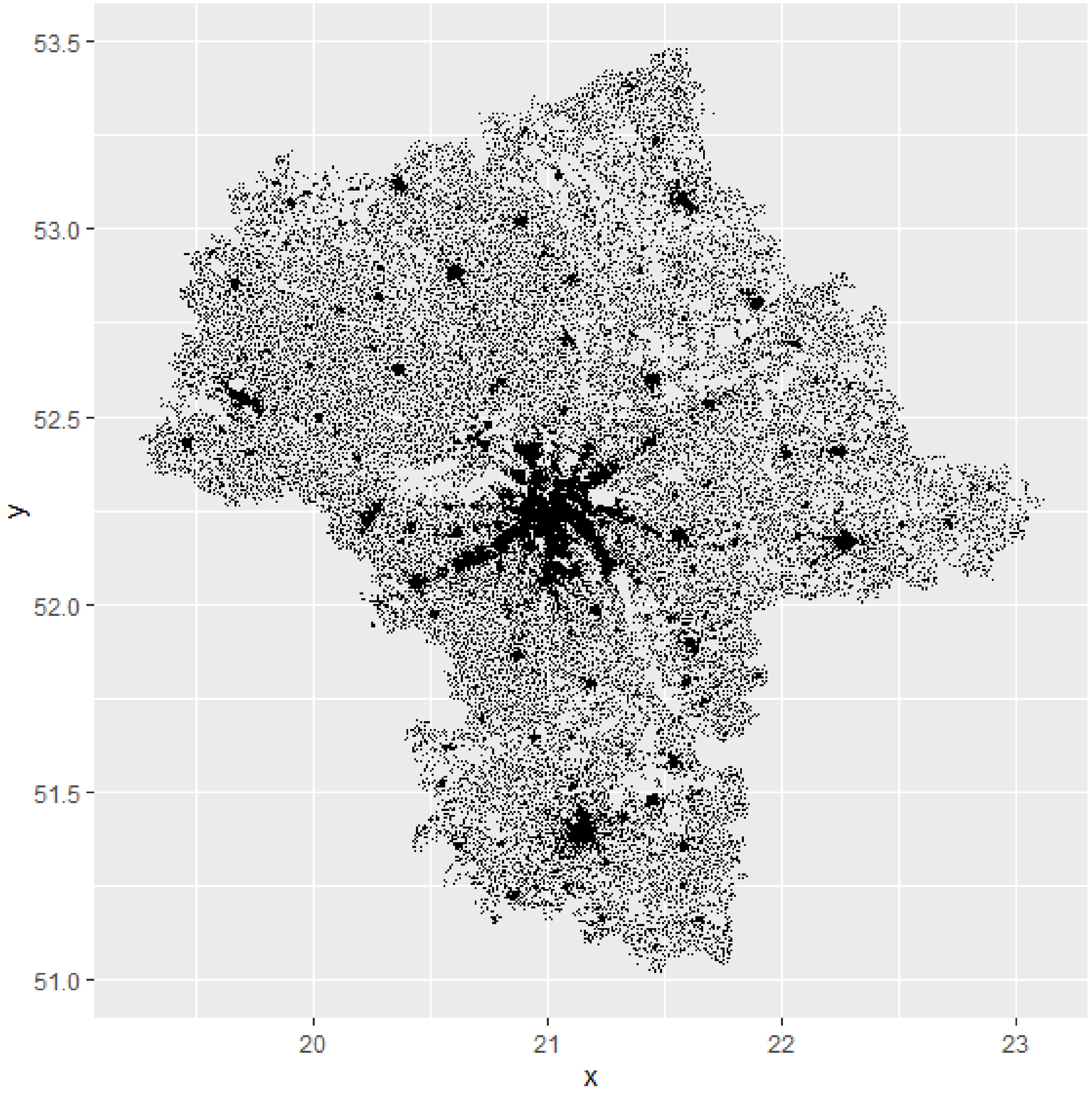
QDC: Quick Density Clustering of geo-located data

