Data Mining: Clustering Assignment

Kamal Sharma
Final year BTech
Computer Science and Engineering
Indian Institute of Technology, Ropar
2017csb1084@iitrpr.ac.in

ABSTRACT

Clustering is one of the ways used to find the structures in the data using unsupervised learning. Clustering methods serve an important part in market analysis and deciding business strategies by find pattern and structures in the clients, sales historic data. The biggest advantage of such unsupervised learning methods is that these don't need a labelled dataset, which save a lot of time in collecting training samples. In this report, we will thoroughly analyze a few cluster algorithms named K-Means, DBScan, EM and DENCLUE over Iris dataset [1] and spiral dataset [2].

Keywords

K-Means, DBScan, Expectation-Maximization, DENCLUE, multivariate normal distribution [3], Principle Component Analysis [4]

Libraries used

Principle Component Analysis (PCA) [4], matplotlib.pyplot, mpl_toolkits.mplot3d.Axes3D, multivariate_normal, numpy, pandas, scipy.stats

1 Datasets

For analyzing different clustering algorithms' performance, we have used 2 different datasets.

Iris Dataset [1]: Dataset consists of 150 unique samples with 4 features/attributes (A1, A2, A3, A4) per sample. See Table 1 for dataset detailed description. Dataset has 3 labels.

Spiral Dataset [2]: Dataset consists of 1000 instances with 2 features (A0, A1) per sample. In dataset, instance and features are arranged in such a way that those create 2 spiral clusters. Dataset has 2 classes. See Table 2 for dataset detailed description. Pictorial representation in Fig 1.

	A1	A2	A3	A4
count	150	150	150	150
mean	5.843333	3.054	3.758667	1.198667
std	0.828066	0.433594	1.76442	0.763161
min	4.3	2	1	0.1
max	7.9	4.4	6.9	2.5

Table 1: Iris dataset description

	A0	A1
count	1000	1000
mean	0	0
std	3.0135	2.9967
min	-6.4889	-5.6911
max	6.4889	5.6911

Table 2: Spiral dataset description

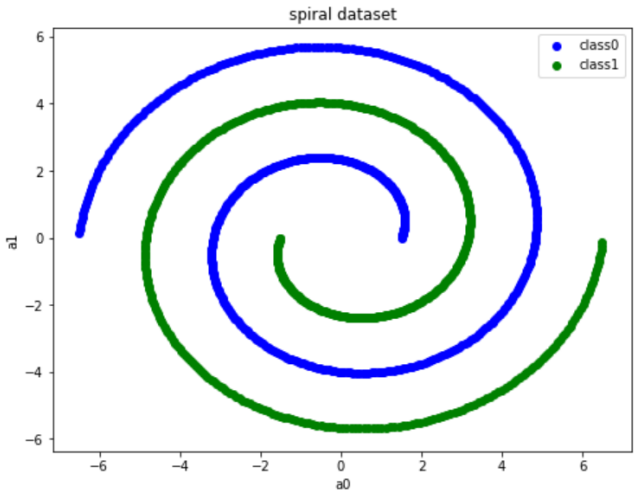

Fig 1: Spiral dataset

As it is not plot 4 dimensional iris dataset, PCA is applied to find 2 principal components of data, say u0 and u1, with minimum information loss. See Fig 2.

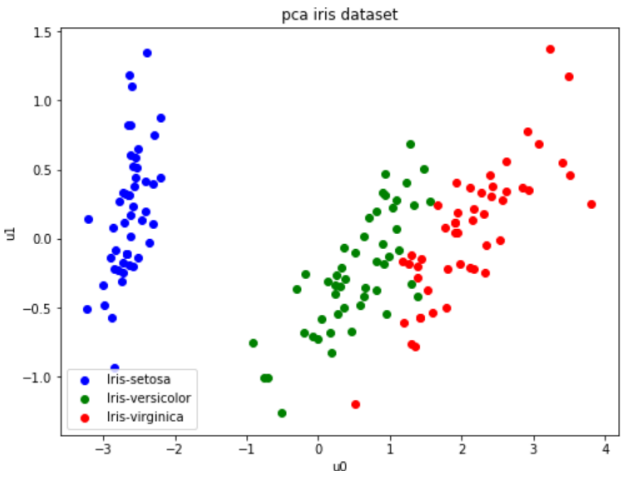

Fig 2: Iris dataset (after applying PCA)

