UNIVERSITY OF CAPE TOWN

DEPARTMENT OF COMPUTER SCIENCE

CS/IT Honours Final Paper 2020

Title: Using UMAP to reduce the dimensionality of molecular dynamics trajectories for optimal HDBSCAN clustering

Author: Wen Kang Lu

Project Abbreviation: ClusterMol

Supervisor(s): Michelle Kuttel

Category	*Min*	*Max*	Chosen
Requirement Analysis and Design	*0*	*20*	0
Theoretical Analysis	*0*	*25*	0
Experiment Design and Execution	*0*	*20*	20
System Development and Implementation	*0*	*20*	5
Results, Findings and Conclusions	*10*	*20*	20
Aim Formulation and Background Work	*10*	*15*	15
Quality of Paper Writing and Presentation	*10*		10
Quality of Deliverables	*10*		10
Overall General Project Evaluation (*this section allowed only with motivation letter from supervisor*)	*0*	*10*	0
Total marks		**80**	80

Using UMAP to reduce the dimensionality of molecular dynamics trajectories for optimal HDBSCAN clustering

Wen Kang Lu
Department of Computer Science
University of Cape Town
wklu99@gmail.com

ABSTRACT

Molecular dynamics plays an important role in determining the physical behaviour of molecules, and clustering is often used to extract the main molecular conformations from simulated trajectories. Here we make use of HDBSCAN to cluster the trajectories of meningococcal Y and W serogroups. With HDBSCAN known to have difficulty clustering high-dimensional data due to "the curse of dimensionality", we also make use of UMAP to preprocess the trajectories beforehand. The results for clustering with and without this preprocessing step were recorded and analysed. We find that while HDBSCAN alone can uncover the main conformational behaviours of meningococcal Y and W, it is unable to reveal each conformation's dominance due to too many frames being considered as noise. With the use of UMAP, HDBSCAN was able to cluster nearly 100% of the frames, extracting the conformational behaviours while accurately representing their relative dominance. Notably, the use of UMAP as a visualisation tool also enables the obtainment of rudimentary information without having to view the molecules themselves, which greatly aids cluster analysis.

CCS CONCEPTS

• **Theory of computation** → **Unsupervised learning and clustering**; • **Applied computing** → *Chemistry*.

KEYWORDS

Molecular dynamics, clustering, HDBSCAN, UMAP, curse of dimensionality, conformational behaviours

1 INTRODUCTION

Molecular dynamics (MD) simulates the behaviour of molecules which are allowed to interact with one another over a fixed period of time; it outputs what is known as the "trajectory" of the behaviour observed. This trajectory is a record of the movement of every atom during the simulation, with respective changes in position over time calculated by solving Newton's equations of motion [11]. Since the effects of molecular interactions occur so rapidly, positions are often updated at the femtosecond timescale [14]. Therefore, trajectories containing even just a few nanoseconds worth of time frames amount to millions of locational data to observe.

The financial assistance of the National Research Foundation (NRF) towards this study is hereby acknowledged. Opinions, findings and conclusions or recommendations expressed in any publication generated by the NRF supported study, is that of the author alone, and the NRF accepts no liability whatsoever in this regard.

MD simulations play an important role for researchers interested in the effects molecular conformations can have in their applications, some of which include drug-receptor interaction and protein folding [20]. Consequently, it is imperative that effective methods exist to gather insight from trajectories as efficiently as possible. A common way to do so is through the use of clustering algorithms that can partition the trajectory into a set of dominant conformations that are deemed structurally unique from one other. The main application for clustering in this paper regards the molecular conformations of polysaccharides. The importance of this is derived from studies suggesting that a polysaccharide's conformational behaviour correlates with its immunogenicity, and therefore efficacy in vaccines [9, 10]. Specifically, the flexible carbohydrate molecules of meningococcal Y and W serogroups are clustered.

Using a set of unlabeled data, clustering is a form of unsupervised learning whereby items deemed similar are grouped together, with items between groups being seen as dissimilar. Many popular algorithms are used for clustering in general, let alone just for molecular dynamics. Although definitions vary, these algorithms can be broadly classified into three categories: hierarchical, partitional, and overlapping clustering [8, 16].

The Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN) algorithm was introduced by Campello et al. [4] as an improvement to the DBSCAN algorithm [18]. As the naming implies, HDBSCAN makes use of a density-based hierarchy where the most dominant clusters are extracted, i.e, it clusters based on regions of high data density and considers areas that are too sparse as noise. The algorithm is relatively fast when compared to other clustering methods and has already seen experimentation in the field of biochemistry [14, 15]. Therefore, it has been chosen for our purposes of clustering MD trajectories.

One issue with HDBSCAN, however, is that it suffers from "the curse of dimensionality" [19]; a term coined by Bellman in 1957 [3]. The phenomenon is observed in the case where some high-dimensional data has an insufficient amount of recorded data points. When dimensionality becomes increasingly large, the amount of data required to form the same meaningful patterns and groups as lower dimensions increases at an exponential rate. This occurs because the feature space itself increases exponentially, so the probability of two data points being similar experiences exponential decay - the data simply becomes too sparse. Since each frame of an MD trajectory consists of a molecule's locational data in a three dimensional space, the number of dimensions for a given frame is 3N, where N is the number of atoms present in the molecule. Frames can often contain hundreds if not thousands of atoms, which gives rise to this phenomenon.

Uniform Manifold Approximation and Projection (UMAP) is a state-of-the-art dimensionality reduction technique developed by McInnes et al. in 2018 [13]. UMAP looks to estimate the underlying topology of some given data and project it to a manifold at some lower dimension, i.e., it aims to preserve as much of the variance explained in the original data while reducing its dimensional complexity. Recent experimentation with UMAP has shown that it is able to improve the clustering performance of algorithms that suffer from high dimensionality [1].

This paper explores HDBSCAN's performance in finding the dominant conformation(s) of meningococcal Y and W, and whether UMAP is able to improve results by using it first as a preprocessing step. Parametric algorithms have been seen to output very different results depending on their given values [20], so there is additional focus on the parameters for both HDBSCAN and UMAP.

2 RELATED WORK

Melvin et al. made use of HDBSCAN and Intelligent Minkowski-Weighted K-Means (iMWK-Means) to discern stability of MD trajectories and detect conformational changes [14]. The focus was to cluster conformations in a non-parametric fashion. To achieve this, the parameters of HDBSCAN and iMWKK-Means were left constant. Notably, the notion of using a dimensionality reduction technique was forgone as the clustering algorithms were considered efficient enough. The authors found that HDBSCAN was good for detecting large-scale structural changes while iMWK-Means was better suited for finer differences in stable systems. The study suggested that the combination of the two algorithms may prove fruitful for even better results.

