Dual Criteria Determination of Natural Clustering Structures in Data

Kaixun Hua
University of Massachusetts Boston
Department of Computer Science
Boston, USA
kingsley@cs.umb.edu

Dan A. Simovici
University of Massachusetts Boston
Department of Computer Science
Boston, USA
dsim@cs.umb.edu

***Abstract*—We present a technique grounded in information theory for determining the "natural" number of clusters existent in a data set. Our approach involves a bi-criteria optimization that makes use of the entropy and the cohesion of a partition. The results are promising and may be applicable in dealing with clusterings of imbalanced data.**

***Index Terms*—clustering; entropy; clustering cohesion; Pareto front; hypervolume**

I. Introduction

Clustering is one of the most important topics in unsupervised learning. It aims to partition a set of objects such that similar objects will be assigned in the same group while those who are dissimilar will be placed in different groups [1]. This definition is not entirely satisfactory, because there exist many similarity measures and the targeted number of groups is not well-defined. In particular, determining the "natural" number of clusters in a data set is a long-standing and challenging problem that has attracted a great number of investigators.

A simple approach to the problem of determining the number of clusters is to generate several partitions with different number of clusters and to choose the best partition based on an internal evaluation index. By plotting the dependency of this index on the number of clusters, it is possible to determine the number of clusters. One of the best-known techniques for the determination of the number of clusters is to check the elbow point on the resulting curve [2]. This elbow is loosely defined as the point of maximum curvature and the desired number of clusters is the cluster coordinate of the elbow point.

An alternative method is the gap statistics which aims to formalize the intuitive approach of the "elbow method" by comparing of the logarithm of the cohesion with a reference distribution of the data [3]. However, this method only works on well-separated datasets. An alternative approach proposed in [4] regards clustering as a supervised classification problem which requires the estimation of "true" class labels. The prediction strength measure evaluates the number of groups that can be predicted from data.

In [5] the largest ratio difference between two adjacent points is used to locally find the elbow point along the curve. Other authors use more than one pairs of points. The first data point with a second derivative above some threshold value is used to specify the elbow point [6], [7], while in [8] the data point with the largest second derivative is used. All these techniques are sensitive to outliers and local trends, which may not be globally significant [2].

Yet another approach to the estimation of the number of clusters is applying consensus clustering [9] and resampling [10]. This involves clustering many samples of the data set, and determining the number of clusters where clusterings of the various samples are the most stable [2]. Consensus clustering or clustering aggregation, has been explored for decades. A formal definition is given in [11], where consensus clustering is defined as a clustering that minimizes the total number of disagreements with a set of clusterings. This technique can deal with a variety of problems such as developing a natural clustering algorithm for categorical data, improve the clustering robustness by combining the results of many clustering algorithms, as well as determine the appropriate number of clusters. In recent years, many approaches have been developed to solve ensemble clustering problems [12], [13], [14], [15], [16], [17] and [18].

As a task of consensus clustering, determining the number of clusters has been considered in several publications. In [19] a hierarchical clustering framework is proposed that combines partitional clustering (k-means) and hierarchical clustering. A random walk technique on the graph defined by a consensus matrix of clusterings is used in [20] to determine the natural number of clusters.

Information-theoretical methods are also applicable for detecting the number of clusters in a dataset by defining a "jump method" of the transformed distortion d on a partition π_d. The highest increase of d indicates the number of clusters with respect to π_d [21]. However, this approach is based on a strong assumption that the clusters are generated based on Gaussian distributions. By integrating $R\acute{e}nyi$ entropy and complement entropy together, Liang *et al* [22] propose a method which can determine the number of clusters on a dataset that has mixed set of feature types. Their approach proposes a clustering validation index which considers within-cluster entropy and between-cluster entropy and the best number of clusters is chosen when such index reaches the maximum.

There also several other methods on detecting the number of clusters in a dataset. In [23] the Maximum Stable Set Problem (MSSP) combined by Continuous Hopfield Network (CHN) is

used to find the natural number of clusters of a data set. The algorithm detects the number of stable sets and uses this to represent the number of clusters.

Shaqsi and Wang [24], [25] work with a similarity parameter and observe that in a certain range of this parameter, the number of clusters formed by their 3-staged algorithm remains constant. The number of clusters that corresponds to the longest interval is chosen as the most appropriate number.

Kolesnikov, Trichina, Kauranne [26] create a parametric modeling of the quantization error to determine the optimal number of clusters in a dataset. This method treats the model parameter as the effective dimensionality of the dataset. By extending the decision-theoretic rough set model an efficient method to detect the number of clusters is presented in [27]. This model applies the Bayesian decision procedure for the construction of probabilistic approximations. Hamerly and Elkan [28] propose an algorithm that based on a statistical test for the hypothesis that a subset of data follows a Gaussian distribution. It only requires one parameter and avoids the calculation of the covariance matrix. An incremental approach called "dip-means", is introduced in [29] with the underlying assumption that each cluster admits a unimodal distribution. The statistic hypothesis test for unimodality (dip-test) is applied on the distribution of distances between one cluster member and others.

In [30] a new method for automatically detecting the number of clusters based on image processing techniques is discussed. This method adopts the key part of Visual Assessment of Cluster Tendency(VAT), and regards the dissimilarity matrix as an image matrix. Image segmentation techniques are applied to an image generated by this matrix, followed by filtering and smoothing to decide the number of clusters in the original data.

Cheung [31] proposes a new novel algorithm that can automatically select the number of clusters by presenting a mechanism to control the strength of rival penalization dynamically.

We propose a new method to evaluate the number of clusters using the metric space of partitions of a dataset. Using an extension of a seminal result of L. de Mántaras [32] we introduce a metric on partitions of finite sets defined by β-entropies of partitions (which generalize the Shannon entropy). The notion of β-entropy was introduced in [33], [34], and axiomatized in [35]. Other significant generalizations of entropy belong to C. Tsallis [36], [37].

Our approach seeks to optimize both clustering partition entropy and the cohesion of clustering. Since partition entropy is anti-monotonic and cluster cohesion are a monotonic function relative to the partial order set of partitions, we can use the Pareto Front to identify the natural number of clusters existent in a data set.

We emphasize that we seek to determine the optimal number of cluster considering clusterings produced by specific algorithms (e.g. k-means or hierarchical approaches) thereby avoiding prohibitively expansive searches over the entire space of partitions of a set.