PRESENTATION OF THE NEW ALGORITHM

Katarzyna Kopczewska

Faculty of Economic Sciences
University of Warsaw, Poland

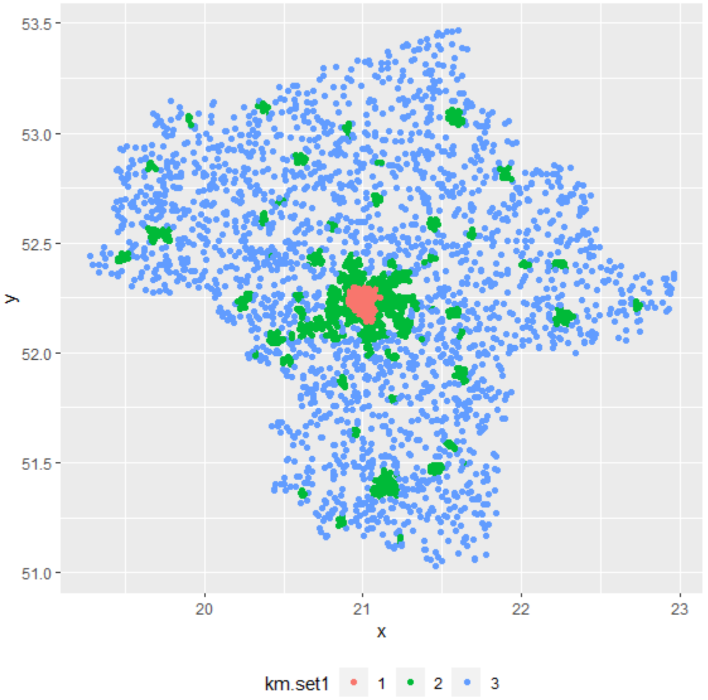


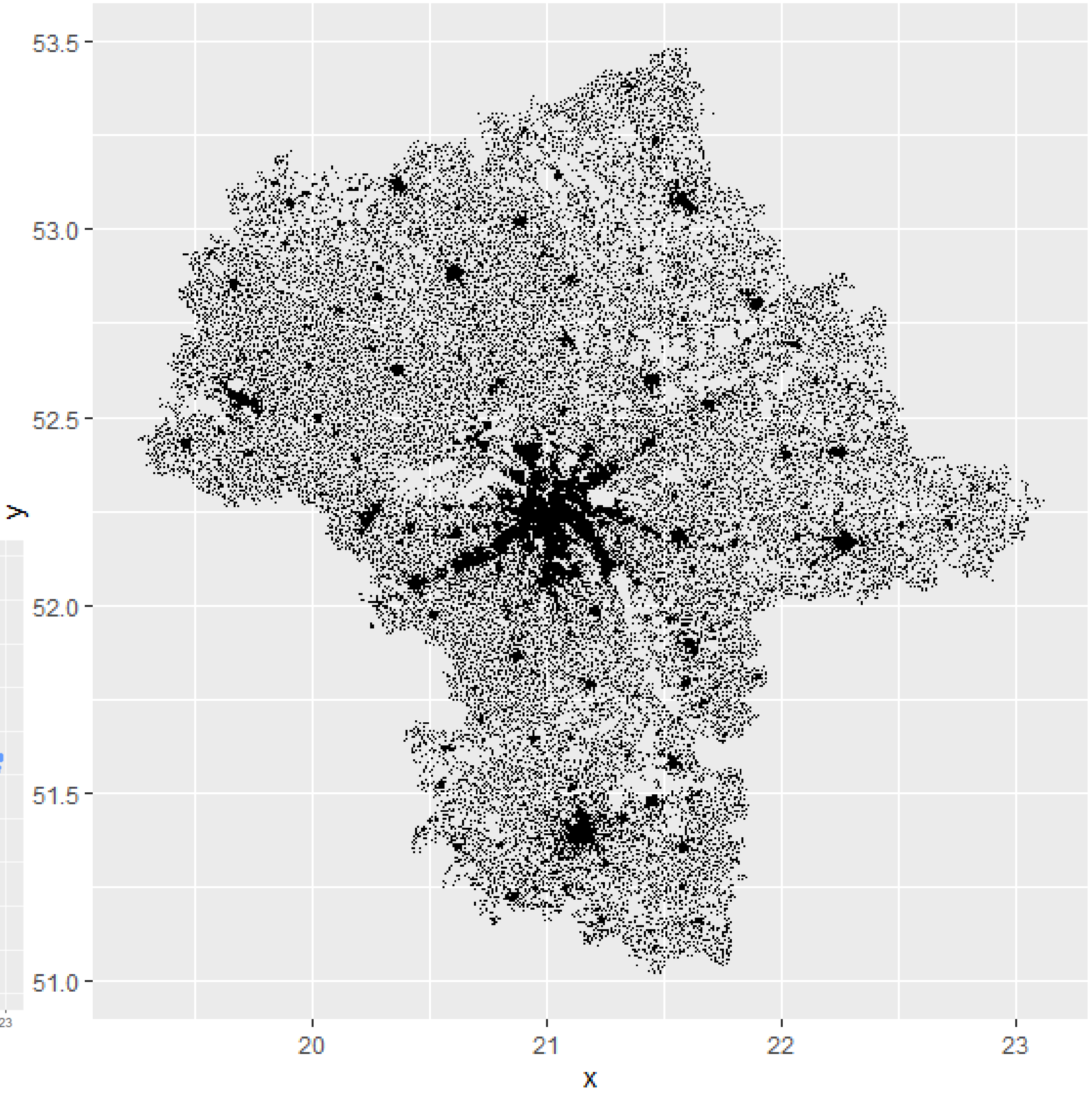
Problem to solve

For a given dataset, assign quickly each point with one of the labels:

- ✓high-density,
- ✓medium-density
- ✓low-density

We want to obtain something like this





Motivation – WHY we need a new method?

- To avoid metodological problems
  - we mostly deal with aggregated data to describe density of territory (e.g. persons / km^2) → this is not applicable to point data
  - to avoid MAUP (Modifiable Areal Unit Problem) when aggregating data
- To answer new research questions
  - to give the ‚label’ to data in terms of its relative location – adding information (another feature) to the dataset
  - to study better the issues of agglomeration, concentration, CBD effects, co-location issues, especially in micro-geography studies
- For monitoring
  - to know if local density structure changes immediately – e.g. to track crowd, traffic
  - for policy insights – to understand human activity at individual (not aggregated) level

Toolbox for density clusters

1 **DBSCAN / Density Peak Clustering**

- points labelled as high-density and noise only (no other levels)
- Poor control over the number of clusters and noise ratio
- Very sensitive to parameters set by users

2 **Gridded data**

- labelling of grid cells (counting aggregated points) – no more for points

3 **Kernel Density Estimation (KDE)**

- Continous values of density
- Very sensitive to parameters set by users

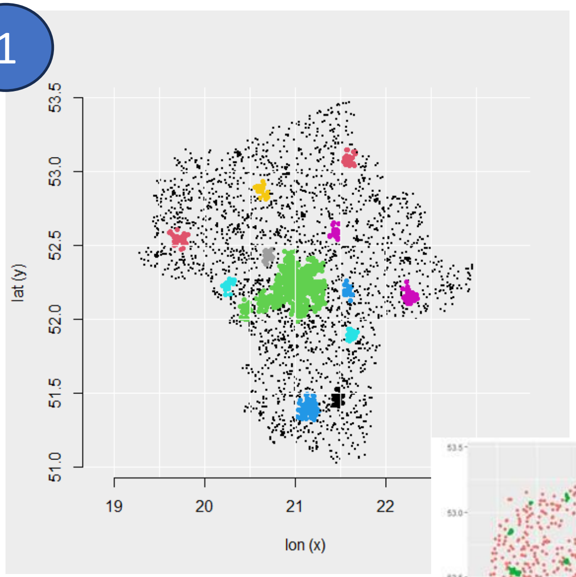
4 **K-means for xy coordinates**

- It outputs the catchment areas, not density clusters

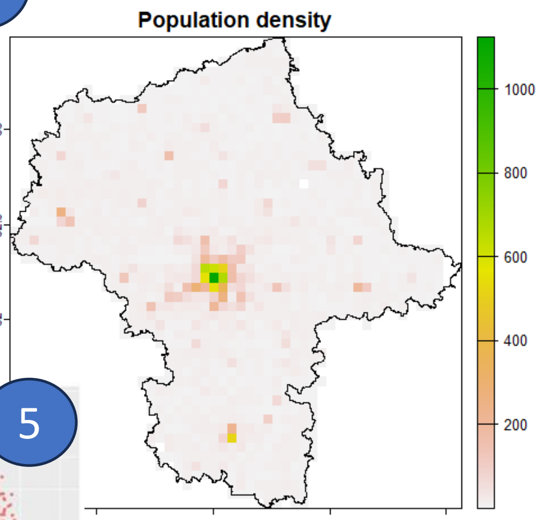
5 **QDC (Quick Density Clustering)**

- Uses distance to kNN and fixed-radius NN
- Yields low, mid and high-density clusters

