Proceedings of the 6th Nordic Signal Processing Symposium - NORSIG 2004
June 9 - 11, 2004
Espoo, Finland

INFORMATION THEORETIC CLUSTERING: A UNIFYING REVIEW OF THREE RECENT ALGORITHMS

Robert Jenssen[1], *Torbjørn Eltoft*[1] *and Jose C. Principe*[2]

[1]Department of Physics, University of Tromsø, Norway
[2]Computational NeuroEngineering Laboratory, University of Florida, USA

ABSTRACT

We provide a unifying review of three recent information-theoretic clustering algorithms. They are all based on the Cauchy-Schwarz distance measure applied to probability density functions (pdfs). However, they employ different optimization techniques, namely a heuristically motivated, a gradient descent-based and a graph spectral scheme. Some of the characteristics of the algorithms are discussed, and comparative remarks are made.

1. INTRODUCTION

Traditionally, the majority of the clustering algorithms rely on a minimum variance criterion. Hence, such algorithms are suitable for clustering Gaussian data, and perform poorly on irregularly shaped clusters. In contrast, clustering based on information-theoretic criteria is attractive since such quantities convey information about pdfs. However, such criteria have been difficult to evaluate because they often require numerical procedures to evaluate integrals.

Recently, Principe et al. [1] proposed a new pdf distance measure based on the Cauchy-Schwarz (CS) inequality. The reason for introducing the CS distance was that it elegantly integrates non-parametric pdf estimation through Parzen windowing. The CS distance has recently been utilized by the current authors as a cost function for clustering in three new algorithms [2, 3, 4], each employing a different optimization scheme. In this paper, we provide a unifying review of these recent methods by comparing the clustering results obtained in the two-cluster case, and by highlighting some of the properties of each algorithm.

In section 2, we define the CS distance. Thereafter, in section 3, we explain the heuristically motivated optimization technique [2]. In section 4, we review the optimization method based on Lagrange multipliers [3], and in section 5 we discuss the spectral optimization procedure [4]. Finally, in section 8 we make our concluding remarks.

THIS WORK WAS PARTIALLY SUPPORTED BY GRANTS ECS-0300340 AND EIA-0135946

R. Jenssen: (+47) 776 46493, robertj@phys.uit.no

2. CAUCHY-SCHWARZ PDF DISTANCE

Based on the Cauchy-Schwarz inequality; $||\mathbf{x}||^2\,||\mathbf{y}||^2 \geq (\mathbf{x}^T\mathbf{y})^2$, we define the CS pdf distance [1];

$$D_{CS} = -\log \frac{\int p(\mathbf{x})q(\mathbf{x})d\mathbf{x}}{\sqrt{\int p^2(\mathbf{x})d\mathbf{x}\ \int q^2(\mathbf{x})d\mathbf{x}}} \geq 0. \qquad (1)$$

In order to obtain (1), inner products between vectors have been replaced by inner products between pdfs, i.e. $\langle p, q\rangle = \int p(\mathbf{x})q(\mathbf{x})d\mathbf{x}$. In order for D_{CS} to equal zero, the two pdfs must overlap completely. It goes to infinity as the overlap between the two pdfs goes to zero.

Assume that we estimate $p(\mathbf{x})$ based on the data points in cluster $C_1 = \{\mathbf{x}_i\}$, $i = 1, \ldots, N_p$, and $q(\mathbf{x})$ based on $C_2 = \{\mathbf{x}_j\}$, $j = 1, \ldots, N_q$. By the Parzen method [1] $\hat{p}(\mathbf{x}) = \frac{1}{N_p}\sum_{i=1}^{N_p} G(\mathbf{x}-\mathbf{x}_i, \sigma^2\mathbf{I})$, and $\hat{q}(\mathbf{x}) = \frac{1}{N_q}\sum_{j=1}^{N_q} G(\mathbf{x}-\mathbf{x}_j, \sigma^2\mathbf{I})$. Here we have used the multi-dimensional Gaussian kernel, $G(\mathbf{x}, \sigma^2\mathbf{I})$. Since maximization of D_{CS} is equivalent to minimization of the argument of the logarithm in (1), we now derive the expression for the latter quantity, which we denote $J_{CS}(C_1, C_2)$. This is done by substituting the pdfs by their Parzen estimates, and by utilizing the convolution theorem for Gaussians, resulting in;

$$J_{CS} = \frac{\sum_{i,j=1}^{N_p,N_q} G_{ij,2\sigma^2\mathbf{I}}}{\sqrt{\sum_{i,i'=1}^{N_p,N_p} G_{ii',2\sigma^2\mathbf{I}}\ \sum_{j,j'=1}^{N_q,N_q} G_{jj',2\sigma^2\mathbf{I}}}}, \qquad (2)$$

where $G_{ij,2\sigma^2\mathbf{I}} = G(\mathbf{x}_i - \mathbf{x}_j, 2\sigma^2\mathbf{I})$. In clustering, the goal is to assign labels to a dataset such that J_{CS} is minimized.

