



MASTERARBEIT | MASTER'S THESIS

Titel | Title

Measuring Quality of Unsupervised Learning: Evaluation of
Density-Based Clustering

verfasst von | submitted by

Marcus Artnr BSc

angestrebter akademischer Grad | in partial fulfilment of the requirements for the degree of
Master of Science (MSc)

Wien | Vienna, 2024

Studienkennzahl lt. Studienblatt | Degree
programme code as it appears on the
student record sheet:

UA 066 645

Studienrichtung lt. Studienblatt | Degree
programme as it appears on the student
record sheet:

Masterstudium Data Science

Betreut von | Supervisor:

Univ.-Prof. Dipl.-Inform.Univ. Dr. Claudia Plant

Mitbetreut von | Co-Supervisor:

Dr. Anna Beer BSc MSc

Acknowledgements

Ich widme diese Arbeit meiner Mutter und meinen Großeltern, welche mich in jeder ihnen möglichen Form während meiner gesamten akademischen Laufbahn unterstützt haben. Ich möchte mich auch bei allen Freunden bedanken, die mir während des letzten Jahres geholfen haben, Arbeit und Freizeit in Einklang zu bringen. Zuletzt möchte ich mich noch bei meinen Betreuerinnen Prof. Claudia Plant und Dr. Anna Beer bedanken, die diese Arbeit überhaupt erst ermöglicht haben.

Abstract

Despite the existence of appropriate evaluation metrics, it remains a significant challenge to assess the quality of a clustering that contains density-based clusters. This can be attributed to both individual weaknesses in the design of existing measures as well as the inherent complexity of the subject.

In this work, we break down three existing density-based metrics into their core components and integrate them into a unified framework that allows for intuitive comparisons. These metrics, along with the recently developed Density-based Internal Evaluation Score for Clustering Outcomes (DISCO), are thoroughly tested in a series of experiments. The insights gained from these experiments are used to identify a collection of real-world, density-based datasets, which does not yet exist in this form but is crucial for advancing and improving density-based clustering methods. Furthermore, we show that dimensionality reduction using UMAP can have unpredictable effects on the structural properties of these datasets. Additionally, we create a Medium blog post that introduces DISCO with practical examples and intuitive explanations, making it accessible to a broader audience.

Kurzfassung

Es ist trotz bereits entsprechender existierende Evaluationsmetriken eine große Herausforderung, die Qualität eines Clusterings zu bestimmen, welches dichte-basierte Cluster enthält. Dies lässt sich auf individuelle Schwächen in der Konstruktion der bereits existierenden Maße, als auch auf die inhärente Komplexität der Thematik zurückführen.

In dieser Arbeit zerlegen wir drei bestehende dichte-basierte Metriken in ihre wesentlichen Bestandteile und überführen diese in ein vereinheitlichtes Framework, welches intuitive Vergleiche zulässt. Diese Metriken werden gemeinsam mit dem kürzlich entwickelten "Density-based Internal Evaluation Score for Clustering Outcomes" (DISCO) in einer Reihe von Experimenten ausführlich getestet. Die gewonnenen Erkenntnisse dienen dazu, eine Sammlung von "real-world", dichte-basierten Datensätzen zu identifizieren, welche in dieser Form bisher nicht existiert, jedoch erheblich für die Weiterentwicklung und Verbesserung dichte-basierter Clustering-Methoden ist. Weiters zeigen wir, dass eine Dimensionalitätsreduktion mittels UMAP unvorhersehbare Auswirkungen auf die strukturellen Eigenschaften dieser Datensätze haben kann. Darüber hinaus erstellen wir einen Medium-Blogpost, der DISCO anhand praktischer Beispiele und intuitiver Erklärungen vorstellt, um es einem breiteren Publikum zugänglich zu machen.

Contents

Acknowledgements	i
Abstract	iii
Kurzfassung	v
List of Tables	ix
List of Figures	xi
1. Introduction	1
2. Related Work	3
2.1. Internal CVIs	3
2.1.1. Non Density-Based CVIs	3
2.1.2. Density-Based CVIs	4
3. Overview of Existing Density-Based CVIs	7
3.1. CVDD	7
3.2. DBCV	9
3.3. DCSI	10
4. On Evaluating Density-Based Clustering Methods	13
4.1. Density-Connectivity	13
4.2. A Novel Measure for Density-Based Clustering	15
4.3. A Tutorial for DISCO, a Novel Density-Based Evaluation Measure	16
4.4. Data for Evaluating Density-Based Clustering Methods	16
5. Experiments	19
5.1. Synthetic Datasets	19
5.1.1. Experiment 1: Reducing the Distance Between Clusters	22
5.1.2. Experiment 2: Increasing the Variance of Clusters	24
5.1.3. Experiment 3: Incorrectly Clustered Points	26
5.1.4. Experiment 4: Increasing the Noise Ratio	28
5.1.5. Experiment 5: Arbitrarily Shaped Clusters With Different Densities	30
5.2. Real-World Datasets	32
5.2.1. UMAP Projections for Density-Based Structure Evaluation	34

Contents

5.3. Results	36
5.3.1. Analysis of Findings on Synthetic Datasets	36
5.3.2. Identification and Analysis of Density-Based Datasets	37
5.3.3. Robustness of Density-Based Structure in UMAP Projections	37
6. Conclusion	39
Bibliography	41
A. Appendix	43
A.1. Simplified Overview	43
A.2. Raw Data	45
A.3. Function Parameters	49

List of Tables

5.1. Results of different CVIs on datasets commonly used as benchmarks in density-based clustering studies. *contains noise	32
5.2. Experimental Results on 10-d UMAP embeddings of Datasets	34
5.3. Experimental Results on 2-d UMAP embeddings of Datasets	35
5.4. "likely density-based" datasets	37
A.1. Overview of considerations regarding some common parameters of existing CVIs	44
A.2. Evaluation of different metrics on two clusters of different dimensionality with decreasing distance between centers	45
A.3. Results of different CVIs on 2-d clusterings with increasing variance and fixed distance between centers	46
A.4. Evaluation of different metrics on two clusters of different dimensionality with increasing variance	46
A.5. Results of different CVIs on 2-d clusterings with fixed distance between centers and varying number of random incorrectly clustered points.	47
A.6. Evaluation of different metrics on four different datasets with clusters of arbitrary shapes and different densities across different dimensionalities.	47
A.7. Evaluation of different metrics on two clusters of different densities across different dimensionalities with increasing noise ratio.	48
A.8. Parameters used for dataset creation in experiments 4 and 5	49

List of Figures

2.2.	Distribution of DBCV scores over 50 runs on the <code>make_blobs</code> [PVG ⁺ 11] dataset with parameters $n_samples = 400$, $centers = 4$, $n_features = 2$, $random_state = 3$ and $cluster_std = 3$. The data is shuffled in every run.	6
4.1.	Two Moons dataset with $n_samples = 150$ and $noise = 0.05$	13
4.2.	Comparison of sorted Euclidean and dc-distance matrices for the Two Moons dataset: The left heatmap displays the sorted Euclidean distance matrix, while the right shows the sorted dc-distance matrix.	14
4.3.	Illustration of the expected properties of datasets used in the evaluation of density-based clustering methods: some datasets are density-based, some exhibit pairwise linear separability, and a few demonstrate both characteristics.	17
5.1.	Pairwise comparisons of dimensions of two 5-dimensional hypersphere clusters, where the difference between cluster centers exists only along the x_1 dimension. The scatter plots illustrate the relationships between different dimensions, while the diagonal histograms show the distribution of each dimension. Notice the clear separation in the x_1 dimension.	20
5.2.	Two dimensional toy example of the generation of a uniformly distributed (hyper)sphere cluster	21
5.3.	2-d clusterings with fixed variance and decreasing distance between centers	22
5.4.	Results for different CVIs on two clusters with decreasing distance between centers for different dimensionalities	23
5.5.	2-d clusterings with fixed centers and increasing variances	24
5.6.	Results for different CVIs on two clusters with fixed distance between centers and increasing variance for different dimensionalities	25
5.7.	2-d clusterings with fixed centers and varying number of random incorrectly clustered points.	26
5.8.	Results of different CVIs on 2-d clusterings with fixed distance between centers and varying number of random incorrectly clustered points.	27
5.9.	Two 2-d clusters with different densities and increasing noise ratios	28
5.10.	Results for different CVIs on two clusters of different densities across different dimensionalities with increasing noise ratio.	29
5.11.	2-d clusters with arbitrary shapes and different densities	30
5.12.	Evaluation of different metrics on four different datasets with clusters of arbitrary shapes and different densities across different dimensionalities.	31

1. Introduction

In the age of the internet, an immense amount of data is collected every second. However, making sense of this data without some form of simplification is often challenging. One effective method of structuring this data meaningfully is through clustering methods. It is widely acknowledged that there is not a single correct way to clustering data; rather, various methods can be applied based on the specific characteristics of the data. Hierarchical clustering, partition-based clustering, and density-based clustering are among the widely used and proven techniques.

Given that the ground truth of a clustering is typically unknown, and there may not even be a singular "ground truth", there is a need for measuring the quality of a clustering solution. One can distinguish between an internal and an external evaluation metric, also known as Cluster Validity Index (CVI). An external evaluation metric compares the clustering solution with the ground truth, whereas an internal metric does not rely on the ground truth. Well-known internal evaluation metrics include the Silhouette Coefficient [Rou87] or the Dunn Index [Dun73]. While these metrics perform well on clusterings of spherical shape, they often struggle to assess the quality of clusterings with arbitrary shapes, such as those that can be obtained through density-based clustering algorithms like DBSCAN [EKSX].

Over the last decade, various groups of researchers have introduced CVIs specifically designed to evaluate density-based clustering solutions, such as the density cluster separability index (DCSI) [GSH23], the Density-Based Clustering Validation index (DBCV) [MAJC⁺14], or the Cluster Validity index based on Density-involved Distance (CVDD) [HZ19]. However, these CVIs do have some shortcomings, ranging from the way they (do not) handle noisy data to a lack of interpretability to struggles in capturing good clusterings in high dimensions or cases with multiple values per clustering.

Furthermore, while these methods have been tested on both synthetic density-based datasets and real-world datasets, there is no evidence provided that the real-world datasets used for evaluation are indeed density-based.

This thesis raises several important questions and challenges. To accurately interpret the results of different metrics, a strong theoretical understanding of their building blocks is essential. Part of this understanding involves the ability to compare the inner workings of each metric. However, the metrics examined in this work often use different terminologies to describe similar concepts, making direct comparisons challenging. To address this, the key properties are broken down and re-embedded into a unified framework with consistent terminology. While a solid theoretical foundation is essential, determining when a CVI fails to assess a reliable clustering requires practical application. Therefore, a systematic evaluation of the three CVIs briefly introduced above, along with the recently developed Density-based Internal Evaluation Score for Clustering Outcomes (DISCO)

1. Introduction

[tKW⁺], is conducted through carefully designed experiments to uncover the strengths and weaknesses of each measure. As mentioned earlier, real-world datasets used to evaluate density-based methods are often only assumed to be density-based. To address this, we identify a collection of datasets that are more likely to exhibit density-based properties, using a novel and robust methodology. This collection provides a more reliable alternative to the current selection of datasets that may or may not be density-based. Furthermore, we provide an intuitive understanding of the DISCO CVI for non-experts through a blog post. For that, a clear and concise explanation of complex ideas, along with practical demonstrations of DISCO’s ability to handle noise effectively is required.

At the beginning of this thesis, the formulas of the existing measures—DCSI, DBCV, and CVDD—are broken down into comprehensible parts for easy reference and embedded into a framework with unified terminology. Then, the new CVI, DISCO, is introduced and explained. This explanation is accompanied by a blog post aimed primarily at non-researchers.

The next steps involve creating a number of synthetic datasets covering different scenarios to test the practical properties of each CVI. These findings are utilized to identify which real-world datasets, used in the evaluation of existing density-based clustering methods, are truly density-based. More specifically, this is achieved by applying classic CVIs, namely the Silhouette Coefficient and the Dunn Index, as well as the reference measures to the ground truth labels of these datasets. Lower values on classic CVIs paired with higher values on the density-based reference measures indicate that the dataset is a suitable benchmark to evaluate algorithms aimed at data containing density-based clusters. The findings won during those experiments are taken into consideration in the interpretation of all scores. Lastly, the effects of a UMAP [MHM20] dimensionality reduction on the global density-based structure of datasets is investigated.

The primary contributions of this thesis include:

- A comprehensive survey and discussion of state-of-the-art methods for density-based clustering evaluation, as well as an overview with unified terminology.
- Extensive experiments across multiple dimensions exploring different settings, revealing strengths and limitations of the CVIs: DCSI, CVDD, DBCV, and DISCO.
- A collection of density-based real-world datasets, identified through a methodology based on the findings of our experiments.
- A blog post covering the novel CVI, DISCO, including an intuitive breakdown of its inner workings as well as comprehensible demonstrations of its noise handling capabilities.

2. Related Work

This section introduces brief explanations of the workings of some established internal CVIs. We compare density-based and non-density-based CVIs, highlighting their advantages and disadvantages briefly. A more detailed breakdown of density-based CVIs is provided in Chapter 3.

2.1. Internal CVIs

To internally evaluate a clustering, labels and data are needed. The labels are obtained through any clustering algorithm and the data is already given. The general idea is to measure intra-cluster distances and inter-cluster distances and relate them in a meaningful way. An internal CVI should yield "good" values for clusterings that constitute of compact clusters with high separation between most clusters. Two famous examples of internal CVIs are the Silhouette Coefficient [Rou87] and the Dunn Index [Dun73], which are not density-based.

2.1.1. Non Density-Based CVIs

Non-density-based CVIs can successfully be used to evaluate the quality of spherical clustering solutions. While this approach is often sufficient, it has significant limitations. Specifically, if there are not only spherical clusters, it lacks the flexibility to effectively handle these arbitrarily shaped or intertwined clusters.

Silhouette Coefficient

A well known CVI that has been used extensively over the last decades is the Silhouette Coefficient. The primary idea of it is to compute silhouettes for every data point, assessing whether that data point is clustered well or not. Intuitively, the silhouette of a point measures whether the points in its own cluster are, on average, closer than the points in other clusters. If they are, the silhouette value is positive; if not, it is negative. [Rou87]

The Silhouette Coefficient $S_{silhouette}$ for a dataset X is defined as [Rou87]:

$$S_{silhouette} = \frac{1}{|X|} \cdot \sum_{x_i \in X} s(x_i), \quad (2.1)$$

where $s(x_i)$ is the silhouette for a data point x_i given by [Rou87]:

2. Related Work

$$s(x_i) = \frac{b(x_i) - a(x_i)}{\max(a(x_i), b(x_i))}. \quad (2.2)$$

Here, $a(x_i)$ is the average distance between the data point x_i and all other points in the same cluster. $b(x_i)$ is the minimum average distance from the data point x_i to all points in the nearest cluster that x_i does not belong to. The Silhouette Coefficient ranges from -1 to 1. On a data point level a value of 1 indicates a well clustered data point while a value of -1 indicates a likely incorrectly clustered data point. This range makes the Silhouette Coefficient easily interpretable, which probably contributes to its success. A downside about this CVI, however is that it does not take arbitrarily shaped clusters into account. One can see this by looking at equation 2.2, which does not contain any density capturing elements.

