A Hierarchical Information Theoretic Technique for the Discovery of Non Linear Alternative Clusterings

Xuan Hong Dang and James Bailey
Department of Computer Science and Software Engineering
The University of Melbourne, Australia
{xdang,jbailey}@csse.unimelb.edu.au

ABSTRACT

Discovery of alternative clusterings is an important method for exploring complex datasets. It provides the capability for the user to view clustering behaviour from different perspectives and thus explore new hypotheses. However, current algorithms for alternative clustering have focused mainly on linear scenarios and may not perform as desired for datasets containing clusters with non linear shapes. Our goal in this paper is to address this challenge of non linearity. In particular, we propose a novel algorithm to uncover an alternative clustering that is distinctively different from an existing, reference clustering. Our technique is information theory based and aims to ensure alternative clustering quality by maximizing the mutual information between clustering labels and data observations, whilst at the same time ensuring alternative clustering distinctiveness by minimizing the information sharing between the two clusterings. We perform experiments to assess our method against a large range of alternative clustering algorithms in the literature. We show our technique’s performance is generally better for non-linear scenarios and furthermore, is highly competitive even for simpler, linear scenarios.

Categories and Subject Descriptors
I.5 [Pattern Recognition]: Clustering

General Terms
Algorithm

Keywords
Alternative Clustering, Information Theoretic Learning, Parzen-window technique

1. INTRODUCTION

Data clustering aims at discovering novel patterns and structures from data. Its objective is to categorize similar data instances into the same classes (or clusters). However, while it may be reasonable to refer to a “best” model in supervised classification, it is less useful to make statements about a single, best clustering [5]. When exploring complex data, different clusterings can exist and they may each be reasonable. For example when analyzing a document dataset, one may find that it is possible to categorize according to either topics or writing styles; or when clustering a gene dataset, it is found that grouping genes based on their functions or structures is equally useful [6]. This challenge has recently stimulated the growing research area of alternative clustering, where the goal is to generate different, yet high quality clusterings or groupings of a given dataset.

Several algorithms have been developed for the task of alternative clustering. Given an input, the reference clustering, the task is to generate another clustering, which is dissimilar from the reference one, yet it is still plausible (i.e. has high quality). It is this dual objective of achieving both dissimilarity and quality that makes the task challenging.

In this paper, we explore another aspect of the alternative clustering problem. We focus on the scenario of non linearity, where cluster shapes may have unusual and non-Gaussian shapes and the border between clusters may not be linearly separable. We show that current algorithms for alternative clustering tend to underperform in this scenario. This motivates us to develop a new algorithm, called NACI (Non-linear Alternative Clustering with Information theory). NACI is a hierarchical technique which uses information theoretic methods to optimize the dual objective functions of both quality and dissimilarity.

In particular, given a predefined (reference) clustering, the NACI algorithm aims to discover an alternative clustering for which i) the mutual information between its cluster labels and data observations is maximized, whereas at the same time ii) the mutual information between the alternative clustering and the reference clustering is minimized. Objective i) helps to reduce the uncertainty within each cluster of the alternative clustering, by ensuring there is a strong (probabilistic) relationship between the cluster labels and the data instances. We later motivate this clustering objective through the use of Fano’s inequality. On the other hand, objective ii) helps to ensure that the alternative clustering is independent (different) from the reference clustering.

The principal technical contribution of our work is the formulation of a well founded alternative clustering objective function, that is purely information theoretic. The advantage of using an information theoretic approach is that it can adapt well to the presence of non linearities. However,
the technical development is not straightforward, requiring
the use of Parzen windows for probability density estima-
tions, as well as approximations based on quadratic mutual
information.

Through an experimental analysis, we show that NACI
performs particularly strongly when finding alternative clus-
terings for non-linear datasets, improving over the state of
the art. Furthermore, even for simpler, more linear datasets,
NACI is able to discover desirable alternative clusterings,
possessing high quality and high dissimilarity to the refer-
ence clustering.

2. RELATED WORK

There are several works related to our research. The clos-
est are those developed in [13, 2, 8, 10], which exploit various
forms of negative information toward the desired clustering
(as opposed to those using prior knowledge to improve clus-
tering results [17, 22, 4]). In [13], the authors proposed
a conditional information bottleneck (CIB) method, which
treats class labels of a given clustering as negative informa-
tion in seeking an alternative clustering. The new data
partition is found by maximizing the information sharing
between the cluster labels and the data features (describ-
ing for data objects), but conditioned on the given refer-
ence clustering. This method, though similar to our work
in that both address the problem from an information the-
ory viewpoint, is rather different in two important aspects.
First, our approach makes no assumption regarding the data
density distribution, whilst CIB requires the availability of
the joint distribution information between cluster labels and
the features, which is known as being hard to formalize[10].
Second, while our algorithm directly minimizes the mutual
information between two clusterings to ensure their inde-
pendence, the CIB approach only conditions on the refer-
ence clustering in the process of encoding properties of the
data features into the new clustering. In other words, it
uses mutual information in a completely different way to
our approach. Another approach, which exploits the effec-
tiveness of pairwise constraints for data clustering [22, 3],
is the COALA technique [2]. From the reference clustering,
COALA generates a set of cannot-link constraints between
pairs of data samples and attempts to find a different clus-
tering that satisfies as many as possible of these constrains.
On the other hand, a line of work developed in [8, 10] takes a rather different approach to alternative clustering by relying
on the notion of orthogonality. In [8], the authors develop
two techniques to find an alternative clustering using or-
thogonal projections. In the first one, data is projected onto
a space that is orthogonal to the space spanned by the set
of mean vectors in the given clustering, while in the second
technique, such a representative vectors are replaced by the
feature space. An alternative clustering is then found by
simply applying a clustering algorithm on this new trans-
formed data. A similar approach is developed in [10] by
which the transformation is applied on the distance matrix
learned from the provided clustering. In comparison, this
work has an advantage compared to[8], since it avoids prob-
lems when the data dimension is smaller than the number of
clusters (e.g., spatial datasets).

