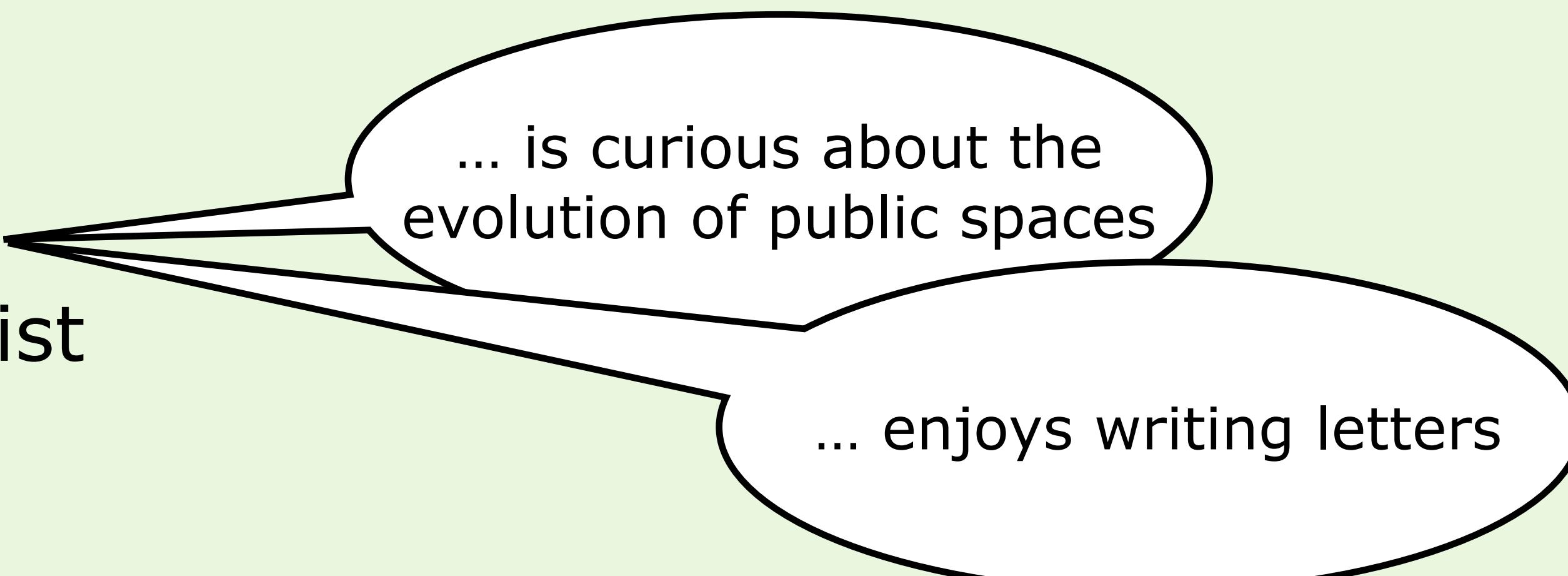


Where have all the post offices gone?

Discovering neighborhood resources with Python and OpenStreetMap

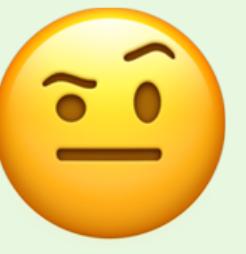
Katie Richardson
Staff Data Scientist
@ Blue Yonder



... is curious about the evolution of public spaces

... enjoys writing letters

It all started when ...

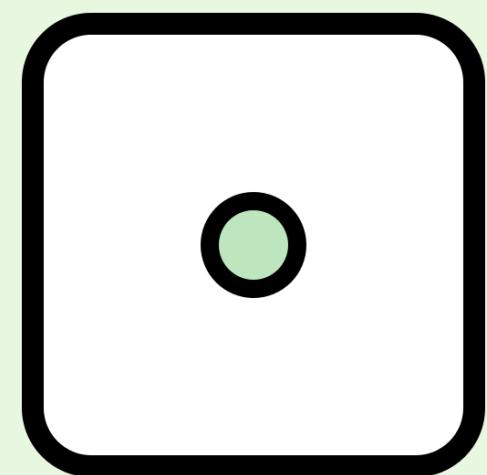


OpenStreetMap

- Database
- Open and free to use
- Community-driven
- Supported by OpenStreetMap Foundation

