

Prep7

Sebastian Montesinos

Due by midnight, Monday April 11

Reminder: Prep assignments are to be completed individually. Upload a final copy of the .Rmd and renamed .pdf to your private repo, and submit the renamed pdf to Gradescope.

Reading

The associated reading for the week is Chapter 17. The focus is on working with Spatial data, which basically means, making maps and showing appropriate data on them.

We will also be introducing the final project this week on Thursday. Remember to submit your reflection due at the same time as this Prep from the Shiny project using the Google form linked on Moodle!

Note: The practice set this week contains one question on networks to cover the material from last week.

1 - Spatial data basics

The chapter introduces a number of spatial data specific terms.

part a - What are two examples of spatial data structure formats?

Solution: Two spatial data structures are shapefiles and KML.

part b - What does EPSG stand for in EPSG codes? Why is it important to know the EPSG number, if applicable for your map?

Solution: EPSG stands for European Petroleum Survey Group, and the codes provide a shorthand for the type of projection/gps system being used for your map.

part c - What is the primary package used in the textbook to work with spatial data?

Solution: The primary package used in the textbook is sf, which provides a set of tidyverse-friendly functions for spatial data.

part d - What term/concept did you find hardest to understand from the reading?

Solution: The idea of coordinate reference systems (CRS) and how to pick/use one when creating a map in R.

2 - Reproducing a map

Section 17.1 introduces *shapefiles*, and includes an example of working with a shapefile to re-create Snow's cholera map. This exercise mostly follows along with the text code.

Setup

Load the **sf** package in the **setup** code chunk. Verify your working directory is the folder *this .Rmd* file is in. Then, create a *data* subfolder in this directory.

part a - Run the code below line-by-line to understand what each part is doing. You can use *command + enter* or *ctrl + enter* to run one selected or highlighted set of code at a time. Confirm that you get a figure similar to that of Figure 17.3 in the textbook.

```
# Download SnowGIS_SHP zip file
download.file("http://rtwilson.com/downloads/SnowGIS_SHP.zip",
              destfile = "data/SnowGIS_SHP.zip")

# Unzip file in same folder
unzip(zipfile = "data/SnowGIS_SHP.zip",
      exdir = "data")

# Create filepath to unzipped files so we don't need to re-type
data_path <- "data/SnowGIS_SHP"

# List files in SnowGIS_SHP
list.files(data_path)

## [1] "Cholera_Deaths.dbf"           "Cholera_Deaths.prj"
## [3] "Cholera_Deaths.sbn"          "Cholera_Deaths.sbx"
## [5] "Cholera_Deaths.shp"          "Cholera_Deaths.shx"
## [7] "OSMap.tfw"                   "OSMap.tif"
## [9] "OSMap_Grayscale.tif"         "OSMap_Grayscale.tif"
## [11] "OSMap_Grayscale.tif.aux.xml" "OSMap_Grayscale.tif.ovr"
## [13] "Pumps.dbf"                  "Pumps.prj"
## [15] "Pumps.sbx"                  "Pumps.shp"
## [17] "Pumps.shx"                  "README.txt"
## [19] "SnowMap.tfw"                "SnowMap.tif"
## [21] "SnowMap.tif.aux.xml"        "SnowMap.tif.ovr"

# List layers
st_layers(data_path)

## Driver: ESRI Shapefile
## Available layers:
##   layer_name geometry_type features fields
## 1 Cholera_Deaths      Point       250      2
## 2       Pumps          Point        8       1
```

```

# Load second layer
cholera_deaths <- st_read(data_path, layer = "Cholera_Deaths")

## Reading layer 'Cholera_Deaths' from data source
##   'C:\Users\Sebastian\OneDrive\Documents\Git\stat231\Homework\SMontesinos_Prep7\data\SnowGIS_SHP'
##   using driver 'ESRI Shapefile'
## Simple feature collection with 250 features and 2 fields
## Geometry type: POINT
## Dimension:      XY
## Bounding box:  xmin: 529160.3 ymin: 180857.9 xmax: 529655.9 ymax: 181306.2
## Projected CRS: OSGB 1936 / British National Grid

class(cholera_deaths)

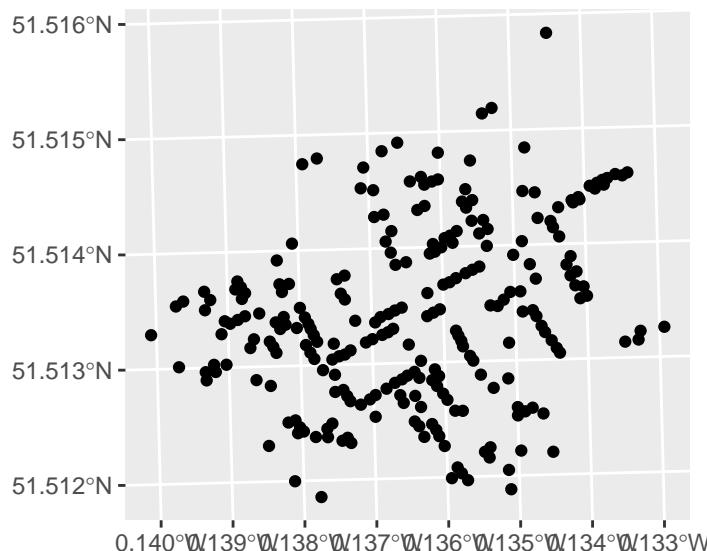
## [1] "sf"           "data.frame"

head(cholera_deaths)

## Simple feature collection with 6 features and 2 fields
## Geometry type: POINT
## Dimension:      XY
## Bounding box:  xmin: 529308.7 ymin: 181006 xmax: 529336.7 ymax: 181031.4
## Projected CRS: OSGB 1936 / British National Grid
##   Id Count          geometry
## 1 0    3 POINT (529308.7 181031.4)
## 2 0    2 POINT (529312.2 181025.2)
## 3 0    1 POINT (529314.4 181020.3)
## 4 0    1 POINT (529317.4 181014.3)
## 5 0    4 POINT (529320.7 181007.9)
## 6 0    2 POINT (529336.7 181006)

# Context-less plot
ggplot(cholera_deaths) +
  geom_sf()

