Statistical Disclosure Control when Publishing on Thematic Maps

Douwe Hut

July 6, 2020

MSc Thesis
Stochastic Operations Research
Applied Mathematics
University of Twente

Daily Supervisors:
prof.dr. M.N.M. van Lieshout (UT)
dr.ir. J. Goseling (UT)
dr.ir. P.P. de Wolf (CBS)
drs. E. de Jonge (CBS)

Graduation Committee:
prof.dr. M.N.M. van Lieshout (UT)
dr.ir. J. Goseling (UT)
dr.ir. P.P. de Wolf (CBS)
prof.dr. A.J. Schmidt-Hieber (UT)

UNIVERSITY OF TWENTE.

Summary

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 report, we are interested in producing a continuous plot of the variable directly from microdata. We will investigate methods to recalculate the original data from the plot and see that it is needed to protect the plot from disclosing sensitive information. We give three methods to do so 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 for each of the methods.

Preface

During the past months I have been working on this research, which concludes my study in Applied Mathematics at the University of Twente. I am grateful to Statistics Netherlands for giving me the opportunity to execute my final project there. I would like to thank Peter-Paul and Edwin for introducing me into the subject of statistical disclosure control, for sharing ideas about possible approaches and for their enthusiasm about the research. Hopefully, you do not mind that of the original possible research directions that were presented to me in the beginning, many were not quite addressed since gradually we discovered another direction.

Furthermore, I would like to thank Jasper en Marie-Colette for taking the time to read my new work and meet me every week to discuss the results. I think the meetings were really fruitful and you kept asking me interesting questions.

About the main results of this research, a paper was written and submitted for the conference Privacy in Statistical Databases 2020, which will organised by the UNESCO Chair in Data Privacy from September 23 to 25. Only a few days before writing this preface, my daily supervisors and I obtained the joyful news that our paper was accepted for presentation during the conference and publication in the Springer LNCS proceedings of the conference. I would like to thank my supervisors for giving me the time to write the paper during my final project, for writing together with me and for all their suggestions and improvements, both on the details and on the overall structure. It was a nice and educational period.

During my final project, but also in the years before, my family has always supported me, which I really appreciate. I would also like to thank my friends for the nice talks and games during lunchbreaks and for telling me to go back to work afterwards.

Next, I would like to say a few words about the current situation worldwide. Halfway this research, the corona virus started spreading in The Netherlands. Just like many others, I took me some time to get used to working at home. Currently, it has been months since I last entered the buildings of Statistics Netherlands in The Hague and the university in Enschede and it feels strange to conclude my master thesis by means of an online presentation, without being able to look the audience into the eyes and say goodbye to my supervisors in person.

Lastly, I appreciate the graduation committee for taking the time and effort taken to read my work.

Contents

1 Introduction

Statistical disclosure control is an important element when producing official statistics of economic, social and demographic data. In many countries, laws ensure that no information on individual subjects like persons and companies can be deduced from the published data. The law applies to tables, but also to the new visualisation techniques that we will consider in this report. The goal of this research and an outline of the report are given in Section 1.3. First, we introduce Statistics Netherlands for which this research was carried out in Section 1.1 and present the traditional way of visualizing data on maps in Section 1.2.

1.1 Statistics Netherlands

Statistics Netherlands (Dutch: Centraal Bureau voor de Statistiek, CBS) is the Dutch national statistical institute. It was founded in 1899, because of the need to have independent and reliable data to understand social issues. Its history is described in great detail by Van Maarseveen and Schreijnders (1999). As of today, Statistics Netherlands has around 2000 employees, divided amongst their offices in The Hague and Heerlen, as well as a small office on Bonaire. This research was specifically carried out for the methodology department in The Hague. That department performs research on the methods with which statistics are created. Examples include developing methods to estimate data that are not yet available or methods to guarantee quality, reliability and unbiasedness of statistics coming from new data sources.

Still today, the primary task of Statistics Netherlands is to gather and publish statistics of the national social and economic data. While they are mainly intended to support policy makers, all of its statistical publications are publicly available. By law, these publications should fulfil the requirement that it is impossible to deduce information from them on a too detailed level, such as on individual persons, households and companies. This means that the statistical institutes face a utility versus disclosure risk trade-off when publishing data.

1.2 Visualisations

Traditionally, statistical institutes mainly publish tabulated data, for which many disclosure control methods exist. Nowadays, more publications make use of other visualisation techniques, such as plots of a spatial distribution on a map. 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.

In Figure 1.1, one can see an instance of this procedure. For the north-western part of the city of Enschede, the average gas consumption per household is shown for different grid cell sizes. A red color indicates a high gas usage, while a blue color indicates a low consumption. To make these visualisations, existing tabular data was used, where each table cell corresponds to a spatial grid cell. However, before publishing the table, it was protected for disclosure control, which in

this particular case means that data is suppressed for grid cells that contain fewer households than a certain threshold value. The plot with a 100m grid might show clearer hot spots of high gas consumption, while also a larger area of the map remains uncoloured due to the disclosure control process.

(a) 500m grid

(b) 100m grid

Figure 1.1: Gas consumption per household in Enschede, averaged over grid cells (`https://www.cbsinuwbuurt.nl`)

Instead of colouring grid cells on the map, it is of course also possible to colour the map on municipality level or neighbourhood level, if the data are available. Drawbacks of giving a single colour to the chosen regions are that the shapes of regions 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 the information from the plot. A smooth plot is often visually more appealing and easier to work with.

1.3 Goal and Outline

For reasons mentioned above, the goal of this project is to make a continuous visualisation on a geographical map, based on measurements that were taken at finitely many points. The visualisation should show spatial patterns of the measurements within the study area, but is not allowed to disclose detailed information about single measurements, since those are regarded as confidential information. In particular, it is interesting to see whether the smooth visualisation process can account for a tailor made form of disclosure control.

First, in Chapter 2, we give an overview of disclosure control methods for tabular data, introduce some preliminaries on creating continuous visualisations on maps and discuss recent research on the topic. Then, Chapter 3 discusses multiple scenarios in which it is clear that the application of disclosure control is needed. For one particular scenario, we explain three closely related methods to do so in Chapter 4. A sensitivity rule is formulated there as well and we prove conditions for our visualisation to be sufficiently protected according to the sensitivity rule. Our approach is illustrated by means of simulations and a case study in Chapter 5 and we make some final remarks in Chapter 6.

2 Preliminaries and Recent Research

Traditionally, statistical institutes mainly publish tabular data, in both quantitative tables and frequency tables. We will focus on the former ones, in which the cell value is a sum of individual contributions on a continuous scale. In Section 2.1, we give a broad overview of these methods. Special attention should be paid to the $p\%$ rule, on which we will base our own sensitivity rule in Chapter 4. We discuss recent research on the subject in Section 2.2 and introduce the concept of kernel smoothing in Section 2.3.

2.1 Tabular Statistical Disclosure Control

For publishing tabular data, different rules exist to determine whether a table cell might disclose sensitive individual information or not. We will discuss the ones most used, introduce additional theory and describe possible procedures to modify the table so as to make it safe for publication, whenever any cells are unsafe to publish.

Throughout this section, we write X for a single table cell, to which $N(X)$ individuals contributed. We let g_i be the contribution of individual i, $i = 1, \ldots, N(X)$ and we assume that the contributions are non-negative and ordered decreasingly, i.e. $g_1 \geq g_2 \geq \ldots \geq g_{N(X)} \geq 0$. This ordering is called the contribution sequence of the table cell. The corresponding cell value that we are willing to publish is denoted by $G = \sum_{i=1}^{N(X)} g_i$.

2.1.1 Sensitivity Rules

Before a table is adapted to make it safe for publication, the cells that might disclose information on a too detailed level have to be indicated. The minimum frequency rule, (n, k) dominance rule, $p\%$ rule and (p, q) rule check whether or not a table cell is safe to publish (Hundepool et al., 2012).

Minimum frequency rule

A very naive approach would be to classify a cell as unsafe whenever its value consists of less than f contributions, for some number f, which means that we just require $N(X) \geq f$. Whenever this rule is implemented, it is often used in combination with another rule.

(n, k) dominance rule

The (n, k) dominance rule, sometimes called n respondents, k percent rule, states that the sum of any combination of n contributions should not exceed k percent of the total cell value. It is easily checked that this is equivalent to stating that the sum of the n largest contributions should not exceed k percent of the cell value. In other words, the cell is safe if

$$\sum_{i=1}^{n} g_i \leq \frac{k}{100} G. \tag{2.1}$$

$p\%$ rule

The $p\%$ rule is based upon the scenario in which an attacker who contributes to a certain table cell tries to get information about another contributor to the cell. He can compute an upper bound for any other contribution, by subtracting his own value from the cell total. Mathematically, the $p\%$ rule states that an upper bound computed in this way, is not allowed to exceed the actual contribution with less than $p \geq 0$ percent for any combination of attacker and target. That is, the relative error of the estimate should be larger than $p\%$:

$$G - g_i \geq \left(1 + \frac{p}{100}\right) g_j \quad \text{for } i \neq j.$$

Obviously, a necessary condition for this equation to hold for any $i \neq j$, is that it holds for $i = 2$, $j = 1$, i.e. the second largest contributor cannot estimate the value of the largest contributor within p percent of the true value. We will now show that this condition is also a sufficient condition. In case we assume $j \neq i \neq 1$, we have

$$G - g_i \geq G - g_2 \geq \left(1 + \frac{p}{100}\right) g_1 \geq \left(1 + \frac{p}{100}\right) g_j,$$

while in case $j \neq i = 1$, we obtain

$$G - g_i = g_2 + G - g_2 - g_1 \geq g_2 + \left(1 + \frac{p}{100}\right) g_1 - g_1 = g_2 + \frac{p}{100} g_1 \geq \left(1 + \frac{p}{100}\right) g_j$$

as well, so that we can formulate an equivalent $p\%$ rule that is easier to check:

$$G - g_2 \geq \left(1 + \frac{p}{100}\right) g_1. \tag{2.2}$$

Most sources only give (2.2) as the $p\%$ rule, without mentioning or showing the derivation above. If (2.2) is not satisfied, the cell is considered unsafe. If $N(X) = 1$ or $N(X) = 2$, where we define $g_2 = 0$ in the former case, we have $G - g_2 = g_1$, so the $p\%$ rule is violated for any value of $p > 0$, which means that this rule implies a minimum frequency rule with $f = 3$.

We also note that an attacker that does not contribute to the table cell can only use G as an estimate for any contribution. Since $G \geq G - g_i$ for $i = 1, \ldots, N(X)$, this situation is captured by the $p\%$ rule as well.

(p, q) rule

The (p, q) or prior-posterior rule is an extension of the $p\%$ rule. It is again based upon the scenario in which a contributor wants to estimate an upper bound for a single other contribution, but now he already knows lower bounds for the other $N(X) - 2$ contributions, guaranteed to have relative errors of at most $q > 0$ percent each. The strategy of the attacker is to subtract his own contribution and the lower bounds from the total cell value. The (p, q) rule then states that this estimate is not allowed to be within $p < q$, $p \geq 0$ percent of the real value, i.e. we require

$$G - g_i - \sum_{k \neq i,j} \left(1 - \frac{q}{100}\right) g_k \geq \left(1 + \frac{p}{100}\right) g_j \quad \text{for } i \neq j,$$

which is equivalent to each of

$$\begin{aligned} G - g_i - g_j - \sum_{k \neq i,j} \left(1 - \frac{q}{100}\right) g_k &\geq \frac{p}{100} g_j, \\ \sum_{k \neq i,j} \frac{q}{100} g_k &\geq \frac{p}{100} g_j, \\ \sum_{k \neq i,j} g_k &\geq \frac{p}{q} g_j, \\ \text{and} \quad G - g_i &\geq \left(1 + \frac{p}{q}\right) g_j, \end{aligned}$$

from which we can follow a similar derivation as at the $p\%$ rule to arrive at the equivalent (p, q) rule

$$G - g_2 \geq \left(1 + \frac{p}{q}\right) g_1. \tag{2.3}$$

We notice that only the fraction p/q defines this rule and that a $(p, 100)$ rule is equivalent to the $p\%$ rule.

2.1.2 ■ Upper Linear Sensitivity Measures

Sensitivity rules are formalised by Cox (1981), who introduced the concept of upper linear sensitivity measures. An upper linear sensitivity measure $S(X)$ of a cell X is a linear combination of the cell contributions. Let us first define $g_i = 0$ whenever $i > N(X)$ for notational convenience and again order the individual contributions to a table cell X as $g_1 \geq g_2 \geq \ldots \geq g_{N(X)} \geq 0$. Then an upper linear sensitivity measure $S(X)$ is defined as $S(X) = \sum_{i=1}^{\infty} w_i g_i$, where the sequence of constants $\{w_i\}_{i=1}^{\infty}$ is called the sequence of weights of $S(X)$. A cell X is called sensitive whenever $S(X) > 0$.

Important upper linear sensitivity measures

The minimum frequency rule defined before is no upper linear sensitivity measure, but the other rules that were introduced are, as can be seen when we write them in the appropriate forms:

$$\begin{aligned} (n, k) \text{ dominance rule:} \quad S(X) &= \sum_{i=1}^{n} \left(1 - \frac{k}{100}\right) g_i + \sum_{i=n+1}^{\infty} -\frac{k}{100} g_i \\ p\% \text{ rule:} \quad S(X) &= \frac{p}{100} g_1 + \sum_{i=3}^{\infty} -g_i \\ (p, q) \text{ rule:} \quad S(X) &= \frac{p}{q} g_1 + \sum_{i=3}^{\infty} -g_i \end{aligned}$$

Cell unions

A respondent that contributes to two cells X_1 and X_2 remains a single respondent in the cell union $X_1 \cup X_2$, with contribution equal to the sum of its contributions to X_1 and X_2. We will give two examples to illustrate the concept of cell unions.

For companies in a single region A, let the value G_1 of cell X_1 be the total electricity consumption and the value G_2 of cell be X_2 equal to the gas consumption. Here, the value $G_1 + G_2$ of the cell union $X_1 \cup X_2$ is the total energy consumption of the companies in region A. The cell has $N(X_1) = N(X_2)$ contributors. Each company makes a contribution to $X_1 \cup X_2$ equal to the sum of its contributions to X_1 and X_2.

As a second example, let the values of cells X_1 and X_2 equal the total energy consumption of companies in region A_1 and A_2, respectively, with $A_1 \cap A_2$ be empty. Then the value $X_1 + X_2$ of the cell union $X_1 \cup X_2$ is the total energy consumption of companies in region $A_1 \cup A_2$ and it has a total of $N(X_1) + N(X_2)$ contributions, since we assume that a company can only be present at one spot, so there is no overlap in companies of A_1 and A_2. Each company makes the same contribution to $X_1 \cup X_2$ as it did to X_1 or X_2. In this case, we could actually say that the companies in A_i have a contribution of 0 to cell X_j for $i \neq j$, so that again each company makes a contribution to $X_1 \cup X_2$ equal to the sum of its contributions to X_1 and X_2.

Subadditivity

A natural thing to do, is relating the sensitivity of a cell union to the sensitivity of the original cells. An upper linear sensitivity measure is called subadditive whenever $S(X_1 \cup X_2) \leq S(X_1) + S(X_2)$ for all possible cells X_1 and X_2 that show the same variable, but possibly for other groups of contributors. If a measure is subadditive, we are sure that the union of two non-sensitive cells is also non-sensitive. Cox (1981) proved that a measure is subadditive if and only if its sequence of weights is non-increasing. This means that all of the three upper linear sensitivity measures that we defined are subadditive.

2.1.3 ■ Table Protection

Once we know which table cells are unsafe for publication, based on a particular sensitivity rule, we should modify the table. According to Hundepool and De Wolf (2012), Statistics Netherlands uses three different methods for this, which we will briefly discuss here. Each of them has its own way in which information loss occurs.

