Mixture of Watson Distributions: A Generative Model for Hyperspherical Embeddings

Avleen S. Bijral, Markus Breitenbach and Greg Grudic
Department of Computer Science
University of Colorado, Boulder
Boulder, CO 80309
{avleen.bijral,markus.breitenbach,gregory.grudic}@colorado.edu

Abstract

Machine learning applications often involve data that can be analyzed as unit vectors on a d-dimensional hypersphere, or equivalently are directional in nature. Spectral clustering techniques generate embeddings that constitute an example of directional data and can result in different shapes on a hypersphere (depending on the original structure). Other examples of directional data include text and some sub-domains of bio-informatics. The Watson distribution for directional data presents a tractable form and has more modeling capability than the simple von Mises-Fisher distribution. In this paper, we present a generative model of mixtures of Watson distributions on a hypersphere and derive numerical approximations of the parameters in an Expectation Maximization (EM) setting. This model also allows us to present an explanation for choosing the right embedding dimension for spectral clustering. We analyze the algorithm on a generated example and demonstrate its superiority over the existing algorithms through results on real datasets.

1 Introduction

Spectral Methods have long gained popularity in areas like clustering and computer vision applications. These methods employ eigenvectors of a sample affinity matrix to form a low-dimensional normalized embedding and use K-means in post-processing steps. There have been tremendous advances in theoretical foundations behind such methods and different frameworks have been proposed to explain their success.

Most of these attempts pose spectral methods as a relaxation of concrete problems (e.g. Markov random walks [1]). One approach [2] attempts to explain the apparently trivial clustering in embedding space as a phenomenon of decrease in the angles between similar vectors in this new space. Specifically, truncating the dimensionality of the embedding enhances the structure in the data, and consequently any further analysis in this representation is more likely to be fruitful.

The authors of [2] also present a brief analysis of such embeddings as directional data on d-dimensional hyperspheres. Directional statistics is primarily concerned with unit vectors or equivalently vectors residing on the surface of a hypersphere of unit radius. The techniques employed therein are very different from usual statistics, and it is this concern that warrants analysis of embeddings as points on a hypersphere. Datasets where standard Mahalanobis type distances are not very effective are more likely to be explained as instances of directional distributions and can be said to possess "directional" properties. Such data exists commonly in domains like bio-informatics and text mining.

More recently, [3] proposed a clustering technique for directional data based on the von-Mises-Fisher (vMF) distribution [4], pitched it against k-means type algorithms [5] and demonstrated the need and efficacy of specific modeling of directional data in machine learning context. The von-Mises-Fisher distribution is analogous to the Gaussian distribution for spherical data, and its form is convenient to work with, albeit with modeling limitations. In [3], the authors present an EM algorithm for mixtures of von-Mises-Fisher distributions and numerical approximations for the parameters involved.

However, the limited modeling capability limits accuracy on noisy, thinly spread clusters since von-Mises-Fisher distribution inherently models only circular or tight clusters. Moreover, spectral embeddings can result in noisy formations in the embedding space, and hence vMF type distributions can not be expected to perform well. In this paper, we first present our intuition about spectral methods, analyze embeddings on

hyperspheres and attempt to correlate known results with our findings. We build upon the above described work to develop an EM-algorithm for mixtures of Watson distributions on hyperspheres and derive fast numerical approximations for the parameters. We go on to demonstrate that our method allows for increased modeling flexibility for spectral embeddings and directional data in general. We test our models on generated and real world datasets and demonstrate the efficacy of our approach over the soft-moVMF algorithm [3]. We also discuss the issue of choosing the right embedding dimension from a Watson distribution perspective. In the next section, we present our intuition on spectral embeddings and how directional distributions are relevant. We present a generative model based on the Watson distribution and present experimental evidence for the same.

