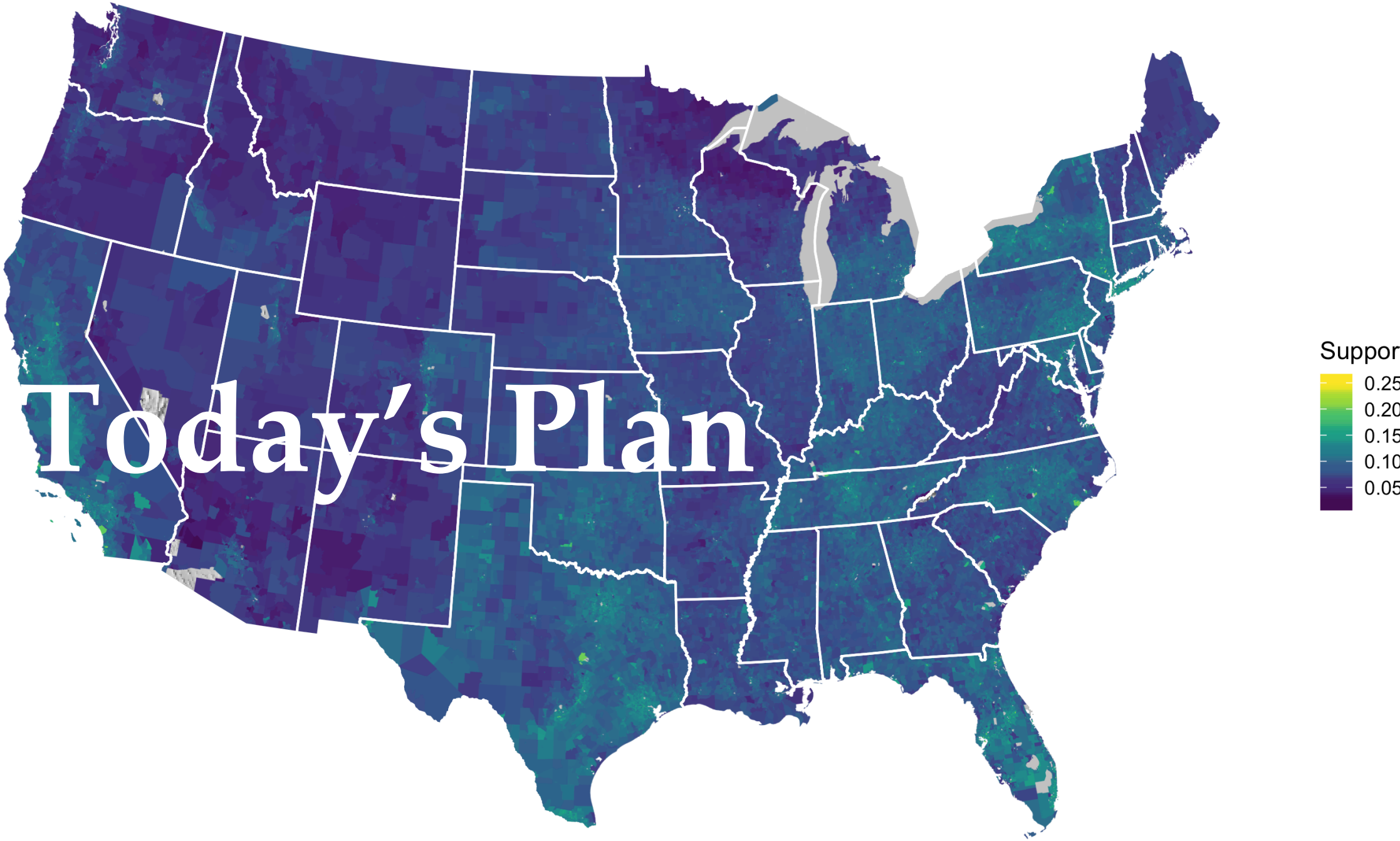
Building Spatial Databases with Attributes

HES 505 Fall 2024: Session 14

Carolyn Koehn

Today's Plan



Objectives

- By the end of today, you should be able to:
  - Define *spatial analysis*
  - Describe the steps in planning a spatial analysis
  - Understand the structure of relational databases
  - Begin building a database for spatial analysis

What is spatial analysis?

What is spatial analysis?