2 K-Means algorithm

It is a portioned bases cluster algorithm to find structures in the dataset. K-Means results into non-overlapping convex clusters.

Convex set: A set S $\subseteq$ R^n is a convex set if for any $x_1, x_2 \in S$ and $0 \leq \gamma \leq 1$, $\gamma x_1 + (1-\gamma) x_2 \in S$. In short every point on line joining x_1 and x_2 must belong to set S.

Idea of K-Means algorithm is to minimize intra cluster distance or maximize inter cluster distance. Let W(C) be the intra cluster distance for cluster C, then

$W(C) = (1/2) * \sum_{m=1}^{k} \sum_{C(i)=m} \sum_{C(i')=m} dist(x_i, x_{i'})$

Or $W(C) = \sum_{m=1}^{k} N_m \sum_{C(i)=m} \| x_i - x_m \|$, where N_m is the number of points in cluster C(i) and x_m is the mean of points in cluster C(i)

For selecting optimal K-value (total number of clusters value), we used sum of squared error metric. We'll select the K after which SSE doesn't decrease much. For SSE vs K curve see Fig 3 and 4.

$$SSE = \sum_{i=1}^{k} \sum_{x \in C_i} (x^T x - n_i u_i^T u_i)$$

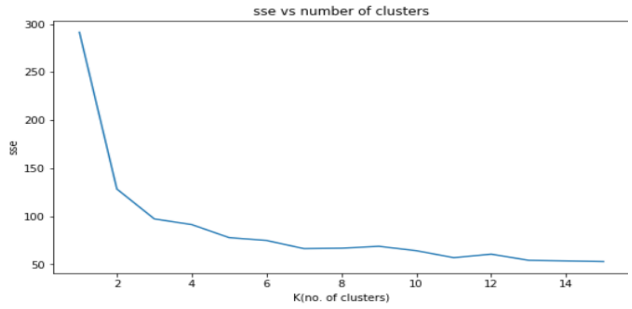

Fig 3: SSE vs K for iris dataset

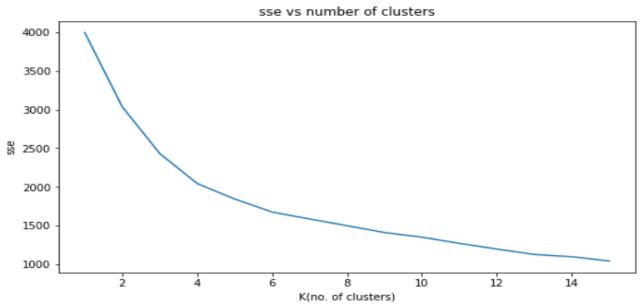

Fig 4: SSE vs K for spiral dataset

Observations:

SSE decreases with increase in number of clusters (K). For iris dataset after K = 3, SSE hasn't decreased much. And for spiral dataset after K = 5, SSE hasn't decreased much. So we continued up with K=3 for iris dataset and K = 5 for spiral dataset. Also with the increase in number of clusters, run/train time of algorithm increases as time complexity of K-Means algorithm is O(knd) (where k is number of clusters, n is total number of samples and d is the number of attributes per sample) which can be clearly observed from Fig 5 and 6.