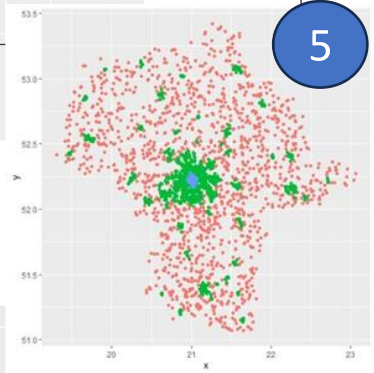
1



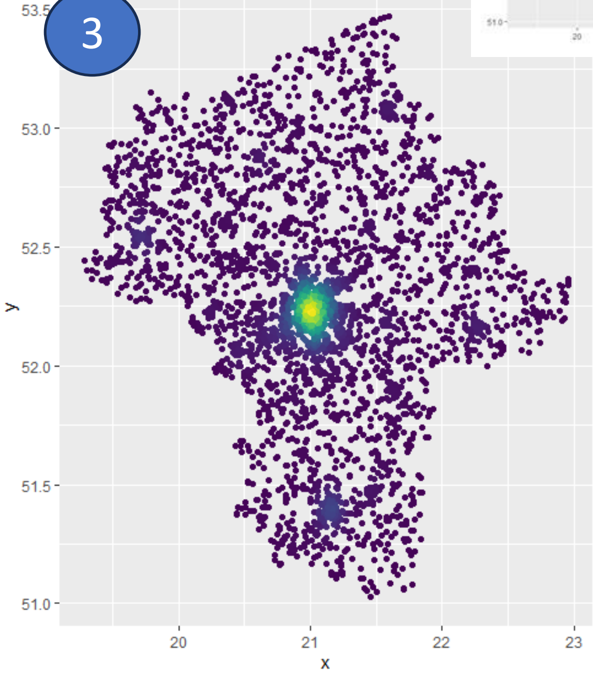
2



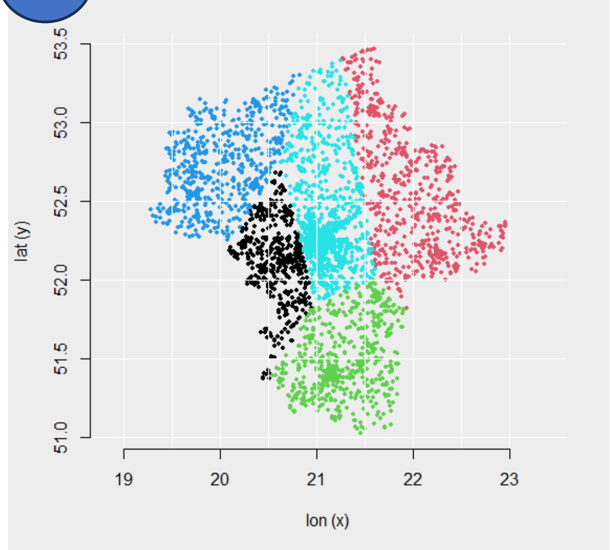
5



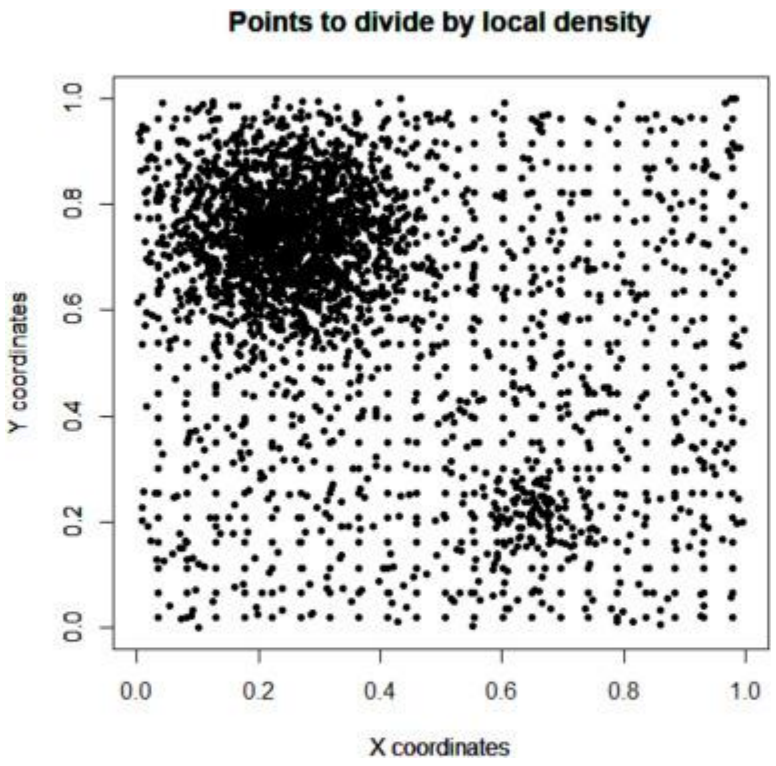
3

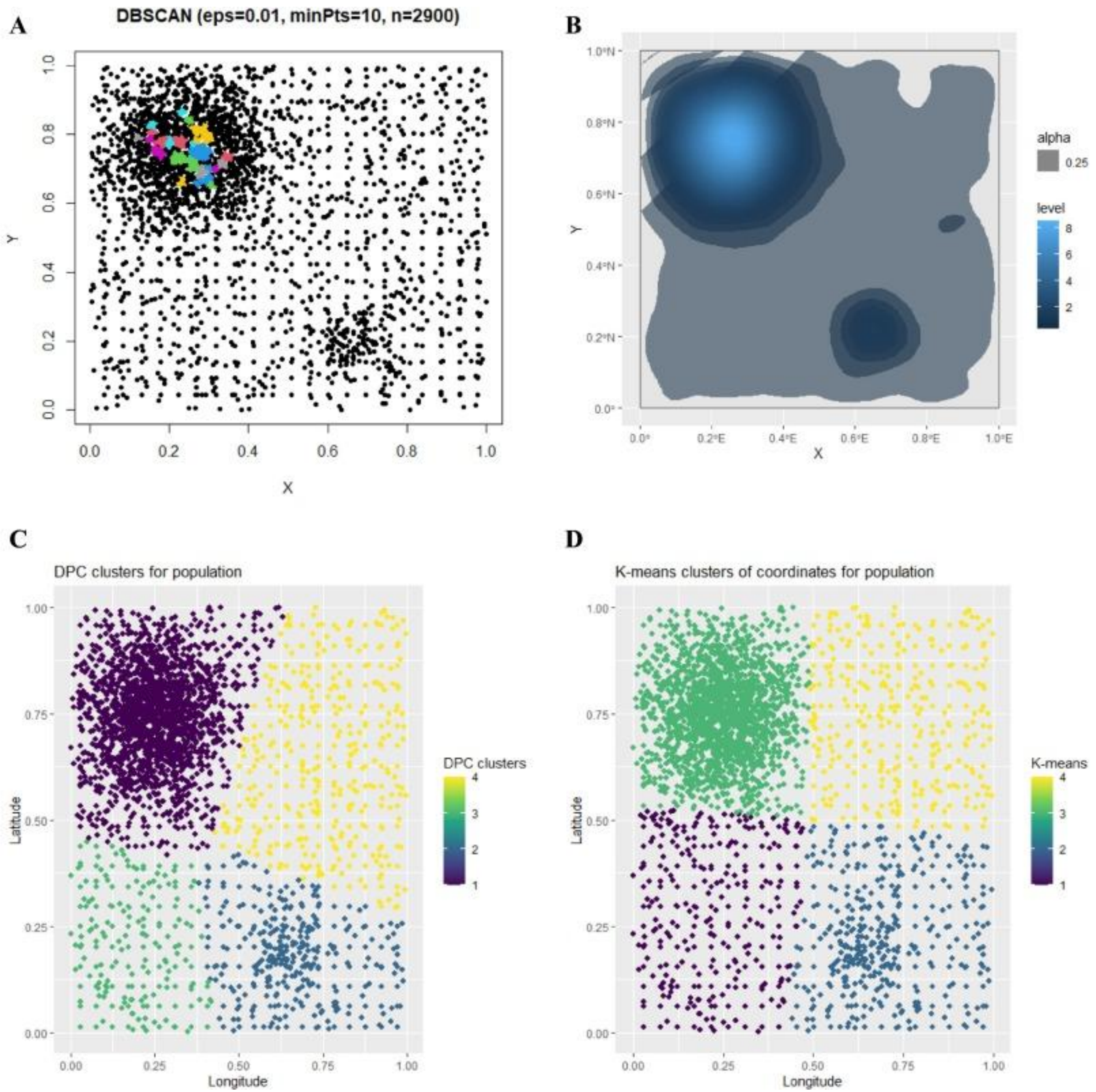


4



Existing metods fail also in simulated designs



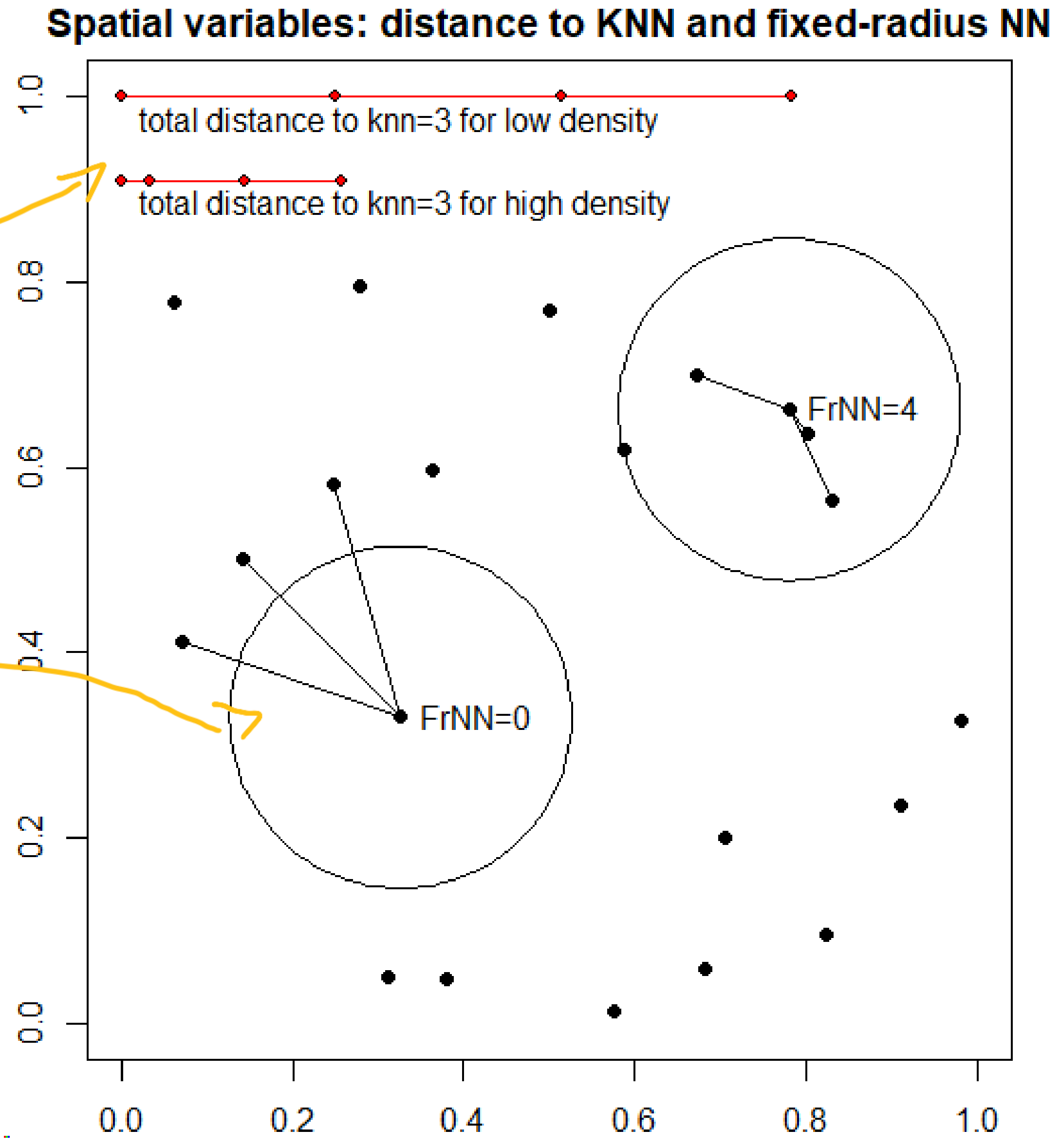


The design of QDC (Quick Density Clustering) (1)

- Intuition to follow the nature of density:
  - **Low-density points:**
    - Far from other points
    - Only a few points around
  - **High-density points:**
    - Close to other points
    - Many points around

QDC is based on two spatial variables:
- **Total distance to k nearest neighbours (knn)**
- **Number of neighbours within a fixed radius (frNN)**

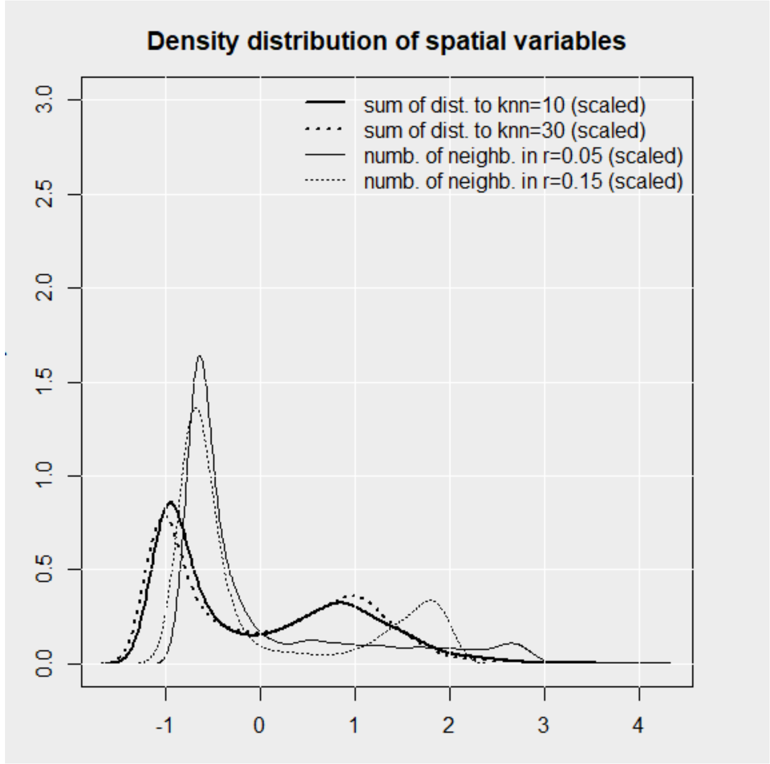
- New solution (Kopczewska, 2025) for dividing data by density, e.g. into 3 density clusters (high, medium, low)
