nature methods

Article

https://doi.org/10.1038/s41592-022-01750-6

A framework for evaluating the performance of SMLM cluster analysis algorithms

Received: 21 July 2021

Accepted: 6 December 2022

Published online: 10 February 2023

Check for updates

Daniel J. Nieves[1,2], Jeremy A. Pike[2,3], Florian Levet[4,5], David J. Williamson[6], Mohammed Baragilly[1,7], Sandra Oloketuyi[8], Ario de Marco[8], Juliette Griffié[9], Daniel Sage[10], Edward A. K. Cohen[11], Jean-Baptiste Sibarita[4], Mike Heilemann[12] & Dylan M. Owen[1,2,13]

Single-molecule localization microscopy (SMLM) generates data in the form of coordinates of localized fluorophores. Cluster analysis is an attractive route for extracting biologically meaningful information from such data and has been widely applied. Despite a range of cluster analysis algorithms, there exists no consensus framework for the evaluation of their performance. Here, we use a systematic approach based on two metrics to score the success of clustering algorithms in simulated conditions mimicking experimental data. We demonstrate the framework using seven diverse analysis algorithms: DBSCAN, ToMATo, KDE, FOCAL, CAML, ClusterViSu and SR-Tesseler. Given that the best performer depended on the underlying distribution of localizations, we demonstrate an analysis pipeline based on statistical similarity measures that enables the selection of the most appropriate algorithm, and the optimized analysis parameters for real SMLM data. We propose that these standard simulated conditions, metrics and analysis pipeline become the basis for future analysis algorithm development and evaluation.

The assembly of biomolecules into clusters is a key process in cell biology[1,2]. Electron microscopy enables high-resolution observation of molecule clustering, but sample preparation can be time-consuming and difficult for intact samples[3]. Super-resolution optical fluctuation imaging can be applied to intact samples[4], but its applicability is limited by the requirement to observe fluorophores at higher density than other techniques. Given these limitations, single-molecule localization microscopy (SMLM) has become one of the most widely used methods to probe protein clustering, readily achieving nanoscale resolution (<20 nm) in intact samples[5]. Regardless of the SMLM modality (for example, direct stochastic optical reconstruction microscopy (dSTORM)[6,7], photoactivated localization microscopy (PALM)[8], or points accumulation for imaging in nanoscale topography (PAINT)[9] and DNA-PAINT[10–12]), data from SMLM experiments consist of a list of coordinates of all localized fluorophores. These data are suitable for the application of statistical methods to describe the spatial arrangement of such data[13]. One of the most common is cluster analysis[14].

Clustering methods can be classified into two groups: global clustering and complete clustering. Global cluster analysis returns an ensemble result; Ripley's K-function[15], nearest neighbor analysis[16]

[1]Institute of Immunology and Immunotherapy, College of Medical and Dental Sciences, University of Birmingham, Birmingham, UK. [2]Centre of Membrane Proteins and Receptors (COMPARE), University of Birmingham, Birmingham, UK. [3]Institute of Cardiovascular Sciences, College of Medical and Dental Sciences, University of Birmingham, Birmingham, UK. [4]Interdisciplinary Institute for Neuroscience, CNRS, IINS, UMR 5297, Université de Bordeaux, Bordeaux, France. [5]Bordeaux Imaging Center, CNRS, INSERM, BIC, UMS 3420, US 4, Université de Bordeaux, Bordeaux, France. [6]Department of Infectious Diseases, School of Immunology and Microbial Sciences, King's College London, London, UK. [7]Department of Mathematics, Insurance and Applied Statistics, Helwan University, Helwan, Egypt. [8]Laboratory of Environmental and Life Sciences, University of Nova Gorica, Rožna Dolina, Slovenia. [9]Laboratory of Experimental Biophysics, Institute of Physics, Ecole Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland. [10]Biomedical Imaging Group, Ecole Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland. [11]Department of Mathematics, Imperial College London, London, UK. [12]Institute of Physical and Theoretical Chemistry, Goethe-University Frankfurt, Frankfurt, Germany. [13]School of Mathematics, University of Birmingham, Birmingham, UK. e-mail: d.owen@bham.ac.uk

https://doi.org/10.1038/s41592-022-01750-6

Table 1 | Properties of different simulated cluster scenarios

Simulation	No. of clusters	No. of molecules per cluster	Elliptical	Cluster width (nm)	No. of background molecules	Non-clustered molecules (%)
Scenario 1	0	N/A	N/A	N/A	300	100
Scenario 2	20	15	N	25	300	50
Scenario 3	20	15	N	25	**1,500**	**80**
Scenario 4	20	**5**	N	25	300	**75**
Scenario 5	**100**	15	N	25	**1,500**	50
Scenario 6	20	**50**	**Y**	*x*=**25** *y*=**75**	**1,000**	50
Scenario 7	20	15	N	**10×25** **10×75**	300	50
Scenario 8	20	**10×5** **10×15**	N	25	**200**	50
Scenario 9	20	**10×15**§ **10×135**#	N	**10×25**§ **10×75**#	**1,500**	50
Scenario 10	20	15	N	25	300 (**gradient**)	50

Parameters used for the simulation of Scenarios 1–10. Bold indicates deviations from Scenario 2 for each scenario, while § and # in Scenario 9 indicate matching cluster parameters in that condition. N/A, not applicable, Y, yes, N, no.

and pair correlation[17] have all been applied to SMLM data[18–22]. Global approaches are statistically robust but provide a limited description of the data; thus, complete clustering approaches have gained popularity. These assign every localization to a specific cluster or into a non-clustered population[23–31]. Complete clustering methods provide rich descriptions of the data such as the number of clusters, cluster shapes and so on.

A consensus framework for assessment of the algorithms used to generate complete clusterings of SMLM data remains to be developed. However, there are existing mathematical means by which to assess the accuracy of clustering algorithms by comparing the result to a known ground truth simulated dataset. One such metric is the adjusted Rand index (ARI), which aims to determine which points have the same cluster classification in the ground truth and the analyzed output[32], normalized to account for chance[33]. ARI is an appropriate metric if the goal is to classify data points into correct clusters, that is, to determine cluster membership. It has a range between −1 and 1, with high positive values representing good agreement. Clustering performance can also be measured geometrically via the overlap between the cluster areas (for example, defined by the convex hull) in the output and the ground truth. This is the basis for the intersection over union (IoU)[34–36] metric, the values of which range from 0 to 1. IoU is therefore suited as a measure of how truthfully SMLM cluster analyses can recapitulate geometric features in the data.

Here, we implement ARI and IoU to score the analysis of a standardized, consensus set of simulated SMLM data mimicking common biological clustering conditions. To demonstrate the framework, we analyze the data with seven diverse algorithms: density-based spatial clustering of applications with noise (DBSCAN)[37], topological mode analysis tool (ToMATo)[28,38], kernel density estimation (KDE), fast optimized clustering algorithm for localizations (FOCAL)[26], cluster analysis by machine learning (CAML)[27], ClusterViSu[25] and SR-Tesseler[24].

The goal is threefold: to enable users to select the best-performing algorithm for their specific application; to enable selection of user-defined analysis settings to maximize the performance of their analysis; and to provide guidance on the future direction of cluster analysis development.

Results

Simulated conditions and data screening

We first generated 10 simulated conditions with a known ground truth in 2,000 × 2,000 nm regions (Scenarios 1–10; Table 1, Fig. 1 and Methods): Scenario 1, non-clustered molecules seeded at completely spatially random positions (Fig. 1a); Scenario 2, 20 clusters of 15 molecules per cluster with 50% of the total molecules being clustered (Fig. 1b); Scenario 3, 20 clusters of 15 molecules per cluster with 20% of the total molecules being clustered (Fig. 1c); Scenario 4, 20 clusters of 5 molecules per cluster with 25% of the total molecules being clustered (Fig. 1d); Scenario 5, 100 clusters of 15 molecules per cluster with 50% of molecules being clustered (Fig. 1e); Scenario 6, 20 elliptically shaped clusters with aspect ratio 3:1 and each having 50 molecules, with 50% of the total molecules being clustered (Fig. 1f); Scenario 7, 10 clusters with a width of 25 nm and 10 clusters with a width of 75 nm, with 50% of the total molecules clustered (Fig. 1g); Scenario 8, 10 clusters with 5 molecules per cluster and 10 clusters with 15 molecules per cluster, with 50% of the total molecules clustered (Fig. 1h); Scenario 9, 10 clusters with 15 molecules per cluster and a cluster width of 25 nm, and a further 10 clusters with 135 molecules and a cluster width of 75 nm, thus maintaining molecule density with increased size, with 50% of the total molecules clustered (Fig. 1i); and Scenario 10, 20 clusters of 15 molecules per cluster with 50% of the total molecules being clustered, but with non-clustered molecule density increasing from left to right across the region (Fig. 1j).