A comparative study utilising UMAP to preprocess data before clustering was performed by Allaoui et al. [2]. The difference in clustering performance between using and not using UMAP with either K-Means, Agglomerative Hierarchical, HDBSCAN, or Gaussian Mixture Model (GMM) showed that in all cases, the use of dimensionality reduction improved clustering accuracy. The improvement was associated with the decrease in dimensional complexity after using UMAP.

3 METHODOLOGY

The (general) steps for clustering MD trajectories with HDBSCAN and testing the effects of UMAP are as follows:

(1) Select the atoms of interest to cluster on if they are only a subset of the entire structure. Not only does doing so reduce the noise generated by uninteresting atoms, the reduction in coordinates reduces the number of dimensions for each frame. This helps improve the efficiency of our algorithms. Let this number of atoms be N.
(2) Use UMAP to perform dimensionality reduction on the trajectory. The dimensions of the data, initially being 3N (x, y, and z coordinate of each atom), are reduced to some lower value.
(3) Cluster the preprocessed trajectories with HDBSCAN, generate relevant output, and create .pdb files of the dominant conformations.
(4) Repeat steps (1) and (3) to get the results of using HDBSCAN without dimensionality reduction.
(5) Analyse and compare the HDBSCAN results when used with and without UMAP.

To assist in running consecutive automated clustering jobs of various input data, a clustering framework was developed to satisfy the need for such a streamlined approach. Python was selected as the programming language as it has access to many data science and machine learning libraries, including the algorithms needed.

3.1 Framework

The framework, colloquially named "ClusterMol" during development, makes use of standard object-oriented design practices. A parser takes in user-inputs and calls the appropriate methods to perform processing and/or clustering jobs. ClusterMol implements 7 algorithms along with a diverse collection of testing data consisting of real-world and artificially generated datasets. Each clustering job can produce various outputs such as resulting cluster labels, visualisations of clusters when dimensionality reduction is used, cluster validity indices, pdb files in the case of MD, etc. Figure 1 illustrates the pipeline of a typical clustering job using ClusterMol.

Since the configuration of each job is entirely dependant on the user's input, doing so can be very time consuming. With this in mind, the framework also allows one to run jobs by creating and inputting the location of a configuration file. Doing so allows for not only easy repetition and adjusting of jobs, but more importantly the automation of them. Each section in a configuration file represents a job with the required parameters from the initial preprocessing (if used) to production of output. Multiple sections can then be written to one file so that ClusterMol is able to read and perform many jobs in succession without user interference.

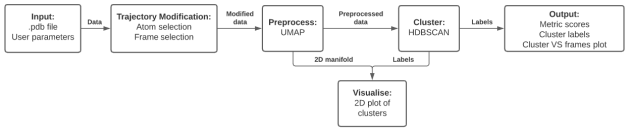

Figure 1: Clustering pipeline using ClusterMol framework.

3.2 HDBSCAN

The HDBSCAN implementation used in ClusterMol was attained from the Python package of the same name based on the works of Campello et al. [4, 12]. The algorithm has two main parameters that heavily influence the output of results. The first is min_cluster_size which changes the size grouping of what we consider to be the minimum for a cluster. The higher the value, the more data points are required in a region for it to be considered a cluster, resulting in fewer of them overall.

The second parameter is min_samples which deals with how HDBSCAN handles noisier data points. Higher values make the algorithm more conservative while clustering, treating more erratic data points as noise. Lower values make the algorithm more opportunistic as it tries to fit noisier data into their own clusters or in an existing one.

3.3 UMAP

The UMAP algorithm, also available as a Python package [13], has two parameters that affect its output in a substantial way. The first is n_components. This parameter is the number of dimensions that UMAP is reducing the data to. While this value requires experimentation for the purposes of clustering, n_components can be set to 2 for the purposes of visualisation. By reducing the dimensionality of a dataset to 2, we are able to plot each feature on a 2D plane. The labels resulting from clustering can then be used to differentiate the data points by colour, as seen in Figure 2.

The second parameter is n_neighbours. This parameter tells the algorithm how to balance its focus between the local and global structure of the presented data. A higher value tells UMAP to focus more on global structure so that groups of data points are placed in such a way that the distance between the groups tell us how different they are, i.e., a group placed far away from another is considered "more different" than if it were placed closer. This feature comes at the detriment of individual data points within the same group not necessarily being closest to other points most similar to them. By focusing on local structure, with a lower n_neighbours value, UMAP ensures that data points within groups that are most similar are close together on the embedding. This, naturally, comes with the trade-off of resulting groups not always being placed in meaningful ways.

3.4 Validation Data

Before clustering MD trajectories, we first clustered test datasets popular in the literature as part of preliminary exploration and validation. The UCI machine learning repository is a freely available online collection of databases that contain many real-life datasets used specifically for the testing of machine learning algorithms [7]. Breast Cancer, Digits, Iris, and Wine were the four datasets chosen, and each dataset has a known number of classes. A summary of them can be found in Table 1.

The four datasets cover a variety of different possible clustering behaviours. This ensures we gain broader insight as to how our techniques approach different kinds of problems. This is especially beneficial with regards to MD, as conformations can often have wildly varied shapes and sizes.

Breast Cancer contains 569 instances of 30 features derived from digitized images of fine needle aspirates of breast mass. There are two classes that can result from these attributes: malignant and benign. The dataset has been chosen for its relatively high feature count.

Digits is a 64-dimensional dataset derived from 8x8 bitmaps of handwritten images. The reduced dataset contains 1797 instances and 10 classes, where each class is a digit from 0 to 9. Digits is being used due to its higher dimensionality and class count.

Iris is a dataset containing three classes of iris flowers: Iris Setosa, Iris Versicolour, and Iris Virginica with 50 instances each. Each instance contains four features. Iris has been selected due to overlap between Iris Versicolour and Iris Virginica as both species have similar attributes.

Wine contains 179 instances of wine cultivated in Italy that belong to one of three types. Each wine has 13 features. The dataset has been chosen due to its well structured classes, so there should be little trouble in differentiating them.

Table 1: Summary of test datasets

Dataset	Instances	Features	Classes
Breast Cancer	569	30	2
Digits	1797	64	10
Iris	150	4	3
Wine	178	13	3

3.5 Molecular Dynamics Data

The MD data used for clustering were the pre-aligned trajectories of meningococcal Y (MenY) and W (MenW) capsular polysaccharides (CPS). These carbohydrate molecules are considered flexible and have varying conformations. The two serogroups along with others are used in polysaccharide vaccines to provide protection against Neisseria meningitidis based diseases. A summary of the data can be found in Table 2. These trajectories were treated as "unseen" data with no known number of classes/conformations for the purposes of experimenting with the different parameters of HDBSCAN and UMAP.

