Clustering of Text Documents by Skewness Maximization

Ata Kabán and Mark Girolami
Department of Computing and Information Systems, University of Paisley
Paisley, PA1 2BE, Scotland

Abstract— **In this paper a variant of ICA is proposed as a computationally attractive alternative approach to text based document clustering. The EM parameter estimation of a modified mixture model produces an ICA approach to document clustering which explicitly maximises a contrast based on distribution asymmetry or *skewness*. This clustering is performed in a low dimensional eigen-space where the data exhibits specific geometric structure induced by the correlation of terms {words} in the high dimensional text representation. Searching for maximally skew projections is an important difference from other approaches to document clustering, which assume symmetrical latent distributions. The relation to mixture modeling has the advantage that the outputs of the resulting algorithm can be interpreted in probabilistic terms. Experimental results are presented from the 20 Newsgroups corpus.**

I. Introduction

The literature has often commented on the abilities of Independent Component Analysis (ICA) for finding meaningful directions in data and applications to Exploratory Projection Pursuit and different data mining problems have been discussed [6], [13], [14]. Despite the well known fact that the type of nonlinearity, or equivalently the objective or contrast function used in ICA makes the algorithm seek different types of structure in the data, in most of the applications the variant of ICA used was arbitrarily chosen. Indeed, there are few such data exploration applications where the latent probability distribution can be derived from the problem, and there are fewer applications which make use of this information.

In this paper we consider the problem of clustering a specific type of data and incorporate our knowledge of the data representation in deriving an appropriate ICA method. The specific data we are considering is vector space representations of text based documents. Previous applications of ICA to text retrieval [9] and clustering [8] indicates this as a promising area of investigation, however the theoretical reasoning or motivation for the use of ICA as a document processing tool has been somewhat lacking. It has been left unexplained why and how ICA may be effective in text mining. On the other hand, Latent Semantic Analysis, which is based on Singular Value Decomposition has a long tradition in document retrieval research, and it has provided one solution for dealing with synonymy and polysemy [11].

We show in this paper that starting from the Principal Component Analysis (PCA) compressed vector space document representation and noting the geometric shape of the compressed data, two modifications to a Gaussian mixture model – in order to better fit the data shape – turns the Maximum Likelihood (ML) mixture-learning to an ICA which specifically maximizes the skewness of the projection. Recognizing this connection, there are several ways to improve the final clustering accuracy; using a larger vocabulary or post-processing the results by EM mixture modeling initialised by the class posteriors provided by the proposed ICA variant. This latter possibility avoids the problem of initialization of EM-based methods which is exacerbated when the data is high dimensional and sparse [2]. It is also worth noting that the resultant ICA algorithm has very fast convergence (3-8 steps), and is computationally simple. In this work it is assumed that the number of clusters or *topics* is known.

In Section 2 we briefly describe the data representation, Section 3 presents and motivates the algorithm, Section 4 gives a discussion, Section 5 presents the experimental results and finally we end with conclusions and future directions of investigation.

II. Data representation and preprocessing

Throughout this paper we will consider high dimensional data vectors, each representing a text document. Each vector is of the same dimensionality as the size of the considered vocabulary. The representation can be binary – 1 indicates the presence- whereas 0 indicates the absence of a word – or frequency count based – each element is the frequency of occurrence of a word in a document. The algorithm which will be derived is not limited to either of these representations, however we obtained better scaled results using the binary coding, as it avoids serious outliers. Dropping the term frequency information does not necessarily imply a loss of useful information, as the number of occurrences of a word in a document may be non-indicative in many situations.

A number of studies have shown that little is to be gained when sophisticated computational linguistics models are used over simple statistical representations. It is also notable that while linear, supervised classification techniques have been reported to achieve high accuracy for document classification [1], theoretically well founded EM based unsupervised clustering methods are somewhat lacking due to the many local optima in the likelihood surface and the high dependence on parameter intitiallization [2]. That is, linear separability of the classes is more probabable due to the sparse nature of the data, however it also provides many local maxima of the likelihood which is a problem in EM mixture modeling.

