Building Spatial Databases with Attributes

HES 505 Fall 2022: Session 14

Matt Williamson

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

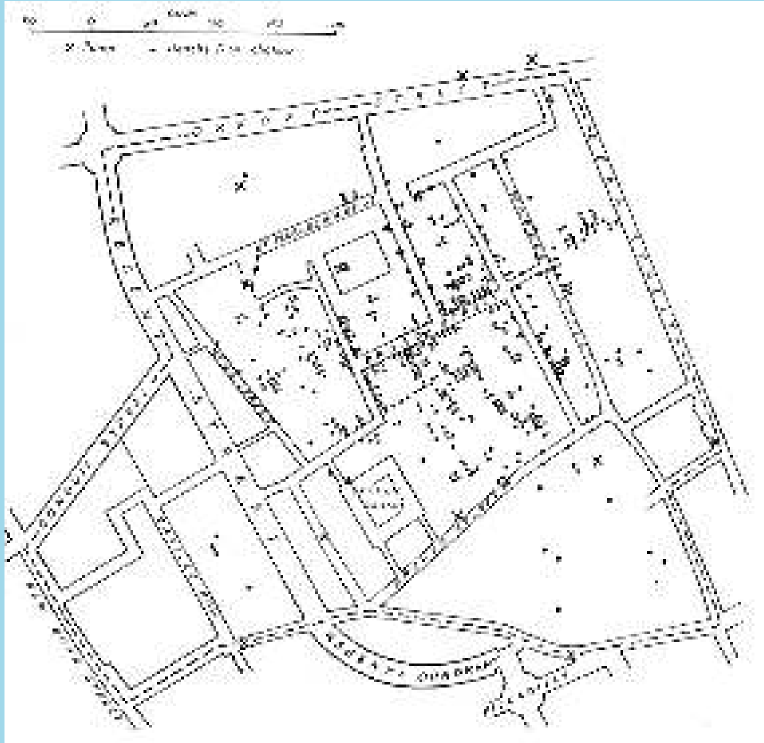
What is spatial analysis?

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

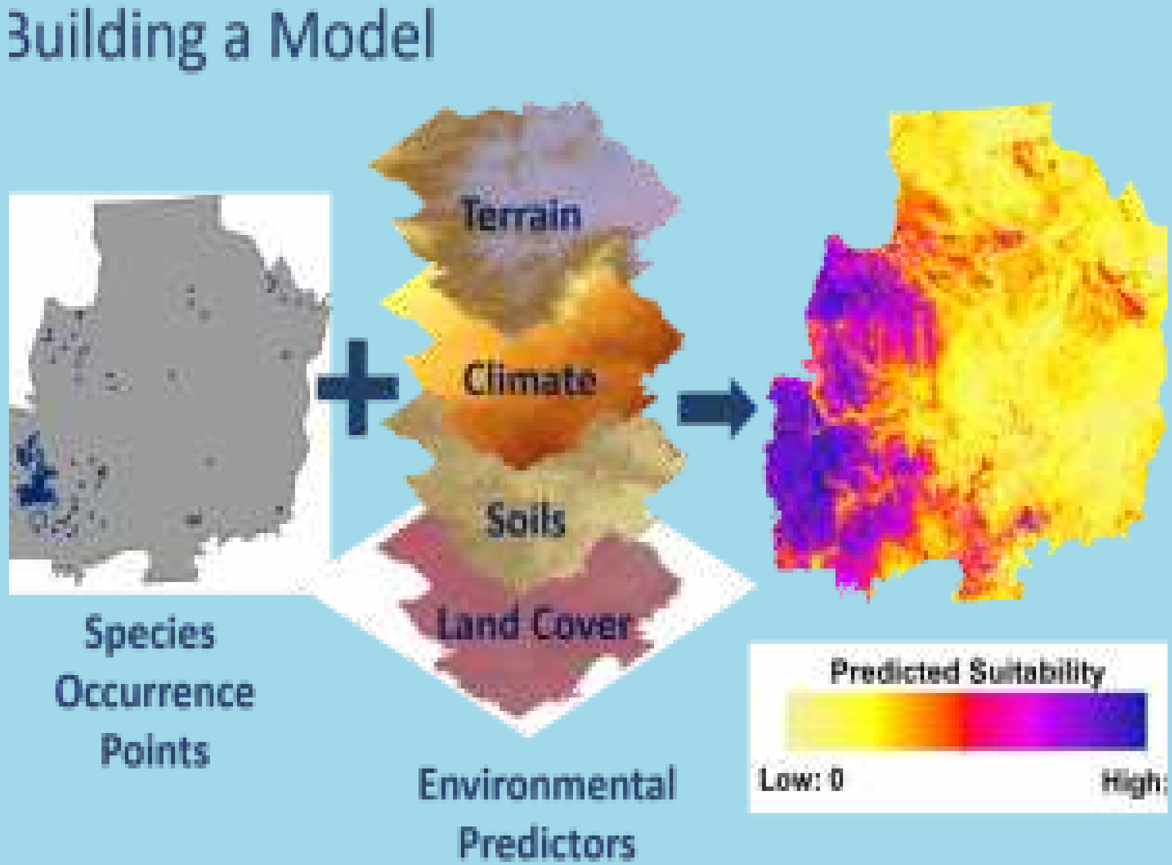
What is spatial analysis?

