Shape-Adaptive Kernel Density Estimation

L.E.N. Baakman

March 9, 2017

Abstract

Kernel density estimation has won in popularity in the past few years. Generally symmetric kernels are used, even though the data of which the density is estimated are not necessarily spread equally in all dimensions. To account for this asymmetric distribution of data we propose the use of shape adaptive kernels: kernels whose shape changes to fit the spread of the data in the local neighborhood of the point of which the density is estimated. We compare the performance of the shape adaptive kernels on both artificial and real world datasets with the performance of symmetric kernels on the unprocessed datasets and on the whitened datasets.

Results

Conclusion

1 Introduction

Estimating densities with kernels has been fairly popular of late; in the medical field this approach has been used to predict dose-volume histograms, which are drawn on when determining radiation doses [5]. Ecologists have explored the habitats of seabirds with density estimation [4]. Ferdosi et al. [3] have described it as “a critical first step in making progress in many areas of astronomy." Within this discipline density estimation is, among other things, used to estimate the density of the cosmic density field, which is required for the reconstruction of the large-scale structure of the universe.

Formally the aim of density estimation is to find the probability density $f(\mathbf{x})$ in d-dimensional Euclidean space underlying N points $\mathbf{x}_1 \dots \mathbf{x}_N$, that have been selected independently from $f(\mathbf{x})$.

Kernel density estimation methods approximate $f(\mathbf{x})$ by placing bumps, referred to as kernels, on the different observations and sums these to arrive at a final density estimate. This paper is concerned with a method to make the shape of these bumps adaptive to the local neighborhood of $\mathbf{x}$. Before introducing the concept of these shape-adaptive kernels we first review the different symmetric kernel density estimation methods that our approach is based on.

One often used approach to kernel density estimation is the Parzen approach[6]. This method estimates the density of $\mathbf{x}$ to be:

$$\hat{f}(\mathbf{x}) = \frac{1}{N}\sum_{j=1}^{N}\frac{1}{\sigma^d}K\left(\frac{\mathbf{x}-\mathbf{x}_j}{\sigma}\right), \tag{1}$$

where d denotes the dimensionality of the data points. The shape of the placed bumps is determined by the kernel function K. The Parzen approach requires the kernel to be a probability density function, i.e. $K(\mathbf{x}) \geq 0$ and $\int K(\mathbf{x}) = 1$. The width of the kernels is controlled by the bandwidth σ [7]. Choosing this bandwidth too small, results in a density estimate with spurious fine structures, whereas kernels that are too wide can over smooth the density estimate. Kernel estimators, such as the Parzen approach, that use kernels of the same width for all $\mathbf{x}_j$, are referred to as fixed-width estimators.

One downside of fixed-width methods is that they cannot respond appropriately to variations in the magnitude of the density function, i.e. the peakedness of the kernel is not data-responsive. Consequently in low density regions the density estimate will have peaks at the few sample points and be too low elsewhere. In areas with high density, the sample points are more densely packed together, which causes the Parzen estimate to spread out [1]. Adaptive-width methods address this disadvantage of the fixed-width methods by allowing the width of the kernel to vary per data point $\mathbf{x}_j$. Breiman, Meisel, and Purcell introduced such a method, which is defined as:

$$\hat{f}(\mathbf{x}) = \frac{1}{N}\sum_{j=1}^{N}(\alpha \cdot D_{j,k})^{-d}K_{\mathcal{G}}\left(\frac{\mathbf{x}-\mathbf{x}_j}{\alpha \cdot D_{j,k}}\right). \tag{2}$$

In Equation (2) $K_{\mathcal{G}}$ represents a Gaussian kernel, α is a multiplicative constant and $D_{j,k}$ the distance between $\mathbf{x}_j$ and the k nearest neighbor of $\mathbf{x}_j$. Comparing Equation (1) with (2) we find that the bandwidth σ, has been replaced with $\alpha D_{j,k}$. In low density regions $D_{j,k}$ will be large, and the kernel will be spread out, in high density regions the converse occurs, this allows the kernel to adapt its width to density of the local neighborhood. Breiman, Meisel, and Purcell use a minimization algorithm on a goodness of fit statistic to find suitable values for k and α. We shall refer to this estimator as the Breiman Estimator.