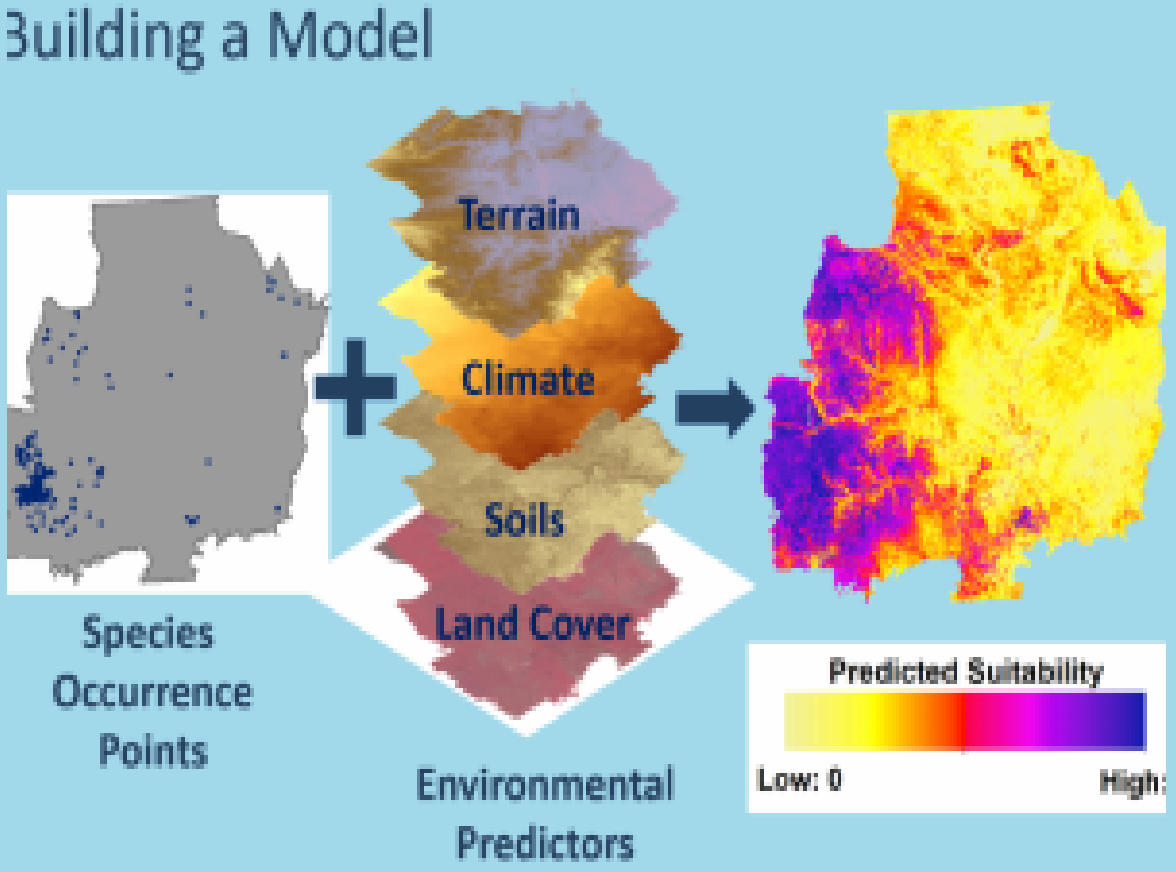
> "The process of examining the locations, attributes, and relationships of features in spatial data through overlay and other analytical techniques in order to address a question or gain useful knowledge. Spatial analysis extracts or creates new information from spatial data".
> — ESRI Dictionary

What is spatial analysis?

- The process of turning maps into information
- Any- or everything we do with GIS
- The use of computational and statistical algorithms to understand the relations between things that co-occur in space.

John Snow's cholera outbreak map

Common goals for spatial analysis



- Describe and visualize locations or events
- Quantify patterns
- Characterize 'suitability'
- Determine (statistical) relations

courtesy of NatureServe

Common pitfalls of spatial analysis

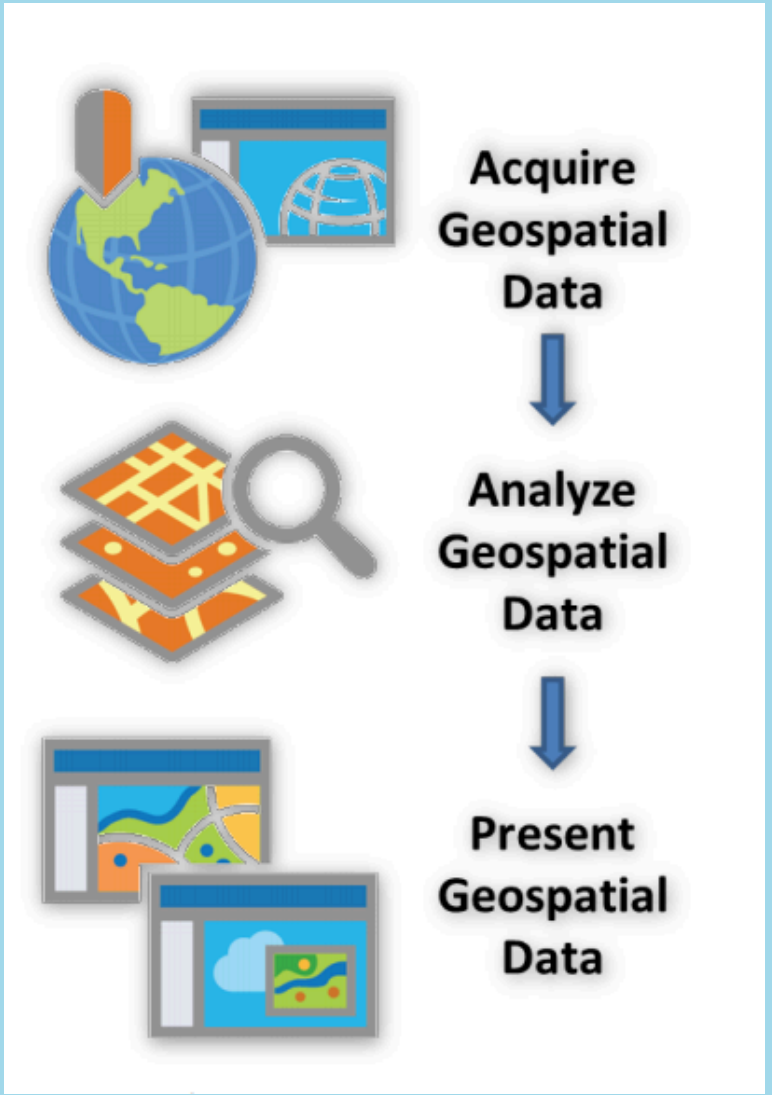
- **Locational Fallacy:** Error due to the spatial characterization chosen for elements of study
- **Atomic Fallacy:** Applying conclusions from individuals to entire spatial units
- **Ecological Fallacy:** Applying conclusions from aggregated information to individuals

