# Statistical Disclosure Control when Publishing on Thematic Maps

Douwe Hut[1], Jasper Goseling[1,3], Marie-Colette van Lieshout[3,1], Peter-Paul de Wolf[2], and Edwin de Jonge[2]

[1] University of Twente, Enschede, The Netherlands
d.a.hut@student.utwente.nl, j.goseling@utwente.nl
[2] Statistics Netherlands, The Hague, The Netherlands
pp.dewolf@cbs.nl, e.dejonge@cbs.nl
[3] Centrum Wiskunde & Informatica, Amsterdam, The Netherlands
m.n.m.van.lieshout@cwi.nl

**Abstract.** The spatial distribution of a variable, such as the energy consumption per company, is usually plotted by colouring regions of the study area according to an underlying table which is already protected from disclosing sensitive information. The result is often heavily influenced by the shape and size of the regions. In this paper, we are interested in producing a continuous plot of the variable directly from microdata and we protect it by adding random noise. We consider a simple attacker scenario and develop an appropriate sensitivity rule that can be used to determine the amount of noise needed to protect the plot from disclosing private information.

## 1 Introduction

Traditionally, statistical institutes mainly publish tabular data. For the tabular data and underlying microdata, many disclosure control methods exist [10]. A straightforward way to visualise the spatial structure of the tabular data on a map is to colour the different regions of the study area according to their value in a table that was already protected for disclosure control. The connection between disclosure control in tables and on maps is investigated in [16,18], for example.

Drawbacks of giving a single colour to the chosen regions are that the shape of the region influences the plot quite a lot and that the regions might not constitute a natural partition of the study area. This makes it difficult for a user to extract information from the plot. A smooth plot is often easier to work with.

To overcome these disadvantages, more and more publications use other visualisation techniques, such as kernel smoothing, that can be used to visualise data originating from many different sources, including road networks [3], crime numbers [6], seismic damage figures [7] and disease cases [8]. More applications and other smoothing techniques are discussed in [4,5,19].

---

The views expressed in this paper are those of the authors and do not necessarily reflect the policy of Statistics Netherlands.

Research involving the confidentiality of locations when publishing smoothed density maps [14,20] shows that it is possible to retrieve the underlying locations whenever the used parameters are published.

Regarding plots of smoothed averages, [13,22] constructed a cartographic map that showed a spatial density of the relative frequency of a binary variable, such as unemployment per capita. The density was defined at any point, not just at raster points, but the final colouring of the map was discretised, as part of the disclosure control. By the fact that often only one of the values of the variable is considered sensitive information, e.g. being unemployed versus being employed, a practical way to protect locations with too few nearby neighbours is assigning them to the non-sensitive end of the frequency scale. Besides assessing the disclosure risk, some utility measures were constructed.

The starting point for the current research is [23], in which plotting a sensitive continuous variable on a cartographic map using smoothed versions of cell counts and totals is discussed. The authors constructed a $p\%$ rule that used the smoothed cell total and smoothed versions of the largest two contributions per cell.

In this paper, we provide another view on the sensitivity of a map that shows a continuous variable and abandon the idea of explicitly using grid cells, so that the result will be a continuous visualisation on a geographical map. First, in Sect. 2, we will introduce some preliminaries. Then, Sect. 3 will show that the application of disclosure control is needed, after which our method to do so is explained in Sect. 4 and guaranteed to sufficiently protect the sensitive information in Sect. 5. We illustrate our approach by means of a case study in Sect. 6 and make some final remarks in Sect. 7.

## 2 Preliminaries and Notation

First, we will introduce some notation. Let $\mathcal{D} \subset \mathbb{R}^2$ be an open and bounded set that represents the study region on which we want to make the visualisation. Let the total population be denoted by $\mathcal{U} = \{\boldsymbol{r}_1, \ldots, \boldsymbol{r}_N\} \subset \mathcal{D}$, for $N \in \mathbb{N}$, in which $\boldsymbol{r}_i = (x_i, y_i)$ is the representation of population element $i$ by its Cartesian coordinates $(x_i, y_i)$. We write $\boldsymbol{r} = (x, y)$ for a general point in $\mathcal{D}$ and $||\boldsymbol{r}|| = \sqrt{x^2 + y^2}$ for the distance of that point to the origin. Associated with each population element is a measurement value. By $g_i \geq 0$, we will denote the value corresponding to population element $i$. As an example, $\mathcal{U}$ could be a set of company locations, where company $i$ has location $\boldsymbol{r}_i$ and measurement value $g_i$, indicating its energy consumption, as in our case study of Sect. 6.

