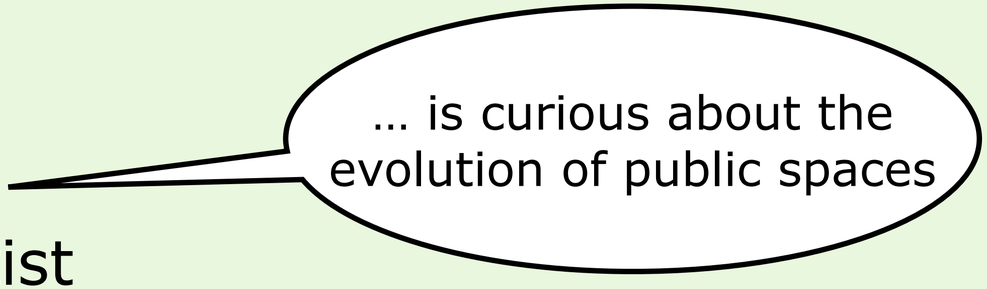
Where have all the post offices gone?

Discovering neighborhood resources with Python and OpenStreetMap

Katie Richardson
Staff Data Scientist
@ Blue Yonder

Where have all the post offices gone?

Discovering neighborhood resources with Python and OpenStreetMap

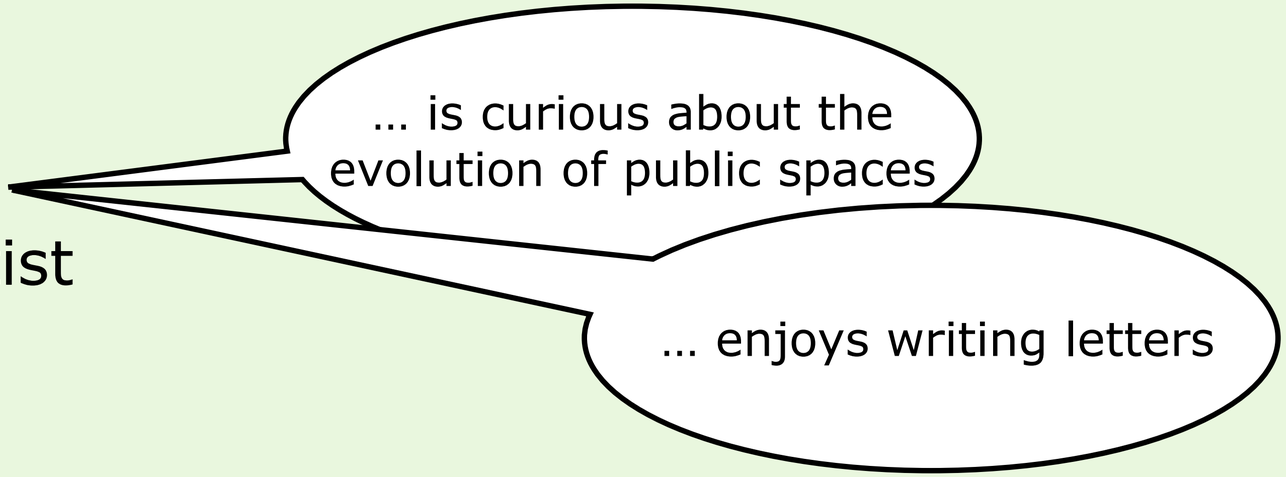


Katie Richardson
Staff Data Scientist
@ Blue Yonder

Where have all the post offices gone?

Discovering neighborhood resources with Python and OpenStreetMap

Katie Richardson
Staff Data Scientist
@ Blue Yonder

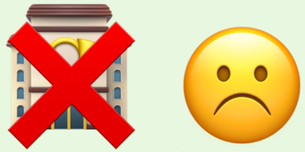


It all started when …

It all started when …

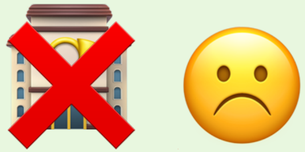
It all started when …


It all started when …

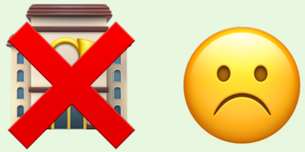


It all started when …

It all started when …



It all started when …



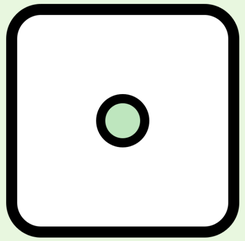
OpenStreetMap

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

OSM Elements

OSM Elements

Node

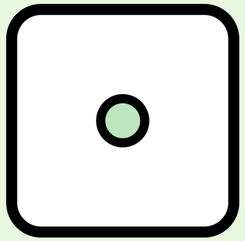


Single point

Has coordinates

OSM Elements

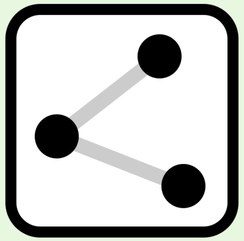
Node



Single point

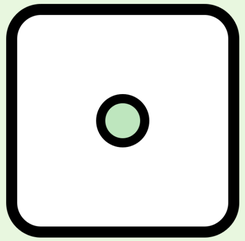
Has coordinates

Way



OSM Elements

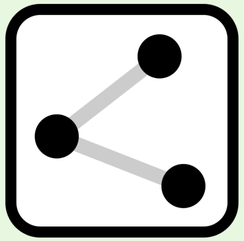
Node



Single point

Has coordinates

Way

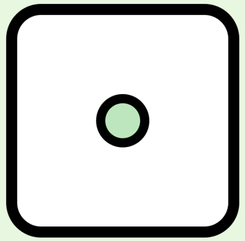


Open

first != last

OSM Elements

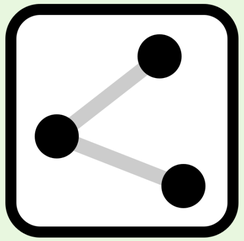
Node



Single point

Has coordinates

Way



Open

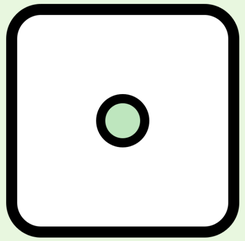
first != last

Closed

first = last

OSM Elements

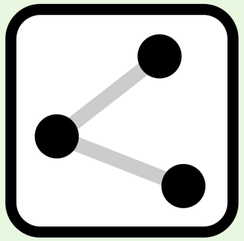
Node



Single point

Has coordinates

Way

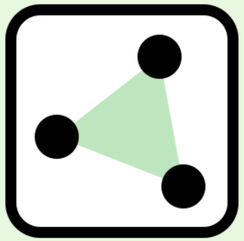


Open

first != last

Closed

first = last

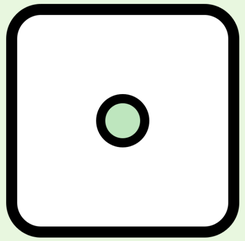


Area

Enclosed, filled

OSM Elements

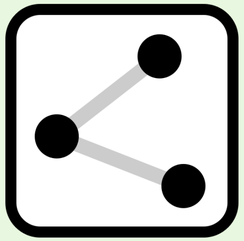
Node



Single point

Has coordinates

Way

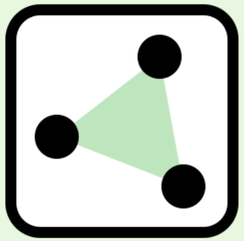


Open

first != last

Closed

first = last



Area

Enclosed, filled

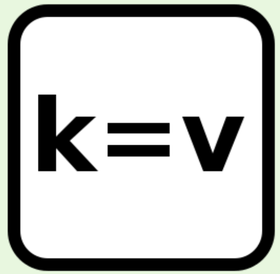
Relation

Collection of elements

Describes logical / geographical relationships

OSM Tags

Tags

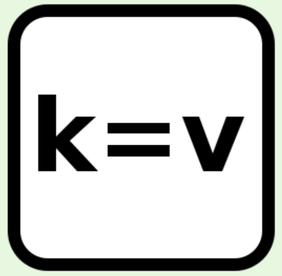


Describe features of elements

Meaning & usage are documented in wiki pages

OSM Tags

Tags



Describe features of elements

Meaning & usage are documented in wiki pages

Tag:boundary=administrative

Country (admin_level=2)	admin_level=*			
	8	9	10	11
new levels: **Germany** *see also Grenzen in Deutschland* *and Diskussionsseite / discussion page*	*Towns, Municipalities / City-districts* Stadt, Gemeinde LAU 2 (aka NUTS 5)	*Parts of a municipality with parish councils /self_government* Stadtbezirk / Gemeindeteil mit Selbstverwaltung	*Parts of a municipality without ...* Stadtteil / Gemeindeteil ohne Selbstverwaltung	*Neighbourhoods* statistical or historical Stadtviertel etc.