OSM Elements

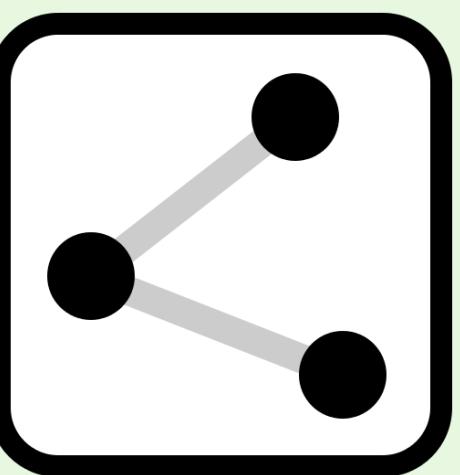
Node



Single point

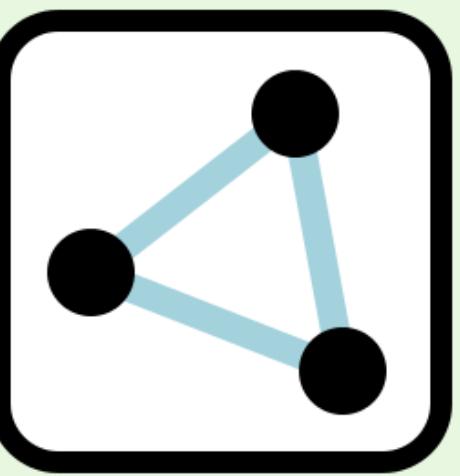
Has coordinates

Way



Open

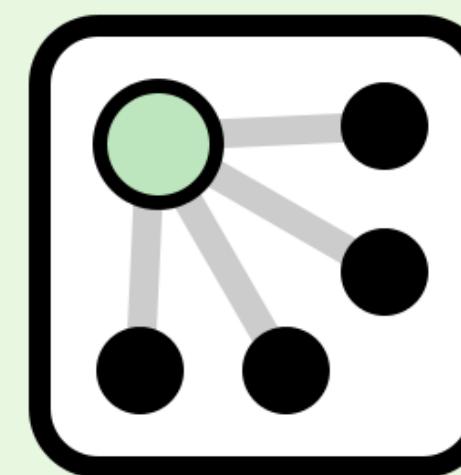
first != last



Closed

first == last

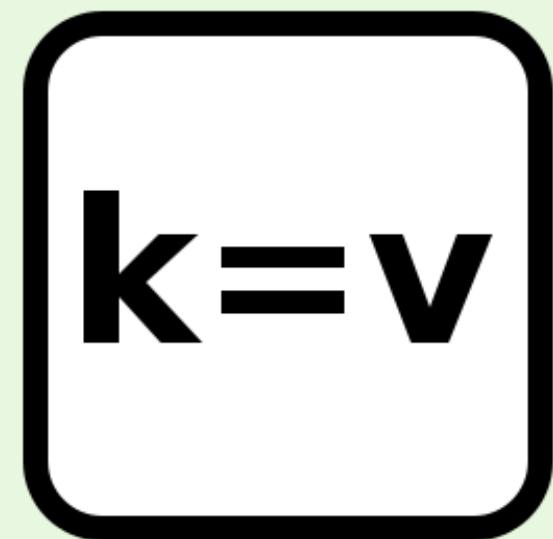
Relation



Collection of elements

Describes logical / geographical relationships

Tags



Describe features of elements

Meaning & usage are documented in wiki pages

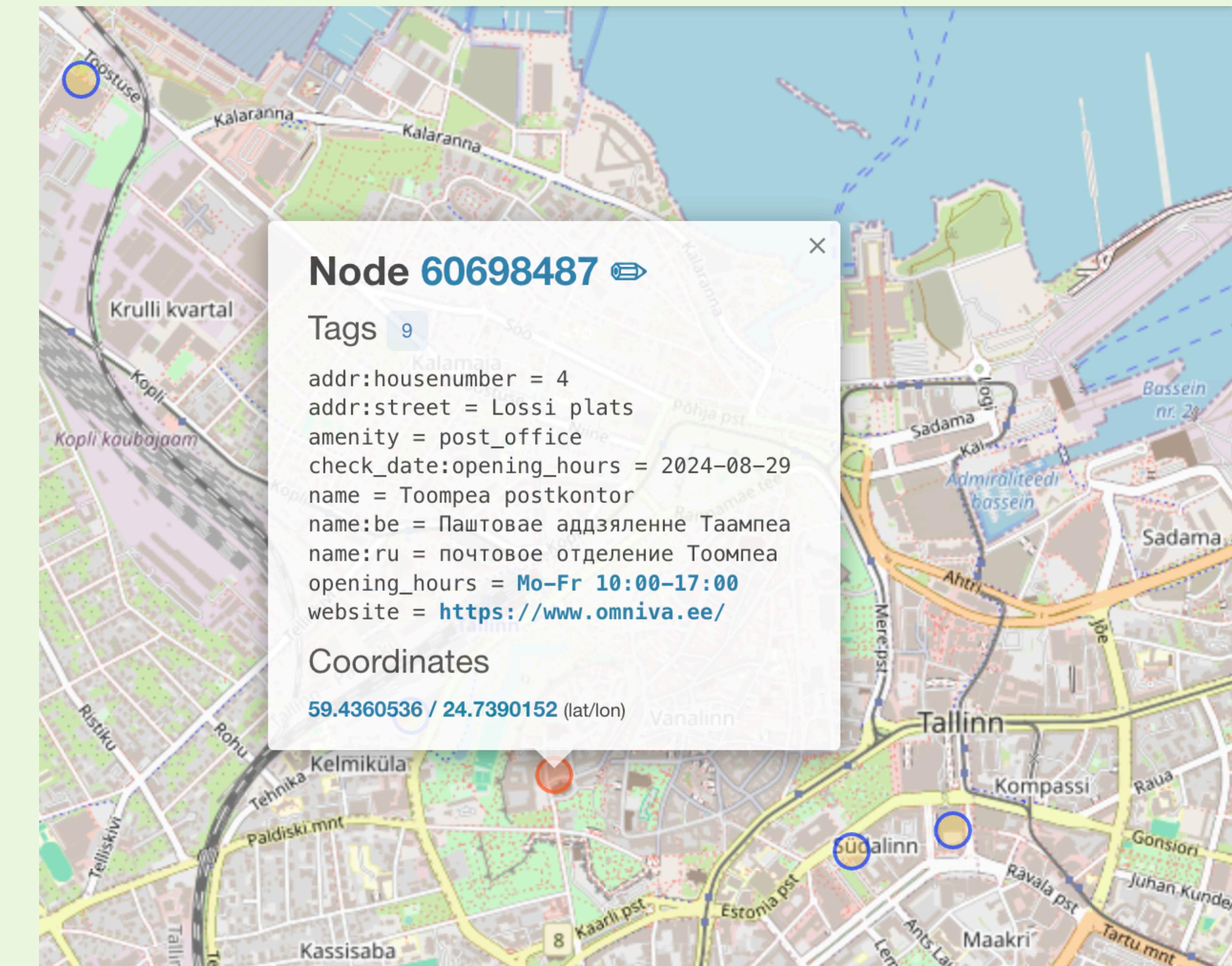
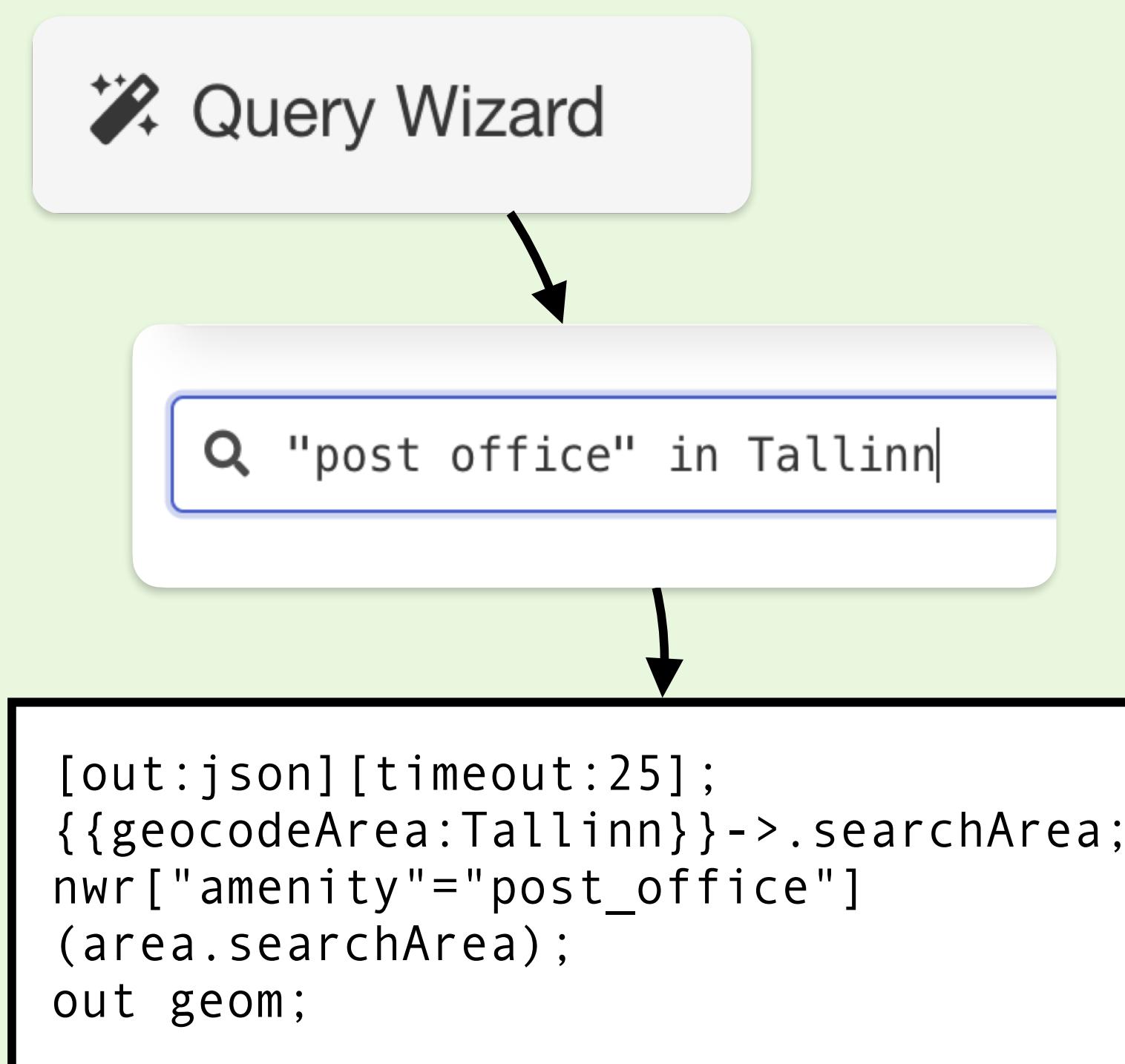
Tag:boundary=administrative

Country (<code>admin_level=2</code>)	6	7	8	9	10
Estonia (proposed)	Countys (<i>maakonnad</i>)	Municipalities (<i>vallad, omavalitsuslikud linnad</i>)	Tallinn and Kohtla-Järve (as settlement units)	Settlement units (<i>asustusüksused: külad, alevikud, alevid, vallasised linnad</i>), city districts (<i>linnaosad</i> , in Tallinn and Kohtla-Järve)	Subdistricts (<i>asumid</i> , in Tallinn)

<https://wiki.openstreetmap.org/wiki/Tag:boundary=administrative>

How are post offices tagged?

