

Dear Reviewer 1

We are thankful for the many helpful comments and suggestions for improving our manuscript. By addressing all of your points, we have made significant changes to the structure of our manuscript. We have also added another example that should be illustrative and inspiring for those readers who do not intend to fully understand all the technical details of the diffKDE. Your comments are shown in grey. Our responses are in black. We numbered comments and responses to allow cross-referencing. The line numbers are referring to the manuscript document with the tracked changes below.

General Comments

The authors present a novel kernel density estimation approach that combines a diffusion-based method with a pilot step to produce better estimates of the bandwidth - a classically difficult problem. The authors also develop a Python package, diffKDE, for easy use of their method and highlight some of the package's capabilities within the manuscript. diffKDE is benchmarked against pre-existing KDE methods using synthetic data and applied to real marine data, highlighting its applicability. However, the material is organized and presented in such a manner that detracts from its scientific contribution. Additionally, as noted in the points below, the verbiage, lack of technical definitions in multiple locations throughout the text, and sectioning of the ideas muddle the presentation of the material. I recommend the authors revise the manuscript and give special attention to the structure of the manuscript and the presentation clarity. I believe this work has potential and I look forward to the reading the revised manuscript.

Specific Comments

The organization of the paper detracts from the scientific contribution. Here are a couple examples:

RC1 – 1 Section 2.1 gives background on the general kernel density estimator and the proposed diffusion-based kernel density estimator with pilot study. Then section 2.2 and 2.3 discuss discretization of the diff-KDE and implementation, respectively. This feels out of order. I would assume general background would be its own section (e.g., make section 2.1 its own section)

RC1 – A1 We made section 2.1.1 its own section named "2 The general kernel density estimator" presenting the general introduction to KDEs.

RC1 – 2 and the proposed method and all associated ideas would be grouped together (e.g., make sections 2.1.2 through 2.4 its own section, say section 3, and order these appropriately).

RC1 – A2 We made the theoretical background of the diffusion KDE an own section named "3 The diffusion-based kernel density estimator" including the subsections for bandwidth selection and pilot estimation. The second and third paragraph of Section 3 are re-ordered and completed to first give a general introduction and motivation for the

diffusion-based KDE and explain how the bandwidth is related to the time propagation and then describe the connection to the Gaussian KDE (see RC2 – A8).

The following final parts of the Methods section are now restructured into two additional sections showing first the theory behind our new approach to the diffKDE in a section “4 The new bandwidth approximation and pilot estimation approach of the diffKDE” and second the explicit discretization and implementation of the diffKDE in “5 Discretization and implementation of the diffKDE”. Redundant subheadings were removed.

We added the phrase: “...on an equidistant spatial grid $(x_i)_{i=0}^n \in \Omega$ of the spatial domain $\Omega \subseteq \mathbb{R}$.” in l. 291.

We deleted the sentences: “The selected implementation is a straightforward approach using equidistant finite differences in space and time and a direct solution of the diffusion equation by an implicit Euler.” and begin the new section 4 with “Our new approach solves the diffusion equation in three stages, ...”

We renamed the section describing the algorithm realized in Python to: “5.4 The diffKDE algorithm with optimized bandwidth” and the final section in this part, which describes the implemented functionality for direct visual output, to: “5.5 Pre-implemented functions for visual outputs”

RC1 – 3 Regarding the order material is presented, there is no initial explanation as to why a pilot estimation is needed. It feels as though section 2.1.4 needs some prior context or motivation.

RC1 – A3 We included an introduction for the need of pilot estimation into the chapter about bandwidth selection as: “This implicit dependency can be solved by so-called pilot estimation steps. Pilot estimates are generally rough estimates of f calculated in an initial step to use them for an approximation of Eq. 17 and 18, which later on serve to calculate a more precise estimate of f . A more detailed introduction into pilot estimation and its specific benefit for diffusion-based KDEs is presented in Sec. 3.2.

Botev et al. (2010) use an iterative scheme to solve the implicit dependency of the bandwidth parameter on the true distribution f . This additional...” in ll. 237 – 241.

RC1 – 4 Section 3 contains the simulation example and real world example. I would make these distinct sections (e.g., following the above point, sections 4 and 5). To this point, the first paragraph of section 3 is confusion. There is no clear distinction between the simulation explanation and the real world example. By separating these, that should mitigate any confusion.

RC1 – A4 We separated the results section, focussing first on artificial data “6 Results on artificial data” and on real marine data “7 Results on marine biogeochemical data and outlook to model calibration”.

In the introductory paragraph we deleted the sentence referring to different data sources and the specific mentioning of the results obtained by the pre-implemented functions (“Different data sources are chosen to best show possibilities and performance of the diffKDE. Additionally, snapshots of the pre-implemented plot routines are given as examples.” ll. 507 – 509).

We moved the final paragraph of the prior Results introduction to the beginning of the new second results section “Results on marine biogeochemical data and outlook to model calibration” and rephrased it to: “The performance of the diffKDE is now illustrated with real data of a) measurements of carbon isotopes (Verwega et al., 2021, Verwega et al., 2021}), b) of plankton size (equivalent spherical diameter) (Lampe et al., 2021) and c) remote sensing data (Sathyendranath et al., 2019, 2021). We chose these data because we propose to apply the diffKDE for the analysis of field data for assessment and optimization of marine biogeochemical- as well as size-based ecosystem models. The carbon isotope data have been collected to constrain model parameter values of a marine biogeochemical model that incorporates this tracer as a prognostic variable (Schmittner and Somes, 2016.” in ll.634 – 641.

Furthermore, we deleted the sentences: “In this final part, we show the diffKDE's performance on real marine biogeochemical field data. We chose two example data: A set of $\delta^{13}\text{C}$ in particulate organic carbon (POC) (Verwega et al., 2021) data and a set of plankton size spectra data (Lampe et al., 2021).” to avoid redundancies.

We incorporated two subsections into “7 Results on marine biogeochemical data and outlook to model calibration” namely “7.1 Performance analyses on organic carbon-13 isotope data” and “7.2 Performance analyses on plankton size spectra data”

I strongly recommend reorganizing the paper by grouping similar ideas by sections with appropriate sub-sections.

RC1 – 5 Some of the verbiage and notation used in this paper may preclude the general audience of GMD from understanding its scientific merit. There are some examples included in the list below, but I recommend the authors check the manuscript and make sure all mathematical terms are defined and avoid using overly technical notation if possible.

RC1 – A5 We clarified and simplified the usage of mathematical terminology at various places in detail listed in the answer to RC2 (see RC2 – A1, RC2 – A4, RC2 – A5, RC2 – A6, RC2 – A7).

Abstract: Some of the sentences need some work. For example:

RC1 – 6 Line 1: A PDF is a function defining the probability of a random variable taking on a specific value. I don't like the use of observed or simulated variables or the phrase, “comprise basic information”, as I think this is a little too weak of wording.

RC1 – A6 We rephrased the sentence to: “Probability density functions (PDFs) provide information about the probability of a random variable taking on a specific value.” in ll. 1-2.

RC1 – 7 Line 6: Starting with “A diffusion-based KDE...” Why is this the case? Maybe rephrase to say, “Diffusion-based KDEs have been shown to provide a useful approach ...”

RC1 – A7 We deleted this sentences and formulated more precisely: “In this study, we designed and developed a new implementation of a diffusion-based KDE as an open source Python tool to make diffusion-based KDE accessible for general use. Our new diffusion-based KDE provides (1) consistency at the boundaries, (2) better resolution of multimodal data, (3) and a family of KDEs with different smoothing intensities. We

demonstrate our tool on artificial data with multiple and boundary close modes and on real marine biogeochemical data, and compare our results against other popular KDE methods.” in ll. 8 – 13.

RC1 – 8 Line 10: What is boundary close data and what details are suppressed? Do you mean your approach produces a smooth surface that is robust to noise and outliers?

RC1 – A8 We rephrased the sentence to: “Our estimator is able to detect relevant multiple modes and it resolves modes that are located closely to a boundary of the observed data interval. Furthermore, our approach produces a smooth graph that is robust to noise and outliers.” in ll. 14 – 17.

RC1 – 9 Also, there are various other places in the manuscript that use “details”, but it is unclear what is being referred to.

RC1 – A9 By details we refer to multiple modes that can be resolved with the diffKDE. We used the term “details” in the example of the plankton size spectra. Since we have now included in the introduction section a description of the relevance and the role of KDE for analyzing continuous size spectra, we explain what we mean by “details”: “The identification of structural details in the size spectra, such as distinct elevations (modes) and troughs within certain size ranges, is useful, since they can reveal some of the underlying structure of the plankton foodweb.” in 52 – 53.

In addition, we wrote in subsection (7.2 Performance analyses on plankton size spectra): “This procedure avoided over-fitting but was also prone to over-smoothing, which can mask details, such as troughs in specific size ranges.” in ll. 679 – 680.

RC1 – 10 The introduction falls a short. There is a nice body of literature on diffusion-based KDE and it would be beneficial to this manuscript if it is put into context with the current literature.

RC1 – A10 We added the reference

Majdara, A., & Nooshabadi, S. (2019). Nonparametric density estimation using copula transform, bayesian sequential partitioning, and diffusion-based kernel estimator. *IEEE Transactions on Knowledge and Data Engineering*, 32(4), 821-826.

as: “..., especially for the resolution of multiple modes (e.g. Majdara and Nooshabadi, 20).” in l 89.

And

Li, G., Lu, W., Bian, J., Qin, F., & Wu, J. (2019). Probabilistic optimal power flow calculation method based on adaptive diffusion kernel density estimation. *Frontiers in Energy Research*, 7, 128.

Santhosh, D., & Srinivas, V. V. (2013). Bivariate frequency analysis of floods using a diffusion based kernel density estimator. *Water Resources Research*, 49(12), 8328-8343.

Xu, X., Yan, Z., & Xu, S. (2015). Estimating wind speed probability distribution by diffusion-based kernel density method. *Electric Power Systems Research*, 121, 28-37.

as: “The improved structure resolution has for example already shown useful for the optimization of photovoltaic power generation (Li et al., 2019), analysis of flood frequencies (Santhosh and Srinivas, 2013) or the prediction of wind speed (Xu et al., 20).” in ll. 90 – 91.

RC1 – 11 Line 19: Any citations to support the claim for the need in other model applications? This seems like a great place to include more references.

RC1 – A11 We have thoroughly revised the first paragraphs of the introduction section, and have included references to studies where the computation and analyses of PDFs seem particularly useful (Dessai et al., 2005; Perkins et al., 2007; Palmer 2012). We also added: “Obtaining high quality approximations of nonparametric PDFs is certainly not limited to applications in the geosciences but is likely desirable in other scientific fields as well. In aquatic ecological research, for example, continuous plankton size spectra can be well derived from PDFs of cell size measurements sorted by individual species or plankton groups (Quintana et al., 2008; Schartau et al., 2010; Lampe et al., 2021). The identification of structural details in the size spectra, such as distinct elevations (modes) and troughs within certain size ranges, is useful, since they can reveal some of the underlying structure of the plankton foodweb. A typical limitation of the approach described in Schartau et al. (2010) and Lampe et al. (2021) is the specification of an estimator for the continuous size spectra, such that all significant details are well resolved.” in ll. 48 – 55.

RC1 – 12 Lines 24-27: Your definition of a PDF does not seem correct. You need to explicitly define each element of your probability space Ω, A, P . Also, I assume A is a σ -algebra and Ω is the sample space. In which case I think you mean to have $f : \Omega \rightarrow [0, \infty)$ (also, open bracket on the right).

RC1 – A12 We simplified the definition of the PDF to the essential parts and re-wrote it to: “Mathematically formulated, PDFs are integrable non-negative functions $f : \Omega \rightarrow [0, \infty)$ from a sample space $\Omega \subseteq \mathbb{R}$ into the non-negative real numbers with $\int_{-\infty}^{\infty} f(x) dx = 1$ ” in l. 56.

RC1 – 13 Also, the sentence, “By definition, ...” should be reworded.

RC1 – A13 We re-phrased the sentence to: “PDFs correspond to the probability P of the occurrence of a data value $X \in \mathbb{R}$ within a specific range $[a, b] \subseteq \mathbb{R}$ via the relationship...” in ll. 59 – 60.

RC1 – 14 Line 41: I like the list, but are there any more current (last 5 or so years) papers that could be included? A cursory literature search returned quite a few recent papers on the topic.

RC1 – A14 See RC2 – A10.

RC1 – 15 Line 48: Define δ -distribution, or say something like, “which will be formally defined in section xxx.”

RC1 – A15 We changed the sentence accordingly and added directly following the definition of the diffKDE: “In general, δ is defined by $\delta(x)=0$ for all $x \in \mathbb{R} \setminus \{0\}$ and $\int_{-\infty}^{\infty} \delta(x) dx = 1$ (Dirac, 1927).” in ll. 207 – 208.

RC1 – 16 Line 75: Are all X_j assumed to come from the same distribution or are they completely independent? Should they be independent or independent and identically distributed?

RC1 – A16 For the general mathematical formulation we need them to be independent and identically distributed. We changed the sentence to: “...independent identically distributed real random variables.” in ll. 118 – 119.

RC1 – 17 Line 78: Is n the same as N ? If so, verify the use of n and N throughout the paper.

RC1 – A17 n will be later on the number of spatial discretization points of the spatial domain Ω . N is the number of data points X_i . We corrected the notation in this formula. As well as in l. 438 in the second paragraph of section 5.4.

RC1 – 18 Line 78 and generally: This is a personal choice, but I feel your paper would be more accessible to the general GMD audience if you change your function notation. Consider

$$f(x; h) = 1/nh \sum K((x-X_j)/h) , \text{ where } f : \mathbb{R} \times \mathbb{R}_{>0} \rightarrow \mathbb{R}_{\geq 0}.$$

RC1 – A18 We changed the function notation in agreement with RC2 to:

$$f(x; h) = 1/nh \sum K((x-X_j)/h) , \text{ where } x \in \mathbb{R}, h \in \mathbb{R}_{>0} \text{ and } f(x; h) \in \mathbb{R}_{\geq 0}$$

Furthermore, we changed Eq. 4

$$\text{MISE}(\hat{f})(h) = E \int_{\mathbb{R}} (\hat{f}(x; t) - f(x))^2 dx, \text{ where } h \in \mathbb{R}_{>0}$$

and Eq. 11 to

$$\Phi_h(x) = \dots , \text{ where } x, \Phi_h(x) \in \mathbb{R}$$

RC1 – 19 Line 86: Should it be $\mathbb{R}_{>0}$ or $\mathbb{R}_{\geq 0}$?

RC1 – A19 It is $\mathbb{R}_{>0}$, because the bandwidth shall always be non-negative.

RC1 – 20 Line 91: It is not clear where equation (6) comes from. It would be helpful if this result, or derivation, is included in the appendix.

RC1 – A20 We changed this part to include a more precise description of the prerequisites necessary for this result as well as a more accessible formulation of it as: “A kernel function K that suffices the additional conditions

$$\int_{\mathbb{R}} K(y) dy = 0, \int_{\mathbb{R}} |K(y)| dy < \infty, \int_{\mathbb{R}} y^2 K(y) dy = k_2 \in \mathbb{R} \setminus \{0\},$$

is a second order kernel as its second moment $\int_{\mathbb{R}} y^2 K(y) dy$ is its first non-zero moment. Those kernels are positive and together with the final condition from Eq. 3 they are PDFs themselves. For the general KDE from Eq. 2 with a second order kernel the optimal bandwidth can be calculated as...” in ll. 141 – 146.

Furthermore, we added in Appendix A a more detailed explanation of the connection between the original idea by Parzen (1962) and our specific application, as well as a source more precisely describing the derivation of this result (Silverman, 1986): “The derivation of the optimal bandwidth choice for a KDE was already described in Parzen

(1962) and can be found in more detail in Silverman (1986). The additional conditions stated in Eq. 7 to the kernel function

$$\int_{\mathbb{R}} K(y) dy = 0 \text{ and } \int_{\mathbb{R}} y^2 K(y) dy = k_2 \in \mathbb{R} \setminus \{0\}$$

correspond to the order of the kernel being equal to 2 (Berliner, 1993). For such kernels Silverman (1986) showed the minimizer of the asymptotic mean integrated squared error to be

$$h = \dots$$

In our context of working with the squared bandwidth $t=h^2$ this optimal bandwidth choice becomes $t = (\|K\|_{L^2}^2 / (N k_2^2 \|f''\|_{L^2}^2))^{2/5}$, which equals Eq. 13.”

RC1 – 21 Lines 98 and 100: see comment 11.

RC1 – A21 We changed the function notations in agreement with RC2 to:

$$K_E(w) = \frac{3}{4} (1 - w^2), \text{ where } w \in \mathbb{R} \text{ and } K_E(w) \in \mathbb{R}_{\geq 0}.$$

and

$$\Phi(w) = \frac{1}{\sqrt{2\pi}} e^{-1/2 w^2}, \text{ where } w \in \mathbb{R} \text{ and } \Phi(w) \in \mathbb{R}_{\geq 0}.$$

RC1 – 22 Line 109: Is the diffusion based approach an alternative method? Better? Also, different approach in what sense? Make this more clear.

RC1 – A22 We expanded this sentence to become an introductory paragraph: “The diffusion-based KDE provides a different approach to Eq. 2 by solving a partial differential equation instead of the summation of kernel functions. This different calculation offers three main advantages: (1) consistency at the boundaries can be ensured by adding Neumann boundary conditions (2) better resolution of multimodal data can be achieved by the inclusion of a suitable parameter function in the differential equation leading to adaptive smoothing intensity (3) a family of KDEs with different bandwidths is produced as a by-product of the numerical solution of the partial differential equation in individual time steps.” in ll. 167 – 172.

RC1 – 23 Line 110: What is “It”? The KDE? Also, “progresses up to an estimate at the final time,” is unclear. What progresses and what estimate?

RC1 – A23 We changed the sentence to: “This KDE solves the partial differential equation describing the diffusion heat process, starting from an initial value based on the input data $(X_j)_{j=1}^N$ and progresses forward in time to a final solution at a fixed time $T \in \mathbb{R}_{>0}$.” in 173 – 175.

RC1 – 24 Line 113: see comment 11.

RC1 – A24 We changed the function notation in agreement with RC2 to:

$$, \text{ where } x \in \mathbb{R}, t \in \mathbb{R}_{>0} \text{ and } \Phi(x; t) \in \mathbb{R}_{\geq 0}$$

RC1 – 25 Line 121: What is the importance of acting inversely proportional to the diffusion quotient? Expand on this point.

RC1 – A25 We added the sentences: “This parameter function allows to influence the intensity of the diffusion applied adaptively depending on the location $x \in R$. Its role and specific choice is discussed in detail in Sec. 4.” in ll. 197 – 198.

RC1 – 26 Line 136: How do you set p properly? It is not clear at this point. (E.g., Line 166 starting with “Choosing p to be ...” could be incorporated here)

RC1 – A26 We changed the sentence to: “Thus, choosing p to be a function allows for a spatially dependent influence on the smoothing intensity, which solves the prior problem of...” in ll. 217 – 219.

RC1 – 27 Line 147: Similar to comment 13, it is not clear where this equation comes from, consider adding its derivation to the appendix. Also, define all the terms in the equation (e.g., ... where $\|\cdot\|_{L2}$ is the L2-norm...).

RC1 – A27 This equation is a direct citation from Botev et al. (2010). We added the sentences: “..., where $\|\cdot\|_{L2}$ is the L2-norm and $\mathbf{E}(\cdot)$ the expected value. The proof of this equation is in detail given in Botev et al. (2010).” in ll. 232 – 233.

RC1 – 28 Line 153: How or why is the additional effort avoided in your approach?

RC1 – A28 We extended the sentence to: “This additional effort is avoided in our approach by directly approximating f with a simple data-based bandwidth approximation in detail described in Sec. 4.” in ll. 242 – 243.

RC1 – 29 Line 168: What is “resolves the unexpected structure”

RC1 – A29 We rephrased the sentence to: “Low smoothing resolves more variability within areas with many similar values (high density), while the intensity of smoothing is increased where data values are more dispersed.”

RC1 – 30 Line 185: The second equal should be an approximate sign. Same with multiple places following. Eq 15, 16, ...

RC1 – A30 We changed the notation in l. 321, Eq. 21, 22, 23, 24, 26, 27, 28 and 30 and rephrased the final sentence of this section to: “By these calculations the solution of the partial differential equation from Eq. 14 can also be approximated by solving the system of ordinary differential equations:” in ll. 339 – 340.

RC1 – 31 Line 201: Bold u ?

RC1 – A31 We changed the font of u to bold.

RC1 – 32 Line 208: Why not an explicit? Does the solver have stability issues? Is this relevant?

RC1 – A32 The explicit Euler is not A-stable, which is the implicit. This is why we chose the implicit approach to guarantee convergence of the KDE. We added the sentence: “The implicit Euler method is chosen at this place, since it is A-stable and by this ensures convergence of the solver.” and rephrased the following to: “Eq. 33 together with the initial value Eq. 16 describes an...” in ll. 351 – 353.

RC1 – 33 Line 220: Define $B_p(0)$.

RC1 – A33 We added: “, where $B_\rho(0) = \{x \in \mathbb{R}; |x - 0| < \rho\} = (-\rho, \rho)$ is the open subset of \mathbb{R} centered around \mathbb{R} with radius ρ .” in l. 363.

RC1 – 34 Line 222: How is $|\Omega|/n = h$ calculated? Moreover, isn't $\Omega = \mathbb{R}$?

RC1 – A34 In general, Ω is a real subset of \mathbb{R} , since we are only calculating over a finite data domain. The spatial stepsize h is the length of the domain Ω divided by the number of spatial discretization points. We rephrased the respective sentence to: “The spatial discretization step size $h \in \mathbb{R}_{>0}$ equals the length of the domain $|\Omega|$ divided by the number of spatial discretization points n , namely $h = |\Omega|/n$. This relationship provides the dependency...” in ll. 365 – 366.

Φ_h shall be defined on \mathbb{R} instead of Ω to make the integration possible (only its support lies in Ω). We corrected the definition of Φ_h in Eq. 36.

RC1 – 35 Line 228: Is $L^1(\mathbb{R})$ defined?

RC1 – A35 We added the definition and rephrased the sentence to: “Then Φ_h is non-negative for all $h \in \mathbb{R}_{>0}$ and as a composition of integrable functions integrable with $\int \Phi_h(x) dx = 1$ (see App. B) and $\Phi_h \in L^1(\mathbb{R}) = \{f : \mathbb{R} \rightarrow \mathbb{R}; f \text{ integrable and } \int |f(x)| dx < \infty\}$.” in ll. 375 – 376.

RC1 – 36 Line 259: Define iqr. A non-statistical audience may not know what this is.

RC1 – A36 We added a definition of iqr as: “The iqr is the interquartile range defined as $\text{iqr}(\text{data}) = q(0.75) - q(0.25)$. The value $q(0.25)$ denotes the lower quartile and describes the value in data, at which 25 % of the elements in data have a value smaller than $q(0.25)$. $q(0.75)$ denotes the upper quartile and describes the analogue value for 75 % (Dekking et al., 2005).” in ll. 297 – 299.

RC1 – 37 Line 271: Is the discretization of Ω the same as Ω ? See line 180. Also, if this is the case, there are multiple instances throughout the paper where this appears.

RC1 – A37 We changed this part to: “...on the discretization $(x_i)_{i=0}^n$ of Ω as...” in l. 303.

Furthermore, we changed a similar instance in l. 433 in the description of the algorithm to be now: “The spatial grid discretizing Ω is setup according...”

RC1 – 38 Line 292: What do you mean by the variable Ω ? Is Ω the spatial domain or something else?

RC1 – A38 We rephrased the sentence to: “This leads to a full set of $n + 1$ equidistant discretization points $(x_i)_{i=0}^n$ saved in a vector variable denoted in Alg. 1 as Ω . The spatial discretization $(x_i)_{i=0}^n$ includes...” in ll. 440 – 441.

RC1 – 39 Table 1: Why is the default number of spatial discretization intervals 1004 and not some value dependent on the dimension of Ω or the concentration and number of observed points?

RC1 – A39 We chose this value to be comparable to the implementation by Hennig (2021) after Botev et. al (2010), where the default value was also set to be $n=1024$ but also possible to be set by the user.

RC1 – 40 Line 309: What is time forward?

RC1 – A40 We changed the sentence to: “The temporal solutions are calculated *timesteps* $\in \mathbb{N}$ times in equidistantly increasing time steps until...” in ll. 459 - 460.

RC1 – 41 Line 324: Why 20 and 10? Any justification?

RC1 – A41 We rephrased this part to: “The times are the 20 timesteps used for the calculation of u as defined in Tab. 1 followed by additional 10 up to the doubled approximated optimal final iteration time $2 T^*$. The time step size for the solutions between T^* and $2 T^*$ are doubled because of the smaller changes in the solution for larger times as for example visible in Fig. 5.” in ll. 475 – 478.

RC1 – 42 Section 2.4: I think this section could use a major rework to make it more readable. Also, there are figures that relate to each of these functions. Why not relate the function to its corresponding figure? For example, “The function call *evol_plot* opens a plot showing the time evolution of the diffKDE (e.g., see Figure 2 for example output). An alternative would be to incorporate this paragraph into section 3.1 and use your simulation example to highlight some of your programs capabilities.

RC1 – A42 We linked the corresponding Figures to the respective function introduction as: “The function call *evol_plot* opens a plot showing the time evolution of the diffKDE (e.g., see Figure 2 for example output).” in l. 489.

and “The function call *pilot_plot* opens that shows the diffKDE together with its pilot estimate p , showing the intensity of local smoothing (e.g., see Fig. 3 for example output).” in ll. 495 – 496.

and “The function call *custom_plot* opens an interactive plot, allowing the user to slide through different approximation stages of the diffKDE (e.g., see Fig. 4 for example output).” in ll. 498 – 499.