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

How are post offices tagged?

- Overpass API

  Query up-to-date database copy

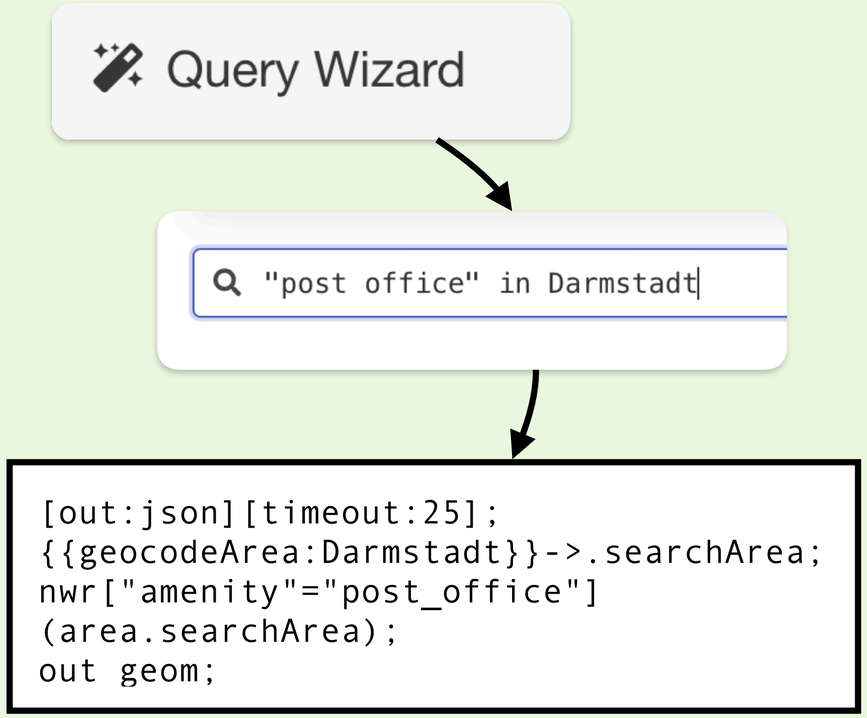
- Overpass Turbo

How are post offices tagged?

- Overpass API

  Query up-to-date database copy

- Overpass Turbo



How are post offices tagged?

- Overpass API

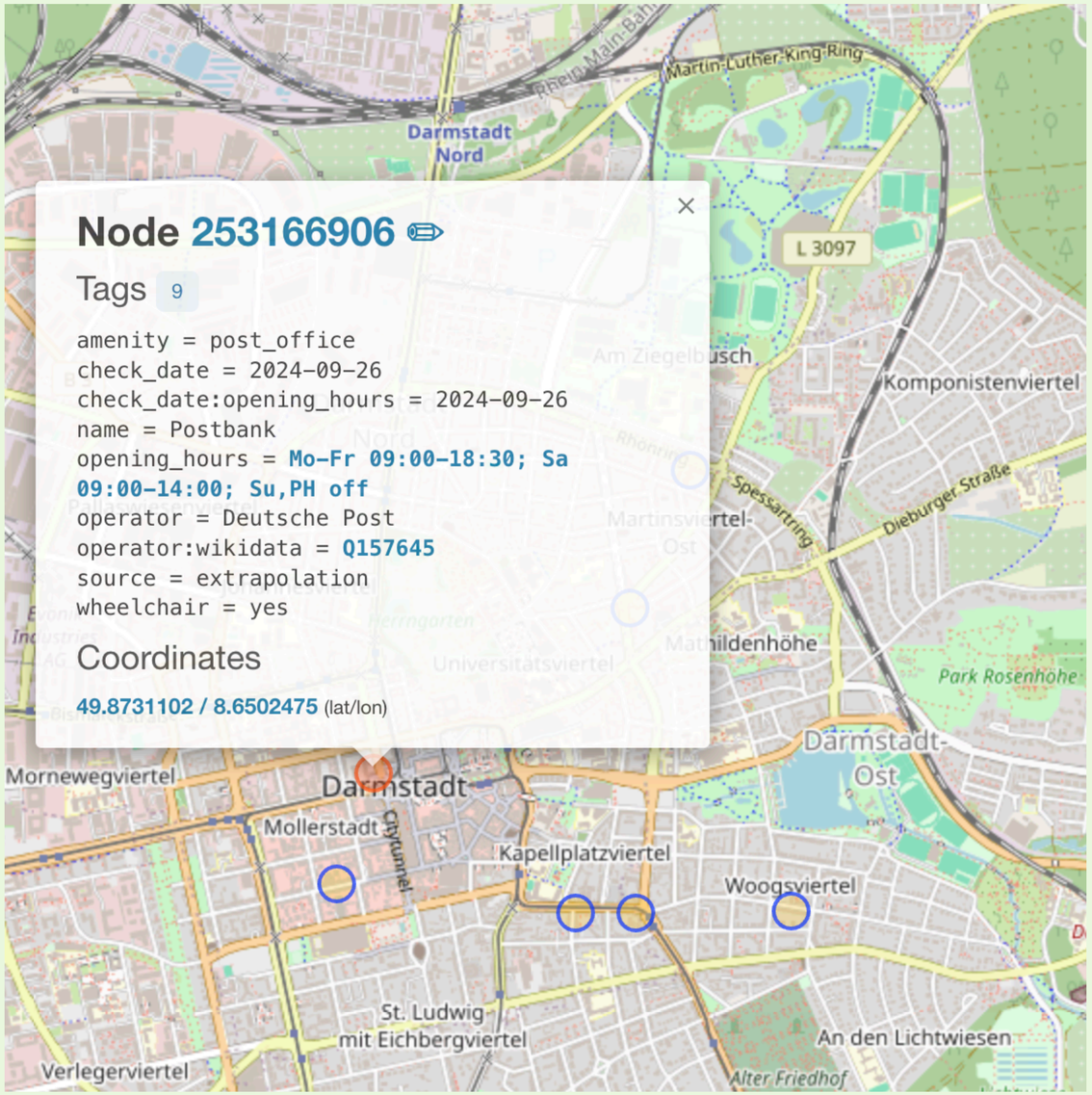
  Query up-to-date database copy

- Overpass Turbo

Query Wizard

"post office" in Darmstadt

```
[out:json][timeout:25];
{{geocodeArea:Darmstadt}}->.searchArea;
nwr["amenity"="post_office"]
(area.searchArea);
out geom;
```