- Overpass API
Query up-to-date database copy
- Overpass Turbo



Getting the data

Geofabrik

- Data extracts for sub regions
- Latest & older data files
- Multiple data formats

```
# latest  
https://download.geofabrik.de/europe/estonia-latest.osm.pbf
```

```
# 01-01-2018  
https://download.geofabrik.de/europe/estonia-180101.osm.pbf
```

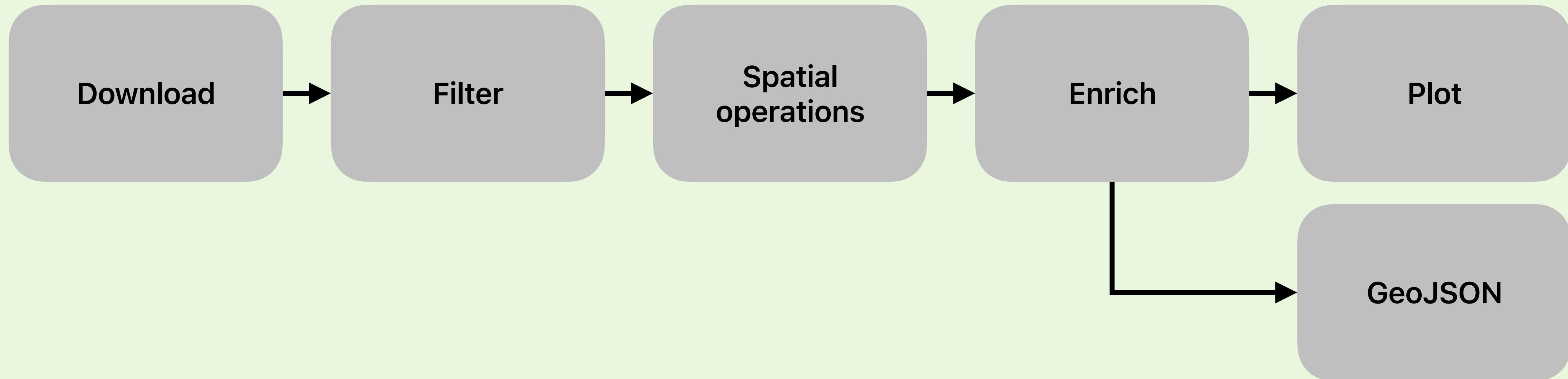


What are .osm.pbf files?

<https://www.geofabrik.de>

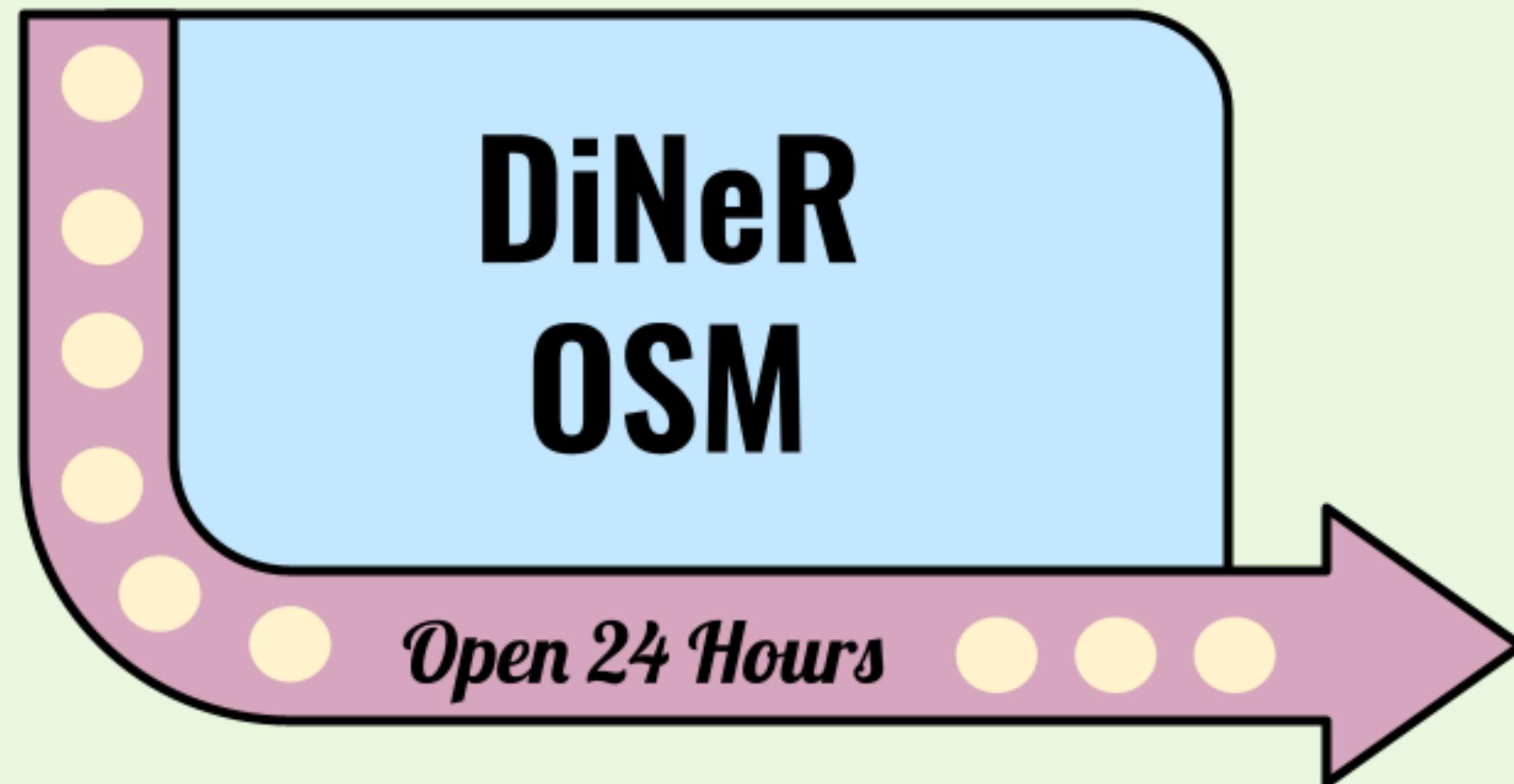
Organizing the project

Goal: Use geospatial data to learn about the distribution of neighborhood resources and how it has changed over time.