Table restructuring

In general, cells with very few contributors or cells with one or two large contributions are sensitive. A straightforward solution would be to merge rows or columns in which the sensitive cells appear, in such a way that there are no sensitive cells left. Of course, it is also possible to restructure the table and additionally use one of the other methods.

Cell suppression

Another frequently used method is to suppress certain cells of the table, which means that the value is simply replaced by a cross. Since the row and column totals are usually provided in the tables, it is not sufficient to suppress only the sensitive cells, but we should also not publish some other cells. Whenever a lower bound for the contributions is known, it will always be possible to find an interval for the suppressed cell values using the marginals. This means that it is needed to specify the acceptable size of these intervals and to carefully decide which secondary cells to suppress.

Additive rounding

Rounding cell values in a table makes sure that the values are only known within a certain interval. In additive rounding, the table is rounded such that its marginals remain the sum of the corresponding cells and the total absolute deviation of the cell values with respect to the original table is minimised. This might mean that the cell values are not always rounded to the nearest multiple of the chosen rounding base.

2.2 Recent Research

Now that the basic theory of tabular disclosure control is covered, we can move on to recent research that combines disclosure control with publishing data on maps. The connection between disclosure control in tables and visualisation techniques on maps is investigated in O'Keefe (2012); Suñé et al. (2017), for example. While in this current report we will focus on recalculating collected measurement values, like the energy consumption of companies, we also include literature on recovering point locations in a density map.

2.2.1 Recovering Point Locations

Research involving the confidentiality of locations when publishing smoothed density maps was executed recently by Wang et al. (2019), for example. Using the theory of Fourier transforms, the authors demonstrated that a kernel density map can be transformed to the original map containing discrete crime locations, but concluded that it is not possible to do so whenever the used parameters are unknown. They state that the error between the recovered map and original map did not indicate a significant pattern when changing the parameters and thus conclude that the process of kernel smoothing can very well protect locational privacy whenever the used bandwidth is not published.

Instead of looking at Fourier transforms, Lee et al. (2019) tried to recover point locations from a kernel smoothed map of disease cases by locating points on the centre lines of contour polygons that were generated from the kernel smoothed surface. For a fine resolution and small bandwidth, their method recovered individual disease cases with a mean error distance that is smaller than a usual parcel size, indicating that patient locations can be recovered with reasonable accuracy.

2.2.2 Binary Variables

De Jonge and De Wolf (2016) constructed a cartographic map that showed a continuous spatial density of the relative frequency of a binary variable, such as unemployment per capita. For most binary variables, only one of the two values is sensitive. For example, it is probably considered undesirable to disclose that a particular person is unemployed, while knowing that a person is employed is no sensitive information.

First, a density of the unemployed population and a density of the total population were made, both using Gaussian kernels. The estimated relative frequency is given by the quotient of the two densities. This estimate is discretised in five levels that correspond to different colours on the map, as part of the disclosure control. Furthermore, this allows for two procedures in case the map might still disclose sensitive information:

- Locations with too few nearby neighbours are sensitive because they might disclose an identifiable group of individuals. These locations are assigned to the bottom colour level.
- Locations where the estimated frequency is larger than a certain maximal allowed frequency, are assigned to the top level. Note that this is only a disclosure protection if the top level also contains locations for which the maximal allowed frequency is not exceeded, since otherwise we would highlight the sensitive locations, which is the opposite of what we want.

At the end of the exploratory paper, the authors mentioned several issues for further research, amongst which:

- Automatic bandwidth selection for spatial data might be an interesting path to investigate. For the moment, choosing the right bandwidth that properly reveals a pattern remains a human task.
- One can also think about an automatic bandwidth that is adaptive to the exact location and

takes, for example the sensitivity of neighbouring locations or local population density into account.

- Additional research is needed to find a more general approach to assess the disclosure risk for spatial plots.
- Spatial estimation smears out points to neighbouring locations, which may introduce density at locations which in reality have no density, like rivers and woods. Is it possible to use boundary kernels to tackle this?
- A special case of the previous remark is that a location that is not sensitive for a small bandwidth might become sensitive for a large bandwidth, because it has many sensitive neighbours.

De Wolf and De Jonge (2017) continued on the same ideas, but provided a stronger mathematical foundation of the disclosure risk that they used before. Also, some utility loss measures were defined, to be able to quantify the decrease of utility of the map after application of statistical disclosure control methods. The different measures are based upon the change in size, location and shape of the hot spots and cold spots that were present in a reference map that used a postulated optimal bandwidth.

2.2.3 Continuous Variables

The starting point for the current research is De Wolf and De Jonge (2018), in which plotting a sensitive continuous variable on a cartographic map using smoothed versions of cell counts and totals is discussed.

The authors considered a continuous variable that is considered sensitive regardless of its value. Unlike in their previous articles, they defined a disclosure risk for areas. They constructed a $p\%$ rule that used the smoothed cell total and smoothed versions of the largest two contributions per cell.

For the disclosure risk measure, the $p\%$ rule (2.2) was used, where the total value G was replaced by the integral of the estimated density over the area, since a continuous estimate was constructed using kernel smoothing. Also, smoothed versions of the largest two contributions per cell were used. Unfortunately, some problems arise with this measure:

- The integrated density is probably not equal to the total of the contributions in the area and it might even be smaller than the largest contribution.
- To overcome this, one might want to use estimates of the individual energy contributions, instead of the actual largest and second largest contributions. However, this makes that in some situations the amount of unsafe grid cells increases with increasing bandwidth, which feels counter-intuitive.

2.3 Kernel Smoothing and Notation

In this section, the concept of kernel smoothing is introduced, which plays an important role in data visualisation.

2.3.1 Notation

First, let us 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 Chapter 5.

2.3.2 ■ Kernel Smoothing

Kernel smoothing is a common way to obtain a smooth visualisation of measurements taken at discrete points (Wand and Jones, 1994). The approach is similar to kernel density estimation (Silverman, 1986). Kernel smoothing overcomes the disadvantages of the straightforward visualisation technique mentioned in Section 1.2, which is why more and more publications make use of it to visualise data originating from many different sources, including road networks (Borruso, 2003), crime numbers (Chainey et al., 2002), seismic damage figures (Danese et al., 2008) and disease cases (Davies and Hazelton, 2010). Other examples and techniques are given in Bowman and Azzalini (1997), for instance.

Essentially, in kernel density estimation, single locations are smeared out to create density bumps around each data point. The density bumps are added and normalised to obtain a total density. In the process of kernel smoothing, no normalisation is applied. In our case, the kernel smoothed population density is given by

$$f_h(\boldsymbol{r}) = \frac{1}{h^2}\sum_{i=1}^{N} k\left(\frac{\boldsymbol{r}-\boldsymbol{r}_i}{h}\right), \quad \boldsymbol{r}\in\mathcal{D}, \tag{2.4}$$

in which $k\colon \mathbb{R}^2 \to \mathbb{R}$ is a so-called kernel function, that is, a non-negative function for which $k(-\boldsymbol{r}) = k(\boldsymbol{r})$ and that integrates over $\mathbb{R}^2$ to 1. 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 Section 4.5 of Wand and Jones (1994).

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

$$\boldsymbol{K}_h = \left(k\left(\frac{\boldsymbol{r}_i-\boldsymbol{r}_j}{h}\right)\right)_{i,j=1}^{N} \tag{2.5}$$

and

$$\boldsymbol{C}_h = \left(\frac{k\left((\boldsymbol{r}_i-\boldsymbol{r}_j)/h\right)}{\sum_{m=1}^{N} k\left((\boldsymbol{r}_i-\boldsymbol{r}_m)/h\right)}\right)_{i,j=1}^{N}. \tag{2.6}$$

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

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

Kernel average smoothers are a regression technique, that tend to find a relation between two variables. As an example, these could be the location and the energy consumption of companies, as in our case study in Chapter 5. A continuous visualisation of the measurement values can be given by the Nadaraya-Watson kernel weighted average (Watson, 1964)

$$m_h(\boldsymbol{r}) = \frac{g_h(\boldsymbol{r})}{f_h(\boldsymbol{r})} = \frac{\sum_{i=1}^{N} g_i k\left((\boldsymbol{r}-\boldsymbol{r}_i)/h\right)}{\sum_{i=1}^{N} k\left((\boldsymbol{r}-\boldsymbol{r}_i)/h\right)}, \quad \boldsymbol{r}\in\mathcal{D}, \tag{2.7}$$

which is obtained by dividing the two densities f_h and g_h and which can be seen as the fraction of an estimate of, in our case, electricity consumption per area and the number of companies per area.

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 (Chacón and Duong, 2018). 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. By writing the Nadaraya-Watson kernel weighted average as

$$m_h(\boldsymbol{r}) = \sum_{i=1}^{N} g_i \frac{k\left(\left(\boldsymbol{r} - \boldsymbol{r}_i\right)/h\right)}{\sum_{j=1}^{N} k\left(\left(\boldsymbol{r} - \boldsymbol{r}_j\right)/h\right)}, \tag{2.8}$$

it is indeed clearly a weighted average of the measurement values.

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. We will prove this in Section 2.3.3.

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 Diggle (1985), Berman and Diggle (1989) and Van Lieshout (2012) for examples. In this report, we will not make further use of these techniques.

2.3.3 ■ Asymptotic Behaviour

It would be interesting to show the asymptotic behaviour of the kernel weighted average, since this gives us insights in the disclosure properties. Here, we use a Gaussian kernel $k(\boldsymbol{r}) = \frac{1}{2\pi}\exp(-\frac{1}{2}||\boldsymbol{r}||^2)$, so that the Nadaraya-Watson estimate will be

$$m_h(\boldsymbol{r}) = \frac{\sum_{i=1}^{N} g_i \ \exp(-||\boldsymbol{r} - \boldsymbol{r}_i||^2/(2h^2))}{\sum_{i=1}^{N} \exp(-||\boldsymbol{r} - \boldsymbol{r}_i||^2/(2h^2))} \tag{2.9}$$

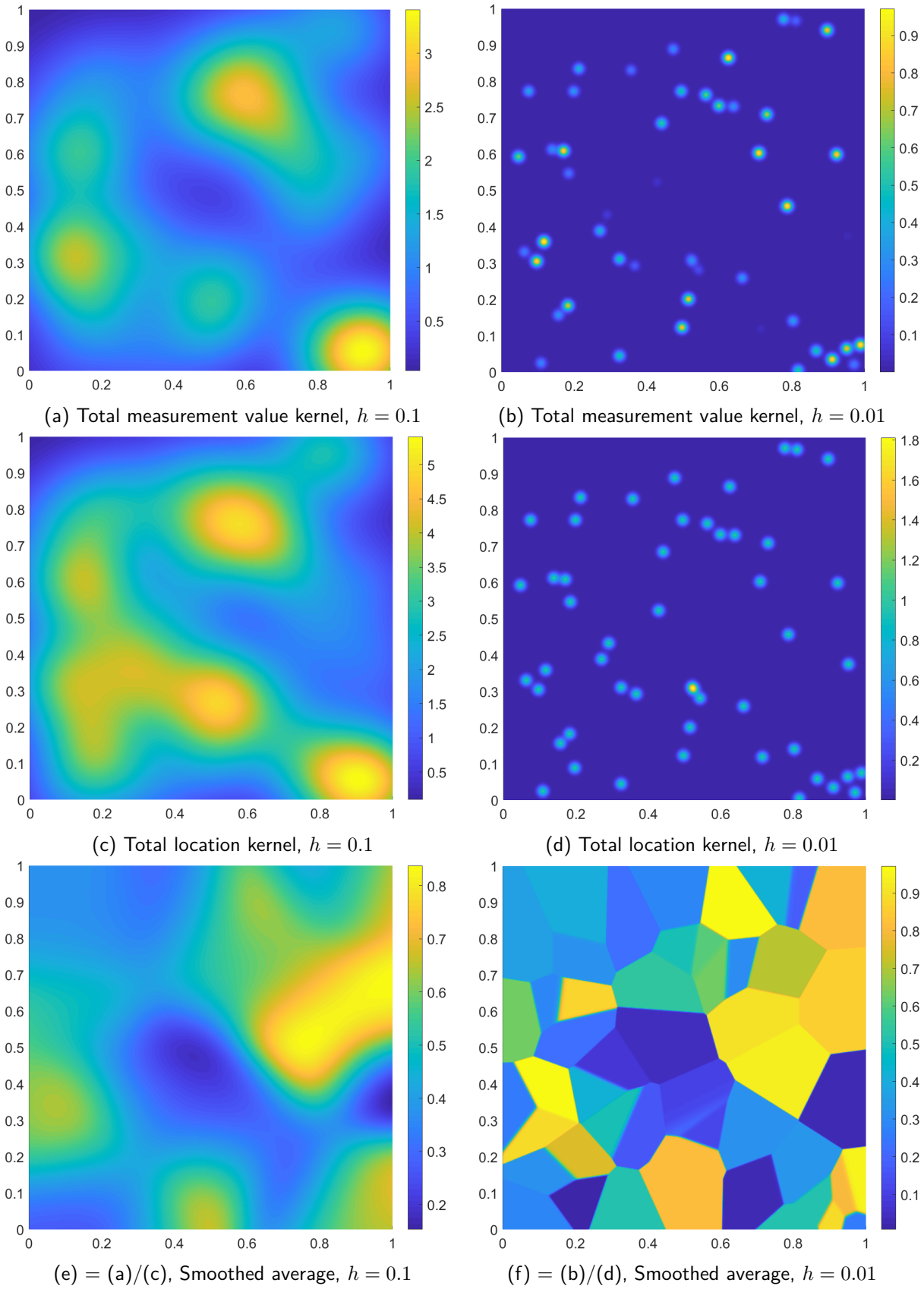
In case the bandwidth h tends to ∞, it can be easily seen that all exponents in (2.9) converge to 0, which means that $m_h(\boldsymbol{r})$ will converge to $(1/N)\sum_{i=1}^{N} g_i$ for all $\boldsymbol{r} \in \mathcal{D}$. When the measurement values fulfill a tabular sensitivity rule, publishing a plot that shows the average value is safe, but of course it does not give any more information than the average value itself.

Next we will show that the Nadaraya-Watson kernel weighted average will converge to the nearest neighbour approximation whenever the Gaussian kernel bandwidth h tends to 0. First, let $N \geq 1$ and assume that $\boldsymbol{r} \in \mathcal{D}$ has a unique nearest neighbour in $\mathcal{U}$. Define $i^*_{\boldsymbol{r}} = \arg\min_{i \in \mathcal{U}}\{||\boldsymbol{r} - \boldsymbol{r}_i||\}$ to be that neighbour. The single nearest neighbour interpolation at location $\boldsymbol{r}$ is $g_{i^*_{\boldsymbol{r}}}$. Then,

$$\begin{aligned}
\lim_{h\to 0} m_h(\boldsymbol{r}) &= \lim_{h\to 0} \frac{\sum_{i=1}^{N} g_i \ \exp(-||\boldsymbol{r} - \boldsymbol{r}_i||^2/(2h^2))}{\sum_{i=1}^{N} \exp(-||\boldsymbol{r} - \boldsymbol{r}_i||^2/(2h^2))} \\
&= \lim_{h\to 0} \frac{\sum_{i=1}^{N} g_i \ \exp\left(\left(||\boldsymbol{r} - \boldsymbol{r}_{i^*_{\boldsymbol{r}}}||^2 - ||\boldsymbol{r} - \boldsymbol{r}_i||^2\right)/(2h^2)\right)}{\sum_{i=1}^{N} \exp\left(\left(||\boldsymbol{r} - \boldsymbol{r}_{i^*_{\boldsymbol{r}}}||^2 - ||\boldsymbol{r} - \boldsymbol{r}_i||^2\right)/(2h^2)\right)} && \text{mult. by } \exp\frac{||\boldsymbol{r} - \boldsymbol{r}_{i^*_{\boldsymbol{r}}}||^2}{2h^2} \\
&= \lim_{h\to 0} \frac{g_{i^*_{\boldsymbol{r}}} + \sum_{i \neq i^*_{\boldsymbol{r}}} g_i \ \exp\left(\left(||\boldsymbol{r} - \boldsymbol{r}_{i^*_{\boldsymbol{r}}}||^2 - ||\boldsymbol{r} - \boldsymbol{r}_i||^2\right)/(2h^2)\right)}{1 + \sum_{i \neq i^*_{\boldsymbol{r}}} \exp\left(\left(\boldsymbol{r} - \boldsymbol{r}_{i^*_{\boldsymbol{r}}}||^2 - ||\boldsymbol{r} - \boldsymbol{r}_i||^2\right)/(2h^2)\right)} && \text{separate } i^*_{\boldsymbol{r}}\text{'th term} \\
&= g_{i^*_{\boldsymbol{r}}}. && \text{exponents div. to } -\infty
\end{aligned}$$

This result indicates that it is unsafe to publish a plot with a very small bandwidth, since an attacker can obtain a particular measurement value by reading off the plot at any location close to the corresponding population element location. More examples methods that generate unsafe plots are given in Chapter 3.

