

Oregon Precincts and the 2020 Presidential Election

Ian McDonald

1/25/2021

Overview

This project is a relatively simple demonstration of mapping technology available to R users, using Oregon election precincts and results from the 2020 general election.

In some ways, the result evokes the New York Times's ["extremely detailed map of the 2020 election"] (<https://www.nytimes.com/interactive/2021/upshot/2020-election-map.html>) which first appeared following the election of 2016. The New York Times tool uses the Mapbox platform and the Open Street Maps layer.

Like my project, this tool adapts precinct data. At the precinct level, compiling this data is labor intensive, and precinct results are usually not aggregated by states along with the results reported by counties. Rather, they are reported by individual counties, and typically not provided as exportable datasets, but as PDF's with inconsistent formatting. They noted, "scraped and standardized precinct-level election results from around the country, and joined this tabular data to precinct GIS data to create a nationwide election map." [[^https://github.com/TheUpshot/presidential-precinct-map-2020](https://github.com/TheUpshot/presidential-precinct-map-2020)].

text[^a-random-footnote-label]

PDF scraping software such as Tabula can help the process but some kind of manual transcription or editing is necessary. As the Times's tool shows, the compilation is incomplete at this writing (2/10/2021). [[^The Times's data compilation involved the efforts of several contributors, including the Open Elections project directed by Derek Willis. See credits at https://github.com/TheUpshot/presidential-precinct-map-2020. \]](#) The data used by the Times are available in a GEOjson download which I will show below.

My project uses shapefiles from various agencies around Oregon and extracts downloaded from OpenElections and some of the counties.

The map uses tmap's Leaflet implementation and let's users drill into a display that shows address-level two party election result with color coding and vote detail. This example displays percentages of the two party presidential election results.

Libraries

This list and sequence should be cleaned up. I'm using the easypackages function *libraries*.

Addresses for input

This chunk can produce an address marker using the geocode_OSM function from tmap. An example is shown here. In the next version this will use a text input for Shiny.

```
address_input <- c("University of Oregon")

address_raw <- geocode_OSM(address_input)

address <- tibble(
  place=c(address_input),
```

```

longitude=address_raw$coords["x"],
latitude=address_raw$coords["y"])

address_sf <- st_as_sf(address, coords = c("longitude", "latitude"),
                      crs = "NAD83", agr = "constant")

```

Shapefiles and sf objects

Easiest to find old fashioned shapefiles with all the headaches described by Lovelace. I read them as sf objects using `st_read` function. In this case, I combine an Oregon shapefile for counties with Clark County in Washington, included because Clark County is part of the Portland Metro area.

I completed the sections for the Portland Metro area first then added the rest of Oregon.

County Shapefiles

I added labels from the USMAP fips lookup. Is there a better lookup table provided by the Census bureau from tidycensus?

Notice that I transform the sf objects into a coordinate reference system of NAD83, although they appear to have been read with that CRS. The `bind_rows` function doesn't appear to work otherwise.

I need to add a state boundary, unless Oregon decides to annex Clark County.

[Need to add a section with sources for all data and shapefiles]