3. HEURISTIC OPTIMIZATION

In [2], a heuristic optimization technique for J_{CS} was proposed. The main idea is to "seed" a number of small initial clusters in the data set, and then to grow

the clusters until all patterns have been labeled. Subsequently, the members of the "worst" cluster are re-clustered, thus reducing the number of clusters by one. This procedure is repeated until only two clusters remain.

Assume that at any time a subset of the feature vectors have been assigned to the clusters $C_1, \ldots, C_K$. The unlabeled feature vector, $\mathbf{x}_j$, being closest to a labeled data point, $\mathbf{x}_i$, as measured by $G_{ij,2\sigma^2}$, is assigned to cluster C_k if;

$$\min_k J_{CS}(C_1, \ldots, C_k + \mathbf{x}, \ldots, C_K), \tag{3}$$

for $k = 1, \ldots, K$. This clustering rule is employed until all N data points have been clustered to one of the K clusters.

To find the "worst" cluster, we eliminate one cluster at a time, and calculate J_{CS} based on the remaining clusters in each case. By eliminating, we mean that the members of the eliminated cluster are considered unlabeled, not contributing to the value of J_{CS}. The "worst" cluster is now selected as the cluster that when eliminated, results in the smallest J_{CS} based on the remaining clusters, because this means that the remaining clusters are the most separated clusters. The members of the "worst" cluster are subsequently re-clustered according to (3). Initially, the K_{in} clusters are "seeded" randomly, having N_{in} members each.

If the affinity matrix, $\mathbf{G} = \left[G_{ij,2\sigma^2\mathbf{I}}\right]_{i,j=1,\ldots,N}$, is created beforehand, this algorithm can be implemented very efficiently. However, to create $\mathbf{G}$ from the available data is an $O(N^2)$ procedure. We will see in section 6 that $\mathbf{G}$ is created automatically. Hence, in the two-cluster case, the only user-specified parameters to the algorithm is K_{in} and N_{in}.

4. LAGRANGE OPTIMIZATION

An optimization technique based on the Lagrange multiplier formalism was proposed in [3]. For each data pattern $\mathbf{x}_i$, $i = 1, \ldots, N$, a membership vector $\mathbf{m}_i$ is defined. If $\mathbf{x}_i$ belongs to cluster C_1 (C_2), correspondingly $\mathbf{m}_i = [1,0]^T$ ($[0,1]^T$). We now rewrite (2), obtaining;

$$J_{CS} = \frac{\frac{1}{2}\sum_{i,j}\left(1 - \mathbf{m}_i^T\mathbf{m}_j\right)G_{ij,2\sigma^2}}{\sqrt{\sum_{i,j} m_{i1}m_{j1}G_{ij,2\sigma^2}\ \sum_{i,j} m_{i2}m_{j2}G_{ij,2\sigma^2}}}, \tag{4}$$

where m_{ik}, $k = 1, 2$, denotes element number k of $\mathbf{m}_i$, and $i, j = 1, \ldots, N$.

In order to minimize (4), we fuzzify the membership vectors such that $\mathbf{m}_i \in [0,1]$, $i = 1, \ldots, N$, and define the following constrained optimization problem:

$$\min_{\mathbf{m}_1,\ldots,\mathbf{m}_N} J_{CS}(\mathbf{m}_1, \ldots, \mathbf{m}_N), \tag{5}$$

subject to $\mathbf{m}_j^T\mathbf{1} - 1 = 0, \quad j = 1, \ldots, N$, where $\mathbf{1} = [1,1]^T$. Now we make a convenient change of variables. Let $m_{ik} = v_{ik}^2$, $k = 1, 2$. Consider;

$$\min_{\mathbf{v}_1,\ldots,\mathbf{v}_N} J_{CS}(\mathbf{v}_1, \ldots, \mathbf{v}_N), \tag{6}$$

subject to $\mathbf{v}_j^T\mathbf{v}_j - 1 = 0, \quad j = 1, \ldots, N$. The constraints for the problem stated in (6) are equivalent to the constraints for (5).The optimization problem, (6), amounts to adjusting the vectors $\mathbf{v}_i$, $i = 1, \ldots, N$, such that;