Dunn Index

An equally popular CVI as the Silhouette Coefficient is the Dunn Index. In essence, the Dunn Index quantifies clustering quality by comparing the largest intra-cluster distance with the smallest inter-cluster distance through division. If the largest intra-cluster distance is smaller than the smallest inter-cluster distance, the Dunn Index will be greater than 1. The more compact the largest cluster and the farther apart the closest clusters are, the higher the Dunn Index will be. [Dun73]

$$S_{dunn} = \frac{\min_{1 \leq i < j \leq K} \delta(C_i, C_j)}{\max_{1 \leq l \leq K} \Delta(C_l)} \quad (2.3)$$

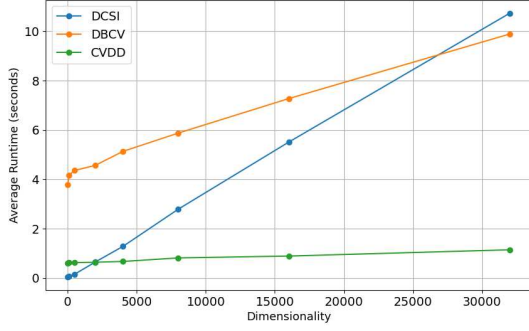
The parameter K is the number of clusters. $\delta(C_i, C_j)$ is the distance between clusters C_i and C_j while $\Delta(C_l)$ is the intra-cluster distance of cluster C_l . The Dunn Index S_{dunn} lies between 0 and ∞ . A higher score indicates better clustering quality. Like the Silhouette Coefficient, the Dunn Index follows a rather simple computation, also not taking any density capturing methods into account, making the assessment of arbitrarily shaped clusters rather poor. Another downside, compared to the Silhouette Coefficient, is that nothing can be said about the clustering quality at the individual data point level.

2.1.2. Density-Based CVIs

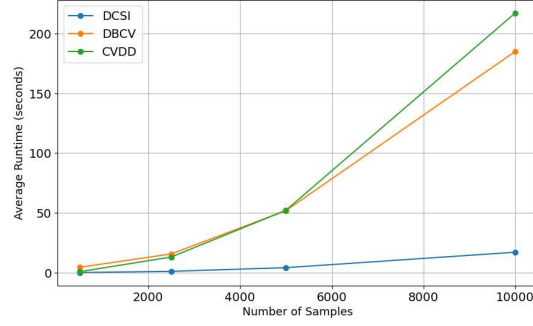
The idea behind density-based CVIs is to evaluate the quality of clusters that are not limited to spherical shapes. These indices are designed to be sensitive to arbitrarily shaped clusters, allowing for a more comprehensive assessment. Furthermore, density-based CVIs are, at least in theory, capable of detecting clusters that may be separated by density rather than by space, which traditional CVIs often fail to recognize.

The three popular CVIs DCSI [GSH23], DBCV [MAJC⁺14] and CVDD [HZ19] all use different approaches to assess the quality of density-based clusters. DCSI, for example, relies on the parameters ε and $minPts$ to define *core points* and thereby identifying dense regions [GSH23]. A *core point* is a point with at least $minPts$ points inside its

2.1. Internal CVIs



(a) Runtime of DCSI, DBCV and CVDD averaged over 5 trials for a random $500 \times d$ matrix, with values 2, 100, 500, 2000, 4000, 8000, 16000 and 32000 for d .



(b) Runtime of DCSI, DBCV and CVDD averaged over 5 trials for a random $n \times 50$ matrix, with values 500, 2500, 5000 and 10000 for n .

ϵ -neighborhood. This is a well known approach, as also the algorithm DBSCAN [EKSX] measures dense regions using *core points*, however the output of DCSI is thus sensitive to these two parameters. In contrast, DBCV captures dense regions by using the *all-points-core-distance* [MAJC⁺14]. As the name suggests, it is computed using all the points in a cluster, thereby eliminating the need for the parameters ϵ or *minPts*. While this avoids the tuning of two parameters, the computation of the *all-points-core-distance* relies on the dimensionality in the exponent to weigh closer points more than points which are far away. This tends to often generate extremely small and sometimes extremely large values for high dimensional datasets, necessitating the use of very high precision, which inherently slows down the runtime. Figure 2.1a visualizes how the average runtime scales with an increasing number of dimensions and Figure 2.1b depicts how it scales with an increasing sample size. Notice how in the left figure DBCV has a sublinear runtime and DCSI a linear one, but because DBCV requires such a high precision (512 bits seems sufficient) it only runs faster than DCSI for a 32000 dimensional dataset. We can also see that the sample size has a stronger effect on the total runtime for DBCV and CVDD than the dimensionality of the dataset. The measure CVDD [HZ19] uses a parameter k to average the distances of the k -nearest neighbors of a point, to estimate the density of a point. To make this density estimate more robust against outliers, an outlier factor and ultimately an outlier penalized distance are computed [HZ19]. CVDD yields values between 0 and ∞ , making the interpretation aspect difficult if there are no reference values.

2. Related Work

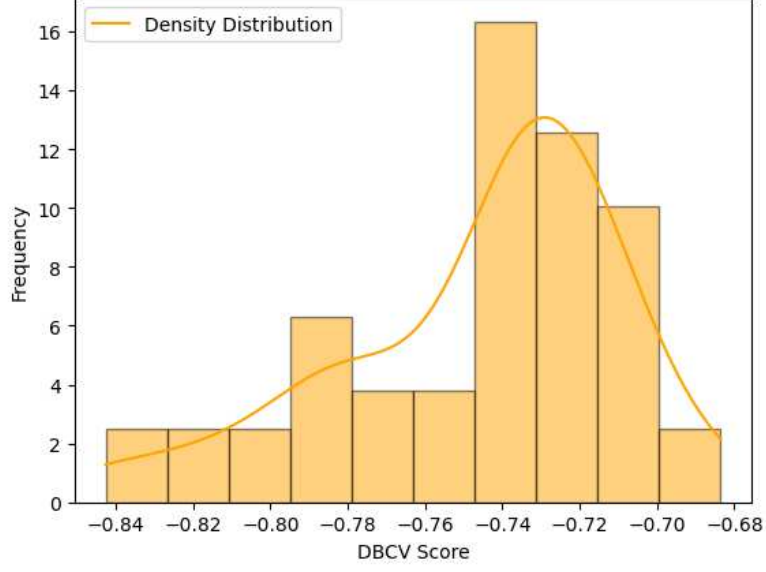


Figure 2.2.: Distribution of DBCV scores over 50 runs on the `make_blobs` [PVG⁺11] dataset with parameters $n_samples = 400$, $centers = 4$, $n_features = 2$, $random_state = 3$ and $cluster_std = 3$. The data is shuffled in every run.

All of the CVIs rely on a minimax distance, which is the longest edge on the shortest path between two points. Note that this can be done by using the maximum edge weight on a Minimum Spanning Tree (MST) [HC20]:

$$D(x_i, x_j)^{MM} = \max_{e \in P(x_i, x_j)} w^\delta(e), \quad (2.4)$$

with $P(x_i, x_j)$ being the set of edges on the MST between points x_i and x_j and δ as the distance metric the MST is built on.

For the construction of the MST, DBCV relies on its mutual reachability distance. However, for this specific measure, it leads to non-deterministic results, which is highly undesirable. A demonstration of this is shown in Figure 2.2. Lastly, it is important to note that none of the discussed measures actively handle noise.

3. Overview of Existing Density-Based CVIs

In this chapter, we present a detailed analysis of three prominent CVIs: CVDD [HZ19], DBCV [MAJC⁺14], and DCSI [GSH23]. By breaking down the underlying formulas of each CVI, we provide a deeper understanding of their theoretical foundations and practical applications. Additionally, this breakdown offers a clearer comparison of the mathematical principles governing these indices, revealing the unique strengths and limitations of each metric. A table with a simplified version of this breakdown can be found in the appendix at A.1. All formulas in this chapter are taken from the respective papers.

3.1. CVDD

CVDD consists of multiple components, each addressing a different aspect of clustering. Below, we break down all the formulas.

First, the **density separation** aspect is handled through the *mutual density factor* $fRel(x_i, x_j)$, which quantifies the likelihood of the points x_i and x_j being density-separated. This factor is calculated as:

$$fRel(x_i, x_j) = 1 - e^{-[Rel(x_i, x_j) + Rel(x_j, x_i) - 2]} \quad (3.1)$$

Here, the *relative density* $Rel(x_i, x_j)$ between the points is defined as the ratio of their densities:

$$Rel(x_i, x_j) = \frac{Den(x_i)}{Den(x_j)}, \quad (3.2)$$

with the density $Den(x_i)$ of a point x_i given by the average distance to its k -nearest neighbors:

$$Den(x_i) = \frac{1}{k} \sum_{x_j \in \mathcal{N}_i} d_E(x_i, x_j), \quad (3.3)$$

where $\mathcal{N}_i$ represents the set of k nearest neighbors of x_i .

Next, **compactness** measures how tightly points within a cluster are packed together. It is computed as the product of the standard deviation and the mean of pairwise distances within the cluster C_i :

$$com(C_i) = \frac{1}{|C_i|} \cdot \text{STD}(C_i) \cdot \text{Mean}(C_i) \quad (3.4)$$

3. Overview of Existing Density-Based CVIs

The mean distance between points in a cluster is given by:

$$\text{Mean}(C_i) = \frac{1}{|C_i| \cdot (|C_i| - 1)} \sum_{x_i, x_j \in C_i} pD(x_i, x_j) \quad (3.5)$$

and the standard deviation is:

$$\text{STD}(C_i) = \sqrt{\frac{1}{|C_i| \cdot (|C_i| - 1) - 1} \sum_{x_i, x_j \in C_i} (pD(x_i, x_j) - \text{Mean}(C_i))^2} \quad (3.6)$$

The *connectivity distance* quantifies the notion of closeness for this CVI. It is defined as the **minimax distance** (the largest edge weight along the shortest path) based on the *directly density-reachable distance* (drD) over all paths p between two points x_i and x_j :

$$\text{conD}(x_i, x_j) = \min_{p \in \mathcal{P}(x_i, x_j)} \max_{e \in p} w^{\text{drD}}(e) \quad (3.7)$$

Here, $\text{drD}(x_i, x_j)$ is a combination of the Euclidean distance $d_E(x_i, x_j)$ and the mutual density factor $f\text{Rel}(x_i, x_j)$:

$$\text{drD}(x_i, x_j) = d_E(x_i, x_j) + f\text{Rel}(x_i, x_j) \cdot nD(x_i, x_j), \quad (3.8)$$

where $nD(x_i, x_j)$ is the sum of the densities of the two points:

$$nD(x_i, x_j) = \text{Den}(x_i) + \text{Den}(x_j) \quad (3.9)$$

To finally determine the **separability** between clusters, we compute the minimum separation between points in different clusters. For a cluster C_i , the separability from other clusters is given by:

$$\text{sep}(C_i) = \min_{x_j \in \mathcal{C}, x_j \notin C_i} \text{sep}(C_i, C_j), \quad (3.10)$$

where $\mathcal{C} = \{C_1, C_2, \dots, C_K\}$ is the set of all clusters. The pairwise separability between two clusters C_i and C_j is defined as:

$$\text{sep}(C_i, C_j) = \min_{x_i \in C_i, x_j \in C_j} DD(x_i, x_j), \quad (3.11)$$

where the *density involved distance* $DD(x_i, x_j)$ is defined as:

$$DD(x_i, x_j) = \sqrt{f\text{Den}(x_i) \cdot f\text{Den}(x_j) \cdot \text{conD}(x_i, x_j)} \quad (3.12)$$

Noise is only handled implicitly through the *Outlier factor*, which compares the density $\text{Den}(x_i)$ of a point to the lowest density of a point in the dataset:

$$f\text{Den}(x_i) = \frac{\text{Den}(x_i)}{\max_{x_j \in X} \text{Den}(x_j)} \quad (3.13)$$

Here, X represents the entire dataset. Lastly, the parameter k (**minPts**) defines the number of nearest neighbors used to compute the density. The density of a point x_i is calculated as:

$$Den(x_i) = \frac{1}{k} \sum_{x_j \in \mathcal{N}_i} d(x_i, x_j) \quad (3.14)$$

where $\mathcal{N}_i$ are the k nearest neighbors of x_i . The choice of k can vary, with potential options being:

$$k = \sqrt{N} \quad \text{or} \quad k = \min(50, \frac{N}{2 \cdot k})$$

The overall score of a clustering is eventually calculated by:

$$CVDD(\Pi) = \frac{\sum_{i=1}^K sep(C_i)}{\sum_{i=1}^K com(C_i)}, \quad (3.15)$$

with $\Pi = \{C_1, C_2, \dots, C_K, C_{noise}\}$ as the clustering solution, where C_{noise} is the set of noise points.

3.2. DBCV

DBCV is another metric that evaluates clustering quality by considering both compactness and density-based separability. This section describes how each component of DBCV is calculated.

First, the *mutual reachability distance* is defined as:

$$d_{mreach}(x_i, x_j) = \max\{a_{ptscoredist}(x_i), a_{ptscoredist}(x_j), d_E(x_i, x_j)\}, \quad (3.16)$$

$$\text{where } a_{ptscoredist}(x) = \left(\frac{\sum_{i=2}^{n_c} \left(\frac{1}{KNN(x, i)} \right)^d}{n_c - 1} \right)^{-\frac{1}{d}}, \quad (3.17)$$

with $KNN(x, i)$ being the Euclidean distance from x to its i -th nearest neighbor.

Here, the *mutual reachability distance* between points x_i and x_j , captures the **density separation** between clusters. It takes into account both, the distance between the points and the density of the points, measured by the *all-points-core-distance*, which is computed on a cluster level.

Next, the **compactness** of a cluster is determined by the *Density Sparseness of a Cluster (DSC)*, which uses the concept of the **minimax distance**. It is computed as the maximum edge length of the MST built over the *mutual reachability distances* within a cluster:

$$DSC(C_i) = \max_{e \in P_i} w^{d_{mreach}}(e) \quad (3.18)$$

3. Overview of Existing Density-Based CVIs

with P_i as the set of all internal edges of the MST built from the points in C_i .

The *Density Separation of a Pair of Clusters (DSPC)* measures the **separability** between two clusters captured by the *all-points-core-distance* and is defined as the minimum *mutual reachability distance* between any internal nodes of the MSTs of the two clusters:

$$DSPC(C_i, C_j) = \min_{n_i \in C_i, n_j \in C_j} d_{mreach}(n_i, n_j), \quad (3.19)$$

where n_i and n_j are internal nodes of the MSTs of clusters C_i and C_j , respectively. An internal node is defined as a node of degree two or higher.

The *Validity Index of a Cluster* $V_c(C_i)$, is then computed using both density separation and compactness in a way reminiscent of the Silhouette Coefficient:

$$V_c(C_i) = \frac{\min_{1 \leq j < K, j \neq i} DSPC(C_i, C_j) - DSC(C_i)}{\max(\min_{1 \leq j < K, j \neq i} DSPC(C_i, C_j), DSC(C_i))} \quad (3.20)$$

The overall DBCV score is then computed as a weighted average:

$$DBC(\Pi) = \sum_{i=1}^K \frac{|C_i|}{|X|} V_c(C_i), \quad (3.21)$$

where K is the number of clusters, $|C_i|$ is the size of cluster C_i , and $|X|$ is the size of the dataset. Notice that DBCV handles **noise** implicitly, because $\sum_{i=1}^K |C_i| \leq |X|$.