In order to visualise the population density, one can use kernel smoothing [19]. The approach is similar to kernel density estimation [17], except that no normalisation is applied. Essentially, density bumps around each data point are created and added to make a total density. In our case, the kernel smoothed population density is given by

$$f_h(\boldsymbol{r}) = \frac{1}{h^2} \sum_{i=1}^{N} k\left(\frac{\boldsymbol{r} - \boldsymbol{r}_i}{h}\right),$$

in which $k\colon \mathbb{R}^2 \to \mathbb{R}$ is a so-called kernel function, that is, a non-negative, symmetric function that integrates to 1 over $\mathbb{R}^2$. The bandwidth $h$ controls the range of influence of each data point. The Gaussian kernel $k(\boldsymbol{r}) = (1/2\pi)\exp(-||\boldsymbol{r}||^2/2)$, the Epanechnikov kernel $k(\boldsymbol{r}) = (2/\pi)(1 - ||\boldsymbol{r}||^2)\mathbb{1}(||\boldsymbol{r}|| \leq 1)$ and the uniform kernel $k(\boldsymbol{r}) = (1/\pi)\mathbb{1}(||\boldsymbol{r}|| \leq 1)$ are common choices, but obviously many others kernel functions exist. Some guidelines are given in Sect. 4.5 of [19].

For the measurements values $g_1, \ldots, g_N$, a density can be constructed by multiplying the kernel corresponding to location $i$ with the value $g_i$:

$$g_h(\boldsymbol{r}) = \frac{1}{h^2}\sum_{i=1}^{N} g_i k\left(\frac{\boldsymbol{r} - \boldsymbol{r}_i}{h}\right).$$

By dividing the two densities $f_h$ and $g_h$, we get the Nadaraya-Watson kernel weighted average [21]

$$m_h(\boldsymbol{r}) = \frac{g_h(\boldsymbol{r})}{f_h(\boldsymbol{r})} = \frac{\sum_{i=1}^{N} g_i k\left((\boldsymbol{r} - \boldsymbol{r}_i)/h\right)}{\sum_{i=1}^{N} k\left((\boldsymbol{r} - \boldsymbol{r}_i)/h\right)}, \quad \boldsymbol{r} \in \mathcal{D}. \tag{1}$$

Whenever $f_h(\boldsymbol{r}) = 0$, it follows that $g_h(\boldsymbol{r}) = 0$ as well and we define $m_h(\boldsymbol{r}) = 0$. This weighted average is an excellent tool for data visualisation and analysis [5]. The ratio $m_h(\boldsymbol{r})$, $r \in \mathcal{D}$ will be the function of which we will investigate disclosure properties and discuss a possible protection method.

Some remarks are in order. Firstly, the bandwidth $h$ influences the smoothness of $m_h$. In the limit case of a very large bandwidth, $m_h$ will be constant, while for small $h$, the plot will contain many local extrema. In the limit case of a very small bandwidth, $m_h$ will be the nearest neighbour interpolation, at least when using a Gaussian kernel. Secondly, note that mass can leak away, since $\mathcal{D}$ is bounded but the kernel is defined on $\mathbb{R}^2$. Consequently, $f_h$ and $g_h$ underestimate the (weighted) population density at $\boldsymbol{r}$ close to the boundary of $\mathcal{D}$. Various techniques to correct such edge effects exist, see [2,9,15].

In this paper, we will frequently use two matrices that are defined in terms of the kernel function, namely

$$\boldsymbol{K}_h = \left(k\left(\frac{\boldsymbol{r}_i - \boldsymbol{r}_j}{h}\right)\right)_{i,j=1}^{N}$$

and

$$\boldsymbol{C}_h = \left(\frac{k\left((\boldsymbol{r}_i - \boldsymbol{r}_j)/h\right)}{\sum_{k=1}^{N} k\left((\boldsymbol{r}_i - \boldsymbol{r}_k)/h\right)}\right)_{i,j=1}^{N}.$$

Lastly, we will write $\Phi^{-1}$ for the standard normal inverse cumulative distribution function.

## 3 Motivation and Attacker Scenario

In this section, we will show that publishing the kernel weighted average reveals exact information on the underlying measurement values. This implies that it is

necessary to apply disclosure control before publishing the plot. Our method to do so will be elaborated on in Sect. 4.

Here, we will restrict our attention to the scenario in which an attacker is able to exactly read off the plot of the kernel weighted average (1) at the population element locations $\boldsymbol{r}_i, i = 1, \ldots, N$. Throughout this paper, we will assume that he is completely aware of the method to produce the kernel weighted average and knows what kernel function, bandwidth and population element locations were used.

