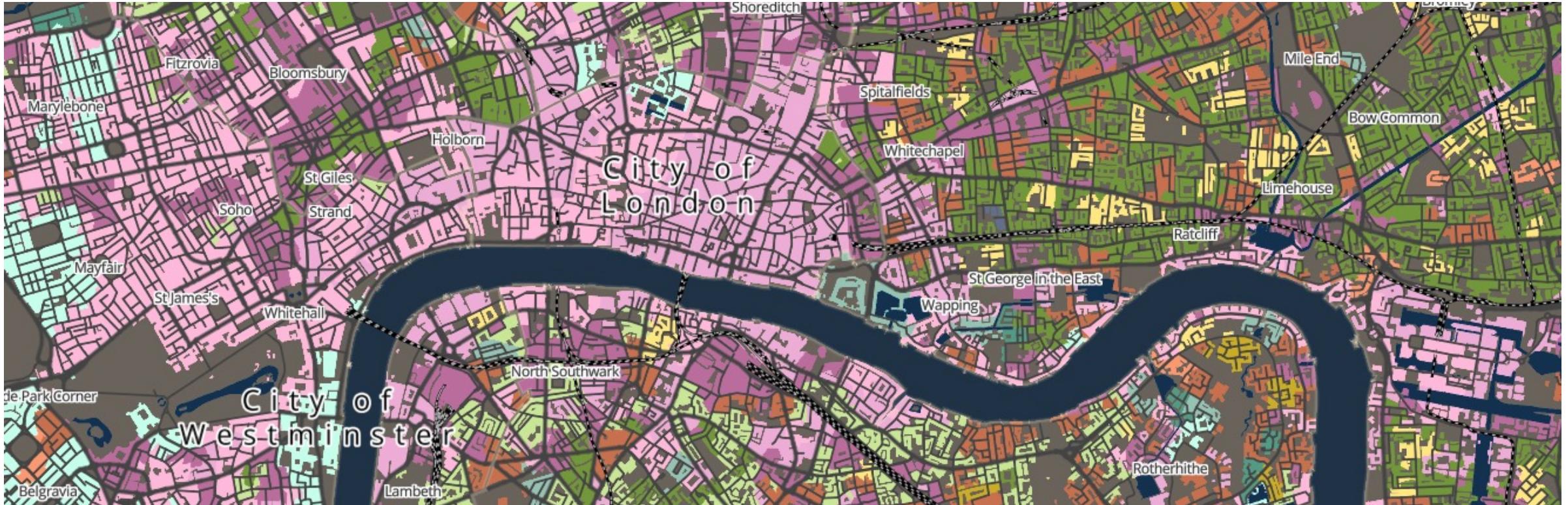


Geocomputation

Point Pattern Analysis



Where are we at?

Part I: Foundational Concepts

W1 Geocomputation: An Introduction

W2 GIScience and GIS software

W3 Cartography and Visualisation



QGIS

W4 Programming for Data Analysis

W5 Programming for Spatial Analysis



R

Where are we at?

Part II: Core Spatial Analysis

W6 Geometric Operations and Spatial Queries

W7 **Point Pattern Analysis**

W8 Spatial Autocorrelation



R

Part III: Advanced Spatial Analysis

W9 Rasters, Zonal Statistics and Interpolation



R

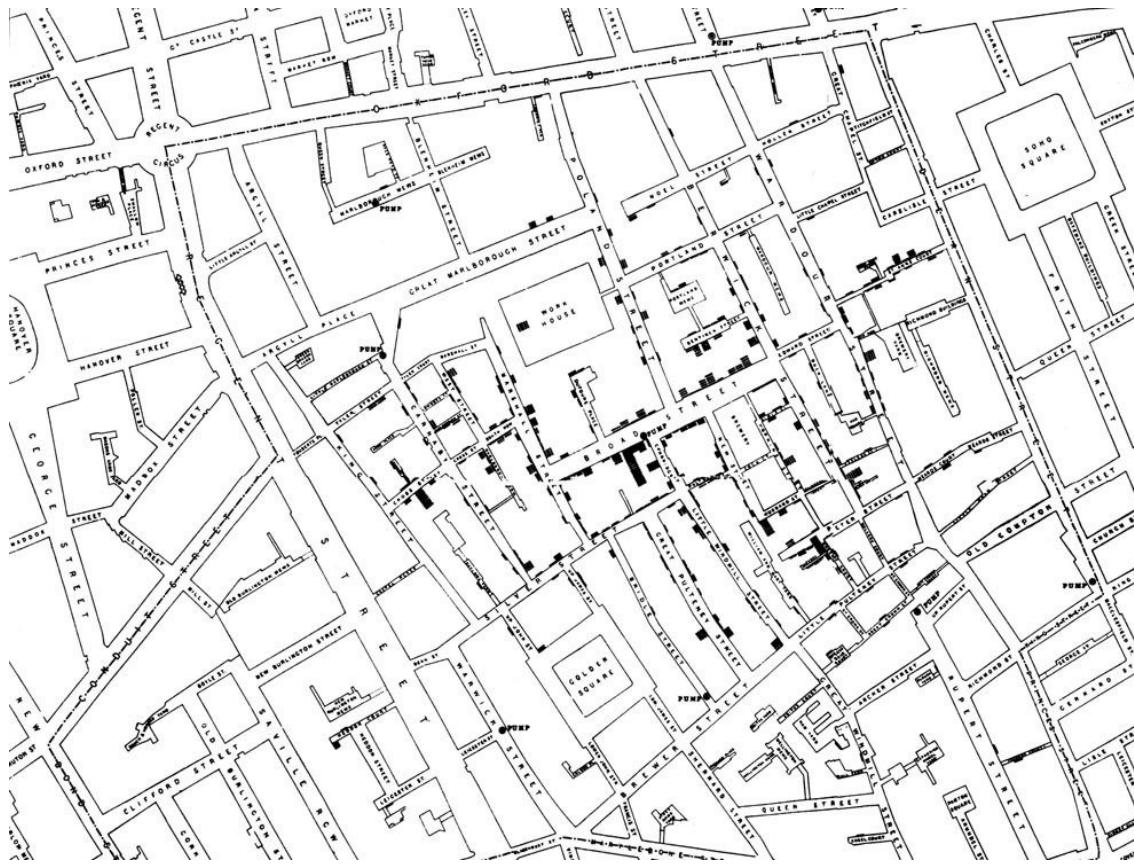
This week

- Focus analysis directly on point events rather than aggregation to some type of administrative geography.
- Roughly three ways to describe or characterise point processes: descriptive (summary) statistics, distance-based methods (average nearest neighbour, Ripley's K), density-based methods (DBSCAN, kernel density estimation).
- Some practical examples.

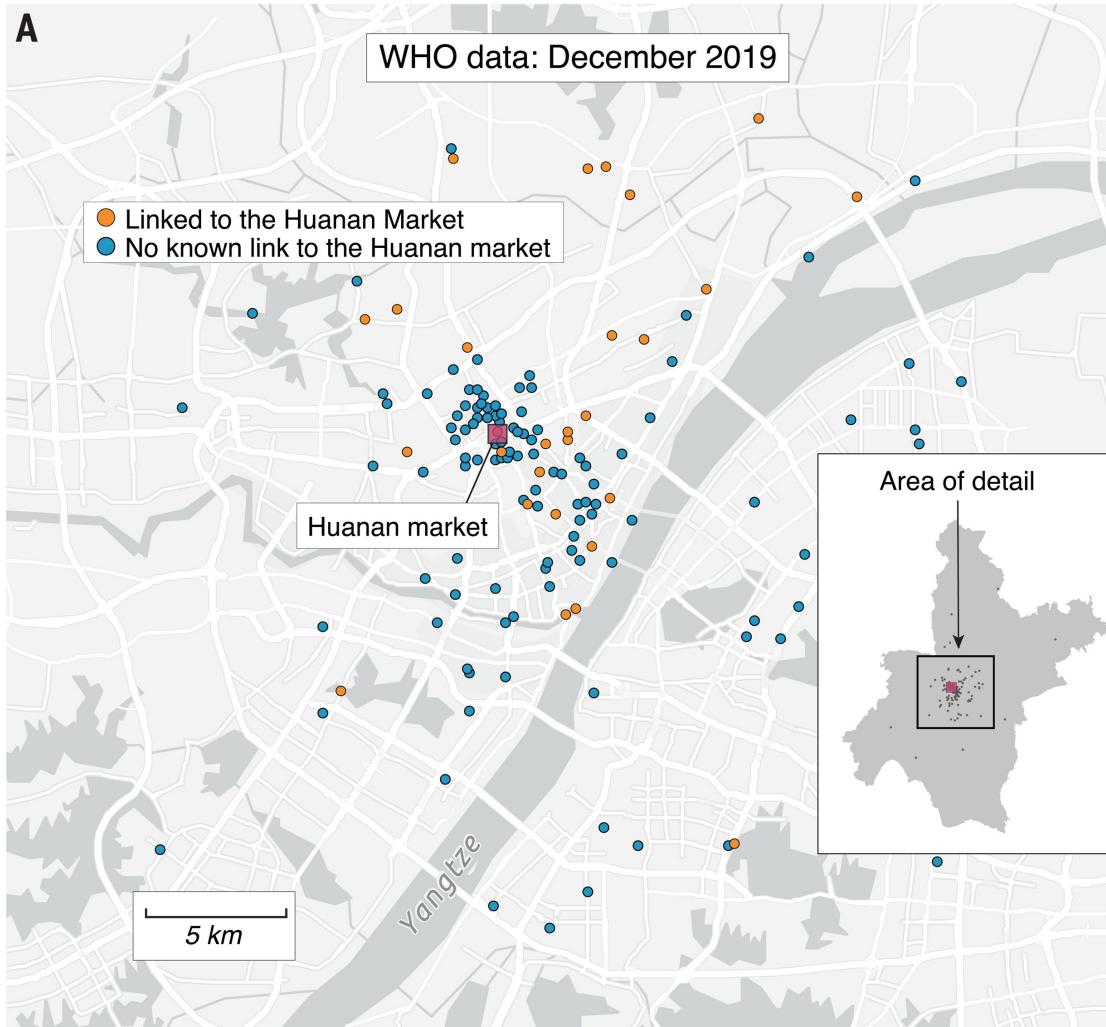
Why do we want to analyse points?

- We use points to represent the location of individuals, features or events (e.g. a crime, report of an illness).
- Spatial patterns within the distribution of these points can reveal relationships.
- It allows us to study the occurrence of the phenomena (i.e. event) or points across a similar attribute value (non-spatial information).

Spatial analysis



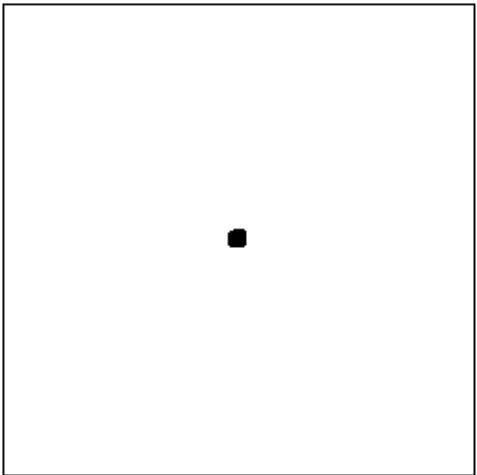
Spatial analysis



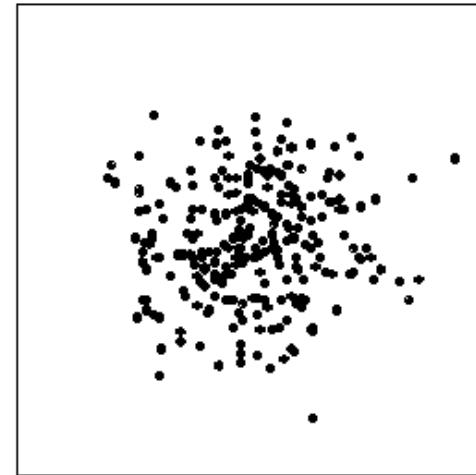
Worobey *et al.* 2022

Why do we want to analyse points?

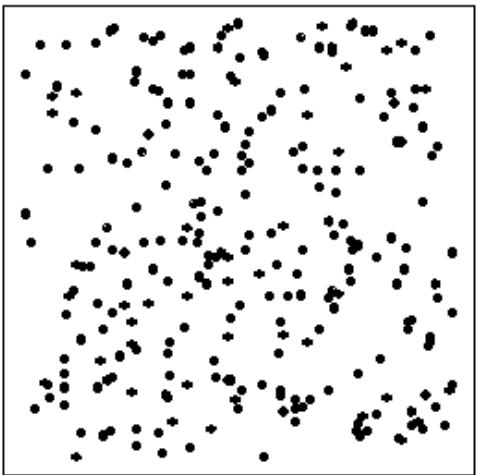
clustered



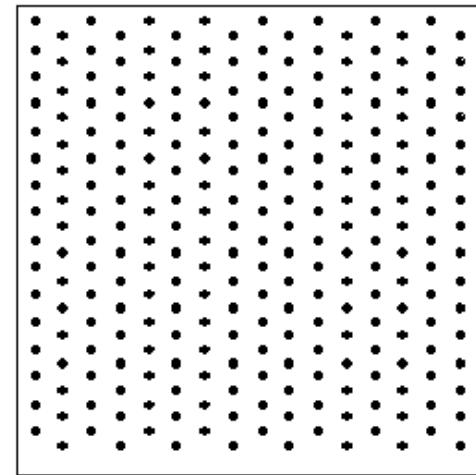
normal



random



regular



How do we analyse points?

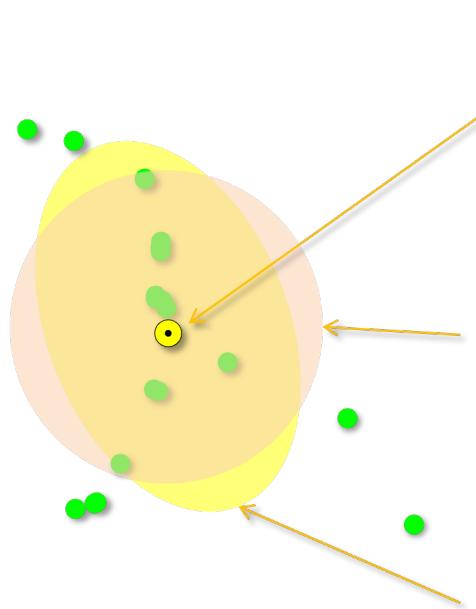
- Prove the existence of clusters using **Tobler's Law** by considering the density of and distance between our points.
- Aim to understand / test whether our Point Pattern is randomly distributed, clustered or dispersed by comparing our observed pattern against the "Poisson" distribution for the same number of "events/points" over the same area in space.

