



**HAL**  
open science

# Graph-Based Spatial Segmentation of Health-Related Areal Data

Vivien Goepp, Jan van de Kastele

► **To cite this version:**

Vivien Goepp, Jan van de Kastele. Graph-Based Spatial Segmentation of Health-Related Areal Data. Computational Statistics and Data Analysis, 2022, 10.48550/arXiv.2206.06752 . hal-03474990

**HAL Id: hal-03474990**

**<https://hal.science/hal-03474990v1>**

Submitted on 27 May 2022

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Graph-Based Spatial Segmentation of Health-Related Areal Data

Vivien Goepp<sup>a,b,c,d,\*</sup>, Jan van de Kasstelee<sup>e</sup>

<sup>a</sup>*Mines ParisTech, PSL Research University,  
CBIO-Centre for Computational Biology, F-75006 Paris, France*

<sup>b</sup>*Institut Curie, PSL Research University, F-75005 Paris, France*

<sup>c</sup>*INSERM, U900, F-75005 Paris, France*

<sup>d</sup>*MAP5, CNRS UMR 8145, 45, rue des Saints-Pères, 75006, Paris, France*

<sup>e</sup>*National Institute for Public Health and the Environment - RIVM, Bilthoven,  
The Netherlands*

---

## Abstract

Smoothing is often used to improve the readability and interpretability of noisy areal data. However there are many instances where the underlying quantity is discontinuous. In this case, specific methods are needed to estimate the piecewise constant spatial process. A well-known approach in this setting is to perform segmentation of the signal using the adjacency graph, as does the graph-based fused lasso. But this method does not scale well to large graphs.

This article introduces a new method for piecewise-constant spatial estimation that *(i)* is fast to compute on large graphs and *(ii)* yields sparser models than the fused lasso (for the same amount of regularization), giving estimates that are easier to interpret.

We illustrate our method on simulated data and apply it to real data on overweight prevalence in the Netherlands. Healthy and unhealthy zones are identified which cannot be explained by demographic or socio-economic characteristics. We find that our method is capable of identifying such zones and can assist policy makers with their health-improving strategies. The implementation of our method in R is publicly available at [github.com/goepp/graphseg](https://github.com/goepp/graphseg).

---

\*Corresponding author: [vivien.goepp@gmail.com](mailto:vivien.goepp@gmail.com)  
CBIO-Centre for Computational Biology, F-75006 Paris, France

*Keywords:* Graph-based signal segmentation, Piecewise constant estimation, Public Health, Areal lattice data, Sparse estimation, Adaptive Ridge, Variable selection

---

## 1. Introduction

Spatial statistics plays a prominent role in epidemiology. Nowadays, the study of health-related outcomes with respect to the geographical location has become widespread. The data for these studies can be of various types, leading to different statistical tools to answer the epidemiological questions at hand [1].

A common problem in spatial statistics is the regularization of spatial data. Most regularization methods perform a spatial smoothing of the data, e.g. kriging for interpolation of data that has been collected at fixed point locations [2] or disease mapping techniques in the case of data has been collected for administrative areas [1]. The main advantage of smoothing is a higher interpretability of the resulting map. The underlying assumption is that the actual truth is smooth.

In some cases however, one may want to obtain a segmented estimation of the spatial distribution, for instance when the true underlying spatial effect is assumed to be discontinuous. Besides, from a policy making point of view, for the purpose of improving the health of a population, it can be of interest to identify zones having a similar health-related status. Because of logistic and administrative efficiencies involved in such health-improving strategies, the areas within such zones should preferably be contiguous. Spatial segmentation techniques provide an objective way to identify such zones.

In demographic and epidemiological studies, the neighborhood in which we live often has an effect on the health-related outcome variable, even when adjusted for demographic variables, like age and sex, and other socio-economic variables, like educational level and income. This leads to the hypothesis that the neighborhood of residence has by itself an impact on the health-related outcome variable. Beyond the demographic and socio-economic factors, people living in the same area tend to share the same habits: the school they attend, the supermarket they shop at, the bank they go to, etc. As an example, demographic studies on longevity focus namely on finding specific geographic areas where the longevity is unexpectedly high. These more-than-expected healthier areas are sometimes referred to as "blue

zones” [3]. These areas are discrete by nature, and such studies use a spatial segmentation of administrative areas to identify these zones.

Thus, instead of looking at the prevalence of a health-related indicator itself, it may be more interesting for policy makers to identify zones that have a higher or lower prevalence than can be expected based on the demographic and socioeconomic composition of neighborhoods alone. Such information is usually not directly available, but van de Kasstele et al. [4] presented a small area estimation model that, as a by-product, provides an estimate of these neighborhood-specific deviations in the form of a spatial random effect term. The goal is to identify healthy and unhealthy zones by performing spatial segmentation on the neighborhood specific deviations that cannot be explained by demographic or socio-economic characteristics.

The main approach for segmentation of piecewise spatial data is to use the graph-fused lasso on the adjacency graph. The graph-fused lasso was first introduced for regression [5] and then extended to multitask regression [6]. Hoeffling [7] introduced a path algorithm for regression with any fused lasso penalty, called generalized fused lasso and Wang et al. [8] developed a method for trend filtering fused lasso on graphs using the ADMM optimization algorithm. Finally, [9] have proposed a fast estimation procedure for the graph fused lasso with any convex loss based on a trail decomposition of the graph and the ADMM algorithm.

In this paper, we introduce a new graph-based sparsity-inducing estimation method that yields sparser estimates than the graph fused lasso. As for the graph fused lasso [7], the method minimizes a likelihood penalized over the differences of the parameter over a graph, using the adjacency structure of the areas. Our method extends the adaptive ridge [10, 11] to a graph fused penalty. The adaptive ridge is a sparsity-inducing iterative method based on iterating over a weighted ridge problem. Since the complexity depends on the number of areas and not on the number of individuals, our method is computationally efficient when there is a large number of areas.

The paper is organized as follows. Section 2.1 introduces the model. In Section 2.2 we investigate the properties of our method, both on simulated data and on real data on overweight prevalence in the Netherlands. Results are shown in Section 3 and are discussed in Section 4.

## 2. Methods and data

### 2.1. The graph-based fused adaptive ridge

In this section, we present our method for estimating the piecewise graph-based signal from a noisy observation.

#### 2.1.1. Segmentation of spatial lattice data

The method presented in this section applies to any signal defined on an undirected, simple graph. However, in this paper, it is applied to the case of lattice spatial data. We first explain how lattice spatial data can be viewed.

Consider a set of  $p \geq 1$  areas forming a partition of a connected subset of  $\mathbb{R}^2$ . In this work, areas often correspond to an administrative division of a territory, for instance census tracts, municipalities, counties, or neighborhoods. Consider a signal  $\mathbf{x} \in \mathbb{R}^p$  (called *spatial effect* in the following) where each component of  $\mathbf{x}$  corresponds to an area. Consider that  $\mathbf{x}$  is a noisy observation of an unobserved effect  $\boldsymbol{\theta}$ :

$$\mathbf{x} = \boldsymbol{\theta} + \boldsymbol{\varepsilon}, \tag{1}$$

where  $\boldsymbol{\theta} \in \mathbb{R}^p$  is the underlying, deterministic signal and  $\boldsymbol{\varepsilon} \in \mathbb{R}^p$  is an error term centered around zero.

The errors need not be normally distributed. When there are, the mean square estimate introduced in this paper corresponds to the negative log-likelihood.

We assume that  $\boldsymbol{\theta}$  is piecewise constant. More precisely, it has the same values on unions of areas, which we call *zones*. The true number of zones is noted  $q$  and hence  $\boldsymbol{\theta}$  only takes  $q$  different values. The zones form a partition of the areas, and each zone is made of a subset of contiguous areas (rook-type contiguity). The goal of this paper is to infer  $\boldsymbol{\theta}$ .

#### 2.1.2. Spatial lattice data as a signal on graph

Consider the adjacency graph  $\mathcal{G} = (\mathcal{E}, \mathcal{V})$  of the areas, where each vertex is an area and two vertices are adjacent if their corresponding areas share a border (rook-type contiguity). We can then view  $\mathbf{x}$  as a signal on this graph (see [12] for a review of signals on graphs). Estimating  $\boldsymbol{\theta}$  can then be viewed as a problem of piecewise constant estimation of the signal on graph.

Note that when estimating  $\boldsymbol{\theta}$  as a signal on graph, we discard any information of the areas as geometrical sets (e.g. Euclidian distance between centroids). This modelization simplifies the problem, but makes the assumption

that adjacency contains sufficiently enough information about how *similar* two areas are.

In applications where there are several connected components (e.g. caused by rivers or the presence of islands) one can consider each component as a separate problem, or connect every pair of components by artificially adding an edge between their closest areas (using the Euclidean distance between their centroids).