2 Spherical Distributions and Embeddings

2.1 Watson Distribution

A unit vector x of dimension-d is said to have the multivariate Watson distribution if its probability density function is given as

$$f(\pm x|\mu,\kappa) = M(\frac{1}{2},\frac{d}{2},\kappa)^{-1} e^{\kappa(\mu^T x)^2} \tag{1}$$

where $M(\frac{1}{2},\frac{d}{2},\kappa)$ is the confluent hyper-geometric function also known as Kummer function (see [6] for more details). There also exists an extensive Matlab library [7] for computing special functions associated with spherical distributions.

The distribution is rotationally symmetric about μ, which is also a unit vector. As κ, the concentration parameter, increases the distribution tends to get more spread out around μ. For $\kappa < 0$, the distributions tends to be a girdle around the hypersphere (for e.g. figure 1(b)). This varied range of the concentration parameter allows for a lot of flexibility in modeling different kinds of embeddings. In contrast, the von-Mises-Fisher distribution given as

$$f(x|\mu,\kappa) = \frac{\kappa^{d/2-1}}{(2\pi)^{d/2} I_{d/2-1}(\kappa)} e^{\kappa\mu^T x} \tag{2}$$

only allows for $\kappa \geq 0$, here $I_{d/2-1}(\kappa)$ is the modified Bessel function of first kind and order $d/2-1$. Moreover the iso-density lines of the distribution are circles. Equivalently points with the same density lie on a circle. This is a far cry from the noise in real world datasets and especially embeddings because of the inadequacy of the Euclidean distance used for estimating the embeddings (see [4] for more details on these distributions).

2.2 Embeddings

Given a sample of points $S = \{x_1, ..., x_n\}$ in $\Re^l$ and an affinity matrix $W \in \Re^{n\times n}$ defined by

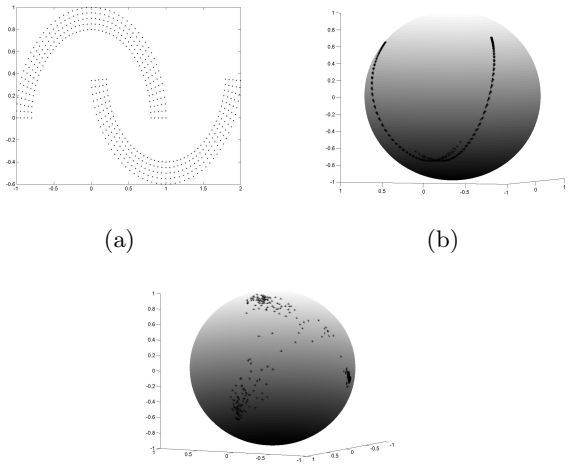
$$W_{ij} = e^{\|x_i - x_j\|^2/2\sigma^2} \tag{3}$$

and the corresponding Laplacian

$$L = D^{-1/2} W D^{-1/2} \tag{4}$$

where D is a diagonal matrix and where the (i,i) element is the sum of the ith row of W.

Let us consider a matrix V whose columns are the d largest eigenvectors (corresponding to d largest eigenvalues) $\{v_1, ..., v_d\}$. The spectral embedding for a point x_i in the sample is then a vector $u_i \in \Re^d$ corresponding to the ith row of the matrix V, such that $\| u_i \| = 1$. Figure 1 displays embeddings derived from generated data and a subsample of USPS digits dataset. It is clear that a model has to be flexible enough to consider girdle type and standard directional data.



(a) (b)

(c)

Figure 1: Spectral Embeddings

In [8], the authors prove a proposition stating that if the off-diagonal blocks of the affinity matrix are zero, then on this hypersphere the points cluster around mutually orthogonal points. For this simple case, modeling the embedding is trivial since the concentration parameter κ is going to be very large and a standard von-Mises-Fisher model would suffice. However, in a realistic case, the non-diagonal blocks are going to be

non-zero and, as is intuitive, the concentration around the cluster means is going to decrease as can be seen in figure 1(c).

