Correlated Histogram Clustering

Brice Brosig
Lone Star Analysis
Denton, TX
bbrosig@lone-star.com

Randal Allen
Lone Star Analysis
Orlando, FL
rallen@lone-star.com

ABSTRACT

Many popular clustering algorithms require *a priori* knowledge of the number of clusters, e.g., k-means and spectral clustering. This requires you to already know some things about the data (which we often don't) or that you have to try to learn things about that data which becomes intractable for large, unstructured, and high-dimensional data. Nearly all popular clustering algorithms use distance as the metric – though not inherently bad all the time – it can introduce hyperparameters like distance thresholds that again, require *a priori* knowledge about the data.

This paper introduces Correlated Histogram Clustering (CHC) which requires no *a priori* knowledge for the number of clusters and assumes nothing about the magnitude of values in any dimension. Designed to handle large, unstructured, high-dimensional, and noisy data, CHC leverages probabilistic techniques to build density estimates rather than using distance metrics. CHC uses the lowland modality algorithm to determine the modes of each dimension and then correlates the modes with points in the original dataset to form a cluster centroid. This cluster centroid may then be used for training, thereby substantially reducing the amount of data needed for supervised learning.

Sorting the significant few data values from the insignificant many in training data using CHC can be used to transform training syllabi by identifying which elements in a syllabus correlate with real skill attainment, and which elements do not accelerate skill attainment. Additional benefits of applying CHC to large, unstructured, high-dimensional, noisy data include dimension reduction and an understanding of the modal nature of the data. In one supervised learning classification application, twenty-seven features were reduced to only three, an 89% reduction of the dataset.

ABOUT THE AUTHORS

Brice Brosig is a Research Engineer at Lone Star Analysis. He contributes to intellectual property development by implementing novel Artificial Intelligence/Machine Learning algorithms and systems as part of the Research and Development team. He has Machine Learning research experience through the Electrical Engineering department and is a Teaching Assistant for the Computer Science department at the University of North Texas – where he is completing his M.S. in Computer Science, received his B.S. in Computer Science, and received his certification in Game Programming.

Randal Allen is the Chief Scientist of Lone Star Analysis. He is responsible for applied research and technology development across a wide range of M&S disciplines and manages intellectual property. He maintains a CMSP with NTSA. He has published and presented technical papers and is co-author of the textbook, "Simulation of Dynamic Systems with MATLAB and Simulink." He holds a Ph.D. in Mechanical Engineering (University of Central Florida), an Engineer's Degree in Aeronautical and Astronautical Engineering (Stanford University), an M.S. in Applied Mathematics and a B.S. in Engineering Physics (University of Illinois, Urbana-Champaign). He serves as an Adjunct Professor/Faculty Advisor in the MAE department at UCF where he has taught over 20 aerospace-related courses.

Correlated Histogram Clustering

Brice Brosig
Lone Star Analysis
Addison, TX
bbrosig@lone-star.com

Randal Allen
Lone Star Analysis
Orlando, FL
rallen@lone-star.com

INTRODUCTION

Unsupervised learning is a machine learning technique that aims to understand the underlying structure of data without any part of the data being a direct indicator of the class that it belongs to – that is, we have no "label" as a field in the dataset or stream data that we are performing unsupervised learning on. Unsupervised learning algorithms are overwhelmingly "clustering" algorithms where each instance of data is assigned to some cluster or group and the output is a labeled dataset.

One can split two types of clustering algorithms, one where *a priori* knowledge of the number of clusters is given to the algorithm as a hyperparameter and the other where the algorithm derives this number of clusters. The advantages that the second has over the first should be obvious: we often do not know the number of clusters associated with a given dataset; in fact, we might be running such an algorithm to find out exactly this piece of information. Popular algorithms that require *a priori* knowledge of the number of clusters are K-Means, Spectral Clustering, and Agglomerative Clustering. Popular algorithms that do not require such knowledge are DBSCAN and BIRCH.

What is needed is a simple clustering algorithm that does not require a priori knowledge of the number of clusters; one that can utilize statistics to identify cluster's centroids rather than distance metrics that become arcane and problem specific. Correlated Histograms Clustering, CHC, does exactly this: it requires no a priori knowledge of the number of clusters, and it uses statistics that are understandable as its metric for determining cluster centroids rather than Euclidean distance (which an overwhelming majority of algorithms use).

