Lecture 03: Spatial Data

Theory and Tools (a.k.a. GIS Tools Lab.)

Bruno Conte

25/Sep/2023

Spatial data in economics: schedule

1. ~~Introduction to (spatial) data and programming in R~~ [18.Sep.2023]

2. ~~Spatial data basics: vector data + assignment~~ [21.Sep.2023]

3. Basic operations with vector data ~~+ assignment~~ [25.Sep.2023]

    - Attribute operations with vector data (slicing, filtering, aggregating)
    - Spatial operations of vetor data (e.g. intersections, touching, etc.)
    - Spatial merging/joining (based on overlaps and/or distances)

4. Geometry operations and miscelanea + follow-up + assignment [28.Sep.2023]

5. Raster data and operations + assignment [02.Oct.2023]

6. Take-home exam [03.Nov.2023]

Main references for this class

1. Lovelace, R., Nowosad, J. and Muenchow, J., 2019. **Geocomputation with R.** Chapman and Hall/CRC.

    - Chapter 3.2 (attribute data operations)
    - Chapter 4 (spatial data operations)

2. Pebesma, E., 2018. Simple Features for R: Standardized Support for Spatial Vector Data. The R Journal 10 (1), 439-446

3. Wickham, H. and Grolemund, G., 2016. R for data science: import, tidy, transform, visualize, and model data. " O'Reilly Media, Inc.".

Vector data operations: attribute and spatial

- Data operations: manipulation of vector data (in terms of geometry and attribute structure). Basic operations are:
  - **Selecting:** restricting the fields (i.e. variables) of a `sf`
  - **Slicing:** restricting the features (i.e. observations) of a `sf`
  - **Filtering:** restricting based on data attributes
  - **Joining/merging:** linking attributes (i.e. data) between different `sf` (or data sets)
  - **Aggregating:** processing attributes (i.e. data) within a `sf` based on some fields
- Operations can be either attribute- or spatial-based

Attribute data operations: selecting (choose fields)

```
world
```

```
## Simple feature collection with 177 features and 10 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -180 ymin: -90 xmax: 180 ymax: 83.64513
## Geodetic CRS:  WGS 84
## # A tibble: 177 × 11
##    iso_a2 name_long      conti…¹ regio…² subre…³ type  area_…⁴     pop
##    <chr>  <chr>          <chr>   <chr>   <chr>   <chr>   <dbl>   <dbl>
##  1 FJ     Fiji           Oceania Oceania Melane… Sove…  1.93e4  8.86e5
##  2 TZ     Tanzania       Africa  Africa  Easter… Sove…  9.33e5  5.22e7
##  3 EH     Western Sahara Africa  Africa  Northe… Inde…  9.63e4 NA
##  4 CA     Canada         North … Americ… Northe… Sove…  1.00e7  3.55e7
##  5 US     United States  North … Americ… Northe… Coun…  9.51e6  3.19e8
##  6 KZ     Kazakhstan     Asia    Asia    Centra… Sove…  2.73e6  1.73e7
##  7 UZ     Uzbekistan     Asia    Asia    Centra… Sove…  4.61e5  3.08e7
##  8 PG     Papua New Gui… Oceania Oceania Melane… Sove…  4.65e5  7.76e6
##  9 ID     Indonesia      Asia    Asia    South-… Sove…  1.82e6  2.55e8
```

Attribute data operations: selecting (choose fields)

```
world %>% select(name_long, continent)
```

```
## Simple feature collection with 177 features and 2 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -180 ymin: -90 xmax: 180 ymax: 83.64513
## Geodetic CRS:  WGS 84
## # A tibble: 177 × 3
##    name_long        continent                                      geom
##    <chr>            <chr>                                <MULTIPOLYGON [°]>
##  1 Fiji             Oceania       (((180 -16.06713, 180 -16.55522, 17…
##  2 Tanzania         Africa        (((33.90371 -0.95, 34.07262 -1.0598…
##  3 Western Sahara   Africa        (((-8.66559 27.65643, -8.665124 27.…
##  4 Canada           North America (((-122.84 49, -122.9742 49.00254, …
##  5 United States    North America (((-122.84 49, -120 49, -117.0312 4…
##  6 Kazakhstan       Asia          (((87.35997 49.21498, 86.59878 48.5…
##  7 Uzbekistan       Asia          (((55.96819 41.30864, 55.92892 44.9…
##  8 Papua New Guinea Oceania       (((141.0002 -2.600151, 142.7352 -3.…
##  9 Indonesia        Asia          (((141.0002 -2.600151, 141.0171 -5.…
```