The Watson distribution not only allows for modeling flexibility, but, regardless of the dimensionality of the data, only two parameters, the κ and the mean μ, need to be estimated. Another problem with spectral methods is choosing the right number of eigenvectors, as a small embedding dimension might not capture the structure in the data appropriately, and a large dimension might obscure any structure. We present an explanation for this problem based on our model.

In the next section, we extend the work in [3], derive a clustering algorithm based on posterior probabilities output by our hybrid-EM for mixtures of Watson distribution and conduct experiments on real world datasets.

3 Mixture of Watson Distributions

Consider a generative model for directional data as a mixture of K Watson distributions. Let $f_j(x|\phi_j)$ be one Watson component for a class corresponding to the parameters $\phi_j = (\mu_j, \kappa_j)$ and $1 \leq j \leq K$. The density for a point generated by this model is then given by

$$f(x|\Phi) = \sum_{j=1}^{K} \alpha_j f_j(x|\phi_j) \tag{5}$$

where $\Phi = (\alpha_1, ..., \alpha_K, \phi_1, ..., \phi_K)$ and α_j are the mixing proportions that sum to one.

Following the standard EM technique [9], the E-step in the EM computes the expectation of the complete likelihood over the distribution of the hidden variables, and the M-step computes the parameters Φ which maximize this expectation. The expectation is given as

$$E[\log(P(X,Y|\Phi))] = \sum_{j=1}^{K}\sum_{i=1}^{n} \log(\alpha_j) p(j|x_i,\Phi) + \sum_{j=1}^{K}\sum_{i=1}^{n} \log(f_j(x_i|\phi_j)) p(j|x_i,\Phi) \tag{6}$$

where $p(j|x_i,\Phi)$ is the posterior distribution of the hidden variable. This expression contains two unrelated terms and can be separately maximized. Allowing for the $\sum_{j=1}^{K} \alpha_j = 1$ constraint, we get [9]

$$\alpha_j = \frac{1}{n} \sum_{i=1}^{n} p(j|x_i,\Phi) \tag{7}$$

To maximize the second term, we have to form the Lagrangian with respect to the constraint $\mu_j^T \mu_j = 1$ and, as discussed before, unlike the EM for von-Mises-Fisher distribution, there is no positivity constraint on κ_j. We now have

$$L = \sum_{j=1}^{K}\sum_{i=1}^{n} \log(f_j(x_i|\phi_j)) p(j|x_i,\Phi) + \sum_{j=1}^{K} \lambda_j (1 - \mu_j^T \mu_j) = \tag{8}$$

$$\sum_{j=1}^{K}\sum_{i=1}^{n} (\kappa_j (x_i^T \mu_j)^2 - \log(M(\kappa_j))) p(j|x_i,\Phi) + \sum_{j=1}^{K} \lambda_j (1 - \mu_j^T \mu_j) \tag{9}$$

where $M(\kappa_j) = M(\frac{1}{2}, \frac{d}{2}, \kappa_j)$ for convenience of notation.

To obtain the update equations for each (μ_j, κ_j), we set each partial derivative of L to zero.

1) Differentiating w.r.t each λ_j

$$\mu_j^T \mu_j = 1 \tag{10}$$

2) Differentiating w.r.t each μ_j

$$\sum_{i=1}^{n} \kappa_j (x_i^T \mu_j) x_i p(j|x_i,\Phi) = \lambda_j \mu_j \tag{11}$$

3) Differentiating w.r.t each κ_j

$$\sum_{i=1}^{n} (x_i^T \mu_j)^2 p(j|x_i,\Phi) = \frac{M'(\kappa_j)}{M(\kappa_j)} \sum_{i=1}^{n} p(j|x_i,\Phi) \tag{12}$$

