STUDIA INFORMATICA 2013
Volume 34 Number 1 (110)

Radosław SOKÓŁ
Silesian University of Technology, Institute of Electrical Engineering and Informatics
Andrzej POLAŃSKI
Silesian University of Technology, Institute of Informatics
Polish-Japanese Institute of Information Technology

COMPARISON OF METHODS FOR INITIALIZING EM ALGORITHM FOR ESTIMATION OF PARAMETERS OF GAUSSIAN, MULTI-COMPONENT, HETEROSCEDASTIC MIXTURE MODELS

Summary. A basic approach to estimation of mixture model parameters is by using expectation maximization (EM) algorithm for maximizing the likelihood function. However, it is essential to provide the algorithm with proper initial conditions, as it is highly dependent on the first estimation ("guess") of parameters of a mixture. This paper presents several different initial condition estimation methods, which may be used as a first step in the EM parameter estimation procedure. We present comparisons of different initialization methods for heteroscedastic, multi-component Gaussian mixtures.

Keywords: expectation-maximization, EM algorithm, pattern matching, initial conditions

PORÓWNANIE METOD INICJALIZAJI ALGORYTMU EM DLA WIELOSKŁADNIKOWYCH HETEROSCEDASTYCZNYCH MIESZANIN ROZKŁADÓW NORMALNYCH

Streszczenie. Algorytm EM (ang. *expectation-maximization*) jest szeroko stosowanym rozwiązaniem problemu estymacji parametrów mieszanin rozkładów prawdopodobieństwa poprzez maksymalizację wiarygodności. Istotne znaczenie dla działania algorytmu mają parametry początkowe, stanowiące pierwsze przybliżenie badanej mieszaniny. Publikacja przybliża kilka metod wyznaczania warunku początkowego dla iteracji algorytmu EM oraz porównuje ich skuteczność dla przypadku heteroscedastycznych, wieloskładnikowych mieszanin rozkładów normalnych.

Słowa kluczowe: expectation maximization, algorytm EM, dopasowanie do wzorca, warunki początkowe

1. Introduction

Mixture models have extremely wide range of applications in statistical data analysis, e.g. [5, 7, 1, 2, 3]. Areas and examples of their applications include detecting structure in animal populations [3], separating sources of variability of experimental data in cellular biology [8], analyses of images, e.g., medical imaging, separating areas of different types [10], estimating concentrations of different chemical compounds or protein or peptide species in samples by means of NMR or protein mass spectrometry [11, 13], analyses of factors behind financial market behaviors [9], and many others. Due to the importance of Gaussian distribution, mixtures of Gaussian distributions play a special role in the area of mixture modeling.

A challenging problem in applications of mixture distributions is fitting mixture parameters to data. A standard approach by maximization of the likelihood function, due to the unobservable mechanism of generation of data by different sources, cannot be carried out analytically. Likelihood maximization for mixture distributions is most often performed by using expectation maximization (EM) algorithm [1, 2, 3], i.e., a recursive procedure involving computing conditional expectations of unknown identities of data source, given available data (E-step) followed by maximization with respect to weights and parameters of mixture distributions (M-step). The construction of the EM procedure guarantees a step by step increase of the likelihood function. Unknown identities of data sources are called hidden or latent variables.

Despite a step-by-step increase of the value of the likelihood function EM iterations may fail to converge or may "stick" to a local maximum corresponding to undesired values of the mixture parameters, far from optimal/reasonable estimates, e.g. [3]. Due to both of these issues, a problem of basic importance is the choice of initial conditions for iterations of the EM algorithm. A good choice of a starting point for EM iterations can result in reducing the probability of erroneous estimation of parameters as well as in faster convergence of EM iterations. Several researches and results concerning the influence of the strategy of the choice of the initial condition for EM iterations on the performance of the whole algorithm were published in the literature, both in monographs [3, 2] and in journal or conference papers [23, 24, 28, 30, 31, 32]. A terminology used in [24] has been referenced in several later papers, so we base on it when describing approaches to setting up initial conditions for EM iterations. The simplest and obvious approach is random initialization (REM) involving generation of initial values of parameters and component weights by using some assumed probability distributions. Random generation can be done by using different probability distributions, often normal or uniform distributions are used for mean values of components and uniform or deterministic values are used for standard deviations and component weights. Parameters of

distributions are chosen such that initial mean values of components fall within ranges given by observed data.

Multiple short runs (SREM) initialization involves multiple repeats of randomly initialized EM recursions which are stopped after given number of iterations (short runs — typically corresponding to 10-20 iterations). The simplest version of short runs initiation involves generating multiple initial values for random methods and starting EM iterations for the one corresponding to highest likelihood. This method is called here multiple initializations EM. Then only one iteration process, namely the one, which attained the highest value of the likelihood function is continued.

Classification (clustering) initialization (CEM) includes a family of approaches where some kind of clustering procedure applied for the data set is used to compute initial parameters for EM iterations. Most often hierarchical clustering or *k*-means clustering algorithms are used.

Finally, stochastic initialization (SEM) stands for an approach where EM iterations are modified, most often in such a way that searching through parameter space is more intensive, which leads to avoiding local maxima of the likelihood function. In addition to SEM methods there are also many other versions of modifications of EM iterations with the analogous aim of improving intensity of searching through parameter space. These methods can be used as algorithms for setting initial values of mixture parameters, after a run of a modified EM iterations standard EM iterations are then applied [28, 33]. Some other methods [35, 34, 17] can in principle be used in the same manner for initiation of EM iterations, but their authors rather recommend replacing EM iterations by the modified versions. Results presented in the overviewed papers concern mixtures of both univariate and vector valued distributions.

Users of available software packages for mixture modeling, [15, 17, 16] are provided with the possibility of setting initial conditions for EM iterations by choosing one of the listed methods.

In this paper we readdress the problem of comparison of methods of initializing EM iterations for mixtures of Gaussian distributions. Papers, where related researches were reported, are [23, 24, 25]. In [23] initialization methods were compared for univariate mixtures of 1, 2 and 3 components, with the additional assumption on homoscedascity (equal variances). Papers [24, 25] concerned methods of initialization of EM iterations for multivariate (two dimensional) Gaussian mixtures with number of components ranging from 2 to 4. In this paper we restrict our study to comparison of methods for initialization of EM iterations for univariate Gaussian mixtures. In the present paper we extend the area explored by previous studies by (i) analyzing mixtures of Gaussian distributions where number of components can take considerably larger values (in numerical experiments we study the range from 5 through

10 components) and (ii) allowing general heteroscedastic (unequal variances) case. We study performances of different initialization methods for mixtures of Gaussian components with (i) different numbers of components, (ii) mixtures with equal variances versus mixtures with unequal variances, (iii) mixtures with different degrees of overlap between components.

An example of an area of applications where assumptions taken in the performed study are justified, is the analysis of spectra, protein mass spectra or nucleic magnetic resonance NMR spectra, or their fragments. Protein mass and NMR spectra can be well modeled with the use of mixtures of Gaussian distributions [12, 13]. There are usually numerous components of the spectra, well modeled by Gaussian distribution functions and different components can have different widths.

We perform analyses of efficiencies of different initialization methods and we formulate some recommendations concerning possible applications and further developments.