3.3. DCSI

DCSI is the final reference CVI to be examined in this chapter. Below, we outline its key components.

To begin with, DCSI allows for the definition of different values ε_i for different clusters. The authors suggest the below formula:

$$\varepsilon_i = \text{median}_{x_j \in C_i} d_E(x_j, x_{(j, \text{minPts}, 2)}), \quad (3.22)$$

where $x_{(j,k)}$ denotes the k -th nearest neighbor of x_j within the cluster C_i . This approach allows DCSI to adapt to varying densities across different clusters.

Next, we evaluate the **compactness** of a cluster. Compactness in DCSI is measured by the *connectedness* of the core points within a cluster:

$$Conn_{DCSI}(C_i) = \max_{e \in P_i} w^{d_E}(e), \quad (3.23)$$

where P_i represents the set of all edges of the MST over Ω_i , the set of core points of cluster C_i . This ensures that the compactness is determined by the longest edge between core points, making it theoretically robust against outliers. Note that this again is the concept of the **minimax distance**.

3.3. DCSI

The **separability** between clusters is defined by the minimum distance between core points of different clusters:

$$Sep_{DCSI}(C_i, C_j) = \min_{x \in \Omega_i, x' \in \Omega_j} d_E(x, x'), \quad (3.24)$$

where Ω_i and Ω_j represent the sets of core points of clusters C_i and C_j , respectively. Notice how through the separate definition of ε_i for each cluster, **density separation** is taken into account.

The selection of **minPts**, which defines the minimum number of points required inside the ε_i – neighborhood to form a corepoint, is not strictly specified. The authors recommend using at least 5, but no clear strategy for its selection is provided.

DCSI relies on pairwise scores between clusters to compute the overall score.

$$DCSI(C_i, C_j) = \frac{q}{1 + q}, \quad \text{and} \quad q = \frac{Sep_{DCSI}(C_i, C_j)}{Conn_{DCSI}(C_i, C_j)} \quad (3.25)$$

Here, $Conn_{DCSI}(C_i, C_j)$ represents the weaker connected of the two clusters C_i and C_j :

$$Conn_{DCSI}(C_i, C_j) = \max\{Conn_{DCSI}(C_i), Conn_{DCSI}(C_j)\} \quad (3.26)$$

Finally, the overall DCSI score is computed as the average of the pairwise scores:

$$DCSI(\Pi) = \frac{2}{K \cdot (K - 1)} \cdot \sum_{i=1}^{K-1} \sum_{j=i+1}^K DCSI(C_i, C_j) \quad (3.27)$$

4. On Evaluating Density-Based Clustering Methods

Evaluating density-based clustering methods is a critical step in ensuring that algorithms perform well across a variety of data types and scenarios. In this chapter, we explore key concepts related to density-based clustering, discuss the new evaluation measure DISCO, and highlight the importance of selecting appropriate datasets for evaluation. By addressing these aspects, we provide a comprehensive understanding of how density-based clustering methods can be effectively assessed.

4.1. Density-Connectivity

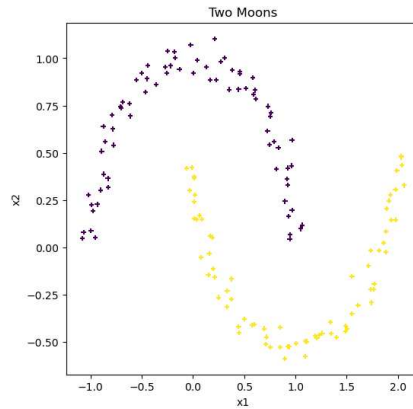


Figure 4.1.: Two Moons dataset with $n_samples = 150$ and $noise = 0.05$

Density-connectivity is a core concept in density-based clustering algorithms like DBSCAN [EKSX]. This notion is rooted in the idea that data points within a cluster should be close to each other, but in terms of density rather than (Euclidean) distance alone. We will introduce a few key concepts used by the authors of DBSCAN. The first concept is that of a point being *directly density-reachable*: a point x_i is *directly density-reachable* from a point x_j if x_j is a *core point* (see Section 2.1.2) and x_i lies within the ϵ -neighborhood of x_j . A broader concept is *density-reachable*: a point x_i is *density-reachable* from x_j if there exists a chain of points between them, where each point is successively *directly density-reachable*. Lastly, two points x_i and x_j are considered *density-connected* if there exists a third point o from which the other two points are *density-reachable* [EKSX].

4. On Evaluating Density-Based Clustering Methods

The recently developed density-connectivity distance (dc-distance) [BDH⁺23] uses these established concepts to capture density-connectivity. At first the *mutual-reachability distance* between all points is computed. It represents the ε , for which any two points are *directly density-reachable*. The definition is given by [BDH⁺23]:

$$d_r(x_i, x_j) = \max\{d_{\text{core}}(x_i), d_{\text{core}}(x_j), d_E(x_i, x_j)\} \quad (4.1)$$

Notice how we saw a similar formula in Section 3.1. This formula relies on the notion of a *core distance* denoted by $d_{\text{core}}(x)$, which is closely related to the concept of a *core point*. It is the distance to the minPts-th closest neighbor of a point. $d_E(x_i, x_j)$ refers to the Euclidean distance between two points.

The second and final step is to build a MST on the mutual reachability distances. This MST connects all data points without forming cycles, while minimizing the total edge weight. Within the MST, the dc-distance between any two points is defined as the largest edge weight along the path connecting them. The general definition of the dc-distance is provided below [BDH⁺23]:

$$d_{dc}(x_i, x_j) = \min_{p \in \mathcal{P}(x_i, x_j)} \max_{e \in p} w^{dr}(e), \quad (4.2)$$

The dc-distance between two points can then be interpreted as the smallest ε value for which these points are *density-connected*.

Figure 4.1 serves as an example dataset to demonstrate how differently the separation between these two clearly separated clusters is quantified for different distance measures.

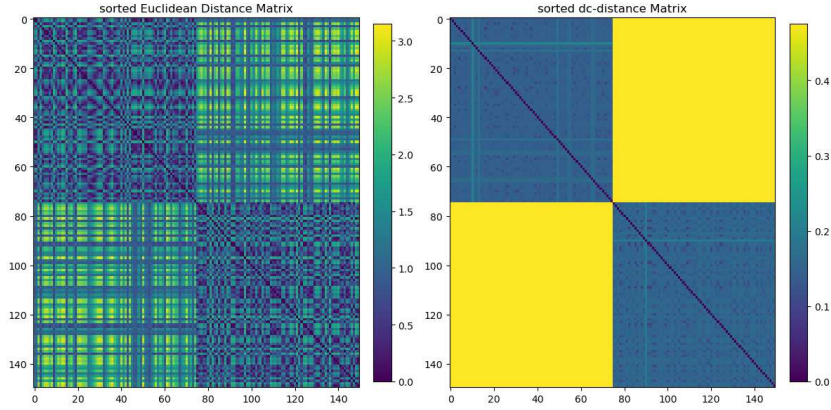


Figure 4.2.: Comparison of sorted Euclidean and dc-distance matrices for the Two Moons dataset: The left heatmap displays the sorted Euclidean distance matrix, while the right shows the sorted dc-distance matrix.

In Figure 4.2, we can see the Euclidean and the dc-distance matrix as heatmaps for the Two Moons dataset. The right heatmap, corresponding to the dc-distance matrix, shows a much clearer distinction between intra-cluster and inter-cluster distances, with strong block-like patterns that highlight well-separated clusters. In contrast, the Euclidean

distance matrix on the left exhibits more diffuse patterns, indicating weaker separation between clusters and a less pronounced difference between intra-cluster and inter-cluster distances.

4.2. A Novel Measure for Density-Based Clustering

Recently, a novel CVI called the Density-based Internal Evaluation Score for Clustering Outcomes (DISCO) [tKW⁺] was introduced. This new CVI combines the intuitive nature of the Silhouette Coefficient with the effective density capturing properties of the earlier discussed dc-distance. On top of that, DISCO also addresses the critical challenge of handling noise within a dataset.

One of the key aspects of DISCO is its ability to handle noise effectively. Unlike other CVIs that may implicitly handle noise or even ignore the presence of noise, DISCO integrates noise handling directly into its formulation. This allows for a more accurate reflection of the clustering structure in datasets where noise is inherent. It distinguishes between "good" and "bad" noise. "Good" noise is when the point is not density-connected to any cluster and when it is in a sparse region, while the opposite of these two characteristics define "bad" noise. For a "bad" noise point, two scores are computed: one quantifies the density of the area in which it lies compared to the density of the sparsest cluster, the other compares the smallest distance between the noise point and the nearest cluster to the largest core distance inside that cluster. The authors give a mathematical formulation of the noise score by the following equations [tKW⁺]:

$$\rho_{\text{noise}}(x_n) = \min(\rho_{\text{sparse}}(x_n), \rho_{\text{far}}(x_n)), \quad (4.3)$$

where

$$\rho_{\text{sparse}}(x_n) = \min_{C_i \in \mathcal{C}} \frac{d_{\text{core}}(x_n) - d_{\text{core}}(C_i)}{\max(d_{\text{core}}(x_n), d_{\text{core}}(C_i))}, \quad (4.4)$$

with

$$d_{\text{core}}(C) = \max_{x \in C} d_{\text{core}}(x) \quad (4.5)$$

and

$$\rho_{\text{far}}(x_n) = \min_{C_i \in \mathcal{C}} \frac{\min_{y \in C_i} d_{dc}(x_n, y) - d_{\text{core}}(C_i)}{\max(\min_{y \in C_i} d_{dc}(x_n, y), d_{\text{core}}(C_i))}. \quad (4.6)$$

The noise score $\rho_{\text{noise}}(x_n)$ lies between -1 and 1 and can directly be compared to the score of a none noise point, which is:

$$\rho_{\text{cluster}}(x) = \min_{C_i \neq C_x} \frac{\widetilde{d}_{dc}(x, C_i) - \widetilde{d}_{dc}(x, C_x)}{\max(\widetilde{d}_{dc}(x, C_i), \widetilde{d}_{dc}(x, C_x))}, \quad (4.7)$$

with

$$\widetilde{d}_{dc}(x, C_i) = \text{avg}_{y \in C_i} d_{dc}(x, y). \quad (4.8)$$

4. On Evaluating Density-Based Clustering Methods

The overall score for the clustering of a dataset is then simply computed as the average of those individual scores. It is given by:

$$\rho(\Pi) = \text{avg}_{x \in X} \rho(x) \quad (4.9)$$

Notice how DISCO uses the same formula as the Silhouette Coefficient, but with the dc-distance instead of the Euclidean distance. This ensures a DISCO score that lies between -1 and 1 for a full dataset.

DISCO also has its limitations however. Since it is tailored to density-based methods, it is sensitive to density-connected clusters. This implies that when clusters exhibit spherical shapes rather than arbitrary ones, it may result in lower values compared to the standard Silhouette Coefficient. We highlight this distinction because the apparent similarities between the two measures might lead to the mistaken belief that direct comparisons are valid. Although this makes sense, as it is constructed in that way, it limits its general usability. Overall, while DISCO builds on several well-established concepts, it introduces innovative approaches. Its competitiveness against existing density-based CVIs is later tested in Section 5.1.

4.3. A Tutorial for DISCO, a Novel Density-Based Evaluation Measure

To facilitate the understanding and application of DISCO, a comprehensive tutorial has been developed in the form of a Medium blog post.¹ This tutorial bridges the gap between the theoretical foundations of the measure and its practical implementation, making it accessible not only to researchers but also to data analysts and practitioners. The tutorial walks users through the mechanics of DISCO, demonstrating how it evaluates clustering outcomes and deals with noise. By providing comprehensible examples, the tutorial ensures that users with different levels of expertise can grasp the nuances of this novel evaluation measure. It is intended to serve as resource for those looking to adopt DISCO for validating density-based clusterings.

4.4. Data for Evaluating Density-Based Clustering Methods

When it comes to evaluating density-based clustering methods, the choice of datasets plays a crucial role. Synthetic as well as real-world datasets are required to properly evaluate a clustering method thoroughly. Unfortunately, many studies in this field tend to overlook the specific properties of real-world datasets, often assuming they are inherently suitable for density-based approaches, without properly checking. Another common issue is the general lack of distinction between clustering data and classification data. Properties that make data suitable for classification do not necessarily make them appropriate for (density-based) clustering [FGK⁺].

¹<https://medium.com/@marcus.artner/disco-internal-evaluation-of-density-based-clustering-d7f3988d27be>

4.4. Data for Evaluating Density-Based Clustering Methods

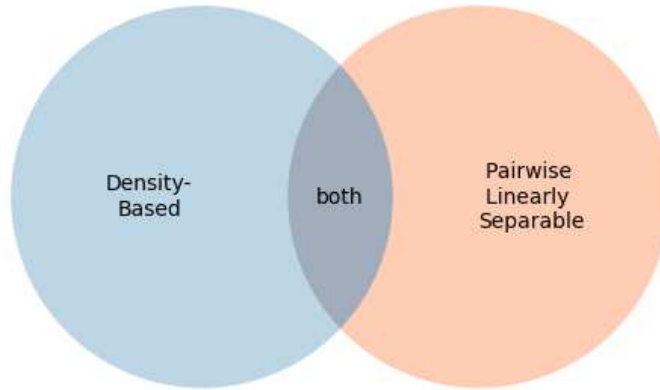


Figure 4.3.: Illustration of the expected properties of datasets used in the evaluation of density-based clustering methods: some datasets are density-based, some exhibit pairwise linear separability, and a few demonstrate both characteristics.

Moreover, many experimental studies focus on two- or three-dimensional datasets, which do not capture the complexity of real-world data that often exist in higher-dimensional spaces. This reliance on low-dimensional datasets can obscure the challenges and strengths of density-based clustering methods when applied to more complex, multi-dimensional data.

Addressing these gaps in dataset selection is essential for a more comprehensive evaluation of density-based clustering methods. In Section 5.2 we conduct an analysis of commonly used real-world datasets in research and explore their suitability for evaluating a density-based clustering method on them. We then identify a collection of datasets that, based on this analysis, are in fact "likely density based". A simplified illustration of the expected outcome of that analysis is given by Figure 4.3.

5. Experiments

This chapter focuses on the practical application of different CVIs. Since we have discussed the theoretical properties and workings of DCSI [GSH23], DBCV [MAJC⁺14], CVDD [HZ19] and DISCO [tKW⁺] earlier, we now evaluate their performances in various settings. Section 5.1 focuses on identifying potential weaknesses and strengths of the CVIs in question under controlled "lab conditions". In Section 5.2, the aforementioned CVIs, along with the widely known Dunn Index [Dun73] and Silhouette Coefficient [Rou87], are evaluated on real-world datasets commonly used in density-based clustering literature. The information gathered during the tests on synthetic datasets is then used to classify the real-world datasets into "likely density-based" and "likely non-density-based".