Let us denote by $\mathbf{D} = (d_{tn})_{t=1..T,n=1..N}$ the observed data of a term by document matrix of the considered document corpus, where T is the dimensionality of terms' space and N is the number of documents in the corpus. The vector $\mathbf{d}_n$ will refer to one column of this matrix.

We assume that the observations are just a sparse, noisy expansion of a dense, K-dimensional latent topic or concepts space. The proposed method has two steps, firstly, based on the previously reported work in the domain, we adopt the space spanned by the first K eigenvectors as the space which the topic-concepts live in, and in the second stage we are looking to identify the clusters in this space. Therefore we project the data onto the first K unit norm eigenvectors, where K is the number of classes we are looking for.

The projected data will be denoted by $\mathbf{X}$

$$\mathbf{X} = \mathbf{U}^T\mathbf{D} \tag{1}$$

where $\mathbf{U}$ contains the first K eigenvectors. These can be efficiently computed for example using the Lanczos algorithm [5] for Singular Value Decomposition of sparse matrices.

Choosing the unit norm for the new basis is just a matter of scaling, and there is no specific requirement to set the variances along the axis to one in this application. Naturally, after compression the classes will generally have an overlapping area, but if sufficient initial dimensions are chosen, than we may hope the overlap to be less severe. The algorithm developed in the sequel will not be able to separate overlapping documents, but separation of those documents falling outside the overlapping area will not be greatly influenced. The PCA compression, besides being optimal in the least squares sense and giving a dense representation of the data in a much smaller dimension, also has the advantage of removing a great amount of noise [10], and on this is based the success of LSA-based retrieval.

III. From class mixture modeling by EM to maximizing the skewness by nonlinear PCA

Let us start by assuming a class mixture model on the compressed data.

$$p(\mathbf{x}_n|\mathbf{w}_k) = \sum_{k=1}^{K} P(k)p(\mathbf{x}_n|k) \tag{2}$$

The data now being continuous, we will start formally from modeling the individual class probabilities as isotropic Gaussians

$$p(\mathbf{x}_n|k, \mathbf{w}_k) \propto exp(-\frac{1}{2}(\mathbf{x}_n - \mathbf{w}_k)^2) \tag{3}$$

where $\mathbf{w}_k$ is the expectation parameter (and also the natural parameter) of the Gaussian, i.e. the mean vector of class k. The instances being independent of each other, the data log-likelihood is

$$logp(\mathbf{x}|\mathbf{W}) = \sum_{n=1}^{N} log(\sum_{k=1}^{K} P(k)p(\mathbf{x}_n|k, \mathbf{w}_k)) \tag{4}$$

For the Expectation Maximization (EM) procedure, it is sufficient to maximize the relative log-likelihood, which in this case is

$$Q = -\sum_{n=1}^{N}\sum_{k=1}^{K} r_{kn}\frac{1}{2}(\mathbf{x}_n - \mathbf{w}_k)^2 + const \tag{5}$$

where r_{kn} is the class posterior probability, i.e.

$$r_{kn} = p(k|\mathbf{x}_n, \mathbf{w}_k) = \frac{P(k)exp[-\frac{1}{2}(\mathbf{x}_n - \mathbf{w}_k)^2]}{\sum_{k'=1}^{K} P(k')exp[-\frac{1}{2}(\mathbf{x}_n - \mathbf{w}_{k'})^2]} \tag{6}$$

which is computed in the E step. The M step performs a Maximum Likelihood (ML) update of the parameters, computed by setting $\partial Q/\partial w_k$, and $\partial(Q + \lambda(\sum_{k=1}^{K} p(k) - 1))/\partial p(k)$ to zero, which gives the familiar result

$$\mathbf{w}_k = \frac{\sum_n r_{kn}\mathbf{x}_n}{\sum_n r_{kn}} \tag{7}$$

and optionally $P(k) = \sum_n p(k|\mathbf{x}_n)/N$.

Mixture modeling has the advantage of being theoretically well founded, however the assumed density models often are unlikely to hold in practice – a fact that makes it unsuitable in some situations.

