Assignment 8: Mapping

Tristen Townsend

OVERVIEW

This exercise accompanies the lessons in Hydrologic Data Analysis on mapping

Directions

1. Change "Student Name" on line 3 (above) with your name.
2. Work through the steps, **creating code and output** that fulfill each instruction.
3. Be sure to **answer the questions** in this assignment document.
4. When you have completed the assignment, **Knit** the text and code into a single pdf file.
5. After Knitting, submit the completed exercise (pdf file) to the dropbox in Sakai. Add your last name into the file name (e.g., "A08_Salk.html") prior to submission.

The completed exercise is due on 23 October 2019 at 9:00 am.

Setup

1. Verify your working directory is set to the R project file,
2. Load the tidyverse, lubridate, cowplot, LAGOSNE, sf, maps, and viridis packages.
3. Set your ggplot theme (can be theme_classic or something else)
4. Load the lagos database, the USA rivers water features shape file, and the HUC6 watershed shape file.

```
library(tidyverse)
```

```
## -- Attaching packages ------------------------------------- tidyverse 1.2.1 --

## v ggplot2 3.2.1     v purrr   0.3.2
## v tibble  2.1.3     v dplyr   0.8.3
## v tidyr   0.8.3     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0

## -- Conflicts ---------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'

## The following object is masked from 'package:base':
##
##     date
```

```
library(cowplot)
```

```
##
## ********************************************************

## Note: As of version 1.0.0, cowplot does not change the

##   default ggplot2 theme anymore. To recover the previous
```

```
##   behavior, execute:
##   theme_set(theme_cowplot())

## *******************************************************

##
## Attaching package: 'cowplot'

## The following object is masked from 'package:lubridate':
##
##     stamp
```

```
library(LAGOSNE)
library(sf)
```

```
## Linking to GEOS 3.7.2, GDAL 2.4.2, PROJ 5.2.0
```

```
library(maps)
```

```
##
## Attaching package: 'maps'

## The following object is masked from 'package:purrr':
##
##     map
```

```
library(viridis)
```

```
## Loading required package: viridisLite
```

```
theme_set(theme_classic())
```

Mapping water quality in lakes

Complete the in-class exercise from lesson 15, to map average secchi depth measurements across states in Maine, considering lake area and lake depth as predictors for water clarity. Steps here are identical to the lesson, with the following edits:

- Make sure all your wrangling is done in this document (this includes basic wrangling of the LAGOS database)
- In your cowplot, do not adjust the legend items (even though they look ugly). Rather, reflect on how you would improve them with additional coding.
- For item 9, **do** run a regression on secchi depth by lake area and a separate regression on secchi depth by lake depth. Make scatterplots of these relationships. Note that log-transforming one of these items may be necessary.

5. Filter the states and secchi depth datasets so that they contain Maine only. For the secchi depth dataset, create a summary dataset with just the mean secchi depth.

```
#Load and wrangle data
LAGOSdata <- lagosne_load()
```

