Annales UMCS Informatica AI IX, 1 (2009)
123–134; DOI: 10.2478/v10065-009-0009-9

Annales UMCS
Informatica
Lublin-Polonia
Sectio AI
http://www.annales.umcs.lublin.pl/

Numerical analysis of EM estimation of mixture model parameters

Małgorzata Plechawska[1*], Łukasz Wójcik[1], Andrzej Polański[2]

[1] *Lublin University of Technology, Lublin, Poland*
[2] *Institute of Computers Science, Silesian University of Technology, Gliwice, Poland*

Abstract

Optimisation of distribution parameters is a very common problem. There are many sorts of distributions which can be used to model environment processes, biological functions or graphical data. However, it is common that parameters of those distribution may be, partially or completely unknown. Mixture models composed of a few distributions are easier to solve. In such a case simple estimation methods may be used to obtain results. Usually models are composed of several distributions. Those distributions may be of the same or different type. Such models are called mixture models. Finding their parameters may be complicated. Usually in such cases iterative methods need to be used. The paper gives a brief survey of algorithms designed for solving mixtures of distributions and problems connected with their usage.

One of the most common method used to obtain mixture model parameters is Expectation-Maximization (EM) algorithm. EM is the iterative algorithm performing maximum likelihood estimation. The authors present the results of adjusting the Gaussian mixture models to the data. It is done with the usage of EM algorithm. The article gives advantages and disadvantages of EM algorithm. Improvements of EM applied in the case of large data are also presented. They help increase efficiency and decrease operation time of the algorithm. Another considered issue is the problem of optimal input parameters selection and its influence on the adjustment results. The authors also present algorithm performance observations.

[*] *E-mail address:* gosiap@pluton.pol.lublin.pl

1. Introduction

Mixture models [**1**] are popular methods of data-sets presentation and analysis. The most common application of mixture models are natural phenomena, biological processes, graphical data, damaged and incomplete data [**2, 3**], classification problems. Single distribution usually represents one process. However, if a sequence of processes occurs, several distributions may be combined. Genes reactions on tissues damage exemplify this. Activation of one set of genes makes the other react. Genes which do not take part in the process have small weight and their rate of activity does not change.
The most popular probability distribution is a Gaussian one. According to the Central Limit Theorem if a number of sample distribution is huge and its variance is finite, distribution statistics may be approximated by the Gaussian function [**4**].

The single Gaussian distribution (Fig. 1) has two parameters: mean and standard deviation. The distribution is given by the formula:

$$f_k(x_n, \mu_k, \sigma_k) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{(x_n - \mu_k)^2}{2\sigma_k^2}\right].$$

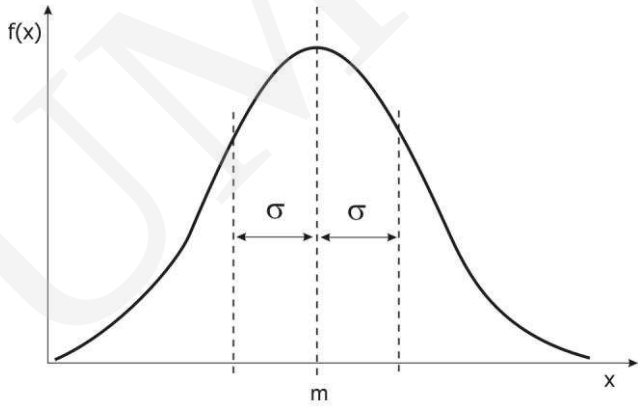


Fig. 1. Gaussian distribution [**5**]

The mixture model, consisting of several Gaussian components is represented by the formula:

$$f^{mix}(x, \alpha_1, \ldots, \alpha_K, p_1, \ldots, p_K) = \sum_{k=1}^{K} \alpha_k f_k(x, p_k)$$

where

$$\alpha_1, \ldots, \alpha_K, p_1, \ldots, p_K \text{ – mixture parameters}$$

$$\alpha_1, \ldots, \alpha_K \text{ – weights } \sum_{k=1}^{K} \alpha_k = 1$$

$$f_k(x, p_k) \text{ – density distribution function.}$$

One can notice that apart from the parameters of Gaussian components, the mixture model must be also described by weights. Every Gaussian component in the model has its own weight, which determines height and importance of this single distribution.

