16: Mapping

Hydrologic Data Analysis | Kateri Salk

Fall 2019

Lesson Objectives

1. Discuss available datasets for watershed mapping
2. Create maps of catchments and water features in R
3. Analyze and communicate the findings of spatial analysis using mapping tools

Opening Discussion

What are the big ideas from last class about mapping?

Session Set Up

```
getwd()
```

```
## [1] "/Users/katerisalk/Box Sync/Courses/Hydrologic Data Analysis/Lessons"
```

```
library(tidyverse)
library(sf)
library(viridis)

theme_set(theme_classic())
```

Spatial datasets: line and polygon objects

Today, we will be working with the USA rivers shape file (see metadata file in README folder) and the Watershed Boundary Dataset, accessed from USGS here: https://viewer.nationalmap.gov/basic/?basemap=b1&category=nhd&title=NHD%20View. We will be using the dataset for the HUC 2 designation 03, which represents the South Atlantic-Gulf region. Additional shapefiles for other regions are available at the link.

Water features line objects

```
waterfeatures <- st_read("../Data/Raw/hydrogl020.dbf")
```

```
## Reading layer `hydrogl020' from data source `/Users/katerisalk/Box Sync/Courses/Hydrologic Data Analy
## Simple feature collection with 76975 features and 12 fields
## geometry type:  LINESTRING
## dimension:      XY
## bbox:           xmin: -179.9982 ymin: 17.67469 xmax: 179.9831 ymax: 71.39819
## epsg (SRID):    NA
## proj4string:    NA
```

```
class(waterfeatures)
```

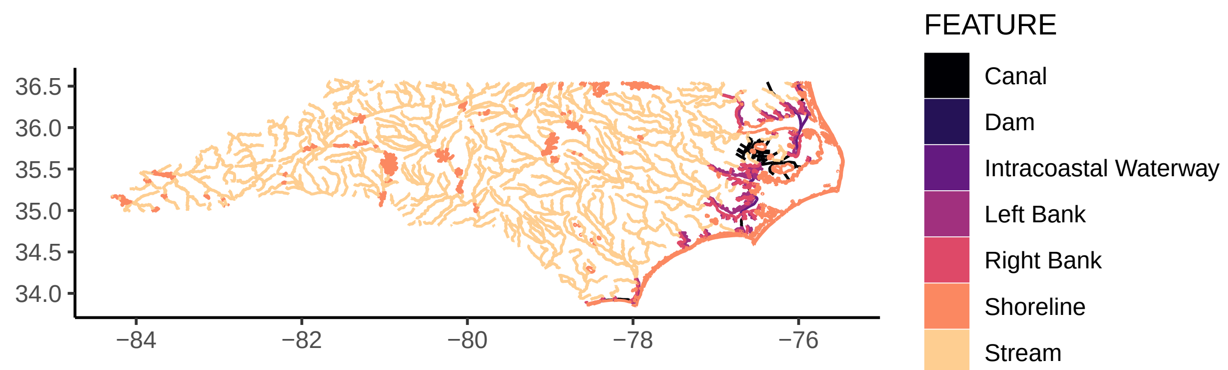
```
## [1] "sf"         "data.frame"
```

```r
# Filter for North Carolina
waterfeatures <- filter(waterfeatures, STATE == "NC")

# Remove a couple feature types we don't care about
waterfeatures <- filter(waterfeatures, FEATURE != "Apparent Limit" & FEATURE != "Closure Line")
```

The .dbf file extension represents an ESRI shapefile. From the `st_read` function, we can see what the bounding box is around the spatial objects as well as the projection.

```r
Waterfeaturesplot <-
ggplot(waterfeatures) +
  geom_sf(aes(fill = FEATURE, color = FEATURE)) +
  scale_color_viridis_d(option = "magma", end = 0.9) +
  scale_fill_viridis_d(option = "magma", end = 0.9)
print(Waterfeaturesplot)
```