Both MenY and MenW trajectories originally contained over 40000 frames but were downscaled to new trajectory files retaining every tenth frame, thus arriving at 4003 frames for each serogroup. This was done for the purpose of expediting clustering and processing jobs while keeping enough meaningful data. 351 atoms were present in each trajectory, of which 56 were selected to cluster. This selection was made so that the atoms we were not interested in did not contribute as noise in the data, i.e., we would not want the same conformation of atoms in our scope of interest to be split into multiple clusters due to the conformations of atoms out of scope.

Table 2: Summary of meningococcus trajectories

Serogroup	Frames	Atoms	Atoms Selected	Features	Average RMSD (Å)
Y	4003	351	56	168	2.18
W	4003	351	56	168	3.29

3.6 Evaluation

The Silhouette score (S) [17] and Davies-Bouldin Index (DB) [6] have been chosen as internal metrics to gauge how well our implementations partition the presented data. Both metrics look to examine the ratio between cohesion within clusters and separation between clusters. Values close to 1 for the Silhouette score indicate good clustering. Scores near 0 indicate overlapping clusters, while a score close to -1 suggests incorrect clustering. For the Davies-Bouldin Index, values closer to 0 indicate good clustering, and values above 1 are considered very poor. Since these metrics require no knowledge of the data's actual classes, they are usable for both the test datasets as well as MD data.

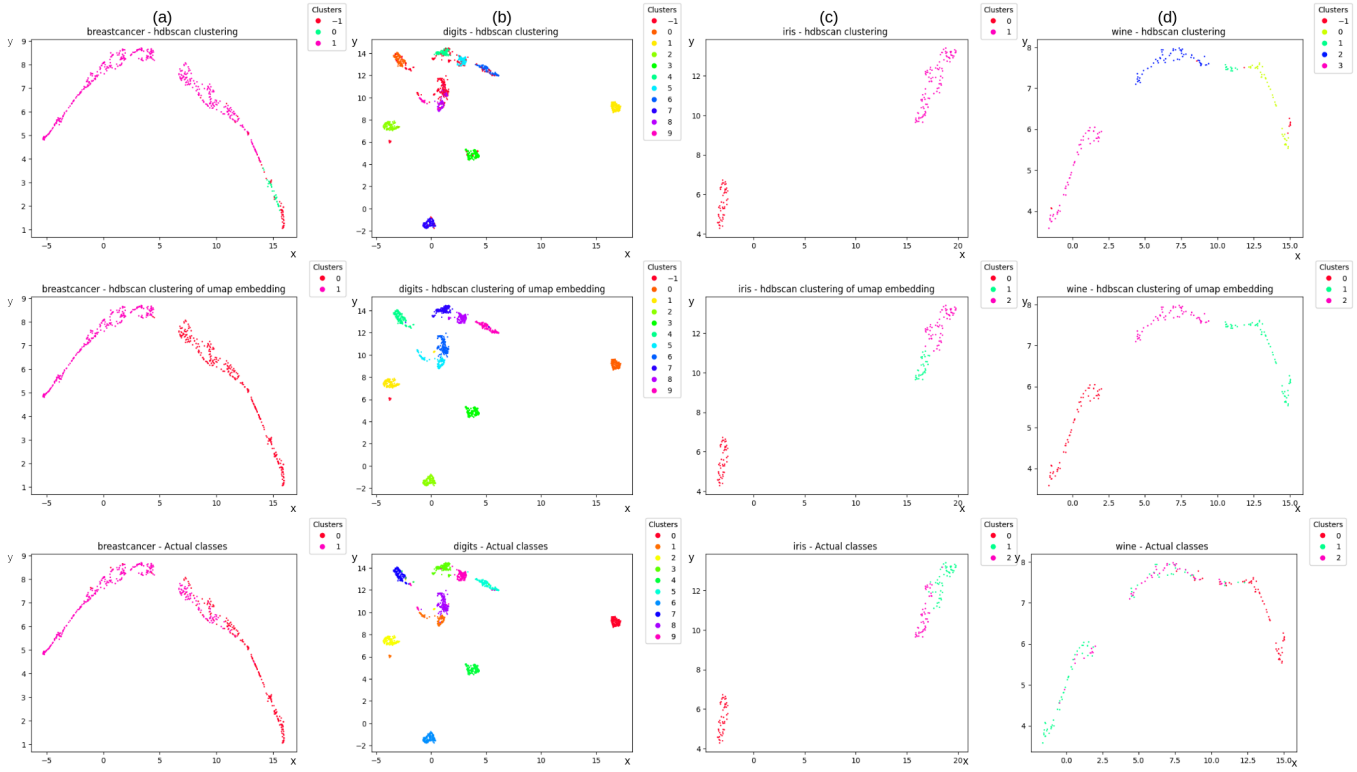

Figure 2: UMAP plots (n_components = 2, n_neighbours = 80) of clustering test data using HDBSCAN with and without dimensionality reduction. The top row is HDBSCAN results, the middle is UMAP^{+}, and the bottom row contains the known classes for each dataset. Columns (a), (b), (c), and (d) illustrate the Breast Cancer, Digits, Iris, and Wine datasets, respectively. Data points belonging to the -1 label are noise found by HDBSCAN. UMAP's ability to be used for visualisation purposes and assist in cluster analysis is highlighted here. Note that the axes for each plot are not meant for interpretation; the points are simply a projection onto a two-dimensional plane.

The Silhouette and Davies-Bouldin Index were adjusted to ignore noise if HDBSCAN found any as it would negatively impact the metrics otherwise. We then called the proportion of data placed into clusters "coverage" (cov.), with a minimum value of 0 (all noise), and a maximum value of 1 (no noise). Regarding the test datasets, the number of clusters found by each method was recorded as k. For MD trajectories, however, the total number of clusters is irrelevant when compared to the number of *dominant* clusters, as those represent the main conformations present for a given molecule. For clustering meningococcal Y and W, we considered k to be the number of clusters which occupied more than 10% of the total trajectory.

While the Silhouette and Davies-Bouldin Index metrics for MD trajectories can give an indication of overall clustering performance, the average root-mean-square deviation (RMSD) for each resulting cluster was calculated to give us actual information about the variation within each cluster. Higher values indicate disorder and a noisier cluster, and lower values indicate an ordered cluster with minimal difference between each frame. It is important to note, however, that getting the lowest possible value was not the aim. A trade-off between cluster count and noise within each cluster had to be made. Should we have focused too much on getting a low RMSD value, the number of clusters produced would increase to meaningless levels as the smallest of differences between frames would result in them being considered as separate conformations.

For clustering the test datasets, all but one parameter from our algorithms were tweaked in such a way as to provide as close to the actual number of classes present in each dataset. This was to show a best-case scenario for clustering with and without UMAP. The only parameter left fixed at 1 was min_samples for HDBSCAN. Since every instance in each dataset has a known class, we wanted as few data points to be considered as noise.

