Space

Session 12

PMAP 8921: Data Visualization with R
Andrew Young School of Policy Studies
Summer 2024

Plan for today

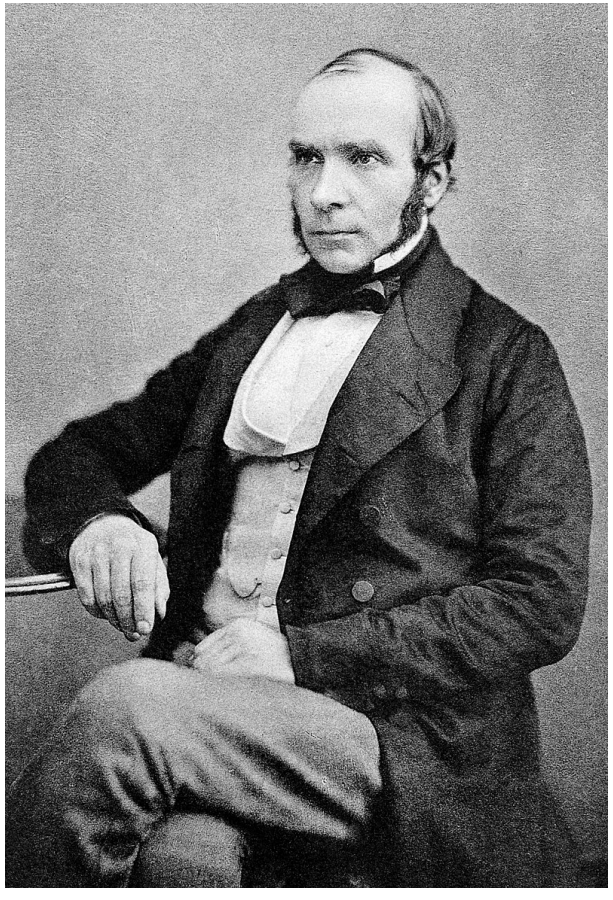
Maps and truth

Putting data on maps

GIS in R with {sf}

Maps and truth

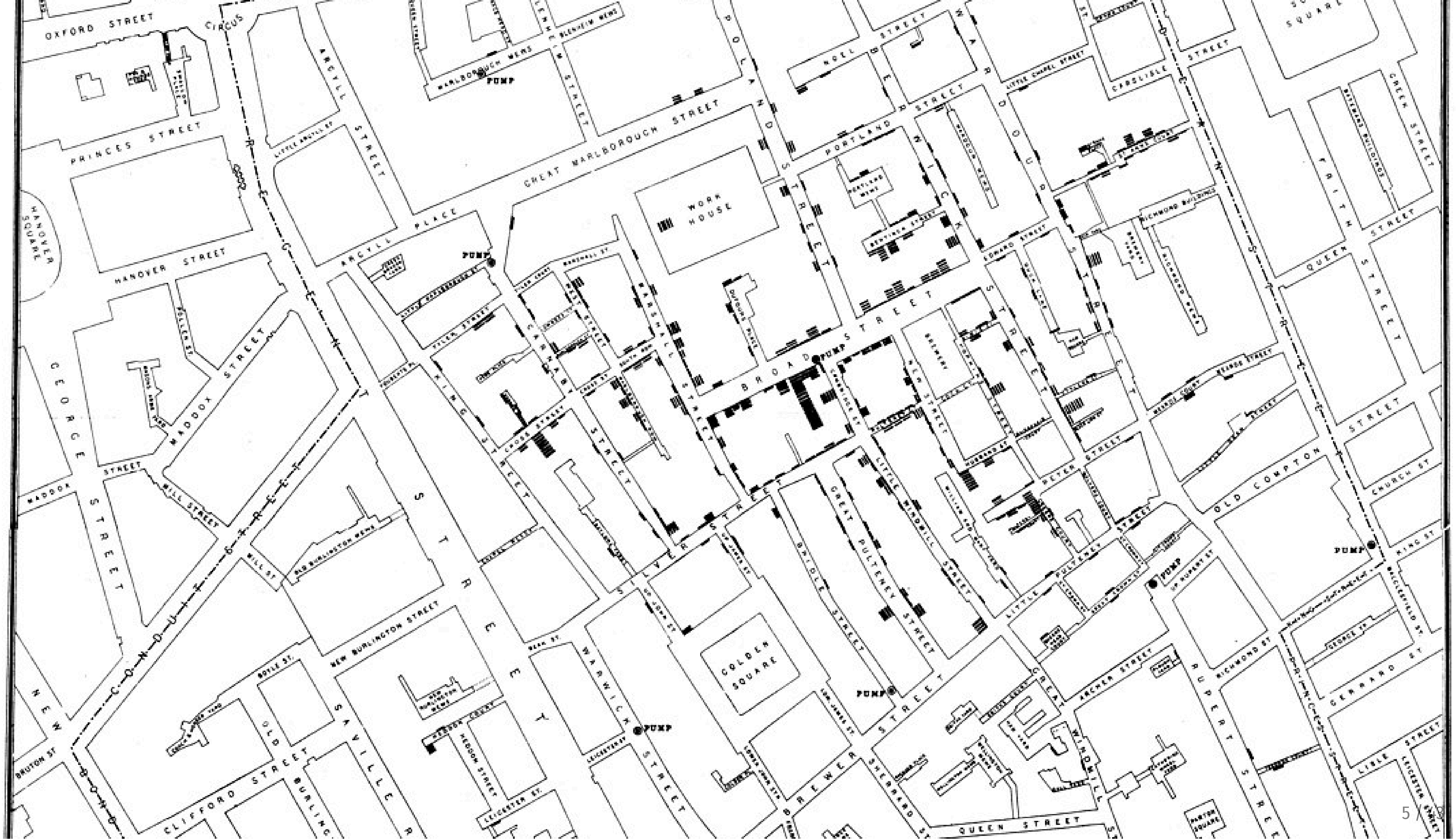
John Snow and 1854 cholera epidemic



This Jo(h)n Snow knows things

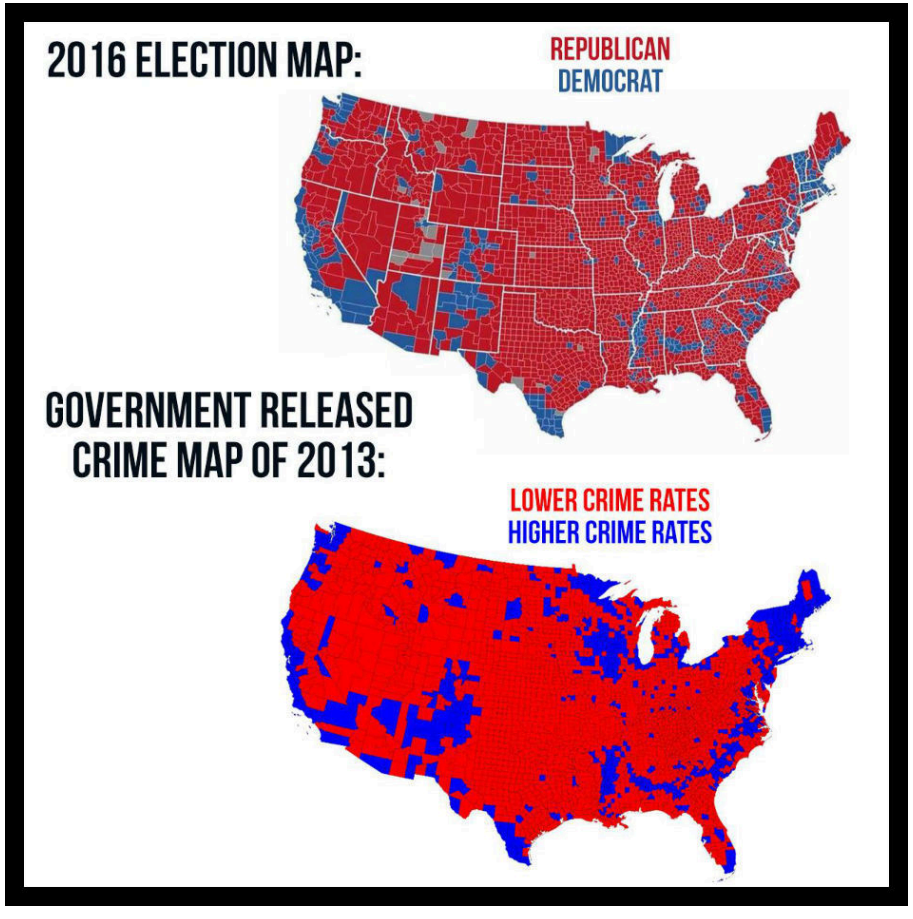
10% of the population of Soho died in a week (!!)

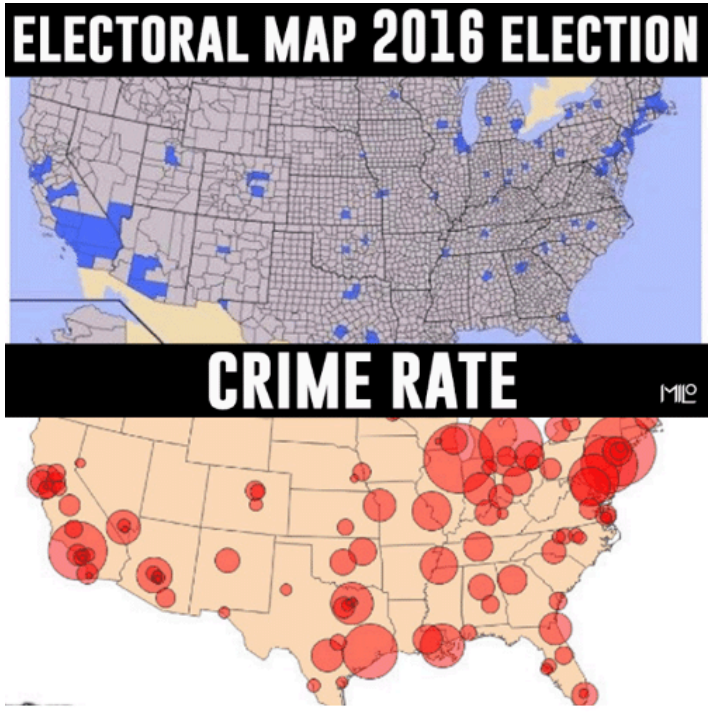
Miasma theory said it was because the air was bad


OXFORD STREET
CIRCUS
PRINCES STREET
HANOVER SQUARE
HANOVER STREET
GEORGE STREET
MADDOX STREET
MILL STREET
ARGYLL STREET
ARGYLL PLACE
MARLBOROUGH MEWS
PUMP
GREAT MARLBOROUGH STREET
POLAND STREET
WORK HOUSE
NOEL STREET
PORTLAND STREET
BERWICK STREET
WARDOUR STREET
DEAN STREET
FRITH STREET
CARLISLE STREET
LITTLE CHAPEL STREET
RICHMOND BUILDINGS
EDWARD STREET
QUEEN STREET
BROAD STREET
MARSHALL STREET
CARNABY STREET
KING STREET
CROSS STREET
SILVER STREET
NEW STREET
PETER STREET
HUBBARD ST
LITTLE WINDMILL STREET
GREAT PULTENEY STREET
BRIDLE STREET
GOLDEN SQUARE
LITTLE PULTENEY STREET
OLD COMPTON STREET
CHURCH STREET
KING ST
GERRARD STREET
LISLE STREET
RUPERT STREET
RICHMOND ST
ARCHER STREET
GREAT WINDMILL STREET
BREWER STREET
WARWICK STREET
NEW BURLINGTON STREET
OLD BURLINGTON MEWS
SAVILLE
OLD BURLINGTON
CLIFFORD STREET
BOYLE ST
BRUTON ST
NEW
STREET

The Broad Street pump

Outright lies





Democrats are as consistent in voting as they are in crime I guess...