### 2.1.3. Estimation procedure

Segmentation of  $\boldsymbol{\theta}$  is done by using a sparsity-inducing method applied to the differences of the values of  $\boldsymbol{\theta}$ . The penalty method we use is the adaptive ridge [10, 11]. This penalized method belong to the class of sparsity-inducing penalized method, like the lasso [13]. It performs feature selection by iterating over re-weighted  $L_2$  norm penalties, which have an explicit solution.

We define the sum-of-squares cost function

$$\ell(\boldsymbol{\theta}) = \frac{1}{2}(\mathbf{x} - \boldsymbol{\theta})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x} - \boldsymbol{\theta}), \quad (2)$$

where  $\boldsymbol{\Sigma}$  is the covariance matrix of  $\boldsymbol{\varepsilon}$ . As mentioned above, this cost function corresponds to the negative log-likelihood when  $\boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$ . When the covariance is not assumed to be known, we set  $\boldsymbol{\Sigma} = \mathbf{I}$  in the cost function.

The estimating procedure is as follows. First, we define the weighted undirected adjacency graph  $\mathcal{G}^{(l)} = (\mathcal{E}, \mathcal{V}^{(l)})$ , where each weighted edge  $\{j, k\} \in \mathcal{V}^{(l)}$  between vertices  $j$  and  $k$  is assigned a positive weight  $v_{j,k}^{(l)} \geq 0$  (by definition  $j$  and  $k$  are not adjacent if  $v_{j,k}^{(l)} = 0$ ) that depends on the iteration step  $l$ . Next, define the penalized log-likelihood  $\ell^{\text{pen}}$  using a weighted  $L_2$  penalty:

$$\ell^{\text{pen}}(\boldsymbol{\theta}, \mathcal{V}^{(l)}) \triangleq \ell(\boldsymbol{\theta}) + \frac{\lambda}{2} \sum_{j \sim k} v_{j,k}^{(l)} (\theta_j - \theta_k)^2, \quad (3)$$

where  $\lambda > 0$  is a smoothing parameter. The set of weighted edges  $\mathcal{V}^{(l)}$  is included as a parameter of  $\ell^{\text{pen}}$  to highlight the dependence on the current weighted graph  $(\mathcal{E}, \mathcal{V}^{(l)})$ . The sum in the latter equation is taken only once per vertex, that is, the sum index is  $\{(j, k) \in \mathcal{E}, j < k\}$ , where an arbitrary ordering of the nodes has been chosen. The edge weights  $v_{j,k}^{(l)}$  play the role of tuning the importance of the difference between areas  $j$  and  $k$  while  $\lambda$  plays the role of tuning the overall regularization.

The adaptive ridge procedure iterates between a step of weighted smoothing and an update of the weights:

$$(i) \quad \boldsymbol{\theta}^{(l)} \triangleq \arg \min_{\boldsymbol{\theta}} \ell^{\text{pen}}(\boldsymbol{\theta}, \mathcal{V}^{(l-1)}) \quad (4a)$$

$$(ii) \quad v_{j,k}^{(l)} = \frac{1}{(\theta_j^{(l)} - \theta_k^{(l)})^2 + \varepsilon}, \quad \{j, k\} \in \mathcal{V}. \quad (4b)$$

where  $\varepsilon > 0$  is a small numerical constant, introduced in order to bound the denominator away from zero for numerical stability. Different choices for the value of  $\varepsilon$  have been proposed, Candès et al. [14] and Daubechies et al. [15] having proposed to update its value at each iteration, decreasing it as the algorithm converges. Numerical experiments [14, 11] have highlighted that the estimation procedure is relatively robust to the choice of  $\varepsilon$ , so we favored setting a constant  $\varepsilon$  ( $\varepsilon = 10^{-6}$  in our implementation). More details about the convergence of Eq. (4) is given in Appendix A.

#### 2.1.4. Implementation and algorithmic considerations

Define the weighted Laplacian matrix associated to the weighted graph  $(\mathcal{E}, \mathcal{V}^{(l)})$ :  $\mathbf{K}^{(l)} = \mathbf{D}^{(l)} - \mathbf{A}^{(l)}$  where  $\mathbf{D}^{(l)}$  is the diagonal matrix giving the weighted degree of each node:  $d_{j,j}^{(l)} = \sum_{k \sim j} v_{j,k}^{(l)}$  and  $\mathbf{A}^{(l)}$  is the weighted adjacency matrix:  $a_{j,k}^{(l)} = v_{j,k}^{(l)}$  if  $j \sim k$  and zero otherwise.

We first rewrite Eq. (3). Using the fact that  $\sum_{j \sim k} v_{j,k}^{(l)} (\theta_j - \theta_k)^2 = \boldsymbol{\theta}^\top \mathbf{K}^{(l)} \boldsymbol{\theta}$  and using Eq. (2), the penalized likelihood can be written

$$\ell^{\text{pen}}(\boldsymbol{\theta}, \mathcal{V}^{(l)}) = \frac{1}{2} (\mathbf{x} - \boldsymbol{\theta})^\top \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\theta}) + \frac{\lambda}{2} \boldsymbol{\theta}^\top \mathbf{K}^{(l)} \boldsymbol{\theta} \quad (5)$$

and the weighted ridge problem in Eq. (4) is solved by the explicit update:

$$\boldsymbol{\theta}^{(l)} = (\boldsymbol{\Sigma}^{-1} + \lambda \mathbf{K}^{(l-1)})^{-1} \boldsymbol{\Sigma}^{-1} \mathbf{x}. \quad (6)$$

In the simple case of independent spatial effects, the precision matrix  $\boldsymbol{\Sigma}^{-1}$  is diagonal with  $j$ -th entry  $(1/\sigma_j^2)$ . If the  $x_j$ s are not assumed independent, as will be the case in our real data application, we assume that  $\boldsymbol{\Sigma}^{-1}$  is sparse. Under this assumption,  $\boldsymbol{\Sigma}^{-1} + \lambda \mathbf{K}^{(l)}$  is sparse positive definite and its inversion can be done using the Cholesky decomposition. In the application, we use a sparse estimate of  $\boldsymbol{\Sigma}^{-1}$ .

*Remark.* The sum of squares function is derived from the likelihood when  $\mathbf{x}$  is Gaussian. However we can use the method in the more general setting where  $\mathbf{x}$  follows any distribution of mean  $\boldsymbol{\theta}$  and covariance matrix  $\boldsymbol{\Sigma}$ . For computational efficiency, our implementation nonetheless requires that  $\boldsymbol{\Sigma}^{-1}$  be sparse.

The computational bottleneck of the iterative procedure is the linear system (6). We use the package `Matrix`, version 1.2-17, which makes use of the `CHOLMOD` library [16] for fast inversion of sparse (semi-)positive definite matrices. Moreover, the matrix  $\boldsymbol{\Sigma}^{-1} + \lambda \mathbf{K}^{(l)}$  has the same sparsity structure at all steps. This can be leveraged to further speed-up the iterative procedure: we compute the symbolic Cholesky decomposition of  $\boldsymbol{\Sigma}^{-1} + \lambda \mathbf{K}^{(l)}$  only once (using function `Matrix::Cholesky`) and update the numerical values at each iteration (using function `Matrix::update`).

The segmentation procedure for one value of  $\lambda$  is given in Algorithm 1. In practice, estimation is performed on a grid of penalties, and the choice of the best  $\lambda$  is done in a second step.

---

**Algorithm 1** Segmentation over a graph using the adaptive ridge

---

```

1: procedure ADAPTIVE-RIDGE( $\mathbf{x}, \boldsymbol{\Sigma}^{-1}, \lambda$ )
2:    $v_{j,k}^{(0)} \leftarrow \mathbb{1}_{j \sim k}$ 
3:    $l \leftarrow 1$ 
4:   do
5:      $\mathbf{K}^{(l-1)} \leftarrow \mathbf{D}^{(l-1)} - \mathbf{A}^{(l-1)}$ 
6:      $\boldsymbol{\theta}^{(l)} \leftarrow (\boldsymbol{\Sigma}^{-1} + \lambda \mathbf{K}^{(l-1)})^{-1} \boldsymbol{\Sigma}^{-1} \mathbf{x}$ 
7:      $v_{j,k}^{(l)} \leftarrow ((\theta_j^{(l)} - \theta_k^{(l)})^2 + \varepsilon)^{-1}$ 
8:      $\delta_{j,k}^{(l)} \leftarrow v_{j,k}^{(l)} (\theta_j^{(l)} - \theta_k^{(l)})^2$ 
9:      $l \leftarrow l + 1$ 
10:  while  $\max_{j,k} |\delta_{j,k}^{(l)} - \delta_{j,k}^{(l-1)}| > \text{tol}$ 
11:  return  $\boldsymbol{\theta}$ 

```

---

The implementation of the method for a sequence of penalties is made faster using a *warm start*: the estimations are performed on an increasing sequence of penalties  $(\lambda_q)$ , and the weights  $v_{j,k}^{(l)}$  obtained at convergence for  $\lambda_q$  are recycled for the first iteration of the estimation  $\lambda_{q+1}$ . This trick is based on the fact that the limit of the adaptive ridge iterations does not vary a lot under a small variation in  $\lambda$ . It significantly reduces the number of iterations needed for convergence for subsequent values of the penalty.