Since $\| \mu_j \| = 1$, from equation 12 and 13 we have

$$\kappa_j \| \sum_{i=1}^{n} (x_i^T \mu_j) x_i p(j|x_i,\Phi) \| = \lambda_j \tag{13}$$

substituting back in equation 13, we get

$$\mu_j = \frac{\sum_{i=1}^{n} (x_i^T \mu_j) x_i p(j|x_i,\Phi)}{\| \sum_{i=1}^{n} (x_i^T \mu_j) x_i p(j|x_i,\Phi) \|} \tag{14}$$

3.1 Approximating κ

Equation 12 presents a highly nonlinear equation that can perhaps be solved by Newton's method, though at a huge computational cost. Moreover, the accuracy of the computation of the Kummer function can also significantly alter results. The Kummer function, however, has properties that we employ to derive an approximation. It can be verified [10] that for the Kummer function there exists a continued fraction for the

ratio of the derivative of the function to the function itself

$$\frac{\kappa M^{'}(\kappa_j)}{M(\kappa_j)} = \frac{(1/2)\kappa}{(d/2) - \kappa + \frac{(1/2+1)\kappa}{(d/2+1)-\kappa \ + \ .....}} \tag{15}$$

Thus, if

$$T = \frac{M^{'}(\kappa_j)}{M(\kappa_j)} = \frac{\sum_{i=1}^{n}(x_i^T\mu_j)^2 p(j|x_i,\Phi)}{\sum_{i=1}^{n} p(j|x_i,\Phi)} \tag{16}$$

from equation 12, then the above equation can be approximately expressed as

$$T\kappa \approx \frac{(1/2)\kappa}{(d/2) - \kappa + T\kappa} \tag{17}$$

Solving this linear equation in κ gives us the following approximation

$$\kappa \approx \frac{1}{2}(\frac{1-Td}{T^2-T}) \tag{18}$$

We add a empirically determined compensating term to improve the numerical accuracy and the final approximation becomes

$$\kappa \approx \frac{1}{2}(\frac{1-Td}{T^2-T}) + \frac{-T^2}{d(T^2-T)} \tag{19}$$

Our approximation works regardless of the dimensionality of the data and outperforms the approximations given in [4], which are mostly for d close to 3. A similar approach was also employed in [3] and was shown to lead to better results. Moreover our approximation also allows κ to take on negative values when the data is thinly spread on the hypersphere, and this property leads to better noise handling, as we demonstrate later.

From these equations, we get the EM algorithm for a mixture of Watson distributions:
E-step: Compute for all points x_i and mixture components $1 \leq j \leq K$

1. $f_j(x_i|\phi_j) = M(\frac{1}{2}, \frac{d}{2}, \kappa_j)^{-1} e^{\kappa_j(\mu_j^T x_i)^2}$
2. $p(j|x_i,\Phi) = \frac{\alpha_j f_j(x_i|\phi_j)}{\sum_{h=1}^{K} \alpha_h f_h(x_i|\phi_j)}$

M-step: Update α_j, μ_j and κ_j for all mixture components

$\alpha_j = \frac{1}{n}\sum_{i=1}^{n} p(j|x_i,\Phi)$

Solve the following non-linear equation for μ_j

$$\mu_j - \frac{\sum_{i=1}^{n}(x_i^T\mu_j)x_i p(j|x_i,\Phi)}{\| \sum_{i=1}^{n}(x_i^T\mu_j)x_i p(j|x_i,\Phi) \|} = 0 \tag{20}$$

Compute T using μ_j

$$\begin{aligned} T &= \frac{\sum_{i=1}^{n}(x_i^T\mu_j)^2 p(j|x_i,\Phi)}{\sum_{i=1}^{n} p(j|x_i,\Phi)} \\ &= \frac{\sum_{i=1}^{n}(x_i^T\mu_j)^2 p(j|x_i,\Phi)}{n\alpha_j} \end{aligned} \tag{21}$$

$$\kappa_j = \frac{1}{2}(\frac{1-Td}{T^2-T}) + \frac{-T^2}{d(T^2-T)} \tag{22}$$

The E-step returns the posterior probabilities $p(j|x_i,\Phi)$ of all the classes, given the point, and are employed in all our experiments. There does not appear to be a closed form solution for μ_j, which can be solved by Newton's method using a standard optimization package. However, the bulk of the computational bottleneck for the procedure is taken care of by the κ approximation.