Fake maps and junk maps

FAST COMPANY

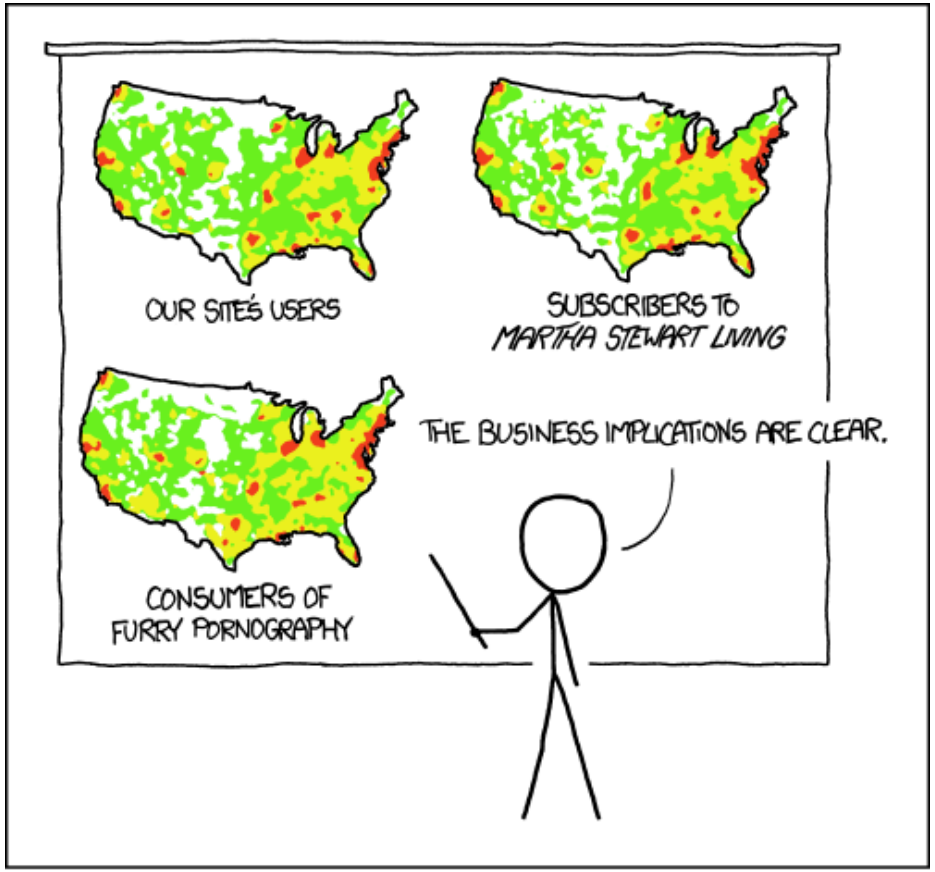
09-05-18

The next great fake news threat? Bot-designed maps

A new study reveals how maps go viral–and why they've become the perfect tool for misinformation.

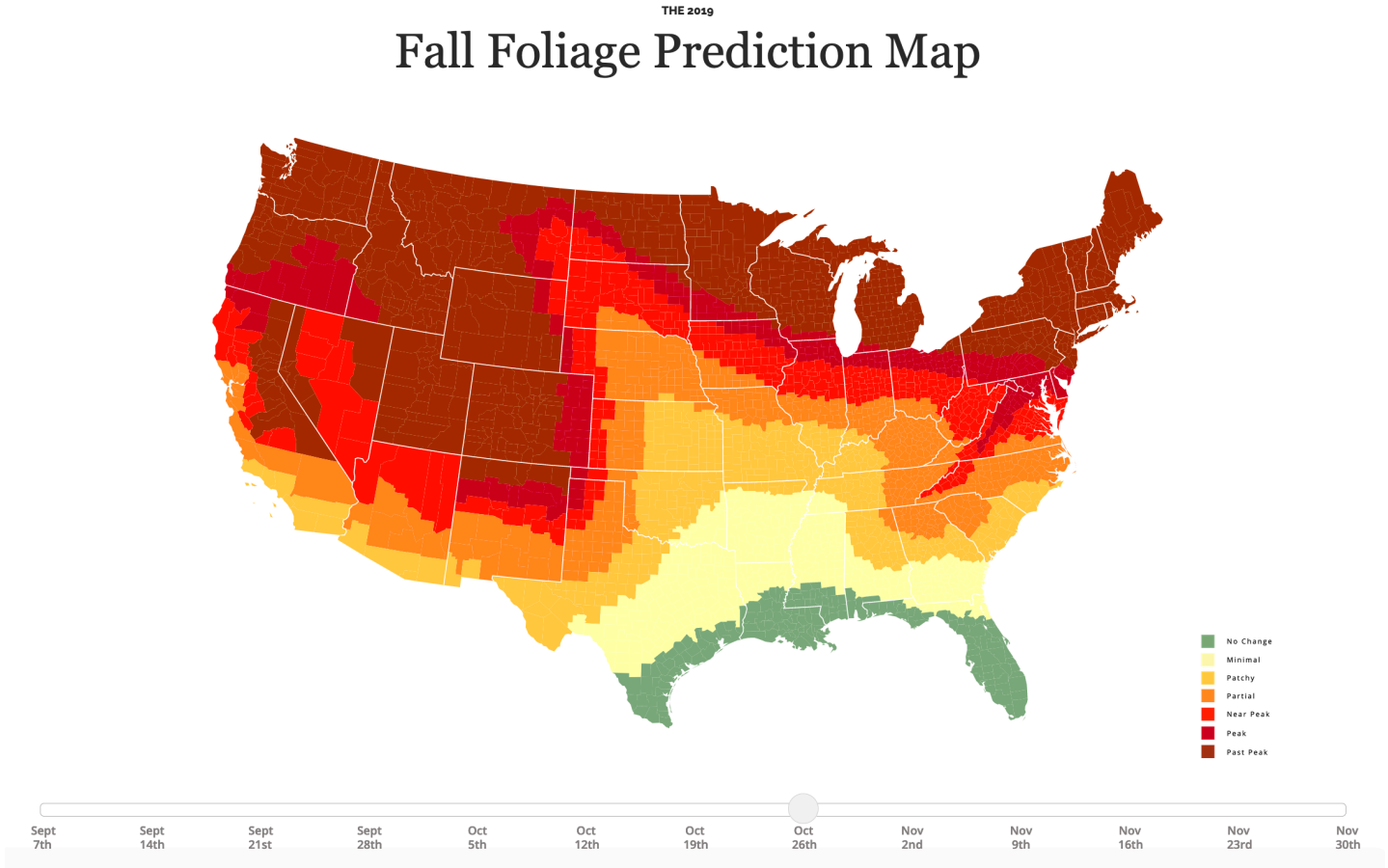
"The next great fake news threat? Bot-designed maps"