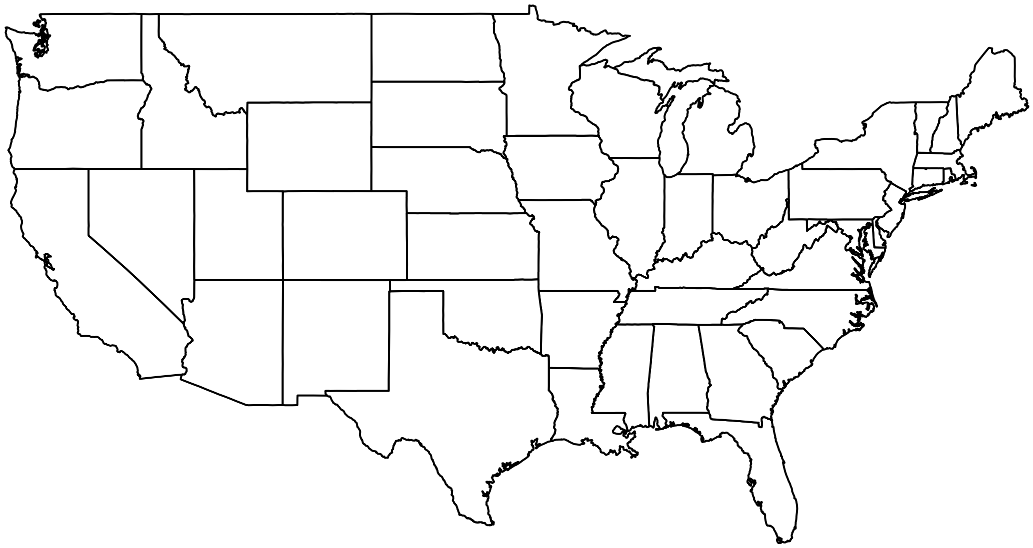
```
## Warning in `_f`(version = version, fpath = fpath): LAGOSNE version
## unspecified, loading version: 1.087.3
```

```
LAGOSlocus <- LAGOSdata$locus
LAGOSstate <- LAGOSdata$state
LAGOSnutrient <- LAGOSdata$epi_nutr
LAGOSlimno <- LAGOSdata$lakes_limno
```

```r
#Join LAGOS data
LAGOScombined <-
  left_join(LAGOSnutrient, LAGOSlocus) %>%
  left_join(., LAGOSlimno) %>%
  left_join(., LAGOSstate) %>%
  filter(!is.na(state)) %>%
  select(lagoslakeid, sampledate, secchi, lake_area_ha, maxdepth, nhd_lat, nhd_long, state)
```

```
## Joining, by = "lagoslakeid"

## Joining, by = c("lagoslakeid", "nhdid", "nhd_lat", "nhd_long")

## Joining, by = "state_zoneid"
```

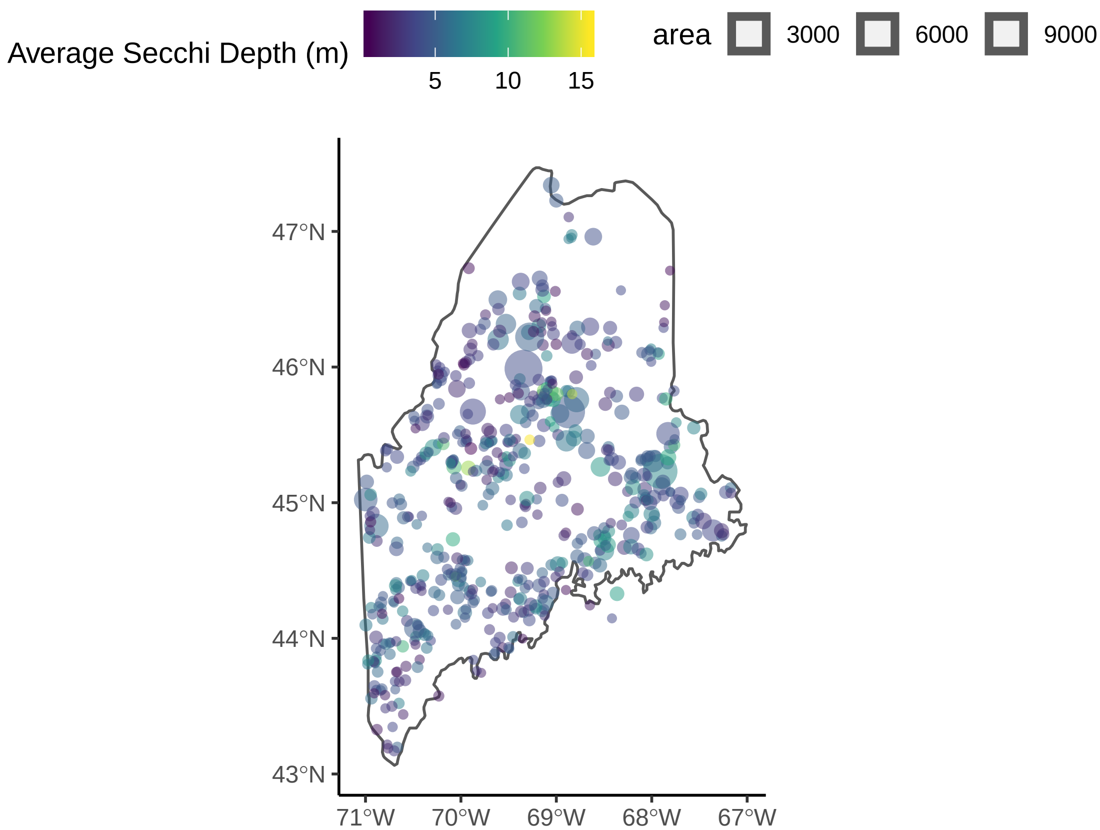
```r
#Filter LAGOScombined for just Maine
maine <- filter(LAGOScombined, state == "ME")

#Summarize Maine secchi data
maine.summary <- maine %>%
  group_by(lagoslakeid) %>%
  summarise(secchi.mean = mean(secchi),
            area = mean(lake_area_ha),
            depth = mean(maxdepth),
            lat = mean(nhd_lat),
            long = mean(nhd_long)) %>%
  drop_na()

#Geometry
states <- st_as_sf(map(database = "state", plot = TRUE, fill = TRUE, col = "white"))
```



```
maine.subset <- filter(states, ID %in% c("maine"))
mainesecchi.spatial <- st_as_sf(maine.summary, coords = c("long", "lat"), crs = 4326)
```