Silverman [7] showed that the minimization procedure used by Breiman, Meisel, and Purcell implicitly uses a k-NN pilot estimate. If pilot estimates are used explicitly the density estimation process becomes:

(i) Find a pilot estimator such that $\forall i\, \tilde{f}(\mathbf{x}_i) > 0$.

(ii) Define local bandwidth factors λ_i by

$$\lambda_i = \left(\frac{\tilde{f}(\mathbf{x}_i)}{\mathrm{GM}\left(\tilde{f}(\mathbf{x}_0), \ldots, \tilde{f}(\mathbf{x}_N)\right)} \right)^{-\beta} \tag{3}$$

where GM denotes the geometric mean and the sensitivity parameter β must lie in the range $[0,1]$.

(iii) Compute the adaptive kernel estimate as

$$\hat{f}(\mathbf{x}) = \frac{1}{N} \sum_{i=1}^{N} (\sigma \cdot \lambda_i)^{-d} K\left(\frac{\mathbf{x} - \mathbf{x}_j}{\sigma \cdot \lambda_i} \right) \tag{4}$$

with K integrating to unity.

The pilot densities computed in step (i) do not need to be sensitive to the fine details of the pilot estimate. Therefore a convenient method, e.g. the Parzen approach, can be used to estimate them. The local bandwidths computed in (ii) depend on the exponent β, if this value is high they are more sensitive to variations in the pilot densities. For $\beta = 0$ Equation (4) reduces to a fixed width kernel density estimation. In the literature two values of β are prevalent. Breiman, Meisel, and Purcell [1] argue that choosing $\beta = {}^1\!/_d$ will ensure that the number of observations covered by the kernel will be approximately the same in all parts of the data. Whereas Silverman favors $\beta = {}^1\!/_2$ independent of the dimension of the data points, as this value results in a bias that can be shown to be of a smaller order than that of the fixed-width kernel estimate.

One disadvantage of the approach taken by Breiman, Meisel, and Purcell is that it is computationally expensive, this is partially due to the Gaussian kernel. Because of the infinite base of this kernel an exponential function has to be evaluated N times to estimate the density of one data point. Wilkinson and Meijer [8] propose to reduce this computational complexity in two ways. Firstly they replace the infinite base Gaussian kernel with a symmetric Epanechnikov kernel in both Equation (1) and Equation (4), i.e. in the computation of the pilot densities and the final densities. This kernel is defined by:

$$K_{\mathcal{E}}(\mathbf{x}) = \begin{cases} \frac{d+2}{2c_d}(1 - \mathbf{x} \cdot \mathbf{x}) & \text{if } \mathbf{x} \cdot \mathbf{x} < 1 \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

where c_d denotes the volume of the d-dimensional unit sphere. It should be noted that the kernel defined in Equation (5) does not have unit variance. this can be corrected by multiplying the bandwidth, σ, with the square root of the variance of $K_{\mathcal{E}}$. There are two advantages to using this kernel, firstly it is computationally much simpler than the Gaussian kernel, in part due to its finite base and secondly it is optimal in the sense of the Mean Integrated Square Error (MISE) [2]. A disadvantage of this kernel is that it is not continuously differentiable. This does not matter when computing the pilot densities, as they are only used to choose the local bandwidths. In the computation of the final densities it is a trade off between a continuously differentiable $\hat{f}$ and a low computational complexity.

The second change Wilkinson and Meijer [8] propose is to compute the pilot densities indirectly. They first compute the pilot densities for the vertices of a grid that covers all data points, before determining the actual pilot densities by multi-linear interpolation. The width of the kernel used for the computation of the pilot densities is computed with

$$\sigma = s \cdot N^{\left(\frac{-1}{d+4}\right)} \left(\frac{8(d+4) \cdot (2\sqrt{\pi})^d}{c_d} \right)^{\frac{1}{d+4}}, \tag{6}$$

where s the square root of the average of the variances of the different dimensions. The final densities are estimated with Equation (4) using the general and local bandwidths estimated with Equation (6) and (3), respectively. This estimator will be referred to as the Modified Breiman Estimator (MBE).