Complete clustering methods are not designed to test for the presence of clustering and can give erroneous results if applied to data without spatial structure. The first stage in an analysis pipeline is therefore to test for clustering in the data. This can be done using Ripley's K-function together with 95% simulation envelopes for complete spatial randomness (Fig. 1). Only datasets for which the estimated K-function lies outside of these envelopes, indicating statistically significant levels of clustering, are appropriate for input into complete clustering algorithms. From our simulated conditions, all but Scenario 1 reject the null hypothesis of a completely random distribution (Fig. 1b–j, bottom panels). Scenario 1 was therefore excluded from further study.

Demonstration of clustering performance analysis

We next analyzed the simulations with the clustering algorithms and scored the results. The selected algorithms encompass a cross-section of those currently available for SMLM cluster analysis and use diverse strategies to generate their output. For those algorithms that required user settings, that is, DBSCAN, ToMATo, KDE, FOCAL and SR-Tesseler, we scanned a wide range of these user inputs to generate hundreds of cluster outputs per algorithm. To illustrate, Fig. 2a shows ground truth and DBSCAN, ToMATo and KDE outputs for a representative realization

https://doi.org/10.1038/s41592-022-01750-6

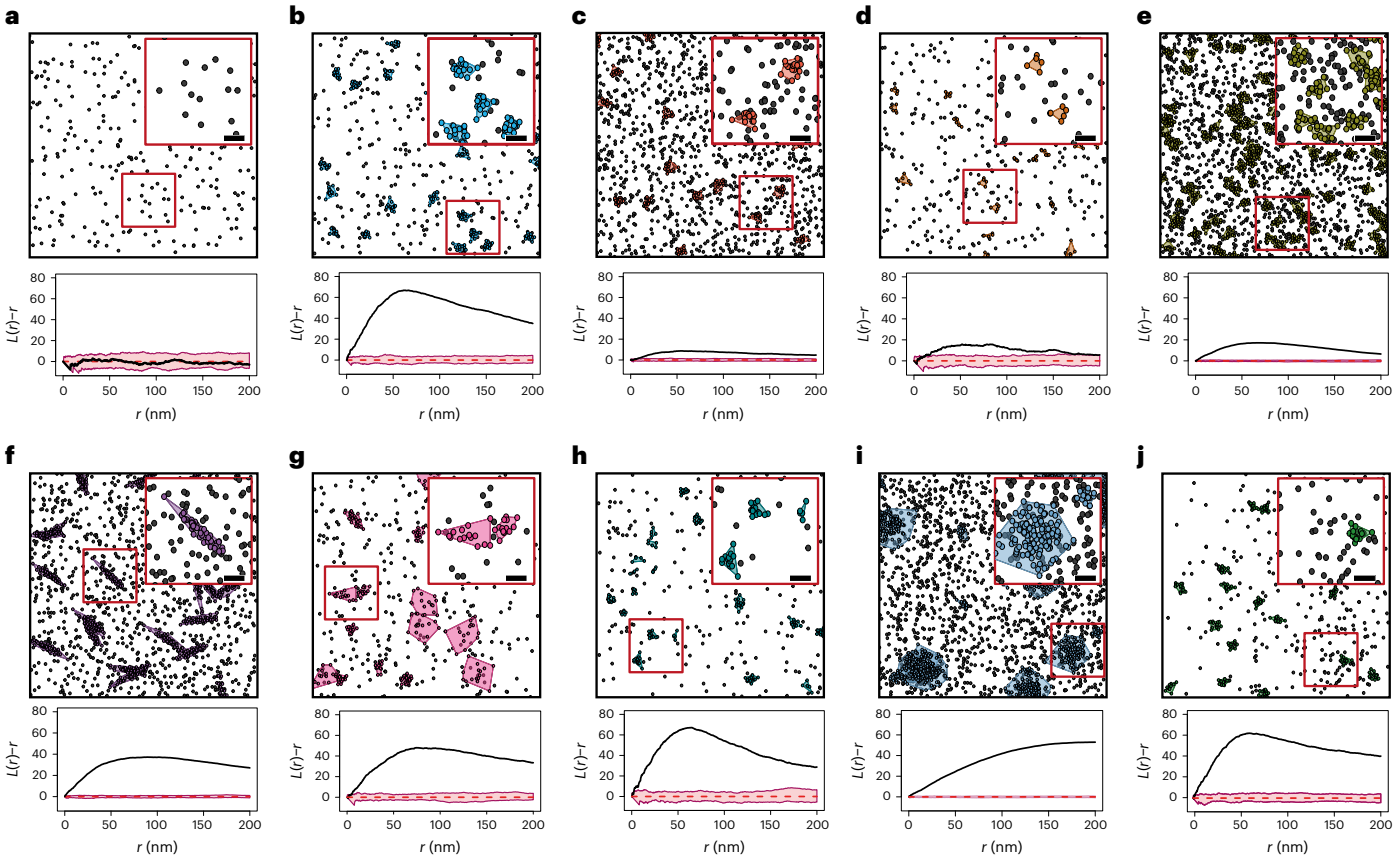

Fig. 1 | Examples of simulated data conditions and pre-evaluation of suitability. a–j, The 10 simulated data conditions: Scenario 1 (**a**), Scenario 2 (**b**), Scenario 3 (**c**), Scenario 4 (**d**), Scenario 5 (**e**), Scenario 6 (**f**), Scenario 7 (**g**), Scenario 8 (**h**), Scenario 9 (**i**) and Scenario 10 (**j**). Top panels: examples of simulation data (colored points and polygons are clustered points, gray points are non-clustered). The insets are zooms of the red boxed regions. Scale bars, 100 nm. Bottom panels: linear representation of Ripley's K-function ($L(r) - r$). The black line represents the ground truth simulation data, and the red line is the mean of 100 simulations using the same number of molecules placed completely randomly, with 95% simulation envelopes (pink).

of Scenario 2. The mean ARI and IoU scores and their variance across the $n = 50$ realizations are shown in Fig. 2b–d. The peak scores are shown in Supplementary Figs. 3–5. For FOCAL (Supplementary Fig. 1) and SR-Tesseler (Supplementary Fig. 2), a single parameter was scanned. ClusterViSu was implemented as published[25], with optimal analysis parameters determined by Monte Carlo simulations. For CAML, the algorithm was trained on separate simulation data corresponding to each condition, and the best-performing model in each case was chosen for analysis of the data. For each algorithm requiring parameter scanning, the mean metric scores for each combination were calculated. The best-performing parameter set in a scenario was that which achieved the highest mean metric score. For DBSCAN, ToMATo, KDE, FOCAL and SR-Tesseler, the optimal user parameters are summarized in Supplementary Table 1.

Performance analysis on simulated data

We next carried out the same performance analysis on all simulated conditions. The mean highest ARI and IoU scores for each condition are summarized in Fig. 3, with the distribution of scores shown in Supplementary Figs. 3–9a,b.

From these data we can see that the best-performing algorithm depends on the choice of metric, which must be motivated by the biological question under study. For example, for Scenario 4, which features sparse data, IoU scores are generally low for all algorithms because it requires accurate mapping of the cluster boundaries, meaning that it is sensitive to misclassification of only a few points in the cluster periphery. ARI is more robust for these types of effects. Conversely, IoU scores remain high for most algorithms in Scenario 9, in which large clusters merge, causing misattribution of points even though the area of the clusters remains well determined. This phenomenon decreases ARI scores.

DBSCAN, ToMATo and KDE perform comparably across all of the conditions according to the framework, with only subtle differences in the mean peak scores and distribution of those values (Fig. 3 and Supplementary Figs. 3–5a,b). Although the algorithms selected here are diverse, DBSCAN, ToMATo and KDE are the most similar in terms of the two user inputs: one relates to a radius of search or integration, and the other operates as a density threshold to terminate clustering. Thus, the generally good performance observed for these three algorithms may point to an inherent adaptability of these algorithms for different clustering situations. In contrast, ClusterViSu, SR-Tesseler and FOCAL often return lower metric scores. All require a threshold to segment clusters, and Scenario 4 is where the performance of these algorithms drops. Conversely, their performance is best in Scenario 3, and comparable to the other algorithms. This suggests that there may be a general issue with these algorithms, such that in data with low point density within clusters, the choice of a threshold, automatically or not, is difficult. This is likely to be a consequence of the points at the edge of the clusters having a similar density to the background molecules.