It has been observed that the geometrical aspect of the PCA compressed document data has the form of a brush which starts from the origin (see Fig. 1), and the angle between the mean directions is less than (or equal to) 90 degrees. This is a consequence of the specific correlational and distributional properties of text data, which is the subject of further investigation. From the perspective of mixture modeling this shape is a disadvantage, because of the overlapping region after the compression and because the Gaussian modeling assumption of the classes is a poor approximation in this situation – there can be gaps or dense sub-regions inside a cluster, which causes local maxima of the likelihood [3]. Mixture modeling also suffers from the

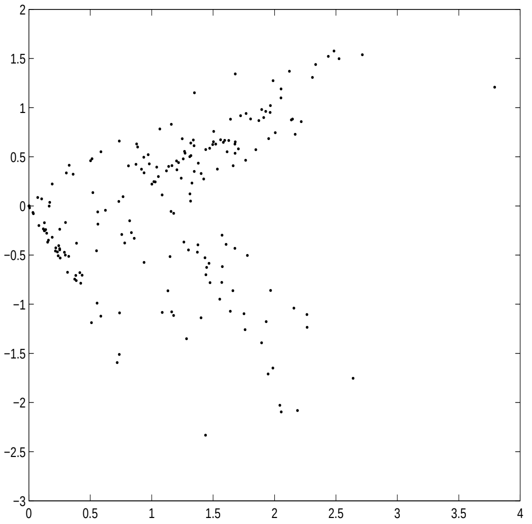

Fig. 1. PCA compressed 200 documents belonging to two classes

initialization problem: after a random initialization, and if the classes are not very distinct in space, there is no guarantee that the same class will not be found several times, while other classes can remain undiscovered.

We will introduce a restriction first: instead of searching for the coordinates of the individual class means we will search for the directions of the individual class means, starting from the origin, which is the common point of all classes. In other words, we fix the norm of the class mean vectors to a constant. This leads to a modified class likelihood, namely (3) modified as follows:

$$\begin{aligned} p(\mathbf{x}_n|k,\mathbf{w}_k) &\propto exp(-\frac{1}{2}(\mathbf{x}_n^2 - 2\mathbf{w}_k{}^T\mathbf{x}_n + \mathbf{w}_k{}^2) && (8) \\ &\propto exp(\mathbf{w}_k{}^T\mathbf{x}_n) && (9) \end{aligned}$$

as $\mathbf{x}^2$ is not parameter-dependent and $||\mathbf{w}_k|| = 1$. Further considering that $P(k) = 1/K$ for each k, which is a reasonable assumption in classification techniques, the class posteriors turn into a softmax transform of the projections:

$$r_{kn} = \frac{exp(\mathbf{w}_k{}^T\mathbf{x}_n)}{\sum_{k'=1}^{K} exp(\mathbf{w}_{k'}{}^T\mathbf{x}_n)} \qquad (10)$$

As we mentioned before, we will normalize the length of the mean vectors to unit norm after each update in the M step. Therefore we can drop the denominator of (7) in the M step, getting a nonlinear Hebbian rule.

$$\mathbf{w}_k = \sum_n r_{kn}\mathbf{x}_n \qquad (11)$$

The approximation of the class means by unit norm vectors will restrict the space of the search, reducing the possibility of getting stuck in a local maximum. However it doesn't solve the problem of possibly finding the same class in several components, depending on the initialization.

The second modification, will enforce the requirement that the mixture components have to be different, which is achieved simply by decorrelating the class mean directions. As these directions are typically not orthogonal, a quasi-orthogonal decorrelation [4] could be used. In the experiments reported we used just the standard orthogonal decorrellation involving matrix square roots. This keeps the computations simple in this context and will also form the link to nonlinear PCA.