The paper is organized as follows. In Section II partition entropy is introduced. Section III illustrates the compromise between cluster entropy and its corresponding cohesion and how it is used on determining the number of clusters. The experimental results are analyzed in Section IV. Conclusions and future work are discussed in Section V.

II. The Metric Space of Partitions of a Finite Set

Properties of generalized entropy defined on partition lattices were explored in [38].

Unless stated otherwise all sets are supposed to be finite. A partition of a set S is a non-empty collection of non-empty subsets of S, $\pi = \{B_1, \ldots, B_n\}$ such that $B_i \cap B_j = \emptyset$ for $i \neq j$ and $\bigcup_{i=1}^{n} B_i = S$. The set of partitions of a set S is denoted by $\mathsf{PART}(S)$.

If $\pi, \sigma \in \mathsf{PART}(S)$ we write $\pi \leqslant \sigma$ if every block of σ is a union of blocks of π. The relation "$\leqslant$" is a partial order on $\mathsf{PART}(S)$ having $\iota_S = \{\{x\} \mid x \in S\}$ as its least element and $\omega_S = \{S\}$ as its largest element, so $\iota_S \leqslant \pi \leqslant \omega_S$ for $\pi \in \mathsf{PART}(S)$. The partially ordered set $(S, \leqslant)$ is a lattice, where $\pi \wedge \sigma = \{B_i \cap C_j \mid B_i \in \pi, C_j \in \sigma \text{ and } B_i \cap C_j \neq \emptyset\}$. The other lattice operation, $\pi \vee \sigma$ has a more complicated description that can be found, for example, in [39].

The partition σ covers the partition π (denoted by $\pi \prec \sigma$) if $\pi \leqslant \sigma$ and there is no partition τ distinct from π and σ such that $\pi \leqslant \tau \leqslant \sigma$. It is known (see [40]) that $\pi \prec \sigma$ if and only if σ is obtained from π by fusing two blocks of π. Of course, if $\pi \leqslant \sigma$, there exists a chain of partitions $\tau_0, \tau_1, \ldots, \tau_n$ such that $\pi = \tau_0$, $\tau_i \prec \tau_{i+1}$ for $0 \leqslant i \leqslant n-1$ and $\tau_n = \sigma$.

If $\pi = \{B_1, \ldots, B_n\} \in \mathsf{PART}(S)$ and $C \subseteq S$, the *trace of π on C* is the partition $\pi_C \in \mathsf{PART}(C)$ given by $\pi_C = \{B_i \cap C \mid B_i \in \pi \text{ and } B_i \cap C \neq \emptyset\}$. Note that we have $\pi \leqslant \sigma$ if and only if $\sigma_B = \omega_B$ for every block B of π.

If $\pi = \{B_1, \ldots, B_n\}$ is a partition of a set S and $\beta > 0$, then its *β-entropy* (introduced in [33], [34]), H_β, is given by:

$$H_\beta(\pi) = \frac{1}{1 - 2^{1-\beta}} \left(1 - \sum_{i=1}^{n} \left(\frac{|B_i|}{|S|}\right)^{\beta}\right). \tag{1}$$

It is immediate that $H_\beta(\omega_S) = 0$.

Note that $\lim_{\beta \to 1} H_\beta = -\sum_{i=1}^{n} \frac{|B_i|}{|S|} \log \frac{|B_i|}{|S|}$, as it can be verified immediately by applying l'Hôpital rule. Thus, the Shannon entropy is a limit case of the generalized entropy.

Let $h_\beta : [0,1] \longrightarrow \mathbb{R}$ be defined by $h_\beta(x) = \frac{x - x^\beta}{1 - 2^{1-\beta}}$, where $\beta > 0$ and $\beta \neq 1$. Since $h''_\beta(x) = \frac{-\beta(\beta-1)x^{\beta-2}}{1-2^{1-\beta}}$, it follows that $h''_\beta(x) \leqslant 0$ because $1 - 2^{1-\beta} \geqslant 0$ when $\beta \geqslant 1$, and $1 - 2^{1-\beta} \leqslant 0$ when $\beta \leqslant 1$. Therefore, h_β is a concave function for $\beta > 0$ and $\beta \neq 1$.

We have $h_\beta\left(\frac{1}{2}\right) = \frac{1}{2}$; the maximum of h_β on the $[0,1]$ interval is achieved at $x_\beta = \beta^{-\frac{1}{\beta-1}}$ and equals $\frac{\beta-1}{1-2^{1-\beta}} \beta^{-\frac{\beta}{\beta-1}}$. The function h_β is subadditive for every $\beta \in (0,1) \cup (1, \infty)$, that is,

$$h_\beta(x + y) \leqslant h_\beta(x) + h_\beta(y)$$

for $x, y \in [0,1]$. Observe that $\lim_{\beta \to 1} h_\beta(x) = x \log_2 \frac{1}{x}$.

Since

$$H_\beta(\pi) = \frac{1}{1-2^{1-\beta}}\left(1-\sum_{i=1}^{n}\left(\frac{|B_i|}{|S|}\right)^\beta\right) = \sum_{i=1}^{n} h_\beta\left(\frac{|B_i|}{|S|}\right),$$

the concavity of h_β implies that the maximum value of $H_\beta(\pi)$ is achieved when $|B_1| = \cdots = |B_n|$ and is equal to $\frac{1}{1-2^{1-\beta}}\left(1-n^{1-\beta}\right)$. Thus, the maximal value of $H_\beta(\pi)$ is obtained when $\pi = \iota_S$ and it is equal to $\frac{1}{1-2^{1-\beta}}\left(1-|S|^{1-\beta}\right)$. Note that the minimal value of $H_\beta(\pi)$ is achieved when $\pi = \omega_S$, $H_\beta(\omega_S) = 0$ and $H_\beta(\pi) = 0$ implies $\pi = \omega_S$.

