IET Systems Biology

Research Article

IET Journals
The Institution of Engineering and Technology

Comparison of clustering approaches with application to dual colour protein data

ISSN 1751-8849
Received on 24th March 2017
Revised 5th July 2017
Accepted on 14th August 2017
E-First on 18th October 2017
doi: 10.1049/iet-syb.2017.0019
www.ietdl.org

Sabrina Siebert[1] ✉, Katja Ickstadt[1], Martin Schäfer[2], Yvonne Radon[3], Peter J. Verveer[3]

[1]*Faculty of Statistics, TU Dortmund University, Dortmund, Germany*
[2]*Chair of Mathematical Optimization, Mathematical Institute, Heinrich Heine University, Düsseldorf, Germany*
[3]*Max-Planck-Institute Dortmund, Dortmund, Germany*
✉ *E-mail: siebert@statistik.tu-dortmund.de*

Abstract: Cells communicate with their environment via proteins, located at the plasma membrane separating the interior of a cell from its surroundings. The spatial distribution of these proteins in the plasma membrane under different physiological conditions is of importance, since this may influence their signal transmission properties. In this study, the authors compare different methods such as hierarchical clustering, extensible Markov models and the gammics method for analysing such a spatial distribution. The methods are examined in a simulation study to determine their optimal use. Afterwards, they analyse experimental imaging data and extend these methods to simulate dual colour data.

1 Introduction

Cluster analysis is a widely used approach for many types of data and applications. This is also represented by the high number of available literature. For example, the PubMed Health [1] search finds 111,848 articles with the keyword cluster analysis.

One application field is genomic data. Here, the aim of the cluster analysis is finding genes with similar functions or genes also which cause the same type of disease. Often gene expression data is employed to find these groups, as described, e.g. in Eisen *et al.* [2] and Sunaga *et al.*[3]. Here, we analyse clusters (and their behaviour) of similar proteins located in the cell membrane, which is also of high interest, see, e.g. Arnau *et al.*[4] or Manley *et al.* [5].

In mammalian cells, the plasma membrane separates the interior from the cell's environment. The cell communicates with the environment via proteins that are embedded in, or attached to, the plasma membrane. The spatial distribution of these proteins is thought to play an important role in signal transmission via the plasma membrane. For instance, the small GTPase Ras forms clusters of about seven molecules with a diameter of around 20 nm, which are thought to be important for the robust transduction of signals into the cell. However, not all proteins form clusters, only a part and the remaining proteins form the background. Therefore, it is of considerable interest to be able to analyse the spatial distribution of such signalling molecules under different physiological conditions, for instance in the presence or absence of hormones that might stimulate a cellular response such as migration or cell division. In recent years, fluorescence microscopy has developed to the point that individual molecules can be imaged in the plasma membrane of cells growing on a glass surface. There are different super-resolution microscopy techniques such as universal Point Accumulation for Imaging in Nanoscale Topography (uPAINT) (Greg and Hosy [6]) or binding activated localization microscopy (BaLM) (Burnette *et al.* [7]). Furthermore, there is mCherry which can also be used for two-colour fluorescence microscopy, see, e.g. Subach *et al.* [8]. Here, we will use data which is obtained by photo-activated localisation light microscopy (PALM) (Betzig *et al.* [9]) and Stochastic Optical Reconstruction Microscopy (STORM) (Huang *et al.* [10]). When using PALM, proteins can be localised by using rapidSTORM (a preprocessing algorithm) with an accuracy of around 20 nm at the bottom plasma membrane of the cell. The result is a list of protein localisations that can be analysed employing clustering or spatial statistical approaches, with the aim of finding and quantifying clusters.

To detect these protein clusters, one can choose different methods of the extensive toolbox of cluster methods, where some are standard methods for such data and some are common cluster methods. One standard method is Ripley's K-function including all transformations such as the L-function (Ripley [11]) which estimates the cluster radius. Another method based on the K-function is the Getis–Franklin method (see, e.g. Rubin-Delanchy *et al.* [12]). Other standard methods are, for example, pair- or cross-correlation (Sengupta *et al.* [13]), the SR-Tesseler method which uses Voronoï diagrams (Levet *et al.* [14]) or the density-based spatial clustering of applications with noise (DBSCAN) (Ester *et al.* [15]). These methods are more for cluster identification.

However though standard, non-spatial cluster methods could be used to find these protein clusters, e.g. hierarchical clustering (see, e.g. Hartigan [16], Jain *et al.* [17] or Kaufman and Rousseeuw [18]). In this paper, we will focus on the proportion of proteins in clusters and will show that standard methods as well as cluster methods will estimate that proportion well, but there can be differences in computation time or simplicity of application (due to parameter choices).

Therefore, we will compare the following five methods for the analysis of PALM data of Ras proteins: the standard hierarchical clustering, a graphical method named average-shifted histogram (ASH, see Scott and Sain [19]), the model-based extensible Markov models (EMMs, see Dunham *et al.* [20]), the standard clustering method DBSCAN (Ester *et al.* [15]) as well as the gammics methods (Schäfer *et al.*, [21]). We chose these five methods to get a survey of different methods, thus we use a standard cluster method (hierarchical clustering), a graphical method (ASH), a Bayesian method (EMM), a standard method for spatial clustering of microscopie data (DBSCAN) and a new approach especially developed for this application field (the gammics method). Finally, we will propose in this paper a new approach on how to combine the different (known) methods in an efficient way and develop a scheme for this procedure. Owing to this scheme, one first choose regions of interest (ROIs) for finding suitable parameter estimates, then applying faster methods for the whole cell or even larger regions. This leads to a considerable reduction of computation time. In addition, we not only adapt this new approach to experimental data but also to (simulated) dual colour data.

Table 1 Simulation parameters

Parameter		Values
overall proportion of points in clusters	p	0.4, 0.8
mean cluster size	μ	4, 8
mean cluster radius	r	15, 30
overall point density	λ	125
proportion of points in metaclusters	$\breve{p}$	0
detection error	σ	20

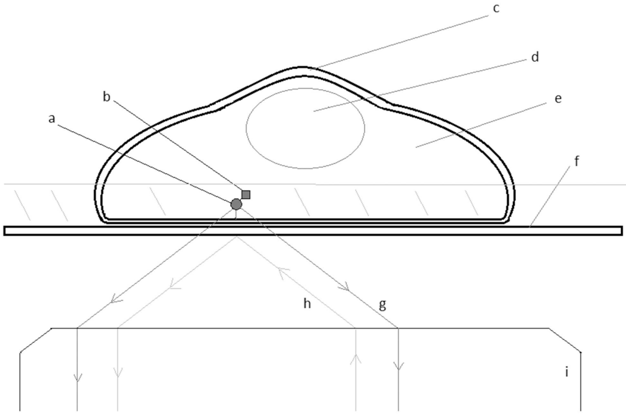

Fig. 1 *Schematic representation of the cell construction and the experimental set-up*
(a) Ras protein in the plasma membrane, ***(b)*** mEos2 (photoactivatable fluorescent protein), **(c)** Plasma membrane, ***(d)*** Cell nucleus, ***(e)*** Cytoplasm, ***(f)*** Object slide, ***(g)*** Emission light, which is reflected by the fluorophore, ***(h)*** Light for stimulation, ***(i)*** Objective/object lens

This paper is structured as follows: in Section 2, we introduce the simulation study as well as the experimental data, concluding with a short description of the different methods. In Section 3, we will compare the different methods on simulated single colour data and develop a new approach on how to combine the different methods to analyse such protein data. In Section 4, we will employ the new approach to experimental single colour data as well as in a dual colour simulation study. Finally, Section 5 contains concluding remarks including a short outlook.

2 Material and methods

In this section, an introduction to the simulation study as well as a description of the experimental data are given. Furthermore, we shortly describe the methods.

2.1 Simulation study

We conduct a simulation study similar to the one described in Schäfer *et al.* [21], which is based on a generalisation of a Matérn cluster process. We adapted this approach to simulate dual colour data.

In these simulations not all points (corresponding to proteins) are clustered, but a given fraction of the points are the so-called monomers or singletons. In a first step, parent points are created using a Poisson process. Normally, in a Matérn cluster process, each parent point would be replaced by a cluster of radius r, but here only a fraction of parent points is replaced. Thus, we create an image of points, where some of which are in a cluster and some not.

For a single colour simulation, we create only one data set with the parameters given in Table 1. An example for such a simulation is shown in Fig. 2. For a simulation of dual colour data, we create two of these data sets, one for each protein. The first one contains the green proteins and the second one the red ones, where the two different colours are a result of the different tagging. These two data sets are merged into a single one considering three different scenarios:

$S1$: The proteins tagged with the two colours are independent: the two data sets are simulated separately.

$S2$: The cluster centres of the two proteins are correlated: the cluster centres of the first data set will be shifted and used as cluster centres for the second data set. The monomers are simulated separately.

$S3$: The clustered proteins are correlated: the clustered points are translated and the monomers are again simulated separately.