> Spatial analysis is an inherently complex endeavor and one that is advancing rapidly. So-called "best practices" for addressing many of these issues are still being developed and debated. This doesn't mean you shouldn't do spatial analysis, but you should keep these things in mind as you design, implement, and interpret your analyses

Workflows for spatial analysis

Workflows for spatial analysis

- Acquisition (not really a focus, but see Resources)
- Geoprocessing
- Analysis
- Visualization



courtesy of University of Illinois

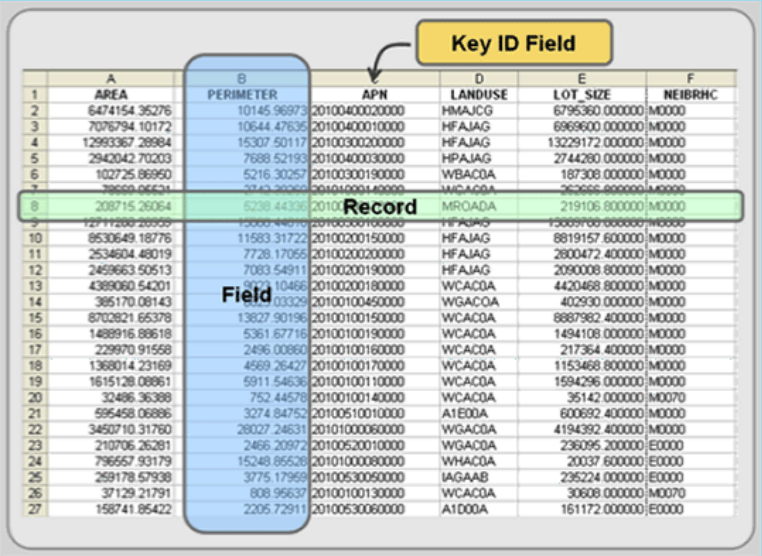
Geoprocessing

Manipulation of data for subsequent use

- Alignment
- Data cleaning and transformation
- Combination of multiple datasets
- Selection and subsetting

Databases and attributes

Databases and attributes



	A	B	C	D	E	F
1	AREA	PERIMETER	APN	LANDUSE	LOT_SIZE	NEIBRHC
2	6474154.35276	10145.96973	20100400020000	HMAJCG	6795360.000000	M0000
3	7076794.10172	10644.47635	20100400010000	HFAJAG	6969600.000000	M0000
4	12993367.28984	15307.50117	20100300200000	HFAJAG	13229172.000000	M0000
5	2942042.70203	7688.52193	20100400030000	HPAJAG	2744280.000000	M0000
6	102725.86960	5216.30257	20100300190000	WBAC0A	187308.000000	M0000
8	208715.26064	5238.44336	20100	MROADA	219106.800000	M0000
10	8530649.18776	11583.31722	20100200150000	HFAJAG	8819157.600000	M0000
11	2534604.48019	7728.17055	20100200200000	HFAJAG	2800472.400000	M0000
12	2459663.50513	7083.54911	20100200190000	HFAJAG	2090008.800000	M0000
13	4389060.54201	9023.10466	20100200180000	WCAC0A	4420468.800000	M0000
14	385170.08143	3183.03329	20100100450000	WGAC0A	402930.000000	M0000
15	8702821.65378	13827.90196	20100100150000	WCAC0A	8887982.400000	M0000
16	1488916.88618	5361.67716	20100100190000	WCAC0A	1494108.000000	M0000
17	229970.91558	2496.00860	20100100160000	WCAC0A	217364.400000	M0000
18	1368014.23169	4569.26427	20100100170000	WCAC0A	1153468.800000	M0000
19	1615128.08861	5911.54636	20100100110000	WCAC0A	1594296.000000	M0000
20	32486.36388	752.44578	20100100140000	WCAC0A	35142.000000	M0070
21	595458.06886	3274.84752	20100510010000	A1E00A	600692.400000	M0000
22	3450710.31760	28027.24631	20101000060000	WGAC0A	4194392.400000	M0000
23	210706.26281	2466.20972	20100520010000	WGAC0A	236095.200000	E0000
24	796557.93179	15248.85528	20101000080000	WHAC0A	20037.600000	E0000
25	269178.57938	3775.17959	20100530050000	IAGAAB	235224.000000	E0000
26	37129.21791	808.95637	20100100130000	WCAC0A	30608.000000	M0070
27	158741.85422	2205.72911	20100530060000	A1D00A	161172.000000	E0000

courtesy of Giscommons

- Previous focus has been largely on *location*
- Geographic data often also includes non-spatial data
- Attributes: Non-spatial information that further describes a spatial feature
- Typically stored in tables where each row represents a spatial feature
  - Wide vs. long format