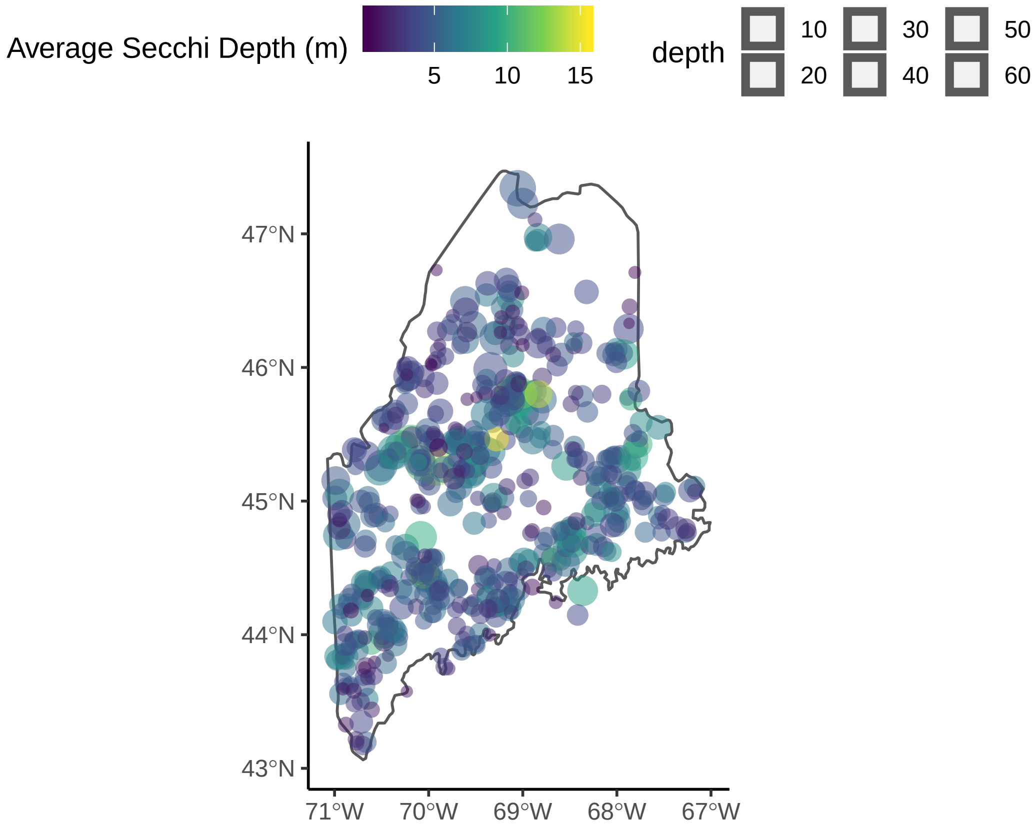
6. Create a plot of mean secchi depth for lakes in Maine, with mean secchi depth designated as color and the lake area as the size of the dot. Remember that you are using size in the aesthetics and should remove the size = 1 from the other part of the code. Adjust the transparency of points as needed.

```
#Plot for mean secchi depth
MEplot.secchi.area <- ggplot() +
  geom_sf(data = maine.subset, fill = "white") +
  geom_sf(data = mainesecchi.spatial, aes(color = secchi.mean, size = area),
          alpha = 0.5) +
  scale_color_viridis_c() +
  labs(color = "Average Secchi Depth (m)") +
  theme(legend.position = "top")
print(MEplot.secchi.area)
```