Fig 5: run time vs K for iris dataset

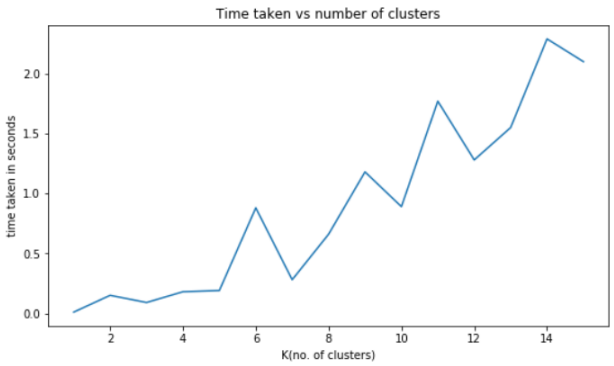

Fig 6: run time vs K for spiral dataset

3 DBScan algorithm

DBScan algorithm falls under Density Based clustering algorithms category. K-Means can results only in convex clusters. The basic idea of this algorithm is to move concentration to local points from global points. All the density connected points belong to the same cluster. X and Y are Density connected if there exists a point Z s.t. X and Y are both density reachable from Z. Goal of the DBScan algorithm is to find maximal density connected points.

Notations:

Nε(x) or ε-neighborhood of x= { y | distance(x, y) $\leq$ ε)

X is a core point if in ε-neighborhood of x, we have at least min_points number of instances/points.

x is density reachable from y if there exists $y=x_1, x_2, \ldots, x_t=x$ s.t. each x_i (except x_t) is a core point and is directly density reachable from other x_j in path x1 to x_{t-1}.

x is directly density reachable from y if $x \in N\varepsilon(y)$ and y is a core point.

For iris dataset number of clusters selected is 3 and for spiral dataset number of clusters selected are 2. As same number of clusters can be achieved by different values of epsilon and min_points, so to break the tie we have selected that epsilon and min_points pair which has highest number of core points.