2. EM algorithm for fitting Gaussian mixture parameters to data

A univariate Gaussian mixture model, which we study in this paper has the form of weighted sum of component probability density functions

$$f^{\text{mix}}(x, p) = \sum_{k=1}^{K} \alpha_k f(x, \mu_k, \sigma_k) \tag{1}$$

where component probability density functions are given by Gaussian distributions

$$f(x, \mu_k, \sigma_k) = \frac{1}{\sigma_k \sqrt{2\pi}} \exp - \frac{1}{2} \left(\frac{x - \mu_k}{\sigma_k} \right)^2 \tag{2}$$

and component weights satisfy normalization condition

$$\sum_{k=1}^{K} \alpha_k = 1. \tag{3}$$

Parameters of component Gaussian distribution functions, means and standard deviations, μ_1, ..., μ_K, σ_1, ..., σ_K, and component weights α_1, ..., α_K, are elements of the parameter vector **p** in (1)

$$\mathbf{p} = [\alpha_1, ..., \alpha_K, \mu_1, ..., \mu_K, \sigma_1, ..., \sigma_K].$$

Given a vector **x** of (independent) available univariate observations

$$\mathbf{x} = [x_1, x_2, ... x_N] \tag{4}$$

the problem of fitting the model (1) (2) to data (4) is solved by maximization of the log likelihood function

$$L(\mathbf{x},\mathbf{p})=\sum_{n=1}^{N}\log f^{\text{mix}}(x_n,\mathbf{p}). \tag{5}$$

One, general assumption is that the number of observations must exceed the number of components by a factor, which is often taken as at least *10*.

Maximization cannot be achieved analytically, therefore expectation maximization (EM) algorithm is used. In order to apply EM algorithm we define hidden (missing) variables as unknown identities of components, χ_1, ..., χ_N, which had generated observations x_1, ..., x_N. Given a parameter guess, denoted by $\mathbf{p}^{\text{old}}$

$$\mathbf{p}^{\text{old}}=\left[\alpha_1^{\text{old}},...,\alpha_K^{\text{old}},\mu_1^{\text{old}},...,\mu_K^{\text{old}},\sigma_1^{\text{old}},...,\sigma_K^{\text{old}}\right] \tag{6}$$

conditional distributions of hidden variables can be computed by using Bayesian formula

$$P[\chi_n=k]=\frac{\alpha_k^{\text{old}}f\left(x_n,\mu_k^{\text{old}},\sigma_k^{\text{old}}\right)}{\sum_{\kappa=1}^{K}\alpha_\kappa^{\text{old}}f\left(x_n,\mu_\kappa^{\text{old}},\sigma_\kappa^{\text{old}}\right)}, \tag{7}$$

and, consequently the following expression for updates for values of parameters can be derived [1, 2, 3]

$$\alpha_k^{\text{new}}=\frac{\sum_{n=1}^{N}P[\chi_n=k]}{N}, \tag{8}$$

$$\mu_k^{\text{new}}=\frac{\sum_{n=1}^{N}x_nP[\chi_n=k]}{\sum_{n=1}^{N}P[\chi_n=k]} \tag{9}$$

and

$$\left(\sigma_k^{\text{new}}\right)^2=\frac{\sum_{n=1}^{N}\left(x_n-\mu_k^{\text{new}}\right)^2P[\chi_n=k]}{\sum_{n=1}^{N}P[\chi_n=k]}, \tag{10}$$

where k = 1, 2, ..., K. It is easily seen, in (9), that iterations (7)-(10) preserve normalization condition for μ_1, ..., μ_K.

After defining

$$\mathbf{p}^{\text{new}}=\left[\alpha_1^{\text{new}},...,\alpha_K^{\text{new}},\mu_1^{\text{new}},...,\mu_K^{\text{new}},\sigma_1^{\text{new}},...,\sigma_K^{\text{new}}\right] \tag{11}$$

and substituting $\mathbf{p}^{\text{old}}=\mathbf{p}^{\text{new}}$, expressions (7)-(11) become a recursion. Expressions (7)-(10) used in a recursive manner are called EM iterations. In order to start the EM iterations some

reasonable initial values of mixture model parameters must be substituted for $\mathbf{p}^{old}$. EM iterations are executed and continued until a suitably defined termination criterion is satisfied.

In the forthcoming sections several methods for computing initial values of mixture parameters will be presented and compared for several artificially created datasets.

2.1. Execution of the EM iterations

Properties of log likelihood functions for mixture distributions as well as convergence of the EM algorithm and EM iterations for estimating parameters of mixtures of distributions were studied and described in [21, 22, 3, 2] and [18, 20, 19]. It is well known that in the general case of unequal variances of components of the Gaussian mixture, the log likelihood (5) is unbounded [21]. Unboundedness of the log likelihood functions results in the fact that global maximum does not exist. It also results in the fact that in practical computations with the use of recursions (7)-(10) a numerical divergence can be encountered. Nevertheless, among local maxima (also called internal maxima) of the log likelihood function there is a sequence, which corresponds to consistent estimates of mixture parameters [21, 22].

Several modifications were proposed in the literature, aimed at modifying the form of the likelihood function / optimization criterion such that unboundedness is removed [36, 37, 38, 39]. In [38] it was demonstrated that modification of the likelihood function by introducing constraints on ratios of variances of components both prevents divergence of iterations and leads to better convergence properties of the EM iterations for Gaussian mixture distributions. Several authors propose some other, partly heuristic, modifications of the form of EM iterations aimed at avoiding divergence and "sticking" to undesired local maxima and/or to speeding up convergence e.g. [33, 34, 35]. These modifications are shown to improve performance of EM iterations for estimation of mixture parameters.

In this paper we take we take the original approach to the execution of the EM iterations, based on using iterations (7)-(10) without any modifications. In this approach, depending on the strategy for setting initial condition, divergence of iterations may take place. In our software implementation for EM iterations occurrences of divergent solutions are detected and registered. Then, statistics of divergent solutions are compared between different methods of initiation of EM iterations.

2.2. Termination criterion

Termination of EM iterations is usually based on either the change of the log likelihood function or on the change of parameter values [3]. Here we take the approach based on the

scaled change of values of parameters between two successive iterations. We use the following formula

$$\varepsilon = \sum_{k=1}^{K} \left| \alpha_k - \alpha_k^{\text{old}} \right| + \frac{1}{K} \sum_{k=1}^{K} \frac{\left| \mu_k - \mu_k^{\text{old}} \right|}{\sigma_k} + \frac{1}{K} \sum_{k=1}^{K} \frac{\left| \sigma_k - \sigma_k^{\text{old}} \right|}{\sigma_k} . \tag{12}$$

If the value of ε given by above formula is lower than the threshold (assumed equal to 10^{-4}) iterations are terminated with a "convergence" status.

The scaling method in the formula (12) is aimed at making the stopping criterion invariant with respect to the size of the decomposition problem (number of components) and with respect to different widths of Gaussian components.

3. Algorithms for setting initial values

In this section we describe algorithms for setting initial conditions for EM iterations used in this study. Each of the method of initiation of EM iterations has many possible variants of deciding on the full set of parameters. Most important is setting mean values (locations) of components. However, methods of setting initial values of component variances and weights have also influence on the performance of the algorithm. Comparison of methods for setting initial conditions for EM iterations requires making some heuristic choices, due to the fact that verifying all possible combinations is computationally too prohibitive. Algorithms presented below were constructed in such a way that component means were computed according to the main idea and then reasonable choices of methods for initial values of component variances and weights were applied.

3.1. True

The first method is initiating EM iterations by using true values of parameters. In the simulation study performed here the "true" method gives a reference for other methods.

3.2. Random methods (REM)

The simplest method of choosing an initial condition for the EM recursions is by generating it in a random manner. Methods reported most frequently in the literature are generating values of means of components by using uniform [23] or Gaussian distribution [3]. Here we use two methods, uniform with bounds defined by the range of observations and the method based on inverting the empirical cumulative distribution.

