Generation of Alternative Clusterings Using the CAMI Approach

Xuan Hong Dang*   James Bailey†

Abstract

Exploratory data analysis aims to discover and generate multiple views of the structure within a dataset. Conventional clustering techniques, however, are designed to only provide a single grouping or clustering of a dataset. In this paper, we introduce a novel algorithm called CAMI, that can uncover alternative clusterings from a dataset. CAMI takes a mathematically appealing approach, combining the use of mutual information to distinguish between alternative clusterings, coupled with an expectation maximization framework to ensure clustering quality. We experimentally test CAMI on both synthetic and real-world datasets, comparing it against a variety of state-of-the-art algorithms. We demonstrate that CAMI's performance is high and that its formulation provides a number of advantages compared to existing techniques.

1 Introduction

Data clustering is an important topic in data mining. However, clustering is a challenging task, whose difficulty is caused by the lack of a unique and precise definition of what a cluster is [22, 19]. Information is not available about the underlying structure of the data, nor is there a unique similarity measure for differentiating clusters. Hence, it is not surprising that there is often no single clustering solution that explains the structure of a given dataset, especially if it is complex and represented in a high dimensional space. This challenge has given rise to the recently emerging area of alternative clustering analysis, whose goal is to seek different partitions (or clusterings), in order to describe different grouping aspects for a given dataset (e.g. [22, 2, 8, 9]).

For example, consider the data given in Figure 1 and assume the number of clusters to be uncovered is 3. It is clear that both of the clustering solutions found in two Figures 1a and 1b are equally valid and important, since they fit the data well and have the same clustering quality. It would be difficult to justify keeping only the first clustering, while omitting the

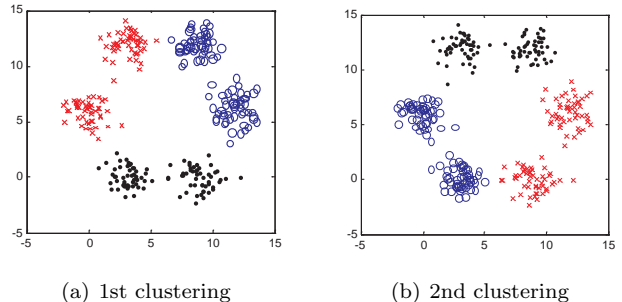

(a) 1st clustering   (b) 2nd clustering

Figure 1: Two alternative clusterings of the same dataset, each with 3 clusters. Point shapes show cluster membership.

second. We can also identify similar examples in real-life applications. For example, in text mining, one can cluster documents by their subjects or writing styles; or in biology, proteins might be classified by either their structure or function. In each case, both clustering solutions are equally important and each could be used to provide a different interpretation of the data.

In this paper, we develop a new and mathematically appealing algorithm called CAMI (Clustering for Alternatives with Mutual Information), to simultaneously uncover a pair clusterings from a dataset. Essentially, it is a regularized expectation maximization technique, which *maximizes* the likelihood of each of the alternative clusterings over the data, while at the same time, *minimizes* the similarity between them. To quantify the dissimilarity between two clustering solutions, we go beyond first and second order statistics by using mutual information based on Shannon's entropy theory [7]. Our motivation here is that mutual information can effectively capture the information shared between two distributions and it fully utilizes the information contained in the data as nonlinearly specialized by the probability density function [7, 27]. Furthermore, by formulating each clustering solution as a mixture model of multivariate normal distributions, we exploit the convolution property of Gaussian kernels and provide an algorithm which leads to a simple form of dual objective optimization function. Consequently, both the E- and M-steps

---

*Department of Computer Science and Software Engineering. The University of Melbourne, Australia (xdang@csse.unimelb.edu.au).

†Department of Computer Science and Software Engineering. The University of Melbourne, Australia (jbailey@csse.unimelb.edu.au).

are an appealing extension of the conventional EM algorithm and they maintain its advantages in computation and implementation.

An important aspect of our approach is that the CAMI algorithm is completely unsupervised. It does not require any a-priori knowledge to search for the various alternate clusterings, except the number of clusters M. This distinguishes it from the majority of previous work in the area, which targets the semi-supervised setting, where either a known clustering must be explicitly provided [8, 9], or some form of side-information is required to support the searching of an alternative one [6, 14]. Such semi-supervised approaches may not always be feasible, since clustering is often used as the first step in data analysis and extra information may be hard to obtain beforehand.

Our contributions in this paper are:

1. We develop and propose a new algorithm, CAMI, to simultaneously discover two alternative clusterings of a dataset. It combines the advantages of two mathematically sound frameworks: information theory and the framework of maximum likelihood. CAMI seeks to optimise two criteria 1) each clustering has high quality of clustering (in terms of maximum likelihood), 2) the mutual information between them is minimized.

2. Unlike nearly all existing techniques, CAMI is a completely unsupervised learning technique. It is the second work in this line but the first one addressing the problem from information theory.

3. We experimentally compare CAMI with seven well known algorithms on both synthetic and real-world datasets. Our results show that CAMI's performance is comparatively very strong and it is very promising to use as the basis for alternative clustering discovery.

2   Related work

Semi Supervised Alternative Clustering: Semi-supervised clustering has been extensively studied in the literature [24, 3, 16, 10]. Most of the existing techniques fall into two general approaches, which use a-priori knowledge as either positive or negative information toward the desired clustering. In the former case, prior knowledge (typically expressed in instances' must-link and cannot-link constraints [18, 24, 4]) is used to improve clustering results by guiding algorithms toward the target clustering. In the latter case, which is more related to our problem, prior knowledge takes the form of negative information about the desired clustering. In [15], authors proposed a conditional information bottleneck (CIB) method which treats class labels of a given clustering as side information in seeking an alternative clustering. The underlying principle of CIB can be summarized as follows. Given two variables X such as data objects and Y as features, CIB attempts to find a clustering C such that the shared information between X and C is minimized, while at the same time the information between Y and C is maximized conditioning on the information provided by the variable Z which represents provided class labels .[1] Although our work adopts an information theoretic approach, there are key differences between CIB and our approach. First, CIB is a semi-supervised algorithm, requiring background information for one of the alternative clusterings, whereas our approach is unsupervised. Second, while our algorithm *directly* minimizes the mutual information between the two clusterings, CIB only *conditions* on the first clustering, while maximizing the mutual information between C and Y. In other words, it uses mutual information in a completely different way to our approach. Subsequent to CIB, the COALA was proposed in [2]. Given a known clustering, COALA generates a set of pairwise cannot-link constraints and it attempts to find a disparate data partition by using these constraints within an agglomerative clustering process. COALA was shown to be very effective with both high qualitative and dissimilar clusterings discovered [2], though the quadratic running time of the hierarchical clustering can be a concern for large datasets.

The line of work developed in [8, 9] takes another approach to alternative clustering, based on the notion of orthogonality. In [8], the authors develop two techniques to find an alternative clustering using orthogonal projections. Intuitively, one can characterize a data partition by a set of representatives (e.g., cluster means). It is then possible to expect that a dissimilar partition might be found by clustering the data in a space orthogonal to the space spanned by such representatives. Both techniques developed in [8] exploit this idea. A similar approach is further developed in [9]. This method is better than [8] in that it does not suffer from the problem in which the data dimension can be smaller than the number of clusters (e.g., spatial datasets). Very recently, this approach has been further extended [21], where the transformation matrix can be regularized by constraints so that a good cluster in the original clustering can still be found in the transformed space.