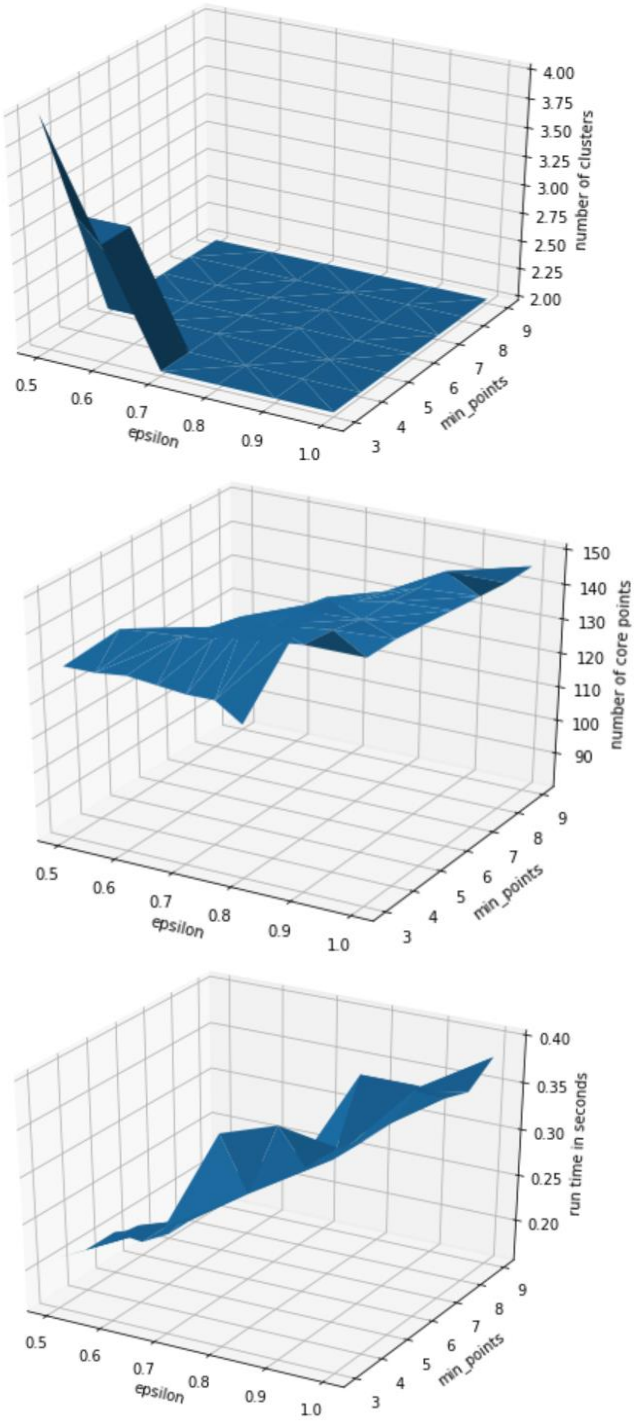

Fig 7: all three rows in this figure corresponds to iris dataset

Observations (Iris dataset):

1. For iris-dataset, K=3 (found by K-Means algorithm using SSE metric)

2. Here in a few epsilon-min_points pair K=3, so we select that pair which has K=3 and has maximum number of core points.
3. Number of core points increses with increase in epsilon and decrease in min_points
4. best epsilon = 0.6, best min_points = 3
5. for this setting K = 3, and number of core points are more as compared to other/

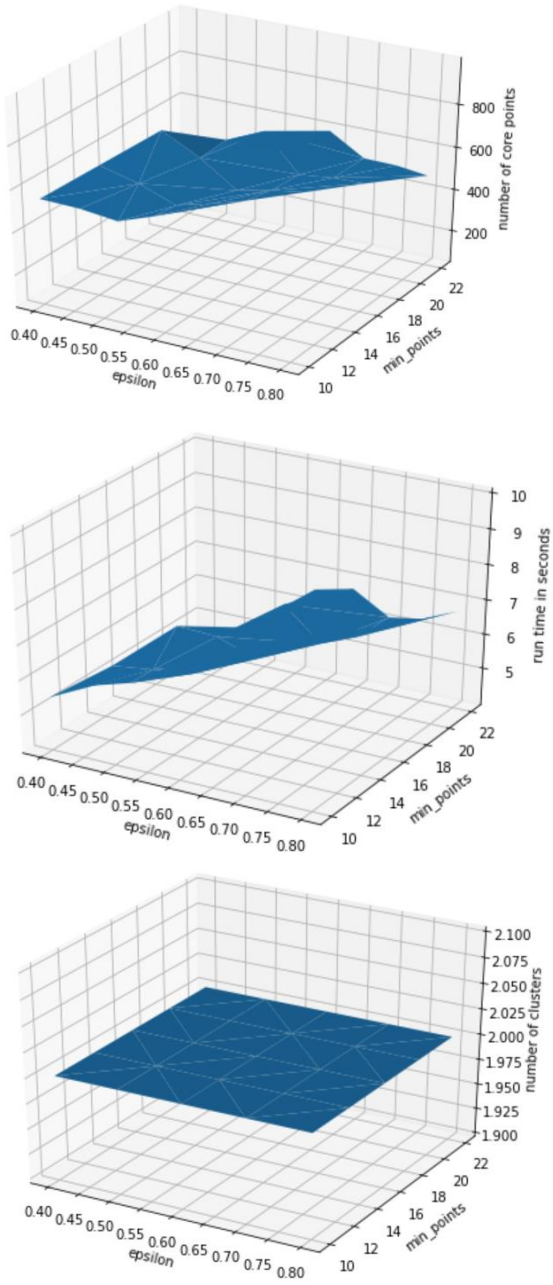

Fig 8: all three rows in this figure corresponds to spiral dataset

Observations (spiral dataset):

1. Number of core points increses with increase in epsilon and decrease in min_points.
2. All the settings of epsilon and min_points pair create either 2 clusters or no clusters.
3. Depending on the larger number of core points, we chose epsilon = 0.8, min_points=13.