Points can be useless



PET PEEVE #208:
GEOGRAPHIC PROFILE MAPS WHICH ARE BASICALLY JUST POPULATION MAPS

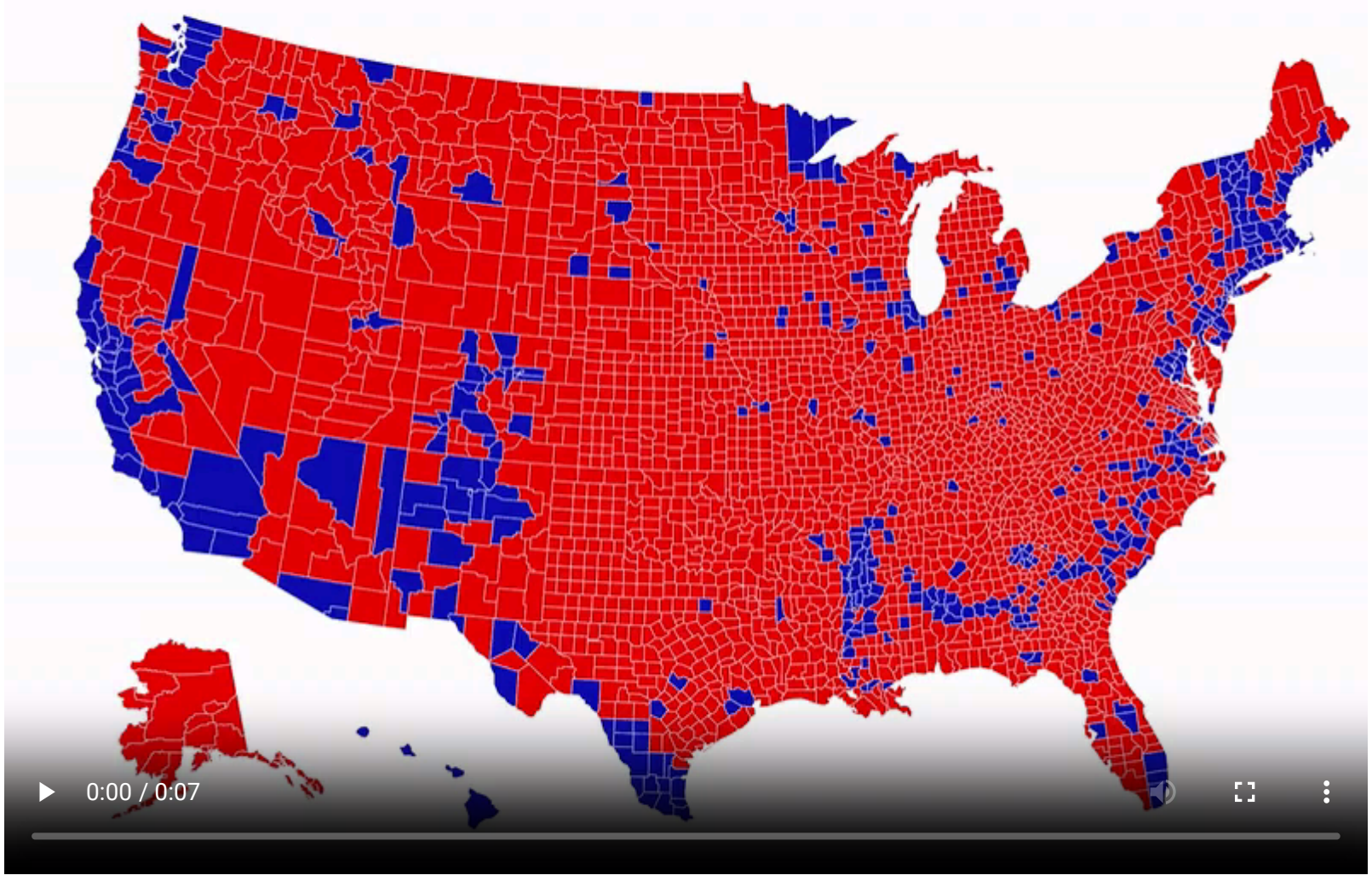
Choropleths can be great



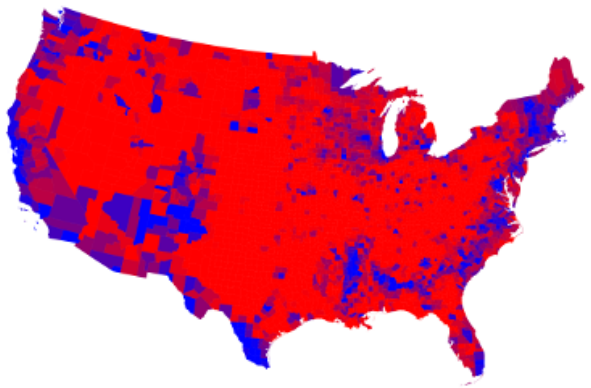
Smoky Mountains 2019 Fall Foliage Prediction Map

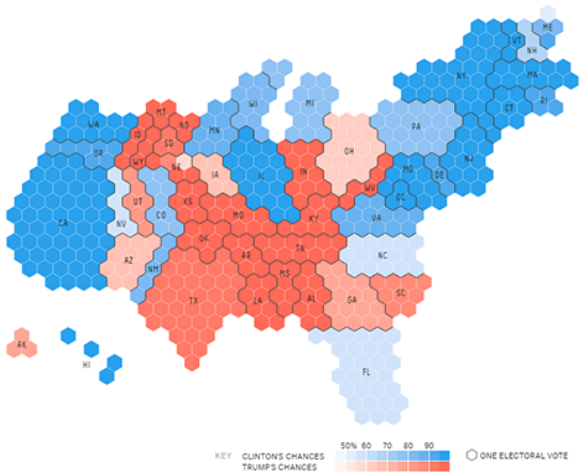
Choropleths can distort

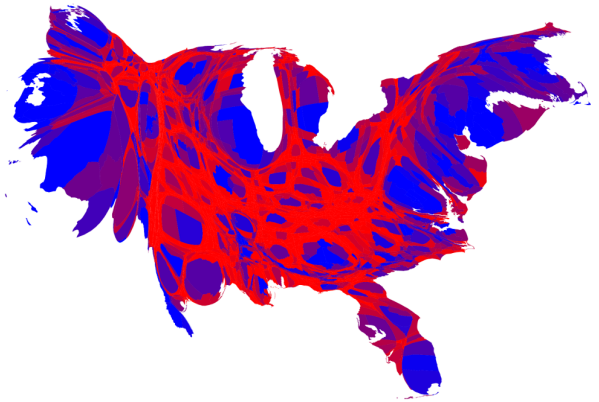
Land doesn't vote



Cartograms



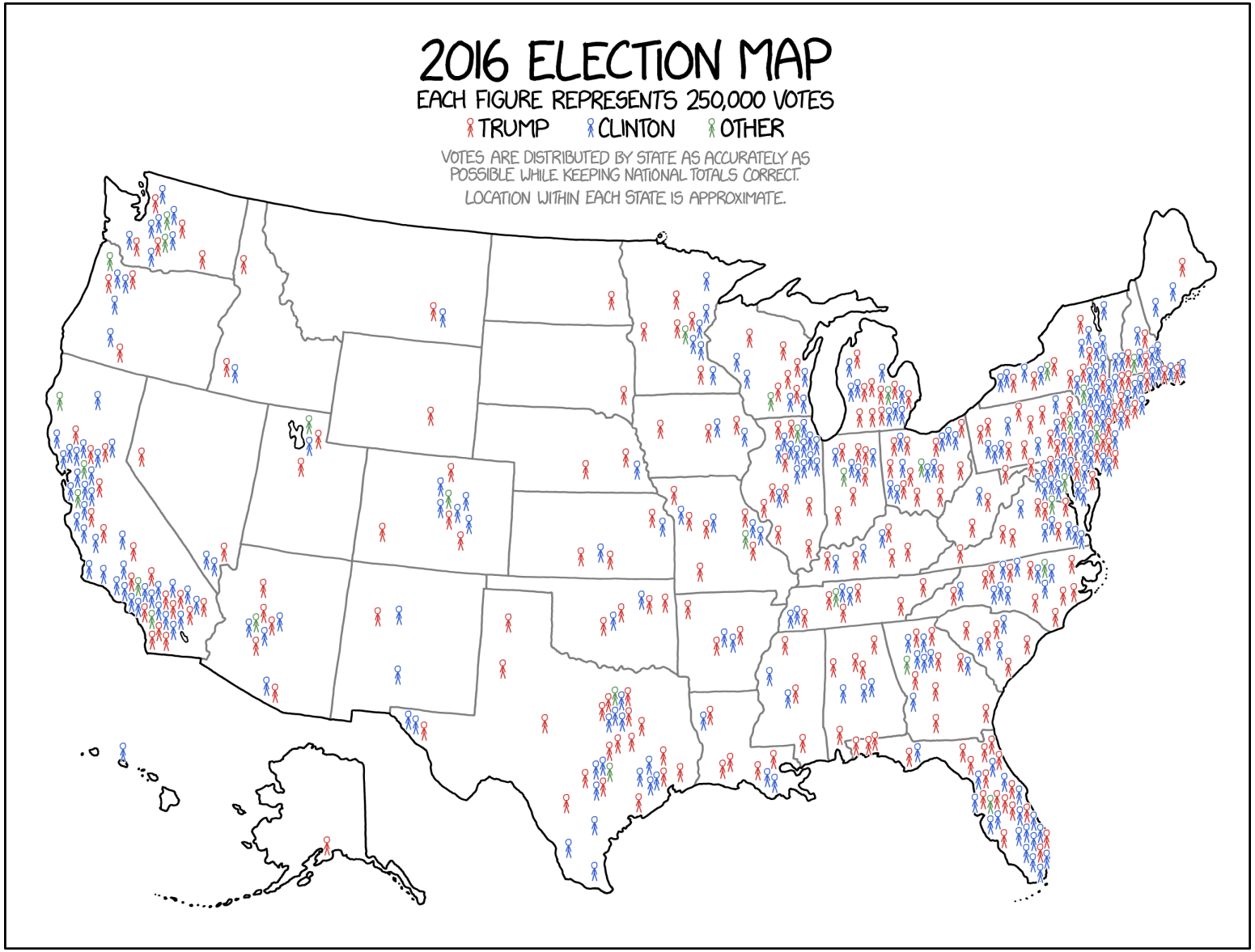

KEY
CLINTON'S CHANCES
TRUMP'S CHANCES
50% 60 70 80 90
ONE ELECTORAL VOTE



Electoral Doctrine

Mapping the 2015 UK General Election

Maps created by Benjamin Hennig
www.viewsoftheworld.net

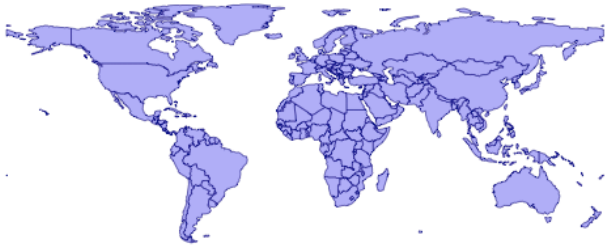

2016 ELECTION MAP
EACH FIGURE REPRESENTS 250,000 VOTES
TRUMP
CLINTON
OTHER
VOTES ARE DISTRIBUTED BY STATE AS ACCURATELY AS POSSIBLE WHILE KEEPING NATIONAL TOTALS CORRECT.
LOCATION WITHIN EACH STATE IS APPROXIMATE.