Watershed boundary polygons

We will upload the HUC6 watershed designations for North Carolina. Note that in the Watersheds_Spatial folder, there are many different shape file datasets going from HUC2 (largest) to HUC16 (smallest) and NWIS drainage lines. These are all downloaded as zip from the Watershed Boundary Dataset (WBD).

```r
HUC6 <- st_read("../Data/Raw/Watersheds_Spatial/WBDHU6.dbf")
```

```
## Reading layer `WBDHU6' from data source `/Users/katerisalk/Box Sync/Courses/Hydrologic Data Analysis/
## Simple feature collection with 33 features and 15 fields
```

```
## geometry type:  MULTIPOLYGON
## dimension:      XY
## bbox:           xmin: -90.6235 ymin: 24.39533 xmax: -75.3981 ymax: 37.52103
## epsg (SRID):    4269
## proj4string:    +proj=longlat +datum=NAD83 +no_defs
```

```r
summary(HUC6$States)
```

```
##       AL    AL,FL AL,FL,GA AL,GA,TN AL,LA,MS    AL,MS       FL    FL,GA
##        1        3        1        1        1        2        6        4
##       GA GA,NC,SC    LA,MS       NC    NC,SC NC,SC,VA    NC,VA       SC
##        2        1        1        4        2        1        2        1
```

```r
HUC6.NC <- HUC6 %>%
  filter(States %in% c("GA,NC,SC", "NC", "NC,SC", "NC,SC,VA", "NC,VA"))
```

Generating maps with mixed features

Let's plot the watershed polygon features underneath the water features. Note that both datasets need to use the same projection in order to be plotted on the same ggplot. In more complex cases, you may want to set the proj4string components to be equal as well, and there is dummy code below (commented out) to show what that process looks like.

Notice that we chose to use the ColorBrewer "paired" palette here (a true qualitative palette), as it would have been difficult to distinguish the continuous viridis color palettes across 10 watersheds.

```r
st_crs(waterfeatures)
```

```
## Coordinate Reference System: NA
```

```r
st_crs(HUC6.NC)
```

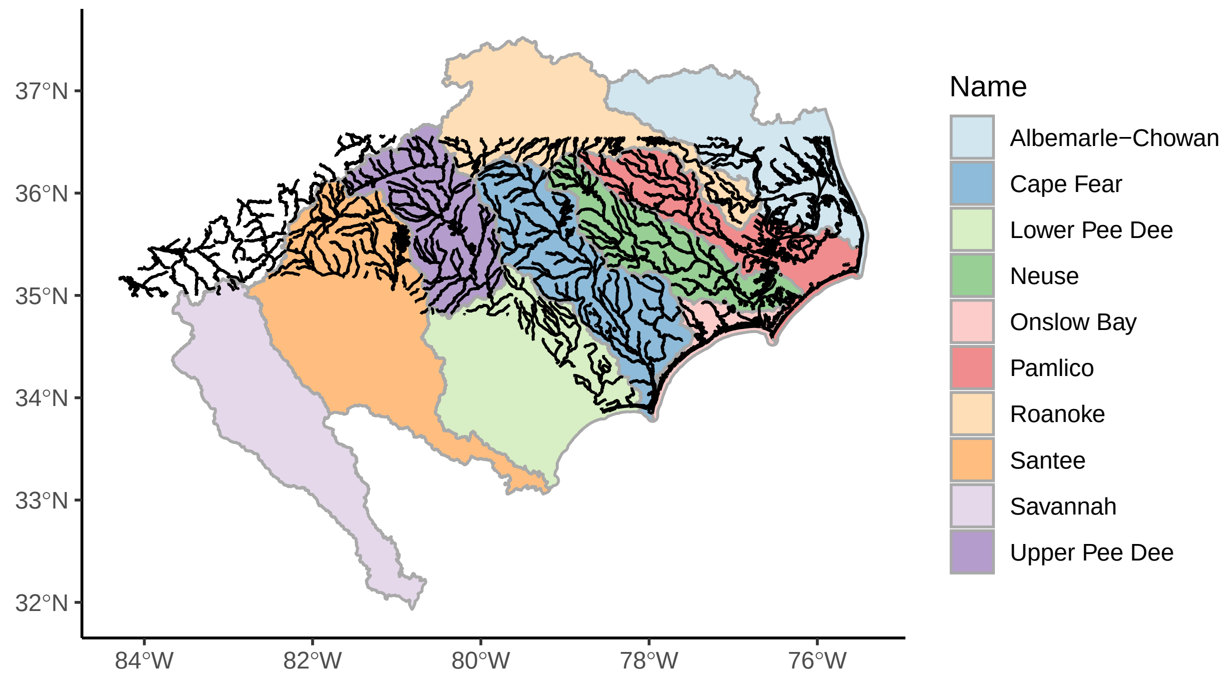
```
## Coordinate Reference System:
##   EPSG: 4269
##   proj4string: "+proj=longlat +datum=NAD83 +no_defs"
```

```r
waterfeatures <- st_set_crs(waterfeatures, 4269)
st_crs(waterfeatures)
```

```
## Coordinate Reference System:
##   EPSG: 4269
##   proj4string: "+proj=longlat +ellps=GRS80 +towgs84=0,0,0,0,0,0,0 +no_defs"
```

```r
#waterfeatures <- waterfeatures %>% st_set_crs(st_crs(HUC6.NC))

NClayers <- ggplot() +
  geom_sf(data = HUC6.NC, aes(fill = Name), color = "darkgray", alpha = 0.5) +
  geom_sf(data = waterfeatures) +
  scale_fill_brewer(palette = "Paired")
print(NClayers)
```