In Figure 2.1, that was based on a simulation, it can be seen that a small bandwidth indeed leads to a nearest neighbour interpolation. For the simulation, 50 independent measurement locations were generated uniformly on the unit square and a standard uniformly distributed measurement value was given to each of those locations. The plots indicated with 'total measurement value kernel' show the numerator of (2.7), the 'total location value kernel' plots show the denominator of (2.7) and the 'smoothed average' plots show (2.7) in its completeness. In Chapter 3, the exact same realisation of $\boldsymbol{r}_i$ and g_i, $i = 1, \ldots, N$ are used for the plots of the smoothed average, so that the reader can visually compare the differences and similarities of the plots.



(a) Total measurement value kernel, $h = 0.1$

(b) Total measurement value kernel, $h = 0.01$

(c) Total location kernel, $h = 0.1$

(d) Total location kernel, $h = 0.01$

(e) = (a)/(c), Smoothed average, $h = 0.1$

(f) = (b)/(d), Smoothed average, $h = 0.01$

Figure 2.1: Gaussian kernel, 50 points

3 Motivation

In this chapter, we show that the plot of (2.7) is often unsafe to publish, since measurement values can be recalculated, when an attacker is aware of all measurement locations $\boldsymbol{r}_i$, $i = 1, \ldots, N$. First, we discuss the use of uniform kernels (Section 3.1) and other kernels with finite support (Section 3.2). Our method in Chapter 4 to protect the plots will be based on the theory in Section 3.3, since we show there that for certain kernels, including the Gaussian kernel, the attacker can always recover the exact measurement values under certain assumptions. Afterwards, in Section 3.4 and 3.5, we say a few words about situations in which we relax some assumptions made in Section 3.3. We conclude with defining the attacker scenario for the remainder of this report in Section 3.6.

3.1 Uniform Kernels

The disk kernel is defined as a uniform density on a disk: $k(\boldsymbol{r}) = \frac{1}{\pi}\mathbb{1}(||\boldsymbol{r}|| < 1)$. In case the disk kernel is used, the Nadaraya-Watson kernel weighted average would be

$$m_h(\boldsymbol{r}) = \frac{\sum_{i=1}^{N} g_i \mathbb{1}(||\boldsymbol{r} - \boldsymbol{r}_i|| < h)}{\sum_{i=1}^{N} \mathbb{1}(||\boldsymbol{r} - \boldsymbol{r}_i|| < h)}, \quad \boldsymbol{r} \in \mathcal{D}. \tag{3.1}$$

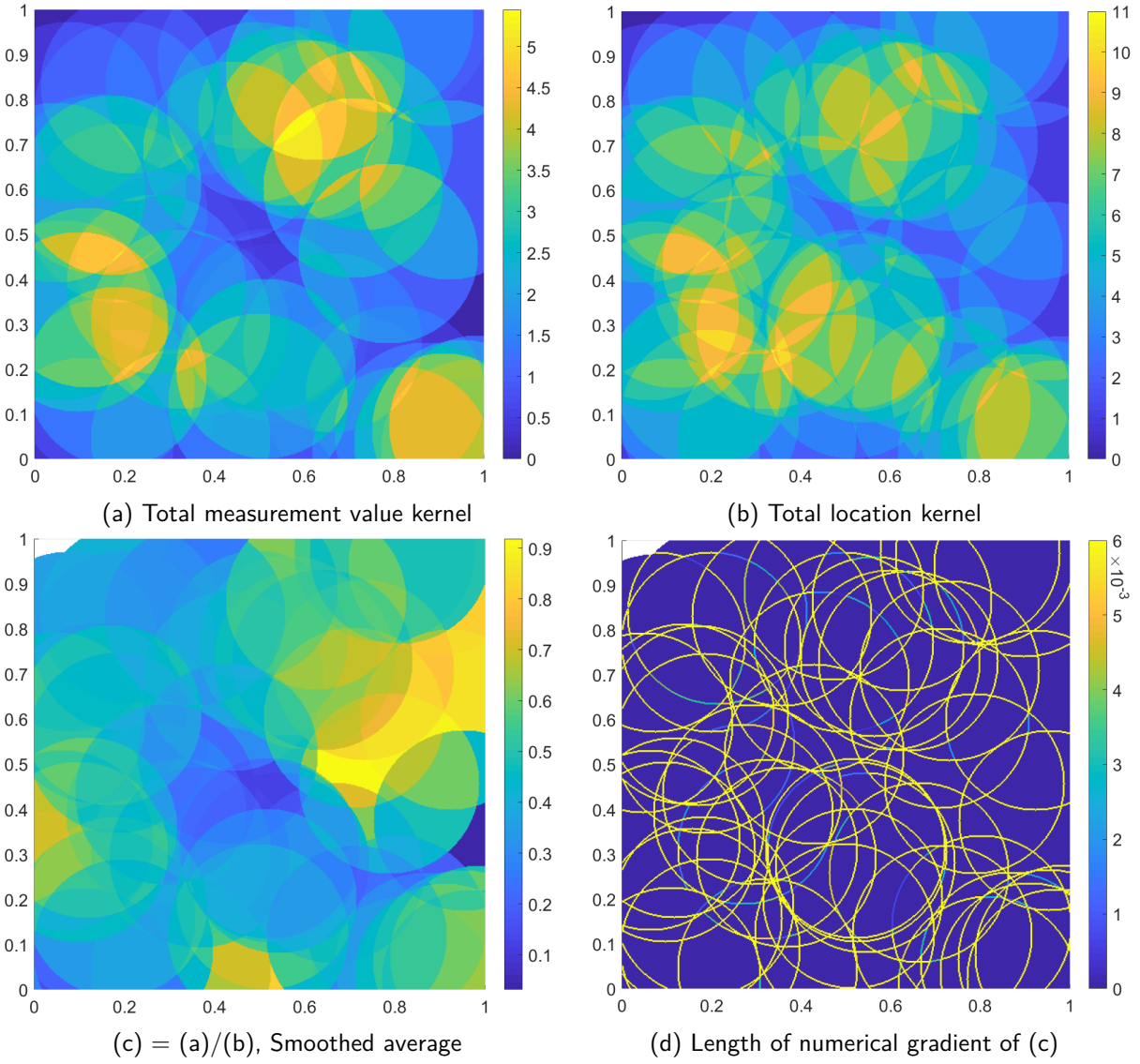
From a plot showing this quantity for each point in space, the value of a single measurement, say g_j is very easily computed, by looking at the plot value $m_h(\boldsymbol{r}_j^+)$ at a point $\boldsymbol{r}_j^+$ that lies at a distance slightly less than h from $\boldsymbol{r}_j$, so that measurement g_j is included in the computation of the plot value, and looking at the plot value $m_h(\boldsymbol{r}_j^-)$ at a point $\boldsymbol{r}_j^-$ that lies in the same direction at a distance slightly more than h from $\boldsymbol{r}_j$, so that the measurement g_j is not included in the computation of the plot value. It is important that only the measurement value on location j accounts for the difference between $m_h(\boldsymbol{r}_j^-)$ and $m_h(\boldsymbol{r}_j^+)$, i.e. we require $||\boldsymbol{r}_j - \boldsymbol{r}_j^+|| \geq h$, $||\boldsymbol{r}_j - \boldsymbol{r}_j^-|| < h$ and $\mathbb{1}(||\boldsymbol{r}_i - \boldsymbol{r}_j^+|| < h) = \mathbb{1}(||\boldsymbol{r}_i - \boldsymbol{r}_j^-|| < h)$ for all $i \neq j$. Provided that the circle around $\boldsymbol{r}$ is small enough to have a part with strict positive length contained into $\mathcal{D}$, it is always possible to find two such points by looking close enough to the boundary, since the company locations and thus all disks are unique and there are only finitely many of them. Note that the two points are not uniquely defined, but their relationship is important.

The attacker should also be able to find the amount of measurements that contribute to the two values, since he can just count the amount of company locations that are within a range of h from the points. Note that the value of h can be easily retrieved by the attacker, since the disk kernels will be very well visible in the plot and it will be easy to measure the distance from the center of a disk to the boundary. Let $n(\boldsymbol{r}) = \sum_{i=1}^{N} \mathbb{1}(||\boldsymbol{r} - \boldsymbol{r}_i|| < h)$ be the amount of measurements contributing to the plot at location $\boldsymbol{r}$. Since this is the denominator in 3.1, the attacker can compute g_j in the following way:

$$
\begin{aligned}
g_j &= \sum_{i=1}^{N} g_i \mathbb{1}(||\boldsymbol{r}_j^+ - \boldsymbol{r}_i|| < h) - \sum_{i=1}^{N} g_i \mathbb{1}(||\boldsymbol{r}_j^- - \boldsymbol{r}_i|| < h) \\
&= \sum_{i=1}^{N} \mathbb{1}(||\boldsymbol{r}_j^+ - \boldsymbol{r}_i|| < h) \frac{\sum_{i=1}^{N} g_i \mathbb{1}(||\boldsymbol{r}_j^+ - \boldsymbol{r}_i|| < h)}{\sum_{i=1}^{N} \mathbb{1}(||\boldsymbol{r}_j^+ - \boldsymbol{r}_i|| < h)} - \\
&\quad \sum_{i=1}^{N} \mathbb{1}(||\boldsymbol{r}_j^- - \boldsymbol{r}_i|| < h) \frac{\sum_{i=1}^{N} g_i \mathbb{1}(||\boldsymbol{r}_j^- - \boldsymbol{r}_i|| < h)}{\sum_{i=1}^{N} \mathbb{1}(||\boldsymbol{r}_j^- - \boldsymbol{r}_i|| < h)} \\
&= n(\boldsymbol{r}_j^+) m_h(\boldsymbol{r}_j^+) - n(\boldsymbol{r}_j^-) m_h(\boldsymbol{r}_j^-).
\end{aligned}
$$

Using the same principles, this method will work for uniform kernels of any shape, since the only information needed is the amount of measurements contributing to the plot close to the kernel boundary.

In Figure 3.1, it is shown that the locations $\boldsymbol{r}_i$ and the bandwidth h are easily retrieved whenever a disk kernel is used. When plotting the length of the numerical gradient of the smoothed average, we see the circular structures even clearer. In the next section, we see that this is also the case for other kernels.



(a) Total measurement value kernel

(b) Total location kernel

(c) = (a)/(b), Smoothed average

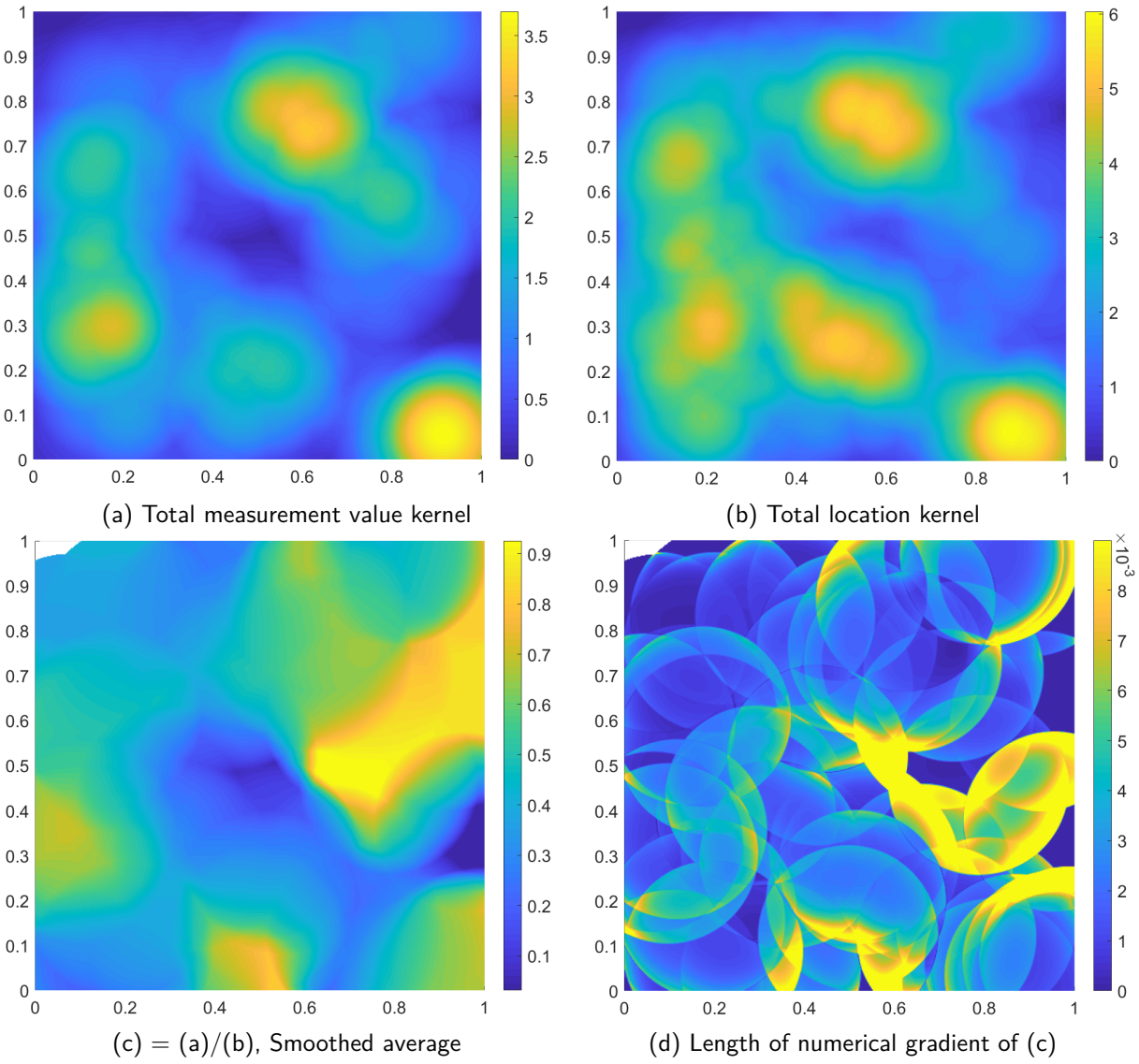
(d) Length of numerical gradient of (c)

Figure 3.1: Disk kernel, $h = 0.2$, 50 points

3.2 Kernels with Discontinuous Derivatives

Whenever kernels with a finite support are used, it might be the case that the kernels are not endlessly differentiable on the boundary. We illustrate this here for one-dimensional kernels, but the pricipal ideas remain valid in two dimensions, since most kernels used there are radially symmetric.