Unlike these orthogonal projection techniques, our approach is unsupervised. Another key difference is that

---

[1]The CIB's minimization function is therefore $F = I(X, C) - \beta I(Y, C|Z)$, where β is a trade-off factor.

our approach seeks alternative clusterings in the original data space (*not an orthogonal one*). Moreover, rather than imposing the strict condition of orthogonality, we minimize the information sharing amongst clusterings; whilst maximizing the likelihood over the data in order to ensure the clustering quality. Notice that, as has been mentioned in [9] and confirmed in our experiments, orthogonality is quite a strong requirement and imposing transformations to satisfy orthogonality does not always ensure clusters can be discriminated better, since data can be distorted by the transformation operation. In addition, once the data is transformed for orthogonality, it may be difficult to interpret the alternative clusterings, since they are represented using completely new co-ordinates.

Unsupervised Alternative Clustering: Different from the semi-supervised setting approaches, interesting recent work in [17] proposes two algorithms to find disparate clusterings in an unsupervised manner. In their first algorithm, the concept of representative vectors is introduced for each clustering solution. The objective function of the k-means method is then subsequently modified by adding terms to account for the orthogonality between mean vectors of one clustering, with respect to the representative vectors of the other. In the second algorithm, it is assumed that the data can be modelled as a sum of mixtures and they associate each mixture with a clustering solution. This leads to the problem of learning a convolution of mixture distributions by which the expectation maximization method can be employed to find the distributions' parameters. Our work is similar to [17] in that we both deal with the problem from the unsupervised learning approach. However, the objective function used by our approach is rather different from the one in [17]. We address the problem from an information theory angle and attempt to minimize the mutual information between clustering solutions. Their work, on the other hand, is not based on mutual information and instead uses dot products to measure and optimise orthogonality between alternative clustering solutions. This use of orthogonality makes their objective function a little similar to some of the semi-supervised settings we have discussed earlier. We provide an experimental comparison later in the paper.

3 CAMI Algorithm Principles

An intuitive problem statement is as follows:

DEFINITION 3.1. *Given a dataset D and a user supplied parameter k, return two (alternative) clusterings C^+ and C^- of D, such that C^+ and C^- are each of high quality, they each have k clusters and the similarity*

between C^+ and C^- is low.

We will later be more precise about what is meant by high quality and high dissimilarity. We begin our description of the CAMI algorithm by reviewing some background.

3.1 Background on General EM theory

In statistical learning, both the greatest gradient and maximum likelihood can be used to estimate the parameters of a density mixture model [20]. Under the framework of maximum likelihood, one maximizes the following log-likelihood function:

$$(3.1) \qquad L(\Theta; \mathcal{X}) = \sum_{i=1}^{N} \log p(x_i | \Theta)$$

where the set of observations in d–dimensional space $\mathcal{X} = \{x_n\}_{n=1}^{N} \subseteq \mathbb{R}^d$ is assumed to be independently drawn from the distribution $p(x|\Theta)$ parameterized by Θ. The function $L(\Theta; \mathcal{X})$ is sometimes called the likelihood of the parameters Θ given the fixed data $\mathcal{X}$. The goal is to find the Θ that maximizes L. The Expectation Maximization (EM) algorithm is a powerful technique to iteratively compute the maximum likelihood when the data observations $\mathcal{X}$ are viewed as incomplete data and the likelihood function can be simplified by assuming the existence of additional but missing data $\mathcal{Y}$ (corresponding to $\mathcal{X}$). With this addition, the complete data is the combination of $\mathcal{X}$ and $\mathcal{Y}$ and the technique tries to maximize the likelihood of this combined data via the iteration of computing two involved steps, named the E- and M-steps. Particularly, in the E-step, the algorithm determines the expectation of the complete data log-likelihood based on the observations $\mathcal{X}$ and the current parameter Θ_t:

$$(3.2) \qquad Q(\Theta|\Theta_t) = E[\log p(\mathcal{X}, \mathcal{Y}|\mathcal{X}, \Theta_t)]$$

In the M-step, it determines a new parameter by maximizing this expectation:

$$(3.3) \qquad \Theta_{t+1} = \arg\max_{\Theta} Q(\Theta|\Theta_t)$$

3.2 Alternative Clustering Objective Function Using Mutual Information

The clustering problem can be viewed as a special case of estimating parameters for a density mixture model. For this approach, one can treat each clustering solution as a mixture of models and the class label C

of data observations plays the role of the missing data $\mathcal{Y}$ in the EM technique. Each model (or distribution) in the mixture corresponds to a cluster and the parameters of each distribution provides a description of the corresponding cluster.

For our problem, given the set of data samples $\mathcal{X} = \{x_1, x_2, ...x_N\}$, we aim to discover two clustering solutions C^+ and C^-, each respectively partitions $\mathcal{X}$ as a whole into M^+ and M^- groups and the similarity between them is to be minimized. C^+ and C^- are parameterized by Θ^+ and Θ^- respectively. Let Θ be the combination parameters of Θ^+ and Θ^-, we therefore define the likelihood function in our case as follows:

(3.4)
$$\widetilde{L}(\Theta; \mathcal{X}) = L(\Theta^+; \mathcal{X}) + L(\Theta^-; \mathcal{X}) + \eta F(C^+; C^- | \Theta)$$

The first two terms on the right hand side of the equation correspond to maximum likelihood functions with respect to each clustering C^+ and C^-, while the third term F measures their dissimilarity. $\eta > 0$ is a regularization parameter which controls the compromise between the degree of the difference and the maximum likelihood.

At this stage, it is natural to ask whether it is reasonable to combine both quality and dissimilarity terms together in the objective function - i.e. do they measure comparable quantities ? Looking ahead, we will shortly see that all three terms of the objective can be expressed in terms of quantities related to probabilities. Hence the objective can loosely be interpreted as a combination of probabilities, with an additional regularization parameter used to provide scaling.

Since we would like to find two partitions over the data that are as disparate as possible, the function F needs to be an effective tool for measuring clustering dissimilarity. We make use of mutual information, since it has been known to exploit well the dependence between components, even in nonlinear cases [27].

In order to formally define mutual information, we need to make use of the quantity entropy. Mathematically, the entropy of a continuous random variable X with probability density function $p(x)$ is defined by:

(3.5)
$$H(X) = - \int p(x) \log p(x) \, dx$$

Entropy can be interpreted as the number of bits on average required to describe a random variable. The negative of entropy is sometimes called information ($I(X) = -H(X)$). The entropy for one random variable also extends to the case of two random variables (joint entropy):

$$H(X, Y) = - \iint p(x, y) \log p(x, y) \, dx \, dy$$

A related concept to the entropy is the Kullback-Leibler divergence. It is a measure of the distance between two distributions $p(x)$ and $q(x)$ and is defined by:

(3.6)
$$KL(p \| q) = \int p(x) \log \frac{p(x)}{q(x)} \, dx$$

Mutual information turns out to be a special case of the KL divergence. It measures the information shared between two objects or in other words, accounts for the amount of information that one random variable contains about another variable. Consider two variables X and Y with a joint probability density function $p(x, y)$ and marginal probability density functions $p(x)$ and $p(y)$, the mutual information $I(X; Y)$ is the KL distance between the joint distribution and the product distribution $p(x)p(y)$:

$$\begin{aligned} I(X; Y) &= \iint p(x, y) \log \frac{p(x, y)}{p(x)p(y)} \, dx \, dy \\ &= KL(p(x, y) \| p(x)p(y)) \end{aligned}$$

which is obviously symmetric. Moreover, two random variables have zero mutual information if and only if they are statistically independent.

