

Lecture 16

Spatial Data and Cartography (Part 2)

10/31/2018

Plotting

Example Data - NC SIDS

```
nc = st_read(system.file("shape/nc.shp", package="sf"), quiet = TRUE) %>%  
  select(-(AREA:CNTY_ID), -(FIPS:CRESS_ID))
```

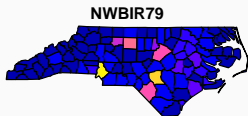
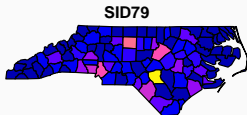
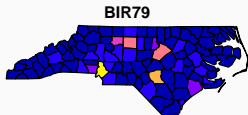
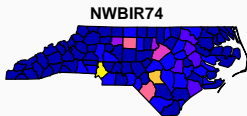
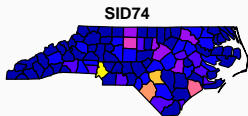
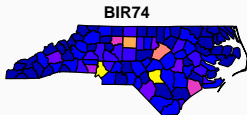
```
tbl_df(nc)
```

```
## # A tibble: 100 x 8
```

```
##   NAME   BIR74  SID74  NWBIR74  BIR79  SID79  NWBIR79  geometry  
##   <fct> <dbl> <dbl>    <dbl> <dbl> <dbl>    <dbl>    <MULTIPOLYGON [°]>  
## 1 Ashe   1091    1      10    1364    0      19 ((((-81.47276 36.23436, ~  
## 2 Alle~  487    0      10    542    3      12 ((((-81.23989 36.36536, ~  
## 3 Surry  3188   5      208   3616   6      260 ((((-80.45634 36.24256, ~  
## 4 Curr~  508    1      123   830    2      145 ((((-76.00897 36.3196, -7~  
## 5 Nort~ 1421   9      1066  1606   3      1197 ((((-77.21767 36.24098, ~  
## 6 Hert~ 1452   7      954   1838   5      1237 ((((-76.74506 36.23392, ~  
## 7 Camd~  286    0      115   350    2      139 ((((-76.00897 36.3196, -7~  
## 8 Gates  420    0      254   594    2      371 ((((-76.56251 36.34057, ~  
## 9 Warr~  968    4      748   1190   2      844 ((((-78.30876 36.26004, ~  
## 10 Stok~ 1612   1      160   2038   5      176 ((((-80.02567 36.25023, ~  
## # ... with 90 more rows
```

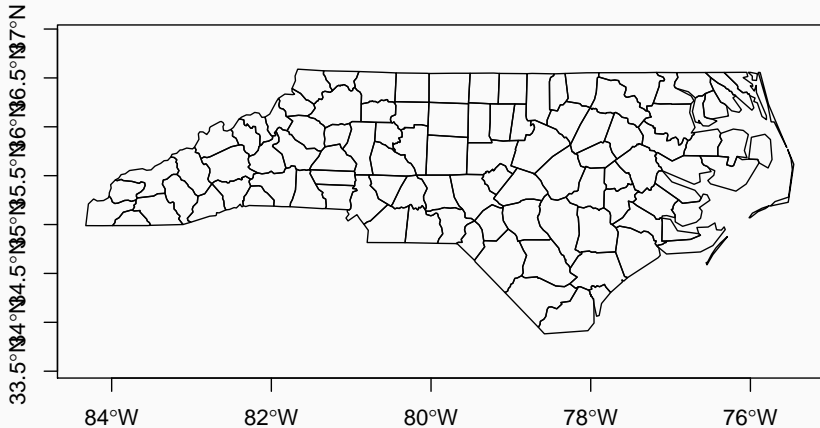
Base Plots

`plot(nc)`



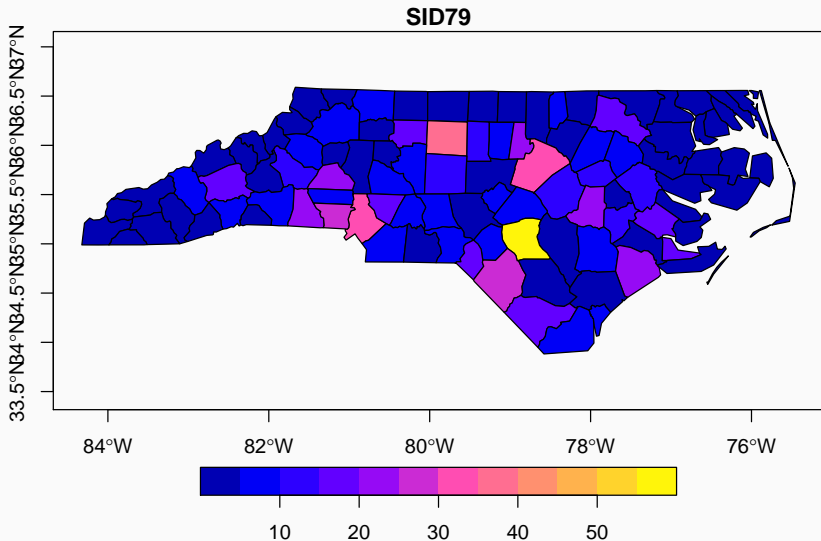
Geometry Plot

```
plot(st_geometry(nc), axes=TRUE)
```



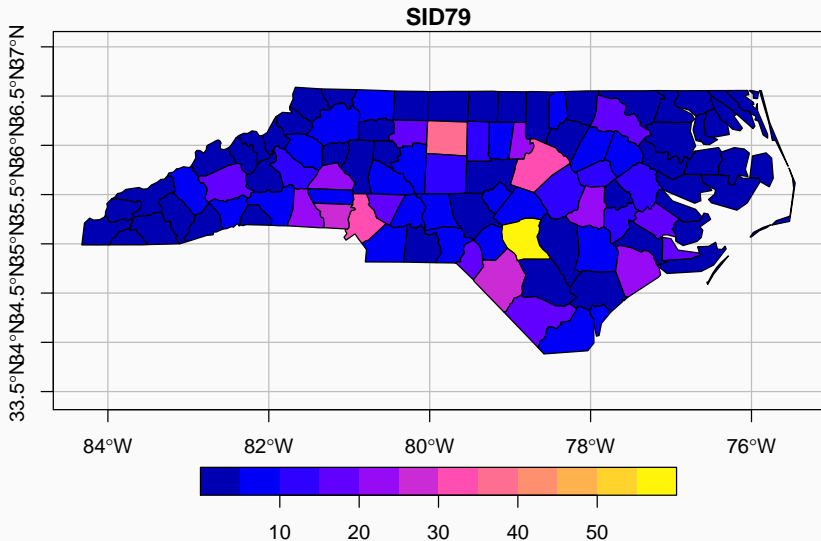
Graticules

```
plot(nc[, "SID79"], axes=TRUE)
```



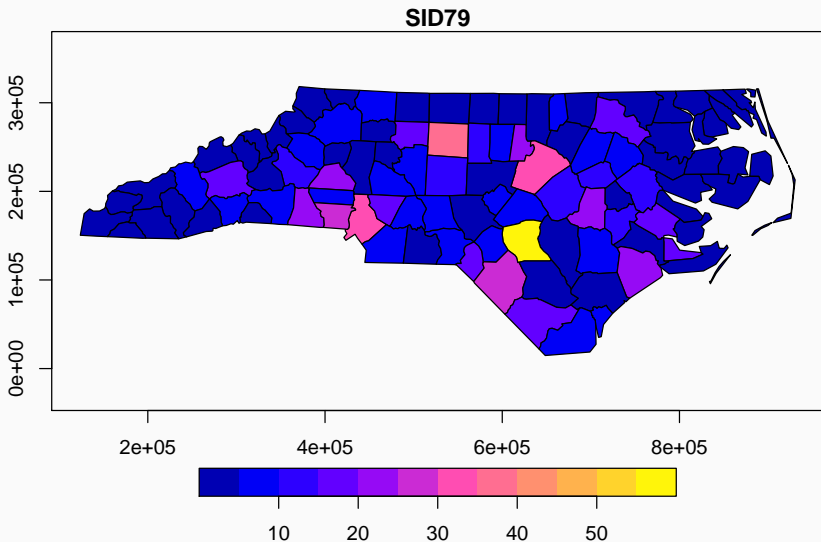
Graticules

```
plot(nc[, "SID79"], graticule=st_crs(nc), axes=TRUE)
```