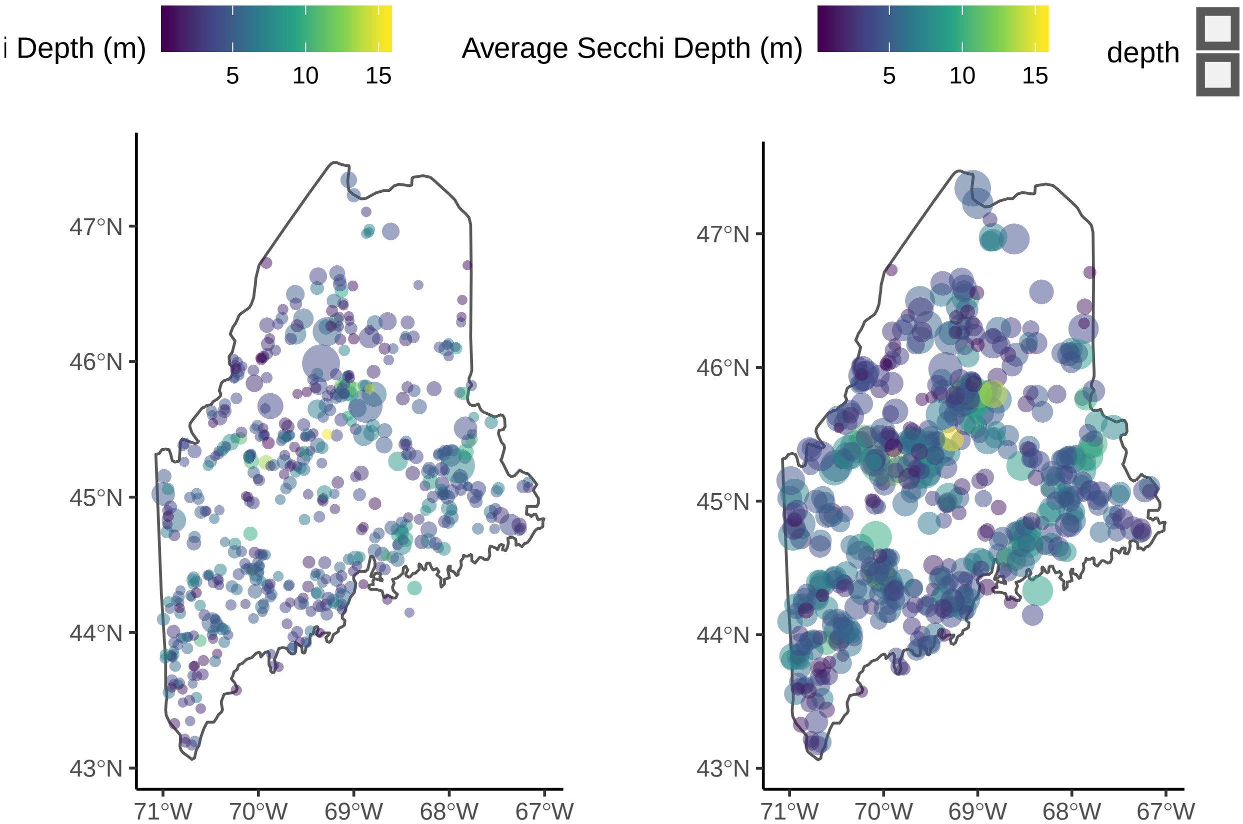
7. Create a second plot, but this time use maximum depth of the lake as the size of the dot.

```
MEplot.secchi.depth<- ggplot() +
  geom_sf(data = maine.subset, fill = "white") +
  geom_sf(data = mainesecchi.spatial, aes(color = secchi.mean, size = depth),
          alpha = 0.5) +
  scale_color_viridis_c() +
  labs(color = "Average Secchi Depth (m)") +
  theme(legend.position = "top")
print(MEplot.secchi.depth)
```



8. Plot these maps in the same plot with the `plot_grid` function. Don't worry about adjusting the legends (if you have extra time this would be a good bonus task).

```
plot_grid(MEplot.secchi.area, MEplot.secchi.depth)
```



What would you change about the legend to make it a more effective visualization?