3.2.1. Uniform (Rand-U)

In the first case, as stated above, initial mean values of components are generated with the use of uniform probability distribution supported on the observed range of data

$$\mu_k^{\text{ini}} = U\left(\min(x), \max(x)\right). \tag{13}$$

In the above expression U(a,b) denotes uniform distribution supported on the interval <a,b>.

Initial values for component standard deviations and for component weights are also generated with the use of uniform distributions, as follows,

$$\sigma_k^{\text{ini}} = U\left(\sigma_{\min}, \sigma_{\max}\right), \tag{14}$$

$$\alpha_k^{*} = U\left(\alpha_{\min}, \alpha_{\max}\right), \tag{15}$$

$$\alpha_k^{\text{ini}} = \frac{\alpha_k^{*}}{\sum_{\kappa=1}^{K} \alpha_\kappa^{*}}. \tag{16}$$

Parameters of the above distributions $\alpha_{\min}$, $\alpha_{\max}$, $\sigma_{\min}$ and $\sigma_{\max}$ are chosen with the use of simple rules, as further described in the "Numerical Experiments" section. Clearly, values of component weights α^*_k in (15) must be scaled by (16) to conform to normalization condition (3). Final values of weights α^{ini}_k, given by (16) follow Dirichlet distribution, e.g. [24].

Abbreviation for random generation of initial means, variances and weights with uniform distribution, (13)-(16), is Rand-U.

3.2.2. Inverse-CDF (R-invCDF)

The second method of random generation of initial mixture parameters uses inversion of the empirical cumulative distribution function (CDF), and is called inverse CDF method. The inverse-CDF method is based on an empirical cumulative distribution function (CDF) of mixture data, given by the formula

$$F^{\text{emp}}(x) = \frac{1}{N}\sum_{n=1}^{N} I_{x \geq x}\ , \tag{17}$$

where $I_{x \geq xi}$ is the indicator function equal to zero for $x < x_i$ and one otherwise, x_i is the i-th sample of observed mixture data. The procedure starts with a selection of K random points, generated by using a uniform distribution:

$$r_k^{*} = U(0,1),\ k = 1...K \tag{18}$$

and then values r_1, r_2, ..., r_K are obtained by sorting r^*_k in an ascending order. Next, these random values (in the <0,1> range) are mapped onto axis of values of observations by inverting empirical cumulative distribution function (17) for the analyzed mixture:

$$\mu_k^{\text{ini}} = \left(F^{\text{emp}}\right)^{-1}(r_k). \tag{19}$$

Clearly, the distribution of $\mu^{\text{ini}}{}_k$ is an approximation of the distribution $f^{\text{mix}}(x,\mathbf{p})$. The intuition behind the inverse-CDF method is that mixture components should be located preferentially in regions of high values of the probability density function, which correspond to high values of slopes of the CDF function.

Initial values for component standard deviations and for component weights are generated with the use of a method different than that for the for uniformly distributed means, (14)-(16). For the case of means generated by the inverse CDF method we use the following rules for component standard deviation and weights

$$\sigma_1^{\text{ini}} = \tfrac{1}{2}\left[\mu_1^{\text{ini}} - \min(x)\right],\ \sigma_k^{\text{ini}} = \tfrac{1}{2}\left[\mu_{k+1}^{\text{ini}} - \mu_{k-1}^{\text{ini}}\right],\ k = 2,...,K-1,$$
$$\sigma_K^{\text{ini}} = \tfrac{1}{2}\left[\max(x) - \mu_K^{\text{ini}}\right] \tag{20}$$

and

$$\alpha_k^{\text{ini}} = \tfrac{1}{2}\left(r_{k+1} - r_{k-1}\right),\ k = 2,...,K, \tag{21}$$

where we additionally assume $r_0 = 0$ and $r_{K+1} = 1$.

Rules for computing standard deviations and weights, (20) and (21) are similar to those used in the case of hierarchical clustering methods, CEM, described in the next section.

Abbreviation for random generation of initial means, variances and weights with uniform distribution, (19)-(21), is R-invCDF.

3.3. Multiple initializations of random methods (Rand-U-20, R-invCDF-20)

We have also implemented initialization algorithms based multiple repetitions of random initialization methods Rand-U and Rand-invCDF. In these algorithms random initializations Rand-U or Rand-invCDF are repeated 20 times and then EM iterations are performed for the set of parameter values, which corresponded to the highest likelihood.

The algorithms for multiple initializations of random methods Rand-U and R-invCDF are abbreviated Rand-U-20 and R-invCDF-20.

3.4. Hierarchical clustering methods (CEM)

Hierarchical clustering methods, e.g. [6], create clusters of samples by using successive operation of merging. For samples (observations) x_1, x_2, ..., x_N we define a distance matrix

$$\mathbf{D} = \left[d\left(x_i, x_j\right) \right] \tag{22}$$

where $d(x_i,x_j)$ is the (Euclidean) distance between x_i and x_j,

$$d\left(x_i, x_j\right) = \left| x_i - x_j \right|. \tag{23}$$

Samples x_{i*} and x_{j*} with the shortest distance between each other

$$i^*, j^* \leftarrow \min_{i,j} d\left(x_i, x_j\right) \tag{24}$$

are then merged into new sample x_z

$$x_{i*}, x_{j*} \rightarrow x_z. \tag{25}$$

3.4.1. Average linkage clustering (Clust-AL)

In the simplest version of hierarchical clustering, called average linkage clustering, x_z is defined as the mean of x_{i*} and x_{j*}

$$x_z = \tfrac{1}{2}\left(x_{i*} + x_{j*}\right) \tag{26}$$

and then the distance matrix is updated such that new distances are defined by using mean operation

$$d\left(x_k, x_z\right) = \tfrac{1}{2}\left[d\left(x_k, x_i\right) + d\left(x_k, x_j\right) \right]. \tag{27}$$

3.4.2. Complete linkage clustering (Clust-CL)

Another version of hierarchical clustering method analyzed here, called complete linkage, uses merging operation in (25), which leads to creation of a union of indexes of samples

$$x_i, x_j \rightarrow x_{\{i,j\}}. \tag{28}$$

The above operation leads to creation of clusters of samples. Defining new distances between clusters $C_m = \{ i_1, i_2, ..., i_M \}$ and $C_l = \{ i_1, i_2, ..., i_L \}$ is based on the following definition

$$d\left(x_{\{i_1,i_2,\ldots i_M\}}, x_{\{i_1,i_2,\ldots i_L\}}\right) = \max d\left(x_{i_m}, x_{j_l}\right). \tag{29}$$

The above definition is also a base for merging clusters.

Both in the average linkage clustering and in complete linkage clustering, the process continues iteratively until the number of clusters assumes the desired value K.

In one dimensional case assumed here, creation of whole N×N distance matrix (22) is not necessary, only above-diagonal elements are important.

3.4.3. Initial values for parameters in initial clustering methods

After average or complete linkage clusters are formed, initial values of means for the EM iterations are computed with the use of the following method. For the m-th cluster containing samples $i_1, i_2, ..., i_M$, $C_m = \{ i_1, i_2, ..., i_M \}$, the corresponding initial mean is computed as

$$\mu_m^{\text{ini}} = \text{mean}\left(\text{x}_{\text{i}_1}, x_{i_2}, ..., x_{i_M}\right), \tag{30}$$

the initial value for standard deviation is computed as

$$\sigma_m^{\text{ini}} = \tfrac{1}{2}\left(x_{i_M}\, 0x_{i_1}\right), \tag{31}$$

and the initial value for component weight is computed by

$$\alpha_m^{\text{ini}} = \frac{\#C_m}{\#\mathbf{x}} = \frac{M}{N}, \tag{32}$$

where $\#C_m$ and $\#\mathbf{x}$ denote numbers of elements in the cluster C and in the vector $\mathbf{x}$, respectively.