Oregon precinct boundaries came from <https://opendata.imspdx.org/dataset/november-2018-election-oregon-results-by-precinct>

```

oregon_counties_sf <- st_read("shapefiles/oregon_counties/counties.shp",
                             quiet = TRUE) %>%
  st_transform(or_counties, crs = "NAD83") %>%
  select(!COUNTY)

washington_state_counties_sf <-
  st_read("shapefiles/washington_counties/WA_County_Boundaries.shp", quiet = TRUE) %>%
  st_transform(crs = "NAD83") %>%
  mutate(STFID = as.character(JURISDIC_5))

washington_state <- st_union(washington_state_counties_sf)

clark_county_sf <- washington_state_counties_sf %>%
  filter(JURISDIC_2 == "Clark")

oregon_counties_and_clark_sf <- bind_rows(oregon_counties_sf, clark_county_sf) %>%
  st_transform(crs = "NAD83")

oregon_fips <- fips_info(oregon_counties_and_clark_sf$STFID) %>%
  mutate(county = gsub(" County", "", county))

oregon_counties_and_clark_sf <- left_join(oregon_counties_and_clark_sf, oregon_fips, by= c("STFID" = "f

oregon_state <- st_union(oregon_counties_sf)

oregon_bb <- st_bbox(oregon_counties_and_clark_sf)

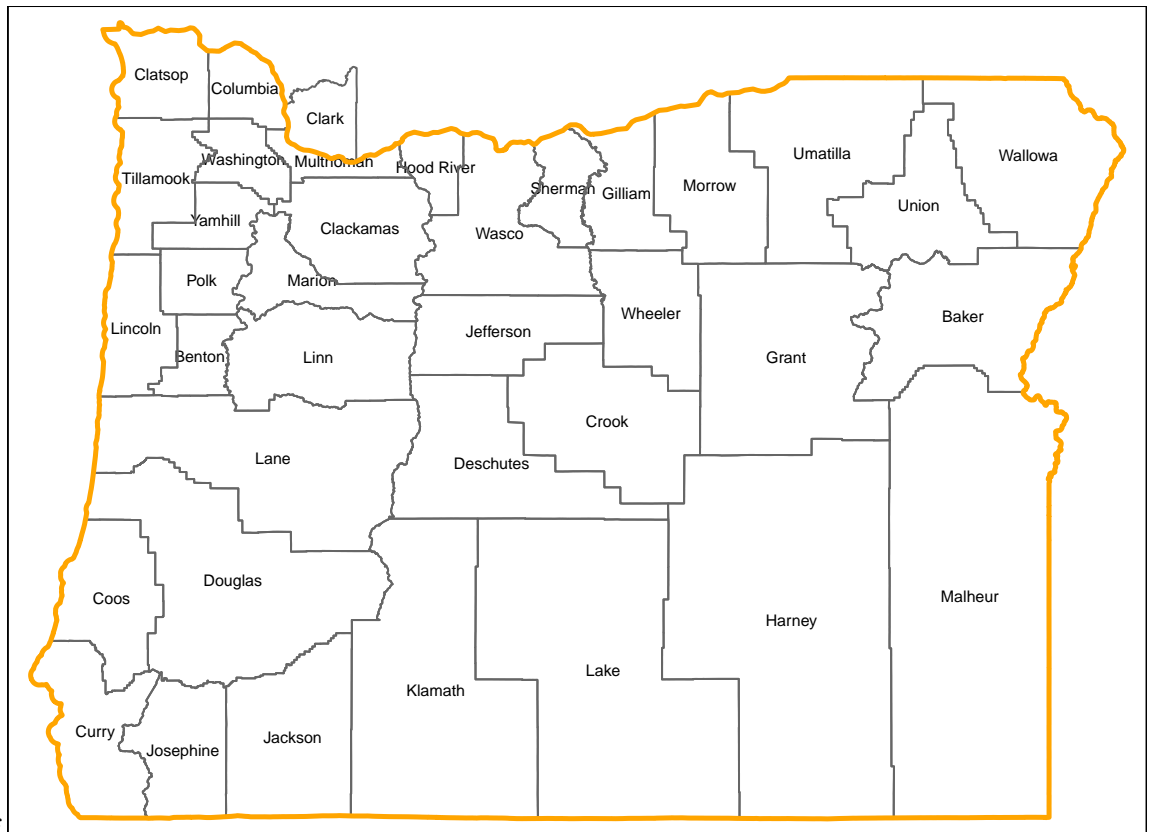
tmap_mode("plot")

```

```
## tmap mode set to plotting
oregon_county_tm <-
  tm_shape(oregon_counties_and_clark_sf, bbox = oregon_bb) +
    tm_text("county", size = 0.5) +
    tm_polygons(alpha = 0, id = "county") +

  tm_shape(oregon_state) +
    tm_borders(col = "orange", lwd = 2.5)

oregon_county_tm
```