Projections

Animated world projections

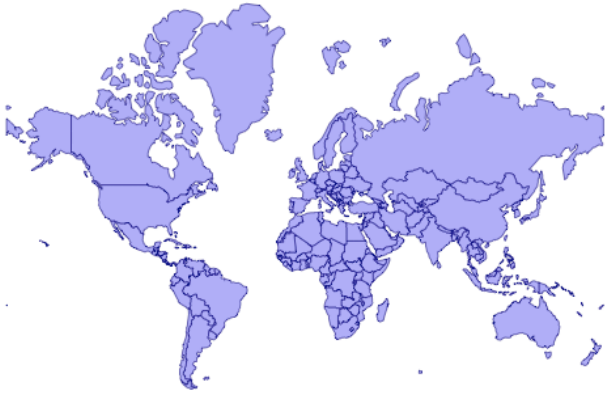
World projections

Longitude-latitude



```
crs = "+proj=longlat +ellps=WGS84"
```

Mercator



```
crs = "+proj=merc"
```

Gall-Peters

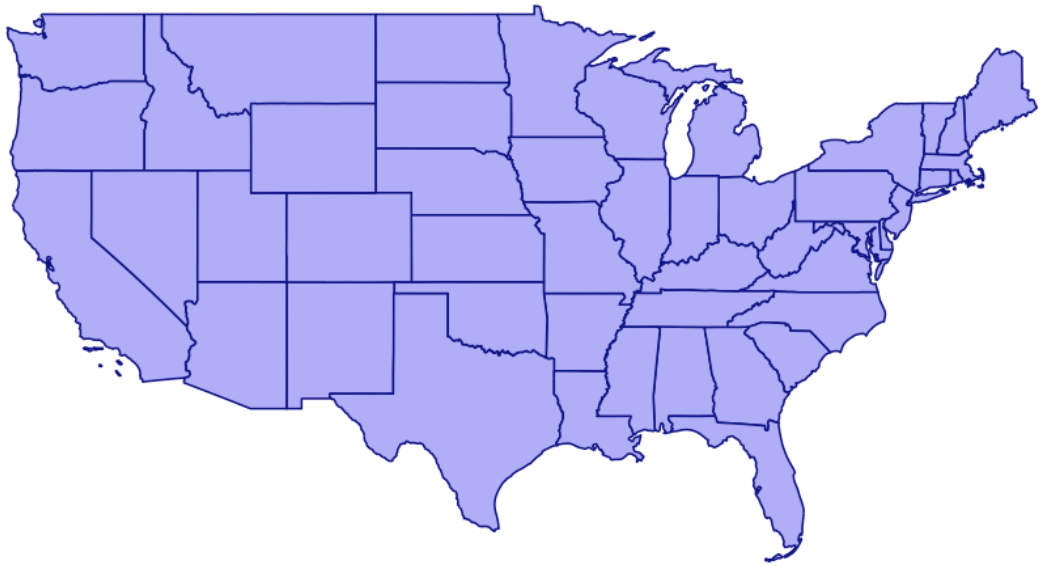
```
crs = "ESRI:54002"
```

Robinson

```
crs = "+proj=robin"
```

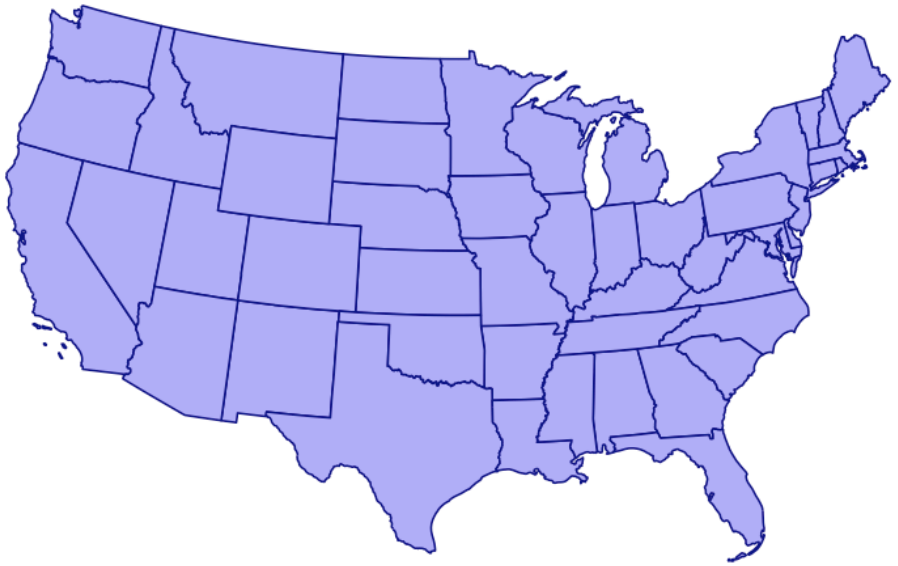
US projections

NAD83



`crs = "EPSG:4269"`

Albers



`crs = "ESRI:102003"`

Finding projection codes

spatialreference.org

epsg.io

proj.org

Most common ones listed on the course website example page

This is an excellent overview of how this all works

And this is a really really helpful overview of all these moving parts

Which projection is best?

None of them

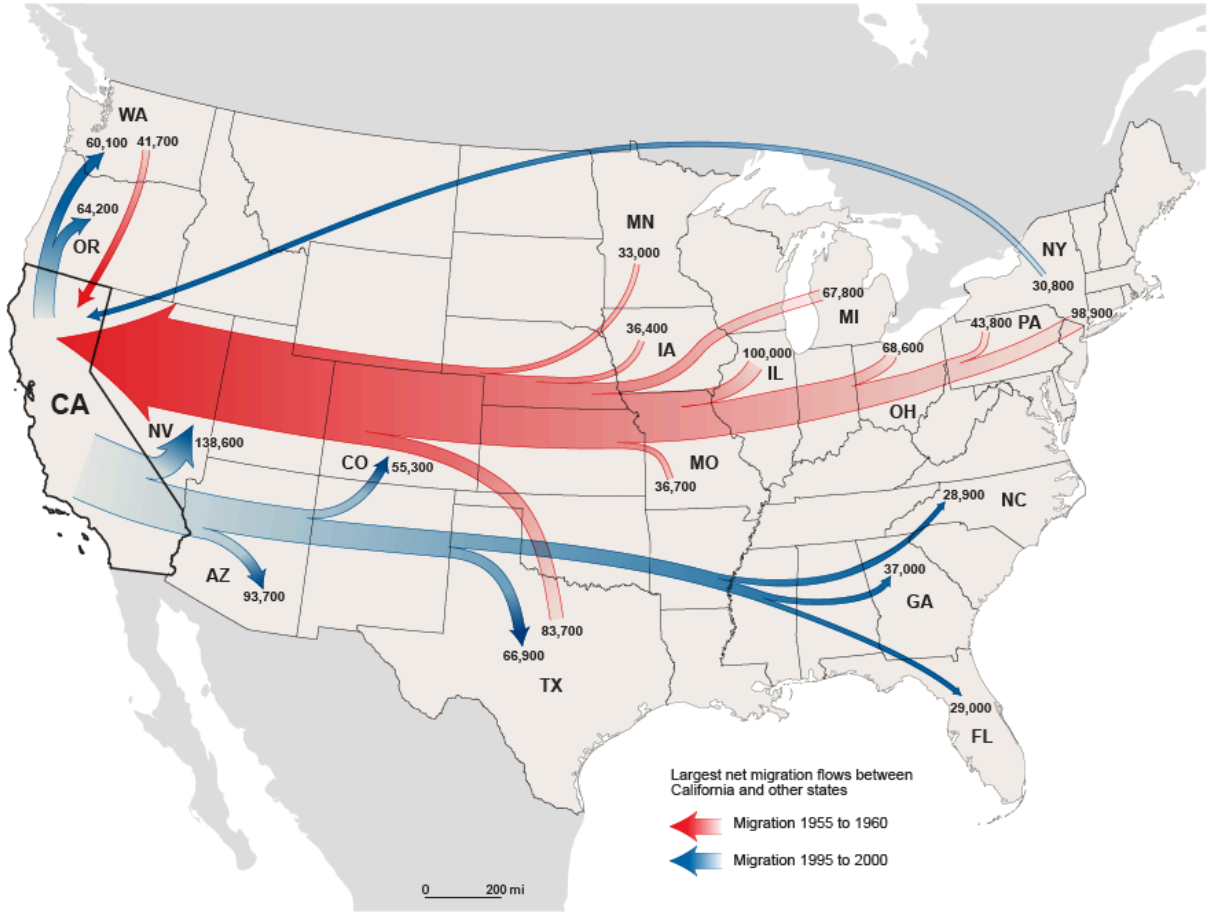
There are no good or bad projections

There are appropriate and inappropriate projections

(but also ew mercator)