Addition of multiple blinking

An inherent property of SMLM data is the presence of multiple points arising from a single molecule. Each ground truth fluorophore may therefore appear as a small cluster, with membership related to the

https://doi.org/10.1038/s41592-022-01750-6

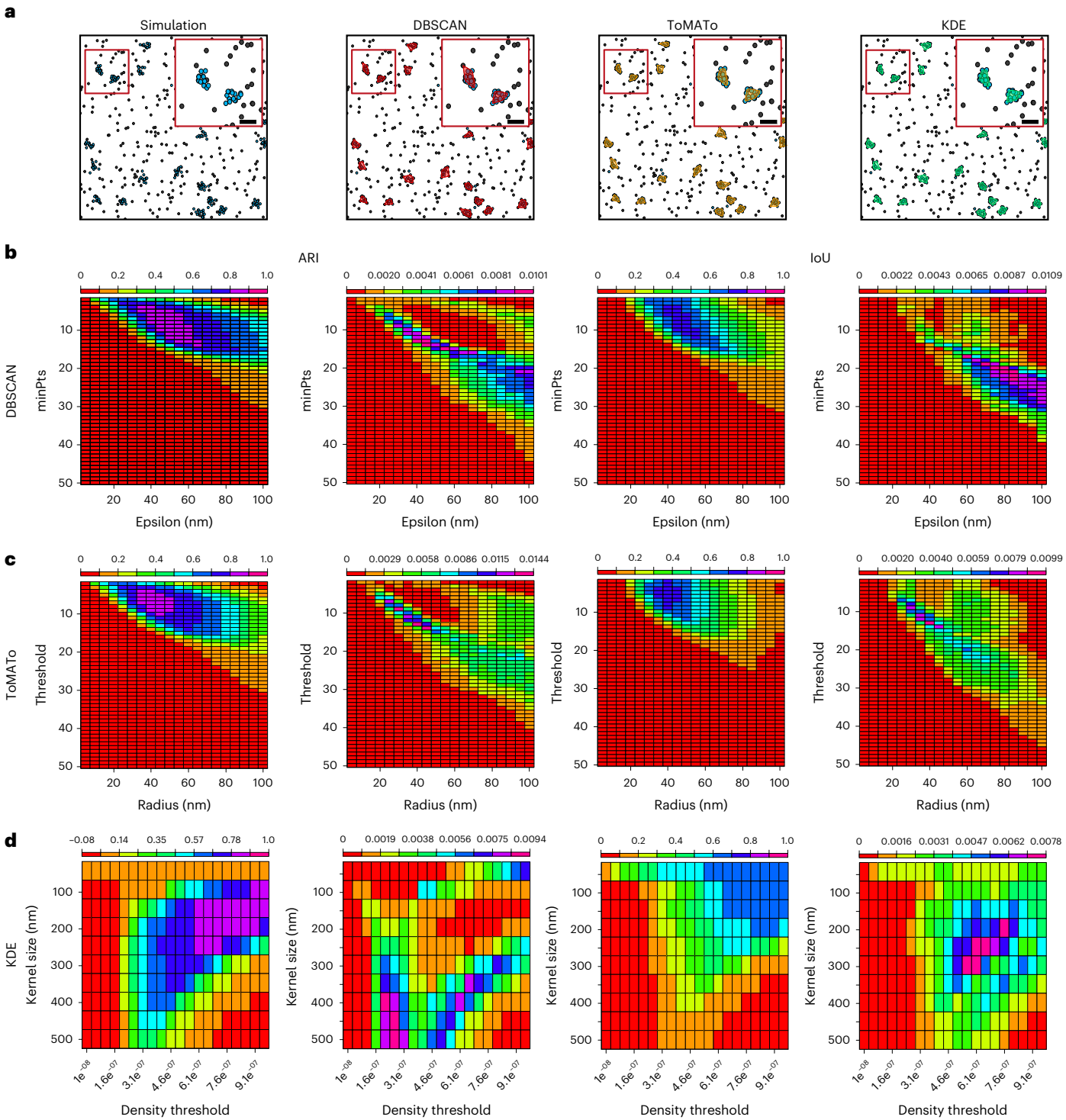

Fig. 2 | Representative example of a multi-parameter scan and performance analysis of DBSCAN, ToMATo and KDE for Scenario 2. a, Example of simulated datasets showing clustered points (blue) and cluster maps generated by parameters achieving peak ARI scores for DBSCAN, ToMATo and KDE. The insets are zooms of the red boxed regions, which show the same clusters for each algorithm. Scale bars, 100 nm. **b**–**d**, Mean and variance of the ARI and IoU scores for DBSCAN (**b**), ToMATo (**c**) and KDE (**d**) for all combinations of user analysis parameters.

number of blinks and size related to the localization precision. We test the performance of the cluster analysis algorithms against simulated data in the presence of non-corrected blinking, for which the probability of blinking followed a geometric distribution[23] set such that a single fluorophore would give 4–5 detections, on average. For all conditions the mean localization precision (σ_{loc}) was approximately 15 nm. It is important to note that with multiple blinking, Scenario 1, that is, completely randomly dispersed ground truth molecules, would appear clustered and pass the Ripley's K-function quality control proposed earlier. Thus, as a general recommendation, the use of correction algorithms to mitigate multiple blinking prior to testing would be advised[39,40].

https://doi.org/10.1038/s41592-022-01750-6

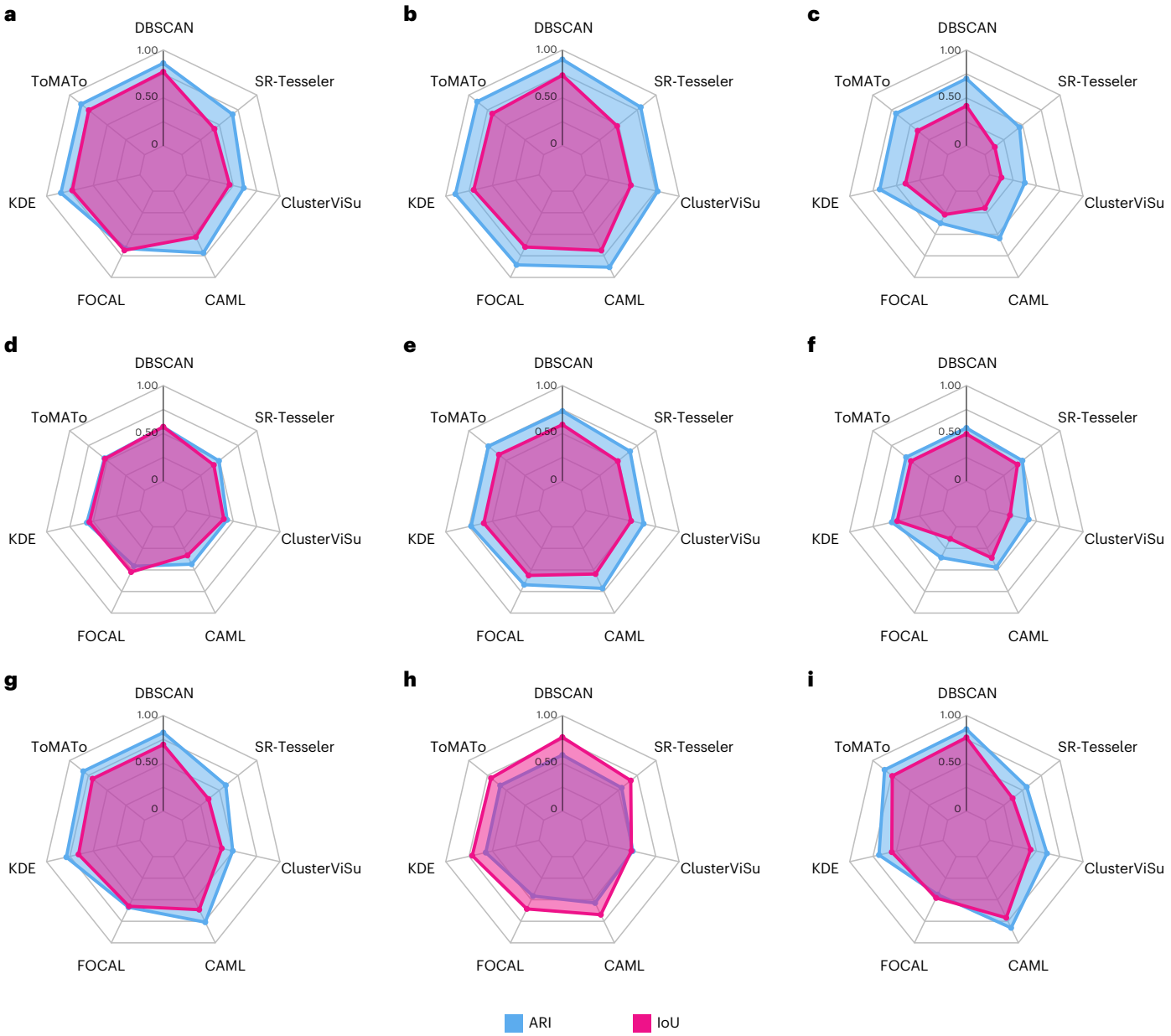

Fig. 3 | Evaluation of clustering algorithms against all ground truth scenarios. a–i, The average peak score for ARI and IoU for Scenario 2 (**a**), Scenario 3 (**b**), Scenario 4 (**c**), Scenario 5 (**d**), Scenario 6 (**e**), Scenario 7 (**f**), Scenario 8 (**g**), Scenario 9 (**h**) and Scenario 10 (**i**) for DBSCAN, ToMATo, KDE, FOCAL, SR-Tesseler, ClusterViSu and CAML.

