Cross-Entropy Clustering

J. Tabor

Faculty of Mathematics and Computer Science, Jagiellonian University, Łojasiewicza 6, 30-348 Kraków, Poland

P. Spurek

Faculty of Mathematics and Computer Science, Jagiellonian University, Łojasiewicza 6, 30-348 Kraków, Poland

Abstract

We build a general and easily applicable clustering theory, which we call cross-entropy clustering (shortly CEC), which joins the advantages of classical k-means (easy implementation and speed) with those of EM (affine invariance and ability to adapt to clusters of desired shapes). Moreover, contrary to k-means and EM, *CEC finds the optimal number of clusters by automatically removing groups which have negative information cost.*

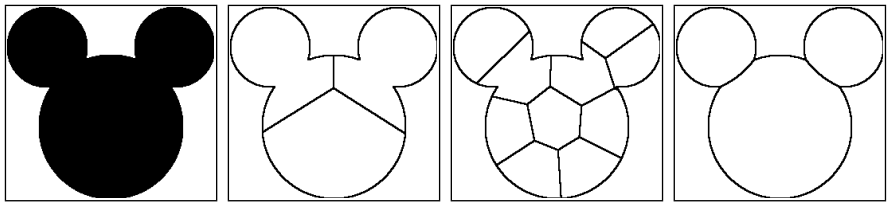
Although CEC, like EM, can be build on an arbitrary family of densities, in the most important case of Gaussian CEC the division into clusters is affine invariant.

Keywords: clustering, cross-entropy, memory compression

1. Introduction

Clustering plays a basic role in many parts of data engineering, pattern recognition and image analysis [1, 2, 3, 4, 5]. Thus it is not surprising that there are many methods of data clustering, many of which however inherit the deficiencies of the first method called k-means [6, 7]. Since k-means has the tendency to divide the data into spherical shaped clusters of similar sizes, it is not affine invariant and does not deal well with clusters of various sizes.

Email addresses: jacek.tabor@ii.uj.edu.pl (J. Tabor), przemyslaw.spurek@ii.uj.edu.pl (P. Spurek)



(a) Mouse-like set. (b) k-means with $k = 3$. (c) k-means with $k = 10$. (d) Spherical CEC.

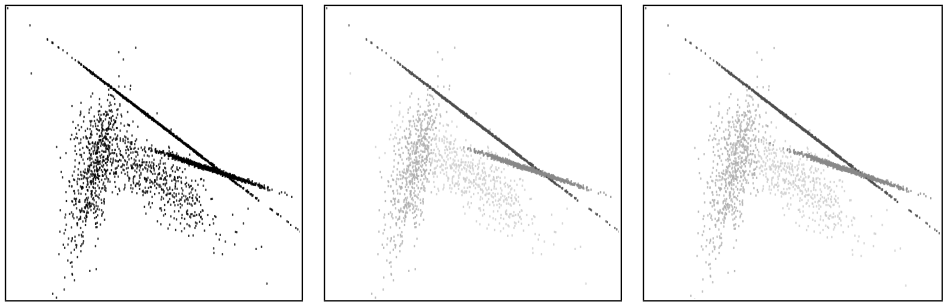
Figure 1: Clustering of the uniform density on mouse-like set (Fig. 1(a)) by standard k-means algorithm with $k = 3$ (Fig. 1(b)) and $k = 10$ (Fig. 1(c)) compared with Spherical CEC (Fig. 1(d)) with initially 10 clusters (finished with 3).

This causes the so-called mouse-effect, see Figure 1(b). Moreover, it does not find the right number of clusters, see 1(c), and consequently to apply it we usually need to use additional tools like gap statistics [8, 9]. Since k-means has so many disadvantages, one can ask why it is so popular. One of the possible answers lies in the fact that k-means is simple to implement and very fast compared to more advanced clustering methods[1] like EM [12, 13].

In our paper we construct a general cross-entropy clustering (CEC) theory which simultaneously joins the clustering advantages of classical k-means and EM. The motivation of CEC comes from the observation that it is often profitable to use various compression algorithms specialized in different data types. We apply this observation in reverse, namely *we group/cluster those data together which are compressed by one algorithm from the preselected set of compressing algorithms.* In development of this idea we were influenced by the classical Shannon Entropy Theory [14, 15, 16, 17] and Minimum Description Length Principle [18, 19]. In particular we were strongly inspired by the application of MDLP to image segmentation given in [20, 21]. A close approach from the Bayesian perspective can be also found in [22, 23].

The above approach allows us automatic reduction of unnecessary clusters: contrary to the case of classical k-means or EM, there is a cost of using each cluster. To visualize our theory let us look at the results of Gaussian

[1]This is excellently summarized in the third paragraph of [10]: "[...] The weaknesses of k-MEANS result in poor quality clustering, and thus, more statistically sophisticated alternatives have been proposed. [...] While these alternatives offer more statistical accuracy, robustness and less bias, they trade this for substantially more computational requirements and more detailed prior knowledge [11]."



(a) Data coming from mixture of 4 Gaussians.

(b) EM clustering with 4 Gaussians.

(c) CEC clustering with initially 10 Gaussians.

Figure 2: Comparison of clustering of mixture of 4 gaussians by EM (with 4 gaussian densities) and Gaussian CEC starting from 10 initial clusters.

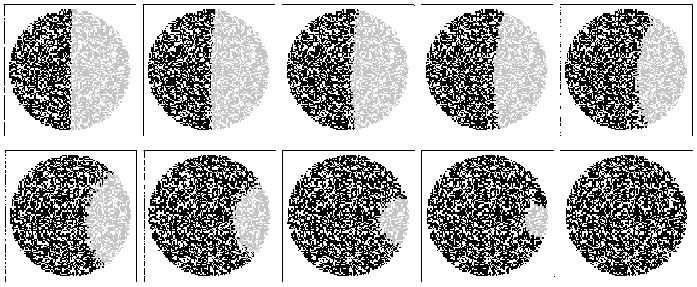


Figure 3: Reduction of cluster by the spherical CEC.

CEC given in Figure 2(c), where we started with $k = 10$ initial randomly chosen clusters which were reduced automatically by the algorithm. The step-by-step view at this process can be seen on Figure 3, in which we illustrate the subsequent steps of the Spherical CEC on random data lying uniformly inside the circle, and divided initially at two almost equal parts.

The clustering limitations of CEC are similar to that of EM, namely we divide the data into clusters of shapes which are reminiscent of the level sets of the family of the densities used. In particular, contrary to the density clustering [24] with use of Gaussian CEC we will not build clusters of complicated shapes. Moreover, in an analogy to k-means, CEC strongly depends on the initial choice of clusters. This is the reason why in the paper we always started CEC at least twenty times from randomly chosen initial conditions to avoid arriving at the local minimum of the cost function. Let us mention that there are clustering methods, see [25], which allow to better minimize

the global minimum, however at the cost of the fixed number of clusters. The advantage comparing to most classical clustering methods [26] lies in the fact that we need only the maximal number of clusters, while we keep the same complexity as k-means.

There are a few probabilistic methods which try to estimate the correct number of clusters. For example in [27] authors use the generalized distance between Gaussian mixture models with different components number by using the Kullback–Leibler divergence [14, 16]. Similar approach is presented in [28] (Competitive Expectation Maximization) which uses Minimum Message Length criterion [29]. In practice one can also directly use the MDLP in clustering [30]. The basic ideological difference lies in the fact that in MDLP we want to take into account the total memory cost of building the model, while in our case, like in EM, we use the classical entropy approach, and therefore assume that the memory cost of remembering the Gaussian (or in general density) parameters is zero.

Another modern clustering method worth mentioning is clearly support vector clustering [31]. In its basic form SVM allows separates the data with the use of hyperplanes while CEC (similarly as EM) allows the quadratic discriminant functions [32]. However, in its general form with use of kernel functions support vector clustering will allow to cluster the date into more complicated sets then CEC, usually at a cost larger numerical complexity. Consequently the CEC framework presented in the paper cannot cluster sufficiently well datasets presented in [32], since they are not well divided into Gaussian-shaped clusters.

For the convenience of the reader we now briefly summarize the contents of the article. The next section is devoted to the gentle introduction to the basic properties of CEC. In particular we show that if the data comes from the known number of Gaussian densities, the basic results of CEC and EM clustering are similar. At the end of this section we discuss applications of CEC on real data-sets. In the following section we introduce notation and recall the necessary information concerning relative entropy. In the fourth section we provide a detailed motivation and explanation of our main idea which allows to reinterpret the cross-entropy for the case of many "coding densities". We also show how to apply classical Lloyds and Hartigan approaches to cross-entropy minimizations.

The last section contains applications of our theory to clustering with respect to various Gaussian subfamilies. We put a special attention on the question whether the given group of data should be divided into two separate

clusters.

First we investigate the most important case of Gaussian CEC and show that it reduces to the search for the partition $(U_i)_{i=1}^k$ of the given data-set U which minimizes the objective cost function:

$$\frac{N}{2}\ln(2\pi e) + \sum_{i=1}^{k} p(U_i)\cdot[-\ln(p(U_i)) + \frac{1}{2}\ln\det(\Sigma_{U_i})],$$

where $p(V)$ denotes the probability of choosing set V and Σ_V denotes the covariance matrix of the set V.

Then we study clustering based on the Spherical Gaussians, that is those with covariance proportional to identity. Comparing Spherical CEC to classical k-means we obtain that clustering is scale and translation invariant and clusters do not tend to be of fixed size. Consequently we do not obtain the mouse effect, see Figure 1(d). To apply Spherical clustering we need the same information as in the classical k-means: in the case of k-means we seek the splitting of the data $U \subset \mathbb{R}^N$ into k sets $(U_i)_{i=1}^k$ such that the value of $\sum_{i=1}^{k} p(U_i)\cdot D_{U_i}$ is minimal, where $D_V = \frac{1}{\mathrm{card}(V)}\sum_{v\in V}\|v-\mathrm{m}_V\|^2$ denotes the mean within cluster V sum of squares (and m_V is the mean of V). It occurs that the Gaussian spherical clustering in $\mathbb{R}^N$ reduces to minimization of

$$\frac{N}{2}\ln(\frac{2\pi e}{N}) + \sum_{i=1}^{k} p(U_i)\cdot[-\ln(p(U_i)) + \frac{N}{2}\ln D_{U_i}].$$

Next we proceed to the study of clustering by Gaussians with fixed covariance. We show that in the case of bounded data the optimal amount of clusters is bounded above by the maximal cardinality of respective ε-net in the convex hull of the data. We finish our paper with the study of clustering by Gaussian densities with covariance equal to $s\mathrm{I}$ and prove that with s converging to zero we obtain the classical k-means clustering (for the similar type of result from the Bayesian point of view we refer the reader to [22]). We also show that with s growing to ∞ data will form one big group.

2. Discussion of CEC

Before proceeding to the more complicated theory we discuss in this section the basic use of CEC and present the intuition behind it. Since in

practical implementations CEC can be viewed as a generalized and modified version of the classical k-means its complexity is that of k-means and one can easily adapt most ideas used in various versions of k-means to CEC. Since CEC is in many aspects influenced by EM we first present a comparison between CEC and EM and summarize the main similarities and differences.

It occurs, see Figure 2, that for the data coming from k "distinct" Gaussian distributions the effects of CEC and EM clustering are very close. However, the basic and crucial advantage of CEC over EM comes from the fact that CEC removes unnecessary clusters while being simpler then EM.

To explain the above consider density f and fixed densities $f_1, \ldots, f_k$ by combination of which we want to approximate f. The basic goal of EM is to find probabilities $p_1, \ldots, p_k$ such that the approximation

$$f \approx p_1 f_1 + \ldots + p_k f_k; \tag{2.1}$$

is optimal, while CEC aims at optimizing (see Theorem 4.2 and Remark 4.1)

$$f \approx \max(p_1 f_1, \ldots, p_k f_k). \tag{2.2}$$

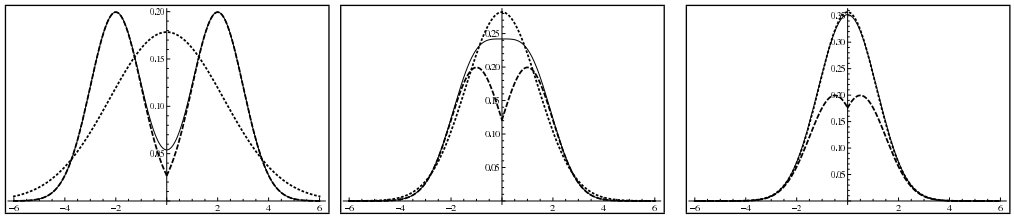
Crucial consequence of the formula (2.2) is that contrary to the earlier approaches based on MLE we approximate f not by a density, as is the case in (2.1), but subdensity. Observe also that the density approximation given by EM will typically be better then that given by CEC.