Fig. 2 illustrates the influence of parameter values on simple, composed of two Gaussians mixture model. The stronger line identifies the envelope, the thinner - single Gaussian. Small difference between the means of components can cause missing or merging Gaussians (Fig. 2c). Weights have also strong influence on the model characteristics. Gaussians with small weights are harder to model and solve (Fig. 2b). Small weights in combination with similar means may lead to more complicated model formation (Fig. 2d).

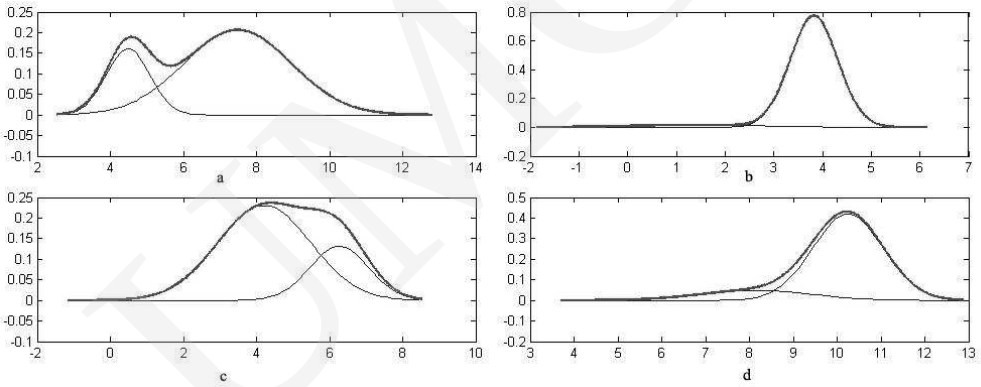


Fig. 2. Simple examples of Gaussian mixtures

2. Mixture model parameters

One of the most problematic issues concerned with mixture models is estimation of their parameters. It is a hard task due to the number and properties of the estimated parameters. Fig. 3 illustrates the example of typical model.

There are several optimization methods which can be used to solve the problem of unknown parameters. The best of them are based on iterative algorithms. Such methods are Newton, quasi-Newton or gradient ones [**6**]. However, those algorithms need partial derivatives vector, which makes them hard to use due

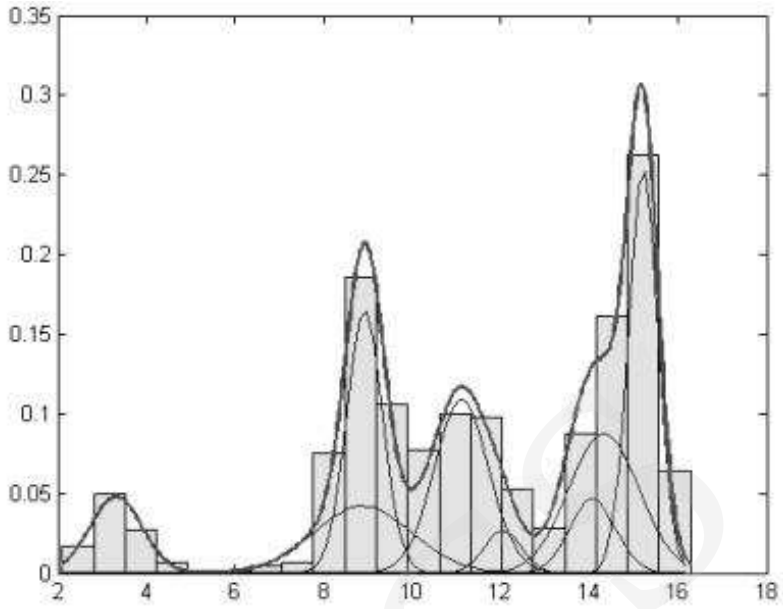


Fig. 3. Gaussian mixture model

to complicated calculations. However, they may be used as an assistance for other algorithms.

The expectation-Maximization algorithm [**7**] is the most common method of computing mixture model parameters. It is an iterative method, composed of two main steps: Expectation (E) and Maximization (M). The standard version of EM algorithm allows for obtaining initial values from randomization. The first step, E, is responsible for calculation of probability value [**8**]:

$$p\left(k|x_n, p^{old}\right) = \frac{\alpha_k^{old} f_k\left(x_n, p^{old}\right)}{\sum_{k=1}^{K} \alpha_k^{old} f_k\left(x_n, p^{old}\right)}$$

where

$p\left(k|x_n, p^{old}\right)$ – probability, that sample x_n belongs to k^{th} component of mixture

p^{old} – set of input parameters

α_k^{old} – weight of k^{th} component

$f_k\left(x_n, p^{old}\right)$ – density distribution function.