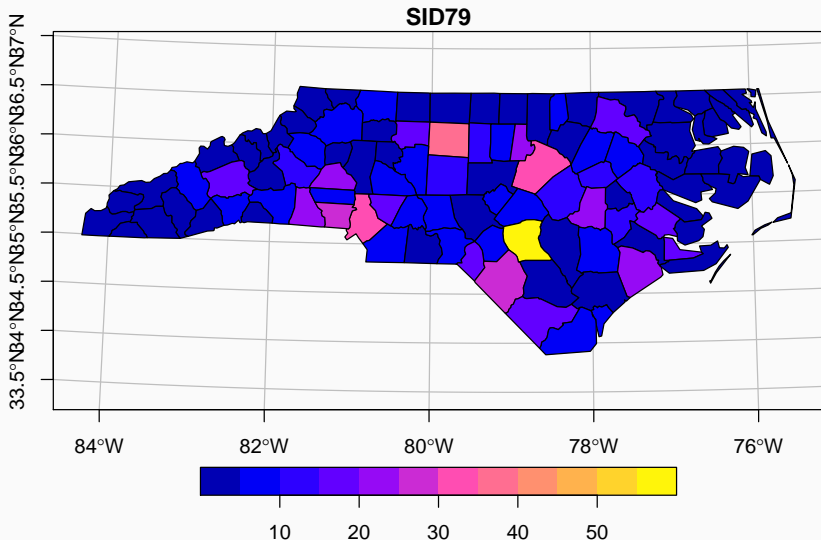
Graticules (EPSG:3631)

```
plot(st_transform(nc[, "SID79"], 3631), axes=TRUE)
```



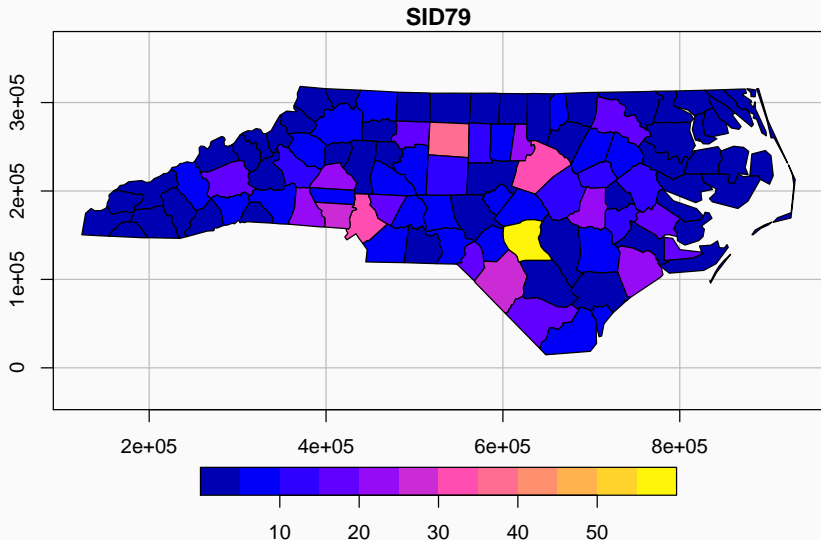
Graticules (EPSG:3631)

```
plot(st_transform(nc[, "SID79"], 3631), graticule=st_crs(nc), axes=TRUE)
```

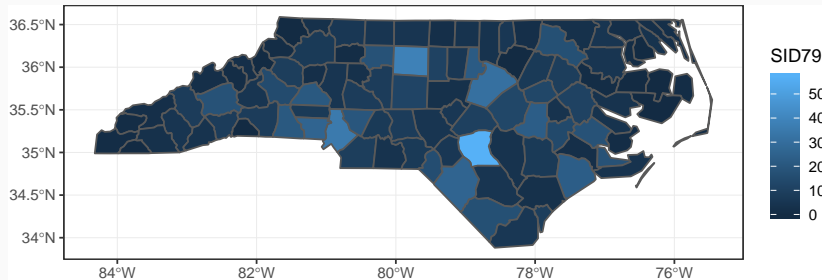


Graticules (EPSG:3631)

```
plot(st_transform(nc[, "SID79"], 3631), graticule=st_crs(3631), axes=TRUE)
```

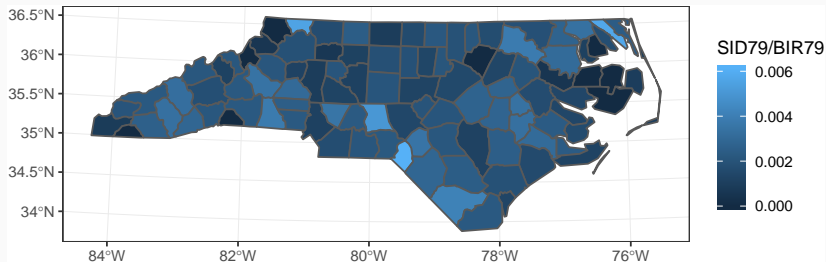


```
ggplot(nc) +  
  geom_sf(aes(fill=SID79))
```



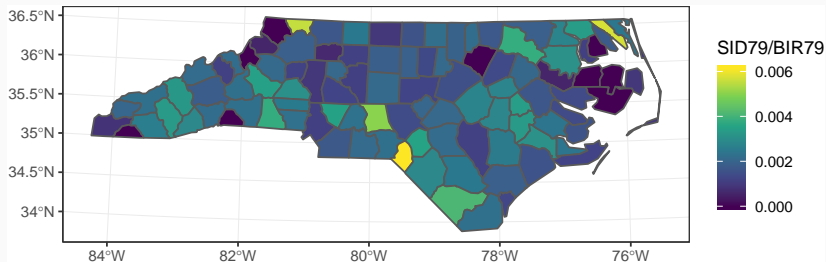
ggplot2 + projections

```
ggplot(st_transform(nc, 3631)) +  
  geom_sf(aes(fill=SID79 / BIR79))
```



ggplot2 + viridis

```
ggplot(st_transform(nc, 3631)) +  
  geom_sf(aes(fill=SID79 / BIR79)) +  
  scale_fill_viridis_c()
```



Example Data - Meuse

```
data(meuse, meuse.riv, package="sp")
```

```
meuse = st_as_sf(meuse, coords=c("x", "y"), crs=28992)
```

```
meuse_riv = st_polygon(list(meuse.riv)) %>% st_sfc() %>% st_set_crs(28992)
```

```
tbl_df(meuse)
```