For instance, the Epanechnikov kernel $k(r) = (3/4)(1-r^2)\mathbb{1}(r < 1)$ has first derivative $\mathrm{d}/\mathrm{d}r\, k(r) = (-3r/2)\mathbb{1}(r < 1)$, which is discontinuous at $r = 1$ and the quartic kernel $k(r) = (15/16)(1 - r^2)^2\mathbb{1}(r < 1)$ has second derivative $\mathrm{d}^2/\mathrm{d}r^2\, k(r) = (15/4)(3r^2 - -1)\mathbb{1}(r < 1)$, which is also discontinuous at the boundary $r = 1$. This will cause discontinuities in the gradient of the plot of the smoothed average, or in a higher-order directional derivative, from which the attacker can deduct the radius of the kernel that was used. In Figure 3.2, the bandwidth is well visible after computing the gradient of the smoothed average of a plot that used the two-dimensional Epanechnikoc kernel, while it is not so apparent in the smoothed average itself. Besides only showing the used bandwidth, it might be possible to recalculate measurement values in a similar manner as in Section 3.1, but using a gradient plot of the smoothed average instead of the smoothed average itself. However, we did not look into this in more detail.



(a) Total measurement value kernel

(b) Total location kernel

(c) = (a)/(b), Smoothed average

(d) Length of numerical gradient of (c)

Figure 3.2: Epanechnikov kernel, $h = 0.2$, 50 points

3.3 System of Linear Equalities

In this section, we show that also when the used kernel is continuous and has continuous derivatives, publishing the kernel weighted average reveals exact information on the underlying measurement values under certain assumptions. This implies that it is necessary to apply disclosure control before publishing the plot. Our method to do so will be elaborated on in Chapter 4.

Here, we restrict our attention to the scenario in which an attacker is able to exactly read off the plot of the kernel weighted average (2.7) at the distinct population element locations $\boldsymbol{r}_i$, $i = 1, \ldots, N$ and is aware of the kernel and bandwidth that 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 (2.7) is a linear combination of the measurement values. When the attacker reads off the plot (2.7) at the exact locations $\boldsymbol{r}_i$, $i = 1, \ldots, N$, he obtains the system

$$m_h(\boldsymbol{r}_i) = \frac{\sum_{j=1}^{N} g_j k\left((\boldsymbol{r}_i - \boldsymbol{r}_j)/h\right)}{\sum_{j=1}^{N} k\left((\boldsymbol{r}_i - \boldsymbol{r}_j)/h\right)}, \quad i = 1, \ldots, N, \tag{3.2}$$

or, in matrix notation,

$$\boldsymbol{m}_h = \boldsymbol{C}_h \boldsymbol{g}, \tag{3.3}$$

with the known plot values $\boldsymbol{m}_h = (m_h(\boldsymbol{r}_i))_{i=1}^N$, unknown measurement value vector $\boldsymbol{g} = (\boldsymbol{g}_i)_{i=1}^N$ and known coefficient matrix $\boldsymbol{C}_h$. Recall the definition of $\boldsymbol{C}_h$ in (2.6) and $\boldsymbol{K}_h$ in (2.5). We know the following about solvability of the system.

Theorem 3.3.1. Whenever $\boldsymbol{K}_h$ is invertible, system (3.3) 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 (3.3) 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 3.3.1 shows that there is at least one configuration of points at which the attacker can read off the plot of (2.7) to retrieve the measurement values g_i, $i = 1, \ldots, N$ exactly. In Section 3.5, we will briefly comment on situations in which the attacker reads of the plot at other points.

Sketches of systems for small N indicate that the system of linear equations is always solvable if a Gaussian kernel is used. Indeed, the Gaussian kernel gives rise to invertible $\boldsymbol{K}_h$, as stated in the following theorem.

Theorem 3.3.2. For the Gaussian kernel, $\boldsymbol{K}_h$ is positive definite and thus invertible for any $h > 0$, $N \in \mathbb{N}$ and configuration of distinct points $\boldsymbol{r}_i$, $i = 1, \ldots, N$.

Proof. Denote by $f : \mathbb{R}^2 \to \mathbb{R}$ the spectral density of the Gaussian kernel, that is, f is the function such that

$$k\left(\frac{\boldsymbol{r}}{h}\right) = \frac{1}{2\pi} \mathrm{e}^{-\frac{||\boldsymbol{r}||^2}{2h^2}} = \int_{\mathbb{R}^2} \mathrm{e}^{i\boldsymbol{w}\cdot\boldsymbol{r}} f(\boldsymbol{w}) \, \mathrm{d}w.$$

According to Van Lieshout (2019), page 19, we have

$$f(\boldsymbol{w}) = \frac{h^2}{4\pi^2} \mathrm{e}^{-\frac{||\boldsymbol{w}||^2 h^2}{2}}.$$

We will show that $\boldsymbol{K}_h$ is positive definite, so that invertibility follows from there. Consider

$$\begin{aligned}
\boldsymbol{a}^T \boldsymbol{K} \boldsymbol{a} &= \sum_{j=1}^{N} \sum_{m=1}^{N} a_j k\left(\frac{\boldsymbol{r}_j - \boldsymbol{r}_m}{h}\right) a_m \\
&= \int_{\mathbb{R}^2} \sum_{j=1}^{N} \sum_{m=1}^{N} a_j a_m \mathrm{e}^{i\boldsymbol{w}\cdot(\boldsymbol{r}_j - \boldsymbol{r}_m)} f(\boldsymbol{w})\, \mathrm{d}\boldsymbol{w} \\
&= \int_{\mathbb{R}^2} \left(\sum_{j=1}^{N} a_j \mathrm{e}^{i\boldsymbol{w}\cdot\boldsymbol{r}_j}\right)\left(\sum_{m=1}^{N} a_m \mathrm{e}^{-i\boldsymbol{w}\cdot\boldsymbol{r}_m}\right) f(\boldsymbol{w})\, \mathrm{d}\boldsymbol{w} \\
&= \int_{\mathbb{R}^2} \left|\sum_{j=1}^{N} a_j \mathrm{e}^{i\boldsymbol{w}\cdot\boldsymbol{r}_j}\right|^2 f(\boldsymbol{w})\, \mathrm{d}\boldsymbol{w}.
\end{aligned}$$

For this integral to be zero, the integrand needs to be zero almost everywhere. Since $f(\boldsymbol{w})$ is strictly positive for all $\boldsymbol{w}$, it follows that the factor $\left|\sum_{i=1}^{N} a_i \exp i\boldsymbol{w}\cdot\boldsymbol{r}_i\right|$ needs to be zero for almost all $\boldsymbol{w}$. By the fact that all points $\boldsymbol{r}_i$ are distinct, this happens only when $a_i = 0$ for all i. To conclude, we have $\boldsymbol{a}^T \boldsymbol{K}_h \boldsymbol{a} > 0$ for non-zero $\boldsymbol{a}$, which means that $\boldsymbol{K}_h$ is positive definite, so $\boldsymbol{K}_h$ is invertible and thus (3.3) is uniquely solvable for the Gaussian kernel. □

3.4 Unknown Bandwidth

Whenever the bandwidth is unknown, the system of equations is not linear anymore. The attacker can overcome this by guessing different bandwidths $\hat{h}$ for his calculations and eventually choosing the bandwidth that seems to work best. Two specific error measures seem natural in this situation: one based on the difference between the actual measurements and the recalculated ones, and one based on the difference between the actual plot and a recalculated plot.

Let $[\hat{g}_1 \cdots \hat{g}_N]^T = \boldsymbol{C}_{\hat{h}} m_h$ be the solution of (3.3) that the attacker obtains when using the possibly incorrect bandwidth $\hat{h}$. The attacker can make a new plot

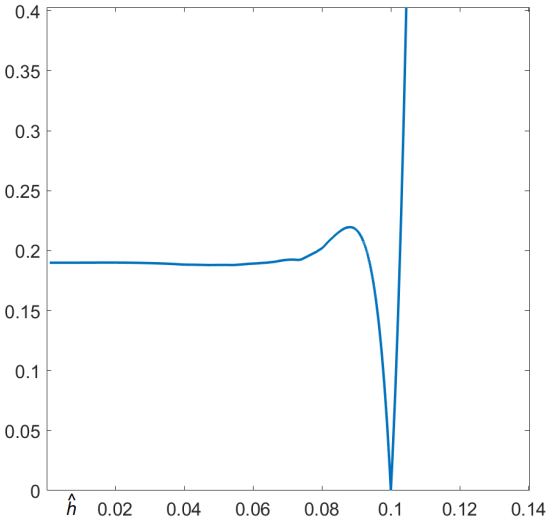
$$\hat{m}_{\hat{h}}(\boldsymbol{r}) = \frac{\sum_{i=1}^{N} \hat{g}_i k\left(\left(\boldsymbol{r} - \boldsymbol{r}_i / \hat{h}\right)\right.}{\sum_{i=1}^{N} k\left(\left(\boldsymbol{r} - \boldsymbol{r}_i\right) / \hat{h}\right)}$$

and compare it to the plot that was originally published. In this way, the attacker will be able to say something about the correctness of his estimates, since this error measure depends only on the actual measurement values through the published plot values.

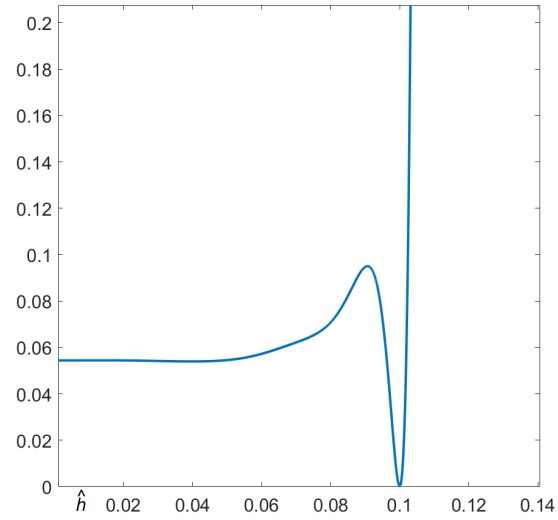
Figure 3.3 contains multiple error plots as a function of $\hat{h}$. Errors in measurement values refer to the differences between g_i and $\hat{g}_i$, for $i = 1, \ldots, N$, whereas errors in plot values refer to the differences between $m_h(\boldsymbol{r})$ and $\hat{m}_{\hat{h}}(\boldsymbol{r})$, for all locations $\boldsymbol{r}$ on a rectangular grid on which the computations are carried out.

The monotonicity of the Figures 3.3c and 3.3d indicate that an attacker will be well able to retrieve the bandwidth that was used to make the plot, after which he will know the measurement values as well.

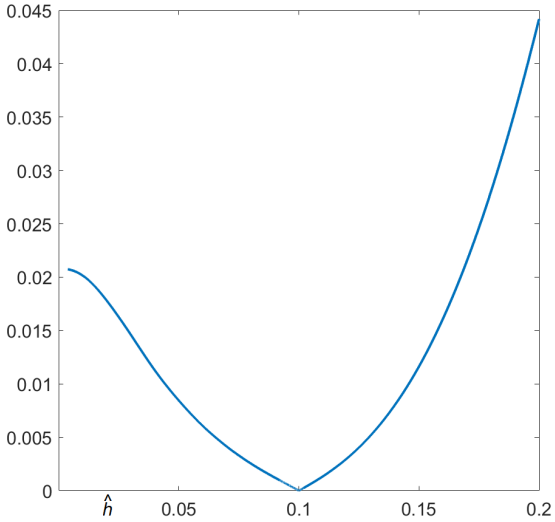
Secondly, we notice that whenever the bandwidth guess $\hat{h}$ is only slightly away from h, the errors in the retrieved measurement values are quite large.



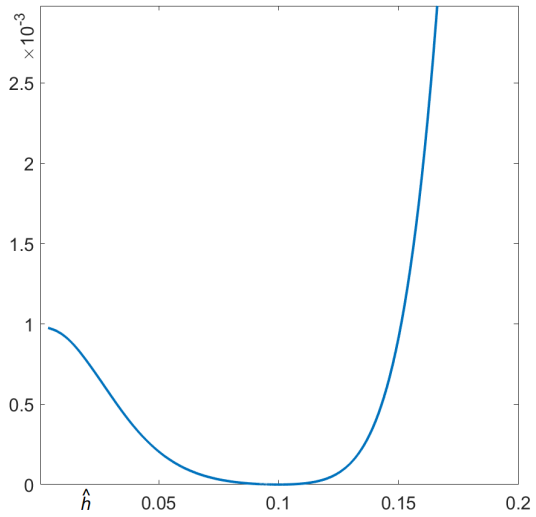
(a) Mean absolute error in measurement values



(b) Mean squared error in measurement values



(c) Mean absolute error in plot values



(d) Mean squared error in plot values

Figure 3.3: Errors as function of bandwidth guess $\hat{h}$, Gaussian kernel, $h = 0.1$, $n = N = 100$ points

Another possible approach would be the usage of a solver for non-linear system of equations. With the command `fsolve` of Matlab, it was seen that this system is well solvable for $N = 100$, when a Gaussian kernel with $h = 0.1$ was used. Note that the number of unknown variables is 101 in this case. To set up the system of equations, we chose to read off the plot at 105 locations, which means that there is a slight oversampling: The attacker uses more equations than there are variables. This improves the quality of the solution, probably by reducing numerical errors. In the simulation, the first 100 recalculation points were taken to be the company locations and the last 5 points were chosen uniformly at random.

In the same situation, but with a bandwidth $h = 0.2$, the results worsened. There were approximately 10 measurement values that were recalculated with errors greater than 2 percent. These were all cases in which three or more company locations lied very close to each other. We think that this is due to the fact that numerical errors play a greater role whenever observed plot values are close to each other, which happens due to the large bandwidth.

3.5 A More General System

In Section 3.3, we saw that for some kernel types, including the Gaussian kernel, the attacker can read off the plot of (2.7) at the population element locations to exactly retrieve the measurement values. Of course, more configurations exist for points on which the attacker can read off the plot. We will briefly discuss a more general system of linear equations.

If the attacker uses the plot value at n points $\boldsymbol{s}_1, \ldots, \boldsymbol{s}_n \in \mathbb{R}^2$, he obtains the system of linear equations

$$\begin{bmatrix} \frac{k((\boldsymbol{s}_1-\boldsymbol{r}_1)/h)}{\sum_{i=1}^N k((\boldsymbol{s}_1-\boldsymbol{r}_i)/h)} & \cdots & \frac{k((\boldsymbol{s}_1-\boldsymbol{r}_N)/h)}{\sum_{i=1}^N k((\boldsymbol{s}_1-\boldsymbol{r}_i)/h)} \\ \vdots & \ddots & \vdots \\ \frac{k((\boldsymbol{s}_n-\boldsymbol{r}_1)/h)}{\sum_{i=1}^N k((\boldsymbol{s}_n-\boldsymbol{r}_i)/h)} & \cdots & \frac{k((\boldsymbol{s}_n-\boldsymbol{r}_N)/h)}{\sum_{i=1}^N k((\boldsymbol{s}_n-\boldsymbol{r}_i)/h)} \end{bmatrix} \begin{bmatrix} g_1 \\ \vdots \\ g_N \end{bmatrix} = \begin{bmatrix} m_h(\boldsymbol{s}_1) \\ \vdots \\ m_h(\boldsymbol{s}_n) \end{bmatrix}. \tag{3.4}$$

Whenever the attacker knows the bandwidth exactly, we are sure that this system of linear equations has a solution, by the construction of the plot values m_h. However, we do not know whether the solution is unique.

For the case $n = N = 2$ we will mention geometric interpretations in the case that a disk kernel or Gaussian kernel is used. To investigate the uniqueness of the solution, let us take a look at the square coefficient matrix in (3.4) that can be obtained whenever the attacker looks up the plot value at two locations to try to obtain both measurement values.

Looking at two plot values for a plot based on two measurement locations, the coefficient matrix will be invertible whenever

$$k\left(\frac{\boldsymbol{s}_1 - \boldsymbol{r}_1}{h}\right) k\left(\frac{\boldsymbol{s}_2 - \boldsymbol{r}_2}{h}\right) \neq k\left(\frac{\boldsymbol{s}_1 - \boldsymbol{r}_2}{h}\right) k\left(\frac{\boldsymbol{s}_2 - \boldsymbol{r}_1}{h}\right). \tag{3.5}$$

We see that this explicitly depends on the kernel function that is used. We will discuss the implications for the disk kernel and the Gaussian kernel.