The second step, M, includes calculation of new parameter values. For the Gaussian mixtures this step is given by formulas [**8**]:

$$\mu_k^{new} = \frac{\sum_{n=1}^{N} x_n p\left(k|x_n, p_{old}\right)}{\sum_{n=1}^{N} p\left(k|x_n, p_{old}\right)}, \quad k = 1, 2, \ldots, K$$

$$(\sigma_k^{new})^2 = \frac{\sum_{n=1}^{N} \left(x_n - \mu_k^{new}\right)^2 p\left(k|x_n, p_{old}\right)}{\sum_{n=1}^{N} p\left(k|x_n, p_{old}\right)}, \quad k = 1, 2, \ldots, K$$

$$\alpha_k^{new} = \frac{\sum_{n=1}^{N} p\left(k|x_n, p_{old}\right)}{N}, \quad k = 1, 2, \ldots, K.$$

The EM algorithm uses the likelihood function for finding the best possible results. The likelihood function $f(x_n, p)$ is a common way of evaluation used in the estimation of probability distribution parameters. The function $f(x_n, p)$ describes the likelihood of x_n observation. It is a good idea to use Maximum Likelihood [**1**] (ML) principle in the process of parameter values calculation. It states, that the best parameters estimation is the one which is most probable. The most probable set of parameters is the one, which is computed from maximizing of the likelihood function [**9**]. The ML principle is given by:

$$L(p, x) = L(p) = f\left(x_1, x_2, \ldots, x_N, p\right) = \prod_{n=1}^{N} f(x_n, p)$$

$$l\left(x_1, x_2, \ldots, x_N, p\right) = \ln\left[L\left(x_1, x_2, \ldots, x_N, p\right)\right] = \sum_{n=1}^{N} \ln\left[f(x_n, p)\right]$$

$$\widehat{p} = \arg\max \prod_{n=1}^{N} f(x_n, p).$$

To make calculations more efficient, the log-likelihood function [**8**] is used. This allows summation instead of multiplication and it does not change the results because monotonicity of $l\left(x_1, x_2, \ldots . x_N, p\right)$ leads to the same location of maximum as $L(p, x)$ does. The Gaussian interpretation of ML principle is as follows:

$$l\left(x_1, x_2, \ldots, x_N, \mu, \sigma\right) = \sum_{n=1}^{N} \left[-\frac{1}{2}\ln(2\pi) - \ln\sigma - \frac{\left(x_n - \mu\right)^2}{2\sigma^2}\right],$$

$$\widehat{\mu} = \frac{1}{N}\sum_{n=1}^{N} x_n, \qquad \sigma^2 = \frac{1}{N}\sum_{n=1}^{N}\left(x_n - \widehat{\mu}\right)^2.$$

Usage of ML principle [**10**] gives a certainty of stability. The ML value is always ascending or stable - it never descends (Fig. 4). It guarantees that there will be no deterioration during algorithm work regardless of any circumstances.

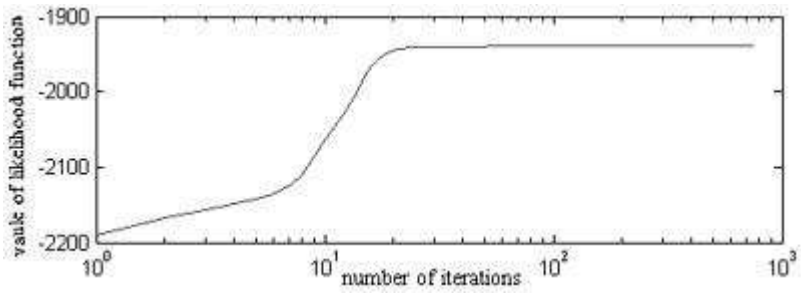


Fig. 4. Log likelihood values

3. Characteristics of EM algorithm

Fig. 5 presents the models composed of the parameters obtained from the EM algorithm. The figures on the left side show an artificially generated model, with 8 - 5 components and the corresponding sample size of 1000. The figures on the right compare those models with those built on the EM results. For each model on the right two corresponding runs of EM are shown on the left. The dotted line illustrates the results of envelopes subtraction.