For $\beta = 2$ we obtain the well-known *Gini* index

$$H_2(\pi) = 2\left(1-\sum_{i=1}^{n}\left(\frac{|B_i|}{|S|}\right)^2\right).$$

Let $\{S_1, \ldots, S_n\}$ be a partition of the set S and let $\pi_1, \ldots, \pi_n$ be n partitions such that $\pi_i \in \mathsf{PART}(S_i)$ for $1 \leqslant i \leqslant n$. Define the partition $\pi_1 + \cdots + \pi_n$ as the partition of S that consists of all blocks of $\pi_1, \ldots, \pi_n$. Then,

$$H_\beta(\pi_1 + \ldots + \pi_n) = H_\beta(\{S_1, \ldots, S_n\}) + \sum_{i=1}^{n}\left(\frac{|S_i|}{|S|}\right)^\beta H_\beta(\pi_i).$$

If $\pi = \{B_1, \ldots, B_m\}$ and let $\sigma = \{C_1, \ldots, C_n\}$ are two partitions in $\mathsf{PART}(S)$, then

$$H_\beta(\pi \wedge \sigma) = H_\beta(\sigma) + \sum_{j=1}^{m}\left(\frac{|C_j|}{|S|}\right)^\beta H_\beta(\pi_{C_j}) = H_\beta(\pi) + \sum_{i=1}^{n}\left(\frac{|B_i|}{|S|}\right)^\beta H_\beta(\sigma_{B_i}).$$

The *conditional* β*-entropy* $H_\beta(\pi|\sigma)$ is defined as

$$H_\beta(\pi|\sigma) = H_\beta(\pi \wedge \sigma) - H_\beta(\sigma).$$

The β-entropy is anti-monotonic, that is, for $\beta \in \mathbb{R}_{>0} - \{1\}$ and $\pi, \sigma \in \mathsf{PART}(S)$, $\pi \leqslant \sigma$ implies $H_\beta(\sigma) \leqslant H_\beta(\pi)$. The conditional β-entropy $H_\beta(\pi|\sigma)$ is anti-monotonic in its first argument and monotonic in its second, that is $\pi_1 \leqslant \pi_2$ implies $H_\beta(\pi_1|\sigma) \geqslant H_\beta(\pi_2|\sigma)$ and $\sigma_1 \leqslant \sigma_2$ implies $H_\beta(\pi|\sigma_1) \geqslant H_\beta(\pi|\sigma_2)$.

A result obtained in [38] (which is a generalization of a result of [32]) shows that the function $d_\beta : \mathsf{PART}(S) \times \mathsf{PART}(S) \longrightarrow \mathbb{R}$ defined by $d_\beta(\pi, \sigma) = H_\beta(\pi|\sigma) + H_\beta(\sigma|\pi)$ is a metric on $\mathsf{PART}(S)$. This function will be used to evaluate distance between clusterings regarded as sets of objects.

III. Dual Criteria Clustering using Entropy and Cohesion

Partition entropy evaluates the imbalance between the sizes of the clusters that constitute a partition. For a fixed number of blocks, the entropy is maximal when blocks have equal sizes. As we saw in Section II, the smaller the partition in the poset $(\mathsf{PART}(S), \leqslant)$ the larger the entropy. Thus, the largest value of the entropy of a partition of S is achieved for ι_S; the smallest value is obtained for the one-block partition ω_S.

Cohesion is a measure of the quality of a clustering π, defined as the within-cluster sum of squared errors and denoted by $\mathsf{sse}(\pi)$.

Let S be the set of objects to be clustered. We assume that S is a subset of $\mathbb{R}^n$ equipped with the Euclidean metric. The *center* $\mathbf{c}_C$ *of a subset* C of S is defined as $\mathbf{c}_C = \frac{1}{|C|}\sum\{\mathbf{o} \mid \mathbf{o} \in C\}$.

For a partition $\pi = \{C_1, C_2, \ldots, C_m\}$ of S the sum of square errors sse of π is defined as

$$\mathsf{sse}(\pi) = \sum_{i=1}^{m}\sum_{\mathbf{o}\in C_i} d^2(\mathbf{o}, \mathbf{c}_{C_i}). \tag{2}$$

The next theorem can be found in [39].

Theorem III.1: Let $\kappa, \lambda \in \mathsf{PART}(S)$. If $\kappa \leqslant \lambda$, then $\mathsf{sse}(\kappa) \leq \mathsf{sse}(\lambda)$.

Proof. It suffices to prove this result for partitions κ, σ such that $\kappa \prec \sigma$. Suppose that $\kappa = \{C_1, \ldots, C_n\}$. Since $\kappa \prec \lambda$, the blocks of κ coincide with the blocks of λ with the exception of two blocks C_j and C_k of κ whose union is the block $C_j \cup C_k$ of λ. The difference in cohesion between λ and κ is

$$\begin{aligned}\mathsf{sse}(\lambda) - \mathsf{sse}(\kappa) &= \sum\{d^2(\mathbf{o}, \mathbf{c}_{C_j\cup C_k}) \mid \mathbf{o} \in C_j \cup C_k\} \\ &\quad - \sum\{d^2(\mathbf{o}, \mathbf{c}_{C_j}) \mid \mathbf{o} \in C_j\} \\ &\quad - \sum\{d^2(\mathbf{o}, \mathbf{c}_{C_k}) \mid \mathbf{o} \in C_k\}.\end{aligned} \tag{3}$$

Since the centroid of $C_j \cup C_k$ is

$$\mathbf{c}_{C_j\cup C_k} = \frac{1}{|C_j \cup C_k|}\sum\{\mathbf{o} \mid \mathbf{o} \in C_j \cup C_k\} = \frac{|C_j|}{|C_j \cup C_k|}\mathbf{c}_{C_j} + \frac{|C_k|}{|C_j \cup C_k|}\mathbf{c}_{C_k},$$

after elementary transformations we obtain

$$\sum\{d^2(\mathbf{o}, \mathbf{c}_{C_j\cup C_k}) - d^2(\mathbf{o}, \mathbf{c}_{C_j}) \mid \mathbf{o} \in C_j\} = \frac{|C_j||C_k|^2}{|C_j \cup C_k|^2}(\mathbf{c}_{C_k} - \mathbf{c}_{C_j})^2.$$

Similarly, we have:

$$\sum\{d^2(\mathbf{o}, \mathbf{c}_{C_j\cup C_k}) - d^2(\mathbf{o}, \mathbf{c}_{C_k}) \mid \mathbf{o} \in C_k\} = \frac{|C_j|^2|C_k|}{|C_j \cup C_k|^2}(\mathbf{c}_{C_k} - \mathbf{c}_{C_j})^2$$

The last two equalities imply

$$\mathsf{sse}(\lambda) - \mathsf{sse}(\kappa) = \frac{|C_j||C_k|}{|C_j \cup C_k|}(\mathbf{c}_{C_k} - \mathbf{c}_{C_j})^2 \geq 0. \tag{4}$$

Theorem III.1 shows that cohesion is an anti-monotonic function on the partially ordered set $(\mathsf{PART}(S), \leqslant)$; we have $\mathsf{sse}(\iota_S) = 0$ and $\mathsf{sse}(\omega_S) = \sum_{\mathbf{o}\in S} \| \mathbf{o} \|^2 - |S| \| \mathbf{c}_S \|^2$. Thus, the entropy varies inversely with the cohesion of partitions.