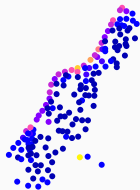
```
## # A tibble: 155 x 13
```

```
##   cadmium copper lead zinc elev dist om ffreq soil lime landuse
## *   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <fct> <fct> <fct> <fct>
## 1  11.7    85  299 1022 7.91 0.00136 13.6 1 1 1 Ah
## 2   8.6    81  277 1141 6.98 0.0122 14 1 1 1 Ah
## 3   6.5    68  199  640 7.8 0.103 13 1 1 1 Ah
## 4   2.6    81  116  257 7.66 0.190 8 1 2 0 Ga
## 5   2.8    48  117  269 7.48 0.277 8.7 1 2 0 Ah
## 6   3      61  137  281 7.79 0.364 7.8 1 2 0 Ga
## 7   3.2    31  132  346 8.22 0.190 9.2 1 2 0 Ah
## 8   2.8    29  150  406 8.49 0.0922 9.5 1 1 0 Ab
## 9   2.4    37  133  347 8.67 0.185 10.6 1 1 0 Ab
## 10  1.6     24  80  183 9.05 0.310 6.3 1 2 0 W
## # ... with 145 more rows, and 2 more variables: dist.m <dbl>,
## # geometry <POINT [m]>
```

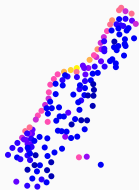
Meuse

```
plot(meuse, pch=16)
```

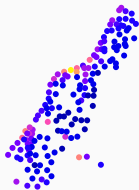
cadmium



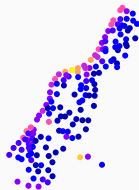
copper



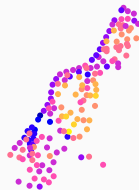
lead



zinc



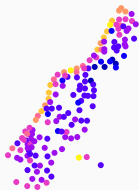
elev



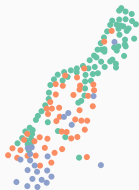
dist



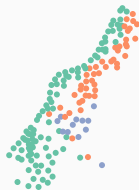
om



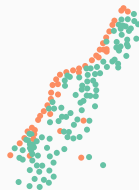
ffreq



soil

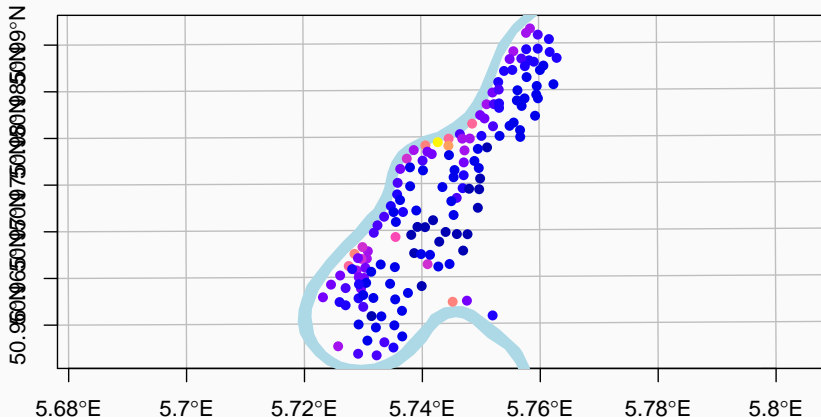


lime

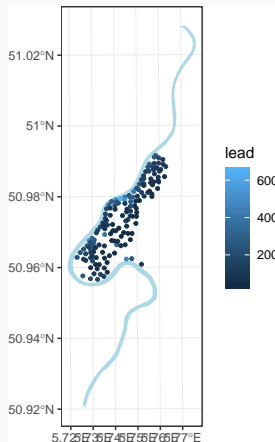


Layering plots

```
plot(meuse_riv, col=adjustcolor("lightblue", alpha.f=1), border = NA,  
     axes=TRUE, graticule=st_crs(4326), xlim=st_bbox(meuse)[c(1,3)], ylim=st_bbox(meuse)  
plot(meuse[, "lead"], pch=16, add=TRUE)
```

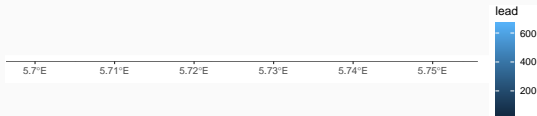



```
ggplot() +  
  geom_sf(data=st_sf(meuse_riv), fill="lightblue", color=NA) +  
  geom_sf(data=meuse, aes(color=lead), size=1)
```



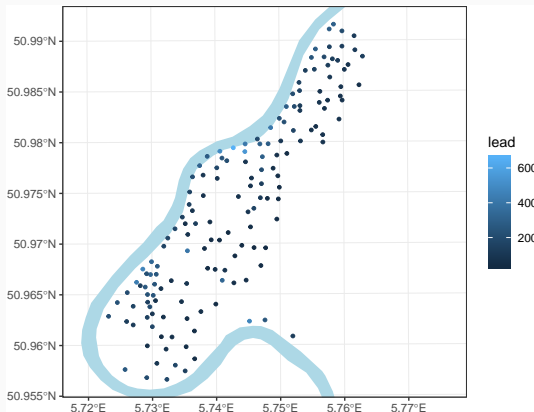
ggplot2 - axis limits

```
ggplot() +  
  geom_sf(data=st_sf(meuse_riv), fill="lightblue", color=NA) +  
  geom_sf(data=meuse, aes(color=lead), size=1) +  
  ylim(50.95, 50.99)
```



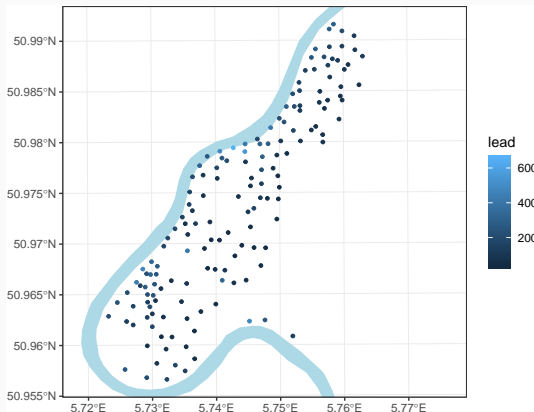
ggplot2 - axis limits

```
ggplot() +  
  geom_sf(data=st_sf(meuse_riv), fill="lightblue", color=NA) +  
  geom_sf(data=meuse, aes(color=lead), size=1) +  
  ylim(329714, 333611)
```



ggplot2 - bounding box

```
ggplot() +  
  geom_sf(data=st_sf(meuse_riv), fill="lightblue", color=NA) +  
  geom_sf(data=meuse, aes(color=lead), size=1) +  
  ylim(st_bbox(meuse)["ymin"], st_bbox(meuse)["ymax"])
```