The optimal clustering parameters for algorithms requiring user-defined parameters is summarized in Supplementary Table 1. The effect of this added blinking step on Scenario 2 is shown in Supplementary Figs. 10–12. The mean peak ARI and IoU scores for each algorithm are summarized for each condition with added fluorophore blinking in Fig. 4, with the distribution about these values shown in Supplementary Figs. 3–9c,d. There is an obvious drop off in performance for all algorithms compared with the case with no blinking. The stochastic nature of the blinking introduces greater heterogeneity into the data, which is demonstrated by the broadening of the peak metric score distributions for ARI, whereas IoU scores generally remain at similar levels. Decreasing the localization precision from $\sigma_{loc} \approx 15$ nm to $\sigma_{loc} \approx 30$ nm, in the case of Scenario 2, further decreases the performance of all algorithms according to both metrics, but with IoU being less affected (Supplementary Figs. 11–13). The framework thus guides how algorithms respond when multiple blinking is present in SMLM data. Given that optimal user parameters required for DBSCAN, ToMATo, KDE and FOCAL shift to higher values, a recommendation would be to err on the side of higher density cut-offs to improve algorithm performance. Interestingly, the distribution of parameters that gives the highest metric scores is widened (Supplementary Figs. 2 and 11), lessening the requirement for precise algorithm optimization.

Demonstration of an analysis pipeline for real, experimental SMLM data

Finally, we sought to develop an integrated analysis pipeline for users to follow when analyzing real, biological data (Fig. 5a). For this we acquired dSTORM data of fibroblast growth factor receptor 1 (FGFR1) distribution on MCF7 cells using a nanobody[41] labeled with Alexa Fluor 647. After using Ripley's K-function to confirm the presence of clustering, the next challenge is to determine which simulated condition most closely recapitulates the experimental data. To determine this, we

https://doi.org/10.1038/s41592-022-01750-6

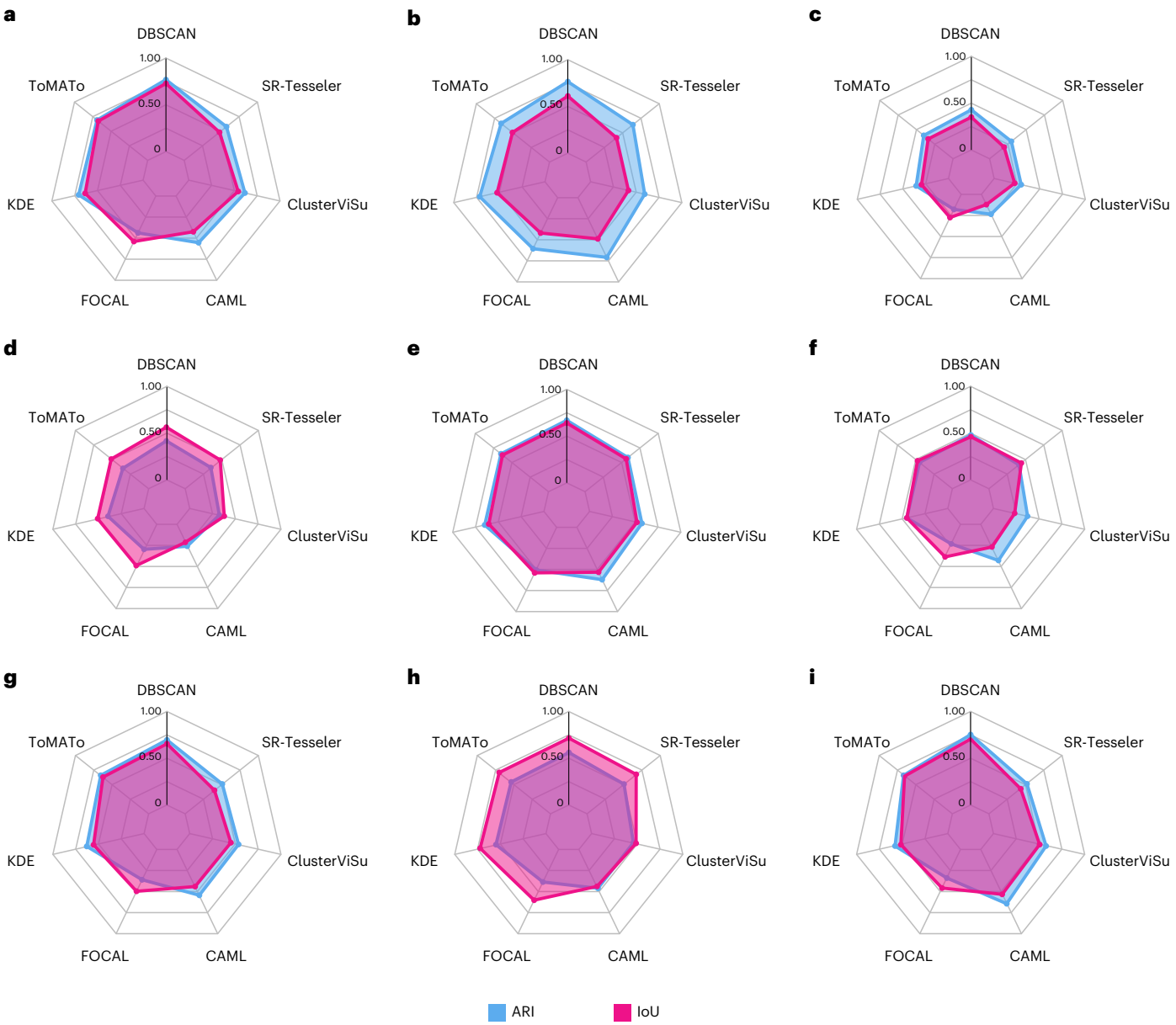

Fig. 4 | Evaluation of clustering algorithms against all conditions with added multiple blinking behavior. a–**i**, The average peak score for ARI and IoU for Scenario 2 (**a**), Scenario 3 (**b**), Scenario 4 (**c**), Scenario 5 (**d**), Scenario 6 (**e**), Scenario 7 (**f**), Scenario 8 (**g**), Scenario 9 (**h**) and Scenario 10 (**i**), with added multiple blinking, for DBSCAN, ToMATo, KDE, FOCAL, SR-Tesseler, ClusterViSu and CAML.

applied a new method for scoring the similarity of point pattern data by calculating a dissimilarity index[42]. The dissimilarity index was calculated for FGFR1 data in Scenarios 2–10 with multiple blinking present (Supplementary Table 2). Scenario 7 had the lowest dissimilarity score for the data, thus, the algorithm seen to perform best in this case was used. For Scenario 7, for the IoU metric both ToMATo and DBSCAN performed equally (Supplementary Figs. 3d and 4d and Supplementary Table 31), but for ARI the performance of DBSCAN was superior (Supplementary Figs. 3c, 4c and Supplementary Table 30). Therefore, DBSCAN was applied to the data with the optimal parameters determined from the peak metric scores for Scenario 7 (Supplementary Table 1 and Fig. 5). Using these parameters we were able to extract quantitative information on the clustering of the data, showing that the mean number of FGFR1 clusters per μm^2 was 2.63 ± 0.86 and the mean cluster size was 0.017 ± 0.013 μm^2 (Fig. 5c,d). Thus, this demonstrates the applicability of the framework to guide cluster analysis of non-simulated SMLM data.

Discussion

SMLM produces data in the form of a pointillist set of localization coordinates. A frequent goal is the cluster analysis of such data to extract quantitative information on molecular aggregation and, to this end, a wide variety of algorithms have been deployed[14]. Given that the development of new algorithms is ongoing, it is advantageous to have a framework in which the performance of these can be systematically evaluated. This is beneficial for both the developers of cluster analyses and the biologists seeking a guide to algorithm selection and optimization. Here, we provide such an environment in the form of standardized simulated conditions, defined performance metrics and a complete evaluation of some of the most widely used algorithms. Our main conclusions are as follows.