5.1. Synthetic Datasets

An effective way of testing a CVI is by using synthetic data, as this is one of the best ways one can be completely sure to know the ground truth to a clustering. Synthetic datasets allow the creation of extreme cases to observe how a CVI responds, offering a unique opportunity to evaluate its robustness. Additionally, synthetic data can be manipulated in a way such that only one parameter in the clustering is changed at a time, which gives complete control over the experiment and transparent insights into the strengths and weaknesses of a metric.

The two dimensional clusterings in Figure 5.3, Figure 5.5 and Figure 5.7 are generated using the `make_blobs` function from the `scikit-learn` library [PVG⁺11], with the parameter `n_samples = 500`. The parameters `centers` and `cluster_std` are changed during the experiments and the other parameters left at default. The clusterings in Figure 5.1 are created using a custom function, with 300 points per cluster. A step-by-step explanation of the function's process is provided in Figure 5.2. In contrast to the `make_blobs` function, which draws points from a Gaussian distribution, the points in these clusters are uniformly distributed to mitigate the curse of dimensionality [BCC57]. The clusterings in Figure 5.11 are created using the `densityDataGen` function from the `datagen` library [PJ]. Further details on the generation of the arbitrarily shaped clusters can be found in the appendix under A.3.

In several of the following figures, a dashed line marks the point where clusters overlap and become density-connected. In the figures, values to the left of the dashed line represent non-overlapping clusters, whereas values to the right illustrate when the clusters are in a state of contact and overlap. The raw data corresponding to all figures in this chapter can be found in the appendix. The settings used to create the datasets in Section 5.1.4 and Section 5.1.5 are also provided in the appendix. The `minPts` parameter of DCSI

5. Experiments

as well as the k parameter of CVDD were set to a value of 5 in all of the following evaluations. DCSI's ε parameter was computed as suggested by the authors of the paper [GSH23]. The implementations of DCSI and CVDD¹, as well as DBCV², used in this chapter are linked below. DBCV's *check_duplicates* parameter is set to **False** and the *use_original_mst_implementation* parameter is set to **True**, while all other parameters are left at their default values.

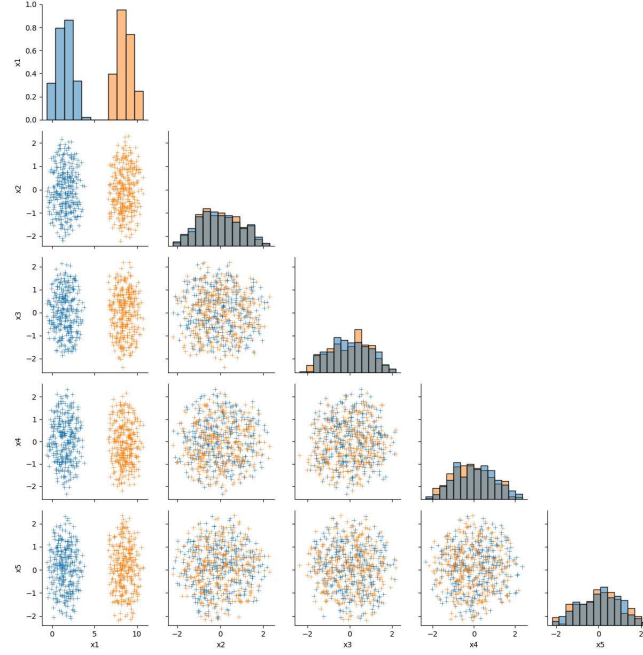
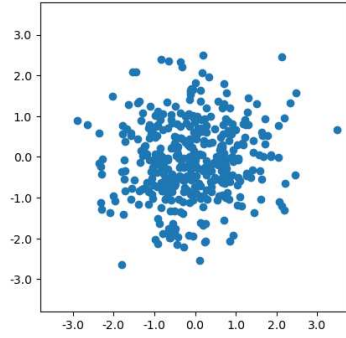


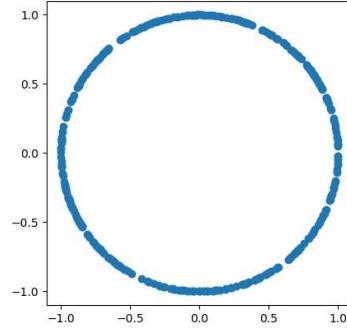
Figure 5.1.: Pairwise comparisons of dimensions of two 5-dimensional hypersphere clusters, where the difference between cluster centers exists only along the x_1 dimension. The scatter plots illustrate the relationships between different dimensions, while the diagonal histograms show the distribution of each dimension. Notice the clear separation in the x_1 dimension.

¹<https://github.com/marcusa98/Thesis>

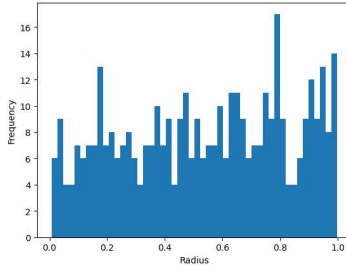
²<https://github.com/FelSiq/DBCv>



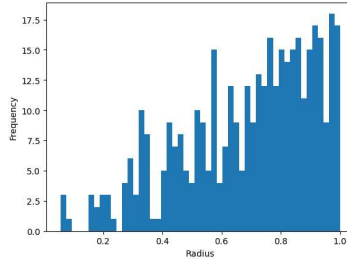
(a) Sampling n vectors from multivariate Gaussian



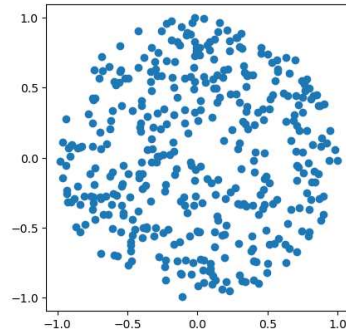
(b) Scaling vectors with their norm to a length of 1



(c) Sampling n radii uniformly between 0 and 1



(d) Exponentiate radii with $1/d$ to take dimensionality into account



(e) Multiply exponentiated radii by normalized points from (b) to obtain uniformly distributed points inside a (hyper)sphere

Figure 5.2.: Two dimensional toy example of the generation of a uniformly distributed (hyper)sphere cluster

5. Experiments

5.1.1. Experiment 1: Reducing the Distance Between Clusters

To assess the sensitivity of the reference CVIs, two blobs are generated and systematically moved closer together while maintaining a constant variance, as illustrated in Figure 5.3. The CVIs are evaluated in each step, to understand their behavior and results are depicted in Figure 5.4. This experiment is also extended to higher dimensions, in which both clusters shared the same coordinates in all but one dimension. In this one dimension, the cluster centers were shifted towards each other, until they overlapped fully again. A snapshot of this concept can be seen in Figure 5.1

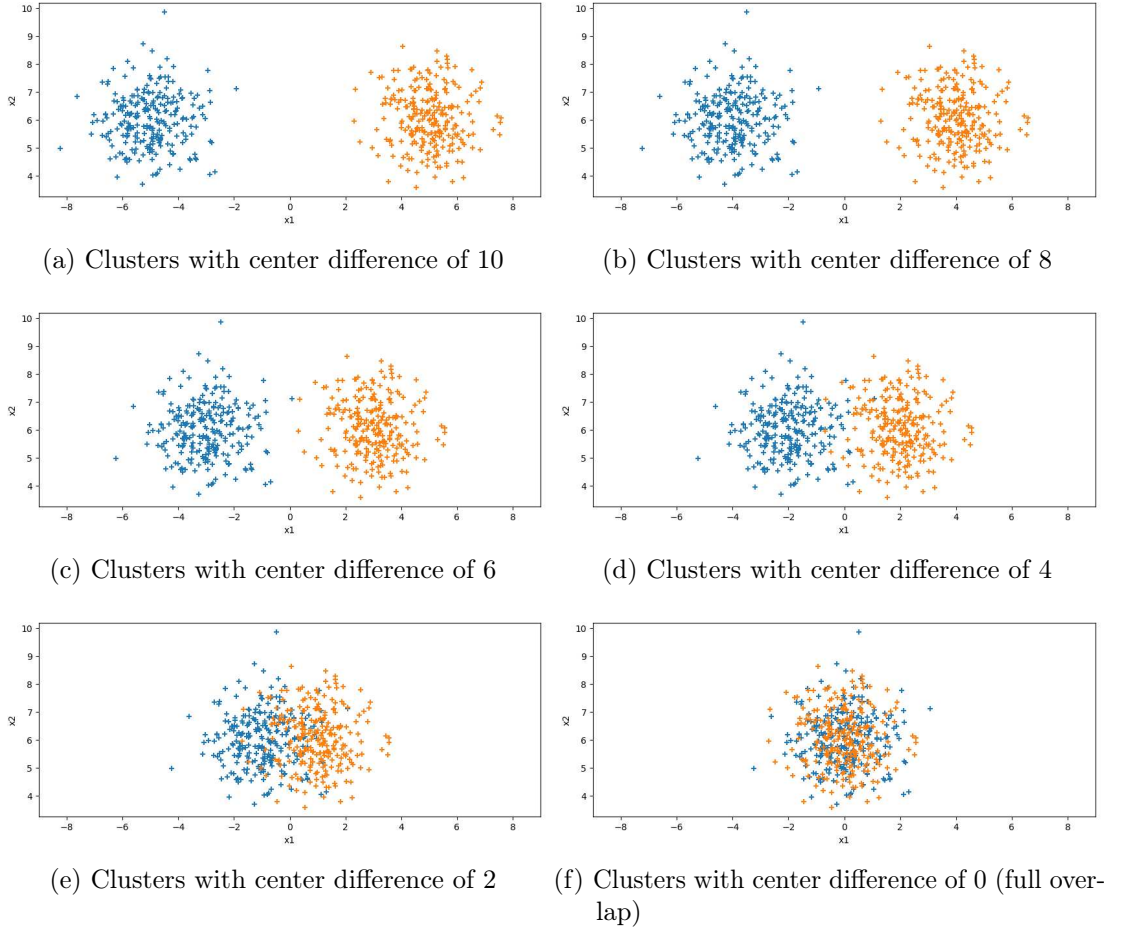


Figure 5.3.: 2-d clusterings with fixed variance and decreasing distance between centers

All CVIs achieve their highest values when the distance between cluster centers is largest across all dimensions. DCSI performs well in lower dimensions; however, for dimensions 10 and above, it yields relatively high values for overlapping (density-connected) clusters,

5.1. Synthetic Datasets

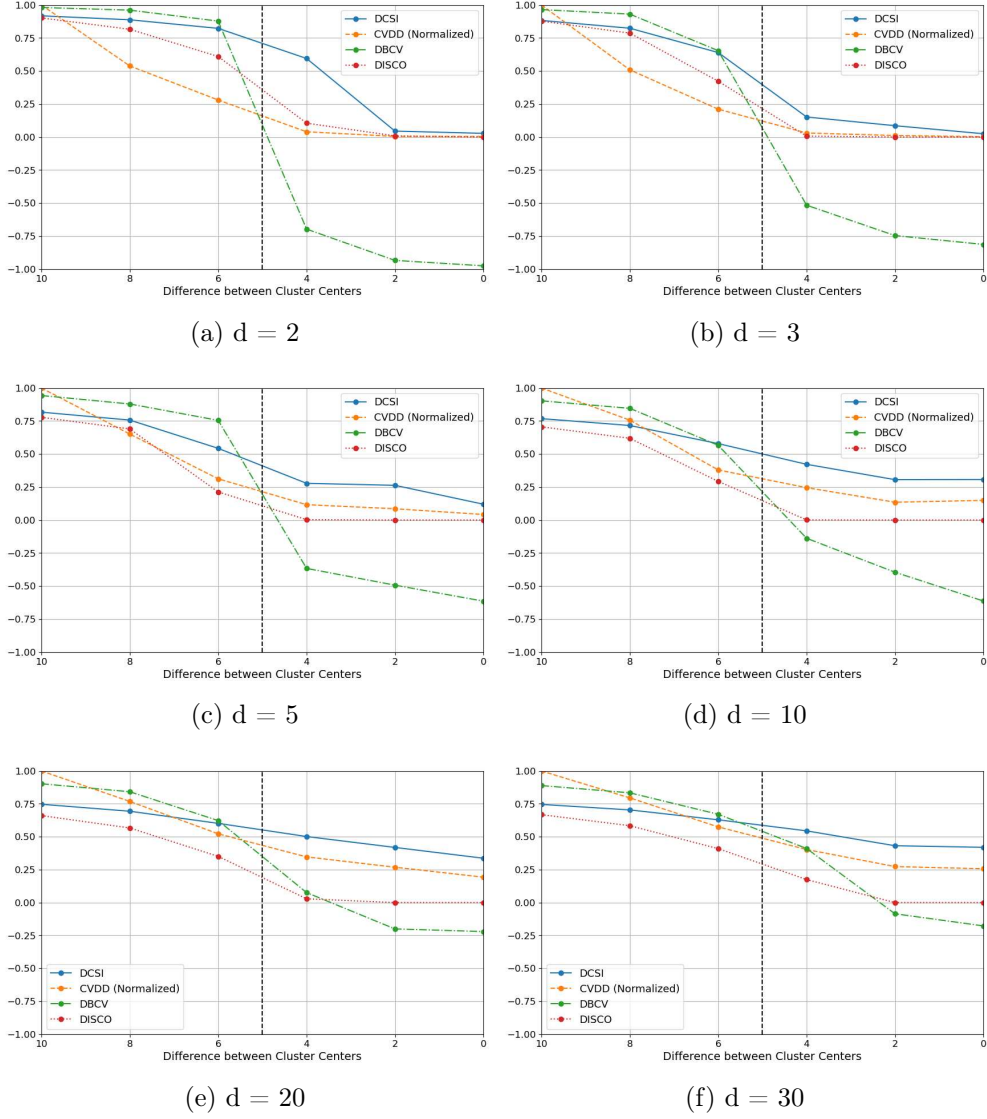


Figure 5.4.: Results for different CVIs on two clusters with decreasing distance between centers for different dimensionalities

indicating a decline in effectiveness. DBCV provides credible values overall. Notice a similar issue to DCSI: in lower dimensions, heavily overlapping clusters result in highly negative values, but these values increase as dimensionality rises. DISCO exhibits consistent behavior across all dimensions, effectively detecting density-connectivity and producing values within the expected range, making it the top performer among all CVIs in this case.

5. Experiments

5.1.2. Experiment 2: Increasing the Variance of Clusters

We are now interested in the behavior of CVIs, when the center is fixed, but the variance is slowly increased to the point that now individual cluster structure can be detected anymore. This experiment is extended to higher dimensions, with the modification that the clusters are uniformly distributed rather than following a Gaussian distribution to avoid overly sparse cluster borders. To achieve a similar scenario as in the 2d experiment, the radius of each hypersphere is increased, while the number of points per cluster and the difference in centers remain fixed at 300 and 6 respectively.