$$\frac{\partial J_{CS}}{\partial \mathbf{v}_i} = \left(\frac{\partial J_{CS}}{\partial \mathbf{m}_i}^T \frac{\partial \mathbf{m}_i}{\partial \mathbf{v}_i}\right)^T = \mathbf{\Gamma}\frac{\partial J_{CS}}{\partial \mathbf{m}_i} \to \mathbf{0}, \tag{7}$$

where $\mathbf{\Gamma} = \text{diag}(2\sqrt{m_{i1}}, 2\sqrt{m_{i2}})$. We force all elements $2\sqrt{m_{ik}}$, $k = 1, 2$, to always be positive by adding a small positive constant ϵ during each membership update. Hence, the direction of the gradients of $\frac{\partial J_{CS}}{\partial \mathbf{v}_i}$ and $\frac{\partial J_{CS}}{\partial \mathbf{m}_i}$ will always be the same. Thus, these scalars can be interpreted as variable step-sizes built into the gradient descent search process, as a consequence of the change of variables that we made. See [3] for the derivation of $\frac{\partial J_{CS}}{\partial \mathbf{m}_i}$.

The necessary conditions for the solution of (6) are commonly generated by constructing the Lagrange function, given by;

$$L = J_{CS}(\mathbf{v}_1, \ldots, \mathbf{v}_N) + \sum_{j=1}^{N} \lambda_j \left(\mathbf{v}_j^T\mathbf{v}_j - 1\right), \tag{8}$$

where λ_j, $j = 1, \ldots, N$, are the *Lagrange multipliers*. The necessary conditions for the extremum of L are given by;

$$\frac{\partial L}{\partial \mathbf{v}_i} = \frac{\partial J_{CS}}{\partial \mathbf{v}_i} + \sum_{k=1}^{N} \lambda_k \frac{\partial}{\partial \mathbf{v}_i}\left(\mathbf{v}_k^T\mathbf{v}_k - 1\right) = \mathbf{0}, \tag{9}$$

$$\frac{\partial L}{\partial \lambda_j} = \mathbf{v}_j^T\mathbf{v}_j - 1 = 0, \tag{10}$$

for $i, j = 1, \ldots, N$. From (9) we derive the following *fixed-point* update rule for the vector $\mathbf{v}_i$ as follows;

$$\frac{\partial J_{CS}}{\partial \mathbf{v}_i} + 2\lambda_i\mathbf{v}_i = \mathbf{0} \Rightarrow \mathbf{v}_i^+ = -\frac{1}{2\lambda_i}\frac{\partial J_{CS}}{\partial \mathbf{v}_i}, \tag{11}$$

$i = 1, \ldots, N$, and where $\mathbf{v}_i^+$ denotes the updated vector.

We solve for the Lagrange multipliers, λ_i, $i = 1, \ldots, N$, by evaluating (10), yielding;

$$\lambda_i = \frac{1}{2}\sqrt{\frac{\partial J_{CS}}{\partial \mathbf{v}_i}^T \frac{\partial J_{CS}}{\partial \mathbf{v}_i}}. \tag{12}$$

In [3] it was shown that by annealing the kernel-size over time, the proposed algorithm is to a large degree

able to avoid local minima. The drawback is that there is at present no principled way to determine a suitable annealing scheme.

After convergence of the algorithm, or after a predetermined number of iterations, we designate the maximum value of the elements of each $\mathbf{m}_i$, $i = 1, \ldots, N$, to one, and the rest to zero. We propose to initialize the vectors $\mathbf{v}_i$, $i = 1, \ldots, N$, randomly such that their elements take small values.

5. SPECTRAL OPTIMIZATION

In [4], it was shown that the CS distance in fact provides us with a new information-theoretic framework for graph-spectral clustering, which also establishes a close link between spectral clustering and non-parametric pdf estimation. In that framework J_{CS} was called the *Information Cut* (IC).

An N-dimensional membership vector, $\mathbf{m}$, is defined, such that $m_i = 1(0)$ if data point $\mathbf{x}_i$ is in $C_1(C_2)$. Eq. (2) is re-written;

$$IC = \frac{\mathbf{m}^T\mathbf{G}(\mathbf{1}-\mathbf{m})}{\sqrt{\mathbf{m}^T\mathbf{G}\mathbf{m}(\mathbf{1}-\mathbf{m})^T\mathbf{G}(\mathbf{1}-\mathbf{m})}}. \qquad (13)$$

where $\mathbf{G}$ is the affinity matrix, discussed in section 3.