1. ARI is the most appropriate metric when clusters have a long tail to their distribution, for example, scenario 4. However, ARI operates best when the level of cluster overlap is low. Given that the IoU takes the *x*,*y* coordinates, as well as the labels, as input,

https://doi.org/10.1038/s41592-022-01750-6

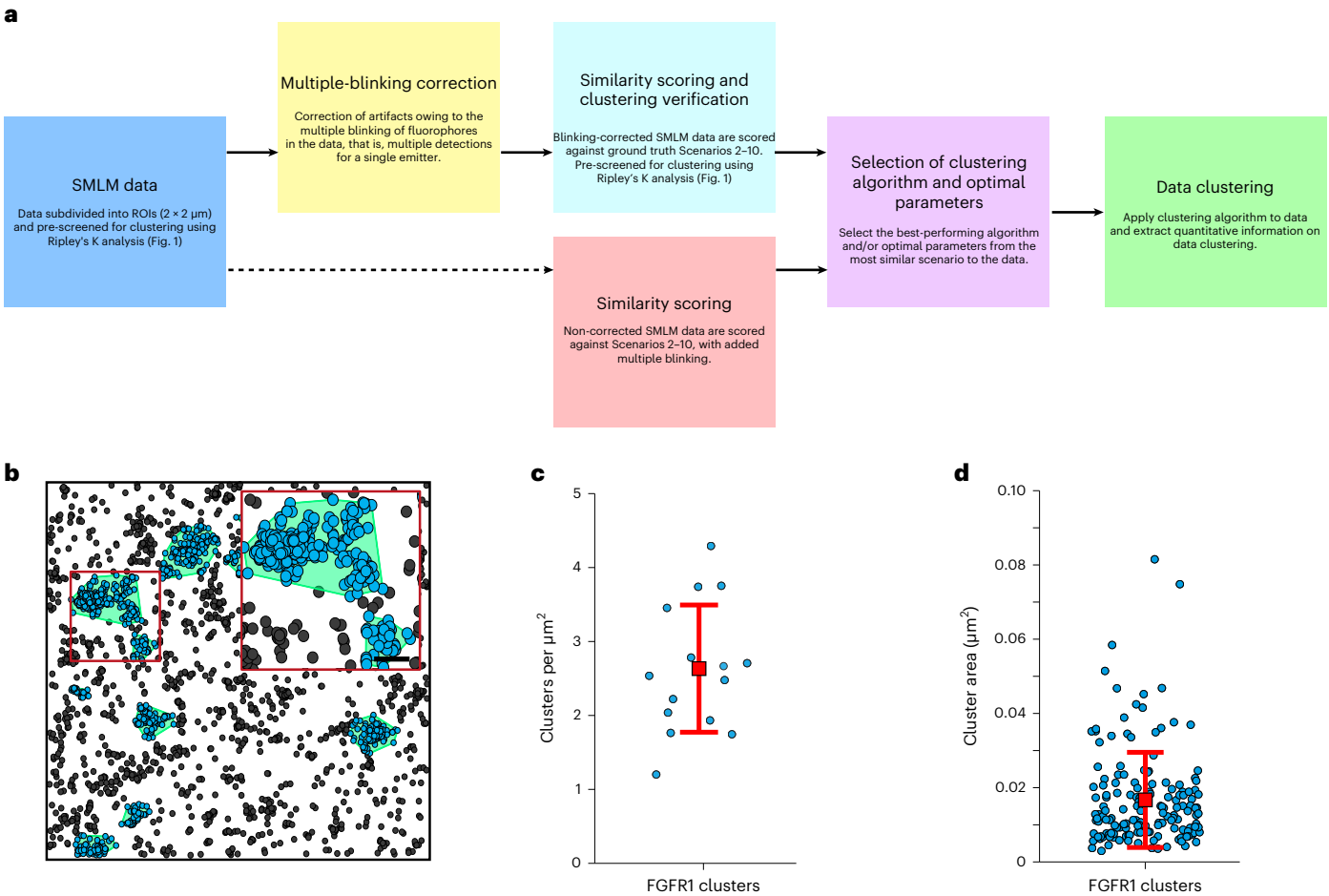

Fig. 5 | Use of the framework for clustering of non-simulated FGFR1 dSTORM data. a, Proposed framework for the clustering of non-simulated SMLM data. **b**, Example of a single region of interest (ROI) of FGFR1-nb-AF647 dSTORM on MCF7 cells clustered using DBSCAN, with optimal settings ($\varepsilon = 65$, minPts = 32) according to those found for Scenario 7 with multiple blinking ($n = 15$, two technical replicates). The inset is a zoom of the red boxed region. Scale bar, 100 nm. **c**,**d**, Number of clusters per square micrometer from the ROIs ($n = 15$, two technical replicates) (**c**) and area of segmented clusters (μm², n = 158) from the ROIs (**d**). The data points are shown in blue and the mean and standard deviation are shown in red.

the metric itself is dependent on the underlying distribution. It may therefore be less useful to compare IoU scores between regions of interest and simulation conditions than between algorithms.

2. Algorithms requiring the most user inputs (DBSCAN, ToMATo, KDE) are most adaptable to diverse clustering conditions and maintain good performance across the board, even in data subject to multiple blinking. This indicates that the ability to fine-tune the performance of these algorithms could be an advantage for SMLM cluster analysis for delivering accuracy, but this approach might hinder reproducibility between labs.
3. Analysis of the variance in performance between regions of interest in the same condition showed areas of relative stability at high scale and low density for DBSCAN, ToMATo and KDE, meaning that it may be safer to err on these sides if the best choice of analysis parameter is uncertain. Some performance may be sacrificed compared with attempting to locate the peak of performance, but results may be more consistent.
4. Scenario 4 (sparse data) was particularly challenging for algorithms with only one parameter, related to density, to set (ClusterViSu, FOCAL, SR-Tesseler), and Scenarios 5 (many clusters) and 7 (diverse cluster sizes) generally scored poorly across the board and users should, therefore, be wary of analyzing regions of interest containing very heterogeneous features.
5. DBSCAN was in general most robust to multiple blinking, therefore it should be considered, especially if no blink correction has been performed.

Overall, when combined with a measure of point-cloud similarity, we have developed an integrated pipeline that enables users to objectively select the most appropriate algorithm for their biological data. While this work assessed performance, there may be other considerations that dictate the choice of algorithm to be used. These include processing time, therefore we have provided timing data for the different algorithms tested in this work (Supplementary Table 38). We anticipate that the framework presented here can be used to evaluate the performance of present cluster analysis algorithms designed for SMLM, and to inform the development of future methodologies. In particular, the poor performance on sparse data suggests avenues for development in the context of low copy-number proteins or as a pathway to high-throughput or live-cell SMLM. Furthermore, the basis of both metrics, that is, point classification and geometric overlap, are easily extendable to 3D, thus widening the applicability of this framework.

Online content

Any methods, additional references, Nature Portfolio reporting summaries, source data, extended data, supplementary information, acknowledgements, peer review information; details of author contributions

and competing interests; and statements of data and code availability are available at https://doi.org/10.1038/s41592-022-01750-6.