Another series of works addressing the alternative clus-
tering problems are those developed in [15] and [9]. Unlike
the work above, these methods attempt to seek two alter-
native clusterings at the same time. In the first work, two
algorithms named Dec-kmeans and ConvEM are developed
which attempt to derive two sets of mean vectors that are
pairwise orthogonal, whereas in the second work [9], an al-
gorithm, known as CAMI seeks two clusterings that share
minimal mutual information. A clear distinction between
these algorithms and ours in this paper is that our algo-
rithm is not limited to spherically shaped clusterings, i.e. it
is able to seek alternative clusterings for non-linear datasets.
We provide experimental comparisons with all the above al-
gorithms in Section 5 of the paper.

3. PRELIMINARIES

3.1 Entropy and Mutual Information

In information theory, the quantity entropy plays a cen-
tral role and is a measure of the uncertainty of a random
variable. Mathematically, let X be a continuous random
variable characterized by the probability distribution p(x),
the Shannon entropy of X is:

\[ H(X) = - \int p(x) \log p(x) dx \]  \hfill (1)

When a variable is known and another is not, the remain-
ing uncertainty is measured by the conditional entropy:

\[ H(Y|X) = - \int \int p(x, y) \log p(y|x) dx dy \]  \hfill (2)

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:

\[ KL(p\|q) = \int p(x) \log \left( \frac{p(x)}{q(x)} \right) dx \]  \hfill (3)

Mutual information, which is of importance in our work,
turns out to be a special case of the KL divergence. It mea-
sures the information shared between two objects or in other
words, it accounts for the amount of information that one
random variable contains about another variable. In spe-
cific form, let X and Y be two random variables with a joint
probability density function p(x, y) and marginal probabil-
ity density functions p(x) and p(y), the mutual informa-
tion I(X;Y) is the KL distance between the joint distribu-
tion and the product of two marginal distributions p(x) and p(y):

\[ I(X;Y) = \int \int p(x, y) \log \left( \frac{p(x, y)}{p(x)p(y)} \right) dx dy \]
\[ = KL(p(x, y)\|p(x)p(y)) \]  \hfill (4)

which is obviously symmetric and non-negative. Impor-
tantly, two random variables have zero mutual information
if and only if they are statistically independent.

3.2 Problem Definition

Given the above definitions, an intuitive problem state-
ment is as follows:

\textbf{Definition 1.} Given a dataset } X = \{x_1, x_2, ..., x_n\} \text{ where each } x_i \text{ is in } d\text{-dimensional space and for which there is an}
existing reference clustering } C^- \text{, find a new alternative clus-
tering } C^+ \text{ from } X \text{, such that } C^+ \text{ is not only high quality but}
also as independent (i.e., different) from $C^-$ as possible. The independence between two clusterings can be quantified by the information sharing between them.

The number of clusters within each clustering $C^+$ and $C^-$ may be different. In our work, for easy of presentation, we assume that they are the same and use $k$ to denote this number. However our methods can be easily adapted to the more general case.

4. CLUSTERING METHOD

4.1 Criterion Function Optimization

For any learning algorithm, the learning process should ultimately transfer the information carried in the data samples into the system’s parameter [20]. It is therefore natural to find an objective function that directly manipulates information. Since mutual information is an essential tool to quantify the statistical relationship between any two random variables, it is intuitive to create a clustering cost function from information theory, which provides a lower bound for the availability of all variables’ probability density functions. Furthermore, the numerical integration of these functions also leads to very high computational complexity.

Fortunately, notice that our clustering objective is to optimize the mutual information, rather than computing it exactly. It has been shown in [16, 20] that as long as a learning process does not require to compute an exact value of the mutual information, but rather to maximize or minimize it, then other practical divergences can be used. Importantly, the extrema of these divergences are also coincident with those of the KL divergence and therefore, the objectives of optimization are not compromised. One of such divergence is presented in [16]:

\[ D(p||q) = \frac{1}{\alpha(\alpha - 1)} \sum_{i=1}^{n} (p^\alpha(x_i) - \alpha p(x_i) + (\alpha - 1)q^\alpha(x_i)) \]

where $\alpha \neq 0, 1$.