Using the plot values, the attacker can set up a system of linear equations to obtain estimates of the measurement values, since the kernel weighted average (1) is a linear combination of the measurement values. When the attacker chooses $N$ points to read off the plot of (1) and uses the exact locations $\boldsymbol{r}_i$ for $i = 1, \ldots, N$, he obtains the system

$$\boldsymbol{m}_h = \boldsymbol{C}_h \, \boldsymbol{g}, \tag{2}$$

with the known plot values $\boldsymbol{m}_h = (m_h(\boldsymbol{r}_i))_{i=1}^N$ and the unknown measurement value vector $\boldsymbol{g} = (\boldsymbol{g}_i)_{i=1}^N$. We know the following about solvability of the system.

**Theorem 1.** *Whenever $\boldsymbol{K}_h$ is invertible, system (2) can be solved uniquely and the attacker can retrieve all measurement values exactly.*

*Proof.* Assume that $\boldsymbol{K}_h$ is invertible. Then $\boldsymbol{C}_h$ is invertible as well, as it is created from $\boldsymbol{K}_h$ by scaling each row to sum to 1. Hence, the linear system (2) is uniquely solvable and an attacker can retrieve the vector $\boldsymbol{g}$ of measurement values by left-multiplying $\boldsymbol{m}_h$ with $\boldsymbol{C}_h^{-1}$. □

In particular, Theorem 1 shows that there is at least one configuration of points at which the attacker can read off the plot of (1) to retrieve the measurement values $g_i, i = 1, \ldots, N$ exactly.

For the Gaussian kernel, amongst others, $\boldsymbol{K}_h$ is positive definite and thus invertible, regardless of $h$, $N$ and $\boldsymbol{r}_i$, $i = 1, \ldots, N$, only provided that all $\boldsymbol{r}_i$ are distinct.

In the remainder of this paper, we will assume an attacker scenario in which the attacker obtains a vector containing the exact plot values at locations $\boldsymbol{r}_i$, $i = 1, \ldots, N$ and left-multiplies that vector by $\boldsymbol{C}_h^{-1}$ to obtain estimates of the measurement values $g_i$, $i = 1, \ldots, N$.

## 4 Proposed Method and Main Result

Our method to prevent the disclosure of sensitive information consists of disturbing the plot of (1), by adding random noise to the numerator $g(\boldsymbol{r})$, $\boldsymbol{r} \in \mathcal{D}$, so that an attacker observes

$$\tilde{m}_h(\boldsymbol{r}) = \frac{\sum_{i=1}^N g_i k\left((\boldsymbol{r} - \boldsymbol{r}_i)/h\right) + \epsilon(\boldsymbol{r})}{\sum_{i=1}^N k\left((\boldsymbol{r} - \boldsymbol{r}_i)/h\right)}, \quad \boldsymbol{r} \in \mathcal{D}, \tag{3}$$

instead of (1), where we define $\tilde{m}_h(\boldsymbol{r}) = 0$ if $f_h(\boldsymbol{r}) = 0$. The random noise $\epsilon(\boldsymbol{r})$ will be generated as a Gaussian random field, with mean 0 and covariance function

$$\operatorname{Cov}(\epsilon(\boldsymbol{r}), \epsilon(\boldsymbol{s})) = \sigma^2 k\left(\frac{\boldsymbol{r}-\boldsymbol{s}}{h}\right), \quad \boldsymbol{r}, \boldsymbol{s} \in \mathcal{D},$$

where $\sigma$ is the standard deviation of the magnitude of the added noise. The kernel $k$ should be a proper covariance function, which is the case when for all $h > 0$, $m \in \mathbb{N}$ and $\boldsymbol{s}_i \in \mathbb{R}^2$, $i = 1, \ldots, m$, the corresponding matrix $\boldsymbol{K}_h$ is positive definite, see Chapt. 1 of [1]. In this way, (3) will be continuous, just as (1), whenever a continuous kernel function is used and $f_h$ vanishes nowhere.

Adding random noise to the plot implies that the attacker's estimates will be stochastic as well. This fact should be captured in a rule that describes whether it is safe to publish the noised kernel weighted average. It brings us to the following sensitivity rule, that states that a plot is considered unsafe to publish when any measurement value estimate that the attacker makes lies with probability greater than $\alpha$ within $p$ percent of the true value. Such a sensitivity rule can be seen as a stochastic counterpart of the well known $p\%$ rule for tabular data, which is elaborated on in [10].