Attribute data operations: slicing (choose observations)

```
world %>% select(name_long, continent) %>% slice(1:2)
```

```
## Simple feature collection with 2 features and 2 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -180 ymin: -18.28799 xmax: 180 ymax: -0.95
## Geodetic CRS:  WGS 84
## # A tibble: 2 × 3
##   name_long continent                                             geom
##   <chr>     <chr>                                       <MULTIPOLYGON [°]>
## 1 Fiji      Oceania   (((180 -16.06713, 180 -16.55522, 179.3641 -16.8…
## 2 Tanzania  Africa    (((33.90371 -0.95, 34.07262 -1.05982, 37.69869 …
```

Attribute data operations: filtering (based on data)

```
world %>% select(name_long, continent) %>% filter(continent=='South America')
```

```
## Simple feature collection with 13 features and 2 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -81.41094 ymin: -55.61183 xmax: -34.72999 ymax: 12.4373
## Geodetic CRS:  WGS 84
## # A tibble: 13 × 3
##    name_long        continent                                        geom
##  * <chr>            <chr>                                  <MULTIPOLYGON [°]>
##  1 Argentina        South America (((-68.63401 -52.63637, -68.25 -53.…
##  2 Chile            South America (((-68.63401 -52.63637, -68.63335 -…
##  3 Falkland Islands South America (((-61.2 -51.85, -60 -51.25, -59.15…
##  4 Uruguay          South America (((-57.62513 -30.21629, -56.97603 -…
##  5 Brazil           South America (((-53.37366 -33.76838, -53.65054 -…
##  6 Bolivia          South America (((-69.52968 -10.95173, -68.78616 -…
##  7 Peru             South America (((-69.89364 -4.298187, -70.79477 -…
##  8 Colombia         South America (((-66.87633 1.253361, -67.06505 1.…
##  9 Venezuela        South America (((-60.73357 5.200277, -60.60118 4.…
```

Attribute data operations: joining (merging data)

```
world %>% select(name_long, continent) %>% filter(continent=='South America') %>%
  left_join(coffee_data) # data of coffee production by country (name_long)
```

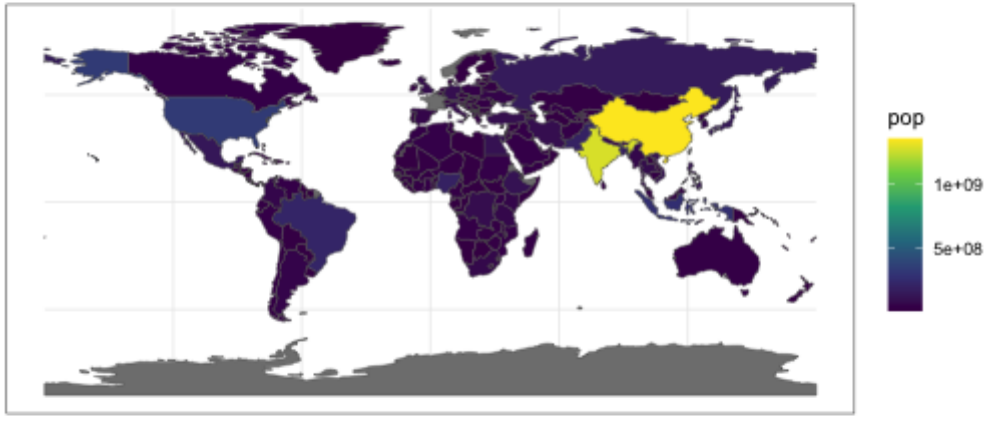
```
## Simple feature collection with 13 features and 4 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -81.41094 ymin: -55.61183 xmax: -34.72999 ymax: 12.4373
## Geodetic CRS:  WGS 84
## # A tibble: 13 × 5
##    name_long        contin…¹                       geom coffe…² coffe…³
##    <chr>            <chr>           <MULTIPOLYGON [°]>   <int>   <int>
##  1 Argentina        South A… (((-68.63401 -52.63637, …      NA      NA
##  2 Chile            South A… (((-68.63401 -52.63637, …      NA      NA
##  3 Falkland Islands South A… (((-61.2 -51.85, -60 -51…      NA      NA
##  4 Uruguay          South A… (((-57.62513 -30.21629, …      NA      NA
##  5 Brazil           South A… (((-53.37366 -33.76838, …    3277    2786
##  6 Bolivia          South A… (((-69.52968 -10.95173, …       3       4
##  7 Peru             South A… (((-69.89364 -4.298187, …     585     625
##  8 Colombia         South A… (((-66.87633 1.253361, -…    1330    1169
```