How do we analyse points?

- Descriptive statistics
- Density-based
- Distance-based

Descriptive statistics

Descriptive statistics



Mean center is the computed average X and Y coordinate values.

$$\bar{s} = \left(\frac{\sum_{i=1}^n x_i}{n}, \frac{\sum_{i=1}^n y_i}{n} \right)$$

Standard distance is a measure of the variance between the average distance of the features to the mean center.

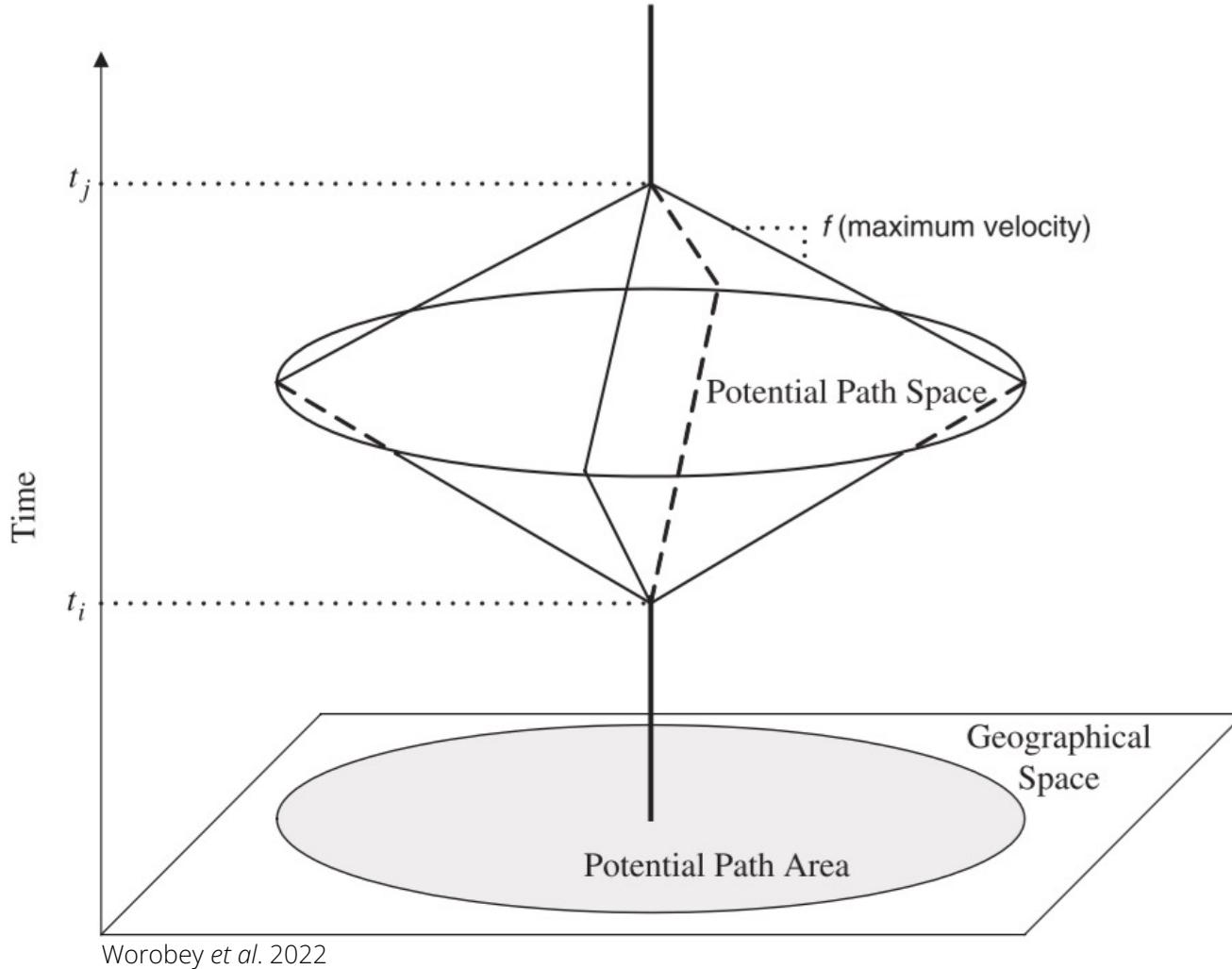
$$d = \sqrt{\frac{\sum_{i=1}^n (x_i - \mu_x)^2 + (y_i - \mu_y)^2}{n}}$$

Standard deviational ellipse computes separate standard distances for each axis.

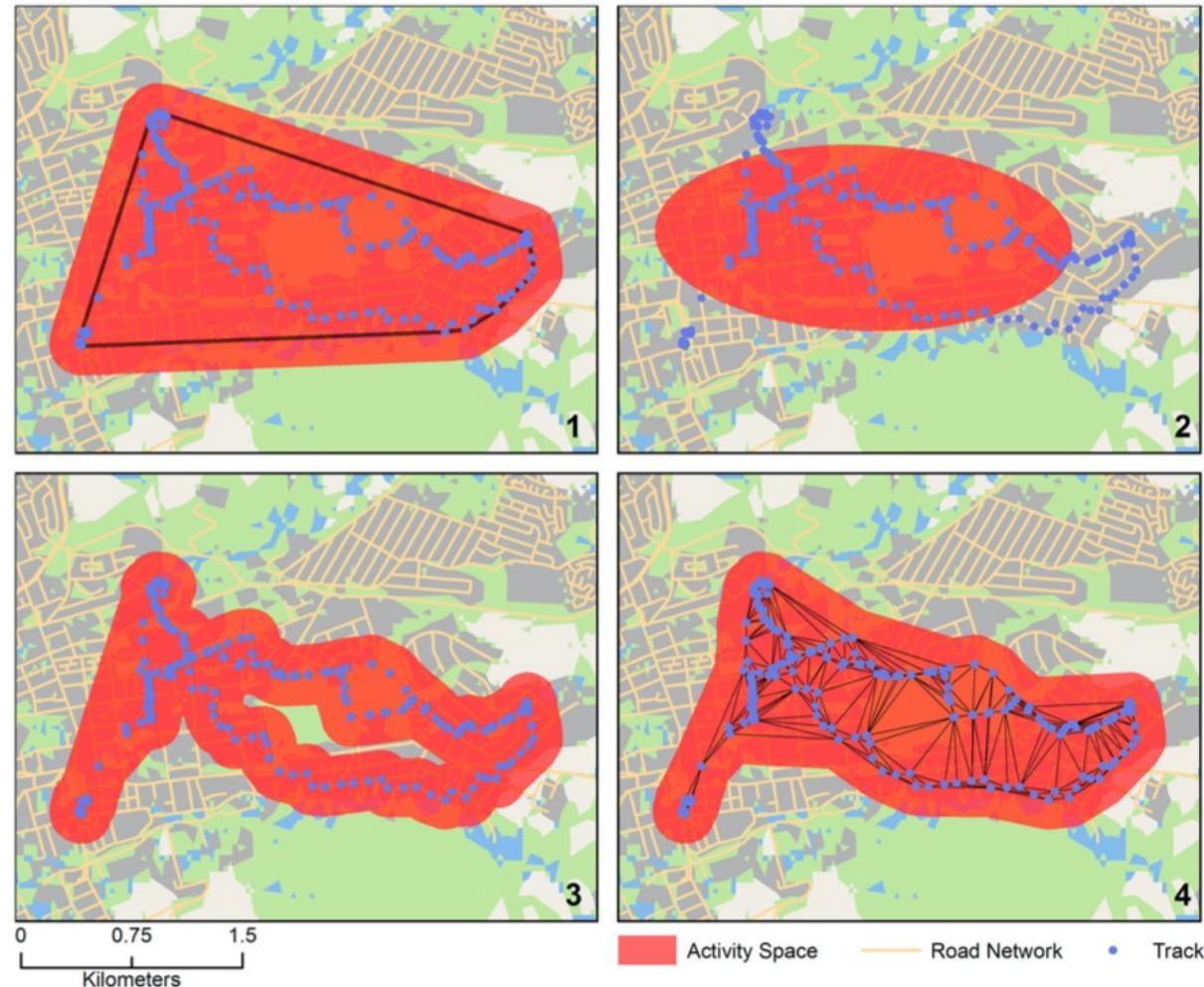
$$d_x = \sqrt{\frac{\sum_{i=1}^n (x_i - \mu_x)^2}{n}}$$

$$d_y = \sqrt{\frac{\sum_{i=1}^n (y_i - \mu_y)^2}{n}}$$

Descriptive statistics



Descriptive statistics



Distance-based measures

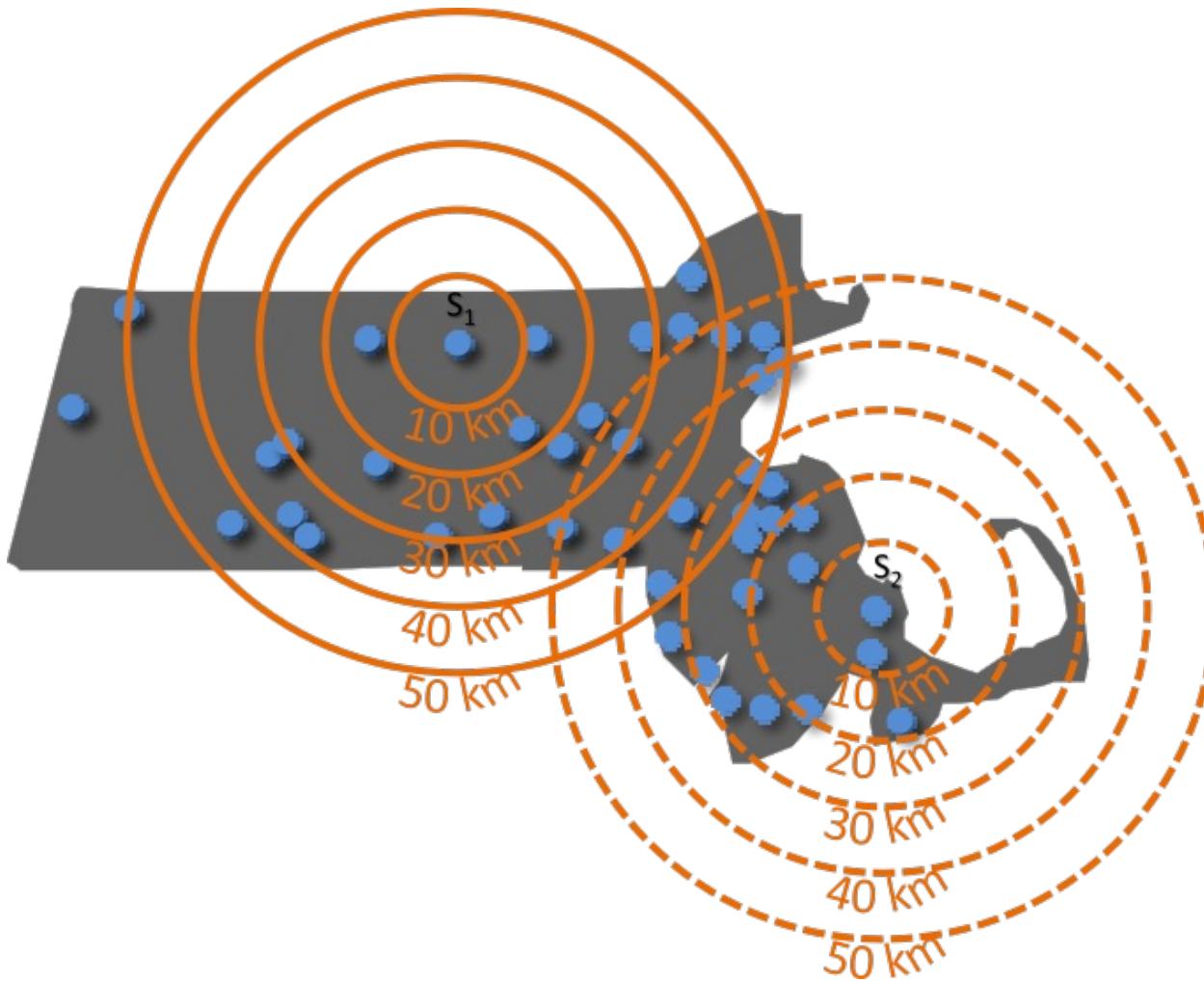
Distance-based measures

- Using distance as a summary measure to describe a point process.
- Measure of spread of the distribution.
- Well-known example is average nearest neighbour (ANN).
- ANN measures the distance between each feature and its nearest neighbor's location. It then averages all these nearest neighbor distances.
- If the average distance is less than the average for a hypothetical random distribution, the distribution of the features being analysed is considered clustered. If the average distance is greater than a hypothetical random distribution, the features are considered dispersed.