shapefiles-1.pdf

Precinct Shapefiles

Metro Portland the rest of the state are processed separately.

```
tmap_mode("plot")
tmap_options(max.categories = 50)

metro_portland_precinct_sf <- st_read("shapefiles/metro_portland_precinct/precinct.shp",
                                       quiet = TRUE) %>%
  select(precinct = PRECINCTID, county = COUNTY, geometry) %>%
  mutate(county = case_when(
    county == "W" ~ "Washington",
    county == "M" ~ "Multnomah",
    county == "C" ~ "Clackamas",
    TRUE ~ "Other"
```

```

    )
  ) %>%

  st_transform(crs = "NAD83")

clark_wa_precincts_sf <- st_read("shapefiles/clark_precinct_shapefiles/Precinct.shp",
                                quiet = TRUE) %>%
  select(precinct = PRECINCT, geometry) %>%
  mutate(precinct = str_c("K", as.character(precinct)), county = "Clark") %>%
  st_transform(crs = "NAD83")

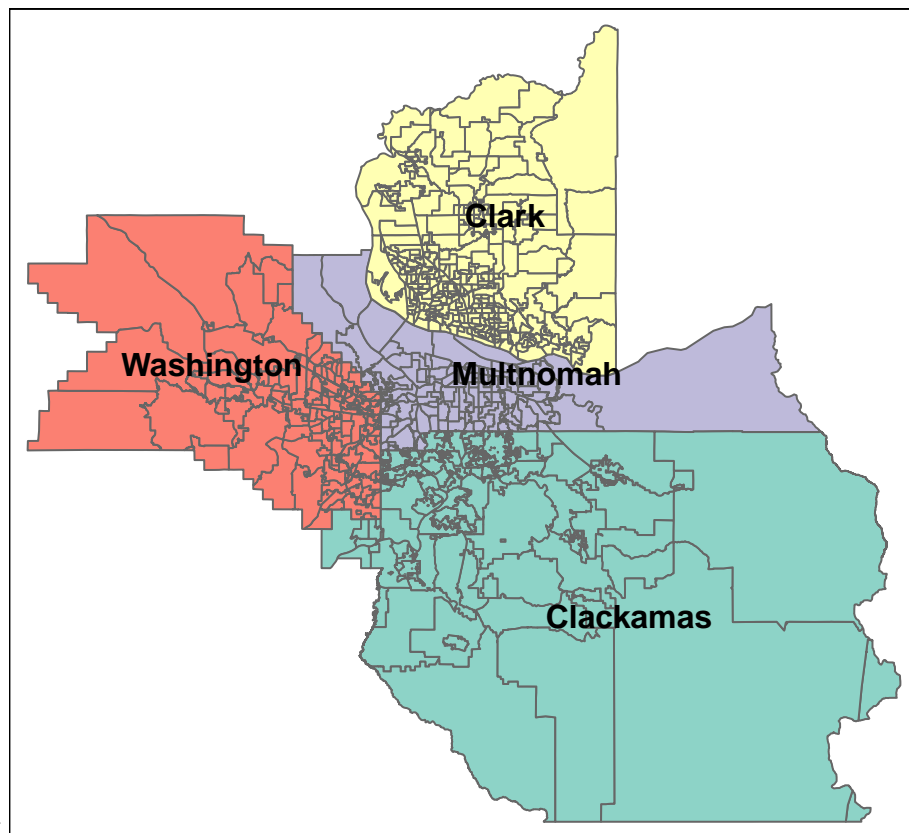
metro_portland_counties_sf <- oregon_counties_and_clark_sf %>%
  filter(county %in% c("Multnomah", "Washington", "Clackamas", "Clark"))

metro_portland_precinct_sf <- bind_rows(metro_portland_precinct_sf,
                                         clark_wa_precincts_sf)

metro_portland_tm <- tm_shape(metro_portland_precinct_sf) +
  tm_polygons(col="county", legend.show = FALSE, id = "county") +
  tm_shape(metro_portland_counties_sf) +
  tm_text("county", fontface = "bold")

metro_portland_tm