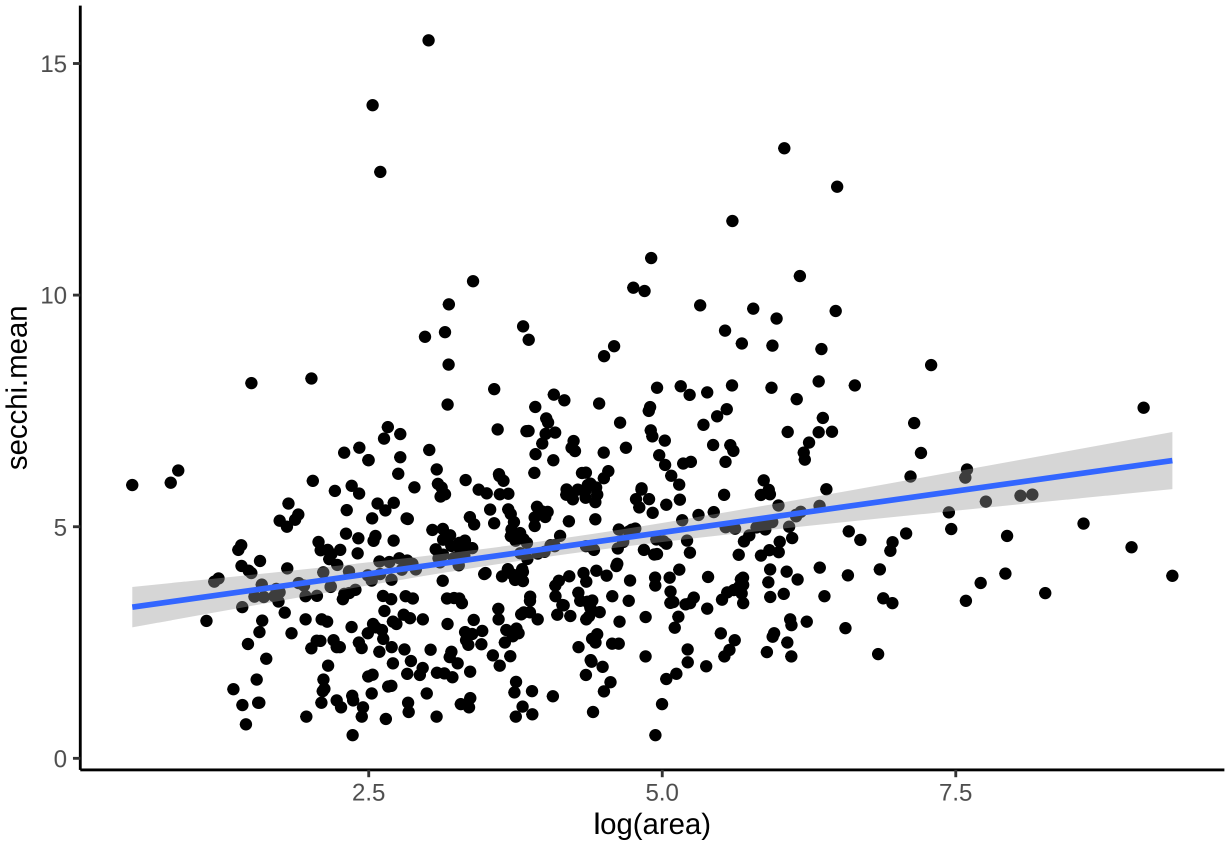
I would make it so 'Average Secchi Depth (m)' wasn't listed twice. I would also clarify the squares that are listed for the size parameter and try to make them circles to match the graph and put a few with different sizes and numbers. I would add units to both of those as well.

9. What relationships do you see between secchi depth, lake area, and lake depth? Which of the two lake variables seems to be a stronger determinant of secchi depth? (make a scatterplot and run a regression to test this)