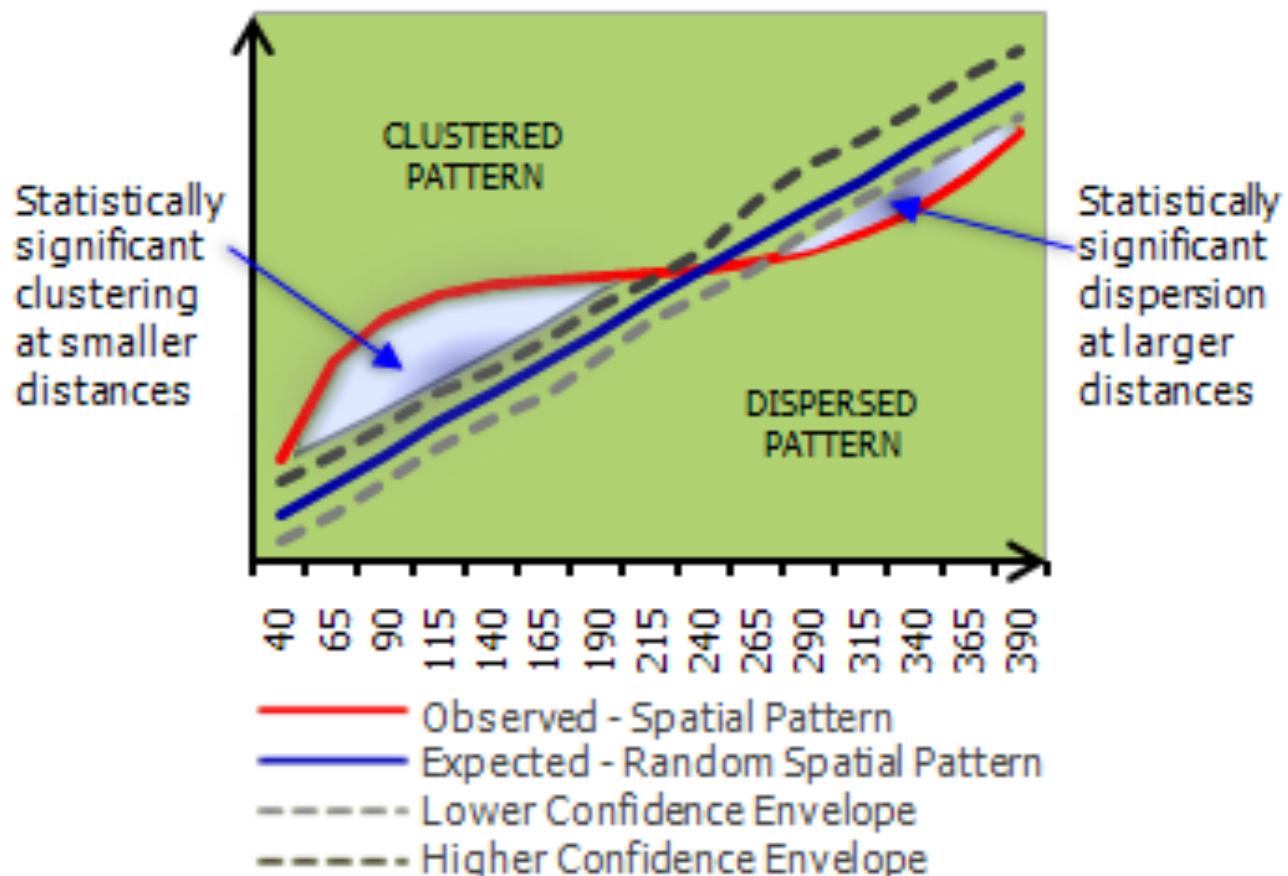
Distance-based measures

- Another example is Ripley's K.
- Ripley's K counts the number of observations within a user defined set of distances and compares this to a hypothetical (random) pattern of observations.
- Ripley's K function is generally calculated at multiple distances allowing you to see how point pattern distributions can change with scale. For example, at near distances, the points could cluster, while at farther distances, points could be dispersed.
- Distance-based measure of dispersion across scales.

Distance-based measure



Distance-based measures



Density-based measures

Density-based measures

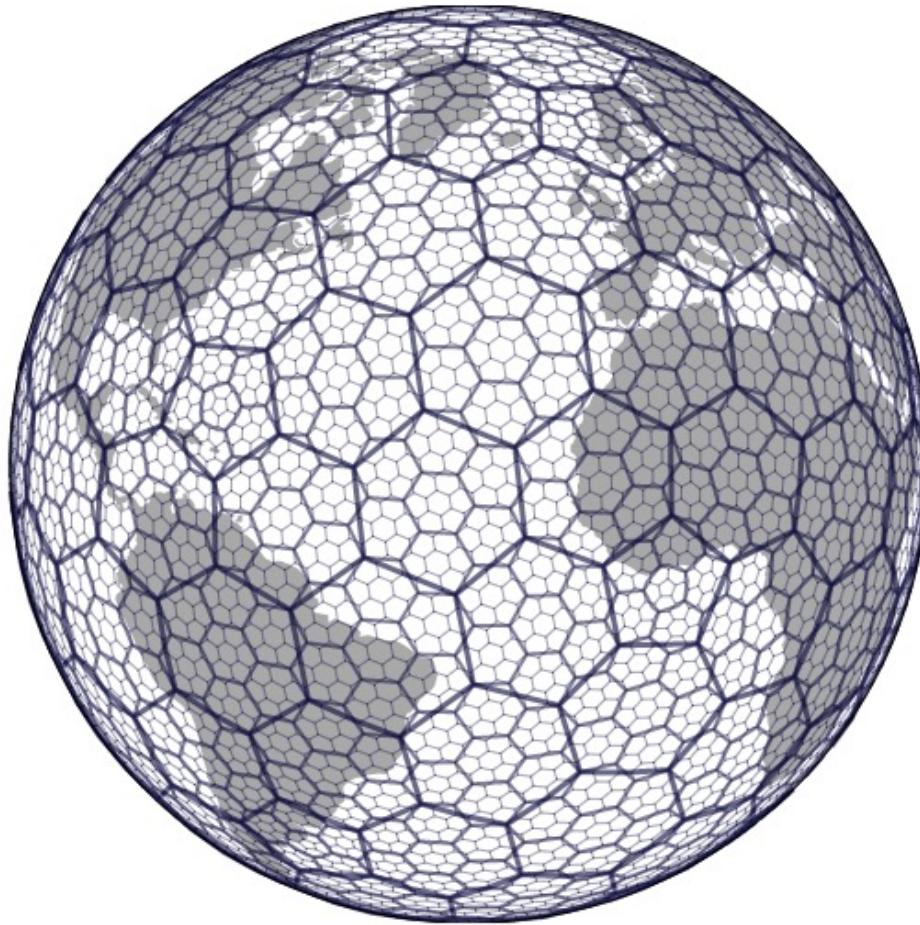
- Measure of the intensity of a point process.
- Density-Based Spatial Clustering of Applications with Noise (DBSCAN) to find high-density clusters.
- Kernel Density Estimation (KDE).

Density-based measures



Gimond, M. 2020. *Geodesic geometry*. [online] <https://mgimond.github.io/Spatial/index.html>

Density-based measures

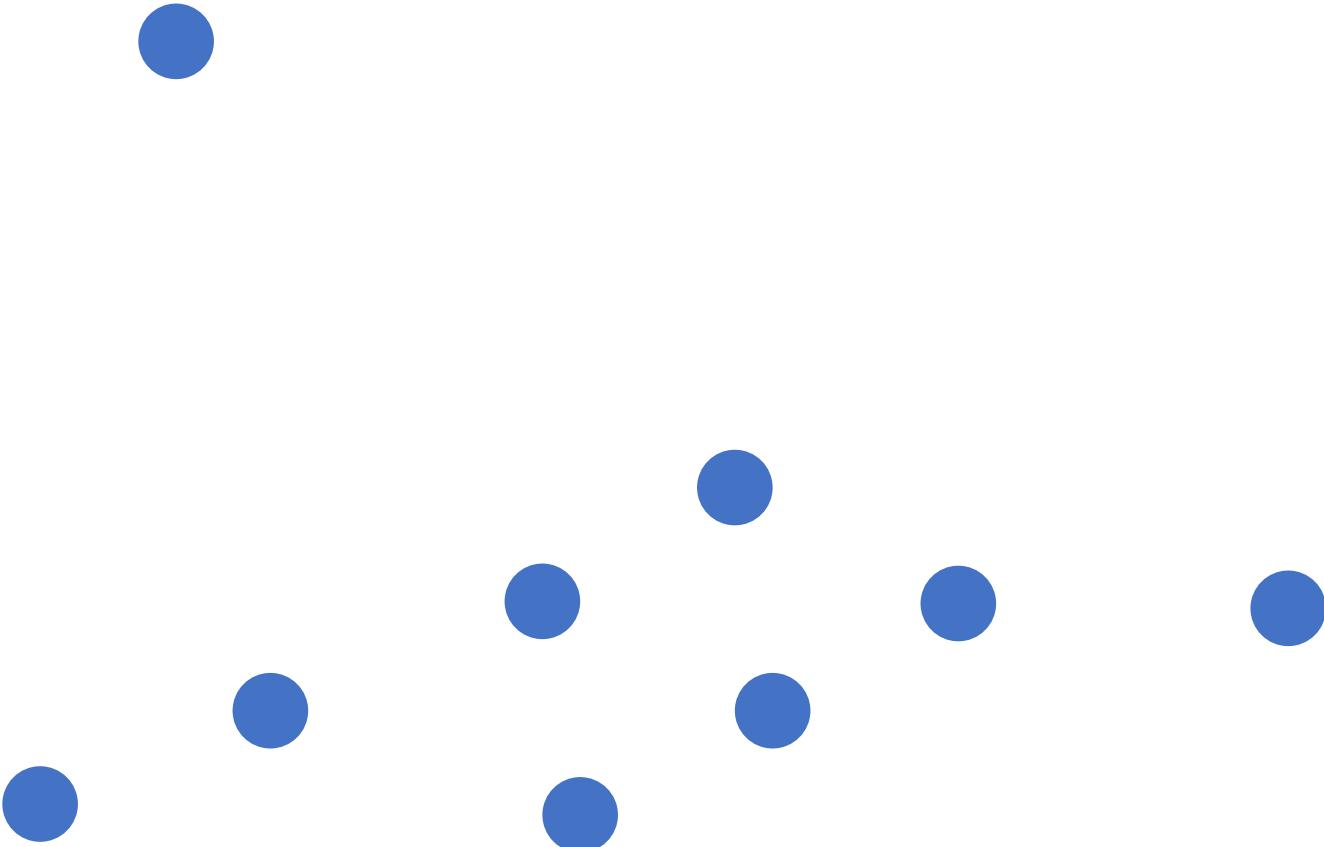


Uber. 2018. *H3: Uber's Hexagonal Hierarchical Spatial Index*. [online] <https://eng.uber.com/h3/>

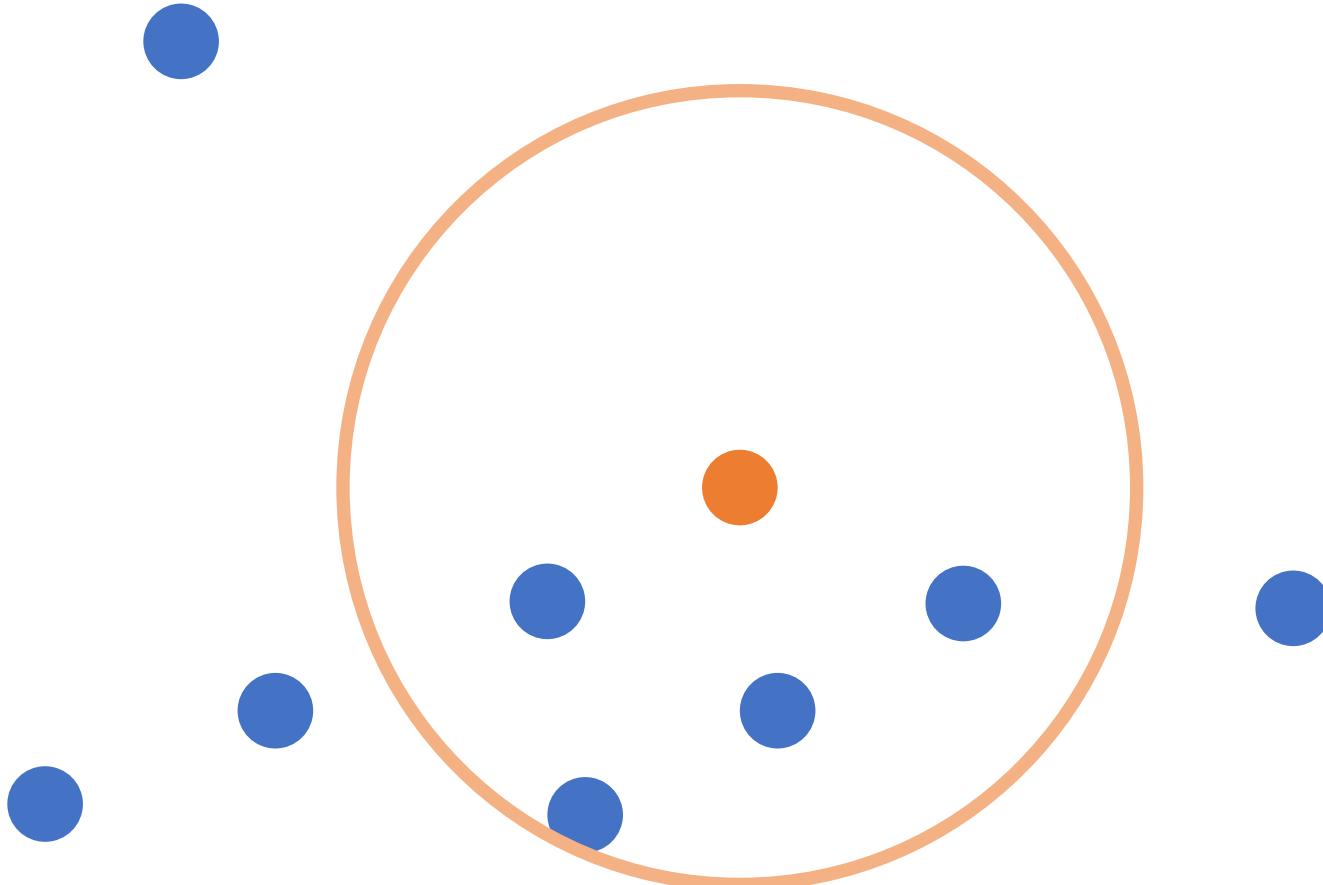
DBSCAN

- Density-Based Spatial Clustering of Applications with Noise (**DBSCAN**).
- DBSCAN is probably one of the most common clustering algorithms and also most cited in scientific literature.
- Functionally DBSCAN detects clusters of points, with noise, that reach the minimum density threshold based on two inputs: `minpts` and `epsilon` neighbourhood radius (`eps`), where `eps` is the maximum distance, or search radius, between each point in a cluster and `minpts` is the minimum number of points to be assigned to a cluster.

