Assignment 9: Spatial Analysis

Emily McNamara

OVERVIEW

This exercise accompanies the lessons in Environmental Data Analytics on spatial analysis.

Directions

1. Use this document to create code for a map. You will **NOT** be turning in the knitted Rmd file this time, only the pdf output for a map.
2. When you have produced your output, submit **only** the pdf file for the map, without any code. Please name your file "StudentName_A09_Spatial.pdf".

The completed exercise is due on Thursday, March 19 at 1:00 pm.

Create a map

You have three options for this assignment, and you will turn in just **one** final product. Feel free to choose the option that will be most beneficial to you. For all options, to earn full points you should use best practices for data visualization that we have covered in previous assignments (e.g., relabeling axes and legends, choosing non-default color palettes, etc.).

Here are your three options:

1. Reproduce figure 1b from the spatial lesson, found in section 3.2.2. You may choose a state other than North Carolina, but your map should still contain the spatial features contained in figure 1b in the "img" folder.

2. Create a new map that mixes spatial and tabular data, as in section 3.3 of the spatial lesson. You may use the maps created in the lesson as an example, but your map should contain data other than precipitation days per year. This map should include:

- State boundary layer
- Basin boundary layer
- Gage layer
- Tabular data (as an aesthetic for one of the layers)

3. Create a map of any other spatial data. This could be data from the spatial lesson, data from our other course datasets (e.g., the Litter dataset includes latitude and longitude of trap sites), or another dataset of your choosing. Your map should include:

- One or more layers with polygon features (e.g., country boundaries, watersheds)
- One or more layers with point and/or line features (e.g., sampling sites, roads)
- Tabular data that correpond to one of the layers, specified as an aesthetic (e.g., total litter biomass at each trap, land cover class at each trap)

Hint: One package that may come in handy here is the `maps` package, which contains several options for basemaps that cover political and geologic boundaries.

```r
library("readr")
library("dplyr")
```

```r
library("tidyr")
library("ggplot2")
library("purrr")
library("sf")
library("ggmap")
library("here")
```

```r
pdf(here("outputs", "pdf_test.pdf"), width = 11, height = 8.5)
ggplot(data = cars) +
  geom_point(aes(x = dist, y = speed))
dev.off()
```

```
## pdf
##   2
```

```r
basins_nf_seplains_raw <- st_read(here("data", "spatial_data", "bas_nonref_SEPlains.shp"))
```

```
## Reading layer `bas_nonref_SEPlains' from data source `/Users/emilymcnamara/Desktop/Env Data Analytics
## Simple feature collection with 1232 features and 3 fields
## geometry type:  POLYGON
## dimension:      XY
## bbox:           xmin: -355995 ymin: 571965 xmax: 1812555 ymax: 2209485
## epsg (SRID):    5070
## proj4string:    +proj=aea +lat_1=29.5 +lat_2=45.5 +lat_0=23 +lon_0=-96 +x_0=0 +y_0=0 +ellps=GRS80 +to
```

```r
gages_raw <- st_read(here("data", "spatial_data", "gagesII_9322_sept30_2011.shp"))
```

```
## Reading layer `gagesII_9322_sept30_2011' from data source `/Users/emilymcnamara/Desktop/Env Data Anal
## Simple feature collection with 9322 features and 14 fields
## geometry type:  POINT
## dimension:      XY
## bbox:           xmin: -6233389 ymin: -47038.1 xmax: 3271609 ymax: 6043894
## epsg (SRID):    5070
## proj4string:    +proj=aea +lat_1=29.5 +lat_2=45.5 +lat_0=23 +lon_0=-96 +x_0=0 +y_0=0 +ellps=GRS80 +to
```

```r
southeast_state_bounds_raw <- st_read(here("data", "spatial_data", "southeast_state_bounds.shp"))
```

```
## Reading layer `southeast_state_bounds' from data source `/Users/emilymcnamara/Desktop/Env Data Analy
## Simple feature collection with 5 features and 17 fields
## geometry type:  MULTIPOLYGON
## dimension:      XY
## bbox:           xmin: 796751.8 ymin: 269281.3 xmax: 1833737 ymax: 1966515
## epsg (SRID):    5070
## proj4string:    +proj=aea +lat_1=29.5 +lat_2=45.5 +lat_0=23 +lon_0=-96 +x_0=0 +y_0=0 +ellps=GRS80 +to
```

```r
my_tabular_data_raw <- read.csv(here("data", "tabular_data", "conterm_climate.txt"))
```

```r
my_proj4 <- "+proj=aea +lat_1=29.5 +lat_2=45.5 +lat_0=23 +lon_0=-96 +x_0=0 +y_0=0 +ellps=GRS80 +towgs84=
my_epsg <- 5070

basins_nf_seplains <- basins_nf_seplains_raw
st_crs(basins_nf_seplains) <- my_proj4
basins_nf_seplains <- basins_nf_seplains %>%
  st_set_crs(my_epsg)
st_crs(basins_nf_seplains)
```