Based on this, a quadratic form of mutual information can be derived by selecting $\alpha = 2$ and extending the equation to continuous densities (the first constant term can be omitted):

\[ I_{R_2}(X;Y) = \iint (p(x,y) - p(x)p(y))^2 \, dx \, dy \]

It is easy to prove that all essential properties of the divergence are preserved, i.e., $I_{R_2}$ is always non-negative, symmetric and equal to 0 if and only if $p(x,y) = p(x)p(y)$.

When applying this quadratic form of mutual information to our alternative clustering problem, it is possible to derive the information sharing between two discrete variables $C^+$ and $C^-$ as follows:
\[ I_{R_2}(C^+; C^-) = \sum_{c_i^+} \sum_{c_j^-} \left( p(c_i^+, c_j^-) - p(c_i^+) p(c_j^-) \right)^2 \]
\[ = \sum_{c_i^+} \sum_{c_j^-} p(c_i^+, c_j^-)^2 + \sum_{c_i^+} \sum_{c_j^-} p(c_i^+)^2 p(c_j^-)^2 - 2 \sum_{c_i^+} \sum_{c_j^-} p(c_i^+, c_j^-) p(c_i^+)^2 p(c_j^-) \]
\[ = \sum_{c_i^+} \int_x p(c_i^+, x)^2 dx \sum_{c_i^+} \int_x p(c_i^+)^2 p(x)^2 dx - 2 \sum_{c_i^+} \int_x p(c_i^+, x) p(c_i^+)^2 p(x)dx \]

and the quadratic mutual information between the continuous variable \( X \) and the discrete variable \( C^+ \):

\[ I_{R_2}(C^+; X) = \sum_{c_i^+} \int_x (p(c_i^+, x) - p(c_i^+) p(x))^2 dx \]
\[ = \sum_{c_i^+} \int_x p(c_i^+, x)^2 dx + \sum_{c_i^+} \int_x p(c_i^+)^2 p(x)^2 dx - 2 \sum_{c_i^+} \int_x p(c_i^+, x)^2 dx \]

where the prior probabilities \( p(c_i^+) \) and \( p(c_j^-) \) are estimated by the number of data samples in each cluster (over \( n \), i.e., \( n_i/n \) and \( n_j/n \). Similarly, the joint probability between \( c_i^+ \) and \( c_j^- \) is estimated by the number of data samples belonging to both \( c_i^+ \) and \( c_j^- \), i.e., \( n_{ij}/n \).

Notice that computing the mutual information in Eq. (9) requires the estimation of variables’ probability density function. Nonetheless, an appealing property of the quadratic mutual information is that it is possible to combine it with the Parzen window method, an effective non-parametric density estimation technique, to simplify the computation. This involves placing a kernel function at each data sample. The density is accordingly evaluated by the sum of kernels. When using a Gaussian function for the kernel, it follows that:

\[ p(x) = \frac{1}{n} \sum_{i=1}^{n} G(x - x_i, \sigma^2) \]  

where \( G(x - x_i, \sigma^2) = \frac{1}{(2\pi\sigma)^d/2} \exp \left\{ -\frac{||x - x_i||^2}{2\sigma^2} \right\} \) is the Gaussian in a \( d \)-dimensional space. An important property with this kernel is that the convolution of two Gaussians remains a Gaussian function:

\[ \int_x G(x - x_i, \sigma^2) G(x - x_j, \sigma^2) dx = G(x_i - x_j, 2\sigma^2) \]  

This equation can be interpreted as the information potential between \( x_i \) and \( x_j \). Therefore, when combining the non-parametric density estimation method with the quadratic mutual information, the computational complexity can be greatly reduced. Specifically, according to the Parzen window method:

\[ p(x|c_i^+) = \frac{1}{n_i} \sum_{\ell=1}^{n_i} G(x - x_\ell, \sigma^2) \]

where \( n_i \) is the number of data points belonging to cluster \( c_i^+ \). Followed by Bayes theorem, the joint probability between \( x \) and \( c_i^+ \) is:

\[ p(c_i^+, x) = p(c_i^+) p(x|c_i^+) = \frac{1}{n} \sum_{\ell=1}^{n} G(x - x_\ell, \sigma^2) \]

The quadratic mutual information in Eq. (9) can thus be decomposed into three terms:

\[ T_{in} = \sum_{c_i^+} \int_x p(c_i^+, x)^2 dx = \frac{1}{n^2} \sum_{c_i^+} \sum_{c_j^-} \sum_{\ell=1}^{n} G(x_\ell - x_m, 2\sigma^2) \]

\[ T_{in} \]

\[ T_{all} = \sum_{c_i^+} \int_x p(c_i^+)^2 p(x)^2 dx \]
\[ = \left( \sum_{c_i^+} \left( \frac{n_i}{n} \right)^2 \right) \left( \frac{1}{n^2} \sum_{\ell=1}^{n} \sum_{m=1}^{n} G(x_\ell - x_m, 2\sigma^2) \right) \]

\[ T_{all} \]

\[ T_{btw} = \sum_{c_i^+} \int_x p(c_i^+, x) p(c_i^+)^2 dx \]
\[ = \frac{1}{n^2} \sum_{c_i^+} \sum_{\ell=1}^{n} \sum_{m=1}^{n} G(x_\ell - x_m, 2\sigma^2) \]

\[ T_{btw} \] is seen as the sum of information potentials between each cluster’s data points and all data points, weighted by the cluster prior.