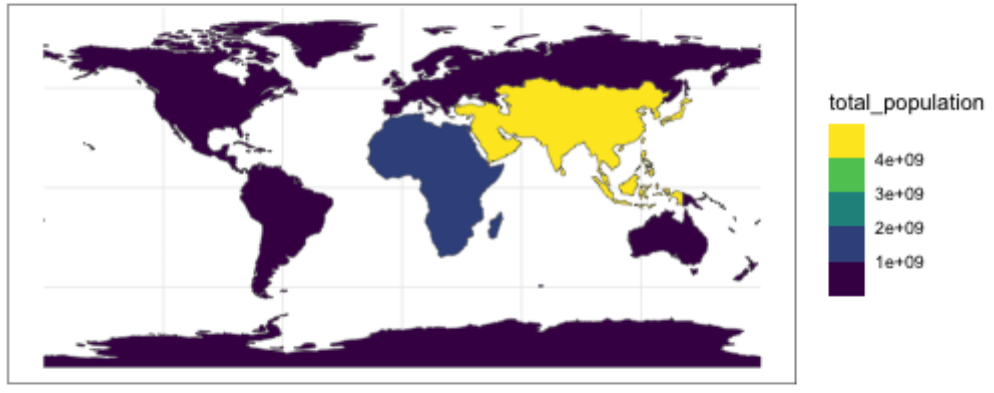
Attribute data operations: aggregating (based on attributes)

```
world %>% select(name_long, continent, pop) %>%
  group_by(continent) %>%
  summarise(total_population = sum(pop, na.rm = T))
```

```
## Simple feature collection with 8 features and 2 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -180 ymin: -90 xmax: 180 ymax: 83.64513
## Geodetic CRS:  WGS 84
## # A tibble: 8 × 3
##   continent                total_population                            geom
##   <chr>                               <dbl>              <MULTIPOLYGON [°]>
## 1 Africa                         1154946633 (((40.43725 -11.76171, 40.…
## 2 Antarctica                           0 (((-48.66062 -78.04702, -4…
## 3 Asia                           4311408059 (((120.295 -10.25865, 118.…
## 4 Europe                          669036256 (((-53.77852 2.376703, -54…
## 5 North America                   565028684 (((-78.21494 7.512255, -78…
## 6 Oceania                          37757833 (((171.9487 -41.51442, 172…
## 7 Seven seas (open ocean)                0 (((68.935 -48.625, 69.58 -…
```