Furthermore, we rephrased ll. 490 – 491 to: “In the background the initial values are drawn. The y-axis range is cut off at 20 %...”

and l. 493 to: “The diffKDE is drawn in a bold blue line.”

and l. 493 to: “...provides the user with insight...”

and l. 500 to: “At the bottom of this plot...”

and deleted the part: “, the intensity of smoothing at different localizations.” in ll. 503 – 504.

RC1 – 43 Figure 5: Are you able to include the true curve as a reference?

RC1 – A43 We included the shaded distribution in the background in consistency with the other graphics generated from known distributions.

RC1 – 44 Figure 6: Can you report the MISE or AMISE for all of the curves so there is some numerical reference on how they perform?

RC1 – A44 The insight into the numerical performance of the four KDEs is presented in Figure 9 and Table C1 by the error between the estimates and the true distribution of the first row of Figure 6 for different sample sizes measured by the Wasserstein distance.

We added the MISE (approximate by 100 different random samples of the respective sample sizes) as a second panel to Fig. 9. and expanded the figure caption to: “(a) shows

the error calculated with the Wasserstein distance and (b) with the MISE. The MISE is calculated after Eq. 2 from 100 different random samples.”

Furthermore, we added the MISE values from Fig. 9 to Table C1 and the sentence: “...and the **MISE** defined in Eq. 4. For the approximation of the expected value in Eq. 4 we applied an averaging of the integral value for 100 different random samples for each observed sample size.” in ll. 605 – 606.

RC1 – 45 Also, what about a third column using 1000 points or so to highlight where diffKDE does really well. It would be nice to see how the other methods perform when diffKDE effectively perfectly captures the truth (e.g., are Botev, Gaussian, Epan still way off when diffKDE is nearly perfect).

RC1 – A45 We added a third column to the figure with 1000 data points and moved the integral values to a table to ensure readability of the graphics. We adapted the caption to: “The plots (a), (b) and (c) show KDEs of random samples of the trimodal distribution defined in Eq. 39, (d), (e) and (f) the same for a lognormal distribution. The left figure column is constructed from 50 random samples, the middle from 100 and the right from 1000.”

Furthermore, we adapted the describing part of the Figure in the corresponding section to: “The random samples are 50, 100 and 1000 data points of each distribution...” in ll. 566 – 567.

And added: “In the 1000 random samples test the diffKDE best detects the left mode and the Botev KDE the two others best. Generally, diffKDE and Botev KDE are closely aligned in this case. As well as Gaussian and Epanechnikov KDE are closely aligned, but with a worse fit of all structures of the true distribution.” in ll. 575 – 578.

As well as: “In the 1000 random samples test with the lognormal distribution are again diffKDE and Botev KDE closely aligned as well as Gaussian and Epanechnikov KDE. The first two are very close to the true distribution, but resolve too much structure of the random sample. The diffKDE resolves more structure in the area close to 0 and becomes smoother towards the tail of the distribution. The Botev KDE performs the other way around and provides a smoother estimate close to 0 and more structure of the random sample towards higher data values.” in ll. 580 – 585.

Finally, we rephrased the last sentence of this paragraph to: “An analysis of the integral of the KDEs over the observed domain is presented in Tab. 3 and reveals...” in l. 585.

The table also includes the integral values from Figure 7 for consistency, which are now also removed from the graphics for better readability. We rephrased the last sentence of the respective paragraph to: “The integrals of the KDEs are also presented in Tab. 3 and our implementation is again always exactly 1.” in ll. 594 – 595.

RC1 – 46 Figure 9: I would either connect the dots with a line or have no line at all. I think the best fit line is odd here.

RC1 – A46 We re-did the figure without the regression, but connected the dots and added a grid. We deleted the sentences in the caption of Figure 9 and its description in the respective section that referred to the regression line.

We moved Table 3 to the appendix and referred to the corresponding KDEs in Figure 9 in the table caption as: “Error convergence of the observed KDEs in Fig. 9.”

Technical Corrections

RC1 – 47 General: Consider using simpler, more precise, descriptions of the concepts. There are various cases, some that are included in the following list, where your message gets lost due to word choice. Below is a list of some of the technical and grammatical issues, but not all. I recommend the authors verify the manuscript is void of technical and grammatical issues before re-submission.

RC1 – A47 We added more simple and concise descriptions to numerous parts in all sections of the manuscript as described (see tracked changes document), in addition to the here listed points in RC2 – A1, RC2 – A4, RC2 – A5, RC2 – A6, RC2 – A7, RC2 – A8, RC2 – A9.

RC1 – 48 Line 2: Comma after geoscience.

RC1 – A48 We changed the sentence to: “In geoscience, data ...” in l. 2.

RC1 – 49 Line 5: No need for a comma, “but incomplete because of the...”

RC1 – A49 We changed the sentence to: “Existing KDEs are valuable but problematic because...” in ll. 5 -6.

RC1 – 50 Line 7: This sentence feels odd. Consider rewording. Something like, “To make diffusion-based KDE accessible for general use, we designed and developed...”

RC1 – A50 We changed the sentence to: “In this study, we designed and developed a new implementation of a diffusion-based KDE as an open source Python tool to make diffusion-based KDE accessible for general use. Our new diffusion-based KDE provides...” in ll. 8-10.

RC1 – 51 Line 8: “We demonstrate our tool on simulated and real marine biogeochemical data individually, and compare our results against other popular KDE method”. Also, be clear on if the simulated data is marine biogeochemical data or is not related to marine data.

RC1 – A51 We used artificial data from known distributions as main test cases, since these provide a direct comparison of the KDEs to the true distribution. Furthermore, we used different marine biogeochemical data to provide some real-world application examples. To clarify this, we changed the sentence to: “We demonstrate our tool on artificial data with multiple and boundary close modes and on real marine biogeochemical data, and compare our results against other popular KDE methods.” in ll. 11 – 13.

RC1 – 52 Line 11: This sentence reads awkwardly. Consider breaking it into two pieces. “The convergence ... smaller error. This is most notable for ... ”

RC1 – A52 We broke the sentence down accordingly to: “..., but with a generally smaller error. This is most notable for...” in ll. 17 – 18.

RC1 – 53 Line 12: I don’t understand the use of “exemplify”.

RC1 – A53 We changed the sentence to: “We discuss the general applicability of such KDEs for data-model comparison in geoscience..” in ll. 18 – 19.

RC1 – 54 Line 13: Is this related to the real-world example? If so, perhaps move it to earlier in the paragraph.

RC1 – A54 Yes, this is one of the real-world examples. We shifted the sentence to ll. 13 – 14.

RC1 – 55 Line 15: Comma after geoscience.

RC1 – A55 We changed the sentence to: “In geoscience, the application of...” in l. 24.

RC1 – 56 Line 18: “Such necessity is not only...”

RC1 – A56 We rephrased the sentence accordingly to: “. Accordingly, there is a strong demand for the analysis of model simulations on various temporal and spatial scales and to evaluate these results against observational data.” in ll. 27 – 29.

RC1 – 57 Line 19: “... applications such as social science and financial or ecological research.”

RC1 – A57 See RC1 – A56.

RC1 – 58 Line 21: “... by some distance or divergence measure between...”

RC1 – A58 We changed the sentence accordingly in ll.42 – 43.

RC1 – 59 Lines 33-43: Some of the sentences in this paragraph read awkwardly. For example, the first three feel like they could be combined to read more clearly. Consider reworking this paragraph.

RC1 – A59 We changed the beginning of the paragraph to: “A KDE is based on a kernel function and a smoothing parameter. The kernel function is ideally chosen to be a PDF itself, usually unimodal and centered around zero (Sheather, 2004). The estimation process sums up the kernel function sequentially centered around each data point.” in ll. 67 – 69.

and later on in the paragraph: “If it is larger, more structure becomes smoothed out (Jones et al., 1996), and information from single data points can get lost.” in ll. 72 – 73.

RC1 – 60 Line 45: No comma after “is possible”

RC1 – A60 We changed the sentence to: “This perspective change is possible because the Gaussian ...” in l. 83.

RC1 – 61 Line 53: “... choose between varying levels of smoothness by design.”

RC1 – A61 We changed the sentence accordingly in ll. 94 – 95.

RC1 – 62 Line 55: “... KDE with an accompanying Python package, diffKDE cite or link package.”

RC1 – A62 We changed the sentence to: “In this study, we present a new, modified diffusion-based KDE with an accompanying Python package, diffKDE (Pelz and Slawig, 2023).” in ll. 96 – 97.

RC1 – 63 Line 59: Remove “, so called”

RC1 – A63 We changed the sentence accordingly in l. 101.

RC1 – 64 Line 61: Starting with “Thus, ...” this sentence is confusion, consider rewording.

RC1 – A64 We rephrased the sentence to: “This allows for an interactive investigation of estimated densities at different smoothing intensities.” in ll. 103 – 104.

RC1 – 65 Line 72: You already defined PDFs, can remove “probability density functions”.

RC1 – A65 We changed the sentence accordingly in l. 114.

RC1 – 66 Line 85: “integrated squared error (MISE) is ...”

RC1 – A66 We changed the sentence accordingly in l. 133.

RC1 – 67 Line 88: Define AMISE.

RC1 – A67 We added the definition as: “In the following, we will work with the asymptotic MISE denoted as AMISE, which describes the asymptotic behavior of the MISE for the bandwidth parameter approaching zero $h \rightarrow 0$, meaning $\lim_{t \rightarrow 0} \text{MISE}(f)(t) / \text{AMISE}(f)(t) = 1$.” in ll. 135 – 136.

RC1 – 68 Line 117: “...Chaudhuri and Marron (2000) and its benefits were ...”

RC1 – A68 We connected the sentences as suggested in ll. 193 – 194.

RC1 – 69 Line 119: What does “is” refer to in “... and is extended ...”

RC1 – A69 We rephrased the sentence to: “Our implementation of the diffusion KDE is based on Chaudhuri and Marron (2000), which we extended by some advancements proposed by Botev et al. (2010):” in ll. 195 – 196.

RC1 – 70 Line 123: No comma after (diffKDE)

RC1 – A70 We changed the sentence accordingly in l. 200.

RC1 – 71 Line 128: “When regarded as a PDE, the Dirac δ -distribution puts all of the probability as the corresponding data point.”

RC1 – A71 We changed the sentence accordingly in ll. 208 – 209.

RC1 – 72 Line 131: This sentence feels out of place, consider incorporating it into a paragraph so it is not a stand alone sentence.

RC1 – A72 We moved the sentence to l. 202 directly following the definition of the diffKDE.

RC1 – 73 Line 133: Obliterates does not feel like the correct word.

RC1 – A73 We changed the sentence to: “). This makes the identification of one single optimal bandwidth unnecessary, which is ideal because the optimal value can be specific to a certain application and is often debated (...).” in ll. 213 – 215.

RC1 – 74 Line 144: Both sentences start with “in”.

RC1 – A74 We changed the first sentence to: “We stressed Eq. 13 that...” in l. 228.

RC1 – 75 Line 154: Unclear, consider rewording.

RC1 – A75 We rephrased the sentence to: “The possible difficulties in finding one single optimal bandwidth (e.g. Scott, 2012) do not arise in the calculation of the diffKDE, by default.” in ll. 244 – 246.

RC1 – 76 Line 171: What is Gaussian here?

RC1 – A76 We rephrased the sentence to: “This approach combines Gaussian KDE and diffKDE interchangeably...” in ll. 264 – 265.

RC1 – 77 Lines 178-179: Incorporate into the paragraph after it.

RC1 – A77 We merged the paragraphs as suggested in ll. 311 – 121 (not visibly resolved in the tracked changes document).

RC1 – 78 Line 199: “...division by p is applied column-wise.”

RC1 – A78 We changed the sentence accordingly in ll. 335 – 336.

RC1 – 79 Line 221: Is fineness the right word? Maybe size? Or resolution?

RC1 – A79 We changed the phrase to: “step size” in l. 364.

RC1 – 80 Figure 1: “The function Φ_h depends on the ...”

RC1 – A80 We changed the sentence accordingly in the caption.

RC1 – 81 Line 228: Is this supposed to be a paragraph? Odd indenting.

RC1 – A81 The new paragraph begins with “Now, let ...”. We corrected the indenting accordingly in l. 374.

RC1 – 82 Line 240: Starting with “It is built...”, re-word this sentence.

RC1 – A82 We rephrased the sentence to: “Our new approach solves the diffusion equation in three stages, where the first two provide pilot estimation steps for the diffKDE.” in ll. 268 – 269.

RC1 – 83 Line 466: Chapter?

RC1 – A83 We changed the sentence to: “Since we have already done this in Sec. 6.2, we...” in ll. 644 – 645.

RC1 – 84 Line 474: Define the euphotic zone.

RC1 – A84 We added the following sentences for clarification: “The euphotic zone describes the uppermost ocean layer that is attenuated with light to enable photosynthesis that produces organic matter (Kirk, 2011). While its depth can vary in nature (Urtizberea et al., 2013), here we pragmatically selected included data in the upper 130 m consistent with the analysis in the data set description (Verwege et al., 2021).” in 654 – 657.

Dear Reviewer 2

We greatly appreciate your constructive and helpful comments. We have addressed all your points and feel that our revised version of the manuscript is a significant improvement. Original comments and questions are shown in grey, while our responses are colored black in the text below. All points and our responses are numbered to allow cross-referencing. The line numbers are referring to the manuscript document with the tracked changes below.

In 'A diffusion-based kernel density estimator (diffKDE, version 1) with optimal bandwidth approximation for the analysis of data in geoscience and ecological research,' Pelz et al. introduce a diffusion-based approach to kernel density estimation, as well as its implementation in Python. From what I understand, they treat input data points as delta function sources in a diffusion equation and run the diffusion calculation forward until a given stopping 'time'; they also allow for a user-specifiable diffusivity, that is a function of data space, allowing higher effective resolution in certain areas of the data space. They show that the KDE estimation method performs comparably to other KDE implementations in the Python scipy package, and they show that it tends to have lower mean-squared error for relatively small ($O(100)$) sample sizes.

This paper presents a novel statistical model, with the apparent innovations in the paper being twofold: (1) the use of a spatially-varying diffusivity in a diffusion-based KDE estimator, and (2) implementation of the method in an open-source language. With respect to the journal, GMD, the paper presents a new method for statistical modeling in general, with applicability to geoscience domains; so it seems to be within scope of the journal (though I have further comments on that below). Based on this, the paper seems publishable in principle within this journal. Overall, this seems like a cool technique!

In its current form, the paper has a number of issues that lead me to recommend a major revision:

RC2 – 1 the paper has widespread use of notation that is not well-explained and is not likely to be widely-known by a geoscience audience;

RC2 – A1 In addition to the specifically mentioned points below and those changes made according to RC1, we also added a more detailed description of the general conditions for a KDE as: "... meaning that K is bounded, integrable, and for the limit $y \rightarrow \infty$ decreases faster to zero than y approaches infinity. The final condition means that K integrates to 1 over the whole real domain, which implies that also the KDE \hat{f} integrates to 1 as it is necessary for a PDF." in ll. 123 – 125.

RC2 – 2 the literature review and discussion misses some key, current literature that is highly relevant; and

RC2 – A2 See RC2 – A10 and RC1 – A10 for a detailed response.

RC2 – 3 the current form of the paper appears to advance a statistical method and only briefly applies the technique to geoscientific data, making it unclear whether this paper is really within the scope of GMD. These issues are described in more detail below.

RC2 – A3 We understand the reviewers comment and have therefore decided to include another example, in which we apply the technique to some regional ocean remote sensing (satellite) data of chlorophyll-a concentration. The additional example of the technique is intended for illustrative purposes, but should also serve as inspiration for similar applications. We dedicated a short subsection (*7.3 Performance analyses on remote sensing data*) to this additional example.

Major comments Widespread use of unclear notation

RC2 – 4 One of the biggest issues with this paper is the widespread use of jargon and notation that is not common in the geosciences. For example, the paper makes widespread use of set notation (e.g., line 75, Equation 2), which is not commonly taught in geoscience curricula (I have no idea what the $f: \mathbb{R} \times \mathbb{R} \dots$ notation in Equation 2 means, and I have quite a bit of training in math and statistics, including having published in statistics journals myself!).

RC2 – A4 We changed the function notation to:

$$f(x; h) = 1/nh \sum K((x-X_j)/h), \text{ where } x \in \mathbb{R}, h \in \mathbb{R}_{>0} \text{ and } f(x; h) \in \mathbb{R}_{\geq 0}$$

and changed the following part to include a definition of the involved sets: “The sets $\mathbb{R}_{>0}$ and $\mathbb{R}_{\geq 0}$ denote the positive real numbers and the non-negative real numbers, respectively. The *kernel function* satisfies...”

Furthermore, we changed the function notation in Eq. 4 to

$$\text{MISE}(\hat{f})(h) = E(\int_{\mathbb{R}} (\hat{f}(x; t) - f(x))^2 dx), \text{ where } h \in \mathbb{R}_{>0}$$

and Eq. 9 to

$$K_E(w) = \frac{3}{4} (1 - w^2), \text{ where } w \in \mathbb{R} \text{ and } K_E(w) \in \mathbb{R}_{\geq 0}$$

and Eq. 10 to

$$\Phi(w) = 1/\sqrt{2\pi} e^{-1/2 w^2}, \text{ where } w \in \mathbb{R} \text{ and } \Phi(w) \in \mathbb{R}_{\geq 0}.$$

and Eq. 11 to

$$\hat{f}(x; t) = \frac{1}{nh} \sum_{j=1}^n K\left(\frac{x - X_j}{h}\right), \text{ where } x \in \mathbb{R}, t \in \mathbb{R}_{>0} \text{ and } \hat{f}(x; t) \in \mathbb{R}_{\geq 0}$$

and Eq. 36 to

$$\Phi_h(x) = \dots, \text{ where } x, \Phi_h(x) \in \mathbb{R}$$

RC2 – 5 In another example, $:=$ notation is used (e.g., line 225), and I am not sure what it means here; I’m used to reading it as ‘is distributed as’, but I don’t think that’s what is meant here

RC2 – A5 The “ $a:=b$ ” notation means that “a is defined as b”. We omitted this notation here, since it is already described in words next to the equation.

RC2 – 6 (and what does the $= :$ mean in equation 30?).

RC2 – A6 Also here (and for the neighboring equations), we have omitted the notation and added describing sentences as:

“The nominator is approximated by the unbiased estimator and denoted as $E_{\sigma} \in \mathbb{R} \dots$ ” in ll. 288 – 289.

and: “... and set to $q_i \in \mathbb{R}$ for all $i \in \{1, \dots, n\} \dots$ ” in l. 291.

and: “For the boundary values we set the second derivative at the lower boundary to $q_0 \in \mathbb{R} \dots$ ” in l. 294.

and: “...and the second derivative at the upper boundary to $q_{n+1} \in \mathbb{R} \dots$ ” in l. 296.

RC2 – 7 If notation like this is to be used, I suggest defining what the notation means at first use. My main concern here is that the notation may end up being a barrier to people reading this paper, which would then also inhibit this paper’s potential impact.

RC2 – A7 We checked our manuscript for further use of this notation.

We omitted the notation in the description of the spatial discretization: “...spatial discretization step size $R_{>0} \quad h = x_i - x_{i-1}$ for all $i \in \{1, \dots, n\}$. For the following calculations, we set $x_{-1} = x_0 - h \in \mathbb{R}$ and $x_{n+1} = x_n + h \in \mathbb{R}$.” in ll. 313 – 315.

and: “Now, we set $\frac{1}{2} \frac{1}{h^2} V \frac{1}{p} = A \in \mathbb{R}^{(n+1) \times (n+1)}$, where $A \in \mathbb{R}^{(n+1) \times (n+1)}$ means that A has real entries and $n+1$ rows and $n+1$ columns, the deviation ...” where we also added the clarification of the use of the set $\mathbb{R}^{(n+1) \times (n+1)}$ in ll. 330 – 332.

We omitted the notation in the description of the temporal discretization in ll. 366 – 367.

And in Section 5.4 we omitted the notation in: “The boundary values are $x_{\min} = \min X \in \mathbb{R}$ and $x_{\max} = \max X \in \mathbb{R}$ by default, ...” in ll. 431 – 432.

and changed the introduction of A_{pilot} to: “reduces to a matrix denoted as A_{pilot}

$\frac{1}{2} \frac{1}{h^2} V = A_{\text{pilot}} \in \mathbb{R}^{(n+1) \times (n+1)}$,

where $A_{\text{pilot}} \in \mathbb{R}^{(n+1) \times (n+1)}$ means that A_{pilot} has real entries and $n+1$ rows and $n+1$ columns.” in ll. 450 – 452.

RC2 – 8 Somewhat related to the above, there is ambiguous usage of the symbol t . In some places it is clear that it is meant as the bandwidth (e.g., line 113). In others (e.g., the left-hand side of Equation 10), it seems to mean ‘time’ in the sense of time-evolution of the diffused quantity u . I don’t think that the bandwidth and the time are meant to be taken as being equivalent in this paper, so a symbol other than t really should be used for the bandwidth. (Especially for this audience, where t is almost always reserved to refer to time.)

RC2 – A8 The time parameter of the diffusion equation t can be identified with the squared bandwidth h^2 , which is the motivation for the definition of a diffusion KDE and links both approaches. Any time t in the solution of the diffusion equation corresponds to a squared bandwidth h^2 of a KDE calculated from the classical sum/function definition. The final iteration time in the solution of the diffusion equation T determines a fixed bandwidth at which the diffKDE is evaluated.

We clarified this, by describing in Section 2 only the general form of a KDE with a general bandwidth h and moving the connection of the time parameter to the squared bandwidth to Section 3. For this, we deleted the sentence: “In the following we will exclusively deal with the squared bandwidth (h^2) and therefore adapt a notation where some t is defined as $h^2 =: t \in \mathbb{R}$.” and replaced in Section 2 the variable t with the variable h and deleted the mentions of “squared” in this context.

We also restructured Section 3 to first explain, how the diffusion KDE is the solution of the diffusion equation: “This KDE solves Eq. 11, the partial differential equation describing the diffusion heat process, starting from an initial value based on the input data $(X_j)_{j=1}^N$ and progresses forward in time to a final solution at a fixed time $T \in \mathbb{R}_{>0}$.

$$\partial/\partial t u(x; t) = \frac{1}{2} d^2/dx^2 u(x; t), x \in \Omega, t \in \mathbb{R}_{>0}$$

The input data are treated as the initial value $u(x, 0)$ at the initial time $t_0 = 0$ and generally set to infinitely high peaks at every data point $X_j, j \in \{1, \dots, N\}$. The time propagation in solving Eq. 11 smooths the initial shape of u meaning that u contains less details of the input data $(X_j)_{j=1}^N$ for increasing values in time $t \in \mathbb{R}_{>0}$. If we observe the solution u of Eq. 11 at a specific fixed final iteration time $T \in \mathbb{R}_{>0}$, this parameter determines the smoothness of the function u and how many details of the input data are resolved. This is an equivalent dependency as already seen for the KDE as the solution of Eq. 2 depending on a bandwidth parameter $h \in \mathbb{R}_{>0}$.” in ll. 170 – 179.

and then how this KDE approach is linked to the Gaussian KDE by adding to the following paragraph: “This function solves Eq. 11 as the Green’s function, where the time parameter $t \in \mathbb{R}_{>0}$ equals the squared bandwidth parameter h^2 (Chaudhuri and Marron, 2000). Consequently, we can use the result of the optimal bandwidth from Eq. 13, only as the squared result as

$$T = (\|K\|_{L_2}^2 / N k_2^2 \|f\|_{L_2}^2)^{2/5}$$

where we denote the optimal bandwidth now with $T \in \mathbb{R}_{>0}$ as this is the final iteration time in the solution of Eq. 11.” in ll. 184 – 189.

RC2 – 9 That said, I’m genuinely confused about Algorithm 1, where lines 4 and 5 of the algorithm seem to clearly be setting T_p and T_f as bandwidths, but line 7 seems to be treating t as time and T_p as a maximum time; so maybe bandwidth and time really are the same in this paper? If so, that needs to be stated very clearly and perhaps repeatedly, since that’s quite unintuitive.

RC2 – A9 We changed the reference to T, T_p and T_f in this section to “final iteration time” and deleted the reference: “...derived by the respective bandwidths.”

Furthermore, we changed the reference to T, T_p and T_f in Section 4 to “final iteration time” and rephrased l. 271 to: “We use a simple bandwidth as variants of the rule of thumb by Silverman (1986) to calculate both of them.”

In the end, I’m not certain I understood the method well enough to review it thoroughly, which is a major concern.

RC2 – 10 The introduction does a good job of describing KDE and the background of KDE, giving references to bandwidth choice up through Scott, 2012. However, there are some recent innovations in kernel density estimation that are highly relevant here:

Bernacchia and Pigolotti (2011) derive a method for choosing both the kernel and its bandwidth in an ‘optimal’ way. Note also that there are several geoscience applications of this approach, which can be found by looking at literature citing this paper.

Davies and Baddeley (2017) describe a spatially adaptive bandwidth approach, which seems relevant here given the spatially-adaptive diffusivity used in this paper

Chacón and Duong (2018) overview KDE methods and the variety of bandwidth selection methods used; this is especially relevant since Duong authored the widely used `ks` package in R, which would be a good, additional state-of-the-science package to compare against in Figures 6, 7, 9.