Geometry Manipulation

Casting

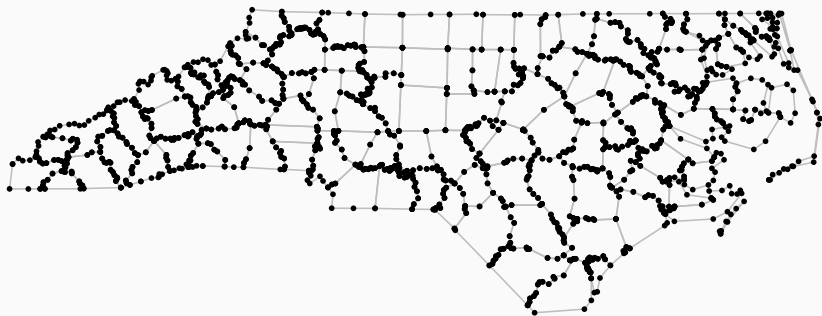
```
nc_pts = st_cast(nc, "MULTIPOINT")
```

```
tbl_df(nc_pts)
```

```
## # A tibble: 100 x 8
```

```
##   NAME BIR74 SID74 NWBIR74 BIR79 SID79 NWBIR79 geometry
## * <fct> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <MULTIPOINT [°]>
## 1 Ashe 1091 1 10 1364 0 19 (-81.47276 36.23436, -81~
## 2 Alle~ 487 0 10 542 3 12 (-81.23989 36.36536, -81~
## 3 Surry 3188 5 208 3616 6 260 (-80.45634 36.24256, -80~
## 4 Curr~ 508 1 123 830 2 145 (-76.00897 36.3196, -76.~
## 5 Nort~ 1421 9 1066 1606 3 1197 (-77.21767 36.24098, -77~
## 6 Hert~ 1452 7 954 1838 5 1237 (-76.74506 36.23392, -76~
## 7 Camd~ 286 0 115 350 2 139 (-76.00897 36.3196, -75.~
## 8 Gates 420 0 254 594 2 371 (-76.56251 36.34057, -76~
## 9 Warr~ 968 4 748 1190 2 844 (-78.30876 36.26004, -78~
## 10 Stok~ 1612 1 160 2038 5 176 (-80.02567 36.25023, -80~
## # ... with 90 more rows
```

```
plot(st_geometry(nc), border='grey')  
plot(st_geometry(nc_pts), pch=16, cex=0.5, add=TRUE)
```

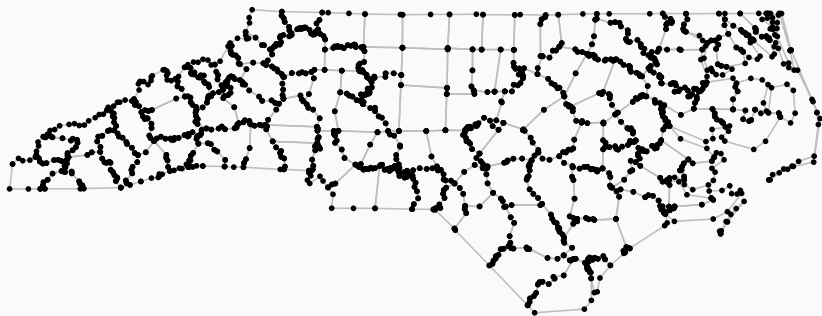


Casting - POINT

```
st_cast(nc, "POINT")
## Simple feature collection with 2529 features and 7 fields
## geometry type: POINT
## 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
## First 10 features:
##      NAME BIR74 SID74 NWBIR74 BIR79 SID79 NWBIR79
## 1 Ashe 1091 1 10 1364 0 19
## 1.1 Ashe 1091 1 10 1364 0 19
## 1.2 Ashe 1091 1 10 1364 0 19
## 1.3 Ashe 1091 1 10 1364 0 19
## 1.4 Ashe 1091 1 10 1364 0 19
## 1.5 Ashe 1091 1 10 1364 0 19
## 1.6 Ashe 1091 1 10 1364 0 19
## 1.7 Ashe 1091 1 10 1364 0 19
## 1.8 Ashe 1091 1 10 1364 0 19
## 1.9 Ashe 1091 1 10 1364 0 19
##
##              geometry
## 1 POINT (-81.47276 36.23436)
## 1.1 POINT (-81.54084 36.27251)
## 1.2 POINT (-81.56198 36.27359)
## 1.3 POINT (-81.63306 36.34069)
## 1.4 POINT (-81.74107 36.39178)
## 1.5 POINT (-81.69828 36.47178)
## 1.6 POINT (-81.7028 36.51934)
## 1.7 POINT (-81.67 36.58965)
```



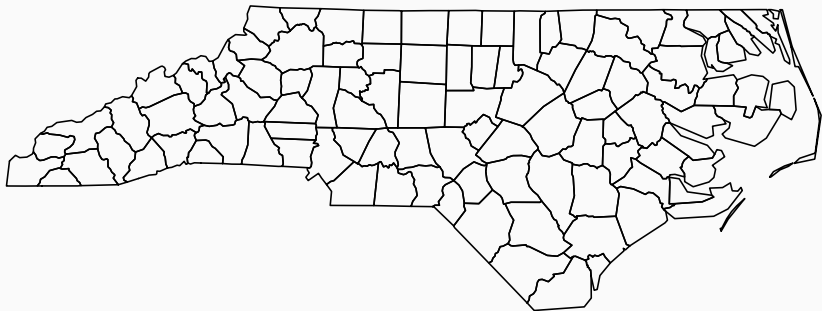
```
plot(st_geometry(nc), border='grey')  
plot(st_geometry(st_cast(nc, "POINT")), pch=16, cex=0.5, add=TRUE)
```



Casting - LINESTRING

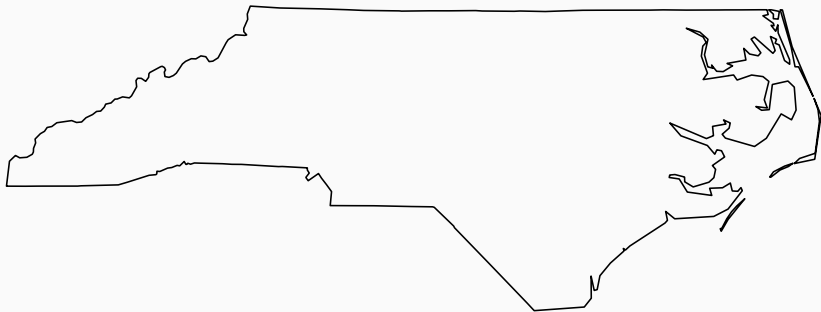
```
st_cast(nc, "MULTILINESTRING") %>% as_tibble()
## # A tibble: 100 x 8
##   NAME BIR74 SID74 NWBIR74 BIR79 SID79 NWBIR79 geometry
## * <fct> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <MULTILINESTRING [°]>
## 1 Ashe 1091 1 10 1364 0 19 ((-81.47276 36.23436, -8~
## 2 Alle~ 487 0 10 542 3 12 ((-81.23989 36.36536, -8~
## 3 Surry 3188 5 208 3616 6 260 ((-80.45634 36.24256, -8~
## 4 Curr~ 508 1 123 830 2 145 ((-76.00897 36.3196, -76~
## 5 Nort~ 1421 9 1066 1606 3 1197 ((-77.21767 36.24098, -7~
## 6 Hert~ 1452 7 954 1838 5 1237 ((-76.74506 36.23392, -7~
## 7 Camd~ 286 0 115 350 2 139 ((-76.00897 36.3196, -75~
## 8 Gates 420 0 254 594 2 371 ((-76.56251 36.34057, -7~
## 9 Warr~ 968 4 748 1190 2 844 ((-78.30876 36.26004, -7~
## 10 Stok~ 1612 1 160 2038 5 176 ((-80.02567 36.25023, -8~
## # ... with 90 more rows
```

```
st_cast(nc, "MULTILINESTRING") %>% st_geometry() %>% plot()
```



Grouping Features

```
nc_state = st_union(nc)
tbl_df(nc_state)
## # A tibble: 1 x 1
##
## * geometry
## 1 (((-76.54427 34.58783, -76.55515 34.61066, -76.53775 34.61392, -76.48448~
plot(nc_state)
```