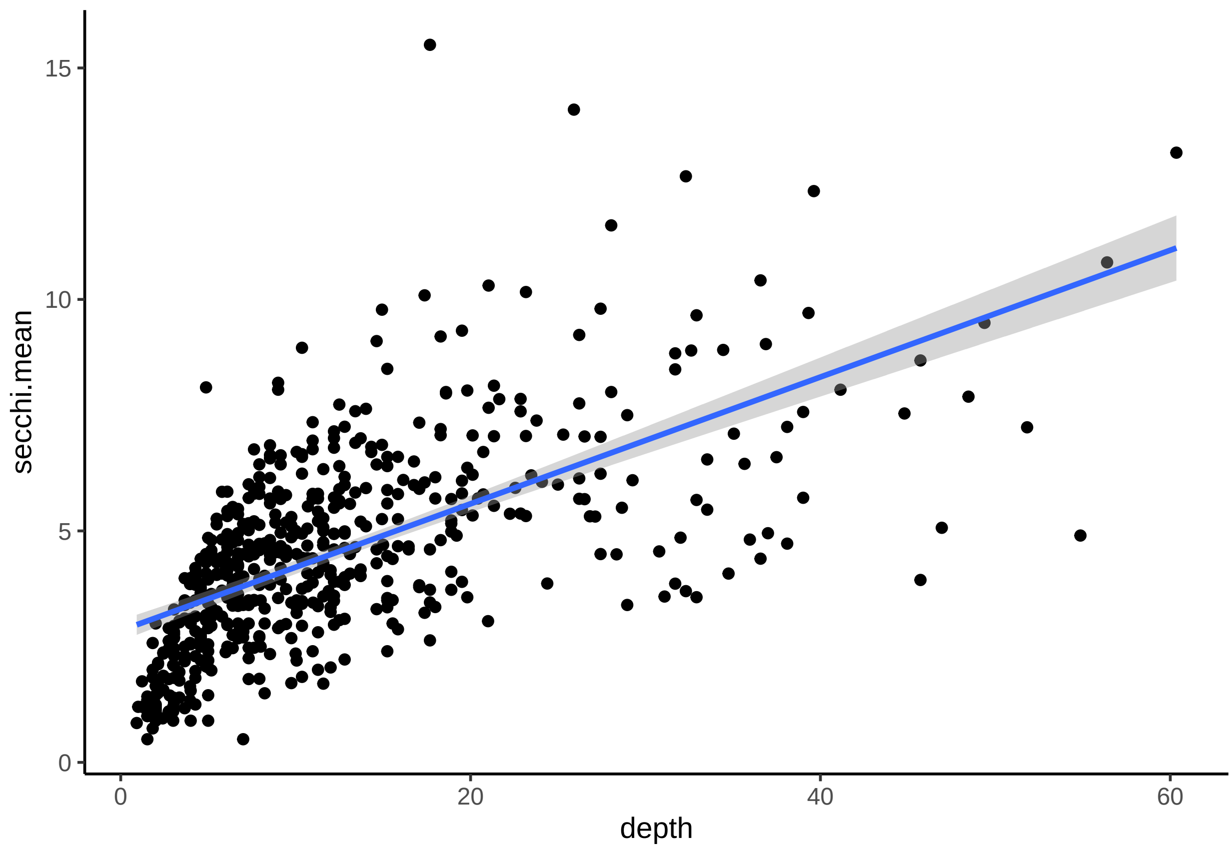
   It looks like there's some variation across the area of lakes and the mean secchi depth, but more consistently seeing a trend in the relationship between lake depth and secchi depth. The deeper the lake the deeper the secchi depth tends to be. So lake depth seems to be a stronger determinant than lake area.

Note: consider log-transforming a predictor variable if appropriate

```r
#Scatterplots
ggplot(maine.summary, aes(x = log(area), y = secchi.mean)) +
  geom_point() +
  geom_smooth(method = "lm")
```




```r
ggplot(maine.summary, aes(x = depth, y = secchi.mean)) +
  geom_point() +
  geom_smooth(method = "lm")
```




```
#Regression
lm.area <- lm(secchi.mean ~ log(area), data = maine.summary)
summary(lm.area)
```

```
##
## Call:
## lm(formula = secchi.mean ~ log(area), data = maine.summary)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -4.355 -1.492 -0.167  1.145 11.335
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.08979    0.24734  12.492  < 2e-16 ***
## log(area)    0.35723    0.05682   6.287 6.62e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.109 on 547 degrees of freedom
## Multiple R-squared:  0.06739,    Adjusted R-squared:  0.06569
## F-statistic: 39.53 on 1 and 547 DF,  p-value: 6.622e-10
```

```
lm.depth <- lm(secchi.mean ~ depth, data = maine.summary)
summary(lm.depth) #Accounts for more variability so depth
```

```
##
```

```
## Call:
## lm(formula = secchi.mean ~ depth, data = maine.summary)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.4592 -1.1484 -0.1410  0.9497 10.2329
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 2.846212   0.115959   24.55   <2e-16 ***
## depth       0.136939   0.007308   18.74   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.705 on 547 degrees of freedom
## Multiple R-squared:  0.3909, Adjusted R-squared:  0.3898
## F-statistic: 351.1 on 1 and 547 DF,  p-value: < 2.2e-16
```

Mapping water features and watershed boundaries

10. Wrangle the USA rivers and HUC6 watershed boundaries dataset so that they include only the features present in Florida (FL). Adjust the coordinate reference systems if necessary to ensure they use the same projection.