Getting the data

Getting the data

Geofabrik

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

Getting the data

Geofabrik

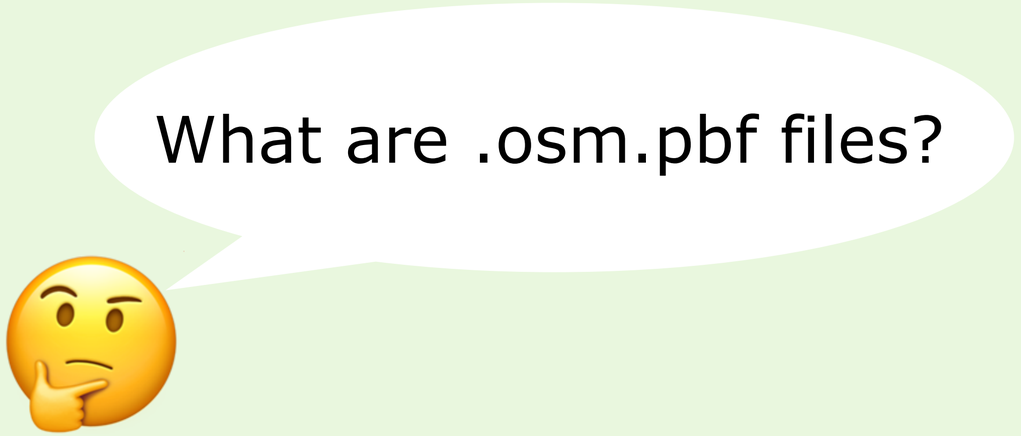
- Data extracts for sub regions
- Latest & older data files
- Multiple data formats

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

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

Getting the data

Geofabrik

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




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

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

Organizing the project

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

Organizing the project

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

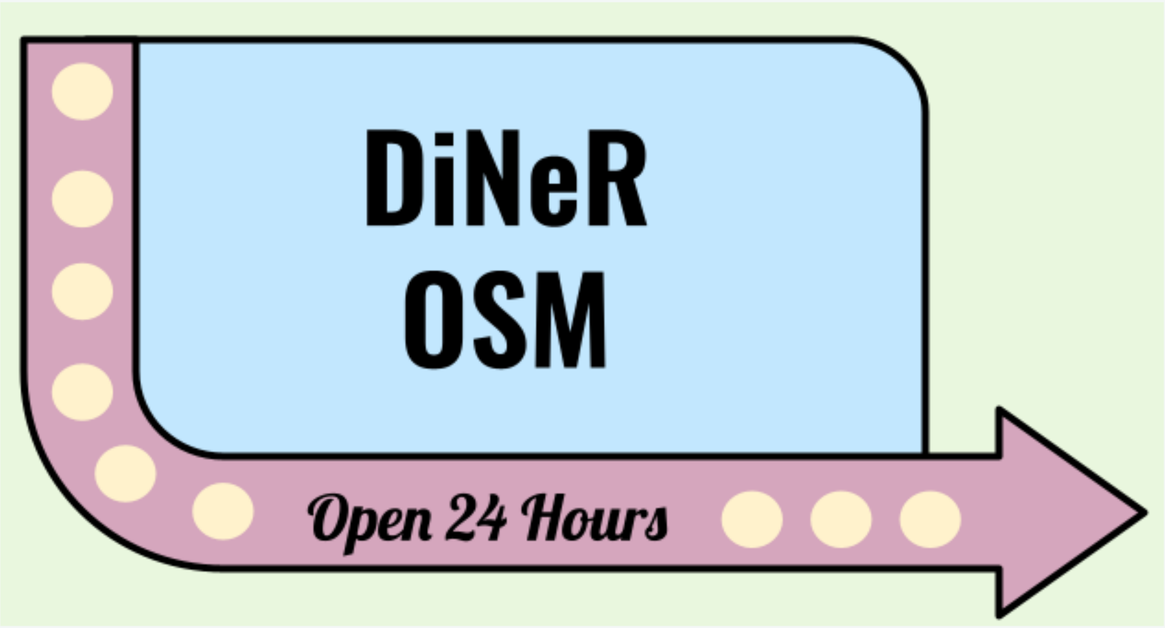
… and I want a catchy name

… and I want a catchy name

DIscovering **NE**ighborhood **R**esources with **O**pen**S**treet**M**ap

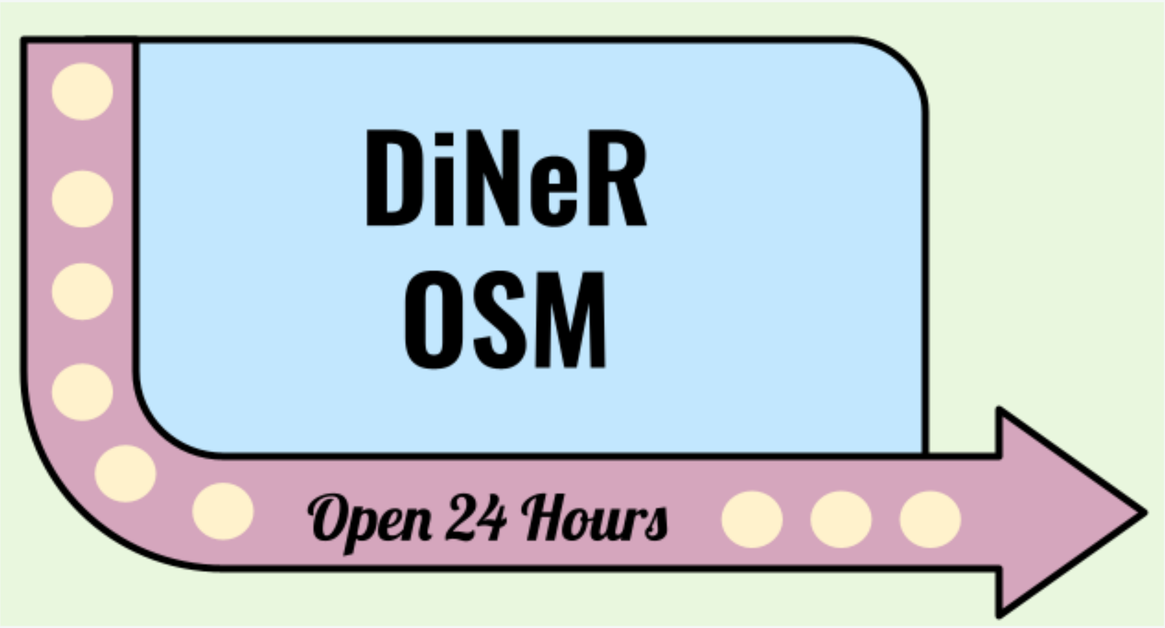
… and I want a catchy name

DIscovering **NE**ighborhood **R**esources with **O**pen**S**treet**M**ap



… and I want a catchy name

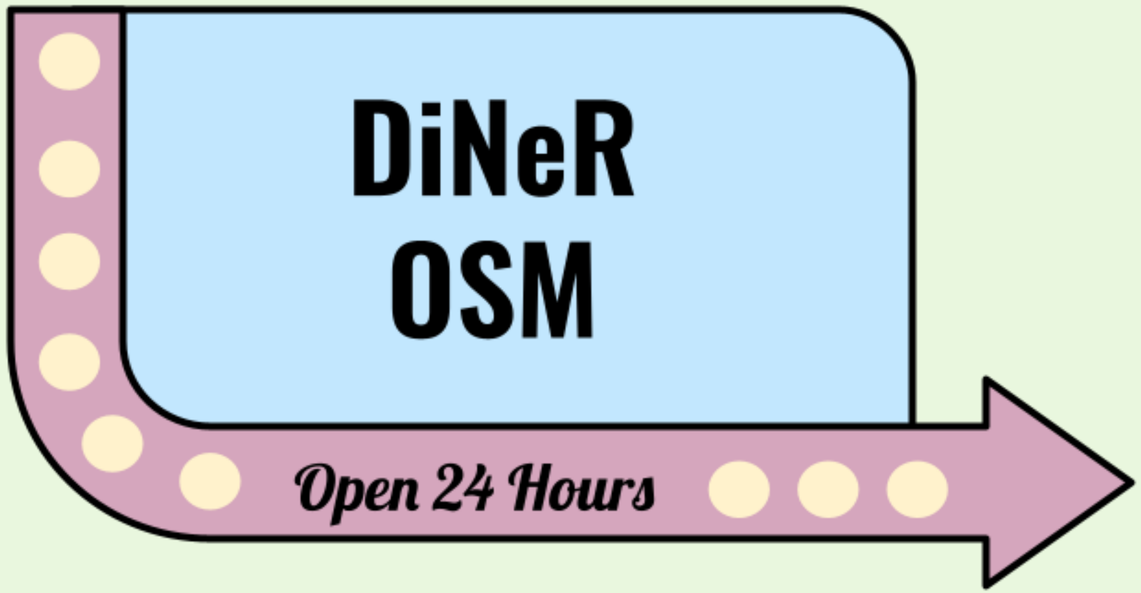
DIscovering **NE**ighborhood **R**esources with **O**pen**S**treet**M**ap




```
diner-osm –region darmstadt –versions latest 2023 2019
```