CORRELATED HISTOGRAMS CLUSTERING

Overview

This approach to clustering leverages statistics and correlates histogram data to determine the centroids of the clusters of data. Rather than using distance as the metric like other algorithms, we look at the distribution along each dimension of the data, consider the modes of those distributions, and then reconcile the modes from each dimension with one another to determine the cluster centroids. By looking at one mode for some dimension, say X, we can find a nearest value, x, which corresponds to the point, (x, y, …) from the data set. We can use one of the other points, say y, to find the nearest peak in dimension Y that *correlates* to modes from different dimensions.

Finding modes

The crux of CHC is finding the modality of each dimension. This is a fairly difficult task especially when dealing with noisy data (real-world data often is). One could use any method to determine this value and then continue with the rest of CHC, in our example we use the Lowland Modality Algorithm (Akinshin, 2020) that makes use of Density Estimates. The goal is to use a density estimate that is accurate to the underlying distribution of our data, robust against noisy data, and represents the modality well. Many density estimates exist and could be candidates; this paper uses Quantile Respective Density Estimates (QRDE) using the Harrel-Davis method since it performs well on noisy data and differentiates modes that are close to one another (Akinshin, 2020).

Modes in the lowland modality algorithm are defined as the highest histogram peak, M, between two other peaks, P_1 and P_2, such that the proportion of the area between M and P_i *just in the histogram bins* and the total rectangular area between the two (where the width is the distance between bin edges and height is the height of P_i) is greater than some threshold value, called the *sensitivity.* This hyperparameter is set by the user and is passed along to the

Lowland algorithm – were one to use a different modality detection technique, there would be no sensitivity parameter (though there could be other hyperparameters associated with that other technique).

For building the Harrel-Davis density estimate, or any density estimate / histogram for that matter, one must input the number of bins. There are many bin "rules of thumb" for histograms and density estimates like the square root rule, Sturges' rule, Rice's Rule, etc. These methods can produce acceptable results for the QRDE but, by default, this method optimizes the Shimizaki Scoring Function to get an *optimal* bin count (Shimizaki and Shinomoto, 2007).

$$\text{score}_n = \frac{(2 * \mu - S^2)}{(max - min)^2}$$

Where μ is the mean number of elements in each bin for a histogram with n bins and S^2 is the variance of the same. *max* and *min* are the max and min from the data set for which the density estimate is being constructed (note that these values will be the same for each estimate as the sample remains the same, only the bins and the values in each bin change). This method has advantage over the typical choices since it consider the statistics of the bin frequencies rather than just the number of data points.

Correlating the Modes

Once one has the modes for each dimension, they need only to find which modes correspond to which data points and then from those data points, which modes are correlated. To demonstrate, consider a 2-dimensional dataset of (*x*, *y*) values. Each mode of dataset X corresponds to some nearest value *in* dataset X which, in turn, corresponds to an (*x*, *y*) point. This corresponding *y* value then corresponds to some nearest mode of the Y dataset.

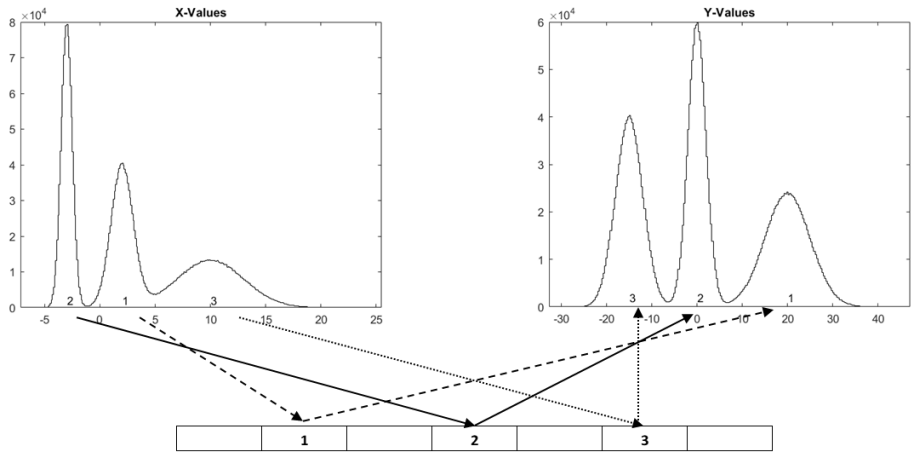

Figure 1 – Correlation of Modes

APPLICATIONS

Consider that the data in Figure 2 was gathered from many training syllabi and plots the training metrics for trainees in each category of the syllabi. Note, of course, that this data set is trivial for one to pick the clusters out of and, furthermore, trivial to fit a curve to. We use it here for demonstrative purposes.