In consequence, the resultant algorithm, in a batch formulation is as follows:

- random initialization of $\mathbf{w}_k$, for each $k = 1..K$

$$\mathbf{W} = \mathbf{W}/\sqrt{(\mathbf{W}^T\mathbf{W})^{-1}} \qquad (12)$$

Iterate until convergence:

- **E step**:

$$\begin{aligned} \mathbf{S} &= \mathbf{W}^T\mathbf{X} && (13) \\ \mathbf{R} &= softmax(\mathbf{S}) && (14) \end{aligned}$$

- **M step**:

$$\begin{aligned} \mathbf{W} &= \mathbf{X}\mathbf{R}^T && (15) \\ \mathbf{W} &= \mathbf{W}/\sqrt{(\mathbf{W}^T\mathbf{W})^{-1}} && (16) \end{aligned}$$

where the matrix $\mathbf{W}$ has $(\mathbf{w}_k)_{k=1..K}$ as columns, $\mathbf{S}$ is a matrix which has $(\mathbf{s}_k)_{k=1..K}$ in rows, $\mathbf{R} = (r_{kn})_{k=1..K,n=1..N}$, and the $softmx(.)$ acts columnwise.

Due to the softmax nonlinearity this is a hybrid between class mixture modeling and nonlinear PCA. The softmax is reminiscent of the class posterior probability computation via Bayes theorem. However it is interesting to notice that (due to decorrelation), in practice the normalization performed by the denominator of the softmax is independent of the other outputs, i.e. the nonlinear outputs will still sum to 1, but each nonlinear output r_{kn} can be expressed separately just in terms of the input vector $\mathbf{x}_n$ and the weight vector of the output $\mathbf{w}_k$, as will be seen shortly. This is because the angles between means are smaller than the orthogonal angle, and the orthogonalisation procedure can approximately keep the same angle between $\mathbf{w}_k$ and $(\mathbf{W}/\sqrt{(\mathbf{W}^T\mathbf{W})^{-1}})_k$ for all k. In consequence, the orthogonal mapping onto the columns of the decorrelated $\mathbf{W}$ excludes (or makes unlikely) the possibility of a projection $\mathbf{s}_k$ being greater when $\mathbf{x}$ is closer (in orthogonal distance) to another column of $\mathbf{W}$ than the k^{th} (Fig.2 gives an illustration in 2D). This is not the case in mixture modeling, where orthogonal projections onto columns of $\mathbf{W}$ (dotted line on Fig.2) have to be softmaxed for determining the coefficient in learning the mixture parameter $\mathbf{w}_k$. Projecting onto the decorrelated columns of $\mathbf{W}$ (continuous line) assures the soft competition, as is obvious from the figure, so even if we drop the normalizing denominator from the softmax, we would still have a competitive process. We can also mention that for comparison of the final outcomes (i.e. clustering the data) it doesn't matter whether the final projection of the data will be onto the direction of the class means'or onto the decorrelated class mean directions – which are in fact the ICA basis vectors in this context – because of the approximately constant angles between them. However, for keeping the relation to probabilistic mixture modeling it is interesting to follow at the same time the estimation process of the class means.

Now, we are interested in the analytical form of r_{kn}, to examine what kind of error function is induced by the modified mixture modeling algorithm. Considered now the columns of $\mathbf{W}$ after decorrelation, from (10) we have that for $\mathbf{x}_n$ lying on the axis k we have $r_{kn} = exp(s_{kn})/(exp(s_{kn}) + K - 1)$, and for $\mathbf{x}_n$ which lies on the middle-angle between all the mean vectors r_{kn} reduces to $1/K$. In general, the r_{kn} has the form

$$r_{kn} = \frac{exp(s_k)}{\sum_{k'=1}^{K} exp(l_{k'}s_k)} \qquad (17)$$

where $l_{k'} \in [0, 1]$, $l_k = 1$ and is dependent on the position of $\mathbf{x}_n$.

Inspecting this family of nonlinearities from the perspective of ICA, the negative of the objective functions associated to these functions for each of the K outputs have the