- Weakly-dependent on hyper-parameters
- Self-calibrating – no need to set the thresholds



The design of QDC (Quick Density Clustering) (2)

Those two spatial variables:

- they carry different information
- there is no linear correlation between them
- they are almost independent of parameters:
  - number of k nearest neighbours (kNN) we use
  - size of radius in which we count neighbours (frNN)



What we can do with two non-linearly correlated variables???

➔ To avoid scale effect one should **standardise/normalise** both variables.
➔ We can use **k-means clustering** to divide observations into **k groups** (clusters) based on those two spatial variables – it is based on distance (dissimilarity measure)

QDC construction

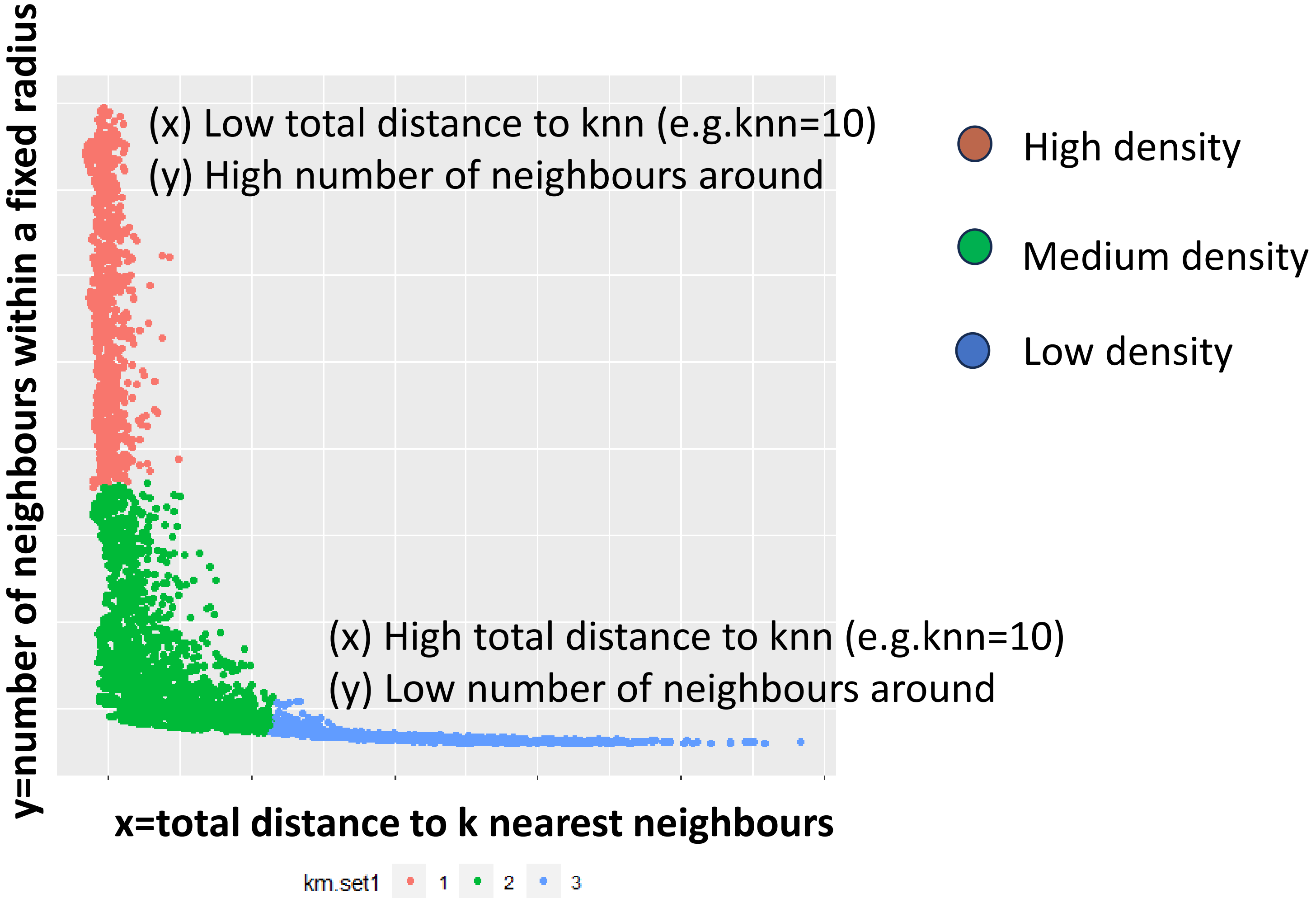
QDC algorithm is executed in the following steps:

1. Calculate for each point the number of nearest neighbours within a fixed radius and normalise
2. Calculate for each point the total distances to k nearest neighbours and normalise
3. Perform k-means clustering on both variables together
4. Obtain the thresholds of the clusters to classify new points

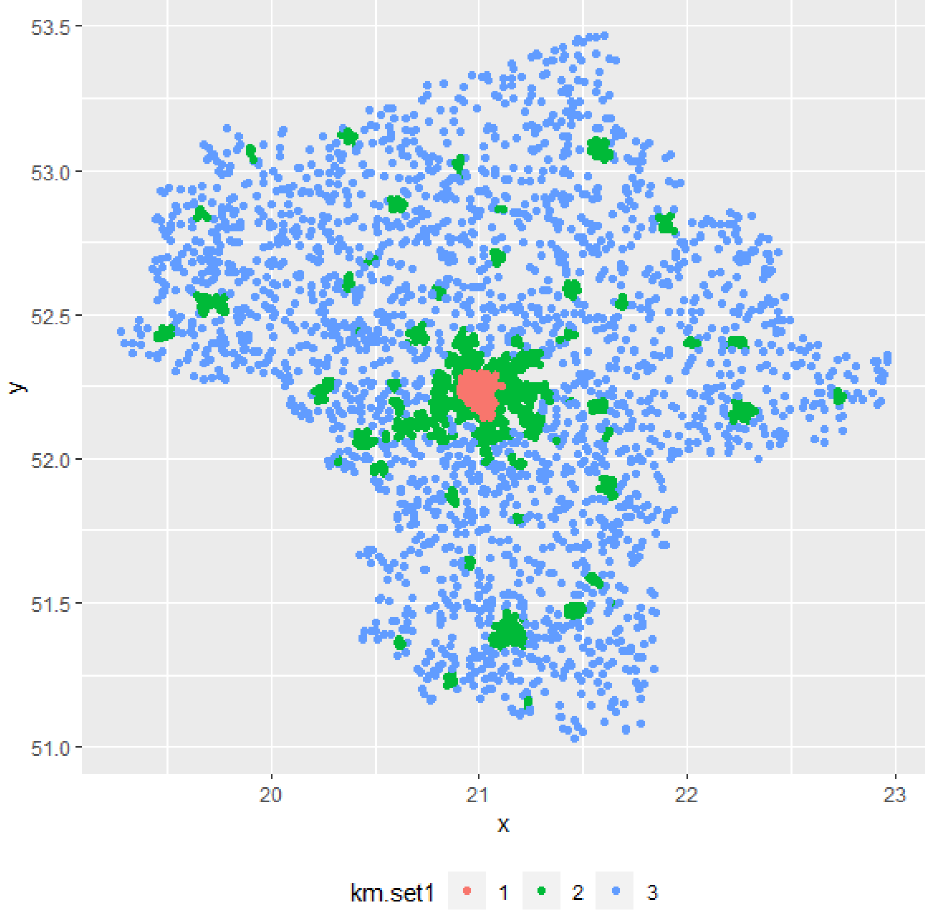
	X	Y	knndist1	frnn1	knndist1.scaled	frnn1.scaled	km.set1	outcome.set1
1	20.73643	52.42706	0.15232063	2	-0.3560706	-0.6624440	3	mid-density
2	21.11411	52.31913	0.10447334	41	-0.5306375	-0.4888737	3	mid-density
3	21.04562	52.14307	0.02639201	236	-0.8155109	0.3789779	3	mid-density
4	21.00613	52.19675	0.02936217	714	-0.8046746	2.5063270	2	high-density
5	20.77757	51.77048	0.33446133	3	0.3084551	-0.6579935	3	mid-density
6	20.67792	52.11957	0.10465465	29	-0.5299761	-0.5422800	3	mid-density

The design of QDC (Quick Density Clustering) (3)

Relation between the two spatial variables



Location of detected clusters (x=longitude, y=latitude)



QDC algorithm

CLUSTERING

k=hyper-parameter, e.g.30

K=hyper-parameter, e.g. 3

r=hyper-parameter, e.g. 0.15

spat.var1 ← ∑dist(knn=k)

spat.var2 ← frnn(r)

spat.var1.s ← (spat.var1-mean(spat.var1))/sd(spat.var1)

spat.var2.s ← (spat.var2-mean(spat.var2))/sd(spat.var2)

data ← (spat.var1.s, spat.var1.s)

kmeans(data, K)

CLASSIFYING

t1←max(min(spat.var1|clust1), …., min(spat.var1|clustK))

t1←max(min(spat.var2|clust1), …., min(spat.var2|clustK))

low-density ←spat.var1>t1

high-density← spat.var2>t2

R Code to execute the whole algorithm

```
popul<-data.frame(x=X, y=Y)                               # dataset
knn.dist<-dbscan::kNNdist(popul[,1:2], 10, all =TRUE)  # spatial variables based on xy
popul$knndist1<-apply(knn.dist,1,sum)                    # sum of distances to knn
agg.radius<-dbscan::frNN(as.matrix(popul[,1:2]), eps=0.05)    # neighbours in radius
popul$frnn1<-unlist(lapply(agg.radius$id, length))            # count nn in fixed radius

popul$knndist1.scaled<-scale(popul$knndist1)             # normalisation
popul$frnn1.scaled<-scale(popul$frnn1)
popul$km.set1<-as.factor(kmeans(popul[ ,5:6], 3)$cluster)     # kmeans clustering

t1<-max(min(popul$knndist1.scaled[popul$km.set1==1]),
min(popul$knndist1.scaled[popul$km.set1==2]), min(popul$knndist1.scaled[popul$km.set1==3])) #
threshold, when knndist>t1 – it is low-density cluster
t2<-max(min(popul$frnn1.scaled[popul$km.set1==1]),min(popul$frnn1.scaled[popul$km.set1==2]),
min(popul$frnn1.scaled[popul$km.set1==3]))  # threshold, when frnn(agg)>t2 – it is high-density
cluster

popul$outcome.set1<-ifelse(popul$knndist1.scaled>t1, "low-density",
ifelse(popul$frnn1.scaled>t2,"high-density", "mid-density"))  # classifies points to clusters

ggplot(popul, aes(x=knndist1.scaled, y=frnn1.scaled, color=km.set1)) +  geom_point() # xy plot
ggplot(popul, aes(x=x, y=y, color=km.set1)) +  geom_point()    # location of clusters
```

Rand Index for alternative scenarios of QDC parameters

Rand Index for QDC clusters	knn=10 r=0.01	knn=100 r=0.01	knn=250 r=0.01	knn=10 r=0.1	knn=100 r=0.1	knn=250 r=0.1
knn=10 r=0.01	NA	0.880	0.934	0.806	0.787	0.744
knn=100 r=0.01	0.880	NA	0.897	0.822	0.833	0.791
knn=250 r=0.01	0.934	0.897	NA	0.781	0.798	0.762
knn=10 r=0.1	0.806	0.822	0.781	NA	0.953	0.912
knn=100 r=0.1	0.787	0.833	0.798	0.953	NA	0.919
knn=250 r=0.1	0.744	0.791	0.762	0.912	0.919	NA