- 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

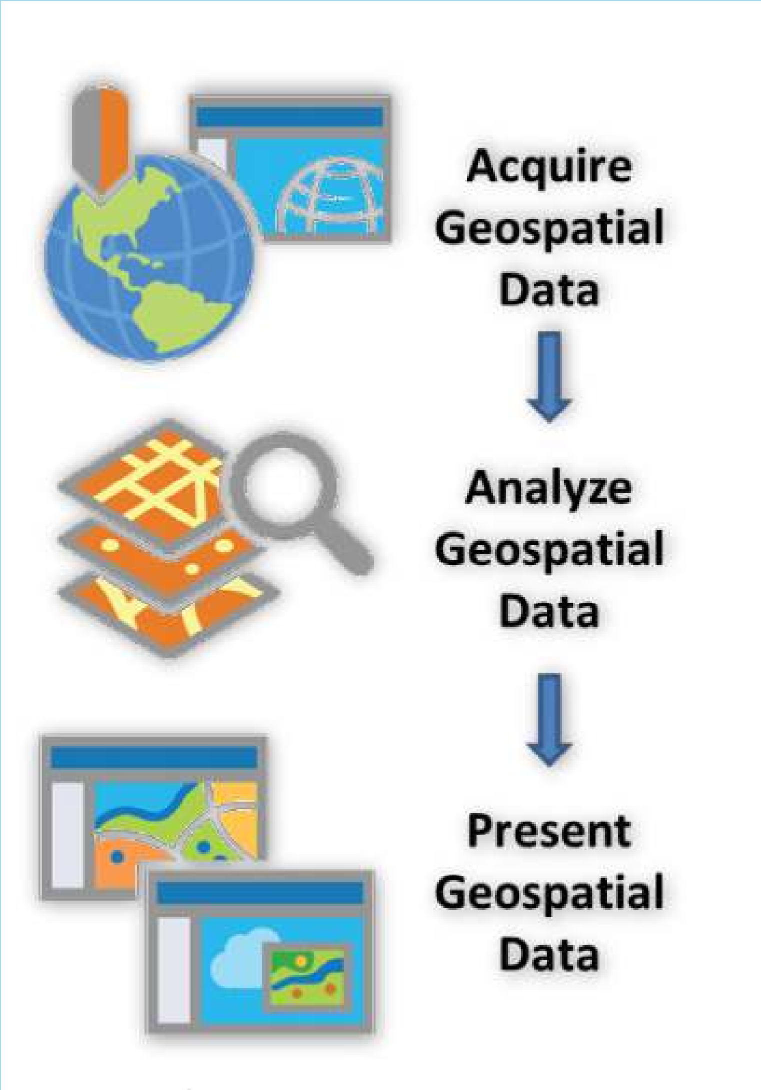
- **Locational Fallacy:** Error due to the spatial characterization chosen for elements of study
- **Atomic Fallacy:** Applying conclusions from individuals to entire spatial units
- **Ecological Fallacy:** Applying conclusions from aggregated information to individuals