**Definition 1.** *For $0 < p \leq 100$ and $0 \leq \alpha < 1$, a plot is said to be* unsafe according to the $(p\%, \alpha)$ rule for an attacker scenario *whenever the estimates $\hat{g}_i$ of $g_i$, $i = 1, \ldots, N$, computed according to the scenario, satisfy*

$$\max_{i=1,\ldots,N} P\left\{\left|\frac{\hat{g}_i - g_i}{g_i}\right| < \frac{p}{100}\right\} > \alpha, \tag{4}$$

*where we take $|(\hat{g}_i - g_i)/g_i| = |\hat{g}_i|$ if $g_i = 0$.*

When applying the $(p\%, \alpha)$ rule, we normally choose $p$ and $\alpha$ to be small, so that a plot is safe when small relative errors in the recalculation happen with small probability. Theorem 1 implies that the plot of (1) cannot be safe for any $(p\%, \alpha)$ rule. Furthermore, we note that high values of $p$ and low values of $\alpha$ correspond to a stricter rule: If a plot is safe according the $(p\%, \alpha)$ rule, then for any $\tilde{p} \leq p$ and $\tilde{\alpha} \geq \alpha$, the plot is also safe according to the $(\tilde{p}\%, \tilde{\alpha})$ rule.

Our main result is the following theorem, that gives the standard deviation of the magnitude of the noise $\epsilon$ in (3) needed to ensure that the plot is safe according to the $(p\%, \alpha)$ rule. In Sect. 5, we will prove the theorem.

**Theorem 2.** *Suppose that the kernel $k\colon \mathbb{R}^2 \to \mathbb{R}$ is a proper covariance function and $g_i > 0$, $i = 1, \ldots, N$. Then the plot of (3) is safe according to the $(p\%, \alpha)$ rule for our attacker scenario of Sect. 3 if*

$$\sigma \geq \frac{p}{100\,\Phi^{-1}\left((1+\alpha)/2\right)} \max_{i=1,\ldots,N} \left\{\frac{g_i}{\sqrt{\left(\boldsymbol{K}_h^{-1}\right)_{ii}}}\right\}. \tag{5}$$

## 5 Proof of Theorem 2

Recall that the attacker observes (3). In matrix notation, (3) reads

$$\boldsymbol{m}_h + \tilde{\boldsymbol{\epsilon}} = \boldsymbol{C}_h\, \boldsymbol{g} + \tilde{\boldsymbol{\epsilon}},$$

where

$$\tilde{\boldsymbol{\epsilon}} = (\tilde{\epsilon}_i)_{i=1}^N = \left( \frac{\epsilon(\boldsymbol{r}_i)}{\sum_{j=1}^N k\left((\boldsymbol{r}_i - \boldsymbol{r}_j)/h\right)} \right)_{i=1}^N . \tag{6}$$

If the attacker left-multiplies the vector of observed plot values by $\boldsymbol{C}_h^{-1}$ to recalculate $\boldsymbol{g}$, just as he could do in (2), he will now make an error, because the observed values are $\boldsymbol{m}_h + \tilde{\boldsymbol{\epsilon}}$ instead of $\boldsymbol{m}_h$. When we write $\hat{\boldsymbol{g}} = (\hat{g}_i)_{i=1}^N$ for the vector of recalculated measurement values, we obtain

$$\hat{\boldsymbol{g}} = \boldsymbol{C}_h^{-1}(\boldsymbol{m}_h + \tilde{\boldsymbol{\epsilon}}) = \boldsymbol{g} + \boldsymbol{C}_h^{-1}\tilde{\boldsymbol{\epsilon}}. \tag{7}$$

Recall that $\boldsymbol{C}_h$ is invertible because $\boldsymbol{K}_h$ is positive definite since $k$ is a proper covariance function.

By the next lemma, that is the result of basic probability theory, it suffices, in order to prove Theorem 2, to show that for our attacker scenario of Sect. 3 and using the plot of (3), for $i = 1, \ldots, N$, the recalculated value $\hat{g}_i$ follows a normal distribution with mean $g_i$ and variance $\sigma^2 \left(\boldsymbol{K}_h^{-1}\right)_{ii}$.