Ferdosi et al. [3] considered the application of density estimation on large datasets, i.e. sets with more than 50 000 points with the dimension of the data points ranging from ten to hundreds of elements. They used the MBE, but introduced as simpler method to estimate the bandwidth. First they determine an intermediate bandwidth for each dimension l of the data according to:

$$\sigma_l = \frac{P_{80}(l) - P_{20}(l)}{\log N}, l = 1, \ldots, d, \qquad (7)$$

where $P_{20}(l)$ and $P_{80}(l)$ are the twentieth and eightieth percentile of the data in dimension l, respectively. The optimal pilot window width, σ, is chosen as the minimum of $\sigma_1, \ldots, \sigma_d$ to avoid oversmoothing.

Although the widths of the kernels used in the estimators proposed by Breiman, Meisel, and Purcell, Wilkinson and Meijer are sensitive to the data, the shapes of the kernels are dependent on the kernel itself not the data. To further increase the responsiveness of the estimator to the data we propose the use of shape-adaptive kernels in density estimation. Not only the width but also the shape of these kernels is steered by the local neighborhood of the data.

A disadvantage of these shape-adaptive kernels could be that in regions where the density of sample points is low, there are insufficient data points to compute the shape of the kernel reliably. Consequently we let the amount of influence exerted by the local data on the shape of the kernel be dependent on the number of the data points in that region.

This paper is organized as follows. Section 2 discusses the proposed shape-adaptive kernels. The experiments used to investigate the performance of these kernels are presented in Section 3. Section 4 gives the results of these experiments, they are discussed in Section 5 and, this paper is concluded in Section 6.

2 Method

We use our shape adaptive kernels in combination with the Modified Breiman Estimator introduced by Wilkinson and Meijer [8]. The grid that the pilot densities are computed on a uniform gird that covers all data points with twenty vertices in each dimension. We choose to use the method proposed by Ferdosi et al. [3] for computing the general bandwidth because of its lower complexity, compared to the method used by Wilkinson and Meijer [8]. We have empirically determined that using $\beta =$ works best in our case. The final densities are estimated according to Equation (4) with a reshaped and scaled Epanechnikov kernel. The Epanechnikov kernel reshaped with the matrix Σ is defined as:

hoe hebben we dat vastgesteld

Een of andere waarde

As of now incorrect and incomplete

$$K_{\mathcal{E}}(\mathbf{x}) = \begin{cases} ? * \frac{d+2}{2c_d}\left(1 - \mathbf{x}\Sigma\mathbf{x}\right) & \text{if } \mathbf{x}\cdot\mathbf{x} < 1 \\ 0 & \text{otherwise.} \end{cases} \qquad (8)$$

The shape of the kernel is determined based on the neighborhood of the pattern, $\mathbf{x}$, whose density we are estimating. For each data point the matrix Σ is determined according to these

Stapjes, en daarna verwijzen!

We determine the neighbors of $\mathbf{x}$ with the k nearest neighbors algorithm (k-NN) with Euclidean distance. This approach is used rather than a fixed-radius neighborhood to ensure that independent of the sparsity of the data the kernel shape is always based on a reasonable number of data points. Furthermore using k-NN instead of the fixed-radius approach allows us to choose k in such a way that the number of patterns the Σ is based on is greater than d. This which makes it extremely improbable that the covariance matrix of the neighborhood is singular. We follow Silverman's [7] recommendation of choosing $k = \sqrt{N}$. To ensure that even the k computed for small high-dimensional data sets satisfies $k \geq d$ we use

$$k = \max\left(\left\lfloor\sqrt{N}\right\rceil, d\right).$$

We let $C_{\mathbf{x}}$ denote the union of $\mathbf{x}$ and its k neighbors, the basic shape of the kernel used for $\mathbf{x}$ is then given by the biased covariance matrix of $C_{\mathbf{x}}$.

To allow the density estimation of each pattern to be influenced by an equal area, before the application of the smoothing factor $\sigma\lambda_i$, the basic shapes of the kernels need to be scaled. To that end we use the eigenellipse, the ellipse defined by the eigenvectors and eigenvalues of $\text{cov}(C_{\mathbf{x}})$. We scale the covariance matrix with the factor S. For the Gaussian kernel this factor is defined as:

$$S_{\mathcal{G}} = \frac{\sigma^2}{\text{GM}\left(\sqrt{\lambda_1}, \ldots, \sqrt{\lambda_d}\right)}, \qquad (9)$$

where λ_j denotes the jth eigenvalue of the jth eigenvector of Σ. This reduces to $S = h\sqrt{h}$ for a standard Gaussian kernel. If the Epanechnikov kernel is used the scaling factor is:

Klopt dit? Wiskundig gezien en met wat we weten van de standard gaussian

$$S_{\mathcal{E}} = ? \qquad (10)$$

The scaling factors ensure that the shape-adapted covariance matrix has the same scale as the covariance matrix that is implicitly used in the Modified Breiman Estimator with a Gaussian kernel.