Common attribute operations

- `sf` designed to work with `tidyverse`
- Allows use of `dplyr` data manipulation verbs (e.g. `filter`, `select`, `slice`)
- Can use `scales` package for units
- Also allows `%>%` to chain together multiple steps
- geometries are "sticky"

Subsetting by Field

Subsetting by Features

- Features refer to the individual observations in the dataset
- Selecting features

```
1  head(world)[1:3, 1:3] %>%
2    st_drop_geometry()
```

```
# A tibble: 3 × 3
  iso_a2 name_long      continent
* <chr>  <chr>          <chr>
1 FJ     Fiji           Oceania
2 TZ     Tanzania       Africa
3 EH     Western Sahara Africa
```

```
1  world %>%
2    filter(continent == "Asia") %>%
3      dplyr::select(name_long, conti
4    st_drop_geometry() %>%
5    head(.)
```

```
# A tibble: 6 × 2
  name_long   continent
  <chr>       <chr>
1 Kazakhstan  Asia
2 Uzbekistan  Asia
3 Indonesia   Asia
4 Timor-Leste Asia
5 Israel      Asia
6 Lebanon     Asia
```

Revisiting the tidyverse

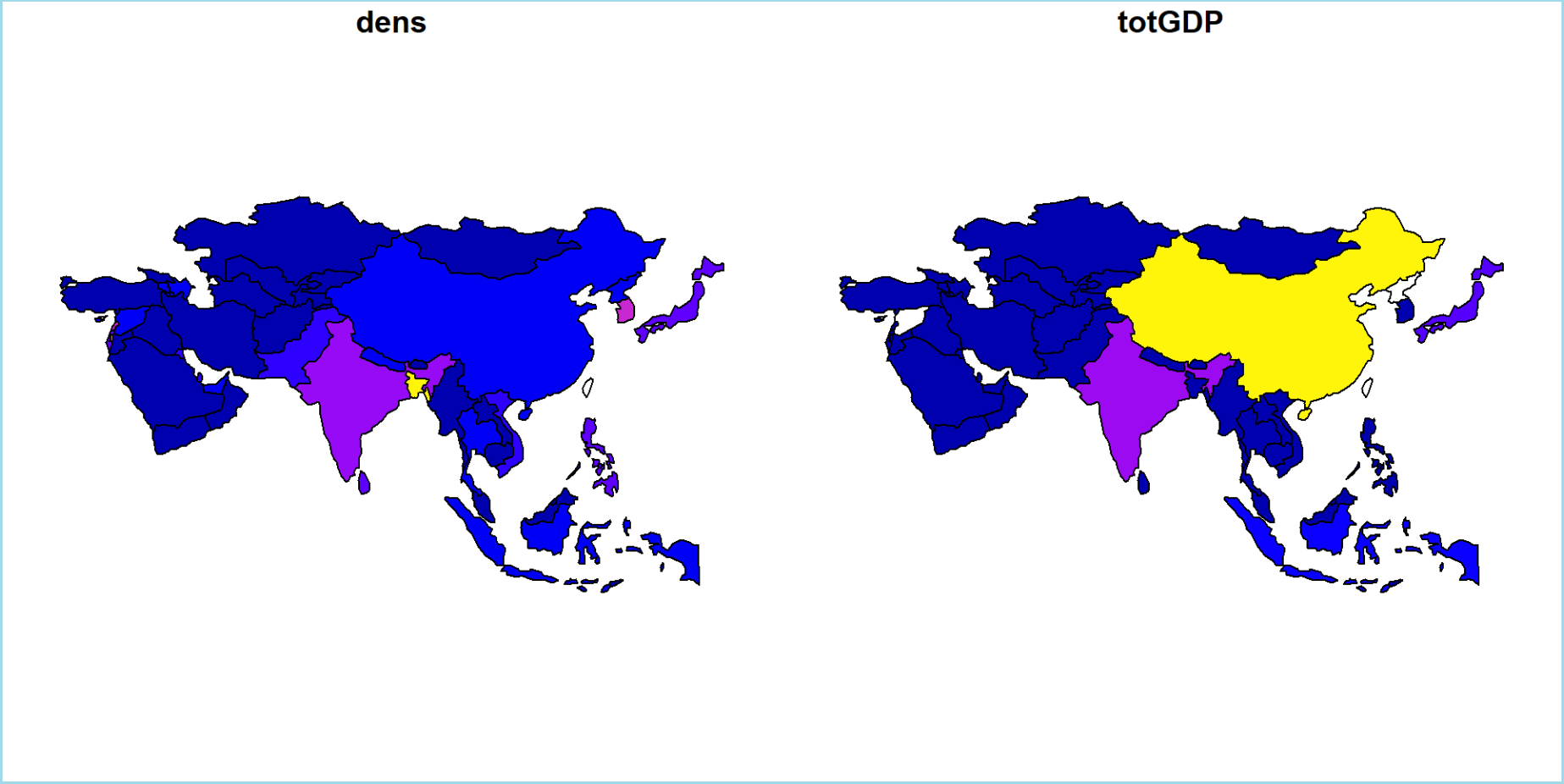
- Creating new fields

```
world %>%
  filter(continent == "Asia") %>%
    dplyr::select(name_long, continent, pop, gdpPercap ,area_km2) %>%
  mutate(., dens = pop/area_km2,
         totGDP = gdpPercap * pop) %>%
  st_drop_geometry() %>%
  head(.)
```

```
# A tibble: 6 × 7
  name_long   continent       pop gdpPercap area_km2   dens  totGDP
  <chr>       <chr>         <dbl>     <dbl>    <dbl>  <dbl>   <dbl>
1 Kazakhstan  Asia       17288285    23587. 2729811.   6.33 4.08e11
2 Uzbekistan  Asia       30757700     5371.  461410.  66.7  1.65e11
3 Indonesia   Asia      255131116    10003. 1819251. 140.   2.55e12
4 Timor-Leste Asia        1212814     6263.   14715.  82.4  7.60e 9
5 Israel      Asia        8215700    31702.   22991. 357.   2.60e11
6 Lebanon     Asia        5603279    13831.   10099. 555.   7.75e10
```