Another relatively recent book overviews KDE methods:

<https://link.springer.com/book/10.1007/978-3-319-71688-6>

I would also add that, for this audience, the introduction would be well-served by listing a number of examples of the use of KDE in geoscience literature.

RC2 – A10 We have been in contact with Alberto Bernacchia after the publication of our preprint at GMD. Their approach differs from ours and reformulates the optimization of the KDE to an optimization of the shape of the kernel function for general bandwidths. Davies and Baddeley (2017) used an adaptive choice of the smoothing parameter depending on the spatial domain and applied an edge correction to achieve consistency at the boundaries, which is not necessary in our approach due to the inclusion of Neumann boundary conditions. We have added this different perspective to our introduction as: “... , to use adaptive bandwidth approaches (Davies and Baddeley, 2017) or to optimize the kernel function shape instead of the bandwidth (Bernacchia and Pigolotti, 2011).” in ll. 75 – 76.

Furthermore, we added Chacón and Duong (2018) and Gramacki (2018) to the reference list for bandwidth selection literature in l. 73 (not visibly resolved in the tracked changes document).

The `ks` package in R uses Gaussian kernels and provides as default the plug-in bandwidth selection, which we included for comparison already as the “GKDE” in the respective figures (calculated from the Python stats package).

Finally, we added the references for KDE applications in geosciences

O’Brien, J. P., O’Brien, T. A., Patricola, C. M., and Wang, S.-Y. S.: Metrics for understanding large-scale controls of multivariate temperature and precipitation variability, *Climate Dynamics*, 53, 3805–3823, <https://doi.org/10.1007/s00382-019-04749-6>, 2019.

Teshome, A. and Zhang, J.: Increase of Extreme Drought over Ethiopia under Climate Warming, *Advances in Meteorology*, 2019, 1–18, <https://doi.org/10.1155/2019/5235429>, 2019.

Ongoma, V., Chen, H., Gao, C., and Sagero, P. O.: Variability of temperature properties over Kenya based on observed and reanalyzed750 datasets, *Theoretical and Applied Climatology*, 133, 1175–1190, <https://doi.org/10.1007/s00704-017-2246-y>, 2017.

to the introduction by: “Well approximated PDFs have been used to benefit data analysis in geosciences (e.g. O’Brien et al., 2019; Teshome and Zhang, 2019; Ongoma et al., 2017).” in ll. 45 – 46.

RC2 – 11 My final concern relates to the scope. In my initial thinking, I considered recommending rejection of this paper because it seems like it might be out of scope for GMD. Most of the content and innovation in the paper relates to a general-purpose statistical method, so it almost seems like this might be more suitable for a statistics journal like JABES.

RC2 – A11 We revised many of the mathematical notations and phrases to make them more accessible (see RC2 – A1, RC2 – A4, RC2 – A5, RC2 – A6, RC2 – A7, RC1 – A12, RC1 – A18, RC1 – A20, RC1 – A27, RC1 – A33, RC1 – A34, RC1 – A35, RC1 – A36, RC1 – A37) and named the benefits of the diffusion KDE for the evaluation of geoscientific data clearly (see RC1 – A22) and repeated them in abstract, introduction and conclusion (see RC2 – A17).

Furthermore, we rephrased the initial paragraph of Section 7.4 to: “In geoscientific research, the derivation and comparison of well resolved PDFs can be useful, as demonstrated in our selected examples. Yet, the significance of resolving details in nonparametric PDFs remains unclear. However, having high resolution PDFs available, as obtained with the diffKDE, is readily of value, and will likely guide further research. An obvious benefit of the diffKDE is its lesser dependence on the specification of a single, albeit optimal, bandwidth. Its application is likely more robust for the assessment of simulation results, either against data or results of other models (e.g. multi model ensembles), which is particularly relevant for evaluations of future climate projections obtained with Earth system models (e.g., Oliver et al., 2022). The presented diffKDE provides a nonparametric approach to estimate PDFs with typical features of geoscientific data. Being able to resolve typical patterns such as multiple or boundary close modes, while being insensitive to noise and individual outliers makes the diffKDE a suitable tool for future work in the calibration and optimization of Earth system models.”

RC2 – 12 I think that part of this impression comes across in the emphasis on the development of the statistical method itself, rather than its application (and/or potential application) in geosciences.

RC2 – A12 In agreement with RC1, we split the results section up into two main sections showing the results on artificial and real geoscientific data individually to highlight the improvements made on each of them (see RC1 – A4).

RC2 – 13 I think this partly relates to the technical notation comment above too; because the notation does not seem typical of a geoscience paper, it kind of feels like it was written for a stats audience.

RC2 – A13 We revised many of the notations to be more accessible (see list provided in RC2 – A11).

RC2 – 14 I spoke with GMD editors about this, and they indicated that they do think it is potentially in scope. But the paper should make a clear case for how this new method advances geoscientific modeling, and I think it might be good to emphasize applications a bit more than the method itself.

RC2 – A14 We originally thought that the examples with the isotopic data and plankton size spectra would be illustrative enough. There are several ways of illustrating the performance of the diffKDE, and the value of estimation of nonparametric probability densities in general. We thought about this and have decided to provide an additional example of the application of the method (described in 7.3 *Performance analyses on remote sensing data*). The example should be simple and clear, as well as an inspiration for future potential applications. Our example shows detailed probability densities that cannot be easily represented by a parametric approach. It addresses temporal changes of spatial patterns within a specified region of interest, which we believe translates well to other applications in the field of geoscience.

RC2 – 15 I also think that the discussion of the method (i.e., section 2) could be revised to be much less rigorous (which would also help with the statistical notation comment); focus on making sure that a large portion of your potential geoscience audience can understand this method (and your innovation to it) rather than focusing on precise mathematical language.

RC2 – A15 We conducted a major revision of Section 2, completely restructuring it (see RC1 – A1, RC1 – A2, RC1 – A3) and simplification and clarification of the statistical details (see RC2 – A11).

RC2 – 16 It also might help to modify the introduction by mentioning some specific uses of KDE in geoscience papers and also modifying the discussion to relate the advances in this paper back to those papers; what might have been different/better about the past papers if they had used your method?

RC2 – A16 We agree and follow the reviewers suggestion. The first paragraphs of the introduction have been revised. We now refer to scientific papers that put emphasis on the use of nonparametric PDFs. However, an evaluation of the quality of the estimated PDFs published in previous studies is not really possible and might not be useful. Even if we could do so, the refined PDF estimates need not automatically alter the scientific inferences documented in these publications. This would go beyond the scope of our method description presented here. The metrics that are applied to quantify the similarity between the PDFs could be more critical, which we also mention.

As far as advances are concerned, there is a clear exception. It is the application of the KDE for the derivation of size spectra, which is the reason we have included such an example. We pick up this topic now already in the introduction, which should make it straightforward for readers to see why we have worked out an example in this regard. See also RC2 – A10.

RC2 – 17 I would also add that the paper should be more explicit about the innovations of this specific paper: in the abstract, the introduction, in the method section, and again in the conclusions.

RC2 – A17 In agreement with RC1, we highlighted the three main advantages of the diffusion KDE in Section 3 as: “This different calculation offers three main advantages: (1) consistency at the boundaries can be ensured by adding Neumann boundary conditions (2) better resolution of multimodal data can be achieved by the inclusion of a suitable parameter function in the differential equation leading to adaptive smoothing intensity (3) a family of KDEs with different bandwidths is produced as a by-product of the numerical solution of the partial differential equation in individual time steps.” in ll. 165 – 169.

We included these benefits in the abstract as: “In this study, we designed and developed a new implementation of a diffusion-based KDE as an open source Python tool to make diffusion-based KDE accessible for general use. Our new diffusion-based KDE provides (1) consistency at the boundaries, (2) better resolution of multimodal data, (3) and a family of KDEs with different smoothing intensities.” in ll. 8 – 11.

and in the introduction: “This different approach comes with three main advantages: (1) consistency at the boundaries (2) better resolution of multimodal data (3) a family of KDEs with different smoothing intensities can be produced as a by-product of the numerical solution.” in ll. 78 – 80.

and in the conclusion as: “We chose this approach to KDE, because it offers three main benefits: (1) consistency at the boundaries (2) better resolution of multimodal data (3) a family of KDEs with different smoothing intensities. We provide our algorithm in an open source Python package.” in ll. 760 – 762.

RC2 – 18 I do see that the innovation of this paper is discussed in lines 544-549, which is great, but the way this is written, it is not clear what specific aspects of this paper are new relative to the citations mentioned.

RC2 – A18 We rephrased this part to: “... Our approach includes a new approximation of the bandwidth, which equals the square root of the final iteration time. We directly approximate the analytical solution of the optimal bandwidth with two pilot estimation steps and finite differences. We calculate the pilot estimates as solutions of a simplified diffusion equation up until final iteration times derived from literature based bandwidths called *rule of thumb* by Silverman (1986). Our new approach results in three subsequent estimations of the PDF, each of them chosen with a finer bandwidth approximation.” in ll. 762 – 768.

Furthermore, we moved the description of our new parts of our method (novelties) to Section 4, becoming a section of its own. And the implementation in Python is described in Section 5.

RC2 – 19 It might help to revise Section 2 such that it is clear which equations are essentially ‘background information’ and which equations (or which parts of the equations) contain the innovation in this paper.

RC2 – A19 In agreement with RC1, we restructured Section 2 completely, now we distinguish between background information with regard to i) the general KDE (Section 2), ii) the diffusion KDE (Section 3), and iii) our new approach (Section 4), and iv) the specific

discretization discretization and implementation of our diffKDE (Section 5) (see RC2 – A15).

A diffusion-based kernel density estimator (diffKDE, version 1) with optimal bandwidth approximation for the analysis of data in geoscience and ecological research

Maria-Theresia Pelz^{1,2}, Markus Schartau², Christopher J. Somes², Vanessa Lampe², and Thomas Slawig¹

¹Department of Computer Science, Kiel University, 24118 Kiel, Germany

²GEOMAR Helmholtz Centre for Ocean Research Kiel, 24105 Kiel, Germany

Correspondence: ts@informatik.uni-kiel.de, mschartau@geomar.de

Abstract. Probability density functions (PDFs) ~~comprise basic~~ provide information about the ~~variability of observed or simulated variables within a system of interest~~ probability of a random variable taking on a specific value. In geoscience, data distributions are often expressed by a parametric estimation of their PDF, such as e.g. a Gaussian distribution. At present there is a growing attention towards the analysis of non-parametric estimation of PDFs, where no prior assumptions about the type of PDF are required. A common tool for such non-parametric estimation is a kernel density estimator (KDE). Existing KDEs are valuable but ~~incomplete, problematic~~ because of the difficulty of objectively specifying optimal bandwidths for the individual kernels. ~~A diffusion-based KDE provides a useful approach to mitigate the difficulty in identifying bandwidths that resolve desired details of multi-modal data while being insensitive to noise. Therefore~~ In this study, we designed and developed a new implementation of a diffusion-based KDE as an open source Python tool. ~~We tested our implementation on artificial and to make~~ diffusion-based KDE accessible for general use. Our new diffusion-based KDE provides (1) consistency at the boundaries, (2) better resolution of multimodal data, (3) and a family of KDEs with different smoothing intensities. We demonstrate our tool on artificial data with multiple and boundary close modes and on real marine biogeochemical data ~~individually and, and compare our results~~ against other popular ~~KDEs~~ KDE methods. We also provide an example for how our approach can be efficiently utilized for the derivation of plankton size spectra in ecological research. Our estimator is able to detect relevant multiple modes and ~~resolve boundary close data while suppressing details induced by noise and individual~~ it resolves modes that are located closely to a boundary of the observed data interval. Furthermore, our approach produces a smooth graph that is robust to noise and outliers. The convergence rate is comparable to the Gaussian estimator, but with a generally smaller error, ~~most notably~~. This is most notable for small data sets with up to around 5000 data points. We ~~exemplify and~~ discuss the general applicability and advantages of such KDEs for data-model comparison in geoscience, ~~in particular for sparse data~~. ~~We also provide an example for how our approach can be efficiently utilized for the derivation of plankton size spectra in ecological research.~~

1 Introduction

~~In geoscience~~

In geoscience, the application of ~~Earth system models (ESMs) numerical models~~ has become an integral part of ~~climate~~
25 ~~research (IPCC, 2022) research~~. Given the complexity of ~~ESMs and the associated manifold of model solutions~~ some models,
such as Earth system models with their descriptions of detailed processes in the ocean, atmosphere, and land, a number of
~~plausible model solutions may exist. Accordingly,~~ there is a strong demand for ~~assessing the agreement of simulation results~~
~~with the analysis of model simulations on various temporal and spatial scales and to evaluate these results against~~ observational
data. ~~In fact, such necessity is not only restricted to simulations with ESM but is transferable to other model applications as well,~~
30 ~~like in social science, in financial or ecological research.~~ A viable evaluation procedure is to compare nonparametric proba-
bility density functions (PDFs) of the data with their simulated counterparts, ~~which may also be quantified.~~ By nonparametric
PDFs, it is meant that no assumptions are made regarding any particular (parametric) probability distribution, such as the
~~normal distribution.~~

Some studies have already documented the advantage of analyzing changes in PDFs, for example, when results of climate
35 models are evaluated on regional scale and their sensitivities to uncertainties in model parameterizations and forcing are
examined (e.g. Dessai et al., 2005; Perkins et al., 2007). Likewise, problems of model parameterizations can be approached
in a stochastic rather than deterministic framework, which requires a simulated probability distribution to compare well
with the probability distribution of truth (Palmer, 2012). Perkins et al. (2007) stressed that the credibility of projecting future
distributions of temperature and precipitation is more likely to be good in cases when the PDFs of hindcast simulation results are
40 similar to the PDFs of the observations. A critical point is the quantification of the similarity between respective nonparametric
PDFs, typically expressed by some distance ~~measure or divergence between respective PDFs (Thorarinsdottir et al., 2013).~~
~~Along with the or divergence measure, as analyzed and discussed in Thorarinsdottir et al. (2013).~~

The examination of the suitability of ~~specific certain~~ divergence functions for data-model assessment ~~as done by Thorarinsdottir et al. (20~~
~~, a necessary prerequisite is the approximation of PDFs based on available data and model results~~ is only one aspect. Another
45 is the quality or representativeness of the estimated PDFs. Well approximated PDFs have been used to benefit data analysis in
geosciences (e.g. O'Brien et al., 2019; Teshome and Zhang, 2019; Ongoma et al., 2017). Obtaining high quality approximations
of nonparametric PDFs is certainly not limited to applications in the geosciences but is likely desirable in other scientific fields
as well. In aquatic ecological research, for example, continuous plankton size spectra can be well derived from PDFs of cell size
measurements sorted by individual species or plankton groups (e.g. Quintana et al., 2008; Schartau et al., 2010; Lampe et al., 2021)
50 . The identification of structural details in the size spectra, such as distinct elevations (modes) and troughs within certain size
ranges, is useful, since they can reveal some of the underlying structure of the plankton foodweb. A typical limitation of the
approach described in Schartau et al. (2010) and Lampe et al. (2021) is the specification of an estimator for the continuous size
spectra, such that all significant details are well resolved.

Mathematically formulated, PDFs are integrable non-negative functions ~~$f: \mathcal{A} \rightarrow [0, \infty]$ from a probability space (Ω, \mathcal{A}, P)~~
55 ~~$f: \Omega \rightarrow [0, \infty)$ from a sample space $\Omega \subset \mathbb{R}$ into the non-negative real numbers.~~ By definition, they allow to directly read the
~~probability with $\int_{-\infty}^{\infty} f(x) dx = 1$.~~ PDFs correspond to the probability P of the occurrence of a data value $X \in \mathbb{R}$ within a

specific range $[a, b] \subseteq \mathbb{R}$ via the relationship

$$P(a < X < b) = \int_a^b f(x) dx \text{ for all } a < b \in \mathbb{R}. \quad (1)$$

The application of kernel density estimators (KDEs) has become a common approach for approximating PDFs in a *non-parametric* way (Parzen, 1962), which means that ~~no~~ probability parameters (~~like e.g.~~ expectation or variance) of the data and ~~no~~ the type of the underlying probability distribution (~~as, e.g.,~~ normal or log-normal) are ~~prescribed or assumed~~ not prescribed. The general concept of KDEs takes into account information of every single data point and treats all of them equally. Consequently, every point's information weighs the same in the resulting estimate ~~;~~ without introducing additional assumptions.

A KDE is based on a kernel function and a smoothing parameter. The kernel function is ideally chosen to be a PDF itself. ~~It is,~~ usually unimodal and centered around zero (Sheather, 2004). ~~In the estimation process,~~ The estimation process sums up the kernel function ~~is~~ sequentially centered around each data point. The sum of these individual kernels is standardized by the number of data points. This ensures that the final estimate is again a PDF by inheriting all properties of its kernels. The smoothing parameter, referred to as bandwidth, determines the smoothness of the estimate. If it is chosen to be small, more details of the underlying data structure become visible. If it is larger, more structure becomes smoothed out (Jones et al., 1996), and information from single data points ~~might can~~ get lost. Hence, it is crucial to determine some kind of an optimal size of the bandwidth parameter to represent a suitable signal-to-noise ratio that allows a separation of significant distinctive features from ambiguous details. The question of optimal bandwidth selection is widely discussed in the literature (~~e.g., Sheather and Jones, 1991; Jones et al., 1996; Heidenreich et al., 2013~~) (e.g., Sheather and Jones, 1991; Jones et al., 1996; Heidenreich et al., 2013). It also takes into account that there might not be one single "optimal" choice for such bandwidth (Abramson, 1982; Terrell and Scott, 1992; Chaudhuri and Marron, 2000; Scott, 2012), to use adaptive bandwidth approaches (Davies and Baddeley, 2017) or to optimize the kernel function shape instead of the bandwidth (Bernacchia and Pigolotti, 2011).

The reformulation of the most common Gaussian KDEs (Sheather, 2004) into a diffusion equation provides a different view on KDE (Chaudhuri and Marron, 2000). This different approach comes with three main advantages: (1) consistency at the boundaries (2) better resolution of multimodal data (3) a family of KDEs with different smoothing intensities can be produced as a by-product of the numerical solution. This perspective change is possible ~~;~~ because the Gaussian kernel function solves the partial differential equation describing the diffusion heat process as the Green function. The time parameter of this differential equation corresponds to the smoothness of the estimate, and thus becomes tantamount to the estimate's bandwidth parameter (Chaudhuri and Marron, 2000). The initial value is typically set to include the δ -distribution of the input data ~~;~~ which will formally be defined in Sec. 3. This differentiates the initial value problem from classical problems, since the δ -distribution is not a proper function itself. In specific applications this diffusion approach delivered convincing results (e.g., Botev et al., 2010; Deniz et al., 2011; Qin and Xiao, 2018) ~~;~~ especially for the resolution of multiple modes (e.g. Majdara and Nooshabadi, 2020). The improved structure resolution has for example already shown useful for the optimization of photovoltaic power generation (Li et al., 2019), analysis of flood frequencies (Santhosh and Srinivas, 2013) or the prediction of wind speed (Xu et al., 2015). However, it tends to resolve too many details or overfit the data in others (e.g., Ma et al., 2019; Chaudhuri and Marron, 2000;

90 Farmer and Jacobs, 2022). One main benefit of the diffusion KDE is that it provides a series of PDF estimates for a sequence of bandwidths by default (Chaudhuri and Marron, 2000). As a consequence, it offers the chance to choose between ~~different grades-varying levels~~ of smoothness by design.

In this study, we present a new, modified diffusion-based KDE ~~for which we provide a Python implementation with an accompanying Python package, diffKDE (Pelz and Slawig, 2023)~~. Our aim is to retain the original idea of diffusion-based KDEs by Chaudhuri and Marron (2000) and Botev et al. (2010), but to avoid the complex fixed-point iteration by Botev et al. (2010). The main objective of our refined approach is to achieve high performance for analyses of high variance and multimodal data sets. Our diffusion-based KDE is based on an iterative approximation that differs from others, using a default optimal bandwidth and two preliminary ~~so-called-pilot~~ estimates. This way the KDE can provide a family of estimates at different bandwidths to choose from in addition to a default solution optimally designed for data from geoscience and ecological research. ~~Thus, This allows for an interactive investigation option-of-these-different-estimates-becomes-possible-in-an-easy-way-of-estimated-densities-at-different-smoothing-intensities.~~

This paper is structured as follows: At first, we will briefly recall the general concept of KDEs. Afterwards, our specific KDE approach will be introduced and described, as developed and implemented in ~~a-our~~ software package. We explain the two pilot estimation steps and the selection of the smoothing parameters. Then the performance of our refined estimator will be compared with other state-of-the-art KDEs, while considering known distributions and real marine biogeochemical data. The real test data include carbon isotope ratios of particulate organic matter ($\delta^{13}C_{POC}$) and plankton size data. Our analyses presented here involve investigations of KDE error, runtime, the sensitivity to data noise and the characteristics of convergence w.r.t. increasing sample size.

2 ~~Theory and methods~~The general kernel density estimator

110 2.1 ~~Kernel density estimation~~

A kernel density estimator (KDE) is a non-parametric statistical tool for the estimation of ~~probability-density-functions (PDFs)~~PDFs. In practice, diverse specifications of KDEs exist that may improve the performance with respect to individual needs. Before we explain our specifications of the diffusion-based KDE, we will provide basic background information about KDEs.

115 For all following let $\Omega \subseteq \mathbb{R}$ be a domain, $X_j \in \Omega$, $j \in \{1, \dots, N\}$, be $N \in \mathbb{N}$ independent identically distributed real random variables.

2.0.1 ~~The general kernel density estimator~~

The most general form of a KDE approximates the true density f of the input data $(X_j)_{j=1}^N$ by

$$\hat{f}: \underline{\mathbb{R} \times \mathbb{R}_{>0}} \rightarrow \underline{\mathbb{R}_{\geq 0}}, (x; h) \mapsto \underline{\frac{1}{nh}} = \underline{\frac{1}{Nh}} \sum_{j=1}^{nN} \underline{K\left(\frac{x - X_j}{h}\right)}, \text{ where } x \in \mathbb{R}, h \in \mathbb{R}_{>0} \text{ and } \hat{f}\left(\underline{x; h}\right) \underline{\mathbb{R}_{\geq 0}}. \quad (2)$$

120 ~~In this formula the~~ The sets $\mathbb{R}_{>0}$ and $\mathbb{R}_{\geq 0}$ denote the positive real numbers and the non-negative real numbers, respectively. The kernel function $K: \mathbb{R} \rightarrow \mathbb{R}_{>0}$ has to satisfy $K: \mathbb{R} \rightarrow \mathbb{R}$ satisfies the following conditions (Parzen, 1962):

$$\sup_{y \in \mathbb{R}} |K(y)| < \infty, \quad \int_{\mathbb{R}} |K(y)| dy < \infty, \quad \lim_{y \rightarrow \infty} |yK(y)| = 0, \quad \int_{\mathbb{R}} K(y) dy = \underline{1.1}, \quad (3)$$

meaning that K is bounded, integrable, and for the limit $y \rightarrow \infty$ decreases faster to zero than y approaches infinity. The final condition means that K integrates to 1 over the whole real domain, which implies that also the KDE \hat{f} integrates to 1 as it is necessary for a PDF.

125 The parameter h determines the smoothness of the estimate calculated by Eq. 2 and is called the *bandwidth parameter* (Silverman, 1986). ~~In the following we will exclusively deal with the squared bandwidth (h^2) and therefore adapt a notation where some t is defined as $h^2 =: t \in \mathbb{R}$.~~ An optimal choice for the bandwidth parameter is regarded as the minimizer of the asymptotic mean integrated squared error between the true density of $(X_j)_{j=1}^N$ and their KDE (Sheather and Jones, 1991). The mean integrated squared error (MISE) is defined as

$$\text{MISE}(\hat{f}): \mathbb{R}_{>0} \rightarrow \mathbb{R}, t \mapsto (\underline{h}) = \mathbf{E} \left(\int_{\mathbb{R}} (\hat{f}(x; \underline{th}) - f(x))^2 dx \right) \underline{dx}, \text{ where } h \in \mathbb{R}_{>0} \quad (4)$$

for all PDFs f and respective KDEs \hat{f} (Scott, 1992). In the following, we will work with the asymptotic MISE denoted as AMISE, which describes the asymptotic behavior of the MISE for the bandwidth parameter approaching zero $h \rightarrow 0$, meaning

$$135 \lim_{h \rightarrow 0} \frac{\text{MISE}(\hat{f})(h)}{\text{AMISE}(\hat{f})(h)} = 1. \quad (5)$$

If now \hat{f} is a KDE and there exists a ~~$t^* \in \mathbb{R}_{>0}$ with~~ $h^* \in \mathbb{R}_{>0}$ with

$$\text{AMISE}(\hat{f})(\underline{th}^*) = \min_{t \in \mathbb{R}_{>0}, h \in \mathbb{R}_{>0}} \text{AMISE}(\hat{f})(\underline{th}), \quad (6)$$

we call ~~t^*~~ h^* the *optimal bandwidth* of \hat{f} by $(X_j)_{j=1}^N$ (Scott, 1992). A kernel function K that suffices the additional conditions

$$140 \int_{\mathbb{R}} yK(y) dy = 0, \quad \int_{\mathbb{R}} y^2 |K(y)| dy < \infty, \quad \int_{\mathbb{R}} y^2 K(y) dy = k_2 \in \mathbb{R} \setminus \{0\}, \quad (7)$$

is a second order kernel as its second moment $\int_{\mathbb{R}} y^2 K(y) dy$ is its first non-zero moment. Those kernels are positive and together with the final condition from Eq. 3 they are PDFs themselves. For the general KDE from ~~this~~ Eq. 2 with a second order kernel, the optimal bandwidth can be calculated ~~according to Parzen (1962) as as~~

$$\underline{th}^* = \left(\frac{f(x) \int K^2(y) dy}{N4 \|f^2\|^2} \frac{\|K\|_{L^2}^2}{Nk_2^2 \|f''\|_{L^2}^2} \right)^{\frac{2}{5} \frac{1}{5}}. \quad (8)$$

145 This result was already obtained by Parzen (1962) and is in more detail discussed in App. A.

~~As we see from Eq. 13~~In Eq. 8, the true density f is involved in the calculation of the optimal bandwidth h^* , which is in turn needed for the approximation of f by a KDE. ~~Thus, and thus prevents~~ a direct derivation of an optimal bandwidth ~~this precluded~~. One possibility of how this implicit relation can be solved is the calculation of pilot estimation steps. Our specific approach to this is shown in Sec. 3.1 and Sec. 3.2.