Entropy and cohesion describe the clustering result from two different perspectives and this suggest that a bi-criterial optimization would be helpful for choosing the best clusterings.

We aim to simultaneously minimize $H(\pi)$ and $\mathsf{sse}(\pi)$ that have inverse types of variations with clusterings considered as partitions. This will allow us to define a natural number of clusters using the Pareto front of this bi-criterial problem. Let $\mathbf{F} : \mathsf{PART}(S) \longrightarrow \mathbb{R}^2$, where

$$\mathbf{F}(\pi) = (H(\pi), \mathsf{sse}(\pi)), \tag{5}$$

where $\pi \in \mathsf{PART}(S)$.

Definition III.2: Let $\pi, \sigma \in \mathsf{PART}(S)$. The partition σ *dominates* π if $H(\sigma) \leqslant H(\pi)$ and $\mathsf{sse}(\sigma) \leqslant \mathsf{sse}(\pi)$.

A partition $\tau \in \mathsf{PART}(S)$ is *Pareto optimal* if there is no other partition that dominates τ.

In principle, several optimal partitions may exist, each with a specific number of clusters. The set of partitions that are not dominated by other partitions is the *Pareto front* of this problem (see [41], [42]).

If a partition π is Pareto optimal, then it is no worse than another partitions from the point of view of ($H(\pi)$ and $\mathsf{sse}(\pi)$) and is better in at least one of these criteria.

To speed up the search for the members of the Pareto front we scalarize the problem by computing a single objective optimization function defined utilizing the concept of hypervolume [43] on entropy and sse. The hypervolume measure is the size of the space covered or size of dominated space (see [43]), is the Lebesgue measure Λ of the union of hypercubes a_i defined by a non-dominated point m_i and a reference point x_{ref} [44].

In our case, we set the reference point at the position that both entropy and sse reaches its maximum. The maximum of entropy will be reached on partition ι_S, while the maximum value of sse is obtained at partition ω_S. Then, the hypervolume that corresponds to a partition π is

$$\mathsf{HV}(\pi) = (H(\iota_S) - H(\pi))(\mathsf{sse}(\omega_S) - \mathsf{sse}(\pi)) \tag{6}$$

The optimal partition for a dataset is obtained as

$$\pi_{opt} = \underset{\pi}{\operatorname{argmax}}\, \mathsf{HV}(\pi). \tag{7}$$

IV. Experimental Results

Our approach was tested on different datasets to evaluate their performance. We used 5 synthetic datasets and 7 real-world datasets.

The 2-dimensional synthetic datasets contain 5 Gaussian distributed clusters; each cluster contains 300 data points produced using the R function RMVNORM implemented by Leisch, F. *et al* [45]. By varying the means and standard deviations, we obtained five different types of clusterings. having the following features:

- clusters that are well separated;
- clusters that are well separated but closer with each other;
- clusters that have different density;
- clusters that have different sizes and number of points;
- clusters that overlap.

The data structures are shown in Figure 1.

Also, we used several real-world data sets which originate from UCI machine learning repository [46].

Iris Data: This dataset contains 150 cases and 4 variables named Sepal.Length, Sepal.Width, Petal.Length and Petal.Width corresponding to 3 species of iris (*setosa*, *versicolor*, and *virginica*) [47].

Wine Recognition Data: These data are the results of a chemical analysis of wines. The analysis determined the quantities of 13 constituents found in each of the three types of wines [48]. The distribution of three classes is as follows: class 1: 59; class 2: 71; class 3: 48.

LIBRAS Movement Database: LIBRAS, acronym of the Portuguese name "$L\acute{I}ngua$ BRAsileira de Sinais", is the official Brazilian sign language. The dataset contains 15 classes of 24 instances each, where each class refers to a hand movement type in LIBRAS. Each instance represents 45 points on a 2-dimensional space, which can be plotted in an ordered way (from 1 through 45 as the x-coordinate) in order to draw the path of the movement.