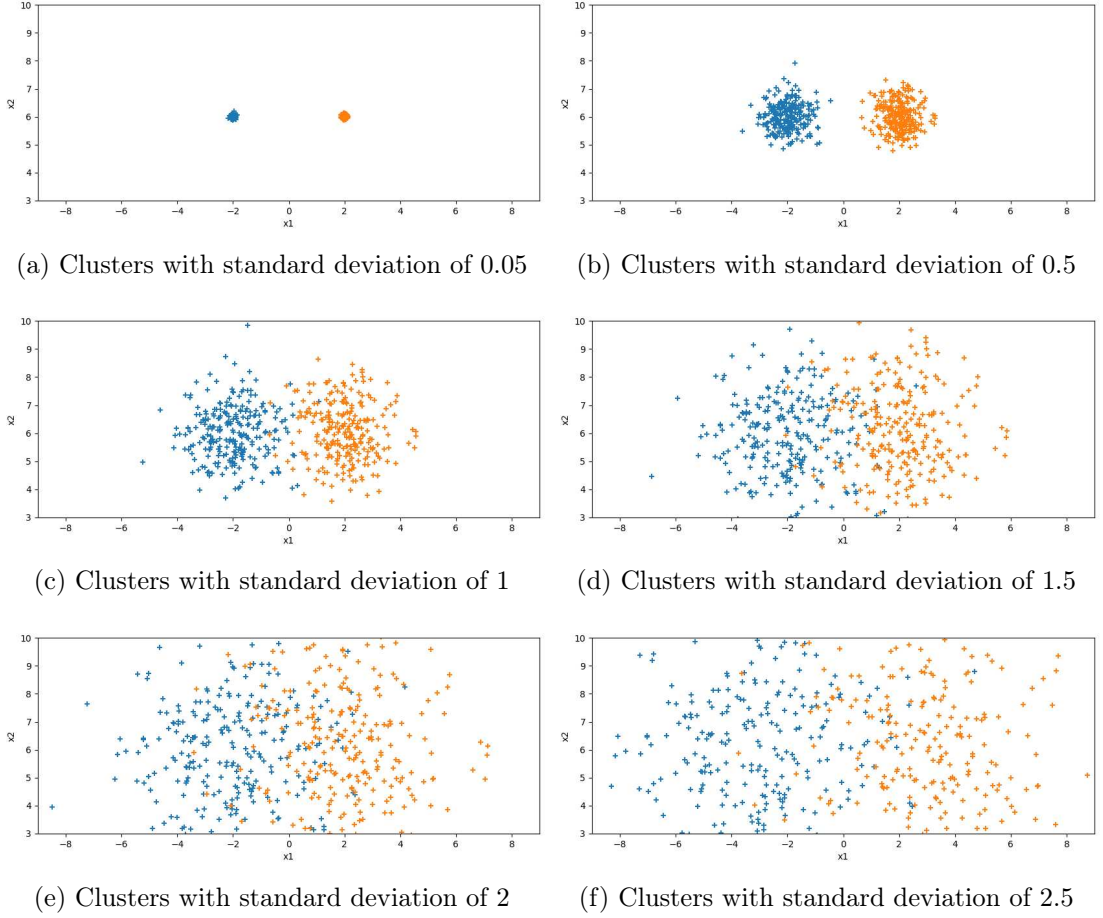


Figure 5.5.: 2-d clusterings with fixed centers and increasing variances

In the two-dimensional scenario, all CVIs start out with a value of approximately 1 when the standard deviation is 0.05. As the standard deviation increases, Normalized CVDD drops to nearly 0, even before the clusters begin to overlap. DCSI shows a slight

5.1. Synthetic Datasets

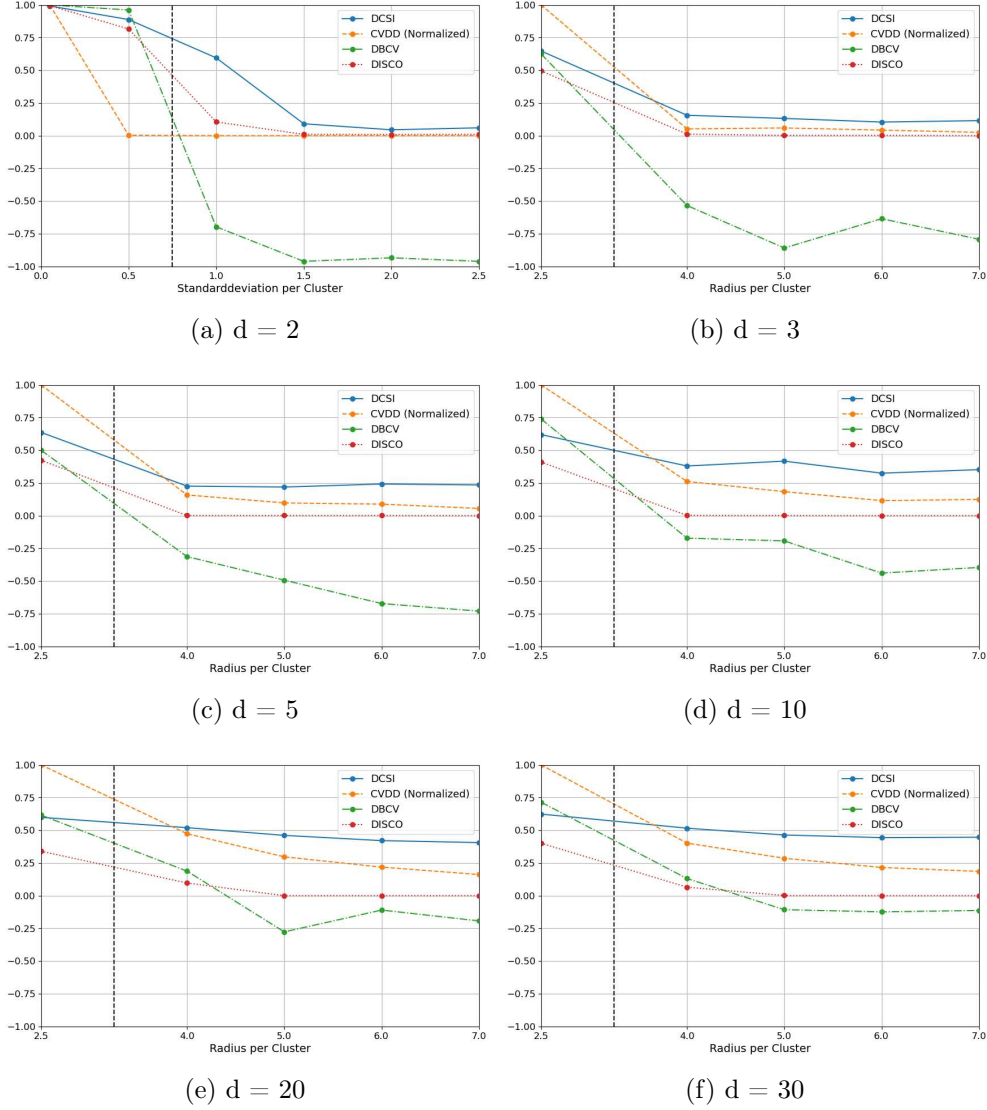


Figure 5.6.: Results for different CVIs on two clusters with fixed distance between centers and increasing variance for different dimensionalities

decline in value with a standard deviation of 1, whereas DBCV and DISCO demonstrate more pronounced sensitivity to the density-connectivity between the clusters. In the higher-dimensional scenarios, a similar trend as in Section 5.1.1 is observed across all CVIs. DISCO maintains consistent sensitivity to overlap between clusters across all dimensions, while DCSI, DBCV, and CVDD become less reliable as dimensionality increases, as their values no longer accurately reflect cluster overlap.

5. Experiments

5.1.3. Experiment 3: Incorrectly Clustered Points

Among the most common errors in clustering is when a point or a group of points gets assigned to the wrong cluster. This can always happen and it is important that a CVI can handle such a case. This experiment is simulating an extreme case of this problem, in which both clusters contain points of the other cluster relatively close to the respective center.

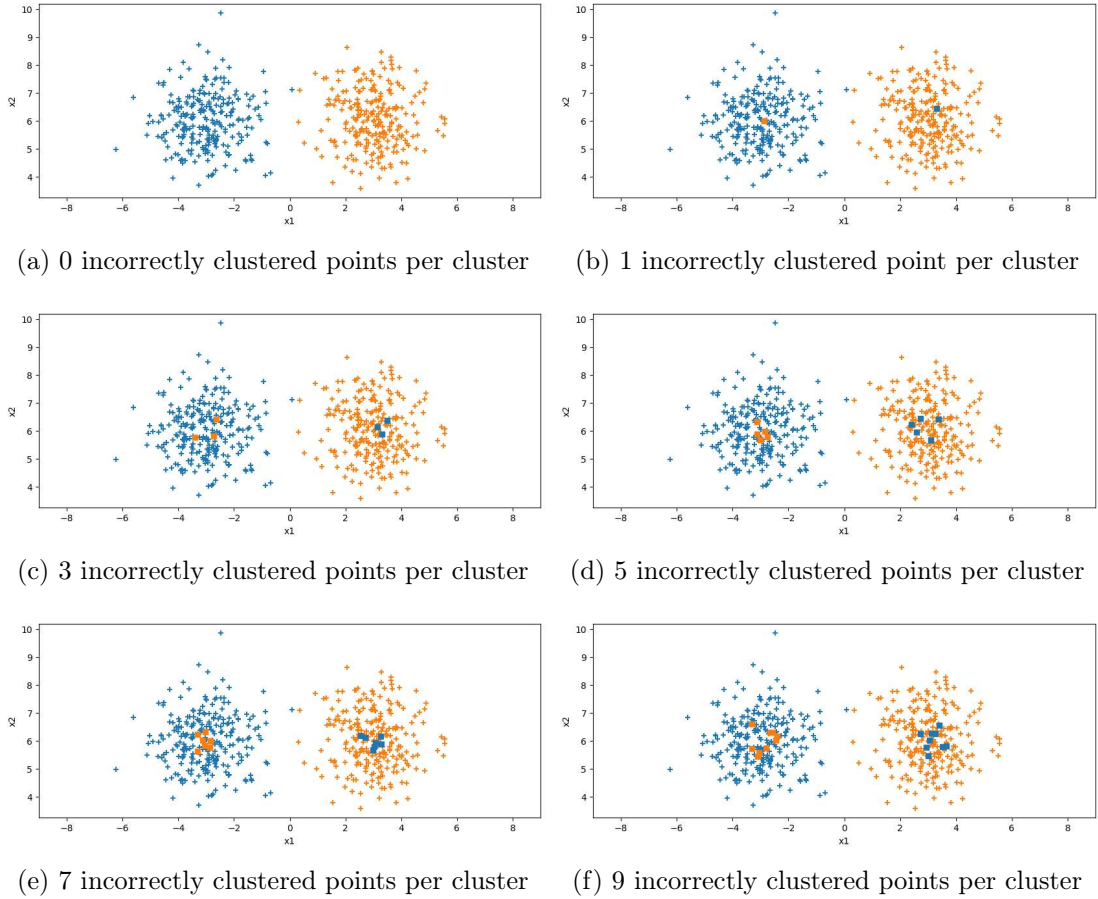


Figure 5.7.: 2-d clusterings with fixed centers and varying number of random incorrectly clustered points.

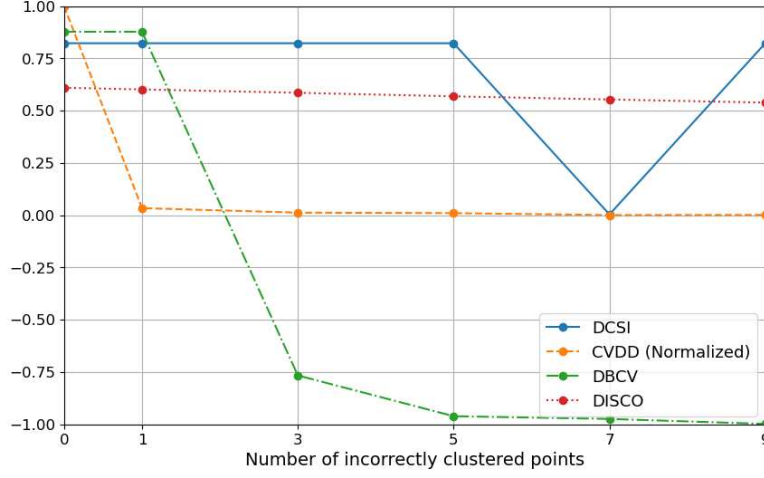


Figure 5.8.: Results of different CVIs on 2-d clusterings with fixed distance between centers and varying number of random incorrectly clustered points.

In Figure 5.8, we observe that the value of DCSI drops to near zero when there are exactly 7 incorrectly clustered points, while it performs well in other cases. This occurs because, in that specific scenario (see Figure 5.7e), the incorrectly clustered points form a core point. This implies two things:

1. DCSI disregards incorrectly clustered points unless they form a core point.
2. DCSI is extremely sensitive to "outlier core points".

Normalized CVDD drops to near zero with just one incorrectly clustered point per cluster, indicating a high sensitivity to outliers. DBCV reacts sensitively for more than one incorrectly clustered point, which is not surprising as it measures separability between clusters by taking the minimum distance between inner nodes of different clusters MSTs and for at least two incorrectly clustered points there exists an inner node inside the wrong cluster. In contrast, DISCO displays a gradual decline in value as the number of incorrectly clustered points increases, showcasing the most intuitive behavior among the four CVIs. These findings are expected to remain consistent across higher dimensions.

5. Experiments

5.1.4. Experiment 4: Increasing the Noise Ratio

When clustering real-world data, it is common to encounter data points that do not fit neatly into any cluster. These outliers are typically labeled as noise. Noise can significantly impact the performance of clustering evaluation metrics. As briefly addressed in Chapter 3, different CVIs handle noise in varying ways, leading to greater metric sensitivity in some cases than in others. Figure 5.9 illustrates a clustering scenario with two arbitrarily shaped clusters of varying densities, where the noise ratio in the dataset is incrementally increased. The `densityDataGen` function facilitates the extension of this and the subsequent experiment to higher dimensions, ensuring that these clusters retain their density-preserving characteristics as dimensionality increases [PJ].