Two analyzed clustering methods, described above, are represented in figures and tables by using abbreviations Clust-AL, and Clust-CL.

3.5. "Maximum over likelihoods" method (Max-Lik)

Implementation of several, different estimation methods gives a possibility of defining and studying a "meta" method, i.e., a method which uses/combines estimates of values of parameters obtained by using other, already implemented methods. We define here a "maximum of likelihoods" method, abbreviated as "Max-Lik" as a method, which takes estimates of mixture parameters equal to the parameters obtained by the method, which led to maximum of all likelihoods. We take maximum of likelihoods over the following set of methods: Rand-U, R-inv-CDF, Rand-U-20, R-inv-CDF-20, Clust-AL and Clust-CL. The method "True" is not on the list, due to the fact that it is only a reference and uses true parameters as starting values, not known in all other initialization methods.

4. Computational experiments

Computational experiments involved generating artificial datasets on the basis of known underlying Gaussian components, and their further analyses by using EM algorithms started with the above described algorithms of setting initial conditions.

4.1. Description of the created datasets

In the computational experiments performed, a total of 12 cases of mixtures of Gaussian distributions have been analyzed. These included two groups, of 5 component and 10 component mixtures. Each of the groups (of 5 and 10 component mixtures) contained 6 cases and each group could be further subdivided into 2 subgroups. One subgroup included mixtures of components with equal variances (homoscedastic) and the other one — components with un-

equal variances (heteroscedastic). Finally, each of the subgroups included 3 mixtures differing in the level of overlap between components, from the case of almost completely disjoint components, through the medium level of overlap to the case of a mixture with high overlap between components. We introduced labeling for cases of mixtures by using three symbols, a number (5 or 10) indicating the number of components, a symbol (O for homoscedascity or V for heteroscedascity) indicating equal or unequal variance and another symbol (N, L or H) indicating the level of overlap between components, no (N), low (L) or high (H). With this labeling, for example the symbol 5VL stands for a 5 component Gaussian mixture with unequal variances of components and low level of overlap between components.

Parameters of all mixtures are given in Tables 1 and 2.

Table 1

Parameters of analyzed 5-component Gaussian mixtures

Symbol	**Parameters**					
5OL	α_i	0.2	0.2	0.2	0.2	0.2
	μ_i	1	2	3	4	5
	σ_i	0.1	0.1	0.1	0.1	0.1
5OM	α_i	0.2	0.2	0.2	0.2	0.2
	μ_i	1	2	3	4	5
	σ_i	0.2	0.2	0.2	0.2	0.2
5OH	α_i	0.2	0.2	0.2	0.2	0.2
	μ_i	1	2	3	4	5
	σ_i	0.3	0.3	0.3	0.3	0.3
5VL	α_i	0.2	0.2	0.2	0.2	0.2
	μ_i	1	4	9	16	25
	σ_i	0.2	0.4	0.6	0.8	1.0
5VM	α_i	0.2	0.2	0.2	0.2	0.2
	μ_i	1	4	9	16	25
	σ_i	0.4	0.8	1.2	1.6	2.0
5VH	α_i	0.2	0.2	0.2	0.2	0.2
	μ_i	1	4	9	16	25
	σ_i	0.7	1.4	2.1	2.8	3.5

Table 2

Parameters of analyzed 10-component Gaussian mixtures

Symbol	**Parameters**										
10OL	α_i	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
	μ_i	0	2	4	6	8	10	12	14	16	18
	σ_i	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2
10OM	α_i	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
	μ_i	0	2	4	6	8	10	12	14	16	18
	σ_i	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4
10OH	α_i	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
	μ_i	0	2	4	6	8	10	12	14	16	18
	σ_i	0.6	0.6	0.6	0.6	0.6	0.6	0.6	0.6	0.6	0.6
10VL	α_i	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
	μ_i	0	2	5	9	14	20	27	35	44	54
	σ_i	0.1	0.25	0.4	0.55	0.7	0.85	1.0	1.15	1.3	1.45
10VM	α_i	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
	μ_i	0	2	5	9	14	20	27	35	44	54
	σ_i	0.2	0.5	0.8	1.1	1.4	1.7	2.0	2.3	2.6	2.9
10VH	α_i	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
	μ_i	0	2	5	9	14	20	27	35	44	54
	σ_i	0.3	0.75	1.2	1.65	2.1	2.55	3.0	3.45	3.9	4.45

Stochastic simulations with random normal numbers generators were used to create artificial datasets. Numbers of samples in each dataset were 1000 for 5 component mixtures and 2000 for 10 component mixtures.

4.2. EM iterations

For all instances of analyzed mixtures of Gaussian distributions, the EM algorithm (7)-(11) was launched many times, with each of the previously described method applied as a procedure for setting initial condition for iterations. Iterations continued unless any of the three possible conditions was encountered (i) termination criterion given in (12) was satisfied, (ii) divergence of iterations was detected (the condition for divergence was assumed as $\min(\sigma_i) \leq 10^{-4}$ or $\min(\alpha_i) \leq 10^{-4}$) or (iii) the upper limit for the number of iterations (assumed equal to 10000) was exceeded. Termination due to condition (i) is called normal while termination due to (ii) or (iii) is called abnormal.

5. Results

In this section we report results of fitting mixture models to generated datasets. First we define performance criteria used to score the obtained results, then we show Tables 3 and 4 which illustrate dependence of values of criteria on the strategy of computing initial condition for iterations.

5.1. Performance criteria

There are many possible methods to score performance of the algorithms for fitting mixture models to data. A criterion commonly used in the literature, based on the (final) value of the log likelihood function, is the percentage (probability) of attaining "maximum" likelihood by a method of initialization of EM iterations [23, 24]. "Maximum" likelihood is understood as the value of the likelihood obtained by using true values of parameters as initial values for EM iterations (for computations 5% margin is allowed). The probability of attaining "maximum" likelihood is denoted here by P(max).

In this study we also use a second, direct criterion, namely a scaled absolute difference between true and estimated locations of components, averaged over all components. Scaling is aimed at making the distribution of errors (differences) invariant with respect to component widths and to components weights. The criterion is defined as follows

$$Q = \frac{1}{K}\sum_{i=1}^{K}\frac{\left|\mu_{\mathrm{i}}^{\mathrm{true}} - \mu_i^{\mathrm{est}}\right|}{\sigma_i^{\mathrm{true}}}\sqrt{N\alpha_i^{\mathrm{true}}}\,. \tag{33}$$

In the above expression, μ_i^{true}, σ_i^{true} and α_i^{true} are true parameters of the analyzed mixture distribution, K is the number of mixture components and N is the sample size. By μ_i^{est} we understand the value of the estimated mixture component mean closest to μ_i^{true}. The above criterion allows scoring one experiment of estimating a mixture parameters. In order to characterize performance of a method we used mean value of criterion Q, mean(Q) following from multiple repetitions of EM iterations (terminated with the status-normal). Clearly, it only makes sense to use criterion Q (33) if EM iterations terminate with the status "normal".