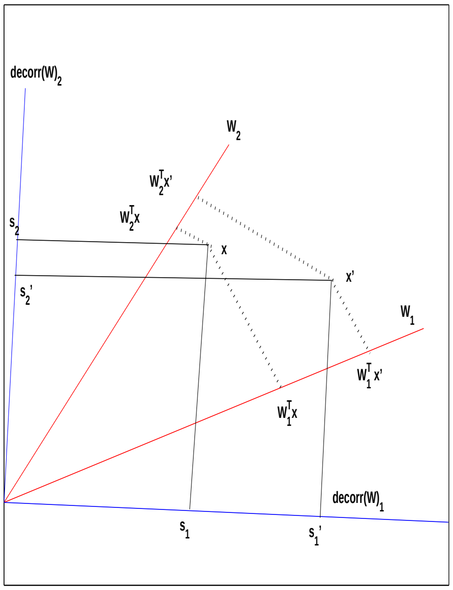

Fig. 2. The mapping mechanism of the modified mixture modeling algorithm, using orthogonal decorrelation. Orthogonal mapping onto the decorrelated class-mean directions (solid lines) – which become ICA basis vectors – instead of the orthogonal mapping onto the estimated class-means (dotted line) during the learning assures the competitive learning and therefore we can drop the softmax' normalization, i.e. output components are not required to depend on each other during learning.

general form

$$Err_k = -E[log[\sum_{k'=1}^{K} exp(l_{k'}s_k)]] \qquad (18)$$

where $l_k = 1$. In other words, starting from maximizing this objective function, the resulting nonlinear PCA algorithm is the same as the previously presented EM algorithm. The difference is just in interpretation: in this case the columns of the decorrelated $\mathbf{W}$ become basis vectors, and r_{kn} is the nonlinear output of the network.

What is interesting to notice is that (18) are all skew functions having similar shapes. Fig.3 shows the input-dependent negative objective function family for the 2D case, which penalize negative values. The negative values are the most penalized in the center (when $l_k = 1$ for all k) and less penalized near the axes of the decorrelated mean vectors (when $l_{k'} = 0 \forall k' \neq k, l_k = 1$).

Summarizing the resultant algorithm, again from the class modeling perspective, we can say that the algorithm still performs class modeling and it learns the class means in the columns of $\mathbf{W}$, because the update rule of $\mathbf{W}$ in the M step is the class-mean update of the mixture modeling algorithm. The responsibilities r_{kn} are computed in a slightly different manner than in standard Gaussian mixture modeling, namely by projecting the data onto the decorrelated mean vectors, which alone assures the competition between mixture components. Variants in computing the class responsibilities are known to stand for different distributional assumptions on the classes in class mixture modeling, whereas the mean update formula is the same for all variants.

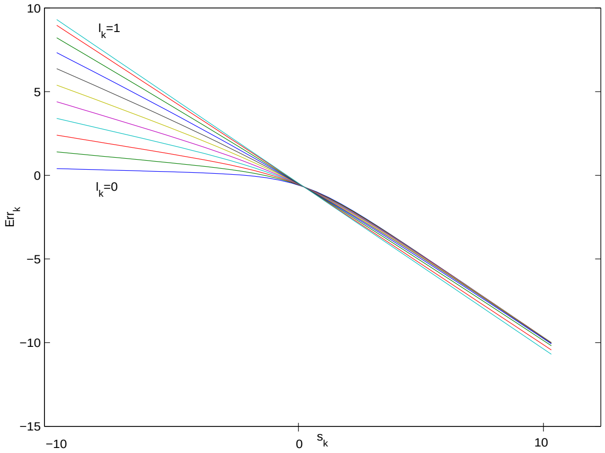

Fig. 3. The input-dependent error function family for the k^{th} output during the nonlinear PCA updating.

From another perspective, the same algorithm performs a form of nonlinear PCA learning which maximizes the skewness of the linear output and whose basis vectors are the decorrelated columns of $\mathbf{W}$.

IV. Discussion

Experiments have shown that any objective function that maximizes some skewness measure on the sources gives similar results, and it is not required in practice to have input-dependent nonlinearities. As a typical example, we can use the third order moment as an objective function, which is a measure of skewness.

$$Err_k = -E[s_k^3] \qquad (19)$$

In this case, (14) is replaced with

$$r_{kn} \propto {s_{kn}}^2 \qquad (20)$$

and the expectation is enclosed in the normalisation. Any other odd moment or combination of odd moments is also possible to use. Naturally, in this context the probabilistic interpretation does not hold, however in all our experiments the softmaxed linear outputs after the algorithm has converged were similar to those using the softmax nonlinearity inside the iterations. Moreover, (20) seemed to work for clustering applications in a broader set of conditions than those presented here, and this forms the subject of further investigations.