References

1. Goyette, J. & Gaus, K. Mechanisms of protein nanoscale clustering. *Curr. Opin. Cell Biol.* **44**, 86–92 (2017).
2. Goyette, J., Nieves, D. J., Ma, Y. & Gaus, K. How does T cell receptor clustering impact on signal transduction? *J. Cell Sci.* **132**, jcs226423 (2019).
3. Prior, I. A., Muncke, C., Parton, R. G. & Hancock, J. F. Direct visualization of Ras proteins in spatially distinct cell surface microdomains. *J. Cell Biol.* **160**, 165–170 (2003).
4. Lukeš, T. et al. Quantifying protein densities on cell membranes using super-resolution optical fluctuation imaging. *Nat. Commun.* **8**, 1731 (2017).
5. Sauer, M. & Heilemann, M. Single-molecule localization microscopy in eukaryotes. *Chem. Rev.* **117**, 7478–7509 (2017).
6. Heilemann, M. et al. Subdiffraction-resolution fluorescence imaging with conventional fluorescent probes. *Angew. Chem. Int. Ed. Engl.* **47**, 6172–6176 (2008).
7. Rust, M. J., Bates, M. & Zhuang, X. Sub-diffraction-limit imaging by stochastic optical reconstruction microscopy (STORM). *Nat. Methods* **3**, 793–795 (2006).
8. Betzig, E. et al. Imaging intracellular fluorescent proteins at nanometer resolution. *Science* **313**, 1642–1645 (2006).
9. Sharonov, A. & Hochstrasser, R. M. Wide-field subdiffraction imaging by accumulated binding of diffusing probes. *Proc. Natl Acad. Sci. USA* **103**, 18911–18916 (2006).
10. Jungmann, R. et al. Single-molecule kinetics and super-resolution microscopy by fluorescence imaging of transient binding on DNA origami. *Nano Lett.* **10**, 4756–4761 (2010).
11. Jungmann, R. et al. Multiplexed 3D cellular super-resolution imaging with DNA-PAINT and Exchange-PAINT. *Nat. Methods* **11**, 313–318 (2014).
12. Nieves, D. J., Gaus, K. & Baker, M. A. B. DNA-based super-resolution microscopy: DNA-PAINT. *Genes (Basel)* **9**, 621 (2018).
13. Nieves, D. J. & Owen, D. M. Analysis methods for interrogating spatial organisation of single molecule localization microscopy data. *Int. J. Biochem. Cell Biol.* **123**, 105749 (2020).
14. Khater, I. M., Nabi, I. R. & Hamarneh, G. A review of super-resolution single-molecule localization microscopy cluster analysis and quantification methods. *Patterns (NY)* **1**, 100038 (2020).
15. Ripley, B. D. Modeling spatial patterns. *J. R. Stat. Soc. B Methodol.* **39**, 172–192 (1977).
16. Cover, T. M. & Hart, P. E. Nearest neighbor pattern classification. *IEEE Trans. Inform. Theory* **13**, 21–27 (1967).
17. van Leeuwen, J. M. J., Groeneveld, J. & de Boer, J. New method for the calculation of the pair correlation function. I. *Physica* **25**, 792–808 (1959).
18. Rossy, J., Owen, D. M., Williamson, D. J., Yang, Z. & Gaus, K. Conformational states of the kinase Lck regulate clustering in early T cell signaling. *Nat. Immunol.* **14**, 82–89 (2013).
19. Williamson, D. J. et al. Pre-existing clusters of the adaptor Lat do not participate in early T cell signaling events. *Nat. Immunol.* **12**, 655–662 (2011).
20. Bar-On, D. et al. Super-resolution imaging reveals the internal architecture of nano-sized syntaxin clusters. *J. Biol. Chem.* **287**, 27158–27167 (2012).
21. Razvag, Y., Neve-Oz, Y., Sajman, J., Reches, M. & Sherman, E. Nanoscale kinetic segregation of TCR and CD45 in engaged microvilli facilitates early T cell activation. *Nat. Commun.* **9**, 732 (2018).
22. Scarselli, M., Annibale, P. & Radenovic, A. Cell type-specific beta2-adrenergic receptor clusters identified using photoactivated localization microscopy are not lipid raft related, but depend on actin cytoskeleton integrity. *J. Biol. Chem.* **287**, 16768–16780 (2012).
23. Mollazade, M. et al. Can single molecule localization microscopy be used to map closely spaced RGD nanodomains? *PLoS One* **12**, e0180871 (2017).
24. Levet, F. et al. SR-Tesseler: a method to segment and quantify localization-based super-resolution microscopy data. *Nat. Methods* **12**, 1065–1071 (2015).
25. Andronov, L., Orlov, I., Lutz, Y., Vonesch, J. L. & Klaholz, B. P. ClusterViSu, a method for clustering of protein complexes by Voronoi tessellation in super-resolution microscopy. *Sci. Rep.* **6**, 24084 (2016).
26. Mazouchi, A. & Milstein, J. N. Fast Optimized Cluster Algorithm for Localizations (FOCAL): a spatial cluster analysis for super-resolved microscopy. *Bioinformatics* **32**, 747–754 (2016).
27. Williamson, D. J. et al. Machine learning for cluster analysis of localization microscopy data. *Nat. Commun.* **11**, 1493 (2020).
28. Pike, J. A. et al. Topological data analysis quantifies biological nano-structure from single molecule localization microscopy. *Bioinformatics* **36**, 1614–1621 (2020).
29. Griffié, J. et al. A Bayesian cluster analysis method for single-molecule localization microscopy data. *Nat. Protoc.* **11**, 2499–2514 (2016).
30. Rubin-Delanchy, P. et al. Bayesian cluster identification in single-molecule localization microscopy data. *Nat. Methods* **12**, 1072–1076 (2015).
31. Nieves, D. J. et al. The T cell receptor displays lateral signal propagation involving non-engaged receptors. *Nanoscale* **14**, 3513–3526 (2022).
32. Rand, W. M. Objective criteria for the evaluation of clustering methods. *J. Am. Stat. Assoc.* **66**, 846–850 (1971).
33. Hubert, L. & Arabie, P. Comparing partitions. *J. Classif.* **2**, 193–218 (1985).
34. Jaccard, P. The distribution of the flora in the alpine zone. 1. *New Phytologist* **11**, 37–50 (1912).
35. Tanimoto, T. T. *An Elementary Mathematical Theory of Classification and Prediction* (IBM, 1958).
36. Margalit, A. & Knott, G. D. An algorithm for computing the union, intersection or difference of two polygons. *Computers Graphics* **13**, 167–183 (1989).
37. Ester, M., Kriegel, H. P., Sander, J., Xiaowei, X. A density-based algorithm for discovering clusters in large spatial databases with noise. In *KDD-96 Proceedings* 226–231 (AAAI, 1996).
38. Chazal, F., Guibas, L. J., Oudot, S. Y. & Skraba, P. Persistence-based clustering in Riemannian manifolds. *J. ACM* **60**, 1–38 (2013).
39. Bohrer, C. H. et al. A pairwise distance distribution correction (DDC) algorithm to eliminate blinking-caused artifacts in SMLM. *Nat. Methods* **18**, 669–677 (2021).
40. Jensen, L. G. et al. Correction of multiple-blinking artefacts in photoactivated localization microscopy. *Nat. Methods* **19**, 594–602 (2022).
41. Monegal, A. et al. Immunological applications of single-domain llama recombinant antibodies isolated from a naive library. *Protein Eng. Des. Sel.* **22**, 273–280 (2009).
42. Baragilly, M., Nieves, D. J., Williamson, D. J., Peters, R. & Owen, D. M. Measuring the similarity of SMLM-derived point-clouds. Preprint at https://www.biorxiv.org/content/10.1101/2022.09.12.507560v1 (2022).

Methods

Ground truth cluster condition simulations

Simulations of ground truth molecule point patterns were generated using scripts written in R (available at https://github.com/JeremyPike/RSMLM). For simulations based on multivariate normal distributions, the cluster centers are randomly generated inside a square field of interest (here, equivalent to 2,000 × 2,000 nm). At each center, molecules are then placed around it according to the random multivariate distribution. For symmetrical clusters, a single value of 25 nm is used for the standard deviation of the multivariate distribution (unless stated otherwise, Table 1), whereas for the elliptically shaped clusters two different standard deviation values, for the minor and major axes, respectively (25 nm and 75 nm), are used. Furthermore, for elliptical clusters they are rotated by a random angle around the center of mass of the generated cluster. Each cluster generated in the simulations possesses a unique 'cluster index' value, that is, molecules from the same cluster will have the same index value. Background molecules are given the index value of 0. For each cluster condition, 50 simulations were generated. The different parameter values used to generate the simulations in this work are summarized in the table below.

Fluorescence blinking cluster simulations

The positions of molecules in the ground truth cluster conditions were used as the basis for simulating data that have the multiple fluorophore blinking and detection precision inherent in SMLM. The simulateSTORM.r script from the RSMLM package (available at https://github.com/JeremyPike/RSMLM) was used to generate the blinking simulations[28]. In brief, the transition between the fluorescent on and off states was modeled using a geometric distribution[23,28] with the probability of transition to the dark state set to 0.2, to generate on average 4–5 fluorescent on-states, and thus, detections per molecule. Blinking was applied to all molecules in the simulations, thus single background molecules were also prone to blinking here. Detections owing to a single molecule will all be assigned the index value of that molecule from the ground truth, for example, if detections are generated from a ground truth molecule with a cluster index of 5, then the detections will also retain the cluster index of 5. The localization uncertainty (σ_{loc}) for each blinking event was determined using a normal distribution centered on the molecule position. Standard deviation for localization uncertainty was set using a log-normal distribution with a mean of 2.8 ($\sigma_{loc} \approx 15$ nm) and 3.2 ($\sigma_{loc} \approx 30$ nm), both with a standard deviation of 0.28 (ref. [23]).

DBSCAN clustering and parameter scanning

The DBSCAN algorithm was implemented in R using the dbscan R package. For DBSCAN there are two parameters: ε, which is the radius of search around each point, and minPts, which is the minimum number of neighboring points within that radius that are required for the point to be assigned to the cluster. Points within ε of clustered points but failing to fulfill minPts are designated the edge of the cluster. For DBSCAN parameter scanning, the ε (nm) and minPts threshold were varied. For the ε values a minimum of 5 nm was used, and stepped by 5 nm up to a maximum value of 100 nm (20 steps). For the minPts threshold a minimum value of 2 was used and stepped by 1 up to a maximum value of 50 (49 steps). Therefore, for each simulation a total of 980 combinations of ε and minPts threshold were used, and the resulting indexing from each combination was retained for further analysis.

ToMATo clustering and parameter scanning

The ToMATo algorithm was implemented in R using the clusterTomato function from the RSMLM library (available at https://github.com/JeremyPike/RSMLM)[28]. For ToMATo parameter scanning, the search radius (nm) and birth density threshold were varied. For the search radius, a minimum of 5 nm was used and stepped by 5 nm up to a maximum value of 100 nm (20 steps). For the birth density threshold, a minimum value of 2 was used and stepped by 1 up to a maximum value of 50 (49 steps). Therefore, for each simulation, a total of 980 combinations of search radius and birth density threshold were used, and the resulting indexing from each combination was retained for further analysis.