$$RI = \frac{a+b}{a+b+c+d}$$

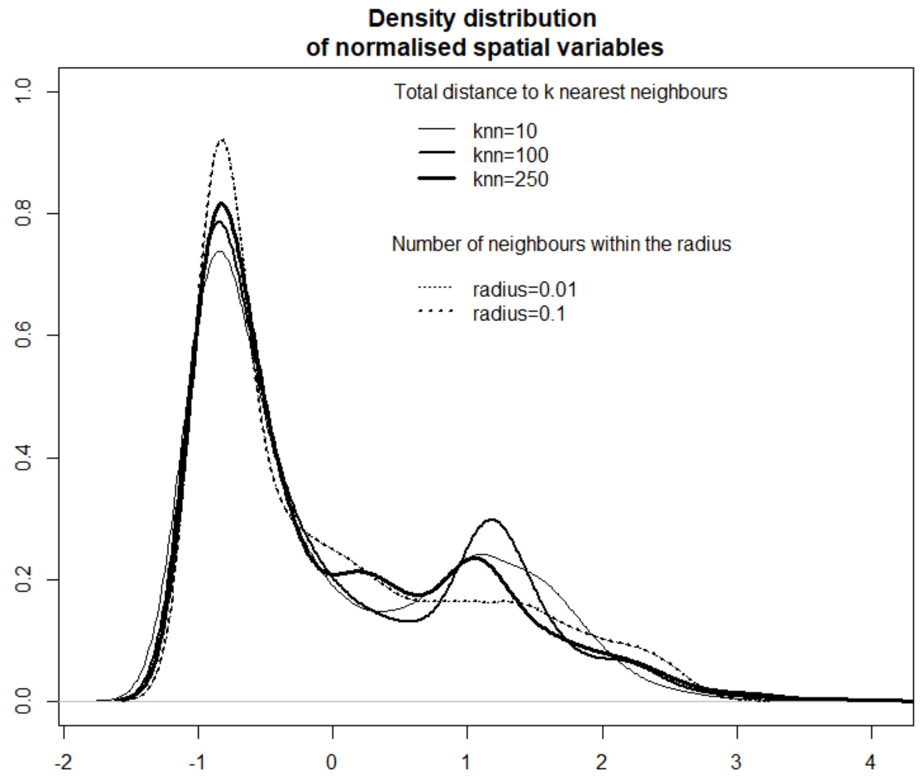
a, b, c, d are the numbers of the possible situations for pairs of pairs of points:

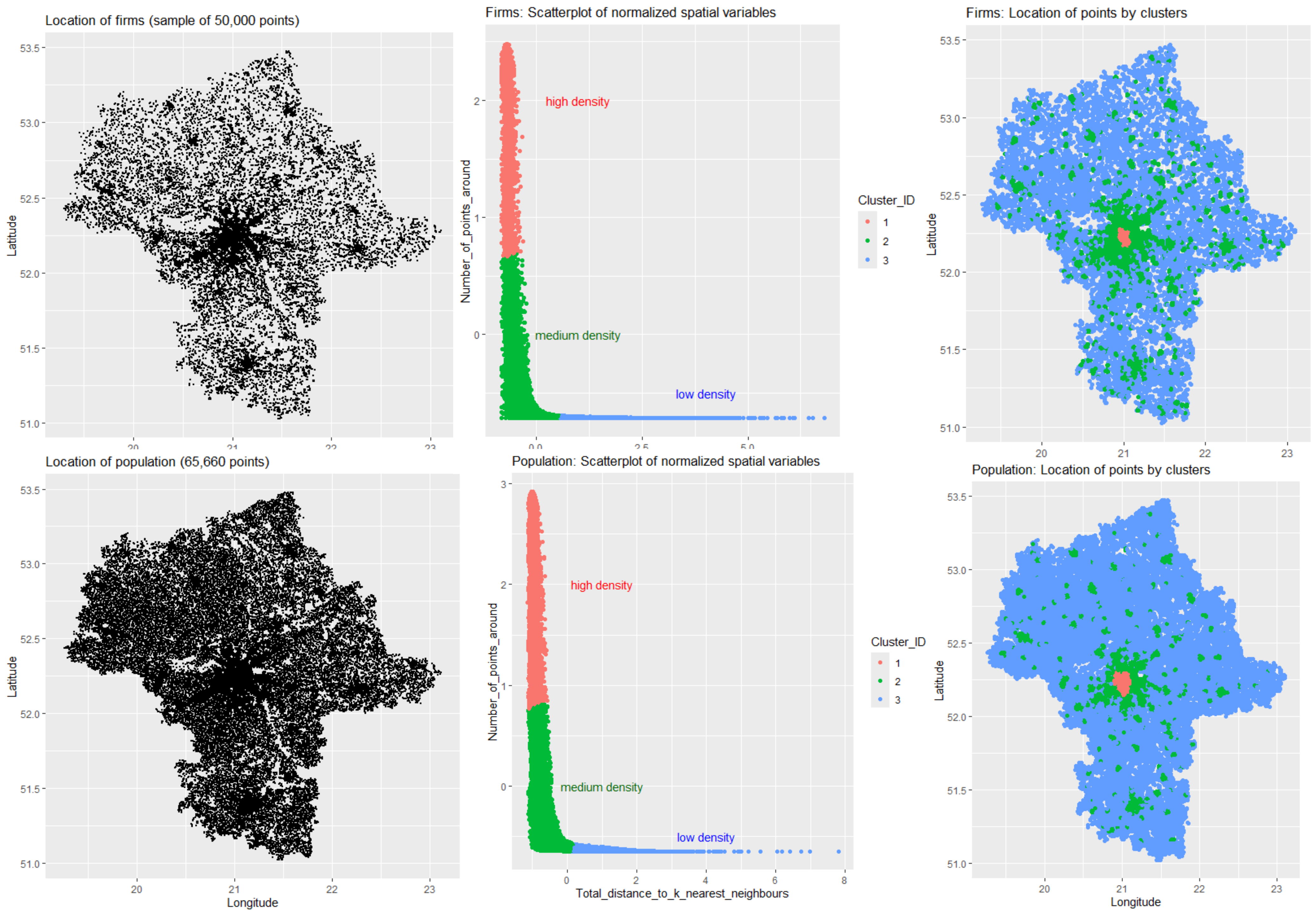
a – in partitioning 1 and 2 both points are in the same cluster [**the same, the same**] – these are stable observations;

b - in partitioning 1 and 2 both points are in different clusters [**different, different**] – these are also stable observations;

c - in partitioning 1 both points are in the same cluster, but in partitioning 2 they are in different clusters [**the same, different**] – these are "jumping" observations;

d - in partitioning 1 both points are in different clusters, but in partitioning 2 they are in the same cluster [**different, the same**] – these are "jumping" observations.