The skewness based objective function is not a surprising result if we have in mind that ultimately the algorithm's goal is to find as sources some quantities proportional to the data class-conditional log-likelihoods – for each class in an individual output dimension. Intuitively, it can be seen that the histogram of the optimal projections will have a peak signifying irrelevant documents and then in a positive direction will have projections of the relevant instances. As the $J_k = E[h_k(s_k)]$ form of object function where $h_k(.)$ is a nonlinear function – in our case ${s_k}^3$ – dates back to infomax and negentropy maximisation the proposed algorithm in this context also minimizes the mutual information between class-conditional data log-likelihoods, which seems a reasonable measure of unsupervised partitioning. The skew objective function for ICA in relation to ML mixture modeling seems also to be rather natural as the link between symmetric objective functions and ML was explored in [15].

Due to the symmetrical nature of the many types of signals which ICA has been applied to it has seldom been necessary to consider asymmetry. In this application however it is one of the key features of the data. In the 2D case for example, beside both the positive and negative directions to the class means (which give both the correct partitioning), there it is also possible to get one positive and one negative mean direction, in which case the classes are confused. In K dimensions the number of unwanted local optima is $2^K - 2$. We conjecture that this is the reason for confusion between two classes in the experiment reported in [8]. There it was assumed that $p(s_k) \propto exp(-logcosh(s_k))$, and the associated error function, $E_k = E[logcosh(s_k)]$ is of course symmetrical. However, in this case, the performance practically depends on classes being distinct enough that PCA compression is of the assumed shape.

V. Simulation Results

Simulations were run on binary coded text documents from the 20 Newsgroups [1] corpus. In all these experiments convergence was achieved in a very few (3-8) iteration steps. Projecting onto the estimated class mean directions resulted in a reasonable accuracy of clustering, even in cases when the initial representation used a small number of terms, which resulted in classes being less clearly defined.

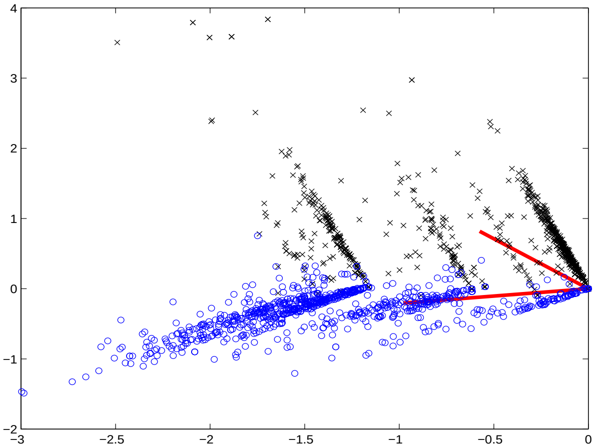

Fig. 4. Estimated first two class-means on a PCA compressed subset of 3000 documents (1000 in each class) from the 20 Newsgroups text corpus

Of course, we cannot expect high accuracy if we start from a rather small initial dictionary, given the fact we drastically compressed the original data. However there are many possibilities to improve on the accuracy. Firstly, the size of the used dictionary allows us to start from a much higher dimensional space, in which case the compressed classes are more likely to have smaller overlapping regions. This doesn't increase much the computational cost, because all we need is to compute a few eigenvectors in the preprocessing stage, which can be efficiently done for example by standard methods, like the Lanczos algorithm [5] or efficient EM-based methods [12] and then the ICA processing takes place in a much smaller dimensionality than the original representation space. Fig.5 shows a case when the number of terms highly exceeded the number of document instances, so the classes are more distinct even after compressing. Fig.6 shows the MAP estimates of class posteriors obtained from the ICA clustering. There were 100 documents in each class, which is nicely revealed by the softmaxed ICA outputs.