Revisiting the `tidyverse`

- Creating new fields

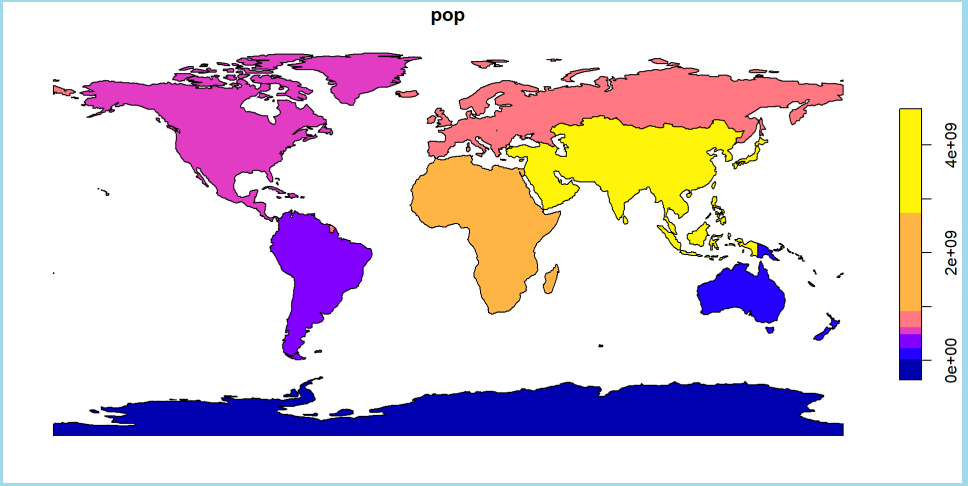


Revisiting the `tidyverse`

- Aggregating data

```
world %>%
  st_drop_geometry(.) %>%
  group_by(continent) %>%
  summarize(pop = sum(pop, na.rm =
```

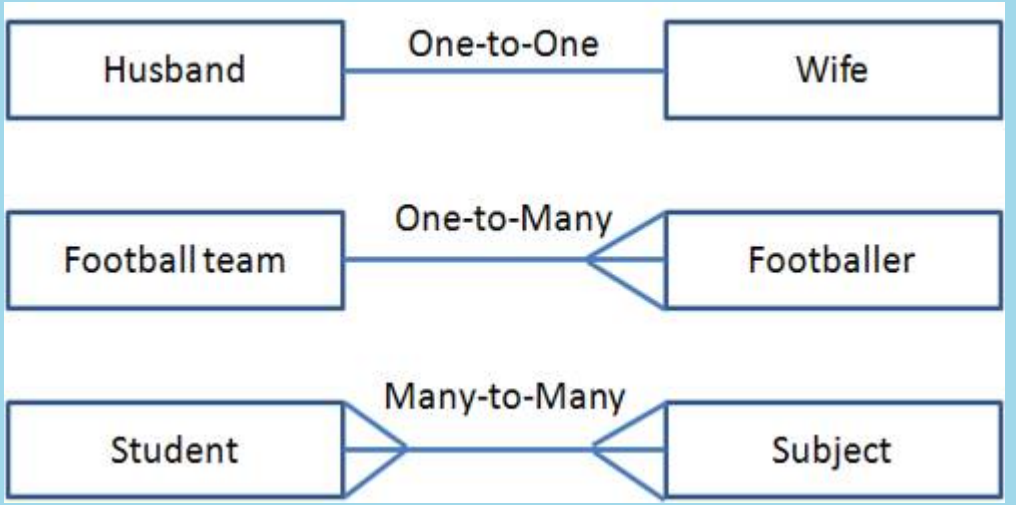
```
# A tibble: 8 × 2
  continent                       pop
  <chr>                         <dbl>
1 Africa                   1154946633
2 Antarctica                        0
3 Asia                     4311408059
4 Europe                    669036256
5 North America             565028684
6 Oceania                    37757833
7 Seven seas (open ocean)           0
8 South America             412060811
```