By using mutual information as a measure to compute the dissimilarity (or independence) between two clustering solutions C^+ and C^-, we wish to minimize the mutual information sharing between them. Therefore, we regularize our likelihood optimization function with this amount as penalty. The objective function now becomes:

$$\widetilde{L}(\Theta; \mathcal{X}) = L(\Theta^+; \mathcal{X}) + L(\Theta^-; \mathcal{X}) - \eta I(C^+; C^- | \Theta)$$

Under the assumption of independence of partitions, mutual information can be written as the sum of pair-wise mutual information between two clusterings. We decompose the third term from the right hand side into a sum of components, yielding:

(3.7)
$$\widetilde{L}(\Theta; \mathcal{X}) = L(\Theta^+; \mathcal{X}) + L(\Theta^-; \mathcal{X}) - \eta \sum_{i,j} I(c_i^+; c_j^- | \theta_{ij})$$

where c_i^+ is the ith cluster of the first clustering C^+ and c_j^- is the jth cluster of the second clustering C^-. The parameter θ_{ij} is again the combination of θ_i^+ and θ_j^- (respectively represented for c_i^+ and c_j^-). We therefore have exploited pairwise mutual information in order to measure the statistical dependence between two clusters, each from a separate clustering.

The degree to which cluster c_i^+ contributes to C^+ is determined by the information sharing between cluster

c_i^+ and all other clusters in the clustering C^-. If c_i^+ has its pairwise mutual information minimized, it is likely to be independent from all clusters of clustering C^-. Therefore, we define the mutual information of cluster c_i^+ with respect to all the other clusters in C^- as:

$$(3.8) \quad I(c_i^+; C^-) = \sum_{c_j^- \in C^-} p(c_i^+, c_j^-) \log \frac{p(c_i^+, c_j^-)}{p(c_i^+)p(c_j^-)}$$

We further observe that since $I(X;Y) = H(X) + H(Y) - H(X,Y) = I(X,Y) - I(X) - I(Y)$, minimizing the mutual information between two objects X and Y is equivalent to minimizing the joint information $I(X,Y) = \iint p(x,y) \log p(x,y) dx dy$. That means the a-priori probability of each cluster can be omitted in the formula of pairwise mutual information above (i.e., Eq. 3.8), without compromising our optimization problem. Our objective function can therefore finally be written as follows:

$$(3.9)$$
$$\widetilde{L}(\Theta; \mathcal{X}) = L(\Theta^+; \mathcal{X}) + L(\Theta^-; \mathcal{X}) - \eta \sum_{i,j} p(c_i^+, c_j^-) \log p(c_i^+, c_j^-)$$

3.3 Learning Parameters

We will consider the alternative clustering problem using the framework of mixture models, which are a powerful tool for probabilistic modeling of data. In addition to their ability to represent complex density functions which are extensively used in various density estimation problems, such models also provide a principled probabilistic approach to cluster data. We employ Gaussian mixtures for our models and thus, each cluster is represented by a single multivariate Gaussian component within the mixture. Specifically, the mixture probability density functions for two clusterings C^+ and C^- have the form:

$$p(x|\Theta^+) = \sum_{i=1}^{M^+} \lambda_i^+ p(x|\theta_i^+) = \sum_{i=1}^{M^+} \lambda_i^+ \mathcal{N}(x - \mu_i^+, \Sigma_i^+)$$

$$p(x|\Theta^-) = \sum_{j=1}^{M^-} \lambda_j^- p(x|\theta_j^-) = \sum_{j=1}^{M^-} \lambda_j^- \mathcal{N}(x - \mu_j^-, \Sigma_j^-)$$

where $\Theta^+ = \{\lambda_{1,\dots,M^+}^+, \theta_{1,\dots,M^+}^+\}$ are parameters corresponding to clustering solution C^+. Each $p(x|\theta_i^+)$ is a multivariate Gaussian density of the component c_i^+ that is parameterized by $\theta_i^+ = (\mu_i^+, \Sigma_i^+)$, where μ_i^+ is the distribution's center and Σ_i^+ is its covariance matrix. λ_i^+'s are the mixing coefficients and they are subject to the condition $\sum_i \lambda_i^+ = 1$. Each λ_i^+ can also be thought of as the a priori probability of each component in the mixture; i.e., $\lambda_i^+ = p(c_i^+)$. A similar explanation applies to clustering C^- with respect to the set of parameters Θ^-.

Given the system setting above, we rewrite the objective function in a more specific form:

$$(3.10)$$
$$\widetilde{L}(\Theta; \mathcal{X}) = \sum_x \log \sum_{i=1}^{M^+} \lambda_i^+ p(x|\theta_i^+) + \sum_x \log \sum_{j=1}^{M^-} \lambda_j^- p(x|\theta_j^-)$$
$$- \eta \sum_i \sum_j p(c_i^+, c_j^-) \log p(c_i^+, c_j^-)$$

By conditioning on the data observations x, the joint distribution of the two components c_i^+ and c_j^- appearing in the last term of the equation can be factorized as:

$$(3.11) \qquad p(c_i^+, c_j^-) = \int p(c_i^+|x) p(c_j^-|x) p(x) dx$$

which implies c_i^+ and c_j^- being statistically independent given the observation x. [2] According to Bayesian theory, it follows that:

$$(3.12) \quad \int p(c_i^+|x) p(c_j^-|x) p(x) dx$$
$$= \int \frac{p(c_i^+) p(x|c_i^+)}{p(x)} \frac{p(c_j^-) p(x|c_j^-)}{p(x)} p(x) dx$$
$$\geq \frac{p(c_i^+) p(c_j^-) \int p(x|c_i^+) p(x|c_j^-) dx}{\int p(x) dx}$$

Notice that $\int p(x) dx = 1$ which is a constant, we thus optimize the lower bound of $p(c_i^+, c_j^-)$ instead (i.e., the numerator of the formula). Our strategy of optimizing the lower bound of the objective is in line with the philosophy of the standard EM algorithm, which also optimizes a lower bound.

On the other hand, the integration term in the numerator essentially is the convolution of two Gaussian kernels, which interestingly has a very simple form [23, 12]:

$$(3.13) \qquad \int \mathcal{N}(x - \mu_i^+, \Sigma_i^+) \mathcal{N}(x - \mu_j^-, \Sigma_j^-) dx$$
$$= \mathcal{N}(\mu_i^+ - \mu_j^-, \Sigma_i^+ + \Sigma_j^-)$$

Observe that the integral computation was replaced by the evaluation of the Gaussian kernel at the location $\mu_i^+ - \mu_j^-$. Hence, the information sharing between two clustering solutions can be estimated as a sum of local interactions, as defined by the kernel, over all pairs of clusters between them. This result can also be interpreted in terms of physics where one makes the analogy between mean vectors (representatives for the

---

[2] In general, Eq. (3.11) does not factorize into $p(c_i^+)p(c_j^-)$, see [5].

clusters) and "physical particles". $\mathcal{N}(\mu_i^+ - \mu_j^-, \Sigma_i^+ + \Sigma_j^-)$ can be interpreted as the potential energy of the vector mean μ_i^+ in the potential field of vector mean μ_j^-, or vice versa. We may call this potential energy an *information potential* and it corresponds to the potential field in physics. Therefore, optimizing the information sharing between two clusterings is equivalent to optimizing the sum of these information potentials, which leads to achieving an equilibrium state, as in the case of moving particles in physics.