Formula (2.2) explains why, contrary to EM, in CEC with the increase of number of clusters we do not always improve the approximation. In consequence it is often profitable to merge two groups. In article [33] authors present slightly different approach which uses dimension reduction and merging condition for reduction of Gaussian components in mixture of densities. In the following example we discuss when clusters are automatically merged in CEC procedure for simple mixture of two Gaussians.

Example 2.1. *Consider two Gaussian densities $\mathcal{N}(s,1)$ and $\mathcal{N}(-s,1)$ with normalized covariance and means $s \geq 0$ and $-s$. Suppose that we want to compare the approximation of the mixture density*

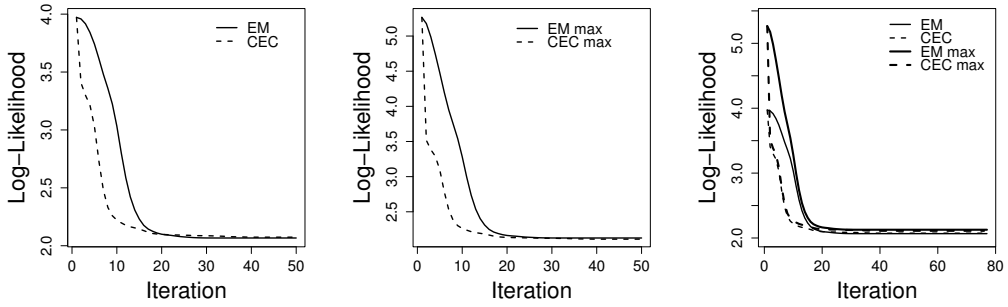
$$f_s := \frac{1}{2}\mathcal{N}(s,1) + \frac{1}{2}\mathcal{N}(-s,1)$$

by one gaussian with that of two gaussians. In the case of EM the approximation by two gaussians will be exact and will return f_s.



(a) $s = 2$ (b) $s = 1$ (c) $s = 0.5$

Figure 4: CEC-type approximation of Gaussian mixture (black line) with one (dotted line) and two gaussians (dashed line).



(a) MLE: $H^{\times}(\mu \| \sum_i p_i f_i)$. (b) CEC: $H^{\times}(\mu \| \max_i p_i f_i)$. (c) MLE and CEC.

Figure 5: EM vs CEC approximation.

Now consider the case of approximation of the form given in (2.2)*. Suppose for simplicity that we approximate by one Gaussian with center at zero and the same standard deviation as f_s, while the analogue approximation by two gaussians will be given by* $\max(\frac{1}{2}\mathcal{N}(s,1), \frac{1}{2}\mathcal{N}(-s,1))$*. Then with large s it is clearly better to approximate f_s with two densities, see Figure 4(a), with s becoming smaller, see Figure 4(b), the decision what is better is not so easy to make at the first glance, while with s small, see Figure 4(c), it is better two approximate f_s by one density. For more detailed study of the above case we refer the reader to Example 5.1.*

However, if the data comes from relatively well-separated Gaussians as is in Figure 2, if we know the number of Gaussians the results of EM and CEC are similar, see Figure 5. In Figure 5(a) we present the graphs with argument denoting the number of iterations through the data-set of the mean value of

Dataset	Nr. of clusters		Rand index		
	original	CEC	EM vs Data	CEC vs Data	EM vs CEC
4 Gaussians	4	4	0.9559388	0.950834	0.982119
Glass data	6	7	0.4888333	0.6898337	0.2680005
Balance data	4	3	0.5391538	0.537559	0.6471846
Yeast data	10	4	0.2227028	0.4540007	0.5875602

Table 1: Comparison of the CEC with clustering by EM.

MLE=$H^{\times}(\mu\|\sum_i p_i f_i)$ obtained for EM by classical gaussian mixtures and Gaussian CEC (we repeated the experiment, which 20 times started from random initial conditions, 100 times) on the data from Figure 2, where $H^{\times}(\mu\|f)$ denotes the cross-entropy of data represented by measure μ with respect to the subdensity f, for detailed explanation see the next Section. In Figure 5(b) we present its analogue for the value minimized by CEC=$H^{\times}(\mu\|\max_i p_i f_i)$, while in 5(c) we present both the above functions on one graph.

As we see, typically CEC at the beginning iterations was decreasing faster then EM, mainly due to the fact that in CEC we used the Hartigans approach which usually decreases faster then the Lloyds approach used in EM. This implies that even for the computation of EM it may be profitable to begin with first few iterations of CEC.

Consequently, if the data is Gaussian-shaped and well-separated, the results of CEC and EM are similar if we know the number of Gaussians in advance. However, in general the results may differ, see the Table 1 in which we compared the effects of CEC with EM on some real-data sets (except for the four Gaussians) taken from the uci-repository `http://archive.ics.uci.edu/ml/`.

If we do not have enough knowledge about the dataset, the pre-partition could directly lead to mis-partition of the data and further affect the clustering. In the case of non–Gaussian data initial partition can influence the final clustering. In Table 2 we present three standard initial conditions. We apply 1000 CEC instances on Wine Data Set from the UCI Repository (with $k = 3$ clusters) with different initial partitioning which use three common approaches. In the first we add elements to cluster randomly (random initialization). The second method (like in classical k–means) is based on choosing randomly centers of clusters and assigning elements to the closest one. In the last case we apply k–means++ method [34]. Numerical experiments show

	Mean nr. of iteration	Log–likelihood		
		mean start	mean end	min
random	11.1	3257.4	2966.0	2769.3
k–means	7.2	3049.8	2985.7	2395.1
k–means ++	7.4	3036.2	2958.2	2329.5

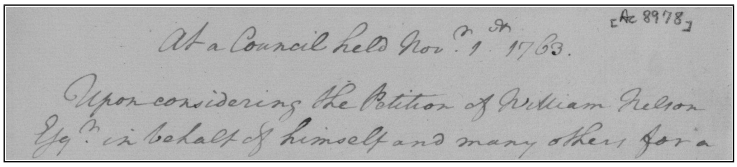
Table 2: Effect of CEC with different initial partitioning in the case of Wine Data Set with $k = 3$.

that most suitable method is k–means++.

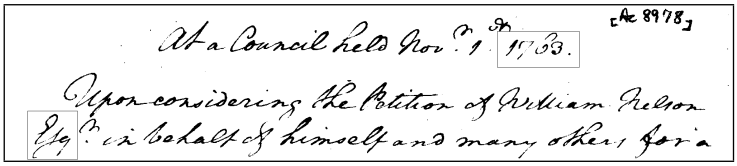
At the end of this section let us mention that the basic practical applications of CEC in image analysis. Let us first show how CEC can be used in image preprocessing in handwriting recognition.

As we know, in the first step we typically apply thresholding. In this case the celebrated Ostu method [35] comes in mind. Since it is in fact equivalent to application of k-means on the histogram of the picture with $k = 2$, it is very fast, as contrary to more dimensional vector it is sufficient to put the border between two groups at all gray levels between 0 and 255. Consequently, we can search through all possible division of the data-space, and therefore Otsu threshold optimal result.

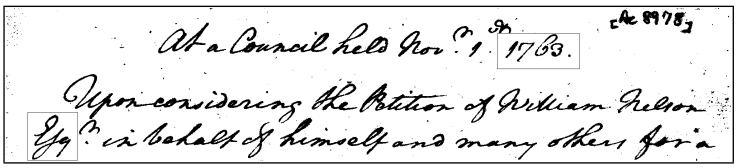
However, although Otsu method works usually very good for the segmentation of pictures, it does not deal so well with the scans of the documents. We will explain it by studying the example given in Figure 6, taken from the database from DIBCO 2009 contest [36]. The reason behind this is that k-means has the natural tendency to divide the data into two groups of similar within cluster sum of squares, and therefore it does not cope well with the case when we have prevailing number of elements from the background, compared to the foreground. In this case as we see in the histogram given in Figure 7, Otsu threshold will have the tendency to put the barrier to much into the foreground (since background is usually more concentrated and consists of more points), see Figure 8 for the consequences on some chosen details. Consequently, as we see after Otsu thresholding we loose some important details which can be of crucial importance in further processing. On the other hand as we see in histogram on Figure 7, CEC will put the barrier in a more reasonable place, since it is in natural way scale invariant (and consequently it copes well with separation of two gaussians with different standard deviations and cardinalities). Observe that one cannot use here EM with the same numerical efficiency as CEC, since for EM (contrary



(a) Scanned text.



(b) Otsu thresholding.



(c) CEC thresholding.

Figure 6: Original scanned text from DIBCO 2009 competition [36] and its comparison with Otsu and CEC thresholdings with two marked details.

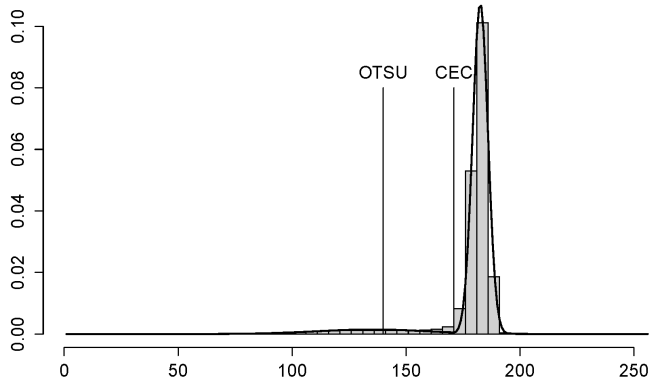
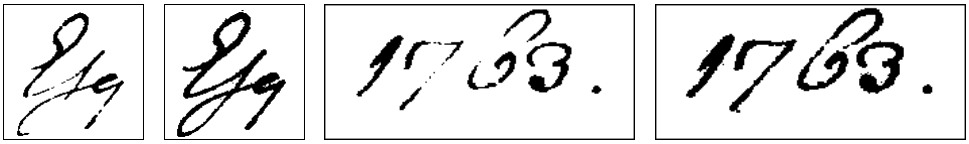


Figure 7: Histogram with Otsu and CEC thresholds with CEC density approximation by two gaussians.



(a) Otsu: detail 1. (b) CEC: detail 1. (c) Otsu: detail 2. (d) CEC: detail 2.

Figure 8: Otsu vs CEC thresholding on details.

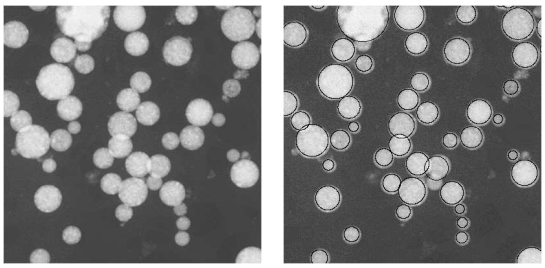
(a) Original image. (b) CEC segmentation.

Figure 9: CEC–base image segmentation.

to Ostu method or CEC) we cannot go through all possible positions of the barrier (these depend on initial condition).

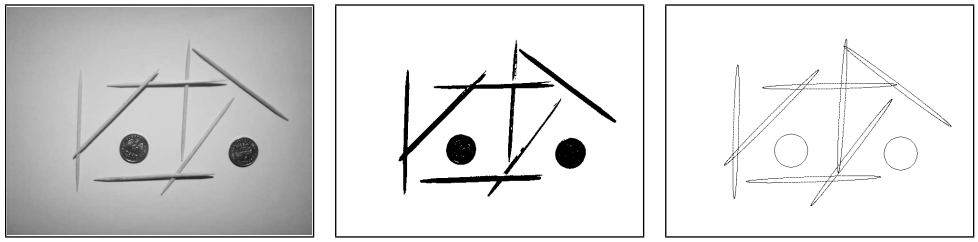
Another more advanced applications can be found in [37, 38, 39], which we briefly discuss beyond. Paper [37] contains a typical applications of clustering in the image segmentation problem. An image is divided into 8×8 squares which are represented as a 192 (we use RGB format) dimensional vectors. Then the PCA is used for dimension reduction (we obtain data in $\mathbb{R}^4$). Projection speeds up the the clustering and, as numerical experiments show, does not affect essentially the segmentation results. Because color and position represent different quantities, we additionally add pixels coordinates, and thus arrive at data in $\mathbb{R}^6$. The effects are close to that from [21], where the MDLP approach was used. In Fig. 9 we present results of CEC segmentation on the example from The Berkeley Segmentation Database (`http://www.eecs.berkeley.edu/Research/Projects/CS/vision/bsds`). Since general Gaussian clustering is affine invariant, authors in [37] show that the results are resistant to possible rescaling of the original picture.

In [38] we have used spherical CEC to discover the circle-like object of



(a) Original image. (b) Segmentation obtained by spherical CEC.

Figure 10: CEC segmentation of Electron Microscopy Images.