2.2 Experimental data

The cell communicates via the cell membrane and signalling proteins with its environment. To understand the communication of a cell, it is important to analyse the signalling proteins, e.g. Ras one important signalling protein, which can be located (amongst others) in the cell membrane. For measuring these proteins in the cell membrane, the experimental set-up is shown in Fig. 1. Here, images of a cell expressing Ras tagged with mEos2 (Ras-mEos2) in the basal plasma membrane (see Figs. 1*a–c*) were obtained by total internal reflection fluorescence (TIRF) microscopy. The TIRF microscopy ensures that only the proteins located in the cell membrane are measured. The data of the Ras-mEos proteins are obtained by photo-activated localisation microscopy (PALM, Betzig *et al.* [9]), an imaging method that can be summarised as follows: the fluorescent tag mEos2 (see Fig. 1*b*) is green in its native state, but can be converted to a red form by ultraviolet (UV) light (see Fig. 1*h*). Using an UV laser, a sparse subset of the mEos2 fluorophores can be converted to the red form, which can then be imaged using TIRF microscopy as individual spots. The spots result from reflected light bundles from the fluorophore. Thereby, the light reflects in all directions, but only the light which is caught by the objective is seen on the image (see Fig. 1*g*). These spots then can be localised with a precision of around 20 nm by fitting a two-dimensional (2D) Gaussian profile to each spot in the image. Prolonged imaging of the red form of mEos2 leads to bleaching of the red fluorophores, and eventually these spots are not visible anymore. At this point, a new set of fluorophores is photo-converted and localised by the same procedure. This is done repeatedly until all proteins are imaged, resulting in a list of protein localisations.

2.3 Statistical methods

As mentioned in Section 1, we will compare and combine five different methods, where some of them are spatial and/or model based. Table 2 only gives a short overview of the methods we use to analyse the simulated as well as the experimental data. A detailed description is given in the supplement in Appendix 1.

3 Results of a first simulation study

In this section, we will employ the methods first for one setting of the single colour simulation study. More precisely, in a first step we will have a look at the method performance for one simulated single colour image and the estimation of the proportion of clustered points for the single colour simulation study. In a second step, we will investigate the performance of methods and their parameters, which worked well for the first simulated setting, on a pool of different settings. Here, we can see if the methods also work well for other proportions of clustered proteins as well as other setting parameters. Finally, we will discuss the results and develop a new approach for analysing spatial protein data. Most computations run on R [22] with the packages **cluster**, **ash**, **rEMM**, **spatstat** and **fpc** [23–28], only the gammics method operates on MATLAB (7.10.0 [29]).

3.1 Method performance

In a first step, we will consider the performance of the methods we introduced in Section 2.3 for a single colour data set. The data set was simulated with an overall proportion of points in clusters of 40%, a mean cluster size of 4 points, a mean cluster radius of 15 nm, an overall point density of 125 points/μm^2 and a detection

Table 2 Short description of the used methods and specification of the corresponding literature; detailed description is given in the supplement Appendix 1

Method	Literature	Short description
hierarchical clustering	Hartigan [16]	• groups data into homogenous groups • puts similar objects into one cluster, where similarity is defined by a distance measure • e.g. silhouette width may be used to get optimal number of clusters
ASH	Scott and Sain [19]	• data divided into *k* bins $B_k = [t_k, t_{k+1})$, where c_k indicates number of observations in B_k and *n* is number of total observations • if all bins are equally spaced with width ω, density histogram is given by $\hat{f}(x) = \frac{c_k}{nh}$ • how to choose t_0 and ω? → assume t_0 as a nuisance parameter → *m* SHs, which are all shifted by distance $\delta = \frac{h}{m}$ ⇒ $B_k^\star = [t_0 + k\delta, t_0 + (k+1)\delta)$ with corresponding c_k's · then ASH is given by $\hat{f}_{\mathrm{ASH}} = \frac{1}{nh}\sum_{i=1-m}^{m-1}\left(1 - \frac{\lvert i \rvert}{m}\right)c_{k+i} \quad \forall x \in B_k^\star$
EMM	Dunham *et al.* [20]	• EMM is a Markov chain which can vary over time → can be interpreted as a directed graph • for every point of time, the EMM consists of Markov chain with nodes N_n and algorithm that alters the Markov chain → algorithm consists of the following three steps: EMMCluster (matching incoming data to existing cluster, EMMIncrement (updating Markov chain, e.g. adding nodes) and EMMDecrement (removing nodes if necessary)
DBSCAN	Ester *et al.* [15]	• algorithm for finding clusters in a data set $\{x_1, \ldots, x_n\}$ using a key parameter based on distances between two points • to that end: Ester *et al.* introduced the theory of density-reachability and density connection w.r.t. two parameters: → ϵ and *MinPts*
gammics	Schäfer *et al.* [21]	• finding clusters in point patterns via a model-based algorithm • the gammics method models squared distances between points and their second-nearest neighbour based on a mixture of two gamma distributions, not point pattern itself • hierarchical Bayesian model and MCMC methods are employed to compute mean size, mean radius and proportion of clustered proteins

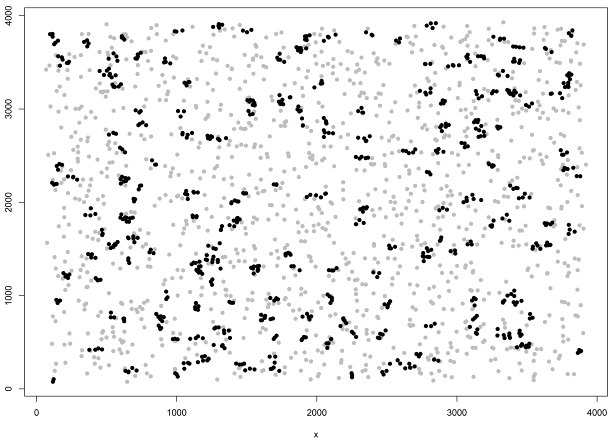

Fig. 2 *Simulated image with overall proportion of points in clusters of 40%, a mean cluster size of 4 points, a mean cluster radius of 15 nm, an overall point density of 125 points/μm^2 and a detection error of 20 nm (the black points are clustered ones)*

error of 20 nm. The resulting distribution is shown in Fig. 2. The data contain 195 clusters (mean proportion of clustered points of ca. 39.2%) and 1976 points altogether. First, we employ classical hierarchical clustering with average linkage. To find the optimal number of clusters, the silhouette width is used. The resulting dendrogram and curve of the silhouette width is shown in Fig. 3. It can be seen that the optimal number of clusters is 451. If we cut the dendrogram such that the resulting number of clusters is 451, we only get 34 singletons and an estimated proportion of proteins in clusters of 98.3%, much higher than the true value of 40%. If we assume that all clusters with at most five points are a singleton, we achieve 331 singletons and an estimated proportion of 83.2%, which is still too high. To solve this problem, we tried to get prior information about the number of clusters by the use of ASHs. The resulting plot of ASH is shown in Fig. 4. With this plot in mind, we guessed that there are round about 150 clusters in this data set. If we assume that the average number of proteins in a cluster is 5 (received as prior information), there are 1976−(150×5) = 1226 singletons. With these prior information the estimated proportion is ca. 37.9%. The resulting clustering is shown in Fig. 1 in the supplement Appendix 2. In summary, it is difficult for hierarchical clustering to distinguish between singletons and clustered proteins, though the proportion is fitted well. Note that hierarchical clustering is mainly used for classification of the points into the clusters here, and that the estimation of the proportion is computed without using hierarchical clustering. We just use the prior knowledge of an average number of 5 points in a cluster and 150 (300) clusters for 40% (80%) points in clusters, estimated by ASH.

For the EMM, a threshold has to be chosen which refers to the dissimilarity and determines if a point is associated to a cluster or not. Thus, here it represents some kind of size of the clusters and we decided to use two different thresholds: 25 and 55 (small and medium cluster sizes). The estimated proportion of singletons for the EMM is the number of singletons divided by the total number of points, thus the proportion of clustered proteins is one minus the proportion of singletons. For a threshold of 25, the estimated proportions of clustered proteins are ca. 35.2% and for a threshold of 55, the estimated proportion is ca. 70.9%. Hence, a threshold of 25 works better than 55 and is roughly the diameter. In Fig. 2, in the supplement Appendix 2 the resulting clustering is shown. It can be seen that it is also difficult for EMM to differentiate between singletons and clustered proteins by EMM. However, the clustering with a threshold of 25 fits the data better than the clustering with a threshold of 55. In summary, EMM depends on prior information of the value of the threshold and when knowing the radius or the diameter we can choose a good value for the threshold.

To obtain prior information about the radius, one can employ Ripley's K-function. For our data the K-function overestimated the radius, such as we did not pursue this approach.