Substituting Eq. (3.13) into the objective function, the following equation is attained:

(3.14)

$$\widetilde{L}(\Theta; \mathcal{X}) = \sum_x \log \sum_{i=1}^{M^+} \lambda_i^+ p(x|\theta_i^+) + \sum_x \log \sum_{j=1}^{M^-} \lambda_j^- p(x|\theta_j^-)$$
$$- \eta \sum_{i=1}^{M^+} \sum_{j=1}^{M^-} p(c_i^+, c_j^-) \log \lambda_i^+ \lambda_j^- \mathcal{N}(\mu_i^+ - \mu_j^-, \Sigma_i^+ + \Sigma_j^-)$$

We do not replace the first $p(c_i^+, c_j^-)$ for a reason clear later. For simplicity, we present the computations for the set of parameters Θ^+ of clustering C^+, the computations for Θ^- are completely analogous.

E-step:

The expectation step of the EM algorithm can be divided into two terms, one is the expectation associated with the likelihood:

(3.15) $p(c_i^+|x_n, \Theta_t^+) = \dfrac{\lambda_i^+ \mathcal{N}(x_n - \mu_i^+, \Sigma_i^+)}{\sum_m \lambda_m^+ \mathcal{N}(x_n - \mu_m^+, \Sigma_m^+)}$

The other is the expectation related to the mutual relationship:

(3.16)

$$p(c_i^+|c_j^-, \Theta_t^+) = \dfrac{\lambda_i^+ \lambda_j^- \mathcal{N}(\mu_j^- - \mu_i^+, \Sigma_j^- + \Sigma_i^+)}{\sum_m \lambda_m^+ \lambda_j^- \mathcal{N}(\mu_j^- - \mu_m^+, \Sigma_j^- + \Sigma_m^+)}$$

Notice that as the summations of both $p(c_i^+|x_n, \Theta_t^+)$ and $p(c_i^+|c_j^-, \Theta_t^+)$ over i are equal to 1 and $-\log(x)$ is convex, we can apply Jensen's inequality to derive the new bound for the objective function. In addition, the information term is normalized by the same amount of $\sum_m \lambda_m^+ \lambda_j^- \mathcal{N}(\mu_j^- - \mu_m^+, \Sigma_j^- + \Sigma_m^+)$, the corresponding $\widetilde{Q}$ function is then derived as follows:[3]

(3.17)

$$\widetilde{Q}(\Theta^+|\Theta_t^+) = \sum_{n=1}^{N} \sum_{i=1}^{M^+} p(c_i^+|x_n) \log \dfrac{\lambda_i^+ \mathcal{N}(x_n - \mu_i^+, \Sigma_i^+)}{p(c_i^+|x_n)}$$
$$- \eta \sum_{i,j} p(c_i^+|c_j^-) \log \lambda_i^+ \lambda_j^- \mathcal{N}(\mu_j^- - \mu_i^+, \Sigma_i^+ + \Sigma_j^-)$$

---
[3]To save space, we omit the term Θ_t^+ in $p(.|.)$.

Deploying the logarithm in the first sum and noticing the term $\sum_{n=1}^N \sum_{i=1}^{M^+} p(c_i^+|x_n) \log p(c_i^+|x_n)$ can be omitted due to the availability of the membership probabilities $p(c_i^+|x_n)$ computed in the E-step. Optimizing $\widetilde{Q}(\Theta^+|\Theta_t^+)$ is thus the same as optimizing the summations:

(3.18)

$$\widetilde{Q}(\Theta^+|\Theta_t^+) = \sum_{n=1}^{N} \sum_{i=1}^{M^+} p(c_i^+|x_n) \log \lambda_i^+ \mathcal{N}(x_n - \mu_i^+, \Sigma_i^+)$$
$$- \eta \sum_{i=1}^{M^+} \sum_{j=1}^{M^-} p(c_i^+|c_j^-) \log \lambda_i^+ \lambda_j^- \mathcal{N}(\mu_j^- - \mu_i^+, \Sigma_i^+ + \Sigma_j^-)$$

M-step:

The M-step involves more computation. First, to find the expression for λ_i^+ we use the Lagrange optimization method subject to the constraint $\sum_i \lambda_i^+ = 1$, and solve the following function (with α is the Lagrange multiplier):

$$\dfrac{\partial}{\partial \lambda_i^+} \left[\widetilde{Q}(\Theta^+|\Theta_t^+) + \alpha \left(\sum_i \lambda_i^+ - 1 \right) \right] = 0$$

or

$$\dfrac{\partial}{\partial \lambda_i^+} \left[\sum_{n=1}^{N} p(c_i^+|x_n) \log \lambda_i^+ - \eta \sum_{j=1}^{M^-} p(c_i^+|c_j^-) \log \lambda_i^+ \lambda_j^- \right] = -\alpha$$

$$\sum_{n=1}^{N} \dfrac{1}{\lambda_i^+} p(c_i^+|x_n) - \eta \sum_{j=1}^{M^-} \dfrac{1}{\lambda_i^+} p(c_i^+|c_j^-) = -\alpha$$

Taking the derivative with respect to all other λ_i^+'s and summing both sides over i gives us:

$$\sum_{n=1}^{N} \sum_{i=1}^{M^+} p(c_i^+|x_n) - \eta \sum_{j=1}^{M^-} \sum_{i=1}^{M^+} p(c_i^+|c_j^-) = -\alpha \sum_{i=1}^{M^+} \lambda_i^+$$
$$N - \eta M^- = -\alpha$$

which results in

(3.19) $\lambda_i^+ = \dfrac{\sum_{n=1}^{N} p(c_i^+|x_n) - \eta \sum_{j=1}^{M^-} p(c_i^+|c_j^-)}{N - \eta M^-}$

The new estimates for the mean vectors can be easily obtained by taking the derivative of the function with respect to μ_i^+:

(3.20)

$$\mu_i^+ = \dfrac{\sum_n p(c_i^+|x_n)/(\Sigma_i^+)x_n - \eta \sum_j p(c_i^+|c_j^-)/(\Sigma_i^+ + \Sigma_j^-)\mu_j^-}{\sum_n p(c_i^+|x_n)/(\Sigma_i^+) - \eta \sum_j p(c_i^+|c_j^-)/(\Sigma_i^+ + \Sigma_j^-)}$$