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

```
##   EPSG: 5070
##   proj4string: "+proj=aea +lat_1=29.5 +lat_2=45.5 +lat_0=23 +lon_0=-96 +x_0=0 +y_0=0 +ellps=GRS80 +t
```

```r
gages <- gages_raw
st_crs(gages) <- my_proj4
gages <- gages %>%
  st_set_crs(my_epsg)
st_crs(gages)
```

```
## Coordinate Reference System:
##   EPSG: 5070
##   proj4string: "+proj=aea +lat_1=29.5 +lat_2=45.5 +lat_0=23 +lon_0=-96 +x_0=0 +y_0=0 +ellps=GRS80 +t
```

```r
southeast_state_bounds <- southeast_state_bounds_raw
st_crs(southeast_state_bounds) <- my_proj4
southeast_state_bounds <- southeast_state_bounds %>%
  st_set_crs(my_epsg)
st_crs(southeast_state_bounds)
```

```
## Coordinate Reference System:
##   EPSG: 5070
##   proj4string: "+proj=aea +lat_1=29.5 +lat_2=45.5 +lat_0=23 +lon_0=-96 +x_0=0 +y_0=0 +ellps=GRS80 +t
```

```r
na_albers_proj4 <- "+proj=aea +lat_1=20 +lat_2=60 +lat_0=40 +lon_0=-96 +x_0=0 +y_0=0 +datum=NAD83 +unit
na_albers_epsg <- 102008

southeast_state_bounds_na_albers <-  sf::st_transform(southeast_state_bounds, crs = na_albers_proj4) %>
  st_set_crs(na_albers_epsg)
```

```
## Warning: st_crs<- : replacing crs does not reproject data; use st_transform for
## that
```

```r
st_crs(basins_nf_seplains)
```

```
## Coordinate Reference System:
##   EPSG: 5070
##   proj4string: "+proj=aea +lat_1=29.5 +lat_2=45.5 +lat_0=23 +lon_0=-96 +x_0=0 +y_0=0 +ellps=GRS80 +t
```

```r
st_crs(gages)
```

```
## Coordinate Reference System:
##   EPSG: 5070
##   proj4string: "+proj=aea +lat_1=29.5 +lat_2=45.5 +lat_0=23 +lon_0=-96 +x_0=0 +y_0=0 +ellps=GRS80 +t
```

```r
st_crs(southeast_state_bounds)
```

```
## Coordinate Reference System:
##   EPSG: 5070
##   proj4string: "+proj=aea +lat_1=29.5 +lat_2=45.5 +lat_0=23 +lon_0=-96 +x_0=0 +y_0=0 +ellps=GRS80 +t
```

```r
st_crs(southeast_state_bounds_na_albers)
```

```
## Coordinate Reference System:
##   EPSG: 102008
##   proj4string: "+proj=aea +lat_1=20 +lat_2=60 +lat_0=40 +lon_0=-96 +x_0=0 +y_0=0 +datum=NAD83 +units
```

2. Create a new map that mixes spatial and tabular data, as in section 3.3 of the spatial lesson. You may use the maps created in the lesson as an example, but your map should contain data other than precipitation days per year. This map should include:

- State boundary layer
- Basin boundary layer
- Gage layer
- Tabular data (as an aesthetic for one of the layers)

```
# select North Carolina (NC)
nc_state_bounds_geom <- southeast_state_bounds %>%
  filter(NAME == "North Carolina") %>%
  st_geometry()


# select watersheds that intersect with NC bounds
nc_basins_nf_seplains <- basins_nf_seplains %>%
  st_intersection(nc_state_bounds_geom)
```

```
## Warning: attribute variables are assumed to be spatially constant throughout all
## geometries
```

```
# check
# add your code here
head(nc_basins_nf_seplains)
```

```
## Simple feature collection with 6 features and 3 fields
## geometry type:  GEOMETRY
## dimension:      XY
## bbox:           xmin: 1367085 ymin: 1571355 xmax: 1494466 ymax: 1632983
## epsg (SRID):    5070
## proj4string:    +proj=aea +lat_1=29.5 +lat_2=45.5 +lat_0=23 +lon_0=-96 +x_0=0 +y_0=0 +ellps=GRS80 +t
##           AREA PERIMETER  GAGE_ID                       geometry
## 231 1280290000    319620 02069000 POLYGON ((1392705 1615494, ...
## 232 2706760000    404340 02071000 POLYGON ((1419434 1619895, ...
## 236 1408740000    357180 02074000 MULTIPOLYGON (((1434765 162...
## 237 5338250000    585840 02075000 POLYGON ((1459785 1626674, ...
## 238 5483280000    604920 02075045 POLYGON ((1465965 1627822, ...
## 239 6697800000    665040 02075500 POLYGON ((1494466 1632983, ...
```

```
# select gages that fall within NC bounds
# add your code here
nc_gages <- gages %>%
  st_intersection(nc_state_bounds_geom)
```

```
## Warning: attribute variables are assumed to be spatially constant throughout all
## geometries
```

```
# Use the NC state boundary we used earlier to select all the stream gages in NC
nc_gages <- gages %>%
  st_intersection(nc_state_bounds_geom)
```

```
## Warning: attribute variables are assumed to be spatially constant throughout all
## geometries
```

```
# take a look at nc_gages
head(nc_gages)
```

```
## Simple feature collection with 6 features and 14 fields
## geometry type:  POINT
## dimension:      XY
## bbox:           xmin: 1386502 ymin: 1593294 xmax: 1733998 ymax: 1665913
```

```
## epsg (SRID):    5070
## proj4string:    +proj=aea +lat_1=29.5 +lat_2=45.5 +lat_0=23 +lon_0=-96 +x_0=0 +y_0=0 +ellps=GRS80 +to
##           STAID                               STANAME   CLASS  AGGECOREGI
## 1219 0204382800 PASQUOTANK RIVER NEAR SOUTH MILLS, NC     Ref  SECstPlain
## 1233   02053200         POTECASI CREEK NEAR UNION, NC     Ref  SECstPlain
## 1234   02053500          AHOSKIE CREEK AT AHOSKIE, NC Non-ref  SECstPlain
## 1254   02068500          DAN RIVER NEAR FRANCISCO, NC Non-ref EastHghlnds
## 1255   02069000            DAN RIVER AT PINE HALL, NC Non-ref    SEPlains
## 1258   02070500             MAYO RIVER NEAR PRICE, NC     Ref    SEPlains
##      DRAIN_SQKM HUC02 LAT_GAGE  LNG_GAGE STATE HCDN_2009 ACTIVE09 FLYRS1900
## 1219   160.7841    03 36.42139 -76.34250    NC      <NA>      yes         2
## 1233   583.6599    03 36.37083 -77.02556    NC       yes      yes        51
## 1234   165.8835    03 36.28028 -76.99944    NC      <NA>      yes        59
## 1254   321.6789    03 36.51500 -80.30306    NC      <NA>      yes        78
## 1255  1280.2920    03 36.31930 -80.05004    NC      <NA>      yes         6
## 1258   672.6420    03 36.53389 -79.99139    NC      <NA>      yes        58
##      FLYRS1950 FLYRS1990                geometry
## 1219         2         2 POINT (1733998 1665913)
## 1233        51        20 POINT (1675686 1648113)
## 1234        59        20 POINT (1679975 1638632)
## 1254        54        17 POINT (1386502 1611186)
## 1255         4         1 POINT (1412304 1593294)
## 1258        38        16 POINT (1413418 1617864)
```

```r
names(nc_gages)
```

```