Joining (a)spatial data

Joining (a)spatial data

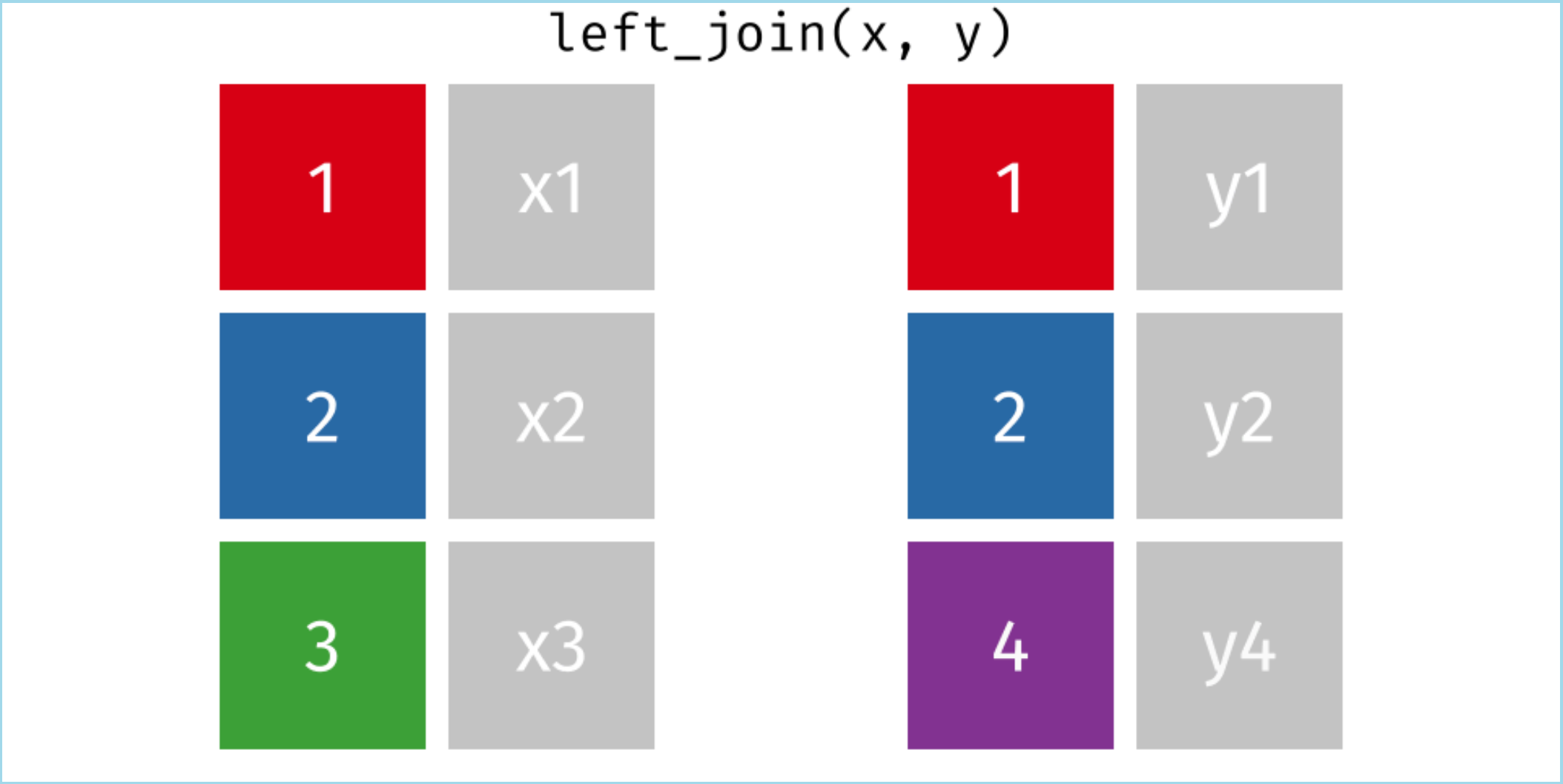
- Requires a “key” field
- Multiple outcomes possible
- Think about your final data form



Left Join

- Useful for adding other attributes not in your spatial data
- Returns all of the records in x attributed with y
- Pay attention to the number of rows!