In order to get a new estimate of the covariance matrix Σ_i^+, we need to take the derivative of Eq. (3.18)

with respect to Σ_i^+. However, observe that the derivative of $\widetilde{Q}$ with respect to Σ_i^+ cannot be solved directly due to the existence of the inverse matrix $(\Sigma_i^+ + \Sigma_j^-)^{-1}$ appearing in the Gaussian kernel. One solution is to use the Cauchy-Schwartz inequality to find a new bound for the function. Particularly, since the Gaussian kernel is always nonnegative, we can write (based on the Cauchy-Schwartz inequality):

$$\frac{1}{2} \times 2 \log\left(\mathcal{N}(\mu_j^- - \mu_i^+, \Sigma_i^+ + \Sigma_j^-)\right)$$

$$= \frac{1}{2} \log\left(\int \mathcal{N}(x - \mu_i^+, \Sigma_i^+)\mathcal{N}(x - \mu_j^-, \Sigma_j^-)dx\right)^2$$

$$\leq \frac{1}{2} \log \int \left(\mathcal{N}(x - \mu_i^+, \Sigma_i^+)\right)^2 dx \int \left(\mathcal{N}(x - \mu_j^-, \Sigma_j^-)\right)^2 dx$$

$$= \frac{1}{2} \log \mathcal{N}(0, 2\Sigma_i^+)\mathcal{N}(0, 2\Sigma_j^-)$$

It follows that the bound for the covariance matrix is given by:

$$\widetilde{Q}(\Theta^+|\Theta_t^+)_{\Sigma_i^+} = \sum_{n=1}^{N} \sum_{i=1}^{M^+} p(c_i^+|x_n) \log \lambda_i^+ \mathcal{N}(x_n - \mu_i^+, \Sigma_i^+)$$

$$- \frac{\eta}{2} \sum_{i=1}^{M^+} \sum_{j=1}^{M^-} p(c_i^+|c_j^-) \log \lambda_i^+ \lambda_j^- \mathcal{N}(0, 2\Sigma_i^+)\mathcal{N}(0, 2\Sigma_j^-)$$

Taking the derivative of this bound with respect to Σ_i^+ and let it equal to 0, the new estimate for the covariance matrix is followed:

$$\frac{\partial}{\partial \Sigma_i^+}\left(\sum_{n=1}^{N} \sum_{i=1}^{M^+} p(c_i^+|x_n) \log \lambda_i^+ \mathcal{N}(x_n - \mu_i^+, \Sigma_i^+)\right.$$

$$\left. - \frac{\eta}{2} \sum_{i=1}^{M^+} \sum_{j=1}^{M^-} p(c_i^+|c_j^-) \log \lambda_i^+ \lambda_j^- \mathcal{N}(0, 2\Sigma_i^+)\mathcal{N}(0, 2\Sigma_j^-)\right)$$

$$= \sum_{n=1}^{N} p(c_i^+|x_n) \left(\frac{-1}{2\Sigma_i^+} + \frac{1}{2\Sigma_i^+}(x_n - \mu_i^+)(x_n - \mu_i^+)^T \frac{1}{\Sigma_i^+}\right)$$

$$- \frac{\eta}{2} \sum_{j=1}^{M^-} p(c_i^+|c_j^-) \left(-\frac{1}{2\Sigma_i^+}\right) = 0$$

or

$$(3.21) \quad \Sigma_i^+ = \frac{\sum_{n=1}^{N} p(c_i^+|x_n)(x_n - \mu_i^+)(x_n - \mu_i^+)^T}{\sum_{n=1}^{N} p(c_i^+|x_n) - \frac{\eta}{2}\sum_{j=1}^{M^-} p(c_i^+|c_j^-)}$$

The role of η: The regularization parameter η acts as a trade-off between how well our algorithm maximizes the log likelihood function and how much it minimizes the similarity between the two clustering solutions. If η is set very high, we favor dissimilarity between the two clusterings over the log likelihood of each individual clustering. The clustering solutions obtained therefore may not fit the data well. Conversely, for a small value of η, the algorithm may return two clustering solutions which are almost identical, and in fact it just becomes the plain EM if η is set to 0. Empirically, we have found that treating this parameter like learning rates in a neural network's training, with the initial value of 15% of the dataset size, produces generally good results (further information is provided in the experimental evaluation). Specifically, we start the algorithm with η initialized at 15% of the dataset size and track the change in objective function. Once the change in objective function becomes sufficiently small, η is thereafter decreased, say to 90% of its value at each iteration, and thus approached to 0 (which also ensures the convergence of the algorithm). The intuition is that we expect each cluster is initially formed in a subspace in which it is mostly independent from those in the opposite clustering. Once such subspaces have been identified, η can then be reduced and the algorithm will converge to solutions which are optimal in terms of likelihood maximization.

4 Experimental Results

We next provide experimental results on both synthetic and real-world datasets. We compare CAMI against seven alternative clustering algorithms: two methods developed in [17] named Dec-kmeans and Conv-EM, the CIB method [15], COALA [2], two methods from [8] denoted by Algo1 and Algo2, and the ADFT algorithm from [9]. For the trade-off factor used in Dec-kmeans and Conv-EM, we follow the heuristic method presented in [17]. With the CIB method, we implement the iterative version [13, 15] and its outputs are post-processed by assigning each data point to the cluster in which it has the highest probability. For ADFT, we implement the gradient descent method integrated with the iterative projection technique (in learning the full family of the Mahalanobis distance matrix) [25, 26]. Also, in order to make the comparison fair, we use the EM technique (instead of k-means) for the approaches developed in [8, 9]. We run each algorithm 10 times and report the average results.

4.1 Clustering Evaluation

We evaluate the clustering results based on both clustering dissimilarity and clustering quality measures. For measuring dissimilarity between two clusterings, we report the values of two different measures. The first is the normalized mutual information [11, 19, 16, 22] and it is defined by: $NMI(C^+; C^-) = I(C^+; C^-)/(H(C^+)H(C^-))$, where $I(C^+; C^-) = \sum_{i=1}^{M^+} \sum_{j=1}^{M^-} \frac{N_{ij}}{N} \log\left(\frac{N.N_{ij}}{N_i^+.N_j^-}\right)$ with N_{ij} denoting the number of shared instances between clus-
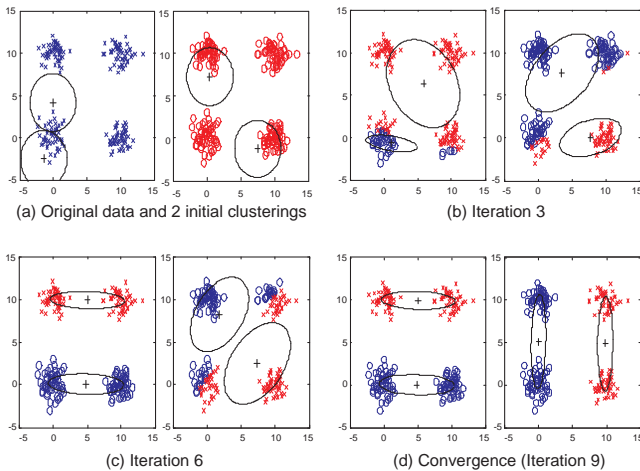
Figure 2: Results for Synthetic1(Syn1) dataset using CAMI.

	NMI	JI	DI	NMI	JI	DI
Methods	Syn1			Syn2		
Dec-kmean	0.12	0.39	2.2	0.39	0.34	2.1
Conv-EM	0.12	0.4	2.15	0.4	0.36	1.9
CIB	0.12	0.4	2.2	0.41	0.39	1.71
COALA	**0**	**0.33**	**2.37**	0.38	0.35	1.31
ADFT	0.12	0.39	2.23	0.62	0.61	2.3
Algo1	0.25	0.41	1.95	0.42	0.41	1.37
Algo2	0.26	0.43	1.9	NA	NA	NA
CAMI	0.1	0.38	2.25	**0.37**	**0.33**	**2.73**

Table 1: Results on synthetic datasets. DI measures quality (higher is better) and NMI and JI measure similarity (lower is better)

ters $c_i^+ \in C^+$ and $c_j^- \in C^-$. The second is the Jaccard index (JI), which was used in [2, 9] to evaluate the dissimilarity between clusterings.