##  [1] "STAID"      "STANAME"    "CLASS"      "AGGECOREGI" "DRAIN_SQKM"
##  [6] "HUC02"      "LAT_GAGE"   "LNG_GAGE"   "STATE"      "HCDN_2009"
## [11] "ACTIVE09"   "FLYRS1900"  "FLYRS1950"  "FLYRS1990"  "geometry"
```

```r
# take a look at my_tabular_data_raw
names(my_tabular_data_raw)
```

```
##  [1] "STAID"             "PPTAVG_BASIN"     "PPTAVG_SITE"      "T_AVG_BASIN"
##  [5] "T_AVG_SITE"        "T_MAX_BASIN"      "T_MAXSTD_BASIN"   "T_MAX_SITE"
##  [9] "T_MIN_BASIN"       "T_MINSTD_BASIN"   "T_MIN_SITE"       "RH_BASIN"
## [13] "RH_SITE"           "FST32F_BASIN"     "LST32F_BASIN"     "FST32SITE"
## [17] "LST32SITE"         "WD_BASIN"         "WD_SITE"          "WDMAX_BASIN"
## [21] "WDMIN_BASIN"       "WDMAX_SITE"       "WDMIN_SITE"       "PET"
## [25] "SNOW_PCT_PRECIP"   "PRECIP_SEAS_IND"  "JAN_PPT7100_CM"   "FEB_PPT7100_CM"
## [29] "MAR_PPT7100_CM"    "APR_PPT7100_CM"   "MAY_PPT7100_CM"   "JUN_PPT7100_CM"
## [33] "JUL_PPT7100_CM"    "AUG_PPT7100_CM"   "SEP_PPT7100_CM"   "OCT_PPT7100_CM"
## [37] "NOV_PPT7100_CM"    "DEC_PPT7100_CM"   "JAN_TMP7100_DEGC" "FEB_TMP7100_DEGC"
## [41] "MAR_TMP7100_DEGC" "APR_TMP7100_DEGC" "MAY_TMP7100_DEGC" "JUN_TMP7100_DEGC"
## [45] "JUL_TMP7100_DEGC" "AUG_TMP7100_DEGC" "SEP_TMP7100_DEGC" "OCT_TMP7100_DEGC"
## [49] "NOV_TMP7100_DEGC" "DEC_TMP7100_DEGC"
```

```r
# check column names of nc_gages to look for joining key
names(nc_gages)
```

```
##  [1] "STAID"      "STANAME"    "CLASS"      "AGGECOREGI" "DRAIN_SQKM"
##  [6] "HUC02"      "LAT_GAGE"   "LNG_GAGE"   "STATE"      "HCDN_2009"
## [11] "ACTIVE09"   "FLYRS1900"  "FLYRS1950"  "FLYRS1990"  "geometry"
```

```r
# use "STAID"
```

```r
nc_gages$STAID <- as.factor(nc_gages$STAID)
my_tabular_data_raw$STAID <- as.factor(my_tabular_data_raw$STAID)

# join the tabular data to nc_gages
nc_gages_climate <- nc_gages %>%
  left_join(my_tabular_data_raw, by = "STAID")
```

```
## Warning: Column `STAID` joining factors with different levels, coercing to
## character vector
```

```r
# check that it worked
names(nc_gages_climate)
```

```
##  [1] "STAID"           "STANAME"         "CLASS"           "AGGECOREGI"
##  [5] "DRAIN_SQKM"      "HUC02"           "LAT_GAGE"        "LNG_GAGE"
##  [9] "STATE"           "HCDN_2009"       "ACTIVE09"        "FLYRS1900"
## [13] "FLYRS1950"       "FLYRS1990"       "PPTAVG_BASIN"    "PPTAVG_SITE"
## [17] "T_AVG_BASIN"     "T_AVG_SITE"      "T_MAX_BASIN"     "T_MAXSTD_BASIN"
## [21] "T_MAX_SITE"      "T_MIN_BASIN"     "T_MINSTD_BASIN"  "T_MIN_SITE"
## [25] "RH_BASIN"        "RH_SITE"         "FST32F_BASIN"    "LST32F_BASIN"
## [29] "FST32SITE"       "LST32SITE"       "WD_BASIN"        "WD_SITE"
## [33] "WDMAX_BASIN"     "WDMIN_BASIN"     "WDMAX_SITE"      "WDMIN_SITE"
## [37] "PET"             "SNOW_PCT_PRECIP" "PRECIP_SEAS_IND" "JAN_PPT7100_CM"
## [41] "FEB_PPT7100_CM"  "MAR_PPT7100_CM"  "APR_PPT7100_CM"  "MAY_PPT7100_CM"
## [45] "JUN_PPT7100_CM"  "JUL_PPT7100_CM"  "AUG_PPT7100_CM"  "SEP_PPT7100_CM"
## [49] "OCT_PPT7100_CM"  "NOV_PPT7100_CM"  "DEC_PPT7100_CM"  "JAN_TMP7100_DEGC"
## [53] "FEB_TMP7100_DEGC" "MAR_TMP7100_DEGC" "APR_TMP7100_DEGC" "MAY_TMP7100_DEGC"
## [57] "JUN_TMP7100_DEGC" "JUL_TMP7100_DEGC" "AUG_TMP7100_DEGC" "SEP_TMP7100_DEGC"
## [61] "OCT_TMP7100_DEGC" "NOV_TMP7100_DEGC" "DEC_TMP7100_DEGC" "geometry"
```

```r
pdf(here("outputs", "spatial_operations_activity_2.pdf"), width = 11, height = 8.5)
ggplot() +
  geom_sf(data = nc_state_bounds_geom, fill = NA) +
  geom_sf(data = nc_basins_nf_seplains, alpha = 0.25) +
  geom_sf(data = nc_gages, lwd = 1) +
  geom_sf(data = nc_gages_climate, aes(color = T_AVG_SITE), size = 3) +
  scale_color_gradient(low = "white", high = "darkgreen") +
  labs(color = "Gage Avg. Annual Air Temp (Celsius)" ) +
  geom_sf(data = nc_state_bounds_geom, fill = NA) +
  theme_bw()
dev.off()
```

```
## pdf
##   2
```