2.1.5. *Choice of the regularization parameter*

In penalized methods, the choice of the penalty  $\lambda$  is a difficult task [17, Section 7]. In a number of statistical settings, the only data-driven criterion for choosing  $\lambda$  is cross-validation. In this setting, the graph structure of the data makes cross-validation computationally too costly since one would need to repeat the cross-validation over a large set of test/train sets. Consequently, we use model selection criteria. We consider the BIC Schwarz [18], the AIC Akaike [19], and generalized cross-validation (GCV) [17, Section 7]:

$$\begin{aligned} \text{BIC}(\lambda) &= 2\ell(\hat{\boldsymbol{\theta}}) + \log(p)e(\lambda), \\ \text{AIC}(\lambda) &= 2\ell(\hat{\boldsymbol{\theta}}) + 2e(\lambda), \\ \text{GCV}(\lambda) &= \frac{2\ell(\hat{\boldsymbol{\theta}})}{p(1 - e(\lambda)/p)^2}, \end{aligned}$$

where  $e(\lambda)$  is the effective model dimension.

The effective dimension is a generalization of the number of parameters to linear fitting methods: if the estimate writes  $\hat{\boldsymbol{\theta}} = \mathbf{S}(\lambda)\mathbf{x}$  with  $\mathbf{S}(\lambda)$  independent from  $\mathbf{x}$ , then we define  $e(\lambda) = \text{Tr}(\mathbf{S}(\lambda))$ . When  $\mathbf{S}$  is a projection (as in the unpenalized linear model), the effective dimension is the number of parameters in the model. In the general case, this formula accounts for both the number of selected parameters and the shrinkage between the parameters. Besides allowing to use cost-effective selection criteria,  $e(\lambda)$  is important in itself: it allows to quantify the degree of freedom of the estimate.

We now derive the effective dimension for the adaptive ridge. As explained in Goepf et al. [20, Section S1], the adaptive ridge Equations (4a) and (4b) is a numerical scheme which minimizes the penalized problem

$$\min_{\boldsymbol{\theta}} \ell(\boldsymbol{\theta}) + \frac{\lambda}{2} \sum_{j \sim k} \log((\theta_j - \theta_k)^2 + \varepsilon)$$

through Fan and Li [21]’s one-step approximation procedure called local quadratic approximation (LQA). Now following the work of Fan and Li [21, 22] for extending the notion of effective dimension to this class of estimating algorithms, we use

$$e(\lambda) \triangleq \text{Tr}((\boldsymbol{\Sigma}^{-1} + \lambda \mathbf{K}^{(l)})^{-1} \boldsymbol{\Sigma}^{-1}) \quad (7)$$

when the iteration step ( $l$ ) is at convergence. There is no theoretical justification for this formula, which is rather motivated by the analogy with linear

smoothers, since our estimate takes the form  $\boldsymbol{\theta} = (\boldsymbol{\Sigma}^{-1} + \lambda \mathbf{K}^{(l)})^{-1} \boldsymbol{\Sigma}^{-1} \mathbf{x}$ , when  $l$  is large enough that the method has converged.

Computing  $e(\lambda)$  only requires solving the linear system in the right-hand side of (7), which comes at a small computational cost since the Cholesky decomposition of  $\boldsymbol{\Sigma}^{-1} + \lambda \mathbf{K}^{(l)}$  is already in memory at convergence of the algorithm.

## 2.2. Application

### 2.2.1. Simulation study

We illustrate our method in six simulation settings. The graph structure used to define the zones and the adjacency between regions is taken from real data. We use the spatial polygons defining several administrative divisions of the Netherlands: regions, municipalities, districts, and neighborhoods. We consider different simulation settings, with varied numbers of areas ( $p$ ) and zones ( $q$ ). We select 6 settings, with  $p$  varying from 390 to 12920 and  $q$  varying from 6 to 390. The datasets are represented in Figure 1 and their properties are summarized in Table 1. We report the average number of areas per zone, which indicates how hard the estimation of the areas and their spatial effect is: the more areas in a zone, the more information there is about its underlying true value. This dataset based on real data is further detailed in Section 2.2.2 and Figure 2.

The data  $\mathbf{x}$  is simulated from Eq. (1), where the true parameter  $\boldsymbol{\theta}$  is constant over each zone and the errors are homoskedastic ( $\boldsymbol{\Sigma} = \sigma \mathbf{I}$ ). The true parameters  $\boldsymbol{\theta}$  are set equal on each zone and are generated as iid Poisson samples of parameter 10. The noise variance  $\sigma^2$  is set to a series of values between 0.1 and 5.

We select the best  $\lambda$  over a grid of 50 values using either of the three criteria. We run the graph-fused lasso using the implementation from the R package `flsa` [23]. We run `flsa` on the same sequence of values of  $\lambda$ .

### 2.2.2. Real data: overweight in the Netherlands

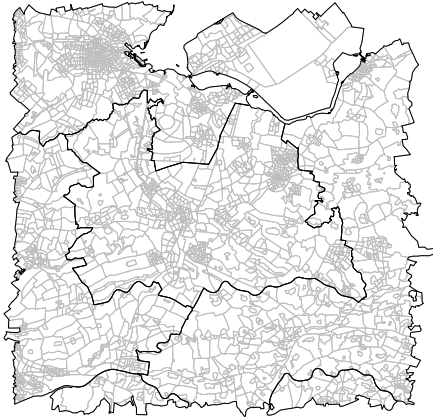
For our real-data application we focus on the estimated overweight prevalence at neighborhood level in the Netherlands in 2016. Overweight is being defined as having a body mass index between 20 and 25. We consider a square region of 75 by 75 kilometers near the center of the Netherlands (the same that is used in simulation Datasets (f)-(e), c.f. Figure 1). Figure 2 shows the region on a map. The region consists of 2,955 neighborhoods. In the



(a) Dataset 1:  $p = 390, q = 12$



(b) Dataset 2:  $p = 650, q = 99$



(c) Dataset 3:  $p = 2955, q = 6$



(d) Dataset 4:  $p = 2955, q = 120$



(e) Dataset 5:  $p = 2955, q = 650$



(f) Dataset 6:  $p = 12920, q = 390$

Figure 1: Areas (grey) and zones (black) of the 6 datasets used in simulation, with their numbers  $p$  and  $q$  respectively.

Table 1: Information on the different simulation designs.

Dataset name	Number of areas ( $p$ )	Number of zones ( $q$ )	Average number of areas per zones
Dataset 1	390	12	32.50
Dataset 2	650	99	6.57
Dataset 3	2955	6	492.00
Dataset 4	2955	99	29.80
Dataset 5	2955	650	4.55
Dataset 6	12920	390	33.13

Netherlands, neighborhoods are defined for administrative use by municipalities and data collection by Statistics Netherlands (CBS). Neighbourhoods are coherent regions that are based on several characteristics like age, geographical barriers such as busy roads, having similar urban and/or architectural features, or having similar functional, social or political characteristics. Neighbourhoods have no formal status.

The goal is to identify spatial zones consisting of adjacent neighborhoods that have higher or lower prevalence than can be expected based on the demographic and socio-economic characteristics of neighborhoods alone.

We follow a two-step procedure. First, we fit the small area estimation model by [4] to our sample data and extract the estimated spatial effect of each neighborhood, given as log-odds ratios, as well as the covariance matrix. Next, we perform the spatial segmentation as described above.

Figure 3 shows the unsegmented spatial random effect for the 2,955 neighborhoods in terms of log-odds ratio's. Blue colours indicate lower than expected log-odds on overweight. Orange colours indicate higher than expected log-odds on overweight. Already clear patterns can be seen, e.g. in large cities like Amsterdam, Utrecht and Amersfoort, the overweight prevalence is lower than expected based on demographic and socio-economic characteristics. In the rural areas in the south, the prevalence is higher than expected.