For measuring clustering quality, we use the Dunn Index, similar to [2, 9]: $DI(C) = \frac{\min_{i \neq j} \{ \delta(c_i, c_j) \}}{\max_{1 \leq \ell \leq k} \{ \triangle(c_\ell) \}}$ where C is a clustering, $\delta: C \times C \to \mathbb{R}_0^+$ is the cluster-to-cluster distance and $\triangle: C \to \mathbb{R}_0^+$ is the cluster diameter measure.

Note that for the NMI and JI measures, *a smaller value is desirable*, indicating higher dissimilarity between clusterings, while for the DI measure, *a larger value is desirable*, indicating a better clustering quality.

Methodology: Recall that CAMI is an unsupervised technique, meaning that it does not require any input clustering to be provided. We will compare CAMI against two classes of alternative clustering algorithms i) unsupervised (Dec-kmeans, Conv-EM) and ii) semi-supervised (COALA, CIB, Algo1, Algo2 and ADFT). In order to compare with the semi-supervised techniques, we need to provide one clustering as input. To try and achieve a fair comparison, we input the higher quality clustering generated by CAMI to each semi-supervised technique. CAMI is then compared against the semi-supervised technique in terms of the quality of the second clustering and the dissimilarity between the two clusterings.

4.2 Synthetic Datasets

For the first synthetic dataset, we use a popular one from [2, 8, 9], consisting of 4 Gaussian sub-classes. Each Gaussian contains 100 2-dimensional data points

and the point distribution is shown in Figure 2a. By setting the number of clusters $M = 2$, the goal of using this dataset is to test whether our algorithm can discover a pair of orthogonal clusterings. Figure 2 shows some selective iterations of CAMI and a comparison of its final result against other techniques is listed in Table 1 (under Syn1). Similar to other techniques, our algorithm works well on this dataset, being able to uncover two disparate clusterings. Looking at Table 1, we also see that the average dissimilarity and quality of CAMI is slightly better than CIB and two other unsupervised techniques Dec-kmeans and Conv-EM, but it is clearly better than Algo1 and Algo2, which we have found to be more sensitive to the initial parameters when clustering in the transformed space. Of the existing methods, COALA seems to have the best performance and its result is stable across all trials. This is justified by the COALA's hierarchical clustering approach.

For the second synthetic dataset, we use a more complicated scenario in which six Gaussians are generated and positioned in a ring shape as depicted in Figure 1. By setting the number of clusters in each solution to be 3, this dataset clearly contains two equally high quality, yet dissimilar clusterings. In Figure 3, we show the most popular clustering outputs of all algorithms from 10 running times. For semi-supervised algorithms, the clustering from Figure 1a was provided as background information to guide the alternative clustering process. The corresponding measures are reported in Table 1 (under Syn2). We note that the average Dunn Index values reported in the Table (for both Syn1 and 2) are computed based on the two clusterings outputted by un-supervised algorithms, while these values for semi-supervised algorithms are based on the (single) alternative clustering found. Also, the transformation performed by Algo2 in [8] is undefined for this experi-

(a) 2 clusterings returned by CAMI

(b) popular clusterings returned by Dec-kmeans

(c) 2nd clustering returned by CIB

(d) 2nd clustering returned by COALA

(e) 2nd clustering returned by ADFT

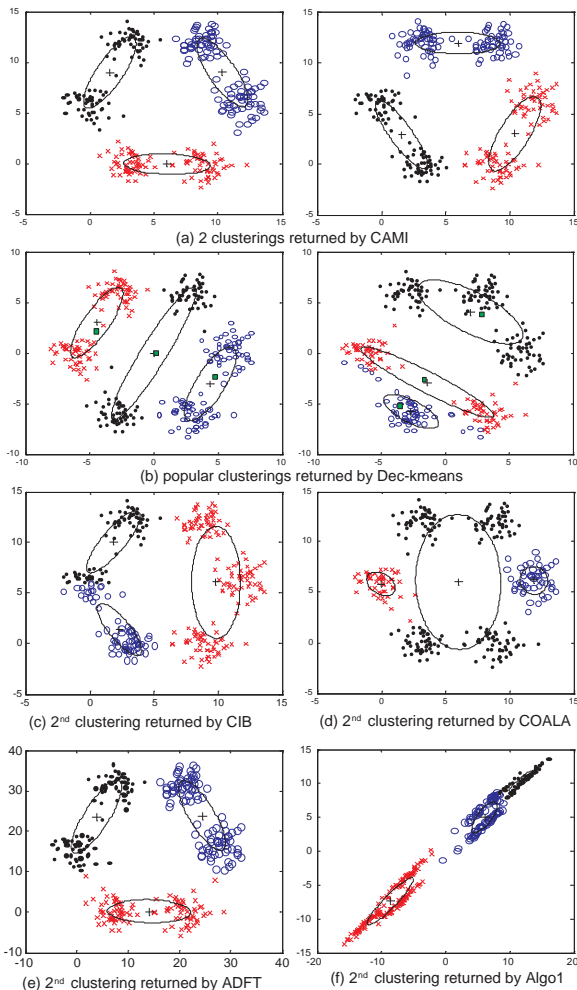(f) 2nd clustering returned by Algo1

Figure 3: Results on Synthetic2. Notice dimensions are scaled in ADFT and Algo1.

ment, since the number of clusters is greater than the data dimensionality.

We observe that while CAMI can easily uncover two dissimilar and high quality clusterings (see Figure 3a), all the other algorithms are less successful in discovering two equally important clusterings, even though the semi-supervised setting were actually provided a first accurate high quality clustering. It is possible to explain this result as follows. First, we found that the outputs of Dec-kmeans and Conv-EM to be quite similar, with Dec-kmeans being slightly better on average. A visualization of the clusterings returned by Dec-kmeans is shown in Figure 3b (blue square points are representative vectors). As observed from this figure, two resultant clusterings are quite dissimilar and orthogonal.

Nonetheless, we can also see that the clusterings are less natural, since points on opposite sides of the ring still being grouped together. This result confirms our values computed in Table 1 in which the normalized mutual information of both Dec-kmeans and Conv-EM is quite small but their Dunn Index is also relatively low.

Second, the alternative clustering uncovered by the CIB method also looks quite unnatural as shown in Figure 3c. This is perhaps because CIB purely relies on the mutual information for its clustering process and does not explicitly have a quality objective. This is also a fundamental difference between CIB and CAMI. With our algorithm, the clustering quality is still ensured by the expectation maximization technique.

The COALA is quite successful in uncovering a second dissimilar clustering, since both its normalized mutual information and Jaccard index are small. However, its clustering quality is rather poor, due to the imbalance amongst the clusters attained (shown in Figure 3d). This is probably because COALA has chosen sub-clusters far apart to merge in its high levels of agglomerative clustering process (to satisfy the cannot-link constraints learnt from the first provided clustering).