> 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

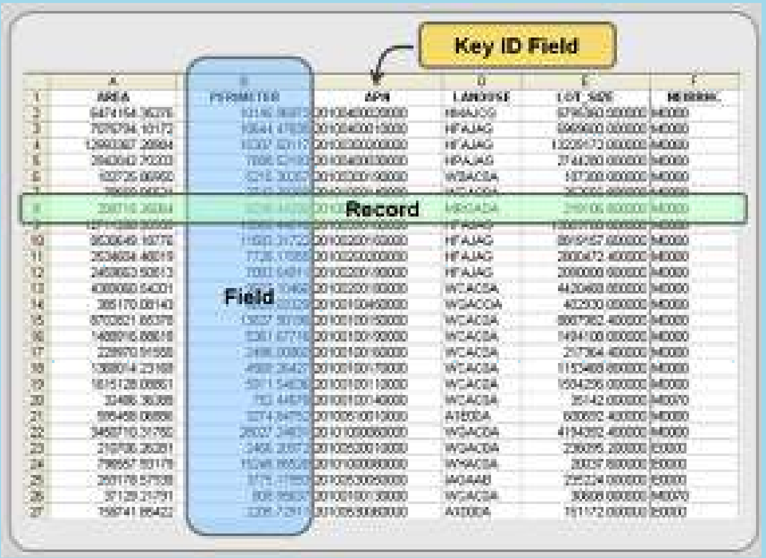
Geoprocessing

Manipulation of data for subsequent use

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

Databases and attributes

Databases and attributes



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
- Can use `scales` package for units
- Also allows `%>%` to chain together multiple steps
- geometries are "sticky"
- Pay attention to masking!!

Revisiting the `tidyverse`

Revisiting the `tidyverse`

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

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

```
world %>%
  filter(continent == "Asia") %>%
    dplyr::select(name_long, conti
  st_drop_geometry() %>%
  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`

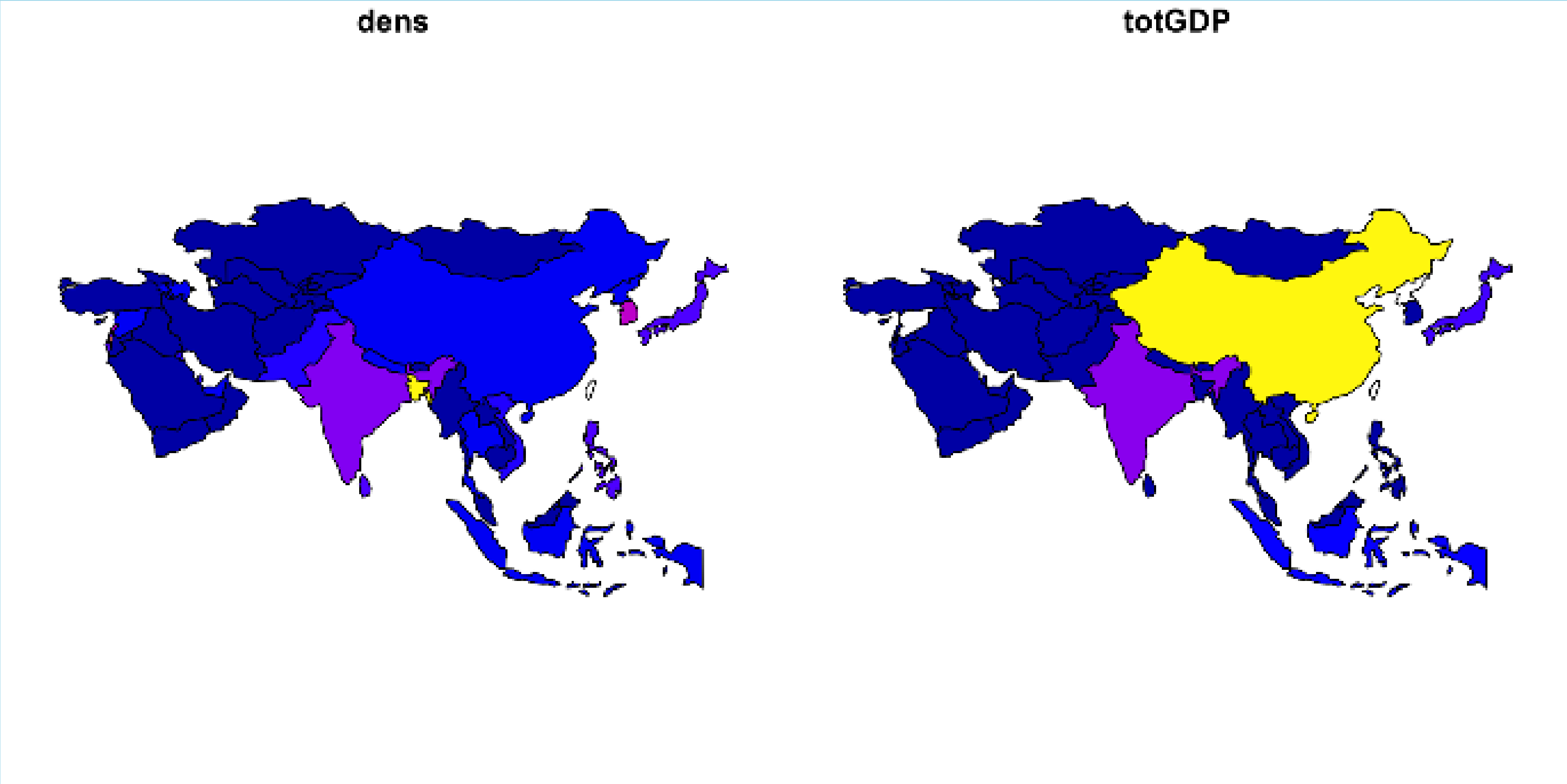
- Creating new fields