We used an updated version of the small area estimation model as described by van de Kasstele et al (2017) [4]. The updated model has several improvements. First, sample data are from 2016 instead of 2012. Second, the model includes educational level as a predictor variable. Third, more two-way interactions are included: age by sex, age by ethnicity, age by marital status, age by educational level, sex by ethnicity, sex by marital status

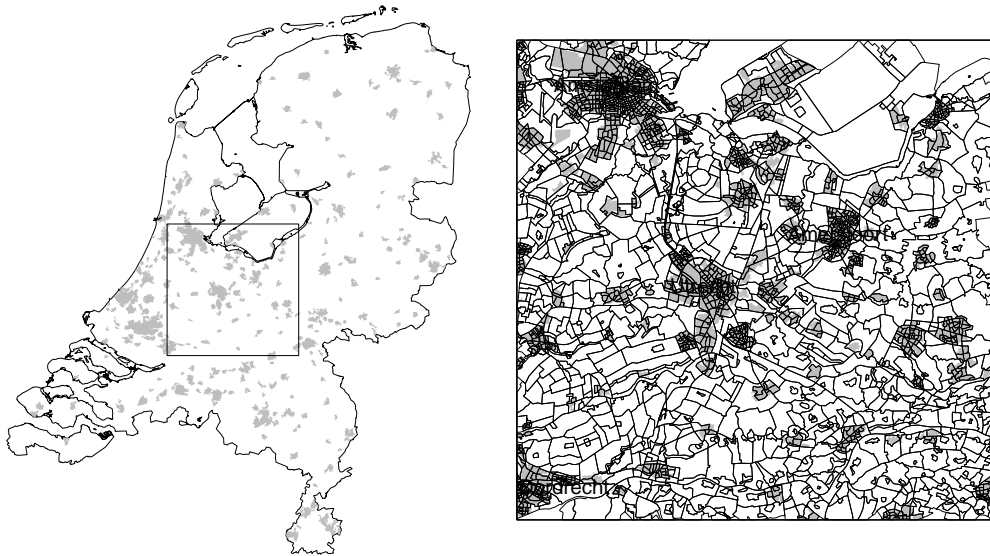


Figure 2: The square 75 by 75 kilometers region of interest located near the centre of the Netherlands. For illustration and orientation, the grey shaded areas are large populated places, e.g. cities like Amsterdam and Utrecht. The map on the right shows the 2,955 neighborhoods in detail.

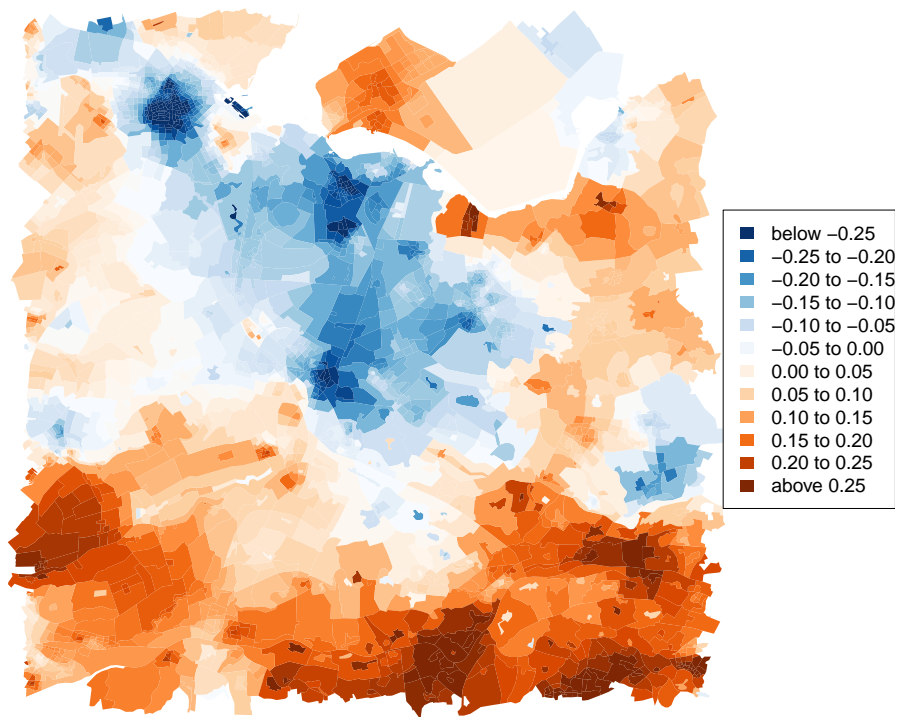


Figure 3: Unsegmented spatial effect for overweight for the 2,955 neighborhoods as estimated by the small area estimation model. Blue colours indicate lower log-odds compared to the expected log-odds, orange colours higher log-odds.

and sex by educational level. Fourth, all predictor variables, both numeric as categorical, entered the model using basis functions and penalization of the regression coefficients. This also enabled automated feature selection, i.e. predictors that are not relevant will not be selected in the model, resulting in a more parsimonious model.

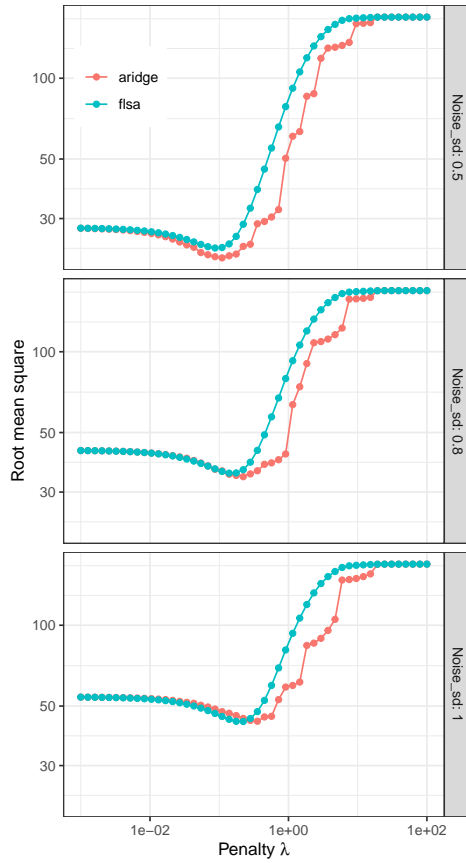
Our method requires the precision matrix as input. We have used the following two-step process to estimate this matrix. First, the covariance matrix  $\Sigma$  of the spatial effect term was extracted from the small area estimation model. Next, we used the graphical lasso [24], as implemented in the package `huge` [25], to generate a sparse estimate of  $\Sigma^{-1}$ .

### 3. Results

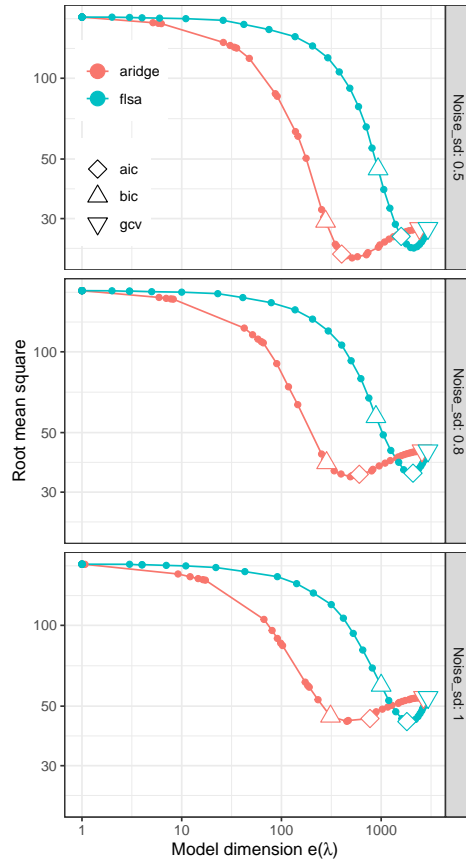
#### 3.1. Simulation study

*The adaptive ridge produces sparser estimates.* We compare the adaptive ridge and the flsa on a single estimate of Dataset 5 ( $p = 2955$ ,  $q = 650$ ) in Figure 4, for  $\sigma = 0.5, 0.8$ , and 1. We present the root mean squared error (RMSE) for varying values of  $\lambda$ . Figure 4a shows that the adaptive ridge performs always better than, or as well as, flsa for the same amount of regularization and Figure 4b shows that *i*) for equal RMSE, the adaptive ridge selects sparser models and *ii*) the AIC criterion is best at selecting the optimal  $\lambda$ . Similar results were observed with the other simulation designs (results not shown here). Extensive simulations (not shown here) across a larger array of values of  $\sigma$  shows that for small  $\sigma$ , GCV outperforms AIC and BIC and for large  $\sigma$ , AIC and BIC are close and outperform GCV. In applications, we recommend using the AIC if estimation performance is prioritized and the BIC if model sparsity is prioritized.

*The adaptive ridge yields better estimates in terms of RMSE.* Table 2 compares adaptive ridge and flsa estimates in terms of RMSE and model dimension across all values of  $\sigma$  and all simulation settings. The adaptive ridge has better estimation performance than flsa (lower RMSE) when  $\sigma$  is not too high ( $\leq 1.1$ ), in which case it has slightly worse but comparable performance. Moreover, it consistently fits sparser estimates.