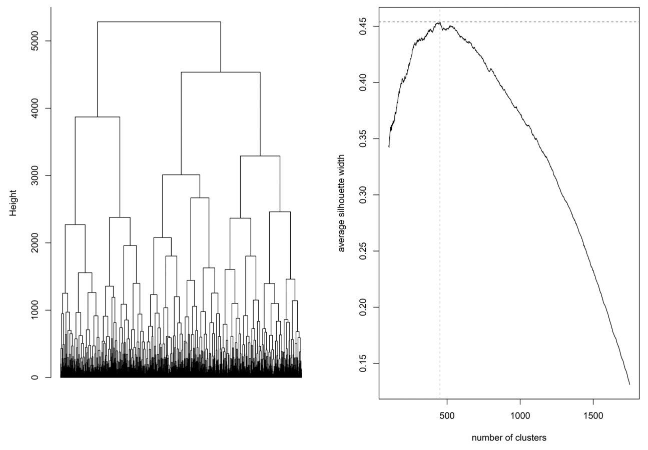

Fig. 3 *Left: dendrogram of the hierarchical clustering with average linkage and right: curve of the average silhouette width values*

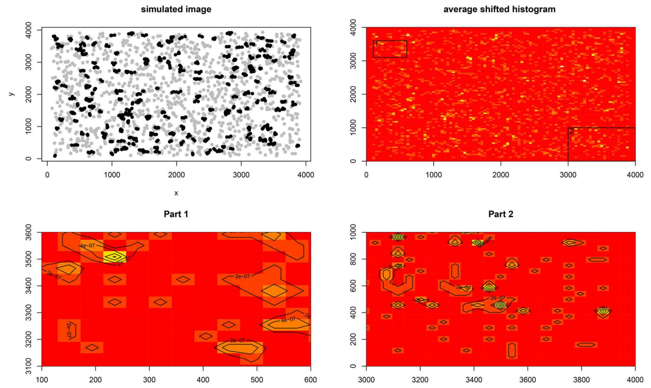

Fig. 4 *Upper left: simulated cell image [same as in Fig. 2 with clustered points (black dots) and background points (grey dots); upper right: contour plot of the ASH for the simulated data; bottom left: contour plot of the ASH for part 1, the marked rectangle (top left) in the upper right contour plot; bottom right: contour plot of the ASH for part 2, the marked rectangle (down right) in the upper right contour plot*

Table 3 Estimated proportion of points in clusters for DBSCAN

Value of ϵ	Estimated proportion of points in clusters, %
25	2.6
50	30.6
75	60.9
100	83.8

Furthermore, we analyse the simulated data using the DBSCAN method. Analogue to the EMM we have to choose again parameters, here ϵ and *MinPts*. Here, we use the default for *MinPts* which is 5, whereas choose four different values for the parameter ϵ which determines whether two points are in one neighbourhood (see also Section 8.4 of Appendix 1): 25, 50, 75 and 100. Owing to the default value for *MinPts*, we decided to use also higher values for ϵ than the threshold of the EMM. The resulting clustering is shown in Fig. 3 in the supplement Appendix 2. As one can see, it fits best for $\epsilon = 50$ which is similar to the estimated proportion of clustered proteins, see Table 3.

Finally, we apply the gammics method for analysing the simulated data. Here, three characteristics: the proportion, the mean radius and the mean size of clusters can be estimated simultaneously. For this example, we get an estimated proportion of points in clusters of ca. 41.5%, a mean radius of 9.9 and a mean size of 3.7. Considering the given simulation adjustment values for the proportion of 40%, for the radius of 15 (where the simulation resulted in a mean cluster radius of 9.9) and a cluster size of 4, the gammics method works well.

3.2 Proportion of clustered points for different simulation parameter settings

Now, all methods with a good performance and good working parameters are adopted for different simulation settings. Therefore, the DBSCAN method with $\epsilon = 25$ is excluded.

For each simulation setting, we simulate four images, where only three parameters (left column) are variable, p, μ and r. In Table 4, the average estimated proportions of points in clusters for the different methods and parameter settings of all simulations are shown.

The remaining columns contain the following estimators:

- $\hat{p}_{\text{prior}}$: The estimated proportion of clustered points, where we assume 150 clusters for $p = 0.4$ and 300 for $p = 0.8$.
- $\hat{p}_{\text{ash}}$: The estimated proportion of clustered points for hierarchical clustering, where we determine the number of clusters by using ASH.
- $\hat{p}_{\text{EMM25}}$: The estimated proportion of points in clusters by using the EMM with a threshold of 25.
- $\hat{p}_{\text{EMM55}}$: The estimated proportion of points in clusters by using the EMM with a threshold of 55.
- $\hat{p}_{\text{db50}}$: Estimated proportion of clustered points using DBSCAN with $\epsilon = 50$.
- $\hat{p}_{\text{db75}}$: The estimated proportion of clustered points using DBSCAN with $\epsilon = 75$.
- $\hat{p}_{\text{db100}}$: The estimated proportion of clustered points using DBSCAN with $\epsilon = 100$.
- $\hat{p}_{\text{gammics}}$: The estimated proportion of clustered points using the gammics method.

Again, note that DBSCAN with $\epsilon = 25$ did not work well in the example before.

The estimated proportions $\hat{p}_{\text{prior}}$ are near the true values if we use 150 and 300 clusters as prior knowledge, respectively. For a low proportion of points in clusters (here 40%) the 'hierarchical clustering' also works well, if prior information is provided by using ASH (i.e. the estimated number of clusters). However, it can be difficult to estimate the number of clusters for a higher proportion of points in clusters (Table 1 in supplement Appendix 3 contains all estimates by ASH). In Fig. 4, in the supplement Appendix 2 the resulting plot for a setting with proportion of 40% on the left and with 80% on the right is shown. One can see that the number of clusters on the right seems to be at most as high as on the left. In case of the EMM method, the estimated values are close to the true proportions for a good choice of the threshold, e.g. for settings with a proportion of clustered points of 40%, the estimated values fit well for a threshold of 25, whereas the estimated proportions fit well in settings with 80% points in clusters for a threshold of 55.

The DBSCAN method works well, but depends on the choice of the parameter ϵ. For example, DBSCAN with $\epsilon = 50$ works well for all settings with $p = 0.4$, with the exception of setting $p = 0.4$, $\mu = 4$, $r = 30$. For a simulated proportion of 80% points in clusters, $\epsilon = 75$ works well (just as well as $\epsilon = 50$ for the settings with $p = 0.8$ and $\mu = 8$).

Furthermore, note that there is a change in the estimated proportion for the DBSCAN method with $\epsilon = 50$ whether the mean size is 4 or 8.

The gammics method estimates the proportion of points in clusters very well. Also, the evaluated values of the mean radius and the mean size are similar to the realised values in the simulations.

3.3 Discussion and new approach

We analyse simulated data and as a result one can say that ASH, the EMM and DBSCAN work well for estimating the proportion of points in clusters if provided prior information (e.g. total number of clusters) is available, or if the (tuning) parameters of the procedure are chosen properly, whereas the gammics method works well even

Table 4 Average estimations of the proportion of points in clusters

Simulation parameter	$\hat{p}_{\mathrm{prior}}$	$\hat{p}_{\mathrm{ash}}$	$\hat{p}_{\mathrm{EMM25}}$	$\hat{p}_{\mathrm{EMM55}}$	$\hat{p}_{\mathrm{db50}}$	$\hat{p}_{\mathrm{db75}}$	$\hat{p}_{\mathrm{db100}}$	$\hat{p}_{\mathrm{gammics}}$
$p = 0.4$, $\mu = 4$, $r = 15$	0.37	0.35	0.37	0.74	0.34	0.64	0.86	0.41
$p = 0.4$, $\mu = 4$, $r = 30$	0.37	0.32	0.37	0.73	0.32	0.64	0.85	0.47
$p = 0.4$, $\mu = 8$, $r = 15$	0.37	0.23	0.43	0.73	0.47	0.62	0.81	0.40
$p = 0.4$, $\mu = 8$, $r = 30$	0.37	0.21	0.42	0.72	0.47	0.63	0.82	0.42
$p = 0.8$, $\mu = 4$, $r = 15$	0.73	0.27	0.53	0.85	0.53	0.79	0.89	0.79
$p = 0.8$, $\mu = 4$, $r = 30$	0.72	0.31	0.48	0.84	0.50	0.77	0.91	0.80
$p = 0.8$, $\mu = 8$, $r = 15$	0.71	0.29	0.67	0.86	0.83	0.88	0.92	0.81
$p = 0.8$, $\mu = 8$, $r = 30$	0.72	0.34	0.62	0.84	0.78	0.87	0.92	0.80

Table 5 Overview of used methods; here we used in terms of computation time only the classification ‘low’, ‘middle’ and ‘high’, since the computation time depends on the complexity of the data and we just want to give a first impression of the computation time in our settings

Method	Computation time	Requirements	Number of stages
hierarchical clustering	low	user has to define optimal number of clusters → definition by using silhouette width or ASH	two stages hierarchical clustering + silhouette width or ASH
ASH	low	user has to define degree of smoothing	one step
EMM	low middle	user has to define threshold (threshold;diameter)	one to two stages (if necessary: estimation for radius)
DBSCAN	low	user has to define ϵ	one stage
gammics	high	no prior knowledge necessary, but can be helpful	one stage

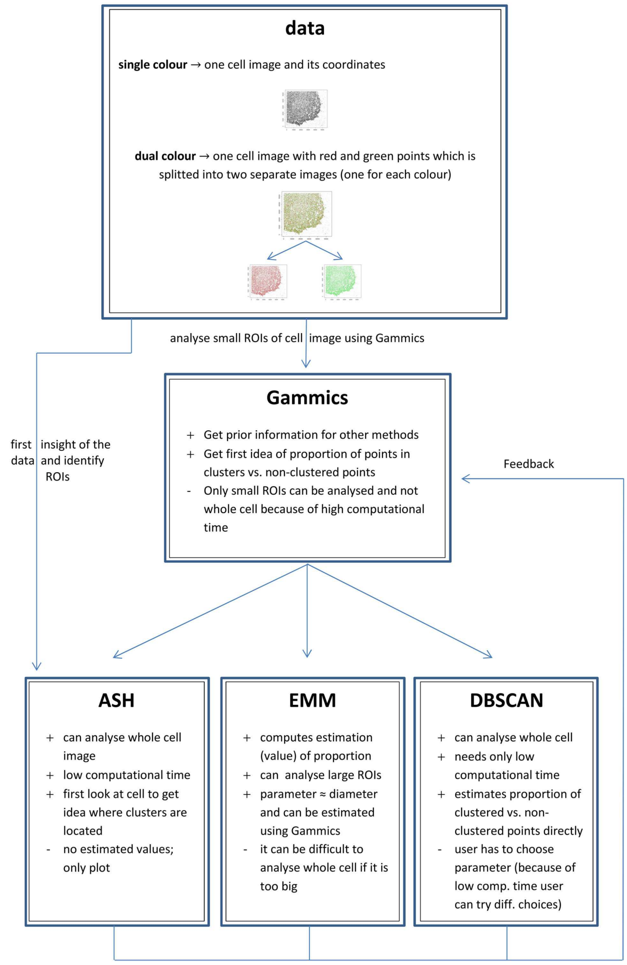
data
single colour → one cell image and its coordinates
dual colour → one cell image with red and green points which is splitted into two separate images (one for each colour)
analyse small ROIs of cell image using Gammics
first insight of the data and identify ROIs
Gammics
+ Get prior information for other methods
+ Get first idea of proportion of points in clusters vs. non-clustered points
- Only small ROIs can be analysed and not whole cell because of high computational time
Feedback
ASH
+ can analyse whole cell image
+ low computational time
+ first look at cell to get idea where clusters are located
- no estimated values; only plot
EMM
+ computes estimation (value) of proportion
+ can analyse large ROIs
+ parameter ≈ diameter and can be estimated using Gammics
- it can be difficult to analyse whole cell if it is too big
DBSCAN
+ can analyse whole cell
+ needs only low computational time
+ estimates proportion of clustered vs. non-clustered points directly
- user has to choose parameter (because of low comp. time user can try diff. choices)