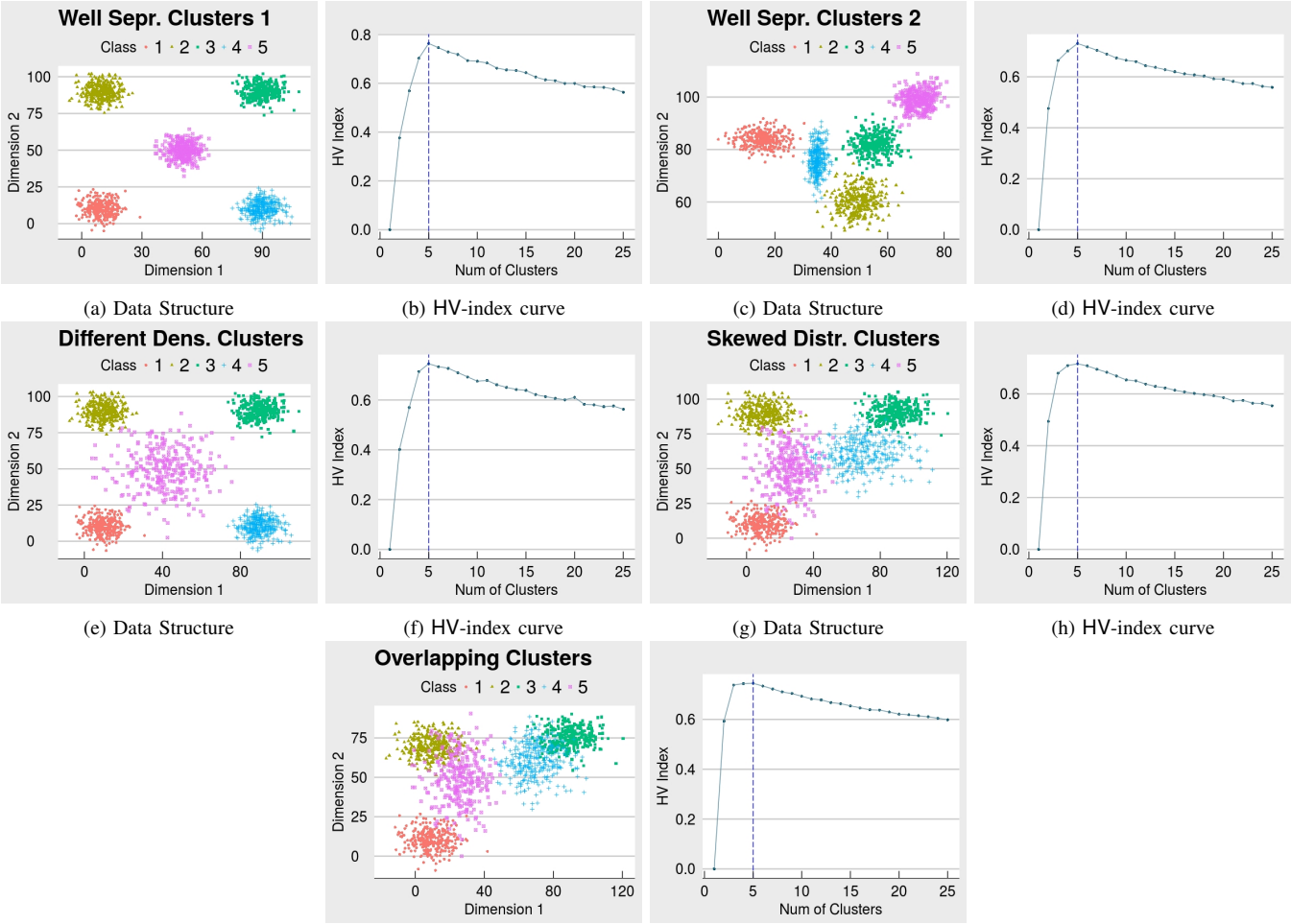
Pen-Based Recognition of Handwritten Digits: The digit database was created by collecting 250 samples from 44 writers. Digits are represented as feature vectors by using linear interpolation between pairs of (x_t, y_t) points. Here x_t and y_t is the coordinate information for the digits at when the writer is written. There are 10 different digits in the data set and the numbers of instance for each digits are roughly the same.

E.coli Dataset: This dataset contains 360 instances and 7 features and is used to predict the protein localization site. 8 classes are embedded into the dataset with the largest class of 143 data points and the smallest one of only 2.

Vowel Recognition: The dataset is generated from speakers' independent recognition of the eleven steady state vowels of British English using a specified training set of LPC derived log area ratios. It consists of a three-dimensional array: voweldata [speaker, vowel, input]. The speakers are indexed by integers 0-89. (Actually, there are fifteen individual speakers, each saying each vowel six times.) The vowels are indexed by integers 0-10. For each utterance, there are ten floating-point input values, with array indices 0-9. It has 990 instances.

Poker Dataset: This dataset records a set of card types people hold in their hands. Each record is an example of a hand consisting of five playing cards drawn from a standard deck of 52. Each card is described using two attributes (suit and rank), for a total of 10 predictive attributes. There is one Class attribute that describes the "Poker Hand". To enhance the performance, in our case, we neglect the Class 0 (Nothing in hand; not a recognized poker hand) due to its heavy weight on the number of data points.

To verify the stability of our method, four other popular methods on determining number of clusters are used for comparison.



(a) Data Structure (b) HV-index curve (c) Data Structure (d) HV-index curve

(e) Data Structure (f) HV-index curve (g) Data Structure (h) HV-index curve

(i) Data Structure (j) HV-index curve

Fig. 1: The original dataset and the HV-index of 5 synthesis datasets and their corresponding clustering structures; the x-axis of the HV-index graph represents the number of cluster of k-means clustering while the y-axis represents the index value.

1) *Gap Statistics:* This method gives the natural number of clusters by defining a gap function as follows:

$$Gap_n(k) = E_n^* \log(W_k) - \log(W_k),$$

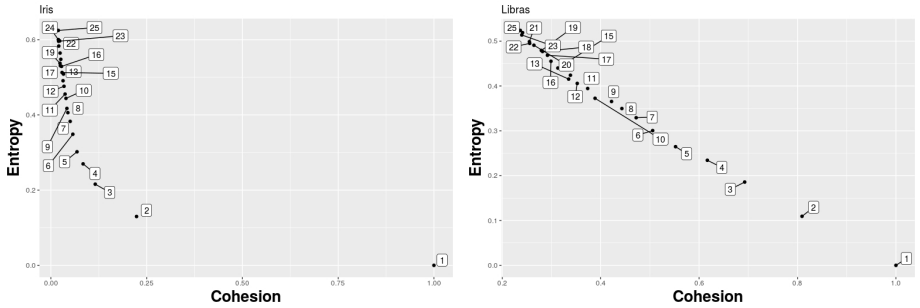
where E_n^* denotes the expectation under a sample of size n from the reference distribution, and W_k is the pooled within-cluster sum of the squares of distances between objects. The estimated $\hat{k}$ will be the value maximizing $Gap_n(k)$ after taking the sampling distribution into account [3].
The idea of this criterion is to standardize the graph of $\log(W_k)$ by comparing it with its expectation under an appropriate full reference distribution of the data. The estimate of the optimal number of clusters is then the value of k for which $\log(W_k)$ falls the farthest below this reference curve. We use the function `clusGap` in R package "cluster" for simulation [49].
2) *Jump method:* It uses the concept of distortion to describe the within-cluster dispersion for a particular partition [21]. The definition of the minimum achievable distortion associated with fitting k centers to the data is

$$d_k = \frac{1}{p} \min_{\mathbf{c}_1,\ldots,\mathbf{c}_k} E[(\mathbf{X}-\mathbf{c}_\mathbf{x})^T \Gamma^{-1} (\mathbf{X}-\mathbf{c}_\mathbf{x})], \quad (8)$$

where $\mathbf{X}$ is a p-dimensional random variable having a mixture distribution of K components, and each with covariance Γ. The $\mathbf{c}_1, \ldots, \mathbf{c}_k$ are a set of candidate cluster centers and $\mathbf{c}_\mathbf{x}$ is the one closest to $\mathbf{X}$.
Equality (8) gives the average Mahalanobis distance, per dimension, between $\mathbf{X}$ and $\mathbf{c}_\mathbf{x}$. If Γ is the identity matrix, distortion will be the mean squared error. The number of cluster k is determined as

$$k = \underset{k}{\operatorname{argmin}}\ d_k^{-Y} - d_{k-1}^{-Y},$$

where Y is an arbitrary value called transformation power and it usually equals to $\frac{p}{2}$.
3) *Prediction Strength:* For a particular dataset S, let $\mathcal{X}_{tr}$ and $\mathcal{X}_{te}$ be the training and testing subset of the data, where $\mathcal{X}_{tr} \cup \mathcal{X}_{te} = S$. Then we partition both $\mathcal{X}_{tr}$