Since we are considering the carbohydrate molecules of the MD data to be unseen, multiple configurations of parameters for both HDBSCAN and UMAP had to be considered as we could no longer base them on how many clusters each produced. Like the validation testing, min_samples for HDBSCAN was set to the minimum value of 1 again. The rationale for this is that although each resulting cluster may be slightly more unstable due to the tolerance of noisier frames, the dominance of each conformation is much more accurately represented. Should too many frames in the trajectory be considered as noise, the ability to infer which clusters are more prominent would be lost.

Clustering with and without dimensionality reduction were referred to as "UMAP^{+}" and "HDBSCAN" methods, respectively. The parameters when naming each configuration for HDBSCAN were

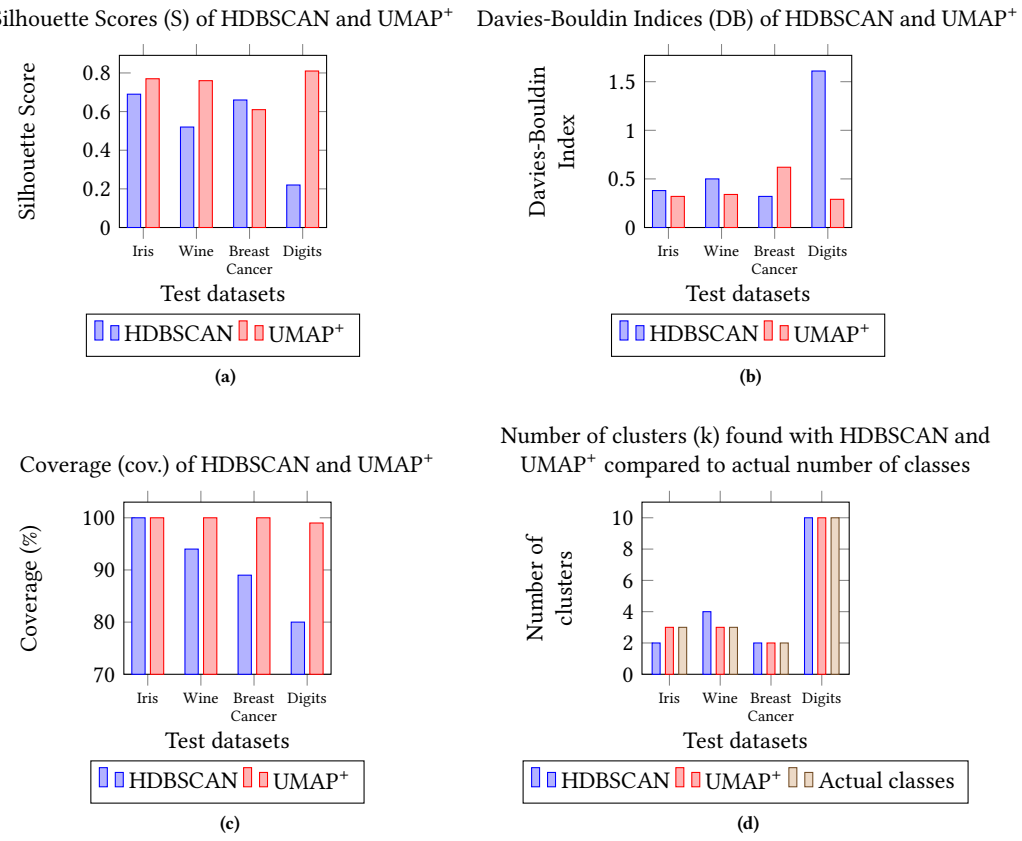

Figure 3: Plots showing clustering results of the test datasets. (a) is the Silhouette Score, (b) is the the Davies-Bouldin Index, (c) is the coverage, and (d) is the number of clusters found. The datasets are labeled in order of increasing dimensionality.

ordered as <min_cluster_size> - <min_samples>. For the UMAP$^+$ approach: <min_cluster_size> - <min_samples> - <n_neighbours> - <n_components>.

4 RESULTS

We first discuss the results from validation with the test datasets and then the clustering of the flexible meningococcus carbohydrate molecules.

4.1 Validation

Figure 2 shows plots of clusters found for each test dataset. Each column represents a specific dataset and is labeled from a to d. Metric scores are listed in Table S1.

4.1.1 Breast Cancer: Figure 2a. Using HDBSCAN on its own yielded S and DB scores of 0.66 and 0.32, respectively. S scores close to 1, and DB scores close to 0 indicate good clustering, so the metrics from HDBSCAN are fair. The HDBSCAN scores were better than the UMAP$^+$ approach, and this is the only dataset where this occurs (Figure 3a and Figure 3b). However, HDBSCAN only managed to cover 89% of the data points, while UMAP$^+$ covered 100% (Figure 3c).

In Figure 2a, although both methods found the correct number of clusters (two), UMAP$^+$ (middle row) allocated the data points far better than HDBSCAN (top row). HDBSCAN was unable to find the cut-off region between the two clusters in the middle of the parabolic shape. UMAP$^+$ clustered each half of the parabola correctly, making it much closer to the known classes (bottom row). Notably, some of the data points in Breast Cancer do not conform to the differentiation between the two classes, and so UMAP$^+$ was unable to separate the overlapping portion in the right half of the parabola.

4.1.2 Digits: Figure 2b. This dataset shows the largest discrepancy in metrics between the two methods. Both approaches attained the correct number of clusters, but metrics using HDBSCAN were much poorer than using UMAP$^+$ (Figure 3a and Figure 3b). In Figure 3c, we see that there is an inverse correlation between the dimensionality of each test data and the coverage of HDBSCAN. The Digits dataset has the highest number of dimensions of the four test datasets, and it causes the HDBSCAN method to yield the lowest coverage of 80%, opposed to the 99% of UMAP$^+$.