Fig. 5 *Flowchart of the ‘feedback analysis’ of combining cluster methods to analyse single or dual colour protein data*

in absence of prior knowledge. Table 5 gives an overview of the methods in terms of computing time, stability and number of stages.

Generally, ASH is a (useful) method to obtain prior information. For example, ASH is a good method to get an overview over the data, e.g. where the clusters or ROIs are. Furthermore, one can estimate the proportion of points in clusters employing ASH and obtain prior information about the average number of proteins in one cluster.

The EMM, DBSCAN and the gammics method work well for estimating the proportion of points in clusters. For the EMM, one has to define a threshold which is approximately the mean diameter of the clusters. DBSCAN also needs a parameter which has to be determined by the user. Only the gammics method needs no special prior information, though this still could be helpful. A disadvantage of the gammics method is its high demands in terms of computation time. Please note that we classified the computation time only in three classes, because the computation time depends on the complexity of the data. We here just wanted to give a first impression of the computation time to the reader.

Hence, our advice is to first analyse the data by using ASH to find small ROIs which can then be analysed with the gammics method. If one has prior information how to choose ROIs, one can skip the ASH step and start directly with the gammics method. Note that the size of the first ROIs depend on the complexity of the data [in the simulation settings, we used an overall point density (points/μm^2) of 125]. Afterwards, the whole cell can be analysed using again ASH (if not used yet), EMM or DBSCAN, where for the DBSCAN, different choices for ϵ can be carried out because of its low computing time. This is also shown in Fig. 5.

4 Results of the new approach

In this section, we will employ the new approach to real single colour data as well as a dual colour simulation study. Again, most computations run on **R** [22] with the packages **cluster**, **ash**, **rEMM**, **spatstat** and **fpc**, only the gammics method operates on MATLAB (7.10.0 [29]).

4.1 Application of the new approach to real single colour data

In this section, we analyse the experimental data as shown in Fig. 6 on the left, which is also used in Schäfer *et al.* [21]. We first analyse four arbitrary chosen ROIs using the gammics method. The results are shown in Table 6, where the mean estimator is marked by a bar and the median estimator is marked by a tilde. The estimated values for the radius and the average size can be used as

Table 6 Average (labelled by $\bar{}$) and median (labelled by $\tilde{}$) estimators for real data analysed by the gammics method

ROI	$\bar{\hat{p}}(\tilde{\hat{p}})$	$\bar{\hat{\mu}}(\tilde{\hat{\mu}})$	$\bar{\hat{r}}(\tilde{\hat{r}})$
1	0.7528 (0.7573)	4.6297 (4.8022)	21.9653 (23.3396)
2	0.6631 (0.6621)	4.4658 (4.3965)	19.6638 (19.2250)
3	0.6583 (0.6580)	5.1815 (5.4357)	17.4426 (18.7699)
4	0.5944 (0.5923)	4.3179 (4.4116)	16.4657 (18.0482)
overall mean	0.6672 (0.6674)	4.6487 (4.7615)	18.8844 (19.8457)

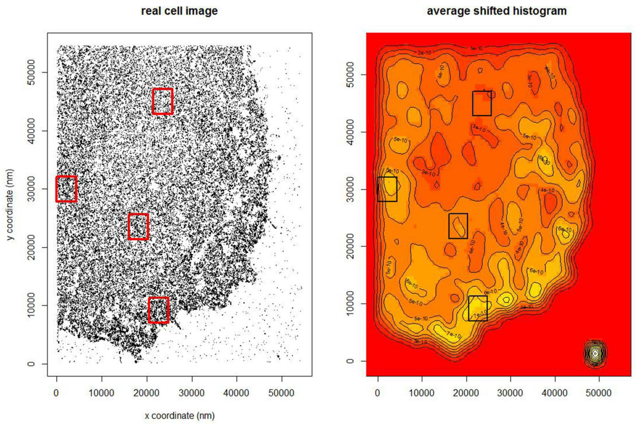

Fig. 6 *Resulting density plot using ASH for the real data; left: plot of the experimental data, where each dot is a measured protein and right: the resulted contour plot of this experimental data*

Table 7 Estimated proportions of points in clusters using EMM

Part of the cell	$\hat{p}_{30}$	$\hat{p}_{35}$	$\hat{p}_{40}$
a	0.8001	0.8369	0.8636
b	0.8131	0.8481	0.8742
c	0.8232	0.8561	0.8808
d	0.8056	0.8417	0.8694

input for other methods, e.g. the EMM. To get a first overview of the data and regions with high density of points, we analyse the data employing ASH. The resulting density plot is shown on the right-hand side in Fig. 6. One can see that the four chosen ROIs do have different point densities and reflect (mainly all) different parts of the cell. For EMM, we showed in Section 3 that the threshold parameter is nearly equal to the diameter. As can be seen in Table 6, the average mean of the radius varies between 17 and 22 nm, thus we use 30, 35 and 40 for this parameter since it should be roughly the diameter. Owing to memory constraints we split the spatial distribution of the cell in four parts, as shown in Fig. 5 in the supplement Appendix 2. The estimated values of the proportion for the four parts are shown in Table 7. As one can see, the estimated values are all similar, but much higher than the estimated values of the different ROIs employing the gammics method. The DBSCAN method works well on the whole cell image. The estimated proportions which depend on the chosen value of parameter $\epsilon \in \{25, 50, 75, 100\}$ are $\hat{p}_{25} = 0.5888$, $\hat{p}_{50} = 0.8155$, $\hat{p}_{75} = 0.8825$ and $\hat{p}_{100} = 0.9190$.

The estimated proportions for the EMM method are similar to those of the DBSCAN method with $\epsilon \in \{50, 75\}$. That confirms the results above.

4.2 Dual colour simulation study

Another field of application for the mentioned methods is dual colour data. Here, the data set consists of the localisations for light dots of the ‘green’ proteins as well as the localisations of the ‘red’ ones. These two images can be separated easily and then analysed as a single image, see Section 3. Here, we will show only results for simulated data. To that end, the ‘green’ image is simulated as above and the ‘red’ one is simulated as described in Section 2.1 with respect to (w.r.t.) the different settings.

To get an idea of what such a simulation looks such as, see Fig. 6 in the supplement Appendix 2: obviously, there are differences between the upper images and the lower ones. As one can see, the regions with high point density are located on different areas in (a) and (b). The regions with high point density in (c) and (d) are similar and only shifted in comparison with those in (a). The upper ones are independent and together represent setting 1. The ones below are the two different depending ‘red’ settings, such that images (a) and (c) create setting 2 and (a) and (d) represent setting 3, as described in Section 2.1.

The results for the gammics method are shown in 2 in the supplement. Note that the estimators work as well as for the single simulation in chapter 3. In summary, the results are similar to those in Section 3 for the three settings. Hence, to get prior information the gammics method is a valid method.

However, note that for setting 2, the mean radius of the ‘red’ proteins is underestimated for all simulations (mostly between 1 and 5 nm instead of 15 and 30 nm). In case of setting 1, the estimated values are similar to those for the ‘green’ proteins, the estimated values in setting 3 tend to overestimate the mean radius in cases of $r = 15$ nm and underestimate in cases of $r = 30$ nm.

The estimated values for the proportion of points in clusters using the other methods are shown in Tables 3–5 in the supplement Appendix 3 for all three settings. In summary, the EMM and DBSCAN work well for a good choice of parameter.

Thus, it is auxiliary to follow the work flow in Fig. 5. At this point of a real data analysis, new ROIs could be defined. These ROIs could be analysed again by using the gammics method, the EMM and DBSCAN. Subject to the size of the ROIs they can be analysed by the gammics method directly; otherwise, one has to choose smaller regions or divide them into smaller ones.

5 Conclusion and outlook

Proteins are an important component of the cell. They can regulate different mechanisms and also the transfer of other molecules through cells. For these reasons, it is important to understand the behaviour of proteins. Here, we concentrate on the clustering behaviour of proteins and the corresponding cluster characteristics.

To get the locations of proteins, fluorescence microscopy can be used, e.g. TRIF microscopy. Here, we analysed simulated data as well as a real data set.