More Grouping

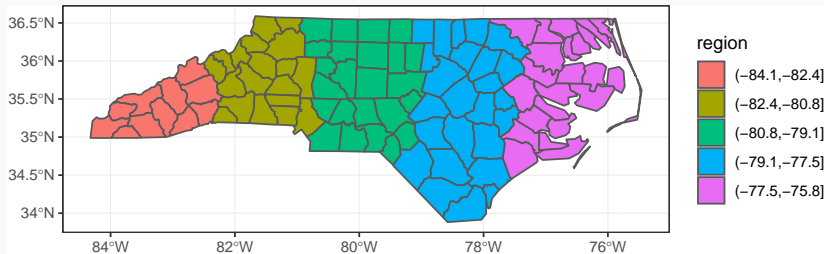
```
nc_cut = nc %>%  
  mutate(X = st_centroid(nc) %>% st_coordinates() %>% .[,1]) %>%  
  mutate(region = cut(X, breaks = 5))
```

```
tbl_df(nc_cut)
```

```
## # A tibble: 100 x 10
```

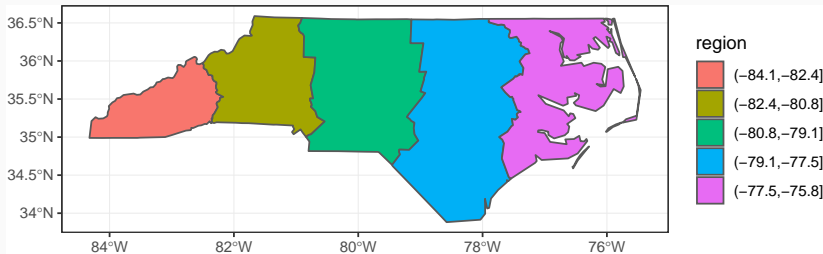
```
##   NAME  BIR74  SID74  NWBIR74  BIR79  SID79  NWBIR79    X region  
##   <fct> <dbl> <dbl>    <dbl> <dbl> <dbl>    <dbl> <dbl> <fct>  
## 1 Ashe  1091    1      10  1364    0      19 -81.5 (-82.~  
## 2 Alle~  487    0      10  542    3      12 -81.1 (-82.~  
## 3 Surry 3188    5      208 3616    6      260 -80.7 (-80.~  
## 4 Curr~  508    1      123  830    2      145 -76.0 (-77.~  
## 5 Nort~ 1421    9     1066 1606    3     1197 -77.4 (-77.~  
## 6 Hert~ 1452    7      954 1838    5     1237 -77.0 (-77.~  
## 7 Camd~  286    0      115  350    2      139 -76.2 (-77.~  
## 8 Gates  420    0      254  594    2      371 -76.7 (-77.~  
## 9 Warr~  968    4      748 1190    2      844 -78.1 (-79.~  
## 10 Stok~ 1612    1      160 2038    5      176 -80.2 (-80.~  
## # ... with 90 more rows, and 1 more variable: geometry <MULTIPOLYGON [°]>
```

```
ggplot(nc_cut) +  
  geom_sf(aes(fill=region))
```



dplyr and sf

```
nc_cut %>%  
  group_by(region) %>%  
  summarize() %>%  
  ggplot() +  
    geom_sf(aes(fill=region))
```



Affine Transformations

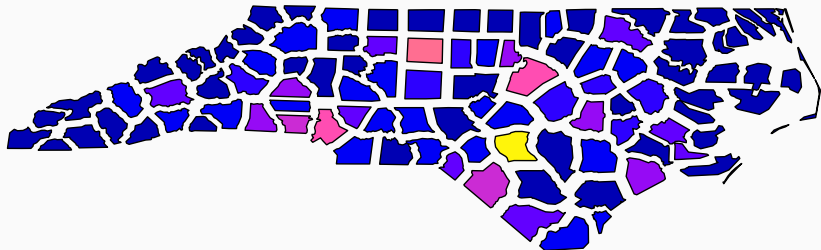
```
rotate = function(a) matrix(c(cos(a), sin(a), -sin(a), cos(a)), 2, 2)  
  
ctrd = st_centroid(nc_state)  
state_rotate = lwgeom::st_make_valid( (nc_state) * rotate(-pi/4) )  
plot(state_rotate, axes=TRUE)
```



Scaling Size

```
ctrd = st_centroid(st_geometry(nc))  
area = st_area(nc) %>% strip_attrs()  
  
nc_rot = nc  
st_geometry(nc_rot) = (st_geometry(nc) - ctrd) * 0.75 + ctrd  
  
plot(nc_rot[, "SID79"])
```

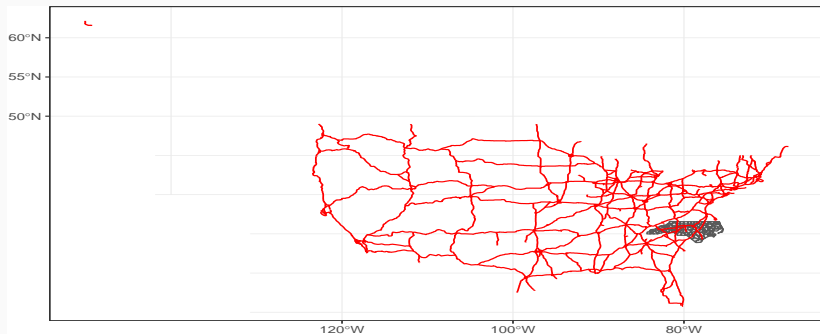
SID79



Highway Example

Highways

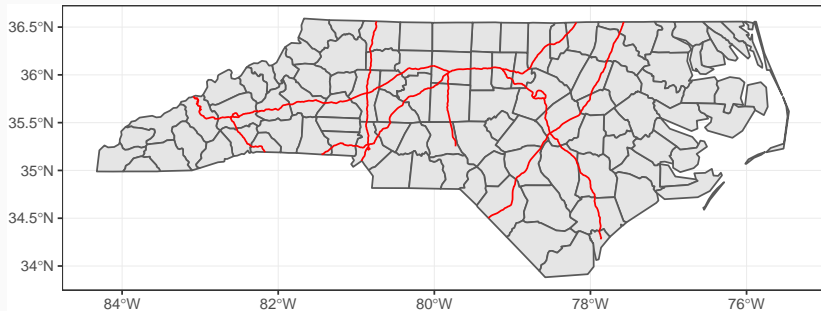
```
hwy = st_read("https://www2.stat.duke.edu/~cr173/Sta444_Fa18/slides/data/us_interstate  
             quiet=TRUE, stringsAsFactors=FALSE) %>% st_transform(st_crs(nc))  
ggplot() +  
  geom_sf(data=nc) +  
  geom_sf(data=hwy, col='red')
```



NC Interstate Highways

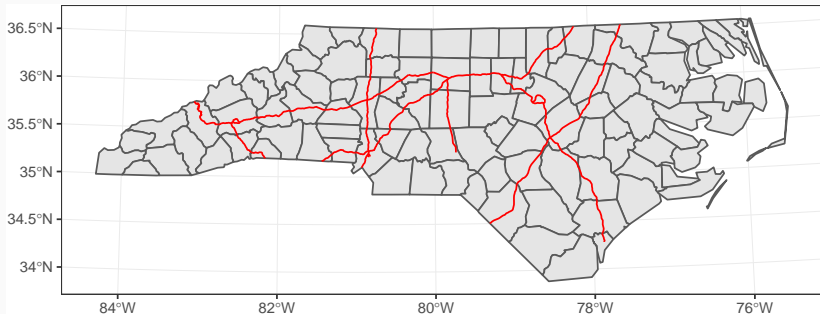
```
hwy_nc = st_intersection(hwy, nc)
## although coordinates are longitude/latitude, st_intersection assumes that they are

ggplot() +
  geom_sf(data=nc) +
  geom_sf(data=hwy_nc, col='red')
```