4 Embedding Dimension

The problem of choosing the right embedding dimension or equivalently the number of eigenvectors of the affinity matrix in spectral clustering is nontrivial, since a too large or too small dimension can obscure the cluster formation in the data. It turns out that the form of the Watson distribution provides us with some explanation of how to choose the ideal embedding dimension.

From an EM perspective, we seek the dimension d that maximizes L from equation, specifically the term $\sum_{i=1}^{n}(x_i^T\mu_j)^2$. It is clear that the dimension that maximizes this term will also maximize the expected log-likelihood. For clarity we assume only one cluster with mean μ. The expression

$$D = \sum_{i=1}^{n}(x_i^T\mu)^2 \tag{23}$$

measures the tightness of the cluster on the hypersphere; the larger D is higher is, the higher the concentration around the mean. It is equivalent to

$$D = \mu^T S\mu = < S\mu, \mu > \tag{24}$$

where $S = \frac{1}{n}\sum_{i=1}^{n} x_i x_i^T$ is the scatter matrix and measures the dispersion about the origin. S is symmetric, and it is well known [11] that, subject to constraint $\mu^T\mu = 1$, D has the maximum value, which is equal to the magnitude of the largest eigenvalue $\lambda(S)$.

$$\lambda(S) = max < S\mu, \mu > \quad s.t. \quad \mu^T\mu = 1 \tag{25}$$

It is reasonable, then, to analyze the change in this eigenvalue when increasing the embedding dimension to $d+1$.

Conjecture 1 *If* $S = \frac{1}{n}\sum_{i=1}^{n} x_i x_i^T$, *where* $x_i \in \Re^d$ *and* $x_i^T x_i = 1$, *if* $\tilde{x}_i \in \Re^{d+1}$ *such that* $\tilde{x_i}^T \tilde{x_i} = 1$ *and* $\tilde{x_i} = [\frac{x_i^T}{\|\tilde{x_i}\|} \frac{a_i}{\|\tilde{x_i}\|}]^T$ *then* $\lambda_1(S) > \lambda_1(\tilde{S})$ *where* λ_1 *is the largest eigenvalue.*

To simplify our analysis lets augment each unit vector x_i with an added dimension having the value a such that $\tilde{x_i} = [\frac{x_i^T}{\sqrt{(1+a^2)}} \quad \frac{a}{\sqrt{(1+a^2)}}]^T$ is also a unit vector.

Then, the resulting scatter matrix $\tilde{S}$ is given by

$$\tilde{S} = \frac{1}{n}\sum_{i=1}^{n}[\frac{x_i^T}{\sqrt{(1+a^2)}} \frac{a}{\sqrt{(1+a^2)}}]^T \quad (26)$$
$$[\frac{x_i}{\sqrt{(1+a^2)}} \quad \frac{a}{\sqrt{(1+a^2)}}]$$

This is equivalent to

$$\tilde{S} = \frac{1}{(1+a^2)}\begin{bmatrix} S & a\sum_{i=1}^{n} x_i/n \\ a\sum_{i=1}^{n} x_i^T/n & a^2 \end{bmatrix} \quad (27)$$

It can be verified [4] that because of the constraint $\tilde{x_i}^T \tilde{x_i} = x_i^T x_i = 1$

$$tr(\tilde{S}) = tr(S) = \sum_{i=1}^{d}\lambda_i(S) = \sum_{i=1}^{d+1}\lambda_i(\tilde{S}) = 1 \quad (28)$$

We can see that because of this constraint the added dimension causes the eigenvalues to distribute mass, and it appears that the largest eigenvalue of $\tilde{S}$ would tend to decrease.