Given the computations above, it is clear that the local interaction, as defined by the kernel in Eq.(11), between any two data instances needs to be computed. Therefore, a matrix \( G \) having size of \( n \times n \) is generated in which at row \( i \) column \( j \), the information potential term \( G(x_i - x_j, 2\sigma^2) \) is computed. Notice that, in practice, only half the number of these interactions need to be evaluated due to the symmetry.

In addition to the \( G \) matrix, two other matrices \( D_{in} \) and \( D_{btw} \), which respectively account for the variation in \( I_{R_2}(C^+; X) \) and \( I_{R_2}(C^+; C^-) \) incurred by merging any pairs of clusters in \( C^+ \), are utilized. Indeed, the combination of these two matrices acts as the similarity matrix in a classical agglomerative clustering technique. Notice that, different from \( G \) whose elements do not change during the clustering process, elements of these two matrices are updated regularly and their size is reduced one upon each merging step of the algorithm. These two matrices are initialized by computing the variation in mutual information when grouping any pair of data samples.

Under the hierarchical clustering framework, there are \( n - k \) iterative steps to merge clusters. Specifically, at each iterative step of this agglomerative clustering, the maximum element from the combined matrix is selected:
\[(\alpha, \beta) = \arg \max_{i,j} \{ D_{ij} - \eta D_{btw} \} \] (14)

Subsequently, cluster \( c_{ij}^u \) is grouped into cluster \( c_{ij}^v \) and this means that the merging leads to the maximum variation in our global objective function. Upon this union, elements located at column and row \( \beta \) will be removed from \( D_{ia} \) and \( D_{btw} \) and it is necessary to update elements in column and row \( \alpha \) of these two matrices. That means a new variation on the mutual information is computed if the updated cluster \( c_{ij}^u \) is combined with any of the rest clusters in \( C^+ \).

For simplicity, we re-use the cluster index notations and denote \( c_{ij}^v \) for the new cluster by merging \( c_{ij}^u \) with any cluster \( c_{ij}^v \) in \( C^- \), then it is clear that:

\[
p(c_{ij}^v) = p(c_{ij}^u) + p(c_{ij}^v) \\
\] \[
p(c_{ij}^v, c_j^v) = p(c_{ij}^u, c_j^v) + p(c_{ij}^v, c_j^v) \] and \[ \]
\[
p(c_{ij}^v, x) = p(c_{ij}^u, x) + p(c_{ij}^v, x) \]

Upon these, the variation in the mutual information can be simply computed. Specifically, the variation of mutual information with respect to each cluster \( c_j^v \) in \( C^- \) is:

\[
\Delta I_{R_2}(C^+; c_j^v) = \int p(c_{ij}^u, x)p(x)dx - \int p(c_{ij}^u, x)p(x)dx \]

The entire variation on \( I_{R_2}(C^+; C^-) \) is therefore:

\[
\Delta I_{R_2}(C^+; C^-) = \sum_j \Delta I_{R_2}(C^+; c_j^v) \] (15)

Analogously, the variation on \( I_{R_2}(C^+; X) \) can be derived by interchanging summations and integrations:

\[
\Delta I_{R_2}(C^+; X) = 2 \left( \int p(c_{ij}^u, x)p(x)dx + \int p(c_{ij}^u)p(x)dx \right) \]

\[
\int p(c_{ij}^u, x)p(x)dx - \int p(c_{ij}^u)p(x)dx \] (16)

in which:

\[
\int p(c_{ij}^u, x)p(x)dx = \frac{1}{n^2} \sum_{k=1}^{n_a} \sum_{\ell=1}^{n_b} G(x_k - x_\ell, 2\sigma^2) \]

\[
\int p(c_{ij}^u)p(x)dx = \frac{n_a n_b}{n^2} \sum_{k=1}^{n_a} \sum_{\ell=1}^{n_b} G(x_k - x_\ell, 2\sigma^2) \]

and

\[
\int p(c_{ij}^u)p(x)dx = \frac{n_a}{n^2} \sum_{k=1}^{n_a} \sum_{\ell=1}^{n_a} G(x_k - x_\ell, 2\sigma^2) \]

Notice that all these summations can be easily obtained from the \( G \) matrix.

Finally, it can be observed from the above that the variation in the two mutual information values may be in different units. This is because \( I_{R_2}(C^+; X) \) is calculated between a clustering solution and the data (i.e. a discrete and continuous variables), whereas \( I_{R_2}(C^+; C^-) \) is computed between two clustering solutions (i.e. two discrete variables). Therefore, in order to avoid this difficulty, the variation with respect to each mutual information is normalized by dividing it by the corresponding quadratic mutual information (Eqs.(8),(9)). This also makes it easier when regularizing the trade-off factor \( \eta \) between these two information’s quantities.