In the case where components of the mixture are disjoint (well separated) and the estimation method (EM algorithm with an initial values close to true values) leads to correct assignment of samples to components, minimal expected value of the criterion Q can be computed theoretically as expected absolute value of a standard normal random variable

$$E(Q) = \int_0^{\infty} x\frac{2}{\sqrt{2\pi}}\exp\left(\frac{x^2}{2}\right)\mathrm{d}x = \frac{2}{\sqrt{2\pi}} \approx 0.7979\,. \tag{34}$$

Since EM iterations can terminate either normally or abnormally, as an additional, supplementary criterion we also use estimated probability of abnormal termination of the EM iterations, denoted as P(abnormal).

5.2. Tables of scores

Results of our experiments are reported in Tables 3 and 4, where for all 12 mixtures described in Tables 1 and 2, we report effects of using initialization algorithms Rand-U, R-inv-CDF, Rand-U-20, R-inv-CDF-20, Clust-AL, Clust-CL, Max-Lik and True. For initialization methods Rand-U, R-inv-CDF, Rand-U-20, R-inv-CDF-20, Clust-AL, Clust-CL and True we report values of mean(Q), P(max) and P(abnormal). The format of an entry of the table is: mean(Q) (P(max)) P (abnormal). For example, application of the Rand-U initialization method for the dataset 5OL leads to output reported as 10.175 (0.66) 0.03, which means

$$\text{mean}(Q) = 10.175,\ P(\text{max}) = 0.66,\ P(\text{abnormal}) = 0.03.$$

For the Max-Lik method the format of entry in Tables 3 and 4 is different. For the Max-Lik method, estimated probability of abnormal termination, P(abnormal) is not reported in Tables 3 and 4, due to the fact that in all experiments it never happened that for a dataset all methods Rand-U, R-inv-CDF, Rand-U-20, R-inv-CDF-20, Clust-AL, Clust-CL terminated with abnormal status. As a conclusion, for Max-Lik method probability P(abnormal) is always equal to 0. Instead of listing all values equal to 0, we report another probability defined as follows. Under the assumption of independence we can theoretically compute probability that at least one of the initialization methods, Rand-U, R-inv-CDF, Rand-U-20, R-inv-CDF-20, Clust-AL, Clust-CL, attained "maximum" likelihood (defined as above). This probability is denoted by P^{Est}(max) and is given by the following formula

$$P^{\text{Est}}(\text{max}) = 1 - \prod_{i \in \mathbf{I}} (1 - P_i(\text{max})). \tag{35}$$

In the above formula **I** = {Rand-U, R-inv-CDF, Rand-U-20, R-inv-CDF-20, Clust-AL, Clust-CL }. If the initialization methods Rand-U, R-inv-CDF, Rand-U-20, R-inv-CDF-20, Clust-AL, Clust-CL are independent (close to independent), we should (approximately) observe

$$P_{\text{Max}-\text{Lik}}(\text{max}) \approx P^{\text{Est}}(\text{max}). \tag{36}$$

The format of entries of the "Max-Lik" column of Table 4 is: mean(Q)$_{\text{Max-Lik}}$ ($P_{\text{Max-Lik}}$(max)) [P^{Est}(max)]. Contemplating reported values of both probabilities in (36), $P_{\text{Max-Lik}}$(max) and P^{Est}(max), allows verifying a hypothesis of independence of different initialization methods.

Table 3

Results of application of initialization methods
Rand-U, R-inv-CDF, Rand-U-20 and R-inv-CDF-20

	Rand-U			**R-inv-CDF**			**Rand-U-20**			**R-inv-CDF-20**		
5ON	10.175	(0.66)	0.03	8.883	(0.70)	0.04	8.093	(0.72)	0.04	7.340	(0.75)	0.02
5OL	4.707	(0.76)	0.0	4.724	(0.73)	0.01	3.450	(0.81)	0.01	2.957	(0.84)	0.01
5OH	2.060	(0.80)	0.0	3.433	(0.67)	0.01	1.986	(0.81)	0.0	2.424	(0.78)	0.01
5VN	23.606	(0.28)	0.24	22.681	(0.39)	0.07	13.452	(0.54)	0.11	14.666	(0.53)	0.04
5VL	16.198	(0.20)	0.07	11.620	(0.38)	0.02	9.620	(0.45)	0.03	6.762	(0.60)	0.0
5VH	5.061	(0.45)	0.05	4.193	(0.64)	0.01	3.407	(0.79)	0.02	3.433	(0.73)	0.01
10ON	9.568	(0.37)	0.18	15.284	(0.20)	0.21	6.483	(0.56)	0.08	13.892	(0.24)	0.13
10OL	6.503	(0.30)	0.05	8.324	(0.21)	0.02	4.736	(0.47)	0.07	7.813	(0.21)	0.04
10OH	4.921	(0.33)	0.09	6.200	(0.25)	0.05	3.826	(0.49)	0.05	5.903	(0.26)	0.03
10VN	39.987	(0.01)	0.34	22.530	(0.06)	0.19	26.111	(0.04)	0.24	17.150	(0.10)	0.13
10VL	17.226	(0.01)	0.11	10.281	(0.09)	0.04	12.871	(0.05)	0.08	8.524	(0.13)	0.03
10VH	8.932	(0.18)	0.19	6.718	(0.35)	0.16	7.353	(0.26)	0.19	6.658	(0.39)	0.15

Table 4

Results of application of initialization methods
Clust-AL, Clust-CL, Max-Lik and True

	Clust-AL			**Clust-CL**			**Max-Lik**			**True**		
5ON	0.797	(1.0)	0.0	0.797	(1.0)	0.0	0.797	(1.0)	[1.0]	0.797	(1.0)	0.0
5OL	0.831	(1.0)	0.0	0.888	(1.0)	0.0	0.831	(1.0)	[1.0]	0.831	(1.0)	0.0
5OH	1.862	(0.85)	0.13	1.989	(0.85)	0.0	1.699	(0.97)	[0.99]	1.655	(1.0)	0.0
5VN	2.094	(0.79)	0.18	31.379	(0.04)	0.08	1.266	(0.98)	[0.98]	0.787	(1.0)	0.0
5VL	16.868	(0.00)	0.19	16.736	(0.02)	0.03	2.344	(0.90)	[0.89]	0.850	(1.0)	0.0
5VH	7.244	(0.12)	0.19	5.434	(0.31)	0.06	2.525	(0.97)	[0.99]	2.318	(1.0)	0.0
10ON	0.797	(1.0)	0.0	0.797	(1.0)	0.0	0.797	(1.0)	[1.0]	0.797	(1.0)	0.0
10OL	0.875	(1.0)	0.0	1.507	(0.91)	0.0	0.875	(1.0)	[1.0]	0.875	(1.0)	0.0
10OH	1.953	(0.78)	0.18	3.100	(0.68)	0.01	2.026	(0.94)	[0.99]	1.765	(1.0)	0.0
10VN	20.659	(0.0)	0.57	40.952	(0.0)	0.07	13.051	(0.19)	[0.20]	0.804	(1.0)	0.0
10VL	17.391	(0.0)	0.22	17.340	(0.0)	0.08	6.392	(0.27)	[0.25]	1.116	(1.0)	0.0
10VH	9.107	(0.12)	0.26	8.909	(0.14)	0.18	6.727	(0.52)	[0.81]	3.949	(0.85)	0.15

6. Discussion

In this section we comment on the comparisons on different initialization strategies shown in Tables 3 and 4. As illustrations of our computational experiments we also give figures, which present in a more comprehensive and detailed way some phenomena observed

when fitting mixture models to data and how they depend on parameters and on initiation strategy. We also try to apply simple computational models to explain values reported in Tables 3 and 4. Further in the Conclusions section we summarize some recommendations concerning practical problems of fitting mixture models to spectroscopic data, which can be drawn from our research.