By the Schur decomposition, $\mathbf{G}$ can be written as; $\mathbf{G} = \mathbf{E}\mathbf{\Lambda}\mathbf{E}^T = \mathbf{E}\mathbf{\Lambda}^{\frac{1}{2}}\mathbf{\Lambda}^{\frac{1}{2}}\mathbf{E}^T$, where the columns of $\mathbf{E}$ contain the eigenvectors, and $\mathbf{\Lambda} = \mathrm{diag}(\lambda_1, \ldots, \lambda_N)$ contains the corresponding eigenvalues in decreasing order. Now we define $\mathbf{u} = \mathbf{\Lambda}^{\frac{1}{2}}\mathbf{E}^T\mathbf{m}$ and $\mathbf{t} = \mathbf{\Lambda}^{\frac{1}{2}}\mathbf{E}^T\mathbf{1}$, where $\mathbf{1}$ is a N-dimensional ones-vector. Eq. (13) can now be re-written;

$$IC = \frac{\mathbf{u}^T(\mathbf{t}-\mathbf{u})}{\sqrt{||\mathbf{u}||^2||\mathbf{t}-\mathbf{u}||^2}} = \cos\angle(\mathbf{u}, \mathbf{t}-\mathbf{u}). \qquad (14)$$

We seek the $\mathbf{u}$ that minimizes (14). In [4] it was shown that that any vector $\mathbf{u}$ of the form $u_i \in \{t_i, 0\}$, $i = 1, \ldots, N$, minimizes the IC because $\cos\angle(\mathbf{u}, \mathbf{t}-\mathbf{u}) = 0$. It remains to determine which $u_i = t_i$ and which $u_i = 0$. The solution for the membership vector is then given by $\mathbf{m} = \mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{E}\mathbf{u}$.

In [4], it was noticed that only a few of the $t_i = \sqrt{\lambda_i}\ \mathbf{e}_i^T\mathbf{1}$, where $\mathbf{e}_i$ is the ith eigenvector, actually deviates significantly from zero. By only considering the M largest t_i, $i = 1, \ldots, N$, and the corresponding eigenvectors and eigenvalues, we can determine a M-dimensional $\hat{\mathbf{u}}$, such that the corresponding solution $\hat{\mathbf{m}}$ is an approximate solution to $\mathbf{m}$.

Define $w_i = 1$ if $\hat{u}_i = t_i$, and $w_i = 0$ if $\hat{u}_i = 0$, $i = 1, \ldots, M$. Thus, the approximate solution can be written;

$$\hat{\mathbf{m}} = \sum_{i=1}^{M} w_i \frac{u_i}{\sqrt{\lambda_i}}\ \mathbf{e}_i = \sum_{i=1}^{M} w_i(\mathbf{e}_i^T\mathbf{1})\ \mathbf{e}_i, \qquad (15)$$

In conclusion, the solution is given as a linearly weighted summation of some of the eigenvectors, where the weighting on each eigenvector is given by the sum of the elements of that eigenvector.

In order to determine which eigenvectors to use, we take as our starting point the eigenvector $\mathbf{e}_1$, corresponding to t_1, in (15). One-by-one, we include the other eigenvectors corresponding to the remaining t_i, $i = 2, \ldots, M$, into the sum, and in each case the IC is calculated. The component that yields the smallest IC-value will be appended permanently to (15) for the projection space. This procedure is terminated if the IC-value increases from one iteration to the next.

The number M of significant elements of $\mathbf{t}$ is determined by selecting the elements whose value is larger than e.g. $0.05 \times t_{i,\max}$, $i = 1, \ldots, N$. To obtain the discrete solution, $\mathbf{m}$, $\hat{\mathbf{m}}$ is thresholded such that elements larger than $1/2$ are given the value one, and elements smaller than $1/2$ are given the value zero.

6. CREATING THE AFFINITY MATRIX

The creation of $\mathbf{G}$ is closely linked to Parzen pdf estimation through σ. Automatic selection of σ is not a trivial task, but methods based on the mean integrated square error (MISE); $MISE(\hat{p}) = E\int\{\hat{p}(\mathbf{x}) - p(\mathbf{x})\}^2 d\mathbf{x}$, have been proposed. Silverman's "rule-of-thumb" [5] can be used to select the kernel-size. We suggest to estimate the "optimal" one-dimensional kernel-size for each dimension of the data, and use the smallest such value as our σ_{opt}, where $\sigma_{\mathrm{opt}} = s1.06N^{-1/5}$ [5], and s is an estimate of the standard deviation of the data.