Disk kernel

If we consider a uniform kernel on a disk, the kernel value can be either 0 or $1/\pi$, which we can combine with (3.5) to see that the coefficient matrix is only invertible if one of $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$ lies within a distance h from $\boldsymbol{r}_1$, the other lies within a distance h from $\boldsymbol{r}_2$ and at most one lies within a distance h from both $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$. Such a configuration is easily made in case the bandwidth is reasonably small compared to the domain.

Gaussian kernel

For the Gaussian kernel, the requirement (3.5) becomes

$$\mathrm{e}^{-(||\boldsymbol{s}_1-\boldsymbol{r}_1||^2+||\boldsymbol{s}_2-\boldsymbol{r}_2||^2)/(2h^2)} \neq \mathrm{e}^{-(||\boldsymbol{s}_1-\boldsymbol{r}_2||^2+||\boldsymbol{s}_2-\boldsymbol{r}_1||^2)/(2h^2)},$$

or, equivalently,

$$||\boldsymbol{s}_1 - \boldsymbol{r}_1||^2 + ||\boldsymbol{s}_2 - \boldsymbol{r}_2||^2 \neq ||\boldsymbol{s}_1 - \boldsymbol{r}_2||^2 + ||\boldsymbol{s}_2 - \boldsymbol{r}_1||^2.$$

Since we have

$$\begin{aligned}
&||\boldsymbol{s}_1 - \boldsymbol{r}_1||^2 + ||\boldsymbol{s}_2 - \boldsymbol{r}_2||^2 - ||\boldsymbol{s}_1 - \boldsymbol{r}_2||^2 - ||\boldsymbol{s}_2 - \boldsymbol{r}_1||^2 \\
= &||\boldsymbol{s}_1||^2 + ||\boldsymbol{r}_1||^2 - 2(\boldsymbol{s}_1 \cdot \boldsymbol{r}_1) + ||\boldsymbol{s}_2||^2 + ||\boldsymbol{r}_2||^2 - 2(\boldsymbol{s}_2 \cdot \boldsymbol{r}_2) - \\
&||\boldsymbol{s}_1||^2 - ||\boldsymbol{r}_2||^2 + 2(\boldsymbol{s}_1 \cdot \boldsymbol{r}_2) - ||\boldsymbol{s}_2||^2 - ||\boldsymbol{r}_1||^2 + 2(\boldsymbol{s}_2 \cdot \boldsymbol{r}_1) \\
= &-2(\boldsymbol{s}_1 \cdot \boldsymbol{r}_1 + \boldsymbol{s}_2 \cdot \boldsymbol{r}_2 - \boldsymbol{s}_1 \cdot \boldsymbol{r}_2 - \boldsymbol{s}_2 \cdot \boldsymbol{r}_1) \\
= &-2(\boldsymbol{s}_1 - \boldsymbol{s}_2) \cdot (\boldsymbol{r}_1 - \boldsymbol{r}_2),
\end{aligned}$$

we obtain

$$||\boldsymbol{s}_1 - \boldsymbol{r}_1||^2 + ||\boldsymbol{s}_2 - \boldsymbol{r}_2||^2 \neq ||\boldsymbol{s}_1 - \boldsymbol{r}_2||^2 + ||\boldsymbol{s}_2 - \boldsymbol{r}_1||^2 \quad \text{if and only if} \quad (\boldsymbol{s}_1 - \boldsymbol{s}_2) \cdot (\boldsymbol{r}_1 - \boldsymbol{r}_2) \neq 0.$$

This means that the system of linear equations has a unique solution whenever the attacker chooses his observations in such a way that the line through his observation points is not perpendicular to the line through the measurement locations.

Tests and simulation involving larger instances for the Gaussian kernel caused us to suspect that for all distinct choices of $\boldsymbol{s}_1, \ldots, \boldsymbol{s}_N$, except maybe on a null set in $\mathbb{R}^N$, the system is uniquely solvable. The details are not pursued in this project. We stress once more that we know that at least one configuration guarantees to lead to a uniquely solvable system for the Gaussian kernel, namely choosing $\boldsymbol{s}_i = \boldsymbol{r}_i$ for $i = 1, \ldots, N$.

3.6 Attacker Scenario

In the remainder of this report, we will assume an attacker scenario in which the attacker obtains a vector containing the exact plot values at the distinct population element 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, $i = 1, \ldots, N$. This scenario is based upon the results in Section 3.3. Of course, we have to assume that $\boldsymbol{K}_h$ is invertible in order for $\boldsymbol{C}_h^{-1}$ to exist.

4 Method

Until now, we have looked into a plot in which the true value of $m_h(\boldsymbol{r})$ was shown. The attacker could recompute the g_i's exactly. In this section, we look at the situation where random noise is added to the plot.

Our method to prevent the disclosure of sensitive information consists of disturbing the plot of (2.7) by adding random noise. We make this clear in Section 4.1. In Section 4.2 we discuss our new sensitivity rule. Then, in Sections 4.3, 4.4 and 4.5, we discuss different methods to add random noise and derive results on the magnitude that is needed to be safe according to our sensitivity rule.

4.1 Noise Propagation

We prevent the disclosure of sensitive information by disturbing the plot of (2.7) with random noise, so that we publish the plot of

$$m_h(\boldsymbol{r}) + \epsilon(\boldsymbol{r}) = \frac{\sum_{j=1}^{N} g_j k\left((\boldsymbol{r}_i - \boldsymbol{r}_j)/h\right)}{\sum_{j=1}^{N} k\left((\boldsymbol{r}_i - \boldsymbol{r}_j)/h\right)} + \epsilon(\boldsymbol{r}), \quad \boldsymbol{r} \in \mathcal{D}, \tag{4.1}$$

for noise $\epsilon\colon \mathbb{R}^2 \to \mathbb{R}$. In particular, this means that according to our attacker scenario, the attacker observes the values

$$m_h(\boldsymbol{r}_i) + \tilde{\epsilon}_i = \frac{\sum_{j=1}^{N} g_j k\left((\boldsymbol{r}_i - \boldsymbol{r}_j)/h\right)}{\sum_{j=1}^{N} k\left((\boldsymbol{r}_i - \boldsymbol{r}_j)/h\right)} + \tilde{\epsilon}_i, \quad i = 1, \ldots, N, \tag{4.2}$$

instead of (3.2), for random noise $\tilde{\epsilon}_i = \epsilon(\boldsymbol{r}_i),\ i = 1, \ldots, N$. In matrix notation,

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

instead of (3.3), with $\tilde{\boldsymbol{\epsilon}} = (\tilde{\epsilon}_i)_{i=1}^{N}$.

According to our attacker scenario of Section 3.6, the attacker will left-multiply the vector of observed values by $\boldsymbol{C}_h^{-1}$. However, he will now make an error, since he observes $\boldsymbol{m} + \tilde{\boldsymbol{\epsilon}}$ instead of $\boldsymbol{m}$. The recalculated values are

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

with its i'th element equal to

$$\hat{g}_i = g_i + \sum_{j=1}^{N} \left(\boldsymbol{C}_h^{-1}\right)_{ij} \tilde{\epsilon}_j, \tag{4.3}$$

instead of $\boldsymbol{g} = \boldsymbol{C}_h^{-1}\boldsymbol{m}$, which means that the attacker makes an error in the recalculation process. In the next sections, we will formulate our sensitivity rule and discover under what conditions this error is large enough to be safe according to that rule, for different choices of the random noise.

4.2 The $(p\%, \alpha)$ Rule

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 α 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 was elaborated on in Section 2.1.1.

Definition 4.2.1. 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

$$\max_{i=1,\ldots,N} P\left\{\left|\frac{\hat{g}_i - g_i}{g_i}\right| < \frac{p}{100}\right\} > \alpha, \tag{4.4}$$

where we take $|(\hat{g}_i - g_i)/g_i| = 0$ if $g_i = 0$. A plot that is not unsafe will be called *safe*.

When applying the $(p\%, \alpha)$ rule, we normally choose p and α to be small, so that a plot is safe when small relative errors in the recalculation happen with small probability. Theorem 3.3.1 implies that the plot of (2.7) cannot be safe for any $(p\%, \alpha)$ rule. Furthermore, we note that high values of p and low values of α 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.

In the remaining sections of this chapter, we will investigate the needed magnitude of the added noise to protect the plot sufficiently. The next lemma, in which we write Φ^{-1} for the standard normal inverse cumulative distribution function, will ease this process.

Lemma 4.2.1. Whenever $\hat{g}_i$ follows a normal distribution with mean g_i, (4.4) is equivalent with

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

Proof. Assume $\hat{g}_i$ follows a normal distribution with mean g_i. Then, we have

$$\begin{aligned}
& P\left\{\left|\frac{\hat{g}_i - g_i}{g_i}\right| < \frac{p}{100}\right\} \\
= \ & 1 - 2P\left\{\hat{g}_i - g_i < -\frac{p\ g_i}{100}\right\} \\
= \ & 1 - 2P\left\{\frac{\tilde{g}_i - g_i}{\sqrt{\mathrm{Var}(\hat{g}_i)}} < -\frac{p\ g_i}{100\ \sqrt{\mathrm{Var}(\hat{g}_i)}}\right\} \\
= \ & 1 - 2P\left\{X < -\frac{p\ g_i}{100\ \sqrt{\mathrm{Var}(\hat{g}_i)}}\right\},
\end{aligned}$$

where X is a standard normal random variable. Note that this derivation is also valid if $g_i = 0$, since we defined $|(\hat{g}_i - g_i)/g_i| = 0$ in that case.

For this whole expression to be larger than α, we require

$$P\left\{X \leq -\frac{p\ g_i}{100\ \sqrt{\mathrm{Var}(\hat{g}_i)}}\right\} < \frac{1-\alpha}{2}.$$

When we write Φ for the cumulative distribution function of a standard normal random variable, the expression above is equivalent to

$$-\frac{p\ g_i}{100\ \sqrt{\mathrm{Var}(\hat{g}_i)}} < \Phi^{-1}\left(\frac{1-\alpha}{2}\right)$$

and

$$\frac{p\ g_i}{100\ \Phi^{-1}\left((1+\alpha)/2\right)\sqrt{\mathrm{Var}(\hat{g}_i)}} > 1,$$

where it should be noted that when we divide by $\Phi^{-1}\left((1-\alpha)/2\right) = -\Phi^{-1}\left((1+\alpha)/2\right)$, which is negative for positive α, the inequality sign flips.

Finally, we take the maximum over i of this expression and obtain

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

□

4.3 Independent Noise on Total Plot

In this section, we will take the noise $\tilde{\epsilon}_i$, $i = 1,\ldots,N$ in (4.2) as independent and identically distributed Gaussian random variables with mean 0 and variance σ^2. In practise, this means that we will add this noise for all pixels of the plot. Because of our attacker scenario, however, only the noise $\tilde{\epsilon}_i$, $i = 1,\ldots,N$ at locations $\boldsymbol{r}_i$, $i = 1,\ldots,N$ plays a role in the derivations in this section. It brings us to the following theorem on safe values for the standard deviation σ.

Theorem 4.3.1. Suppose that $\boldsymbol{K}_h$ is invertible, $g_i \geq 0$, $i = 1,\ldots,N$ and $\tilde{\epsilon}_i$, $i = 1,\ldots,N$ are independent and identically distributed Gaussian random variables with mean 0 and variance σ^2. Then the plot of (4.2) is safe according to the $(p\%,\alpha)$ rule for our attacker scenario in Section 3.6 if

$$\sigma \geq \frac{p}{100\,\Phi^{-1}\left((1+\alpha)/2\right)} \max_{i=1,\ldots,N}\left\{\frac{g_i}{\sqrt{\sum_{j=1}^{N}\left(\boldsymbol{C}_h^{-1}\right)_{ij}^2}}\right\}. \tag{4.5}$$

Proof. Take $\tilde{\epsilon}_i$, $i = 1,\ldots,N$ as independent and identically distributed Gaussian random variables with mean 0 and variance σ^2. Continuing from (4.3), this implies that the i-th recalculated value $\hat{g}_i$, as a linear combination of independent Gaussian random variables, will follow a normal distribution with mean g_i and variance

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

Combining this with Lemma 4.2.1 gives us

$$\frac{p}{100\,\sigma\,\Phi^{-1}\left((1+\alpha)/2\right)} \max_{i=1,\ldots,N}\left\{\frac{g_i}{\sqrt{\sum_{j=1}^{N}\left(\boldsymbol{C}_h^{-1}\right)_{ij}^2}}\right\} > 1,$$

as a condition to be *unsafe* according to the $(p\%,\alpha)$ rule, from which it is only a small step to conclude that the plot is *safe* according to the $(p\%,\alpha)$ if

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

□

4.4 Continuous Noise on Total Plot

In the previous section, we presented a way to protect the values that the attacker obtains at the company locations, by adding independent and identically distributed noise. When publishing a smoothed average plot, one would then add the noise for all locations, i.e. for all pixels on which the plot value is calculated. However, the resulting image would look grainy by the independence of the noise on neighbouring pixels. This does not fulfill our requirement for a continuous visualisation. For that reason, we will consider a form of continuous noise in this section.

One way in which continuous noise can be created, is by means of a Gaussian random field. A sequence of random variables is a Gaussian random field whenever any subsequence follows a multivariate normal distribution. According to Van Lieshout (2019), a Gaussian random field is completely defined by its mean and covariance functions. Abrahamsen (1997) tells us that a function must be positive definite in order to be a valid covariance function. For the Gaussian kernel, positive definiteness was proven in Theorem 3.3.2.

For our application, we will choose the mean of the Gaussian random field to be zero everywhere, since we do not intend to predefine a positive or negative bias at any location. We take the covariance function of the noise $\epsilon\colon \mathbb{R} \to \mathbb{R}^2$ in (4.1) as

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

where σ influences the magnitude of the added noise. In this way, (4.2) will be continuous, just as (2.7), whenever a continuous kernel function is used and f_h vanishes nowhere. In particular, this means for the realisations $\tilde{\epsilon}_i = \epsilon(\boldsymbol{r}_i),\ i = 1, \ldots, N$ of the random noise that the attacker sees at the measurement locations according to (4.2), that they follow a multivariate normal distribution with mean zero and covariance

$$\operatorname{Cov}\left(\tilde{\epsilon}_i, \tilde{\epsilon}_j\right) = \sigma^2 \left(\boldsymbol{K}_h\right)_{ij}, \quad i, j = 1, \ldots, N,$$

The following theorem gives a safe bound for the standard deviation of the noise.