150 There exists a variety of available choices for the type of kernel function K , which all have their individual benefits and shortcomings. Amongst them are ~~for example, for example~~, the uniform, triangle, or the Epanechnikov kernel (Scott, 1992):

$$K_E: \mathbb{R} \rightarrow \mathbb{R}_{\geq 0}, (w \mapsto) \equiv \frac{3}{4} (1 - w^2), \text{ where } w \in \mathbb{R} \text{ and } K_E(w) \in \mathbb{R}_{\geq 0}. \quad (9)$$

A common choice for K is the Gaussian kernel (Sheather, 2004):

$$\Phi: \mathbb{R} \rightarrow \mathbb{R}_{\geq 0}, (w \mapsto) \equiv \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}w^2}, \text{ where } w \in \mathbb{R} \text{ and } \Phi(w) \in \mathbb{R}_{\geq 0}. \quad (10)$$

155 The standard KDE from Eq. 2 – despite being widely applied and investigated – comes with several disadvantages in practical applications (Khorramdel et al., 2018). For example, severe boundary bias can occur when applied on a compact interval (Marron and Ruppert, 1994). It means that a kernel function with a specified bandwidth, attributed to a single point nearby the boundary, may actually exceed the boundary. Furthermore, it can lack a proper response to variations in the magnitude of the true density f (Breiman et al., 1977). The introduction of a parameter that depends on the respective data region can address
160 the latter (Breiman et al., 1977). Unfortunately, no true independent local bandwidth strategy exists (Terrell and Scott, 1992), meaning that in all local approaches there is still an influence of neighboring data points on each locally chosen bandwidth.

2.0.1 ~~The diffusion-based kernel density estimator~~

3 The diffusion-based kernel density estimator

The diffusion-based KDE provides a different approach to Eq. 2. ~~It solves by solving a partial differential equation instead of~~
165 ~~the summation of kernel functions. This different calculation offers three main advantages:~~ (1) consistency at the boundaries can be ensured by adding Neumann boundary conditions, (2) better resolution of multimodal data can be achieved by the inclusion of a suitable parameter function in the differential equation leading to adaptive smoothing intensity, (3) a family of KDEs with different bandwidths is produced as a by-product of the numerical solution of the partial differential equation in individual time steps.

170 This KDE solves Eq. 11, the partial differential equation describing the diffusion heat process, starting from an initial value based on the input data $(X_j)_{j=1}^N$, ~~that progresses up to an estimate at a final~~ and progresses forward in time to a final solution at a fixed time $T \in \mathbb{R}_{>0}$.

$$\frac{\partial}{\partial t} u(x; t) = \frac{1}{2} \frac{d^2}{dx^2} u(x; t), x \in \Omega, t \in \mathbb{R}_{>0} \quad (11)$$

The input data are treated as the initial value $u(x, 0)$ at the initial time $t_0 = 0$ and generally set to infinitely high peaks at every
175 data point $X_j, j \in \{1, \dots, N\}$. The time propagation in solving Eq. 11 smooths the initial shape of u meaning that u contains

less details of the input data $(X_j)_{j=1}^N$ for increasing values in time $t \in \mathbb{R}_{>0}$. If we observe the solution u of Eq. 11 at a specific fixed final iteration time $T \in \mathbb{R}_{>0}$, this parameter determines the smoothness of the function u and how many details of the input data are resolved. This is an equivalent dependency as already seen for the KDE as the solution of Eq. 2 depending on a bandwidth parameter $h \in \mathbb{R}_{>0}$.

180 An advantageous connection to Eq. 2 is that the widely applied Gaussian kernel is a fundamental solution of this differential equation. Precisely, the Gaussian kernel from Eq. 10 as applied in the construction of a Gaussian KDE depends on the location $x \in \mathbb{R}$ and the smoothing parameter $h \in \mathbb{R}_{>0}$ and has the form $(x; t) \mapsto \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2} \left(\frac{x-X_j}{\sqrt{t}} \right)^2}$ for any $j \in \{1, \dots, N\}$.

$$\Phi(x; t) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2} \left(\frac{x-X_j}{\sqrt{t}} \right)^2}, \text{ where } x \in \mathbb{R}, t \in \mathbb{R}_{>0} \text{ and } \Phi(x; t) \in \mathbb{R}_{\geq 0}. \quad (12)$$

This function solves

$$185 \quad \frac{\partial}{\partial t} u(x; t) = \frac{1}{2} \frac{d^2}{dx^2} u(x; t), x \in \Omega, t \in \mathbb{R}_{>0}$$

Eq. 11 as the Green's function, where the time parameter $t \in \mathbb{R}_{>0}$ equals the squared bandwidth parameter h^2 (Chaudhuri and Marron, 2000). Consequently, we can use the result of the optimal bandwidth from Eq. 8, only as the squared result as

$$T = \left(\frac{\|K\|_{L^2}^2}{N k_2^2 \|f''\|_{L^2}^2} \right)^{\frac{5}{6}}, \quad (13)$$

where we denote the optimal bandwidth now with $T \in \mathbb{R}_{>0}$ as this is the final iteration time in the solution of Eq. 11. This idea to use the diffusion heat equation to calculate a KDE was first proposed by Chaudhuri and Marron (2000). Its and its benefits were widely explored in Botev et al. (2010).

Our implementation of the diffusion KDE is based on Chaudhuri and Marron (2000) and is, which we extended by some advancements proposed by Botev et al. (2010): We included a parameter function $p \in C^2(\Omega, \mathbb{R}_{>0})$ with $\|p''\|_{\infty} < \infty$ into Eq. 11, acting inversely to a diffusion quotient. This parameter function allows to influence the intensity of the diffusion applied adaptively depending on the location $x \in \mathbb{R}$. Its role and specific choice is discussed in detail in Sec. 4. Boundary conditions are set to be Neumann and the initial value being a normalized sum of the δ -distributions centered around the input data points. In the following, we call a function $u \in C^{2,1}(\Omega \times \mathbb{R}_{>0}, \mathbb{R}_{\geq 0})$ the *diffusion kernel density estimator* (diffKDE), if it solves the diffusion partial differential equation

$$\frac{\partial}{\partial t} u(x; t) = \frac{1}{2} \frac{d^2}{dx^2} \left(\frac{u(x; t)}{p(x)} \right), \quad x \in \Omega, t \in \mathbb{R}_{>0}, \quad (14)$$

$$200 \quad \frac{\partial}{\partial x} \left(\frac{u(x; t)}{p(x)} \right) = 0, \quad x \in \partial\Omega, t \in \mathbb{R}_{>0}, \quad (15)$$

$$u(x; 0) = \frac{1}{N} \sum_{j=1}^N \delta(x - X_j), \quad x \in \Omega. \quad (16)$$

The final iteration time $T \in \mathbb{R}_{>0}$ of the solution process of Eq. 14 is called the squared *bandwidth* of the diffKDE. In Eq. 16, the data are incorporated as initial values via the Dirac δ -distribution, i.e., a generalized function which takes the value infinity

at its argument and zero anywhere else. ~~Regarded as PDF, it puts all probability in~~ In general, δ is defined by $\delta(x) = 0$ for all $x \in \mathbb{R} \setminus \{0\}$ and $\int_{-\infty}^{\infty} x \delta(x) = 1$ (Dirac, 1927). When regarded as a PDE, the Dirac δ -distribution puts all of the probability as the corresponding data point. The δ -distribution can be defined exactly as a limit of functions, the so-called Dirac sequence. In actual implementations, it has to be approximated, see Sec. 5.3.

~~The final iteration time $T \in \mathbb{R}_{>0}$ of the solution process of Eq. 14 is called the squared bandwidth of the diffKDE.~~

This specific type of KDE has several advantages. First of all, it naturally provides a sequence of estimates for different smoothing parameters (Chaudhuri and Marron, 2000). This ~~obliterates identifying~~ makes the identification of one single optimal bandwidth ~~whose existence is questioned unnecessary,~~ which is ideal because the optimal value can be specific to a certain application and is often debated (e.g., Abramson, 1982; Terrell and Scott, 1992; Chaudhuri and Marron, 2000; Scott, 2012). ~~Even more~~ Additionally, such a sequence allows a specification of the estimate's smoothness that is most appropriate for the analysis. The parameter function p introduces adaptive smoothing properties (Botev et al., 2010). Thus, ~~setting-choosing p properly to be a function allows for a spatially dependent influence on the smoothing intensity, which~~ solves the prior problem of having to locally adjust the bandwidth to the respective region to prevent oversmoothing of local data structure (Breiman et al., 1977; Terrell and Scott, 1992; Pedretti and Fernández-Garcia, 2013). In contrast to local bandwidth adjustments, local variations of the smoothing intensity can be applied to resolve multimodal data as well as values close to the boundary.

3.0.1 Bandwidth selection

3.1 Bandwidth selection

~~According-~~ We now focus on the selection of the optimal squared bandwidth $T \in \mathbb{R}_{>0}$ according to the relationship $T = h^2$ between final iteration time T of the diffKDE and bandwidth parameter h (Chaudhuri and Marron, 2000), ~~we from now on focus on the selection of the optimal squared bandwidth $T \in \mathbb{R}_{>0}$ and.~~ In the following we refer to this as the *bandwidth selection* for simplicity.

~~In~~ We stressed in Eq. 13 ~~we stressed~~ that the optimal choice of the bandwidth parameter depends on the true density f . In our setup of the diffKDE, the analytical solution for T of Eq. 6 depends not only on the true density f , but also on the parameter function p . It can be calculated as

$$T^* = \left(\frac{\mathbf{E} \left(\sqrt{p(X)} \right)}{2N \sqrt{\pi} \left\| \left(\frac{f}{p} \right)'' \right\|_{L^2}^2} \right)^{\frac{2}{5}}, \quad (17)$$

~~(Botev et al., 2010).~~ The ~~where $\|\cdot\|_{L^2}$ is the L^2 -norm and $\mathbf{E}(\cdot)$ the expected value. The proof of this equation is in detail given~~ in Botev et al. (2010). The role of the parameter function p is in detail described in Sec. 3.2.

In the simplified setup with $p = 1$ as in Eq. 11, the analytical optimal solution of Eq. 13 becomes

$$T_{(p=1)}^* = \left(\frac{1}{2N \sqrt{\pi} \|f''\|_{L^2}^2} \right)^{\frac{2}{5}}. \quad (18)$$

Still, the smoothing parameter depends on the unknown density function f and its derivatives. So we will need to find a suitable approximation of f , which might again be dependent of f and so on. ~~Botev et al. (2010) use p . This implicit dependency can be solved by so-called pilot estimation steps. Pilot estimates are generally rough estimates of f calculated in an initial step to use them for an approximation of Eq. 17 and 18, which later on serve to calculate a more precise estimate of f . A more detailed introduction into pilot estimation and its specific benefit for diffusion-based KDEs is presented in Sec. 3.2.~~

~~Botev et al. (2010) used an iterative scheme to solve this implicit dependency the implicit dependency of the bandwidth parameter on the true distribution f . This additional effort is avoided in our approach by directly approximating f with a simple data-based bandwidth approximation based on two pilot estimation steps in detail described in Sec. 4.~~

The ~~prior claimed possibility of no existence of possible difficulties in finding~~ one single optimal bandwidth ~~for complicated densities (e.g. Scott, 2012) is, by default, no problem for the diffKDE. A solution to this problem is to create (e.g. Scott, 2012) do not arise in the calculation of the diffKDE by default. This problem is solved by creating~~ a family of estimates from different bandwidth parameters (Breiman et al., 1977), ranging from oversmoothed estimates to those with beginning oscillations (Sheather, 2004). For the diffKDE, the progression of the time t up to a final iteration time T is equivalent to the creation of such a family of estimates. ~~For the diffKDE we~~ We thus only need to find a suitable optimal final iteration time T^* . Then, the temporal solution of Eq. 14 provides solutions for the diffKDE for the whole sequence of the temporal discretization time steps smaller than T^* , which we can then use as the requested family of estimates.

3.1.1 Pilot estimation

3.2 Pilot estimation

A crude first estimate of the true density f can serve as a pilot estimation step for several purposes (Abramson, 1982; Sheather, 2004). The most obvious in our case is to obtain an estimate of f for the calculations of the optimal bandwidth in Eq. 17. The second purpose is its usage for the definition of the parameter function p in Eq. 14. Setting this as an estimate of the true density itself introduces locally adaptive smoothing properties (Botev et al., 2010). Since p appears in the denominator in the diffusion equation, it operates conversely to a classical diffusion coefficient. Choosing p to be a function allows for a spatially dependent influence on the smoothing intensity as follows. At points where the function p is small, the smoothing becomes more pronounced, whereas, ~~Whereas~~ if p is larger, the smoothing is less intense. ~~This resolves the expected structure in data dense areas, but expands in sparsely sampled areas~~ Low smoothing resolves more variability within areas with many similar values (high density), while the intensity of smoothing is increased where data values are more dispersed. Eventually, we calculate two pilot estimates – one for p and one for f – to support the calculation of the diffKDE. We set both pilot estimates to be the solution of Eq. 11 with an optimal smoothing parameter approximating Eq. 18. This approach combines Gaussian KDE and diffKDE interchangeably to make best use of both of their benefits (Chung et al., 2018).

3.3 Discretization of diffusion kernel density estimator

4 The new bandwidth approximation and pilot estimation approach of the diffKDE

265 Our new approach solves the diffusion equation in three stages, where the first two provide pilot estimation steps for the diffKDE. The three chosen bandwidths increase in complexity and accuracy over this iteration. This algorithm was implemented in Python 3.

For the optimal final iteration time T^* from Eq. 17, we need the parameter function p as well as the true density f . We approximate them both by a simple KDE, each as pilot estimation steps. We use for both cases the simplified diffKDE defined in Eq. 11, without additional parameter functions. We denote the final iteration times for p and f as $T_p, T_f \in \mathbb{R}_{>0}$, respectively. We use a simple bandwidth as variants of the *rule of thumb* by Silverman (1986) to calculate both of them.

We begin to estimate T_p , which is the final iteration time for the KDE that serves as p . It shall be the smoothest of the three estimates, since p limits the resolution fineness of the diffKDE as a lower boundary. This occurs because the diffKDE converges to this parameter function and hence never resolves less details than p itself (Botev et al., 2010).

275 As seen in Eq. 18, the optimal bandwidth for the approximation of p is depending on the second derivative of f . We therefore need to make some initial assumption about f . For a first simplification, we assume that f belongs to the normal distribution family. Then the variance can be estimated by the standard deviation of the data. This leads us to the parametric approximation of the bandwidth T_p (Silverman, 1986)

$$T_p = \left(\frac{1}{2N\sqrt{\pi}\|f''\|_{L^2}^2} \right)^{\frac{5}{2}} = \left(\frac{1}{2N\sqrt{\pi}\sigma^{-5}\|\Phi''\|_{L^2}^2} \right)^{\frac{5}{2}} = \left(\frac{1}{2N\sqrt{\pi}\sigma^{-5}\frac{3}{8}\frac{1}{\sqrt{\pi}}} \right)^{\frac{5}{2}} = \sigma^2 \left(\frac{4}{3}N \right)^{-\frac{5}{2}}, \quad (19)$$

280 whose estimate is known to be overly smooth on multimodal distributions.

To calculate the final iteration time T_f for the approximation of f in Eq. 17, we choose a refined approximation of Eq. 18 that has been proposed by Silverman (1986) as

$$T_f = \left(0.9 \min \left(\sigma, \frac{iqr(data)}{1.34} \right) \right)^2 N^{-\frac{2}{5}}. \quad (20)$$

285 The *iqr* is the interquartile range defined as $iqr(data) = q(0.75) - q(0.25)$. The value $q(0.25)$ denotes the lower quartile and describes the value in *data*, at which 25 % of the elements in *data* have a value smaller than $q(0.25)$. $q(0.75)$ denotes the upper quartile and describes the analogue value for 75 % (Dekking et al., 2005).

We approximate optimal final iteration time T^* from Eq. 17 by calculating p and f by Eq. 11, based on Eq. 19, and Eq. 20 respectively, on an equidistant spatial grid $(x_i)_{i=0}^n \subset \bar{\Omega}$ of the spatial domain $\Omega \subset \mathbb{R}$. The nominator is approximated by the unbiased estimator and denoted as $E_\sigma \in \mathbb{R}$ (Botev et al., 2010)

$$290 \mathbf{E}(p(X)) \approx \frac{1}{n+1} \sum_{i=0}^{n+1} \sqrt{p(x_i)} = E_\sigma \quad (21)$$

and the second derivative in the denominator by finite differences (McSwiggan et al., 2016) and set to $q_i \in \mathbb{R}$ for all $i \in \{1, \dots, n\}$

$$\left(\frac{f}{p}\right)''(x_i) \approx \frac{1}{h^2} \left(\frac{f}{p}(x_{i+1}) - 2\frac{f}{p}(x_i) + \frac{f}{p}(x_{i-1}) \right) = q_i. \quad (22)$$

For the boundary values we set the second derivative at the lower boundary to $q_0 \in \mathbb{R}$

$$295 \quad \left(\frac{f}{p}\right)''(x_0) \approx \frac{1}{h^2} \left(2\frac{f}{p}(x_1) - 2\frac{f}{p}(x_0) \right) = q_0 \quad (23)$$

and the second derivative at the upper boundary to $q_{n+1} \in \mathbb{R}$

$$\left(\frac{f}{p}\right)''(x_{n+1}) \approx \frac{1}{h^2} \left(2\frac{f}{p}(x_{n-1}) - 2\frac{f}{p}(x_n) \right) = q_{n+1}. \quad (24)$$

We set the finite differences approximation from Eq. 22, Eq. 23 and Eq. 24 as a discrete function with image $\mathbf{q} := (q_0, \dots, q_{n+1})$.

300 In this way we derived an already discrete formula for approximation of the optimal squared bandwidth $T^* \in \mathbb{R}_{>0}$ of the diffKDE on the discretization $(x_i)_{i=0}^n$ of Ω as

$$T^* = \left(\frac{E_\sigma}{2N\sqrt{\pi}\|\mathbf{q}\|_{L^2}^2} \right)^{\frac{2}{5}}. \quad (25)$$

The L^2 -norm is calculated on the discretized versions of f and p by array operations. The integration is performed by the *trapz* function of the SciPy integrate package (Gommers et al., 2022), the square root is part of the math package (Van Rossum, 2020)

~

305 5 Discretization and implementation of the diffKDE

Equation 14 is solved numerically using a spatial and temporal discretization. The discretization is based on finite differences and sparse matrices in Python. A similar approach can be found in a diffusion-based kernel density estimator for linear networks implemented in R by McSwiggan et al. (2016).

5.0.1 **Spatial discretization**

310 5.1 Spatial discretization

We start with the description of the discretization of the spatial domain $\Omega \subseteq \mathbb{R}$. This will reduce the partial differential equation in Eq. 14 into a system of linear ordinary differential equations.

Let $n \in \mathbb{N}$ and $(x_i)_{i=0}^n \subseteq \bar{\Omega}$, an equidistant discretization of Ω with $x_{i-1} < x_i$ and $\mathbb{R}_{>0} \ni h := x_i - x_{i-1}$ **spatial discretization**
step size $\mathbb{R}_{>0} \ni h = x_i - x_{i-1}$ for all $i \in \{1, \dots, n\}$. For the following calculations, we set $x_{-1} := x_0 - h \in \mathbb{R}$ and $x_{n+1} := x_n + h \in \mathbb{R}$.

315 and $x_{n+1} = x_n + h \in \mathbb{R}$. Let u be the solution of the diffKDE and p its parameter function, both as defined in Sec. 3. We assume that u and p are both defined on x_{-1} and x_{n+1} and we set $u_i = u(x_i)$ and $p_i = p(x_i)$ for all $i \in \{-1, \dots, n+1\}$.

Let $t \in \mathbb{R}_{>0}$. We approximate Eq. 15 at $x = x_0$ by applying a first order central difference quotient as

$$0 = \frac{\partial}{\partial x} \left(\frac{u(x_0; t)}{p(x_0)} \right) \approx \frac{1}{2h} \left(\frac{u_1(t)}{p_1} - \frac{u_{-1}(t)}{p_{-1}} \right).$$

This implies

320
$$\frac{u_{-1}(t)}{p_{-1}} = \frac{u_1(t)}{p_1}.$$

We approximate Eq. 14 at $x = x_0$ by applying a second order central difference quotient

$$u'_0(t) \approx \frac{1}{2} \frac{1}{h^2} \left(\frac{u_1(t)}{p_1} - 2 \frac{u_0(t)}{p_0} + \frac{u_{-1}(t)}{p_{-1}} \right) = \frac{1}{2} \frac{1}{h^2} \left(2 \frac{u_1(t)}{p_1} - 2 \frac{u_0(t)}{p_0} \right). \quad (26)$$

Analogously, we approximate Eq. 15 and Eq. 14 at $x = x_n$ again by first and second order central difference quotients, respectively. This gives

325
$$u'_n(t) \approx \frac{1}{2} \frac{1}{h^2} \left(\frac{u_{n+1}(t)}{p_{n+1}} - 2 \frac{u_n(t)}{p_n} + \frac{u_{n-1}(t)}{p_{n-1}} \right) = \frac{1}{2} \frac{1}{h^2} \left(2 \frac{u_{n-1}(t)}{p_{n-1}} - 2 \frac{u_n(t)}{p_n} \right). \quad (27)$$

Finally, we derive from Eq. 14 by applying a second order central difference quotient for all $i \in \{1, \dots, n-1\}$:

$$u'_i(t) \approx \frac{1}{2} \frac{1}{h^2} \left(\frac{u_{i+1}(t)}{p_{i+1}} - 2 \frac{u_i(t)}{p_i} + \frac{u_{i-1}(t)}{p_{i-1}} \right). \quad (28)$$

Now, we identify $\mathbf{p} := (p_0, \dots, p_n) \in \mathbb{R}^{n+1}$, $\mathbf{u}'(t) := (u'_0(t), \dots, u'_n(t)) \in \mathbb{R}^{n+1}$ and $\mathbf{u}(t) := (u_0(t), \dots, u_n(t)) \in \mathbb{R}^{n+1}$ with their spatial discretizations. Furthermore, we define $\mathbf{v}_{\text{upper}} := (2, 1, \dots, 1) \in \mathbb{R}^n$, $\mathbf{v}_{\text{main}} := (-2, \dots, -2) \in \mathbb{R}^{n+1}$ and $\mathbf{v}_{\text{lower}} :=$
 330 $(1, \dots, 1, 2) \in \mathbb{R}^n$ to be the upper, main and lower diagonal of the tridiagonal matrix $\mathbf{V} \in \mathbb{R}^{(n+1) \times (n+1)}$. Now, we set

$$\frac{1}{2} \frac{1}{h^2} \mathbf{V} \frac{1}{\mathbf{p}} =: \mathbf{A} \in \mathbb{R}^{(n+1) \times (n+1)}, \quad (29)$$

where $\mathbf{A} \in \mathbb{R}^{(n+1) \times (n+1)}$ means that \mathbf{A} has real entries and $n+1$ rows and $n+1$ columns, the division by \mathbf{p} is **meant to be applied** column-wise. Then, Eq. 26, Eq. 27 and Eq. 28 can be summarized as a linear system of ordinary differential equations:

335
$$\mathbf{u}'(t) \approx \frac{1}{2} \frac{1}{h^2} \mathbf{V} \frac{\mathbf{u}(t)}{\mathbf{p}} = \mathbf{A} \mathbf{u}(t). \quad (30)$$

By these calculations the solution of the partial differential equation from Eq. 14 **becomes a can also be approximated by solving the** system of ordinary differential equations:

$$\mathbf{u}'(t) = \mathbf{A} \mathbf{u}(t), \quad t \in \mathbb{R}_{>0}. \quad (31)$$

5.1.1 Temporal discretization

340 5.2 Temporal discretization

The time-stepping applied to solve the ordinary differential equation from Eq. 31 and Eq. 16 is again built on equidistant steps forward in time. Let $\Delta \in \mathbb{R}_{>0}$ small and set $t_0 := 0$ and $t_k := t_{k-1} + \Delta$ $t_0 = 0$ and $t_k = t_{k-1} + \Delta$ for all $k \in \mathbb{N}$. Set $u_{k,i} := u(x_i, t_k) \in \mathbb{R}$ $u_{k,i} = u(x_i, t_k) \in \mathbb{R}$ for all $i \in \{0, \dots, n+1\}$ and $k \in \mathbb{N}_0$ and identify $\mathbf{u}_k := (u_{k,i})_{i=0}^n \in \mathbb{R}^{n+1}$ $\mathbf{u}_k = (u_{k,i})_{i=0}^n \in \mathbb{R}^{n+1}$ for all $k \in \mathbb{N}_0$ with their discretizations.