3 Experiment

Describe the experiment: estimate the density of the different datasets with three different methods: MBE, SAMBE, MBE with prewhitened data.

Introduce the subsections

3.1 Datasets

We examine the performance of the estimators on on two groups of datasets: a number of simulated datasets with known desnity fields and a real world dataset with an unkown density field.

Simulated Datasets

The simulated datasets are based on the simulated datasets used by Ferdosi et al. [3]. Figure 1 presents scatter plots of the data sets. The definitions of the these simulated data sets are shown in Table 1.

Dataset 1, shown in Figure 1a, is the most simple set in this group. It is an unimodal Gaussian distribution with random noise added.

The second dataset, depicted in Figure 1b, contains two Gaussian distributions with different covariance matrices and uniform noise. The means and covariance matrices of the Normal distributions are such that the two distribtions are very unlikely to overlap.

Dataset 3, represented in Figure 1c, consists of four different normal distributions and uniformly distributed noise. The four Gaussian distributions are placed in such a way that it is unlikely that among them any overlap occurs.

Figure 1d illustrates dataset 4. This set consists of a horizontal wall-like structure and a vertical filament-like structure.

The fifth dataset, shown in Figure 1e, contains three intersecting walls. For each point in these walls its position in two of the three dimensions is drawn from a uniform distubtion, the third coordiante is sampled from a Gaussian distribution.

We expect comparable performance from alle estimators on dataset one through three, as other than the randomly sampled noise these sets only contain data sampled from a Gaussian distribution with a diagonal covariance matrix. Which results in an equal spread of the data in all dimensions for the non-noise data. Dataset four and five are clearly spread more in one dimension than in others, thus we expect that the shape adaptive estimator and the estimator on the whitened data will perform better on these sets than the MBE estimator.

The increasing complexity of these datasets allows us to investigate the performance of the classifier on simple situations, one cluster of data with some noise, to complex density fields that should better approximate real world data. The advantage of using simulated data is that the true densities of the data points are known, which allows us to test how well the different methods estimate the densities.

Real World Datasets

Iets over real world data

3.2 Error Measures

To quantify the performance of the estimators on the simulated datasets we use the Mean Squared Error (MSE):

$$\mathrm{MSE}\left(\hat{f}(\cdot)\right) = \frac{1}{N}\sum_{i=0}^{N}\left(\hat{f}(\mathbf{x}i) - f(\mathbf{x}i)\right)^2.$$

Iets over het quantificeren van de resultaten op real world data.

4 Results

5 Discussion

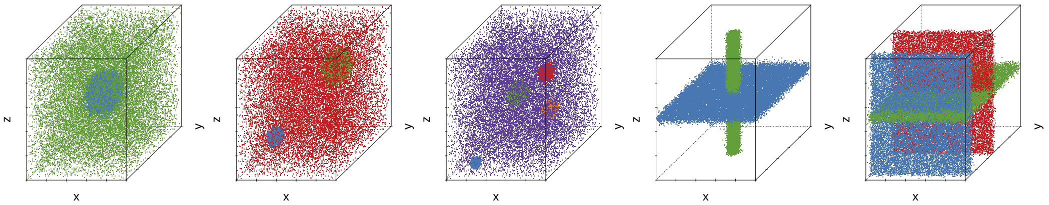
6 Conclusion

References

[1] L. Breiman, W. Meisel, and E. Purcell. “Variable Kernel Estimates of Multivariate Densities”. In: *Technometrics* 19.2 (1977), pp. 135–144.

[2] V.A. Epanechnikov. “Non-Parametric Estimation of a Multivariate Probability Density”. In: *Theory of Probability & Its Applications* 14.1 (1969), pp. 153–158.

[3] B.J. Ferdosi et al. “Comparison of Density Estimation Methods for Astronomical Datasets”. In: *Astronomy & Astrophysics* 531 (2011).