The clustering found by ADFT (in Figure 3e) is overly similar to the original clustering that was provided as background knowledge. Here, it seems the inversion of the stretcher matrix is not able to help in uncovering a dissimilar clustering. This might be due to that a stretcher matrix is intrinsically a diagonal matrix with entries acting as stretching factors along each dimension. And by the nature of the first clustering, the inversion (i.e., orthogonality) of such matrix in this case has simply scaled all dimensions with almost the same factor.

Finally, Algo1 has attempted to find the dissimilar clustering by projecting data on a space orthogonal to the mean vectors of the first provided clustering. Such a transformation, nonetheless, has distorted the data (as plotted in the Figure 3f) and consequently prevented it from deducing the correct hidden clustering. Moreover, although Algo1 seems to fit well the data in its transformed space, we observe that it is difficult to interpret the resultant clustering due to its unnatural shape visualized in the original data space.

This experiment highlights an interesting case where the output of CAMI is superior to the the existing methods. It also demonstrates well that various techniques based on orthogonal transformations or projections do not always ensure discovery of a dissimilar and high quality clustering. Conversely, CAMI's approach of maximizing the likelihood over the data (to meet quality criteria), while at the same time minimiz-

Figure 4: The average face image for each cluster found by CAMI. 1st and 2nd rows respectively represent for the 1st and 2nd clusterings

Methods	NMI	JI	DI
Dec-kmeans	0.25	0.31	1.65
Conv-EM	0.28	0.32	1.63
CIB	0.3	0.34	1.59
COALA	0.29	0.32	1.62
ADFT	0.33	0.35	1.54
Algo1	0.38	0.39	1.51
Algo2	0.41	0.42	1.45
CAMI	**0.21**	**0.29**	**1.68**

Table 2: Results on CMUFace dataset. DI measures quality (higher is better) and NMI and JI measure similarity (lower is better)

ing the mutual information sharing amongst clusterings (to ensure the dissimilarity) leads to a better result.

4.3 CMUFace Dataset

We have shown that our algorithm works well on synthetic datasets. In this section and the next one, we compare CAMI against the other algorithms on some real datasets. We begin with the CMUFace data from the UCI KDD repository [1]. This is an interesting dataset, since data instances can be partitioned in several different ways (e.g. by individual, by pose, etc.). The dataset consists of images of 20 people taken at various features such as facial expressions (neutral, happy, sad, angry), head positions (left, right or straight), and eye states (open or sunglasses). Each person has 32 images captured in every combination of these features. We randomly select 3 people along with all their images and run the CAMI algorithm with $M = 3$, to see if the clusterings found yield useful information. As a pre-processing step, we apply PCA

and use the dimensionality that retains at least 90% of the original data's variance.

After running CAMI with $M = 3$ on this dataset, we obtain two dissimilar clustering solutions. Clusters' means of each solution are shown in Figure 4. Graphically, it is possible to observe that the uncovered clusterings explain two distinct ways that the images are grouped. The clustering in the first row clearly shows that images are categorized into different persons, while the clustering in the second row reveals that they are grouped by different poses. Due to space constraints, we do not show the output pictures of other techniques. However, we compare them via the measures reported in Tables 2. We note that for semi-supervised algorithms, the clustering which was grouped by person was provided as background information (since this is the easier clustering to discover). The Dunn Index in Table 2 is therefore reported for the second clustering, which is expected to be based on different poses.

From Table 2, we see that all three unsupervised algorithms are performing well with this dataset. However, CAMI with its minimizing information sharing between clusterings is better than the unsupervised algorithms Dec-kmeans and Conv-EM. ADFT and COALA are better than Algo1 and 2, which find the alternative clustering in a completely transformed space. However, their performance is still worse than CAMI. CAMI is slightly better than COALA when the dissimilarity is measured in terms of Jaccard Index, but clearly better in terms of NMI. Its clustering quality measured using the Dunn Index is also better than that of COALA and considerably better than those of Algo1 and 2.

4.4 Other Real-World Datasets

We further compare the eight algorithms on three real-world datasets selected from the UCI repository: Vowel, Segmentation and Vehicle Silhouette. For the Segmentation dataset, three attributes 5,7 and 9 are removed as they were reported to be repetitive with attributes 4,6 and 8 [1]. We report the Dunn Index of both CAMI and these semi-supervised algorithms using the second clustering obtained from each dataset (recall that the higher qualitative clustering returned by CAMI is provided as background to other semi-supervised techniques). The results are shown in Table 3.

We also compare CAMI directly against the unsupervised algorithms Dec-kmeans and Conv-EM in Table 4. In this table, the quality reported is the average Dunn Index of the two clusterings that are outputted by the algorithms.

It can be seen that the clustering results of CAMI on these datsets are also consistently better than those of the other algorithms. In all cases, CAMI outperforms

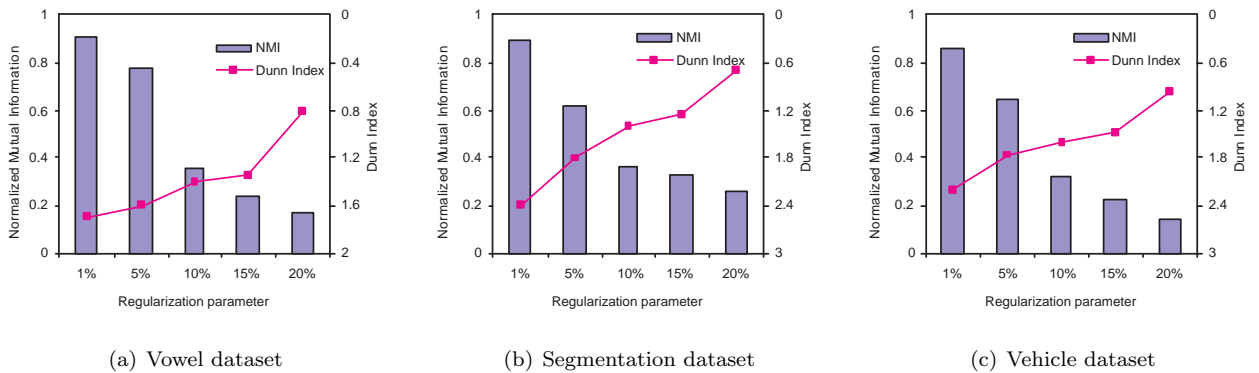(a) Vowel dataset   (b) Segmentation dataset   (c) Vehicle dataset

Figure 5: Impact on CAMI of varying the regularization parameter η on the clustering performance. For an ideal result, NMI should be low and Dunn index should be high.