We compared five different methods for analysing such data sets: hierarchical clustering, ASH, EMM, DBSCAN and the gammics method. Hierarchical clustering did not work well, but we showed that ASH is a good method to get a first overview of the data and also for estimating the proportion of points in clusters if prior information is available. For estimating the proportion of proteins in clusters, we employed a standard method (DBSCAN) as well as new methods (EMM and the gammics method). All three methods worked well. EMM and DBSCAN require the specification of an additional parameter, but run very fast. In contrast, the gammics method needs higher running time, but additionally estimates the mean radius and the mean size.

Thus our advice is to apply the new ‘feedback analysis’. After getting an overview of the data and perhaps prior information, employing ASH and/or the gammics method analyse the whole cell to estimate the proportion of clustered points, e.g. adopting EMM and DBSCAN, respectively. The scheme of this proposed new approach is shown in Fig. 5.

For future research, the identification of points in clusters is an interesting starting point and should be the next step. We showed that the estimation of the proportion of points in clusters employing EMM (and also hierarchical clustering) works well, but the identification of the points in clusters needs to be improved. If we can identify the points in clusters, we can compare the clusters and corresponding points for dual colour data. This would open up the possibility to have a closer look at the points in neighbouring clusters with different colours and draw conclusions of the (spatial) correlation of the two proteins.

Furthermore, we did not have a look at the robustness of our new approach or the applicability to other application fields. This should be done in a next step to analyse the performance of the flowchart for other types of data and settings.

6 Acknowledgments

The financial support of the Deutsche Forschungsgemeinschaft (grant no. IC 5/3-1 to Katja Ickstadt, grant no. VE 579/3-1 to Peter Verveer and grant no. SCHW 1508/3-1 to Holger Schwender) and the German Ministry of Education and Research (BMBF; FORSYS initiative, grant no. 0315257 to Peter Verveer) is gratefully acknowledged.

7 References

[1] PubMed Help [Internet]. Bethesda (MD): 'National Center for Biotechnology Information (US); 2005-. PubMed Help'. Available at https://www.ncbi.nlm.nih.gov/books/NBK3827/, accessed May 2017

[2] Eisen, M.B., Spellman, P.T., Brown, P.O., *et al.*: 'Cluster analysis and display of genome-wide expression patterns', *PNAS*, 1998, **95**, (25), pp. 14863–14868

[3] Sunaga, D.Y., Nievola, J.C., Ramos, M.P.: 'Statistical and biological validation methods in cluster analysis of gene expression'. Sixth Int. Conf. Machine Learning and Applications (ICMLA), 2007, pp. 494–499

[4] Arnau, V., Mars, S., Marín, I.: 'Iterative cluster analysis of protein interaction data', *Bioinformatics*, 2005, **21**, (2), pp. 364–378

[5] Manley, S., Gillette, J.M., Patterson, G.H., *et al.*: 'High-density mapping of single-molecule trajectories with photoactivated localization microscopy', *Nat. Methods*, 2008, **5**, (2), pp. 155–157

[6] Greb, C., Hosy, E.: 'Universal PAINT–dynamic super-resolution microscopy', 2015

[7] Burnette, D.T., Sengupta, P., Dai, Y., *et al.*: 'Bleaching/blinking assisted localization microscopy for super-resolution imaging using standard fluorescent molecules', *Proc. Natl. Acad. Sci.USA*, 2011, **108**, (52), pp. 21081–21086, doi:10.1073/pnas.1117430109

[8] Subach, F.V., Patterson, G.H., Manley, S., *et al.*: 'Photoactivatable mCherry for high-resolution two-color fluorescence microscopy', *Nat. Methods*, 2009, **6**, (2), pp. 153–159

[9] Betzig, E., Patterson, G.H., Sougrat, R., *et al.*: 'Imaging intracellular fluorescent proteins at nanometer resolution', *Am. Assoc. Adv. Sci.*, 2006, **313**, (5793), pp. 1642–1645, doi:10.1126/science.1127344

[10] Huang, B., Babcock, H., Zhuang, X.: 'Breaking the diffraction barrier: super-resolution imaging of cells', *Cell*, 2010, **143**, (7), pp. 1047–1058

[11] Ripley, B.D.: 'Modelling spatial patterns', *J. R. Stat. Soc. Ser. B*, 1977, **39**, (2), pp. 173–212

[12] Rubin-Delanchy, P., Burn, G.L., Griffie, J., *et al.*: 'Bayesian cluster identification in single-molecule localization microscopy data', *Nat. Methods*, 2015, **12**, (11), pp. 1072–1076

[13] Sengupta, P., Jovanovic-Talisman, T., Skoko, D., *et al.*: 'Probing protein heterogeneity in the plasma membrane using PALM and pair correlation analysis', *Nat. Methods*, 2011, **8**, (11), pp. 969–975

[14] Levet, F., Hosy, E., Kechkar, A., *et al.*: 'SR-Tesseler: a method to segment and quantify localization-based super-resolution microscopy data', *Nat. Methods*, 2015, **12**, (11), pp. 1065–1071

[15] Ester, M., Kriegel, H.-P., Sander, J., *et al.*: 'A density-based algorithm for discovering clusters in large spatial databases with noise'. Proc. Second Int. Conf. Knowledge Discovery and Data Mining (KDD-96), 1996, pp. 231–266, ed. U. M. F. Evangelos Simoudis, Jiawei Han AAAI Press

[16] Hartigan, J.A.: '*Clustering algorithms*' (Wiley, New York, 1975)

[17] Jain, A.K., Murty, M.N., Flynn, P.J.: 'Data clustering: a review', *ACM Comput. Surv.*, 1999, **31**, (3), pp. 264–323

[18] Kaufman, L., Rousseeuw, P.J.: '*Finding groups in data: an introduction to cluster analysis*' (Wiley, Wiley series in probability and mathematical statistics, New Jersey, 2005)

[19] Scott, D.W., Sain, S.R.: 'Multidimensional density estimation'. in Rao, C.R., Wegman, E.J., Solka, J.L. (EDs.): '*Handbook of statistics*' (Elsevier, 2005), vol. **24**, pp. 229–261

[20] Dunham, M.H., Meng, Y., Huang, J.: 'Extensible Markov model'. Proc. IEEE ICDM Conf., IEEE, 2004, pp. 371–374

[21] Schäfer, M., Radon, Y., Klein, T., *et al.*: 'A Bayesian mixture model to quantify parameters of spatial clustering', *Comput. Stat. Data Anal.*, 2015, **92**, pp. 163–176

[22] R Core Team: '*R: a language and environment for statistical computing*' (R Foundation for Statistical Computing, Vienna, Austria, 2013). Available at http://www.R-project.org/

[23] Baddeley, A., Turner, R.: 'Spatstat: an R package for analyzing spatial point patterns', *J. Stat. Softw.*, 2005, **12**, (6), pp. 1–42

[24] Hahsler, M., Dunham, M.H.: 'rEMM: extensible Markov model (EMM) for data stream clustering in R', R package version 1.0-8, 2014. Available at http://CRAN.R-project.org/package=rEMM

[25] Hahsler, M., Dunham, M.H.: 'rEMM: extensible Markov model for data stream clustering in R', *J. Stat. Softw.*, 2010, **35**, (5), pp. 1–31

[26] Ch, Hennig.: 'Fpc: flexible procedures for clustering, R package version 2.1-7', 2014. Available at http://CRAN.R-project.org/package=fpc

[27] Maechler, M., Rousseeuw, P., Struyf, A., *et al.*: 'Cluster: cluster analysis basics and extensions, R package version 1.14.4 – for new features, see the 'Changelog' file (in the package source), 2013

[28] Original, S., Scott, D.W.: 'Report by Gebhardt, A. adopted to recent S-PLUS by Kaluzny, S.: ash: David Scott's ASH routines'. R Package, version 1.0-14, 2013. Available at http://CRAN.R-project.org/package=ash

[29] MATLAB: '*Version 7.10.0 (R2010a)*' (The MathWorks Inc., Natick, MA, 2010)

[30] Johnson, S.C.: 'Hierarchical clustering schemes', *Psychometrika*, 1967, **32**, (3), pp. 241–254

[31] Ankerst, M., Breunig, M.M., Kriegel, H.-P., *et al.*: 'OPTICS: ordering points to identify the clustering structure'. ACM SIGMOD Int. Conf. Management of Data, 1999, pp. 49–60, ACM Press

[32] Argiento, R., Cremaschi, A., Guglielmi, A.: 'A 'density-based' algorithm for cluster analysis using species sampling Gaussian mixture models', *J. Comput. Graph. Stat.*, 2014, **23**, (4), pp. 1126–1142

8 Appendix 1: Methods

In this section, a detailed description of the clustering methods we used is given.

Cluster analysis is used to find structures in data x_i, $i = 1, \dots, n$, e.g. groups of the same kind. Hence, one assumes that the data can be grouped into h homogeneous clusters. To ensure the homogeneity, the cluster should only contain similar objects. To that end, the similarity of two points is measured by the distance. The goal of the cluster analysis is to classify n objects or observations into h groups, the clusters, in which every cluster should contain at most one and maximal all possible objects. Popular distance measures $\mathrm{d}(x_i, x_j)$ are the Manhattan block matrix or the L_q norm, where for $q = 2$ it is the Euclidean distance *(for detailed information about distance measures see Johnson* [30]*).*

8.1 Hierarchical clustering

To find clusters using hierarchical clustering, there are two possible approaches: the agglomerative one and the divisive one. The agglomerative approach starts with the finest partition, i.e. each object forms one cluster. In the following iterations, the two 'nearest' clusters are merged until there is only one big cluster. The divisive approach starts with one cluster containing all observations and iteratively splits the most inhomogeneous cluster.