The data points in clusters were allocated well with both methods. However, much of the data classified as noise in the HDBSCAN

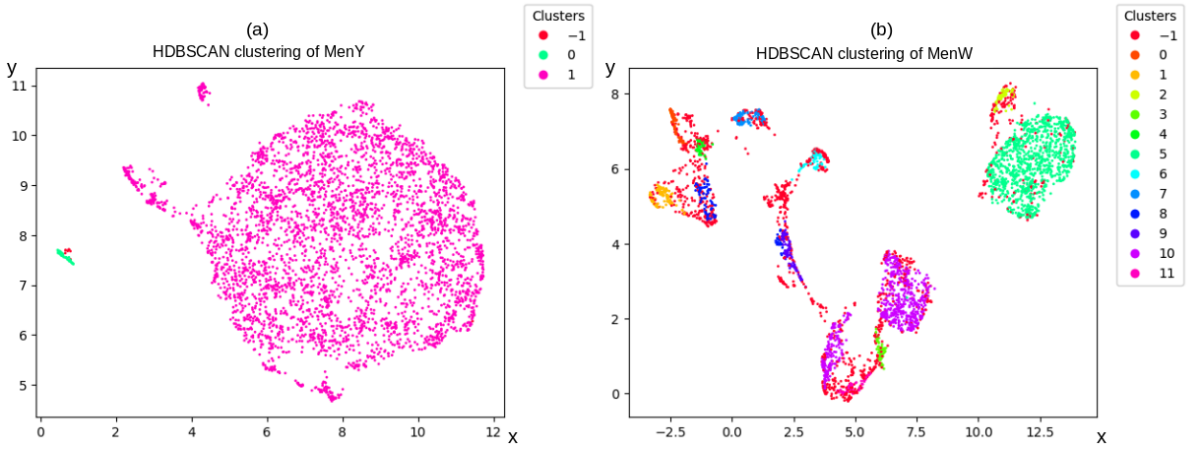

Figure 4: UMAP plots (n_components = 2, n_neighbours = 80) visualising MenY and MenW results using HDBSCAN (min_cluster_size = 50, min_samples = 1). Data points belonging to the -1 label are noise found by HDBSCAN. UMAP's ability to estimate the topology of the data reinforces the results found through clustering. MenY seems to have one extremely dominant conformation while MenW has a few. Note that the axes for each plot are not meant for interpretation; the points are simply a projection onto a two-dimensional plane.

method belonged to one of the classes of Digits, as seen when comparing the top and bottom row of Figure 2b. This further suggests that HDBSCAN's ability to cluster suffers when dealing with high dimensionality.

4.1.3 Iris: Figure 2c. Both HDBSCAN and UMAP$^+$ had reasonable metric scores for this dataset (Figure 3a and Figure 3b). Interestingly, HDBSCAN was unable to find the correct the number of clusters (three) as it was unable to differentiate between the two Iris species that were very similar. UMAP$^+$ correctly defined all three clusters, albeit not allocating all instances correctly. Both methods covered 100% of the data, and this dataset is the only one where HDBSCAN achieved full coverage for. This, again, reinforces the idea of dimensionality affecting HDBSCAN's performance, as the Iris dataset is only four-dimensional (Figure 3c).

4.1.4 Wine: Figure 2d. The UMAP$^+$ method yielded better metric scores (S = 0.76, DB = 0.34) than HDBSCAN (S = 0.52, DB = 0.50) and attained the correct number of classes (three). HDBSCAN found four classes and covered 84% of the data (Figure 3c). UMAP$^+$ had 100% coverage, but neither approach allocated the data points well. We found that although the classes themselves are well defined, not all instances of wine seem to conform to them, suggesting that perhaps not every feature is relevant in class definition.

4.1.5 Summary: Figure 3. Excluding the Digits dataset, the Silhouette and Davies-Bouldin metrics from HDBSCAN were competitive with UMAP$^+$, even though HDSCAN did not find the correct number of clusters for Iris and Wine. This suggests that while the two metrics are good for gauging overall cluster performance, it alone cannot be used to determine the number of clusters present in a dataset. UMAP$^+$ was able to find the correct number of clusters for all four datasets (Figure 3d).

Figure 3c clearly indicates an inverse correlation between the dimensionality of a dataset to HDBSCAN's coverage. Using UMAP$^+$, this relationship no longer exists, as reducing the number of dimensions first with UMAP allows HDBSCAN to cover 99%+ of each dataset.

4.2 Clustering MenY and MenW

Metric scores and average RMSD values for clustering the MenY and MenW datasets are shown in Table S2, Table S3, Table S4, and Table S5 under Supplementary Material. For each table, each row represents a clustering outcome using one particular configuration of parameters, which follows the naming scheme mentioned at the end of Section 3.6 - HDBSCAN: <min_cluster_size> - <min_samples>, UMAP$^+$: <min_cluster_size> - <min_samples> - <n_neighbours> - <n_components>. Plots showing these metrics are also in Supplementary Material in Figure S1.

4.2.1 MenY. HDBSCAN obtained very interesting results for this trajectory. While configurations with min_cluster_size = 25 and 50 (25-1 and 50-1) yielded poor metrics (Figure S1a and Figure S1c) and coverage above 90%, configurations with min_cluster_size > 50 (75-1 and 100-1) allocated none of the frames to clusters (Figure 5a). Using UMAP$^+$ produced slightly improved metric values overall but achieved close to 100% coverage for every configuration. UMAP$^+$ had no cases of 0% coverage even when using the same min_cluster_size value of 100 as HDBSCAN (Figure 5a). Excluding HDBSCAN's configurations with min_cluster_size > 50 (75-1, 100-1), every other parameter configuration used for HDBSCAN and UMAP$^+$ resulted in one dominant conformation (Figure 4c).

UMAP's ability to visualise high-dimensional data was used to assist in cluster analysis (Figure 4a). Using HDBSCAN with parameter min_cluster_size = 50, there is a very large central cluster (cluster 1), which again suggests that there is one dominant conformation of MenY. The only other cluster found (cluster 0) is very small in comparison and suggests that the dominant conformation takes up the majority of the MenY trajectory.