... and I want a catchy name

DIscovering **N**Eighborhood Resources with **O**pen**S**treet**M**ap



```
[server]
url = "https://download.geofabrik.de"

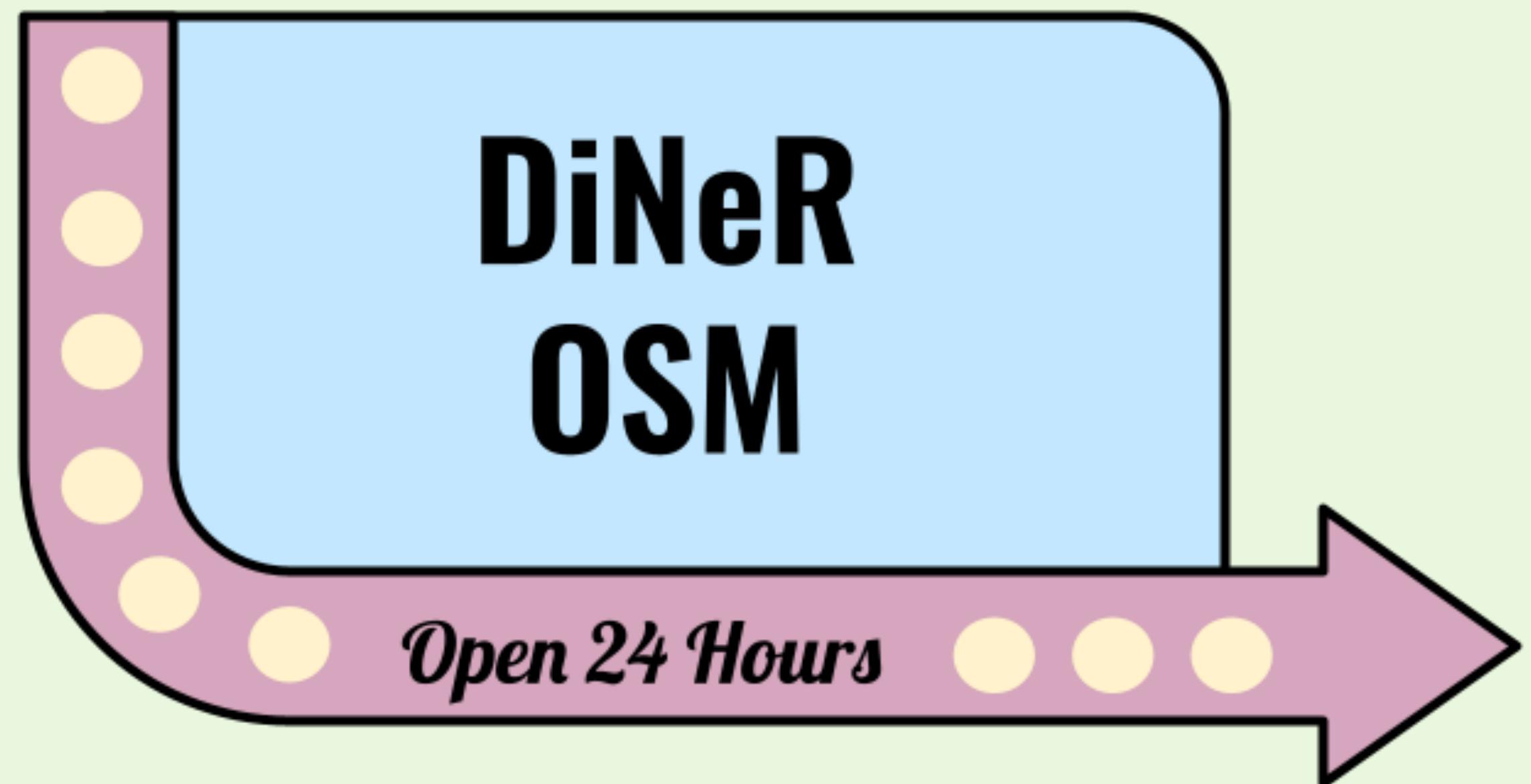
[regions]
tallinn = "europe/estonia"

[versions]
2019 = "190101.osm.pbf"
2023 = "230101.osm.pbf"
latest = "latest.osm.pbf"
```

```
diner-osm visualize --region tallinn --versions latest 2023 2019
```

... and I want a catchy name

DIscoving NEighborhood Resources with OpenStreetMap



```
[tallinn.areas]  
entity = "area"  
tags.boundary = "administrative"  
tags.admin_level = "9"
```

```
[tallinn.places]  
entity = "node"  
keys = ["name"]  
tags.amenity = "post_office"
```

```
diner-osm visualize --region tallinn --versions latest 2023 2019
```

Get neighborhoods

pyosmium

- Read file
- Filter the data
- Enhance with geometries
- Add attributes

```
fp_areas = (
    osmium.FileProcessor("estonia-latest.osm.pbf")
    .with_areas()
    .with_filter(EntityFilter(osmium.osm.AREA))
    .with_filter(TagFilter(("boundary", "administrative")))
    .with_filter(TagFilter(("admin_level", "9")))
    .with_filter(GeoInterfaceFilter())
    .with_filter(EnrichAttributes())
)
```

Get post offices

pyosmium

- Read file
- Filter the data
- Enhance with geometries
- Add attributes

```
fp_places = (
    osmium.FileProcessor("estonia-latest.osm.pbf")
    .with_areas()
    .with_filter(EntityFilter(osmium.osm.NODE))
    .with_filter(KeyFilter(("name")))
    .with_filter(TagFilter(("amenity", "post_office")))
    .with_filter(GeoInterfaceFilter())
    .with_filter(EnrichAttributes())
)
```

Custom Filter

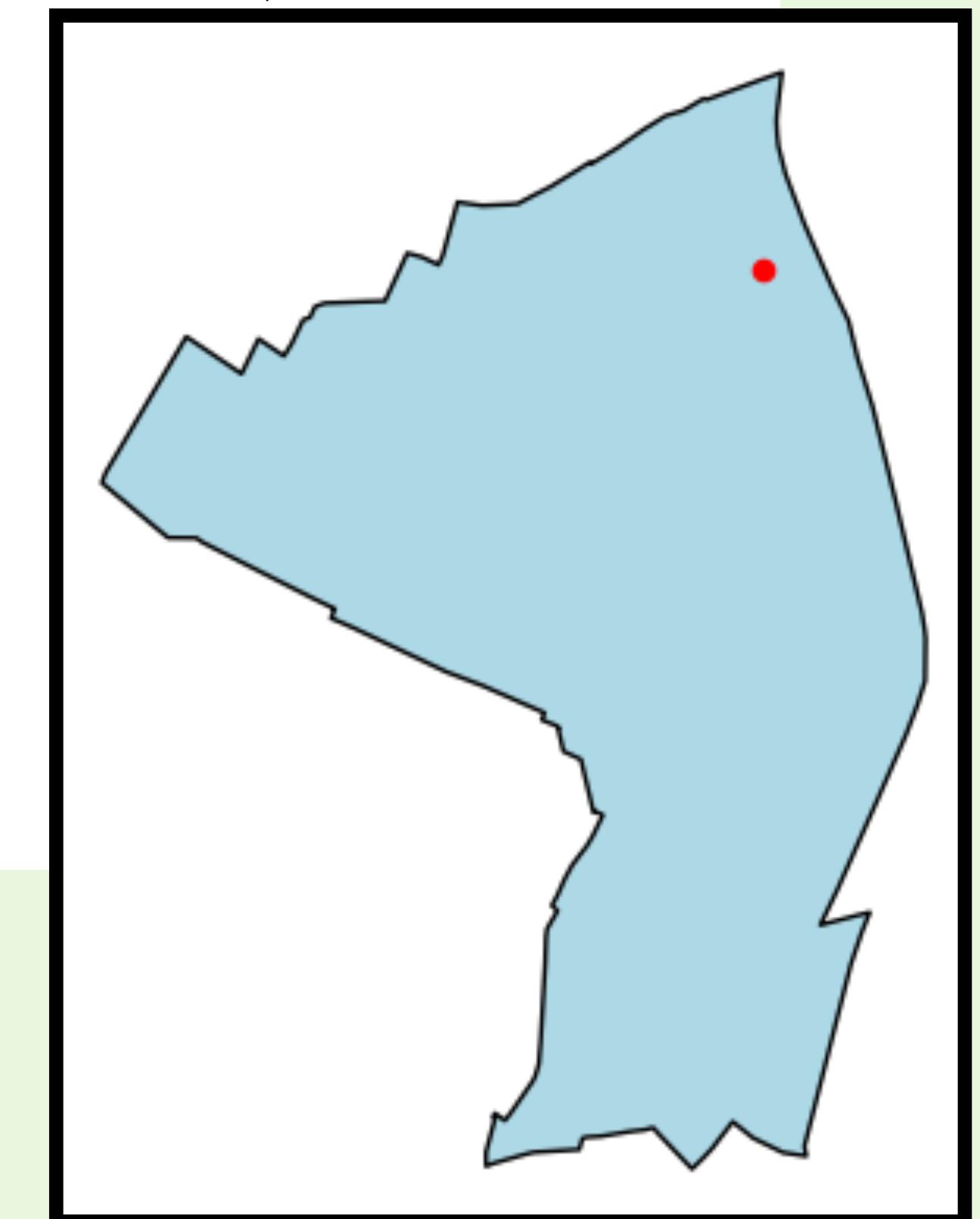
In which neighborhood are post offices located?

