EXERCISE: Clipping}
386 #Select the Triangle County from the
387 triCo_sf <- counties_sf_utm %>%
388   filter(NAME %in% c('Chatham', 'Durham', 'Orange', 'Wake'))
389
390 #Grab the intersecting HUC_8s
391 triHucs_1 <- huc8_sf_utm[triCo_sf,]
392 mapview(triHucs_1)
393
394 triHucs_2 <- huc8_sf_utm %>%
395   st_filter(triCo_sf)
396 mapview(triHucs_2)
397
398 #Intersect the HUC_8s
399 truHucs_clipped = huc8_sf_utm %>%
400   st_intersection(triCo_sf)
401 mapview(truHucs_clipped)|
402 ▸ ```
```

3.4 Select counties w/in 30 km of pt

```
```{r Select counties within a 30 km area from the site}
the_site_counties <- theSite_sfc_transformed %>%
 st_buffer(.,dist=30000)

counties_3k <- counties_sf_utm %>%
 st_filter(the_site_counties)

mapview(counties_3k) +
 mapview(the_site_counties,col.regions='red',alpha=0.1)
```
```