345 We use an implicit Euler method to approximate Eq. 31 for all $k \in \mathbb{N}_0$

$$\mathbf{u}_{k+1} = \Delta \mathbf{A} \mathbf{u}_{k+1} + \mathbf{u}_k \quad (32)$$

from which we obtain

$$\mathbf{u}_k = (\mathbf{I}_{n+1} - \Delta \mathbf{A}) \mathbf{u}_{k+1} \text{ for all } k \in \mathbb{N}_0. \quad (33)$$

~~Together~~ The implicit Euler method is chosen at this place, since it is A-stable and by this ensures convergence of the solver.

350 Eq. 33 together with the initial value Eq. 16 ~~this~~ describes an implementation-ready time stepping procedure. The linear equation for \mathbf{u}_{k+1} will be solved in every time step $k \in \mathbb{N}_0$.

5.2.1 Initial value

5.3 Initial value

The initial value in Eq. 16 depends on the δ -distribution (Dirac, 1927). The δ -distribution is not a proper function, but can be
355 calculated as a limit of a suitable function sequence. A common approximation for the δ -distribution is to use a Dirac sequence (Hirsch and Lacombe, 1999). Such is a sequence $(\Phi_n)_{n \in \mathbb{N}}$ of integrable functions that are non-negative and satisfy

$$\int \Phi_n(x) dx = 1 \text{ for all } n \in \mathbb{N} \quad (34)$$

and

$$\lim_{n \rightarrow \infty} \int_{\mathbb{R} \setminus \mathcal{B}_\rho(0)} \Phi_n dx = 0 \text{ for all } \rho \in \mathbb{R}_{>0}, \quad (35)$$

360 where $\mathcal{B}_\rho(0) = \{x \in \mathbb{R}; |x - 0| < \rho\} = (-\rho, \rho)$ is the open subset of \mathbb{R} centered around \mathbb{R} with radius ρ . For our implementation we define a Dirac sequence $(\Phi_h)_{h \in \mathbb{R}_{>0}}$ depending on the spatial discretization fineness-step size $h \in \mathbb{R}_{>0}$ as an approximation of δ in Eq. 16. The relationship $\frac{|\Omega|}{n} = h$ spatial discretization step size $h \in \mathbb{R}_{>0}$ equals the length of the domain $|\Omega|$ divided by the number of spatial discretization points n , namely $h = \frac{|\Omega|}{n}$. This relationship provides the dependency of Φ_h on $n \in \mathbb{N}$ and the equivalence of the limits $n \rightarrow \infty$ and $h \rightarrow 0$ in this framework. In the following we give the specific definition
365 of our function sequence of choice and proof that this indeed defines a proper Dirac sequence.

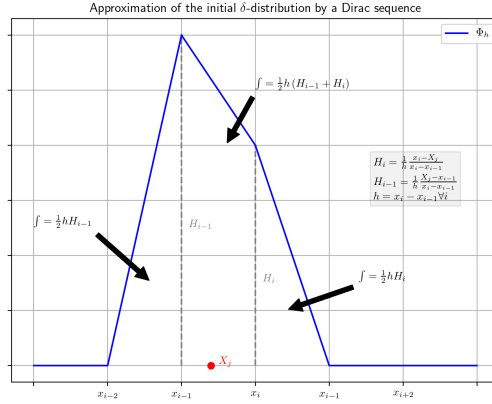


Figure 1. Dirac sequence $(\Phi_h)_{h \in \mathbb{R}_{>0}}$ for the approximation of the δ -distribution in the initial value in Eq. 16. The function Φ_h depends on the spatial discretization fineness h and converges to δ for $h \rightarrow 0$. The function Φ_h is piecewise linear with a peak at each data point $X_j, j \in \{1, \dots, N\}$ integrating to 1.

We assume $0 \in \Omega$. Then there exists an $i \in \{0, \dots, n\}$ with $0 \in [x_{i-1}, x_i]$. If not readily defined, we set $x_{i-2} := x_{i-1} - h \in \mathbb{R}$ and $x_{i+1} := x_i + h \in \mathbb{R}$. We define (see also Fig. 1)

$$\Phi_h: \Omega \rightarrow \mathbb{R}, \left(x \mapsto\right) \approx \begin{cases} \frac{x_i}{h^3}x + \frac{x_i|x_{i-2}|}{h^3}, & x \in [x_{i-2}, x_{i-1}) \\ \frac{x_i+x_{i-1}}{h^3}x + x_i \frac{x_i+x_{i-1}}{h^3} - \frac{x_{i-1}}{h^2}, & x \in [x_{i-1}, x_i) \\ \frac{x_{i-1}}{h^3}x + \frac{x_{i+1}|x_{i-1}|}{h^3}, & x \in [x_i, x_{i+1}] \\ 0 & \text{else.} \end{cases}, \text{ where } x, \Phi_h(x) \in \mathbb{R} \quad (36)$$

370 Dirac sequence $(\Phi_h)_{h \in \mathbb{R}_{>0}}$ for the approximation of the δ -distribution in the initial value in Eq. 16. The function Φ_h is depending on the spatial discretization fineness h and converges to δ for $h \rightarrow 0$. The function Φ_h is piecewise linear with a peak at each data point $X_j, j \in \{1, \dots, N\}$ integrating to 1.

Then $\Phi_h \in L^1(\mathbb{R})$. Then Φ_h is non-negative for all $h \in \mathbb{R}_{>0}$ and as a composition of integrable functions integrable with $\int \Phi_h(x) dx = 1$ (see Appendix App. B) and $\Phi_h \in L^1(\mathbb{R}) = \{f: \mathbb{R} \rightarrow \mathbb{R}; f \text{ integrable and } \int |f(x)| dx < \infty\}$.

Now, let $\rho \in \mathbb{R}_{>0}$ and set $h = \frac{\rho}{2} \in \mathbb{R}_{>0}$. Then we have by Eq. 36

$$375 \int_{\mathbb{R} \setminus \mathcal{B}_\rho(0)} \Phi_h dx = \int_{-\infty}^{\rho} \Phi_h dx + \int_{\rho}^{\infty} \Phi_h dx = \int_{-\infty}^{\rho} 0 dx + \int_{\rho}^{\infty} 0 dx = 0,$$

and it follows

$$\lim_{h \rightarrow 0} \int_{\mathbb{R} \setminus \mathcal{B}_\rho(0)} \Phi_h dx = 0 \text{ for all } \rho \in \mathbb{R}_{>0}. \quad (37)$$

Hence Eq. 36 defines a Dirac sequence. We use Φ_h for the approximation of the δ -distribution in our implementation of Equation 16.

380 The concept of the Dirac sequence also provides the justification to generally rely on the δ -distribution in the construction of the initial value of the diffKDE. The Gaussian kernel defined in Eq. 10 that solves the diffusion equation as a fundamental solution is again a Dirac-sequence (Boccaro, 1990). This link connects the diffKDE directly back to the δ -distribution.

5.4 **Implementation of the diffusion kernel density estimator** The diffKDE algorithm realization in Python

385 ~~The selected implementation is a straight forward approach using equidistant finite differences in space and time and a direct solution of the diffusion equation by an implicit Euler. It is build on the three times sequent solution of the diffusion equation, providing two pilot estimates for the calculation of the final diffKDE. The three chosen bandwidths increase in complexity and accuracy over this iteration. The implementation is realized in Python 3.~~

5.4.1 **Selection of pilot function and optimal bandwidth**

390 ~~For the optimal bandwidth T^* from Eq. 17 we need the parameter function p as well as the true density f . We approximate them both by a simple KDE, each as pilot estimation steps. We use for both cases the simplified diffKDE defined in Eq. 11, without additional parameter functions. We denote the bandwidths for p and f as $T_p, T_f \in \mathbb{R}_{>0}$, respectively. We use a simple bandwidth as variants of the *rule of thumb* by Silverman (1986) for both of them.~~

~~We begin to estimate T_p , which is the bandwidth for the KDE that serves as p . It shall be the smoothest of the three estimates, since p limits the resolution fineness of the diffKDE as a lower boundary. This is, because the diffKDE converges to this parameter function and hence never resolves less details than p itself (Botev et al., 2010).~~

~~As seen in Eq. 18, the optimal bandwidth for the approximation of p is depending on the second derivative of f . We therefore need to make some initial assumption about f . For a first simplification, we assume that f belongs to the normal distribution family. Then the variance can be estimated by the standard deviation of the data. This leads us to the parametric approximation of the bandwidth T_p (Silverman, 1986)~~

$$400 \quad T_p = \left(\frac{1}{2N\sqrt{\pi}\|f''\|_{L^2}^2} \right)^{\frac{2}{5}} = \left(\frac{1}{2N\sqrt{\pi}\sigma^{-5}\|\Phi''\|_{L^2}^2} \right)^{\frac{2}{5}} = \left(\frac{1}{2N\sqrt{\pi}\sigma^{-5}\frac{3}{8}\frac{1}{\sqrt{\pi}}} \right)^{\frac{2}{5}} = \sigma^2 \left(\frac{4}{3}N \right)^{-\frac{2}{5}},$$

~~whose estimate is known to be overly smooth on multimodal distributions.~~

~~To calculate the bandwidth T_f for the approximation of f in Eq. 17 we choose a refined approximation of Eq. 18, which has been proposed by Silverman (1986) as~~

$$T_f = \left(0.9 \min \left(\sigma, \frac{iqr(data)}{1.34} \right) \right)^2 N^{-\frac{2}{5}}.$$

405 We approximate optimal bandwidth T^* from Eq. 17 by calculating p and f by Eq. 11, based on Eq. 19, and Eq. 20 respectively. The nominator is approximated by the unbiased estimator (Botev et al., 2010)

$$\underline{\mathbf{E}(p(X)) = \frac{1}{n+1} \sum_{i=0}^{n+1} \sqrt{p(x_i)} =: E_\sigma}$$

and the second derivative in the denominator by finite differences (McSwiggan et al., 2016)

$$\underline{\left(\frac{f}{p}\right)''(x_i) = \frac{1}{h^2} \left(\frac{f}{p}(x_{i+1}) - 2\frac{f}{p}(x_i) + \frac{f}{p}(x_{i-1})\right) =: q_i}$$

410 for all $i \in \{1, \dots, n\}$. For the boundary values we set

$$\underline{\left(\frac{f}{p}\right)''(x_0) = \frac{1}{h^2} \left(2\frac{f}{p}(x_1) - 2\frac{f}{p}(x_0)\right) =: q_0}$$

and

$$\underline{\left(\frac{f}{p}\right)''(x_{n+1}) = \frac{1}{h^2} \left(2\frac{f}{p}(x_{n-1}) - 2\frac{f}{p}(x_n)\right) =: q_{n+1}}.$$

We set the finite differences approximation from Eq. 22, Eq. 23 and Eq. 24 as a discrete function with image $\mathbf{q} := (q_0, \dots, q_{n+1})$.

415 In this way we derived an already discrete formula for approximation of the optimal squared bandwidth $T^* \in \mathbb{R}_{>0}$ of the diffKDE on the discretization Ω as

$$\underline{T^* = \left(\frac{E_\sigma}{2N\sqrt{\pi}\|\mathbf{q}\|_{L^2}^2}\right)^{\frac{2}{5}}.$$

The L^2 -norm is calculated on the discretized versions of f and p by array operations. The integration is performed by the *trapz* function of the SciPy integrate package (Gommers et al., 2022), the square root is part of the math package (Van Rossum, 2020)

420 :-

5.4.1 The diffKDE algorithm with optimized bandwidth

The implementation is realized implementation was conducted in Python and its concept shown in Alg. 1. We use-used the Python libraries Numpy (Harris et al., 2020) and SciPy (Virtanen et al., 2020; Gommers et al., 2022) and the Python Math module (Van Rossum, 2020) for data preprocessing, calculation of the bandwidths, setup of the differential equations and their

425 solution. The algorithm iteratively calculates three KDEs: first the two for the approximations of p and f as the pilot estimation steps described in Sec. 4 and the last one being u the solution of the diffKDE built on the two prior. All three KDEs are calculated by solving the diffusion equation up to the respective final iteration time. The solution is realized in *while*-loops solving Eq. 33. The two pilot estimation steps can be calculated simultaneously, since they are independent of each other and only differ in their final iteration times T_p and T_f . All input variables are displayed in Tab. 1 the return values listed in Tab. 2.

Algorithm 1 Finite differences based algorithm for the implementation of the diffusion KDE.

Note: the routine solve(\mathbf{M}, \mathbf{b}) means that the system $\mathbf{M}\mathbf{x} = \mathbf{b}$ is solved.

Require: $\mathbf{X} \in \mathbb{R}^N$, $n \in \mathbb{N}$, *timesteps* $\in \mathbb{N}$, $x_{min} \in \mathbb{R}$, $x_{max} \in \mathbb{R}$

```

1:  $h \leftarrow (x_{max} - x_{min}) / (n - 4)$ 
2:  $\Omega \leftarrow (x_{min} - 2h, x_{min} - h, \dots, x_{max} + h, x_{max} + 2h) \in \mathbb{R}^{n+1}$ 
3:  $\mathbf{p}, \mathbf{f}, \mathbf{u} \leftarrow \Phi_h$ 
4:  $T_p \leftarrow \sigma^2 \left(\frac{4}{3}N\right)^{-\frac{2}{5}}$ 
5:  $T_f \leftarrow \left(0.9 \min\left(\sigma, \frac{igr(data)}{1.34}\right)\right)^2 N^{-\frac{2}{5}}$ 
6:  $t \leftarrow 0$ ,  $\Delta_p \leftarrow T_p / \textit{timesteps}$ ,  $\Delta_f \leftarrow T_f / \textit{timesteps}$ 
7: while  $t < T_p$  do
8:    $\mathbf{p} \leftarrow \text{solve}(\mathbf{I}_{n+1} - \Delta_p \mathbf{A}_{pilot}, \mathbf{p})$ 
9:    $\mathbf{f} \leftarrow \text{solve}(\mathbf{I}_{n+1} - \Delta_f \mathbf{A}_{pilot}, \mathbf{f})$ 
10:   $t \leftarrow t + \Delta_p$ 
11: end while
12:  $\mathbf{q} \leftarrow \sqrt{\int_{\Omega} \left(\left(\frac{\mathbf{f}}{\mathbf{p}}\right)''\right)^2 dh}$ 
13:  $E_{\sigma} \leftarrow \frac{1}{n+1} \sum_{i=0}^{n+1} \sqrt{p(x_i)}$ 
14:  $T \leftarrow \left(\frac{E_{\sigma}}{2N\sqrt{\pi q^2}}\right)^{\frac{2}{5}}$ 
15:  $t \leftarrow 0$ ,  $\Delta \leftarrow T / \textit{timesteps}$ 
16: while  $t < T$  do
17:   $\mathbf{u} \leftarrow \text{solve}(\mathbf{I}_{n+1} - \Delta \mathbf{A}, \mathbf{u})$ 
18:   $t \leftarrow t + \Delta$ 
19: end while
20: return  $\mathbf{u}, \Omega, \Phi_h, \mathbf{p}, \textit{stages}, \textit{times}$ 

```

430 The spatial grid discretizing Ω is setup according to the description in Sec. 5.1 in lines 1 and 2 of Alg. 1. It consists of $n \in \mathbb{N}$ intervals, where n can be set by the user. The boundary values are $x_{min} := \min X \in \mathbb{R}$ and $x_{max} := \max X \in \mathbb{R}$ $x_{min} = \min X \in \mathbb{R}$ and $x_{max} = \max X \in \mathbb{R}$ by default, but can also be chosen individually. Setting the boundary values to an individually chosen interval in the function call results in a clipping of the used data to this smaller one before KDE calculation. Outside the interval boundaries, the diffKDE adds two additional discretization points to make it applicable for the case of a

435 data point $X_i, i \in \{0, \dots, n+1\}$ $X_j, j \in \{1, \dots, N\}$ being directly located at one of the boundaries. This way it is possible to construct the initial value defined in Eq. 36, which takes into account the two neighbouring discretization points in each direction. This leads to a full set of $n+1$ equidistant discretization points ~~saved in the variable Ω~~ $(x_i)_{i=0}^n$ saved in a vector variable denoted in Alg. 1 as Ω . The spatial discretization Ω $(x_i)_{i=0}^n$ includes an inner discretization between the handed in (or default set) interval endpoints x_{min} and x_{max} of $n-4$ equally sized inner discretization intervals.

440 The Dirac sequence Φ_h for the implementation of the initial value is defined in Eq. 36 and we use the same for initialization of all three approximations of the the PDF (p, f, u) in line 3 of Alg. 1. In its calculation, the algorithm searches for each

Table 1. Input variables: The only required input variable for the calculation of the diffKDE is a one dimensional data set as an array like type. All other variables are optional, with some prescribed defaults. On demand the user can set individual lower and upper bounds for their data evaluation under the diffKDE as well as the number of used spatial and temporal discretization intervals. The individual selection of the final iteration time provides the opportunity to choose the specific smoothing grade on demand.

index	name	type	default	description
0	<i>data</i>	array like	required	input data $X \in \mathbb{R}$
1	<i>xmin</i>	float	$\min X$	lower data boundary for KDE calculation
2	<i>xmax</i>	float	$\max X$	upper boundary for KDE calculation
3	<i>n</i>	integer	1004	number of spatial discretization intervals in Ω
4	<i>timesteps</i>	integer	20	number of temporal discretization intervals
5	<i>T</i>	float	T^*	final iteration time for diffKDE

$j \in \{1, \dots, N\}$ for the $i \in \{1, \dots, n+1\}$ with x_i being the closest right neighbour of X_j . Then the initial value is constructed by assigning the values $\frac{1}{h} \frac{x_i - X_j}{x_i - x_{i-1}}$ and $\frac{1}{h} \frac{X_j - x_{i-1}}{x_i - x_{i-1}}$ at grid point x_i and x_{i-1} , respectively, and zero elsewhere. These values are corresponding to the weighed heights H_i and H_{i-1} displayed in Fig. 1. The final initial value is the normalized sum of all these individual approximations of the δ -distribution. All three used KDEs (p, f, u) are initialized with this initial value.

In the pilot estimation steps, we calculate the KDEs for p and for f required for the set up of the bandwidth-final iteration time T^* for the diffKDE. The bandwidths-final iteration times T_p and T_f for p and f , respectively, are calculated based on the input data X in lines 4 and 5 of Alg. 1 as described in Sec. 4. Then, the KDEs are calculated by solving a linear ordinary differential equation by an implicit Euler in the first *while*-loop in lines 7 to 9 of Alg. 1. For the pilot estimation steps calculating p and f the matrix \mathbf{A} defined in Eq. 29 does not incorporate a parameter function and reduces to a matrix denoted as \mathbf{A}_{pilot} .

$$\frac{1}{2} \frac{1}{h^2} \mathbf{V} = \mathbf{A}_{pilot} \in \mathbb{R}^{(n+1) \times (n+1)}, \quad (38)$$

where $\mathbf{A}_{pilot} \in \mathbb{R}^{(n+1) \times (n+1)}$ means that \mathbf{A}_{pilot} has real entries and $n+1$ rows and $n+1$ columns. Apart from this, the solutions for the pilot KDEs are the same as for the final diffKDE. The two pilot KDEs can be solved simultaneously, since they share their matrix \mathbf{A}_{pilot} and have independent pre-computed bandwidths-final iteration times. The difference in their bandwidths is implemented in different time step sizes Δ_p and Δ_f for p and f , respectively, which are initialized in line 6 of Alg. 1 directly before this first *while*-loop. The time-forward is-temporal solutions are calculated $timesteps \in \mathbb{N}$ times in equidistant-equidistantly increasing time steps until each individual final iteration time derived-by-the-respective-bandwidths. Since we solve implicitly, there is no restriction to the time step size. But a larger *timesteps* parameter reduces the numerical error proportional to the step size parameters Δ_p and Δ_f . In this temporal solution we rely on the fact that the involved matrices are sparsely covered. The applied solver is part of the SciPy Python library and designed for efficient solution of linear systems including sparse matrices (Virtanen et al., 2020; Gommers et al., 2022).

Table 2. Return values of the diffKDE: The return variable of the diffKDE is a vector. Its first entry is the diffKDE evaluated on the spatial grid. Its second entry is the spatial grid Ω .

index	name	type	size	description
0	u	Numpy array	$n + 1$	diffKDE values on Ω
1	Ω	Numpy array	$n + 1$	spatial discretization

The final ~~bandwidth-iteration time~~ T for the diffKDE solution u is calculated after the calculations of p and f , using them both as described in Sec. 4 in lines 12 to 14 of Alg. 1. For the diffKDE u , the differential equation is given in Eq. 31 and the solution ~~approach-by-uses~~ the implicit Euler ~~approach~~ in Eq. 33. This is implemented in a second *while*-loop described in lines
465 16 to 18 in Alg. 1 ~~and-apart-, which is separated~~ from the final iteration time T^* and the matrix \mathbf{A} identical to the calculations in the pilot step.

The return value is a vector providing the user the diffKDE, along with some main parameters and the opportunity to also evaluate different approximation stages. It provides in the first and second entry ~~of~~ the diffKDE and the spatial discretization Ω . The third entry is the initial value Φ_δ and the fourth pilot estimate p that influences the adaptive smoothing. The last ~~return~~
470 ~~values-two-vectors-are-handed-back:-values-to-return-are-the-two-vectors~~ $stages$ and $times$. These include the approximation stages of the diffKDE and the respective times exceeding the default optimal solution stored in the diffKDE ~~and-providing~~
~~also-some-oversmoothed-solution-, providing some oversmoothed solutions~~ for individual evaluations. The times are the 20 timesteps used for the calculation of u ~~and-10-additional-with-doubled-stepsize-reaching-as-defined-in-Tab.1-followed-by~~
~~additional-10~~ up to the doubled approximated optimal final iteration time $2T^*$. ~~The time step size for the solutions between T^*~~
475 ~~and T^* are doubled because of the smaller changes in the solution for larger times as for example visible in Fig. 5.~~

Possible problems are caught in *assert* and *if* clauses. ~~First-of-all~~ ~~Initially~~, the data is reshaped to a Numpy array for the case of a list handed in and it is made sure that this is non-empty. For the case of numerical issues leading to a pilot estimate including zero values, the whole pilot is set back equal to 1 to ensure numerical convergence. ~~Similar-is-done~~ ~~Accordingly~~ for the case of NaN value being delivered for the optimal bandwidth for the diffKDE, in which case this is also set to the bandwidth
480 chosen for f in Eq. 20.

5.5 Pre-implemented ~~functions for~~ visual outputs

Besides the standard use to calculate a diffKDE at an approximated optimal final iteration time for direct usage, we also included three possibilities to generate a direct visual output, one of them being interactive. Matplotlib (Hunter, 2007) provides the software measures for creating the plots. Most methods are part of the submodule Pyplot, ~~whereas~~ the interactive plot is
485 based on the submodule Slider.

The function call *evol_plot* opens a plot showing the time evolution of the diffKDE ~~-(e.g., see Fig. 2 for example output).~~ The plot includes drawings of the data points on the x-axis. In the background the initial values are drawn, ~~but.~~ The y-axis range is cut off at 20 % above the global maximum of the diffKDE to preserve focus of the graphic on the diffKDE and evolution. The evolution is presented by drawings of the individual time evolution stages using the sequential color map Viridis. ~~In the front~~
490 ~~the~~ The diffKDE is drawn in a bold blue line. This visualization of the evolution provides the user with insight into the data distribution and their respective influence on the final form of the diffKDE.

The function call *pilot_plot* opens that shows the diffKDE together with its pilot estimate p , showing the intensity of local smoothing ~~-(e.g., see Fig. 3 for example output).~~ With this the user has the possibility to gain insight to the influence of this pilot estimator on the performance of the diffKDE. This plot also includes the data points on the x-axis.

495 The function call *custom_plot* opens an interactive plot, allowing the user to slide through different approximation stages of the diffKDE ~~-(e.g., see Fig. 4 for example output).~~ This feature is based on the Slider module from the Matplotlib library (Hunter, 2007) and opens a plot showing the diffKDE. ~~On~~ At the bottom of this plot is a scale that shows the time, initially being set to the optimal iteration time derived from Eq. 17 in the middle of the scale. By clicking to the scale, the user can display the evolution stages at the respective (closest) iteration time. This reaches down to the initial value and up to the doubled
500 optimal iteration time. This interactive tool provides the user a simple tool to follow the estimate at different bandwidths, ~~the intensity of smoothing at different localizations.~~ With the help of such plot it is possible to decide on whether the diffKDE is desired to be applied with a final iteration time that is different from the default.

6 Results ~~and Discussion~~ on artificial data

In the following we document the performance of the diffKDE on artificial and real marine biogeochemical data. ~~Different data sources are chosen to best show possibilities and performance of the diffKDE. Additionally, snapshots of the pre-implemented plot routines are given as examples.~~ Whenever not stated otherwise, we used the default values of the input variables stated in Tab. 2 in the calculation of the diffKDE.