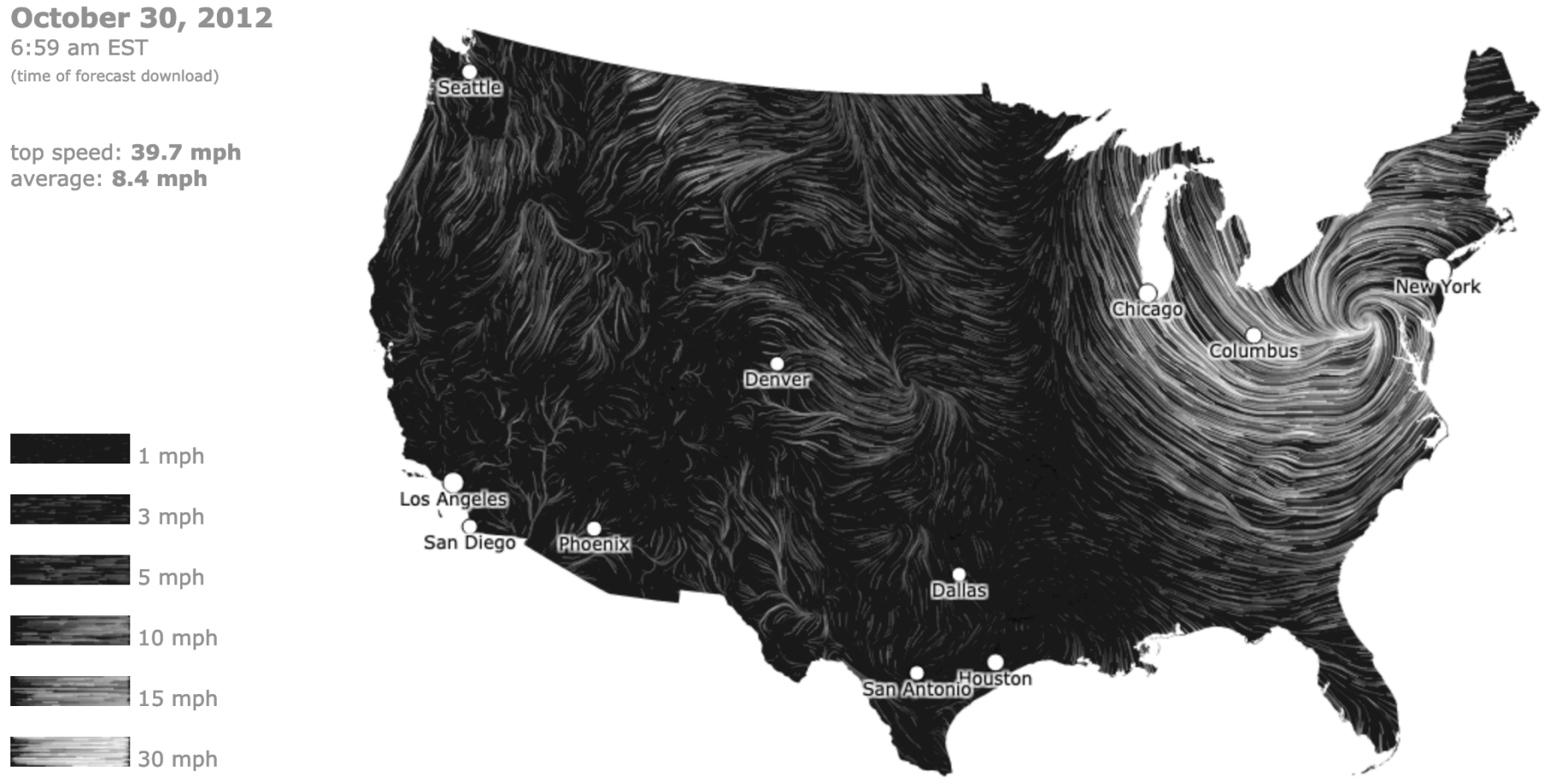
Putting data on maps

Maps with lines



US Census Bureau: Net migration between California and other states

Maps with lines

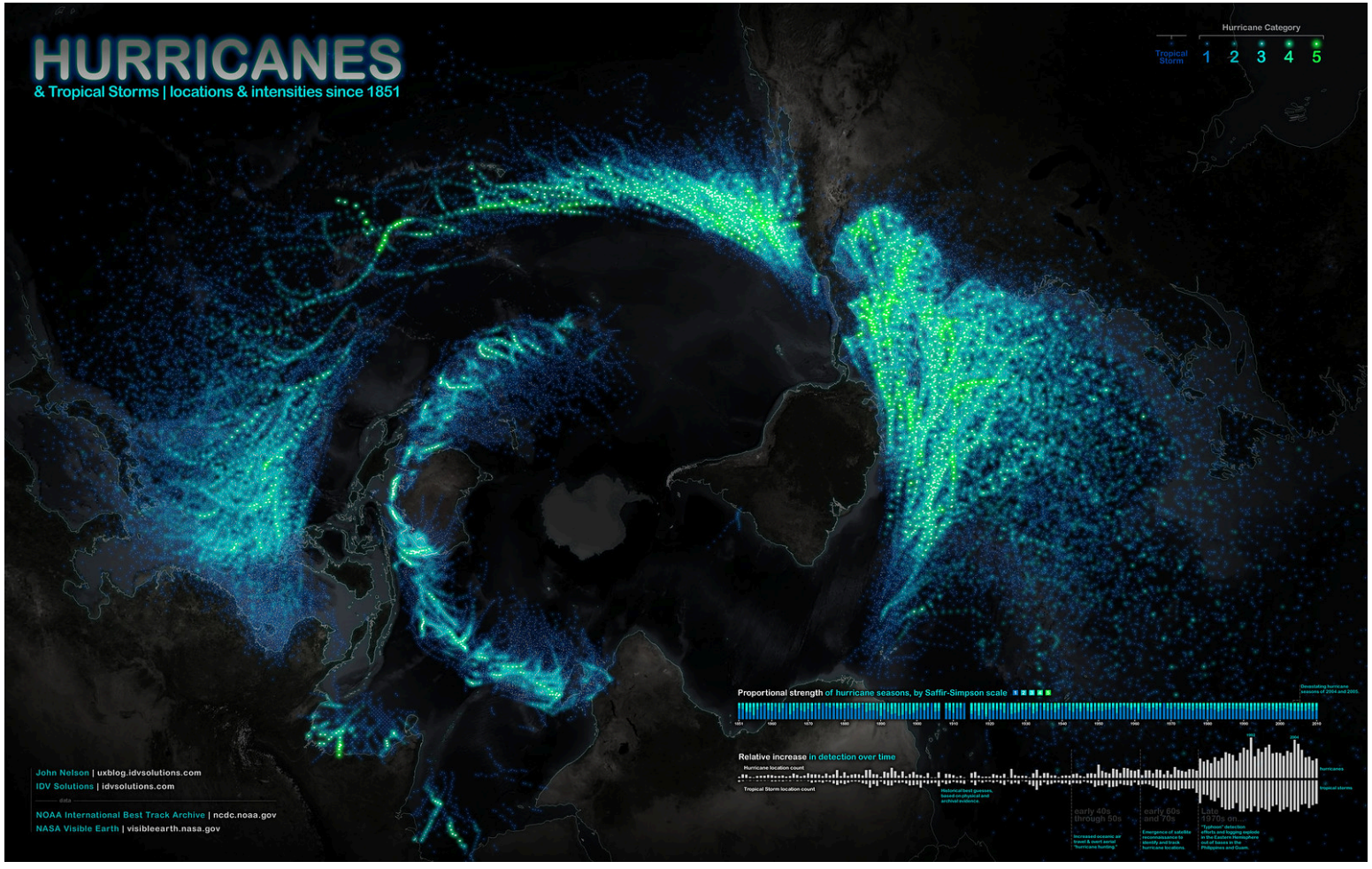


hint.fm Live Wind Map

American Rivers, Streams, and Waterways

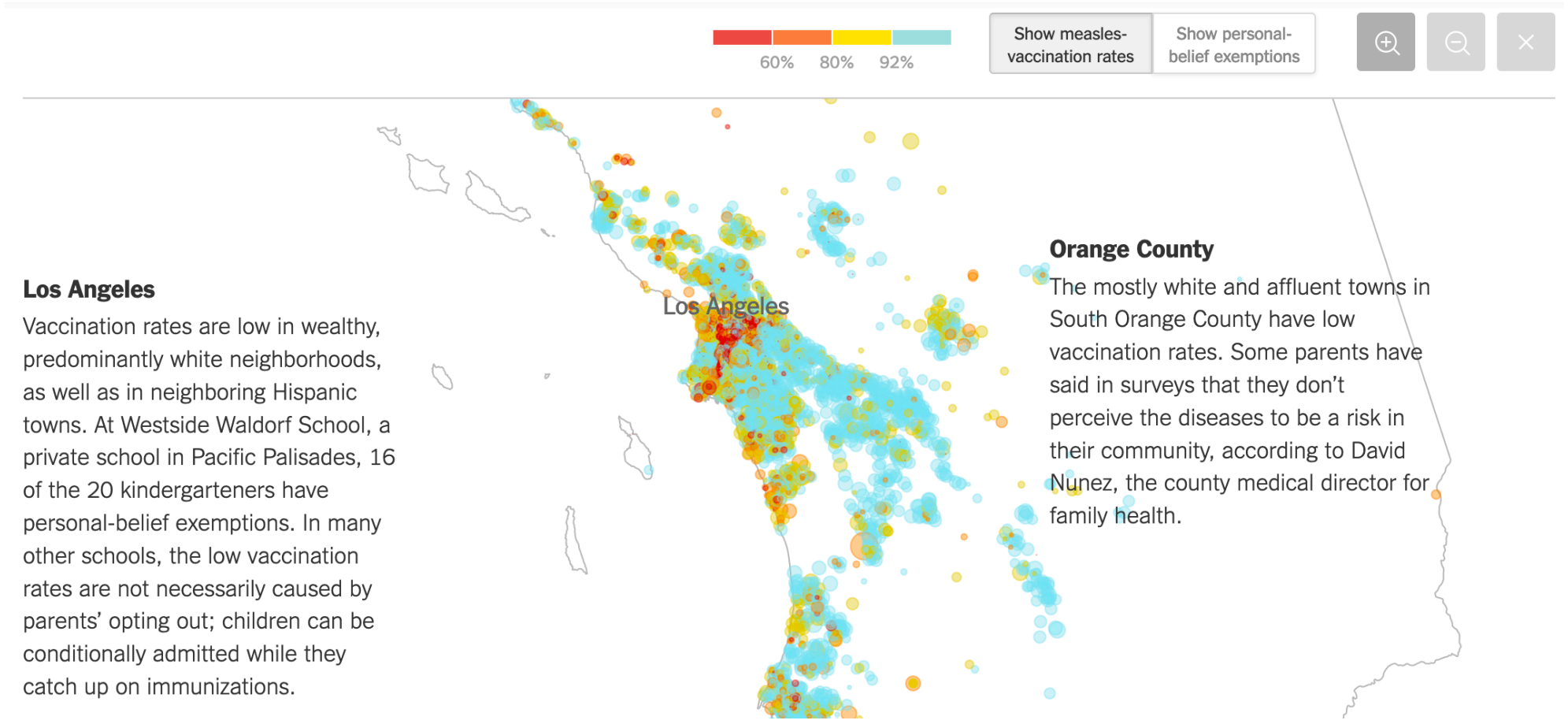
Kieran Healy

Maps with points