… and I want a catchy name

DIscovering **NE**ighborhood **R**esources with **O**pen**S**treet**M**ap

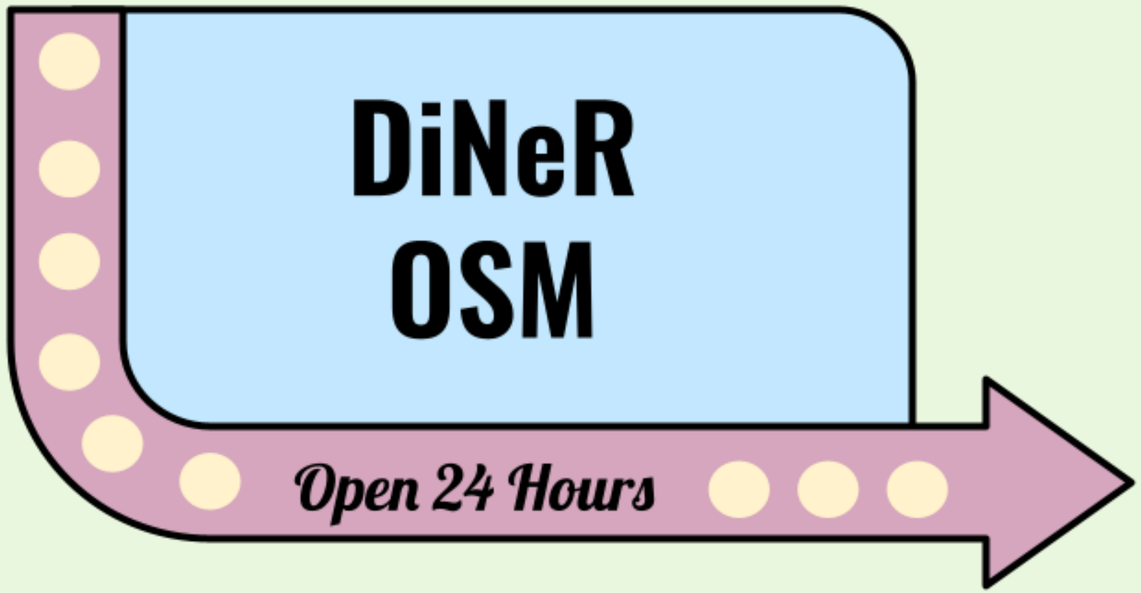



```
[server]
url = "https://download.geofabrik.de"

[regions]
darmstadt = "europe/germany/hessen"

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

```
diner-osm –region darmstadt –versions latest 2023 2019
```

… and I want a catchy name

DIscovering **NE**ighborhood **R**esources with **O**pen**S**treet**M**ap




```
[darmstadt.areas]
admin_level = "9"


[darmstadt.places]
entity = "node"
keys = ["name"]
tags.amenity = "post_office"
tags.operator = "Deutsche Post"
```

```
diner-osm –region darmstadt –versions latest 2023 2019
```

Get neighborhoods

pyosmium

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

Get neighborhoods

pyosmium

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

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

Get post offices

pyosmium

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

```
fp_nodes = (
    osmium.FileProcessor("hessen-latest.osm.pbf")
    .with_locations()
    .with_filter(EntityFilter(NODE))
    .with_filter(KeyFilter(("name")))
    .with_filter(TagFilter(("amenity", "post_office")))
    .with_filter(TagFilter(("operator", "Deutsche Post")))
    .with_filter(GeoInterfaceFilter(tags=tags_to_keep))
    .with_filter(EnrichAttributes())
)
```

Get post offices

pyosmium

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

```
fp_nodes = (
    osmium.FileProcessor("hessen-latest.osm.pbf")
    .with_locations()
    .with_filter(EntityFilter(NODE))
    .with_filter(KeyFilter(("name")))
    .with_filter(TagFilter(("amenity", "post_office")))
    .with_filter(TagFilter(("operator", "Deutsche Post")))
    .with_filter(GeoInterfaceFilter(tags=tags_to_keep))
    .with_filter(EnrichAttributes())
)
```

In which neighborhood are post offices located?

GeoPandas

- From feature iterable
- Spatial joins

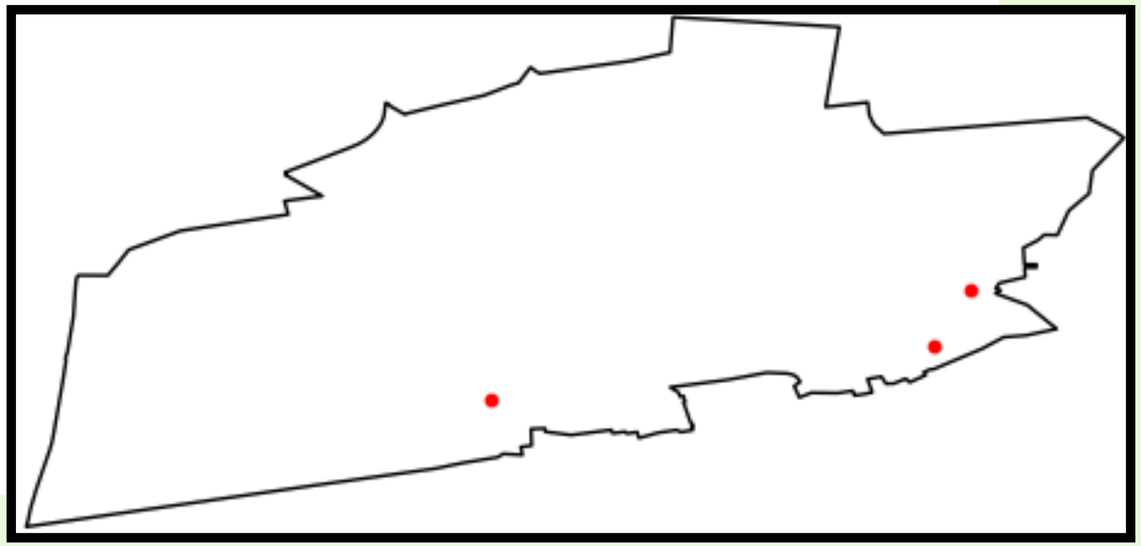
```
gdf_areas = GeoDataFrame.from_features(fp_areas)
gdf_nodes = GeoDataFrame.from_features(fp_nodes)

gdf = gdf_areas.sjoin(
        df=gdf_nodes,
        how="left",
        predicate="contains",
        lsuffix=“area",
        rsuffix="node",
)
```

In which neighborhood are post offices located?

GeoPandas

- From feature iterable
- Spatial joins

```
gdf_areas = GeoDataFrame.from_features(fp_areas)
gdf_nodes = GeoDataFrame.from_features(fp_nodes)

gdf = gdf_areas.sjoin(
        df=gdf_nodes,
        how="left",
        predicate="contains",
        lsuffix=“area",
        rsuffix="node",
)
```



Narrow scope & calculate area

GeoPandas

- Clip by geometry / query
- Get areas of neighborhoods

```
# bbox clip mask
gdf.clip([9.5, 10.4, 53.3, 53.75])