6.1. Comments on the created and analyzed datasets

The datasets created and analyzed in this paper have one common feature, which can be called linear or sequential structure. By this name we mean that possible overlaps occur only between neighboring components. In contrast, artificially created datasets for benchmarking algorithms for analyzing mixtures of density functions, which can be found in the literature can exhibit more complicated overlap structures (e.g., claw-like probability density functions [3]). Confining the analysis to a narrower class of probability density functions is aimed at making possible obtaining some explicit conclusions on the influence of the structure of data on the performance of different initialization methods. The structures of the analyzed datasets (Tables 1 and 2) were chosen in order to support studies on the influence of (i) homoscedascity versus heteroscedascity, (ii) the extent of the overlap between Gaussian components and (iii) the number of components in the mixture, on the performance of different methods of initialization of EM iterations. Comments on these influences are given in the following subsections.

6.2. Comments on comparisons

When contemplating entries of Tables 3 and 4 one can observe that using different methods of initialization of EM iterations can lead to substantial differences in the obtained results. Theoretical lower bound, given by equation (34) is attained by True method for all mixtures with no overlap between components and by both clustering methods CL and AL but only for homoscedastic, non overlapping mixtures 5ON and 10ON. For the case of homoscedastic mixtures with low or high level of overlap between components, 5OL, 5OH, 10OL, 10OH, initialization methods basing of hierarchical clustering still perform very well, similarly or only slightly worse than the True method. However, performance of hierarchical clustering methods changes for the case of heteroscedastic mixtures. For this case hierarchical clustering initialization methods lead to substantially worse results and can even be outperformed by some of random initialization methods. The decline of the performance of hierarchical clustering methods for heteroscedastic case is further amplified by increasing the number of components in the mixture. Two clustering methods, CL and AL can lead to different out-

comes. The average linkage method AL performs better for mixtures with no or low overlap. This method, however, ho has also always higher probabilities of abnormal termination, P(abnormal).

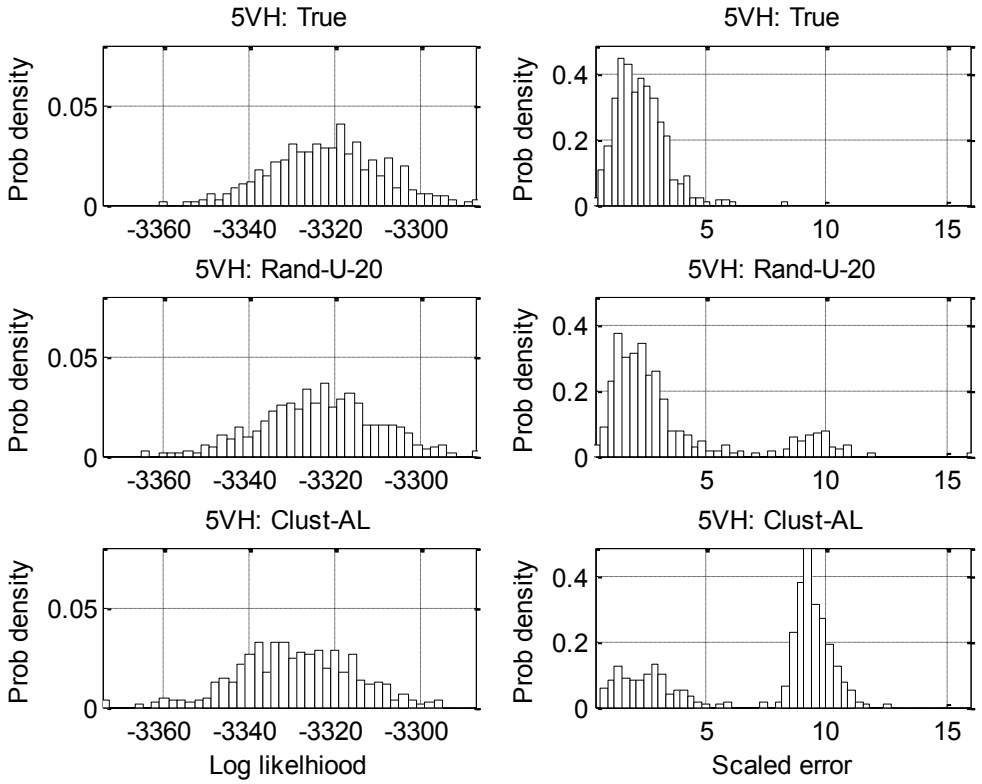

Fig. 1. Plots in the right panel of the figure demonstrate multi-modality of distributions of scaled errors, which occurs despite quite strong overlap between mixture components. In contrast, distributions of log likelihoods are here unimodal, with shapes well approximated by normal distributions

Rys. 1. Wykresy po prawej stronie rysunku przedstawiają wielomodalność rozkładów przeskalowanych błędów, występującą mimo znacznego przekrywania się składników mieszaniny. Rozkład wiarygodności logarytmicznej w tym przypadku jest unimodalny, a jego kształt można przybliżyć rozkładem normalnym

Random methods show similarities in their outcomes in the sense that they always lead to considerably higher average errors than the True method. The methods based on inverting CDF, lead to higher probabilities of abnormal termination, which results from its strategy of setting initial variances, leading to narrower initial components. Variants with multiple initializations always lead to some improvement of the average scaled error.

The Max-Lik is a reasonable method of aggregating (combining) results obtained with different methods of initializing EM iterations. As seen in Table 4, it leads to substantial improvement of the value of mean scaled error in all analyzed mixture datasets.

6.3. Comments on distributions of scaled errors and log likelihoods

Probability distributions of scaled errors and log likelihoods obtained in the performed computational are often multi-modal, which follows from the fact that analyzed datasets are multi-component. In Figure 1, as an example, we show plots corresponding to probability distributions of log likelihoods (left) and scaled errors (right) in the experiment 5VH, for three initialization methods True, Rand-U-20 and Clust-AL.

In the cases where there is a high overlap between mixture components one can observe the existence of the so called "local spurious maximizers" in the obtained solutions. This phenomenon is well known in the literature, e.g. [3]. By existence of "local spurious maximizers" we mean a situation where orders of values of log likelihoods and (scaled) errors, corresponding to some of obtained solutions to estimation of mixture parameters problems, are inverted. In other words, normally the greater value of the likelihood would imply smaller value of the scaled error, but there are situations that for a generated dataset two solutions X and Y are such that likelihood of X > likelihood of Y, but scaled error of X > scaled error of Y. One example of a spurious local maximizer encountered in computational experiments is presented in Figure 2.