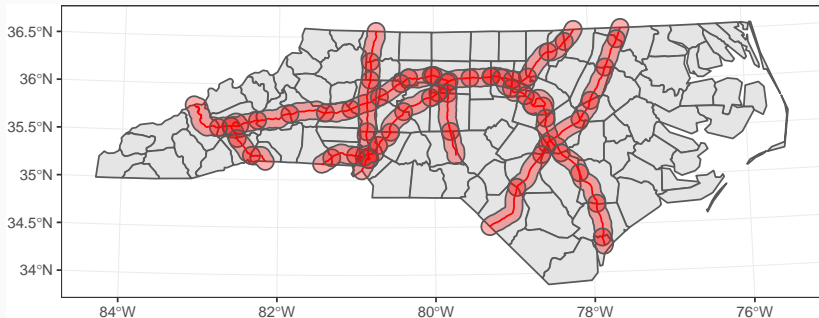
Counties near the interstate (Projection)

```
nc_utm = st_transform(nc, "+proj=utm +zone=17 +datum=NAD83 +units=m +no_defs")  
  
ggplot() +  
  geom_sf(data=nc_utm) +  
  geom_sf(data=hwy_nc, col='red')
```



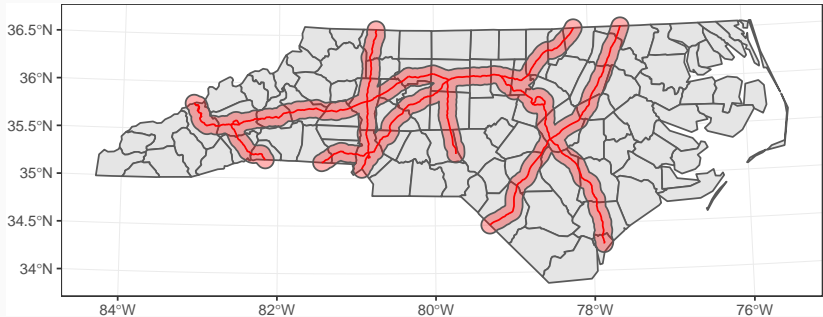
Counties near the interstate (Buffering)

```
hwy_nc_buffer = hwy_nc %>%  
  st_transform("+proj=utm +zone=17 +datum=NAD83 +units=m +no_defs") %>%  
  st_buffer(10000)  
  
ggplot() +  
  geom_sf(data=nc_utm) +  
  geom_sf(data=hwy_nc, color='red') +  
  geom_sf(data=hwy_nc_buffer, fill='red', alpha=0.3)
```



Counties near the interstate (Buffering + Union)

```
hwy_nc_buffer = hwy_nc %>%  
  st_transform("+proj=utm +zone=17 +datum=NAD83 +units=m +no_defs") %>%  
  st_buffer(10000) %>%  
  st_union() %>%  
  st_sf()  
  
ggplot() +  
  geom_sf(data=nc_utm) +  
  geom_sf(data=hwy_nc, color='red') +  
  geom_sf(data=hwy_nc_buffer, fill='red', alpha=0.3)
```



Example

How many counties in North Carolina are within 5, 10, 20, or 50 km of an interstate highway?

Gerrymandering Example

NC House Districts - 112th Congress

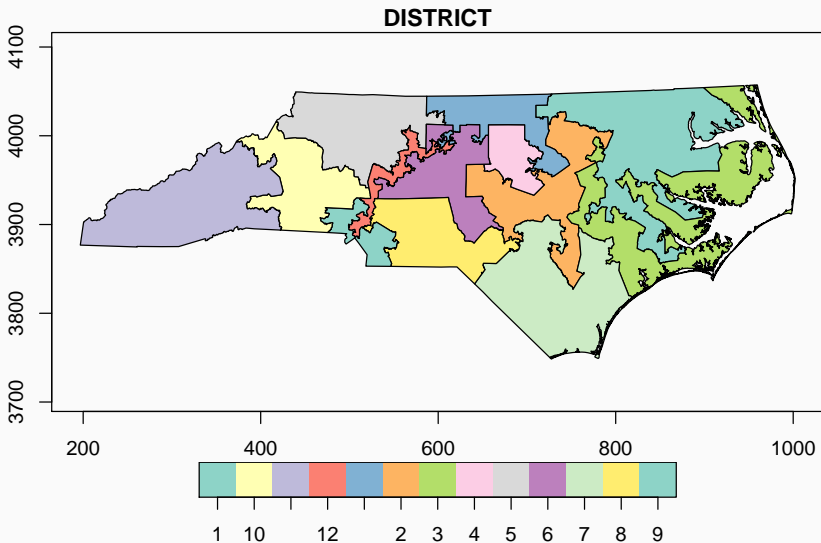
```
nc_house = st_read(  
  "https://www2.stat.duke.edu/~cr173/Sta444_Fa18/slides/data/nc_districts112  
  stringsAsFactors = FALSE, quiet = TRUE) %>%  
  select(ID, DISTRICT)
```

```
tbl_df(nc_house)
```

```
## # A tibble: 13 x 3
```

```
##   ID          DISTRICT          geo  
##   <chr>       <chr>          <MULTIPOLYGON [° ]  
## 1 037108112~ 1      (((-77.32845 35.35031, -77.35398 35.32799, -77.336  
## 2 037108112~ 2      (((-78.89928 35.12619, -78.89763 35.12859, -78.892  
## 3 037108112~ 3      (((-75.68266 35.23291, -75.68113 35.23237, -75.682  
## 4 037108112~ 4      (((-78.77926 35.78568, -78.77947 35.77568, -78.794  
## 5 037108112~ 5      (((-79.8968 36.38075, -79.89213 36.37108, -79.8928  
## 6 037108112~ 6      (((-80.4201 35.68953, -80.41483 35.68918, -80.4111  
## 7 037108112~ 7      (((-77.59169 34.40907, -77.58699 34.40611, -77.585  
## 8 037108112~ 8      (((-78.93373 34.95909, -78.94074 34.95789, -78.943  
## 9 037108112~ 9      (((-80.93058 35.18181, -80.9244 35.16754, -80.9210  
## 10 037108112~ 10     (((-81.04032 35.40447, -81.30021 35.41461, -81.287  
## 11 037108112~ 11     (((-84.00768 35.37262, -84.00831 35.37888, -84.011  
## 12 037108112~ 12     (((-80.4996 35.6493, -80.51926 35.63314, -80.52016  
## 13 037108112~ 13     (((-79.90775 36.3818, -79.90694 36.38382, -79.9016
```

```
nc_house = nc_house %>%  
  st_transform("+proj=utm +zone=17 +datum=NAD83 +units=km +no_defs")  
plot(nc_house[, "DISTRICT"], axes=TRUE)
```



Measuring Compactness - Reock Score

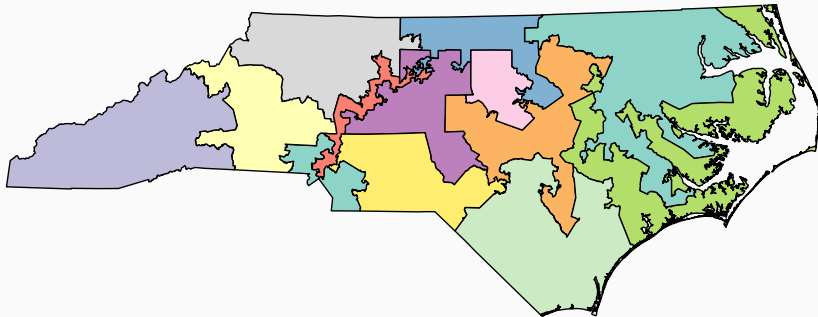
The Reock score is a measure of compactness that is calculated as the ratio of the area of a shape to the area of its minimum bounding circle.

```
circs = nc_house %>% st_geometry() %>% lwgeom::st_minimum_bounding_circle()  
  
sub = nc_house$DISTRICT == 1  
plot(circs[sub])  
plot(nc_house[sub,"DISTRICT"], add=TRUE)
```



```
plot(nc_house[, "DISTRICT"])  
plot(circls, add=TRUE)
```