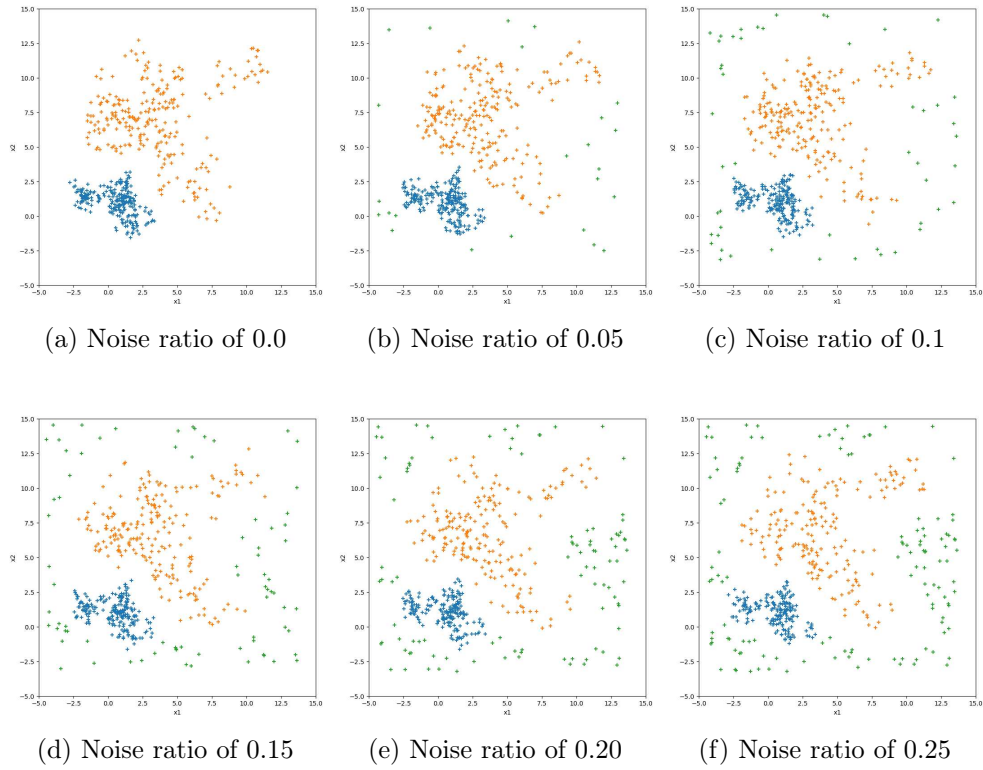


Figure 5.9.: Two 2-d clusters with different densities and increasing noise ratios

The dots at a noise ratio of 0 in Figure 5.10 can be interpreted as how well two clusters of arbitrary shape and different densities are handled by the CVIs. We observe that although the clusters are well separated as can be seen in 5.10a, the values of DCSI and DISCO are mostly in the 0.4 and 0.3 range respectively across all dimensions. These

5.1. Synthetic Datasets

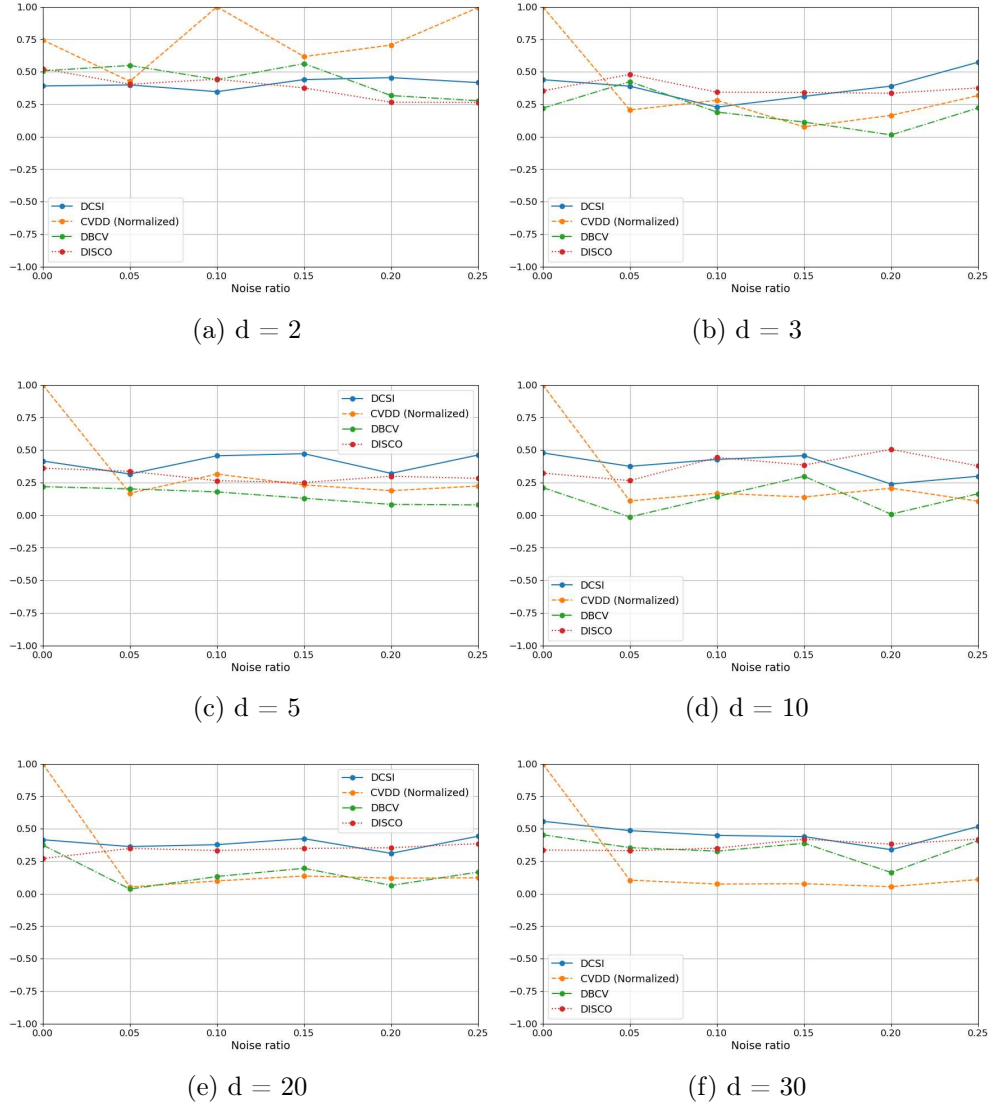


Figure 5.10.: Results for different CVIs on two clusters of different densities across different dimensionalities with increasing noise ratio.

values remain rather consistent for both CVIs for different noise ratios, which is desirable. Normalized CVDD displays sensitive behavior to the introduction of noise. The values remain steady between noise ratios of 0.05 and 0.25 however. DBCV exhibits more volatile behavior compared to the other DCSI and DISCO, likely due to its non-deterministic nature. While most values are positive, a few hover around 0.0.

5. Experiments

5.1.5. Experiment 5: Arbitrarily Shaped Clusters With Different Densities

The main purpose of density-based CVIs is to be able to assess the quality of a clustering, where clusters do not need to follow a spherical shape. A good CVI should be able to recognize non spherical shapes and also clusters of different densities. In Figure 5.11 we can see four different datasets with clusters of arbitrary shape and varying density. The number of clusters (classes) increases by one in each dataset. Figure 5.12 shows the results of different CVIs on these four datasets.

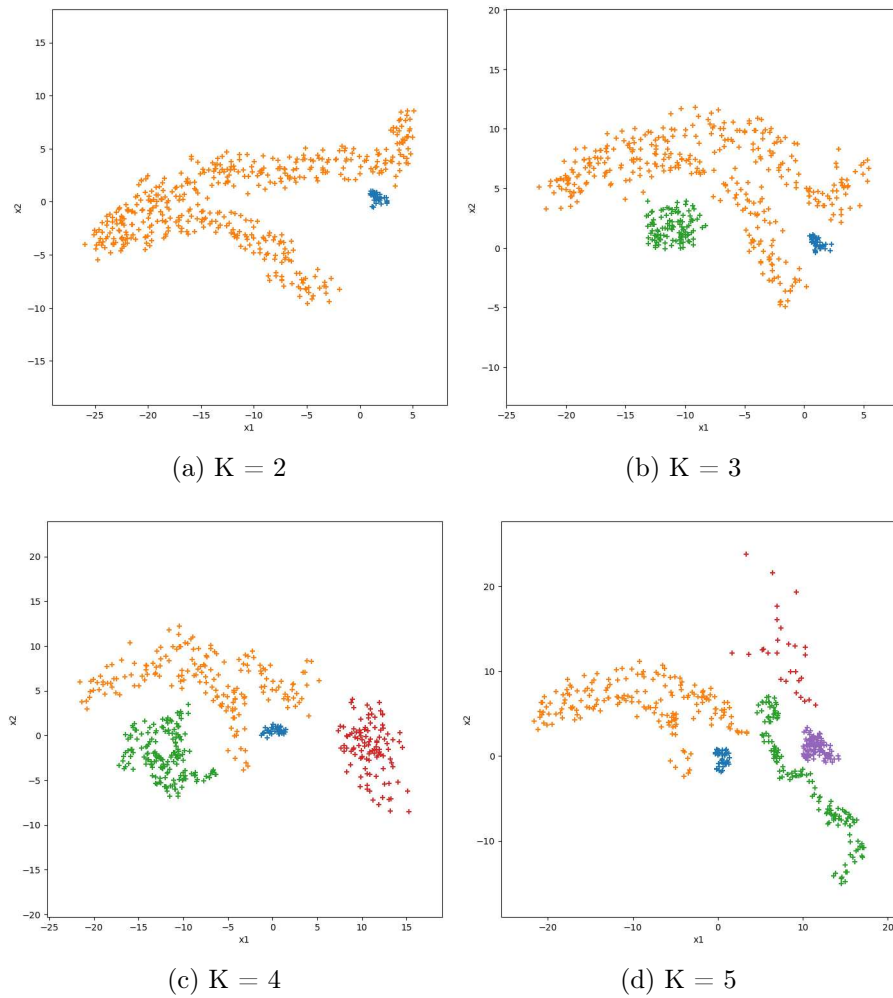


Figure 5.11.: 2-d clusters with arbitrary shapes and different densities

5.1. Synthetic Datasets

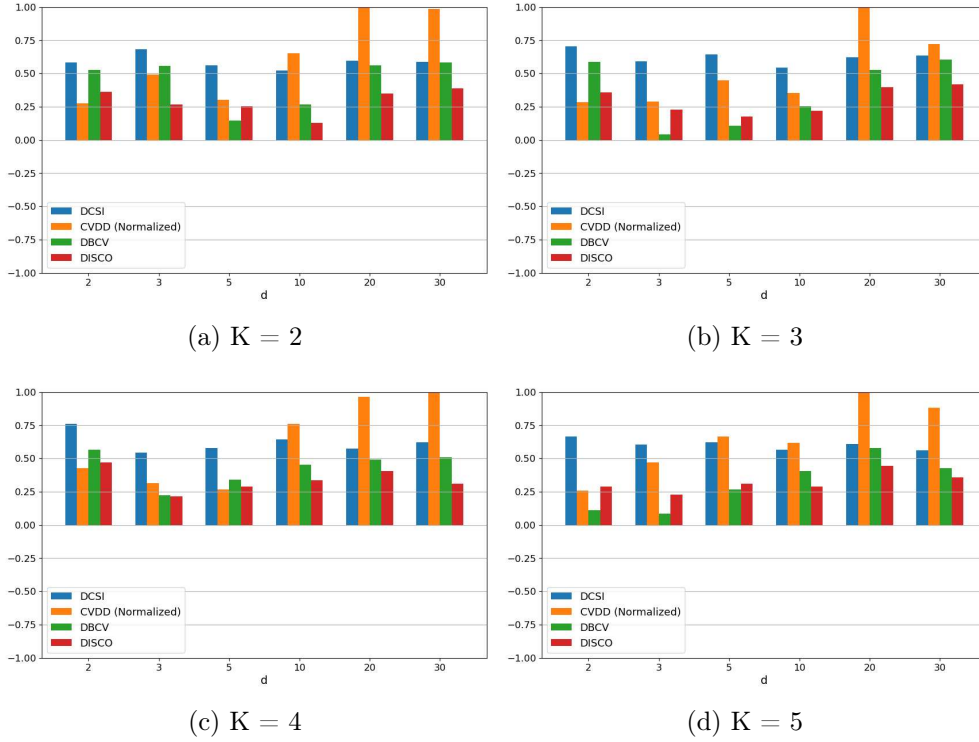


Figure 5.12.: Evaluation of different metrics on four different datasets with clusters of arbitrary shapes and different densities across different dimensionalities.

Values of DCSI and DISCO are consistent with increasing dimensions. Note that values are around 0.6 and 0.3 respectively. This gives an idea of what we can expect on real-world density-based datasets later in Section 5.2. Normalized CVDD increases in value for clusterings with two classes with increasing dimensionality. As more classes are introduced, the values seem to stabilize better for increasing dimensions, however, for clustering d, Normalized CVDD again does not stabilize with increasing dimension, making it hard to draw a general conclusion. DBCV values generally fall within a realistic range, but random outliers close to 0.0 appear for $d = 3$ in clusterings b and d. This again is likely due to DBCV's non-deterministic nature.

5. Experiments

5.2. Real-World Datasets

This section evaluates the density-based CVIs: DCSI, DBCV, CVDD and DISCO along with the widely known Dunn Index and Silhouette Coefficient on 28 datasets, commonly used by the research community in the field of density-based clustering, to identify appropriate real-world datasets that are density-based. Note that two datasets, Synth High and Synth Low were synthetically created and exhibit density-based structure by default. They serve as a reference point in this and the following section. Many of the datasets evaluated in this chapter were loaded via the `ClustPy` package [LMPB23], which sources data from multiple external repositories. For details on dataset origins, please refer to the documentation of the package. Results in Table 5.1 have been color-coded with green colored cells hinting at a density-based and red colored cells hinting at non density-based dataset for the density-based CVIs. For the Silhouette Coefficient and the Dunn Index green indicates pairwise linear separability.

Dataset	n	d	K	Source	DCSI	DBCV	CVDD	Dunn	Silh.	DISCO	Lin. Sep
Coil20	1440	16384	20	[LMPB23]	0.64	0.36	0.02	0.24	0.15	0.3	✗
Coil100	7200	49152	100	[LMPB23]	0.67	0.33	0.01	0.09	0.16	0.26	✗
HTRU2	17898	8	2	[LMPB23]	0.03	-1	0.36	0	0.26	0.34	✗
Ionosphere	351	34	2	[KLN]	0.17	-0.89	2.3	0.05	0.15	0.32	✗
Iris	150	4	3	[KLN]	0.67	-0.11	154.42	0.06	0.5	0.27	✓
Seeds	210	7	3	[KLN]	0.5	-0.75	41.74	0.05	0.41	0.05	✓
Synth High*	5000	100	10	[PJ]	0.88	0.86	2,334.89	0.03	0.42	0.73	✓
Synth Low*	5000	100	10	[PJ]	0.9	0.83	2,744.53	0.03	0.53	0.68	✓
WDBC	569	30	2	[KLN]	0.14	-0.93	0	0	0.51	0.33	✓
CMU Faces P.	624	960	20	[LMPB23]	0.55	-0.15	0.02	0.26	0.17	0.07	✗
CMU Faces P. + G.	624	960	24	[LMPB23]	0.54	-0.62	0.01	0.14	0.08	-0.01	✗
FMNIST	70000	784	10	[LMPB23]	0.37	-0.91	2.12	0.07	0.05	-0.07	✗
Glass	214	9	6	[LMPB23]	0.35	-0.91	0.3	0.02	-0.09	-0.12	✗
HAR	10299	561	6	[LMPB23]	0.5	-0.76	593.26	0.09	0.06	-0.01	✗
Keck Gestures	25457	120000	15	[LMPB23]	0.1	-1	0	0.01	-0.04	-0.02	✗
Keck People	25457	120000	4	[LMPB23]	0.21	-0.95	0.02	0.01	0.03	-0.02	✗
KMNIST	70000	784	10	[LMPB23]	0.33	-0.84	3.27	0.14	0.01	-0.05	✗
Letter rec.	20000	16	26	[LMPB23]	0.33	-0.85	43.98	0.03	0.01	-0.02	✗
Mice	1077	68	8	[LMPB23]	0.33	-0.84	18.33	0.05	-0.01	-0.02	✗
MNIST	70000	784	10	[LMPB23]	0.41	-0.75	5.93	0.15	0.04	-0.02	✗
Movement	360	91	15	[KLN]	0.43	-0.67	6.17	0.04	0.02	-0.04	✗
Optdigits	5620	64	10	[LMPB23]	0.5	-0.37	66.14	0.18	0.17	0.06	✗
Pendigits	10992	16	10	[LMPB23]	0.42	-0.77	2.47	0.04	0.18	0.13	✗
USPS	9298	256	10	[LMPB23]	0.43	-0.82	0.41	0.03	0.14	-0.01	✗
Weizmann Action	5701	77760	10	[LMPB23]	0.23	-0.92	0.03	0.09	-0.04	-0.05	✗
Weizmann People	5701	77760	9	[LMPB23]	0.33	-0.52	0.04	0.18	0	0	✗
Wine	178	13	3	[KLN]	0.29	-0.97	0.01	0	0.2	-0.01	✗
Yeast	1484	8	10	[KLN]	0.32	-0.96	19.05	0.01	0	-0.07	✗

Table 5.1.: Results of different CVIs on datasets commonly used as benchmarks in density-based clustering studies. *contains noise

All datasets are evaluated using the CVI parameters outlined at the beginning of this chapter, with a few exceptions. For the Yeast dataset, the *minPts* parameter of DCSI is adjusted to 4 due to the significant class imbalance, where one class has only 5 instances. Consequently, a point in this class can have at most 4 points within its core distance. For the Letter Recognition dataset, the parameter *k* of CVDD is increased to 30. This dataset contains many duplicate entries, and a smaller *k* value results in a

density of 0 for some points, leading to a division by zero error in the computation of relative density. DBCV’s `use_original_mst_implementation` parameter is set to `False` for the datasets Keck Gestures, Keck People, Weizmann Action and Weizmann People due to runtime reasons. In Section 5.2.1 the `umap-learn` library is utilized and only the parameter `n_components` is manually set when instantiating the embeddings.

