Operations With Vector Data II

HES 505 Fall 2024: Session 11

Carolyn Koehn

Today's Plan

Objectives

- By the end of today, you should be able to:
  - Translate pseudocode commands into functional workflows
  - Articulate the importance of key arguments to `sf` functions
  - Generate new attributes and geometries from existing data.

Motivating Questions

Example questions

- What is the chronic heart disease risk of the 10 ID tracts that are furthest from hospitals?
- How may of ID are served by more than 1 hospital?
- What is the difference between the average risk of chronic heart disease in the tracts served by at least two hospitals compared to those that aren't served by any?

Key assummptions

- All hospital locations are contained in the landmarks dataset
- A hospital service area is defined as a 50km radius
- Hospital service areas can cross state lines.

Example 1

> What is the chronic heart disease risk of the 10 ID tracts that are furthest from hospitals?

What do we need to know?

- Where are the hospitals?
- How far are the hospitals from ID tracts?
- Which tracts are the furthest?
- What is the CHD risk?

Pseudocode

```
1. Load the hospital and cdc datasets
2. Align the data
3. Filter cdc so it only has Idaho tracts
4. Calculate distance from hospitals
5. Find top 10 tracts based on distance
6. Map chronic heart disease risk
```

Adding Functions

1. Load the hospital and cdc datasets

```
1 library(tidyverse)
2 library(sf)
3 library(tmap)
4 hospital.sf <- read_csv("../../data/2023/vectorexample/hospitals_pr
5   st_as_sf(., coords = c("longitude", "latitude"))
6 st_crs(hospital.sf)
```

```
Coordinate Reference System: NA
```

```
1 cdc.sf <- read_sf("../../data/2023/vectorexample/cdc_nw.shp")
2 st_crs(cdc.sf)$epsg
```

```
[1] 4269
```

Adding Functions

2. Align the data

```
st_crs(hospital.sf) <- 4326

hospital.sf.proj <- hospital.sf %>%
  st_transform(., crs=st_crs(cdc.sf))

st_crs(hospital.sf.proj) == st_crs(cdc.sf)
```

```
[1] TRUE
```

```
identical(st_crs(hospital.sf.proj), st_crs(cdc.sf))
```

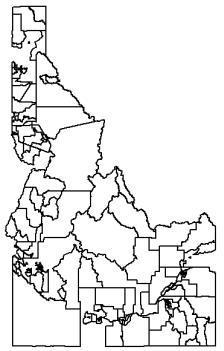
```
[1] TRUE
```

Adding Functions

3. Filter cdc so it only has Idaho tracts

```
1  cdc.idaho <- cdc.s
2    filter(STATEFP =
```

```
1  plot(st_geometry(cdc.idaho))
```



Adding Functions

4. Calculate distance from hospitals

```
1 nearest.hosp <- st_nearest_feature(cdc.idaho, hospital.sf.proj)
2 str(nearest.hosp)
```

```
int [1:191] 6 45 45 45 3 3 3 3 6 3 ...
```

```
1 nearest.hosp.sf <- hospital.sf.proj[nearest.hosp,]
2 hospital.dist <- st_distance(cdc.idaho, nearest.hosp.sf, by_element
3 str(hospital.dist)
```

```
Units: [m] num [1:191] 29414 46610 39432 32817 23548 ...
```

Adding Functions

5. Find top 10 counties based on distance

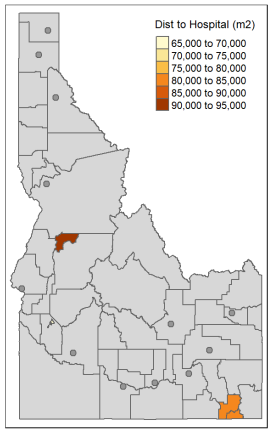
```
1 cdc.idaho.hosp <- cdc.idaho %>%
2   mutate(., disthosp = hospital.dist)
3
4 cdc.furthest <- cdc.idaho.hosp %>%
5   slice_max(., n=10, order_by= disthosp)
6
7 head(cdc.furthest$disthosp)
```

```
Units: [m]
[1] 94622.55 83296.77 80916.73 70646.03 70292.69 69877.25
```

Adding Functions

6. Map chronic heart disease risk

```
1  library(tmap)
2
3  tm_shape(tigris::counties("ID", progress_bar=FALSE)) +
4    tm_polygons() +
5    tm_shape(cdc.furthest) +
6    tm_polygons("disthosp", title="Dist to Hospital (m2)") +
7    tm_shape(hospital.sf.proj[cdc.idaho,]) +
8    tm_symbols(size=0.25)
```



Example 2

> How may of ID are served by more than 1 hospital?
> km^2

What do we need to know?