As Fig. 5 shows, parameters estimation may be a problematic issue. The examples show that some Gaussians are more difficult to solve than others. Close nearness of means, in conjunction with similar standard deviation values may result in merging Gaussians. In such a case flat, stretched Gaussian occurs to preserve a proper number of components. In other cases, instead of flat Gaussian, one can find low components with small standard deviations. This may result in unwanted peaks. Peaks may be also a result of mistaken estimation of small-weighted component. The examples show the principle: the smaller weight of Gaussian, the worse match is found. This phenomenon illustrates algorithm characteristic - it is easier to solve high-weighted components. Good estimators of components are found quickly in the first several algorithm iterations. In most cases the researcher is interested only in high-weighted component estimation. Low-weighted components need more time to estimate. However, too high accuracy results in lengthened calculation time and excessive concentration on the low-weighted components. The next interesting issue is the shape of envelope subtraction lines, shown in Fig. 5. In all cases these lines have similar, sinus-shaped look in the areas corresponding to the overlapping components. This property can be used in error prediction simulations.

Estimation mistakes can be also a result of errors in Maximum Likelihood principle operation. The ML principle always tries to find global maximum of likelihood function but sometimes it sticks to a local one [**11**]. Good illustration of this process is presented in Fig. 6a. The charts show the dependence of means