### 4.3 Kernel Parameter Setting

One of the key advantages in our algorithm is that it makes no prior assumption about the probability density functions and these functions are approximated directly from data using the non-parametric method. However, it should be noticed that the success of this approach is dependent on an appropriate selection for the kernel parameter, that is the standard deviation \( \sigma \). It is shown in [19] that if the kernel width \( \sigma \) is annealed toward zero at a sufficiently low rate as \( n \) tends to infinity, then the Parzen window density estimator will be asymptotically unbiased and consistent. However, for most practical applications where data are finite, the kernel size should be selected in such a way that it balances out the bias and variance, which essentially being derived from the optimization of the mean integrated squared error between an estimator \( \hat{p}(x) \) and the true density \( p(x) \): \( MISE\{\hat{p}(x)\} = \int \mathbb{E}\left\{[\hat{p}(x) - p(x)]^2\right\}dx \).

In our work, we choose \( \sigma = \hat{\sigma} \left( \frac{4 \ln 2 + \delta}{n^{d/2} \pi^d} \right)^{1/4} \), which is derived by applying the least square cross-validation and the normal reference rule [23] to minimize the generalization error above. It was also further experimentally observed that by setting \( \sigma \) at this value, the interaction between samples that are far distant is still considered, while the interaction between close data samples remains emphasized. As shown in our experimental section, this \( \sigma \) selection results in good clusterings for most of the datasets examined.

### 4.4 Algorithm Complexity

The proposed algorithm requires to compute the matrix \( G \) of information potentials between any pair of data samples. The complexity of this operation requires \( O(dn^2) \) where \( d \) is the cost of calculating the interaction according to Eq.(11), and \( n \) is the number of data observations. The calculation of mutual information’s variation when merging any two data samples takes \( O(n^2) \). At each merging step, the maximum value is selected from the combination of two matrices \( D_{ia} \) and \( D_{btw} \). By using the priority queue data structure [7] that supports the search and deletion in \( O(\log n) \) from this matrix, this step thus takes \( O(n \log n) \). The calculation of updating information variation according to Eqs.(15) and

\[ \int p(c_{ij}^u)p(x)dx = \frac{n_a}{n^2} \sum_{k=1}^{n_a} \sum_{\ell=1}^{n_a} G(x_k - x_\ell, 2\sigma^2) \]

where \( \hat{\sigma} = \frac{1}{n} \sum_i \sigma_i \) and \( \sigma_i \)'s are the diagonal elements of the sample covariance matrix.
(16) takes constant time given the availability of information potential matrix $G$. Since there are $(n-k)$ merging steps for the agglomerative clustering, the computation is $O(n^2 \log n)$. The overall complexity of our proposed algorithm is therefore $O(n^2 \log n + d n^2)$. Considering that $d$ is usually smaller than $\log n$, it is possible to say that the final complexity is $O(n^2 \log n)$, which is the same as that of a conventional hierarchical clustering using group-average similarity.

5. EXPERIMENTAL RESULTS

In this section, we provide experimental results on both synthetic and real-world benchmark datasets. Our NACI algorithm is compared against eight methods, including five semi-supervised alternative clustering algorithms: the CIB method [13], COALA [2], two methods from [8] denoted by Alg01 and Alg02, and the ADFT algorithm[10]; and three unsupervised alternative clustering algorithms: the Dec-kmeans, ConvEM from [15], and our previous algorithm CAMI [9]. Unless otherwise indicated, we set $\eta = 0.2$ as the default value for NACI. For the CIB method, we implement the iterative version [12, 13] and its outputs are post-processed by assigning each data point to the cluster to 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) [24, 25]. We also use the EM technique as the background clustering technique for the approaches developed in [8, 10]. For the Dec-kmeans, ConvEM and CAMI, we follow the heuristic method described in their work to set the trade-off factor between the clustering quality and dissimilarity. Since both NACI and COALA are developed based on agglomerative hierarchical clusterings, which are not sensitive to the initial parameters but possibly to the data instances’ order. Therefore, when running them, we randomly swap the order amongst data samples. We run each algorithm 20 times and report the average results.

5.1 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 and also the most popular one is the normalized mutual information [18, 21, 14, 11]: $\text{NMI}(C^+; C^-) = I(C^+; C^-)/(H(C^+)H(C^-))$, where $I(C^+; C^-) = \sum_{i=1}^{M^+} \sum_{j=1}^{M^-} \frac{n_{ij}}{n} \log \left( \frac{n_{ij}}{n_i n_j} \right)$ with $n_{ij}$ denoting the number of shared instances between clusters $c^+_i \in C^+$ and $c^-_j \in C^-$. The second is the Jaccard index (JI) [2, 10]: $J(C^+; C^-) = \frac{n_{ii}}{n_{ii} + n_{i0} + n_{0i}}$ in which $n_{ii}$ is the number of pairs of samples in the same cluster for both $C^+$ and $C^-$, $n_{i0}$ and $n_{0i}$ are the number of samples’ pairs belonging to the same cluster in one solution, but not in the other.