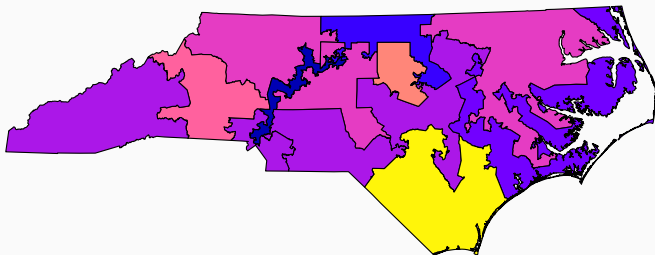
DISTRICT



Calculating Reock

```
nc_house = nc_house %>%  
  mutate(reock = st_area(nc_house) / st_area(circs))  
plot(nc_house[, "reock"])
```

reock [1]



0.2 0.3 0.4 0.5 0.6

```
tbl_df(nc_house) %>%  
  arrange(reock) %>%  
  print(n=13)
```

```
## # A tibble: 13 x 4
```

```
##   ID      DISTRICT reock      geo  
##   <chr>   <chr>   <S3: uni> <MULTIPOLYGON [km]  
## 1 0371081~ 12      0.115548~ (((545.2989 3945.167, 543.5279 3943.366, 5  
## 2 0371081~ 13      0.237185~ (((597.9666 4026.851, 598.0375 4027.076, 5  
## 3 0371081~ 3       0.266378~ (((984.071 3911.853, 984.2139 3911.8, 984.  
## 4 0371081~ 2       0.303422~ (((691.4142 3889.056, 691.5584 3889.327, 6  
## 5 0371081~ 9       0.338722~ (((506.3209 3893.208, 506.885 3891.626, 50  
## 6 0371081~ 8       0.341645~ (((688.6585 3870.456, 688.021 3870.31, 687  
## 7 0371081~ 11      0.343731~ (((226.7524 3918.52, 226.7167 3919.217, 22  
## 8 0371081~ 6       0.377542~ (((552.4693 3949.669, 552.9463 3949.632, 5  
## 9 0371081~ 1       0.377659~ (((833.6772 3918.082, 831.4469 3915.52, 83  
## 10 0371081~ 5      0.398978~ (((598.9506 4026.745, 599.3817 4025.677, 5  
## 11 0371081~ 10     0.410830~ (((496.3392 3917.9, 472.7445 3919.064, 473  
## 12 0371081~ 4      0.480027~ (((700.7066 3962.453, 700.712 3961.343, 69  
## 13 0371081~ 7      0.623895~ (((813.3007 3812.784, 813.7436 3812.471, 8
```

Raster Data

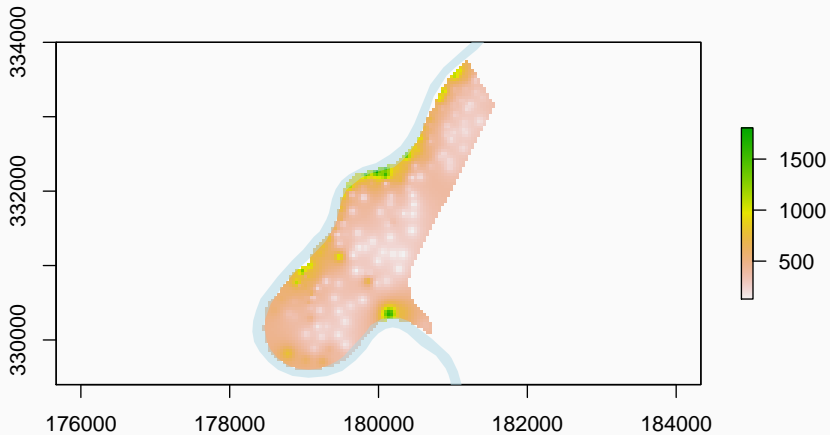
Example data - Meuse

```
meuse_rast = raster(system.file("external/test.grd", package="raster"))
```

```
meuse_rast
```

```
## class      : RasterLayer
## dimensions : 115, 80, 9200  (nrow, ncol, ncell)
## resolution : 40, 40  (x, y)
## extent     : 178400, 181600, 329400, 334000  (xmin, xmax, ymin, ymax)
## coord. ref.: +init=epsg:28992 +towgs84=565.237,50.0087,465.658,-0.406857,0.350733,
## data source : /usr/local/lib/R/3.5/site-library/raster/external/test.grd
## names      : test
## values     : 128.434, 1805.78  (min, max)
```

```
plot(meuse_rast)  
plot(meuse_riv, add=TRUE, col=adjustcolor("lightblue",alpha.f = 0.5), border=NA)
```



raster class

```
str(meuse_rast)
## Formal class 'RasterLayer' [package "raster"] with 12 slots
## ..@ file      :Formal class '.RasterFile' [package "raster"] with 13 slots
## .. .. ..@ name      : chr "/usr/local/lib/R/3.5/site-library/raster/external/t
## .. .. ..@ datanotation: chr "FLT4S"
## .. .. ..@ byteorder  : Named chr "little"
## .. .. .. ..- attr(*, "names")= chr "value"
## .. .. ..@ nodatavalue : num -3.4e+38
## .. .. ..@ NAchanged   : logi FALSE
## .. .. ..@ nbands     : int 1
## .. .. ..@ bandorder  : Named chr "BIL"
## .. .. .. ..- attr(*, "names")= chr "value"
## .. .. ..@ offset     : int 0
## .. .. ..@ toptobottom : logi TRUE
## .. .. ..@ blockrows  : int 0
## .. .. ..@ blockcols  : int 0
## .. .. ..@ driver     : chr "raster"
## .. .. ..@ open       : logi FALSE
## ..@ data       :Formal class '.SingleLayerData' [package "raster"] with 13 slots
## .. .. ..@ values     : logi(0)
## .. .. ..@ offset     : num 0
## .. .. ..@ gain       : num 1
## .. .. ..@ inmemory   : logi FALSE
## .. .. ..@ fromdisk   : logi TRUE
## .. .. ..@ isfactor   : logi FALSE
## .. .. ..@ attributes: list()
## .. .. ..@ haveminmax: logi TRUE
## .. .. ..@ min        : num 128
```

raster features

```
extent(meuse_rast)
```

```
## class      : Extent  
## xmin       : 178400  
## xmax       : 181600  
## ymin       : 329400  
## ymax       : 334000
```

```
dim(meuse_rast)
```

```
## [1] 115  80  1
```

```
res(meuse_rast)
```

```
## [1] 40 40
```

```
projection(meuse_rast)
```

```
## [1] "+init=epsg:28992 +towgs84=565.237,50.0087,465.658,-0.406857,0.350733,-1.87035,
```