Kernel density estimation clustering and parameter scanning

KDE was performed using the kde2d function from the MASS R package. A two-dimensional (2D) matrix was generated using the minimum and maximum dimensions from the simulation data, with each element in the matrix corresponding to a 1 nm^2 region. The simulation coordinates are then convolved with a 2D Gaussian kernel within this 2D matrix, and the densities in each of these 1 nm^2 regions after convolution are calculated. This 2D density matrix can then be thresholded according to a specific density value, and higher density regions above the cut-off are considered as clustered. These regions are then used to assign points from the real data into clusters. For KDE parameter scanning, the 2D Gaussian kernel width and the density threshold were varied. For the kernel width a minimum of 50 nm was used and stepped by 50 nm up to a maximum value of 500 nm (10 steps). For the density threshold a minimum value of 1×10^{-8} was used and stepped by 0.5×10^{-7} up to a maximum value of 0.96×10^{-6} (20 steps). Therefore, for each simulation, a total of 100 combinations of kernel size and density threshold were used, and the resulting indexing from each combination was retained for further analysis.

FOCAL clustering and parameter scanning

FOCAL clustering was performed using a custom-written script in R, following the method described in ref. [26]. In brief, the *x* and *y* coordinates from the simulations were discretized into a matrix in which each element corresponded to a 10 × 10 nm region. The number of points in the region was summed. The matrix was then convolved using a 3 × 3 square filter, that is, for a single element the sum was taken of that element and the surrounding eight elements, and the value entered into a new matrix. The convolved matrix was then used, and the minL threshold was applied (scanning from 1 to 20; 20 steps). Elements in the convolved matrix greater than or equal to the threshold were designated as core points, and elements with values greater than zero that share a boundary with a core point were designated as boundary points. All elements remaining from this process were taken as a mask, and masks that had more than nine elements (that is, a 3 × 3 array) were retained to filter the point sources. The surviving masks were used to determine clusters from *x* and *y* coordinates falling within these masks, with points indexed depending on the mask they fell into. These cluster indexes were retained for further analysis.

ClusterViSu clustering

Clustering was performed as described in ref. [25], using the ClusterViSu software package with automatic thresholding. The indexes of points in clusters after thresholding were retained for further analysis.

SR-Tesseler clustering and parameter scanning

Clustering was performed using the SR-Tesseler method from ref. [24]. For SR-Tesseler parameter scanning, the local density threshold was varied in the range 0.5–4.9 in 0.2 increments, constituting 22 steps, with the cluster indexes retained at each increment for further analysis.

CAML training and clustering

CAML was performed using scripts available from https://gitlab.com/quokka79/caml and run in Python (v3.8.5), as previously described[27]. Training datasets were generated according to the same conditions in Table 1 but across a 40 × 40 µm field of view to ensure that sufficient points could be collected to train a supervised machine learning model. Ten repeat images were generated for each scenario. The distances and relative angle from a point to its 1,000 nearest neighbors was determined for all points in all images and pooled together with the label (non-clustered or clustered). From this, 2.5×10^5 or 0.5×10^5 points of

each label were selected at random. The distance, angle and label data for each cluster scenario were then used to train a supervised machine learning model, for 5–100 epochs, specifically for that scenario. For the non-blinking datasets, the model layers were arranged and configured similarly to those already published. For the blinking datasets, an input layer was connected to three fully connected hidden layers (each with 1,000 units), and the last hidden layer output was flattened and connected to the final output layer to yield the point's predicted label. Trained models were then used to evaluate the cluster simulations data. Model-labeled points were grouped into 'like clusters' and the labels refined, for example to avoid clusters containing single points, as previously described. The predicted label for each point was appended to the input data file and saved for ARI and IoU analysis.

Adjusted Rand index scoring

Cluster index results from the clustering and parameter scanning are used to compute the ARI. The ARI calculation was implemented in R using the function mclustcomp from the dbscan library (the full ARI R script is available at https://github.com/DJ-Nieves/ARI-and-IoU-cluster-analysis-evaluation). In brief, the Rand index (RI) is a measure of the similarity between two sets of cluster indexes, and is calculated using

$$\mathrm{RI} = \frac{a+b}{\binom{n}{2}}$$

where a is the number of agreements between the ground truth and the cluster results, and b is the number of disagreements between the ground truth and the clustering results. $\binom{n}{2}$ is the binomial coefficient describing the number of unordered pairs between sets a and b.

The ARI corrects for the random chance of points being assigned to the correct clusters. For clusterings X and Y, the common elements between the two can be summarized using a contingency table:

X/Y	Y_1	Y_2	⋯	Y_j	Sums
X_1	n_{11}	n_{12}	⋯	n_{1s}	a_1
X_2	n_{21}	n_{22}	⋯	n_{2s}	a_2
⋮	⋮	⋮	⋱	⋮	⋮
X_i	n_{i1}	n_{i2}	⋯	n_{ij}	a_i
Sums	b_1	b_2	⋯	b_j	$\sum_{ij} n_{ij} = n$

Where n_{ij} is the number of elements in common between X_i and Y_j, a_i is the sum of the contingency table for row i, and b_j is the sum of the contingency table for column j. The ARI is calculated as follows;

$$\mathrm{ARI} = \frac{\sum_{ij}\binom{n_{ij}}{2} - \left[\sum_i \binom{a_i}{2}\sum_j \binom{b_j}{2}\right] \Big/ \binom{n}{2}}{\frac{1}{2}\left[\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2}\right] - \left[\sum_i \binom{a_i}{2}\sum_j \binom{b_j}{2}\right] \Big/ \binom{n}{2}}$$

For each parameter combination the clustering result (that is, the cluster indexes) is compared with that of the ground truth simulation, and the ARI is calculated for that parameter combination. For fluorescent blinking simulations data, the clustering results are compared with the cluster indexes generated by the simulateSTORM algorithm as the ground truth condition. This is repeated for all parameter combinations to generate an ARI matrix.

Intersection of union scoring

The IoU scoring was implemented in custom-written script in R (https://github.com/DJ-Nieves/ARI-and-IoU-cluster-analysis-evaluation). A convex hull was used to identify the molecule coordinates at the edge of each ground truth cluster. A filled polygon for each cluster was then generated inside a binary image matching the limits of the data (pixel area = 1 nm^2). All cluster images were then added together to generate a single image, and then the image was flattened to generate a combined binary image. This process was performed for the ground truth clustering as well as each of the clustering results from the cluster parameter scanning. IoU is calculated as follows:

$$\mathrm{IoU} = \frac{\text{Area of overlap between the polygons}}{\text{Combined area of the polygons}}$$

For the calculation from our binary images, the ground truth image was added to the cluster result image, thus giving a single image in which the overlapping pixels had a value of 2. The number of pixels with value > 0 were equal to the combined area (nm^2), whereas the number of pixels with value > 1 were equal to the area of overlap (nm^2). For fluorescent blinking simulations data, the clustering results are compared with an image generated form the convex hulls of cluster indexes generated by the simulateSTORM algorithm as the ground truth condition. This is repeated for all parameter combinations to generate an IoU matrix.

FGFR1 (C8) nanobody production and conjugation to AlexaFluor647

The sequence corresponding to the anti-FGFR1 nanobody C8 (ref. [41]) was cloned in a modified pET14 vector. The resulting constructs possess a 6xHis tag and a free cysteine at the carboxy terminus[43]. The nanobody fusion was produced and affinity purified as previously described[44]. Purified FGFR1 nanobody (500 nM) was incubated for 1 h with a 5× molar excess of Maleimide-Alexa Fluor 647 dye (2.5 µM, AF647, Life Technologies). Excess dye was removed with repeated rounds of Zeba spin column desalting (three 2 min spins at 1,000 g, ThermoFisher) yielding FGFR1 nanobody labeled with a single fixed AF647 dye in 1× PBS (FGFR1-nb-AF647).

dSTORM sample staining, imaging and data processing

MCF7 cells (A150645-1106; antibodies.com) were fixed with warm 4% (v/v) paraformaldehyde (PFA) in 1× PBS for 15 min at room temperature and then washed with 1× PBS. Fixed samples were blocked using 5% (w/v) bovine serum albumin (7030; Sigma Aldrich) in 1× PBS for 30 min, and then incubated with FGFR1-nb-AF647 (2 µg ml^{-1}) in 5% (w/v) bovine serum albumin for a further 30 min. Samples were then washed with 1× PBS to remove excess unbound FGFR1-nb-AF647 and a post-fixation step (4% PFA for 5 min) was used to preserve the labeled sample for later dSTORM imaging. dSTORM imaging was performed in an oxygen-scavenging buffer system (10 mM Tris with 50 mM NaCl, pH 8.0, supplemented with 20% (w/v) glucose, 0.8 mg ml^{-1} glucose oxidase (G2133; Sigma), 40 mg ml^{-1} catalase (C100; Sigma) and 10 mM cysteamine (30070; Sigma)), with samples mounted freshly with buffer just prior to imaging. dSTORM images were acquired with an ONI nanoimager using total internal reflection fluorescence (TIRF) imaging. Excitation using the 405 nm laser line was used for conversion of fluorophores from the dark state. Stochastic blinking of fluorophores was captured using an sCMOS camera with an integration time of 50 ms per frame. Images were taken until most of the fluorescence events inside the field of view had ceased, resulting in images stacks of approximately 10,000–20,000 frames. All images were 512 × 256 pixels with a pixel size of 117 nm.