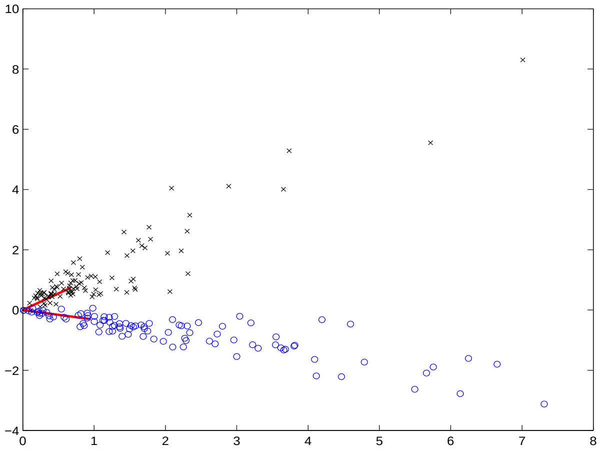

Fig. 5. First two estimated class means when initial representation used 800 terms.

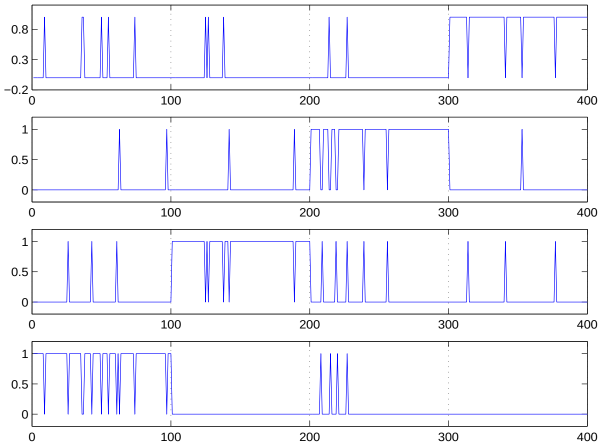

Fig. 6. MAP class posterior estimates obtained in a 4 classes experiment. The initial space was 800 dimensional.

Another advantage of the method is that the mean estimates of the missclassified instances have typically class posteriors around $1/K$. This is because they exist around

[1] http://www.cs.cmu.edu/ textlearning/

the boundary between the classes. In consequence we can identify the uncertainty in the decision – contrary to mixture modeling techniques which tend to produce extreme probabilites [2] due to the class conditional independence assumption in those models. However this feature requires a bit of care, because the scaling of the produced class posteriors also has be taken into account, as it depends on the data from the moment we fixed the means' norm to unity.

Another use and possible improvement of this method is to take the produced class posterior probabilities as initial estimates and pass them to another type of clustering algorithm. EM-based algorithms for example can produce good data representations, but typically are highly sensitive to initial conditions. Starting from the initial estimates provided by this very fast algorithm can give the benefit of starting from a closer point to the solution. This can be achieved by supervised training in the initialization phase, where the supervisor is the ICA result. This post-processing has the advantage of refining the class posterior estimates from a proper probabilistic treatment of the multidimensional data. However it is to be noticed that the benefit depends on the probabilistic model formulated in the second clustering algorithm fitting the data sufficiently well (otherwise an increase in the likelihood doesn't give an increase in accuracy, moreover it can decrease it).

As we dealt with binary coded data, we formulated a Bernoulli and a Poisson mixture model using EM and initialized them using class posteriors obtained by ICA. This resulted in very fast convergence of the likelihood in each of these experiments, and the Poisson model was able to improve over ICA's result, whereas the Bernoulli model did not. This is because the data is sparse and the Poisson model is a better fit to the data. The most significant improvements were registered when the representation used a small (insufficient) number of terms.

For example on a subset consisting of 5000 documents (1000 in each class) of the 20 Newsgroups corpus, with an initial document representation using a vocabulary of 100 terms, the ICA method gave 60.02% accuracy and the Poisson mixture post-processing was able to improve it to 78.28%. On another subset consisting of 3000 documents (1000 in each class), ICA gave an accuracy of 81.76% and the Poisson mixture modeling starting from here finally gave 89.86%. By accuracy we mean the percentage of document instances classified in accordance to the pre-defined manual labeling. The labels were available only for the evaluation purpose, the confusion matrix was computed and the ICA result was permuted after rearranging the results in accordance with the confusion matrix. It also can be noted that columns of $\mathbf{W}$, are normalized estimates of the the compressed class' means, which also has probabilistic meaning, namely $(\mathbf{WU}^T)_{ik} \propto p(word_i|k)$, i.e. individual peaks in the columns of $\mathbf{W}$ identify an estimate of keywords for each class.