In Table 5.1, column `n` and `d` reflect the dataset size and its dimensionality respectively. The column `K` represents the number of classes in the dataset. Datasets highlighted in bold are likely to exhibit a density-based structure, while the others do not. This determination is mostly based on the performance of DISCO, which is found to be the most reliable CVI in the previous section. DCSI sometimes yields misleading values for overlapping clusters, especially in dimensions higher than 10, which makes it only somewhat reliable for the five datasets: HTRU2, Iris, Seeds, Glass, and Yeast. The fact that DBCV is non-deterministic and experiences volatility, along with the fact that it yields extremely low values for most of these real-world datasets is the reason it is excluded from the decision. A dataset is classified as "likely density-based" if its DCSI value exceeds 0.4 and it has less than ten dimensions or if its DISCO value surpasses 0.3. These thresholds are established based on our findings in the previous section, where values above these thresholds are typically associated with well-clustered datasets. Additionally, if either the Dunn Index or Silhouette Coefficient exceeds 0.3, the dataset is considered pairwise linearly separable, as indicated by the column "Lin. Sep". Note that this does not guarantee pairwise linear separability between all clusters. Out of the 28 datasets evaluated, only 9 are classified as "likely density-based" and out of these 9, 5 also indicate pairwise linear separability, which implies that non density-based clustering methods likely achieve decent results as well. As observed in some previous experiments, there is a degree of disagreement between the CVIs in these results. Notable discrepancies between DCSI and DISCO include the CMU Faces P. and CMU Faces P. + G. datasets, where DCSI assigns values above 0.5, while DISCO remains below 0.1. Similar inconsistencies are seen for the HAR, MNIST, Movement, Optdigits, Seeds, and USPS datasets, where DCSI values exceed 0.4, but DISCO values fall significantly short of its threshold. In contrast, for the HTRU2, Ionosphere, and WDBC datasets, DISCO surpasses its threshold, while DCSI remains below. Mutual agreement between these two metrics is only found in the Coil20, Coil100, Synth High, and Synth Low datasets. Interestingly, the non-deterministic DBCV measure aligns with this mutual agreement.

Previously, researchers relied on datasets that were only assumed to be suitable for testing their methods. This analysis, particularly the identification of 9 (7 without the synthetic ones) datasets likely to have a density-based structure, offers future researchers in the field of density-based clustering new, more accurate benchmarks for testing their methods. Consequently, by using these selected datasets, future studies will yield more meaningful evaluations and comparisons, leading to better method development and validation in this domain.

5. Experiments

5.2.1. UMAP Projections for Density-Based Structure Evaluation

In this section, we apply the Uniform Manifold Approximation and Projection (UMAP) [MHM20] technique to datasets that either had a DISCO value of greater than 0.3 or a DCSI value of greater than 0.4. UMAP is a dimensionality reduction method that aims to preserve the local structure of high-dimensional data in a lower-dimensional space. By projecting these datasets into lower dimensional spaces, we evaluate whether the intrinsic density-based characteristics identified by the CVIs are maintained after dimensionality reduction. This analysis will help us understand the robustness of the global density-based structure when subjected to UMAP’s non-linear transformation. The datasets are embedded into two different spaces: once into a ten-dimensional space and once into a two-dimensional space. The results on these embeddings can be seen in Table 5.2 and Table 5.3 respectively.

Dataset	DCSI	Dunn	Silh.	DISCO	Lin. Sep.
Ionosphere	0.08	0	0.12	−0.01	✗
WDBC	0.08	0	0.36	0.4	✗
Movement	0.46	0	−0.06	−0.07	✗
MNIST	0.57	0	0.56	0.44	✓
Synth High	0.27	0	−0.04	−0.05	✗
Synth Low	0.48	0	0.11	0.08	✗
HAR	0.43	0	0.29	0.17	✗
Pendigits	0.44	0	0.6	0.69	✓
Coil20	0.85	0	0.6	0.55	✓
Coil100	0.91	0	0.52	0.48	✓
CMU Faces Person	0.72	0	0.42	0.36	✗
CMU Faces Person Glasses	0.81	0	0.05	0	✗
Optdigits	0.83	0	0.74	0.68	✓
USPS	0.89	0	0.62	0.55	✓

Table 5.2.: Experimental Results on 10-d UMAP embeddings of Datasets

In Table 5.2 we can see that DCSI yields mostly very similar values in the 10-d UMAP embedding compared to the original datasets. The datasets Synth High and Synth Low get lower scores. Interestingly DISCO yields higher scores for datasets such as MNIST, Coil100, CMU Faces Person, Optdigits and USPS, which were all below the threshold of 0.3 before, leading to new mutual agreements on density-basedness between DCSI and DISCO for the datasets: MNIST, Pendigits, Coil100, CMU Faces Person, Optdigits and USPS. The two datasets Synth High and Synth Low achieve really low scores, below the threshold of 0.3 as well. The Silhouette Coefficient yields higher scores for most datasets, indicating a better pairwise linear separability inside the embedding.

Dataset	DCSI	Dunn	Silh.	DISCO	Lin. Sep.
Iris	0.74	0.02	0.66	0.47	✓
Ionosphere	0.01	0	0.14	−0.01	✗
WDBC	0.04	0	0.37	0.44	✗
Movement	0.4	0	−0.09	−0.1	✗
Seeds	0.47	0	0.53	0.38	✓
MNIST	0.63	0	0.54	0.58	✓
Synth High	0.21	0	−0.22	0	✗
Synth Low	0.36	0	−0.03	0.03	✗
HAR	0.34	0	0.28	0.18	✗
HTRU2	0	0	0.16	0.08	✗
Pendigits	0.31	0	0.48	0.66	✓
Coil20	0.83	0	0.57	0.54	✓
Coil100	0.9	0	0.5	0.46	✓
CMU Faces Person	0.68	0	0.39	0.37	✗
CMU Faces Person Glasses	0.8	0	0.01	−0.02	✗
Optdigits	0.72	0	0.64	0.62	✓
USPS	0.93	0	0.61	0.57	✓

Table 5.3.: Experimental Results on 2-d UMAP embeddings of Datasets

Table 5.3 looks quite similar to Table 5.2. It is worth mentioning however that now both datasets Synth High and Synth Low have fallen far below the threshold of 0.3 for DISCO and have also gotten lower scores than in the 10-d embedding for DCSI. It serves as a reminder that UMAP does not guarantee to retain the global structure of datasets, when it fails to do so on datasets that were generated to be density-based. Similar to the results in Table 5.2, the Silhouette Coefficient yields relatively high scores for most datasets in the 2-D embedding, suggesting improved pairwise linear separability compared to the original data. This improvement is likely due to UMAP’s ability to preserve local structures—particularly in terms of k -neighborhood relationships—while simultaneously spreading out clusters.

5.3. Results

5.3.1. Analysis of Findings on Synthetic Datasets

The most critical property of a density-based CVI is its ability to reliably assess clusterings of arbitrary shapes and varying densities. In Section 5.1.5, we observe that DCSI and DISCO perform well across four different datasets, achieving consistent results across all dimensions. This consistency enables meaningful comparisons between clusterings on the original dataset and clusterings on the dataset with reduced dimensions. In contrast, DBCV displays high volatility, making it less reliable. CVDD is challenging to analyse without reference labels, but it also demonstrates inconsistency across different dimensionalities, which diminishes its effectiveness.

Another crucial attribute of a good CVI is its ability to handle noise. As shown in Figure 5.10, DCSI and DISCO once again deliver promising results. These two CVIs maintain consistent performance across various dimensions, even as the noise ratio increases from 0 up to 0.25. This finding is more impressive for DISCO, as DCSI does not take noise into account. In contrast, CVDD exhibits a sharp decline with the introduction of noise, but stays consistent for the datasets with noise. DBCV exhibits volatile behavior with some datasets yielding values at around 0, while others with a higher noise ratio achieve reasonably high values.

When evaluating scenarios with density-connected spherical clusters, DCSI provides decent values up to three dimensions, although it does not detect density-connectivity exceptionally well. Beyond three dimensions, however, its scores remain relatively high even when the clusters fully overlap, rendering it unreliable. DISCO, on the other hand, produces accurate values across all dimensions, with values dropping to near zero almost immediately when clusters begin to be density-connected, as illustrated in Figure 5.3d. DBCV provides trustworthy values for well-separated spheres. It reacts, similar to DISCO, rather sensitively to density-connected clusters. In higher dimensions, it yields higher values for density-connected clusters than in lower dimensions, exposing an inconsistency. CVDD also reacts sensitively to overlapping clusters in lower dimensions, but becomes less sensitive in higher ones, displaying a similar inconsistency as DBCV.

Finally, in the extreme case where points are incorrectly clustered near the center of another cluster, as tested in Section 5.1.3, DCSI does not react unless these points form a core point. This behavior is questionable, as it leaves significant room for error. DISCO, on the other hand, shows a gradual decline in value, which is the desired response. CVDD experiences a sharp drop when one incorrectly clustered point is introduced per cluster, but the score decreases more gradually as more points are introduced. DBCV behaves similar to CVDD, except it takes two incorrectly clustered points for it to start dropping sharply in value. Notice that for DBCV, Equation 3.19 implies that, although unlikely, for any dataset with n observations, it theoretically only takes two incorrectly clustered points near an inner node of the MST of another cluster, for every cluster to represent a bad separation for the whole dataset. The *minPts* parameter for DCSI can at least in theory be adjusted. It is difficult to say how CVDD behaves in a multicluster setting for this scenario.

In summary, while all CVIs demonstrated strengths and weaknesses in various scenarios, DISCO and DCSI emerged as the top performers overall. Between the two, DISCO detects density-connectivity more effectively, as evidenced by its behavior in scenarios with overlapping clusters. Although DCSI produces somewhat reliable results in lower-dimensional scenarios, high-dimensional data and incorrectly clustered points remain problematic in our experiments. DBCV performed well in some cases but proved to be too inconsistent, in part due to its non-deterministic nature.

5.3.2. Identification and Analysis of Density-Based Datasets

Determining which datasets are truly density-based can be approached in various ways. One method is to select datasets that exceed the previously established thresholds of 0.4 for DCSI or 0.3 for DISCO. Alternatively, we could restrict our selection to datasets that surpass both thresholds simultaneously or one could also try to take the number of classes and dimension into consideration. These are just a few of the many possible criteria for identifying density-based datasets.

In our analysis, we choose to define "likely density-based" datasets as the ones which surpass the DISCO threshold of 0.3 in Table 5.1. The goal of identifying these density-based datasets is to provide future researchers with a reliable set of benchmark datasets for the development of future algorithms. DISCO demonstrated "conservative" behavior in its outputs, never producing a "false positive" in our experiments according to our opinion, which gives us confidence in its accuracy for these datasets as well. In contrast, DCSI occasionally yielded "false positives". While this approach to identifying density-based datasets is not without its limitations, it serves as a starting point, and future researchers are encouraged to refine and improve upon the methodology presented here.

The table below presents the final datasets we view as "likely density-based":

Dataset	n	d	K	Lin. Sep
Coil20	1440	16384	20	✗
Coil100	7200	49152	100	✗
HTRU2	17898	8	2	✗
Ionosphere	351	34	2	✗
Iris	150	4	3	✓
Seeds	210	7	10	✓
WDBC	569	30	2	✓

Table 5.4.: "likely density-based" datasets

5.3.3. Robustness of Density-Based Structure in UMAP Projections

It is common practice to use dimensionality reduction methods on large datasets. However, the results of Section 5.2.1 show us that one needs to be careful, when using UMAP as a

5. Experiments

dimensionality reduction technique before clustering the dataset. Values in Table 5.2 and Table 5.3 reveal discrepancies when compared to the results in Table 5.1 (Note that for the datasets HTRU2, Iris and Seeds there only exists a 2-d UMAP embedding due to these datasets having less than 10 dimensions). While some values remain relatively consistent between the original and projected data, there are also notable differences. For instance, DISCO produces significantly higher scores for the MNIST, USPS, and Optdigits datasets, yet much lower scores for Synth High and Synth Low. Similar changes are observed for DCSI and the Silhouette Coefficient, with the latter showing improvements across many datasets.

An important thing to take into consideration when applying dimensionality reduction techniques in general is the intrinsic dimensionality of a dataset, which in simple terms is the number of required dimensions to retain most of the variance contained in the dataset [PJ]. If a datasets dimensions are reduced to a smaller number than its intrinsic dimensionality, the inherent structure is bound to change significantly, which is likely what happened with the Synth High and Synth Low datasets after the UMAP transformation.

The key takeaway here is that while UMAP embeddings can improve the clusterability of some datasets, they can also diminish it. UMAP inherently changes the properties of a dataset, and it is difficult to predict how these changes will impact clustering results. Users should be aware that applying a clustering algorithm to a UMAP embedding may yield different outcomes than clustering the original data. Therefore, whenever possible, clustering algorithms should be run on the original data to ensure reliable results. In research, it is especially important to avoid tuning CVIs and algorithms based on UMAP embeddings, as this may lead to skewed evaluations.

6. Conclusion

In summary, this work investigates four density-based CVIs. The analysis begins with a theoretical comparison of three of these CVIs, where each one is broken down into its fundamental components. The key properties of each measure are then embedded into a unified framework, providing a common terminology to facilitate clearer comparisons. These three CVIs are then tested alongside the newly developed DISCO measure in five experiments with varying parameters. These experiments consider different scenarios, which we require a reliable CVI to perform well on. The findings from these experiments are then used to identify a collection of "likely density-based" real-world datasets. To the best of our knowledge this is the first one of its kind. The collection is intended to help evaluate future density-based clustering methods, and thereby contribute to the advancement of the field. In addition to the dataset collection, we also contribute a comprehensive blog post that breaks down DISCO into its fundamental components, providing a clear and detailed explanation of the CVI. The blog post includes easy to understand examples using toy datasets, guiding readers through its capabilities. It serves as an accessible entry point for anyone looking to understand or apply the DISCO CVI, expanding the reach and impact of this research.