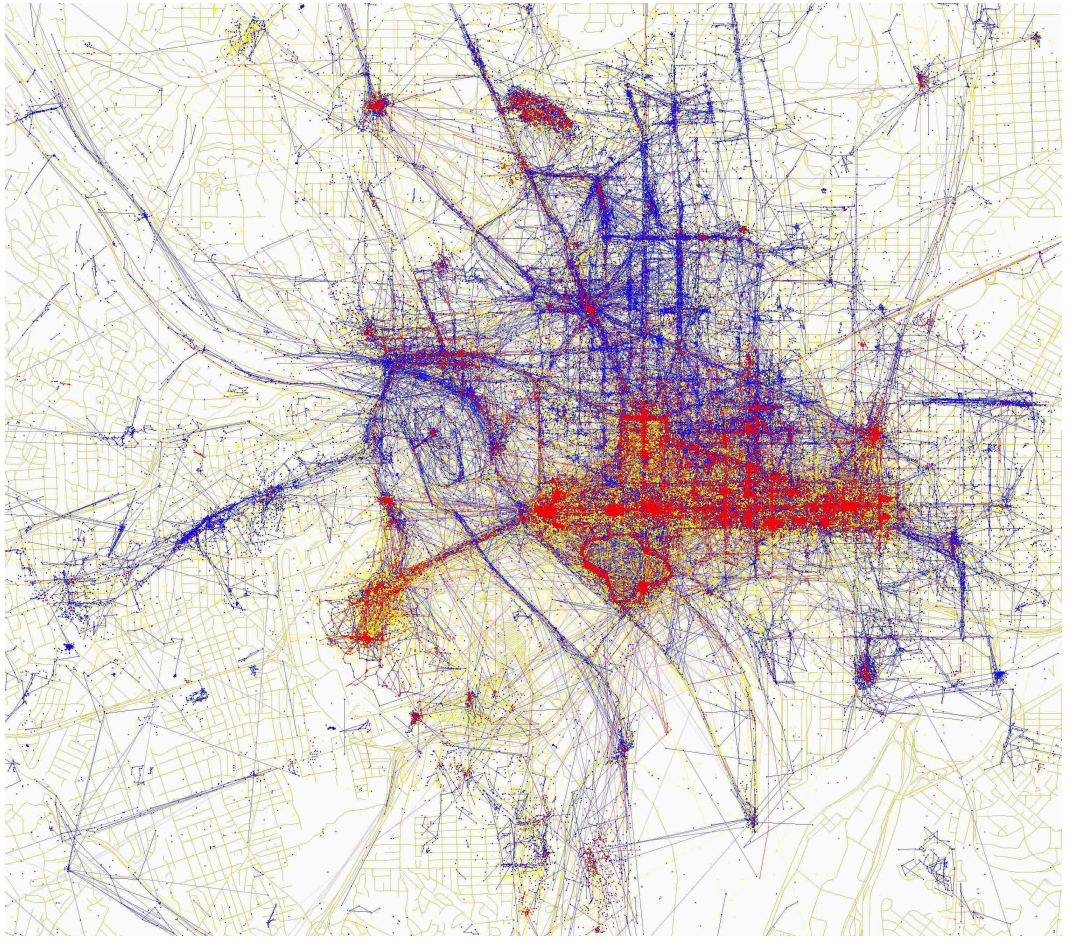
Every hurricane since 1851, by IDV solutions

Maps with points



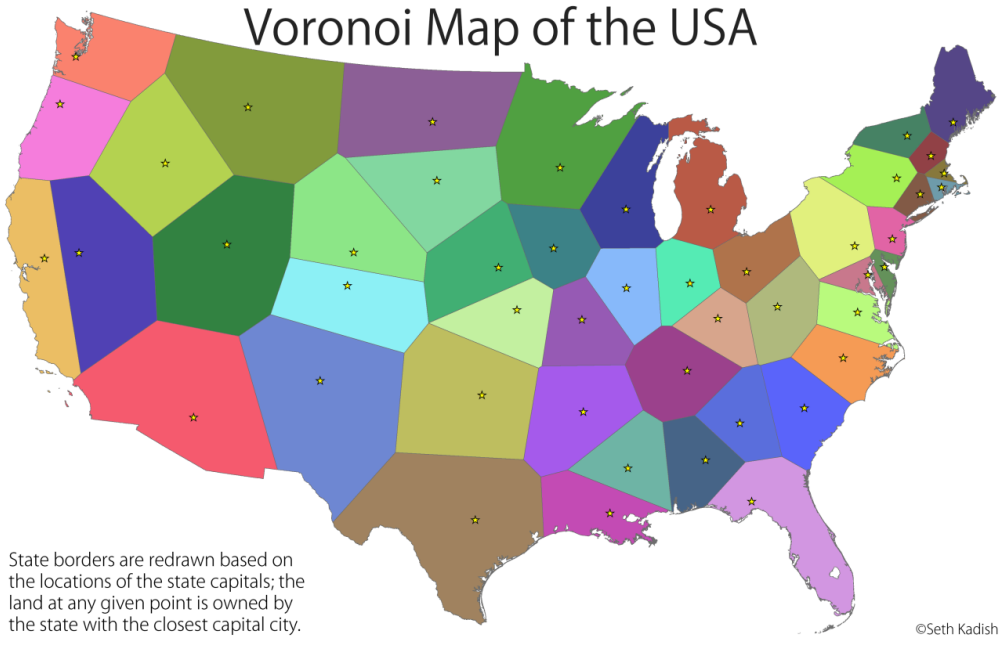
The New York Times, "Vaccination Rates for Every Kindergarten in California

Maps with points

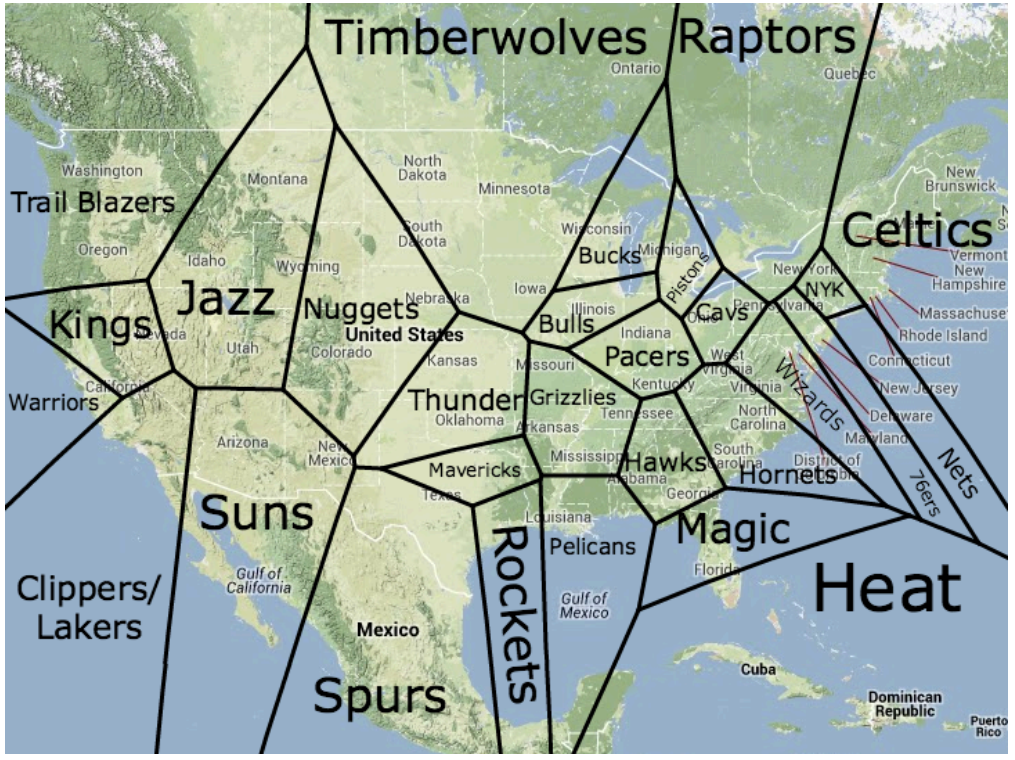


Locals vs. tourists in DC (blue = locals; red = tourists; yellow = unknown)