For measuring clustering quality we divide into two cases: if true class labels are known, the agreement between clustering results and the correct labels is calculated by the F-measure: $F = 2P \times R / (P + R)$, in which $P$ and $R$ are respectively the precision and recall. If true class labels are not known, we use the Dunn Index, similar to [2, 10]: $\text{DI}(C) = \min_{C^+ \neq C^-} \{ \delta(C^+; C^-) \}$ where $C$ is a clustering, $\delta: C \times C \rightarrow \mathbb{R}^+_0$ is the cluster-to-cluster distance and $\triangle: C \rightarrow \mathbb{R}^+_0$ 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 F-measure and Dunn Index, a larger value is desirable, indicating a better clustering quality. Also, since methods like Dec-kmeans, ConvEM and CAMI do not require existing clusterings to be provided and instead seek two alternative clusterings at the same time, we try to achieve a fair comparison with them by reporting the higher values of F-measure in the case true labels are available, and averaging the Dunn Index when the class labels are not known.

5.2 Synthetic Datasets

Four synthetic datasets are used to evaluate the performance of our proposed clustering technique against other algorithms. For the first dataset Syn1, we use a popular one from [2, 8, 10, 15], which consists of 4 Gaussian sub-classes. Each Gaussian contains 200 points in 2-dimensional data space. The goal of using this dataset, when setting $k = 2$, is to test whether our algorithm can discover an alternative clustering that is orthogonal to the existing one. For the second synthetic dataset Syn2, we use a more complicated one in which 6 Gaussian sub-classes are located in a ring shape. Different from Syn1, this dataset consists two equally important clusterings (with $k = 3$) that are not orthogonal and it is not possible to find them by simply projecting the data on either of the subspaces. We generate the third and fourth datasets by replacing a non-Gaussian shape for each subclass in the first synthetic dataset. By using these datasets, we aim to test the ability of our algorithm in uncovering non-linear clusterings.

Figures 1(a), (b) and (c) show clustering solutions respectively for the first three synthetic datasets. Clusterings in the top graphs of each figure are provided as pre-existing reference solutions to each semi-supervised algorithm and in the bottom graphs, we demonstrate the alternative clusterings returned by the NACI, which exactly match second important clusterings included in each dataset. We compare the performances of all algorithms via the results summarized in Table 1. As can be seen from the table, like other alternative clustering methods, our proposed algorithm can
NACI’s alternative clustering
Algo1’s alternative clustering
ADFT’s alternative clustering
COALA’s alternative clustering
CIB’s alternative clustering
Dec-kmeans’s two alternative clusterings
CAMI’s two alternative clusterings
Pre-defined clustering

Figure 2: Alternative clusterings returned by all algorithms on Syn4 dataset. The first graph presents for the pre-defined clustering.

<table>
<thead>
<tr>
<th>Methods</th>
<th>NMI</th>
<th>JI</th>
<th>F</th>
<th>NMI</th>
<th>JI</th>
<th>F</th>
<th>NMI</th>
<th>JI</th>
<th>F</th>
<th>NMI</th>
<th>JI</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alg0</td>
<td>0.25</td>
<td>0.41</td>
<td>0.83</td>
<td>0.42</td>
<td>0.41</td>
<td>0.62</td>
<td>0.2</td>
<td>0.42</td>
<td>0.78</td>
<td>0.28</td>
<td>0.34</td>
<td>0.63</td>
</tr>
<tr>
<td>Alg02</td>
<td>0.26</td>
<td>0.43</td>
<td>0.81</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>0.21</td>
<td>0.44</td>
<td>0.76</td>
<td>0.28</td>
<td>0.34</td>
<td>0.63</td>
</tr>
<tr>
<td>ADFT</td>
<td>0.12</td>
<td>0.39</td>
<td>0.92</td>
<td>0.62</td>
<td>0.61</td>
<td>0.57</td>
<td>0.14</td>
<td>0.36</td>
<td>0.86</td>
<td>0.36</td>
<td>0.36</td>
<td>0.62</td>
</tr>
<tr>
<td>COALA</td>
<td>0</td>
<td>0.33</td>
<td>1</td>
<td>0.38</td>
<td>0.35</td>
<td>0.53</td>
<td>0.18</td>
<td>0.41</td>
<td>0.79</td>
<td>0.25</td>
<td>0.37</td>
<td>0.58</td>
</tr>
<tr>
<td>CIB</td>
<td>0.12</td>
<td>0.39</td>
<td>0.91</td>
<td>0.41</td>
<td>0.39</td>
<td>0.72</td>
<td>0.24</td>
<td>0.46</td>
<td>0.69</td>
<td>0.37</td>
<td>0.39</td>
<td>0.53</td>
</tr>
<tr>
<td>Dec-kmeans</td>
<td>0.12</td>
<td>0.39</td>
<td>0.93</td>
<td>0.39</td>
<td>0.34</td>
<td>0.68</td>
<td>0.12</td>
<td>0.35</td>
<td>0.87</td>
<td>0.22</td>
<td>0.34</td>
<td>0.62</td>
</tr>
<tr>
<td>ConvEM</td>
<td>0.1</td>
<td>0.38</td>
<td>0.95</td>
<td>0.37</td>
<td>0.33</td>
<td>0.92</td>
<td>0.11</td>
<td>0.35</td>
<td>0.88</td>
<td>0.21</td>
<td>0.34</td>
<td>0.63</td>
</tr>
<tr>
<td>CAMI</td>
<td>0</td>
<td>0.33</td>
<td>1</td>
<td>0.35</td>
<td>0.32</td>
<td>1</td>
<td>0</td>
<td>0.33</td>
<td>1</td>
<td>0</td>
<td>0.33</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 1: The clustering performances of all algorithms on four synthetic datasets (for NMI and JI measures, lower values are better, and for F-measure, higher values are better). Results are not available for Alg02 on Syn2 dataset since its transformation is undefined when number of clusters greater than data dimensionality.