Theorem 4.4.1. Suppose that $\boldsymbol{K}_h$ is positive definite, $g_i \geq 0$, $i = 1, \ldots, N$ and $\tilde{\epsilon}_i$, $i = 1, \ldots, N$ follows a multivariate normal distribution with mean 0 and covariance matrix $\sigma^2 \boldsymbol{K}_h$. Then the plot of (4.2) is safe according to the $(p\%, \alpha)$ rule for our attacker scenario in Section 3.6 if

$$\sigma \geq \frac{p}{100\,\Phi^{-1}\left((1+\alpha)/2\right)} \max_{i=1,\ldots,N} \left\{ \frac{g_i}{\sqrt{\sum_{j=1}^{N}\sum_{k=1}^{N}\left(\boldsymbol{C}_h^{-1}\right)_{ij}\left(\boldsymbol{C}_h^{-1}\right)_{ik}\left(\boldsymbol{K}_h\right)_{jk}}} \right\} \tag{4.6}$$

Proof. Take $\boldsymbol{K}_h$ positive definite and $\tilde{\epsilon}_i$, $i = 1, \ldots, N$ as multivariate normal random variables with mean 0 and covariance matrix $\sigma^2 \boldsymbol{K}_h$. Continuing from (4.3), this implies that the i-th recalculated value $\hat{g}_i$, as a linear combination of independent Gaussian random variables, will follow a normal distribution with mean g_i and variance

$$\begin{aligned}
\operatorname{Var}(\hat{g}_i) &= \operatorname{Var}\left(\sum_{j=1}^{N}\left(\boldsymbol{C}_h^{-1}\right)_{ij}\tilde{\epsilon}_j\right) \\
&= \sum_{j=1}^{N}\sum_{k=1}^{N}\left(\boldsymbol{C}_h^{-1}\right)_{ij}\left(\boldsymbol{C}_h^{-1}\right)_{ik}\operatorname{Cov}\left(\tilde{\epsilon}_j, \tilde{\epsilon}_k\right) \\
&= \sigma^2 \sum_{j=1}^{N}\sum_{k=1}^{N}\left(\boldsymbol{C}_h^{-1}\right)_{ij}\left(\boldsymbol{C}_h^{-1}\right)_{ik}\left(\boldsymbol{K}_h\right)_{jk}.
\end{aligned}$$

Combining this with Lemma 4.2.1 gives us

$$\frac{p}{100\,\sigma\,\Phi^{-1}\left((1+\alpha)/2\right)}\max_{i=1,\ldots,N}\left\{\frac{g_i}{\sqrt{\sum_{j=1}^{N}\sum_{k=1}^{N}\left(\boldsymbol{C}_h^{-1}\right)_{ij}\left(\boldsymbol{C}_h^{-1}\right)_{ik}\left(\boldsymbol{K}_h\right)_{jk}}}\right\}>1,$$

as a condition to be *unsafe* according to the $(p\%,\alpha)$ rule, from which it is only a small step to conclude that the plot is *safe* according to the $(p\%,\alpha)$ if

$$\sigma\geq\frac{p}{100\,\Phi^{-1}\left((1+\alpha)/2\right)}\max_{i=1,\ldots,N}\left\{\frac{g_i}{\sqrt{\sum_{j=1}^{N}\sum_{k=1}^{N}\left(\boldsymbol{C}_h^{-1}\right)_{ij}\left(\boldsymbol{C}_h^{-1}\right)_{ik}\left(\boldsymbol{K}_h\right)_{jk}}}\right\}.$$

□

4.5 Continuous Noise on Numerator

Our last method to consider for the addition of random noise to the plot of (2.7), is adding it to the numerator of (2.7), so that an attacker observes

$$m_h(\boldsymbol{r}_i)+\tilde{\epsilon}_i=\frac{\sum_{j=1}^{N}g_j k\left((\boldsymbol{r}_i-\boldsymbol{r}_j)/h\right)+\epsilon_i}{\sum_{j=1}^{N}k\left((\boldsymbol{r}_i-\boldsymbol{r}_j)/h\right)},\quad i=1,\ldots,N,\tag{4.7}$$

which means that we take $\tilde{\epsilon}_i$ in (4.2) equal to

$$\tilde{\epsilon}_i=\frac{\epsilon_i}{\sum_{j=1}^{N}k\left((\boldsymbol{r}_i-\boldsymbol{r}_j)/h\right)},\quad i=1,\ldots,N.\tag{4.8}$$

Again, we choose to take the continuous noise on the numerator te be a Gaussian field. For the values that the attacker observes at the measurement locations, this means that ϵ_i, $i=1,\ldots,N$ follows a multivariate normal distribution with mean 0 and covariance

$$\operatorname{Cov}\left(\epsilon_i,\epsilon_j\right)=\sigma^2 k\left(\frac{\boldsymbol{r}_i-\boldsymbol{r}_j}{h}\right),\quad i,j=1,\ldots,N.$$

The left-hand side of 4.8 allows for an other interpretation: Instead of saying that random noise is added to the numerator density of the kernel smoother, one might also say that the distortion at a point is inversely proportional to the population density at that point.

Again, the resulting plot of (4.2) will be continuous, just as (2.7), whenever a continuous kernel function is used and f_h vanishes nowhere. The following theorem gives a safe bound for the standard deviation of the noise.

Before stating the safe lower bound for σ, we will prove a result on the inverses of $\boldsymbol{K}_h$ and $\boldsymbol{C}_h$.

Lemma 4.5.1. Suppose that $\boldsymbol{K}_h$ is invertible. Then

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

Proof. Recall that $\boldsymbol{C}_h$ is invertible whenever $\boldsymbol{K}_h$ is. We will work out the two required matrix multiplications to show that the matrix on the right hand side is indeed the inverse of $\boldsymbol{K}_h$. To begin,

$$\sum_{k=1}^{N}\frac{\left(\boldsymbol{C}_h^{-1}\right)_{ik}}{\sum_{m=1}^{N}\left(\boldsymbol{K}_h\right)_{km}}\left(\boldsymbol{K}_h\right)_{kj}=\sum_{k=1}^{N}\left(\boldsymbol{C}_h^{-1}\right)_{ik}\left(\boldsymbol{C}_h\right)_{kj}=\mathbb{1}(i=j).$$

Secondly,

$$\begin{aligned}\sum_{k=1}^{N}(\boldsymbol{K}_h)_{ik}\frac{\left(\boldsymbol{C}_h^{-1}\right)_{kj}}{\sum_{m=1}^{N}(\boldsymbol{K}_h)_{jm}} &= \frac{\sum_{m=1}^{N}(\boldsymbol{K}_h)_{im}}{\sum_{m=1}^{N}(\boldsymbol{K}_h)_{jm}}\sum_{k=1}^{N}(\boldsymbol{C}_h)_{ik}\left(\boldsymbol{C}_h^{-1}\right)_{kj}\\ &= \frac{\sum_{m=1}^{N}(\boldsymbol{K}_h)_{im}}{\sum_{m=1}^{N}(\boldsymbol{K}_h)_{jm}}\mathbb{1}(i=j)\\ &= \mathbb{1}(i=j).\end{aligned}$$

□

Using this result, we can prove the following theorem.

Theorem 4.5.1. Suppose that $\boldsymbol{K}_h$ is positive definite, $g_i \geq 0$, $i = 1, \ldots, N$ and ϵ_i, $i = 1, \ldots, N$ follow a multivariate normal distribution with mean 0 and covariance matrix $\sigma^2\boldsymbol{K}_h$. Then the plot of (4.7) is safe according to the $(p\%, \alpha)$ rule for our attacker scenario in Section 3.6 if

$$\sigma \geq \frac{p}{100\,\Phi^{-1}\left((1+\alpha)/2\right)}\max_{i=1,\ldots,N}\left\{\frac{g_i}{\sqrt{\left(\boldsymbol{K}_h^{-1}\right)_{ii}}}\right\} \tag{4.9}$$

Proof. Take $\boldsymbol{K}_h$ positive definite and $\tilde{\epsilon}_i$, $i = 1, \ldots, N$ as multivariate normal random variables with mean 0 and covariance matrix $\sigma^2\boldsymbol{K}_h$. Continuing from (4.3), this implies that the i-th recalculated value $\hat{g}_i$, as a linear combination of independent Gaussian random variables, will follow a normal distribution with mean g_i and variance

$$\begin{aligned}\mathrm{Var}(\hat{g}_i) &= \mathrm{Var}\left(\sum_{j=1}^{N}\left(\boldsymbol{C}_h^{-1}\right)_{ij}\tilde{\epsilon}_j\right)\\ &= \sum_{j=1}^{N}\sum_{k=1}^{N}\left(\boldsymbol{C}_h^{-1}\right)_{ij}\left(\boldsymbol{C}_h^{-1}\right)_{ik}\mathrm{Cov}\left(\tilde{\epsilon}_j, \tilde{\epsilon}_k\right)\\ &= \sum_{j=1}^{N}\sum_{k=1}^{N}\frac{\left(\boldsymbol{C}_h^{-1}\right)_{ij}\left(\boldsymbol{C}_h^{-1}\right)_{ik}}{\sum_{m=1}^{N}(\boldsymbol{K}_h)_{jm}\sum_{m=1}^{N}(\boldsymbol{K}_h)_{km}}\mathrm{Cov}\left(\epsilon_j, \epsilon_k\right)\\ &= \sigma^2\sum_{j=1}^{N}\sum_{k=1}^{N}\frac{\left(\boldsymbol{C}_h^{-1}\right)_{ij}\left(\boldsymbol{C}_h^{-1}\right)_{ik}}{\sum_{m=1}^{N}(\boldsymbol{K}_h)_{jm}}(\boldsymbol{C}_h)_{kj}\,.\end{aligned}$$

In the third equality above, we wrote $\tilde{\epsilon}_j$ and $\tilde{\epsilon}_k$ in terms of ϵ_j and ϵ_k, respectively, and took the factors outside the covariance. In the last step, we substituted $\sigma^2\left(\boldsymbol{K}_h\right)_{jk} = \sigma^2\left(\boldsymbol{C}_h\right)_{kj}\sum_{m=1}^{N}\left(\boldsymbol{K}_h\right)_{km}$ for $\mathrm{Cov}(\epsilon_j, \epsilon_k)$. We continue by rearranging factors and working out the matrix multiplication:

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

We use Lemma 4.5.1 to obtain

$$\mathrm{Var}(\hat{g}_i) = \sigma^2 \left(\boldsymbol{K}_h^{-1} \right)_{ii}, \tag{4.10}$$

which we can combine with Lemma 4.2.1 to get

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

as a condition to be *unsafe* according to the $(p\%, \alpha)$ rule. From here it is only a small step to conclude that the plot is *safe* according to the $(p\%, \alpha)$ if

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

□

For the limit case of a very small bandwidth h, the diagonal of $\boldsymbol{K}_h$ will be close to 1 and the off-diagonal entries will be close to 0, which means that also $\boldsymbol{K}_h^{-1}$ will be a diagonal matrix. In that case, we use (4.10) to conclude that the variance in the recalculated values equals the variance of the ϵ_i's that were introduced. This makes sense, since the for a small bandwidth, $m(\boldsymbol{r}_i) \approx g_i$, as was seen in Section 2.3.3, and thus the variance of ϵ_i will be the variance of $\tilde{g}_i$.

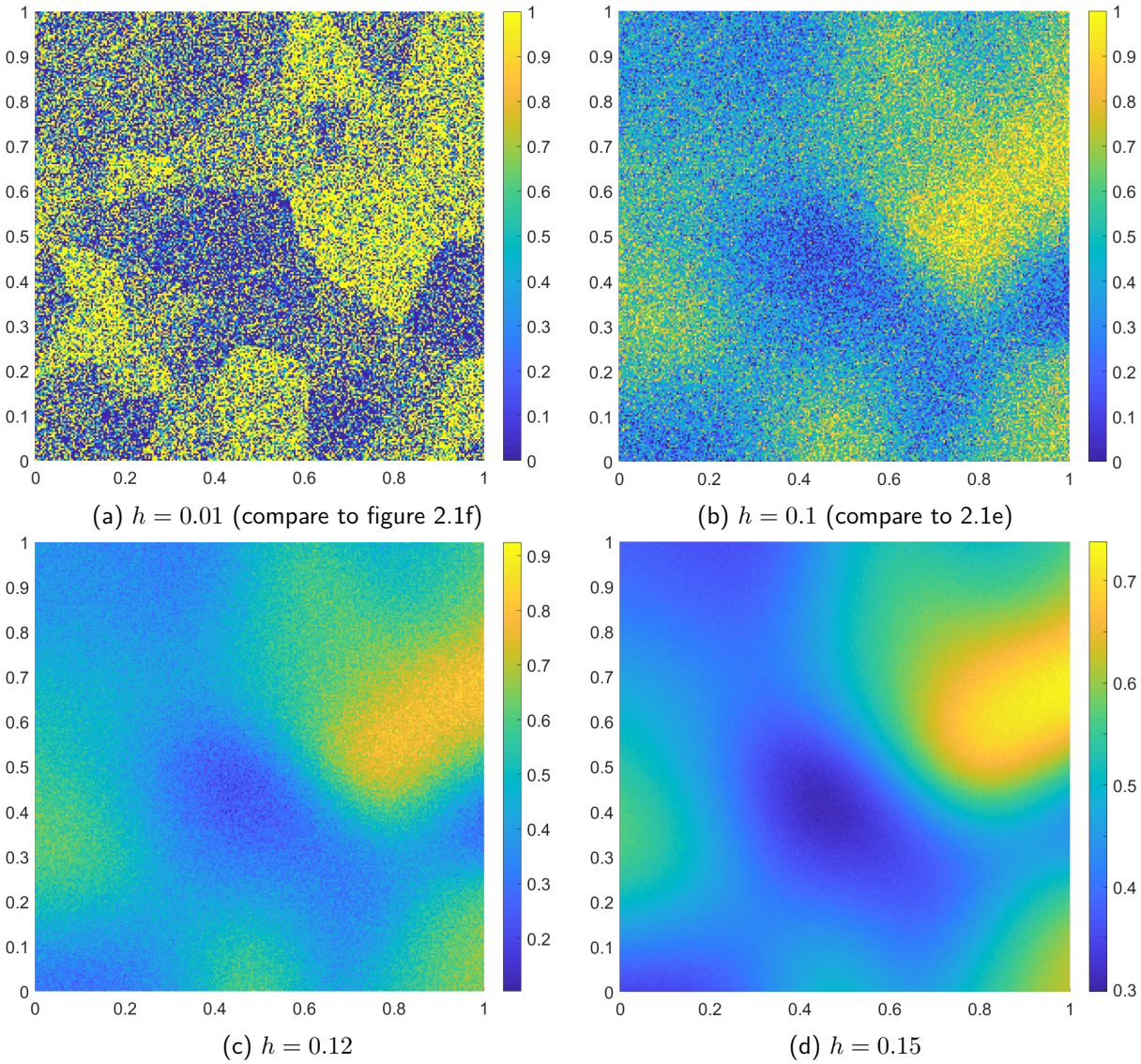
5 Simulations and Case Study

For the three methods that were elaborated on in Chapter 4, we want to be able to compare unprotected plots with protected plots. This means so we cannot use original confidential data, since it would disclose sensitive information. For that reason, we show plots of uniform random data in Section 5.1 and of a more realistic data set in Section 5.2.

5.1 Simulations

First, in Figure 5.1, independent noise per pixel is added to the smoothed average for different bandwidths, where the standard deviation σ is chosen equal to the lower bound in Theorem 4.3.1. The population element locations and measurement values were taken to be identical to the ones of the plots in Chapters 2 and 3. We clearly see that for the larger bandwidth, the required magnitude of the noise is smaller and the noise is visually less apparent. Adding noise independently per pixel makes that the plots look grainy. Also, changing the amount of pixels will visually change the plot to a large extent.

Then, for the simulations in Figures 5.2-5.4, 100 population elements were used. Their locations were chosen uniformly at random on the unit square and their measurement values uniformly at random on $[0, 1]$. For all figures, a Gaussian kernel was used and the smoothed average was protected according to a $(10\%, 0.1)$ rule for the attacker scenario of Section 3.6. In Figures 5.2-5.4, continuous noise is added to the smoothed average, where the standard deviation σ is chosen equal to the lower bound in Theorem 4.4.1 for noise on the total plot and the lower bound in Theorem 4.5.1 for noise on the numerator of the smoothed average. Per figure, the two different methods used the same realisation of the Gaussian field, apart from the standard deviation, so that the disturbances can be compared well. Again, we see that a larger bandwidth requires a smaller magnitude of the noise. In Figure 5.2, the noise is well visible for both protection methods, where in Figure 5.3 the differences are more hard to find, especially between the original smoothed average and the plot with noise on the numerator. In Figure 5.4, the plots are similar to a very large extend. If we consider the net disturbances of the two methods, we see that for all bandwidths, the method with noise on the total plot disturbs the plot more severely than the method with noise on the numerator.