Dissimilarity scoring of real data versus simulated data

A dissimilarity score was calculated according to the method in ref. [42]. A total of $n = 15$ (two technical replicates) experimental data regions of interest were compared, pairwise, with the same number of simulated regions of interest from every scenario (including multiple blinking) and the mean dissimilarity index was calculated.

Statistical analyses

Peak metric score distributions in a scenario from each algorithm were tested for normality using the Shapiro–Wilk normality test (stats package; R Studio). A non-parametric pairwise Wilcoxon rank sum test with Bonferroni correction for multiple comparisons was applied to determine whether the distributions emerge from the same population, with the null hypothesis rejected at a significance threshold of 0.05. The *P* values for the pairwise comparisons for each clustering algorithm in a scenario, for both peak metric score distributions, are presented in Supplementary Tables 3–37.

Reporting summary

Further information on research design is available in the Nature Portfolio Reporting Summary linked to this article.

Data availability

Both the simulation and the real SMLM data used as the basis for this work are available for download at https://github.com/DJ-Nieves/ARI-and-IoU-cluster-analysis-evaluation without restriction. Source data are provided with this paper.

Code availability

R code for calculating ARI and IoU for clustering results against a ground truth scenario is available for download at https://github.com/DJ-Nieves/ARI-and-IoU-cluster-analysis-evaluation without restriction.

References

43. Ambrosetti, E. et al. Quantification of circulating cancer biomarkers via sensitive topographic measurements on single binder nanoarrays. *ACS Omega* **30**, 2618–2629 (2017).
44. Veggiani, G. & de Marco, A. Improved quantitative and qualitative production of single-domain intrabodies mediated by the co-expression of Erv1p sulfhydryl oxidase. *Protein Expr. Purif.* **79**, 111–114 (2011).

Acknowledgements

D.M.O. acknowledges funding from BBSRC grant BB/R007365/1. M.H. acknowledges funding by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation, Project-ID 259130777, SFB 1177; GRK 2566). D.M.O. and M.B. acknowledge funding from the Alan Turing Institute.

Author contributions

D.J.N. wrote simulation and analysis code, produced simulations, performed cluster analyses, acquired dSTORM data and wrote the manuscript. J.A.P. wrote the simulation code. F.L. and D.J.W. performed analyses. M.B. performed dissimilarity measurements. S.O. and A.d.M. produced the FGFR1 nanobody. J.G., D.S., E.A.K.C., J.A.P., J.-B.S. and M.H. contributed ideas and concepts. D.M.O. conceived the work and wrote the manuscript. All authors contributed to the drafting and writing of the manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41592-022-01750-6.

Correspondence and requests for materials should be addressed to Dylan M. Owen.

Peer review information *Nature Methods* thanks Marek Cebecauer and the other, anonymous, reviewers for their contribution to the peer review of this work. Primary Handling Editor: Rita Strack, in collaboration with the *Nature Methods* team. Peer reviewer reports are available.

nature research

Corresponding author(s): Dylan M. Owen

Last updated by author(s): Nov 23, 2022

Reporting Summary

Nature Research wishes to improve the reproducibility of the work that we publish. This form provides structure for consistency and transparency in reporting. For further information on Nature Research policies, see our Editorial Policies and the Editorial Policy Checklist.

Statistics

For all statistical analyses, confirm that the following items are present in the figure legend, table legend, main text, or Methods section.

n/a	Confirmed	
☐	☒	The exact sample size (*n*) for each experimental group/condition, given as a discrete number and unit of measurement
☒	☐	A statement on whether measurements were taken from distinct samples or whether the same sample was measured repeatedly
☐	☒	The statistical test(s) used AND whether they are one- or two-sided *Only common tests should be described solely by name; describe more complex techniques in the Methods section.*
☒	☐	A description of all covariates tested
☒	☐	A description of any assumptions or corrections, such as tests of normality and adjustment for multiple comparisons
☐	☒	A full description of the statistical parameters including central tendency (e.g. means) or other basic estimates (e.g. regression coefficient) AND variation (e.g. standard deviation) or associated estimates of uncertainty (e.g. confidence intervals)
☐	☒	For null hypothesis testing, the test statistic (e.g. *F*, *t*, *r*) with confidence intervals, effect sizes, degrees of freedom and *P* value noted *Give P values as exact values whenever suitable.*
☒	☐	For Bayesian analysis, information on the choice of priors and Markov chain Monte Carlo settings
☒	☐	For hierarchical and complex designs, identification of the appropriate level for tests and full reporting of outcomes
☒	☐	Estimates of effect sizes (e.g. Cohen's *d*, Pearson's *r*), indicating how they were calculated

Our web collection on statistics for biologists contains articles on many of the points above.

Software and code

Policy information about availability of computer code

Data collection	ONI Nanoimager software v1.19.4, FIJI (ImageJ) v2.9.0/1.53t, ThunderSTORM ImageJ plugin (dev-2018-04-13-pSMLM)
Data analysis	Custom R code available for metrics: https://github.com/DJ-Nieves/ARI-and-IoU-cluster-analysis-evaluation ClusterViSu 1.1.2: https://github.com/andronovl/SharpViSu/tree/master/Installer TOMATO: https://github.com/JeremyPike/RSMLM CAML: https://gitlab.com/quokka79/caml DBSCAN: dbscan R library v1.1-5: https://cran.r-project.org/web/packages/dbscan/index.html KDE: https://github.com/DJ-Nieves/ARI-and-IoU-cluster-analysis-evaluation FOCAL: https://github.com/DJ-Nieves/ARI-and-IoU-cluster-analysis-evaluation SR-TESSELER: https://github.com/flevet/SR-Tesseler Ripley's K-function: Spatstat R libarary v1.63-0 R: v3.6.1 RStudio 1.2.1335 simulateSTORM.r from RSMLM: https://github.com/JeremyPike/RSMLM ROI similarity tool: https://www.biorxiv.org/content/10.1101/2022.09.12.507560v1.supplementary-material

For manuscripts utilizing custom algorithms or software that are central to the research but not yet described in published literature, software must be made available to editors and reviewers. We strongly encourage code deposition in a community repository (e.g. GitHub). See the Nature Research guidelines for submitting code & software for further information.

Data

Policy information about availability of data

All manuscripts must include a data availability statement. This statement should provide the following information, where applicable:

- Accession codes, unique identifiers, or web links for publicly available datasets
- A list of figures that have associated raw data
- A description of any restrictions on data availability

All simulated and experimental data are available without restriction at https://github.com/DJ-Nieves/ARI-and-IoU-cluster-analysis-evaluation

Field-specific reporting

Please select the one below that is the best fit for your research. If you are not sure, read the appropriate sections before making your selection.

☒ Life sciences ☐ Behavioural & social sciences ☐ Ecological, evolutionary & environmental sciences

For a reference copy of the document with all sections, see nature.com/documents/nr-reporting-summary-flat.pdf

Life sciences study design

All studies must disclose on these points even when the disclosure is negative.

Sample size	No sample size calculation was performed. Sample size was determined to approximately match typical sample sizes found in the literature
Data exclusions	None
Replication	2 technical replicates were performed, data was consistent between replicates
Randomization	No randomization as no comparison of experimental data sets is made
Blinding	No blinding as no comparison of experimental data sets is made

Reporting for specific materials, systems and methods

We require information from authors about some types of materials, experimental systems and methods used in many studies. Here, indicate whether each material, system or method listed is relevant to your study. If you are not sure if a list item applies to your research, read the appropriate section before selecting a response.

Materials & experimental systems

n/a	Involved in the study
☒	☐ Antibodies
☐	☒ Eukaryotic cell lines
☒	☐ Palaeontology and archaeology
☒	☐ Animals and other organisms
☒	☐ Human research participants
☒	☐ Clinical data
☒	☐ Dual use research of concern

Methods

n/a	Involved in the study
☒	☐ ChIP-seq
☒	☐ Flow cytometry
☒	☐ MRI-based neuroimaging

Eukaryotic cell lines

Policy information about cell lines

Cell line source(s)	MCF7 from Antibodies.com A150645-1106
Authentication	Not authenticated
Mycoplasma contamination	Not tested
Commonly misidentified lines (See ICLAC register)	None (MCF-7 not listed)