VI. Conclusions and future work

We have presented a fast method for clustering text documents, by a skewness maximizing variant of ICA. The algorithm follows from some straightforward modifications of Gaussian mixture modeling by EM and taking into consideration the data-specific shape in the eigen-space. Simulations confirmed the importance of the skewness assumption on the latent priors for the document clustering/classification problem. This is in contrast to the usual assumptions of symmetry in ICA.

Further studies are intended to go in three directions: (1) relaxing the assumption that the number of classes is known, by incorporating some model estimation techniques, (2) finding a way of exploiting other prior knowledge like a number of available manually labeled instances, and (3) building a more flexible clustering mechanism from this type of clustering machine in a mixed or hierarchical manner [16], which would allow a more structured representation for data access and retrieval.

Acknowledgments This work is funded by the British Library and Information Commission, Grant Number LIC-RE-092 '*Improved Online Information Access*'.

References

[1] Dumais, Susan; Platt, John; Hackerman, David; Sahami, Mehran, *Inductive Learning Algorithms and Representations for Text categorization*, 7th International Conference on Information and Knowledge Management (1998) http://www.cs.umbc.edu/cikm

[2] Sahami, Mehan *Using Machine Learning to Improve Information Access*, PhD Thesis, Stanford University, 1998

[3] Bishop, C.M. *Neural Networks for Pattern Recognition.* Oxford University Press, 1995.

[4] A. Hyvärinen, R. Cristescu and E. Oja. *A fast algorithm for estimating overcomplete ICA bases for image windows.* In Proc. Int. Joint Conf on Neural Networks, Washington D.C., 1999.

[5] Berry MW. *Large-Scale Sparse Singular Value Computations.* The International Journal of Super-computer Applications Vol. 6 No.1, 1992.

[6] Girolami, M., *Self-Organising Neural Networks*, Springer, 1999.

[7] Girolami, M., *Hierarchic dichotomizing of polychotomous data – an ICA based data mining tool*, First International Workshop on ICA and Signal Separation, Aussois, France – January 11-15, 1999, pp.143-148.

[8] Kolenda, T., Hansen, L.K, Sigurdsson, S. *Independent Components in Text*, in Advances in Independent Component Analysis, Springer-Verlag, (Editor M, Girolami), pp 241-262.

[9] Isbell, C.L., Viola,P. Restructing Sparse High Dimensional Data for Effective Retrieval. Advances in Neural Information Processing Systems 11, 480-486, 1998.

[10] Yang, Y. *Noise Reduction in a Statistical Approach to Text Categorisation*, SIGIR 1995: 256-263. http://www.cis.hut.fi/ aapo/pub.html

[11] Deerwester S., Dumais S.T., Furnas G.W., Landauer, T.K.,Harshman R. *Indexing by Latent Semantic Analysis.* J Amer Soc Inf Sci 41, 6, 391-407, 1990.

[12] Rosipal, R. *An Expectation Maximization Approach to Nonlinear Component Analysis*, University of Paisley, Technical Report No 5.

[13] Lee,TW; Lewicki, M.S.; Sejnowski, T. *ICA Mixture Models For Unsupervised Classification And Automatic Context Switching*, 1'st International Workshop on Independent Component Analysis (1999) pp.209-214.

[14] A.D. Back and A.S. Weigend: *A first application of independent component analysis to extracting structure from stock returns*, Int. Journal of Neural Systems, vol. 8, No.4, August, 1997, pp. 473-484.

[15] Oja, E. *Nonlinear PCA Criterion and ML in ICA*, First International Workshop on ICA and Signal Separation, Aussois, France – January 11-15, 1999, pp.143-148.

[16] Pajunen, P and Girolami, M. Implementing Decisions in Binary Decision Trees Using Independent Component Analysis. Submitted to Second International Workshop on ICA and Signal Separation, Helsinki, Finland, 2000. 2000.