**Lemma 1.** *Whenever $\hat{g}_i$ follows a normal distribution with mean $g_i$, (4) is equivalent with*

$$\max_{i=1,\ldots,N} \frac{p\ g_i}{100\ \Phi^{-1}\left(\frac{1+\alpha}{2}\right)\sqrt{\mathrm{Var}(\hat{g}_i)}} > 1.$$

Now, let us compute the variance of the recalculated measurement values. For all $i = 1, \ldots, N$, combining (7) with the fact that $\epsilon(\boldsymbol{r_i})$, $i = 1, \ldots, N$, follows a multivariate normal distribution with zero mean and covariance matrix $\sigma^2 \boldsymbol{K}_h$, the $i$-th recalculated value $\hat{g}_i$ will follow a normal distribution with mean $g_i$ and variance

$$\mathrm{Var}(\hat{g}_i) = \sum_{j=1}^N \sum_{k=1}^N \mathrm{Cov}\left( \left(\boldsymbol{C}_h^{-1}\right)_{ij} \tilde{\epsilon}_j, \left(\boldsymbol{C}_h^{-1}\right)_{ik} \tilde{\epsilon}_k \right).$$

Rewriting $\tilde{\epsilon}_j$ and $\tilde{\epsilon}_k$ according to (6), taking factors outside the covariance term and substituting $\sigma^2 \left(\boldsymbol{K}_h\right)_{jk} = \sigma^2 \left(\boldsymbol{C}_h\right)_{kj} \sum_{m=1}^N \left(\boldsymbol{K}_h\right)_{km}$ for $\mathrm{Cov}(\epsilon_j, \epsilon_k)$, we obtain

$$\mathrm{Var}(\hat{g}_i) = \sigma^2 \sum_{j=1}^N \sum_{k=1}^N \frac{\left(\boldsymbol{C}_h^{-1}\right)_{ij} \left(\boldsymbol{C}_h^{-1}\right)_{ik}}{\sum_{m=1}^N \left(\boldsymbol{K}_h\right)_{jm}} \left(\boldsymbol{C}_h\right)_{kj}.$$

Now, we can work out the multiplications of inverse matrices and use

$$\boldsymbol{K}_h^{-1} = \left( \frac{\left(\boldsymbol{C}_h^{-1}\right)_{ij}}{\sum_{m=1}^N \left(\boldsymbol{K}_h\right)_{jm}} \right)_{i,j=1}^N$$

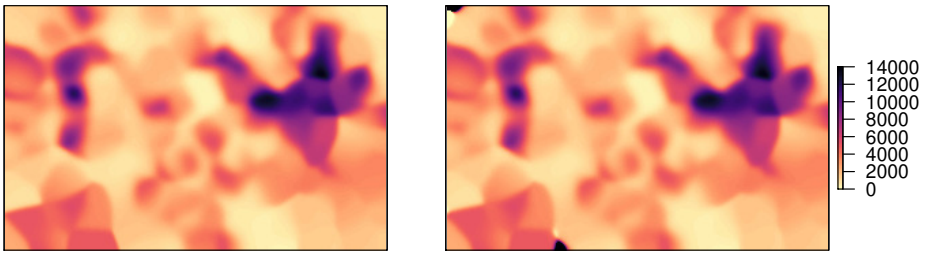


**Fig. 1.** Unprotected (left panel) and protected (right panel) kernel weighted average of our entire synthetic dataset, according to a $(10\%, 0.1)$ rule for a Gaussian kernel with bandwidth $h = 250\,\mathrm{m}$

to get the result

$$\mathrm{Var}(\hat{g}_i) = \sigma^2 \left(\boldsymbol{K}_h^{-1}\right)_{ii},$$

which, together with Lemma 1, proves Theorem 2.

## 6 Case Study

We want to be able to compare unprotected plots with protected plots, so we cannot use original, confidential data. Hence we used a synthetic dataset, based on real data of energy consumption by enterprises. The original data contained enterprises in the region 'Westland' of The Netherlands. This region is known for its commercial greenhouses as well as enterprises from the Rotterdam industrial area. We perturbed the locations of the enterprises and we assigned random values for the energy consumption drawn from a log-normal distribution with parameters estimated from the original data. We introduced some spatial dependency in the energy consumption to mimic the compact industrial area and the densely packed greenhouses. The final dataset consists of some 8348 locations and is also included in the sdcSpatial R-package that can be found on CRAN [12].

Figure 1 shows the unprotected kernel weighted average (1) and the protected kernel weighted average (3) that satisfies the $(10\%, 0.1)$ rule. A Gaussian kernel with a bandwidth of 250 m was used. We computed a safe lower bound for the standard deviation $\sigma$ of the random noise by (5). The plot of (3) resulting from that computation looks almost exactly identical to the plot of (1). Only at parts of the boundary where the population density is very small, the added disturbance is perceptible by the eye.