The distance of two clusters and the homogeneity of a cluster, respectively, can be computed in different ways as described in Hartigan [16], e.g. based on the average-linkage approach.

To find the optimal number of clusters, one can use the silhouette width. It represents the goodness of fit for each observation for a given clustering. The silhouette width for observation x_i is given by

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}, \tag{1}$$

where $a(i)$ is the average distance between x_i and all other observations in the same cluster and $b(i)$ is the minimum of all distances of x_i and all other clusters. Thus, the silhouette width $s(i)$ takes a value between -1 and 1. Observations with a silhouette width value near 1 are represented well, and observations with $s(i) < 0$ are not.

Now, the average silhouette width for all observations for a given numbers of clusters can be computed. The optimal number of clusters h_{opt} is given by the number h of clusters that maximises the average silhouette width.

8.2 Average SH

Assume the data to be divided into $k^\star$ bins, labelled $0, \dots, k^\star - 1$. Let the kth bin be defined by the interval $B_k = [\tau_k, \tau_{k+1})$ and c_k is the corresponding number of observations in this interval. Then, the density histogram is given by $\hat{f}(x) = (c_k/n(\tau_{k+1} - \tau_k))$, where n is the total number of observations. Often the bins are equally spaced, thus $\omega = \tau_{k+1} - \tau_k \quad \forall k$ and $\tau_0 = 0$. The density histogram then results in $\hat{f}(x) = (c_k/n\omega)$. In this case, there is only one unknown parameter ω. That is why it is often called 'non-parametric' (Scott and Sain [19]).

If τ_0 is unknown, there is the unknown parameter pair (ω, τ_0). In this case, τ_0 can be seen as a 'nuisance' parameter which can be purged by m ASH, where each histogram is shifted by a factor $\delta = (\omega/m)$ from the previous histogram.

Let the intervals $B_k^\star = [\tau_0 + k\delta, \tau_0 + (k+1)\delta)$ all be equal spaced with width δ and c_k the corresponding counts. Then the ASH is constant over all intervals and

$$\hat{f}_{ASH} = \frac{1}{n\omega} \sum_{i=1-m}^{m-1} \left(1 - \frac{|i|}{m}\right) c_{k+i}, \quad \text{for } x \in B_k^\star. \tag{2}$$

In addition to equal weights for the SHs, any other choices for $\hat{f}_{\text{ASH}} = (1/n\omega) \sum_{i=1-m}^{m-1} w_m(i) c_{k+i}$, for $x \in B_k^\star$ is possible, e.g. a weight defined by a kernel $\text{ker}(x)$. Since $m \to \infty$ the ASH approximates the 'kernel estimator'

$$\hat{f}_{\text{ker}} = \frac{1}{n\omega} \sum_{i=1}^{n} \text{ker}\left(\frac{x - x_i}{\omega}\right) = \frac{1}{n} \sum_{i=1}^{n} \text{ker}_\omega(x - x_i), \tag{3}$$

where $\text{ker}_\omega(x) = (1/\omega)\text{ker}[(x/\omega)]$.

For vector-valued data, the extension of ASH estimators is straight forward. An alternative to these estimators are mixture models, which are fitted using the expectation-maximization (EM) algorithm. More information can be found in Scott and Sain [19].

8.3 Extensible MMs

An EMM is a Markov chain which is varying in time. This Markov chain can be interpreted as a directed graph (with fixed structure). The EMM has the following properties.

The EMM is built up of a Markov chain with nodes N_n at every point in time and an algorithm which alters the Markov Chain. This algorithm consists of the three parts: EMMCluster, EMMIncrement and EMMDecrement.

Thereby EMMCluster assigns an object to a cluster at a given time t which is represented by the nodes. If the object is not close enough to the other clusters, this object is assigned to a new cluster. Whether the object is close enough to an existing cluster is determined by a threshold defined by the user. This step can be described as a kind of 'nearest-neighbour' algorithm. EMMIncrement computes transition probabilities of the Markov chain by retaining the counts as an indicator how many 'visitors' the node has. Moreover finally, EMMDecrement can reduce the size of the EMM if it gets too large.

The EMM still learns during the application phase and is 'generic incremented model whose nodes can have any kind of representative' (Dunham *et al.* [20]). It is possible to match states during clustering. Furthermore, it is allowed to add new nodes and also delete present nodes. Finally, the EMM can be applied to online data.

Detailed information of the EMM can be found in Dunham *et al.* [20].

8.4 DBSCAN

DBSCAN is a commonly adopted algorithm for finding clusters in a data set $\{x_1, \dots, x_n\}$ and employs the distance between two points as a key number for indicating a cluster. The distance function is arbitrary, but it defines the shape of the neighbourhood, e.g. in 2D with a Manhattan distance, the shape is rectangular. Therefore, Ester *et al.* [15] introduced the theory of density-reachability w.r.t. two key parameters, ϵ and *MinPts*. A point x_i is density-reachable for x_j if the distance between x_i and x_j is at most ϵ and if there are at least *MinPts* points in the ϵ neighbourhood of x_j, i.e. within a distance ϵ of x_j.

Furthermore, a set of points $x_1, \dots, x_m$, $m \le n$, is density connected, if for each pair of points x_i and x_j, $i, j \in \{1, \dots, m\}$, there is a permutation $x_i = x_1^\star, \dots, x_m^\star = x_j$, a set of density-collected points (i.e. a set in which $x_{(i+1)}^\star$ is directly density-reachable from $x_i^\star$).

Accordingly, if the two parameters ϵ and *MinPts* are given (or, ideally, known) the algorithm can start with an arbitrary point $x_\star$ and search for all density-reachable points of $x_\star$ w.r.t. to parameters ϵ and *MinPts*. If $x_\star$ is a kernel point (of a cluster), DBSCAN finds a cluster w.r.t. the parameter ϵ and *MinPts*. Is the point a boundary point, i.e. $x_\star$ lies on the boundary of a cluster, there are no density-reachable points of $x_\star$ and the algorithm chooses another point.

Since ϵ und *MinPts* are global parameters, several problems can occur, e.g. two 'near' clusters with different densities can be merged or a boundary point can belong to two different clusters. On account of such problems, another run of DBSCAN can be necessary. Moreover, there are also modifications of the DBSCAN algorithm such as the ordering points to identify the clustering structure (OPTICS) method or Bayesian DBSCAN (bDBSCAN) (Ankerst *et al.* [31] and Argiento *et al.* [32]). However, in this paper we will use the original DBSCAN algorithm.

8.5 Gammics

The gammics method (Schäfer *et al.* [21]) is another method for finding clusters in a point pattern, where a cluster has to consist of at least two objects. A nice feature of this method is the possibility of estimating the proportion, the cluster size and the cluster radius simultaneously.

To achieve this, the gammics method does not model the point pattern itself, but rather the squared distances between a point and its κth nearest neighbour, here we use $\kappa = 2$. The cluster radius and size are then estimated algorithmically.

Assume the data to be realisations of a random variable X_i, $i = 1, \dots, n$, in which they are random point coordinates in an endless region $P \subset \mathbb{R}^2$. Let D_i be the distance between the point X_i and its nearest neighbour X_j with realisations $d_i = d(x_i, x_j)$. The approach is to model a function of D_i^2 which indicates if a point X_i is clustered or not, because the squared distances obviously differ between clustered and non-clustered points. This function is given by

$$Y_i(D_i^2) = \begin{cases} 1 & X_i \text{ is part of a cluster} \\ 0 & X_i \text{ is not part of a cluster}. \end{cases} \tag{4}$$

For D_i^2, we fit two gamma distributions (with density $p(x) = [1/\Gamma(\alpha)\beta^\alpha]x^{\alpha-1}\text{e}^{-(\alpha/\beta)}$), one representing the clustered points and one the non-clustered. Thus, we can write

$$D_i^2 | Y_i = k \sim \text{gamma}(\alpha_k, \beta_k) \quad k = 0, 1 \tag{5}$$

$$\begin{aligned} \alpha_k | Y_i = k &\sim \text{gamma}(a_k, b_k) \quad k = 0, 1 \quad \text{and} \\ \frac{1}{\beta_k} | Y_i = k &\sim \text{gamma}(c_k, d_k) \quad k = 0, 1. \end{aligned} \tag{6}$$

Furthermore, 'the implicit allocation of points to one of the distributions is carried out by means of $Y = (Y_1, \dots, Y_n)'$ '(Schäfer *et al.* [21]) and we assume the Y_i to be Bernoulli distributed. The mixture proportion is then given by the following beta distribution:

$$Y_i \sim \text{Bernoulli}(p_c), \tag{7}$$

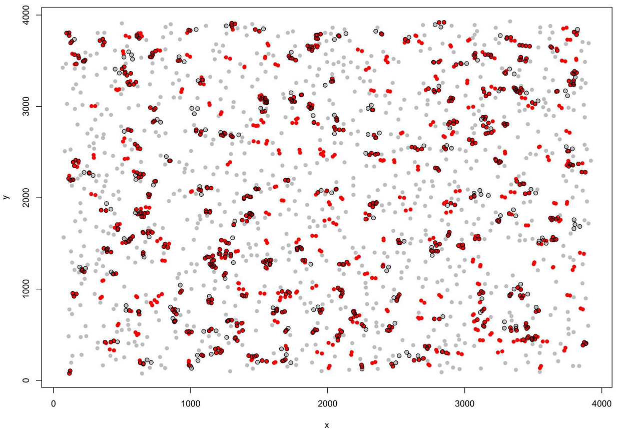

Fig. 7 *Results of clustering using ASH and hierarchical clustering (red points are clustered, grey points are singletons and the points with black circles are the real clustered points)*