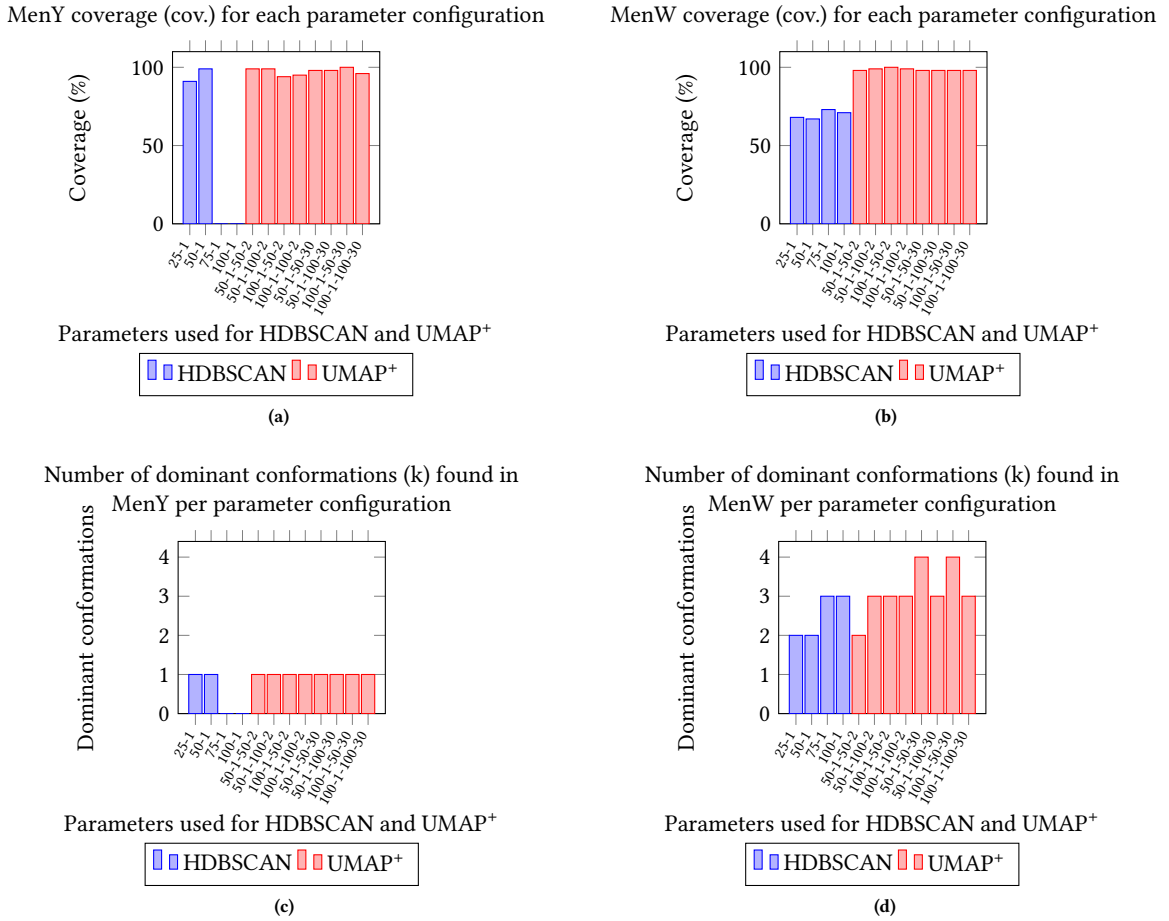

Figure 5: Plots showing clustering results of MenY and MenW. (a) and (b) illustrate the % coverage attained in MenY and MenW, respectively. (c) and (d) show the number of dominant clusters/conformations (k) found in MenY and menW, respectively. The x-axis for each plot consists of the parameter configurations used, and follows the naming scheme mentioned at the end of Section 3.6. HDBSCAN: <min_cluster_size> - <min_samples>. UMAP$^+$: <min_cluster_size> - <min_samples> - <n_neigbours> - <n_components>.

The average RMSD value for the dominant MenY conformation per configuration is in agreement between HDBSCAN and UMAP$^+$. The trajectory's original average of 2.18Å is reduced to between 1.89Å and 2.13Å for the dominant conformation by our methods, excluding the results with min_cluster_size > 50 (75-1 and 100-1) from HDBSCAN. This means that the average difference between frames in the dominant conformation was reduced by 3.3% - 13.3%. While the reduction is not dramatic, this is not surprising given that the dominant conformation consists of nearly the entire trajectory itself.

4.2.2 MenW. HDBSCAN was able to cluster MenW in a consistent manner in terms of metric performance. Silhouette and Davies-Bouldin scores were poor (Figure S1b and Figure S1d), but coverage was maintained at an average of 70%, with the highest coverage being 73% with min_cluster_size = 75 (Figure 5b). UMAP$^+$ was not only able to yield improved metrics compared to HDBSCAN, but also covered close to 100% of the frames present in all configurations. This difference in coverage is illustrated in Figure 5b. Figure 5d shows us the number of dominant conformations found in MenW per configuration. We see an indication of there being around 3 of such conformations. This result is reinforced when examining the UMAP embedding of MenW in Figure 4b. There are large clusters in the top left and right corner, as well as in the middle of the plot. The vast number of red data points that should belong to nearby clusters is a result of poor coverage, where 33% of the data was categorised as noise by HDBSCAN (min_cluster_size = 50 and min_samples = 1).

Once again, there were similar average RMSD values of dominant conformations between HDBSCAN and UMAP$^+$.The original RMSD of the trajectory was reduced from 3.29Å to below 2.78Å at all times, with many conformations having an RMSD near 2Å, which is a 39.2% reduction in noise. This suggests that while HDBSCAN was unable to cover the MenW trajectory well, it was able to correctly allocate the frames it did cluster. Had it not, the average RMSD in each cluster would be near the original value of 3.29. For UMAP$^+$,

it shows us that UMAP allows HDBSCAN to cover nearly 100% of the trajectory without significantly increasing noise within each cluster.

In Figure 6, we now compare the conformations found between MenY and MenW using UMAP$^+$ with min_cluster_size = 100, min_samples = 1, n_neighbours = 50, and n_components = 30 (100-1-50-30). MenY's dominant conformation lasts for 95% of its trajectory whereas MenW's dominant conformations cumulatively last for 78%. Although the structure of MenW is clearly more variable than MenY, it is interesting to note that the most dominant conformation for MenW (29%) is extremely similar to that of MenY. These results fall in line with findings in the literature. Kuttel et al. had also found that MenY consisted of one dominant conformation that took up 88% of the trajectory. MenW was also found to have four dominant conformations that each consisted of more than 10% of the trajectory (44%, 15%, 11%, 11%), with the most dominant conformation of MenW being similar to that of MenY [10]. The differences in cluster sizes with our findings can be attributed to the use of a different clustering algorithm [5] by Kuttel et al.

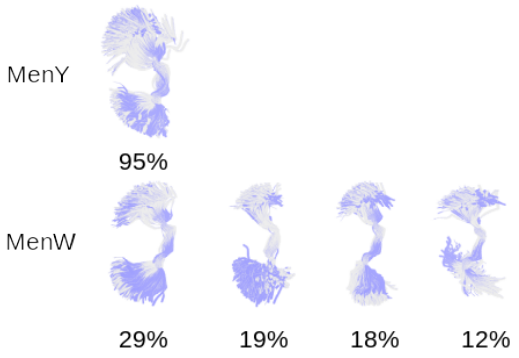

Figure 6: Comparison between the dominant conformations of MenY and MenW found with the UMAP$^+$ method. The parameters used were 100-1-50-30. The proportion that each conformation lasts for per trajectory is expressed as a percentage. We see that the most dominant conformation of MenW is very similar to that of MenY.

5 DISCUSSION

From the test data, UMAP$^+$ achieved almost 100% coverage across all four cases while also delivering better clustering results than HDBSCAN. The inverse correlation between dimensionality and coverage was evident in Figure 3c, and suggested that our experimentation with MD trajectories should see similar trends.

HDBSCAN's inability to cluster MenY with min_cluster_size set to 75 and 100 suggested that the data was too sparse for those parameters to be effective. We saw, however, that preprocessing MenY first with UMAP allowed HDBSCAN to cluster with the same min_cluster_size values. We intuit that "the curse of dimensionality" was indeed in effect, and that by reducing the dimensionality, HDBSCAN was then able to find the appropriate conformations. This was further reinforced when we found that the coverage for MenW increased from an average of 70% to near 100% when UMAP was used.