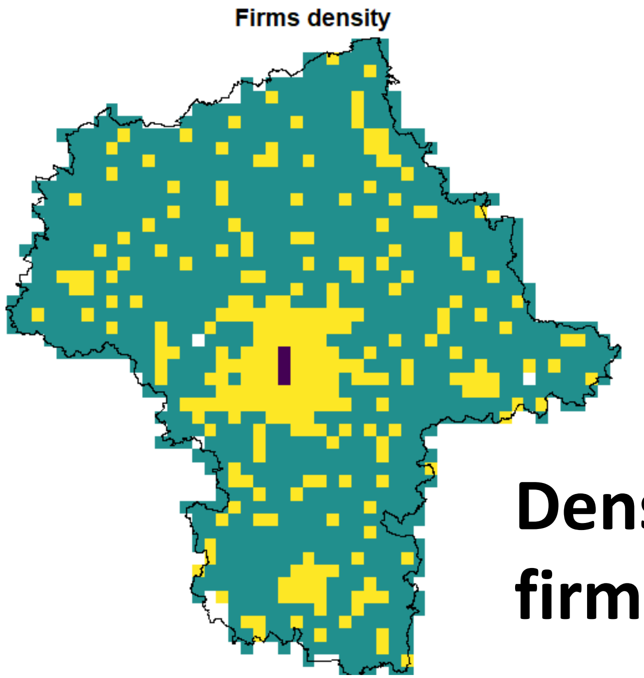




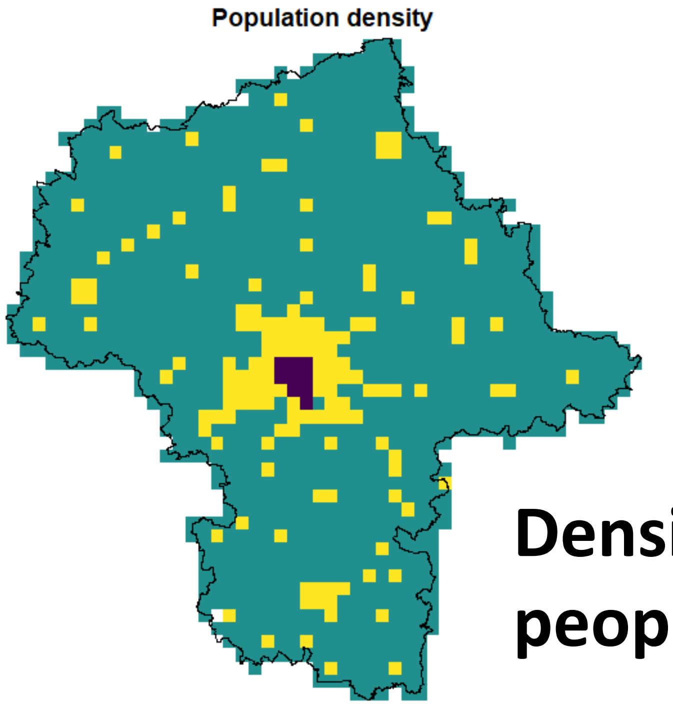
Density of firms

Density of people

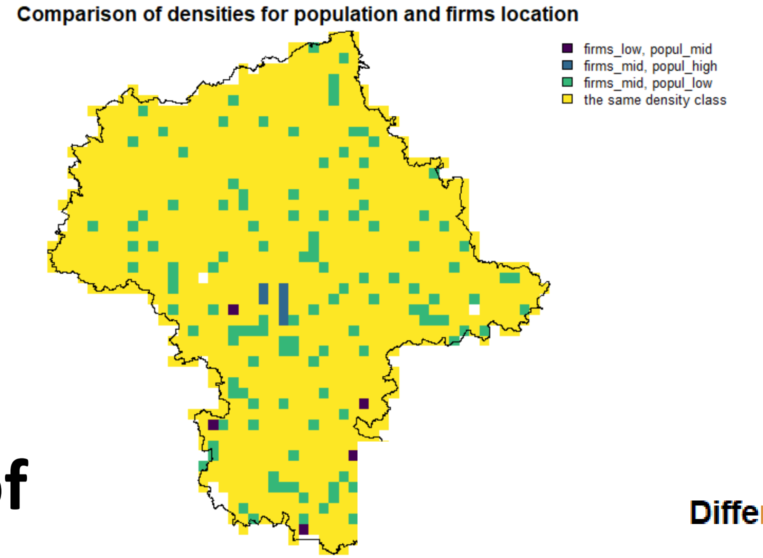
Analysis on grids

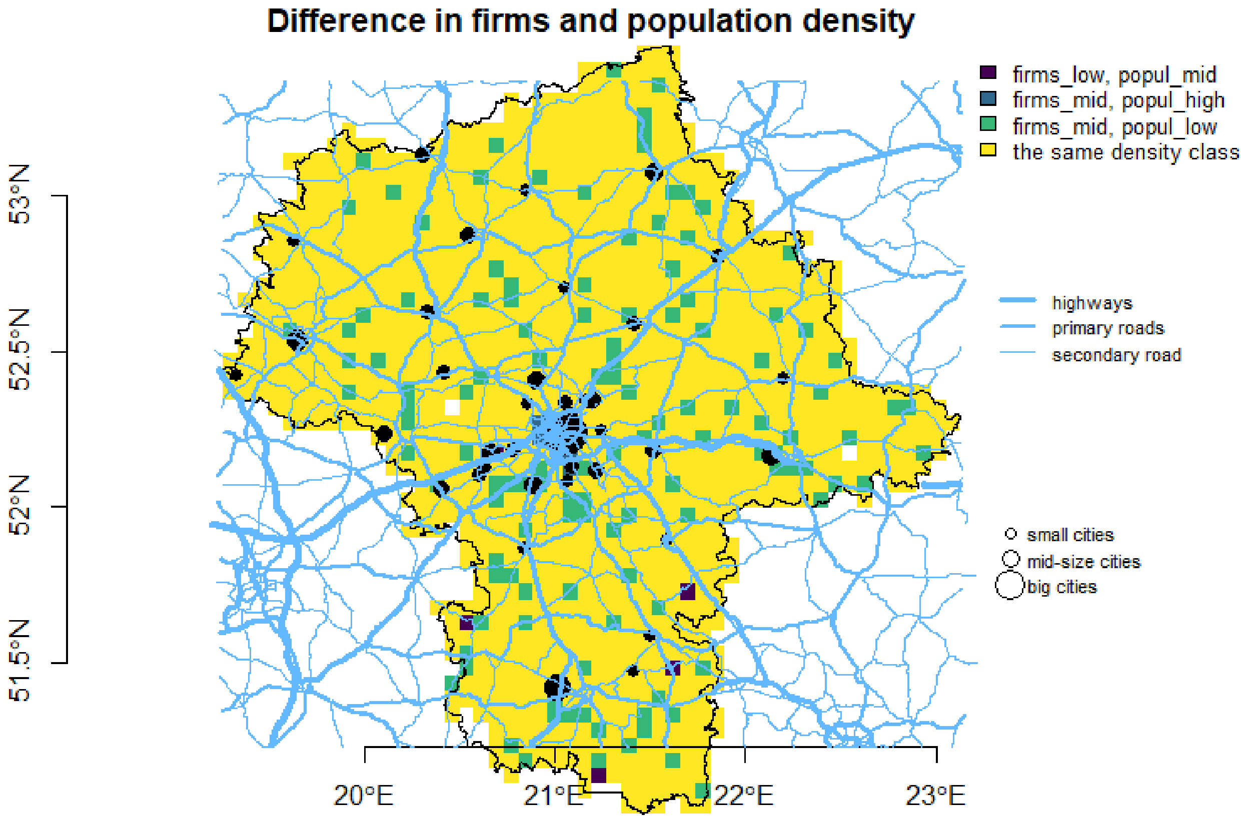


Density of firms



Density of people





Features of algorithms to consider

Questions to asnwer	QDC
What is the input to the algorithm?	Geolocated point data
What is the result of the algorithm?	Each point is classified into one of density clusters (high / medium / low); one can control the number of clusters
What are the parameters?	**Radius** (to check a number of points within it) and **k nearest neighbours** (to calculate the total distance)
How are the parameters set? Is there any benchmarking?	Parameters are set **arbitrarily**, but due to normalisation their absolute value is of little importance
Are the parameters sensitive to sample size and/or subsampling?	**No**, the same parameters can be used for any subsample; in a subsample, the distance to k nearest neighbours is usually greater than in a full sample, but the change is for all units and after normalisation it does not matter much
Is the result sensitive to the values of the parameters?	**No**, the classification is very similar in all scenarios
Is the result sensitive to sample size?	**No**, there is no relationship between sample size and result

Self-calibrating mechanism and de-calibration flags

Self-calibration

- By normalisation, we do not care what values of parameters express „dense”, „close”
- With any reasonable values of parameters, results are stable – the method is quite robust
- We do not set arbitrary the thresholds as in DBSCAN
- We can use the same parameters in a subsample and it still works (e.g. for 1/10 sample, the number of NN in radius changes)

Streaming data (new data)

- For a new point we know its location and we can get its neighbourhood – we can calculate both spatial variables
- We need to normalise it – mostly we will use mean and sd from the training
- What if parameters change – our model decalibrates
- By using streaming parameters (Welford’s online algorithm) we can check if our new parameters are still similar to the baseline parameters, if not we get a flag

$$\bar{x}_n = \frac{(n-1)\bar{x}_{n-1} + x_n}{n} = \bar{x}_{n-1} + \frac{x_n - \bar{x}_{n-1}}{n}$$

$$s_n^2 = \frac{(n-2)}{(n-1)} s_{n-1}^2 + \frac{\left(x_n - \bar{x}_{n-1}\right)^2}{n}$$

Is this method OK?

The methods should fulfil some criteria:

a) work quickly (due to quick k-means implementations)

b) do not involve deep pre-studies to get parameters (best, when their outcome weakly depends on parameters set)

c) can set the high/low-density benchmark autonomously or use reference given by the user

d) are suitable for big data

e) can easily work with stream data

f) have the self-calibrating or at least self-noticing mechanism giving an alert if the previously calibrated model stops being valid due to structural change in new data. This kind of (semi)autonomous (self-service) method is desired by users due to low computational cost and high analytical gain.