GeoPandas

- From feature iterable
- Spatial joins

```
gdf_areas = GeoDataFrame.from_features(fp_areas)
gdf_places = GeoDataFrame.from_features(fp_places)
```

```
gdf = gdf_areas.sjoin(
    df=gdf_places,
    how="left",
    predicate="contains",
    lsuffix="area",
    rsuffix="place",
)
```



Transform the data

GeoPandas

- Clip by geometry
- Calculate area

```
# bbox clip mask  
gdf.clip([24.72, 59.42, 24.78, 59.44])  
  
# GeoDataFrame clip mask  
gdf.clip(tallinn_area_gdf)  
  
gdf[ "sqkm" ] = gdf.set_crs(epsg=4326) \  
    .to_crs(epsg=32633) \  
    .geometry.area / 1_000_000
```

Transform the data

GeoPandas

- Clip by geometry
- Calculate area

geometry	name	wikidata	id	osm_url	count
MULTIPOLYGON (((24.7488 59.3764, 24.74818 59.3...))	Kesklinna linnaosa	Q1230929	r351870	https://www.osm.org/relation/351870	3
MULTIPOLYGON (((24.64588 59.46923, 24.64588 59...))	Põhja-Tallinna linnaosa	Q1632061	r350769	https://www.osm.org/relation/350769	2
POLYGON ((24.81569 59.41122, 24.81493 59.41153...))	Lasnamäe linnaosa	Q1017526	r355154	https://www.osm.org/relation/355154	2
POLYGON ((24.67393 59.41931, 24.6799 59.42443,...))	Kristiine linnaosa	Q999834	r353790	https://www.osm.org/relation/353790	1
POLYGON ((24.63976 59.3953, 24.63989 59.39533,...))	Mustamäe linnaosa	Q1632467	r350487	https://www.osm.org/relation/350487	1

Interactive Visualisations

Bokeh

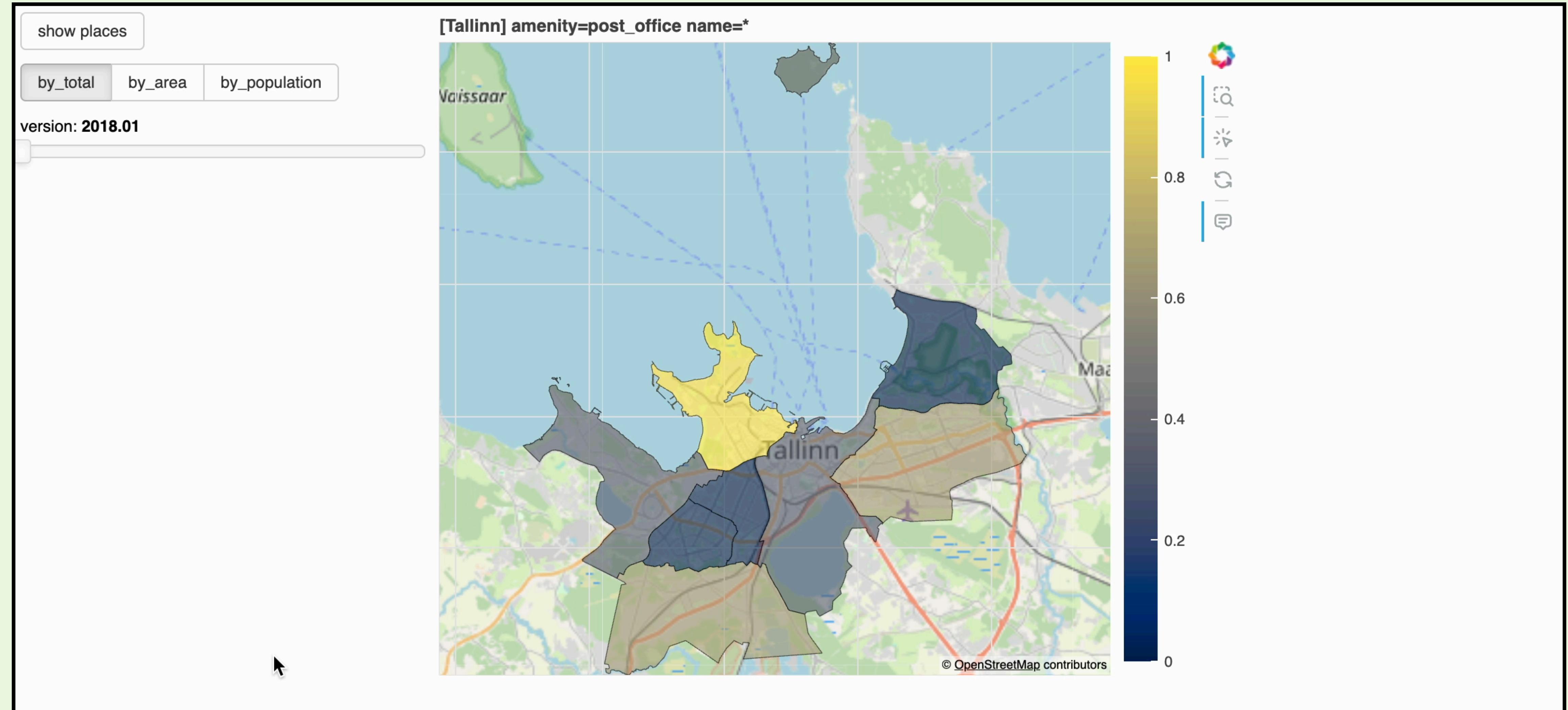
- GeoJSONDataSource
- CustomJS callbacks
- Hover & Tap tools
- Toggles & Sliders

```
areas_data = GeoJSONDataSource(geojson=gdf.to_json())

plot = figure()

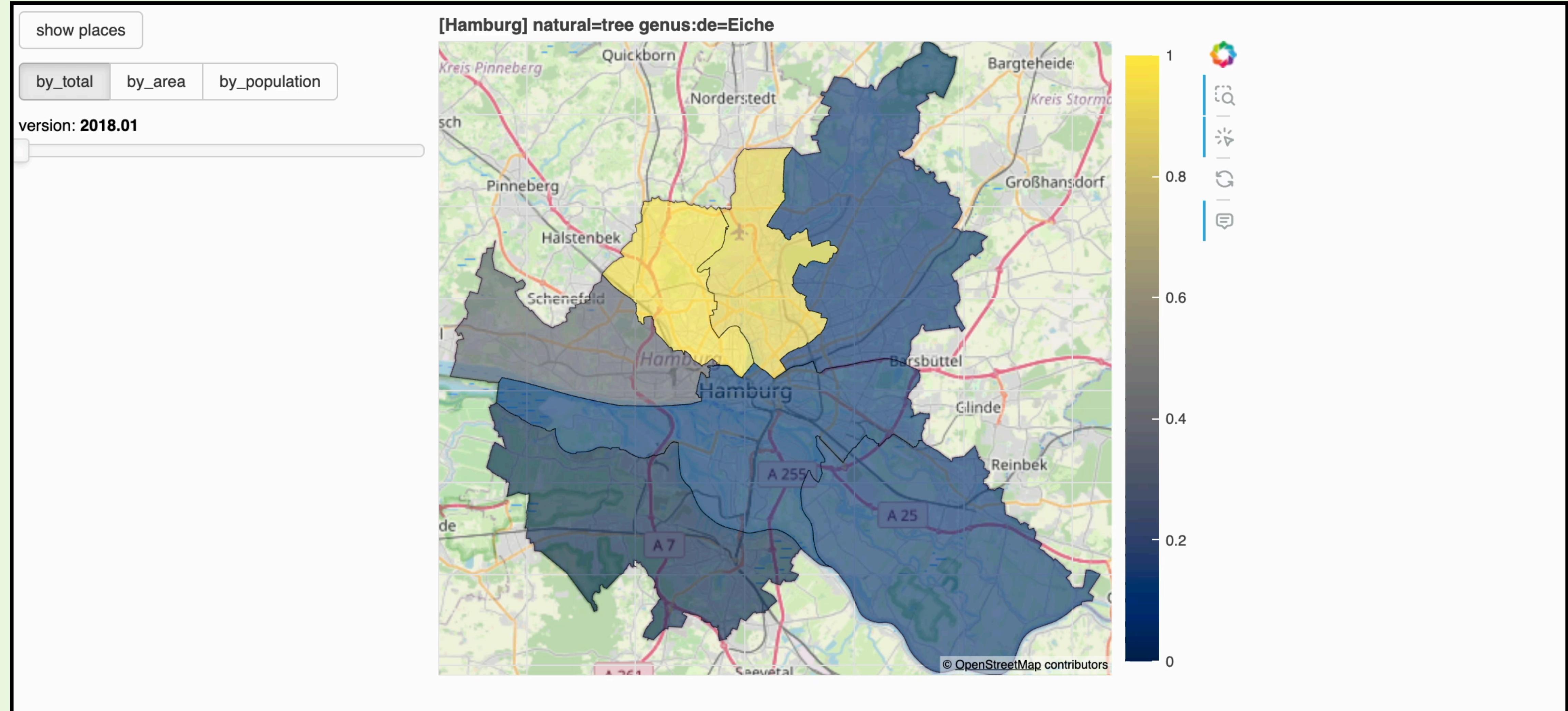
areas = plot.patches(source=areas_data)

show(plot)
```