(a) (b) (c)

Figure 11: The result of CEC based ellipse detection algorithm: Fig 11(a) – original image, Fig 11(b) – binarized image, the input for algorithm, Fig 11(c) – outcome of algorithm.

various sizes on the Electron Microscopy Images of Ghanite nano-particles, see Fig. 10. We applied the spherical CEC on the binarized images. Observe that in this case one could not easily effectively use Gaussian mixture models based on spherical Gaussians, since EM does not discover the number of groups. Morever, the use of spherical Gaussians in EM needs complicated numerical optimizations [40], [41, Chapter 6]. Consequently, the advantage of CEC is the speed of the calculations and the ability to discover the right number of circle shapes of different sizes.

In [39] the method of detection ellipse-like shapes is presented. Authors use CEC with different Gaussian models which allow to distinguish elliptical shapes which represent different objects. Consequently, semi-supervised classification based on CEC is obtained. The effects of the algorithm on can be observed on Fig. 11. The method allows to divide the data into groups containing ellipses of similar shapes.

3. Cross-entropy

Since CEC is based on choosing the optimal (from the memory point of view) coding algorithms, we first establish notation and present the basics of cross-entropy compression. Assume that we are given a discrete probability distribution ν on a finite set $X = \{x_1, \ldots, x_m\}$ which attains the values x_i with probabilities f_i. Then, roughly speaking [14], the optimal code-length[2] is given by the entropy

$$h(\nu) := \sum_{i=1}^{m} f_i \cdot (-\ln f_i) = \sum_{i=1}^{m} \mathrm{sh}(f_i),$$

where $\mathrm{sh}(x)$ denotes the Shannon function defined by $-x \cdot \ln x$ if $x > 0$ and $\mathrm{sh}(0) := 0$.

Let us proceed to the case of continuous probability measure ν on $\mathbb{R}^N$ (with density f_ν). The role of entropy is played by *differential entropy* (which corresponds to the limiting value of discrete entropy of coding with quantization error going to zero [14]):

$$h(\nu) := \int f_\nu(x) \cdot (-\ln f_\nu(x))dx = \int \mathrm{sh}(f_\nu(x))dx,$$

where f_ν denotes the density of the measure ν.

In our case we need to consider codings produced by *subdensities*, that is nonnegative measurable functions $f : \mathbb{R}^N \to \mathbb{R}_+$ such that $\int_{\mathbb{R}^N} f(x)dx \leq 1$. Thus the differential code-length connected with subdensity f is given by

$$l(x) = -\ln f(x). \tag{3.1}$$

From now on, if not specified, by μ we denote either continuous or discrete probability measure on $\mathbb{R}^N$. If we code measure μ by the code optimized for a subdensity f we arrive at the definition of cross-entropy.

Definition 3.1. *We define the* cross-entropy of μ with respect to subdensity f *by:*

$$H^{\times}(\mu \| f) := \int -\ln f(y)d\mu(y). \tag{3.2}$$

[2]We accept arbitrary, not only integer, code-lengths and compute the value of entropy in NATS.

It is well-known that if μ has density f_μ, the minimum in the above integral over all subdensities is obtained for $f = f_\mu$ (and consequently the cross-entropy is bounded from below by the differential entropy). One can easily get the following:

Observation 3.1. *Let f be a given subdensity and A an invertible affine operation. Then $H^\times(\mu \circ A^{-1} \| f_A) = H^\times(\mu \| f) + \ln|\det A|$, where f_A is a subdensity defined by*

$$f_A : x \to f(A^{-1}x)/|\det A|, \tag{3.3}$$

and $\det A$ *denotes the determinant of the linear component of A.*

In our investigations we will be interested in (optimal) coding for μ by elements of a set of subdensities $\mathcal{F}$, and therefore we put

$$H^\times(\mu \| \mathcal{F}) := \inf\{H^\times(\mu \| f) : f \in \mathcal{F}\}.$$

One can easily check that if $\mathcal{F}$ consists of densities then the search for $H^\times(\mu \| \mathcal{F})$ reduces to the maximum likelihood estimation of measure μ by the family $\mathcal{F}$. Thus by $\mathrm{MLE}(\mu \| \mathcal{F})$ we will denote the set of all subdensities from $\mathcal{F}$ which realize the infimum:

$$\mathrm{MLE}(\mu \| \mathcal{F}) := \{f \in \mathcal{F} : H^\times(\mu \| f) = H^\times(\mu \| \mathcal{F})\}.$$

In proving that the clustering is invariant with respect to the affine transformation A we will use the following simple corollary of Observation 3.1:

Corollary 3.1. *Let $\mathcal{F}$ be the subdensity family and $A : \mathbb{R}^N \to \mathbb{R}^N$ an invertible affine operation. By $\mathcal{F}_A$ we denote $\{f_A : f \in \mathcal{F}\}$, where f_A is defined by* (3.3)*. Then*

$$H^\times(\mu \circ A^{-1} \| \mathcal{F}_A) = H^\times(\mu \| \mathcal{F}) + \ln|\det A|. \tag{3.4}$$

By m_μ and Σ_μ we denote the mean and covariance of the measure μ, that are

$$\mathrm{m}_\mu = \tfrac{1}{\mu(\mathbb{R}^N)} \int x \, d\mu(x), \Sigma_\mu = \tfrac{1}{\mu(\mathbb{R}^N)} \int (x - \mathrm{m}_\mu)(x - \mathrm{m}_\mu)^T d\mu(x).$$

For measure μ and measurable set U such that $0 < \mu(U) < \infty$ we introduce the probability measure $\mu_U(A) := \frac{1}{\mu(U)}\mu(A \cap U)$, and use the abbreviations

$$\begin{aligned} \mathrm{m}_U^\mu &:= \mathrm{m}_{\mu_U} = \tfrac{1}{\mu(U)} \textstyle\int_U x \, d\mu(x), \\ \Sigma_U^\mu &:= \Sigma_{\mu_U} = \tfrac{1}{\mu(U)} \textstyle\int_U (x - \mathrm{m}_\mu)(x - \mathrm{m}_\mu)^T \, d\mu(x). \end{aligned}$$

Given symmetric positive matrix Σ, we recall that by the Mahalanobis distance [42, 43] we understand $\|x-y\|_\Sigma := (x-y)^T\Sigma^{-1}(x-y)$. By $\mathcal{N}(\mathrm{m},\Sigma)$ we denote the normal density with mean m and covariance Σ.

The basic role in Gaussian cross-entropy minimization is played by the following result which says that we can reduce computation to gaussian families. Since its proof is essentially known part of MLE, we provide here only its short idea.

Theorem 3.1. *Let μ be a discrete or continuous probability measure with well-defined covariance matrix, and let $\mathrm{m}\in\mathbb{R}^N$ and positive-definite symmetric matrix Σ be given.*

Then

$$H^\times\big(\mu\|\mathcal{N}(\mathrm{m},\Sigma)\big) = H^\times\big(\mu_\mathcal{G}\|\mathcal{N}(\mathrm{m},\Sigma)\big),$$

where $\mu_\mathcal{G}$ denotes the probability measure with Gaussian density of the same mean and covariance as μ (that is the density of $\mu_\mathcal{G}$ equals $\mathcal{N}(\mathrm{m}_\mu,\Sigma_\mu)$).

Consequently

$$H^\times\big(\mu\|\mathcal{N}(\mathrm{m},\Sigma)\big) = \frac{N}{2}\ln(2\pi)+\frac{1}{2}\|\mathrm{m}-\mathrm{m}_\mu\|^2_\Sigma+\frac{1}{2}\mathrm{tr}(\Sigma^{-1}\Sigma_\mu)+\frac{1}{2}\ln\det(\Sigma). \quad (3.5)$$

Sketch of the proof. We consider the case when μ is a continuous measure with density f_μ. One can easily see that by applying trivial affine transformations and (3.4) it is sufficient to prove (3.5) in the case when $\mathrm{m}=0$ and $\Sigma=\mathrm{I}$. Then we have

$$\begin{aligned}
&H^\times(\mu\|\mathcal{N}(0,\mathrm{I})) = \textstyle\int f_\mu(x)\cdot[\frac{N}{2}\ln(2\pi)+\frac{1}{2}\ln\det(\mathrm{I})+\frac{1}{2}\|x\|^2]dx\\
&=\tfrac{N}{2}\ln(2\pi)+\tfrac{1}{2}\textstyle\int f_\mu(x)\|(x-\mathrm{m}_\mu)+\mathrm{m}_\mu\|^2dx\\
&=\tfrac{N}{2}\ln(2\pi)+\tfrac{1}{2}\textstyle\int f_\mu(x)[\|x-\mathrm{m}_\mu\|^2+\|\mathrm{m}_\mu\|^2+2(x-\mathrm{m}_\mu)\circ\mathrm{m}_\mu]dx\\
&=\tfrac{N}{2}\ln(2\pi)+\tfrac{1}{2}\mathrm{tr}(\Sigma_\mu)+\tfrac{1}{2}\|\mathrm{m}_\mu\|^2.
\end{aligned}$$

□

By $\mathcal{G}$ we denote the set of all normal densities, while by $\mathcal{G}_\Sigma$ we denote the set of all normal densities with covariance Σ. As a trivial consequence of the Theorem 3.1 we obtain the following proposition.

Proposition 3.1. *Let Σ be a fixed positive symmetric matrix. Then* $\mathrm{MLE}(\mu\|\mathcal{G}_\Sigma) = \{\mathcal{N}(\mathrm{m}_\mu,\Sigma)\}$ *and* $H^\times(\mu\|\mathcal{G}_\Sigma) = \frac{N}{2}\ln(2\pi)+\frac{1}{2}\mathrm{tr}(\Sigma^{-1}\Sigma_\mu)+\frac{1}{2}\ln\det(\Sigma)$.

Now we consider cross-entropy with respect to all normal densities.

Proposition 3.2. *We have* $\mathrm{MLE}(\mu\|\mathcal{G}) = \{\mathcal{N}(\mathrm{m}_\mu, \Sigma_\mu)\}$ *and*

$$H^\times(\mu\|\mathcal{G}) = \frac{1}{2}\ln\det(\Sigma_\mu) + \frac{N}{2}\ln(2\pi e). \tag{3.6}$$

Proof. Since entropy is minimal when we code a measure by its own density, we easily obtain that

$$H^\times(\mu\|\mathcal{G}) = H^\times(\mu_\mathcal{G}\|\mathcal{G}) = H^\times(\mu_\mathcal{G}\|\mathcal{N}(\mathrm{m}_\mu, \Sigma_\mu))$$

$$= H(\mu_\mathcal{G}) = \frac{1}{2}\ln\det(\Sigma_\mu) + \frac{N}{2}\ln(2\pi e).$$

Consequently the minimum is realized for $\mathcal{N}(\mathrm{m}_\mu, \Sigma_\mu)$. □

Due to their importance and simplicity we also consider Spherical Gaussians $\mathcal{G}_{(\cdot\mathrm{I})}$, whose covariance matrix is proportional to I:

$$\mathcal{G}_{(\cdot\mathrm{I})} = \bigcup_{s>0} \mathcal{G}_{s\mathrm{I}}.$$

We will need the denotation for the mean squared distance from the mean

$$D_\mu := \int \|x - \mathrm{m}_\mu\|^2 d\mu(x) = \mathrm{tr}(\Sigma_\mu),$$

which will play in Spherical Gaussians the analogue of covariance. As is the case for the covariance, we will use the abbreviation

$$D_U^\mu := D_{\mu_U} = \frac{1}{\mu(U)}\int_U \|x - \mathrm{m}_U^\mu\|^2 d\mu(x).$$

Observe, that if $\Sigma_\mu = s\mathrm{I}$ then $D_\mu = Ns$. In the case of measures on the real line $\sqrt{D_\mu}$ is exactly the standard deviation of the measure μ. It occurs that D_μ can be naturally interpreted as a square of the "mean radius" of the measure μ: for the uniform probability measure μ on the sphere $S(x, R) \subset \mathbb{R}^N$ we clearly get $\sqrt{D_\mu} = R$.

By λ we denote the Lebesgue measure on $\mathbb{R}^N$. Recall that λ_U denotes the probability measure defined by $\lambda_U(A) := \lambda(A \cap U)/\lambda(U)$.