7. PERFORMANCE STUDIES

We cluster the datasets shown in Fig. 1. In the heuristic method we use $K_{\mathrm{in}} = N_{\mathrm{in}} = 10$. For the Lagrange method, we select the upper limit for σ to be $1.5 \times \sigma_{\mathrm{opt}}$ and the lower limit to be $0.1 \times \sigma_{\mathrm{opt}}$, where the annealing is performed linearly over 300 iterations. These values are selected based on our experience. The spectral method is fully automatic.

Fig. 1 (a) show the result consistently obtained by the heuristic method. The Lagrange method obtains a similar result. This method is naturally slower than the other two methods, because of the annealing. The spectral algorithm obtains the result shown in Fig. 1 (b), which is also very good.

Fig. 1 (c) show the result obtained by both the heuristic and the spectral method. It is a perfect clustering on a challenging dataset. The Lagrange method does not handle this dataset as shown in Fig. 1 (d), even for very slow annealing rates. It may be that the algorithm gets trapped in a local minimum.

Finally, we cluster the four-dimensional IRIS dataset using the data corresponding to the two iris-plants (50 instances of each) which are known to overlap and to have a non-linear cluster boundary. The true clustering is shown in Fig. 1 (e) projected onto its first two principal components. Once again, the heuristic method performs very well, consistently yielding an error rate of $4-10\%$. The best result is shown in Fig. 1 (f). The Lagrange method obtaines similar results at best, but vary more for this dataset. The spectral method fails. The reason is that σ_{opt} is determined too large for the eigenvectors to exhibit discriminatory power. By manually selecting a smaller kernel-size, we are able to obtain reasonable results also for the spectral method.

8. CONCLUSIONS

We have reviewed three recent information-theoretic clustering algorithms, and shown that they can to a large degree handle irregularly shaped clusters, as opposed to many variance-based clustering algorithms. These algorithms rely on different optimization techniques, and hence exhibit different characteristics. The heuristic algorithm has in fact the overall best performance on the studied datasets. But it is dependent on two user-specified parameters, even though it has been shown to be robust with regard to these parameters [2]. For the Lagrange method the annealing scheme has to be selected, for which no principled approach exists. The spectral method reveals a very interesting link between information-theory and graph spectral methods, is fully automatic, but may be more sensitive to the kernel-size that the heuristic method. We only studied the two-cluster case, but all methods can easily be extended to $K > 2$ clusters.

9. REFERENCES

[1] J. Principe, D. Xu, and J. Fisher, "Information Theoretic Learning," in *Unsupervised Adaptive Filtering*, S. Haykin (Ed.), John Wiley & Sons, 2000, vol. I, Chapter 7.

[2] R. Jenssen, J. C. Principe, and T. Eltoft, "Cauchy-Schwartz pdf Divergence Measure for non-Parametric Clustering," in *IEEE Norway Section Signal Processing Symposium*, Bergen, Norway, 2003.

[3] R. Jenssen, D. Erdogmus, K. E. Hild, J. C. Principe, and T. Eltoft, "Cauchy-Schwarz Distance for Fuzzy Clustering using Parzen Kernel Annealing," *Submitted to IEEE Transactions on Pattern Analysis and Machine Intelligence.*

[4] R. Jenssen, T. Eltoft, and J. C. Principe, "Information Theoretic Spectral Clustering," in *Submitted to International Joint Conference on Neural Networks*, Budapest, Hungary, 2004.

[5] B. W. Silverman, *Density Estimation for Statistics and Data Analysis*, Chapman and Hall, 1986.

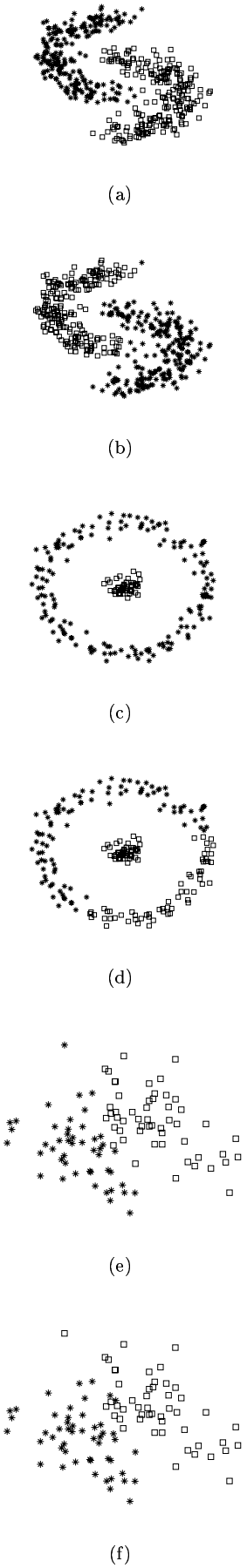

Figure 1: Datasets used in clustering experiments.