(a) RMSE as a function of  $\lambda$



(b) RMSE as a function of the model dimension

Figure 4: Root mean squared error (RMSE) of one estimate in Dataset 5 as a function of (a) the penalty  $\lambda$  and (b) the model dimension  $e(\lambda)$ . The left-hand figure shows that at the optimal  $\lambda$ , the adaptive ridge yields an estimate with a RMSE close or slightly less than that of flsa. The right-hand figure shows that it yields a way sparser model (for the same amount of RMSE, since both curves have similar minimal values). For both methods, the AIC criterion is best at selecting the optimal  $\lambda$ .



Table 2: Performance estimation and selected number of zones of the adaptive ridge compared to Hoeffing’s fused lasso signal approximator (lfsa), using the AIC.

Noise standard deviation	Model dimension		RMSE		
	Adaptive Ridge	FLSA	Adaptive Ridge	FLSA	No Regularization
<b>Dataset 1</b>					
0.1	<b>11.1 ± 0.0214</b>	43 ± 1.41	<b>0.0815 ± 0.00297</b>	0.313 ± 0.0685	0
0.3	<b>11.4 ± 0.236</b>	41.5 ± 2.12	<b>0.123 ± 0.00747</b>	0.408 ± 0.0858	0
0.5	<b>15 ± 3.01</b>	36.5 ± 4.95	<b>0.253 ± 0.0264</b>	0.505 ± 0.0236	0
0.7	<b>16.2 ± 3.58</b>	40.5 ± 0.707	<b>0.358 ± 0.0971</b>	0.433 ± 0.0979	0
0.9	<b>16.7 ± 4.12</b>	48.5 ± 4.95	<b>0.467 ± 0.0335</b>	0.5 ± 0.00638	0
1.1	<b>26.4 ± 13</b>	49.5 ± 4.95	0.699 ± 0.0717	<b>0.614 ± 0.0112</b>	0
2	75.4 ± 0.109	<b>73.5 ± 31.8</b>	1.61 ± 0.0144	<b>0.922 ± 0.0976</b>	0
5	<b>153 ± 13.6</b>	362 ± 2.12	<b>4.6 ± 0.0833</b>	4.67 ± 0.0421	0
<b>Dataset 2</b>					
0.1	<b>45.3 ± 1.21</b>	146 ± 0.707	0.356 ± 0.0186	<b>0.322 ± 0.00223</b>	2.5
0.3	<b>44.8 ± 0.549</b>	162 ± 4.95	<b>0.392 ± 0.00749</b>	0.525 ± 0.0935	7.6
0.5	<b>49.1 ± 4.04</b>	146 ± 4.24	<b>0.402 ± 0.0213</b>	0.794 ± 0.0157	13
0.7	<b>38.5 ± 0.0574</b>	150 ± 6.36	<b>0.613 ± 0.0326</b>	0.809 ± 0.0163	18
0.9	<b>50.7 ± 2.07</b>	156 ± 19.1	<b>0.683 ± 0.0774</b>	0.946 ± 0.19	24
1.1	<b>68.9 ± 0.522</b>	157 ± 26.9	<b>0.771 ± 0.0132</b>	0.956 ± 0.142	28
2	<b>128 ± 26.8</b>	332 ± 55.2	1.6 ± 0.0531	<b>1.31 ± 0.0421</b>	51
5	<b>260 ± 33</b>	599 ± 9.9	4.66 ± 0.277	<b>4.64 ± 0.325</b>	130
<b>Dataset 3</b>					
0.1	<b>60.6 ± 0.135</b>	298 ± 2.83	<b>0.178 ± 0.00229</b>	0.213 ± 0.000189	5.5
0.3	<b>61.6 ± 1.56</b>	282 ± 12	<b>0.185 ± 0.0031</b>	0.404 ± 0.00924	16
0.5	<b>61.3 ± 3.64</b>	286 ± 2.12	<b>0.241 ± 0.0106</b>	0.544 ± 0.00403	28
0.7	<b>67 ± 4.07</b>	306 ± 0.707	<b>0.331 ± 0.00976</b>	0.568 ± 0.0106	38
0.9	<b>89 ± 24.3</b>	254 ± 17.7	<b>0.41 ± 0.00876</b>	0.703 ± 0.00372	48
1.1	<b>141 ± 58.2</b>	291 ± 4.24	<b>0.57 ± 0.0662</b>	0.713 ± 0.000775	59
2	543 ± 65.8	<b>486 ± 3.54</b>	1.57 ± 0.0246	<b>0.878 ± 0.0277</b>	110
5	<b>1010 ± 143</b>	2630 ± 12	4.55 ± 0.0702	<b>4.49 ± 0.019</b>	270
<b>Dataset 4</b>					
0.1	<b>61 ± 0.947</b>	298 ± 2.83	<b>0.176 ± 0.0082</b>	0.213 ± 0.000189	5.5
0.3	<b>62.9 ± 2.58</b>	282 ± 12	<b>0.175 ± 0.0194</b>	0.404 ± 0.00924	16
0.5	<b>64.8 ± 4.46</b>	286 ± 2.12	<b>0.237 ± 0.0213</b>	0.544 ± 0.00403	28
0.7	<b>70.4 ± 9.32</b>	306 ± 0.707	<b>0.325 ± 0.0228</b>	0.568 ± 0.0106	38
0.9	<b>90.7 ± 19.6</b>	254 ± 17.7	<b>0.424 ± 0.0215</b>	0.703 ± 0.00372	48
1.1	<b>147 ± 42.8</b>	291 ± 4.24	<b>0.569 ± 0.0412</b>	0.713 ± 0.000775	59
2	506 ± 94.4	<b>486 ± 3.54</b>	1.53 ± 0.0778	<b>0.878 ± 0.0277</b>	110
5	<b>1110 ± 96.9</b>	2630 ± 12	4.58 ± 0.0818	<b>4.49 ± 0.019</b>	270
<b>Dataset 5</b>					
0.1	<b>429 ± 33.7</b>	1130 ± 14.1	<b>0.187 ± 0.0637</b>	0.216 ± 0.00045	5.5
0.3	<b>417 ± 27.7</b>	1240 ± 10.6	<b>0.272 ± 0.0227</b>	0.429 ± 0.00182	16
0.5	<b>508 ± 14.6</b>	1360 ± 180	<b>0.391 ± 0.00397</b>	0.555 ± 0.0629	28
0.7	<b>610 ± 41</b>	1430 ± 12	<b>0.534 ± 0.0165</b>	0.652 ± 0.00449	38
0.9	<b>661 ± 20.7</b>	1710 ± 247	<b>0.719 ± 0.00217</b>	0.748 ± 0.0141	48
1.1	<b>788 ± 197</b>	2000 ± 28.3	0.921 ± 0.0331	<b>0.871 ± 0.0218</b>	59
2	<b>1080 ± 2.49</b>	2560 ± 103	1.83 ± 0.0201	<b>1.76 ± 0.0351</b>	110
5	<b>1730 ± 73</b>	2900 ± 9.19	<b>4.88 ± 0.0242</b>	4.93 ± 0.02	270
<b>Dataset 6</b>					
0.1	<b>1260 ± 15.7</b>	4220 ± 25.5	<b>0.506 ± 0.00908</b>	0.758 ± 0.000811	12
0.3	<b>1240 ± 58.2</b>	4450 ± 33.2	<b>0.539 ± 0.0335</b>	0.769 ± 0.000792	34
0.5	<b>1240 ± 14.3</b>	4090 ± 31.8	<b>0.601 ± 0.00416</b>	1.02 ± 0.00611	57

Table 2: Performance estimation and selected number of zones of the adaptive rid (*continued*)

Noise standard deviation	Model dimension		RMSE		
	Adaptive Ridge	FLSA	Adaptive Ridge	FLSA	No Regularization
0.7	<b>1170 ± 19.6</b>	3920 ± 632	<b>0.762 ± 0.00647</b>	1.18 ± 0.179	80
0.9	<b>1340 ± 75.6</b>	3660 ± 59.4	<b>0.828 ± 0.0391</b>	1.32 ± 0.00631	100
1.1	<b>1440 ± 2.84</b>	3900 ± 23.3	<b>0.961 ± 0.0141</b>	1.34 ± 0.00494	130
2	<b>2200 ± 332</b>	6610 ± 13.4	1.67 ± 0.0175	<b>1.47 ± 0.000724</b>	230
5	<b>5540 ± 60</b>	11500 ± 11.3	4.62 ± 0.0131	<b>4.49 ± 0.016</b>	570

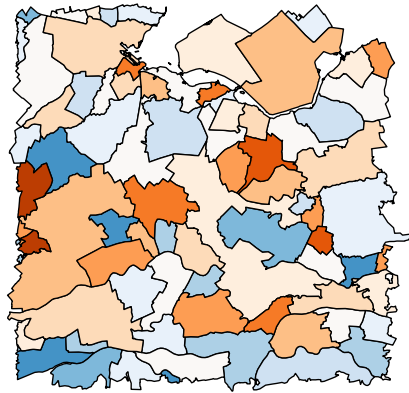
*The adaptive ridge recovers the zones well.* Our method performs well in terms of estimating  $\mathbf{q}$  (in terms of RMSE). We now verify that it performs well at estimating the zones.