Proposition 3.3. *We have* $\mathrm{MLE}(\mu\|\mathcal{G}_{(\cdot\mathrm{I})}) = \{\mathcal{N}(\mathrm{m}_\mu, \frac{D_\mu}{N}\mathrm{I})\}$ *and*

$$H^\times(\mu\|\mathcal{G}_{(\cdot\mathrm{I})}) = \frac{N}{2}\ln(D_\mu) + \frac{N}{2}\ln(2\pi e/N). \tag{3.7}$$

Proof. Clearly by Proposition 3.1

$$H^{\times}(\mu\|\mathcal{G}_{(\cdot\mathrm{I})}) = \inf_{s>0} H^{\times}(\mu\|\mathcal{G}_{s\mathrm{I}}) = \inf_{s>0}\left(\frac{1}{2s}D_{\mu} + \frac{N}{2}\ln s + \frac{N}{2}\ln(2\pi)\right).$$

Now by easy calculations we obtain that the above function attains minimum for $s = D_{\mu}/N$ and equals the RHS of (3.7). □

At the end we consider the cross-entropy with respect to $\mathcal{G}_{s\mathrm{I}}$ (spherical Gaussians with fixed scale). As a direct consequence of Proposition 3.1 we get:

Proposition 3.4. *Let $s > 0$ be given. Then* $\mathrm{MLE}(\mu\|\mathcal{G}_{s\mathrm{I}}) = \{\mathcal{N}(\mathrm{m}_{\mu}, s\mathrm{I})\}$ *and*

$$H^{\times}(\mu\|\mathcal{G}_{s\mathrm{I}}) = \frac{1}{2s}D_{\mu} + \frac{N}{2}\ln s + \frac{N}{2}\ln(2\pi).$$

4. Many coding subdensities

In the previous section we considered the coding of a μ-randomly chosen point $x \in \mathbb{R}^N$ by the code optimized for the subdensity f. Since it is often better to "pack/compress" parts of data with various algorithms, assume that we are given a sequence of k subdensities $(f_i)_{i=1}^k$, which we interpret as coding algorithms.

Suppose that we want to code x by j-th algorithm from the sequence $(f_i)_{i=1}^k$. By (3.1) the length of code of x corresponds to $-\ln f_j(x)$. However, this code itself is clearly insufficient to decode x if we do not know which coding algorithm was used. Therefore to uniquely code x we have to add to it the code of j. Thus if l_j denotes the length of code of j (in NATS), the "final" length of the code of the point x is the sum of l_j and the length of the code of the point x:

$$\text{code-length of } x = l_j - \ln f_j(x).$$

Since the coding of the algorithms has to be acceptable, the sequence $(l_i)_{i=1}^k$ has to satisfy the Kraft's inequality and therefore if we put $p_i = e^{-l_i}$, we can consider only those $p_i \geq 0$ that $\sum_{i=1}^{k} p_i \leq 1$. Consequently without loss of generality (by possibly shortening the expected code-length), we may restrict to the case when $\sum_{i=1}^{k} p_i = 1$.

We discuss the case when points from $U_i \subset \mathbb{R}^N$ are coded by the subdensity f_i. Observe that although U_i have to be pairwise disjoint, they do not have to cover the whole space $\mathbb{R}^N$ – we can clearly omit the set with μ-measure zero. To formalize this, the notion of μ-partition[3] for a given continuous or discrete measure μ is convenient: we say that a pairwise disjoint sequence $(U_i)_{i=1}^k$ of Lebesgue measurable subsets of $\mathbb{R}^N$ is a *μ-partition* if $\mu\left(\mathbb{R}^N \setminus \bigcup_{i=1}^k U_i\right) = 0$.

To sum up: we have the "coding" subdensities $(f_i)_{i=1}^k$ and $p \in P_k$, where

$$P_k := \{(p_1, \ldots, p_k) \in [0,1]^k : \sum_{i=1}^k p_i = 1\}.$$

As U_i we take the set of points of $\mathbb{R}^N$ we code by density f_i. Then for a μ-partition $(U_i)_{i=1}^k$ we obtain the code-length function

$$x \to -\ln p_i - \ln f_i(x) \text{ for } x \in U_i,$$

which is exactly the code-length of the subdensity[4]

$$p_1 f_1|_{U_1} \cup \ldots \cup p_k f_k|_{U_k}. \tag{4.1}$$

In general we search for those p and μ-partition for which the expected code-length given by the cross-entropy $H^\times\big(\mu \| \bigcup_{i=1}^k p_i f_i|_{U_i}\big)$ will be minimal.

Definition 4.1. *Let $(\mathcal{F}_i)_{i=1}^k$ be a sequence of subdensity families in $\mathbb{R}^N$, and let a μ-partition $(U_i)_{i=1}^k$ be given. Then we define*

$$\biguplus_{i=1}^k (\mathcal{F}_i|U_i) := \Big\{ \bigcup_{i=1}^k p_i f_i|_{U_i} : (p_i)_{i=1}^k \in P_k, (f_i)_{i=1}^k \in (\mathcal{F}_i)_{i=1}^k \Big\}.$$

Observe that $\biguplus_{i=1}^k (\mathcal{F}_i|U_i)$ denotes those compression algorithms which can be built by using an arbitrary compression subdensity from $\mathcal{F}_i$ on the set U_i.

Our basic aim is to find a μ-partition $(U_i)_{i=1}^k$ for which $H^\times\big(\mu \| \biguplus_{i=1}^k (\mathcal{F}_i|U_i)\big)$ is minimal. In general it is NP-hard problem even for k-means [44], which is

[3]We introduce μ-partition as in dealing in practice with clustering of the discrete data it is natural to partition just the dataset and not the whole space.

[4]Observe that this density is defined for μ-almost all $x \in \mathbb{R}^N$.

the simplest limiting case of Spherical CEC (see Observation 5.7). However, in practice we hope to find a sufficiently good solution by applying either Lloyd's [45, 46] or Hartigan's method [1, Chapter 4], [47].

Since the most common and simple optimization heuristic for k-means cost is Lloyd's method, we first discuss its adaptation for CEC. The basis of Lloyd's approach to CEC is given in the following two results which show that

- given $p \in P_k$ and $(f_i)_{i=1}^k \in (\mathcal{F}_i)_{i=1}^k$, we can find a partition $(U_i)_{i=1}^k$ which minimizes the cross-entropy $H^\times\big(\mu\| \bigcup_{i=1}^k p_i f_i|_{U_i}\big)$;
- for a partition $(U_i)_{i=1}^k$, we can find $p \in P_k$ and $(f_i)_{i=1}^k \in (\mathcal{F}_i)_{i=1}^k$ which minimizes $H^\times\big(\mu\| \bigcup_{i=1}^k p_i f_i|_{U_i}\big)$.

We first show how to minimize the value of cross-entropy being given a μ-partition $(U_i)_{i=1}^k$. From now on we interpret $0 \cdot x$ as zero even if $x = \pm\infty$ or x is not properly defined.

Observation 4.1. *Let $(f_i) \in (\mathcal{F}_i)$, $p \in P_k$ and $(U_i)_{i=1}^k$ be a μ-partition. Then*

$$H^\times\big(\mu\| \bigcup_{i=1}^k p_i f_i|_{U_i}\big) = \sum_{i=1}^k \mu(U_i) \cdot \big(-\ln p_i + H^\times(\mu_{U_i}\|f_i)\big)\,. \tag{4.2}$$

Proof. We have

$$\begin{aligned} H^\times\big(\mu\| \bigcup_{i=1}^k p_i f_i|_{U_i}\big) &= \sum_{i=1}^k \int_{U_i} -\ln p_i - \log_d f_i(x) d\mu(x) \\ &= \sum_{i=1}^k \mu(U_i) \cdot \big(-\ln p_i - \int \ln(f_i(x)) d\mu_{U_i}(x)\big)\,. \end{aligned}$$

□

Proposition 4.1. *Let the sequence of subdensity families $(\mathcal{F}_i)_{i=1}^k$ be given and let $(U_i)_{i=1}^k$ be a fixed μ-partition. We put $p = (\mu(U_i))_{i=1}^k \in P_k$.*

Then

$$H^\times\big(\mu\| \biguplus_{i=1}^k (\mathcal{F}_i|U_i)\big) = H^\times\big(\mu\| \bigcup_{i=1}^k p_i f_i|_{U_i}\big) = \sum_{i=1}^k \mu(U_i) \cdot [-\ln(\mu(U_i) + H^\times(\mu_{U_i}\|\mathcal{F}_i)].$$

Proof. We apply the formula (4.2)

$$H^{\times}\big(\mu\| \bigcup_{i=1}^{k} \tilde{p}_i f_i|_{U_i}\big) = \sum_{i=1}^{k} \mu(U_i) \cdot \big(-\ln \tilde{p}_i + H^{\times}(\mu_{U_i}\|f_i)\big)\,.$$

By the property of classical entropy we know that the function $P_k \ni \tilde{p} = (\tilde{p}_i)_{i=1}^k \to \sum_{i=1}^{k} \mu(U_i) \cdot (-\ln \tilde{p}_i)$ is minimized for $\tilde{p} = (\mu(U_i))_i$. □

The above can be equivalently rewritten with the use of notation:

$$h_\mu(\mathcal{F}; W) := \begin{cases} \mu(W) \cdot \big(-\ln(\mu(W)) + H^{\times}(\mu_W\|\mathcal{F})\big) \text{ if } \mu(W) > 0, \\ 0 \text{ otherwise.} \end{cases}$$

Thus $h_\mu(\mathcal{F}; W)$ tells us what is the minimal cost of compression of the part of our dataset contained in W by subdensities from $\mathcal{F}$. By Proposition 4.1 if $(U_i)_{i=1}^k$ is a μ-partition then

$$H^{\times}\big(\mu\| \biguplus_{i=1}^{k} (\mathcal{F}_i|U_i)\big) = \sum_{i=1}^{k} h_\mu(\mathcal{F}_i; U_i). \tag{4.3}$$

Observe that, in general, if $\mu(U) > 0$ then $H^{\times}(\mu_U\|\mathcal{F}) = \ln(\mu(U)) + \frac{1}{\mu(U)} h_\mu(\mathcal{F}; U)$. Consequently, if we are given a μ_U-partition $(U_i)_{i=1}^k$, then

$$H^{\times}(\mu_U\| \biguplus_{i=1}^{k} (\mathcal{F}_i|U_i)) = \ln(\mu(U)) + \frac{1}{\mu(U)} \sum_{i=1}^{k} h_\mu(\mathcal{F}_i; U_i).$$

Theorem 4.1. *Let the sequence of subdensity families $(\mathcal{F}_i)_{i=1}^k$ be given and let $(U_i)_{i=1}^k$ be a fixed μ-partition.*

We put $p = (\mu(U_i))_{i=1}^k \in P_k$. We assume that $\mathrm{MLE}(\mu_{U_i}\|\mathcal{F}_i)$ *is nonempty for every $i = 1..k$. Then for arbitrary*

$$f_i \in \mathrm{MLE}(\mu_{U_i}\|\mathcal{F}_i) \text{ for } i = 1, \ldots, k, \tag{4.4}$$

we get

$$H^{\times}\big(\mu\| \biguplus_{i=1}^{k} (\mathcal{F}_i|U_i)\big) = H^{\times}\big(\mu\| \biguplus_{i=1}^{k} p_i f_i|_{U_i}\big).$$

Proof. Directly from the definition of MLE we obtain that

$$H^{\times}(\mu_{U_i}\|\tilde{f}_i) \geq H^{\times}(\mu_{U_i}\|\mathcal{F}_i) = H^{\times}(\mu_{U_i}\|f_i)$$

for $\tilde{f}_i \in \mathcal{F}_i$. □

The following theorem is a dual version of Theorem 4.1 – for fixed $p \in P_k$ and $f_i \in \mathcal{F}_i$ we seek optimal μ-partition which minimizes the cross-entropy.

By the support of measure μ we denote the support of its density if μ is continuous and the set of support points if it is discrete.

Theorem 4.2. *Let the sequence of subdensity families $(\mathcal{F}_i)_{i=1}^k$ be given and let $f_i \in \mathcal{F}_i$ and $p \in P_k$ be such that $\operatorname{supp}(\mu) \subset \bigcup_{i=1}^k \operatorname{supp}(f_i)$. We define $l : \operatorname{supp}(\mu) \to (-\infty, \infty]$ by*

$$l(x) := \min_{i\in\{1,\ldots,k\}} [-\ln p_i - \ln f_i(x)].$$

We construct a sequence $(U_i)_{i=1}^k$ of measurable subsets of $\mathbb{R}^N$ recursively by the following procedure:

- $U_1 = \{x \in \operatorname{supp}(\mu) : -\ln p_1 - \ln f_1(x) = l(x)\}$;
- $U_{l+1} = \{x \in \operatorname{supp}(\mu) \setminus (U_1 \cup \ldots \cup U_l) : -\ln p_{l+1} - \ln f_{l+1}(x) = l(x)\}$.