The combination of different parameters for the HDBSCAN method showed that the algorithm was quite sensitive to them, at least when regarding MenY. For UMAP$^+$, however, this sensitivity was generally reduced even with varying UMAP parameters. There was no single parameter configuration that completely outperformed the others. While the method is still parametric, UMAP seems to provide robustness when clustering for exploratory analysis.

The lack of meaningful difference in RMSD values achieved between HDBSCAN and UMAP$^+$ suggests that UMAP was able to improve the coverage of HDBSCAN clustering without introducing levels of noise that inhibit accurate cluster analysis, i.e., UMAP allowed HDBSCAN to more accurately determine the dominance of each conformation present. Interestingly, the RMSD values attained by the HDBSCAN method contradict the poor clustering performance suggested by the Silhouette score and Davies-Bouldin Index. As such, this bears further investigation in future work.

UMAP has also shown that apart from improving clustering results, it is immensely useful in cluster analysis as a whole due its ability to visualise high-dimensional data. As such, it may be used to gain rudimentary knowledge about MD trajectories without having to view the molecular structures themselves.

6 CONCLUSIONS

Clustering structural conformations from molecular dynamics trajectories is important for researchers to efficiently gather insights from them. The applications of HDBSCAN with and without UMAP preprocessing were compared to determine if HDBSCAN could find the dominant conformations of meningococcal Y and W serogroups, and if UMAP would be able to improve its clustering performance.

Without UMAP, HDBSCAN was found to provide good conformational results but was unable to accurately determine dominance due to its inability to cluster many of the frames. By using UMAP to preprocess the trajectories beforehand, HDBSCAN was able to cluster almost all of the frames present in both meningococcal Y and W trajectories while still accurately determining their conformational behaviours, allowing us to deduce the dominance of each conformation present. These findings suggest that using UMAP as a preprocessing step greatly benefits the clustering of MD trajectories with HDBSCAN, as it no longer suffers from sparse data resulting from high dimensionality.

7 ACKNOWLEDGEMENTS

I would like to thank my colleagues Nicholas Limbert and Robyn McKenzie for their support in developing the ClusterMol framework used to achieve the results in this paper. I also present my gratitude to my supervisor Michelle Kuttel who guided me throughout this paper with utmost enthusiasm, and for providing the meningococcal Y and W trajectories used for clustering. This work is based on the research supported wholly by the National Research Foundation of South Africa.

REFERENCES

[1] Mebarka Allaoui, Mohammed Lamine Kherfi, and Abdelhakim Cheriet. 2020. Considerably Improving Clustering Algorithms Using UMAP Dimensionality Reduction Technique: A Comparative Study. In *Image and Signal Processing*, Abderrahim El Moataz, Driss Mammass, Alamin Mansouri, and Fathallah Nouboud

(Eds.). Springer International Publishing, Cham, 317–325.
[2] Mebarka Allaoui, Mohammed Lamine Kherfi, and Abdelhakim Cheriet. 2020. *Considerably Improving Clustering Algorithms Using UMAP Dimensionality Reduction Technique: A Comparative Study.* 317–325.
[3] Richard Bellman. 2003. *Dynamic Programming.* Dover Publications, New York.
[4] Ricardo J. G. B. Campello, Davoud Moulavi, and Joerg Sander. 2013. Density-Based Clustering Based on Hierarchical Density Estimates. In *Advances in Knowledge Discovery and Data Mining*, Jian Pei, Vincent S. Tseng, Longbing Cao, Hiroshi Motoda, and Guandong Xu (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 160–172.
[5] Xavier Daura, Karl Gademann, Bernhard Jaun, Dieter Seebach, Wilfred F van Gunsteren, and Alan E Mark. 1999. Peptide Folding: When Simulation Meets Experiment. *Angewandte Chemie (International ed.)* 38, 1-2 (1999), 236–240.
[6] David L Davies and Donald W Bouldin. 1979. A Cluster Separation Measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-1, 2 (1979), 224–227.
[7] Dheeru Dua and Casey Graff. 2017. UCI Machine Learning Repository. http://archive.ics.uci.edu/ml
[8] E.R Hruschka, R.J.G.B Campello, A.A Freitas, and A.C.P.L.F de Carvalho. 2009. A Survey of Evolutionary Algorithms for Clustering. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 39, 2 (2009), 133–155.
[9] Michelle Mary Kuttel and Neil Ravenscroft. [n.d.]. Conformation and cross-protection in Group B Streptococcus serotype III and Streptococcus pneumoniae serotype 14: a molecular modeling study.
[10] Michelle M Kuttel, Zaheer Timol, and Neil Ravenscroft. 2017. Cross-protection in Neisseria meningitidis serogroups Y and W polysaccharides: A comparative conformational analysis. *Carbohydrate research* 446-447 (2017), 40–47.
[11] Erik Lindahl. 2015. Molecular dynamics simulations. *Methods in molecular biology (Clifton, N.J.)* 1215 (2015), 3–26.
[12] Leland McInnes and John Healy. 2017. Accelerated Hierarchical Density Clustering. (2017).
[13] Leland McInnes, John Healy, and James Melville. 2018. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. (2018).
[14] Ryan Melvin, Ryan Godwin, Jiajie Xiao, William Thompson, Kenneth Berenhaut, and Freddie Salsbury. 2016. Uncovering Large-Scale Conformational Change in Molecular Dynamics without Prior Knowledge. *Journal of Chemical Theory and Computation* 12, 12 (2016), 6130–6146. http://search.proquest.com/docview/1880017835/
[15] Ryan L Melvin, Jiajie Xiao, Ryan C Godwin, Kenneth S Berenhaut, and Freddie R Salsbury. 2018. Visualizing correlated motion with HDBSCAN clustering. *Protein science* 27, 1 (2018), 62–75.
[16] Satyasai Jagannath Nanda and Ganapati Panda. 2014. A survey on nature inspired metaheuristic algorithms for partitional clustering. *Swarm and Evolutionary Computation* 16 (2014), 1–18.
[17] Peter J Rousseeuw. 1987. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics* 20, C (1987), 53–65.
[18] Jörg Sander, Martin Ester, Hans-Peter Kriegel, and Xiaowei Xu. 1998. Density-Based Clustering in Spatial Databases: The Algorithm GDBSCAN and Its Applications. *Data Mining and Knowledge Discovery* 2, 2 (1998), 169–194.
[19] Erich Schubert, Jörg Sander, Martin Ester, Hans Kriegel, and Xiaowei Xu. 2017. DBSCAN Revisited, Revisited: Why and How You Should (Still) Use DBSCAN. *ACM Transactions on Database Systems (TODS)* 42, 3 (2017), 1–21.
[20] Jianyin Shao, Stephen W Tanner, Nephi Thompson, and Thomas E Cheatham. 2007. Clustering Molecular Dynamics Trajectories: 1. Characterizing the Performance of Different Clustering Algorithms. *Journal of chemical theory and computation* 3, 6 (2007), 2312–2334.