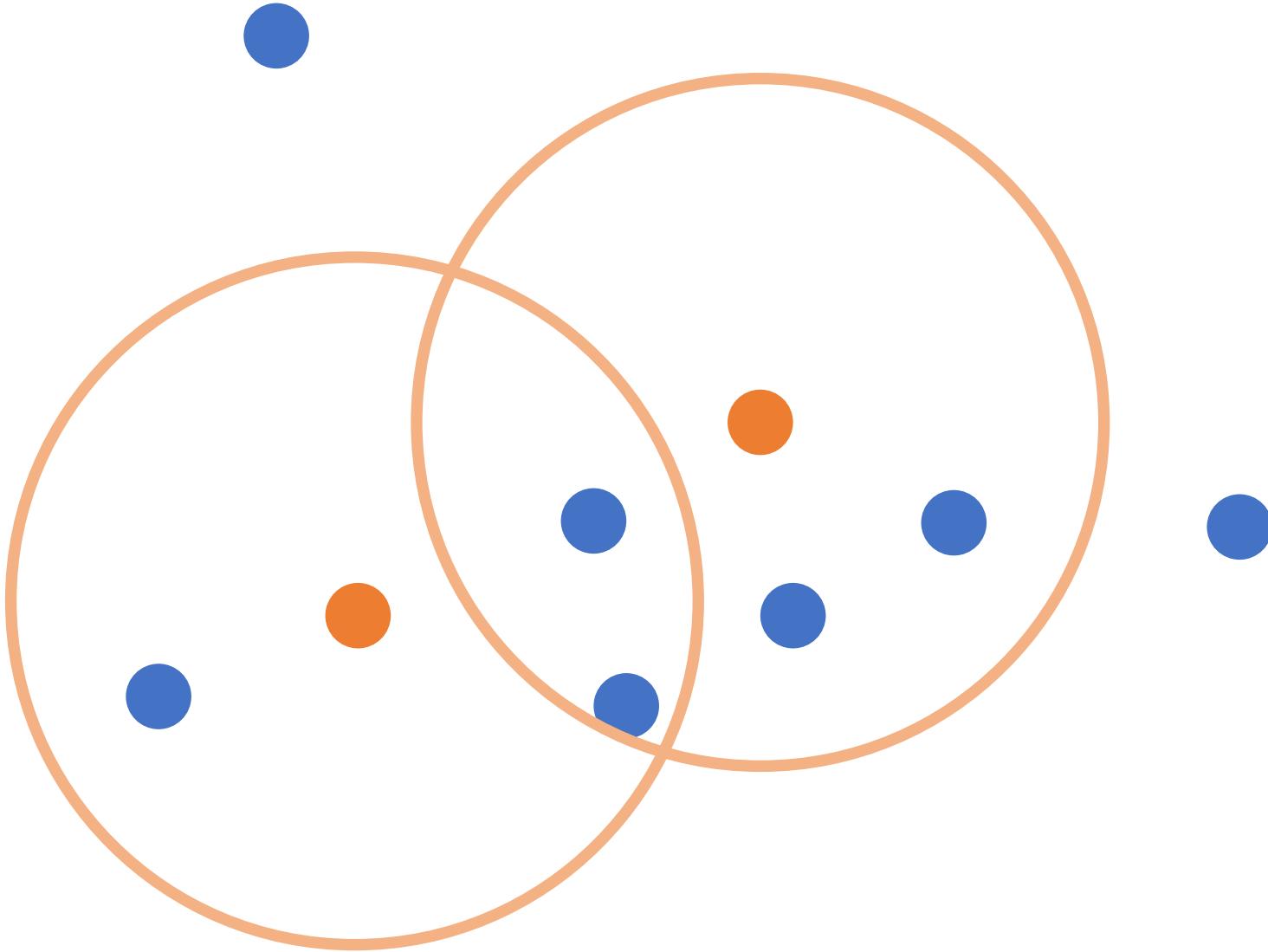
DBSCAN



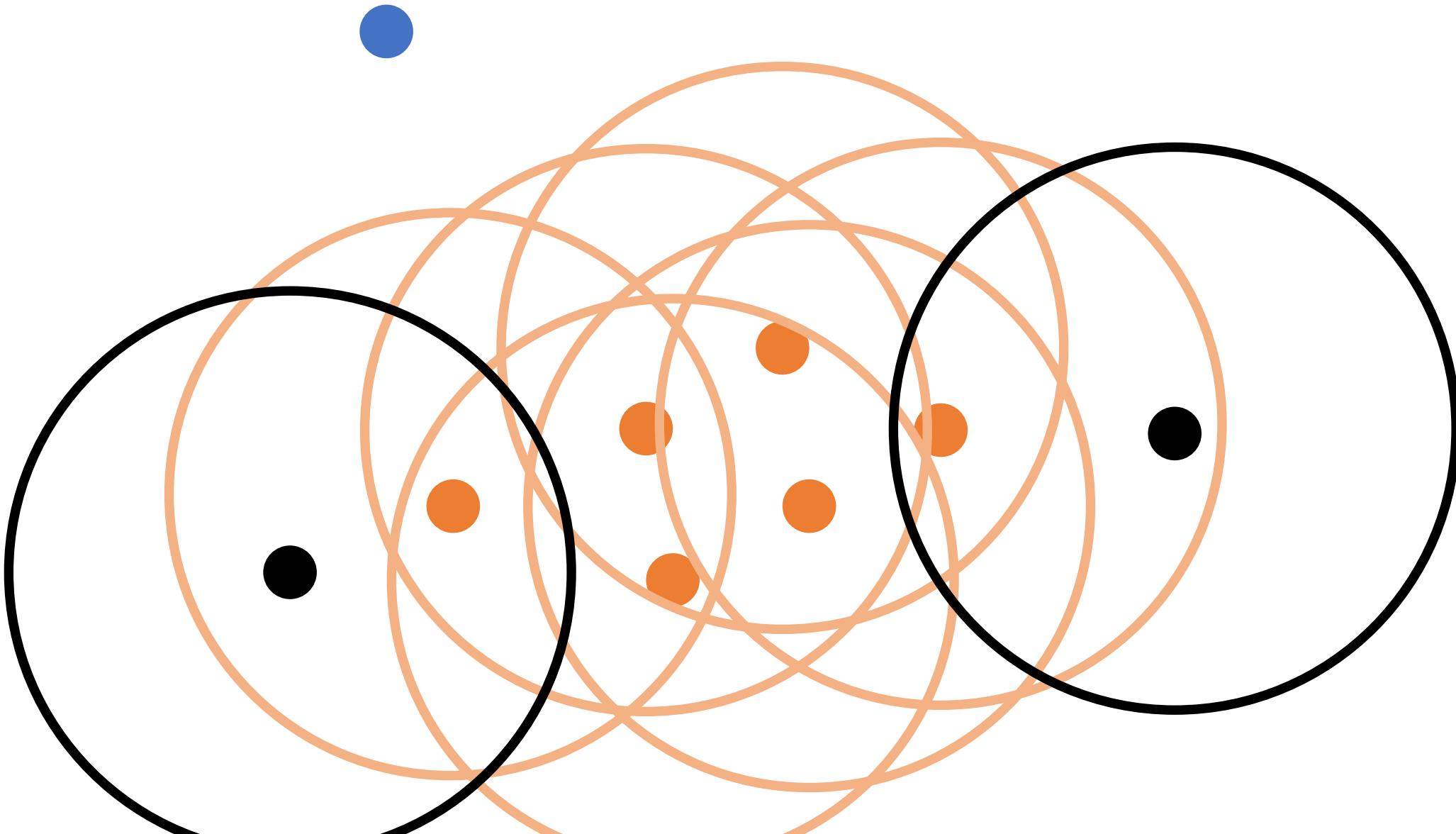
DBSCAN



DBSCAN



DBSCAN



DBSCAN

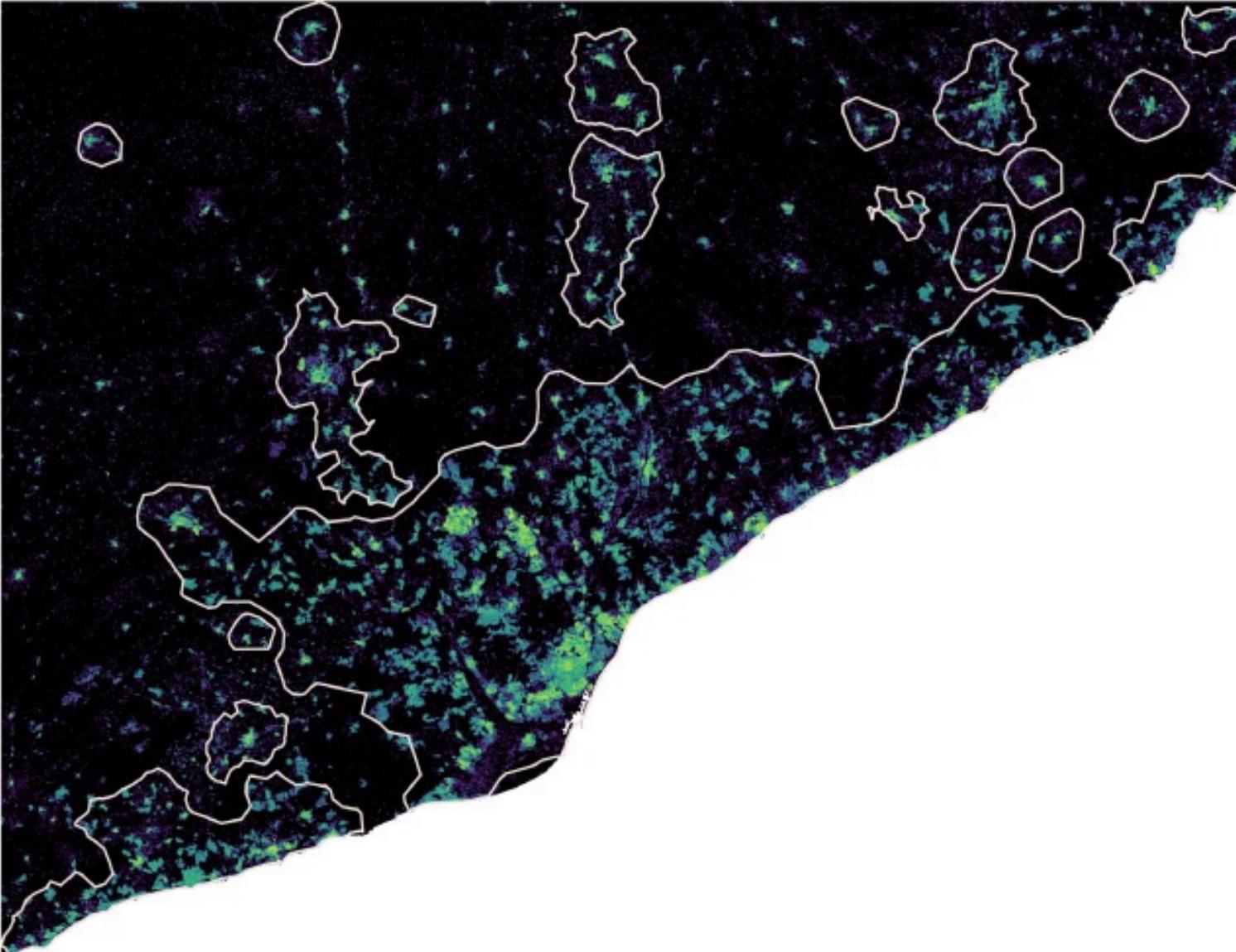


DBSCAN

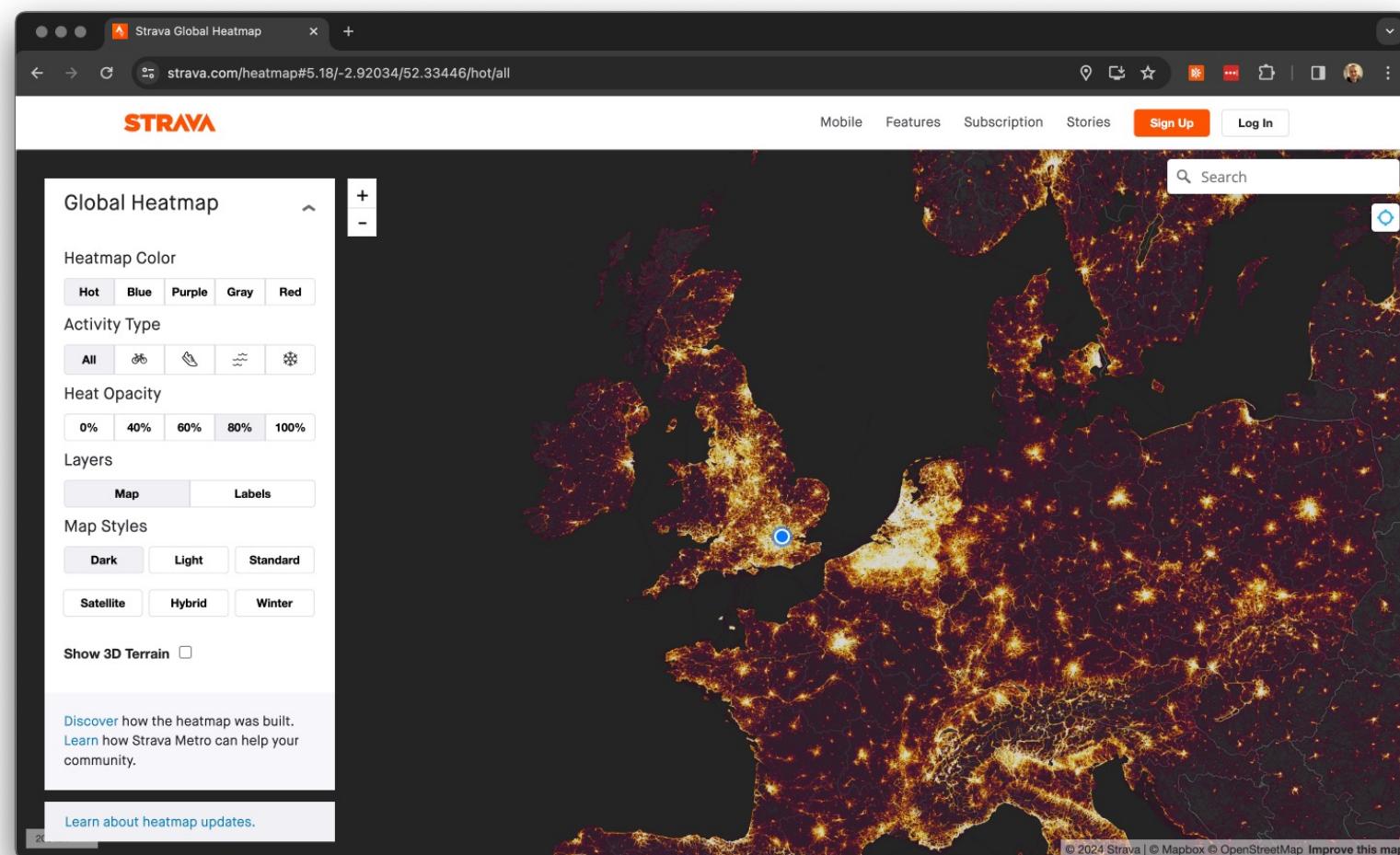
- No need to specify number of clusters *a priori*.
- DBSCAN can find non-linearly separable clusters (i.e. arbitrarily shaped).
- However: highly dependent on the distance measure and not robust to data sets with large differences in densities as the parameters cannot be chosen to cover both situations.