Then $(U_i)_{i=1}^k$ is a μ-partition and

$$H^{\times}\big(\mu\| \bigcup_{i=1}^k p_i f_i|_{U_i}\big) = \inf\{H^{\times}\big(\mu\| \bigcup_{i=1}^k p_i f_i|_{V_i}\big) : \mu\text{-partition } (V_i)_{i=1}^k\}.$$

Proof. Since $\operatorname{supp}(\mu) \subset \bigcup_{i=1}^k \operatorname{supp}(f_i)$, we obtain that $(U_i)_{i=1}^k$ is a μ-partition. Moreover, directly by the choice of $(U_i)_{i=1}^k$ we obtain that

$$l(x) = \ln(\bigcup_{i=1}^k p_i f_i|_{U_i})(x) \text{ for } x \in \operatorname{supp}(\mu),$$

and consequently for an arbitrary μ-partition $(V_i)_{i=1}^k$ we get

$$\begin{aligned} &H^{\times}(\mu\| \bigcup_{i=1}^k p_i f_i|_{V_i}) = \int \bigcup_{i=1}^k \big[-\ln(p_i) - \ln(f_i|_{V_i}(x))\big] d\mu(x) \\ &\leq \int \bar{l}(x) d\mu(x) = \int \bigcup_{i=1}^k \big[-\ln(p_i) - \ln(f_i|_{U_i}(x))\big] d\mu(x). \end{aligned}$$

□

Remark 4.1. *Observe that the function* $\bigcup_{i=1}^{k} p_i f_i|_{U_i}$ *constructed in the above theorem coincides (possibly except for set of* μ*-measure zero) with subdensity* $\max_{i=1..k} p_i f_i$*. This implies that the aim of CEC lies in minimization of the value of* $H^{\times}\big(\mu\| \max_{i=1..k} p_i f_i\big)$ *with respect to nonnegative* $p_i : \sum_i p_i = 1$ *and arbitrary* $f_i \in \mathcal{F}_i$*.*

As we have mentioned before, Lloyd's approach is based on alternate use of steps from Theorems 4.1 and 4.2. In practice we usually start by choosing initial densities and set probabilities p_k equal: $p = (1/k, \ldots, 1/k)$ (since the convergence is to local minimum we commonly start from various initial condition several times).

Observe that directly by Theorems 4.1 and 4.2 we obtain that the sequence $n \to h_n$ is decreasing. One hopes that limit h_n converges (to enhance that chance we usually start many times from various initial clustering) to the global infimum of $H^{\times}\big(\mu\| \biguplus_{i=1}^{k} (\mathcal{F}_i|U_i)\big)$.

To show a simple example of cross-entropy minimization we first need some notation. We are going to discuss the Lloyds cross-entropy minimization of discrete data with respect to $\mathcal{G}_{\Sigma_1}, \ldots, \mathcal{G}_{\Sigma_K}$. As a direct consequence of (4.3) and Proposition 3.1 we obtain the formula for the cross entropy of μ with respect to a family of Gaussians with covariances $(\Sigma_i)_{i=1}^k$.

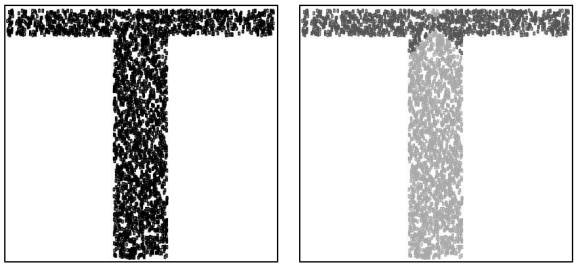
Observation 4.2. *Let* $(\Sigma_i)_{i=1}^k$ *be fixed positive symmetric matrices and let* $(U_i)_{i=1}^k$ *be a given* μ*-partition. Then*

$$H^{\times}(\mu\| \biguplus_{i=1}^{k} (\mathcal{G}_{\Sigma_i}|U_i)) =$$
$$\tfrac{N}{2}\ln(2\pi) + \sum_{i=1}^{k} \mu(U_i)\big[-\ln(\mu(U_i)) + \tfrac{1}{2}\mathrm{tr}(\Sigma_i^{-1}\Sigma_{U_i}^{\mu}) + \tfrac{1}{2}\ln\det(\Sigma_i)\big].$$

Example 4.1. *We show Lloyd's approach to cross-entropy minimization of the set* Y *showed on Figure 12(a). As is usual, we first associate with the data-set* Y *the probability measure defined by the formula*

$$\mu := \frac{1}{\mathrm{card}\mathrm{Y}} \sum_{y\in \mathrm{Y}} \delta_y,$$

where δ_y *denotes the Dirac delta at the point* y*.*



(a) T-like set. (b) Partition into two groups.

Figure 12: Effect of clustering by Lloyd's algorithm on T-like shape.

Next we search for the μ-partition $\mathrm{Y} = \mathrm{Y}_1 \cup \mathrm{Y}_2$ which minimizes $H^{\times}(\mu \| (\mathcal{G}_{\Sigma_1}|\mathrm{Y}_1) \uplus (\mathcal{G}_{\Sigma_2}|\mathrm{Y}_2))$, where $\Sigma_1 = [300, 0; 0, 1]$, $\Sigma_2 = [1, 0; 0, 300]$. The result is given on Figure 12(b), where the dark gray points which belong to Y_1 are "coded" by density from $\mathcal{G}_{\Sigma_1}$ and light gray belonging to Y_2 and are "coded" by density from $\mathcal{G}_{\Sigma_2}$.

Due to its nature to use Hartigan method [1, Chapter 4], [47] we have to divide the data-set (or more precisely the support of the measure μ) into "basic parts/blocks" from which we construct our clustering/grouping (typically the role of blocks is played by single data-points). Suppose that we have a fixed μ-partition[5] $\mathcal{V} = (V_i)_{i=1}^n$. The aim of Hartigan method is to find a μ-partition build from elements of $\mathcal{V}$ which has minimal cross-entropy by subsequently reassigning membership of following elements of partition $\mathcal{V}$.

Consider k coding subdensity families $(\mathcal{F}_i)_{i=1}^k$. To explain Hartigan approach more precisely we need the notion of *group membership function* $\mathrm{gr} : \{1, \ldots, n\} \to \{0, \ldots, k\}$ which describes the membership of i-th element of partition, where 0 value is a special symbol which denotes that V_i is as yet unassigned. In other words: if $\mathrm{gr}(i) = l > 0$, then V_i is a part of the l-th group, and if $\mathrm{gr}(i) = 0$ then V_i is unassigned.

We want to find such $\mathrm{gr} : \{1, \ldots, n\} \to \{1, \ldots, k\}$ (thus all elements of $\mathcal{V}$ are assigned) that $\sum_{i=1}^{k} h_\mu(\mathcal{F}_i; \mathcal{V}(\mathrm{gr}^{-1}(i))$ is minimal. Basic idea of Hartigan is relatively simple – we repeatedly go over all elements of the partition $\mathcal{V} = (V_i)_{i=1}^n$ and apply the following steps:

[5] By default we think of it as a partition into sets with small diameter.

- if the chosen set V_i is unassigned, assign it to the first nonempty group;
- reassign V_i to those group for which the decrease in cross-entropy is maximal;
- check if no group needs to be removed/unassigned, if this is the case unassign its all elements;

until no group membership has been changed.

To practically apply Hartigans algorithm we still have to determine the initial group membership. In most examples in this paper we initialized the cluster membership function randomly. However, one can naturally speed the clustering by using some more intelligent cluster initializations like [48].

To implement Hartigan approach for discrete measures we still have to add a condition when we unassign given group. For example in the case of Gaussian clustering in $\mathbb{R}^N$ to avoid overfitting we cannot consider clusters which contain less then $N+1$ points. In practice while applying Hartigan approach on discrete data we usually removed clusters which contained less then three percent of all data-set.

Observe that in the crucial step in Hartigan approach we compare the cross-entropy after and before the switch, while the switch removes a given set from one cluster and adds it to the other. Since

$$h_\mu(\mathcal{F}; W) = \mu(W) \cdot \big(- \ln(\mu(W)) + H^\times(\mu_W \| \mathcal{F}) \big),$$

basic steps in the Hartigan approach reduce to computation of $H^\times(\mu_W \| \mathcal{F}))$ for $W = U \cup V$ and $W = U \setminus V$. This implies that to apply efficiently the Hartigan approach in clustering it is of basic importance to compute

- $H^\times(\mu_{U \cup V} \| \mathcal{F})$ for disjoint U, V;
- $H^\times(\mu_{U \setminus V} \| \mathcal{F})$ for $V \subset U$.

Since in the case of Gaussians to compute the cross-entropy of μ_W we need only covariance Σ^μ_W, our problem reduces to computation of $\Sigma_{U \cup V}$ and $\Sigma_{U \setminus V}$. Here the following well-known result can be useful:

Theorem 4.3. *Let U, V be Lebesgue measurable sets with finite and nonzero μ-measures.*

a) Assume additionally that $U \cap V = \emptyset$. Then

$$\mathrm{m}^{\mu}_{U\cup V} = p_U \mathrm{m}^{\mu}_U + p_V \mathrm{m}^{\mu}_V,$$
$$\Sigma^{\mu}_{U\cup V} = p_U \Sigma^{\mu}_U + p_V \Sigma^{\mu}_V + p_U p_V (\mathrm{m}^{\mu}_U - \mathrm{m}^{\mu}_V)(\mathrm{m}^{\mu}_U - \mathrm{m}^{\mu}_V)^T,$$

where $p_U = \frac{\mu(U)}{\mu(U)+\mu(V)}$, $p_V := \frac{\mu(V)}{\mu(U)+\mu(V)}$.

b) Assume that $V \subset U$ is such that $\mu(V) < \mu(U)$. Then

$$\mathrm{m}^{\mu}_{U\setminus V} = q_U \mathrm{m}^{\mu}_U - q_V \mathrm{m}^{\mu}_V,$$
$$\Sigma^{\mu}_{U\setminus V} = q_U \Sigma^{\mu}_U - q_V \Sigma^{\mu}_V - q_U q_V (\mathrm{m}^{\mu}_U - \mathrm{m}^{\mu}_V)(\mathrm{m}^{\mu}_U - \mathrm{m}^{\mu}_V)^T,$$

where $q_U := \frac{\mu(U)}{\mu(U)-\mu(V)}$, $q_V := \frac{\mu(V)}{\mu(U)-\mu(V)}$.

We want to find such $\mathrm{gr} : \{1, \ldots, n\} \to \{1, \ldots, k\}$ (thus all elements of $\mathcal{V}$ are assigned) that

$$\sum_{i=1}^{k} h_\mu(\mathcal{F}_i; \mathcal{V}(\mathrm{gr}^{-1}(i))$$

is minimal. Basic idea of Hartigan is relatively simple – we repeatedly go over all elements of the partition $\mathcal{V} = (V_i)_{i=1}^n$ and apply the following steps:

- if the chosen set V_i is unassigned, assign it to the first nonempty group;
- reassign V_i to those group for which the decrease in cross-entropy is maximal;
- check if no group needs to be removed/unassigned, if this is the case unassign its all elements;

until no group membership has been changed.

5. Clustering with respect to Gaussian families

In the proceeding part of our paper we study the applications of our theory for clustering, where by *clustering we understand division of the data into groups of similar type.* Therefore since in clustering we consider only one fixed subdensity family $\mathcal{F}$ we will use the notation

$$h_\mu(\mathcal{F}; (U_i)_{i=1}^k) := \sum_{i=1}^{k} h_\mu(\mathcal{F}; U_i), \tag{5.1}$$

Algorithm 1 (**HARTIGAN-BASED CEC**):

input
dataset X
number of clusters $k > 0$
initial clustering $X_1, \ldots, X_k$
Gaussian family $\mathcal{F}$
cluster reduction parameter $\varepsilon > 0$
define
cluster membership function
cl: $X \ni x \to l \in \{1, \ldots, k\}$ such that $x \in X_l$
cluster cost function $E(X_i)$ where
$E(Y) = p(-\ln(p) + H^{\times}(Y \| \mathcal{F}))$ and $p = \frac{\mathrm{card} Y}{\mathrm{card} X}$
repeat
for $x \in X$ **do**
for $l = 1, \ldots, k$: $x \notin X_l$ **do**
if $E(X_i \cup \{x\}) + E(X_{\mathrm{cl}(x)} \setminus \{x\}) < E(X_i) + E(X_{\mathrm{cl}(x)})$ **then**
switch x to X_i
update cl
if $\mathrm{card} X_i < \varepsilon \cdot \mathrm{card} X$ **then**
delete cluster X_i
update cl by attaching elements of X_i to existing clusters
end if
end if
end for
end for
until no switch for all subsequent elements of X

for the family $(U_i)_{i=1}^k$ of pairwise disjoint Lebesgue measurable sets. We see that (5.1) gives the total memory cost of disjoint $\mathcal{F}$-clustering of $(U_i)_{i=1}^k$.