If the embedding dimension is too large the dispersion on the hypersphere would be very high, and any structure in the data would be obscured. At the same time, if the embedding dimension is too low we could loose information about the data. It seems intuitive to choose an embedding dimension close to the number of clusters. As we show later, our experiments corroborate this fact.

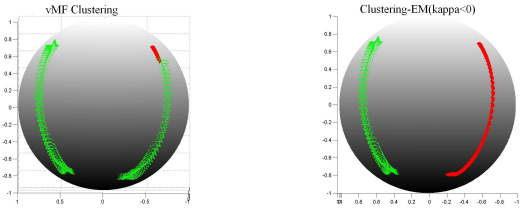
5 Experimental Results

We evaluate our algorithm on generated data and a variety of real datasets. In all experiments we averaged the results over 20 runs, and κ was initialized to 10 for every class. The mean μ was randomly chosen. For each experiment, spectral embeddings were precomputed, and the algorithms were run on the computed embeddings.

We compare against the soft-moVMF algorithm. No comparison against K-Means based Spectral Clustering is necessary as it was demonstrated in [3] that soft-moVMF always performs equal or better to K-Means on directional data. We demonstrate that our algorithm outperforms soft-moVMF in cases where the data is very noisy. To this end we make plots showing that the proposed algorithm performs equal or better than soft-moVMF while increasing the embedding dimension of the data. Note that in the usual settings of Spectral Clustering, the embedding dimension is set to the number of clusters [8].

Ideally, a clustering algorithm would discover clusters that are meaningful to its users. Therefore, we use data sets in which the labels are known, namely USPS digits, 20 Newsgroups and Yahoo20. We measure performance by using Mutual Information as in [3].

5.1 Generated Data



(a) (b)

Figure 2: Clustering two-moon embeddings with soft-moVMF and the proposed algorithm.

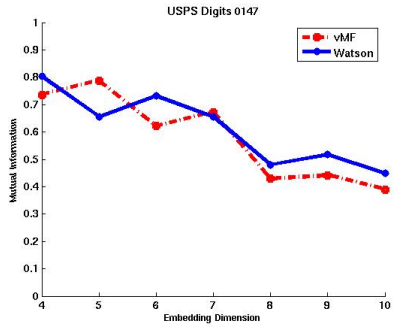
In this section, we study the behavior of our clustering-EM algorithm on a simple generated example derived from the three dimensional spectral embeddings for the two-moons dataset (figure 1(a)). Figure 1(b) displays the embeddings as two separate bands on a sphere and the clusterings obtained by our algorithm for $\kappa < 0$ and soft-moVMF for $\kappa > 0$.

For $\kappa > 0$, soft-moVMF clustering tends to put every point in the same cluster resulting in a very large concentration parameter (κ) for one cluster and a very small concentration for the other. Whereas for $\kappa < 0$, the Watson clustering is perfect and ultimately converges to roughly equal negative concentration parameters. It is evident that the proposed algorithm can handle more topologies than the soft-moVMF algorithm, where κ is constrained to be greater than zero.

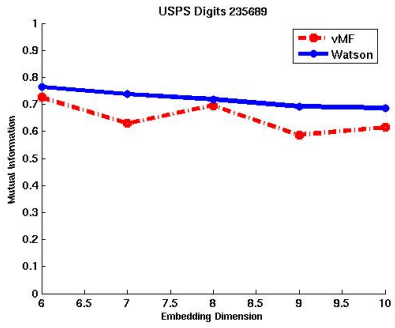
5.2 Clustering USPS Digits

We use a subset of the USPS Zipcodes Dataset. The images are scaled to mean zero and then embedded in spectral space. The kernel sigma used was $\sigma = 0.01$.