easily identify the orthogonal clustering for the first simple synthetic dataset. Its performance on the second dataset is also accurate, although the two alternative clusterings solutions have been deliberately designed to be non-orthogonal. Notice that, unlike Syn1, there are no dominant features that fully support any of the clusterings in this dataset. Therefore, the methods developed based on orthogonal space transformation [8, 10] or orthogonal clusters’ mean projection [15] are usually less successful in discovering the second alternative clustering. Only CAMI and NACI are able to achieve high accurate results since they both adopt the approach of mutual information minimization. However, different from CAMI, which slightly suffers from the problem of initial parameters sensitivity, NACI can completely avoid this since it is designed based on the hierarchical clustering technique. In addition, though we tested the strategy at which data samples were randomly swapped before each running, it was still found that NACI’s performance was more consistent across all trials.²

²We refer the reader to our previous work [9] for detailed justification on the clustering outputs of the other algorithms.

We provide the clustering output of NACI for the Syn3 dataset in Figure 1(c) and all algorithms for the Syn4 dataset in Figure 2 (the clustering outputs of Alg02 and ConvEM are omitted since they are very much similar to those of Alg01 and Dec-kmeans, respectively). For these two synthetic datasets, we aim to test the algorithms’ ability in uncovering nonlinear clusterings. It is clear that methods like Alg01 and Alg02, or Dec-kmeans, ConvEM and CAMI are unable to identify the correct alternative clustering, since their core algorithms are tied to a particular spherical clustering technique (e.g., k-means, EM). Moreover, the data projection on a space orthogonal to the set of mean vectors used in Alg01 or analogously to the feature subspace used in Alg02, does not help them uncover the second alternative clustering, since the data is distorted and seems to be more overlapping by these transformations; whereas the clustering approaches used in ConvEM and CAMI only ensure a low value of decorrelation between two alternative clusterings, but cannot guarantee that accurate clusterings can be found for these nonlinear clustering shapes due to the EM technique. Similarly, by inverting the stretcher ma-
to different poses. This obviously provides another mean-
clearly reveal that images have been partitioned according
data. While pictures in the first row show that they rep-
different, yet equally important clustering on the set of image
terns. The alternative clustering returned by NACI provides another dif-
in the NACI’s alternative clustering (in the bottom graphs).

principal components that cover 90% of the original data’s
processing step, the PCA technique is applied to reduce the
is known which image comes from which person, this forms
images captured in every combination of these features. We
take 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. Since it
is known which image comes from which person, this forms
an existing partition of the dataset. We run NACI and the other
algorithms with this provided clustering. As a pre-
processing step, the PCA technique is applied to reduce the
number of dimensions, in which we retain the number of first
principal components that cover 90% of the original data’s
variance.

In Figure 3, we show the mean vectors of the provided
clustering (in the top graphs) and the mean vectors returned
in the NACI’s alternative clustering (in the bottom graphs).
Pictorially, it is possible to observe that the uncovered alter-
native clustering returned by NACI provides another dif-
fersent, yet equally important clustering on the set of image
data. While pictures in the first row show that they rep-
resent for different individuals, pictures in the second row
clearly reveal that images have been partitioned according
to different poses. This obviously provides another mean-
ingful interpretation about this dataset. For specific results,