(a) Pareto Front for Iris Dataset (b) Pareto Front for Libras Dataset

Fig. 2: The Pareto Front of solution of Equation (5) for Iris and Libras dataset using k-means clustering algorithm. The labelled points represents *Pareto optimal* partitions and the label shows the corresponding number of clusters. x-axis represents the cohesion while y-axis is the entropy. Both are normalized into $[0, 1]$.

and $\mathcal{X}_{te}$ into k clusters. Let $\pi_{tr} = A_1, \ldots, A_k$ and $\pi_{te} = B_1, \ldots, B_k$ be the partitions for $\mathcal{X}_{tr}$ and $\mathcal{X}_{te}$, respectively. The prediction strength of S given k is defined in [4] as

$$PS(k) = \min_{1 \leqslant l \leqslant k} \frac{\sum_{i \neq j} \{\delta((x_i, x_j), \mathcal{X}_{tr}) \mid x_i, x_j \in B_l\}}{|B_l|(|B_l| - 1)},$$

where δ is an indicator function. If we assign x_i and x_j to their nearest centroids in π_{tr}, say $\mathbf{c}_{A_i}$ and $\mathbf{c}_{A_j}$, respectively. If $A_i = A_j$, then $\delta((x_i, x_j), \mathcal{X}_{tr}) = 1$, otherwise, it is 0. This method is mainly implemented with the help of function `prediction.strength` in R package "fpc" [50].

4) *ACA-DTRS:* The Automatically Clustering Algorithm using Decision-Theoretic Rough Set model (ACA-DTRS) introduced in [27] can detect the number of clusters by utilizing the concept of rough set. It creates a clustering validity index $Risk(CS_t)$ based on the similarity matrix to guide the choice of a number of clusters.

Our method performs well on several synthetic datasets, as shown in Figure 1.

The HV-index is designed to solve the multi-objective Equality (5) which consider two general validation indexes of clustering.

Since we are seeking to minimize both the entropy and the cohesion, the region of feasible solutions should have a convex structure in the left-lower bound. If the algorithm can cluster the dataset well, the partition generated from it will be close to the bound of the region of feasible solution. Thus, the set of pairs $(H(\pi), \mathsf{sse}(\pi))$ for different partitions will form a convex curve. Figure 2 shows the pair of entropy and sse of k-means clustering results with different number of clusters on *Iris* and *Libras* dataset.

The HV index scalarizes the Equation (5) using the hypervolume indicator. Both entropy and cohesion are normalized to values in $[0, 1]$. The entropy on partition is defined as the generalized entropy in Equation (1) with parameter β. As illustrated in last part of Section III, different values of β will affect detection of the natural number of clusters. The β value we pick is given in Table I for each dataset.

In most cases, we choose $\beta = 1.0001$ in our experiments. As mentioned in Section II, $H(\mathbf{p}) = \lim_{\beta \to 1} H_\beta(\mathbf{p})$ is *Shannon Entropy*.

The natural number of clusters is successfully determined for all synthetic data sets. The HV index method worked for the dataset with overlapping clusters for $\beta = 0.95$ even if the value on 3 is also relatively high (Figure 1j).

Tables I and II give the results of the application of the algorithm on total of 12 different datasets (5 synthetic and 7 real datasets). The HV index works well on all of those 5 synthetic datasets (designated as Well-Separated I, Well-Separated II, Different Densities, Skewed Densities, and Overlapping Clusters). In Table II, we show that our algorithms outperform some existing algorithms both in detecting the correct number of clusters and in time performance. All experiments were performed on a 64-bit, Lenovo X1-Carbon laptop with Core i7 and 8GiB memory.

The flexibility afforded by generalized entropies allows choosing β to improve results in the case of imbalanced data sets.

Experiments suggest that small values of β may compensate for the sizes of small clusters and thus provide a more accurate estimation of the natural number of clusters. We verified this assumption on both synthetic (the skewed distributed dataset as shown in Figure 1g) and real data.

For data sets involved in the experiments a random portion of one of the clusters was removed and we sought to determine the number of clusters in the resulting imbalanced data using the dual criteria algorithm. In the *Iris* data set we eliminated a portion of the *versicolor* cluster. As shown in Figure 3a, 3b, to retrieve the correct number (in our case, it is 5 and 3) better results are obtained with values of β that are less than 1.

For the data set *Wine*, there are three unbalanced clusters: the size of the largest one is almost twice as the size of the smallest. To maintain consistency, we randomly removed 50% to 90% of the largest cluster, so that we can have roughly the same situation as in previous two examples. Figure 3c still

TABLE I: Comparison between the number of clusters for datasets

Data Sets	Cardinality of Datasets	actual no. of Classes	β	natural number of clusters			
				HV Index	Gap Stat.	Jump Mthd.	Pred. Strgth.
Well Sepr. I	900	5	1.00	**5**	5	5	3
Well Sepr. II	900	5	1.00	**5**	5	5	5
Diff. Density	900	5	1.00	**5**	5	5	5
Skewed Dist.	900	5	1.00	**5**	5	30	5
Overlapping	900	5	0.95	**5**	3	3	**5**
Iris	150	3	1.00	**3**	4	24	3
Libras	360	15	1.00	**13**	6	30	2
PenDigits	10992	10	1.20	**9**	22	29	6

TABLE II: Detecting the number of clusters for additional datasets including CPU times(s)

Data Sets	Cardinality of Datasets	actual no. of Classes	β	natural number of clusters				
				HV Index	Gap Stat.	Jump Mthd.	Pred. Strgth.	ACA-DTRS
Wine	178	3	1.00	4 (0.648)	1 (1.219)	28 (0.933)	**3** (2.006)	5 (2.11)
E. coli	336	8	0.9	**7** (0.646)	6 (1.899)	25 (1.3)	3 (1.957)	6 (13.791)
Vowel	990	11	0.8	**9** (1.353)	4 (5.672)	29 (1.525)	4 (2.896)	4 (52.203)
Poker(1-9)	511308	9	1.4	**10** (477.107)	4 (18.885)	29 (1574.23)	2 (2080.16)	8 (665.406)

shows a similar dependency of the quality of the clustering for smaller values of β.