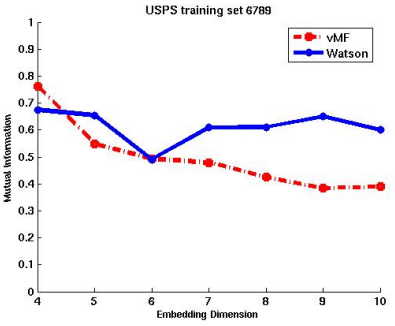
The eigenvector embeddings are normalized to length one such that each point lies on the d dimensional hypersphere. In this embedding space, we compare the performance of soft-moVMF and the EM-Watson algorithm.



(a)

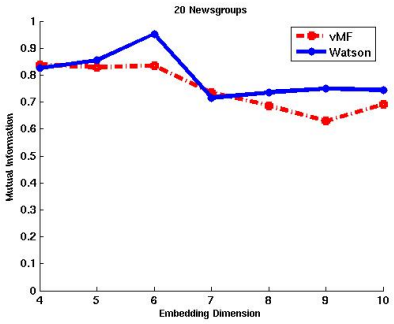


(b)

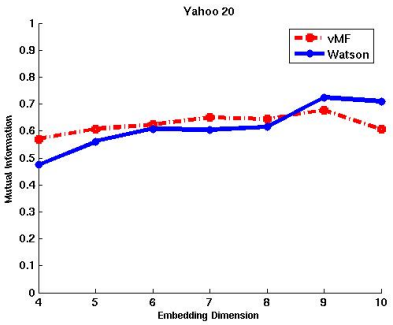


(c)

Figure 3: **soft-moVMF vs. EM-Watson:** Mutual Information vs. Embedding Dimension (a) USPS Digits 0147 Test-set (b) USPS Digits 235689 Test-set (c) USPS Digits 6789



(a)



(b)

Figure 4: **soft-moVMF vs. EM-Watson:** Mutual Information vs. Embedding Dimension (a) 20 Newsgroups (b) Yahoo20

We setup different subsets in order to test the performance. We first start with the digits $\{6, 7, 8, 9\}$, as they are more difficult to distinguish than the commonly used digits 1-4.

We also choose two subsets from the USPS test-set, namely $\{0, 1, 4, 7\}$ and $\{2, 3, 5, 6, 8, 9\}$. We use the test-set as it is known to be more noisy than the training set.

In figure 3(a)-(c), we can see that our algorithm performs as well or better than soft-moVMF. For a higher embedding dimension our algorithm outperforms soft-moVMF consistently.

5.3 Clustering 20 Newsgroups

In this experiment, we cluster natural language text from the 20 newsgroups dataset (version 20-news-18828). We choose a subset of the topics in *rec.** which contains autos, baseball, hockey and motorcycles (4 classes). The articles were preprocessed using the Rainbow software package [12] with the following options: (1) skipping any header, as they contain the correct newsgroup; (2) stemming all words; (3) removing stop words; (4) ignoring words that occur in 5 or fewer documents. By removing documents that have less than 5 words, we obtained 3970 document vectors in 8014 dimensional space. The documents were nor-

malized into TFIDF representation. As the kernel for the affinity matrix, we use the function

$$K(x, y) = e^{-\frac{(1-y^T x)}{\sigma^2}} \quad (29)$$

for text-data as suggested in [13]. The kernel-sigma is set to $\sigma = 3$. We can see in figure 4(a) that our algorithm performs equally well or better than soft-moVMF.

5.4 Clustering Yahoo20

We obtained a subset of the Yahoo-20 data set that contains 2340 news articles belonging to 6 different news topics. We pre-process the data by extracting the raw text of the news article out of the HTML. Additionally we removed phrases that give away the class right away (e.g. copyright statements). After standard pre-processing steps, we then selected the top 1000 words based on information gain which resulted in a sparse 2340×1000 document-term matrix. As the kernel for the affinity matrix we use the equation (29) with $\sigma = 0.3$ and then compute the embedding.