(a) *Set 1* **(b)** *Set 2* **(c)** *Set 3* **(d)** *Set 4* **(e)** *Set 5*

Figure 1: *Scatter plot representation of the simulated datasets defined in Table 1.*

Set	Component	Fraction	Distribution
1	Trivariate Gaussian 1	$^2/_3$	$(x,y,z) \sim \mathcal{N}\big([50,50,50],\mathrm{diag}(30)\big)$
	Uniform random background	$^1/_3$	$(x,y,z) \sim \mathcal{U}\big([0,0,0],[100,100,100]\big)$
2	Trivariate Gaussian 1	$^1/_3$	$(x,y,z) \sim \mathcal{N}\big([25,25,25],\mathrm{diag}(5)\big)$
	Trivariate Gaussian 2	$^1/_3$	$(x,y,z) \sim \mathcal{N}\big([65,65,65],\mathrm{diag}(20)\big)$
	Uniform random background	$^1/_3$	$(x,y,z) \sim \mathcal{U}\big([0,0,0],[100,100,100]\big)$
3	Trivariate Gaussian 1	$^1/_6$	$(x,y,z) \sim \mathcal{N}\big([24,10,10],\mathrm{diag}(2)\big)$
	Trivariate Gaussian 2	$^1/_6$	$(x,y,z) \sim \mathcal{N}\big([33,70,40],\mathrm{diag}(10)\big)$
	Trivariate Gaussian 3	$^1/_6$	$(x,y,z) \sim \mathcal{N}\big([90,20,80],\mathrm{diag}(1)\big)$
	Trivariate Gaussian 4	$^1/_6$	$(x,y,z) \sim \mathcal{N}\big([60,80,23],\mathrm{diag}(5)\big)$
	Uniform random background	$^1/_3$	$(x,y,z) \sim \mathcal{U}\big([0,0,0],[100,100,100]\big)$
4	Wall-like structure 1	$^1/_2$	$(x,z) \sim \mathcal{U}\big([0,0],[100,100]\big)$, $(y) \sim \mathcal{N}\big(5,5\big)$
	Filament-like structure	$^1/_2$	$(x,y) \sim \mathcal{N}\big([50,50],\mathrm{diag}(5)\big)$, $(z) \sim \mathcal{U}\big(0,100\big)$
5	Wall-like structure 1	$^1/_3$	$(x,z) \sim \mathcal{U}\big([0,0],[100,100]\big)$, $(y) \sim \mathcal{N}\big(10,5\big)$
	Wall-like structure 2	$^1/_3$	$(x,y) \sim \mathcal{U}\big([0,0],[100,100]\big)$, $(z) \sim \mathcal{N}\big(50,5\big)$
	Wall-like structure 3	$^1/_3$	$(x,z) \sim \mathcal{U}\big([0,0],[100,100]\big)$, $(y) \sim \mathcal{N}\big(50,5\big)$

Table 1: *The simulated datasets used to test the estimators. The column 'Fraction' indicates for each component of the dataset which fraction of the total number of points of the data set is part of that component.* $\mathcal{N}(\mu,\Sigma)$ *denotes a Gaussian distribution with mean* μ *and covariance matrix* Σ. *A diagonal matrix with the value* x *on the diagonal is represented as* $\mathrm{diag}(x)$. $\mathcal{U}(a,b)$ *denotes a uniform distribution with its minimum and maximum set to a and b, respectively.*

[4] Kirsty J Lees, Andrew J Guerin, and Elizabeth A Masden. “Using kernel density estimation to explore habitat use by seabirds at a marine renewable wave energy test facility”. In: *Marine Policy* 63 (2016), pp. 35–44.

[5] Johanna Skarpman Munter and Jens Sjőlund. “Dose-volume histogram prediction using density estimation”. In: *Physics in Medicine and Biology* 60.17 (2015), p. 6923.

[6] E. Parzen. “On Estimation of a Probability Density Function and Mode”. In: *The Annals of Mathematical Statistics* 33.3 (1962), pp. 1065–1076.

[7] B.W. Silverman. *Density Estimation for Statistics and Data Analysis*. Monographs on Statistics and Applied Probablity. Springer-Science+Business Media, B.V., 1986.

[8] M.H.F. Wilkinson and B.C. Meijer. “DATAPLOT: A Graphical Display Package for Bacterial Morphometry and Fluorimetry Data”. In: *Computer Methods and Programs in Biomedicine* 47.1 (1995), pp. 35–49.