We can first assess this by visual inspection. Figure 5 illustrates one estimate from Dataset 4, with  $\sigma = 0.5$  and using the AIC. Out of the 99 true zones, our method estimates 161 different areas where flsa estimates 1231. The overlay plot (Figures 5d) and 5f)) shows that the adaptive ridge estimate has a few more areas of small size, but recovers the correct shape of most areas. In comparison, the flsa estimates too many zones.

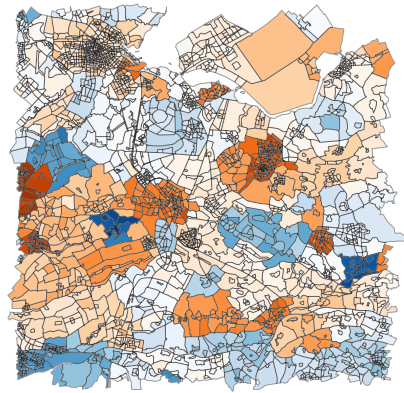
We can also quantify how well the estimated zones fit the true zones. Consider the problem of estimating zones as a problem of clustering, where the data points are the areas. Using this viewpoint, we use the Rand index [26] to measure how well each method estimates the true zones. We obtain a Rand index of 0.98 for the adaptive ridge and 0.96 for flsa and an adjusted Rand index of 0.85 for our method and 0.07 for flsa.

*Comparing adaptive ridge and flsa runtime.* We want to compare the computation time of our method to that of flsa. To that end we run both methods on planar graphs of different size. We use the adjacency graph of the neighborhoods on the whole of Netherlands ( $p = 12920$ , see Figure 1f) and consider connected subgraphs by selecting a node and adding all vertices within a certain (graph-based) distance of that node. We generate signals as iid normalized Gaussian samples and run both methods on a grid of 50 values of  $\lambda$ . The computing times are summarized in Figure 6. The flsa is faster than the adaptive ridge for small graphs (less than 1000 nodes) and the adaptive ridge is faster for large graphs (more than 3000 nodes).

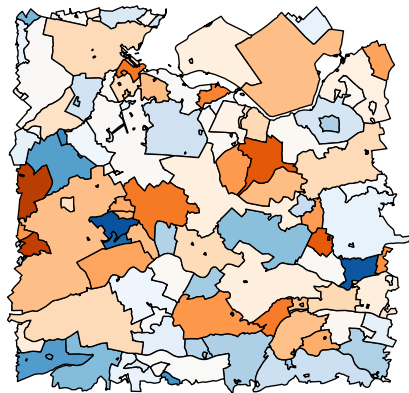
Runtime experiments (not show here) show that our method’s runtime edge increases as the graph is more connected.



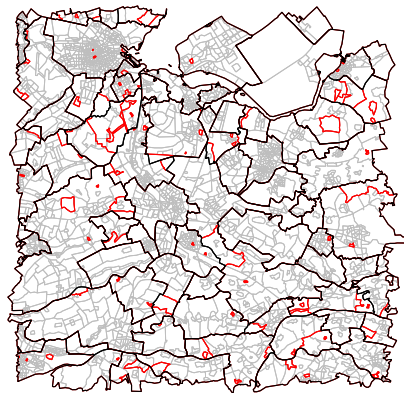
(a) True signal (areas not shown)



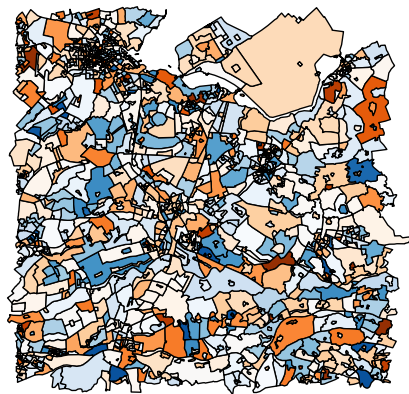
(b) Noisy signal



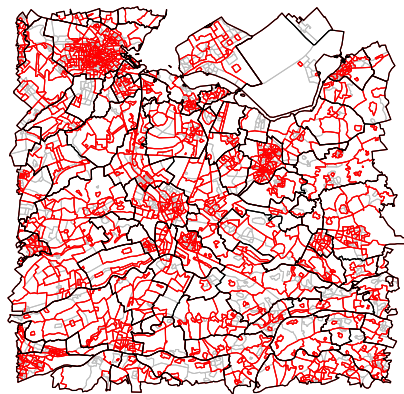
(c) Estimated signal - agraph



(d) Difference between true and estimated - agraph



(e) Estimated signal - flsa



18 (f) Difference between true and estimated - flsa

Figure 5: Illustration of the piecewise constant estimate obtained by our method on Dataset 4. *a)* True piecewise constant signal  $\theta$ ; *b)* Raw signal  $x$  generated using  $\sigma = 0.5$ ; *c)* and *e)* Estimated signal  $\hat{\theta}$  using the AIC; *d)* and *f)* Representation of the difference between the true (black) and estimated (red) signals. The panels share the same color scale.

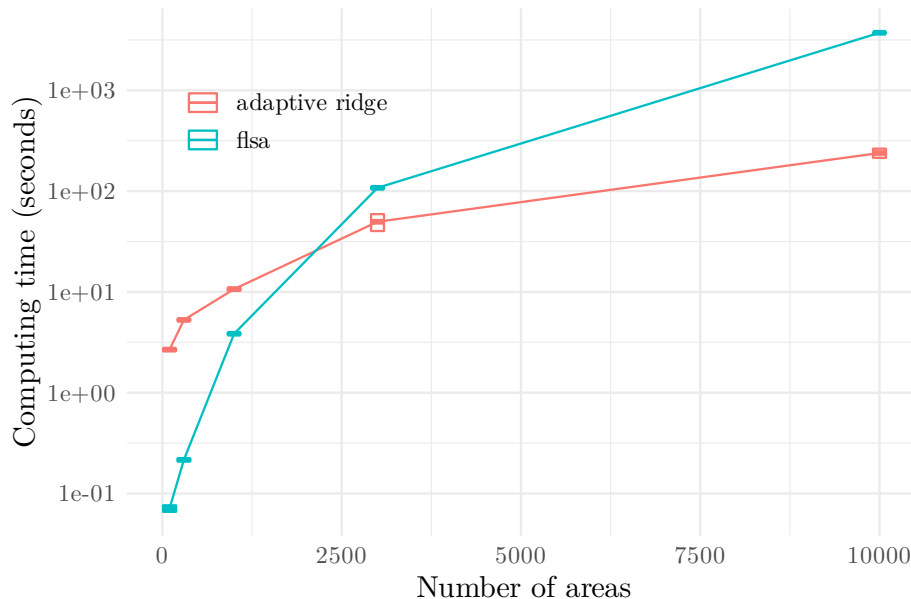


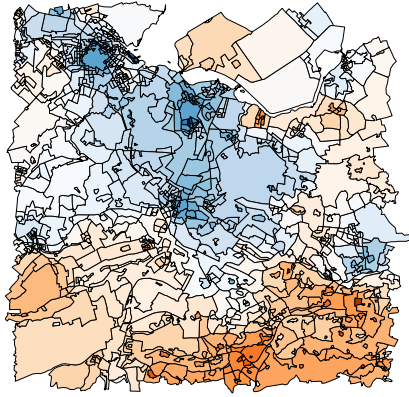
Figure 6: Computing time in seconds of both methods for datasets of different sample sizes.

### 3.2. Spatial segmentation of overweight in the Netherlands

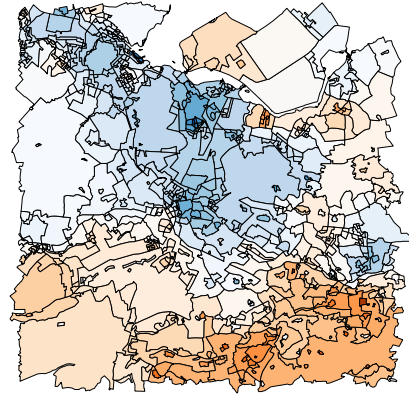
Figure 7 shows the result of the spatial segmentation of the log-odds ratios, i.e., the neighborhood effects, for the 2,955 neighborhoods in the Netherlands. The models selected by the selection criteria are either underpenalized (model dimension of 2558.3 for GCV) or overpenalized (model dimensions of 18.9 for AIC and 4.7 for BIC). Consequently, we display on Figure 7 the spatial segmentation for four different penalties, corresponding to model dimensions of (a) 1020, (b) 700, (c) 457, and (d) 158.