For all three data sets, we recorded the average point of the range for each portion of the reduced cluster and apply linear regression. The regression results presented in Figure 3 show that regression lines have positive slopes, which indicate that smaller values of β yield better results for large imbalances created by reduction in size of one of the clusters.

V. Conclusions

We presented a new performant algorithm for the detection of the number of clusters in a dataset in the context of a given clustering algorithm (k-means).

Our heuristics seeks to determine the number of clusters existent in a data set as the number of blocks of a partition produced by specific algorithms (in our case, the k-means algorithm) that maximizes the hypervolume attached to these clusterings. Pareto Fronts are used utilized to identify the desired partition.

The consistency of results produced by experiments performed on both synthetic and real datasets using a variety of algorithms confirm that this technique gives a relatively cheaper technique comparing with existing methods.

We intend to focus our attention on clustering imbalanced data, where the generalized entropy and a metric generated by this entropy seem promising.

References

[1] A. K. Jain, "Data clustering: 50 years beyond k-means," *Pattern recognition letters*, vol. 31, no. 8, pp. 651–666, 2010.

[2] S. Salvador and P. Chan, "Determining the number of clusters/segments in hierarchical clustering/segmentation algorithms," in *Tools with Artificial Intelligence, 2004. ICTAI 2004. 16th IEEE International Conference on*, 2004, pp. 576–584.

[3] R. Tibshirani, G. Walther, and T. Hastie, "Estimating the number of clusters in a data set via the gap statistic," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 63, no. 2, pp. 411–423, 2001.

[4] R. Tibshirani and G. Walther, "Cluster validation by prediction strength," *Journal of Computational and Graphical Statistics*, vol. 14, no. 3, pp. 511–528, 2005.

[5] T. Chiu, D. Fang, J. Chen, Y. Wang, and C. Jeris, "A robust and scalable clustering algorithm for mixed type attributes in large database environment," in *Proceedings of the seventh ACM SIGKDD international conference on knowledge discovery and data mining*, 2001, pp. 263–268.

[6] M. Ester, H.-P. Kriegel, J. Sander, X. Xu *et al.*, "A density-based algorithm for discovering clusters in large spatial databases with noise." in *Kdd*, vol. 96, no. 34, 1996, pp. 226–231.

[7] A. Foss and O. R. Zaïane, "A parameterless method for efficiently discovering clusters of arbitrary shape in large datasets," in *Data Mining, 2002. ICDM 2003. Proceedings. 2002 IEEE International Conference on*, 2002, pp. 179–186.

[8] R. Scott Harris, D. R. Hess, and J. G. Venegas, "An objective analysis of the pressure-volume curve in the acute respiratory distress syndrome," *American Journal of Respiratory and Critical Care Medicine*, vol. 161, no. 2, pp. 432–439, 2000.

[9] S. Monti, P. Tamayo, J. Mesirov, and T. Golub, "Consensus clustering: a resampling-based method for class discovery and visualization of gene expression microarray data," *Machine learning*, vol. 52, no. 1-2, pp. 91–118, 2003.

[10] V. Roth, T. Lange, M. Braun, and J. Buhmann, "A resampling approach to cluster validation," in *Compstat*. Springer, 2002, pp. 123–128.

[11] A. Gionis, H. Mannila, and P. Tsaparas, "Clustering aggregation," *TKDD*, vol. 1, no. 1, pp. 1–30, 2007.

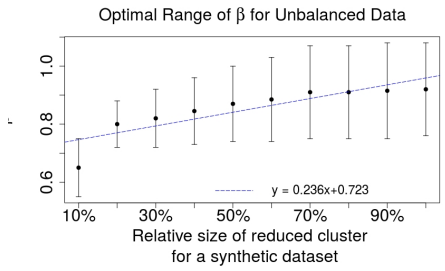
[12] T. Li and C. H. Q. Ding, "Weighted consensus clustering," in *Proceedings of the SIAM International Conference on Data Mining, SDM 2008, April 24-26, 2008, Atlanta, Georgia, USA*, 2008, pp. 798–809.

[13] T. Li, C. H. Q. Ding, and M. I. Jordan, "Solving consensus and semi-supervised clustering problems using nonnegative matrix factorization," in *Proceedings of the 7th IEEE International Conference on Data Mining (ICDM 2007), October 28-31, 2007, Omaha, Nebraska, USA*, 2007, pp. 577–582.

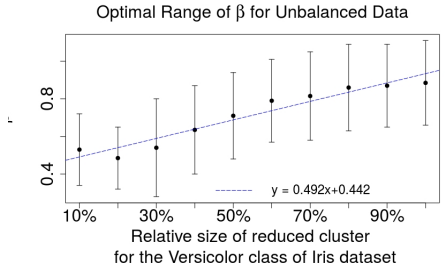
[14] C. H. Q. Ding and X. He, "Cluster aggregate inequality and multi-level hierarchical clustering," in *Knowledge Discovery in Databases: PKDD 2005, 9th European Conference on Principles and Practice of Knowledge Discovery in Databases, Porto, Portugal, October 3-7, 2005, Proceedings*, 2005, pp. 71–83.

[15] J. Azimi and X. Fern, "Adaptive cluster ensemble selection," in *IJCAI 2009, Proceedings of the 21st International Joint Conference on Artificial Intelligence, Pasadena, California, USA, July 11-17, 2009*, 2009, pp. 992–997.

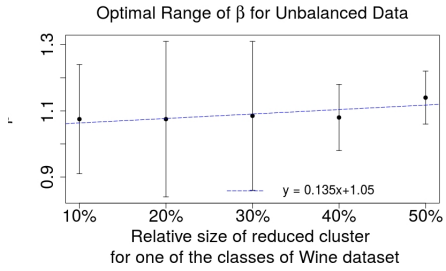
[16] X. Z. Fern and C. E. Brodley, "Solving cluster ensemble problems by bipartite graph partitioning," in *Machine Learning, Proceedings of the Twenty-first International Conference (ICML 2004), Banff, Alberta, Canada, July 4-8, 2004*, 2004.