Comparison between K-Means and DBScan algorithms:

For Iris dataset K-Means perform better but for spiral dataset K-Means miserably fails as now clusters are no longer convex in shape. For spiral dataset cluster are spherical in d-dimensions. Time complexity for DBScan algorithm is of order n^2 while for K-Means time complexity is linear in terms of n. Hence in both datasets K-Means is faster than DBScan algorithm. See Fig 9 and 10.

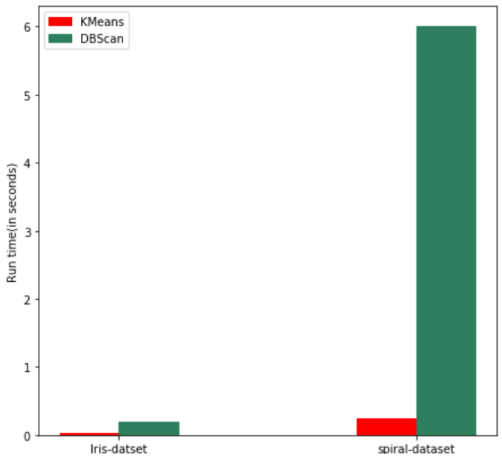

Fig 9: run time comparison

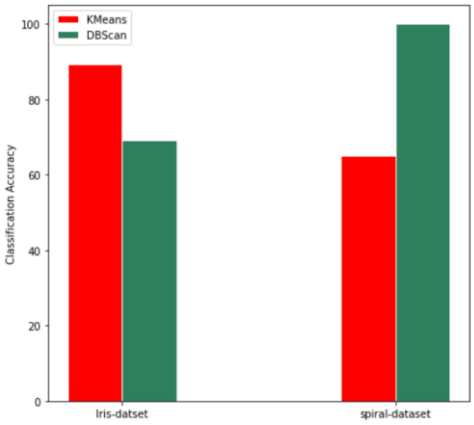

Fig 10: Classification accuracy comparison

4 EM algorithm

Expectation-Maximization algorithm assumes that all the points in the dataset came from the K-Gaussian Mixture models (GMMs). For a fix value of K, we have K means, K covariance matrix and K prior probabilities corresponding to K-GMMs. Let u_i, Σ_i and $P(C_i)$ are mean, covariance matrix and the prior probability of i^{th} cluster distribution. This implies probability density function at x is:

$$f(x) = \sum_{i=1}^{k} P(C_i) Normal(x \mid u_i, \Sigma_i)$$

Let $\theta = \{u_i, \Sigma_i, P(C_i) \mid \text{for every } i = \{1, 2, \ldots, K\}\}$

$P(\text{Dataset} \mid \theta) = \prod_{i=1}^{K} P(x_i \mid \theta)$

In EM algorithm, we start with random value of means, covariance matrix and prior probabilities. Using this we can find

$$P(C_j \mid x_i) = w_{ij} = \frac{Normal(x_i \mid u_j, \Sigma_j) * P(C_j)}{\sum_{j=1}^{K} Normal(x_i \mid u_j, \Sigma_j) * P(C_j)}$$

Now using this w_{ij} we can find u_j, Σ_j and $P(C_j)$ such that it maximizes the Expectation of log likelihood of P(Dataset | θ).

$$u_j = \frac{\sum w_{ij} x_i}{\sum w_{ij}}$$

$$\Sigma_j = \frac{\sum_{i=1}^{n} w_{ij} (x_i - u_j)(x_i - u_j)^T}{\sum w_{ij}}$$

$$P(C_j) = \left(\frac{i}{n}\right) * \sum w_{ij}$$

Repeat this Expectation-Maximization cycle till norm of change in means is less than ε. Clusters start changing their positions w.r.t mean and covariance matrix. Using these parameters θ, we can create a probability density function at any point x using

$$f(x) = \sum_{i=1}^{k} P(C_i) Normal(x \mid u_i, \Sigma_i)$$

See Fig 11, 12 to see change in clusters and pdf w.r.t. iterations.