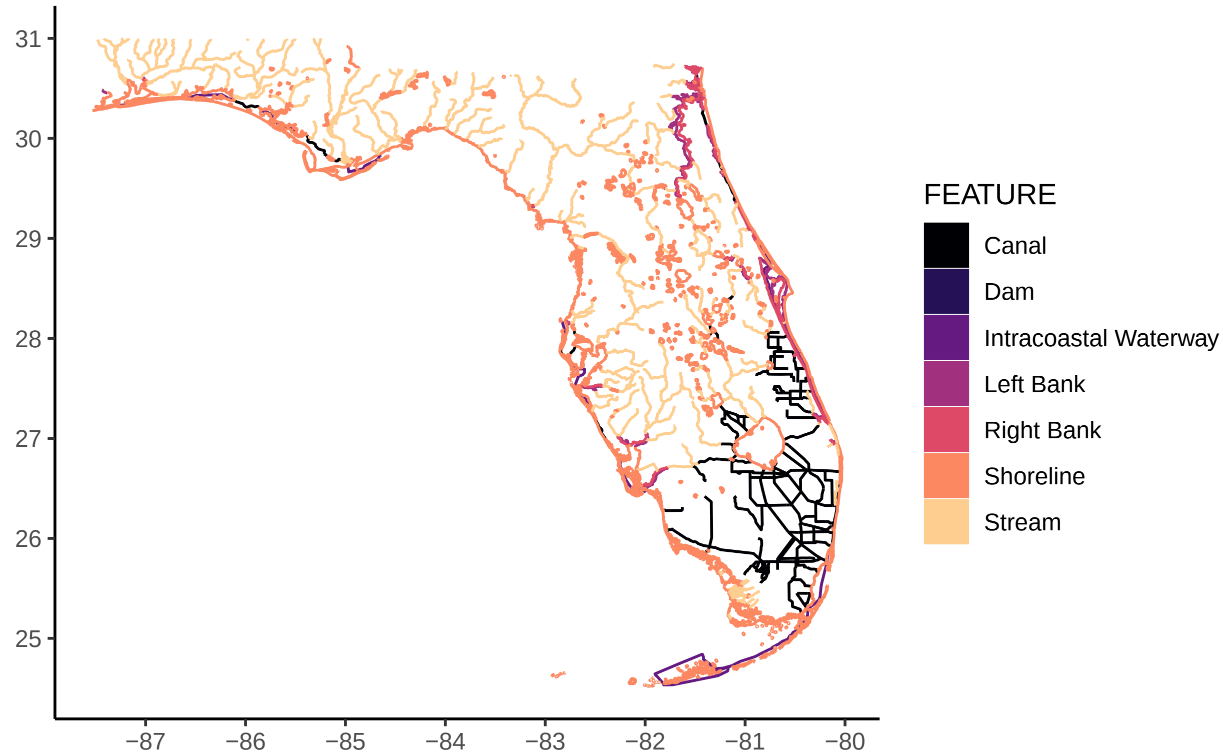
```r
waterfeatures <- st_read("./Data/Raw/hydrogl020.dbf")
```

```
## Reading layer `hydrogl020' from data source `/Users/Tristen/OneDrive - Duke University/Fall 2019/Hyd
## 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
```

```r
class(waterfeatures)
```

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

```r
# Filter for Florida
waterfeatures <- filter(waterfeatures, STATE == "FL")

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

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)
```




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

```
## Reading layer `WBDHU6' from data source `/Users/Tristen/OneDrive - Duke University/Fall 2019/Hydrolog
## 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 +ellps=GRS80 +towgs84=0,0,0,0,0,0,0 +no_defs
```

```
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
```

```
HUC6.FL <- HUC6 %>%
  filter(States %in% c("AL,FL", "AL,FL,GA", "FL", "FL,GA"))
```