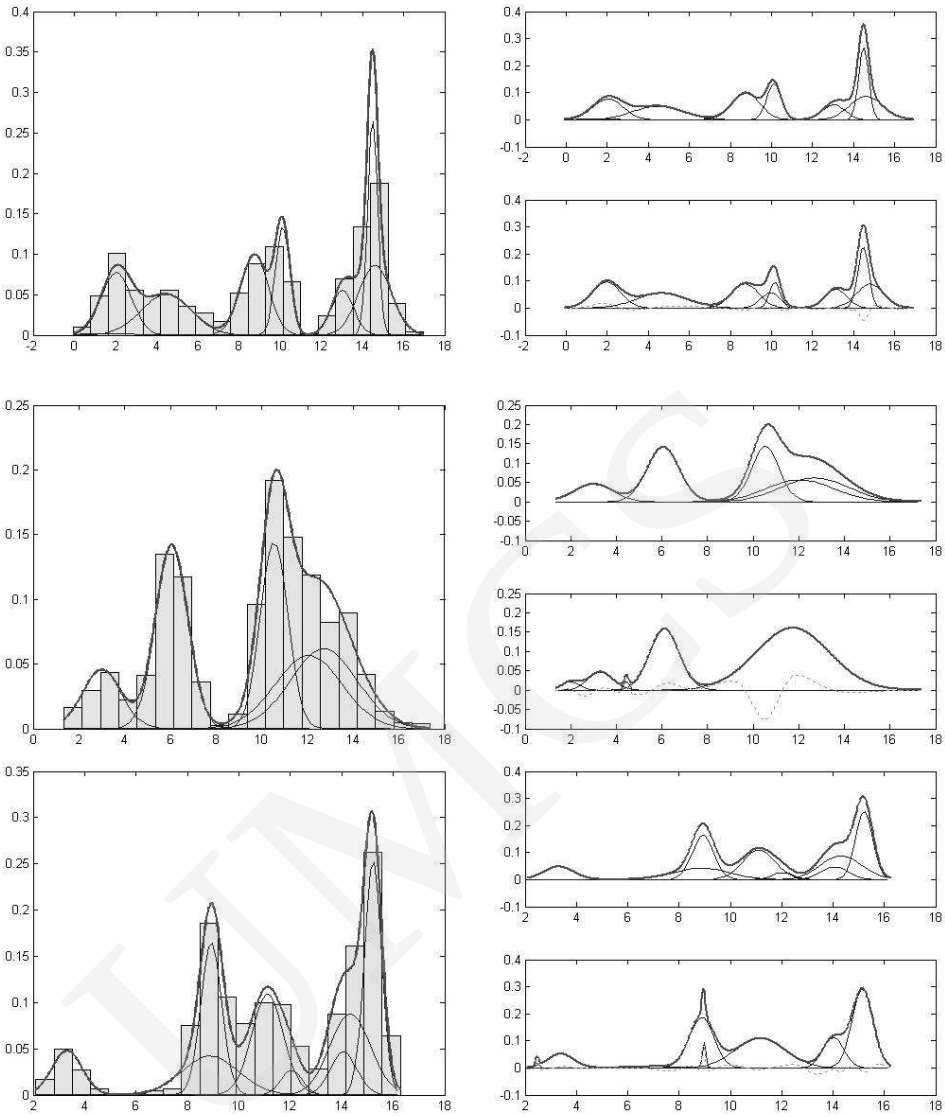


Fig. 5. Parameters estimation results

on standard deviations with consideration of weights. Weights are represented by the size of circles on the charts. This way of parameters presentation gives opportunity to check the quality of the obtained results and it makes easy to match the true model and the estimation parameters. This matching is important because the EM algorithm does not return estimated parameters in the same order as they are in the model. This enables matching and measuring the distance between the true and estimated models, especially when the model is composed of many elements. the multiple repetition method can be used to

handle the local maximum problem. It offers to use many ML calculations (for example 50 or 100) instead of one and choose estimation with the highest value of ML. Another improvement is to repeat the whole above process many times and each time draw the best parameters (those corresponding to the highest likelihood) in the chart. As a result (Fig. 6b) the obtained parameters should oscillate between on the corresponding parameters of the true model.

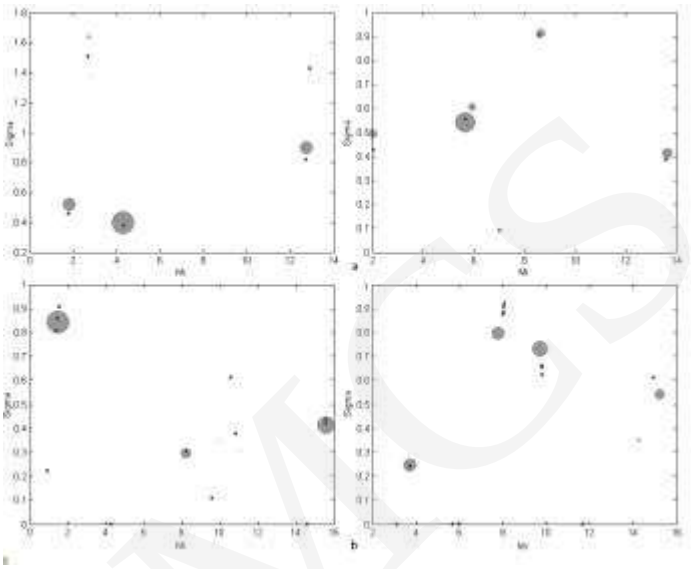
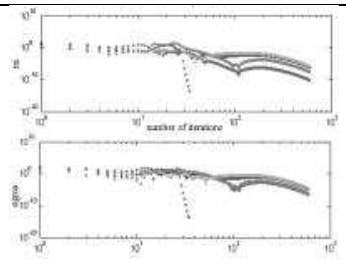
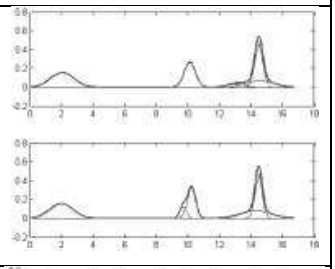
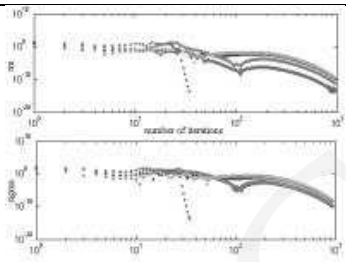
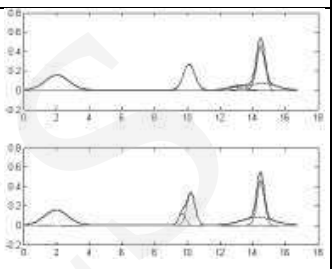
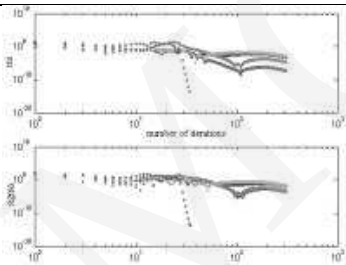
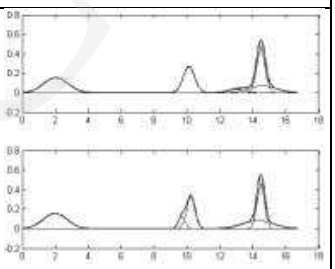
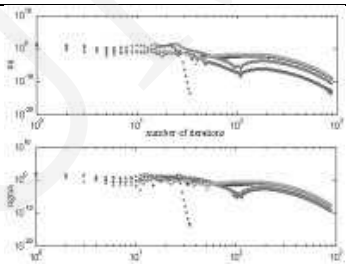
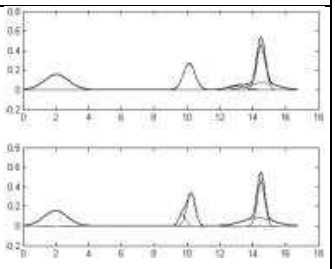
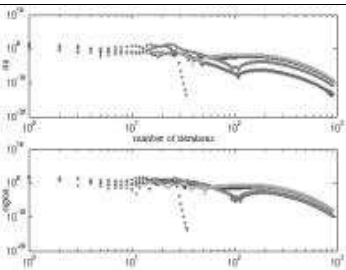
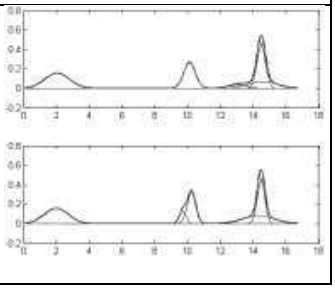