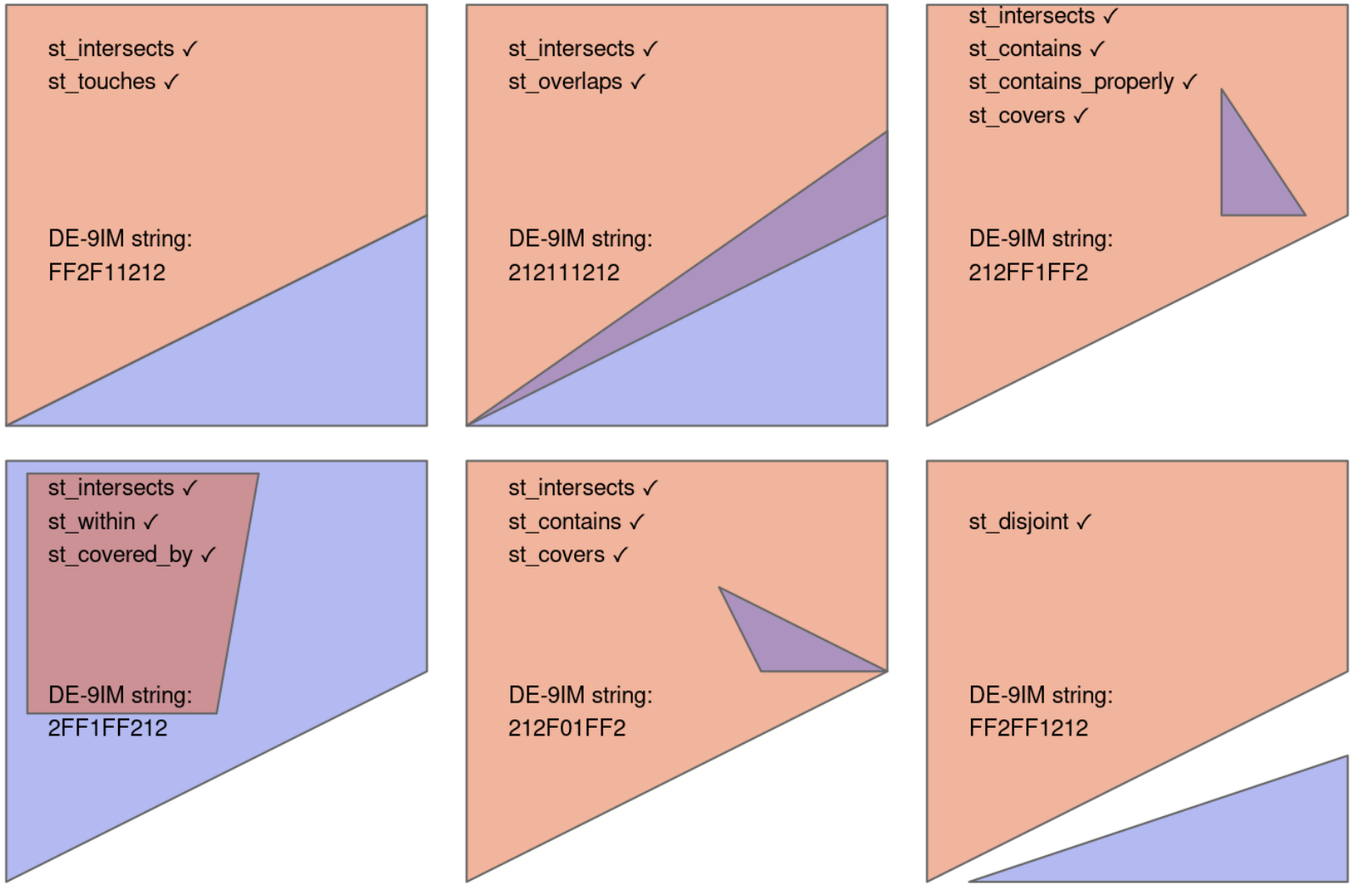
Attribute data operations: aggregating (based on attributes)