```



shapefiles-1.pdf

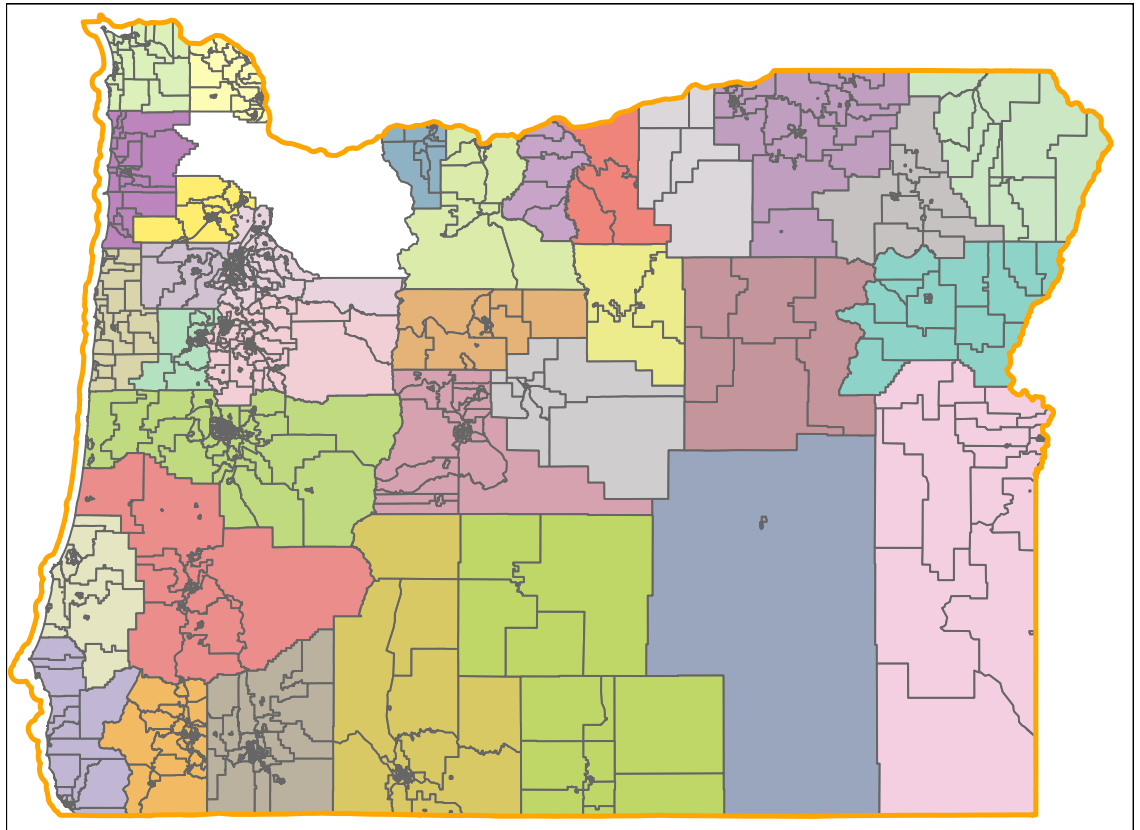
```

or_precinct_sf <- st_read("shapefiles/oregon_precincts/OregonPrecinctsNov2018.shp", quiet = TRUE) %>%
  filter(!County %in% c("Multnomah", "Washington", "Clackamas")) %>%
  st_transform("NAD83") %>%
  rename(county = County, precinct = Precinct)

or_precinct_tm <- tm_shape(or_precinct_sf) +
  tm_polygons(col="county", legend.show = FALSE, id = "county") +
  tm_shape(oregon_state) +
  tm_borders(col = "orange", lwd = 2.5)

or_precinct_tm

```



Vote data for presidential election (two major parties only)

This section processes the various files with precinct level election results. A lot of inconsistencies about candidate and party labeling are handled in these routines.

Oregon, excluding Metro Portland

Lane County vote totals, hand tabulated from county pdf

```

lane_votes <- read_csv("data/votes/lane.csv", col_types = cols(.default = "d", precinct = "c")) %>%
  mutate(county = "Lane") %>%
  rename(DEM = Biden, REP = Trump) %>%
  mutate(OTH = Total - DEM - REP) %>%
  select(county, precinct, DEM, REP, OTH)

```

Marion, Polk, and Yamhill County voter data. Note these use the OpenElections standard format. Portland

Metro counties generated further down.

This chunk was necessary because the consolidated state file for Open Elections had issues with these three counties. In theory, this function could be applied to all individual county files.

```
open_election_col_types = cols(.default = "c", votes = "d")

counties <- c("marion", "polk", "yamhill")
counties_csv_names <- str_c("data/votes/20201103__or__general__", counties, "__precinct.csv")

marion_polk_yamhill_votes <- map_df(counties_csv_names, open_election_single_county_votes)
```

Tillamook County, hand tabulated from the published county pdf

```
tillamook_votes <- read_csv("data/votes/tillamook_votes.csv", col_types = cols(.default = "c", DEM = "d",
  rename(precinct = precinct_vote_file) %>%
  select(county, precinct, DEM, REP, OTH)
```