Left Join



Left Join

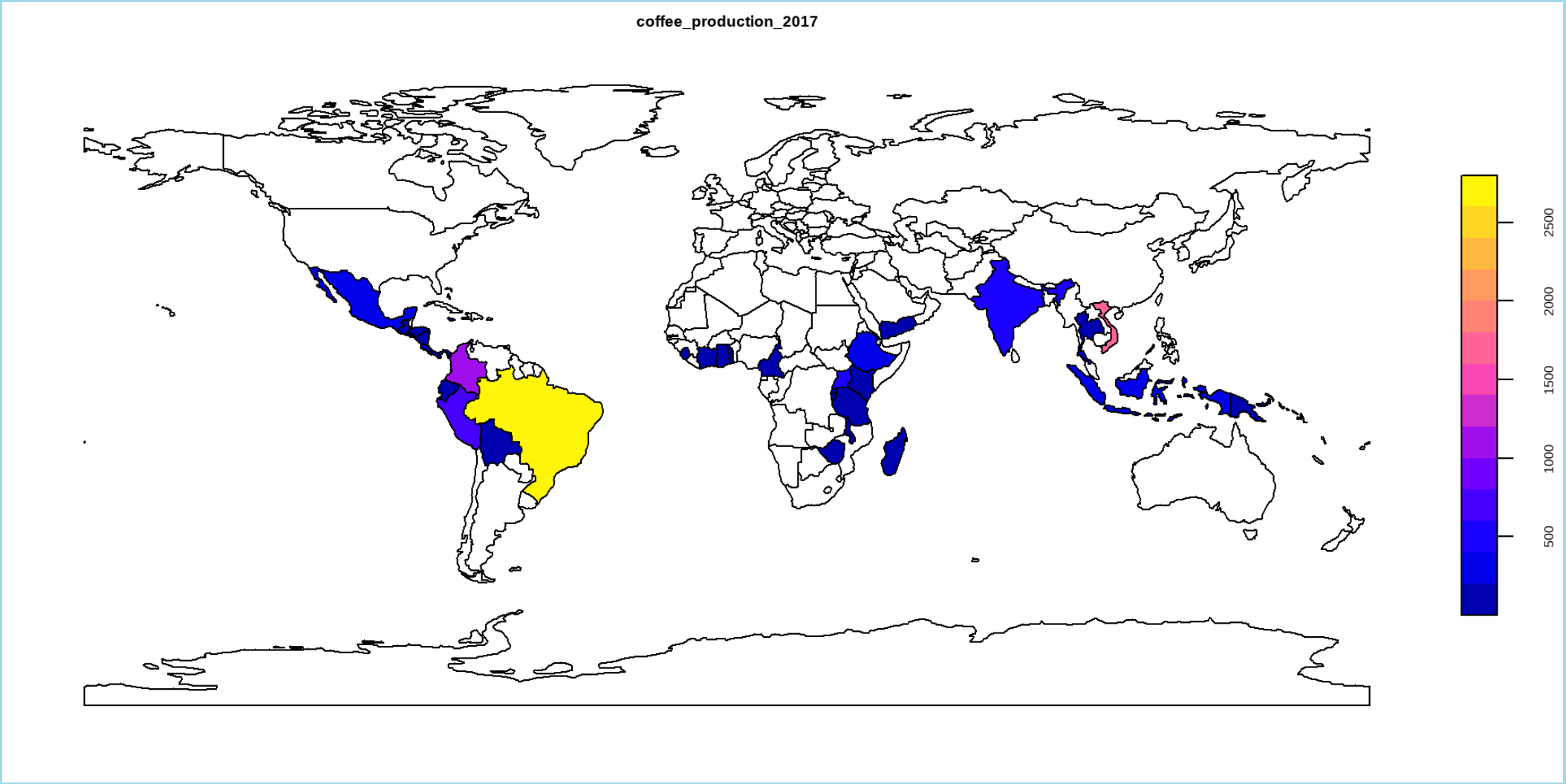
```
1 head(coffee_data)
```

```
# A tibble: 6 × 3
  name_long
coffee_production_2016
coffee_production_2017
  <chr>
<int>                  <int>
1 Angola
NA                     NA
2 Bolivia
3                      4
3 Brazil
3277                   2786
4 Burundi
37                     38
5 Cameroon
```

```
1 world_coffee = left_join(world, cc
2 nrow(world_coffee)
```

```
[1] 177
```