# GeoDataFrame clip mask
gdf.clip(cities_gdf.query("name == 'Darmstadt'"))

gdf["sqkm"] = gdf.set_crs(epsg=4326)\
                 .to_crs(epsg=25833)\
                 .geometry.area / 1_000_000
```

Narrow scope & calculate area

GeoPandas

- Clip by geometry / query
- Get areas of neighborhoods

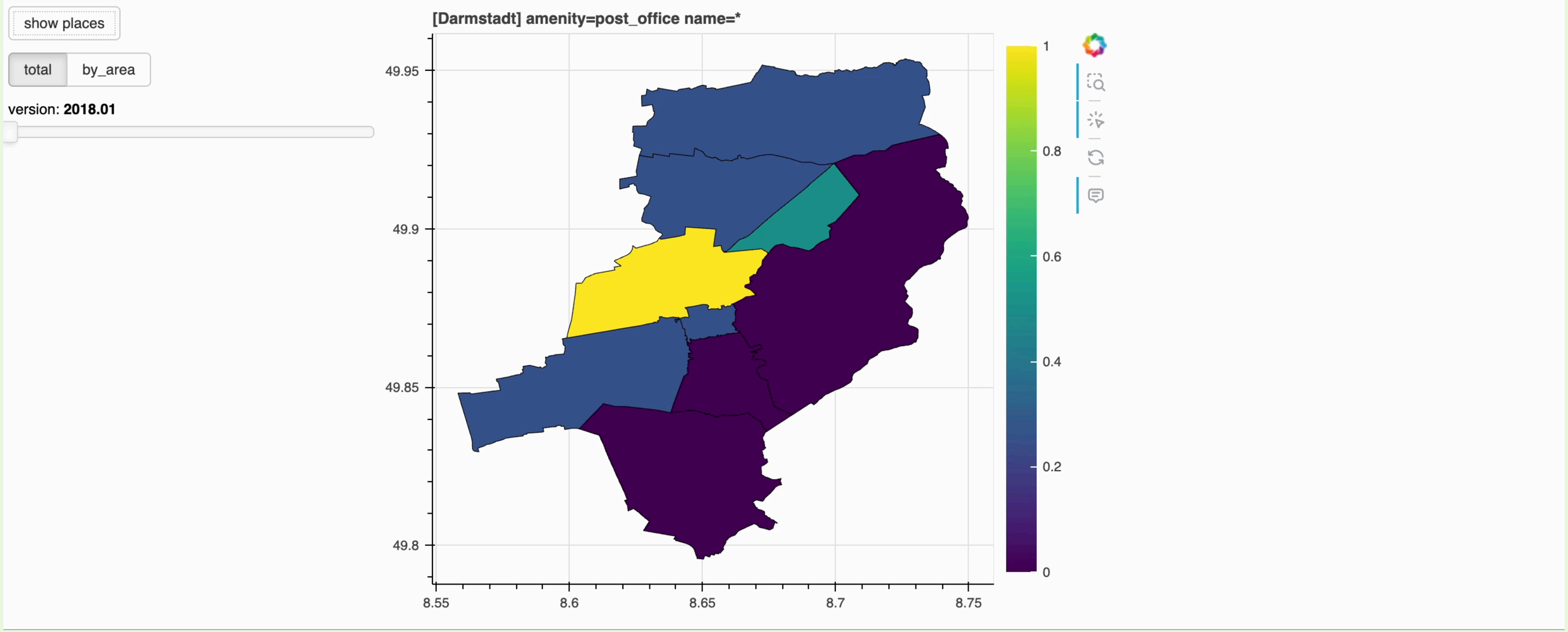
geometry	name	wikidata	id	osm_url	count
POLYGON ((8.64126 49.87164, 8.64261 49.87202, ...	Darmstadt-Mitte	Q1166438	r6604794	https://www.osm.org/relation/6604794	2
POLYGON ((8.63439 49.89673, 8.64077 49.89762, ...	Darmstadt-Nord	Q1166439	r6604800	https://www.osm.org/relation/6604800	1
POLYGON ((8.6189 49.91331, 8.6189 49.91333, 8....	Arheilgen	Q1166422	r6606314	https://www.osm.org/relation/6606314	1

Interactive Visualisations

Bokeh

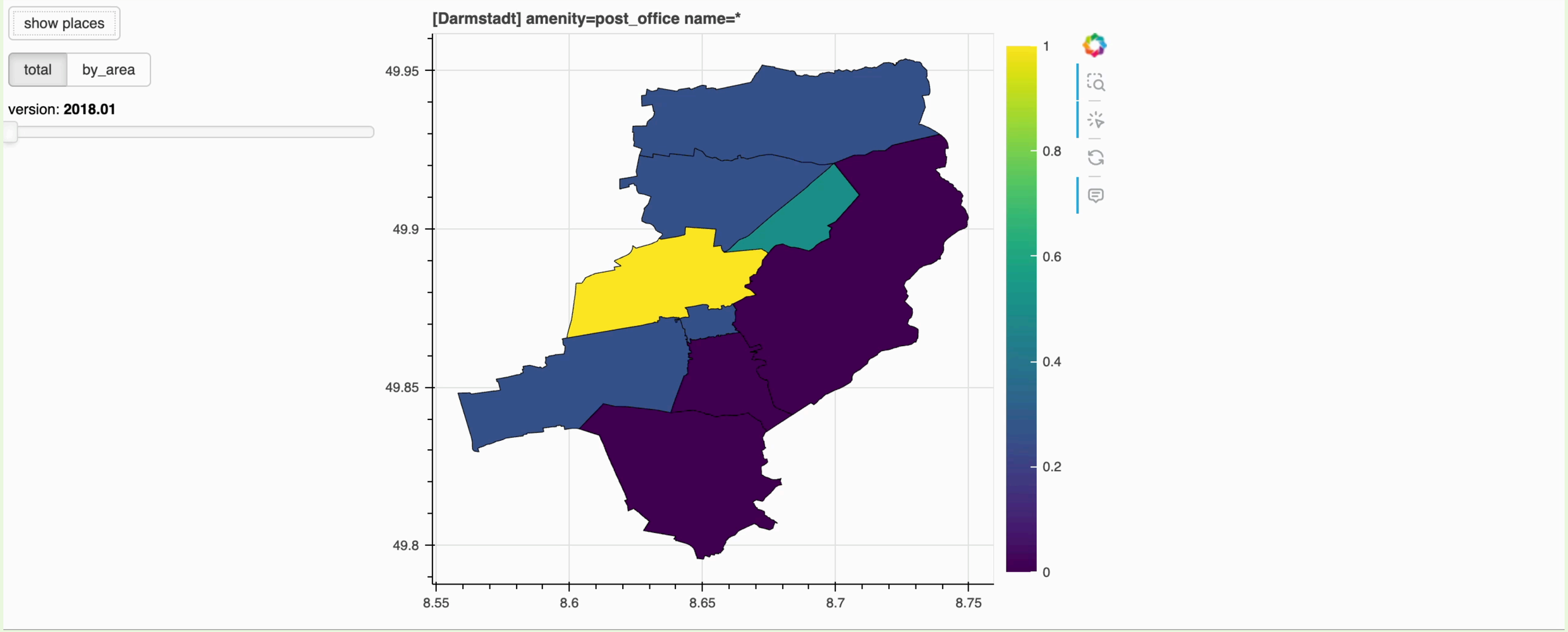
- GeoJSONDataSource
- Tools, widgets
- Custom JS callbacks

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

plot = figure()

areas = plot.patches(source=areas_data)

show(plot)
```