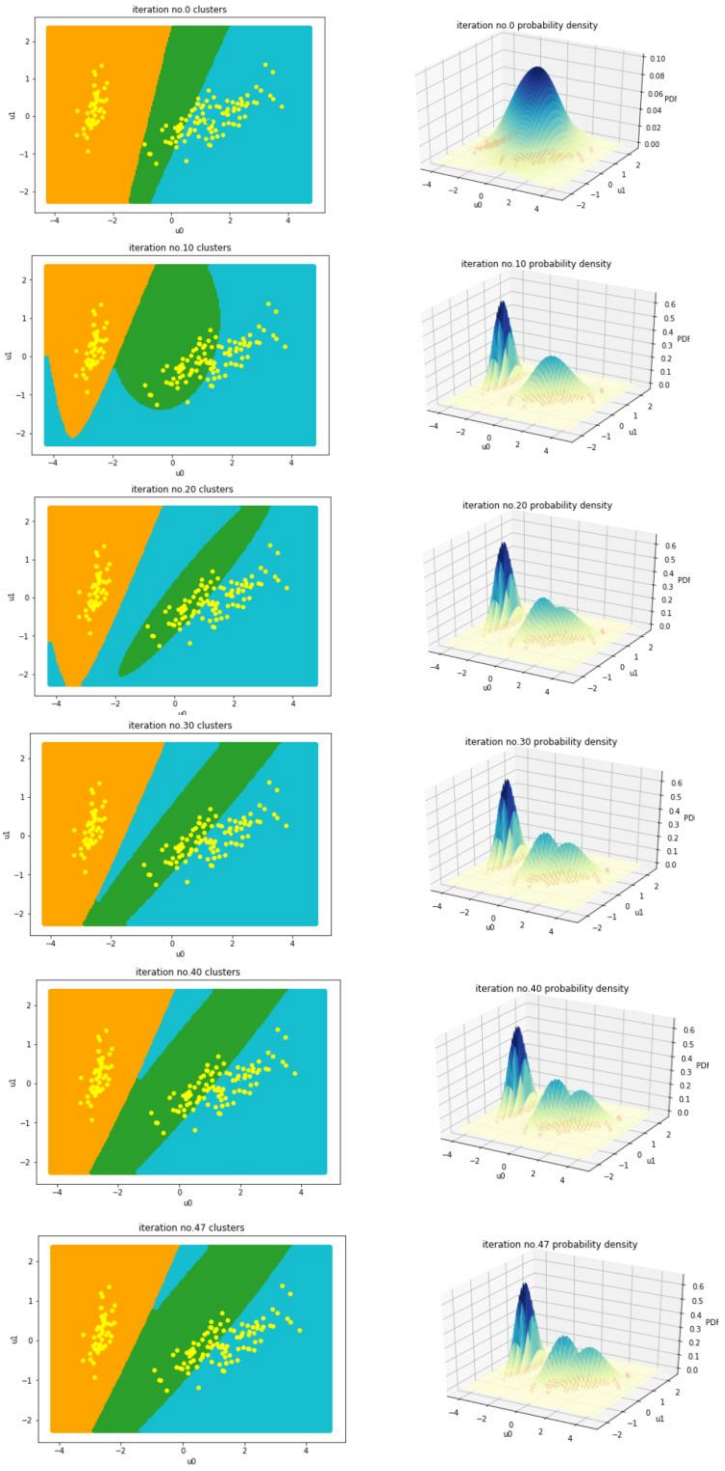

Fig 11: Change in pdf and clusters, for iris dataset. Convergence after 48 iterations