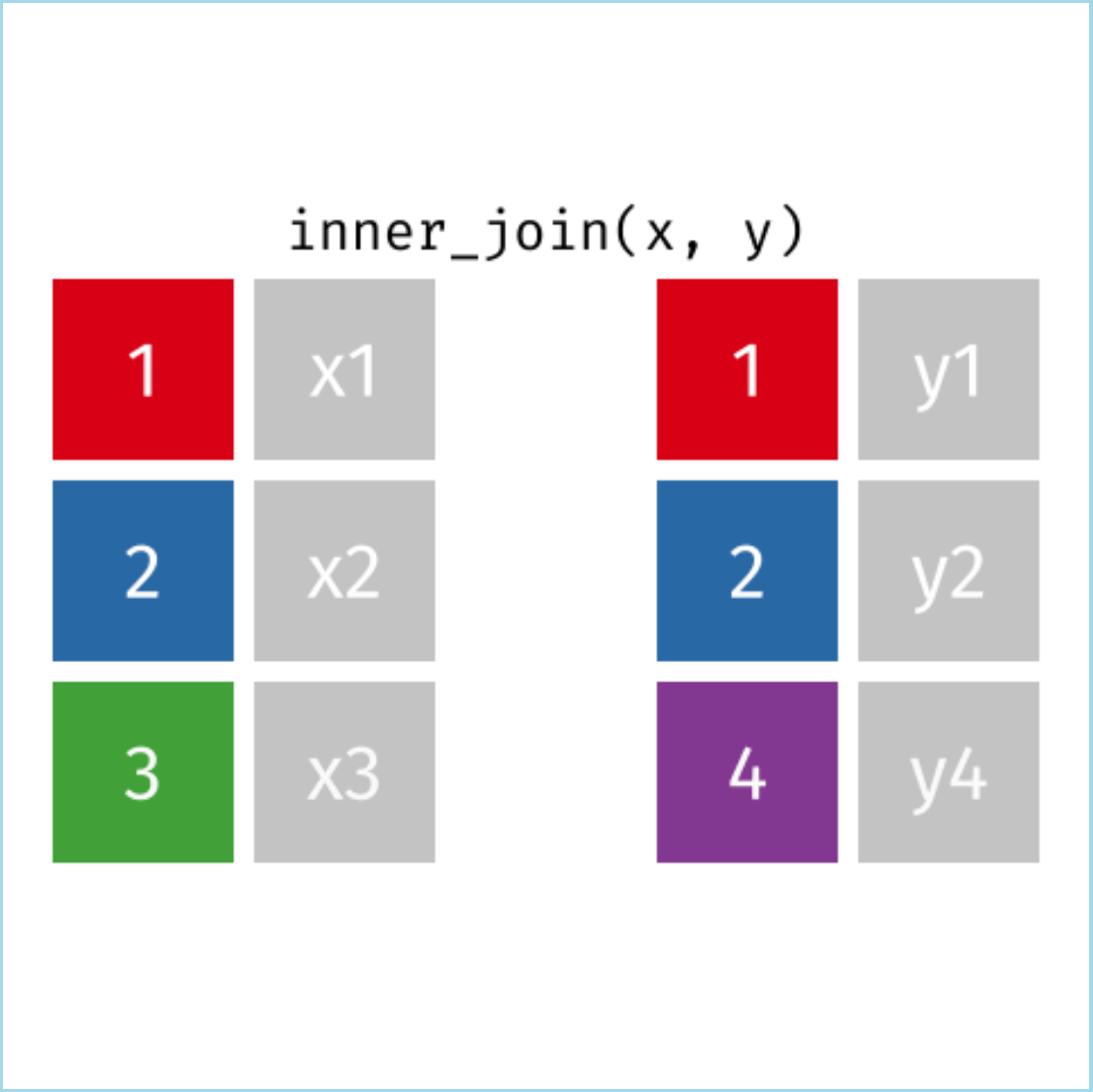
Left Join



Inner Join

- Useful for subsetting to “complete” records
- Returns all of the records in x with matching y
- Pay attention to the number of rows!

Inner Join



Inner Join

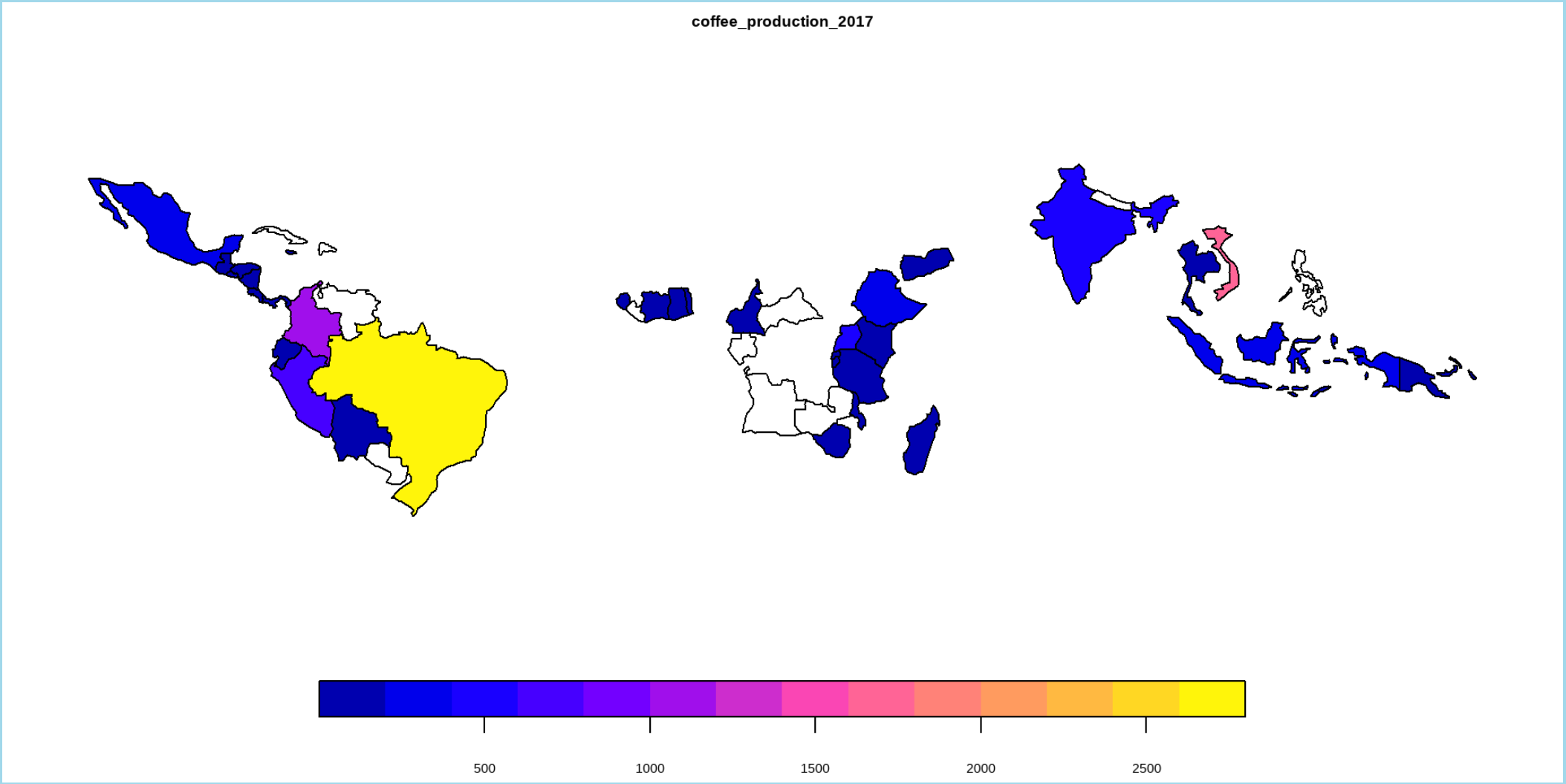
```
1 world_coffee_inner = inner_join(wo
2 nrow(world_coffee_inner)
```

```
[1] 45
```

```
1 setdiff(coffee_data$name_long, wor
```

```
[1] "Congo, Dem. Rep. of" "Others"
```

Inner Join



Other Joins

- `right_`, `outer_`, and `anti_`
- Spatial Joins (next week)