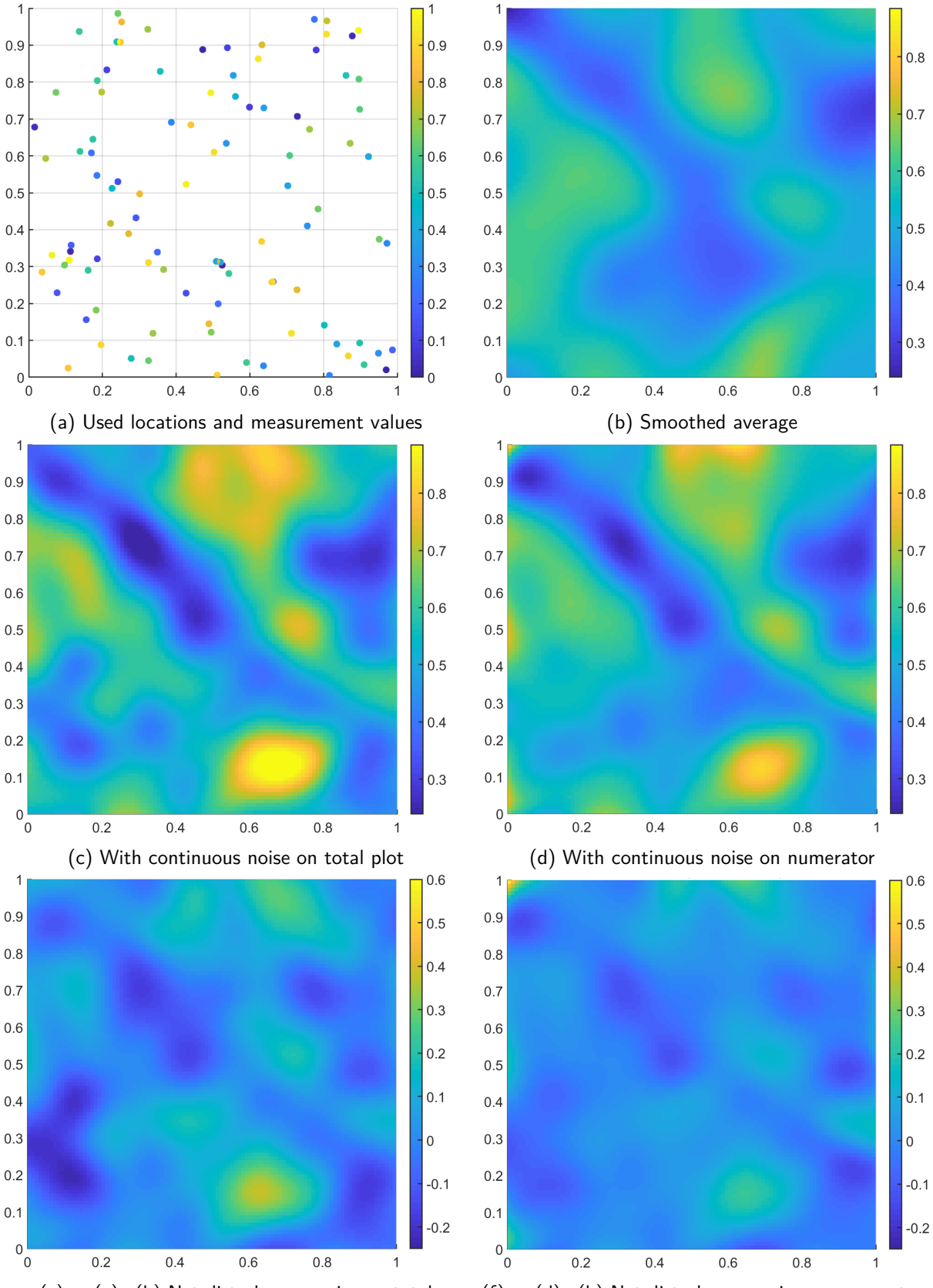
(a) $h = 0.01$ (compare to figure 2.1f)

(b) $h = 0.1$ (compare to 2.1e)

(c) $h = 0.12$

(d) $h = 0.15$

Figure 5.1: Noised smoothed average, Gaussian kernel, 50 points, 250^2 pixels, $(10\%, 0.1)$ rule



(a) Used locations and measurement values

(b) Smoothed average

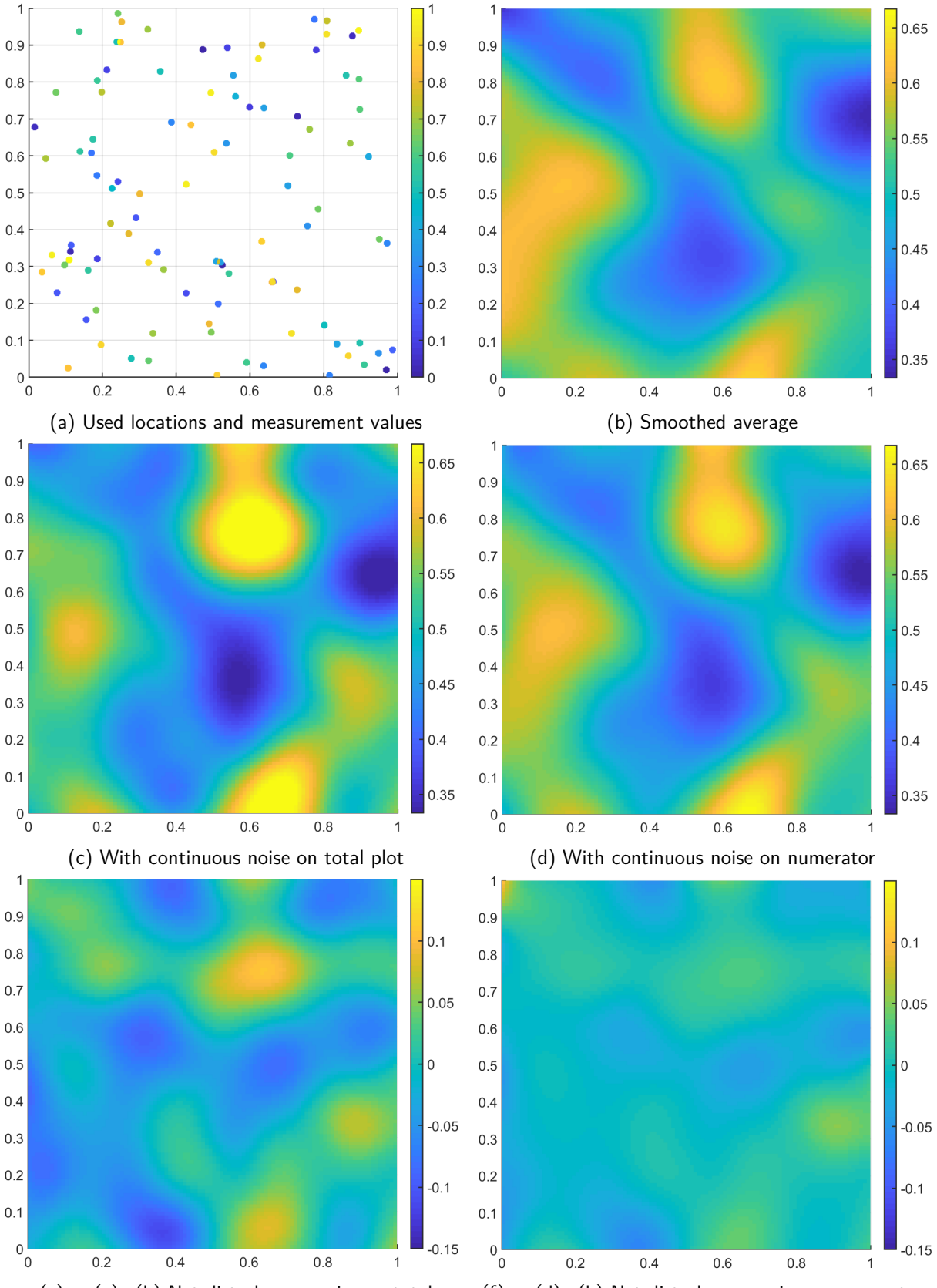
(c) With continuous noise on total plot

(d) With continuous noise on numerator

(e) = (c)−(b) Net disturbance, noise on total

(f) = (d)−(b) Net disturbance, noise on numerator

Figure 5.2: Plots using 100 uniformly chosen locations on the unit square, with measurement values uniformly chosen on $[0, 1]$. A Gaussian kernel was used with $h = 0.09$. The noised images are protected according to a $(10\%, 0.1)$ rule.



(a) Used locations and measurement values

(b) Smoothed average

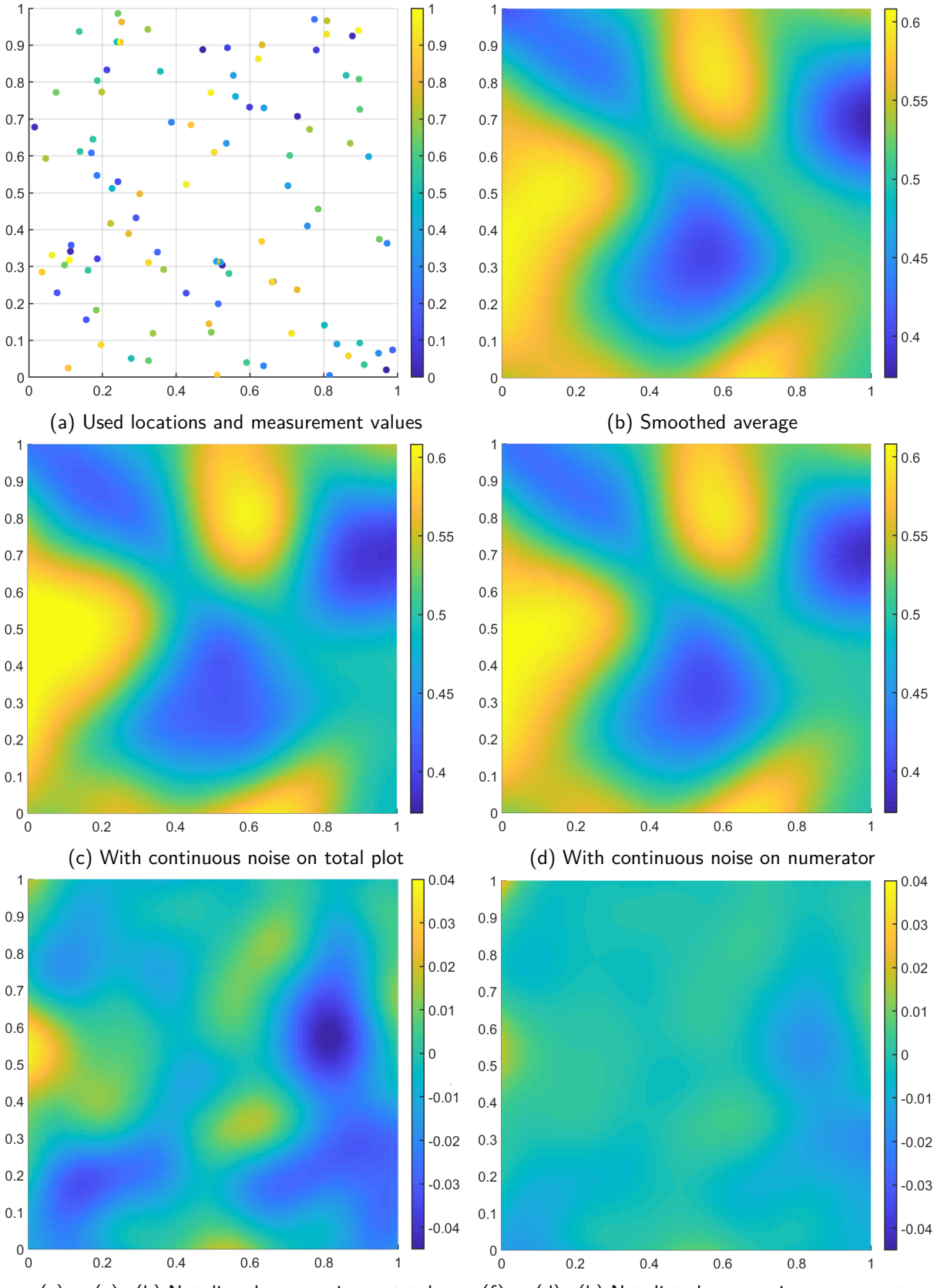
(c) With continuous noise on total plot

(d) With continuous noise on numerator

(e) = (c)−(b) Net disturbance, noise on total

(f) = (d)−(b) Net disturbance, noise on numerator

Figure 5.3: Plots using 100 uniformly chosen locations on the unit square, with measurement values uniformly chosen on $[0, 1]$. A Gaussian kernel was used with $h = 0.11$. The noised images are protected according to a $(10\%, 0.1)$ rule.



(a) Used locations and measurement values

(b) Smoothed average

(c) With continuous noise on total plot

(d) With continuous noise on numerator

(e) = (c)−(b) Net disturbance, noise on total

(f) = (d)−(b) Net disturbance, noise on numerator

Figure 5.4: Plots using 100 uniformly chosen locations on the unit square, with measurement values uniformly chosen on $[0, 1]$. A Gaussian kernel was used with $h = 0.13$. The noised images are protected according to a $(10\%, 0.1)$ rule.

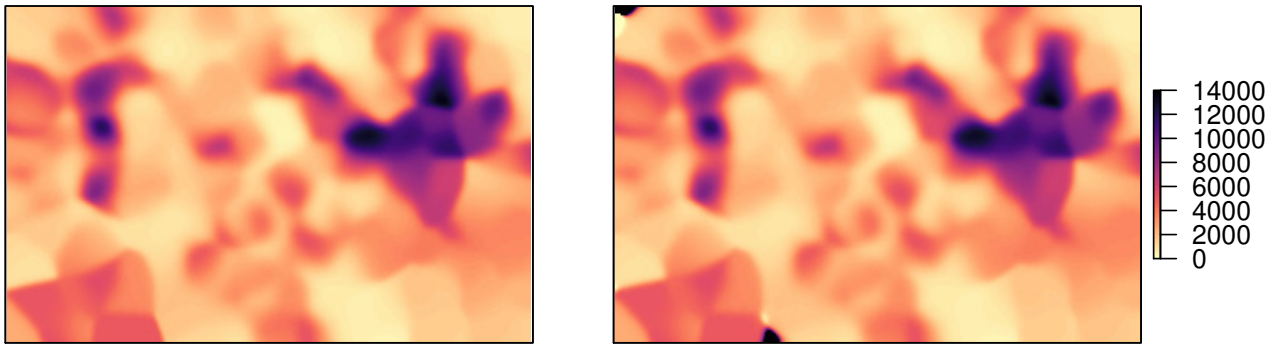


Figure 5.5: Unprotected (left panel) and protected (right panel) kernel weighted average of our entire synthetic dataset, according to a $(10\%, 0.1)$ rule for a Gaussian kernel with bandwidth $h = 250\,\text{m}$

5.2 Case Study

Instead of using uniform random data, we will use a synthetic dataset here, based on real data of energy consumption by enterprises. It is included in the sdcSpatial R-package that can be found on CRAN (De Jonge and De Wolf, 2019). This synthetic dataset of 8348 locations is based upon original data of 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. The locations of the enterprises were perturbed and random values were assigned for the energy consumption drawn from a log-normal distribution with parameters estimated from the original data. Spatial dependency in the energy consumption was introduced to mimic the compact industrial area and the densely packed greenhouses.

Figure 5.5 shows the unprotected kernel weighted average (2.7) and the protected kernel weighted average with noise on the numerator 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 σ of the random noise by (4.5). The plot of (4.2) resulting from that computation looks almost exactly identical to the plot of (2.7). 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 Figure 5.6 for $h = 100\,\text{m}$ and in Figure 5.7 for $h = 80\,\text{m}$, while Figure 5.8 displays the spatial structure of the locations in our entire synthetic dataset and the subset thereof.

We see that that the necessary disturbance to the plot is smaller in Figure 5.7 than in Figure 5.6. In order to be able to compare the results for different bandwidths, Figure 5.9 contains two graphs that show the influence of the bandwidth on σ for our synthetic data set. Note that the total disturbance of the plot is also influenced by the denominator of (4.2), 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 σ 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 a diagonal matrix. For $h \leq 5$ a single company with a very large energy consumption dominates the value of σ. Since this company is not present in the subset that we work with, a smaller σ may be used for the subset, also for $h \leq 5$.