Voronoi maps

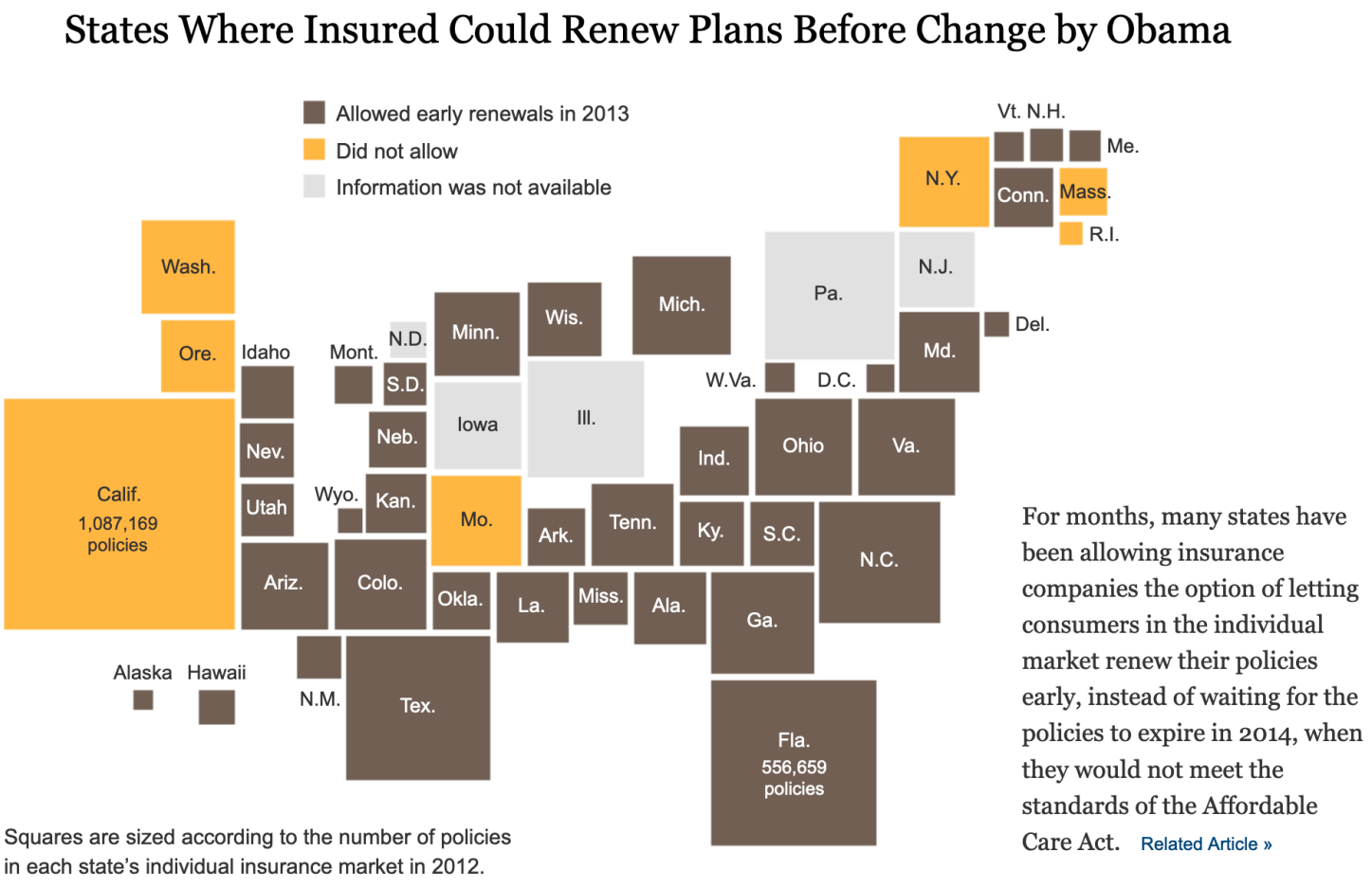


Voroni state boundaries, by Seth Kadish

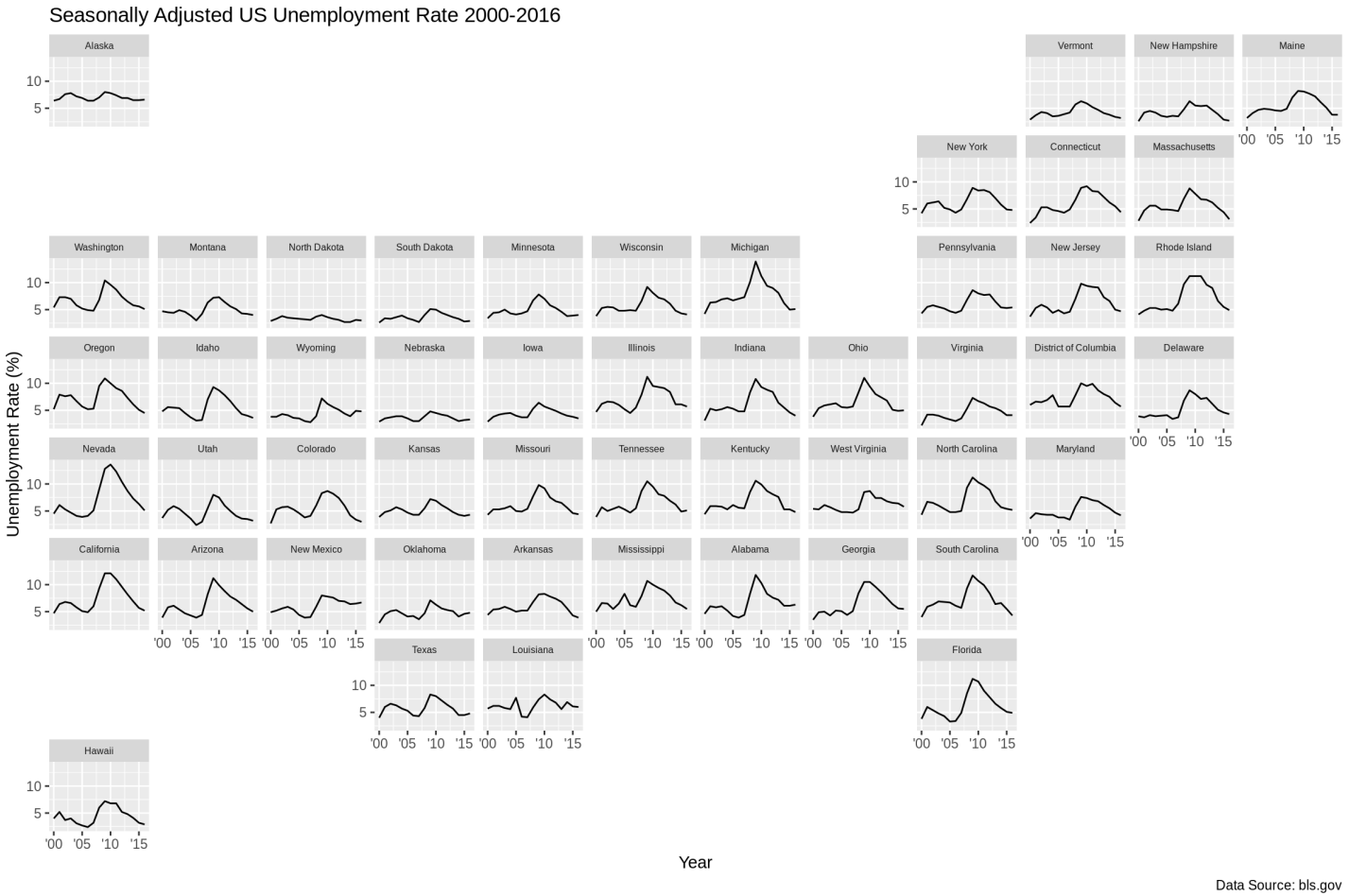


Closest NBA teams

Maps with shapes



Small multiples that look like maps



`facet_geo()` in the **geofacet** package

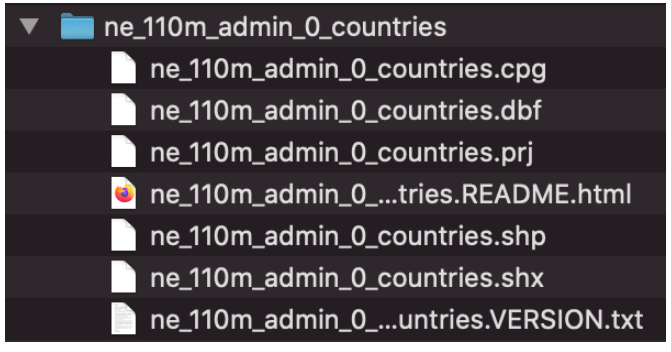
GIS in R with {sf}

Shapefiles

Geographic information is shared as shapefiles

These are *not* like regular single CSV files!

Shapefiles come as zipped files with a bunch of different files inside



Structure of a shapefile

```r
library(sf)
world_shapes <- read_sf("data/ne_110m_admin_0_countries/ne_110m_admin_0_countries.shp")
```

```
## Simple feature collection with 7 features and 3 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -180 ymin: -18 xmax: 180 ymax: 83
## Geodetic CRS:  WGS 84
## # A tibble: 7 × 4
##   TYPE              GEOUNIT                  ISO_A3                              geometry
##   <chr>             <chr>                    <chr>                     <MULTIPOLYGON [°]>
## 1 Sovereign country Fiji                     FJI    (((180 -16, 180 -17, 179 -17, 179 -17…
## 2 Sovereign country Tanzania                 TZA    (((34 -0.95, 34 -1.1, 38 -3.1, 38 -3.…
## 3 Indeterminate     Western Sahara           ESH    (((-8.7 28, -8.7 28, -8.7 27, -8.7 26…
## 4 Sovereign country Canada                   CAN    (((-123 49, -123 49, -125 50, -126 50…
## 5 Country           United States of America USA    (((-123 49, -120 49, -117 49, -116 49…
## 6 Sovereign country Kazakhstan               KAZ    (((87 49, 87 49, 86 48, 86 47, 85 47,…
## 7 Sovereign country Uzbekistan               UZB    (((56 41, 56 45, 59 46, 59 46, 60 45,…
```

Where to find shapefiles

Natural Earth for international maps

US Census Bureau for US maps

For anything else...