For testing our implementation against a known true PDF, we first constructed a three-modal distribution. The objective is to assess the diffKDE's resolution and to exemplify the pre-implemented plot routines. The distribution was constructed from
510 three Gaussian kernels centered around $\mu_1 = 3$, $\mu_2 = 6.5$ and $\mu_3 = 9$ and with variances $\sigma_1^2 = 1$, $\sigma_2^2 = 0.7^2$ and $\sigma_3^2 = 0.5^2$, each of them with a relative contribution of 30 %, 60 % and 10 %, respectively:

$$f(x) = 0.3 \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}(x-3)^2} + 0.6 \frac{1}{0.7\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-6.5}{0.7}\right)^2} + 0.1 \frac{1}{0.5\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-9}{0.5}\right)^2}. \quad (39)$$

~~The performance of the diffKDE is then illustrated with real data of a) measurements of carbon isotopes (Verwega et al., 2021; Verwega et al. and b) of plankton size (equivalent spherical diameter) (Lampe et al., 2021). We chose these data because we propose to~~
515 ~~apply the diffKDE for the analysis of field data for assessment and optimization of marine biogeochemical as well as size-based ecosystem models. The carbon isotope data have been collected to constrain model parameter values of a marine biogeochemical model that incorporates this tracer as a prognostic variable (Schmittner and Somes, 2016).~~

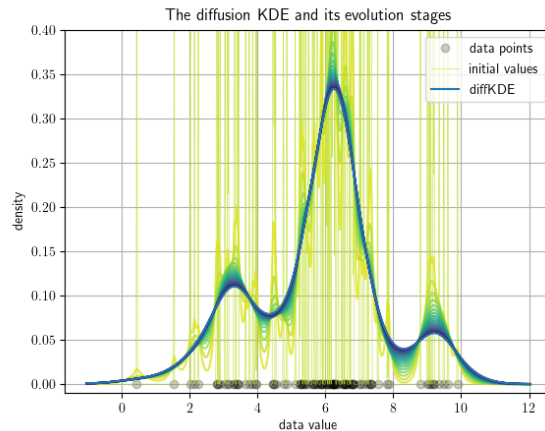


Figure 2. Pre-implemented direct visual output of the evolution process of the diffKDE. The input data are 100 samples randomly collected from Eq. 39. The individual data points are drawn on the x-axis. The y-axis represents the estimated probability density. The light yellow vertical lines in the background are the initial value of the the diffKDE. The temporal evolution of the solution of Eq. 33 is visualized by the sequent color scheme from light yellow over green to the bold blue graph in the front. The final diffKDE at the approximated optimal final iteration time represents as this graph the end of the time evolution.

6.1 Pre-implemented outputs

As described in Sec. 5.5, we included three plot functions in the diffKDE implementation. All of them open pre-implemented
520 plots, to give an impression of the special features that come with the diffKDE. An overview of the three possible direct visual outputs of the diffKDE software is described below.

First we [outline the possibility demonstrate how](#) to display the diffKDE's evolution. By calling the `evol_plot` function, a plot opens that shows all temporal evolution stages of the solution of Eq. 33. The temporal progress is visualized by a sequential colorscheme progressing from light yellow over different shades of green to dark blue. On the x-axis, all used data points are
525 drawn and in the background a cut-off part of the initial value in light yellow as the beginning of the temporal evolution. The final diffKDE is plotted as a bold blue line in front of the evolution process. This gives the user an insight in the distribution of the initial data and their influence on the shape of the estimate. As an example of the default setting, we created an evolution plot from 100 random samples of Eq. 39 visualized in Fig. 2. The second example shows the possibility of displaying the diffKDE together with the pilot estimate p by the function `pilot_plot`. This is the parameter function in Eq. 16 responsible for
530 the adaptive smoothing. Where this function is larger, the smoothing is less intense and allows more structure in the estimate of the diffKDE. Contrarily where it is smaller, the smoothing becomes more pronounced and data gaps are better smoothed out. The result of the diffKDE is shown together with its parameter function p in figure Fig. 3 on the same random sample of the distribution from Eq. 39 as before.

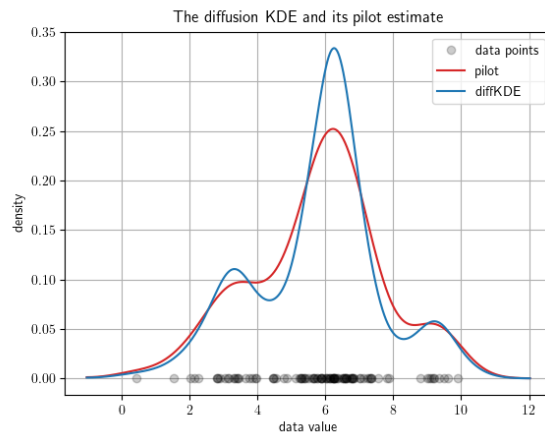


Figure 3. The diffKDE and its pilot estimate p . The input data are 100 samples randomly collected from Eq. 39. The data points are drawn on the x-axis. The y-axis represents the estimated density of the diffKDE in blue and the pilot estimate in red.

Lastly, we illustrate example snapshots of the interactive option to investigate different smoothing stages of the diffKDE by the function. We chose simpler and smaller example data for this demonstration, because these are better suited for visualization of this tool's possibilities. The function `custom_plot` opens an interactive graphic, starting with a plot of the approximated optimal default solution of the diffKDE at T^* . In this graphic the user is able to individually choose, by a slider, the iteration time at which the desired approximation stage of the diffKDE can be seen. The time can be chosen from 0, where the initial value is shown, up until the doubled approximated optimal time ($2 \times T^*$). A reset button sets the graphic back to its initial stage of the diffKDE at T^* . Four snapshots of this interactive experience are drawn in Fig. 4.

6.2 Performance analyses on known distributions and in comparison to other KDEs

In this section we present results obtained by random samples of the trimodal distribution from Eq. 39 and lognormal distributions with differing parameters. Wherever suitable, the results are compared to other commonly used KDEs. These include the most common Gaussian KDE with the kernel function from Eq. 10 (Gommers et al., 2022), the Epanechnikov KDE with the kernel function from Eq. 9 (Pedregosa et al., 2012) and an improved implementation of the Gaussian KDE by Botev et al. (2010) in a Python implementation by Hennig (2021). We begin with an example of how the user may choose individually different smoothing grades of the diffKDE, then compare the different KDEs with the true distribution, followed by investigating the influence of noise on different KDEs, and finally show the convergence of different KDEs to the true distribution with increasing sample size.

We start with an individual selection of the approximation stages. This is one of the main benefits of the diffKDE compared to standard KDEs by providing naturally a family of approximations. This family can be observed by the function `custom_plot`. Individual members can be produced by setting the bandwidth parameter T in the function call of the diffKDE. This gives the

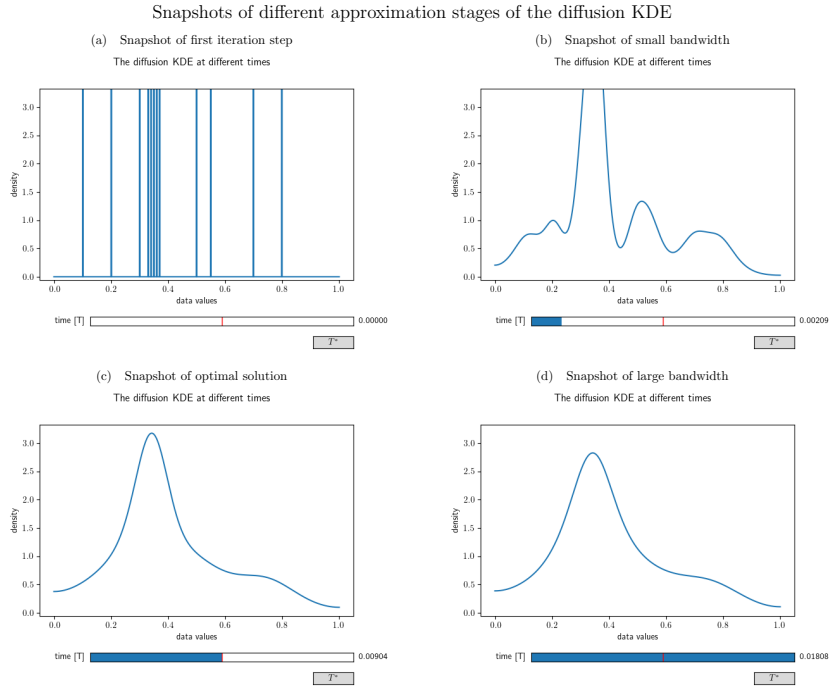


Figure 4. Different snapshots from the interactive visualization of the diffusion KDE generated from the artificial data set $(0.1, 0.2, 0.3, 0.33, 0.34, 0.35, 0.36, 0.37, 0.5, 0.55, 0.7, 0.8)$. (a) shows the output at $time = 0$ and hence the initial value. (b) shows an intermediate smoothing stage of the diffKDE. (c) shows the diffKDE of the input data at the approximated optimal iteration time T^* . This is the initial stage of the interactive graphic. By clicking the button on the lower right, the graphic can be reset to this stage. (d) shows an oversmoothed version of the diffKDE at the doubled approximated optimal iteration time.

user the chance to choose among more and less smooth approximations. A selection of such approximations along with the default solution are shown in Fig. 5 on a random sample of 50 data points from the trimodal distribution in Eq. 39. The plot
555 shows how smaller iteration times resolve more structure in the estimate, while a substantially larger iteration time has only little influence on the increased smoothing of the diffKDE.

~~From now on~~ In the following we only work with the default solution of the diffKDE at T^* . We start with comparisons of the diffKDE and the three other popular KDEs directly to the underlying true distribution. The three other KDEs are the Gaussian KDE in an implementation from SciPy (Gommers et al., 2022), the Epanechnikov KDE in an implementation from
560 Scikit-learn (Pedregosa et al., 2012) and an improved Gaussian KDE by Botev et al. (2010) in a Python implementation by Hennig (2021).

We use differently sized random samples of the known distribution from Eq. 39 and the standard lognormal distribution both over $[-1, 12]$, for a direct comparison of the accuracy of the KDEs. The random samples are 50 ~~and~~, 100 ~~and~~ 1000 data points of each distribution and all four KDEs are calculated and plotted together in Fig. 6. The underlying true distribution is plotted

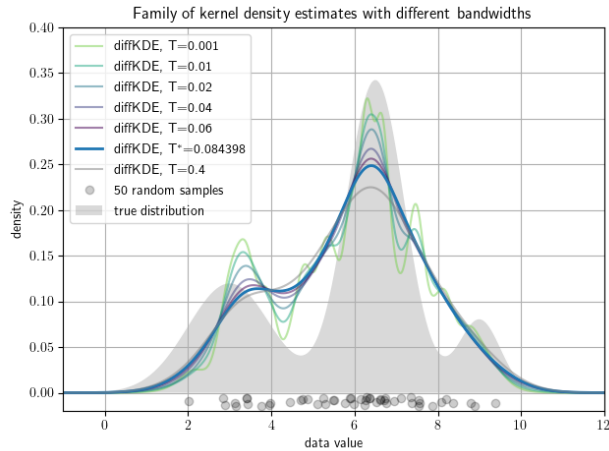


Figure 5. Family of diffKDEs evaluated at different bandwidths: A data set of 50 random samples drawn as grey circles on the x-axis serve to show the possibility to investigate a whole family of estimates by the diffKDE. The bold blue line represents the default solution of the diffKDE by solving the diffusion equation up to the approximated optimal final iteration time T^* . The other colors depict more detailed prior approximation stages with smaller bandwidth, i.e. earlier iteration times, and a smoother estimate with a far larger iteration time.

565 in the background to visually assess the approximation accuracy. In general, the diffKDE resolves more of the details of the structure of the true distribution, while not being too sensitive to patterns introduced by the selection of the random sample and individual outliers. For the 50 random samples test of the trimodal distribution, all KDEs do not detect the third mode and only the diffKDE and the Epanechnikov KDE detect the second. The magnitude of the main mode is also best resolved by these two. In the 100 random samples test of the trimodal distribution, the diffKDE and the Botev KDE are able to detect

570 all three modes. The main mode is best resolved by the diffKDE, whereas the third mode best by the Botev KDE. In both test cases for the trimodal distribution, the Gaussian KDE is the smoothest and the Epanechnikov KDE provides the least smooth graph. ~~For 50 as well as for 100 random samples drawn from the lognormal distribution the magnitude and the~~ In the 1000

random samples test the diffKDE best detects the left mode and the Botev KDE the two others best. Generally, diffKDE and Botev KDE are closely aligned in this case. The Gaussian and Epanechnikov KDEs are also closely aligned, but with a worse

575 fit of all structures of the true distribution. The steep decline to 0 is best reproduced by the diffKDE particularly with low random sample sizes. The Gaussian KDE always performs the worst. The Botev KDE is generally also close to the diffKDE, but resolves in the tail of the distribution too much influence of individual outliers. In the 1000 random samples test with the lognormal distribution are again diffKDE and Botev KDE closely aligned as well as Gaussian and Epanechnikov KDE. The first two are very close to the true distribution, but resolve too much structure of the random sample. The diffKDE resolves

580 more structure in the area close to 0 and becomes smoother towards the tail of the distribution. The Botev KDE performs the other way around and provides a smoother estimate close to 0 and more structure of the random sample towards higher data values. An analysis of the ~~the~~ integral of the KDEs over the observed domain is presented in Tab. 3 and reveals that the diffKDE is the only one that integrates to 1 in all test cases.

Compared KDE performance on known distributions

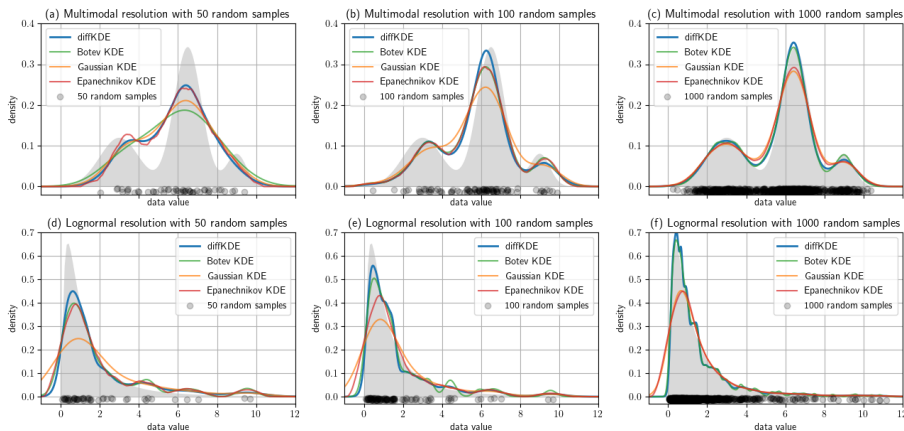


Figure 6. Test cases with known distributions: The plots (a) and (b) and (c) show KDEs of random samples of the trimodal distribution defined in Eq. 39, (d), (e) and (f) the same for a lognormal distribution. The left figure column is constructed from 50 random samples, the right-middle from 100 and the right from 1000. In all plots the true distribution is drawn in grey in the background and the random data sample as grey dots on the x-axis. Each subfigure shows four KDEs: the diffKDE, the Botev KDE, the Gaussian KDE and the Epanechnikov KDE. In the labels of the KDEs are also the integrals over the interval $[-1, 12]$ given for each of the KDEs

Table 3. Integrals of the KDEs displayed in Fig. 6 and Fig. 7.

Graphic	diffKDE	Botev KDE	Gaussian KDE	Epanechnikov KDE
Figure 6 (a)	1.0	0.999984	0.999984	0.999998
Figure 6 (b)	1.0	0.999999	0.999677	1.0
Figure 6 (c)	1.0	1.0	0.999989	1.0
Figure 6 (d)	1.0	0.999955	0.961448	0.999999
Figure 6 (e)	1.0	1.0	0.987128	0.999998
Figure 6 (f)	1.0	0.996	0.995571	0.996372
Figure 7 (a)	1.0	0.9986	0.894	0.9163
Figure 7 (b)	1.0	0.9094	0.802	0.8968
Figure 7 (c)	1.0	0.9999	0.9986	0.9617
Figure 7 (d)	1.0	1.0	0.9983	0.996
Figure 7 (e)	1.0	0.7703	0.0914	0.6171
Figure 7 (f)	1.0	0.6825	0.0337	0.5718

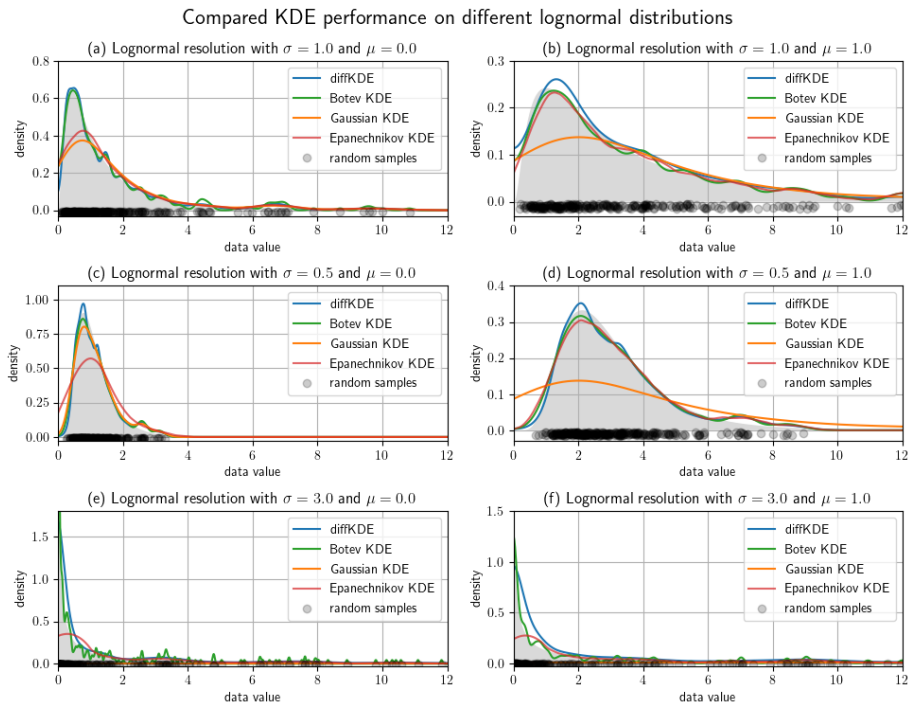


Figure 7. Lognormal test cases with different mean and variance parameters. Of each distribution 300 random samples were taken and the diffKDE, the Botev KDE, the Gaussian KDE and the Epanechnikov KDE calculated and plotted together with the true distribution. The random data sample is drawn as gray circles on the x-axis. (a) and (b) use $\sigma = 1$, (c) and (d) $\sigma = 0.5$ and (e) and (f) $\sigma = 3$ in their underlying normal distributions. The means are in the left column $\mu = 0$, the right column $\mu = 1$ of the underlying normal distribution.

We refined the test cases from Fig. 6 by investigating a lognormal distribution with different parameters and a restriction to the interval $[0, 12]$ in Fig. 7. We varied mean and variances of the normal distribution and used two different means and three different variances resulting in six test cases. All of them are run with 300 random samples and again with all four KDEs. The larger the variance becomes, the more structure of individual data points is resolved by the Botev KDE. The Gaussian KDE fails for increasing variance too, resulting in intense oversmoothing. The Epanechnikov KDE performs well for smaller variances and larger means, but also oversmooths in the other cases. The diffKDE is generally one of the closest to the true distribution, while not resolving too much of the structure introduced by the choice of the random sample, especially for increased variances. But this too tends to resolve too much structure in the vicinity of the mode for smaller variances. The [integral of integrals of the KDEs are also presented in Tab. 3 and](#) our implementation is again always exactly 1.

Now, we show the performance of the diffKDE on increasingly large data sets. We still use the trimodal distribution from Eq. 39. We start with four larger random data samples ranging from 100 to 10 million data points of the trimodal distribution and then being restricted to our core area of interest $[-1, 12]$. We calculate the diffKDE from all of them as well as the respective runtime on a consumer laptop from 2020. We compare the results again to the true distribution in Fig. 8. All of the estimates

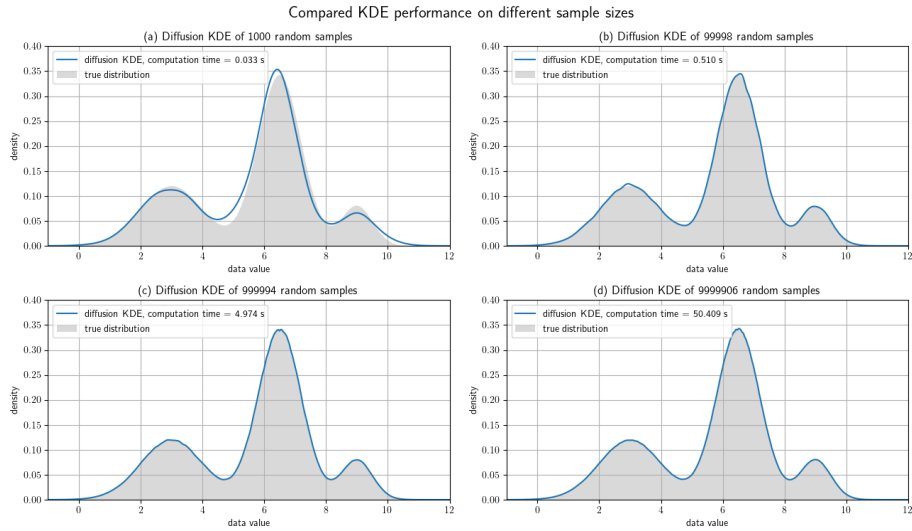


Figure 8. Test cases with different sample sizes: All four plots show the diffKDE of random samples of the known trimodal distribution defined in Eq. 39. (a) is calculated from a subsample of 100 data points, (b) 100,000, (c) 1,000,000 and (d) 10,000,000, all cut to the interval $[-1, 12]$ and hence lacking a few data points. The true numbers of incorporated data points in the four test cases are given in the respective sub-headings. The measured computing time on a 2020 MacBook Air is also drawn in the respective label.

could be calculated in less than one minute. For 100 data points there is still an offset to the true distribution visible in the estimate. For the larger data samples the estimate only shows some minor uneven areas, which smooth out until the largest test case.

600 Furthermore, we investigated the convergence of the diffKDE to the true distribution, again in comparison to the three other KDEs. The error between the respective KDE and the true distribution is calculated by the Wasserstein distance (Panaretos and Zemel, 2019) with $p = 1$ by a SciPy function `wasserstein_1` and the MISE defined in Eq. 4. For the approximation of the expected value in Eq. 4 we applied an averaging of the integral value for 100 different random samples for each observed sample size. We used increasingly large random samples from the trimodal distribution starting with 10 and reaching up to 1 million. The errors
605 calculated for each of the KDEs on each of the random samples are listed in Tab. C1. The values from Tab. C1 are visualized in Fig. 9 on a log-scale and with a linear regression for each KDE's error values. The diffKDE, the Gaussian and the Botev show a similar steep decline, while the Epanechnikov KDE far slower decreases its error with increased sample size. The diffKDE and the Botev KDE generally show similar error values, the diffKDE relatively smaller ones on smaller data samples, the Botev KDE relatively smaller ones on data samples larger than around 5000.

610 **Error convergence sample size** $error_{diffKDE}$ $error_{BKDE}$ $error_{GKDE}$ $error_{EKDE}$ 10 0.02354 0.03618 0.02662 0.0273
50 0.01813 0.02484 0.02182 0.02017 100 0.00422 0.00724 0.01371 0.00702 150 0.00664 0.00937 0.01526 0.00933 200
0.00787 0.00894 0.01522 0.00967 300 0.0053 0.00621 0.01385 0.00849 400 0.00368 0.00484 0.01147 0.0081 500 0.0027
0.00324 0.01057 0.00761 750 0.00361 0.00343 0.00999 0.00807 1000 0.00321 0.00238 0.00933 0.00785 2000 0.00235

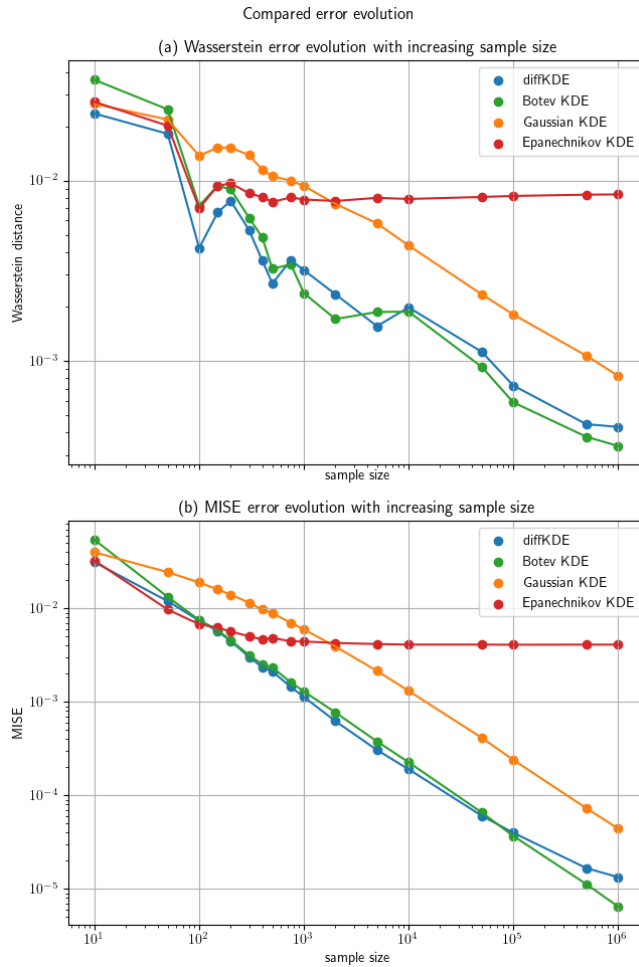


Figure 9. The evolution of the errors of the diffKDE, the Gaussian KDE, the Epanechnikov KDE and the Botev KDE are drawn on log-scale against the increasing sample size on the x-axis. The (a) shows the error has been calculated with the Wasserstein distance and (b) with the log-scale MISE. The MISE is constructed calculated after Eq. 4 from the discrete values of the individual errors for all four KDEs 100 different random samples.