The aim of $\mathcal{F}$-clustering is to find a μ-partition $(U_i)_{i=1}^k$ (with possibly empty elements) which minimizes

$$H^{\times}\big(\mu\| \biguplus_{i=1}^{k}(\mathcal{F}|U_i)\big) = h_\mu(\mathcal{F};(U_i)_{i=1}^k) = \sum_{i=1}^{k}\mu(U_i)\cdot[-\ln(\mu(U_i)) + H^{\times}(\mu_{U_i}\|\mathcal{F})].$$

Observe that the amount of sets (U_i) with nonzero μ-measure gives us the number of clusters into which we have divided our space.

In many cases, we want the clustering to be invariant to translations, change of scale, isometry, etc.

Definition 5.1. *Suppose that we are given a probability measure μ. We say that the clustering is* A-invariant *if instead of clustering μ we will obtain the same effect by*

- *introducing $\mu_A := \mu\circ A^{-1}$ (observe that if μ corresponds to the data* Y *then μ_A corresponds to the set A(Y));*
- *obtaining the clustering $(V_i)_{i=1}^k$ of μ_A;*
- *taking as the clustering of μ the sets $U_i = A^{-1}(V_i)$.*

This problem is addressed in following observation which is a direct consequence of Corollary 3.4:

Observation 5.1. *Let $\mathcal{F}$ be a given subdensity family and A be an affine invertible map. Then*

$$H^{\times}\big(\mu\| \biguplus_{i=1}^{k}(\mathcal{F}|U_i)\big) = H^{\times}\big(\mu\circ A^{-1}\| \biguplus_{i=1}^{k}(\mathcal{F}_A|A(U_i))\big) + \ln|\det A|.$$

As a consequence we obtain that if $\mathcal{F}$ is A-invariant, that is $\mathcal{F} = \mathcal{F}_A$, then the $\mathcal{F}$ clustering is also A-invariant.

The next important problem in clustering theory is the question how to verify cluster validity. Cross entropy theory gives a simple and reasonable answer – namely from the information point of view the clustering

$$U = U_1\cup\ldots\cup U_k$$

is profitable if we gain on separate compression by division into $(U_i)_{i=1}^k$, that is when $h_\mu(\mathcal{F};(U_i)_{i=1}^k) < h_\mu(\mathcal{F};U)$. This leads us to the definition of *relative $\mathcal{F}$-entropy of the splitting $U = U_1 \cup \ldots \cup U_k$*:

$$d_\mu(\mathcal{F};(U_i)_{i=1}^k) := h_\mu(\mathcal{F};U) - h_\mu(\mathcal{F};(U_i)_{i=1}^k).$$

Trivially if $d_\mu(\mathcal{F};(U_i)_{i=1}^k) > 0$ then we gain in using clusters $(U_i)_{i=1}^k$. Moreover, if $(U_i)_{i=1}^k$ is a μ-partition then

$$d_\mu(\mathcal{F};(U_i)_{i=1}^k) = H^\times\big(\mu\|\mathcal{F}\big) - H^\times\big(\mu\|\biguplus_{i=1}^k(\mathcal{F}|U_i)\big).$$

5.1. Gaussian Clustering

From now on we fix our attention on Gaussian clustering (we use this name instead $\mathcal{G}$-clustering). By Observation 5.1 we obtain that the Gaussian clustering is invariant with respect to affine transformations. By joining Proposition 3.2 with (5.1) we obtain the basic formula on the Gaussian cross-entropy.

Observation 5.2. *Let $(U_i)_{i=1}^k$ be a sequence of pairwise disjoint measurable sets. Then*

$$h_\mu(\mathcal{G};(U_i)_{i=1}^k) = \sum_{i=1}^k \mu(U_i)\cdot[\tfrac{N}{2}\ln(2\pi e) - \ln(\mu(U_i)) + \tfrac{1}{2}\ln\det(\Sigma^\mu_{U_i})]. \tag{5.2}$$

In the case of Gaussian clustering due to the large degree of freedom we are not able to obtain in the general case a simple formula for the relative entropy of two clusters. However, we can easily consider the case of two groups with equal covariances.

Theorem 5.1. *Let us consider disjoint sets $U_1, U_2 \subset \mathbb{R}^N$ with identical covariance matrices $\Sigma^\mu_{U_1} = \Sigma^\mu_{U_2} = \Sigma$. Then*

$$d_\mu(\mathcal{G};(U_1,U_2))/(\mu(U_1)+\mu(U_2)) = \tfrac{1}{2}\ln(1+p_1p_2\|\mathrm{m}^\mu_{U_1} - \mathrm{m}^\mu_{U_2}\|^2_\Sigma) - \mathrm{sh}(p_1) - \mathrm{sh}(p_2),$$

where $p_i = \mu(U_i)/(\mu(U_1)+\mu(U_2))$.

Consequently $d_\mu(\mathcal{G};(U_1,U_2)) > 0$ iff

$$\|\mathrm{m}^\mu_{U_1} - \mathrm{m}^\mu_{U_2}\|^2_\Sigma > p_1^{-2p_1-1}p_2^{-2p_2-1} - p_1^{-1}p_2^{-1}. \tag{5.3}$$

Proof. By (5.2)

$$d_\mu(\mathcal{G};(U_1,U_2))/(\mu(U_1)+\mu(U_2)) = \tfrac{1}{2}[\ln\det(\Sigma^\mu_{U_1\cup U_2}) - \ln\det(\Sigma)] - \mathrm{sh}(p_1) - \mathrm{sh}(p_2).$$

By applying Theorem 4.3 the value of $\Sigma^\mu_{U_1\cup U_2}$ simplifies to $\Sigma + p_1p_2\mathrm{m}\mathrm{m}^T$, where $\mathrm{m} = (\mathrm{m}^\mu_{U_1} - \mathrm{m}^\mu_{U_2})$, and therefore we get

$$\begin{aligned}
&d_\mu(\mathcal{G};(U_1,U_2))/(\mu(U_1)+\mu(U_2))\\
&= \tfrac{1}{2}\ln\det(\mathrm{I} + p_1p_2\Sigma^{-1/2}\mathrm{m}\mathrm{m}^T\Sigma^{-1/2}) - \mathrm{sh}(p_1) - \mathrm{sh}(p_2)\\
&= \tfrac{1}{2}\ln\det(\mathrm{I} + p_1p_2(\Sigma^{-1/2}\mathrm{m})(\Sigma^{-1/2}\mathrm{m})^T) - \mathrm{sh}(p_1) - \mathrm{sh}(p_2).
\end{aligned}$$

Since $\det(I+\alpha vv^T) = 1+\alpha\|v\|^2$ (to see this it suffices to consider the matrix of the operator $I + \alpha vv^T$ in the orthonormal base which first element is $v/\|v\|$), we arrive at

$$d_\mu(\mathcal{G};(U_1,U_2))/(\mu(U_1)+\mu(U_2)) = \tfrac{1}{2}\ln(1 + p_1p_2\|\mathrm{m}\|^2_\Sigma) - \mathrm{sh}(p_1) - \mathrm{sh}(p_2).$$

Consequently $d_\mu(\mathcal{G};(U_1,U_2)) > 0$ iff

$$\ln(1 + p_1p_2\|\mathrm{m}\|^2_\Sigma) > 2\mathrm{sh}(p_1) + 2\mathrm{sh}(p_2),$$

which is equivalent to $1 + p_1p_2\|\mathrm{m}\|^2_\Sigma > p_1^{-2p_1}p_2^{-2p_2}$. □

Remark 5.1. *As a consequence of* (5.3) *we obtain that if the means of U_1 and U_2 are sufficiently close in the Mahalanobis $\|\cdot\|_\Sigma$ distance, then it is profitable to glue those sets together into one cluster.*

Observe also that the constant in RHS of (5.3) *is independent of the dimension. We mention it as an analogue does not hold for Spherical clustering, see Observation 5.4.*

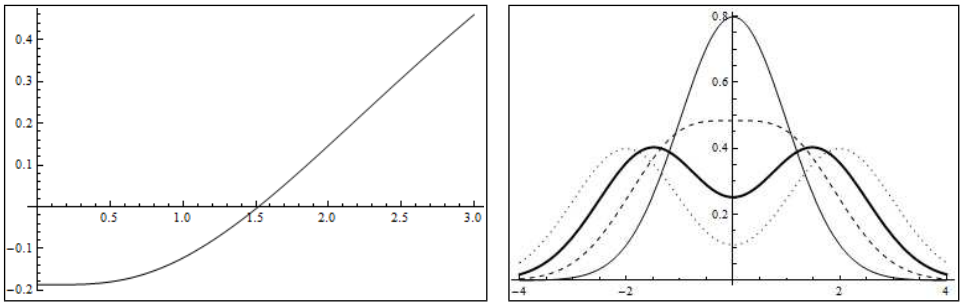
Example 5.1. *Consider the probability measure μ_s on $\mathbb{R}$ given as the convex combination of two gaussians with means at s and $-s$, with density*

$$f_s := \tfrac{1}{2}\mathcal{N}(s,1) + \tfrac{1}{2}\mathcal{N}(-s,1),$$

where $s \geq 0$. Observe that with $s \to \infty$ the initial density $\mathcal{N}(0,1)$ separates into two almost independent gaussians.

To check for which s the Gaussian divergence will see this behavior, we fix the partition $(-\infty,0),(0,\infty)$. One can easily verify that

$$\begin{aligned}
&d_{\mu_s}\big(\mathcal{G};((-\infty,0),(0,\infty))\big) =\\
&-\ln(2) + \tfrac{1}{2}\ln(1+s^2) - \tfrac{1}{2}\ln[1 - \tfrac{2e^{-s^2}}{\pi} + s^2 - \sqrt{\tfrac{8}{\pi}}se^{-\frac{s^2}{2}}\mathrm{Erf}(\tfrac{s}{\sqrt{2}}) - s^2\mathrm{Erf}(\tfrac{s}{\sqrt{2}})^2].
\end{aligned}$$



(a) Plot of Gaussian divergence. (b) Densities.

Figure 13: Convex combination of gaussian densities. Black thick density is the bordering density between one and two clusters

Consequently, see Figure 13(a), there exists $s_0 \approx 1.518$ *such that the clustering of* $\mathbb{R}$ *into two clusters* $((-\infty, 0), (0, \infty))$ *is profitable iff* $s > s_0$. *On figure 13(b) we show densities* f_s *for* $s = 0$ *(thin line);* $s = 1$ *(dashed line);* $s = s_0$ *(thick line) and* $s = 2$ *(points).*

This theoretical result which puts the border between one and two clusters at s_0 *seems consistent with our geometrical intuition of clustering of* μ_s.

5.2. Spherical Clustering

In this section we consider spherical clustering which can be seen as a simpler version of the Gaussian clustering. By Observation 5.1 we obtain that Spherical clustering is invariant with respect to scaling and isometric transformations (however, it is obviously not invariant with respect to affine transformations).

Observation 5.3. *Let* $(U_i)_{i=1}^k$ *be a* μ*-partition. Then*

$$h_\mu(\mathcal{G}_{(\cdot \mathrm{I})}; (U_i)_{i=1}^k) = \sum_{i=1}^k \mu(U_i) \cdot [\tfrac{N}{2} \ln(2\pi e/N) - \ln(\mu(U_i)) + \tfrac{N}{2} \ln D_{U_i}^\mu]. \quad (5.4)$$

To implement Hartigan approach to Spherical CEC and to deal with Spherical relative entropy the following trivial consequence of Theorem 4.3 is useful.

Corollary 5.1. *Let* U, V *be measurable sets.*

a) If $U \cap V = \emptyset$ *and* $\mu(U) > 0, \mu(V) > 0$. *Then*

$$D_{U \cup V}^\mu = p_U D_U^\mu + p_V D_V^\mu + p_U p_V \|\mathrm{m}_U^\mu - \mathrm{m}_V^\mu\|^2,$$

where $p_U := \frac{\mu(U)}{\mu(U)+\mu(V)}$, $p_V := \frac{\mu(V)}{\mu(U)+\mu(V)}$.

b) If $V \subset U$ *is such that* $\mu(V) < \mu(U)$ *then*

$$D_{U\setminus V} \;=\; q_U D_U^\mu - q_V D_V^\mu - q_U q_V \|\mathrm{m}_U^\mu - \mathrm{m}_V^\mu\|^2,$$

where $q_U := \frac{\mu(U)}{\mu(U)-\mu(V)}$, $q_V := \frac{\mu(V)}{\mu(U)-\mu(V)}$.

Example 5.2. *We considered in Figure 1(a) the uniform distribution on the set consisting of three circles. We started CEC with initial choice of 10 clusters, as a result of Spherical CEC we obtained clustering into three "almost circles" see Figure 1(d) – compare this result with the classical k-means with* $k = 3$ *and* $k = 10$ *on Figures 1(b) and 1(c).*

Let us now consider when we should join two groups.

