

Narrative

Zachary Klamann

12/09/2019

1. Substantive goal:

The initial goal of the project was to collect data on violent events in the Spartacist uprising, map those onto specific areas that featured some basic socio-economic and election data to see whether violence was more prevalent in some areas than others. That initial attempt did not work out for a number of reasons, but mostly because the socioeconomic data and the map data I was able to find were in different administrative units, despite being supposedly pulled from the same statistical records. As such, they couldn't be mapped together.

Instead, using the data I was able to find, I have coded major violent events at two levels. The first is major violent events in the 'Revolutionary' period of the Weimar Republic – roughly from the Mutiny in Kiel in November 1918 that officially spelled the end of the German War effort and the German monarchy to the Beer Hall Putsch in Munich in October 1923. The second is the major events of the Spartacist Uprising in January 1919 Berlin. These are demonstrated graphically with maps.

2. Data Collection

This was the biggest challenge. Working with historic data in another language added a level of complexity. For the violent events, I relied on a variety of sources. For the Spartacist Uprising, I based the major events and the coding of basic information on Mark Jones's *Founding Weimar*, a recent piece of historical work on violence in the revolutionary period. This gave the largest events and cross-referenced participant counts based on newspapers covering the ideological spectrum. For the major events, I relied partially on background knowledge, partially on timelines from Wikipedia, and then cross-referenced participant numbers with Pierre Broue's massive book on the revolutionary period. That gave decent numbers, but some had to be estimated for the purposes of graphing.

The map data was a different issue entirely. There are quite a few shapefiles for the Weimar period, including really comprehensive, yearly-ones from Mosaic, but for my initial research none of them matched with the social and election data I had gotten. For some reason, both claimed to have gotten their information from the German Imperial Statistical Office but somehow the MOsaic map had 200 fewer districts and different district numbers, so there was no way to merge the datasets. Ultimately, then, I switched over to the current project, subsetting for the Berlin districts. For the Spartacist map, I used a modern map of the Berlin *Orteile* (neighborhoods) that I cross-referenced with a period map to make sure they're still roughly the same. To get the raster of the period map, I used a massive photo file from a Harvard archive. Finally, for the map of Germany, I used a shapefile from a 1990s article on Nazi voting in 1930.

3. Cleaning and pre-processing the data

Maps: For the violent events across Germany map, all I had to do initially was dissolve the internal boundaries using `ms_dissolve()` from the `rmapshaper` package. The map for Berlin was considerably more frustrating. I initially plotted the data over a blank Berlin shapefile, but it wasn't very interesting or descriptive. So, I wanted to get an actual map of Berlin from the period as the backdrop for the event data. The easiest way

to ensure this actually mapped correctly was to georeference an image file then load it as a raster file. This required finding a high-quality image file, downloading QGIS and learning how exactly one uses QGIS to georeference an image file. So, I loaded into QGIS a shapefile of Germany, courtesy of the EU, and my Orteile Map of Berlin, courtesy of the city of Berlin. These referenced to a CRS that the EU uses automatically, which I importantly did not realize. To georeference the image file, I needed to import it through the raster georeference function then plot points on the image file with coordinates so the image could be plotted onto space. Sadly, the actual shapefiles I had loaded into QGIS had no distinguishing features so I couldn't use those to get coordinates, and instead used Googlemaps, but these were basic long-lat coordinates and when I tried to get the image to rasterize, it wouldn't load into the correct position because of this particular coordinate system the shapefiles were using. I eventually found an online translating service and redid the process before getting it to load. I then messed with the layers on this so now the district boundaries in Berlin show up lightly on the mapfile to give a bit of structure. The next step was translating this map into a useable format in R. I saved it as a geoTiff file and loaded it as a RasterStack. Except, when it plotted it looked like a heat map of some kind. The internet told me it had an extra level, so I took out the alpha level, and it now plotted in RGB and acted like any raster file, so that gave me a useable, historic shapefile!

Next was getting the event files into geospatial files. As the code references a bit, I got pretty tripped up giving them CRS. Initially it didn't work. One issue was I entered latitude as x values and longitude as y values, but that needed to be flipped, so for awhile I was pretty confused as to why transformations weren't working. Eventually, I figured out that I needed to assign them some random CRS first then transform them into the CRS their corresponding maps used. I've put this issue in here, simplified from how many different things I tried.

```
map1919 <- readOGR(dsn = "../Data/spatial_data/mosaic_data_1931set/German_Empire_1919_Stat_v.1.0.shp")
violent_events <- read.csv("../Data/spatial_data/Original datasets/violent_events.csv")

#Let's take a look at the structure of this:
str(violent_events)

#Needs to be transformed to a SPDF so it can be plotted
coordinates(violent_events) <- c("POINT_X", "POINT_Y")

proj4string(violent_events) <- CRS("+init=epsg:4326")
proj4string(violent_events)

#Still in the wrong zone of the UTM, so we need to transform to match
violent_events <- spTransform(violent_events, proj4string(map1919))

#let's check the ranges
bbox(violent_events)
bbox(map1919)

#so something has gone wrong. Let's troubleshoot here
violent_events <- read.csv("../Data/spatial_data/Original datasets/violent_events.csv")
proj4string(map1919)
coordinates(violent_events) <- c("POINT_Y", "POINT_X")
projection(violent_events) <- "+init=epsg:4326"

violent_events <- spTransform(violent_events, CRS("+proj=longlat +datum=WGS84 +no_defs +ellps=WGS84 +towgs84=0,0,0"))

map1919 <- spTransform(map1919, CRS("+proj=longlat +datum=WGS84 +no_defs +ellps=WGS84 +towgs84=0,0,0"))

bbox(violent_events)
bbox(map1919)
```

#SUCCESS! They now map onto the same space and look decently alright!