```

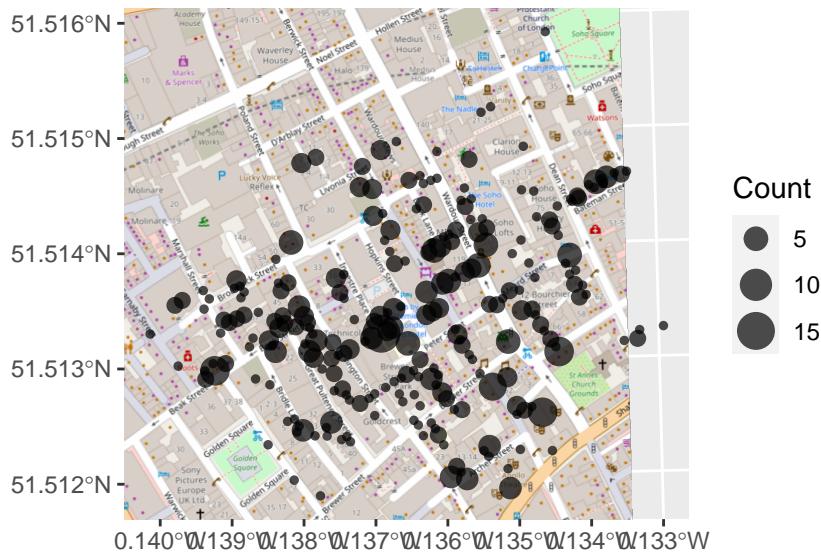


Solution: The map generated is like the one displayed in 17.3

part b - Now use the **ggspatial** package to overlay the London street map. Make sure you (install then) load the **ggspatial** package in the **setup** code chunk before running the code. What is wrong with this map?

```
ggplot(cholera_deaths) +  
  annotation_map_tile(type = "osm", zoomin = 0) +  
  geom_sf(aes(size = Count), alpha = 0.7)
```

Zoom: 17

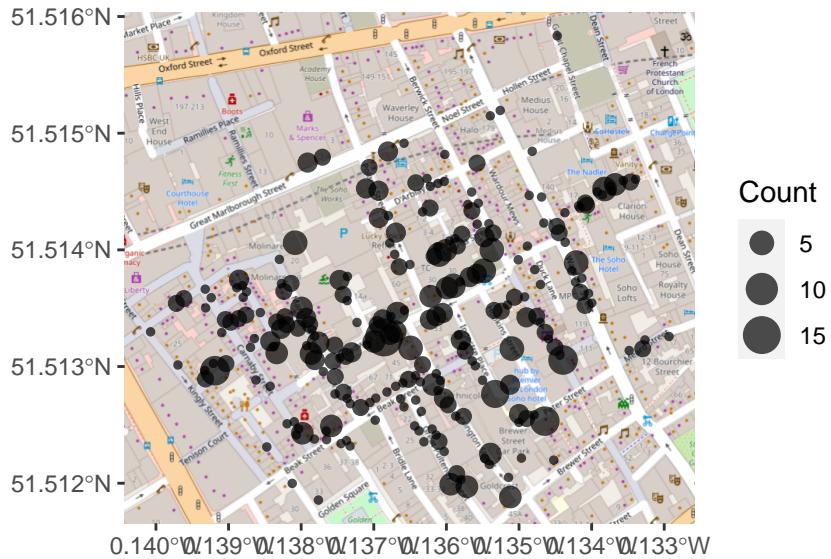


Solution: The coordinates in the ggspatial package and choleraDeaths are not in the same units.