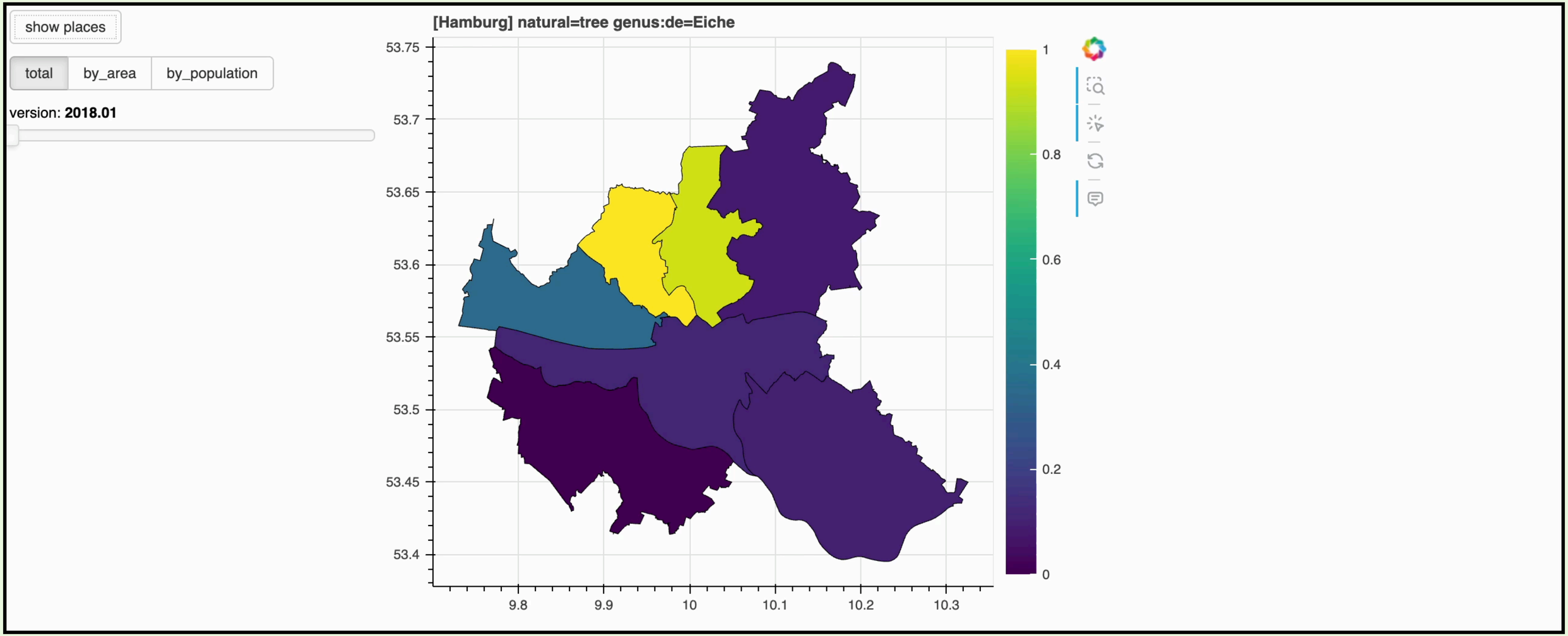
Post offices in Darmstadt

Map data from OpenStreetMap



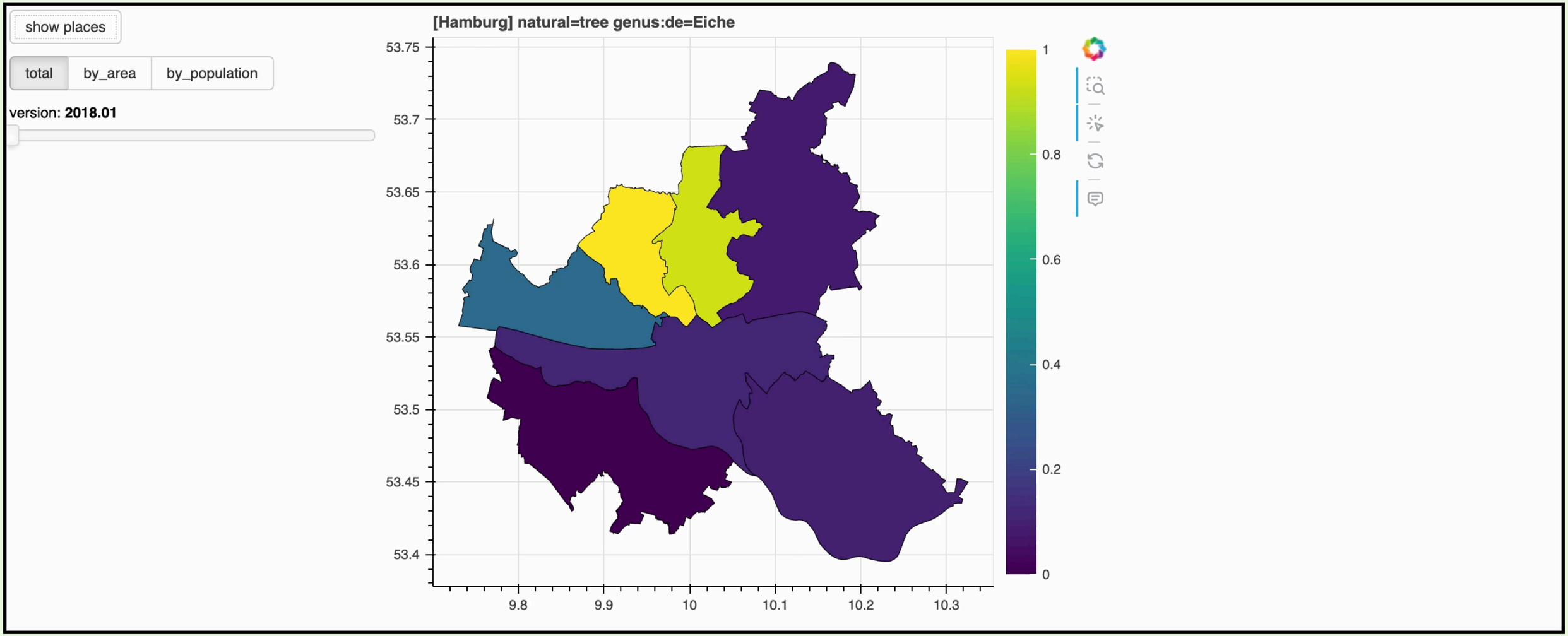
Post offices in Darmstadt

Map data from OpenStreetMap



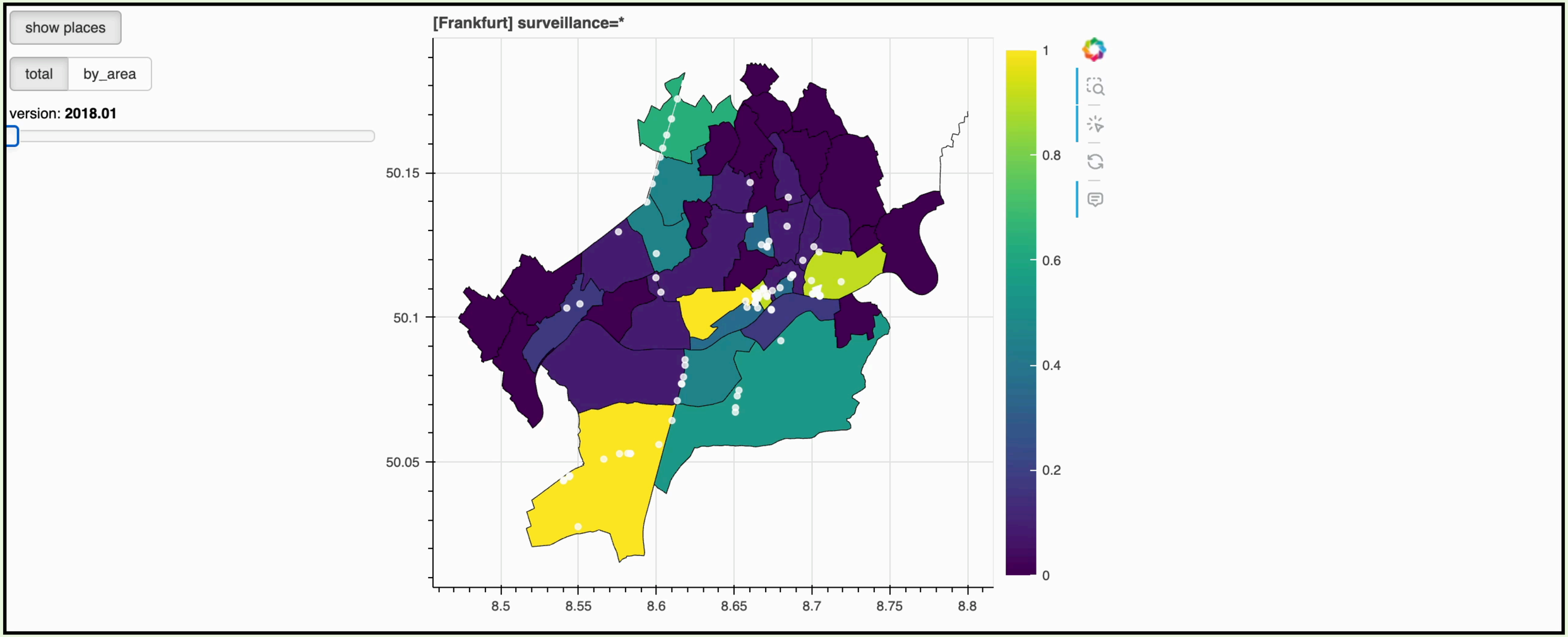
Oak trees in Hamburg

Map data from OpenStreetMap