The fusion of areas is not happening uniformly. This can be seen for example in the south-west corner north of Dordrecht, where higher-than-expected dark region does not get fused with its neighbouring areas, while the background gets more and more segmented into larger zones with piecewise constant values.

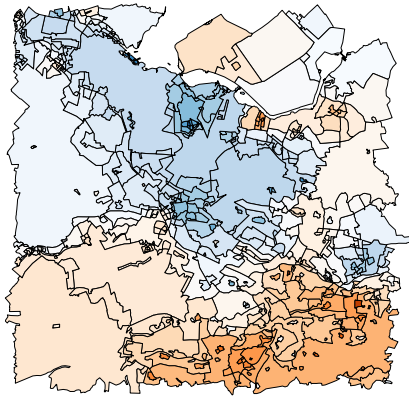
Remarkably, large number of neighborhoods are selected as being different from their surrounding neighborhoods, even as the penalty gets very large. Typical examples is the IJburg island east of Amsterdam, the former fishers town Bunschoten-Spakenburg north of Amersfoort, and some small villages in the south.



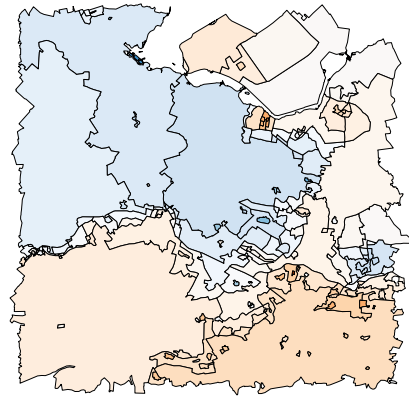
(a) With 1021 zones



(b) With 700 zones



(c) With 457 zones



(d) With 158 zones

Figure 7: Segmented spatial effect for overweight for different penalties. The results correspond to model dimensions of (a) 1020, (b) 700.0, (c) 457, and (d) 158 (corresponding to 1097, 744, 504, and 163 selected zones, respectively). The selected zones are delimited with thick lines.

## 4. Discussion

### 4.1. Simulation study

We have run our method on simulated data from 6 simulation settings, varied levels of noise standard deviation. We have compared it to the *flsa*, in terms of RMSE, model dimension, and ability to recover an accurate estimate of the zones. Our method estimates signals of slightly lower RMSE for most noise levels (except for the high noise levels), but of greatly lower dimensions. This comes from the fact that the adaptive ridge solves an  $L_q$  penalized problem, with  $q \rightarrow 0^+$  [see 20, Section S1]. This is a desired property for obtaining zones that are easier to interpret. Moreover, the estimation of the zones, seen as a task of clustering the areas, is of better quality for our method. Our method runs slower than *flsa* for small and medium number of areas ( $\leq 1000$ ) but runs faster for large numbers of areas ( $\geq 3000$ ).

### 4.2. Real-data application

To start with, for the real-data application, the identification of zones with higher or lower prevalence than expected was done in two steps: first generate data, then apply the segmentation to these data. This is done for two reasons. First, it is for computation reasons impossible to combine these two steps. The iterative nature of the adaptive ridge would require refitting the small area estimation model many times. One model fit takes a few minutes, so the total computation time would take hours or days. Second, the small area estimation model is fit on individual data. These data are only available in a secured environment hosted by Statistics Netherlands. Outside this environment, it is only allowed to work with aggregated data.

With regard to the spatial segmentation, our method has identified zones that have higher or lower overweight prevalence than can be expected based on the demographic and socio-economic characteristics of neighborhoods alone. However, the total number of zones is still relatively large, here in the order of hundreds of zones. This is caused by neighborhoods that have a particular high or low value compared to their adjacent neighborhoods. From a practical point of view, one still has to visualize the results.

We have found that the BIC does not provide a satisfying selection: only one zone. The question of which penalty parameter to favor depends on the how simple *vs* intricate we want the *blue zones* that are inferred by the model to be. We note that different penalties provide different levels of information: 7a simplifies the original data significantly while retaining much

information about neighborhoods with specifically higher or lower prevalence than expected, while 7d gives extended zones which are easier to interpret.

This collection of estimates with decreasing levels of granularity can serve policy makers as a tool to choose not only the optimal spatial distribution of the neighborhoods, but also the spatial scale to target their health-improving strategies. For example, a policy maker could focus either on the zone in the south-east (high odds ratio) when targeting on a large scale, or, on the other hand, focus on a specific neighborhood when targeting on a small scale.

We only considered overweight here. We also looked at other health-related indicators, like smoking, alcohol use and self-reported health. If blue zones would exist in the Netherlands, we would expect to see similar patterns of the spatial effect term. However, this was not the case. For example, the spatial term for overweight was negatively correlated with the spatial term for smoking (-0.24) and heavy drinking (-0.49). We found a "positive" correlation with self-reported health (0.41), i.e. a lower than expected overweight is positively associated with a better self-reported health. It is however outside the scope of this paper to explain these discrepancies.

The interpretability of the blue zones obtained by segmenting the overweight spatial term is somewhat limited. Considering the good performance of our method on simulated data, this does not question the quality of the method. Rather, it hints that our method may be too simplistic for this application, as no other covariates are included inside the model.

#### *4.3. Methodological discussion & conclusion*

This work introduces a method for graph-based signal segmentation applied to detecting piecewise constant effects in areal data.

The application is based on the assumptions that the areal discretization is adapted to the geographical distribution, in that not too much information about the distribution is lost when discretizing. Moreover, the areal data is considered only through its adjacency structure. Thus, the segmentation obtained is of good quality if the graphical distance is close to the geographical distance. This is usually the case for most administrative units, as the division into areas is fairly regular.

Besides having data values for each area, our method also requires a measure of precision of these values. This determines the weights that areas receive. If a value has been observed with a low precision, the corresponding area gets less weight and will be more quickly fused with its adjacent areas. It is also important to take the correlation (i.e. covariance or precision matrix)

between areas into account. A positive correlation will oppose the penalization, while a negative correlation will facilitate the penalization. If no covariance information is available, one can still use a diagonal matrix with variances. If no variance information is available, one can use the identity matrix. In that case all areas receive equal weights.

For large graphs, our iterative method for segmentation over a graph is competitive with similar fused-type penalties. Its application to spatial data has shown to yield sparse models, which can make the spatial effect simpler to interpret. Note that our method applies to disconnected, non-planar graphs, with possible applications to graph signal processing.

Our method also has some limitations. It makes the underlying assumption that the division into areas is regular enough, which is not always met in spatial statistics. Moreover, when  $\Sigma$  is known, the method is computationally efficient under the assumption that  $\Sigma$  is sparse, which can limit its application to other types of problems.

Future works include extending the method to the case of linear regression:  $\mathbf{x} \sim \mathcal{N}(\mathbf{X}\boldsymbol{\theta}, \Sigma)$ , where  $\mathbf{X}$  is the design matrix. Indeed, in this case, the penalized likelihood takes the form  $\frac{1}{2}(\mathbf{x} - \mathbf{X}\boldsymbol{\theta})^\top \Sigma^{-1}(\mathbf{x} - \mathbf{X}\boldsymbol{\theta}) + \frac{\lambda}{2}\boldsymbol{\theta}^\top \mathbf{K}^{(l)}\boldsymbol{\theta}$ . Consequently, our method can be extended to the linear model with graph-based penalty by replacing 6 with  $\boldsymbol{\theta}^{(l)} = (\mathbf{X}^\top \Sigma^{-1} \mathbf{X} + \lambda \mathbf{K}^{(l-1)})^{-1} \mathbf{X}^\top \Sigma^{-1} \mathbf{x}$ .

Further, the model can be extended to the general linear model, where the errors follow a distribution inside the exponential family. Indeed, the generalized linear model is estimated using the Iteratively Reweighted Least Squares (IRLS) procedure [27, Section 2.5], which solves a reweighted ridge problem. Since the adaptive ridge is also based on iterations over a ridge problem, extension to the exponential family of distributions would consist in replacing the matrix  $\Sigma^{-1}$  in (6) with a diagonal matrix, the entries of which are update at each step  $l$ .

To summarize, we have presented a new method for spatial segmentation of areal data. It uses the adaptive ridge technique to penalize over the differences between adjacent areas. The method yields a segmented estimate of the spatial effect in a computationally efficient way. The model only requires areal data values and the adjacency structure as input. The method is shown to perform well, yielding estimates sparser than the lasso-based method we used for comparison. The method can assist policy makers with their health-improving strategies. An implementation of our method is publicly available as an R package at [github.com/goepp/graphseg](https://github.com/goepp/graphseg).