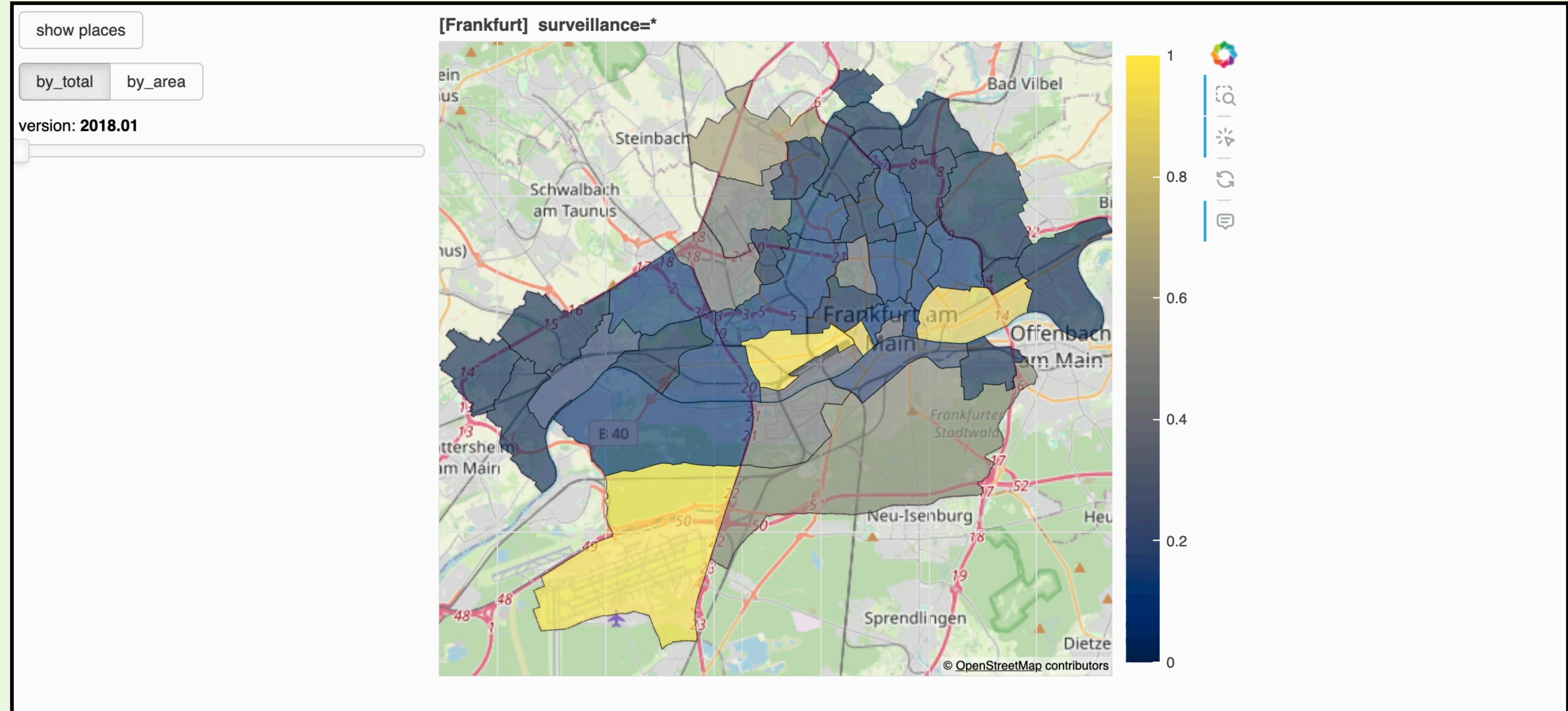
Post offices in Tallinn

Map data from OpenStreetMap



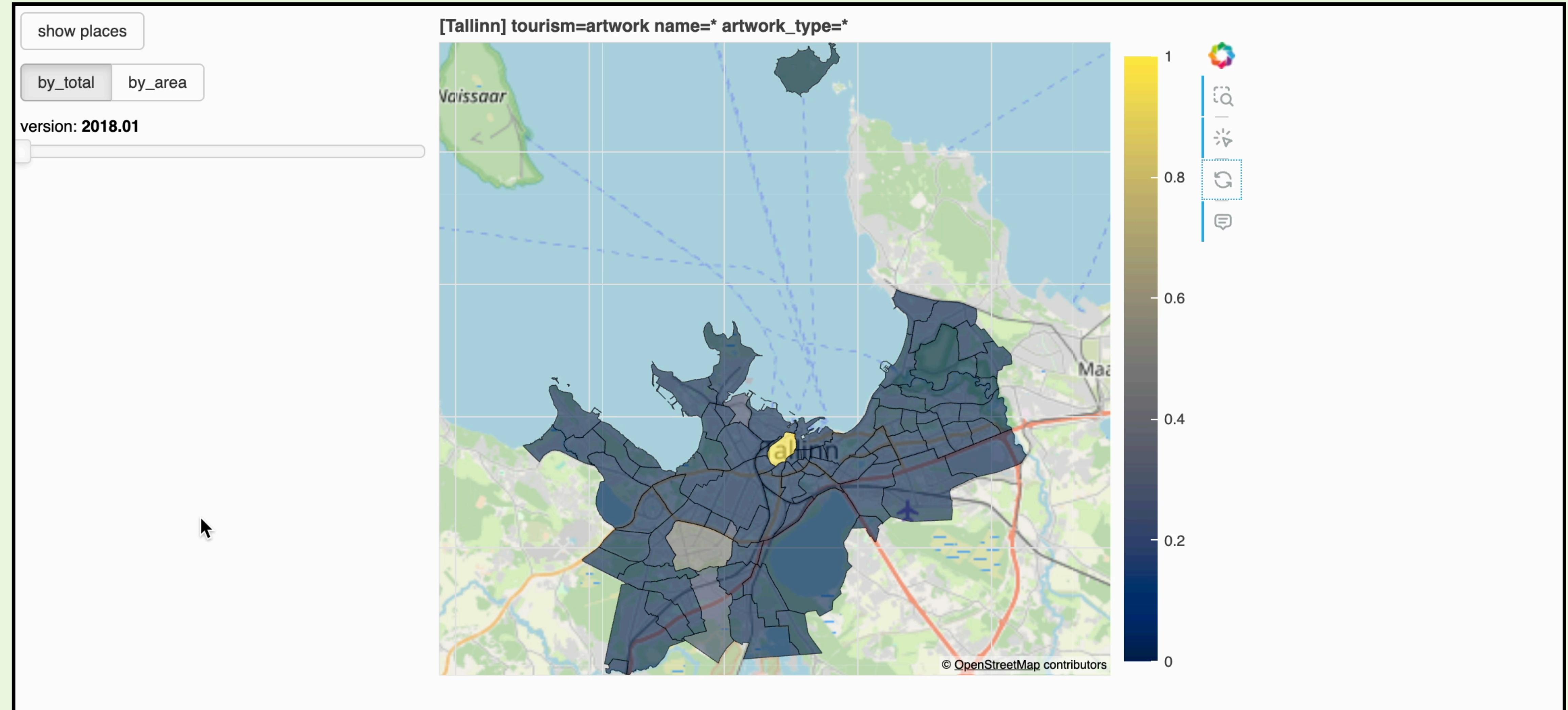
Oak trees in Hamburg

Map data from OpenStreetMap



Surveillance in Frankfurt

Map data from OpenStreetMap



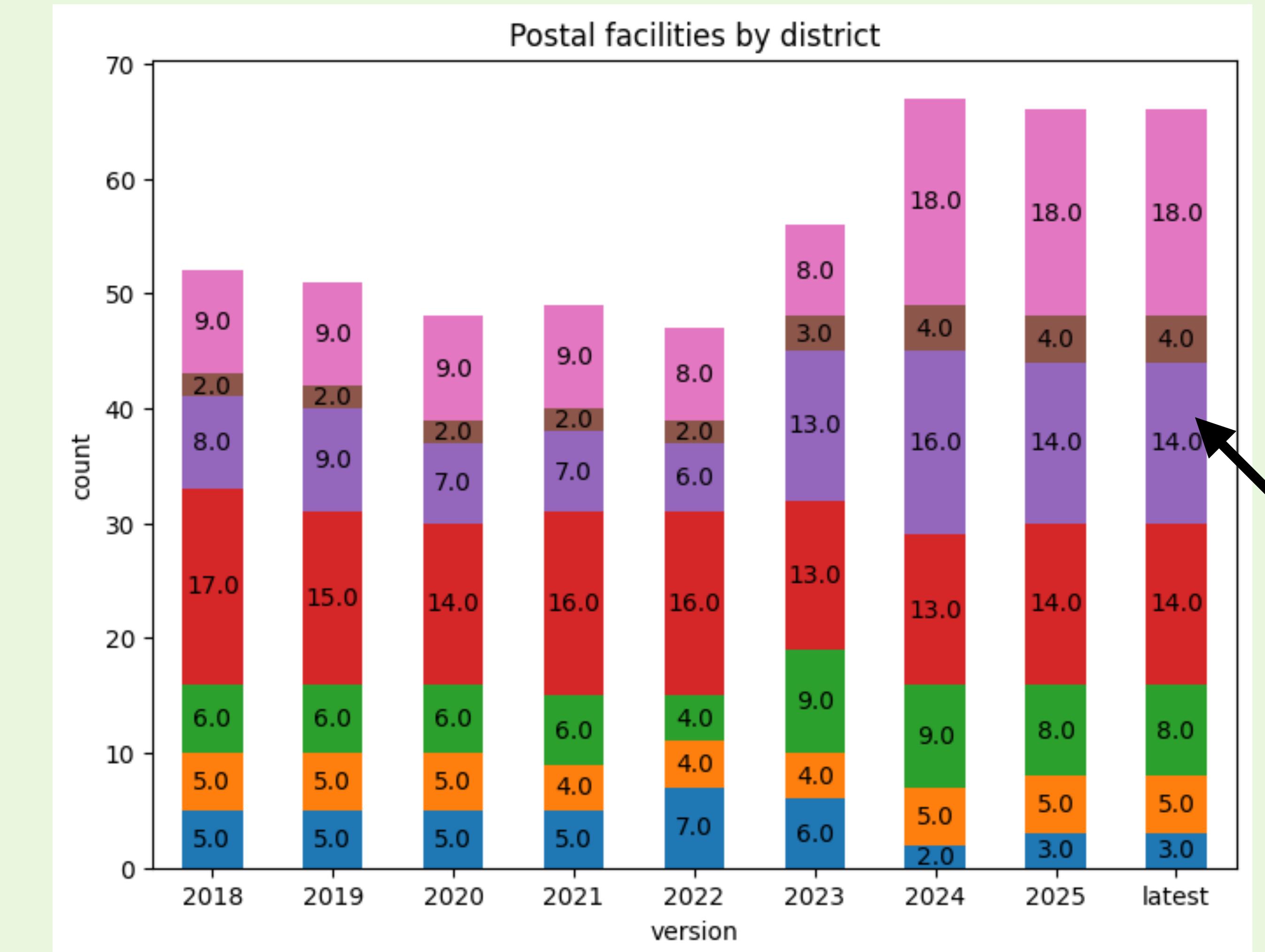
Public artwork in Tallinn

Map data from OpenStreetMap

So where did the post offices go?

Was my experience reflected in the data?