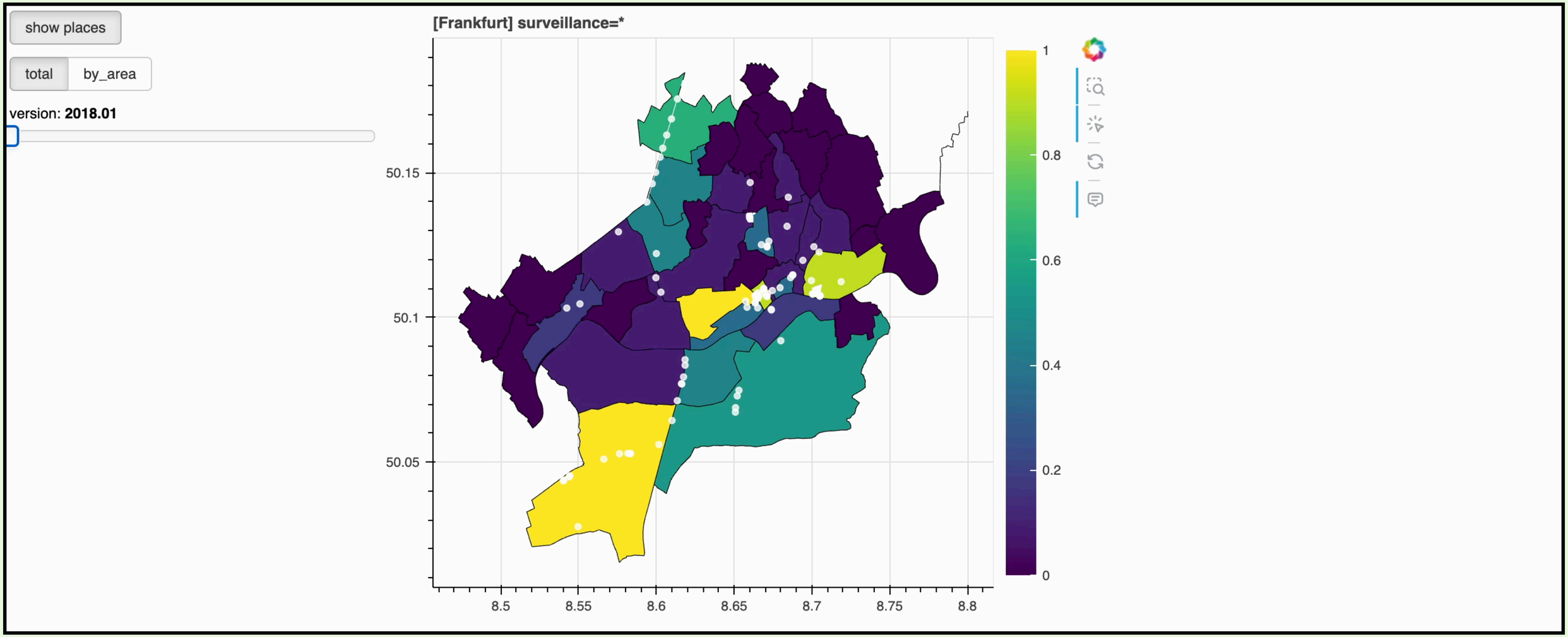
Oak trees in Hamburg

Map data from OpenStreetMap



Surveillance in Frankfurt

Map data from OpenStreetMap



Surveillance in Frankfurt

Map data from OpenStreetMap

So where did the post offices go?

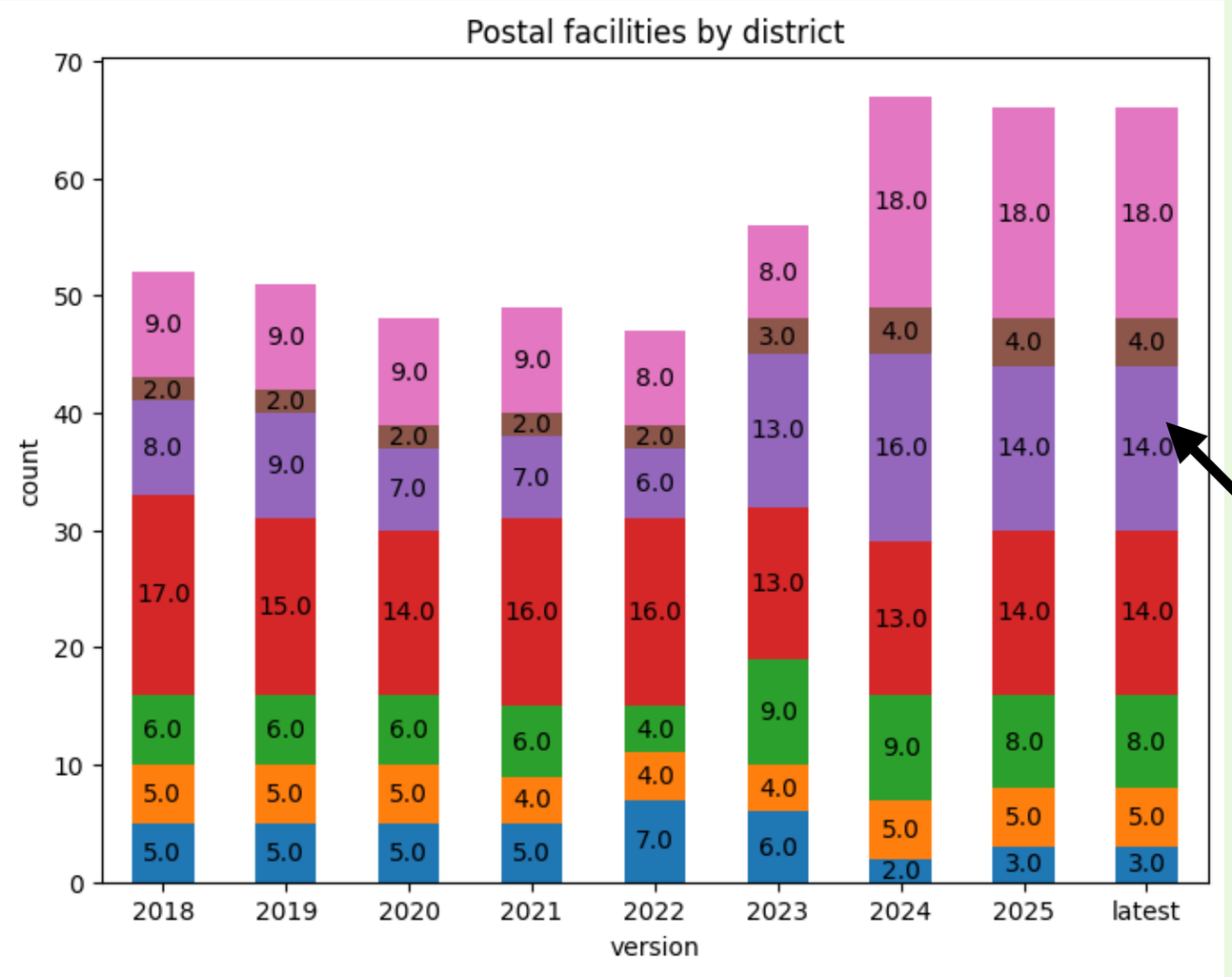
Was my experience reflected in the data?