When the bandwidth would be taken smaller, the standard deviation of the noise would become large enough for the disturbance to be visually apparent. However, working on this scale, it would be hard to see the details in that situation. Thus, we plotted a subset of the data, restricting ourselves to a square of 2 km $\times$ 2 km and all 918 enterprises contained in that square. The results of

our method on the data subset are visible in Fig. 2 for $h = 100\,\mathrm{m}$ and in Fig. 3 for $h = 80\,\mathrm{m}$, while Fig. 4 displays the spatial structure of the locations in our entire synthetic dataset and the subset thereof.

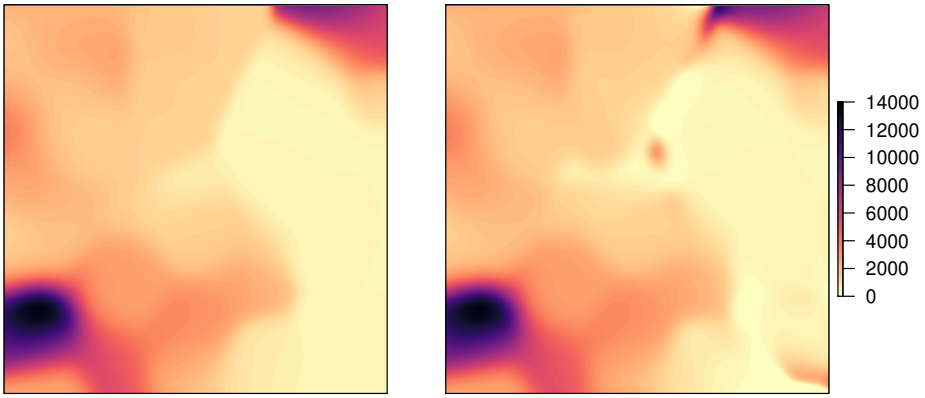


**Fig. 2.** Unprotected (left panel) and protected (right panel) kernel weighted average of a part of our synthetic dataset, according to a $(10\%, 0.1)$ rule for a Gaussian kernel with bandwidth $h = 100\,\mathrm{m}$

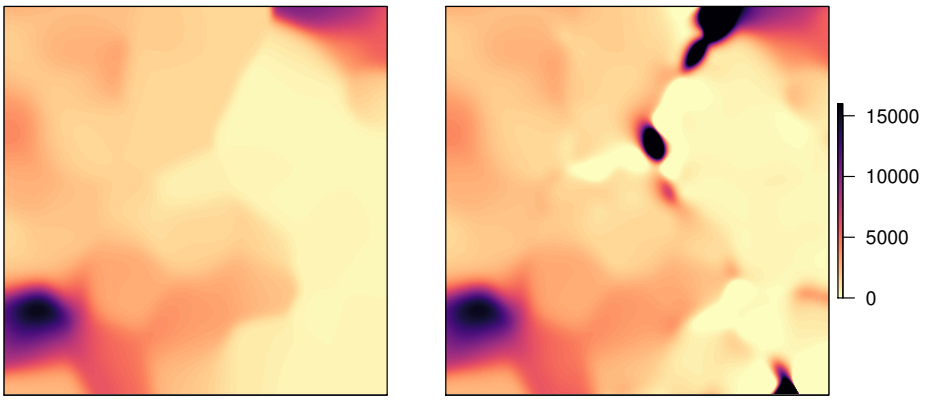


**Fig. 3.** Unprotected (left panel) and protected (right panel) kernel weighted average of a part of our synthetic dataset, according to a $(10\%, 0.1)$ rule for a Gaussian kernel with bandwidth $h = 80\,\mathrm{m}$

We see that the necessary disturbance to the plot is smaller in Fig. 3 than in Fig. 2. In order to be able to compare the results for different bandwidths, Fig.