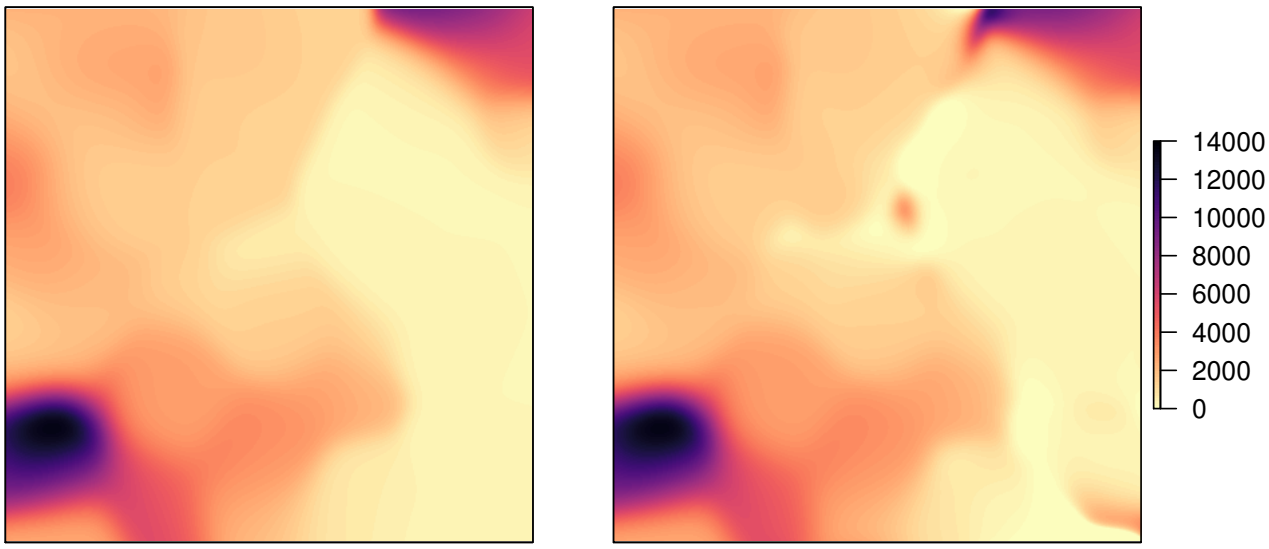


Figure 5.6: Unprotected (left panel) and protected (right panel) kernel weighted average of a part of our synthetic dataset, according to a $(10\%, 0.1)$ rule for a Gaussian kernel with bandwidth $h = 100\,\text{m}$

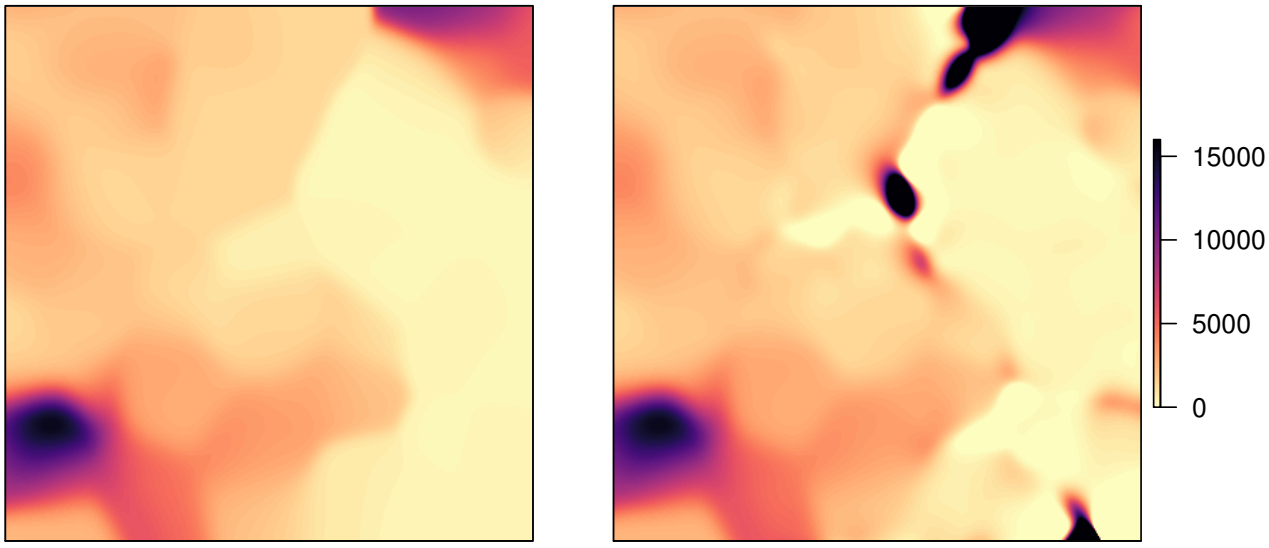


Figure 5.7: Unprotected (left panel) and protected (right panel) kernel weighted average of a part of our synthetic dataset, according to a $(10\%, 0.1)$ rule for a Gaussian kernel with bandwidth $h = 80\,\text{m}$

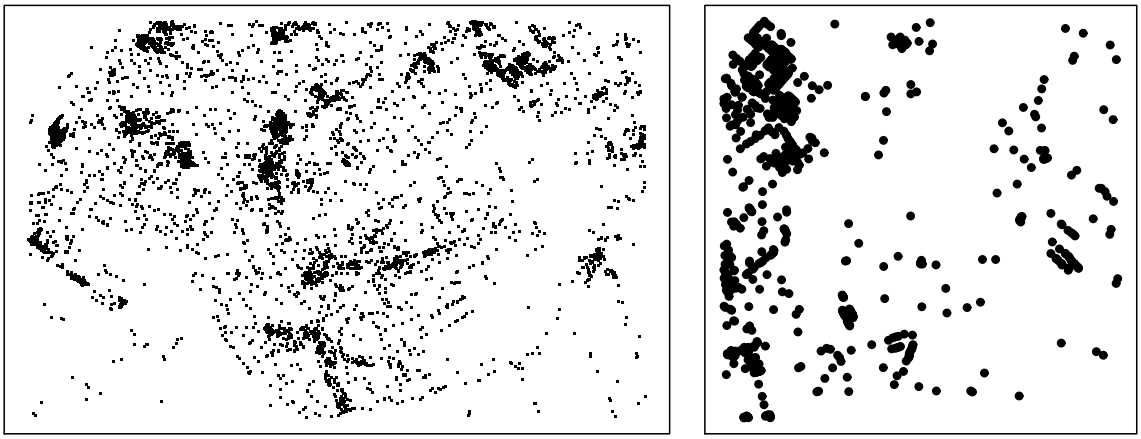


Figure 5.8: Map of enterprise locations in our entire dataset (left panel) and in the data subset (right panel)

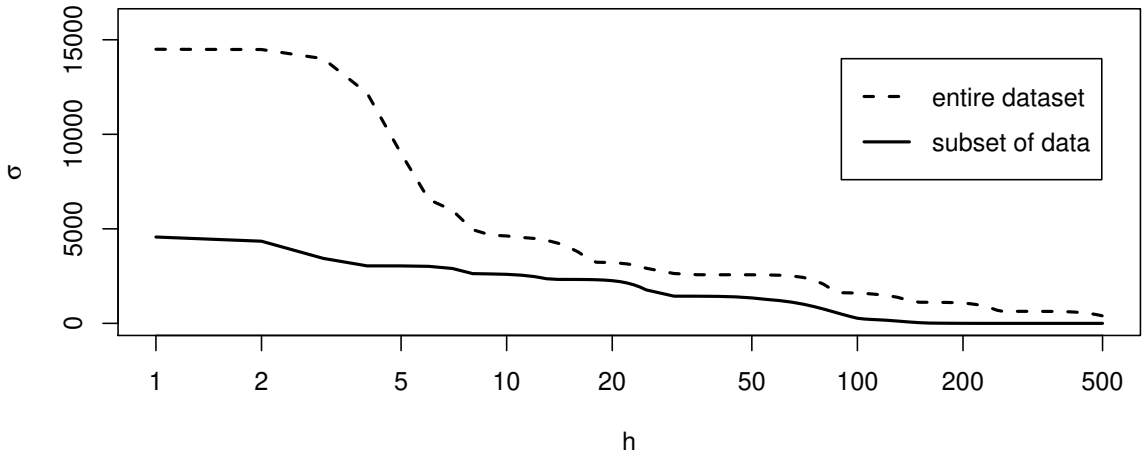


Figure 5.9: Standard deviation σ of added noise for different bandwidths

6 Discussion and Recommendations

In this report we looked into the disclosure properties of kernel smoothed average plots. We found that when an attacker is aware of both the kernel and the bandwidth used to produce the map, the original measurement values can be retrieved for some kernel types, by means of reading off the plotted values at the distinct population element locations and estimating the measurement values by solving a system of linear equations. For that reason, we introduced a new sensitivity rule that is applicable in this scenario. To protect the plot, we proposed to disturb the data by adding random noise. For three specific noise types, we derived a rule on how large the disturbance to the plot should be before publishing it.

The first of the three methods involved adding noise independently per pixel (Section 4.3. However, this type does not fulfill our continuity requirement and also we want to mention that it remains unclear if this method protects the plot well enough. It might be that in practice an attacker will take an average of the pixels close to a measurement location to obtain a better estimate of the plot value at that location.

Simulations that we considered for two continuous methods to add noise, indicated that in general the protected plot using noise on the numerator of the kernel weighted average (Section 4.5) suffers from a smaller net distortion than the plot using noise added to the total kernel weighted average (Section 4.4), which makes that the former plot is visually more attractive. Also, the noise on the numerator can be chosen to a computationally more elegant formula. Concluding, we propose to use the noise on the numerator and consider that our main result.

To investigate the efficacy of this type of noise 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. This is in line with the limit cases we considered in Section 2.3.3. 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. This could be seen very clearly in the case study.

We close with some final remarks and perspectives. First of all, note that the addition of noise in our method might lead to negative or extremely large values of (4.2) at locations where the population density is very small. In our case study, these locations were given the minimal or maximal colour scale values, to result in a realistic map for the user. In practical implementations, where the bandwidth might automatically become smaller when a user zooms in, one could choose to use a 'zoom stop' whenever the net disturbance of the plot becomes to large.

Secondly, our method requires that all $\boldsymbol{r}_i$, $i = 1, \ldots, N$ are distinct. We think that one of the most interesting future extensions is 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.

Another extension of the method might lie in taking boundary corrections into account and choosing the bandwidths per measurement location instead of having a single bandwidth for the whole plot. In that way, less densely populated areas might use a greater bandwidth, which protects them from disclosure control, without influencing densely populated regions on the other side of the plot.

Furthermore, it would be interesting to look at the utility of our plot for different bandwidth choices. Figure 5.9 is a first step in this direction but more research is needed.

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.

Bibliography

Abrahamsen, P. (1997), *A review of Gaussian random fields and correlation functions*. Norwegian Computing Center, www.nr.no/directdownload/2437/Abrahamsen_-_A_Review_of_Gaussian_random_fields_and_correlation.pdf.

Berman, M. and Diggle, P. (1989), "Estimating weighted integrals of the second-order intensity of a spatial point process." *Journal of the Royal Statistical Society*, 51, 81–92.

Borruso, G. (2003), "Network density and the delimitation of urban areas." *Transactions in GIS*, 7, 177–191.

Bowman, A.W. and Azzalini, A. (1997), *Applied smoothing techniques for data analysis*. Oxford University Press.

Chacón, J.E. and Duong, T. (2018), *Multivariate kernel smoothing and its applications*. CRC Press.

Chainey, S., Reid, S., and Stuart, N. (2002), "When is a hotspot a hotspot? a procedure for creating statistically robust hotspot maps of crime." In *Innovations in GIS 9: Socio-economic applications of geographic information science* (Kidner, D., Higgs, G., and White, S., eds.), 21–36, Taylor and Francis.

Cox, L.H. (1981), "Linear sensitivity measures in statistical disclosure control." *Journal of Statistical Planning and Inference 5*, 153–164. www.doi.org/10.1016/0378-3758(81)90025-2.

Danese, M., Lazzari, M., and Murgante, B. (2008), "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*, 415–429, Springer. LNCS 5072.

Davies, T.M. and Hazelton, M.L. (2010), "Adaptive kernel estimation of spatial relative risk." *Statistics in Medicine*, 29, 2423–2437.

Diggle, P.J. (1985), "A kernel method for smoothing point process data." *Journal of the Royal Statistical Society*, 34, 138–147. www.doi.org/10.2307/2347366.

Hundepool, A., Domingo-Ferrer, J., Franconi, L., Giessing, S., Schulte Nordholt, E., Spicer, K., and De Wolf, P.-P. (2012), *Statistical Disclosure Control*. Wiley series in Survey Methodology, John Wiley & Sons, Ltd. ISBN: 978-1-119-97815-2.

Hundepool, A. and De Wolf, P.-P. (2012), *Method Series - Statistical Disclosure Control*. Statistics Netherlands, www.cbs.nl/en-gb/our-services/methods/statistical-methods/output/output/statistical-disclosure-control.

De Jonge, E. and De Wolf, P.-P. (2016), "Spatial smoothing and statistical disclosure control." In *Privacy in Statistical Databases* (Domingo-Ferrer, Josep and Pejić-Bach, Mirjana, eds.), 107–117, Springer. LNCS 9867.

De Jonge, E. and De Wolf, P.-P. (2019), *sdcSpatial: Statistical Disclosure Control for Spatial Data*. `https://CRAN.R-project.org/package=sdcSpatial`. R package version 0.2.0.9000.

Lee, M., Chun, Y., and Griffith, D.A. (2019), "An evaluation of kernel smoothing to protect the confidentiality of individual locations." *International Journal of Urban Sciences*, 23, 335–351, DOI: 10.1080/12265934.2018.1482778.

Van Lieshout, M.N.M. (2012), "On estimation of the intensity function of a point process." *Methodology and Computing in Applied Probability*, 14, 567–578.

Van Lieshout, M.N.M. (2019), *Theory of Spatial Statistics - A Concise Introduction*. CRC Press.

Van Maarseveen, J.G.S.J. and Schreijnders, R. (eds.) (1999), *Welgeteld een eeuw*. Statistics Netherlands.

O'Keefe, C.M. (2012), "Confidentialising maps of mixed point and diffuse spatial data." In *Privacy in Statistical Databases*, 226–240, Springer.

Silverman, B.W. (1986), *Density estimation for statistics and data analysis*. Chapman & Hall.

Suñé, E., Rovira, C., Ibáñez, D., and Farré, M. (2017), "Statistical disclosure control on visualising geocoded population data using quadtrees." In *extended abstract at NTTS 2017*, `http://nt17.pg2.at/data/x_abstracts/x_abstract_286.docx`.

Wand, M.P. and Jones, M.C. (1994), *Kernel smoothing*. CRC Press.

Wang, Z., Liu, L., Zhou, H., and Lan, M. (2019), "How is the confidentiality of crime locations affected by parameters in kernel density estimation?" *International Journal of Geo-Information*, 8, 544–556, DOI: 10.3390/ijgi8120544.

Watson, G.S. (1964), "Smooth regression analysis." *Sankhya: The Indian Journal of Statistics*, 26, 359–372. `www.jstor.org/stable/25049340`.

De Wolf, P.-P. and De Jonge, E. (2017), "Location related risk and utility." Presented at UNECE/Eurostat worksession Statistical Data Confidentiality, 20–22 September, Skopje, `https://www.unece.org/fileadmin/DAM/stats/documents/ece/ces/ge.46/2017/3_LocationRiskUtility.pdf`.

De Wolf, P.-P. and De Jonge, E. (2018), "Safely plotting continuous variables on a map." In *Privacy in Statistical Databases* (Domingo-Ferrer, Josep and Montes, Francisco, eds.), 347–359, Springer. LNCS 11126.