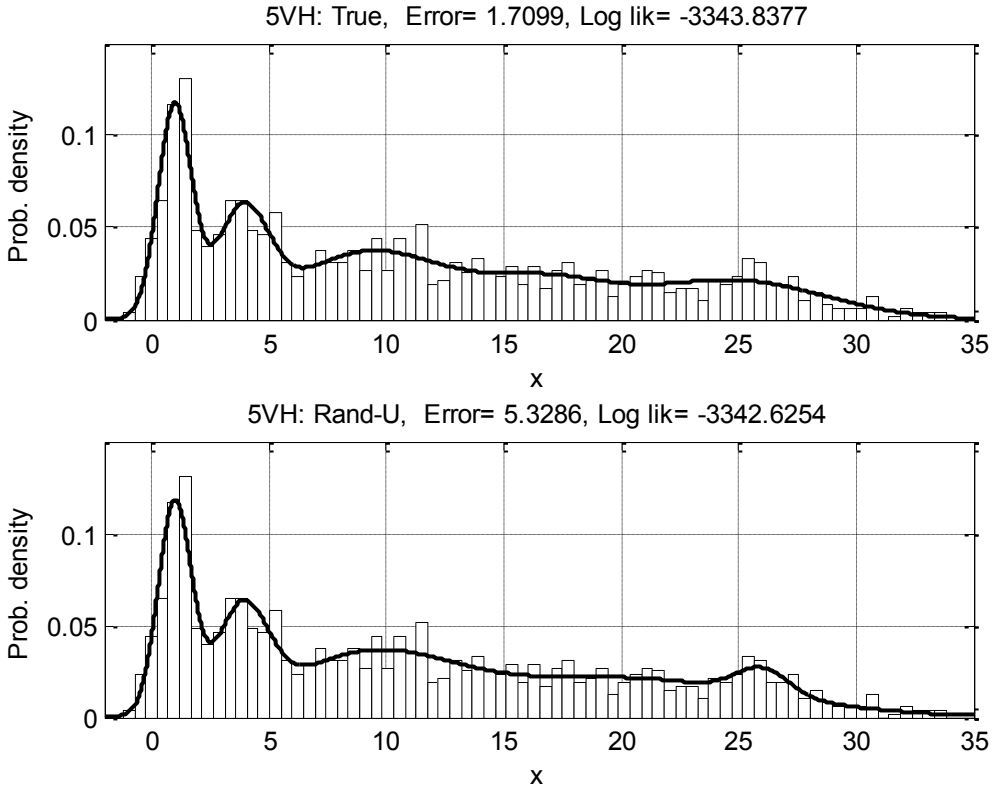

Fig. 2. Illustration of the phenomenon of spurious local maximizers
Rys. 2. Ilustracja zjawiska pozornych lokalnych maksimów

We do not report exact statistics here, but occurrences of spurious local maximizers are rather rare in the analyzed datasets. Despite the existence of spurious local maximizers values of the criteria mean(Q) and P(max), are strongly negatively correlated. Correlation coefficient, computed over all simulation experiments is -0.76. Another fact, confirming that spurious local maximizers do not have a very strong impact on statistics of solutions is observed high efficiency of the Max-Lik method. It should, nevertheless, be noted that frequency of occurrence of spurious local maximizers increases with the increase of the number of components in the mixture model.

7. Conclusions

In this paper we have presented a comparison study of methods for initialization of EM iterations for univariate, multi component, heteroscedastic Gaussian mixtures models. Comparing different strategies for initialization of the EM algorithm is a surprisingly complex issue. Comparisons are complicated by facts that, (i) EM algorithm, despite its simplicity, can exhibit complex behavior (unboundedness, non-convergence), and (ii) initialization and application of the EM algorithm for estimating mixture parameters requires using many parameters and making many decisions, whose influence on the results interfere one with another. Understanding this influence requires systematic studies.

A set of methods implemented in this study includes mostly algorithms already described and tested in the cited references. One exception is the inverse CDF method , which according to our knowledge was not previously researched. It, however, belongs to the group of random methods and is quite similar to other methods in this group. The contribution, compared to previous papers is exploring the case of multi - component heteroscedastic mixtures. Similarly to previous studies [23, 24, 25] we must report that no single initialization method (Rand-U, R-inv-CDF, Rand-U-20, R-inv-CDF-20, Clust-AL and Clust-CL) can outperform others in all experiments. The method, which always leads to an improvement in terms of average scaled error, is our meta method Max-Lik.

We have assumed that the number of components in the mixture was known. In practical applications this assumption is rather never true. The problem of estimating the number of components in the mixture is a separate research area, and is approach by different methods, e.g., by using Bayesian Information Criterion (BIC) [3].

As far as implications of our study for analyses of real data are concerned, it should be stated that the initialization algorithms in their original forms, studied in the literature, are rather not enough efficient for practical computations for proteomic or NMR spectra. For practical applications further improvements are necessary. An obvious way of develop-

ing/improving algorithms is including a priori knowledge of the structure of mixtures into the algorithm (both to initialization methods and to execution phase of EM). This knowledge can be available for real data and includes lower and upper bounds on variances of components and bounds on overlaps between components. Another way is developing more efficient and sophisticated initialization algorithms, which can be an area of further studies.

Acknowledgments

This work was financially supported by the European FP6 project GENEPI low RT, by the Polish Ministry of Science Grant: "Efficient methods of genome browsing based on the Burrows Wheeler Transform".

BIBLIOGRAPHY

1. Dempster A. P., Laird N. M., Rubin D. B.: Maximum likelihood from incomplete data via the EM algorithm (with discussion). J. R. Statist. Soc., Ser. B, vol. 39, 1977, p. 1÷38.
2. McLachan G. J., Krishnan T.: The EM Algorithm and Extensions. Wiley, 1997.
3. McLachan G. J., Peel W.: Finite Mixture Distributions. Wiley, 2000.
4. Bohning D., Seidel W.: Recent developments in mixture models. Comput. Statist. Data Anal., 41 (2003), p. 349÷357.
5. Recent Developments in Mixture Model, Special Issue. Computational Statistics and Data Analysis Volume 41, Issues~3-4, 28 January 2003.
6. Hastie T., Tibshirani R., Friedman J.: The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Second edition, Springer Verlag, Berlin 2009.
7. Sociological Methods and Research, Special Issue. Volume 29 Issue 3, 2001.
8. Gyllenberg M., Koski T., Lund T.: Applying the EM-algorithm to Classification of Bacteria}. Proceedings of the International ICSC Congress on Intelligent Systems and Applications, 2000.
9. Mazzocchi, M.: Time patterns in UK demand for alcohol and tobacco: an application of the EM algorithm. Computational Statistics and~Data Analysis, 50 (9), 2006, p. 2191÷2205.
10. Gooya A., Biros G., Davatzikos C.: An EM Algorithm for Brain Tumor Image Registration: A Tumor Growth Modeling Based Approach. IEEE Computer Society Conference on~Computer Vision and~Pattern Recognition, 2010.

11. Miller M. I., Chen S. C., Kuefler D. A., Davignon D. A.: Maximum Likelihood and the EM Algorithm for 2D NMR Spectroscopy. Journal of Magnetic Resonance, Volume 104, Issue 3, 1993, p. 247÷257.
12. Dijkstra M, Roelofsen H, Vonk RJ, Jansen R. C.: Peak quantification in surface-enhanced laser desorption/ionization by using mixture models. Proteomics 2006; 6(19):5106-16.
13. Noy K, Fasulo D.: Improved model-based, platform-independent feature extraction for mass spectrometry. Bioinformatics 2007; 23(19):2528-35.
14. Davis L.: Handbook of Genetic Algorithms. New York, Van Nostrand Reinhold, 1991.
15. McLachlan, G. J., Peel D., Basford K. E., Adams P.: The EMMIX software for the fitting of mixtures of normal and t-components. Journal of Statistical Software, 1999, 4(2).
16. Biernacki C., Celeux G., Govaert G., Langrognet F.: Model-based cluster and discriminant analysis with the MIXMOD software. Computational Statistics & Data Analysis, 2006, Volume 51, Issue 2, p. 587÷600.
17. Richardson S., Green P. J.: On Bayesian analysis of mixtures with an unknown number of components (with discussion). Journal of the Royal Statistical Society, 1997, B 59, p. 731÷792.
18. Wu J. C. F.: On the Convergence Properties of the EM Algorithm. The Annals of Statistics, 1983, Vol. 11, No. 1, p. 95÷103
19. Ma J., Xu L.: Asymptotic convergence properties of the EM algorithm with respect to the overlap in the mixture, Neurocomputing 68, 2005, p. 105÷129.
20. Xu L., Jordan M.: On Convergence Properties of the EM Algorithm for Gaussian Mixtures. Neural Computation, vol. 8, 1996, p. 129÷151.
21. Kiefer J., Wolfowitz J.: Consistency of the Maximum Likelihood Estimator in the Presence of Infinitely Many Incidental Parameters, Ann. Math. Statist. 1956, Volume 27, Number 4, p. 887÷906.
22. Peters B. C.,Walker H. F.: An iterative procedure for obtaining maximum likelihood estimators of the parameters for a mixture of normal distributions. SIAM Journal on Applied Mathematics 35, 1978, p. 362÷378.
23. Karlis D., Xekalaki E.: Choosing initial values for the EM algorithm for finite mixtures. Computational Statistics and Data Analysis~41, 2003, p. 577÷590.
24. Biernacki C., Celeux G., Govaert G.: Choosing starting values for the EM algorithm for getting the highest likelihood in multivariate Gaussian mixture models. Computational Statistics & Data Analysis, 2003, vol. 41, p. 561÷575.