```
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

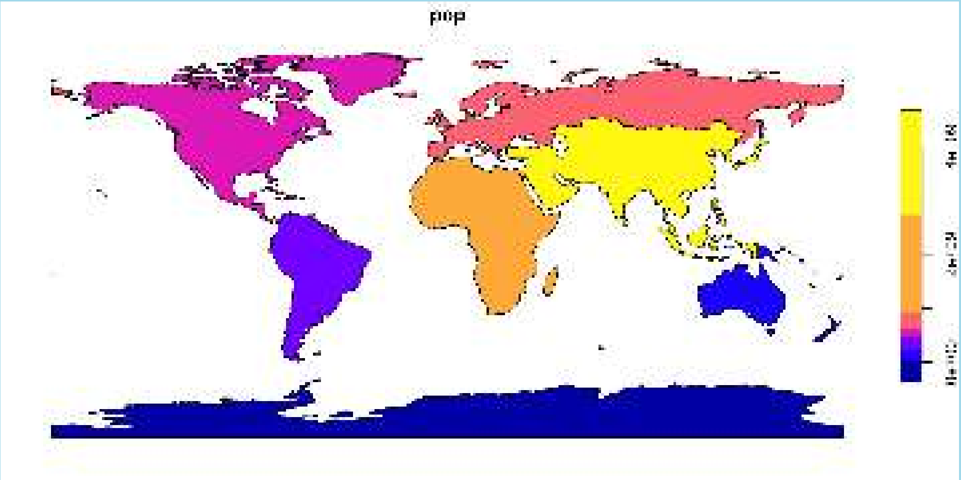


Revisiting the `tidyverse`

- Aggregating data

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

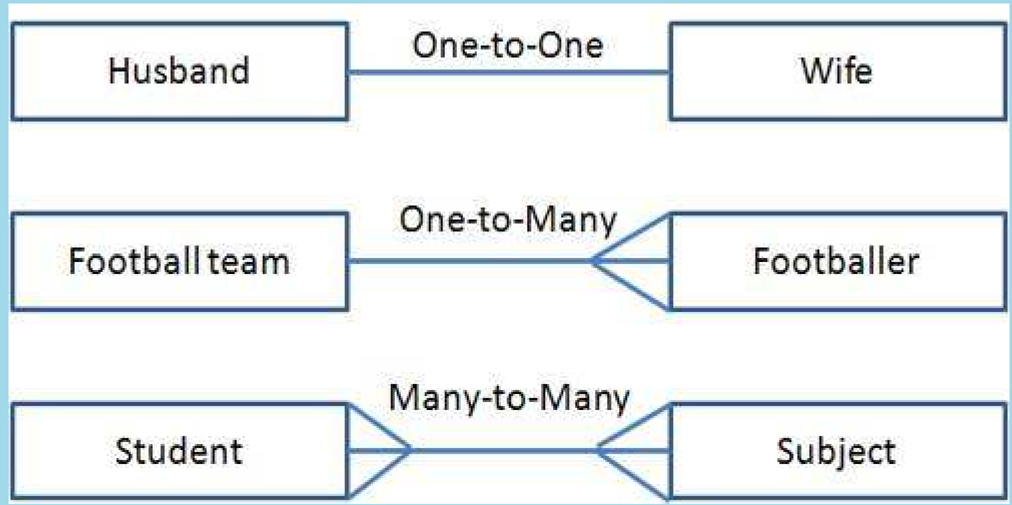
```
# A tibble: 8 × 2
  continent                      pop