Was I asking the correct question?

So where did the post offices go?

Was my experience reflected in the data?

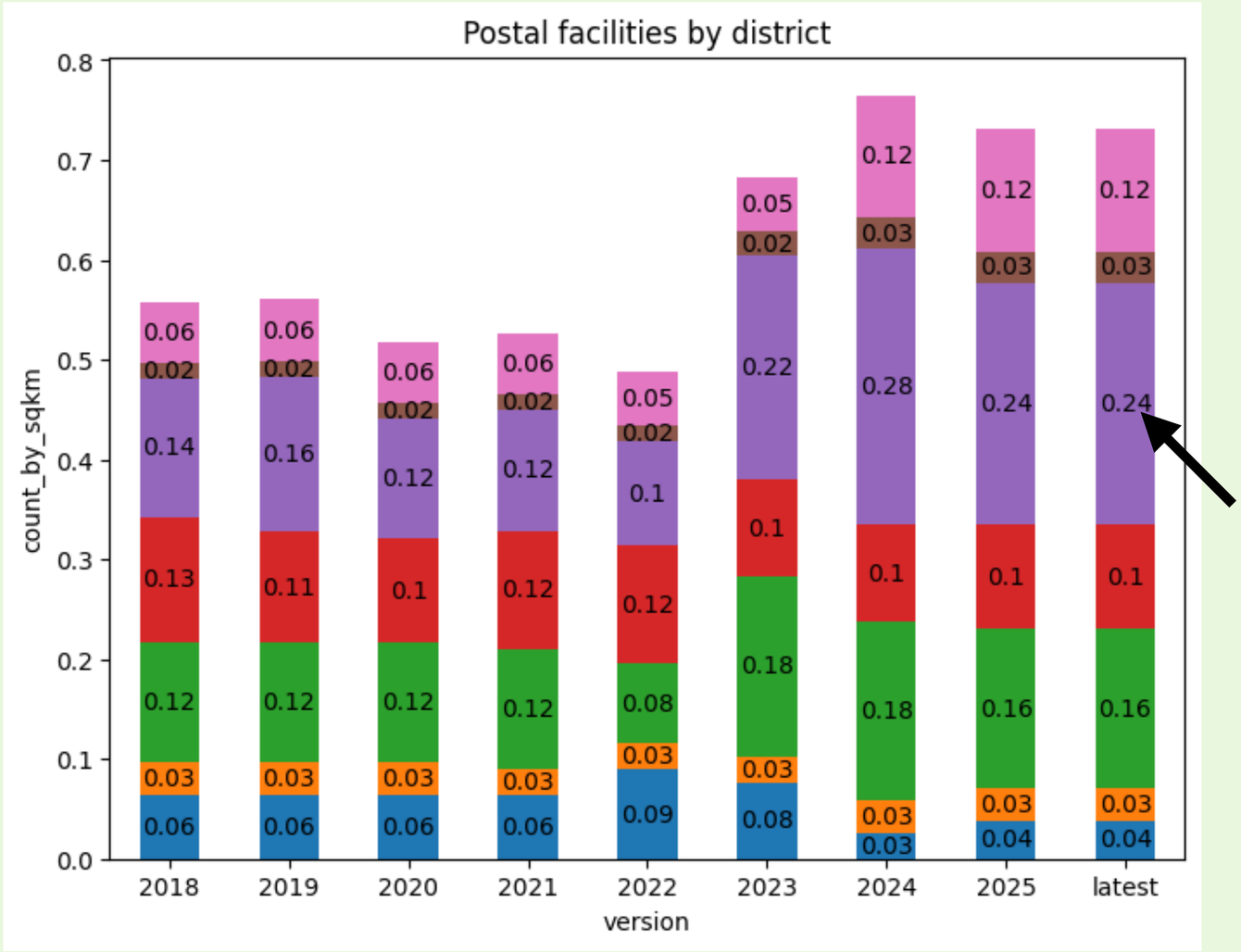
Was I asking the correct question?



So where did the post offices go?

Was my experience reflected in the data?

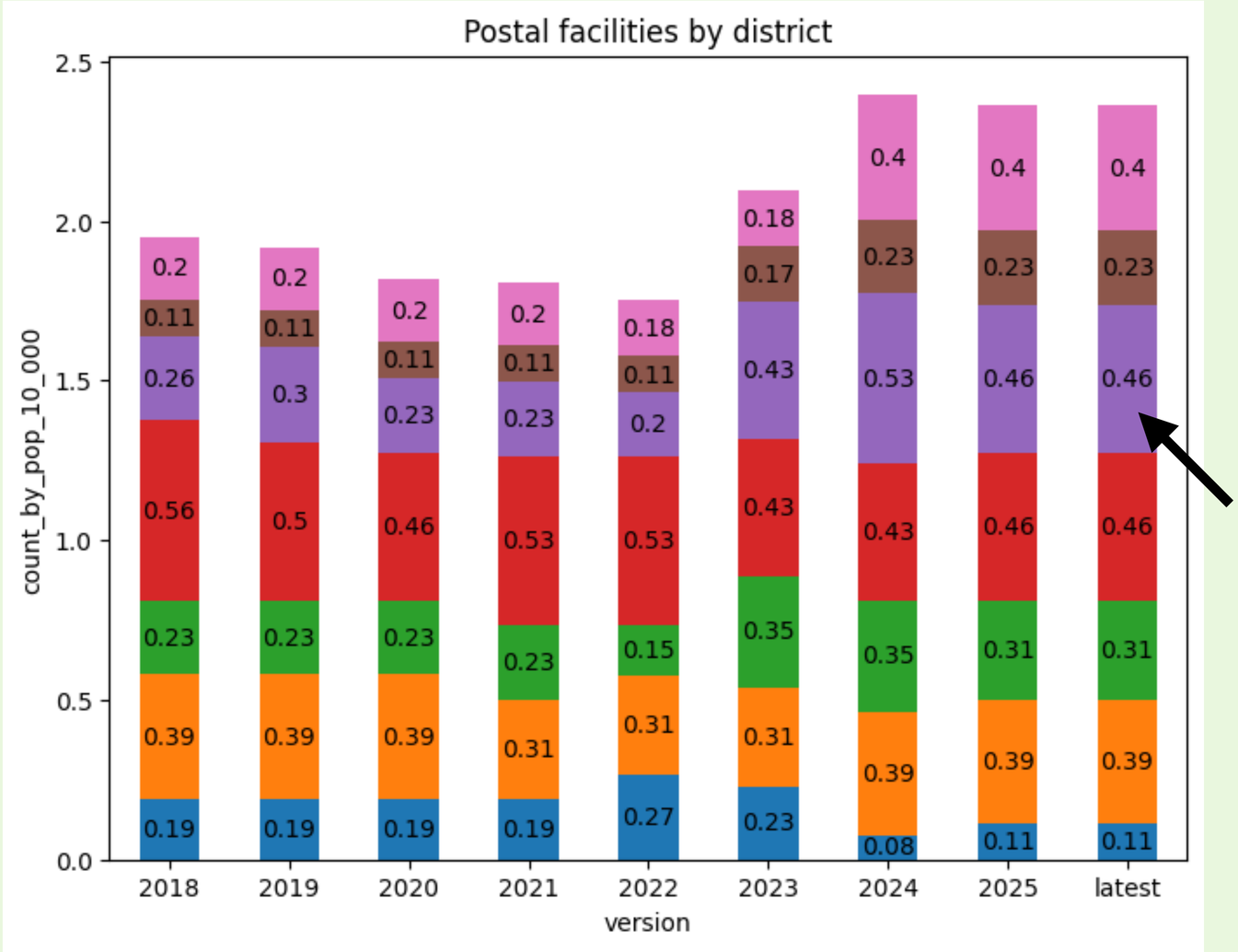
Was I asking the correct question?



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