DBSCAN

Arribas-Bel et al. 2021. Building(s and) cities: Delineating urban areas with a machine learning algorithm.
Journal of Urban Economics 125: 103217



Kernel density estimation



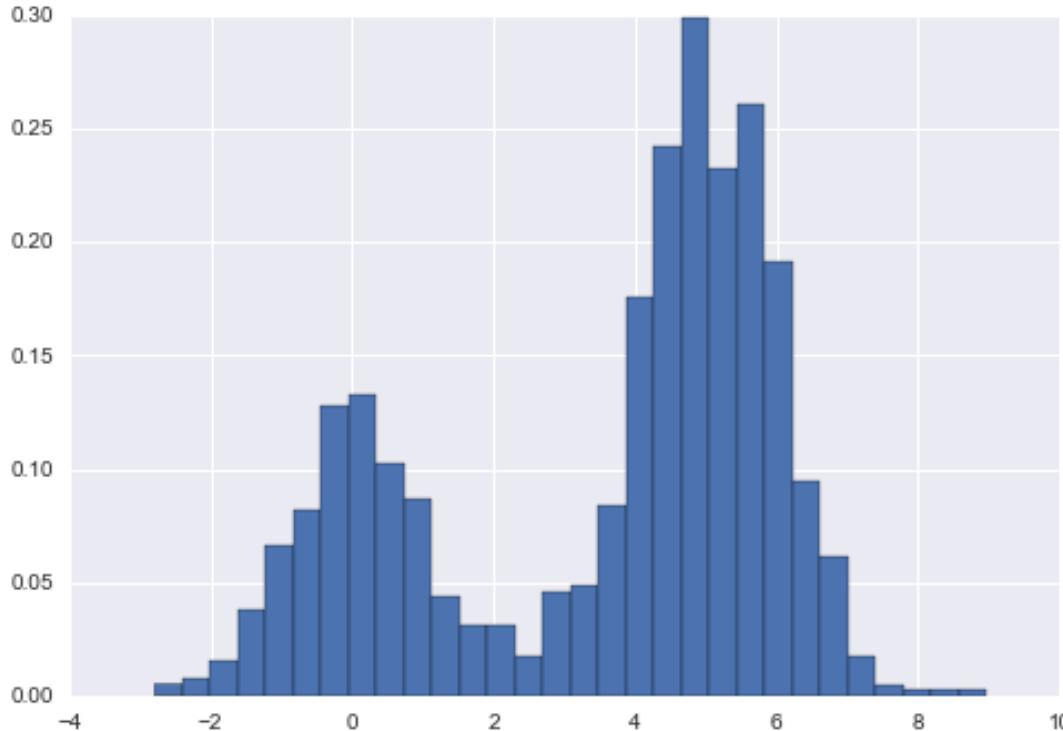
Kernel density estimation

- Heatmap: Kernel density estimation (KDE) is a non-parametric way to estimate the probability density function of a random variable.
- It uses local information defined by windows (called kernels) to estimate densities of specified features at given locations.
- In essence it is a smoothing function where a continuous curve is created, based on a finite data sample.

Kernel density estimation

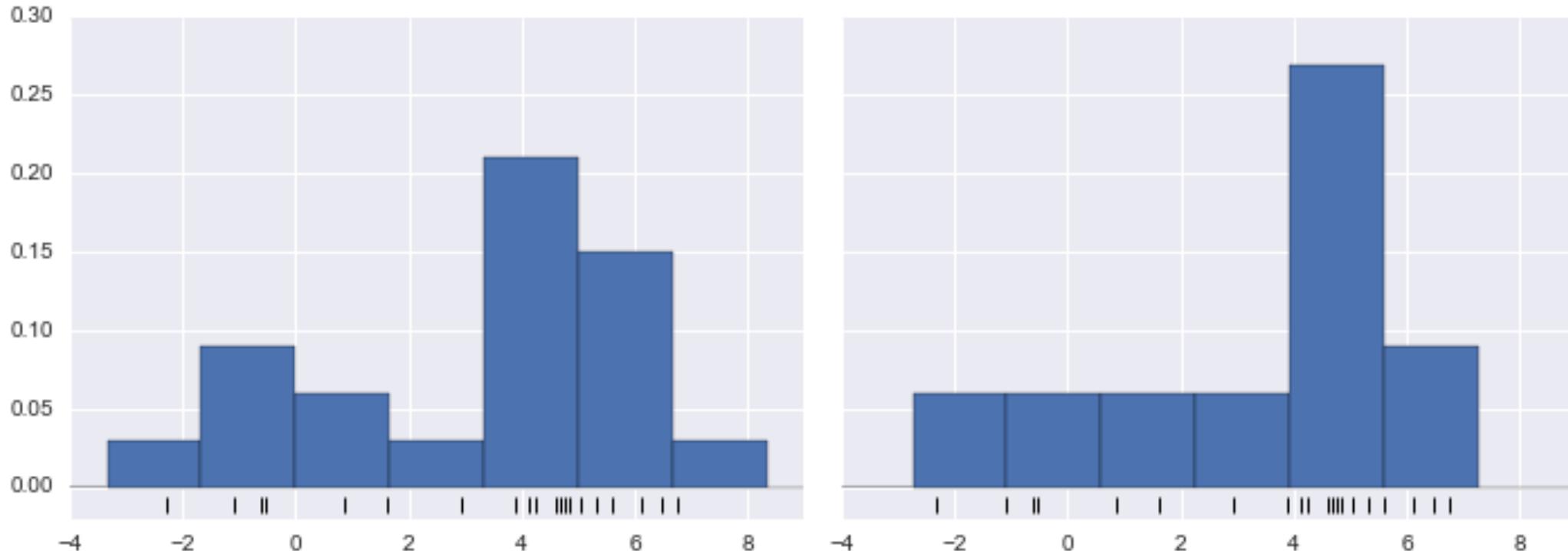
```
Tycho — python — 85x25
>>> def make_data(N, f=0.3, rseed=1):
...     rand = np.random.RandomState(rseed)
...     x = rand.randn(N)
...     x[int(f * N):] += 5
...     return x
...
[>>> x = make_data(1000)
[>>> x
array([-1.62434536e+00, -6.11756414e-01, -5.28171752e-01, -1.07296862e+00,
       8.65407629e-01, -2.30153870e+00,  1.74481176e+00, -7.61206901e-01,
       3.19039096e-01, -2.49370375e-01,  1.46210794e+00, -2.06014071e+00,
      -3.22417204e-01, -3.84054355e-01,  1.13376944e+00, -1.09989127e+00,
      -1.72428208e-01, -8.77858418e-01,  4.22137467e-02,  5.82815214e-01,
      -1.10061918e+00,  1.14472371e+00,  9.01590721e-01,  5.02494339e-01,
       9.008555949e-01, -6.83727859e-01, -1.22890226e-01, -9.35769434e-01,
      -2.67888080e-01,  5.30355467e-01, -6.91660752e-01, -3.96753527e-01,
      -6.87172700e-01, -8.45205641e-01, -6.71246131e-01, -1.26645989e-02,
      -1.11731035e+00,  2.34415698e-01,  1.65980218e+00,  7.42044161e-01,
      -1.91835552e-01, -8.87628964e-01, -7.47158294e-01,  1.69245460e+00,
       5.08077548e-02, -6.36995647e-01,  1.90915485e-01,  2.10025514e+00,
       1.20158952e-01,  6.17203110e-01,  3.00170320e-01, -3.52249846e-01,
      -1.14251820e+00, -3.49342722e-01, -2.08894233e-01,  5.86623191e-01,
       8.38983414e-01,  9.31102081e-01,  2.85587325e-01,  8.85141164e-01,
      -7.54397941e-01,  1.25286816e+00,  5.12929820e-01, -2.98092835e-01,
       4.88518147e-01, -7.55717130e-02,  1.13162939e+00,  1.51981682e+00,
```

Kernel density estimation



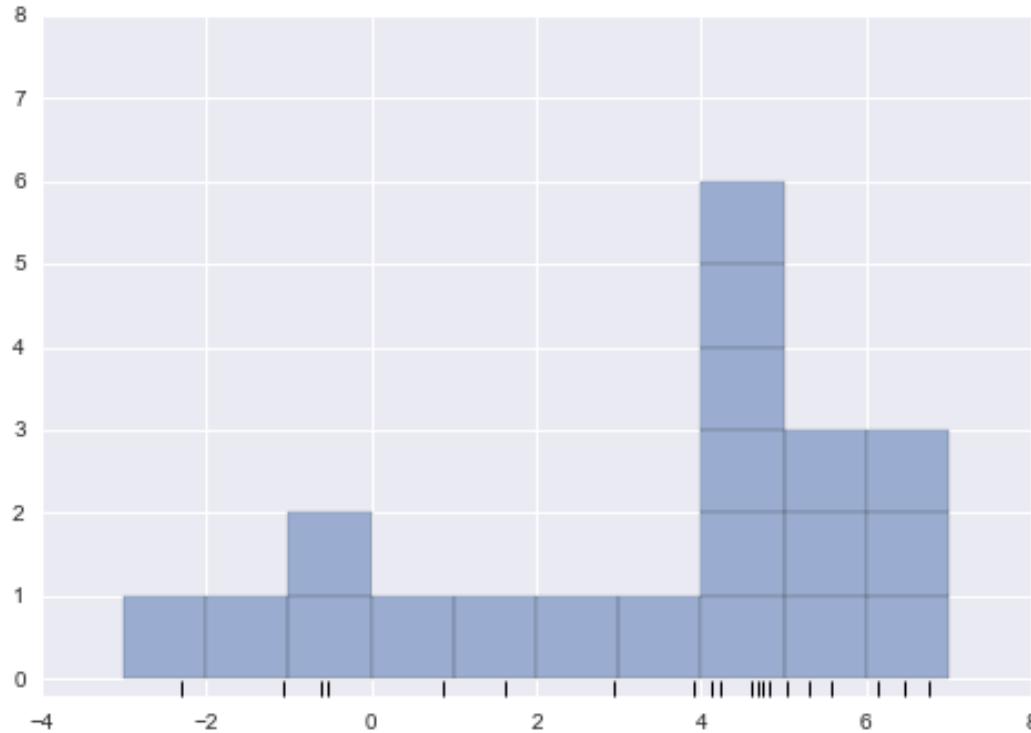
VanderPlas, J. 2016. *Python data science handbook: essential tools for working with data*.
O'Reilly Media, Inc.

Kernel density estimation



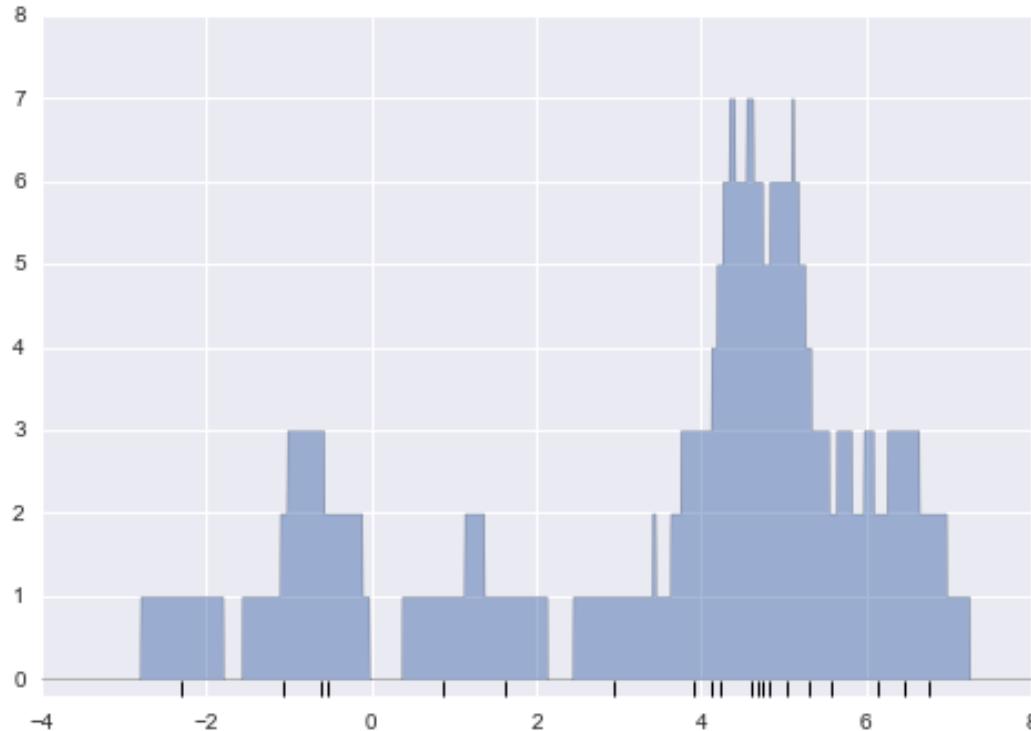
VanderPlas, J. 2016. *Python data science handbook: essential tools for working with data*. O'Reilly Media, Inc.