0.00171 0.00743 0.00771 5000 0.00154 0.00187 0.00578 0.00802 10000 0.00199 0.00188 0.00437 0.00791 50000 0.00113
 615 0.00093 0.00234 0.00811 100000 0.00074 0.00059 0.00181 0.00822 500000 0.00048 0.00038 0.00108 0.00835 1000000
 0.00046 0.00034 0.00084 0.00838

Finally, we investigated the noise sensitivity of the diffKDE compared to the three other KDEs on data containing artificially introduced noise. We again used the trimodal distribution from Eq. 39 and 1000 random samples. From this, we created noised data $X_\theta \in \mathbb{R}^N$ by

620 $(X_\theta)_i = (X)_i + (-1)^\tau \text{rand} 10^{-2}\theta\sigma$ for all $i \in \{1, \dots, 1000\}$, (40)

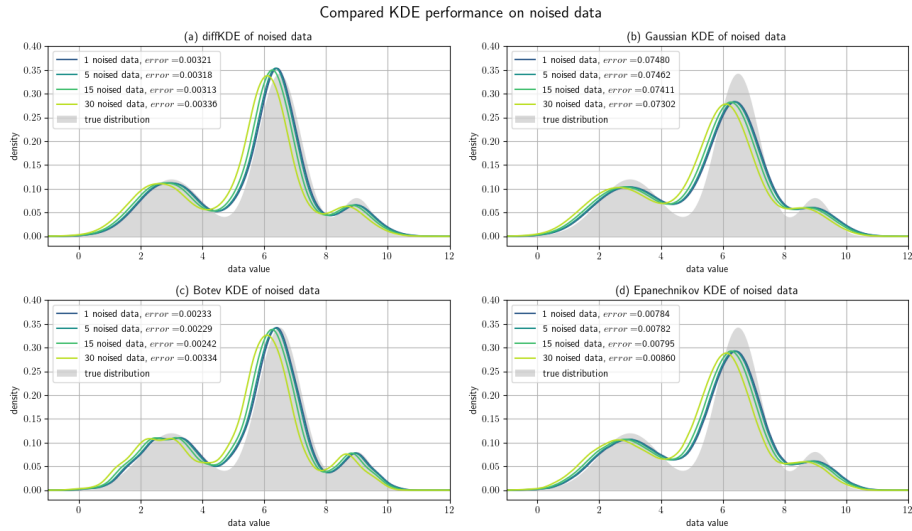


Figure 10. Noised data experiments: A random sample of 1000 data points of the trimodal distribution is artificially noised by differing amounts of the standard deviation. (a) shows the resulting diffKDEs of the differently noised data, (b) the Gaussian KDE, (c) the Botev KDE and (d) the Epanechnikov KDE. In all four panels the original true distribution is drawn in grey in the background. The values of the error between the KDEs and the original true distribution are also part of the respective labels.

where $\theta \in \{0, 1, 5, 15, 30\}$ defines the percentage of noise with respect to the standard deviation $\sigma \in \mathbb{R}$. $\tau \in \{1, 2\}$ was chosen randomly as well as $rand \in [0, 1]$. The error is again expressed by the Wasserstein distance between the original probability density and the respective KDE. The results are visualized in Fig. 10 with an individual panel for each KDE. The error of the Epanechnikov KDE is overall the largest and also increases to the largest. The Gaussian KDE produces the second largest error, but this even decreases with increased noise. The Botev KDE produces the smallest errors, but for increased noise this increases and approaches the magnitude of the one from the diffKDE. The error of the diffKDE only minimally responds to increased noise in the data. Visually, all four KDEs follow a similar pattern of a shift to the left of the graph. The Botev KDE additionally resolves more structure of the noised data as the noise increases.

6.3 Performance analyses on biogeochemical data

7 Results on marine biogeochemical data and outlook to model calibration

In this final part, we show the diffKDE's performance on real marine biogeochemical field data. The performance of the diffKDE is now illustrated with real data of a) measurements of carbon isotopes (Verwega et al., 2021; Verwega et al., 2021), b) of plankton size (equivalent spherical diameter) (Lampe et al., 2021) and c) remote sensing data (Sathyendranath et al., 2019, 2021). We chose two example data: A set of $\delta^{13}\text{C}$ in particulate organic carbon (POC) (Verwega et al., 2021) data and a set of plankton size spectra data (Lampe et al., 2021). These data because we propose to apply the diffKDE for the analysis of field data for assessment and optimization of marine biogeochemical- as well as size-based ecosystem models. The carbon isotope

data have been collected to constrain model parameter values of a marine biogeochemical model that incorporates this tracer as a prognostic variable (Schmittner and Somes, 2016).

Both data sets were already analyzed using KDEs in their original publications (Verwega et al., 2021; Lampe et al., 2021).
640 Here we expand these analyses by a comparison of the KDEs used in the respective publications to the new implementation of the diffKDE. For the $\delta^{13}\text{C}_{\text{POC}}$ data, the Gaussian KDE was the one used in the data description publication. Since we have already done this in ~~the previous chapter~~Sec. 6.2, we furthermore added the Epanechnikov and the Botev KDE to these graphics. For the plankton size spectra data, we only compared the diffKDE to the two Gaussian KDEs used in the respective publication to preserve the clarity of the resulting figures.

645 7.1 Performance analyses on organic carbon-13 isotope data

The $\delta^{13}\text{C}_{\text{POC}}$ data (Verwega et al., 2021) was collected to serve for direct data analyses as well as for future model assessments (Verwega et al., 2021). We show here the Gaussian KDE as it was used in the data publication in a direct comparison to the diffKDE. Furthermore, we added the Epanechnikov and the Botev KDE. Since in this case no true known PDF is available, we have to compare the four estimates and subjectively judge their usefulness. In Fig. 11 we show the KDEs on four different
650 subsets of the $\delta^{13}\text{C}_{\text{POC}}$ data: a) the full data set, b) a restriction to the core data interval of $[-35, -15]$, where 98.65 % of the data is located, and then even further restricted to c) the euphotic zone and d) only data sampled in the 1990s. The euphotic zone describes the upper ocean layer with sufficient light to enable photosynthesis that produces organic matter (Kirk, 2011). While its depth can vary in nature (Urtizberea et al., 2013), here we pragmatically selected included data in the upper 130 m consistent with the analysis in the data set description (Verwega et al. 2021). In all three cases that involve
655 deep ocean measurements, the Botev KDE produces strong oscillations while the Gaussian KDE strongly smoothes the dip between the modes at around $\delta^{13}\text{C}_{\text{POC}} = -24$ and $\delta^{13}\text{C}_{\text{POC}} = -22$ and mostly the one between around $\delta^{13}\text{C}_{\text{POC}} = -28$ and $\delta^{13}\text{C}_{\text{POC}} = -24$. The Epanechnikov KDE resolves more structure than the Gaussian, but still less pronounced than the diffKDE. Especially in the full data analysis, the diffKDE reveals the most structure while not resolving smaller data features of individual data points. The KDEs from the euphotic zone data are all reasonably smooth. The Gaussian KDE is again the smoothest
660 and missing the mode at $\delta^{13}\text{C}_{\text{POC}} = -22$ completely. The other three KDEs resolve a similar amount of data structure. The Botev KDE reveals a better distinction between the modes at around $\delta^{13}\text{C}_{\text{POC}} = -24$ and $\delta^{12}\text{C}_{\text{POC}} = -22$ while the diffKDE shows the first one more pronounced. These observations are consistent with those from the experiments from Fig. 7 and Fig. Figure 10, where especially the Gaussian and the Botev KDE struggle with the resolution of data with increasing variances or noise. From the four here observed $\delta^{12}\text{C}_{\text{POC}}$ data sets the euphotic zone data shown in panel (c) in Fig. 11 has with 7.78 the
665 smallest standard deviation. The other shown data has variances 13.91, 10.96 and 9.61 for panels (a), (b) and (d), respectively.

7.2 Performance analyses on plankton size spectra data

Another example demonstrates the performance of the diffKDE if applied to plankton size data (Lampe et al., 2021). The data of size, abundance of protist plankton was originally collected for resolving changes in plankton community size-structure, providing complementary insight for investigations of plankton dynamics and organic matter flux (e.g., Nöthig et al., 2015).

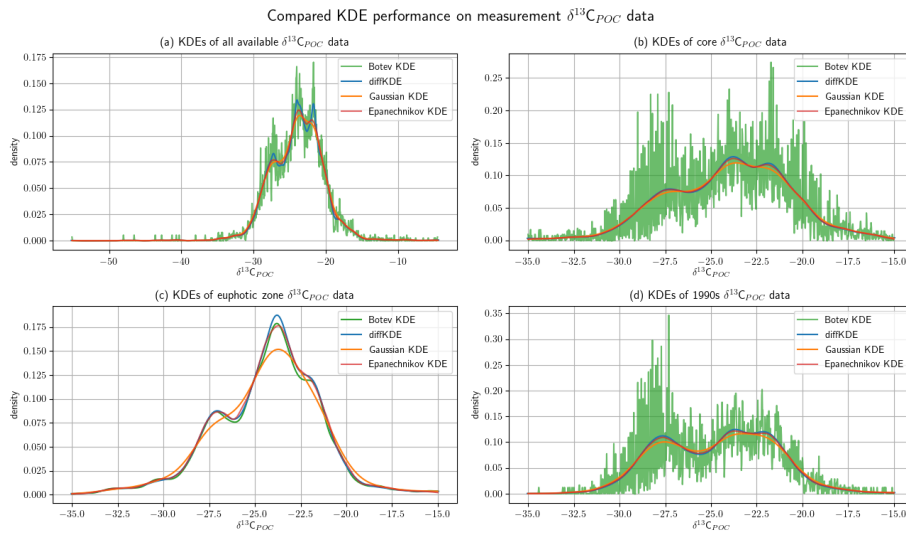


Figure 11. Comparison of KDE performance on marine biogeochemical field data: The $\delta^{13}\text{C}_{\text{POC}}$ data (Verwege et al., 2021) is in detail described in Verwege et al. (2021) and is covering all major world oceans, the 1960s to 2010s and reaches down into the deep ocean. In all four panels the diffKDE is plotted together with the Gaussian, the Epanechnikov and the Botev KDE. (a) Shows KDEs from all available data, (b) shows the KDEs of the data restricted to the core data values of $[-35, -15]$, (c) shows the KDEs from only euphotic zone data with values in $[-35, -15]$ and (d) the KDEs from all 1990s data with values in $[-35, -15]$.

670 In the study of Lampe et al. (2021) a KDE was applied for the derivation of continuous size spectra of phytoplankton and microzooplankton that can potentially be used for the calibration and assessment of size-based plankton ecosystem models. In their study they used a Gaussian KDE, as proposed in Schartau et al. (2010), but with two different approaches for generating plankton size spectra. Uncertainties, also with respect to optimal bandwidth selection, were accounted for in both approaches by analyzing ensembles of pseudo-data resampled from original microscopic measurements. Smooth plankton spectra were

675 obtained using the *combined* approach, where all phytoplankton and all zooplankton data were lumped together respectively and single bandwidths were calculated for every ensemble member (set of resampled data). This procedure avoided overfitting but was also prone to over-smoothing. ~~More structured, which can mask details, such as troughs in specific size ranges.~~ More details in the size spectra were ~~obtained with the resolved with tan~~ obtained with the resolved with tan *composite* approach, where individual size spectra were ~~calculated for each species or genus and then pieced together, were assembled.~~ calculated for each species or genus and then pieced together, were assembled. Since the variance within species or genus groups

680 is smaller than within the large groups 'phytoplankton' or 'zooplankton', ~~resulting bandwidths~~ the individual bandwidths, and therefore the degree of smoothing, ~~were~~ were considerably smaller than obtained in the *combined* approach. This computationally expensive method revealed many details in the spectra, but at the same time tended to resolve narrow peaks that were either clearly insignificant or remained difficult to interpret (see supplemental material in Lampe et al. (2021)). The here proposed diffKDE is tested with resampled data used for the simpler *combined* approach. The objective is to identify details in the size

685 spectra that remained previously unresolved while insignificant peaks, as found in the *composite* approach, become smoothed

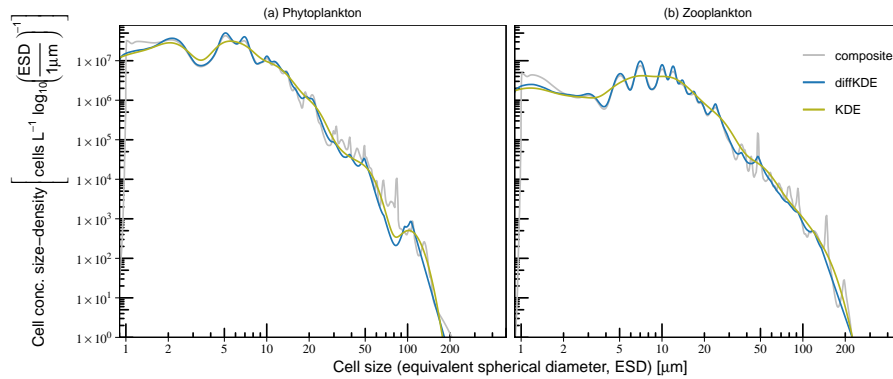


Figure 12. Comparison of KDE performance on (a) phytoplankton and (b) microzooplankton size spectra. The construction of composite and combined size spectra is described in Lampe et al. (2021) and based on Gaussian KDEs. Smoother combined spectra are the result of one KDE with a common bandwidth for all data. More structured composite spectra were assembled from taxon-specific spectra with individual, hence smaller, bandwidths.

out. Figure 12 shows the performance of the diffKDE in comparison to the original combined and composite spectra that were derived as ensemble means of estimates obtained with a Gaussian KDE. The spatial discretization of the diffKDE was set to $n=600$ $n=1000$ to be comparable to the other already published KDEs in this case. The diffKDE seems to meaningfully combine the advantages of the two Gaussian KDE approaches in both spectra, of the phytoplankton and microzooplankton respectively. With the diffKDE it is possible to generate estimates that display more detailed structure of the composite KDE for cell sizes smaller than 10. In some size ranges, such as between 5 and 50 μm , in particular in the microzooplankton spectrum. Concurrently, detailed variations, as caused by overfitting in the composite spectra, become suppressed for cell sizes larger than 10 μm . Thus, with the diffKDE it is possible to generate a single robust estimate that otherwise is only achieved by analyzing a series of composite of a series of individual estimates of a Gaussian KDE.

The application of the diffKDE for analysing details in plankton size spectra, or generally in particle size spectra, reduces computational efforts considerably.

7.3 Future application to model calibration Performance analyses on remote sensing data

The robustness of Earth system models is crucial for providing reliable climate projections for a sustainable development into Earth's future. Such models can assist the understanding of past and present and predict future conditions in the Earth system. Earth system models simulate the ocean's element cycling (e.g., Ilyina et al., 2013) and with this the ocean's carbon uptake capacity (e.g., Frölicher et al., 2015). They serve to assess the current and future state of our climate system and provide projections for different mitigation scenarios. This information can be used to support a sustainable development in our climate system (IPCC, 2022). As a consequence, political decisions depend on reliable projections to construct a safe pathway into Earth's future.

Calibration can increase the reliability of Earth system models (e.g., Oliver et al., 2022). For this purpose, a metric calculates the difference between simulated model output and measured field data. This metric defines the target or cost function in an optimization process, where unknown or uncertain model parameters are identified or estimated by numerical algorithms. This process is sometimes also called "tuning" of the model. The result is usually a single or multiple sets of "optimal" parameters. They provide the

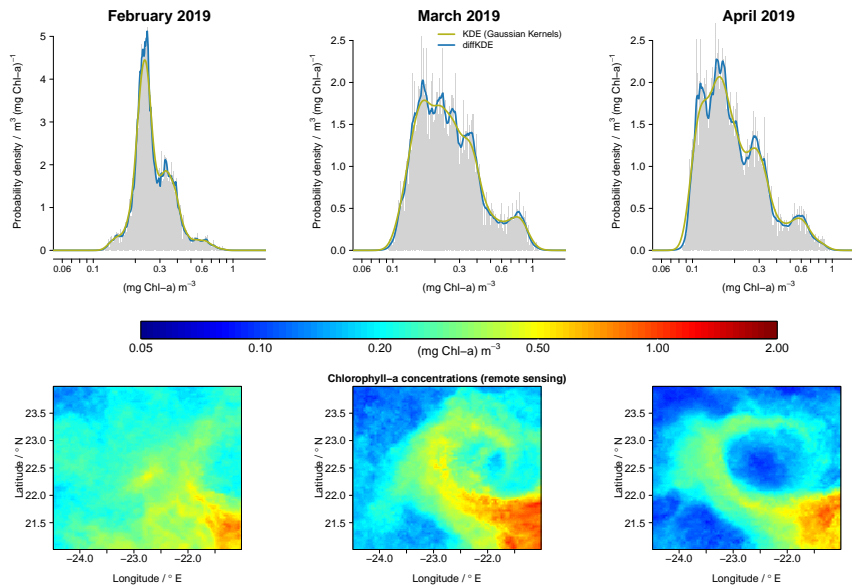


Figure 13. Comparison of KDE performance using monthly means (February, March, and April) of chlorophyll-a concentration derived from remote sensing data of the year 2019. The PDFs (upper panel) represent the temporal development of spatial variability seen in a subregion of the Mauritanian upwelling system (lower panel).

710

concentration within an off-shore ocean region (approximately $350 \text{ km} \times 330 \text{ km}$) that exhibits substantial mesoscale and sub-mesoscale variability. The selected area is part of the of the Mauritanian upwelling system, located at the Moroccan coast of North Africa. This eastern boundary upwelling is known for the formation and spread of filaments, with some distinct characteristics in terms of the spatial variability in temperature and the distribution of plankton (e.g., Sylla et al., 2019; Romero et al., 2020;

715

. For this example we use remote sensing (satellite) data of monthly mean chlorophyll-a concentration from the year 2019, as processed and made available through the Ocean-Colour Climate Change Initiative (OC-CCI) (Sathyendranath et al., 2019, 2021). We deliberately chose three months in which an eddy-like filamentous structure of elevated chlorophyll-a concentration developed after February, and then evolved during a two months period (March and April). Such development reveals specific spatial patterns, of low and increased chlorophyll-a, which leave a clear imprint in the corresponding PDFs, as depicted in Figure 13. In Figure 13 we find multiple modes, as well as a shift towards low chlorophyll-a concentrations that are much

720

better resolved when the diffKDE is applied. Such details derived with the diffKDE could be compared with complementary PDFs, for example as obtained from remote sensing sea surface temperature data within the same time period, and we might

gain further insight into the underlying processes involved in generating distinctive spatial and temporal patterns. Also, in case of simulations of mesoscale and sub-mesoscale processes, we should not expect to obtain a model solution with filamentous structures at identical times at the same places. But, we may regard a model's performance as credible if the model configuration with results closest to the incorporated field data's solution yields similar spatial structures visually within the same region, perhaps at some different time, and the associated PDFs may then be directly assessed against the PDFs obtained from the remote sensing data.

7.4 Future application to model assessment and calibration

In geoscientific research, the derivation and comparison of well resolved PDFs can be useful, as demonstrated in our selected examples. Yet, the significance of resolving details in nonparametric PDFs remains unclear. However, having high resolution PDFs available, as obtained with the diffKDE, is readily of value, and will likely guide further research. An obvious benefit of the diffKDE is its lesser dependence on the specification of a single, albeit optimal, bandwidth. Its application is likely more robust for the assessment of simulation results, either against data or results of other models (e.g. multi model ensembles), which is particularly relevant for evaluations of future climate projections obtained with Earth system models (e.g., Oliver et al., 2022). The presented diffKDE provides a nonparametric approach to estimate PDFs with typical features of geoscientific data. Being able to resolve typical patterns such as multiple or boundary close modes, while being insensitive to noise and individual outliers makes the diffKDE a suitable tool for future work in the calibration and optimization of Earth system models.

Comparison of model and field data requires additional processing to account for spatial-temporal differences between collected samples and model resolution. Typically, simulation results are available at every single spatial grid point and in every time step. In comparison, field data are usually sparsely available only. Interpolating such sparse field data can introduce high uncertainty (e.g., Oliver et al., 2022). PDFs provide a useful approach to investigate data independent of the number of data points available (Thorarinsdottir et al., 2013). A comparison of two such functions can easily resolve the issue of non-equal field observations and simulation results. Histograms are commonly used as an approach to compare and ultimately constrain the distribution of model data to observations. However, many issues arise including the subjective selection of intervals and histograms not being proper PDFs themselves.

The presented diffKDE provides a ~~non-parametric~~ nonparametric approach to estimate PDFs with typical features of geoscientific data. Being able to resolve typical patterns such as multiple or boundary close modes, while being insensitive to noise and individual outliers makes the diffKDE a suitable tool for future work in the calibration and optimization of Earth system models.

750 **8 Summary and conclusions**

In this study we constructed and tested an estimator (KDE) of probability density functions (PDFs) that can be applied for analysing geoscientific and ecological data. KDEs allow the investigation of data with respect to their probability distribution, and PDFs can be derived even for sparse data. To be well suited for geoscientific data, the KDE must work fast and reliably on

differently sized data sets, while revealing multimodal details as well as features nearby data boundaries. A KDE should not be overly sensitive to noise introduced by measurement errors or by numerical uncertainties. Such an estimator can be applied for direct data analyses or can be used to construct a target function for model assessment and calibration.

We presented a novel implementation of a KDE based on the diffusion heat process (diffKDE). This idea was originally proposed by Chaudhuri and Marron (2000) and its benefits in comparison to traditional KDE approaches were widely investigated by Botev et al. (2010). ~~Our approach combines the solution of the diffusion equation with two pilot estimation steps that correspond to the Gaussian KDE. We used an~~ We chose this approach to KDE, because it offers three main benefits: (1) consistency at the boundaries (2) better resolution of multimodal data (3) a family of KDEs with different smoothing intensities. We provide our algorithm in an open source Python package. Our approach includes a new approximation of the optimal bandwidth for the diffKDE by a central differential quotient and plug-in of the pilot estimates. For their bandwidths we used variations of the bandwidth, which equals the square root of the final iteration time. We directly approximate the analytical solution of the optimal bandwidth with two pilot estimation steps and finite differences. We calculate the pilot estimates as solutions of a simplified diffusion equation up until final iteration times derived from literature based bandwidths called rule of thumb by Silverman (1986). Our new approach results in three subsequent estimations of the PDF, each of them chosen with a finer bandwidth approximation.

Finite differences build the fundamentals of our discretization. The spatial discretization are equidistant finite differences. The δ -distribution in the initial value is discretized by piecewise linear functions along the spatial discretization points constructing a Dirac-sequence. For the timestepping we applied an implicit Eulerian algorithm on an ordinary differential equation set up by a tridiagonal matrix corresponding to the diffusion equation on the spatial equidistant grid.

Our diffKDE implementation includes pre-implemented default output options. The first is the visualization of the diffusion time evolution showing the sequence of all solution steps from the initial values to the final diffKDE. This lets a user [see-view](#) the influence of individual data points and outlier accumulations on the final diffKDE and how this decreases over time. The second is the visualization of the pilot estimate that is also included in the partial differential equation to introduce adaptive smoothing properties. This provides the user an easy insight into the adaptive smoothing as well as the lower boundary of structure resolution given by this parameter function. Finally, an interactive plot provides a simple opportunity to explore all of these time iterations and look even beyond the optimal bandwidth and see smoother estimates.

Our implementation is fast and reliable on differently sized and multimodal data sets. We tested the implementation for up to 10 million data points and obtained acceptably fast results. A comparison of the diffKDE on known distributions together with classically employed KDEs showed reliable and often superior performance. For comparison we chose a SciPy implementation (Gommers et al., 2022) of the most classical Gaussian KDE (Sheather, 2004), an Scikit implementation (Pedregosa et al., 2012) of an Epanechnikov KDE (Scott, 1992) and a Python implementation (Hennig, 2021) of the improved Gaussian KDE developed by Botev et al. (2010). We designed multimodal and different boundary-close distributions and found our implementation to generate the most reliable estimates across a large range of sample sizes (Fig. 9). The diffKDE was neither prone to oversmoothing nor overfitting of the data, which we could observe in the other tested KDEs. A noise sensitivity test in comparison to the other KDEs also showed a good stability of the diffKDE against noise in the data.

An assessment of the diffKDE on real marine biogeochemical field data in comparison to usually employed KDEs reveals superior performance of the diffKDE. We used carbon isotope and plankton size spectra data and compared the diffKDE to the KDEs that were used to explore the data in the respective original data publications. On the carbon isotope data, we furthermore applied all previous KDEs for comparison. In both cases we were able to show that the diffKDE resolves relevant features of the data while not being sensitive to individual outliers or uncertainties (noise) in the data. We were able to obtain a best possible and reliable representation of the true data distribution, better than those derived with other KDEs.

In future studies the diffKDE may potentially be used for the assessment, calibration and optimization of marine biogeochemical and Earth system models. Already a plot of PDFs, of field data and simulation results respectively, may provide visual insight into some shortcomings of the applied model. A target function can be constructed by adding a distance like the Wasserstein distance (Panaretos and Zemel, 2019) or other useful metrics for the calibration of climate models that can be investigated (Thorarinsdottir et al., 2013). Thus, KDE applications such as our diffKDE can greatly simplify comparisons of differently sized field and simulation data sets.

Code availability. The exact version of the diffKDE implementation (Pelz and Slawig, 2023) used to produce the results used in this paper is archived on Zenodo: <https://doi.org/10.5281/zenodo.7594915>.