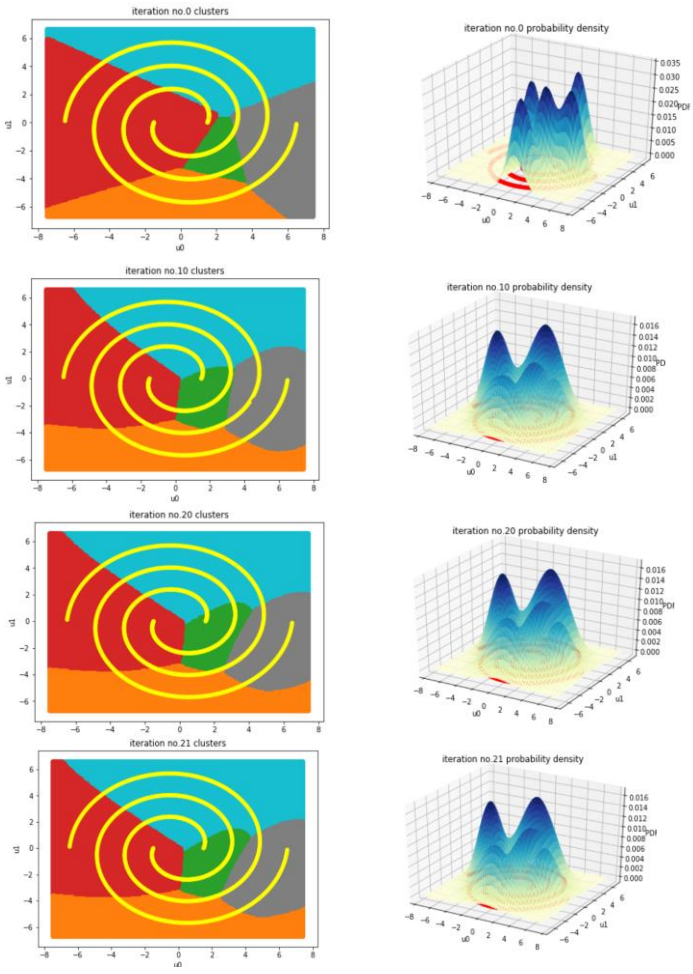

Fig 12: Change in pdf and clusters, for spiral dataset. Convergence after 22 iterations. K = 5 is selected from –Means SSE measure. For even K=2 EM doesn't perform good.

For iris-dataset EM algorithm perform better finding all the three clusters but as data in the spiral data set yields to non-convex clusters, EM algorithm is not performing well.

5 DENCLUE algorithm

DENCLUE assumes the probability density function comes from the Gaussian kernel. Estimated pdf using Gaussian kernel function is:

$$f_{hat}(x) = \frac{1}{nh^d}\sum_{i=1}^{n}\frac{1}{\left(\sqrt{2\pi}\right)^d}exp(\frac{-(x-x_i)^T(x-x_i)}{2h^2})$$

Assuming covariance matrix is d*d identity matrix. By applying gradient ascent we can find local maxima or density attractor for an point x.

$$x_{opt} = \frac{\sum_{i=1}^{n} K\left(\frac{x-x_i}{h}\right)x_i}{\sum_{i=1}^{n} K(\frac{x-x_i}{h})}$$

where K is the Gaussian kernel function.

If there exists a path between two local maxima or density attractors such that each point in the path has probability value greater than ξ, then the cluster or region of both density attractors can be merged to take single cluster.

Estimated probability density function using Gaussian kernels with different values of h can be seen in Fig 13 and 14.