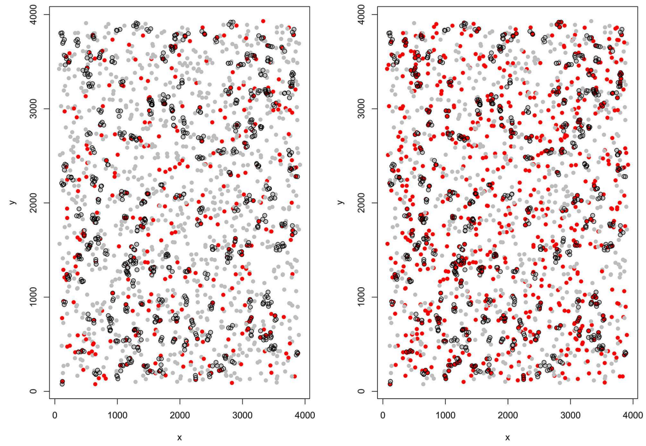

Fig. 8 *Results of clustering using EMM with two different thresholds (red points are clustered, grey points are singletons and the points with black contour are the real clustered points); left: EMM threshold = 25 and right: EMM threshold = 55*

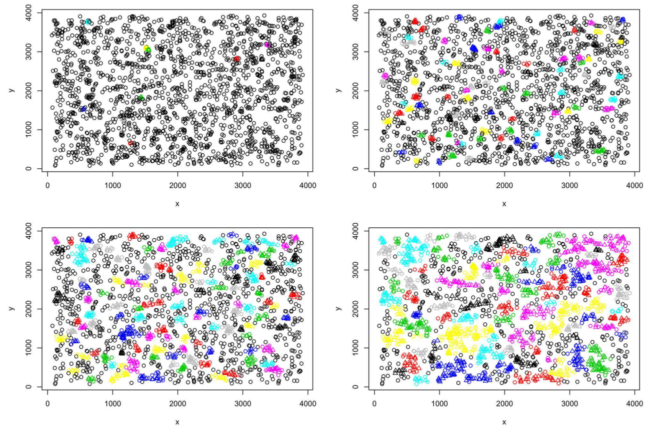

Fig. 9 *Result of clustering by DBSCAN with different choices of the parameter ϵ; top left: $\epsilon = 25$, top right: $\epsilon = 50$, bottom left: $\epsilon = 75$, bottom right: $\epsilon = 100$*

$$p_c \sim \text{Beta}(e,\ f)\,. \tag{8}$$

The mixture model underlying the gammics method is given by (4)–(8). The hyperparameters a_0, a_1, b_0, b_1, c_0, c_1, d_0, d_1, e and f have to be defined by the user. In our application, we use $a_0 = 3$, $a_1 = 2$, $b_0 = 1$, $b_1 = 1$, $c_0 = 1$, $c_1 = 4$, $d_0 = 0.5$, $d_1 = 1$ and $e = f = 1$.

The estimation of mean cluster radius and mean cluster size depends on the allocations as well as on the basic group, i.e. clustered versus non-clustered.

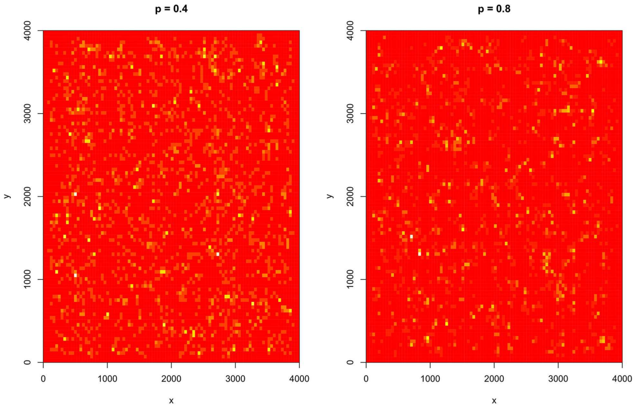

Fig. 10 *ASH for a simulation with p = 0.4 (left) and p = 0.8 (right)*

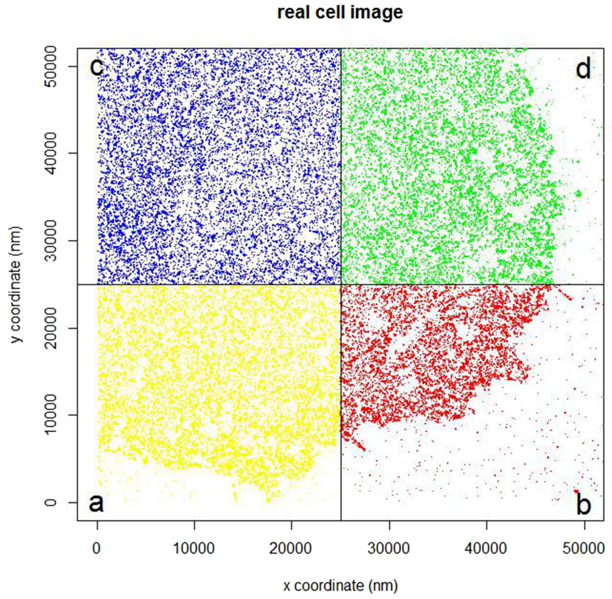

Fig. 11 *Experimental data fragmented into four parts; these parts are used for the EMM-analysis, because the EMM method cannot analyse the whole cell in one step. Thus we split the cell in these four parts to analyse the experimental data by using the EMM method*

The cluster structure is then derived from the fit of the two gamma distributions algorithmically, where first the intersection L_c between the densities is calculated. It can be written as

$$L_c = \{x \mid p_c \cdot p(x \mid \alpha_{y,\,1},\ \beta_{y,\,1}) = (1 - p_c) \cdot p(x \mid \alpha_{y,\,0},\ \beta_{y,\,0})\}, \tag{9}$$

where p is the density of the gamma distribution. Second, the distance D_q of a point X_q to its nearest neighbour is determined by $q = \text{argmin}_{i:D_i^2 \ge L_c} D_i^2$. Thus, two points are clustered in the same group (even) if the distance is higher than L_c, but there exist other points between those two, such that no distance between 'nearest neighbours' is more than L_c.

The model is finally fitted by a Gibbs sampling Markov chain Monte Carlo (MCMC) approach including a Metropolis step for updating the shape parameters of the gamma distribution.

9 Appendix 2

See Figs. 7–12.

10 Appendix 3

See Tables 8–12.

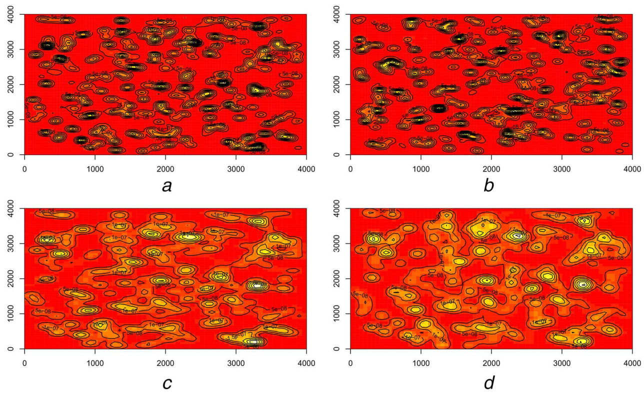

Fig. 12 *Dual colour simulation with parameter settings p = 0.4, μ = 4, r = 15 and all three settings analysed by ASH*
(a) Green proteins, ***(b)*** Red proteins for setting 1, ***(c)*** Red proteins for setting 2, ***(d)*** Red proteins for setting 3

Table 8 Estimated number of clusters by using ASH, where green $\widehat{=}$ simulation of green proteins, red1$\widehat{=}$simulation of red proteins and setting 1, red2 $\widehat{=}$ simulation of red proteins and setting 2 and red3 $\widehat{=}$ simulation of red proteins and setting 3