Supplementary Material

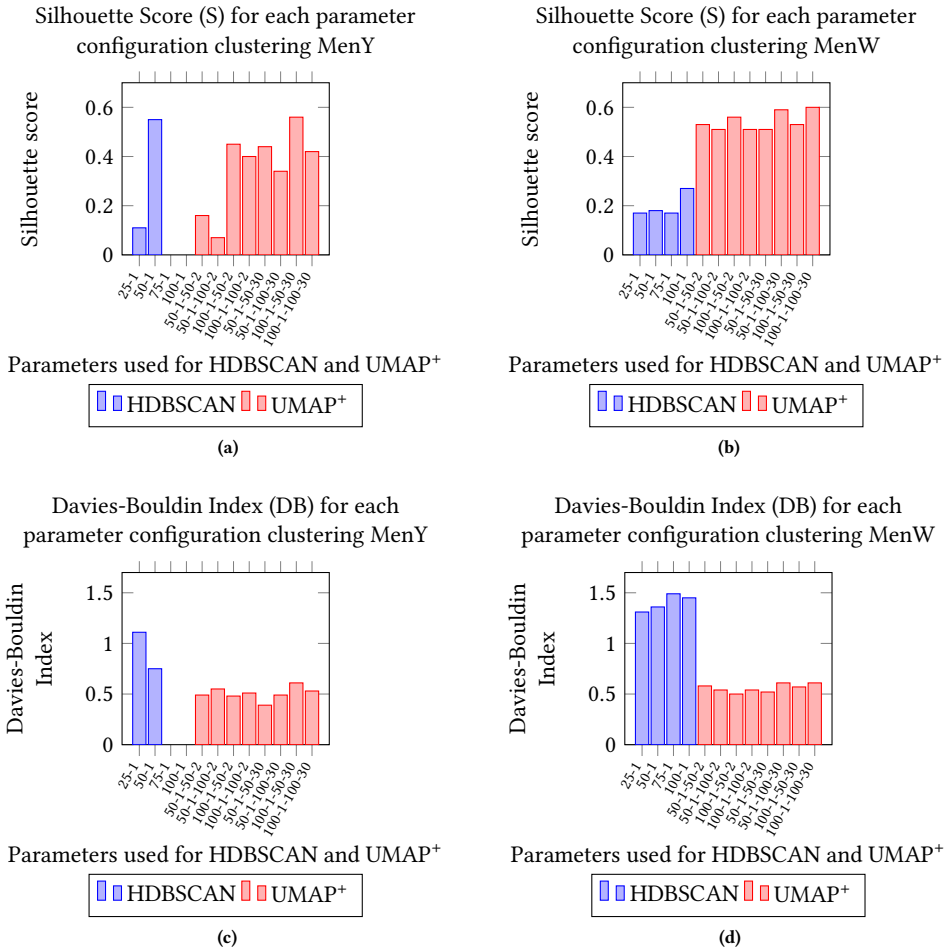

Figure S1: Plots showing clustering results of MenY and MenW. (a) and (b) illustrate the Silhouette Score (S) attained in MenY and MenW, respectively. (c) and (d) show the Davies-Bouldin Index (DB) found in MenY and menW, respectively. The x-axis for each plot consists of the parameter configurations used, and follows the naming scheme mentioned at the end of Section 3.6. HDBSCAN: <min_cluster_size> - <min_samples>. UMAP⁺: <min_cluster_size> - <min_samples> - <n_neigbours> - <n_components>.

Table S1: Metrics from clustering test datasets

	HDBSCAN				UMAP⁺			
Dataset	S	DB	cov.	k	S	DB	cov.	k
Breast Cancer	0.66	0.32	0.89	2	0.61	0.62	1	2
Digits	0.22	1.61	0.80	10	0.81	0.29	0.99	10
Iris	0.69	0.38	1	2	0.77	0.32	1	3
Wine	0.52	0.50	0.94	4	0.76	0.34	1	3

Table S2: Metric scores of HDBSCAN results clustering meningococcal Y and W

	MenY				MenW			
Parameters	S	DB	cov.	k	S	DB	cov.	k
25-1	0.11	1.11	0.91	1	0.17	**1.31**	0.68	2
50-1	**0.55**	**0.75**	**0.99**	1	0.18	1.36	0.67	2
75-1	-	-	0	0	0.17	1.49	**0.73**	3
100-1	-	-	0	0	**0.27**	1.45	0.71	3

Table S3: Metric scores of UMAP$^+$ results clustering meningococcal Y and W

	MenY				MenW			
Parameters	S	DB	cov.	k	S	DB	cov.	k
50-1-50-2	0.16	0.49	0.99	1	0.53	0.58	0.98	2
50-1-100-2	0.07	0.55	0.99	1	0.51	0.54	0.99	3
100-1-50-2	0.45	0.48	0.94	1	0.56	**0.50**	**1**	3
100-1-100-2	0.40	0.51	0.95	1	0.51	0.54	0.99	3
50-1-50-30	0.44	**0.39**	0.98	1	0.51	0.52	0.98	4
50-1-100-30	0.34	0.49	0.98	1	0.59	0.61	0.98	3
100-1-50-30	**0.56**	0.61	**1**	1	0.53	0.57	0.98	4
100-1-100-30	0.42	0.53	0.96	1	**0.60**	0.61	0.98	3

Table S4: Average RMSD values (Å) for the top 4 dominant conformations found per HDBSCAN configuration

	MenY				MenW			
Parameters	1	2	3	4	1	2	3	4
25-1	1.89				1.95	2.28		
50-1	2.13				1.95	2.28		
75-1	2.18*				1.95	2.48	2.71	2.59
100-1	2.18*				2.13	2.48	2.71	2.59

Table S5: Average RMSD values (Å) for the top 4 dominant conformations found per UMAP$^+$ configuration

	MenY				MenW			
Parameters	1	2	3	4	1	2	3	4
50-1-50-2	1.99				2.83	2.07		
50-1-100-2	1.99				2.85	2.07	2.82	
100-1-50-2	1.99				2.83	2.07	2.78	
100-1-100-2	1.99				2.85	2.07	2.82	
50-1-50-30	2.02				2.05	2.78	2.12	2.56
50-1-100-30	2.10				2.23	2.78	2.12	
100-1-50-30	2.07				2.05	2.78	2.28	2.56
100-1-100-30	2.02				2.23	2.78	2.27	