Kernel density estimation



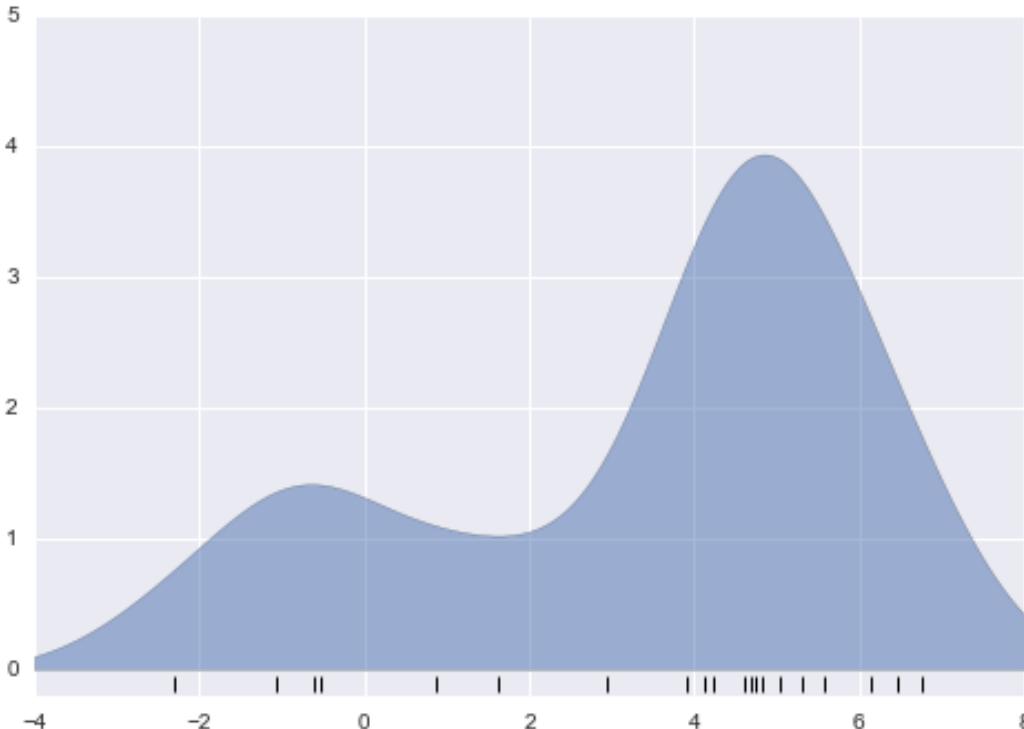
VanderPlas, J. 2016. *Python data science handbook: essential tools for working with data*.
O'Reilly Media, Inc.

Kernel density estimation



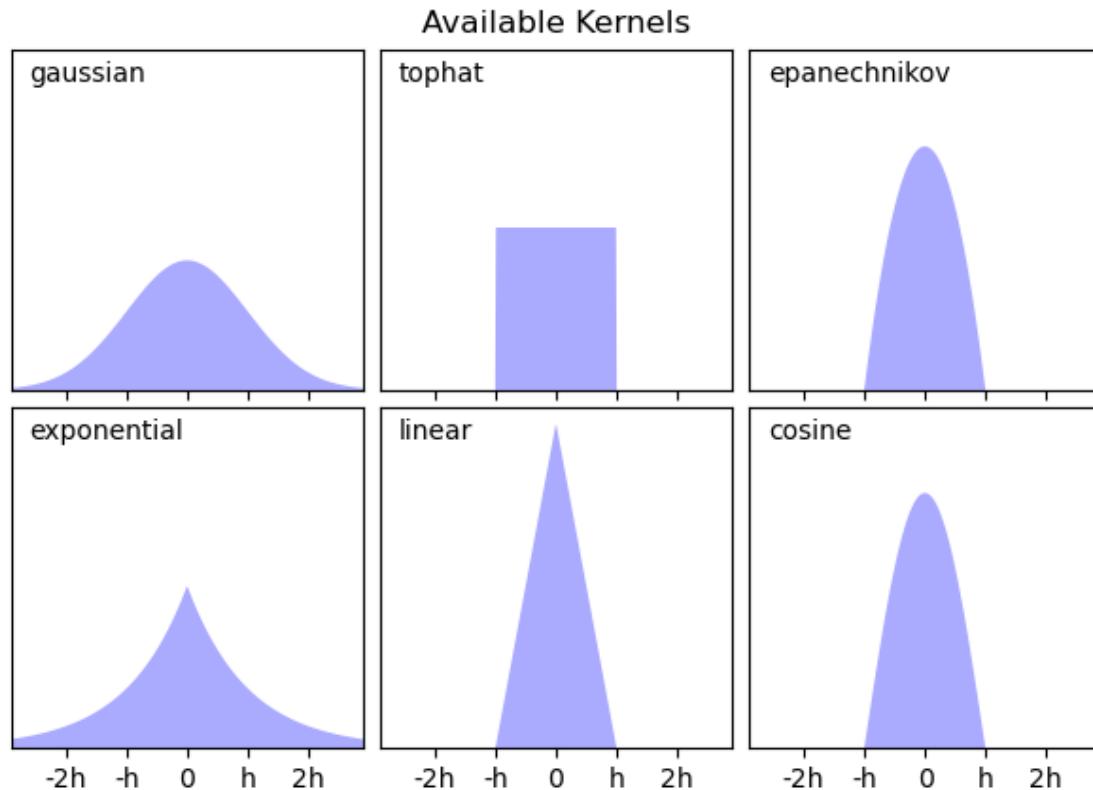
VanderPlas, J. 2016. *Python data science handbook: essential tools for working with data*.
O'Reilly Media, Inc.

Kernel density estimation



VanderPlas, J. 2016. *Python data science handbook: essential tools for working with data*.
O'Reilly Media, Inc.

Kernel density estimation



Scikit-learn. 2020. *Density estimation*. [online] <https://scikit-learn.org/stable/modules/density.html>

I UNDERSTAND

BUT WHY DO WE CARE?

makeameme.org

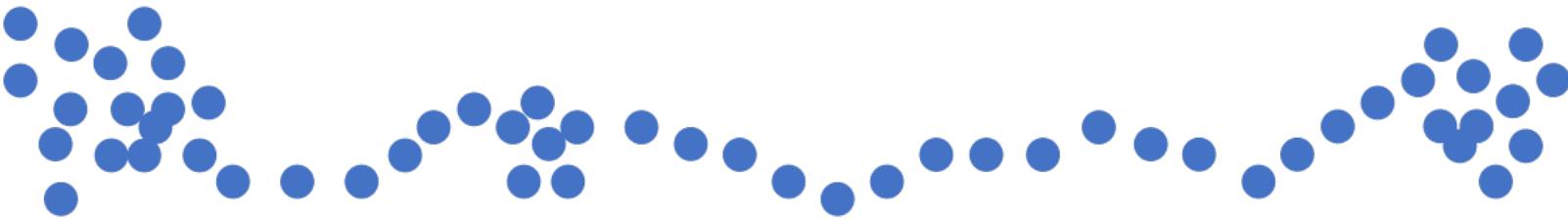
Practical examples

Two examples

Point pattern analysis 'in action' in some actual research:

- Not all point data sets can be fully analysed with summary measures, e.g. analysis of GPS trajectory data ('trajectory data').
- Some problems do not per se look like a 'point pattern analysis' problem, but actually can be conceived as such, e.g. surname profiling.

Trajectory data



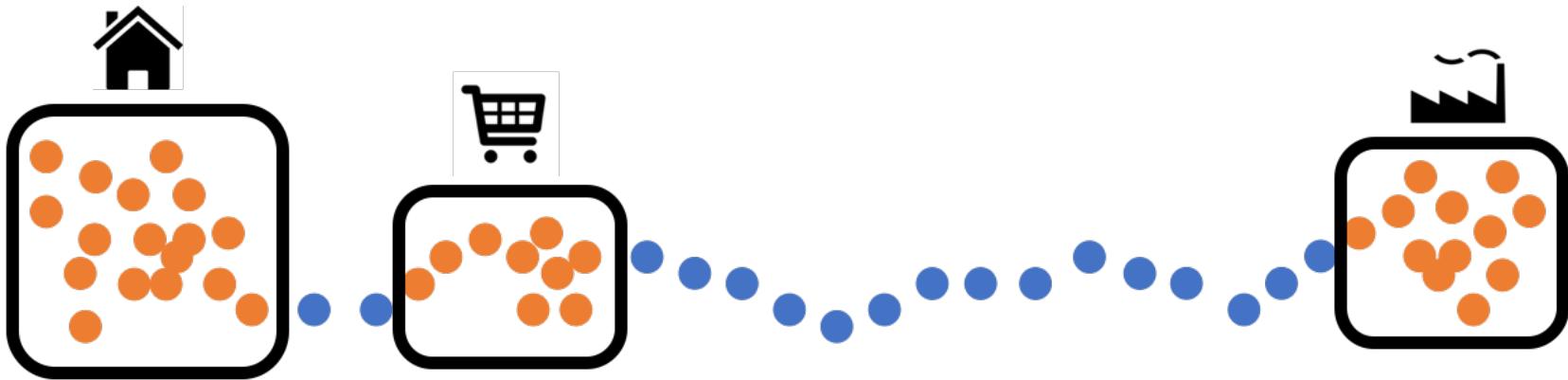
Van Dijk, J. T. 2018. Identifying activity-travel points from GPS-data with multiple moving windows.
Computers, Environment and Urban System 70: 84-101

Trajectory data

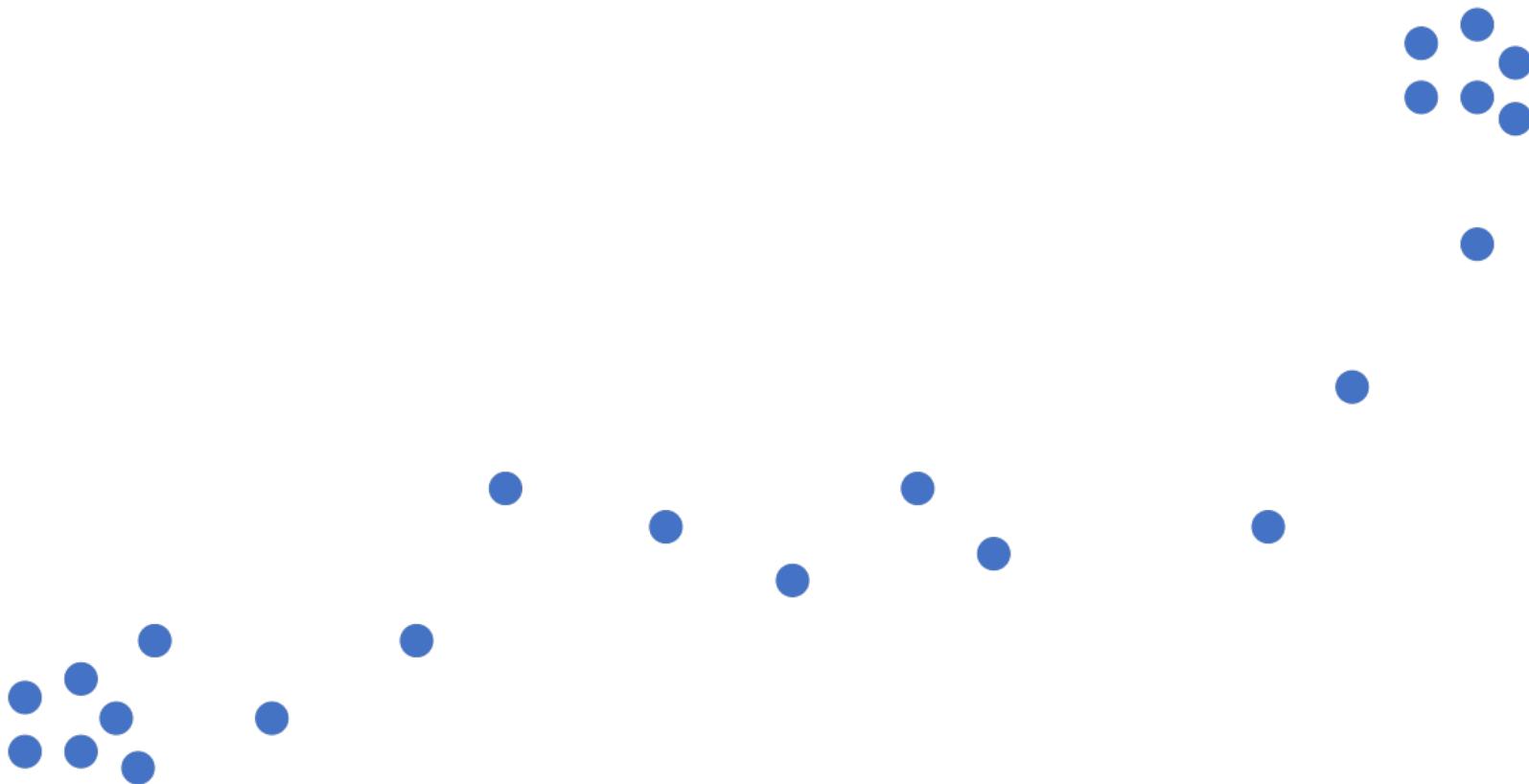


Van Dijk, J. T. 2018. Identifying activity-travel points from GPS-data with multiple moving windows.
Computers, Environment and Urban System 70: 84-101