4. Visualization

For the visualizations, I ran into a couple problems. I wanted to do animated maps, cause that sounds fun and would be a good way of showing time-series data. I went through a couple online tutorials, mainly using tmap as I was using sp objects rather than th sf objects that go with ggplot. The coding itself is pretty straightforward, just requiring that you use facet wrapping but instead of [by =] you use [along =] and save the file through tm_animate. I got to that point and it refused to work. The issue was the animate functions rely on ImageMagick to create their files. I first downloaded ImageMagick, but got the same error, so redownloaded it and restarted R with no improvement. I then looked through some discussions of the issue on StackExchange and Reddit to see that it might be the way tm_animate was calling on the convert.exe program in ImageMagick. So I tried to create a shortcut to that file because the path might have been messed up by the space in “Program Files”. That didn’t work. Neither did redefining the path. I still have absolutely no idea why it started working the next day, but suddenly it would animate! This was the final thing I did:

```
#This did not work initially so troubleshooting
library(animation)
Sys.setenv(PATH = paste("C:/Program Files/ImageMagick/bin",
                        Sys.getenv("PATH"), sep = ";"))
library(installr)
install.ImageMagick(URL = "http://www.imagemagick.org/script/binary-releases.php")
install.packages("magick")
```

For the Spartacist map, I decided to not animate because it didn’t look as good animated as it does as a static map. There are too few events, given the uprising only had about three days of intense fighting. Ultimately, with animating, much of my time has been learning the various tools involved with tmap and trying to get the aesthetics to behave like I want them to. For now, I still haven’t gotten the maps how I want them. The huge disparity in the number of participants – in Berlin this ranged from a few dozen people shooting at each other to as many as 100,000 protesters – made it hard to make the smaller events, that are still very important, visible. Tm_bubble has been pretty frustrating to use because I can’t get the size of the bubbles to customize, but ultimately, the map file and event data is legible. Whenever I save it, the margins are way off, too, so that’s frustrating, but I’ve tried messing with the margins in tm_layout to no avail.

The visualization of Berlin is in the end.

5. Future Work

I have a few ideas on how to move forward with this. I’d first like to actually plot a comprehensive list of events at both these levels and second I’d like to do more work with QGIS on historic map data. Ultimately, completing comprehensively coded datafiles for the areas that saw communist and proto-fascistic uprisings in Germany would be great. It would give me some leverage on answering questions about spatial distributions of violence during the Revolutionary period, which could be combined with available data on who participated and to what extent. As far as the computational side, buliding map files that include far more information on the landscape and features of the city would be great and can be done with QGIS.

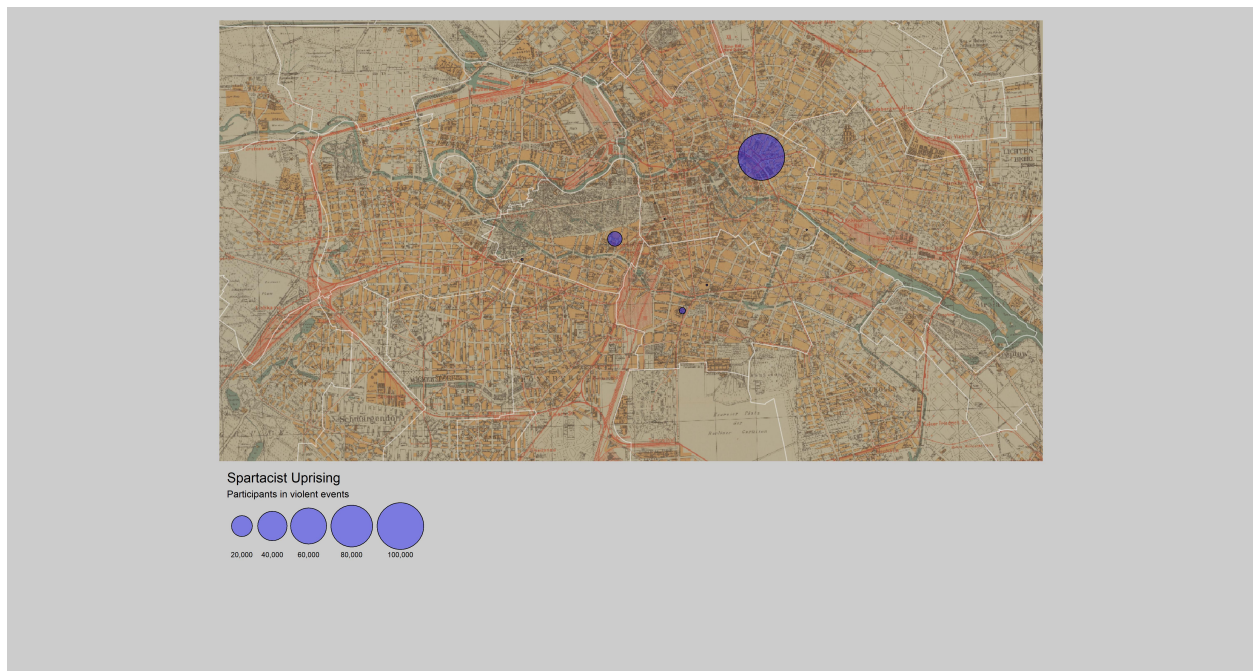


Figure 1: Major Violent Events in Berlin, January 1919