(a) $k = 5$, Synthetic Data



(b) $k = 3$, Iris Data



(c) $k = 3$, Wine Data

Fig. 3: Range of β that yields correct k clusters for the modified dataset.

[17] H. Liu, T. Liu, J. Wu, D. Tao, and Y. Fu, "Spectral ensemble clustering," in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2015, pp. 715–724.

[18] J. Wu, H. Liu, H. Xiong, J. Cao, and J. Chen, "K-means-based consensus clustering: A unified view," *IEEE Transactions on Knowledge and Data Engineering*, vol. 27, no. 1, pp. 155–169, 2015.

[19] L. Zheng, T. Li, and C. H. Q. Ding, "A framework for hierarchical ensemble clustering," *TKDD*, vol. 9, no. 2, pp. 9:1–9:23, 2014.

[20] G. von Winckel, "Determining the number of clusters via iterative consensus clustering," 2013.

[21] C. A. Sugar and G. M. James, "Finding the number of clusters in a dataset," *Journal of the American Statistical Association*, 2011.

[22] J. Liang, X. Zhao, D. Li, F. Cao, and C. Dang, "Determining the number of clusters using information entropy for mixed data," *Pattern Recognition*, vol. 45, no. 6, pp. 2251–2265, 2012.

[23] A. Karim, C. Loqman, and J. Boumhidi, "Determining the number of clusters using neural network and max stable set problem," *Procedia Computer Science*, vol. 127, pp. 16–25, 2018.

[24] J. Al-Shaqsi and W.Wang, "A novel three staged clustering algorithm," in *IADIS European Conference on Data Mining, Algarve,Portugal*, 2009, pp. 9–16.

[25] J. Al-Shaqsi and W. Wang, "Estimating the predominant number of clusters in a dataset," *Intelligent Data Analysis*, vol. 17, no. 4, pp. 603–626, 2013.

[26] A. Kolesnikov, E. Trichina, and T. Kauranne, "Estimating the number of clusters in a numerical data set via quantization error modeling," *Pattern Recognition*, vol. 48, no. 3, pp. 941–952, 2015.

[27] H. Yu, Z. Liu, and G. Wang, "An automatic method to determine the number of clusters using decision-theoretic rough set," *International Journal of Approximate Reasoning*, vol. 55, no. 1, pp. 101–115, 2014.

[28] G. Hamerly and C. Elkan, "Learning the k in k-means," in *Advances in neural information processing systems*, 2004, pp. 281–288.

[29] A. Kalogeratos and A. Likas, "Dip-means: an incremental clustering method for estimating the number of clusters," in *Advances in neural information processing systems*, 2012, pp. 2393–2401.

[30] L. Wang, C. Leckie, K. Ramamohanarao, and J. Bezdek, "Automatically determining the number of clusters in unlabeled data sets," *IEEE Transactions on knowledge and Data Engineering*, vol. 21, no. 3, pp. 335–350, 2009.

[31] Y.-m. Cheung, "On rival penalization controlled competitive learning for clustering with automatic cluster number selection," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 11, pp. 1583–1588, 2005.

[32] R. L. de Mántaras, "A distance-based attribute selection measure for decision tree induction," *Machine Learning*, vol. 6, pp. 81–92, 1991.

[33] Z. Daroczy, "Generalized information function," *Information and Control*, vol. 16, pp. 36–51, 1970.

[34] J. F. Havrda and F. Charvat, "Qualification method of classification processes, the concept of structural k-entropy," *Kybernetika*, vol. 3, pp. 30–35, 1967.

[35] D. A. Simovici and S. Jaroszewicz, "An axiomatization of partition entropy," *IEEE Trans. Information Theory*, vol. 48, no. 7, pp. 2138–2142, 2002.

[36] C. Tsallis, "Possible generalization of Boltzmann-Gibbs statistics," *Journal of Statistical Physics*, vol. 52, pp. 479–487, 1988.

[37] S. D. Silva and P. Rathie, "Shannon, Lévy and Tsallis: A note," *Applied Mathematical Sciences*, vol. 2, pp. 1359–1363, 2008.

[38] D. A. Simovici, "On generalized entropy and entropic metrics," *Multiple-Valued Logic and Soft Computing*, vol. 13, no. 4-6, pp. 295–320, 2007.

[39] D. Simovici and C. Djeraba, *Mathematical Tools for Data Mining*, 2nd ed. London: Springer, 2014.

[40] G. Birkhoff, *Lattice Theory*, 3rd ed. Providence, RI: American Mathematical Society, 1973.

[41] R. T. Marler and J. S. Arora, "Survey of multi-objective optimization methods for engineering," *Structural and multidisciplinary optimization*, vol. 26, no. 6, pp. 369–395, 2004.

[42] V. Pareto, *Manuale di economia politica*. Societa Editrice, 1906, vol. 13.

[43] E. Zitzler and L. Thiele, "Multiobjective optimization using evolutionary algorithms-a comparative case study," in *International Conference on Parallel Problem Solving from Nature*. Springer, 1998, pp. 292–301.

[44] C. A. Coello, G. B. Lamont, D. Van Veldhuizen *et al.*, *Evolutionary algorithms for solving multi-objective problems*. Springer, 2007.

[45] A. Genz, F. Bretz, T. Miwa, X. Mi, F. Leisch, F. Scheipl, B. Bornkamp, M. Maechler, T. Hothorn, and M. T. Hothorn, "Package mvtnorm," 2016.

[46] M. Lichman, "UCI machine learning repository," 2013. [Online]. Available: http://archive.ics.uci.edu/ml

[47] R. A. Becker, J. M. Chambers, and A. R. Wilks, "The new s language," *Pacific Grove, Ca.: Wadsworth & Brooks, 1988*, vol. 1, 1988.

[48] S. Aeberhard, D. Coomans, and O. D. Vel, "Comparison of classifiers in high dimensional settings," *Dept. Math. Statist., James Cook Univ., North Queensland, Australia, Tech. Rep*, no. 92-02, 1992.

[49] M. Maechler, P. Rousseeuw, A. Struyf, M. Hubert, and K. Hornik, *cluster: Cluster Analysis Basics and Extensions*, 2016, R package version 2.0.5 — For new features, see the 'Changelog' file (in the package source).

[50] C. Hennig, *fpc: Flexible Procedures for Clustering*, 2015, R package version 2.1-10. [Online]. Available: https://CRAN.R-project.org/package=fpc