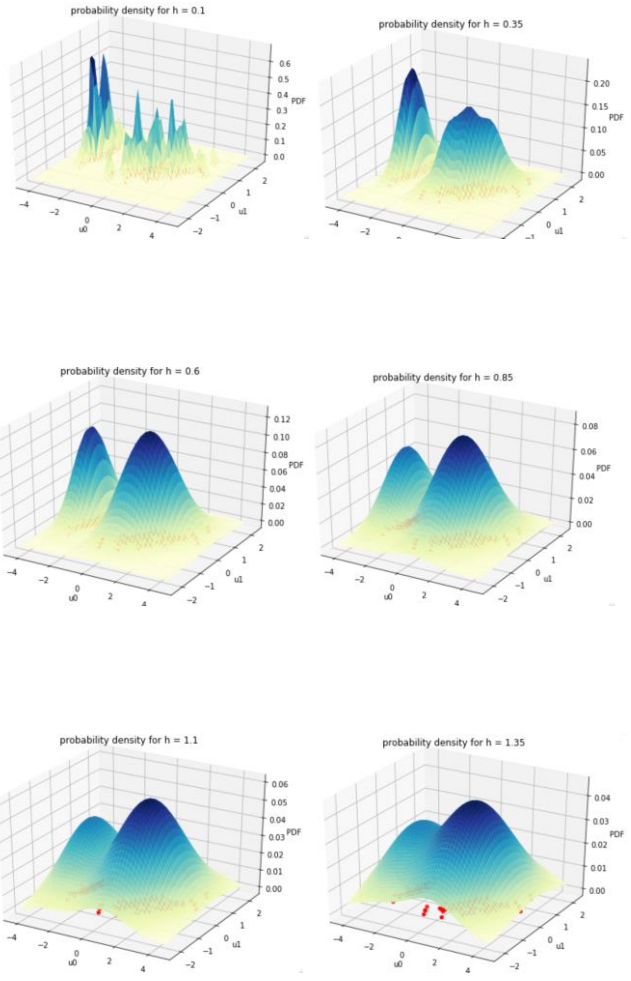

Fig 13: Gaussian kernel pdf for PCA applied iris dataset

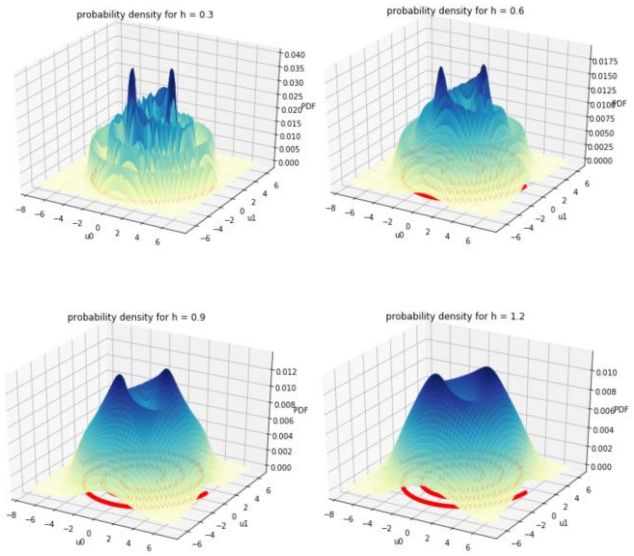

Fig 14: Gaussian kernel pdf for spiral dataset

With the increase in h, number of local maxima decreases as weights to the local points smoothen out due to increase in variance of the normal distribution. Probability density function has peaks near to the points of dataset for lower values of h. For h = epsilon (in DBScan) and ξ = min_points (in DBScan), both DBScan and DENCLUE performs similar. For iris-dataset EM algorithm perform better finding all the three clusters but as data in the spiral data set yields to non-convex clusters, EM algorithm is not performing well. Although DENCLUE performs decent on spiral dataset but EM have a clear edge on iris dataset yielding same number of clusters as the labels.

REFERNECES:

[1]: Iris Dataset (https://archive.ics.uci.edu/ml/datasets/iris)

[2]: Spiral Dataset (https://github.com/milaan9/Clustering-Datasets).

[3]: https://en.wikipedia.org/wiki/Multivariate_normal_distribution

[4]: https://scikitlearn.org/stable/modules/generated/sklearn.decomposition.PCA.html