Fig. 6. Mean and standard deviation dependency

Another issue is the influence of the stop criterion choice on the performance of the algorithm. The most common criterion is that based on the likelihood function values. If the difference between two c consecutive calculated likelihood function values is smaller than the defined accuracy ε, $\varepsilon > 0$, the algorithm terminates. However, there are also other stopping conditions consisting in the rate of the estimators change. This rate can be obtained with different measures and combinations of those measures. Table 1 presents the results of comparison of: maximum change of single parameter values (relative and absolute changes), Euclidean and chi2 distance. The table contains the number of iterations needed to gain the results and charts illustrating the shape of parameters and comparing the obtained parameters.

The results Table 1 confirm, that the type of stop criterion does not have substantial influence on parameters estimation. All estimators have very similar values. The only difference is in the estimation time. The values in brackets represent the number of iterations taken from a few other calculation attempts.

Table 1. Comparison of EM distance methods

Measure type	Amount of iterations	Taking shape of parameters	Comparison obtained parameters
Standard likelihood function	594 (1866 594 2042 727)		
Absolute changes of parameters	948 (3572 951 4772 989)		
Relative changes of parameters	309 (531 316 296 725)		
Euclidean distance	885 (3252 856 4431 933)		
Chi2 distance	913 (3417 902 4269 890)		

According to those data, the best results are given by simple relative changes of parameters.

One needs to remember that the defined accuracy has great influence on algorithm efficiency and speed. The same accuracy may be optimal for one distance type whereas for the others may not be good enough. This may lead to differences in number of iterations or quality of estimations. Every distance method needs another accuracy due to different values of distance parameters. The accuracy depends on a number of model components and points. There is a need to estimate the accuracy by empirical testing.

4. Methods of EM improvement

Initial convergence of EM is satisfactory because estimation gets to the vicinity of the limit values very fast. But after that, the progress decreases and the algorithm approaches the solution quite arduously (Fig. 7).

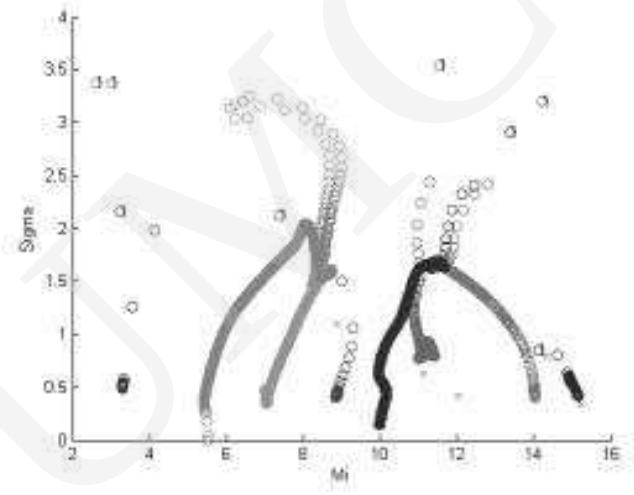


Fig. 7. Convergence of EM algorithm

To decrease the computational time analytical work should be done, which, unfortunately, leads to increasing the complexity. There are many variants of the EM algorithm. One of them is the incremental version of EM algorithm (IEM). This implementation is based on dividing the observed data into equal B blocks. The procedure of IEM takes the E-step for only one block of the observed data at a time and next the B-step is taken. A simple "scan" of algorithm consist of B partial E-steps and B M-steps. As a result new information is collected faster. In simulation of McLachlan and Ng performance of IEM

[**12**] the algorithm was tested against the simple EM algorithm. A sample set $n = 256 \times 256$ was generated (Table 2).