Seg.	Algo1	Algo2	ADFT	COALA	CIB	CAMI
NMI	0.54	0.44	0.53	0.47	0.45	**0.36**
JI	0.39	0.31	0.37	0.34	0.31	**0.27**
DI	1.3	1.25	1.29	1.21	1.32	**1.4**
Veh.						
NMI	0.4	0.39	0.37	0.27	0.35	**0.23**
JI	0.4	0.44	0.39	0.34	0.42	**0.32**
DI	1.26	1.47	1.42	**1.54**	1.38	1.5
Vow.						
NMI	0.43	0.47	0.54	0.38	0.4	**0.24**
JI	0.2	0.22	0.38	0.28	0.26	**0.11**
DI	1.24	1.29	**1.4**	1.3	1.25	1.33

Table 3: Results on three real world datasets for CAMI versus semi-supervised algorithms. DI measures quality (higher is better) and NMI and JI measure similarity (lower is better)

Seg.	Dec-kmeans	Conv-EM	CAMI
NMI	0.39	0.41	**0.36**
JI	0.29	0.3	**0.27**
DI	1.26	1.28	**1.45**
Veh.			
NMI	0.26	0.25	**0.23**
JI	0.36	0.34	**0.32**
DI	1.4	1.4	**1.53**
Vow.			
NMI	0.27	0.31	**0.24**
JI	0.17	1.19	**0.11**
DI	1.26	1.23	**1.38**

Table 4: Results on three real world datasets of CAMI versus unsupervised algorithms. DI measures quality (higher is better) and NMI and JI measure similarity (lower is better)

both semi-supervised and unsupervised algorithms in terms of normalized mutual information and Jaccard Index, indicating its two clusterings are more dissimilar compared to those returned by the other algorithms. For the Vowel dataset, CAMI's dissimilarity value is significantly better than those of semi-supervised techniques. On the other hand, its clustering quality, measured in Dunn Index, is also very competitive. It is the highest for the Segmentation dataset and only slightly smaller than that of COALA in the Vehicle dataset and ADFT in the Vowel dataset.

We also tested an alternate strategy whereby the class labels of the dataset were used to form a clustering as input to the semi-supervised techniques. This made little difference to the results though and we do not report them here due to space constraints.

4.5 Impact of Regularization Parameter

As mentioned in Sections 3.2 and 3.3, the parameter η has been used to regularize the trade-off between the degree of the dissimilarity between two clustering solutions and their clustering quality. We next report the behavior of CAMI when this parameter is varied.

In Figure 5, the relationship between the normalized mutual information, the Dunn Index, and the regularization parameter η is shown for three real world datasets: Vowel, Segmentation and Vehicle. The results are reported when η is varied between 1% and 20% of each dataset's size. As expected, we observe that CAMI favors the quality of each resultant clustering over the dissimilarity between them when η is set to be small. This is indicated by the high average values of the Dunn Index and close to 1 of the normalized mutual information. As η increases, values for both NMI and Dunn Index decrease, implying that the clustering solutions are more dissimilar, yet the clustering quality has been

somewhat compromised. Three graphs in Figure 5 truly show the inverse relationship between the dissimilarity and the quality of clusterings, and they provide information helpful for suggesting an appropriate η value. As observed from all three graphs, to achieve both requirements of high qualitative and dissimilar clusterings, the best value of η can be set around 15%, since the average value of Dunn Index in this range is relatively high, whereas that value of the NMI remains small.

5 Conclusions

Searching for alternative clusterings is an important problem in exploratory data analysis with important practical significance. We have addressed this difficult problem by developing CAMI, an algorithm that uses an appealing and mathematically well founded combination of expectation maximization (to guarantee quality) and mutual information (to ensure dissimilarity).

CAMI operates in a completely unsupervised manner, not requiring a background clustering like most other techniques. We have tested CAMI on both synthetic and real-life datasets and compared it against seven existing techniques. The experimental results show that CAMI has strong performance overall, and in most of the cases, its clustering quality and dissimilarity are better than those of the semi-supervised clustering techniques. Its performance is also superior to the only two existing unsupervised algorithms, which confirms its advantages of combining mutual information and the maximum likelihood framework.

For future work, we plan to extend the CAMI algorithm and its mathematical framework to simultaneously uncover more than two alternative clusterings. Another interesting avenue is to improve it to work with statistical models other than the Gaussian mixture model.

References

[1] A. Asuncion and D.J. Newman, *UCI machine learning repository*, 2007.

[2] E. Bae and J. Bailey, *Coala: A novel approach for the extraction of an alternate clustering of high quality and high dissimilarity*, ICDM Conference, pages 53–62, 2006.

[3] S. Basu, A. Banerjee, and R. Mooney, *Active semi-supervision for pairwise constrained clustering*, SDM Conference, pages 333–344, 2004.

[4] Mikhail Bilenko, Sugato Basu, and Raymond J. Mooney, *Integrating constraints and metric learning in semi-supervised clustering*, ICML Conference, 2004.

[5] C. Bishop, *Pattern Recognition and Machine Learning*, Springer-Verlag, 2006.

[6] G. Chechik and N. Tishby, *Extracting relevant structures with side information*, NIPS Conference, pages 857–864, 2002.

[7] Thomas M. Cover and Joy A. Thomas, *Elements of Information Theory*, Wiley-Interscience, August 1991.

[8] Y. Cui, X. Fern, and J. Dy, *Non-redundant multi-view clustering via orthogonalization*, ICDM Conference, pages 133–142, 2007.

[9] I. Davidson and Z. Qi, *Finding alternative clusterings using constraints*, ICDM Conference, pages 773–778, 2008.

[10] I. Davidson, S. Ravi, and M. Ester, *Efficient incremental constrained clustering*, SIGKDD Conference, pages 240–249, 2007.

[11] X. Fern and W. Lin, *Cluster ensemble selection*, Stat. Anal. Data Min, 1(3):128–141, 2008.

[12] E. Gokcay and C. Principe, *Information theoretic clustering*, IEEE Trans. Pattern Anal. Mach. Intell., 24(2):158–171, 2002.

[13] D. Gondek, *Non-redundant clustering*, PhD Thesis, Brown University, 2005.

[14] D. Gondek, Hofmann, and Thomas, *Non-redundant data clustering*, KAIS Journal, 12(1):1–24, 2007.

[15] D. Gondek and T. Hofmann, *Conditional information bottleneck clustering*, ICDM Conference, pages 36–42, 2003.

[16] D. Gondek, S. Vaithyanathan, and A. Garg, *Clustering with model-level constraints*, SDM Conference, 2005.

[17] P. Jain, R. Meka, and I. Dhillon *Simultaneous unsupervised learning of disparate clusterings*, SDM Conference, pages 858–869, 2008.

[18] K.Wagstaff and C.Cardie *Clustering with instance-level constraints*, ICML Conference, pages 1103–1110, 2000.

[19] M. Law, A. Topchy, and A. Jain, *Multiobjective data clustering*, CVPR Conference, pages 424–430, 2004.

[20] L.Xu and M. Jordan, *On convergence properties of the em algorithm for gaussian mixtures*, Neural Computation, 8:129–151, 1996.

[21] Z. Qi and I. Davidson, *A principled and flexible framework for finding alternative clusterings*, SIGKDD Conference, pages 240–249, 2009.

[22] A. Topchy, A. Jain, and W. Punch, *A mixture model for clustering ensembles*, SDM Conference, 2004.

[23] K. Torkkola, *Feature extraction by non parametric mutual information maximization*, J. Mach. Learn. Res., 3:1415–1438, 2003.

[24] K. Wagstaff, C. Cardie, S. Rogers, and S. Schrodl, *Constrained k-means clustering with background knowledge*, ICML Conference, pages 577–584, 2001.

[25] E. Xing, A. Ng, M. Jordan, and S. Russell, *Distance metric learning with application to clustering with side-information*, NIPS Conference, pages 505–512, 2002.

[26] L. Yang and R. Jin, *Distance metric learning: A comprehensive survey*, 2006.

[27] R. Yang and M. Zwolinski, *Mutual information theory for adaptive mixture models*, Trans. Pattern Anal. Mach. Intell., 23(4), 2001.