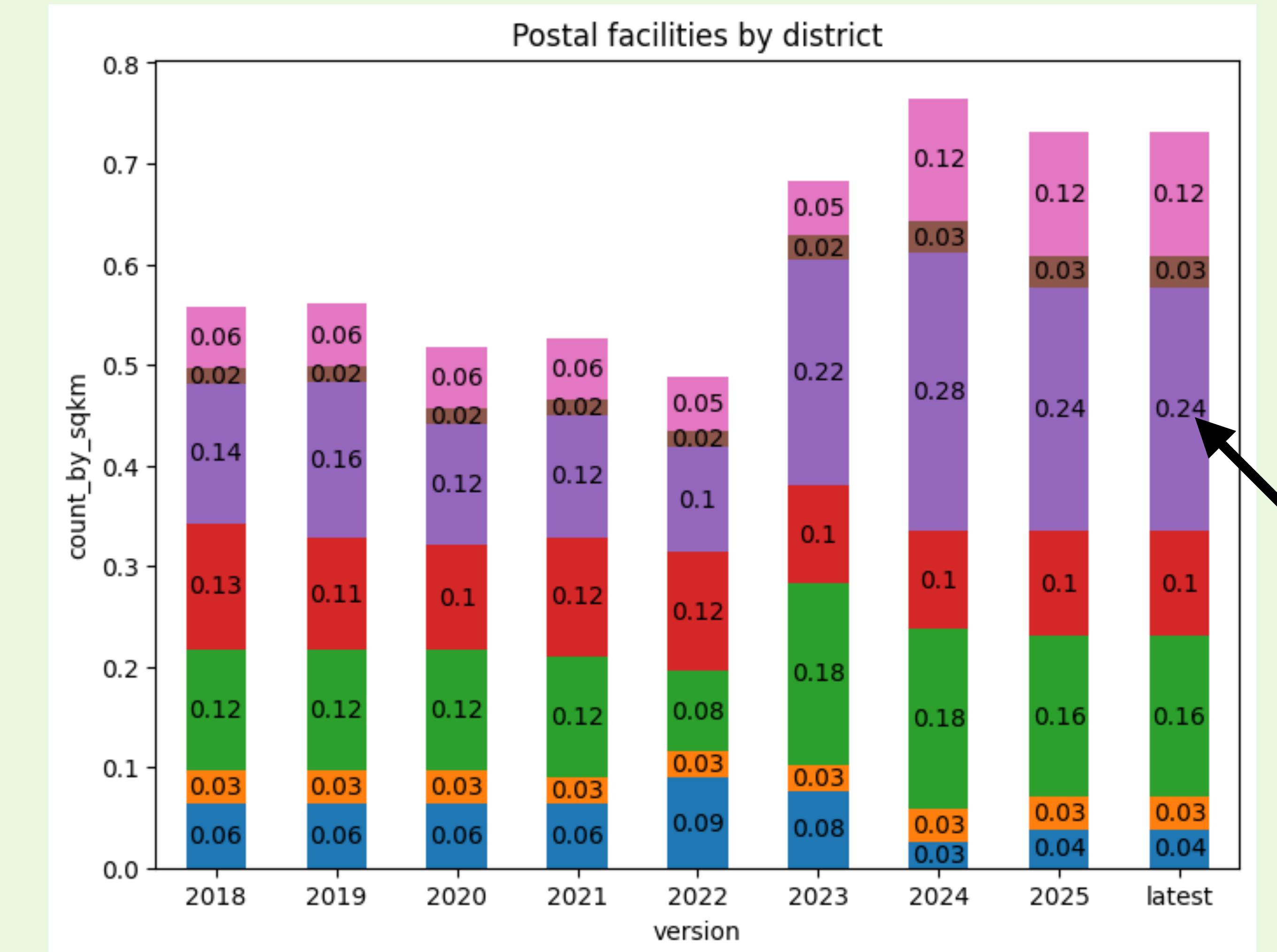
Was I asking the correct question?



So where did the post offices go?

Was my experience reflected in the data?

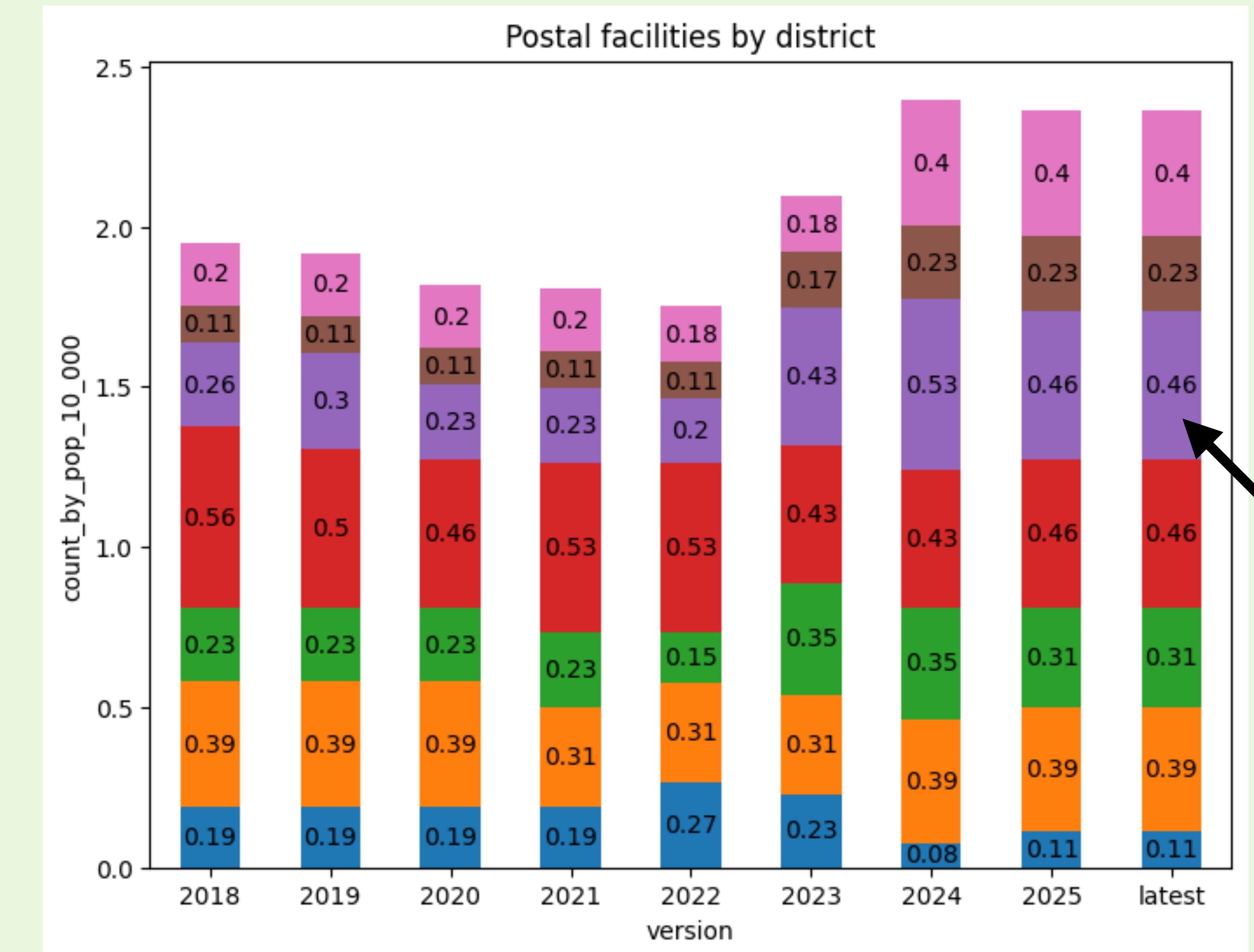
Was I asking the correct question?



So where did the post offices go?

Was my experience reflected in the data?

Was I asking the correct question?



Go discover your neighborhood!



KatieBSC / **diner-osm**