  <chr>                        <dbl>
1 Africa                  1154946633
2 Antarctica                       0
3 Asia                    4311408059
4 Europe                   669036256
5 North America            565028684
6 Oceania          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



Joining (a)spatial data

Joining (a)spatial data

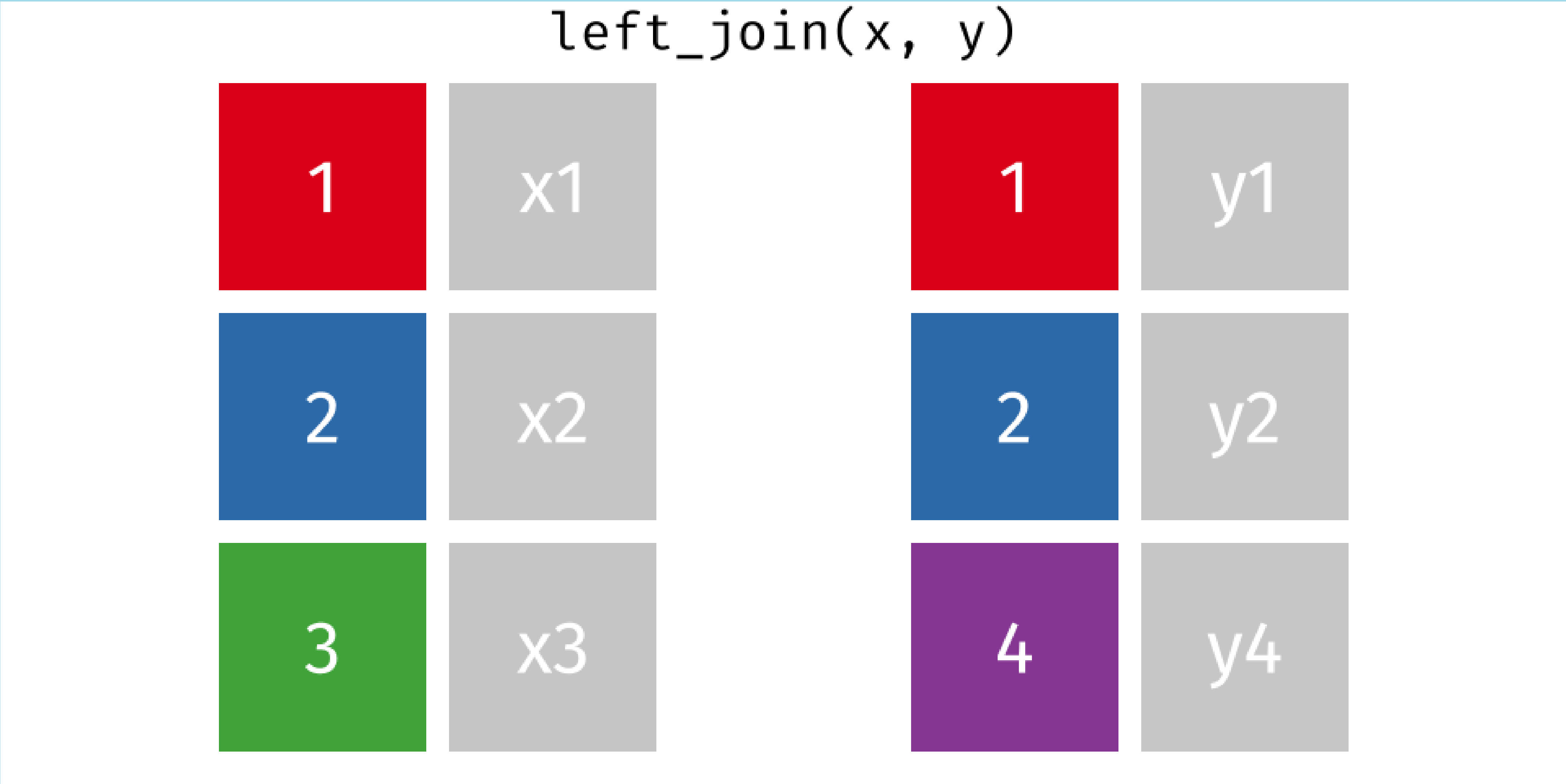
- Requires a “key” field
- Multiple outcomes possible
- Think about your final data form



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

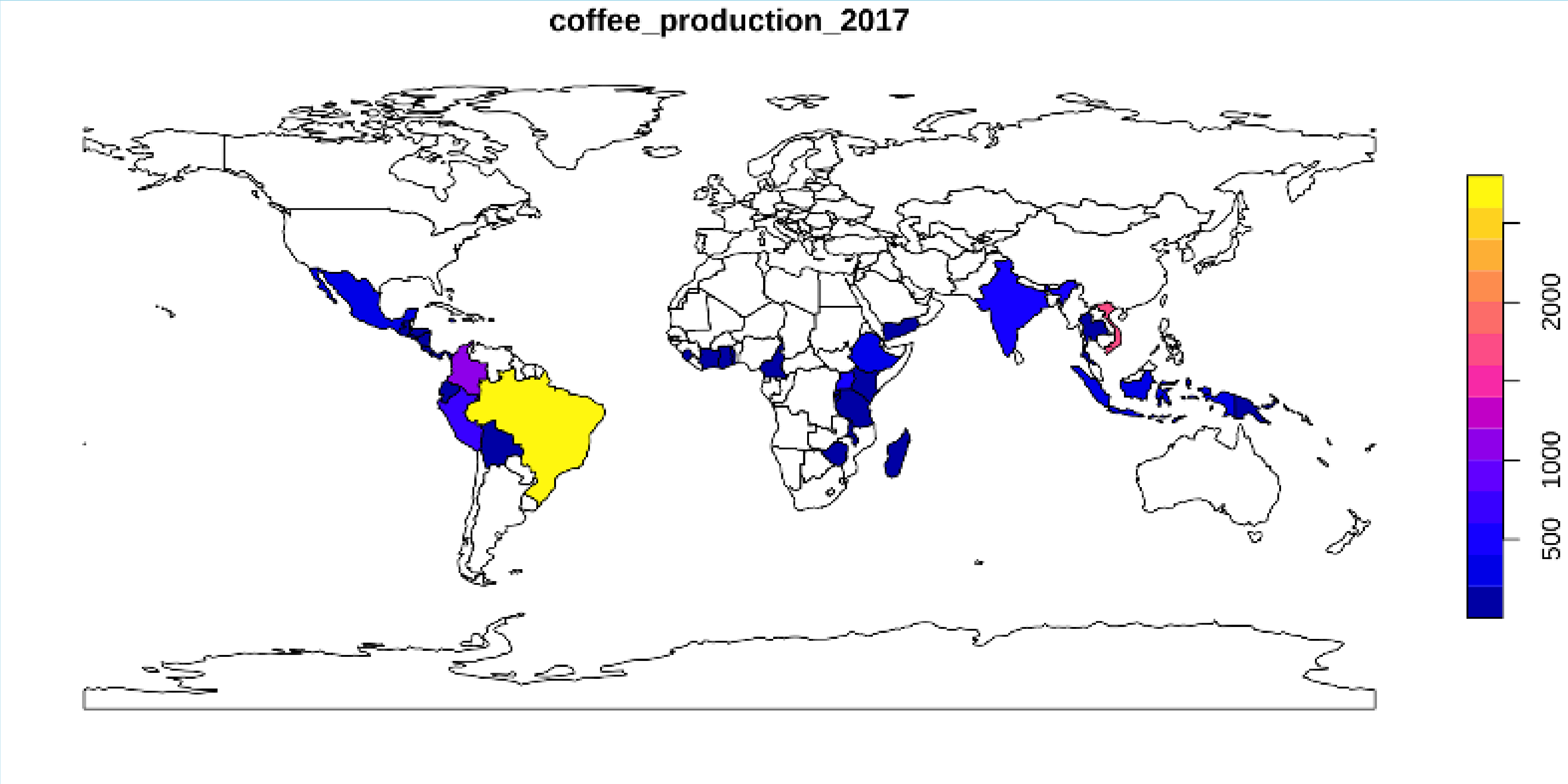
```r
head(coffee_data)
```