Table 2. [**12**] Performance of IEM and EM algorithm

Algorithm	CPU Times Overall (E-stop, M-step)	No. of Scans
Standard EM	601 (4.76, 1.07)	101
Incremental EM		
B=4	456 (4.89, 1.26)	72
B=8	427 (4.89, 1.26)	67
B=16	414 (4.89, 1.26)	65
B=32	408 (4.90, 1.25)	64
B=64	405 (4.90, 1.28)	63
B=128	407 (4.90, 1.28)	63
B=256	411 (4.94, 1.32)	63
B= 65536	2352 (28.40, 6.87)	63

Only in the case where the number of blocks was established to a size of the data set the IEM was slow. All other simulations show that EM has slow convergence and incremental implementation, IEM, is faster.

Another variant of algorithm is Lazy EM [**12**]. The main idea is to specify a threshold for selecting subsets of the data upon which E-step and M-step will be performed. In other words, the method assumes that for each iteration not all data is of equal significance.

The third method used for accelerating the EM algorithm is sparse EM. In E-step some posterior probabilities are often close to zero. The sparse method selects only relative probabilities of a given data point. This algorithm can be combined with the incremental version by performing partial E-step and sparse E-step.

5. Conclusions

EM is one of the best algorithms of mixture parameters estimation. It can be also used in grouping and clustering tasks. It is a stable method, giving good results in the case of huge amount of data processing. Many improvements of EM have been found, which makes EM more efficient. They are helpful in acceleration or dealing with huge data-sets. However, EM is not free of disadvantages and difficulties. It is very sensitive to the initial values - improper values may lengthen the time of work or cause a local maximum problem. Another issue is slow convergence and high complexity, especially in the M step. There is also a need of multiple repetitions, which has additional influence on working time.

Acknowledgement

The author is thankful to Prof Joanna Polańska for assistance in the preparation of the paper.

References

[1] Dempster A. P., Laird N. M. and Rubin D. B., *Maximum likelihood from in-complete data via the EM algorithm*, J. R. Stat. Soc. 39(1) (1977) 138.

[2] Nassar C., Soleymani M., *Joint sequence detection and phase estimation using the EM algorithm* Electrical and Computer Engineering, 1994, Conference Proceedings 1 (1994) 296.

[3] Miłosz M., *Data Mining as a Modern Method of Data Analysis*, News of the Kazan State University of Architecture and Engineering 1(9) (2008) 162.

[4] Watała C., *Biostatystyka - wykorzystanie metod statystycznych w pracy badawczej w naukach biomedycznych*, Bielsko-Biała 2002.

[5] Soliński B., Strona uniwersytecka - Statystyka http://www.zarz.agh.edu.pl/bsolinsk/statystyka.html.

[6] Grega W., Strona uniwersytecka - Metody Optymalizacji http://aq.ia.agh.edu.pl/Aquarium/Dydaktyk/Wyklady/MO/2005-06/Wyklad04.pdf

[7] Everitt B. S., Hand D. J., *Finite Mixture Distributions*, Chapman and Hall, New York 1981.

[8] Polański A., Kimmel M., *Bioinformatics*, Springer, Berlin Heidelberg 2006.

[9] Hand D., Mannila H., Smyth P., *Principles of data mining*, Massachusetts Institute of Technology 2001.

[10] Polański A., et al., *Application of the Gaussian mixture model to proteomic MALDI-ToF mass spectra*, Journal of Computational Biology, Gliwice, 2007.

[11] Ishikawa Y., Nakano R., *Landscape of a Likelihood Surface for a Gaussian Mixture and its use for the EM algorithm*. Proceedings of the International Joint Conference on Neural Networks, IJCNN 2006, part of the IEEE World Congress on Computational Intelligence, pp. 1434-1440. WCCI 2006, Vancouver, BC, Canada 2006.

[12] McLachlan G., Peel D., *Finite Mixture Models*, The University of Queensland 2000.