Trajectory data

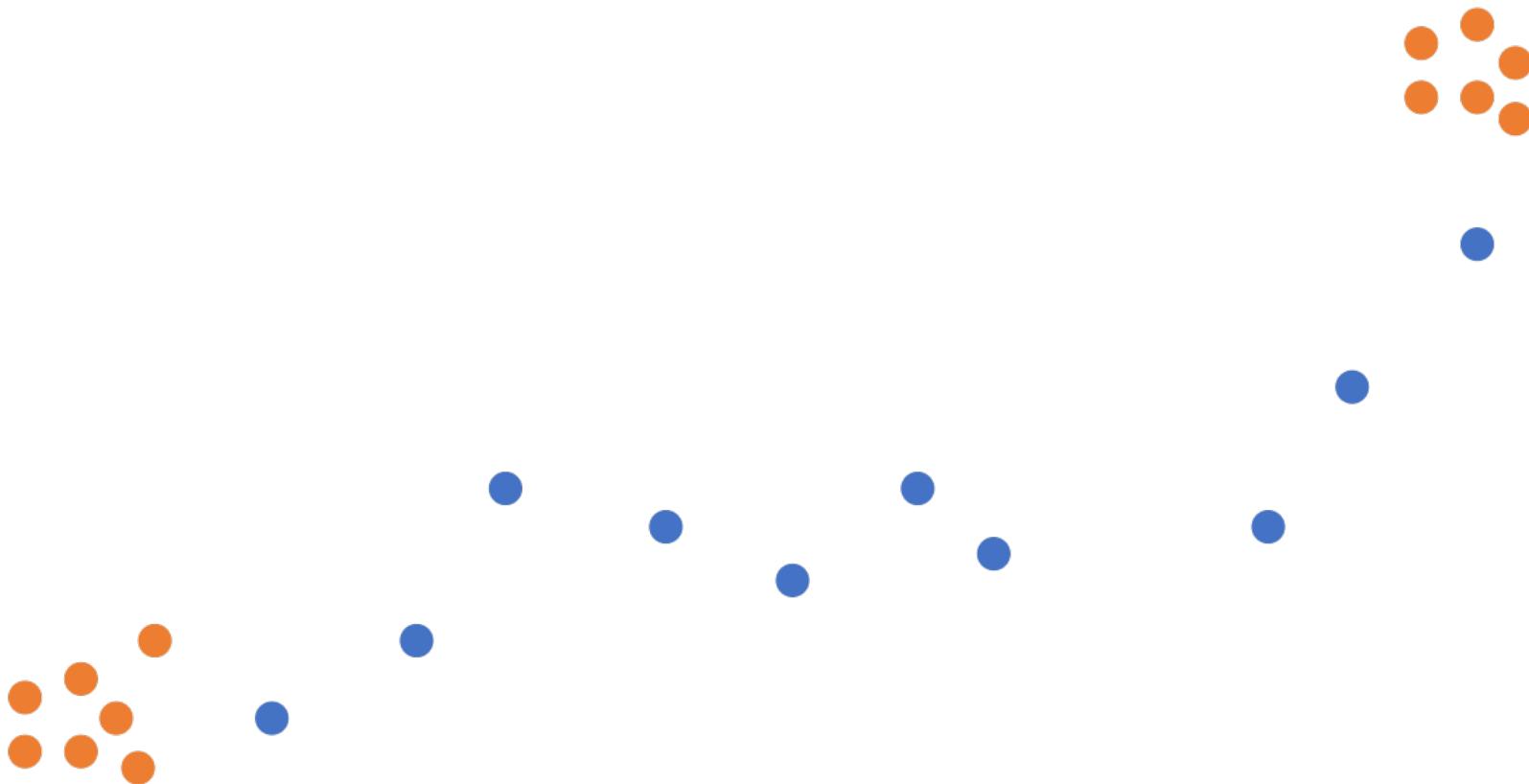


Trajectory data



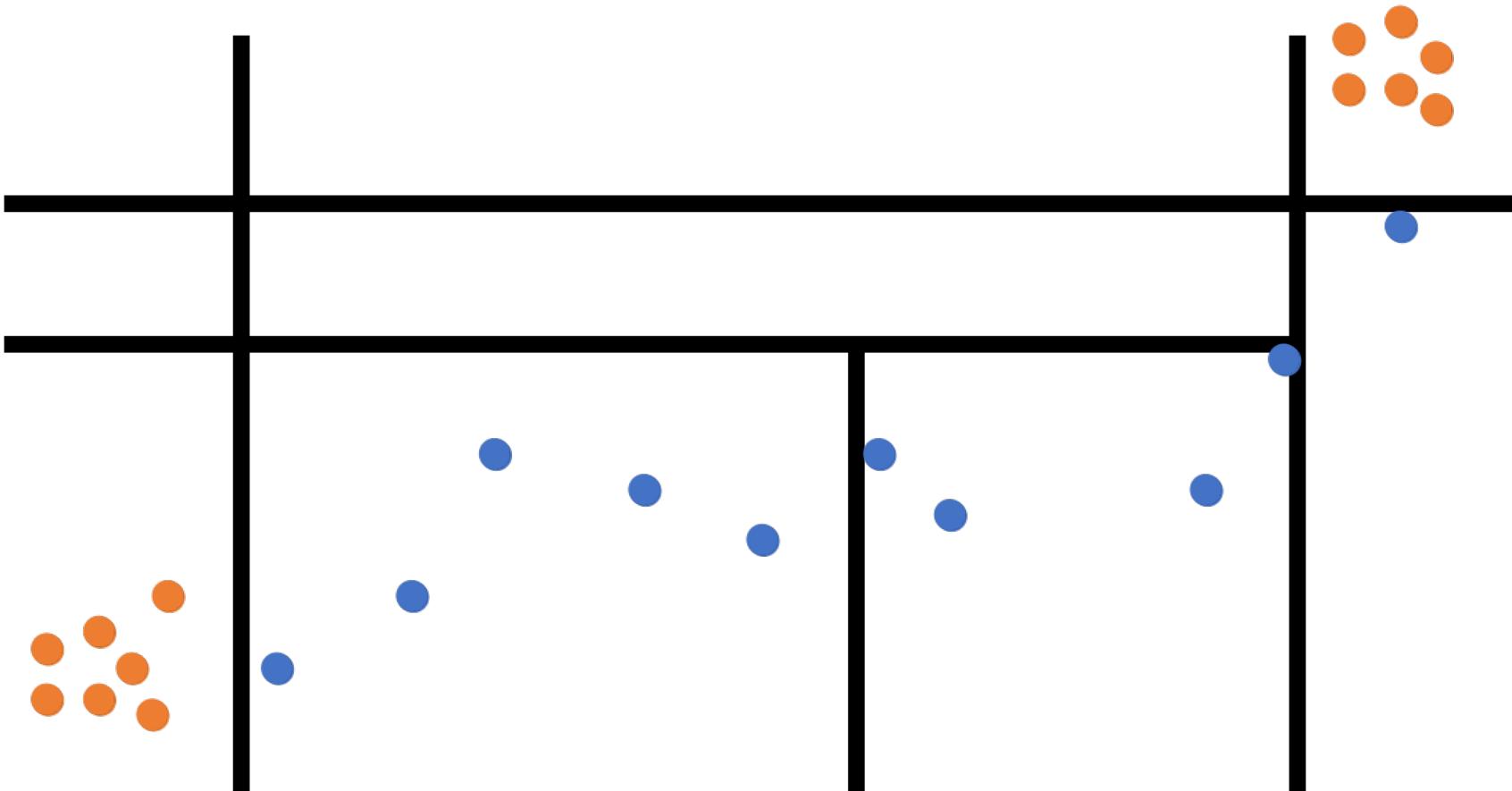
Van Dijk, J. T. & T. de Jong. 2017. Post-processing GPS tracks in reconstructing travelled routes in a GIS-environment: Network subset selection and attribute adjustment. *Annals of GIS* 23(3): 203-217

Trajectory data



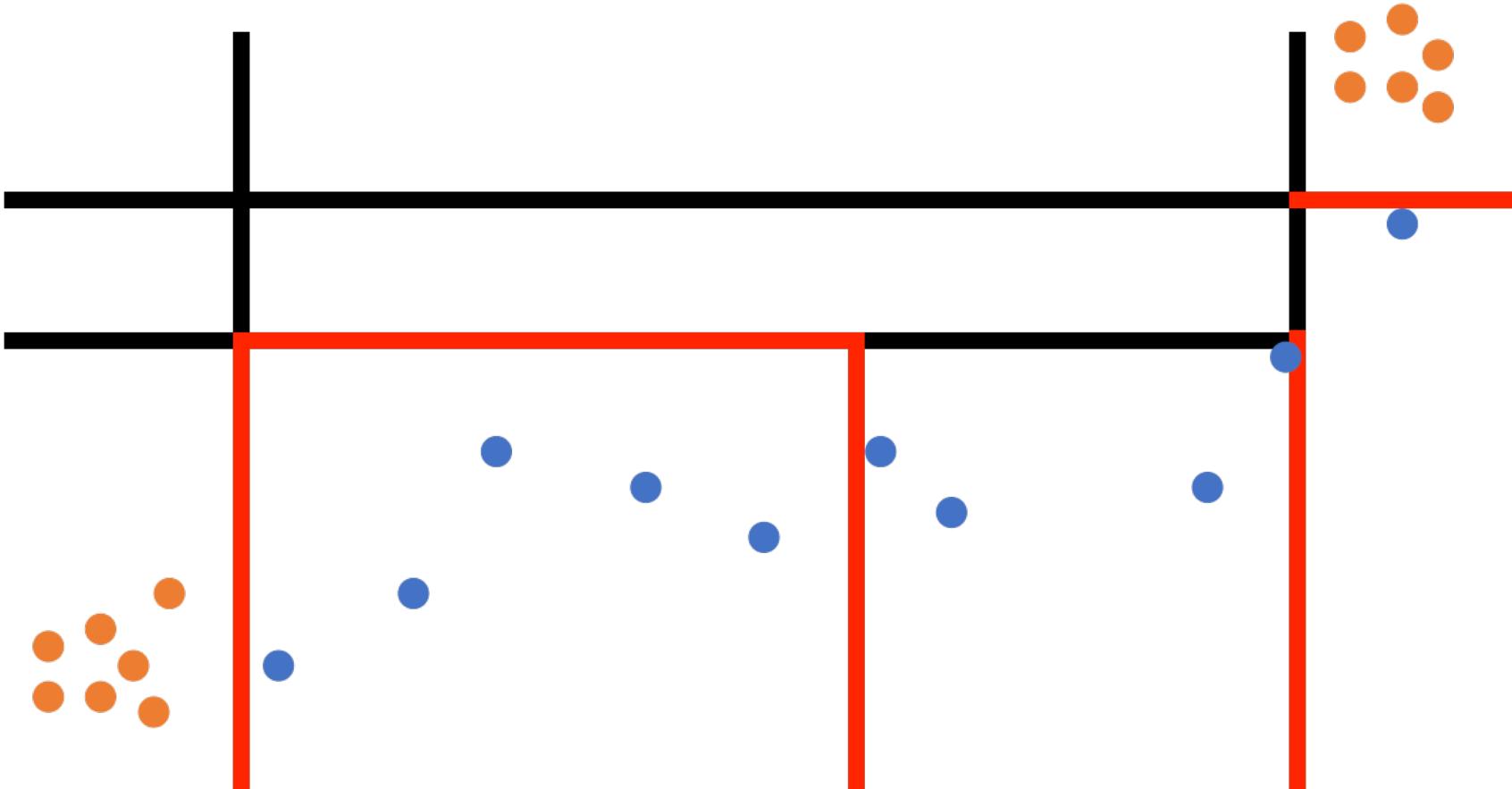
Van Dijk, J. T. & T. de Jong. 2017. Post-processing GPS tracks in reconstructing travelled routes in a GIS-environment: Network subset selection and attribute adjustment. *Annals of GIS* 23(3): 203-217

Trajectory data



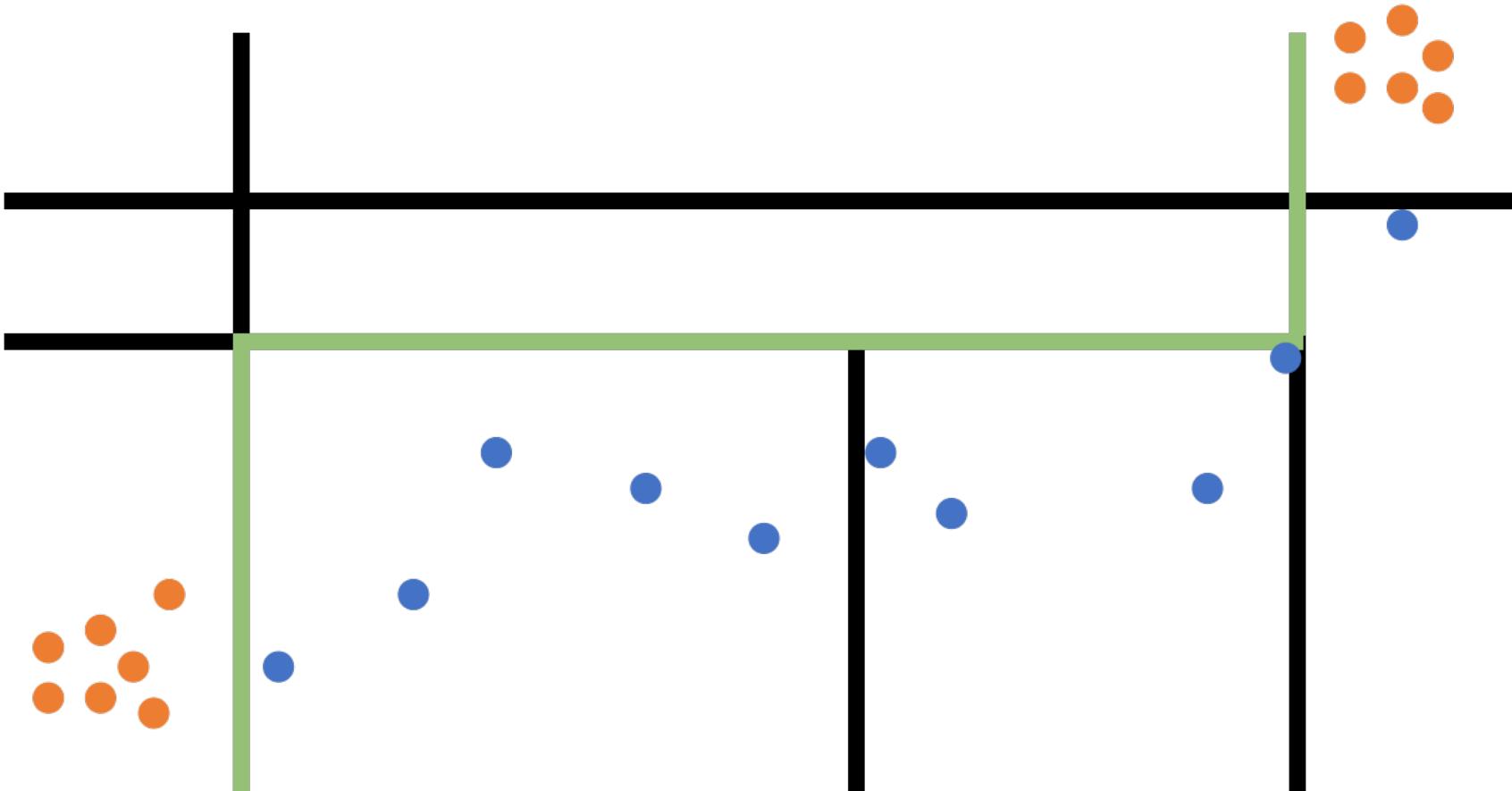
Van Dijk, J. T. & T. de Jong. 2017. Post-processing GPS tracks in reconstructing travelled routes in a GIS-environment: Network subset selection and attribute adjustment. *Annals of GIS* 23(3): 203-217