part c - Set the coordinates from the cholera data as the `espg:27700` coordinate system using `st_set_crs()`, then transform them to the `espg:4326` system using `st_transform()`, and finally plot the new, correctly projected data. What does "crs" stand for in this code?

```
choleralatlong <- choleraDeaths %>%  
  st_set_crs(27700) %>%  
  st_transform(4326)  
  
ggplot(choleralatlong) +  
  annotation_map_tile(type = "osm", zoomin = 0) +  
  geom_sf(aes(size = Count), alpha = 0.7)
```

Zoom: 17



Solution: ‘crs’ stands for coordinate reference system, a method for keeping track of geographic locations that comes in multiple forms.

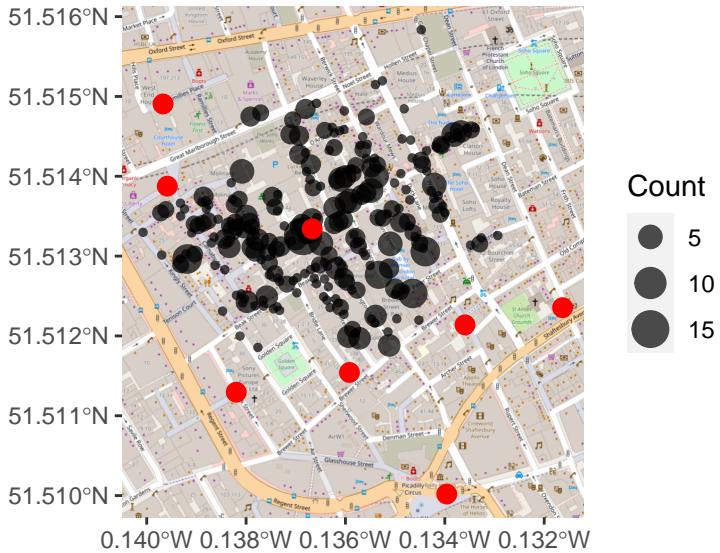
part d - Repeat the layer loading and coordinate transformation procedure to add the water pumps to the plot. Looking at your plot, do you agree that the water pump on Broad Street seems to have been the problem for this outbreak?

```
pumps_latlong <- st_read(data_path, layer = "Pumps") %>%
  st_set_crs(27700) %>%
  st_transform(4326)

## Reading layer 'Pumps' from data source
##   'C:\Users\Sebastian\OneDrive\Documents\Git\stat231\Homework\SMontesinos_Prep7\data\SnowGIS_SHP'
##   using driver 'ESRI Shapefile'
## Simple feature collection with 8 features and 1 field
## Geometry type: POINT
## Dimension:      XY
## Bounding box:  xmin: 529183.7 ymin: 180660.5 xmax: 529748.9 ymax: 181193.7
## Projected CRS: OSGB 1936 / British National Grid

ggplot(cholera_latlong) +
  annotation_map_tile(type = "osm", zoomin = 0) +
  geom_sf(aes(size = Count), alpha = 0.7) +
  geom_sf(data = pumps_latlong, size = 3, color = "red")

## Zoom: 17
```



Solution: This plot shows nearly all the deaths concentrated near people at the broad street water pump, so I agree that this pump seems to have been the problem for the outbreak.

part e - Finally, try out the code below to create a dynamic map using the **leaflet** package (install and load as before!). Zoom in and out of the map to confirm that (1) there is a death in the middle of Hopkins Street, and (2) there is a pump near the intersection of Broadwick Street and Lexington Street. What seem to be the main types of businesses along the modern day Kingly street?

```
# create dynamic map
leaflet() %>%
  addTiles() %>%
  addCircleMarkers(data = cholera_latlong,
    radius = ~ Count,
    color = "navy",
    stroke = FALSE,
    fillOpacity = 0.7) %>%
  addCircleMarkers(data = pumps_latlong,
    radius = 6,
    color = "red",
    stroke = FALSE,
    fillOpacity = 0.7)
```

Solution: The main businesses on modern-day Kingsly street seem to be restaurants and department stores.