Simulation parameter	Green	Ash	Red 1	Ash	Red 2	Ash	Red 3	Ash	All 1	Ash	All 2	Ash	All 3	Ash
$p = 0.4$, $\mu = 4$, $r = 15$	202	150	207	150	202	130	202	120	409	400	404	350	404	300
$p = 0.4$, $\mu = 4$, $r = 15$	194	120	222	130	194	120	194	130	416	300	388	300	388	250
$p = 0.4$, $\mu = 4$, $r = 15$	195	150	208	150	195	180	195	150	403	400	390	350	390	300
$p = 0.4$, $\mu = 4$, $r = 15$	196	150	183	100	196	180	196	150	379	350	392	350	392	300
$p = 0.4$, $\mu = 4$, $r = 30$	205	150	210	150	205	120	205	150	415	400	410	400	410	350
$p = 0.4$, $\mu = 4$, $r = 30$	194	120	205	150	194	150	194	120	399	400	388	300	388	300
$p = 0.4$, $\mu = 4$, $r = 30$	205	130	211	100	205	150	205	130	416	300	410	280	410	350
$p = 0.4$, $\mu = 4$, $r = 30$	210	130	204	150	210	150	210	200	414	300	420	280	420	300
$p = 0.4$, $\mu = 8$, $r = 15$	92	80	102	100	92	100	92	120	194	200	184	180	184	180
$p = 0.4$, $\mu = 8$, $r = 15$	114	100	121	100	114	130	114	120	235	200	228	250	228	200
$p = 0.4$, $\mu = 8$, $r = 15$	102	90	106	120	102	120	102	120	208	200	204	180	204	180
$p = 0.4$, $\mu = 8$, $r = 15$	102	100	96	90	102	100	102	100	198	250	204	200	204	200
$p = 0.4$, $\mu = 8$, $r = 30$	89	80	98	90	89	80	89	90	187	200	178	180	178	180
$p = 0.4$, $\mu = 8$, $r = 30$	101	90	102	100	101	100	101	90	203	200	202	180	202	180
$p = 0.4$, $\mu = 8$, $r = 30$	96	80	89	90	96	100	96	100	185	200	192	200	192	200
$p = 0.4$, $\mu = 8$, $r = 30$	103	90	108	90	103	100	103	120	211	180	206	200	206	200
$p = 0.8$, $\mu = 4$, $r = 15$	444	100	405	120	444	120	444	130	849	300	888	350	888	300
$p = 0.8$, $\mu = 4$, $r = 15$	424	90	468	120	424	100	424	120	892	250	848	250	848	250
$p = 0.8$, $\mu = 4$, $r = 15$	402	150	454	170	402	150	402	150	856	300	804	400	804	300
$p = 0.8$, $\mu = 4$, $r = 15$	432	100	416	120	432	150	432	120	848	400	864	350	864	400
$p = 0.8$, $\mu = 4$, $r = 30$	441	120	426	180	441	150	441	150	867	200	882	250	882	250
$p = 0.8$, $\mu = 4$, $r = 30$	428	150	418	150	428	180	428	180	846	350	856	300	856	300
$p = 0.8$, $\mu = 4$, $r = 30$	396	150	428	200	396	150	396	180	824	300	792	350	792	350
$p = 0.8$, $\mu = 4$, $r = 30$	427	100	411	100	427	150	427	150	838	250	854	200	854	200
$p = 0.8$, $\mu = 8$, $r = 15$	211	120	183	130	211	150	211	130	394	300	422	350	422	300
$p = 0.8$, $\mu = 8$, $r = 15$	217	100	200	100	217	100	217	130	417	250	434	250	434	250
$p = 0.8$, $\mu = 8$, $r = 15$	203	150	188	170	203	150	203	150	391	400	406	350	406	350
$p = 0.8$, $\mu = 8$, $r = 15$	211	120	233	130	211	150	211	130	444	300	422	250	422	300
$p = 0.8$, $\mu = 8$, $r = 30$	180	150	196	150	180	180	180	170	376	320	360	300	360	350
$p = 0.8$, $\mu = 8$, $r = 30$	192	150	214	120	192	120	192	120	406	300	384	280	384	300
$p = 0.8$, $\mu = 8$, $r = 30$	197	150	198	120	197	150	197	130	395	300	394	300	394	300
$p = 0.8$, $\mu = 8$, $r = 30$	210	120	210	150	210	150	210	130	420	300	420	300	420	280

Table 9 Average estimators obtained using the gammics method for all parameter choices for the simulations, where $\bar{\hat{r}}$ is the average estimator for the radius, $\bar{\hat{\mu}}$ is the average estimator for the size and $\bar{\hat{p}}$ is the average estimator for the proportion of points in clusters and also green ≙ simulation of green proteins, red1 ≙ simulation of red proteins and setting 1, red2 ≙ simulation of red proteins and setting 2, red3 ≙ simulation of red proteins and setting 3

Simulation	Green			Red 1			Red 2			Red 3		
	$\bar{\hat{r}}$	$\bar{\hat{\mu}}$	$\bar{\hat{p}}$	$\bar{\hat{r}}$	$\bar{\hat{\mu}}$	$\bar{\hat{p}}$	$\bar{\hat{r}}$	$\bar{\hat{\mu}}$	$\bar{\hat{p}}$	$\bar{\hat{r}}$	$\bar{\hat{\mu}}$	$\bar{\hat{p}}$
$p = 0.4$, $\mu = 4$, $r = 15$	9.16	3.74	0.43	9.08	6.82	0.43	8.48	3.57	0.78	9.39	7.08	0.82
$p = 0.4$, $\mu = 4$, $r = 30$	8.98	3.73	0.46	8.61	6.89	0.42	8.31	3.64	0.81	8.76	6.63	0.81
$p = 0.4$, $\mu = 8$, $r = 15$	9.80	3.84	0.43	9.60	6.85	0.42	8.45	3.68	0.81	8.94	6.67	0.81
$p = 0.4$, $\mu = 8$, $r = 30$	8.43	3.74	0.44	8.40	6.59	0.44	8.24	3.62	0.81	8.17	6.66	0.84
$p = 0.8$, $\mu = 4$, $r = 15$	14.71	3.68	0.50	12.80	6.13	0.45	12.84	3.60	0.80	13.27	6.36	0.81
$p = 0.8$, $\mu = 4$, $r = 30$	15.20	3.81	0.49	11.61	5.93	0.45	13.07	3.50	0.79	11.95	6.01	0.81
$p = 0.8$, $\mu = 8$, $r = 15$	15.01	3.81	0.48	12.87	5.95	0.42	13.34	3.72	0.80	12.31	6.00	0.81
$p = 0.8$, $\mu = 8$, $r = 30$	15.34	3.69	0.54	12.78	5.80	0.45	13.44	3.75	0.81	11.92	5.87	0.82

Table 10 Average estimations of the proportion of points in clusters for red proteins of dual colour data set, setting 1 (independent proteins)

Simulation parameter	$\hat{p}_{\text{prior}}$	$\hat{p}_{\text{EMM25}}$	$\hat{p}_{\text{EMM55}}$	$\hat{p}_{\text{db50}}$	$\hat{p}_{\text{db75}}$	$\hat{p}_{\text{db100}}$
$p = 0.4$, $\mu = 4$, $r = 15$	0.37	0.39	0.73	0.33	0.63	0.85
$p = 0.4$, $\mu = 4$, $r = 30$	0.36	0.35	0.73	0.33	0.64	0.87
$p = 0.4$, $\mu = 8$, $r = 15$	0.37	0.43	0.72	0.46	0.61	0.80
$p = 0.4$, $\mu = 8$, $r = 30$	0.37	0.40	0.73	0.45	0.62	0.82
$p = 0.8$, $\mu = 4$, $r = 15$	0.71	0.55	0.85	0.55	0.79	0.91
$p = 0.8$, $\mu = 4$, $r = 30$	0.69	0.48	0.84	0.49	0.79	0.92
$p = 0.8$, $\mu = 8$, $r = 15$	0.71	0.66	0.85	0.81	0.87	0.91
$p = 0.8$, $\mu = 8$, $r = 30$	0.70	0.63	0.86	0.80	0.89	0.92

Table 11 Average estimations of the proportion of points in clusters for red proteins of dual colour data set, setting 2 (correlated cluster centres)

Simulation parameter	$\hat{p}_{\text{prior}}$	$\hat{p}_{\text{EMM25}}$	$\hat{p}_{\text{EMM55}}$	$\hat{p}_{\text{db50}}$	$\hat{p}_{\text{db75}}$	$\hat{p}_{\text{db100}}$
$p = 0.4$, $\mu = 4$, $r = 15$	0.37	0.36	0.67	0.33	0.55	0.74
$p = 0.4$, $\mu = 4$, $r = 30$	0.37	0.36	0.67	0.33	0.56	0.74
$p = 0.4$, $\mu = 8$, $r = 15$	0.37	0.41	0.66	0.44	0.55	0.71
$p = 0.4$, $\mu = 8$, $r = 30$	0.37	0.43	0.66	0.47	0.56	0.70
$p = 0.8$, $\mu = 4$, $r = 15$	0.73	0.55	0.83	0.54	0.75	0.86
$p = 0.8$, $\mu = 4$, $r = 30$	0.72	0.56	0.84	0.54	0.75	0.86
$p = 0.8$, $\mu = 8$, $r = 15$	0.71	0.68	0.85	0.82	0.86	0.89
$p = 0.8$, $\mu = 8$, $r = 30$	0.72	0.67	0.84	0.81	0.85	0.89

Table 12 Average estimations of the proportion of points in clusters for red proteins of dual colour image, setting 3 (correlated points)

Simulation parameter	$\hat{p}_{\text{prior}}$	$\hat{p}_{\text{EMM25}}$	$\hat{p}_{\text{EMM55}}$	$\hat{p}_{\text{db50}}$	$\hat{p}_{\text{db75}}$	$\hat{p}_{\text{db100}}$
$p = 0.4$, $\mu = 4$, $r = 15$	0.37	0.34	0.67	0.30	0.54	0.74
$p = 0.4$, $\mu = 4$, $r = 30$	0.37	0.31	0.65	0.26	0.54	0.73
$p = 0.4$, $\mu = 8$, $r = 15$	0.37	0.39	0.65	0.44	0.55	0.71
$p = 0.4$, $\mu = 8$, $r = 30$	0.37	0.38	0.67	0.43	0.57	0.72
$p = 0.8$, $\mu = 4$, $r = 15$	0.73	0.50	0.82	0.48	0.74	0.87
$p = 0.8$, $\mu = 4$, $r = 30$	0.72	0.45	0.80	0.44	0.74	0.87
$p = 0.8$, $\mu = 8$, $r = 15$	0.71	0.64	0.84	0.80	0.87	0.89
$p = 0.8$, $\mu = 8$, $r = 30$	0.72	0.59	0.83	0.75	0.85	0.90