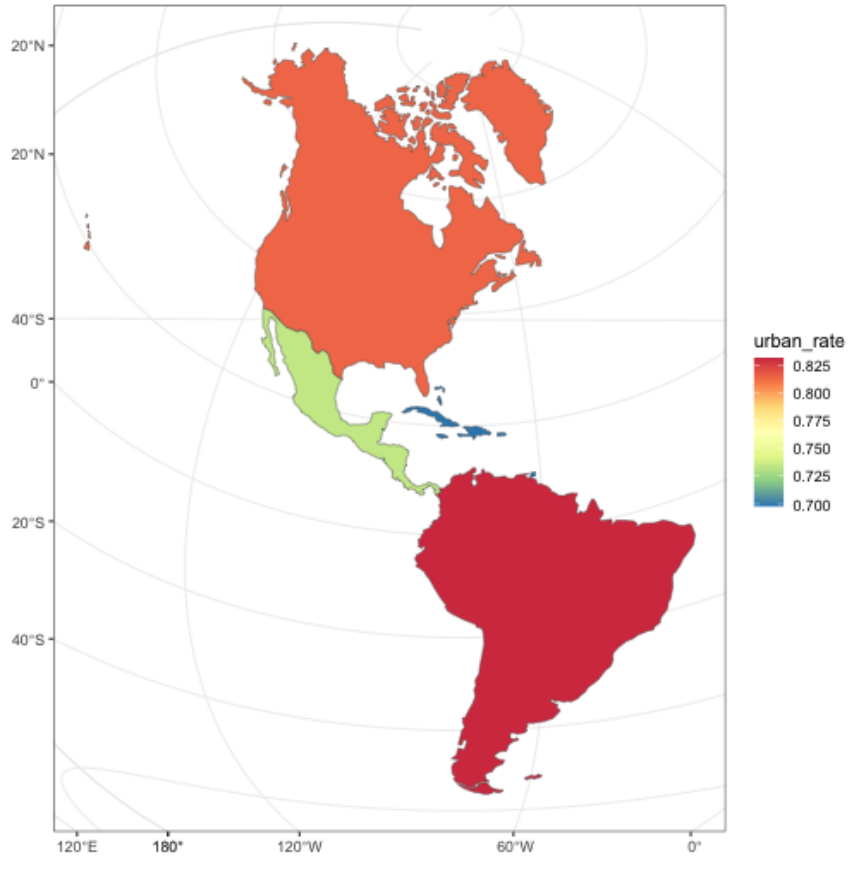
Spatial data operations

- Same intution, but now spatial aspects determine the operations
  - Before: based on the underlying attributes
- Spatial relationship of `sf` objects: determined by different **topological relations**
  - Examples: intersection, containing, touching, etc.
- Intuition (and workflow with data): the same as with attribute data
- **Detailed exposition:** on class material `01_class03.R`
- Next: types of topological relationships


st_intersects ✓
st_touches ✓
DE-9IM string:
FF2F11212
st_intersects ✓
st_overlaps ✓
DE-9IM string:
212111212
st_intersects ✓
st_contains ✓
st_contains_properly ✓
st_covers ✓
DE-9IM string:
212FF1FF2
st_intersects ✓
st_within ✓
st_covered_by ✓
DE-9IM string:
2FF1FF212
st_intersects ✓
st_contains ✓
st_covers ✓
DE-9IM string:
212F01FF2
st_disjoint ✓
DE-9IM string:
FF2FF1212

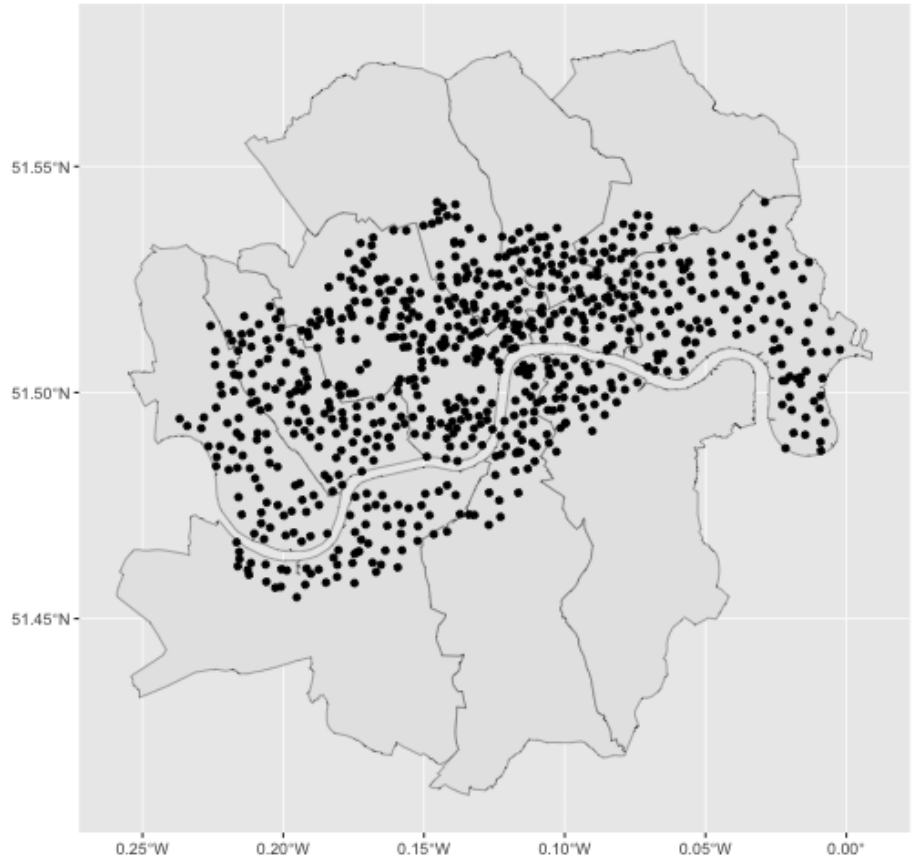
Hands-in: your turn! (1/2)