Consolidate the five additional counties with or_precincts_votes

```
or_precincts_votes <- bind_rows(or_precincts_votes,
  tillamook_votes,
  lane_votes,
  marion_polk_yamhill_votes
)
```

read in a lookup table that matches vote file precinct names and shape file precinct names

```
lookup <- read_csv("data/votes_shapefile_lookup.csv",
  col_types = cols(.default = "c")) %>%
  mutate(county = str_to_title(tolower(county)))
```

combine lookup table and shapefile

Two columns have white space in the text and needs to be trimmed. Could replace regular expressions with simpler tidy version.

```
or_precinct_sf$precinct_vote_file <- gsub('\\s+', '', or_precinct_sf$precinct_vote_file)
or_precincts_votes$precinct <- gsub('\\s+', '', or_precincts_votes$precinct)
```

```
or_precinct_sf <- inner_join(or_precinct_sf, or_precincts_votes,
  by = c("county" = "county",
    "precinct_vote_file" = "precinct")) %>%
  select(county, precinct = precinct_vote_file, REP, DEM, OTH, geometry) %>%
  st_transform(crs = "NAD83")
```

```
or_precinct_df <- st_drop_geometry(or_precinct_sf)
```

Metro Portland Vote tables

Multnomah, Clackamas, and Washington, plus Clark County WA

Clackamas County is complicated because their reports combine several precincts. This chunk consolidates the affected precincts

The vote data was generated by hand before it became available in the Open Elections extract.

Voting tables and sf objects joined here

```
check <- anti_join(metro_portland_precinct_df,
  metro_portland_votes,
```

```

                                by=c("county", "precinct"))

metro_portland_precinct_sf <- inner_join(metro_portland_precinct_sf,
                                metro_portland_votes,
                                by=c("county", "precinct")) %>%
  select(county, precinct, DEM, REP, OTH)

or_precinct_sf <- bind_rows(or_precinct_sf,
                                metro_portland_precinct_sf
                                )

or_precinct_df <- st_drop_geometry(or_precinct_sf)

```

##The Leaflet Map

Some additional formatting variables added here.

```

or_precinct_sf <- or_precinct_sf %>%
  mutate(vpct = ifelse(DEM + REP > 0, DEM/(DEM + REP), .5)) %>%
  mutate(pwinner = ifelse(vpct >=.5,
                          "DEM",
                          "REP"),
          DPR = DEM + REP + OTH) %>%
  mutate(pct_lbl = ifelse(pwinner == "DEM",
                          str_c(pwinner, " ", format(vpct*100, digits = 3)),
                          str_c(pwinner, " ", format(100 - vpct*100, digits = 2)))) %>%
  mutate(hover = str_c(county, " ", precinct, ": ", pct_lbl, " ", DPR)) %>%
  select(county, precinct, vpct, DEM, REP, OTH, hover)

```

The final or_precinct_votes table has this structure:

```

or_precinct_df <- st_drop_geometry(or_precinct_sf)
or_precinct_df %>%
  filter(county=="Multnomah" & (precinct == "M3605" | precinct == "M3604"))

##      county precinct    vpct  DEM REP OTH      hover
## 1 Multnomah   M3604 0.7926829  130  34   6 Multnomah M3604: DEM  79.27 170
## 2 Multnomah   M3605 0.8751550 2117 302  48 Multnomah M3605: DEM  87.52 2467

```

#find some table functionality besides kbl that works

Draw the map

#Eliminate NaN's from the table listed here. If a precinct has zero votes it winds up with this result

```

or_precinct_df %>% filter(is.na(hover))

```

```

## [1] county  precinct vpct    DEM      REP      OTH      hover
## <0 rows> (or 0-length row.names)

```

```

tmap_mode("view")

```

tmap mode set to interactive viewing

```

tm_min = 0; tm_max = 1
min_max_bound <- function(x, min=tm_min, max=tm_max) {