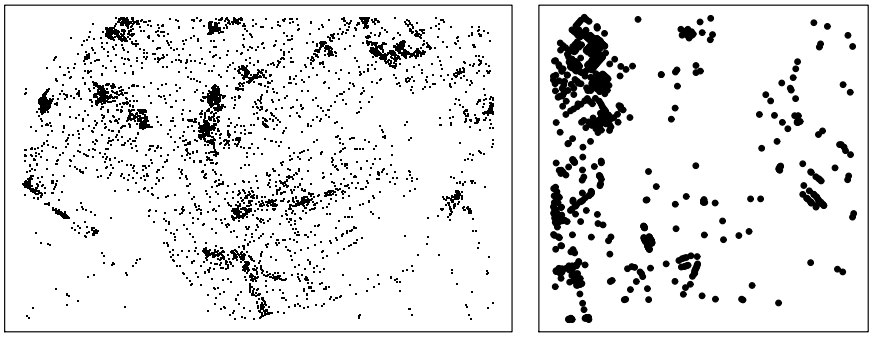

**Fig. 4.** Map of enterprise locations in our entire dataset (left panel) and in the data subset (right panel)

5 contains two graphs that show the influence of the bandwidth on $\sigma$ for our synthetic data set. Note that the total disturbance of the plot is also influenced by the denominator of (3), that increases with increasing bandwidth if the used kernel is decreasing in $||\boldsymbol{r}||$. The graph of the entire dataset shows a steep decrease of $\sigma$ around $h = 5$. This is caused by the quick increase of the diagonal elements of $\boldsymbol{K}_h^{-1}$ due to $\boldsymbol{K}_h$ becoming less similar to the identity matrix. For $h \leq 5$ a single company with a very large energy consumption dominates the value of $\sigma$. Since this company is not present in the subset that we work with, a smaller $\sigma$ may be used for the subset, also for $h \leq 5$.

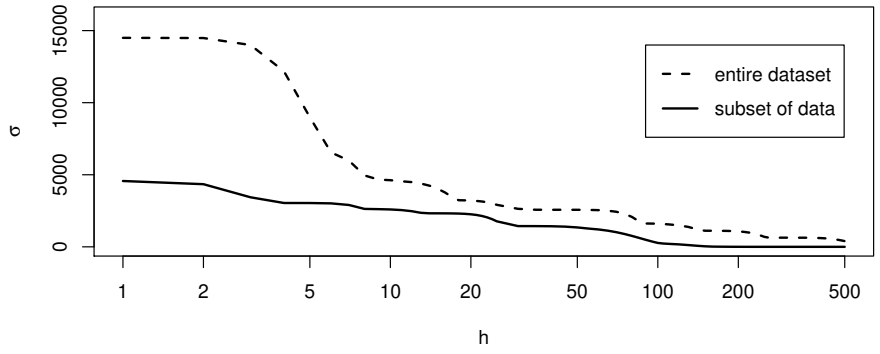


**Fig. 5.** Standard deviation $\sigma$ of added noise for different bandwidths

## 7 Discussion

In this paper we introduced a new sensitivity rule that is applicable in the scenario that an attacker knows both the kernel and the bandwidth used to produce the map, reads off the plotted values at the population elements and estimates the measurement values by solving a system of linear equations. To protect the plot, we proposed to disturb the data by adding noise and derived a rule on how large the disturbance to the plot should be before publishing it.

To investigate the efficacy of the proposed method a case study was carried out. It indicated that for a bandwidth that is large relative to the population density, the disturbance needed was very small. When zooming in, however, the disturbance to the plot was visually apparent.

During this research, some other interesting results were found that fall outside the scope of this paper. For details we refer to [11]. For instance, in our attacker scenario we assumed that the bandwidth is known to the attacker. If the bandwidth were unknown to the attacker, simulations indicate that in many cases, the bandwidth can be retrieved from the plot of (1) by repeatedly guessing a bandwidth, solving the linear system for that bandwidth, making a plot using the recalculated values and the guessed bandwidth and calculating the similarity between the original and the recovered plot.

Secondly, many kernels with a compact support, including the uniform and Epanechnikov kernel, are discontinuous or not infinitely differentiable at the boundary of their support. An attacker can often use such information to obtain the bandwidth or a single measurement value by considering plot values close to that boundary.

We close with some final remarks and perspectives. At first glance, it might seem more natural to add noise to the kernel weighted average itself rather than to the numerator of (1). However, typically more noise should then be added, resulting in a less visually attractive map. Furthermore, the proposed method agrees with the intuition that densely populated areas need less protection, since the standard deviation of the noise is inversely proportional to the kernel smoothed population density. Note that the addition of noise in our method might lead to negative or extremely large values of (3) at locations where the population density is very small. In our figures, these locations were given the minimal or maximal colour scale values, to result in a realistic map for the user.

It would be interesting to look at the utility of our plot for different bandwidths. Fig. 5 is a first step in this direction but more research is needed.

Our method requires that all $\boldsymbol{r}_i$, $i = 1, \ldots, N$ are distinct. It would be interesting to look into a scenario in which population elements can have the same location, since these might partly protect each other for disclosure. If one would introduce grid cells and use a single location for elements in the same cell, a similar analysis could lead to explicitly taking the resolution of the plot into account. Alternatively, rounding the plot values or using a discrete color scale may be a useful approach to obtaining some level of disclosure control.

Finally, we restricted ourselves to a single simple attacker scenario. It would be interesting to investigate alternative scenarios in which the attacker is particularly interested in a single value, uses other locations to read off the plot or tries to eliminate the added noise.