Trajectory data



Van Dijk, J. T. & T. de Jong. 2017. Post-processing GPS tracks in reconstructing travelled routes in a GIS-environment: Network subset selection and attribute adjustment. *Annals of GIS* 23(3): 203-217

Trajectory data



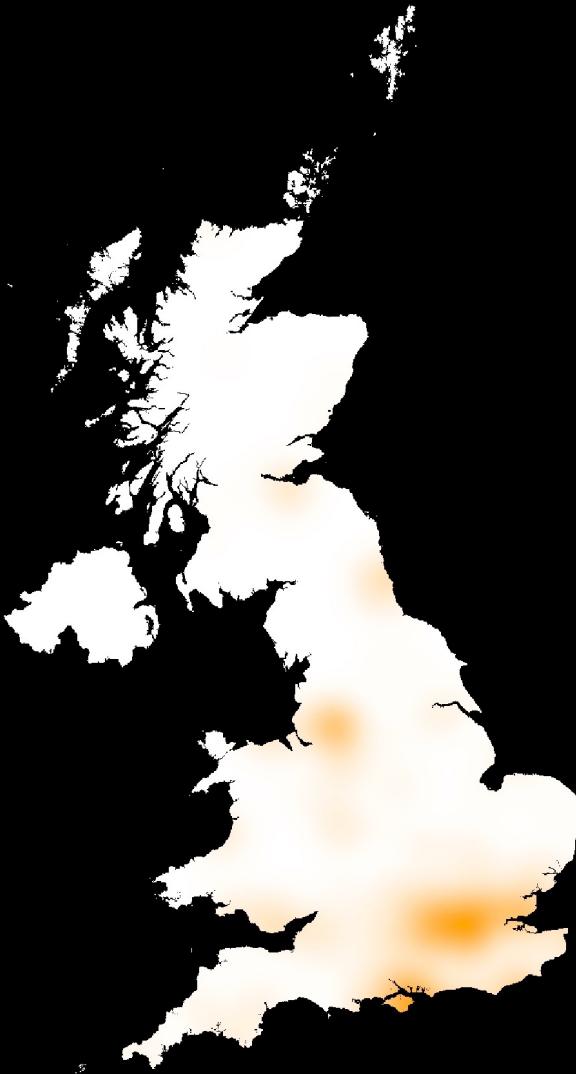
Van Dijk, J. T. & T. de Jong. 2017. Post-processing GPS tracks in reconstructing travelled routes in a GIS-environment: Network subset selection and attribute adjustment. *Annals of GIS* 23(3): 203-217

Surname profiling

- Personal names contain socio-demographic information.
- But also: many surnames contain spatial information at a variety of scales relating to the origins of many of their bearers.
- Data: Historic Census of Population 1851-1911, Consumer Registers 1997– 2016.
- Total of 1.2 million surnames with locations (Historic Parishes, unit postcodes and geo-coded addresses for several years of data).



"Van Dijk"



“Lansley”



“Rossall”

Surname profiling

Combination on various point pattern analysis techniques:

- kernel densities to map the surname concentrations of names found in Great Britain executed over a 1000m x 1000m grid.
- deconstruction of grids as sparse matrices to optimise storage and database retrieval (storing 1.2 million KDEs is challenging), followed by DBSCAN to create contours of highest relative density (vectorisation).

| | | |
|---|------------|------------|
| 0 | 0 | 0 |
| 0 | 0.5 | 0.9 |
| 0 | 0.7 | 0 |
| 0 | 0 | 0 |

1

2

| | | |
|---|-----------|-----------|
| 0 | 0 | 0 |
| 0 | 50 | 90 |
| 0 | 70 | 0 |
| 0 | 0 | 0 |

3

| |
|-----------|
| 0 |
| 0 |
| 0 |
| 0 |
| 0 |
| 50 |
| 90 |
| 70 |
| 0 |
| 0 |
| 0 |
| 0 |

4

5

| | |
|----|-----------|
| 1 | 0 |
| 2 | 0 |
| 3 | 0 |
| 4 | 0 |
| 5 | 0 |
| 6 | 50 |
| 7 | 90 |
| 8 | 70 |
| 9 | 0 |
| 10 | 0 |
| 11 | 0 |
| 12 | 0 |

6,7,8;50,90,70

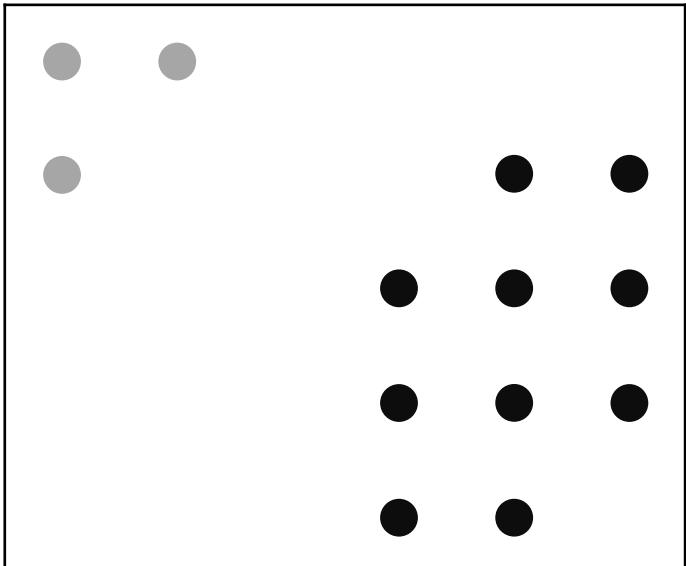
1

| | | | | | |
|-----------|-----------|----|-----------|-----------|-----------|
| 50 | 90 | 20 | 0 | 10 | 10 |
| 80 | 10 | 0 | 20 | 40 | 50 |
| 30 | 0 | 10 | 40 | 60 | 80 |
| 0 | 0 | 20 | 50 | 70 | 50 |
| 0 | 0 | 30 | 50 | 90 | 30 |

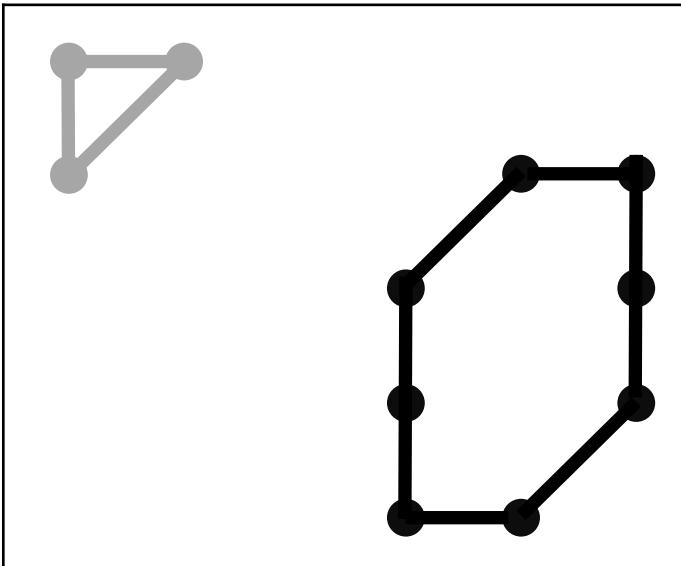
2

| | | | | | |
|-----------|-----------|--|--|-----------|-----------|
| 50 | 90 | | | | |
| 80 | | | | 40 | 50 |
| | | | | 40 | 60 |
| | | | | 60 | 80 |
| | | | | 50 | 70 |
| | | | | 50 | 90 |

3



4

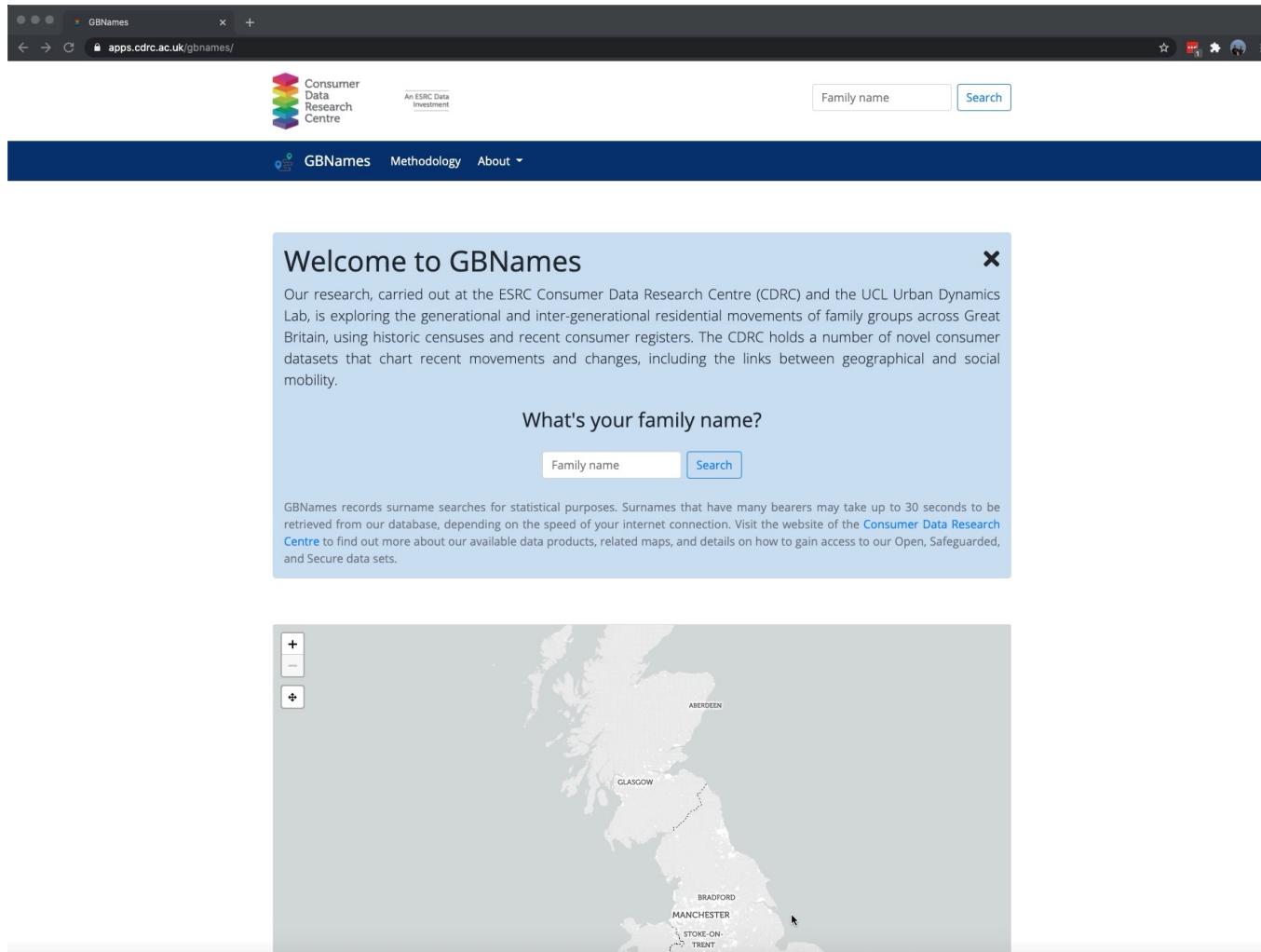


Surname profiling

Why does the DBSCAN work?

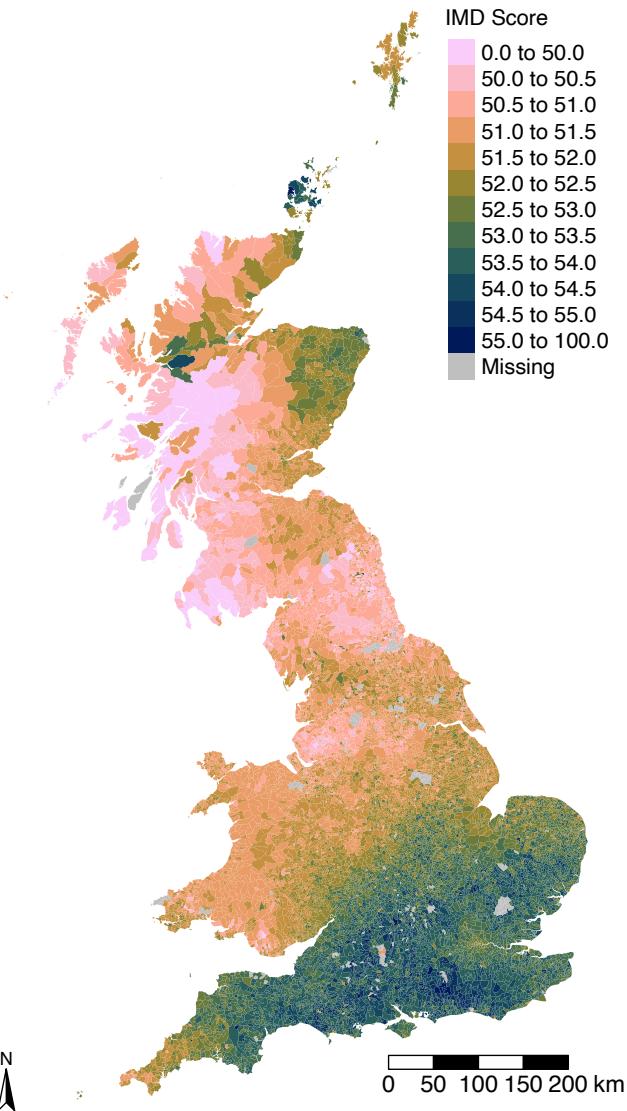
- Because the 1000×1000 m grid resolution never changes, we know that two adjacent grid centre points are always within a distance of approximately 1415 metres and we can thus use a distance constrained neighbourhood search.
- It is fast.

Surname profiling



Van Dijk, J. T. & P. A. Longley. 2020. Interactive display of surname distributions in historic and contemporary Great Britain. *Journal of Maps* 16(1): 68-76

Surname profiling



Longley, P. A., Van Dijk, J. T., & Lan, T. 2021. The geography of intergenerational social mobility in Great Britain. *Nature Communications* 12: 6050.

Conclusion

- Point pattern analysis rather than aggregation to some type of administrative geography – we talked through several techniques today.
- Different approaches possible depending on the type of question you try to answer (e.g. characterisation of a point process versus cluster identification).
- Not all data may present themselves as clear candidates for these types of analyses.

Questions

Justin van Dijk

j.t.vandijk@ucl.ac.uk