Figure 3: NACI’s results on CMUFace dataset

<table>
<thead>
<tr>
<th>Methods</th>
<th>NMI</th>
<th>JI</th>
<th>F(pose)</th>
<th>F(person)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algo1</td>
<td>0.31</td>
<td>0.34</td>
<td>0.68</td>
<td>0.87</td>
</tr>
<tr>
<td>Algo2</td>
<td>0.33</td>
<td>0.36</td>
<td>0.67</td>
<td>0.84</td>
</tr>
<tr>
<td>ADFT</td>
<td>0.29</td>
<td>0.33</td>
<td>0.69</td>
<td>0.89</td>
</tr>
<tr>
<td>COALA</td>
<td>0.27</td>
<td>0.32</td>
<td>0.71</td>
<td>0.87</td>
</tr>
<tr>
<td>CIB</td>
<td>0.28</td>
<td>0.34</td>
<td>0.69</td>
<td>0.86</td>
</tr>
<tr>
<td>Dec-kmeans</td>
<td>0.26</td>
<td>0.32</td>
<td>0.72</td>
<td>0.9</td>
</tr>
<tr>
<td>ConvEM</td>
<td>0.28</td>
<td>0.33</td>
<td>0.7</td>
<td>0.89</td>
</tr>
<tr>
<td>CAMI</td>
<td>0.24</td>
<td>0.31</td>
<td>0.74</td>
<td>0.89</td>
</tr>
<tr>
<td>NACI</td>
<td>0.2</td>
<td>0.24</td>
<td>0.81</td>
<td>0.94</td>
</tr>
</tbody>
</table>

Table 2: Results on the CMUFace dataset.
for clustering quality comparison amongst the nine clustering techniques. We report the results of all techniques on these datasets in Table 3.

Looking at this table, we see that NACI also performs well with all three datasets. Note that these datasets have a much higher degree of non-linearity, compared to the ones we have already examined. It can be noted that NACI typically finds more dissimilar clusterings (as measured by both NMI and JI) compared to those of other algorithms. Its clusters found in the alternative clustering are also well separated as indicated by the small values of Dunn index. This measure is only slightly larger than that of CAMI in the Vehicle dataset and ADFT in Vowel dataset. This might happen, since the Dunn Index is essentially computed by the averaging distances between pairs of points in two clusters over the maximum cluster diameter, and thus somewhat supports clusters returned by CAMI and ADFT. However, overall, one can observe that our algorithm tends to achieve more decorrelated (i.e., more different) clusterings, whereas its clustering quality is very competitive in all three datasets. Its performance in the Segmentation dataset is better than all other algorithms.

### 5.5 Parameter Sensitivity

There are two parameters that may impact the performance of our NACI algorithm: the kernel width $\sigma$ and the regularization factor $\eta$. We have conducted a series of experiments to examine the sensitivity of the results on these parameters. For the kernel width $\sigma$, though its variation can affect the algorithm’s performance, we have found that for most of cases, setting it to the value derived in Section 4.3 often leads to good and stable clustering results (as reported in the previous sections). Due to lack of space, we can only present here the main observations with respect to the parameter $\eta$, which regularizes for the relative importance between two quantities of mutual information.

Since both the variations computed in Eqs. (15) and (16) have been normalized by the corresponding mutual information, we set the $\eta$ to be within the unit interval. In Figure 4, we show the relationship between the normalized mutual information, the Dunn index and the values of $\eta$ for three real-world datasets: Segmentation, Vehicle Silhouette, and Vowel. The results are reported when $\eta$ is varied from 0.1 to 0.5 with a step of 0.05. As we expected, small values of $\eta$ lead to compact clustering solutions (in terms of Dunn index), but such results seem to be quite overlapping with the given clusterings (shown by the high values of NMI). As the values of $\eta$ increase, the clustering quality somewhat reduces but the alternative clusterings are more decorrelated from the existing solutions. However, once $\eta$ is above 0.3, it was observed the fluctuations happened with the NMI’s values. This can be explained by the hierarchical clustering approach of NACI, where it tends to combine clusters which overly support small values of the information between two clusterings at the beginning, but such decisions cannot be undone at a later time where it converges to a small number of final clusters. Likewise, it was seen that the resultant clusters in the alternative clusterings in this case were also very imbalanced, which made the Dunn index located at small values as well. Nevertheless, as we observed from all three graphs in Figure 4, high quality and dissimilar alternative clusterings can be achieved if $\eta$ is set around 0.2 since the Dunn Index in this range is relatively high, whereas that value of NMI is also small.

### 6. CONCLUSIONS

In this paper we have proposed a novel algorithm called NACI, to discover alternative clusterings, which are of high quality, yet distinctively different from a provided reference clustering. We address the problem purely from information theory, in which the clustering quality is achieved by maximizing the mutual information between cluster labels and the data observations (this implicitly ensures a strong probabilistic relationship between them), whereas the dissimilarity between two alternative clusterings is achieved by the minimization of the mutual information between them. To fully exploit the information embedded in the data, we employed the Parzen window method for probability density estimation. Such a non-parametric technique does not impose
any assumptions regarding the data distribution and further enables practical computations, when combined with the quadratic mutual information form. These features made NACI particularly suitable for the challenging scenario where datasets have non-linear structures.

We evaluated the performance of our algorithm on a number of synthetic and real-world benchmark datasets, comparing against eight well-known existing approaches. The experimental results show NACI is able to achieve excellent performance for non-linear cases, and is able to obtain highly competitive performance even for simpler, more linear structures. We believe that NACI is a powerful tool for alternative clustering discovery and exploration.

7. REFERENCES