Theorem 5.2. *Let* U_1 *and* U_2 *be disjoint measurable sets with nonzero* μ*-measure. We put* $p_i = \mu(U_i)/(\mu(U_1) + \mu(U_2))$ *and* $\mathrm{m}_i = \mathrm{m}_{U_i}^\mu$, $D_i = D_{U_i}^\mu$ *for* $i = 1, 2$. *Then*

$$\begin{aligned} &d_\mu(\mathcal{G}; (U_1, U_2))/(\mu(U_1) + \mu(U_2)) \\ &= \tfrac{N}{2} \ln(p_1 D_1 + p_2 D_2 + p_1 p_2 \|\mathrm{m}_1 - \mathrm{m}_2\|^2) - p_1 \tfrac{N}{2} \ln D_1 - p_2 \tfrac{N}{2} \ln D_2 - \mathrm{sh}(p_1) - \mathrm{sh}(p_2). \end{aligned}$$

Consequently, $d_\mu(\mathcal{G}_{(\cdot \mathrm{I})}; (U_1, U_2)) > 0$ *iff*

$$\|\mathrm{m}_1 - \mathrm{m}_2\|^2 > \frac{D_1^{p_1} D_2^{p_2}}{p_1^{2p_1/N} p_2^{2p_2/N}} - (p_1 D_1 + p_2 D_2).$$

Proof. By (5.4)

$$\begin{aligned} &d_\mu(\mathcal{G}_{(\cdot \mathrm{I})}; (U_1, U_2))/(\mu(U_1) + \mu(U_2)) \\ &= \tfrac{N}{2} \ln(D_{U_1 \cup U_2}^\mu) - p_1 \tfrac{N}{2} \ln D_1 - p_2 \tfrac{N}{2} \ln D_2 - \mathrm{sh}(p_1) - \mathrm{sh}(p_2). \end{aligned}$$

Since by Corollary 5.1

$$D_{U_1+U_2}^\mu = p_1 D_1 + p_2 D_2 + p_1 p_2 \|\mathrm{m}_1 - \mathrm{m}_2\|^2,$$

we obtain that $d_\mu(\mathcal{G}_{(\cdot \mathrm{I})}; (U_1, U_2)) > 0$ iff $\|\mathrm{m}_1 - \mathrm{m}_2\|^2 > \frac{D_1^{p_1} D_2^{p_2}}{p_1^{2p_1/N} p_2^{2p_2/N}} - (p_1 D_1 + p_2 D_2)$. □

Observation 5.4. *Let us simplify the above formula in the case when we have sets with identical measures $\mu(U_1) = \mu(U_2)$ and $D := D^{\mu}_{U_1} = D^{\mu}_{U_2}$. Then by the previous theorem we should glue the groups together if*

$$\|\mathrm{m}_1 - \mathrm{m}_2\| \leq \sqrt{4^{1/N} - 1} \cdot r,$$

where $r = \sqrt{D}$. So, as we expected, when the distance between the groups is proportional to their "radius" the joining becomes profitable.

Another, maybe less obvious, consequence of

$$4^{1/N} - 1 \approx \frac{\ln 4}{N} \to 0 \text{ as } N \to \infty$$

is that with the dimension N growing we should join the groups/sets together if their centers become closer. This follows from the observation that if we choose two balls in $\mathbb{R}^N$ with radius r and distance between centers $R \geq 2r$, the proportion of their volumes to the volume of the containing ball decreases to zero with dimension growing to infinity.

5.3. Fixed covariance

In this section we are going to discuss the simple case when we cluster by $\mathcal{G}_\Sigma$, for a fixed Σ. By Observation 5.1 we obtain that $\mathcal{G}_\Sigma$ clustering is translation invariant (however, it is obviously not invariant with respect to scaling or isometric transformations).

Observation 5.5. *Let Σ be fixed positive symmetric matrix. and let $(U_i)_{i=1}^k$ be a sequence of pairwise disjoint measurable sets. Then*

$$\begin{aligned} &h_\mu(\mathcal{G}_\Sigma; (U_i)_{i=1}^k) \\ &= \sum_{i=1}^k \mu(U_i) \cdot (\tfrac{N}{2} \ln(2\pi) + \tfrac{1}{2} \ln \det(\Sigma)) + \sum_{i=1}^k \mu(U_i) \cdot [-\ln(\mu(U_i)) + \tfrac{1}{2} \mathrm{tr}(\Sigma^{-1} \Sigma^{\mu}_{U_i})]. \end{aligned}$$

This implies that in the $\mathcal{G}_\Sigma$ clustering we search for the partition $(U_i)_{i=1}^k$ which minimizes

$$\sum_{i=1}^k \mu(U_i) \cdot \big[-\ln(\mu(U_i)) + \frac{1}{2} \mathrm{tr}(\Sigma^{-1} \Sigma^{\mu}_{U_i}) \big].$$

Now we show that in the $\mathcal{G}_\Sigma$ clustering, if we have two groups with centers/means sufficiently close, it always pays to "glue" the groups together into one.

Theorem 5.3. *Let U_1 and U_2 be disjoint measurable sets with nonzero μ-measure. We put $p_i = \mu(U_i)/(\mu(U_1) + \mu(U_2))$. Then*

$$d_\mu(\mathcal{G}_\Sigma; (U_1, U_2))/(\mu(U_1) + \mu(U_2)) = p_1 p_2 \|\mathrm{m}^\mu_{U_1} - \mathrm{m}^\mu_{U_2}\|^2_\Sigma - \mathrm{sh}(p_1) - \mathrm{sh}(p_2). \tag{5.5}$$

Consequently $d_\mu(\mathcal{G}_\Sigma; (U_1, U_2)) > 0$ iff

$$\|\mathrm{m}^\mu_{U_1} - \mathrm{m}^\mu_{U_2}\|^2_\Sigma > \frac{\mathrm{sh}(p_1) + \mathrm{sh}(p_2)}{p_1 p_2}.$$

Proof. We have

$$\begin{aligned} &d_\mu(\mathcal{G}_\Sigma; (U_1, U_2))/(\mu(U_1) + \mu(U_2)) \\ &= \tfrac{1}{2}\mathrm{tr}(\Sigma^{-1}\Sigma^\mu_{U_1\cup U_2}) - \tfrac{p_1}{2}\mathrm{tr}(\Sigma^{-1}\Sigma^\mu_{U_1}) - \tfrac{p_2}{2}\mathrm{tr}(\Sigma^{-1}\Sigma^\mu_{U_2}) - \mathrm{sh}(p_1) - \mathrm{sh}(p_2). \end{aligned} \tag{5.6}$$

Let $\mathrm{m} = \mathrm{m}^\mu_{U_1} - \mathrm{m}^\mu_{U_2}$. Since $\Sigma^\mu_{U_1\cup U_2} = p_1\Sigma^\mu_{U_1} + p_2\Sigma^\mu_{U_2} + p_1p_2\mathrm{m}\mathrm{m}^T$, and $\mathrm{tr}(AB) = \mathrm{tr}(BA)$, (5.6) simplifies to (5.5). □

Observe that the above formula is independent of deviations in groups, but only on the distance of the centers of weights (means in each groups).

Lemma 5.1. *The function*

$$\{(p_1, p_2) \in (0,1)^2 : p_1 + p_2 = 1\} \to \frac{\mathrm{sh}(p_1) + \mathrm{sh}(p_2)}{p_1 p_2}$$

attains global minimum $\ln 16$ *at* $p_1 = p_2 = 1/2$.

Proof. Consider

$$w : (0,1) \ni p \to \frac{\mathrm{sh}(p) + \mathrm{sh}(1-p)}{p(1-p)}.$$

Since w is symmetric with respect to $1/2$, to show assertion it is sufficient to prove that w is convex.

We have

$$w''(p) = \frac{2(-1+p)^3\ln(1-p) + p\,(-1+p-2p^2\ln(p))}{(1-p)^3p^3}.$$

Since the denominator of w'' is nonnegative, we consider only the numerator, which we denote by $g(p)$. The fourth derivative of g equals $12/[p(1-p)]$. This implies that

$$g''(p) = 4(-2 + 3(-1+p)\ln(1-p) - 3p\ln(p))$$

is convex, and since it is symmetric around $1/2$, it has the global minimum at $1/2$ which equals

$$g''(1/2) = 4(-2 + 3\ln 2) = 4\ln(8/e^2) > 0.$$

Consequently $g''(p) > 0$ for $p \in (0,1)$, which implies that g is convex. Being symmetric around $1/2$ it attains minimum at $1/2$ which equals $g(1/2) = \frac{1}{4}\ln(4/e) > 0$, which implies that g is nonnegative, and consequently w'' is also nonegative. Therefore w is convex and symmetric around $1/2$, and therefore attains its global minimum $4\ln 2$ at $p = 1/2$. □

Corollary 5.2. *If we have two clusters with centers* m_1 *and* m_2*, then it is always profitable to glue them together into one group in* $\mathcal{G}_\Sigma$*-clustering if*

$$\|m_1 - m_2\|_\Sigma < \sqrt{\ln 16} \approx 1.665.$$

As a direct consequence we get:

Corollary 5.3. *Let* μ *be a measure with support contained in a bounded convex set* V*. Then the number of clusters which realize the cross-entropy* $\mathcal{G}_\Sigma$ *is bounded from above by the maximal cardinality of an* ε*-net (with respect to the Mahalanobis distance* $\|\cdot\|_\Sigma$*), where* $\varepsilon = \sqrt{4\ln 2}$*, in* V*.*

Proof. By k we denote the maximal cardinality of the ε-net with respect to the Mahalanobis distance.

Consider an arbitrary μ-partition $(U_i)_{i=1}^l$ consisting of sets with nonempty μ-measure. Suppose that $l > k$. We are going to construct a μ-partition with $l-1$ elements which has smaller cross-entropy then (U_i).

To do so consider the set $(\mathrm{m}_{U_i}^\mu)_{i=1}^l$ consisting of centers of the sets U_i. By the assumptions we know that there exist at least two centers which are closer then ε – for simplicity assume that $\|\mathrm{m}_{U_{l-1}}^\mu - \mathrm{m}_{U_l}^\mu\|_\Sigma < \varepsilon$. Then by the previous results we obtain that

$$h_\mu(\mathcal{G}_\Sigma; U_{l-1} \cup U_l) < h_\mu(\mathcal{G}_\Sigma; U_{l-1}) + h_\mu(\mathcal{G}_\Sigma; U_l).$$

This implies that the μ-partition $(U_1, \ldots, U_{l-2}, U_{l-1} \cup U_l)$ has smaller cross-entropy then $(U_i)_{i=1}^l$. □

5.4. Spherical CEC with scale and k-means

We recall that $\mathcal{G}_{s\mathrm{I}}$ denotes the set of all normal densities with covariance $s\mathrm{I}$. We are going to show that for $s \to 0$ results of $\mathcal{G}_{s\mathrm{I}}$-CEC converge to k-means clustering, while for $s \to \infty$ our data will form one big group.

Observation 5.6. *For the sequence $(U_i)_{i=1}^k$ we get*

$$h_\mu(\mathcal{G}_{s\mathrm{I}}; (U_i)_{i=1}^k) = \sum_{i=1}^k \mu(U_i) \cdot (\tfrac{N}{2} \ln(2\pi s) - \ln \mu(U_i) + \tfrac{N}{2s} D^\mu_{U_i}).$$

Clearly by Observation 5.1 $\mathcal{G}_{s\mathrm{I}}$ clustering is isometry invariant, however it is not scale invariant.

To compare k-means with Spherical CEC with fixed scale let us first describe classical k-means from our point of view. Let μ denote the discrete or continuous probability measure. For a μ-partition $(U_i)_{i=1}^k$ we introduce the *within clusters sum of squares* by the formula

$$\begin{aligned}&\mathbf{ss}(\mu \| (U_i)_{i=1}^k) := \sum_{i=1}^k \textstyle\int_{U_i} \|x - \mathrm{m}^\mu_{U_i}\|^2 d\mu(x)\\ &= \sum_{i=1}^k \mu(U_i) \textstyle\int \|x - \mathrm{m}^\mu_{U_i}\|^2 d\mu_{U_i}(x) = \sum_{i=1}^k \mu(U_i) \cdot D^\mu_{U_i}.\end{aligned}$$

Remark 5.2. *Observe that if we have data* Y *partitioned into* $\mathrm{Y} = \mathrm{Y}_1 \cup \ldots \cup \mathrm{Y}_k$*, then the above coincides (modulo multiplication by the cardinality of* Y*) with the classical within clusters sum of squares. Namely, for discrete probability measure $\mu_Y := \frac{1}{\mathrm{card}(\mathrm{Y})} \sum_{y \in \mathrm{Y}} \delta_y$ we have $\mathbf{ss}(\mu_\mathrm{Y} \| (\mathrm{Y}_i)_{i=1}^k) =$*

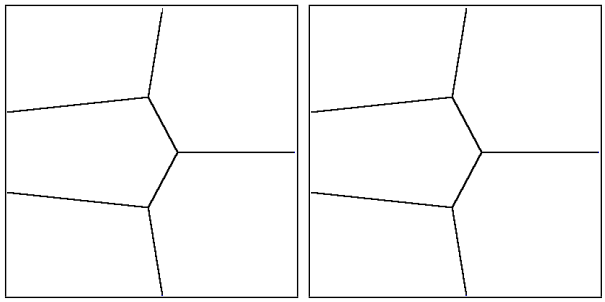
$\frac{1}{\mathrm{card}(\mathrm{Y})} \sum_{i=1}^k \sum_{y \in \mathrm{Y}_i} \|y - \mathrm{m}_{\mathrm{Y}_i}\|^2$.