- Combine `world` (`sf`) and `worldbank_df` (`data.frame`)
- Filter only countries in America
- Calculate urban rate by `subregion`
- urban rate = urban population/total population
- Plot of Americas by subregions' urban rates:



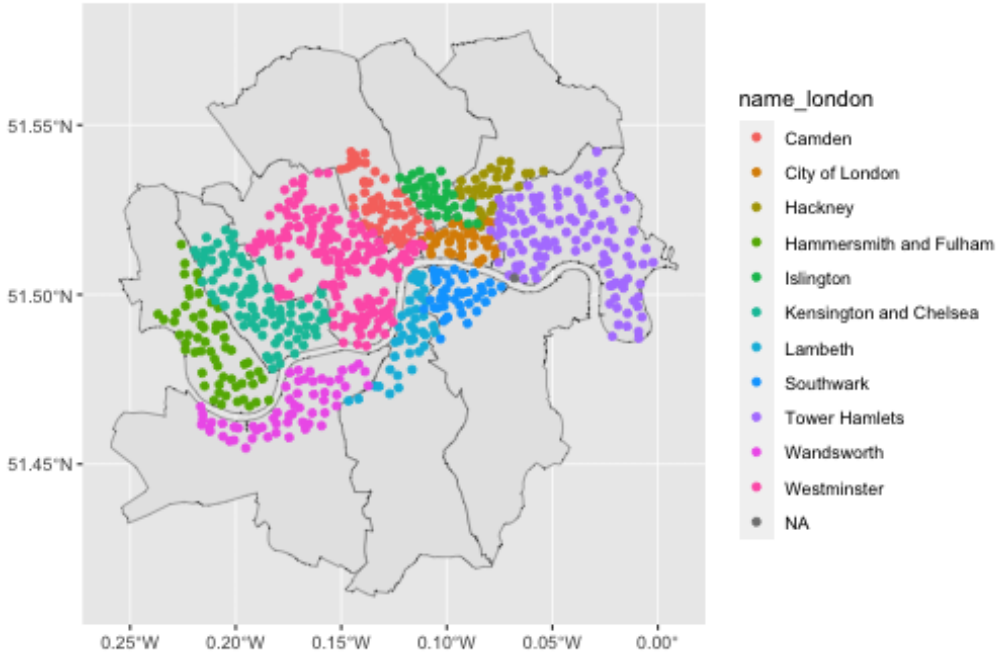
Hands-in: your turn! (2/2)

- Combine `lnd` (Great London) and `cycle_hire` (location of bike stations)
- Filter London regions with bike stations, **plot the two together**
- Join both datasets, plot bike stations by London neighborhood
- Aggregate datasets, plot London neighborhoods by number of bikes



Hands-in: your turn! (2/2)

- Combine `lnd` (Great London) and `cycle_hire` (location of bike stations)
- Filter London regions with bike stations, plot the two together
- Join both datasets, **plot bike stations by neighborhood**
- Aggregate datasets, plot London neighborhoods by number of bikes



Hands-in: your turn! (2/2)

- Combine `lnd` (Great London) and `cycle_hire` (location of bike stations)
- Filter London regions with bike stations, plot the two together
- Join both datasets, plot bike stations by London neighborhood
- Aggregate datasets, plot **London neighborhoods by number of bikes**