Scales

1:10m = 1:10,000,000

1 cm = 100 km

1:50m = 1:50,000,000

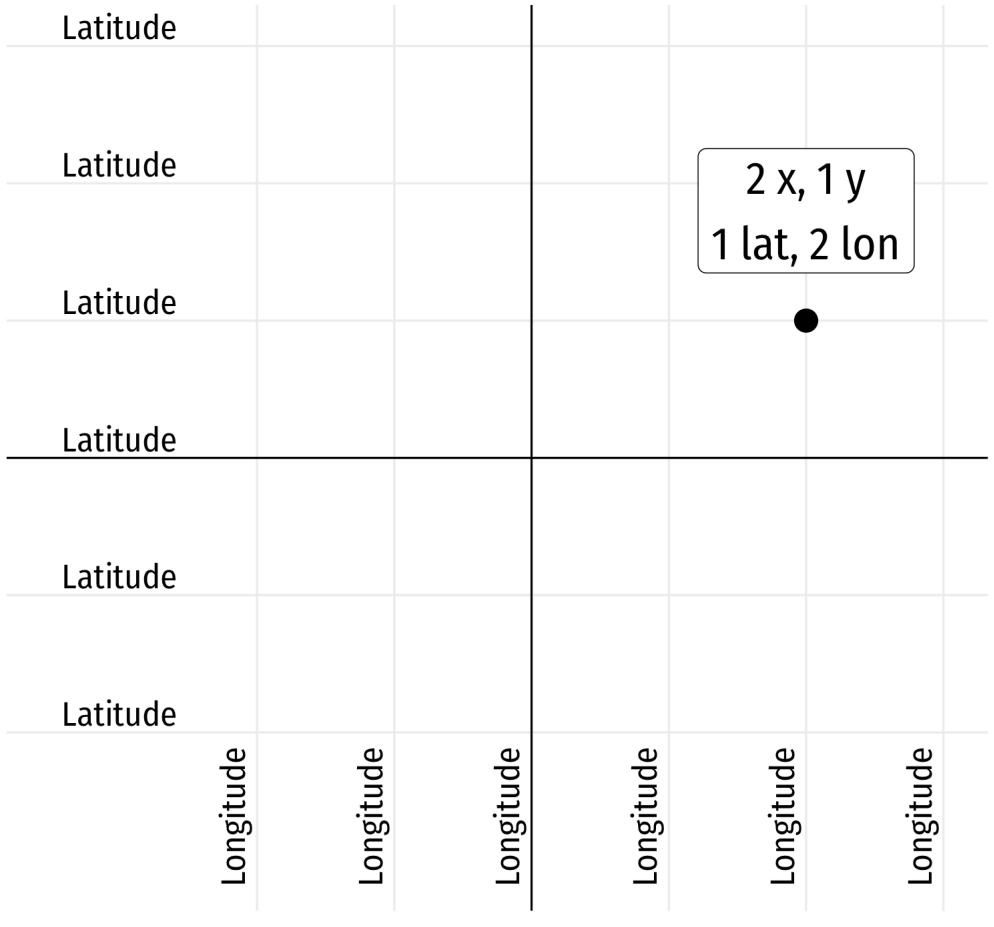
1cm = 500 km

1:110m = 1:110,000,000

1 cm = 1,100 km

Using too high of a resolution makes your maps slow and huge

Latitude and longitude

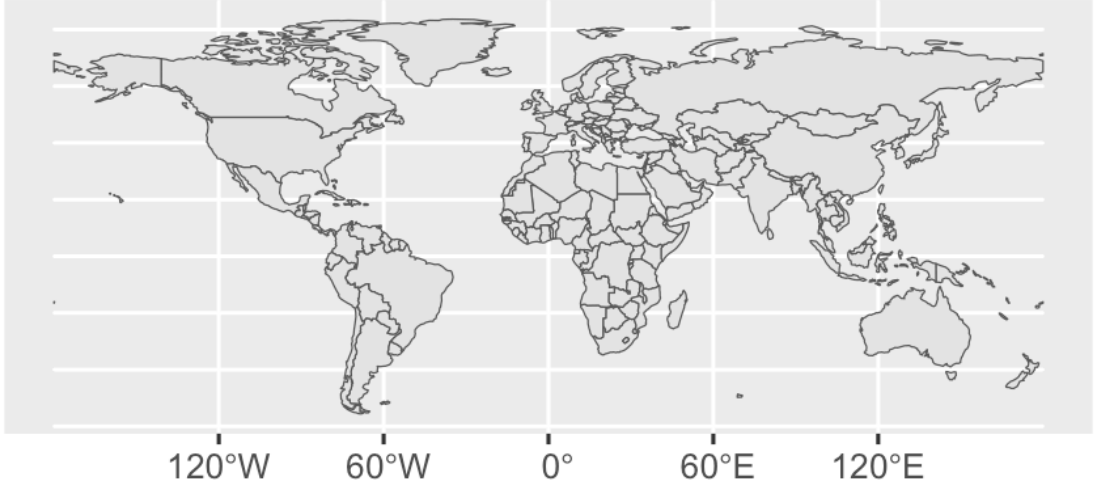


via @sarahbellmaps

The magic `geometry` column

As long as you have a magic `geometry` column, all you need to do to plot maps is `geom_sf()`

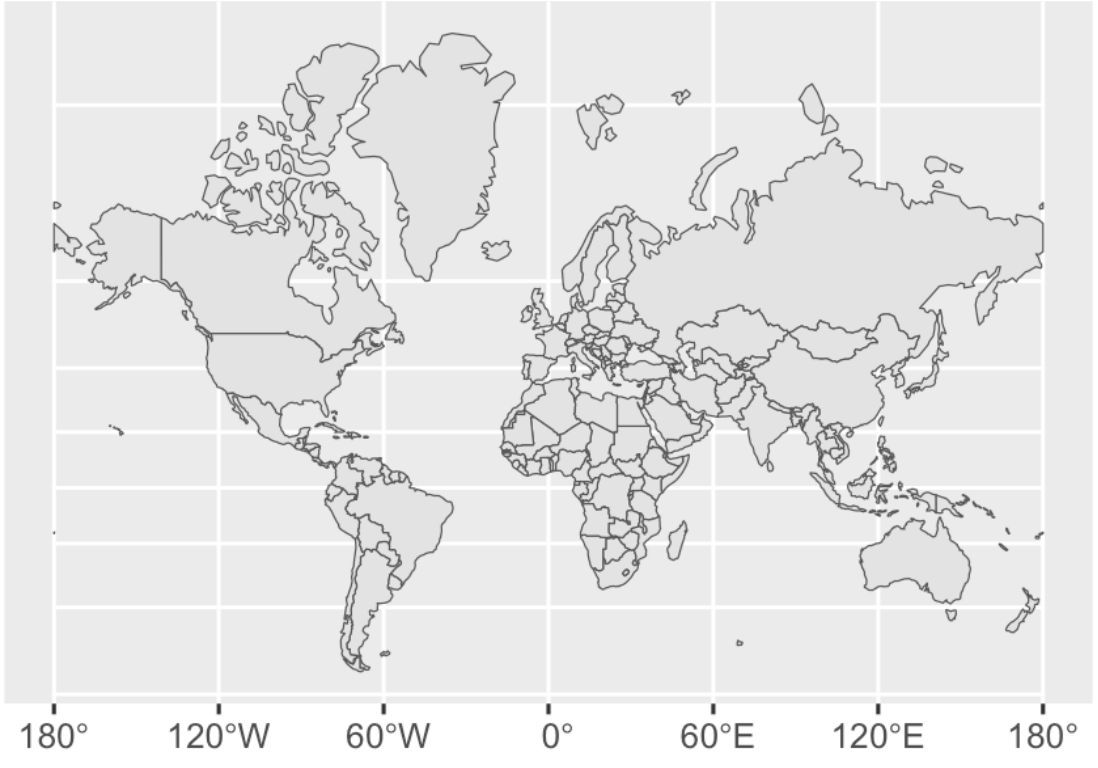
```
ggplot() +
  geom_sf(data = world_shapes)
```



The magic `geometry` column

Use `coord_sf()` to change projections

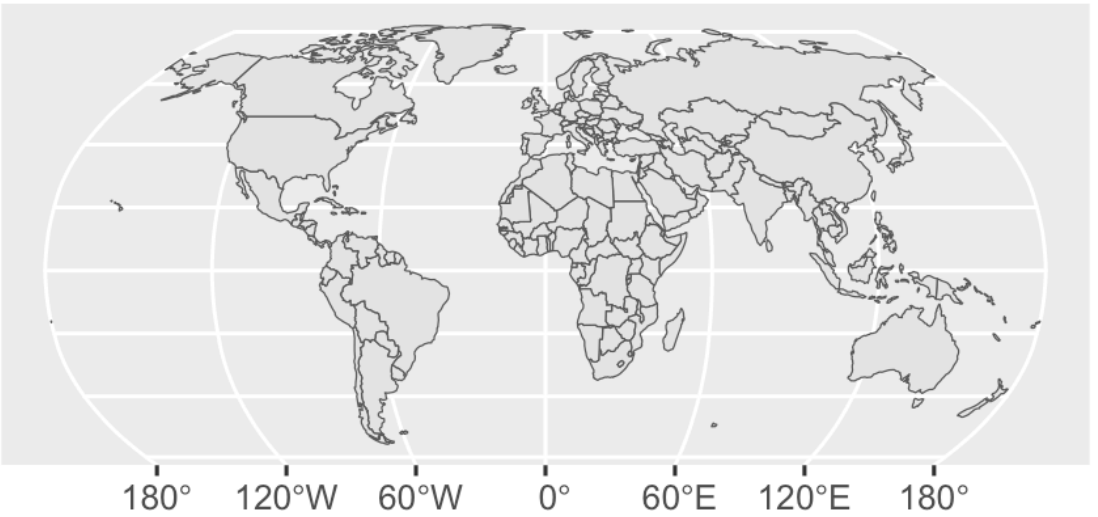
```
ggplot() +
  geom_sf(data = world_shapes) +
  coord_sf(crs = "+proj=merc")
```



The magic `geometry` column

Use `coord_sf()` to change projections

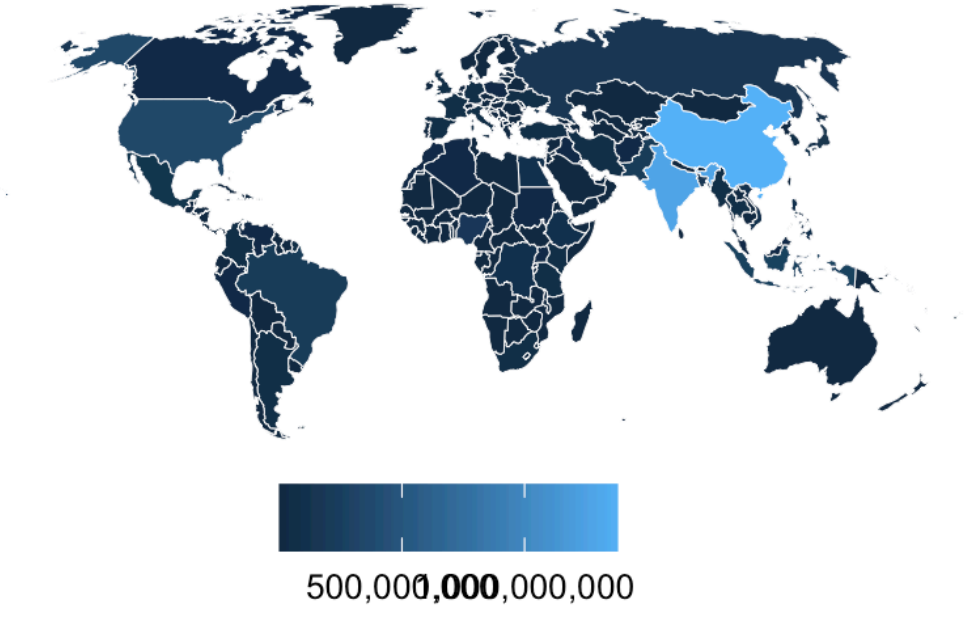
```
ggplot() +
  geom_sf(data = world_shapes) +
  coord_sf(crs = "+proj=robin")
```



Use aesthetics like normal

All regular ggplot layers and aesthetics work

```
ggplot() +
  geom_sf(data = world_shapes,
          aes(fill = POP_EST),
          color = "white", linewidth = 0.15) +
  coord_sf(crs = "+proj=robin") +
  scale_fill_gradient(labels = scales::label_
  labs(fill = NULL) +
  theme_void() +
  theme(legend.position = "bottom")
```



No `geometry` column?

Make your own with `st_as_sf()`

```
other_data
```

```
## # A tibble: 2 × 3
##   city            long   lat
##   <chr>          <dbl> <dbl>
## 1 Atlanta        -84.4  33.8
## 2 Washington, DC -77.1  38.9
```

```
other_data |>
  st_as_sf(coords = c("long", "lat"),
           crs = st_crs("EPSG:4326"))
```

```
## Simple feature collection with 2 features and 1 field
## Geometry type: POINT
## Dimension:     XY
## Bounding box:  xmin: -84 ymin: 34 xmax: -77 ymax: 39
## Geodetic CRS:  WGS 84
## # A tibble: 2 × 2
##   city                geometry
## * <chr>             <POINT [°]>
## 1 Atlanta             (-84 34)
## 2 Washington, DC      (-77 39)
```

{sf} is for all GIS stuff

Draw maps

Calculate distances between points

Count observations in a given area

Anything else related to geography!

See here or here for full textbooks

`geom_sf()` is today's standard

You'll sometimes find older tutorials and StackOverflow answers about using `geom_map()` or {ggmap} or other things

Those still work, but they don't use the same magical {sf} system with easy-to-convert projections and other GIS stuff

Stick with {sf} and `geom_sf()` and your life will be easy