While this work gives good insights to the inner workings of the tested CVIs through experiments, there is more potential for further investigations:

- Most experiments are only conducted with two clusters. This simple setting helps understand the fundamental behavior of the CVIs, however it is of interest to extend this to a higher number of clusters, since that is more realistic in practice. A higher number of clusters can lead to many new scenarios not covered in this work. The distribution of scores for each CVI might also be affected.
- Investigate the distribution of scores further. Although the metrics DBCV and DISCO in theory both yield scores between -1 and 1, they have a noticeably different distribution. Potential future work could aim at adjusting the score distribution to a uniform distribution for easier interpretability.
- Adapt the collection of density-based real world datasets. This collection has been identified through the findings in our work, which are naturally limited. More advanced experiments can yield new insights and lead to a more sophisticated approach on choosing appropriate datasets.

Bibliography

- [BCC57] R. Bellman, Rand Corporation, and Karreman Mathematics Research Collection. *Dynamic Programming*. Rand Corporation research study. Princeton University Press, 1957.
- [BDH⁺23] Anna Beer, Andrew Draganov, Ellen Hohma, Philipp Jahn, Christian M.M. Frey, and Ira Assent. Connecting the dots – density-connectivity distance unifies dbscan, k-center and spectral clustering. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '23, page 80–92, New York, NY, USA, 2023. Association for Computing Machinery.
- [Dun73] J. C. Dunn. A fuzzy relative of the isodata process and its use in detecting compact well-separated clusters. *Journal of Cybernetics*, 3(3):32–57, 1973.
- [EK SX] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise.
- [FGK⁺] Ines Färber, Stephan Günnemann, Hans-Peter Kriegel, Peer Kröger, Emmanuel Müller, Erich Schubert, Thomas Seidl, and Arthur Zimek. On Using Class-Labels in Evaluation of Clusterings.
- [GSH23] Jana Gauss, Fabian Scheipl, and Moritz Herrmann. Dcsi – an improved measure of cluster separability based on separation and connectedness, 2023.
- [HC20] Morteza Haghir Chehreghani. Unsupervised representation learning with minimax distance measures. *Machine Learning*, 109(11):2063–2097, 2020.
- [HZ19] Lianyu Hu and Caiming Zhong. An internal validity index based on density-involved distance. *IEEE Access*, 7:40038 – 40051, 03 2019.
- [KLN] Markelle Kelly, Rachel Longjohn, and Kolby Nottingham. The uci machine learning repository.
- [LMPB23] Collin Leiber, Lukas Miklautz, Claudia Plant, and Christian Böhm. Benchmarking deep clustering algorithms with clustpy. In *2023 IEEE International Conference on Data Mining Workshops (ICDMW)*, pages 625–632. IEEE, 2023.
- [MAJC⁺14] Davoud Moulavi, Pablo Andretta Jaskowiak, Ricardo Campello, Arthur Zimek, and Joerg Sander. Density-based clustering validation. 04 2014.

Bibliography

- [MHM20] Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction, 2020.
- [PJ] Anna Beer Collin Leiber Thomas Seidl Philipp Jahn, Christian M. M. Frey. Data with density-based clusters: A generator for systematic evaluation of clustering algorithms. accepted at ECML PKDD 2024.
- [PVG⁺11] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.
- [Rou87] Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65, 1987.
- [tKW⁺] **Anna Beer***, Lena Krieger*, Pascal Weber*, Martin Ritzert, Ira Assent, and Claudia Plant. DISCO: Internal Evaluation of Density-Based Clustering. Under review, 2024.

A. Appendix

A.1. Simplified Overview

While each density-based CVI differs in its approach, they all fundamentally evaluate two key aspects of clustering: Separability between clusters and compactness within a cluster. Many of these indices utilize a concept known as the minimax distance, and understanding at which point this distance is applied can be insightful for interpreting their behavior. Additionally, it is crucial to assess how these CVIs account for varying densities and how (or if) they handle noise within the dataset. Furthermore, most CVI formulas can be categorized into families such as the Silhouette Coefficient or Dunn Index, based on their structural similarity. Table A.1 provides a simplified overview of these characteristics for the CVIs DCSI, DBCV and CVDD, highlighting their unique approaches to measuring cluster validity. Note that DCSI's family was assigned "Dunn", due to the way q is computed (see Section 3.3).

A. Appendix

CVI	Separability	Compactness	Minimax Distance	Different Densities	Noise	MinPts	Family
CVDD	$sep(C_i) = \min_{x_j \in C, x_j \notin C_i} sep(C_i, C_j)$	$com(C_i) = \frac{1}{ C_i } \cdot \text{STD}(C_i) \cdot \text{Mean}(C_i)$	$conD(x_i, x_j) = \max_{e \in P} u^{drD}(e)$ $\min_{p \in P} p(\alpha_i, \alpha_j)$	$fRel(x_i, x_j) = \frac{fRel(x_i, x_j)}{1 - e^{-[Rel(x_i, x_j) + Rel(x_i, x_j) - 2]}}$	$fDen(x_i) = \frac{fDen(x_i)}{\max_{x_j \in X} Den(x_j)}$	unclear	Dunn
DBCv	$DSPC(C_i, C_j) = \min_{n_i \in C_i, n_j \in C_j} d_{mreach}(n_i, n_j)$	$DSC(C_i) = \max_{e \in P_i} u^{d_{mreach}}(e)$	$DSC(C_i) = \max_{e \in P_i} u^{d_{mreach}}(e)$	$d_{mreach} = \max\{a_{pls} \text{coredist}(o_i), a_{pls} \text{coredist}(o_j), d(o_i, o_j)\}$	implicitly handled	not needed	Silhouette
DCSI	$SeppDCSI(C_i, C_j) = \min_{u_x \in \Omega_i, x' \in \Omega_j} d_E(x, x')$	$CommDCSI(C_i) = \max_{e \in P_i} u^{d_E}(e)$	$CommDCSI(C_i) = \max_{e \in P_i} u^{d_E}(e)$	cluster specific ε_i	-	unclear	Dunn

Table A.1.: Overview of considerations regarding some common parameters of existing CVIs

A.2. Raw Data

Difference	DCSI	CVDD	BCV	DISCO
d = 2				
10.0	0.918	838.443	0.981	0.902
8.0	0.888	451.189	0.961	0.816
6.0	0.823	235.092	0.878	0.61
4.0	0.595	33.369	-0.697	0.105
2.0	0.045	4.25	-0.934	0.009
0.0	0.028	4.135	-0.975	-0.001
d = 3				
10.0	0.883	12801.736	0.965	0.878
8.0	0.824	6516.022	0.931	0.787
6.0	0.64	2703.339	0.654	0.423
4.0	0.152	393.442	-0.516	0.007
2.0	0.086	149.464	-0.746	-0.0
0.0	0.025	34.874	-0.813	-0.0
d = 5				
10.0	0.817	7390.724	0.943	0.778
8.0	0.757	4812.269	0.879	0.69
6.0	0.543	2306.911	0.755	0.213
4.0	0.278	860.655	-0.366	0.003
2.0	0.263	631.964	-0.493	-0.0
0.0	0.12	312.219	-0.614	0.0
d = 10				
10.0	0.768	5480.806	0.903	0.706
8.0	0.716	4146.767	0.846	0.619
6.0	0.579	2079.962	0.566	0.293
4.0	0.422	1344.979	-0.138	0.001
2.0	0.306	740.121	-0.395	0.0
0.0	0.307	816.195	-0.613	-0.0
d = 20				
10.0	0.747	4831.409	0.903	0.661
8.0	0.694	3712.582	0.842	0.567
6.0	0.602	2524.666	0.623	0.351
4.0	0.502	1676.832	0.076	0.029
2.0	0.419	1298.813	-0.2	-0.0
0.0	0.337	933.31	-0.22	-0.0
d = 30				
10.0	0.746	4874.778	0.889	0.668
8.0	0.704	3878.046	0.834	0.584
6.0	0.629	2803.025	0.671	0.41
4.0	0.545	1965.763	0.411	0.175
2.0	0.432	1334.292	-0.085	0.0
0.0	0.42	1253.516	-0.177	-0.0

Table A.2.: Evaluation of different metrics on two clusters of different dimensionality with decreasing distance between centers

A. Appendix

Standarddev.	DCSI	CVDD	DBCV	DISCO
0.05	0.992	287744.064	1.0	0.994
0.5	0.888	902.379	0.961	0.816
1.0	0.595	33.369	-0.697	0.105
1.5	0.091	2.2	-0.961	0.01
2.0	0.045	2.125	-0.934	0.009
2.5	0.06	1.889	-0.961	0.01

Table A.3.: Results of different CVIs on 2-d clusterings with increasing variance and fixed distance between centers

Radius	DCSI	CVDD	DBCV	DISCO
d = 3				
2.5	0.648	2922.5	0.626	0.496
4.0	0.156	151.264	-0.534	0.012
5.0	0.132	172.67	-0.859	0.002
6.0	0.104	123.863	-0.635	0.002
7.0	0.115	75.073	-0.792	0.0
d = 5				
2.5	0.637	2692.938	0.501	0.424
4.0	0.226	426.985	-0.313	0.001
5.0	0.219	262.357	-0.493	0.001
6.0	0.242	237.529	-0.672	0.001
7.0	0.235	149.582	-0.73	0.0
d = 10				
2.5	0.62	2733.904	0.74	0.41
4.0	0.38	715.747	-0.172	0.002
5.0	0.417	502.658	-0.193	0.001
6.0	0.325	314.735	-0.439	0.0
7.0	0.352	339.279	-0.396	0.0
d = 20				
2.5	0.599	2444.147	0.616	0.34
4.0	0.52	1158.333	0.188	0.097
5.0	0.461	725.874	-0.277	0.0
6.0	0.421	533.81	-0.11	0.0
7.0	0.406	395.115	-0.193	0.0
d = 30				
2.5	0.624	2705.617	0.715	0.401
4.0	0.516	1087.513	0.131	0.065
5.0	0.464	774.162	-0.107	0.001
6.0	0.444	583.115	-0.124	0.0
7.0	0.447	502.7	-0.113	0.0

Table A.4.: Evaluation of different metrics on two clusters of different dimensionality with increasing variance

Num. Points	DCSI	CVDD	DBCV	DISCO
0.0	0.823	235.092	0.878	0.61
1.0	0.823	8.055	0.878	0.602
3.0	0.823	2.979	-0.766	0.586
5.0	0.823	2.432	-0.961	0.569
7.0	0.004	0.226	-0.974	0.554
9.0	0.823	0.436	-0.998	0.539

Table A.5.: Results of different CVIs on 2-d clusterings with fixed distance between centers and varying number of random incorrectly clustered points.

d	DCSI	CVDD	DBCV	DISCO
Clustering a				
2.0	0.583	334.851	0.529	0.361
3.0	0.684	595.304	0.558	0.269
5.0	0.562	364.208	0.146	0.255
10.0	0.524	791.843	0.266	0.127
20.0	0.595	1210.142	0.56	0.35
30.0	0.589	1192.124	0.583	0.389
Clustering b				
2.0	0.703	226.412	0.588	0.359
3.0	0.591	228.038	0.041	0.23
5.0	0.642	354.475	0.106	0.178
10.0	0.545	281.146	0.254	0.218
20.0	0.622	791.441	0.528	0.396
30.0	0.636	570.622	0.603	0.418
Clustering c				
2.0	0.759	105.085	0.567	0.472
3.0	0.545	78.096	0.226	0.216
5.0	0.579	65.933	0.339	0.287
10.0	0.645	188.06	0.453	0.336
20.0	0.575	237.924	0.494	0.405
30.0	0.623	246.879	0.508	0.31
Clustering d				
2.0	0.665	5.305	0.112	0.289
3.0	0.603	9.685	0.084	0.23
5.0	0.622	13.674	0.267	0.309
10.0	0.564	12.709	0.405	0.288
20.0	0.61	20.554	0.579	0.446
30.0	0.56	18.169	0.428	0.36

Table A.6.: Evaluation of different metrics on four different datasets with clusters of arbitrary shapes and different densities across different dimensionalities.

A. Appendix

Noise Ratio	DCSI	CVDD	DBCV	DISCO
d = 2				
0.0	0.391	89.073	0.507	0.524
0.05	0.399	51.003	0.549	0.404
0.1	0.347	119.599	0.44	0.443
0.15	0.44	73.8	0.563	0.376
0.2	0.455	84.447	0.317	0.266
0.25	0.417	118.96	0.277	0.264
d = 3				
0.0	0.439	288.745	0.22	0.353
0.05	0.389	59.618	0.424	0.48
0.1	0.228	81.012	0.19	0.343
0.15	0.311	22.04	0.113	0.341
0.2	0.39	47.584	0.014	0.335
0.25	0.574	91.582	0.222	0.376
d = 5				
0.0	0.416	416.18	0.219	0.361
0.05	0.316	70.19	0.202	0.337
0.1	0.456	131.373	0.179	0.265
0.15	0.472	96.708	0.13	0.251
0.2	0.321	78.423	0.083	0.299
0.25	0.462	93.193	0.079	0.283
d = 10				
0.0	0.478	710.628	0.212	0.323
0.05	0.375	77.618	-0.013	0.266
0.1	0.428	120.186	0.143	0.443
0.15	0.457	98.938	0.299	0.385
0.2	0.239	147.485	0.008	0.505
0.25	0.299	76.524	0.165	0.378
d = 20				
0.0	0.416	960.774	0.375	0.271
0.05	0.363	50.121	0.036	0.349
0.1	0.378	94.903	0.133	0.333
0.15	0.424	131.141	0.196	0.349
0.2	0.311	115.705	0.064	0.354
0.25	0.443	117.635	0.167	0.386
d = 30				
0.0	0.558	1376.601	0.454	0.337
0.05	0.486	144.17	0.355	0.332
0.1	0.449	102.855	0.328	0.351
0.15	0.44	106.003	0.389	0.42
0.2	0.339	75.418	0.164	0.382
0.25	0.518	150.43	0.41	0.42

Table A.7.: Evaluation of different metrics on two clusters of different densities across different dimensionalities with increasing noise ratio.

A.3. Function Parameters

The fixed parameters for the datasets created in Sections 5.1.4 and 5.1.5 are presented in the table below. For experiment 5, four datasets are created. For the dataset with two clusters in Figure 5.11a, the parameter *clunum* is set to 2 and the *core_num* parameter is [20,30] for the dataset with three clusters in Figure 5.11b the parameter *clunum* is 3 and *core_num* is [20,30,35] etc. The *momentum* parameter is modified in the same way as *core_num*.

Parameter	Experiment 4	Experiment 5
<i>clunum</i>	2	5
<i>seed</i>	12	7
<i>radius</i>	0.7	default
<i>clu_ratios</i>	[0.5,0.5]	default
<i>core_num</i>	[50,70]	[20,30,35,25,20]
<i>dens_factors</i>	True	[0.4, 2, 1, 2, 0.9]
<i>momentum</i>	[0,0]	[0.3,0.6,0.5,0.4,0.1]

Table A.8.: Parameters used for dataset creation in experiments 4 and 5