In classical k-means the aim is to find such μ-partition $(U_i)_{i=1}^k$ which minimizes the within clusters sum of squares

$$\sum_{i=1}^k \mu(U_i) \cdot D^\mu_{U_i}, \tag{5.7}$$

while in $\mathcal{G}_{s\mathrm{I}}$-clustering our aim is to minimize

$$\sum_{i=1}^k \mu(U_i) \cdot (-\frac{2s}{N} \ln \mu(U_i) + D^\mu_{U_i}).$$



(a) k-means with $k=5$. (b) $\mathcal{G}_{s\mathrm{I}}$-CEC for $s = 5 \cdot 10^{-5}$ and 5 clusters.

Obviously with $s \to 0$, the above function converges to (5.7), which implies that k-means clustering can be understood as the limiting case of $\mathcal{G}_{s\mathrm{I}}$ clustering, with $s \to 0$.

Example 5.3. *We compare on Figure 5.3 $\mathcal{G}_{s\mathrm{I}}$ clustering of the square $[0,1]^2$ with very small $s = 5{\cdot}10^{-5}$ to k-means. As we see we obtain optically identical results.*

Observation 5.7. *We have*

$$\begin{aligned} &0 \leq \mathbf{ss}(\mu\|(U_i)_{i=1}^k) - [-s\ln(2\pi s) + \tfrac{2s}{N} H^{\times}\big(\mu\| \biguplus_{i=1}^{k} (\mathcal{G}_{s\mathrm{I}}|U_i)\big)] \\ &= \tfrac{s}{2N} \sum_{i=1}^{k} \mu(U_i) \cdot \ln(\mu(U_i)) \leq \tfrac{\ln(k)}{2N} s. \end{aligned}$$

This means that for an arbitrary partition consisting of k-sets $\mathbf{ss}(\mu\|\cdot)$ can be approximated (as $s \to 0$) with the affine combination of $H^{\times}\big(\mu\|\mathcal{G}_{s\mathrm{I}}\big)$, which can be symbolically summarized as interpretation of k-means as $\mathcal{G}_{0\cdot\mathrm{I}}$ clustering.

If we cluster with $s \to \infty$ we have tendency to build larger and larger clusters.

Proposition 5.1. *Let μ be a measure with support of diameter d. Then for*

$$s > \frac{d^2}{\ln 16}$$

the optimal clustering with respect to $\mathcal{G}_{s\mathrm{I}}$ will be obtained for one large group.

More precisely, for every $k > 1$ and μ-partition $(U_i)_{i=1}^k$ consisting of sets of nonempty μ-measure we have

$$H^{\times}\big(\mu\|\mathcal{F}\big) < H^{\times}\big(\mu\| \biguplus_{i=1}^{k} (\mathcal{F}|U_i)\big).$$

Proof. By applying Corollary 5.2 with $\Sigma = s\mathrm{I}$ we obtain that we should always glue two groups with centers $\mathrm{m}_1, \mathrm{m}_2$ together if $\|\mathrm{m}_1 - \mathrm{m}_2\|^2_{s\mathrm{I}} < \ln 16$, or equivalently if $\|\mathrm{m}_1 - \mathrm{m}_2\|^2 < s \ln 16$. □

Concluding, if the radius tends to zero, we cluster the data into smaller and smaller groups, while for the radius going to ∞, the data will have the tendency to form only one group.

6. Conclusions and future plans

In the paper we have constructed CEC: a fast hybrid between k-means and EM, which allows easy simultaneous use of various Gaussian mixture models in clustering. Moreover, due to its nature CEC automatically removes unnecessary clusters and therefore can be successfully applied in typical situations where EM was used. Our method was successfully applied in image segmentation [37] and disk and ellipse pattern recognition [38, 39].

To practically use CEC we need two parameters: the initial maximal number of clusters k (which we usually fixed at 10) and the percent ε of population below which we deleted given cluster (we usually fix ε at 2 percent). The cost function CEC aims to minimize is given in the case of Gaussian densities by

$$\frac{N}{2}\ln(2\pi e) + \sum_{i=1}^{k} p(U_i)\cdot[-\ln(p(U_i)) + \frac{1}{2}\ln\det(\Sigma_{U_i})],$$

while for spherical Gaussians by

$$\frac{N}{2}\ln(\frac{2\pi e}{N}) + \sum_{i=1}^{k} p(U_i)\cdot[-\ln(p(U_i)) + \frac{N}{2}\ln \mathrm{tr}\Sigma_{U_i}].$$

In our implementation we used the Hartigans approach to find the minimum of the above cost functions.

In future we plan to apply CEC as a preprocessing method in data classification. In particular we want to use CEC as an alternative method for one-class SVM, MDLP and density clustering.

[1] J. Hartigan, Clustering algorithms, John Willey and Sons, 1975.

[2] A. Jain, R. Dubes, Algorithms for clustering data, Prentice-Hall, Inc., 1988.

[3] A. Jain, M. Murty, P. Flynn, Data clustering: A Review, ACM Computing Surveys 31 (1999) 264–323.

[4] A. Jain, Data clustering: 50 years beyond K-means, Pattern Recognition Letters 31 (2010) 651–666.

[5] R. Xu, D. Wunsch, Clustering, Wiley-IEEE Press, 2009.

[6] H. Bock, Clustering methods: a history of K-Means algorithms, Selected contributions in data analysis and classification (2007) 161–172.

[7] H. Bock, Origins and extensions of the k-means algorithm in cluster analysis, Journal Electronique dHistoire des Probabilités et de la Statistique Electronic Journal for History of Probability and Statistics 4 (2008).

[8] R. Tibshirani, G. Walther, T. Hastie, Estimating the number of clusters in a data set via the gap statistic, Journal of the Royal Statistical Society: Series B (Statistical Methodology) 63 (2001) 411–423.

[9] B. Mirkin, Choosing the number of clusters, Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 1 (2011) 252–260.

[10] V. Estivill-Castro, J. Yang, Fast and robust general purpose clustering algorithms, PRICAI 2000 Topics in Artificial Intelligence (2000) 208–218.

[11] S. Massa, M. Paolucci, P. Puliafito, A new modeling technique based on Markov chains to mine behavioral patterns in event based time series, DataWarehousing and Knowledge Discovery (1999) 802–802.

[12] G. McLachlan, T. Krishnan, The EM algorithm and extensions, volume 274, Wiley New York, 1997.

[13] A. Samé, C. Ambroise, G. Govaert, An online classification EM algorithm based on the mixture model, Statistics and Computing 17 (2007) 209–218.

[14] T. Cover, J. Thomas, J. Wiley, et al., Elements of information theory, volume 6, Wiley Online Library, 1991.

[15] D. MacKay, Information theory, inference, and learning algorithms, Cambridge Univ Pr, 2003.

[16] S. Kullback, Information theory and statistics, Dover Pubns, 1997.

[17] C. Shannon, A mathematical theory of communication, ACM SIGMOBILE Mobile Computing and Communications Review 5 (2001) 3–55.

[18] P. Grünwald, The minimum description length principle, The MIT Press, 2007.

[19] P. Grünwald, I. Myung, M. Pitt, Advances in minimum description length: Theory and applications, the MIT Press, 2005.

[20] Y. Ma, H. Derksen, W. Hong, J. Wright, Segmentation of multivariate mixed data via lossy data coding and compression, Pattern Analysis and Machine Intelligence, IEEE Transactions on 29 (2007) 1546–1562.

[21] A. Yang, J. Wright, Y. Ma, S. Sastry, Unsupervised segmentation of natural images via lossy data compression, Computer Vision and Image Understanding 110 (2008) 212–225.

[22] B. Kulis, M. I. Jordan, Revisiting k-means: New algorithms via bayesian nonparametrics, in: Proceedings of the 29th International Conference on Machine Learning (ICML), Edinburgh, UK, 2012, pp. 513–520.

[23] K. Kurihara, M. Welling, Bayesian k-means as a maximization-expectation algorithm, Neural computation 21 (2009) 1145–1172.

[24] H.-P. Kriegel, P. Kröger, J. Sander, A. Zimek, Density-based clustering, Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 1 (2011) 231–240.

[25] K. Rose, E. Gurewitz, G. Fox, A deterministic annealing approach to clustering, Pattern Recognition Letters 11 (1990) 589–594.

[26] M. Halkidi, Y. Batistakis, M. Vazirgiannis, On clustering validation techniques, Journal of Intelligent Information Systems 17 (2001) 107–145.

[27] J. Goldberger, S. T. Roweis, Hierarchical clustering of a mixture model, in: Advances in Neural Information Processing Systems, pp. 505–512.

[28] B. Zhang, C. Zhang, X. Yi, Competitive em algorithm for finite mixture models, Pattern recognition 37 (2004) 131–144.

[29] M. A. T. Figueiredo, A. K. Jain, Unsupervised learning of finite mixture models, Pattern Analysis and Machine Intelligence, IEEE Transactions on 24 (2002) 381–396.

[30] R. S. Wallace, T. Kanade, Finding natural clusters having minimum description length, in: Pattern Recognition, 1990. Proceedings., 10th International Conference on, volume 1, IEEE, pp. 438–442.

[31] A. Ben-Hur, D. Horn, H. T. Siegelmann, V. Vapnik, Support vector clustering, The Journal of Machine Learning Research 2 (2002) 125–137.

[32] W. Härdle, L. Simar, Applied multivariate statistical analysis, Springer, 2012.

[33] S. Dasgupta, Learning mixtures of gaussians, in: Foundations of Computer Science, 1999. 40th Annual Symposium on, IEEE, pp. 634–644.

[34] D. Arthur, S. Vassilvitskii, k-means++: The advantages of careful seeding, in: Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms, Society for Industrial and Applied Mathematics, pp. 1027–1035.

[35] N. Otsu, A threshold selection method from gray-level histogram, IEEE Trans. on Systems, Man, and Cybernetics 9 (1979) 62–66.

[36] B. Gatos, K. Ntirogiannis, I. Pratikakis, Icdar 2009 document image binarization contest (dibco 2009), in: Document Analysis and Recognition, 2009. ICDAR'09. 10th International Conference on, IEEE, pp. 1375–1382.

[37] M. Śmieja, J. Tabor, Image segmentation with use of cross-entropy clustering, in: Proceedings of the 8th International Conference on Computer Recognition Systems CORES 2013, Springer, pp. 403–409.

[38] P. Spurek, J. Tabor, E. Zając, Detection of disk-like particles in electron microscopy images, in: Proceedings of the 8th International Conference on Computer Recognition Systems CORES 2013, Springer, pp. 411–417.

[39] J. Tabor, K. Misztal, Detection of elliptical shapes via cross-entropy clustering, in: Pattern Recognition and Image Analysis, Springer, 2013, pp. 656–663.

[40] J. D. Banfield, A. E. Raftery, Model-based gaussian and non-gaussian clustering, Biometrics (1993) 803–821.

[41] W. L. Martinez, A. R. Martinez, Exploratory data analysis with MATLAB, CRC Press, 2005.

[42] C. Davis-Stober, S. Broomell, F. Lorenz, Exploratory data analysis with MATLAB, Psychometrika 72 (2007) 107–108.

[43] R. De Maesschalck, D. Jouan-Rimbaud, D. Massart, The mahalanobis distance, Chemometrics and Intelligent Laboratory Systems 50 (2000) 1–18.

[44] D. Aloise, A. Deshpande, P. Hansen, P. Popat, NP-hardness of Euclidean sum-of-squares clustering, Machine Learning 75 (2009) 245–248.

[45] S. Lloyd, Least squares quantization in pcm, Information Theory, IEEE Transactions on 28 (1982) 129–137.

[46] J. MacQueen, et al., Some methods for classification and analysis of multivariate observations, in: Proceedings of the fifth Berkeley symposium on mathematical statistics and probability, volume 1, California, USA, p. 14.

[47] M. Telgarsky, A. Vattani, Hartigans method: k-means clustering without voronoi, Journal of Machine Learning Research-Proceedings Track 9 (2010) 820–827.

[48] D. Arthur, S. Vassilvitskii, k-means++: The advantages of careful seeding, in: Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms, Society for Industrial and Applied Mathematics, pp. 1027–1035.