Thematic areas of point data

Urban management
- resilient and smart cities
- quality of life
- urban traffic and crowd
- pedestrian flow
- urban heat island (UHI)
- thermal remote sensing
- air quality monitoring
- public transport accessibility
- places with exceeded capacity
- urban facilities and infrastructure
- spatial planning
- housing and real estates
- urban ecosystems management
- IoT-based monitoring
- urban digital twins

Environment and climate
- sea level changes
- emissions and pollutions
- surface temperature and rainfalls
- weather forecasts
- vegetation
- floods, droughts and other natural hazards
- seismic activity (earthquakes, volcano eruptions)
- risk maps of disasters and natural hazards
- wildfire risk
- biodiversity and species distribution
- microclimate
- wildlife tracking and conservation
- invasive species monitoring

Socio-economic development
- census / geo-referenced population data
- education and literacy
- public and private transportation
- access to water and sanitation services
- public sector policy and public finance
- spatial mobility and migration patterns
- urbanization degree
- poverty and wealth
- employment
- crime and safety
- internet access and conectivity

Agriculture
- mapping food / crop production
- food / crop production management
- crop yield prediction
- pest attacks
- livestock

Infrastrutcure
- supply chains, transportation and logistics
- navigation and geo-tracking
- telecommunication and GIS
- construction
- maritime activity and military services
- UAV / UAS (unmanned aerial vehicles / systems)
- astronomy, star systems, satellites

Health
- spread of diseases
- predicting disease risks
- maps of vaccinations
- healthcare facilities and services
- healthcare usage
- brain mapping

Natural resources
- deforestation and reforestation
- energy resources and infrastructure,
- green energy
- land use, landscape, terrain
- soil condition and properties
- distribution of water, hydrology, rivers
- geothermia
- oil, gas, coal and metals mining

Business and economy
- shops and customers
- marketing, recommendations,
- customer geo-targeting and segmentation
- business location
- innovations, R&D
- logistics

Security and defence
- geospatial intelligence
- border surveillance
- distaster response and emergency management
- cybersecurity and location-based threat detection

Culture, sports and recreation
- tourism
- archeology
- cultural ecosystem services and landscape
- cultural heritage
- geotagged social media
- sports analytics
- outdoor recreation
- event management and crowd control

Computers, Environment and Urban Systems 119 (2025) 102289

Contents lists available at ScienceDirect

Computers, Environment and Urban Systems

journal homepage: www.elsevier.com/locate/ceus

Analysing local spatial density of human activity with quick density clustering (QDC) algorithm

Katarzyna Kopczewska *

Faculty of Economic Sciences, University of Warsaw, Warsaw 00-927, Poland

ARTICLE INFO

Keywords:
Spatial density clustering
Human activity
Geospatial analysis

ABSTRACT

This paper deals with the local spatial density of human activity. By understanding and quantifying the spatial distribution of interrelated phenomena such as business location and population settlement at the micro level, it is possible to track local under- and over- spatial representation in socio-economic development. The modelling of spatial density using point data is crucial for territorially targeted policies and business decisions. Weak stream of studies in this field is a consequence of lack of methods. This study presents quick density clustering (QDC), a novel algorithm for classifying geolocated point data into low, medium and high density clusters. QDC uses two spatial features - the sum of distances to k-nearest neighbours (kNN) and the number of neighbours within a fixed radius (frNN) - to generate parameter robust, interpretable clusters. By normalising these metrics and applying K-means clustering, QDC captures both local and global density variations, making it suitable for analysing human activity at urban and regional scales. Empirical validation demonstrates its accuracy and effectiveness in partitioning point data into density clusters and comparing density groups in grids. The QDC provides a robust framework for advancing density-based studies in socio-economic research as well as environmental science and spatial statistics

QDC

Katarzyna Kopczewska

2025-04-03

Reproducible codes for QDC (Quick Density Clustering)

This document shows how to reproduce the graphics and analysis that were presented in a paper on "Analysing local spatial density of human activity with quick density clustering (QDC) algorithm"

Kopczewska K. (2025), Analysing local spatial density of human activity with quick density clustering (QDC) algorithm, Computers, Environment and Urban Systems,

Reading the data for the analysis

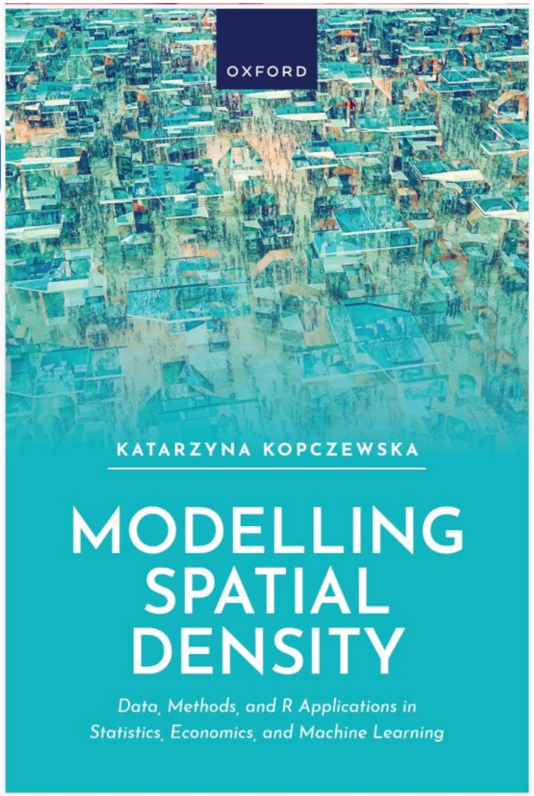
Datasets for point data are available at Figshare, please download:

- for population is at https://figshare.com/s/b0feffe666b4f4bece93?file=39484516 - these are the locations (x,y) of points (always needed)
- for firms at https://figshare.com/articles/dataset/Firms_in_Mazovian_region_Poland_2012_/22215634?file=46207716 - these are the locations (x,y) of points (always needed)

Overview

Description

Author Information



Modelling Spatial Density

Data, Methods, and R Applications in Statistics, Econometrics, and Machine Learning

Katarzyna Kopczewska

- Provides a novel research toolbox for spatial analytics
- Enables reproducability of all methods presented through R code
- Presents quantitative methodologies in inclusive and accessible language for social and environmental scientists

£45.00

Add to Basket

Paperback

This item is not yet published, but may be pre-ordered now for delivery when available.

Notify Me When In Stock

Published: 09 October 2025 (Estimated)

336 Pages

234x156mm

ISBN: 9780198975182

So, what's next?

- **github.com/poktam/spatialWarsaw**

- Functions used in the book and many more
- **R package almost ready**

spatialWarsaw R package

Repository for spatialWarsaw project in R

The functions collected in the spatialWarsaw package support spatial analysis on geolocalised points. They use spatial machine learning, spatial statistics and spatial econometric approaches. You can detect density clusters (with `QDC`, `ClustCont`, `ClustDisjoint`, `bootdbscan` functions), measure global and local agglomeration of points (`SPAG`, `ETA`, `FLE` functions), construct the spatial weight matrix W from Voronoi polygons (`tessW` function), determine the best knn structure of W with AIC (`bestW` function), check the correlation between spatial lags with different knn (`corrSpatialLags` function), and study semi-variance by expanding knn (`semiVarKnn` function). You can test and generate spatial point patterns that follow Benford's law (with `SpatBenfordTest`, `SpatBenfordPattern`). In spatial econometrics, it can run switching regime models - regression in density subgroups (`ssr` function) and bootstrapped spatial regression (`BootSpatReg` function), generate out-of-sample predictions (`SpatPredTess` function), approximate standard errors for large samples (`ApproxSERoot2` function), and rasterise and cluster GWR coefficients (with `rastClustGWR` function) to check their spatiotemporal stability (`STS` function).
Please note that the package is still under development, although all the functions provided work. Currently we are working on cleaning up the code and creating help for specific functions. The package will soon be updated with sample data.

Thank you very much!

Dr hab. Katarzyna Kopczewska, prof.ucz.
Katedra Data Science / Department of Data Science
Wydział Nauk Ekonomicznych / Faculty of Economic Sciences
Uniwersytet Warszawski / University of Warsaw

@ kkopczewska@wne.uw.edu.pl

@Katarzyna Kopczewska
@SpatialWarsaw

UNIVERSITY OF WARSAW | FACULTY OF ECONOMIC SCIENCES