```
# 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
```

```r
world_coffee = left_join(world, co
nrow(world_coffee)
```

```
[1] 177
```

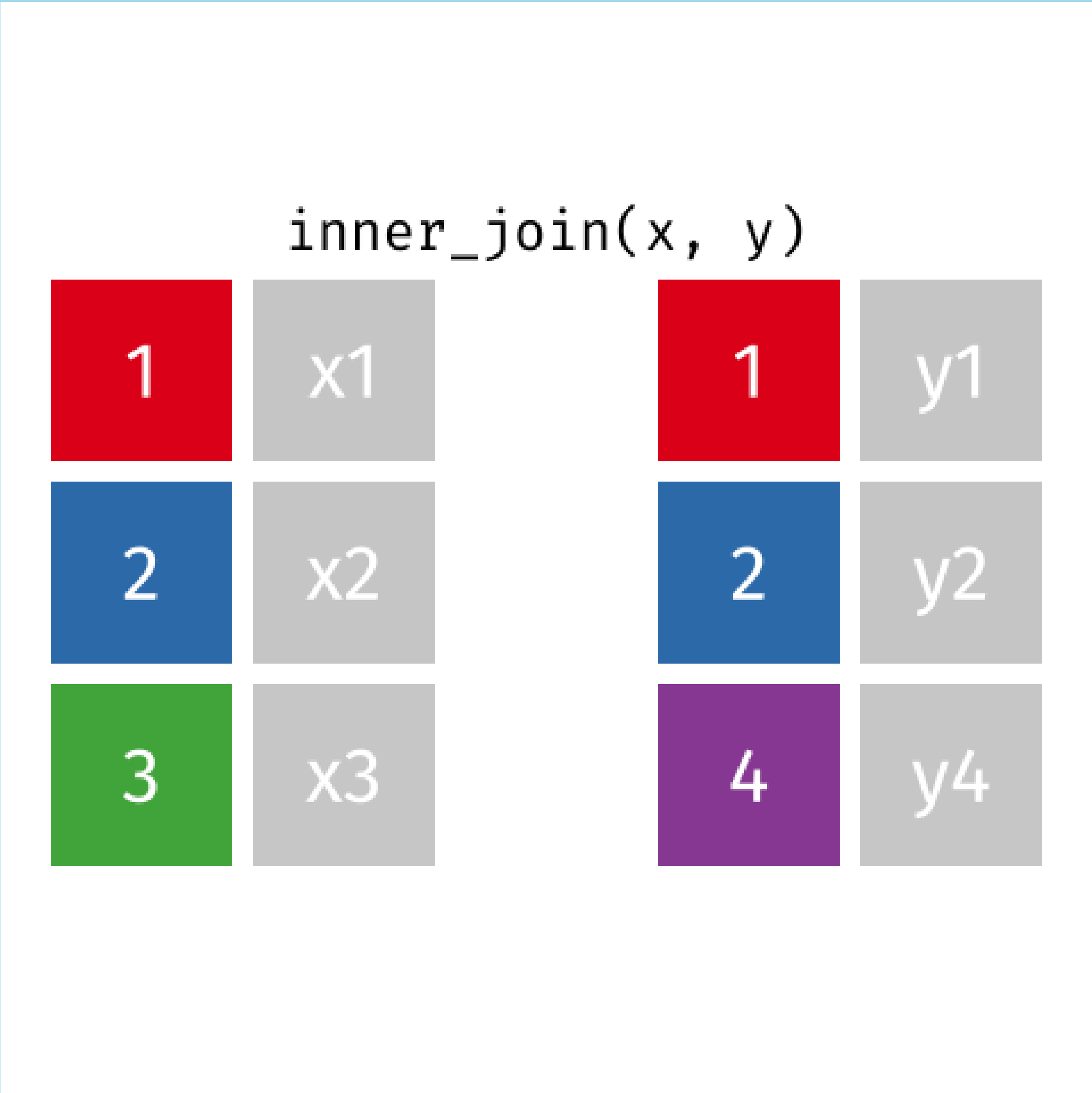
Left Join



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

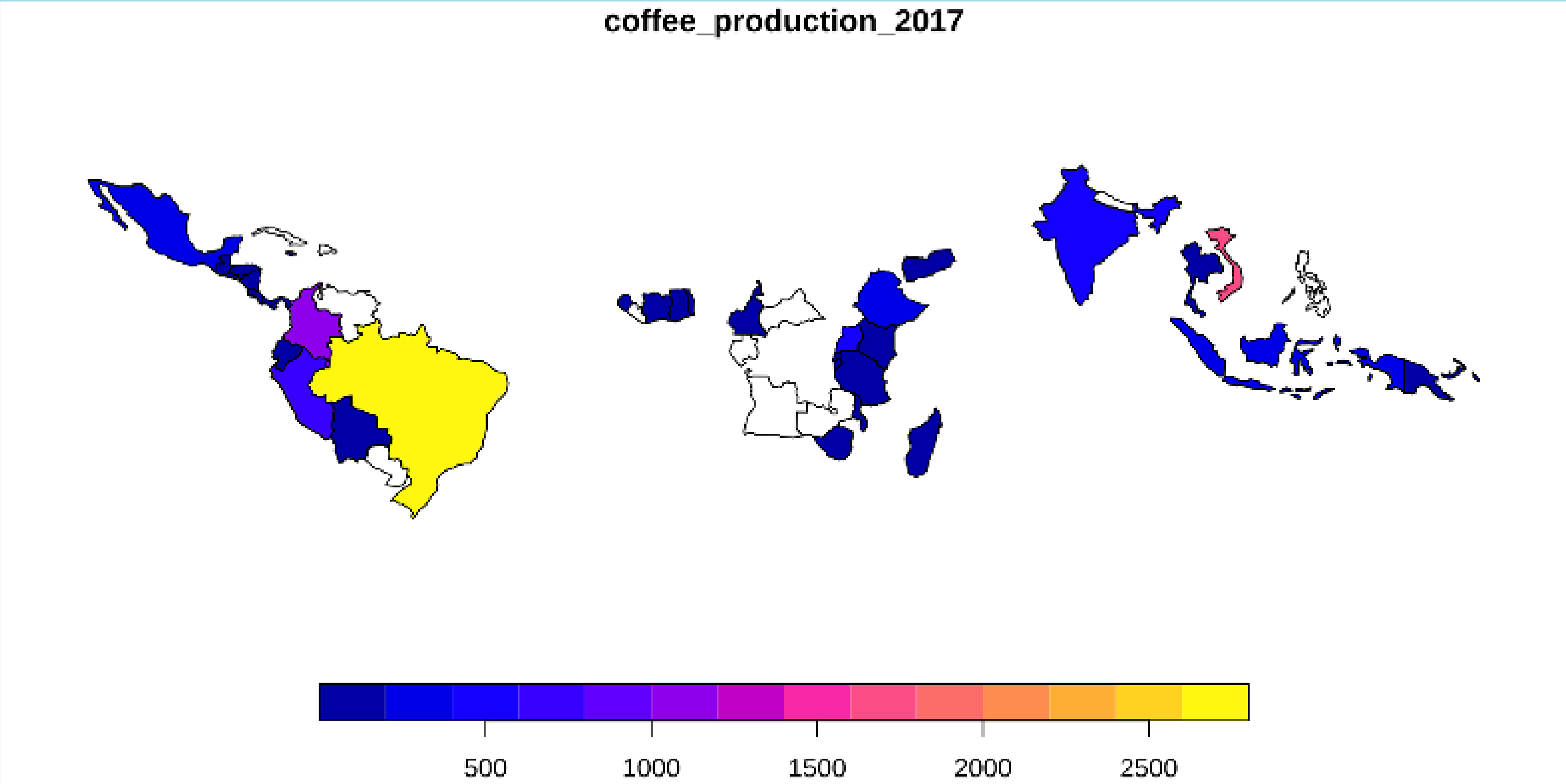
```r
world_coffee_inner = inner_join(wo
nrow(world_coffee_inner)
```

```
[1] 45
```

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

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

Inner Join



Other Joins

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