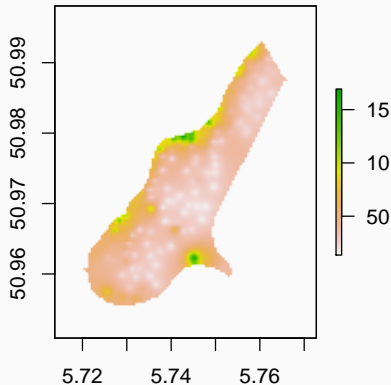
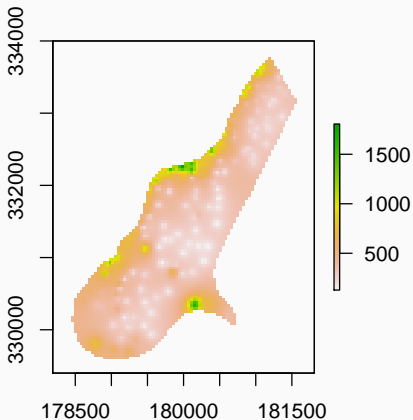
```
meuse_rast[20,]
```

```
## [1]      NA      NA      NA      NA      NA      NA      NA      NA  
## [9]      NA      NA      NA      NA      NA      NA      NA      NA  
## [17]     NA     NA     NA     NA     NA     NA     NA     NA  
## [25]     NA     NA     NA     NA     NA     NA     NA     NA  
## [33]     NA     NA     NA     NA     NA     NA     NA     NA  
## [41]     NA     NA     NA     NA     NA     NA     NA     NA  
## [49]     NA     NA     NA     NA     NA     NA     NA     NA  
## [57]     NA     NA     NA  749.536  895.292  791.145  607.186  511.044  
## [65]  468.404  399.325  350.362  306.180  300.483  310.082  283.940  285.771  
## [73]  304.709  309.690  301.799  308.753  328.357  345.611      NA      NA
```

Rasters and Projections

```
library(rgdal)
meuse_rast_ll = projectRaster(meuse_rast, crs="+proj=longlat +datum=NAD27 +no_defs")

par(mfrow=c(1,2))
plot(meuse_rast)
plot(meuse_rast_ll)
```



meuse_rast

```
## class      : RasterLayer
## dimensions  : 115, 80, 9200 (nrow, ncol, ncell)
## resolution  : 40, 40 (x, y)
## extent     : 178400, 181600, 329400, 334000 (xmin, xmax, ymin, ymax)
## coord. ref. : +init=epsg:28992 +towgs84=565.237,50.0087,465.658,-0.406857
## data source : /usr/local/lib/R/3.5/site-library/raster/external/test.grd
## names      : test
## values     : 128.434, 1805.78 (min, max)
```

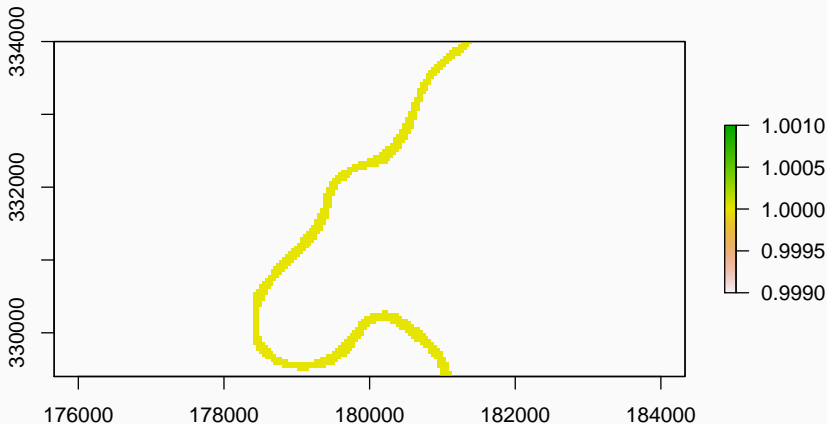
meuse_rast_ll

```
## class      : RasterLayer
## dimensions  : 131, 91, 11921 (nrow, ncol, ncell)
## resolution  : 0.000569, 0.00036 (x, y)
## extent     : 5.717362, 5.769141, 50.95089, 50.99805 (xmin, xmax, ymin, ymax)
## coord. ref. : +proj=longlat +datum=NAD27 +no_defs +ellps=clrk66 +nadgrids
## data source : in memory
## names      : test
## values     : 135.647, 1693.578 (min, max)
```

Simple Features \longleftrightarrow Rasters

```
meuse_riv_rast = rasterize(meuse_riv, meuse_rast)
## Error in (function (classes, fdef, mtable) : unable to find an inherited method for

meuse_riv_rast = rasterize(as(meuse_riv, "Spatial"), meuse_rast)
plot(meuse_riv_rast)
```



Rasters and Spatial Models

```
head(meuse)
```

```
## Simple feature collection with 6 features and 12 fields
## geometry type: POINT
## dimension: XY
## bbox: xmin: 181025 ymin: 333260 xmax: 181390 ymax: 333611
## epsg (SRID): 28992
## proj4string: +proj=sterea +lat_0=52.156160555555555 +lon_0=5.387638888888889 +k=0.
## cadmium copper lead zinc elev dist om freq soil lime landuse
## 1 11.7 85 299 1022 7.909 0.00135803 13.6 1 1 1 Ah
## 2 8.6 81 277 1141 6.983 0.01222430 14.0 1 1 1 Ah
## 3 6.5 68 199 640 7.800 0.10302900 13.0 1 1 1 Ah
## 4 2.6 81 116 257 7.655 0.19009400 8.0 1 2 0 Ga
## 5 2.8 48 117 269 7.480 0.27709000 8.7 1 2 0 Ah
## 6 3.0 61 137 281 7.791 0.36406700 7.8 1 2 0 Ga
## dist.m geometry
## 1 50 POINT (181072 333611)
## 2 30 POINT (181025 333558)
## 3 150 POINT (181165 333537)
## 4 270 POINT (181298 333484)
## 5 380 POINT (181307 333330)
## 6 470 POINT (181390 333260)
```

```
head(st_coordinates(meuse))
```

```
## X Y
## 1 181072 333611
## 2 181025 333558
## 3 181165 333537
## 4 181298 333484
```

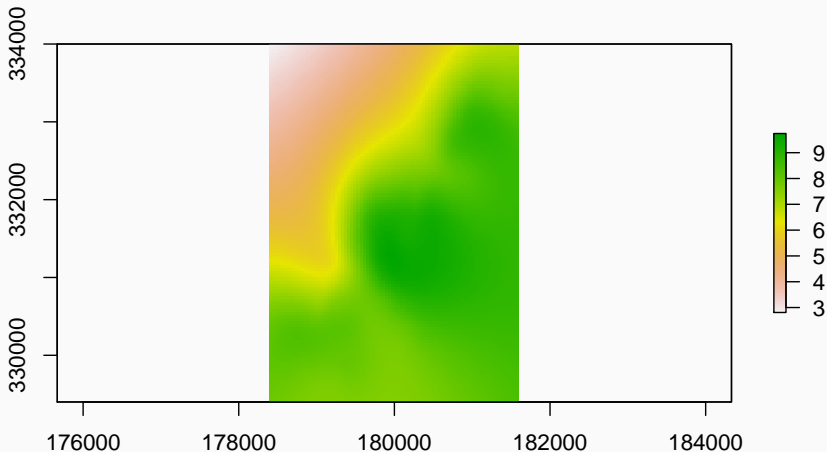


```
library(fields)
```

```
tps = Tps(x = st_coordinates(meuse), Y=meuse$elev)  
pred_grid = xyFromCell(meuse_rast, seq_along(meuse_rast))
```

```
meuse_elev_pred = meuse_rast  
meuse_elev_pred[] = predict(tps, pred_grid)
```

```
plot(meuse_elev_pred)
```



ggplot and rasters

```
p = rasterToPolygons(meuse_elev_pred) %>% st_as_sf()  
  
(ggplot() + geom_sf(data=meuse, aes(color=elev), size=1)) +  
(ggplot() + geom_sf(data=p, aes(fill=test), color=NA))
```