```

```

    min(max(x,min),max)
}

rwb <- colorRampPalette(c("#ff0000", "white", "#0000fa"))(256)
midpoint <- .4
pctiles <- midpoint + c(-.5, -.4, -.2, -.05, 0, .05, .2, .4, .5)
pctiles <- c(tm_min, map_dbl(pctiles, min_max_bound), tm_max)

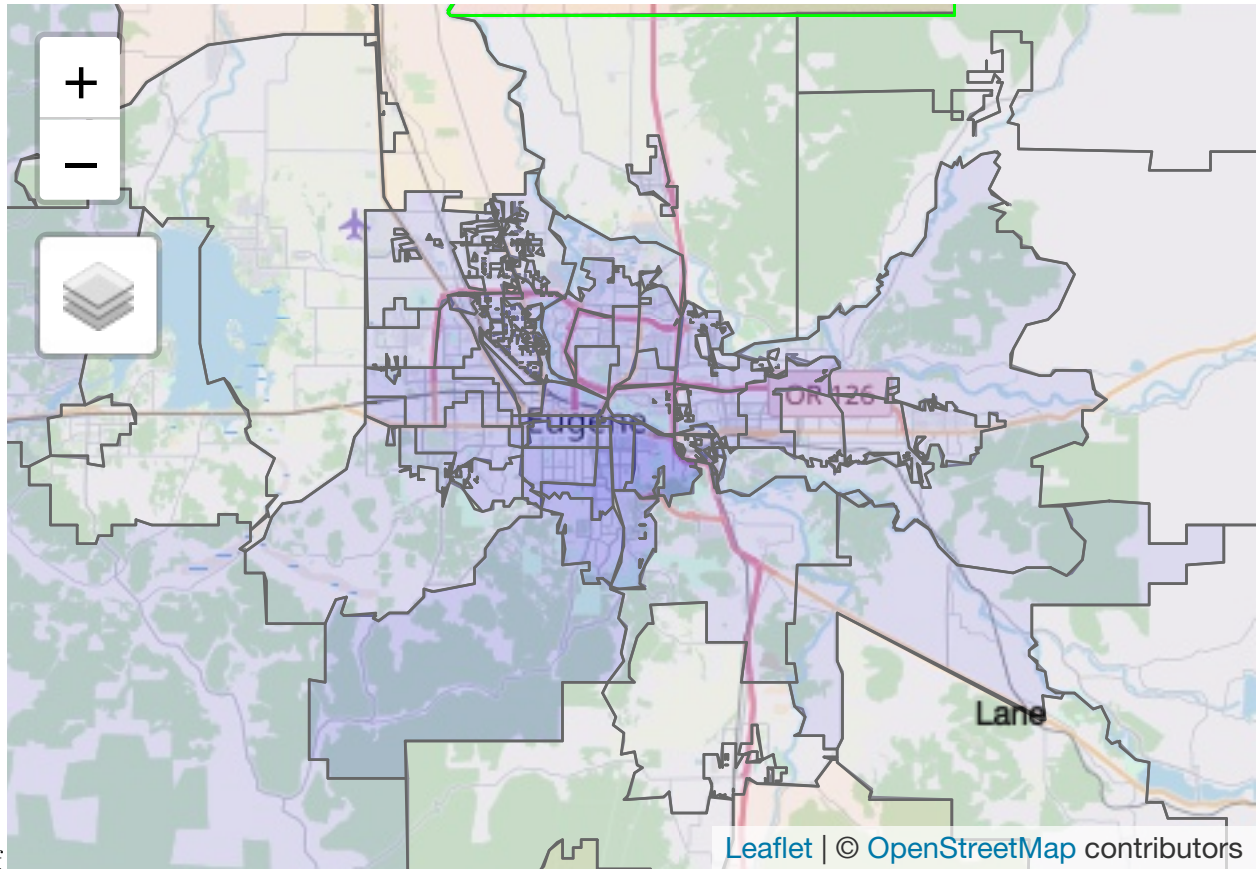
bb_or <- bb(or_precinct_sf)
bb_small <- bb(address_input, ext=10)

tmap_options(basemaps = c(Canvas = "OpenStreetMap"))

or_precincts_tm <-
  tm_shape(or_precinct_sf, bbox=bb_small) +
    tm_polygons(col = "vpct",
               n=length(10),
               id = "hover",
               style = "fixed",
               breaks = pctiles,
               alpha = 0.3,
               palette = rwb,
               legend.show = FALSE) +
  #tm_view(set.zoom.limits = c(6,18)) +
  tm_shape(oregon_counties_and_clark_sf, bbox=bb_small) +
    tm_borders(lwd = 1, col = "green") +
    tm_text("county", size = 1.0, fontface = "bold")

or_precincts_tm

```

the Map-1.pdf