Appendix A: Optimal bandwidth choice for the general KDE

The derivation of the optimal bandwidth choice for a KDE was already described in Parzen (1962) and can be found in more detail in Silverman (1986). The additional conditions stated in Eq. 7 to the kernel function

$$\int_{\mathbb{R}} yK(y) dy = 0 \text{ and } \int_{\mathbb{R}} y^2 K(y) dy = k_2 \in \mathbb{R} \setminus \{0\} \quad (\text{A1})$$

correspond to the order of the kernel being equal to 2 (Berliner, 1993). For such kernels Silverman (1986) showed the minimizer of the asymptotic mean integrated squared error to be

$$h = \left(\frac{\int_{-\infty}^{\infty} K(y)^2 dy}{Nk_2^2 \int_{-\infty}^{\infty} f''(x)^2 dx} \right)^{\frac{1}{5}} = \left(\frac{\|K\|_{L^2}^2}{Nk_2^2 \|f''\|_{L^2}^2} \right)^{\frac{1}{5}} \quad (\text{A2})$$

In our context of working with the squared bandwidth $t = h^2$ this optimal bandwidth choice becomes

$$t = \left(\frac{\|K\|_{L^2}^2}{Nk_2^2 \|f''\|_{L^2}^2} \right)^{\frac{2}{5}}, \quad (\text{A3})$$

which equals Eq. 13.

Appendix B: Integral property of the Dirac sequence

Here, we briefly give the proof of the integral property of the used Dirac sequence Φ_h defined in Equation 36. Let $h \in \mathbb{R}_{>0}$.

815 Then we obtain

$$\begin{aligned}
 \int \Phi_h(x) dx &= \int_{x_{i-2}}^{x_{i-1}} \Phi_h(x) dx + \int_{x_{i-1}}^{x_i} \Phi_h(x) dx + \int_{x_i}^{x_{i+1}} \Phi_h(x) dx \\
 &= \frac{1}{2} (x_{i-2} - x_{i-1}) \frac{1}{x_{i-2} - x_{i-1}} \frac{x_i}{x_i - x_{i-1}} + \frac{1}{2} (x_i - x_{i-1}) \left(\frac{1}{x_{i-2} - x_{i-1}} \frac{x_i}{x_i - x_{i-1}} + \frac{1}{x_{i+1} - x_i} \frac{-x_{i-1}}{x_i - x_{i-1}} \right) \\
 &\quad + \frac{1}{2} (x_{i+1} - x_i) \frac{1}{x_{i+1} - x_i} \frac{-x_{i-1}}{x_i - x_{i-1}} \\
 &= \frac{1}{2} h \frac{1}{h} \frac{x_i}{h} + \frac{1}{2} h \left(\frac{1}{h} \frac{x_i}{h} + \frac{1}{h} \frac{-x_{i-1}}{h} \right) + \frac{1}{2} h \frac{1}{h} \frac{-x_{i-1}}{h} \\
 820 &= \frac{1}{2} \frac{x_i}{h} + \frac{1}{2} \frac{x_i}{h} - \frac{1}{2} \frac{x_{i-1}}{h} - \frac{1}{2} \frac{x_{i-1}}{h} \\
 &= \frac{x_i - x_{i-1}}{h} \\
 &= 1.
 \end{aligned} \tag{B1}$$

Appendix C: Error convergence of observed KDEs

825 Tab. C1 shows the error values calculated by the Wasserstein distance and the MISE between the true distribution and the respective KDEs. The used distribution is the trimodal from Eq. 39 and the values plotted in Fig. 9.

Author contributions. MTP set up the manuscript and developed the implementation of the diffusion-based kernel density estimator. VL conducted the comparison experiments with the plankton size spectra. MS conducted the comparison experiments with the remote sensing data. CJS edited the manuscript. MS and TS edited the manuscript and supported the development of the implementation of the diffusion-based kernel density estimator.

830 *Competing interests.* The contact author has declared that neither they nor their co-authors have any competing interests.

Acknowledgements. The first author is funded through the Helmholtz School for Marine Data Science (MarDATA), Grant No. HIDSS-0005.

Table C1. Error convergence of the observed KDEs in Fig. 9: The first column lists the sample sizes used for the calculation in each row. The following four lines show the error between the four observed KDEs and the true distribution calculated by the Wasserstein distance. The final four columns contain the equivalent errors, but calculated by the MISE.

<u>sample size</u>	<u>W_{d1LKDE}</u>	<u>W_{BKDE}</u>	<u>W_{GKDE}</u>	<u>W_{EKDE}</u>	<u>$MISE_{d1LKDE}$</u>	<u>$MISE_{BKDE}$</u>	<u>$MISE_{GKDE}$</u>	<u>$MISE_{EKDE}$</u>
10	0.0235	0.0362	0.0266	0.0273	0.0313	0.0544	0.04	0.0327
50	0.0181	0.0248	0.0218	0.0202	0.012	0.0133	0.0246	0.0098
100	0.0042	0.0072	0.0137	0.007	0.0074	0.0075	0.019	0.0068
150	0.0066	0.0094	0.0153	0.0093	0.0057	0.0059	0.016	0.0062
200	0.0078	0.0089	0.0152	0.0097	0.0045	0.0046	0.0141	0.0057
300	0.0053	0.0062	0.0139	0.0085	0.003	0.0032	0.0116	0.0051
400	0.0036	0.0048	0.0115	0.0081	0.0024	0.0025	0.0098	0.0047
500	0.0027	0.0032	0.0106	0.0076	0.0021	0.0024	0.009	0.0049
750	0.0036	0.0034	0.01	0.0081	0.0015	0.0016	0.007	0.0045
1000	0.0032	0.0024	0.0093	0.0079	0.0011	0.0013	0.006	0.0044
2000	0.0024	0.0017	0.0074	0.0077	0.0006	0.0008	0.0039	0.0043
5000	0.0015	0.0019	0.0058	0.008	0.0003	0.0004	0.0022	0.0042
10000	0.002	0.0019	0.0044	0.0079	0.0002	0.0002	0.0013	0.0041
50000	0.0011	0.0009	0.0024	0.0081	0.00006	0.00007	0.0004	0.0041
100000	0.0007	0.0006	0.0018	0.0082	0.00004	0.00004	0.0002	0.0041
500000	0.0005	0.0004	0.0011	0.0084	0.00002	0.00001	0.00007	0.0041
1000000	0.0005	0.0003	0.00081	0.0084	0.00001	0.000007	0.00004	0.0041

References

- Abramson, I. S.: On bandwidth variation in kernel estimates-a square root law, *The annals of Statistics*, pp. 1217–1223, 1982.
- Berlinet, A.: Hierarchies of higher order kernels, *Probability Theory and Related Fields*, 94, 489–504, <https://doi.org/10.1007/bf01192560>, 1993.
- Bernacchia, A. and Pigolotti, S.: Self-Consistent Method for Density Estimation, *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 73, 407–422, <https://doi.org/10.1111/j.1467-9868.2011.00772.x>, 2011.
- Boccarda, N.: *Functional Analysis - An Introduction for Physicists*, Academic Press, Inc., 1990.
- Botev, Z. I., Grotowski, J. F., and Kroese, D. P.: Kernel density estimation via diffusion, *The annals of Statistics*, 38, 2916–2957, <https://doi.org/10.1214/10-AOS799>, 2010.
- Breiman, L., Meisel, W., and Purcell, E.: Variable kernel estimates of multivariate densities, *Technometrics*, 19, 135–144, 1977.
- Chacón, J. E. and Duong, T.: *Multivariate kernel smoothing and its applications*, CRC Press, 2018.

- Chaudhuri, P. and Marron, J.: Scale space view of curve estimation, *ANNALS OF STATISTICS*, 28, 408–428, <https://doi.org/10.1214/aos/1016218224>, 2000.
- 845 Chung, Y.-W., Khaki, B., Chu, C., and Gadh, R.: Electric Vehicle User Behavior Prediction Using Hybrid Kernel Density Estimator, in: 2018 IEEE International Conference on Probabilistic Methods Applied to Power Systems (PMAPS), pp. 1–6, <https://doi.org/10.1109/PMAPS.2018.8440360>, 2018.
- Davies, T. M. and Baddeley, A.: Fast computation of spatially adaptive kernel estimates, *Statistics and Computing*, 28, 937–956, <https://doi.org/10.1007/s11222-017-9772-4>, 2017.
- 850 Dekking, F. M., Kraaikamp, C., Lopuhaä, H. P., and Meester, L. E.: *A Modern Introduction to Probability and Statistics*, Springer London, <https://doi.org/10.1007/1-84628-168-7>, 2005.
- Deniz, T., Cardanobile, S., and Rotter, S.: A PYTHON Package for Kernel Smoothing via Diffusion: Estimation of Spike Train Firing Rate, *Front. Comput. Neurosci. Conference Abstract: BC11 : Computational Neuroscience & Neurotechnology Bernstein Conference & Neurex Annual Meeting 2011*, 5, <https://doi.org/10.3389/conf.fncom.2011.53.00071>, 2011.
- 855 Dessai, S., Lu, X., and Hulme, M.: Limited sensitivity analysis of regional climate change probabilities for the 21st century, *Journal of Geophysical Research: Atmospheres*, 110, <https://doi.org/https://doi.org/10.1029/2005JD005919>, 2005.
- Dirac, P. A. M.: The physical interpretation of the quantum dynamics, *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character*, 113, 621–641, <https://doi.org/10.1098/rspa.1927.0012>, 1927.
- Farmer, J. and Jacobs, D. J.: MATLAB tool for probability density assessment and nonparametric estimation, *SoftwareX*, 18, 101 017, <https://doi.org/10.1016/j.softx.2022.101017>, 2022.
- 860 Frölicher, T. L., Sarmiento, J. L., Paynter, D. J., Dunne, J. P., Krasting, J. P., and Winton, M.: Dominance of the Southern Ocean in Anthropogenic Carbon and Heat Uptake in CMIP5 Models, *Journal of Climate*, 28, 862–886, <https://doi.org/10.1175/jcli-d-14-00117.1>, 2015.
- Gommers, R., Virtanen, P., Burovski, E., Weckesser, W., Oliphant, T. E., Cournapeau, D., Haberland, M., Reddy, T., alexbr, Peterson, P., Nelson, A., Wilson, J., endolith, Mayorov, N., Polat, I., van der Walt, S., Laxalde, D., Brett, M., Larson, E., Millman, J.,
865 Lars, peterbell10, Roy, P., van Mulbregt, P., Carey, C., eric jones, Sakai, A., Moore, E., Kai, and Kern, R.: *scipy/scipy: SciPy 1.8.0*, <https://doi.org/10.5281/zenodo.5979747>, 2022.
- Gramacki, A.: *Nonparametric Kernel Density Estimation and Its Computational Aspects*, Springer International Publishing, <https://doi.org/10.1007/978-3-319-71688-6>, 2018.
- Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J.,
870 Kern, R., Picus, M., Hoyer, S., van Kerkwijk, M. H., Brett, M., Haldane, A., del Río, J. F., Wiebe, M., Peterson, P., Gérard-Marchant, P., Sheppard, K., Reddy, T., Weckesser, W., Abbasi, H., Gohlke, C., and Oliphant, T. E.: Array programming with NumPy, *Nature*, 585, 357–362, <https://doi.org/10.1038/s41586-020-2649-2>, 2020.
- Heidenreich, N.-B., Schindler, A., and Sperlich, S.: Bandwidth selection for kernel density estimation: a review of fully automatic selectors, *AStA Advances in Statistical Analysis*, 97, 403–433, <https://doi.org/10.1007/s10182-013-0216-y>, 2013.
- 875 Hennig, J.: *John-Hennig/KDE-diffusion: KDE-diffusion 1.0.3*, <https://doi.org/10.5281/zenodo.4663430>, 2021.
- Hirsch, F. and Lacombe, G.: *Elements of Functional Analysis*, Springer, 1999.
- Hunter, J. D.: Matplotlib: A 2D graphics environment, *Computing in science & engineering*, 9, 90–95, <https://doi.org/10.1109/mcse.2007.55>, 2007.

- Ilyina, T., Six, K. D., Segschneider, J., Maier-Reimer, E., Li, H., and Núñez-Riboni, I.: Global ocean biogeochemistry model HAMOCC: Model architecture and performance as component of the MPI-Earth system model in different CMIP5 experimental realizations, *Journal of Advances in Modeling Earth Systems*, 5, 287–315, <https://doi.org/10.1029/2012ms000178>, 2013.
- IPCC: *Climate Change 2022: Mitigation of Climate Change. Contribution of Working Group III to the Sixth Assessment Report of the Intergovernmental Panel on Climate Change* [P.R. Shukla, J. Skea, R. Slade, A. Al Khourdajie, R. van Diemen, D. McCollum, M. Pathak, S. Some, P. Vyas, R. Fradera, M. Belkacemi, A. Hasija, G. Lisboa, S. Luz, J. Malley, (eds.)], <https://doi.org/10.1017/9781009157926>, 2022.
- Jones, M. C., Marron, J. S., and Sheather, S. J.: A Brief Survey of Bandwidth Selection for Density Estimation, *Journal of the American Statistical Association*, 91, 401–407, <https://doi.org/10.1080/01621459.1996.10476701>, 1996.
- Khorramdel, B., Chung, C. Y., Safari, N., and Price, G. C. D.: A Fuzzy Adaptive Probabilistic Wind Power Prediction Framework Using Diffusion Kernel Density Estimators, *IEEE Transactions on Power Systems*, 33, 7109–7121, <https://doi.org/10.1109/tpwrs.2018.2848207>, 2018.
- Kirk, J. T. O.: *Light and Photosynthesis in Aquatic Ecosystems - Third edition*, Cambridge Univ. Press, 2011.
- Lampe, V., Nöthig, E.-M., and Schartau, M.: Spatio-Temporal Variations in Community Size Structure of Arctic Protist Plankton in the Fram Strait, *Frontiers in Marine Science*, 7, <https://doi.org/10.3389/fmars.2020.579880>, 2021.
- Li, G., Lu, W., Bian, J., Qin, F., and Wu, J.: Probabilistic Optimal Power Flow Calculation Method Based on Adaptive Diffusion Kernel Density Estimation, *Frontiers in Energy Research*, 7, <https://doi.org/10.3389/fenrg.2019.00128>, 2019.
- Ma, S., Sun, S., Wang, B., and Wang, N.: Estimating load spectra probability distributions of train bogie frames by the diffusion-based kernel density method, *International Journal of Fatigue*, 132, 105–132, <https://doi.org/10.1016/j.ijfatigue.2019.105352>, 2019.
- Majdara, A. and Nooshabadi, S.: Nonparametric Density Estimation Using Copula Transform, Bayesian Sequential Partitioning, and Diffusion-Based Kernel Estimator, *IEEE Transactions on Knowledge and Data Engineering*, 32, 821–826, <https://doi.org/10.1109/tkde.2019.2930052>, 2020.
- Marron, J. S. and Ruppert, D.: Transformations to reduce boundary bias in kernel density estimation, *Journal of the Royal Statistical Society: Series B (Methodological)*, 56, 653–671, <https://www.jstor.org/stable/2346189>, 1994.
- McSwiggan, G., Baddeley, A., and Nair, G.: Kernel Density Estimation on a Linear Network, *Scandinavian Journal of Statistics*, 44, 324–345, <https://doi.org/10.1111/sjos.12255>, 2016.
- Nöthig, E.-M., Bracher, A., Engel, A., Metfies, K., Niehoff, B., Peeken, I., Bauerfeind, E., Cherkasheva, A., Gäbler-Schwarz, S., Hardge, K., Kiliyas, E., Kraft, A., Mebrahtom Kidane, Y., Lalande, C., Piontek, J., Thomisch, K., and Wurst, M.: Summertime plankton ecology in Fram Strait - a compilation of long- and short-term observations, *Polar Research*, 34, 23–349, <https://doi.org/10.3402/polar.v34.23349>, 2015.
- O'Brien, J. P., O'Brien, T. A., Patricola, C. M., and Wang, S.-Y. S.: Metrics for understanding large-scale controls of multivariate temperature and precipitation variability, *Climate Dynamics*, 53, 3805–3823, <https://doi.org/10.1007/s00382-019-04749-6>, 2019.
- Oliver, S., Cartis, C., Kriest, I., Tett, S. F. B., and Khatiwala, S.: A derivative-free optimisation method for global ocean biogeochemical models, *Geoscientific Model Development*, 15, 3537–3554, <https://doi.org/10.5194/gmd-15-3537-2022>, 2022.
- Ongoma, V., Chen, H., Gao, C., and Sagero, P. O.: Variability of temperature properties over Kenya based on observed and reanalyzed datasets, *Theoretical and Applied Climatology*, 133, 1175–1190, <https://doi.org/10.1007/s00704-017-2246-y>, 2017.
- Palmer, T. N.: Towards the probabilistic Earth-system simulator: a vision for the future of climate and weather prediction, *Quarterly Journal of the Royal Meteorological Society*, 138, 841–861, <https://doi.org/10.1002/qj.1923>, 2012.

- Panaretos, V. M. and Zemel, Y.: Statistical Aspects of Wasserstein Distances, *Annual Review of Statistics and Its Application*, 6, 405–431, <https://doi.org/10.1146/annurev-statistics-030718-104938>, 2019.
- Parzen, E.: On estimation of a probability density function and mode, *The annals of mathematical statistics*, 33, 1065–1076, 1962.
- 920 Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Müller, A., Nothman, J., Louppe, G., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E.: Scikit-learn: Machine Learning in Python, <https://doi.org/10.48550/ARXIV.1201.0490>, 2012.
- Pedretti, D. and Fernández-García, D.: An automatic locally-adaptive method to estimate heavily-tailed breakthrough curves from particle distributions, *Advances in Water Resources*, 59, 52–65, <https://doi.org/10.1016/j.advwatres.2013.05.006>, 2013.
- 925 Pelz, M.-T. and Slawig, T.: Diffusion-based kernel density estimator (diffKDE), <https://doi.org/10.5281/ZENODO.7594915>, 2023.
- Perkins, S. E., Pitman, A. J., and McAneney, N. J. H. J.: Evaluation of the AR4 Climate Models’ Simulated Daily Maximum Temperature, Minimum Temperature, and Precipitation over Australia Using Probability Density Functions, *Journal of Climate*, 20, 4356 – 4376, <https://doi.org/https://doi.org/10.1175/JCLI4253.1>, 2007.
- Qin, B. and Xiao, F.: A Non-Parametric Method to Determine Basic Probability Assignment Based on Kernel Density Estimation, *IEEE Access*, 6, 73 509–73 519, <https://doi.org/10.1109/ACCESS.2018.2883513>, 2018.
- 930 Quintana, X. D., Brucet, S., Boix, D., López-Flores, R., Gascón, S., Badosa, A., Sala, J., Moreno-Amich, R., and Egozcue, J. J.: A nonparametric method for the measurement of size diversity with emphasis on data standardization, *Limnology and Oceanography: Methods*, 6, 75–86, <https://doi.org/https://doi.org/10.4319/lom.2008.6.75>, 2008.
- Romero, O. E., Baumann, K.-H., Zonneveld, K. A. F., Donner, B., Hefter, J., Hamady, B., Pospelova, V., and Fischer, G.: Flux variability of phyto- and zooplankton communities in the Mauritanian coastal upwelling between 2003 and 2008, *Biogeosciences*, 17, 187–214, <https://doi.org/10.5194/bg-17-187-2020>, 2020.
- 935 Santhosh, D. and Srinivas, V. V.: Bivariate frequency analysis of floods using a diffusion based kernel density estimator, *Water Resources Research*, 49, 8328–8343, <https://doi.org/10.1002/2011wr010777>, 2013.
- Sathyendranath, S., Brewin, R. J., Brockmann, C., Brotas, V., Calton, B., Chuprin, A., Cipollini, P., Couto, A. B., Dingle, J., Doerffer, R., Donlon, C., Dowell, M., Farman, A., Grant, M., Groom, S., Horseman, A., Jackson, T., Krasemann, H., Lavender, S., Martinez-Vicente, V., Mazeran, C., Mélin, F., Moore, T. S., Müller, D., Regner, P., Roy, S., Steele, C. J., Steinmetz, F., Swinton, J., Taberner, M., Thompson, A., Valente, A., Zühlke, M., Brando, V. E., Feng, H., Feldman, G., Franz, B. A., Frouin, R., Gould, R. W., Hooker, S. B., Kahru, M., Kratzer, S., Mitchell, B. G., Muller-Karger, F. E., Sosik, H. M., Voss, K. J., Werdell, J., and Platt, T.: An Ocean-Colour Time Series for Use in Climate Studies: The Experience of the Ocean-Colour Climate Change Initiative (OC-CCI), *Sensors*, 19, <https://doi.org/10.3390/s19194285>, 2019.
- 945 Sathyendranath, S., Jackson, T., Brockmann, C., Brotas, V., Calton, B., Chuprin, A., Clements, O., Cipollini, P., Danne, O., Dingle, J., Donlon, C., Grant, M., Groom, S., Krasemann, H., Lavender, S., Mazeran, C., Mélin, F., Müller, D., Steinmetz, F., Valente, A., Zühlke, M., Feldman, G., Franz, B., Frouin, R., Werdell, J., and Platt, T.: Global chlorophyll-a data products gridded on a geographic projection, Version 5.0, NERC EDS Centre for Environmental Data Analysis [data set], 2021.
- Schartau, M., Landry, M. R., and Armstrong, R. A.: Density estimation of plankton size spectra: a reanalysis of IronEx II data, *Journal of Plankton Research*, 32, 1167–1184, <https://doi.org/10.1093/plankt/fbq072>, ISBN: 0142-7873, 2010.
- 950 Schmittner, A. and Somes, C. J.: Complementary constraints from carbon (^{13}C) and nitrogen (^{15}N) isotopes on the glacial ocean’s soft-tissue biological pump, *Paleoceanography*, pp. 669–693, <https://doi.org/10.1002/2015PA002905>, 2016.
- Scott, D. W.: *Multivariate density estimation: theory, practice, and visualization*, John Wiley & Sons, 1992.

- Scott, D. W.: Multivariate density estimation and visualization, in: Handbook of computational statistics, pp. 549–569, Springer, 955 <https://doi.org/10.1007/978-3-642-21551-3-19>, 2012.
- Sheather, S. J.: Density Estimation, *Statistical Science*, 19, 588–597, <https://doi.org/10.1214/088342304000000297>, 2004.
- Sheather, S. J. and Jones, M. C.: A reliable data-based bandwidth selection method for kernel density estimation, *Journal of the Royal Statistical Society: Series B (Methodological)*, 53, 683–690, 1991.
- Silverman, B.: Density estimation, *Monographs on Statistics and Applied Probability*, 1986.
- 960 Sylla, A., Mignot, J., Capet, X., and Gaye, A. T.: Weakening of the Senegalo–Mauritanian upwelling system under climate change, *Climate Dynamics*, 53, 4447–4473, <https://doi.org/10.1007/s00382-019-04797-y>, 2019.
- Terrell, G. R. and Scott, D. W.: Variable kernel density estimation, *The Annals of Statistics*, pp. 1236–1265, <https://www.jstor.org/stable/2242011>, 1992.
- Teshome, A. and Zhang, J.: Increase of Extreme Drought over Ethiopia under Climate Warming, *Advances in Meteorology*, 2019, 1–18, 965 <https://doi.org/10.1155/2019/5235429>, 2019.
- Thorarinsdottir, T. L., Gneiting, T., and Gissibl, N.: Using Proper Divergence Functions to Evaluate Climate Models, *SIAM/ASA Journal on Uncertainty Quantification*, 1, 522–534, <https://doi.org/10.1137/130907550>, 2013.
- Urtizbera, A., Dupont, N., Rosland, R., and Aksnes, D. L.: Sensitivity of euphotic zone properties to CDOM variations in marine ecosystem models, *Ecological Modelling*, 256, 16–22, <https://doi.org/10.1016/j.ecolmodel.2013.02.010>, 2013.
- 970 Van Rossum, G.: The Python Library Reference, release 3.8.2, Python Software Foundation, 2020.
- Versteegh, G. J. M., Zonneveld, K. A. F., Hefter, J., Romero, O. E., Fischer, G., and Mollenhauer, G.: Performance of temperature and productivity proxies based on long-chain alkane-1, mid-chain diols at test: a 5-year sediment trap record from the Mauritanian upwelling, *Biogeosciences*, 19, 1587–1610, <https://doi.org/10.5194/bg-19-1587-2022>, 2022.
- Verwega, M.-T., Somes, C. J., Schartau, M., Tuerena, R. E., Lorrain, A., Oschlies, A., and Slawig, T.: Description of a global marine 975 particulate organic carbon-13 isotope data set, *Earth System Science Data*, 13, 4861–4880, <https://doi.org/10.5194/essd-13-4861-2021>, 2021.
- Verwega, M.-T., Somes, C. J., Tuerena, R. E., and Lorrain, A.: A global marine particulate organic carbon-13 isotope data product, <https://doi.org/10.1594/PANGAEA.929931>, 2021.
- Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., 980 van der Walt, S. J., Brett, M., Wilson, J., Millman, K. J., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., Carey, C. J., Polat, İ., Feng, Y., Moore, E. W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E. A., Harris, C. R., Archibald, A. M., Ribeiro, A. H., Pedregosa, F., van Mulbregt, P., and SciPy 1.0 Contributors: SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python, *Nature Methods*, 17, 261–272, <https://doi.org/10.1038/s41592-019-0686-2>, 2020.
- Xu, X., Yan, Z., and Xu, S.: Estimating wind speed probability distribution by diffusion-based kernel density method, *Electric Power Systems 985 Research*, 121, 28–37, 2015.