- Where are the hospitals?
- What is the service area for each hospital?
- Where do those service areas overlap?
- How big is the overlap area?

Pseudocode

```
1. Load the hospital dataset and add projection
2. Buffer hospitals by service area
3. Find intersection of service areas
4. Calculate area of overlap
```

Adding Functions

1. Load the hospital dataset and add projection

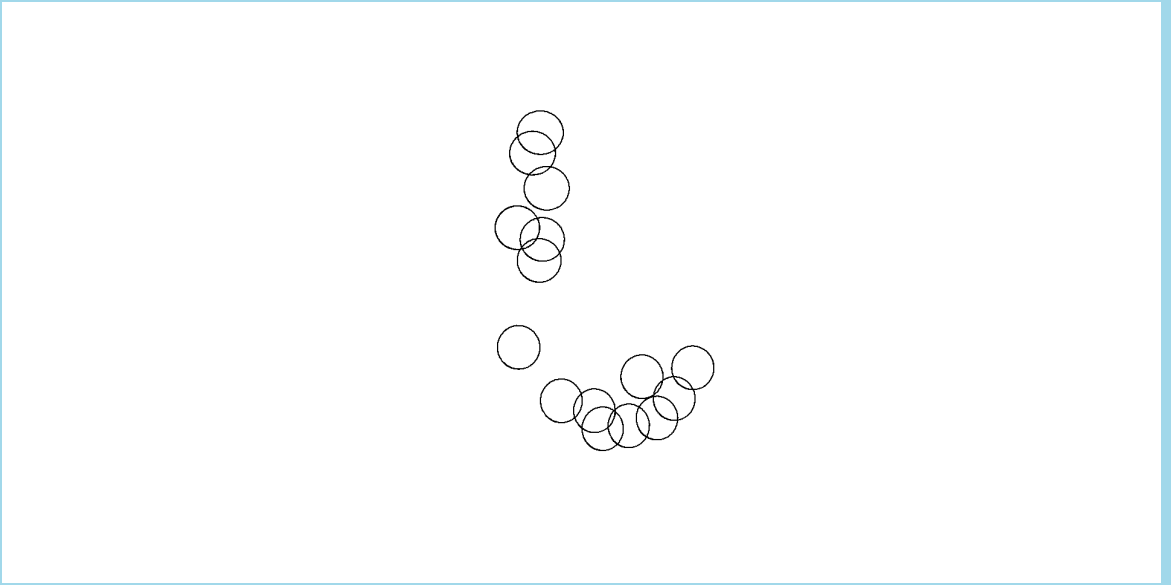
```
1 hospital.sf <- read_csv("../../data/2023/vectorexample/hospitals_pr
2   st_as_sf(., coords = c("longitude", "latitude"))
3
4 st_crs(hospital.sf) <- 4326
```

Adding Functions

2. Buffer hospitals by service area

```
1 hospital.buf <- hc
2   filter(STATEFP =
3   st_buffer(., dis
```

```
1 plot(st_geometry(hospital.buf))
```



Adding Functions

3. Find intersection of service areas

```
1 hospital.int <- hospital.buf %>%
2   st_intersection()
3 all(st_is_valid(hospital.int))
```

Troubleshooting Process

Googling error code with package name and R lead to this issue page: https://github.com/r-spatial/sf/issues/2143

```
hospital.buf <- hospital.buf %>%
  # project to planar CRS to get rid of warning
  st_transform(., crs = 5070) %>%
  # remove +/- duplicate buffer
  filter(!row_number() %in% c(7,8))


hospital.int <- hospital.buf %>%
  st_intersection(.)
all(st_is_valid(hospital.int))
```

```
[1] TRUE
```

```
hospital.int.overlaps <- hospital.int %>%
  filter(n.overlaps > 1)
```

Adding Functions

4. Calculate area of overlap

```
1 overlap.areas <- st_area(hospital.int.overlaps)
2
3 area_m2 <- sum(overlap.areas) + units::set_units(pi*50000^2, m^2)
4
5 units::set_units(area_m2, km^2)
```

```
32664.78 [km^2]
```

Example 3

> What is the difference between the average risk of chronic heart disease in the counties served by at least two hospitals compared to those that aren't served by any?

What do we need to know?

Pseudocode

Adding Functions

Plotting the Results