11. Create a map of watershed boundaries in Florida, with the layer of water features on top. Color the watersheds gray (make sure the lines separating watersheds are still visible) and color the water features by type.

```
st_crs(waterfeatures)
```

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

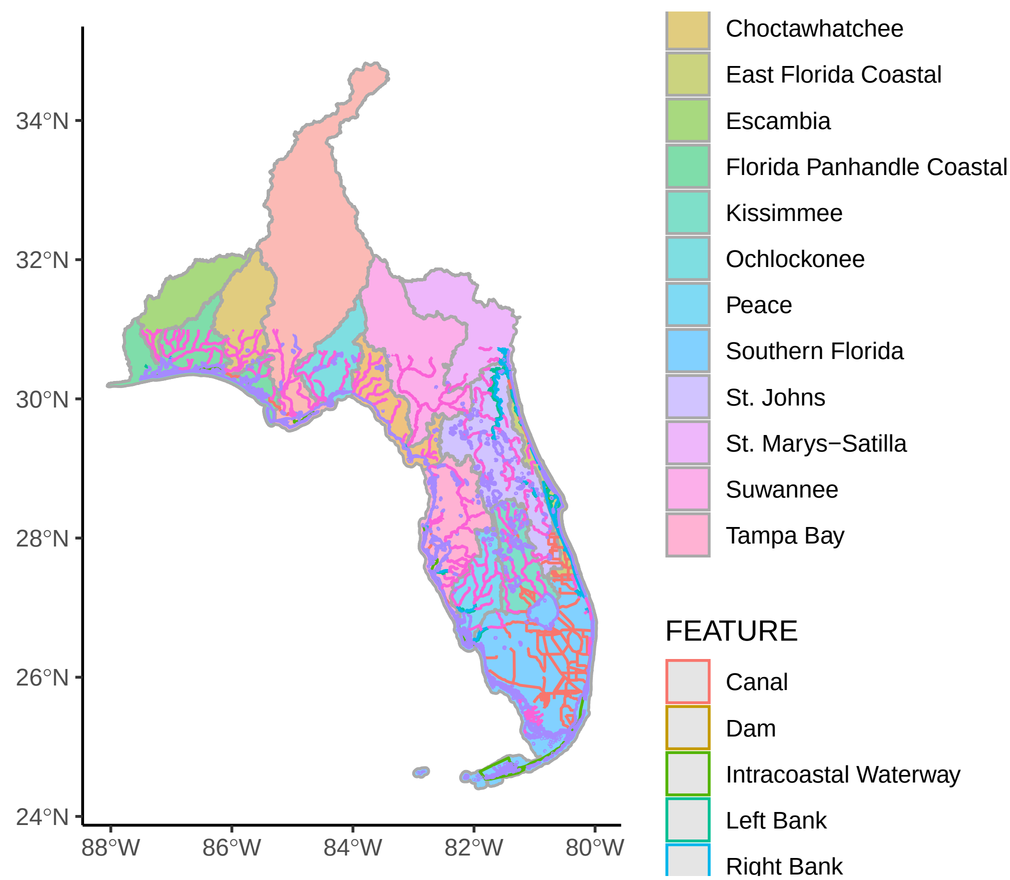
```r
st_crs(HUC6.FL)
```

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

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



12. What are the dominant water features in Florida? How does this distribution differ (or not) compared to North Carolina?

    Florida has lots of streams and canals (and shoreline, of course). North Carolina seems to be mostly dominated by just streams, with a bit of left bank features (unsure what that means though).

Reflection

13. What are 2-3 conclusions or summary points about mapping you learned through your analysis?

    This lesson was a good reminder how natural features and watersheds dont follow the typically boundaries we draw between places in our head.

    Maps can be useful tools to identify trends in data.

14. What data, visualizations, and/or models supported your conclusions from 13?

    Map of features and Florida - lots of watersheds were crossing state boundaries.

    Map of secchi depth vs lake area and lake depth help guide how to test correlation.

15. Did hands-on data analysis impact your learning about mapping relative to a theory-based lesson? If so, how?

    Yes. Practicing how to adjust geometry in the code and how to deal with actual data to make the best visualization was useful.

16. How did the real-world data compare with your expectations from theory?

    They matched!