### *Acknowledgements*

The authors would like to thank Pr. Olivier Bouaziz for his useful remarks which helped improve the quality of this paper. This work was partially funded by the National Institute for Public Health and the Environment (RIVM) through its Strategic Research Programme (SPR) which contributes to solutions to societal challenges through interdisciplinary research and by supporting innovation and capacity building at RIVM. This work was also partially funded by the French Foundation for Medical Research ("Fondation pour la Recherche Médicale").

### *Declaration of competing interests.*

The authors declare no competing interests.

### *Author contributions*

Conceptualization JvdK and VG; Data curation JvdK; Formal analysis VG and JvdK; Funding acquisition JvdK; Investigation VG; Methodology VG; Project administration VG; Resources VG and JvdK; Software VG; Supervision JvdK and VG; Validation JvdK and VG; Visualization VG; Roles/Writing – original draft JvdK and VG; Writing – review & editing N/A

### **References**

- [1] A. B. Lawson, S. Banerjee, R. P. Haining, M. D. Ugarte, Handbook of Spatial Epidemiology, CRC Press (2016) 704.
- [2] N. Cressie, Statistics for Spatial Data, Wiley Series in Probability and Statistics, 1993.
- [3] M. Poulain, G. M. Pes, C. Grasland, C. Carru, L. Ferrucci, G. Baggio, C. Franceschi, L. Deiana, Identification of a geographic area characterized by extreme longevity in the Sardinia island: The AKEA study, *Experimental Gerontology* 39 (2004) 1423–1429.
- [4] J. van de Kasstele, L. Zwakhals, O. Breugelmans, C. Ameling, C. van den Brink, Estimating the prevalence of 26 health-related indicators at neighbourhood level in the Netherlands using structured additive regression, *International Journal of Health Geographics* 16 (2017).

- [5] S. Kim, K.-A. Sohn, E. P. Xing, A multivariate regression approach to association analysis of a quantitative trait network, *Bioinformatics* 25 (2009) i204–i212.
- [6] X. Chen, S. Kim, Q. Lin, J. G. Carbonell, E. P. Xing, Graph-Structured Multi-task Regression and an Efficient Optimization Method for General Fused Lasso, *arXiv:1005.3579 [cs, math, stat]* (2010). [arXiv:1005.3579](#).
- [7] H. Hoefling, A Path Algorithm for the Fused Lasso Signal Approximator, *Journal of Computational and Graphical Statistics* 19 (2010) 984–1006.
- [8] Y.-X. Wang, J. Sharpnack, A. J. Smola, R. J. Tibshirani, Trend Filtering on Graphs, *Journal of Machine Learning Research* 17 (2016) 1–41.
- [9] W. Tansey, J. G. Scott, A Fast and Flexible Algorithm for the Graph-Fused Lasso, *arXiv:1505.06475 [stat]* (2015). [arXiv:1505.06475](#).
- [10] R. C. A. Rippe, J. J. Meulman, P. H. C. Eilers, Visualization of Genomic Changes by Segmented Smoothing Using an L0 Penalty, *PLoS ONE* 7 (2012) e38230.
- [11] F. Frommlet, G. Nuel, An Adaptive Ridge Procedure for L0 Regularization, *PLoS ONE* 11 (2016) e0148620.
- [12] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, P. Vandergheynst, The Emerging Field of Signal Processing on Graphs: Extending High-Dimensional Data Analysis to Networks and Other Irregular Domains, *IEEE Signal Processing Magazine* 30 (2013) 83–98. [arXiv:1211.0053](#).
- [13] R. Tibshirani, Regression Shrinkage and Selection via the Lasso, *Journal of the Royal Statistical Society. Series B* 58 (1996) 267–288.
- [14] E. J. Candès, M. B. Wakin, S. P. Boyd, Enhancing Sparsity by Reweighted  $\ell_1$  Minimization, *Journal of Fourier Analysis and Applications* 14 (2008) 877–905.
- [15] I. Daubechies, R. DeVore, M. Fornasier, C. S. Gunturk, Iteratively re-weighted least squares minimization for sparse recovery, *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences* 63 (2008) 1–38.

- [16] Y. Chen, T. A. Davis, W. W. Hager, S. Rajamanickam, Algorithm 887: CHOLMOD, Supernodal Sparse Cholesky Factorization and Update/Downdate, *ACM Transactions on Mathematical Software* 35 (2008) 1–14.
- [17] T. Hastie, J. Friedman, R. Tibshirani, *The Elements of Statistical Learning*, Springer Series in Statistics, 2nd ed., Springer New York, 2009.
- [18] G. Schwarz, Estimating the Dimension of a Model, *The Annals of Statistics* 6 (1978) 461–464.
- [19] H. Akaike, A new look at the statistical model identification, *IEEE Transactions on Automatic Control* 19 (1974) 716–723. doi:10.1109/TAC.1974.1100705.
- [20] V. Goepp, J.-C. Thalabard, G. Nuel, O. Bouaziz, Regularized bidimensional estimation of the hazard rate, *The International Journal of Biostatistics* (2021). doi:10.1515/ijb-2019-0003.
- [21] J. Fan, R. Li, Variable Selection via Nonconcave Penalized Likelihood and its Oracle Properties, *Journal of the American Statistical Association* 96 (2001) 1348–1360.
- [22] J. Fan, R. Li, Statistical Challenges with High Dimensionality: Feature Selection in Knowledge Discovery, *Proceedings of the International Congress of Mathematicians III* (2006) 595–622. arXiv:math/0602133.
- [23] H. Hoeffling, *flsa: Path Algorithm for the General Fused Lasso Signal Approximator*, 2020. URL: <https://CRAN.R-project.org/package=flsa>, r package version 1.5.2.
- [24] J. Friedman, T. Hastie, R. Tibshirani, Sparse Inverse Covariance Estimation with the Graphical Lasso, *Biostatistics* 9 (2008) 432–441.
- [25] T. Zhao, H. Liu, K. Roeder, J. Lafferty, L. Wasserman, The huge Package for High-dimensional Undirected Graph Estimation in R, *Journal of Machine Learning Research* 13 (2012) 1059–1062.
- [26] W. M. Rand, Objective criteria for the evaluation of clustering methods, *Journal of the American Statistical association* 66 (1971) 846–850.

- [27] P. McCullagh, J. A. Nelder, *Generalized Linear Models*, Chapman & Hall/CRC Monographs on Statistics & Applied Probability, 2 ed., Chapman and Hall, 1989.

## Appendix A. Convergence criterion for the algorithm

Define the weighted differences

$$\delta_{j,k}^{(l)} \triangleq v_{j,k}^{(l)} (\theta_j^{(l)} - \theta_q^{(l)})^2 = \frac{(\theta_j^{(l)} - \theta_q^{(l)})^2}{(\theta_j^{(l)} - \theta_q^{(l)})^2 + \varepsilon}.$$

With  $\varepsilon$  very small, as the sequence  $\boldsymbol{\theta}^{(l)}$  converges, the  $\delta_{j,k}^{(l)}$ s tend to either zero if the two values are close to equal, or one if the two values are different. Consequently, this quantity serves to diagnose the convergence of the algorithm.

More precisely, the stopping criteria is when the absolute differences between two consecutive values  $\delta_{j,k}$  are smaller than a fixed tolerance, for all pairs  $j \sim k$ . This tolerance parameter is set to  $10^{-8}$  in our implementation. We chose a very small tolerance to make sure that the algorithm has converged, but we advise taking a higher tolerance (e.g.,  $10^{-6}$ ) makes the estimating procedure run faster (for illustration, the real data application runs on a Intel Core i7 CPU in 1,130 seconds with a tolerance of  $10^{-8}$ , versus 625 seconds with a tolerance of  $10^{-6}$ ).

At convergence, the estimated model is not sparse. Therefore we use a cutoff of 0.99 to round the  $\delta_{j,k}^{(l)}$ s to zero or one. The cutoff is purposefully set to a high value, which ensures that we only remove vertices between adjacent neighborhoods with very different spatial effects. Thus, our method is conservative in separating two neighborhoods.

Adjacent areas with a weighted difference of zero have been estimated to have the same spatial effect. This yields a partition of the areas into a set of connected subgraphs who each have been estimated to have the same underlying spatial effect. These subgraphs are the estimated zones.

Note that the algorithm developed here induces shrinkage of  $\hat{\boldsymbol{\theta}}$ . This is in contrast with the initial implementation of [11], which uses a two-step approach, using the adaptive ridge to estimate the zones, and then estimating the effect on each zones using unpenalized estimation. Our approach using shrinkage was giving better results on simulations (results not shown here).