## References


1. Abrahamsen, P.: A review of gaussian random fields and correlation functions. Tech. Rep. 917, Norwegian Computing Center (1997)
2. Berman, M., Diggle, P.: Estimating weighted integrals of the second-order intensity of a spatial point process. Journal of the Royal Statistical Society 51, 81–92 (1989)
3. Borruso, G.: Network density and the delimitation of urban areas. Transactions in GIS 7(2), 177–191 (2003)
4. Bowman, A.W., Azzalini, A.: Applied smoothing techniques for data analysis. Oxford University Press (1997)
5. Chacón, J.E., Duong, T.: Multivariate kernel smoothing and its applications. CRC Press (2018)
6. Chainey, S., Reid, S., Stuart, N.: When is a hotspot a hotspot? a procedure for creating statistically robust hotspot maps of crime. In: Kidner, D., Higgs, G., White, S. (eds.) Innovations in GIS 9: Socio-economic applications of geographic information science. pp. 21–36. Taylor and Francis (2002)
7. Danese, M., Lazzari, M., Murgante, B.: Kernel density estimation methods for a geostatistical approach in seismic risk analysis: the case study of Potenza hilltop town (southern Italy). In: ICCSA 2008, Part I. pp. 415–429. Springer (2008), LNCS 5072
8. Davies, T.M., Hazelton, M.L.: Adaptive kernel estimation of spatial relative risk. Statistics in Medicine 29(23), 2423–2437 (2010)
9. Diggle, P.J.: A kernel method for smoothing point process data. Journal of the Royal Statistical Society 34, 138–147 (1985)
10. Hundepool, A., Domingo-Ferrer, J., Franconi, L., Giessing, S., Schulte Nordholt, E., Spicer, K., De Wolf, P.P.: Statistical Disclosure Control. Wiley series in Survey Methodology, John Wiley & Sons, Ltd (2012), ISBN: 978-1-119-97815-2
11. Hut, D.A.: Statistical disclosure control when publishing on thematic maps. Master's thesis, University of Twente, Enschede, the Netherlands (2020)
12. de Jonge, E., de Wolf, P.P.: sdcSpatial: Statistical Disclosure Control for Spatial Data, `https://CRAN.R-project.org/package=sdcSpatial`, r package version 0.2.0.9000
13. de Jonge, E., de Wolf, P.P.: Spatial smoothing and statistical disclosure control. In: Domingo-Ferrer, J., Pejić-Bach, M. (eds.) Privacy in Statistical Databases. pp. 107–117. Springer (2016), LNCS 9867
14. Lee, M., Chun, Y., Griffith, D.A.: An evaluation of kernel smoothing to protect the confidentiality of individual locations. International Journal of Urban Sciences 23(3), 335–351 (2019), DOI: 10.1080/12265934.2018.1482778
15. van Lieshout, M.N.M.: On estimation of the intensity function of a point process. Methodology and Computing in Applied Probability 14, 567–578 (2012)
16. O'Keefe, C.M.: Confidentialising maps of mixed point and diffuse spatial data. In: Privacy in Statistical Databases. pp. 226–240. Springer (2012)
17. Silverman, B.W.: Density estimation for statistics and data analysis. Chapman & Hall (1986)

18. Suñé, E., Rovira, C., Ibáñez, D., Farré, M.: Statistical disclosure control on visualising geocoded population data using quadtrees. In: extended abstract at NTTS 2017 (2017), http://nt17.pg2.at/data/x_abstracts/x_abstract_286.docx
19. Wand, M.P., Jones, M.C.: Kernel smoothing. CRC Press (1994)
20. Wang, Z., Liu, L., Zhou, H., Lan, M.: How is the confidentiality of crime locations affected by parameters in kernel density estimation? International Journal of Geo-Information 8(12), 544–556 (2019), DOI: 10.3390/ijgi8120544
21. Watson, G.S.: Smooth regression analysis. Sankhya: The Indian Journal of Statistics 26(4), 359–372 (1964)
22. de Wolf, P.P., de Jonge, E.: Location related risk and utility. Presented at UNECE/Eurostat worksession Statistical Data Confidentiality, 20–22 September, Skopje (2017), https://www.unece.org/fileadmin/DAM/stats/documents/ece/ces/ge.46/2017/3_LocationRiskUtility.pdf
23. de Wolf, P.P., de Jonge, E.: Safely plotting continuous variables on a map. In: Domingo-Ferrer, J., Montes, F. (eds.) Privacy in Statistical Databases. pp. 347–359. Springer (2018), LNCS 11126