5.5 Analysis

We note that in most experiments when the embedding dimension is close to the number of actual clusters, our algorithm performs comparably to soft-moVMF. However, when we increase the embedding dimension the clusters start to spread thinly on the hypersphere as explained in section 4. This makes the clusters less concentrated around the mean, or equivalently, the noise in the data increases.

Our algorithm performs better in these situations for most cases. Since Watson distribution can model thinly spread data better than von-Mises-Fisher distribution and this added modeling capability allows it to handle less concentrated or noisy clusters. In the experiments we observed that our EM algorithm assigns negative κ parameters to classes to model the increasing noise levels as we increase the embedding dimension.

6 Discussion and Future Work

In this paper, we presented the mixture of Watson distributions as a generative model for spectral embeddings and directional data. We demonstrated that our algorithm tends to peform significantly better than soft-moVMF on noisy thinly spread clusters. We also presented an explanation for the choice of embedding dimension. This discussion implied that, as we increase the embedding dimension, the concentration around the means (μ) decrease, and the unit vectors tend to spread out on the hypersphere. In the future, we intend to extend our model to very high dimensional problems by speeding up the M-step for the μ parameter. We also hope to analyze the convergence of the proposed EM algorithm in the light of the approximation used. There are other directional models such as the Bingham and the Kent distribution, which could be explored in lieu of their enhanced modeling capacity. However, the number of parameters to be estimated in these models is quite large and would require significantly more training data.

References

[1] M. Meila and J. Shi. A random walks view of spectral segmentation. In *Proceedings of the 7th International Conference on Artificial Intelligence and Statistics*, 2001.

[2] Band M. and Huang K. A unifying theorem for spectral embedding and clustering. In C. M. Bishop and B. J. Frey, editors, *Proceedings of the 9th International Conference on Artificial Intelligence and Statistics*, January 2003.

[3] A. Banerjee, I. S. Dhillon, Ghosh J., and Sra S. Clustering on the unit hypersphere using von mises-fisher distributions. *Journal of Machine Learning Research*, 6:1345–1382, Sep 2005.

[4] K.V. Mardia and P. Jupp. *DirectionalStatistics*. John Wiley and Sons Ltd., 2000.

[5] Dhillon I.S. and Modha D.S. Concept decompositions for large sparse text data using clustering. *Machine Learning*, 42(1):143–175, 2001.

[6] Abramowitz M. and Stegun I.A. *Handbook of Mathematical Functions*. Dover Publ. Inc., New York, 1974.

[7] Barrowes B. Matlab routines for computation of special functions. http://ceta.mit.edu/comp_spec_func/.

[8] A. Y. Ng, M. I. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In T. G. Dietterich, S. Becker, and Z. Ghahramani, editors, *Advances in Neural Information Processing Systems 14*, Cambridge, MA, 2002. MIT Press.

[9] Bilmes J. A gentle tutorial on the em algorithm and its application to parameter estimation for gaussian mixture and hidden markov models. Technical Report ICSI-TR-97-021, University of California, 1997.

[10] W. Gautschi. Anomalous convergence of a continued fraction for ratios of kummer functions. *Mathematics of Computation*, (31):994–999, 1977.

[11] R. Horn and C. Johnson. *Matrix Analysis*. Cambridge University Press, 1999.

[12] Andrew Kachites McCallum. Bow: A toolkit for statistical language modeling, text retrieval, classification and clustering. http://www.cs.cmu.edu/ mccallum/bow, 1996.

[13] D. Zhou, O. Bousquet, T.N. Lal, J. Weston, and B. Schölkopf. Learning with local and global consistency. In L. Saul S. Thrun and B. Schölkopf, editors, *Advances in Neural Information Processing Systems 16*, Cambridge, Mass., 2004. MIT Press.