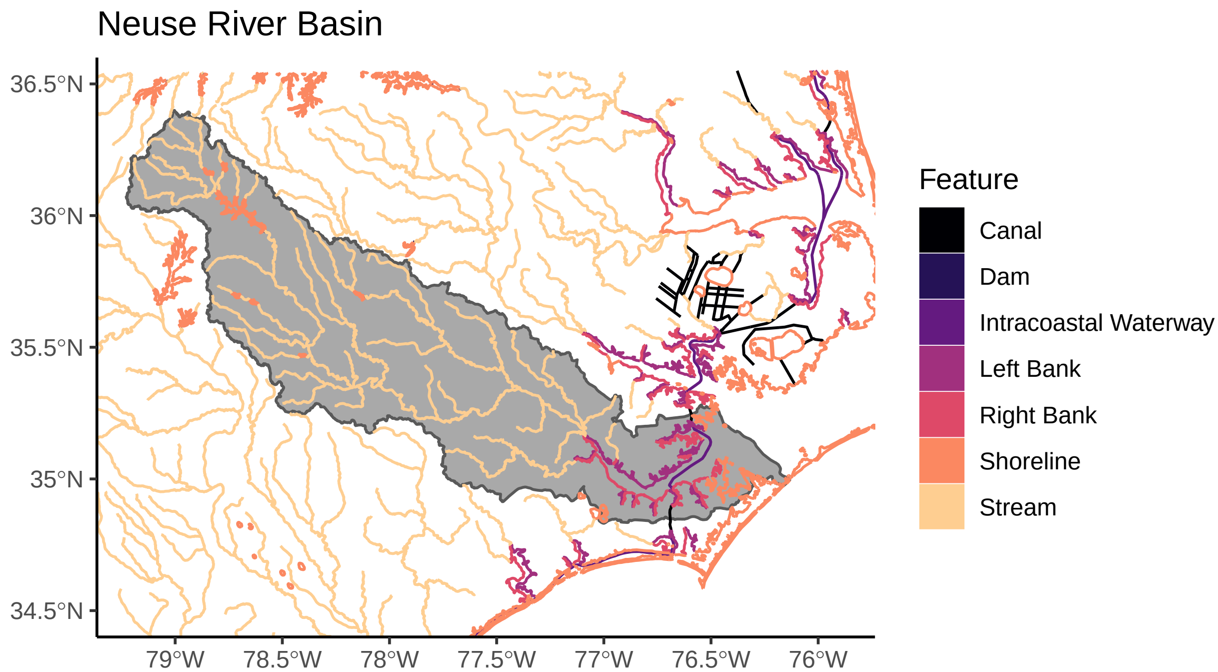


Let's say we wanted to zoom in on the Neuse River basin. For today, we will place a bounding box on the map, but there are more complex ways to "clip" spatial dataests to only map onto specific boundaries. This can be particularly helpful when you are working with multiple spatial objects that do not have a common column from which to filter.

More information here: https://geocompr.robinlovelace.net/

```
HUC6.Neuse <- filter(HUC6.NC, Name == "Neuse")

Neusebasin <- ggplot() +
  geom_sf(data = HUC6.Neuse, fill = "darkgray") +
  geom_sf(data = waterfeatures, aes(fill = FEATURE, color = FEATURE)) +
  xlim(c(-79.2, -75.9)) +
  ylim(c(34.5, 36.5)) +
  labs(title = "Neuse River Basin", color = "Feature", fill = "Feature") +
  scale_color_viridis_d(option = "magma", end = 0.9) +
  scale_fill_viridis_d(option = "magma", end = 0.9)
print(Neusebasin)
```



What features are prevalent in the Neuse River basin? Where do certain features occur in the hydrologic network? How does this distribution compare to nearby basins?

Closing Discussion

What could you do with the catchment and water features datasets as base layers for an analysis of physical or water quality data?