25. Biernacki C.: Initializing EM using the properties of its trajectories in Gaussian mixtures. Statistics and Computing, 2004, vol. 14, p. 267÷279.
26. Yang M. S., Lai C. Y., Lin C. Y.: A robust EM clustering algorithm for Gaussian mixture models, Pattern Recognition, vol. 45, 2012, p. 3950÷3961.
27. Fayyad U. M., Reina C., Bradley P. S.: Initialization of iterative refinement clustering algorithms. Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining (KDD-98), 1998, p. 194÷198.
28. Ishikawa Y., Nakano R.: Obtaining EM Initial Points by Using the Primitive Initial Point and Subsampling Strategy, Proceedings of International Joint Conference on Neural Networks, 2007, p. 1115÷1120.
29. Pereira J. R. G., Cabral C. R. B., Marques L. A., da Costa J. M. J.: An Empirical Comparison of EM Initialization Methods and Model Choice Criteria for Mixtures of Skew-Normal Distributions, Technical Report, 10.01.2012, (http://www.ime.unicamp.br-/sinape/sites/default/files/Pereira_Cabral_Marques_Costa_0.pdf).
30. Bessadok A., Hansen P., Rebai A.: EM algorithm and Variable Neighborhood Search for fitting Finite Mixture Model parameters, Proceedings of the International Multiconference on Computer Science and Information Technology, 2009, p. 725÷733.
31. Meila M., Heckerman D.: An Experimental Comparison of Several Clustering and Initialization Methods, Microsoft Research Technical report MSR-TR-98-06, UAI 1998 and Machine Learning Journal, 2000, vol. 42, p. 9÷42.
32. Maitra R.: Initializing partition-optimization algorithms. IEEE/ACM Transactions on Computational Biology and Bioinformatics 6, 2009, p. 144÷157.
33. Reddy C. K., Rajaratnam B.: Learning mixture models via component-wise parameter smoothing, Computational Statistics and Data Analysis, vol. 54, 2010, p. 732÷749.
34. Pernkopf F., Bouchaffra D.: Genetic-Based EM Algorithm for Learning Gaussian Mixture Models, IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 27, 2005, p. 1344÷1348.
35. Li L., Ma J.: A BYY Split-and-Merge EM Algorithm for Gaussian Mixture Learning. F. Sun et al. (Eds.): ISNN 2008, Part I, LNCS 5263, 2008, p. 600÷609.
36. Yao W.: A profile likelihood method for normal mixture with unequal variance Journal of Statistical Planning and Inference, vol. 140, 2010, p. 2089÷2098.
37. Hathaway R. J.: A constrained formulation of maximum-likelihood estimation for normal mixture distributions. Annals of Statistics vol. 13, 1985, p. 795÷800.
38. Hathaway R. J.: A constrained EM algorithm for univariate mixtures Journal of Statistical Computation and Simulation vol. 23, 1986, p. 211÷230.

39. Ingrassia S.: A likelihood-based constrained algorithm for multivariate normal mixture models. Statistical Methods & Applications, vol. 13, 2004, p. 151÷590.
40. Fisher W. D.: On Grouping for Maximum Homogeneity. Journal of the American Statistical Association, Vol. 53, No. 284. 1958, p. 789÷798.
41. Engelman L., Hartigan J. A.: Percentage points of a test for cluster. J Am Stat Assoc vol. 64, 1969, p. 1647÷1648.
42. Jensen R. E.: A Dynamic Programming Algorithm for Cluster Analysis, Operations Research, vol. 17, 1969, p. 1034÷1057.

Wpłynęło do Redakcji 5 grudnia 2012 r.

Omówienie

Mieszaniny rozkładów prawdopodobieństwa mają szerokie zastosowanie w statystycznej analizie danych, między innymi w zakresie analizy struktury populacji zwierząt, rozpoznawania wpływów różnych czynników na rynki finansowe, rozdzielania źródeł zmienności w danych eksperymentalnych czy też analizy wyników obrazowania medycznego. Szczególne znacznie mają skończone mieszaniny rozkładów normalnych, w których z góry znana liczba składowych rozkładów prawdopodobieństwa Gaussa (2) tworzy nową wypadkową funkcję gęstości prawdopodobieństwa (1) z pewnym zadanym udziałem każdej ze składowych.

Czołowym problemem pojawiającym się w praktycznych zastosowaniach mieszanin rozkładów normalnych jest estymacja parametrów rozkładów na podstawie danych eksperymentalnych. Szeroko stosowanym i opisanym w literaturze rozwiązaniem tego problemu jest algorytm EM, który w sposób iteracyjny na zmianę wyznacza szacunkowe prawdopodobieństwo przynależności poszczególnych obserwacji do składowych mieszaniny oraz maksymalizuje wiarygodność poprzez dobór wag i parametrów poszczególnych składowych mieszaniny. Idea algorytmu EM zapewnia wzrost wiarygodności obserwacji w każdej kolejnej iteracji.

Zbieżność algorytmu EM oraz jakość uzyskanego rozwiązania są zależne od początkowej estymacji parametrów mieszaniny (warunku początkowego). Według wiedzy Autorów, w literaturze kwestia metod doboru warunku początkowego nie jest dostatecznie opisana. Publikacje jej dotyczące ograniczają się do jedno- lub dwuwymiarowych mieszanin od 2 do 4 składowych, przy czym często są to mieszaniny o równych wartościach wariancji wszystkich składowych.

W powyższej publikacji Autorzy rozszerzają wiedzę na temat doboru warunku początkowego algorytmu EM przez analizę mieszanin jednowymiarowych (i) o większej liczbie skła-

dowych (od 5 do 10) oraz (ii) o różnych wartościach wariancji składowych (heteroscedastyczne). Zostały zaprezentowane wyniki badań w zależności od (i) liczby składowych, (ii) zróżnicowania wariancji składowych, (iii) zróżnicowania przekrycia poszczególnych składowych.

Addresses

Radosław SOKÓŁ: Silesian University of Technology,Institute of Electrical Engineering and Informatics, ul. Akademicka 10, 44-100 Gliwice, Poland, radoslaw.sokol@polsl.pl.
Andrzej POLAŃSKI: Silesian University of Technology, Institute of Informatics, Akademicka 16, 44-100 Gliwice, Poland, andrzej.polanski@polsl.pl; Polish-Japanese Institute of Information Technology, ul. Legionów 2, 41-900 Bytom, Poland.