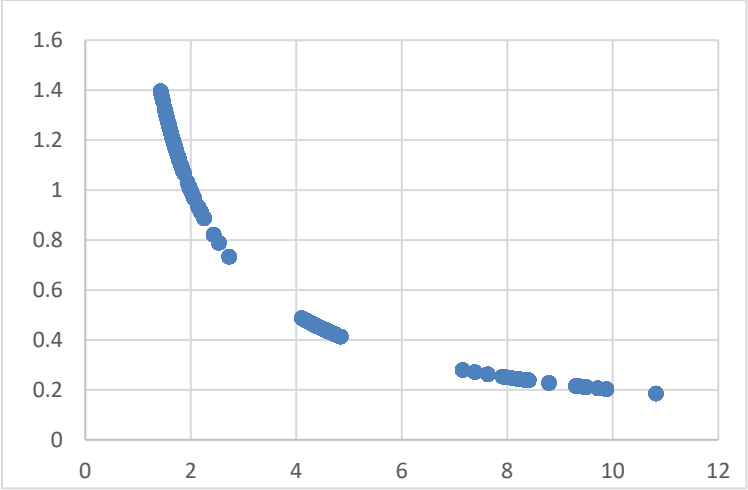

Figure 2 – Scatter of example data from training syllabus, skill attainment

An insight that could be gleaned from such a dataset is *what types of trainees does this syllabus produce?* In figure 2, we see that we form 3 distinct clusters based off the results of two training criteria. The *centroids* of these clusters would give one a very simple view into what metrics on which criteria correspond to the skill attainment that your training program wants. We will apply Correlated Histograms Clustering to this dataset to find exactly this.

Below are two histograms representing this multimodal statistical model based on 10,000 pairs of points, (a, b) such that a in dataset A and b in dataset B. By examining the Figures 2, 3, and 4 one will surmise this is most likely a tri-modal distribution.

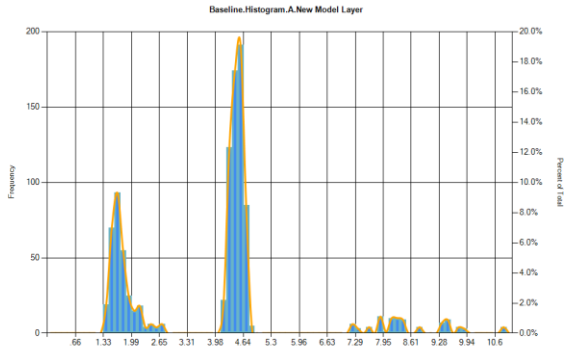

Figure 3 – Tri-Modal Histogram for Data Set A

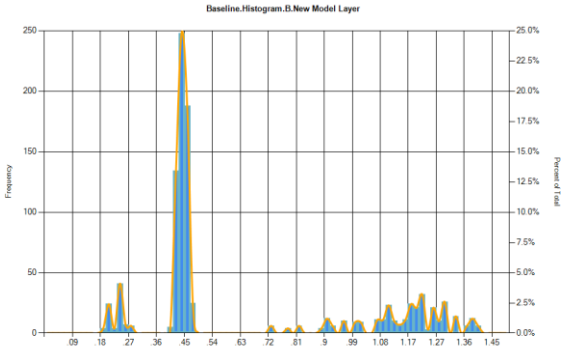

Figure 4 – Tri-Modal Histogram for Data Set B

Modes

The construction of density estimates alongside an algorithm for interpreting that estimate is used to determine the modality of each data set. To get density estimates that are sensitive enough to determine the different modes in the dataset yet robust enough to not over or underestimate the number of modes, we use the Harrell-Davis method for QRDE and the Lowland Modality Algorithm. Below are the density estimate plots for datasets A and B.

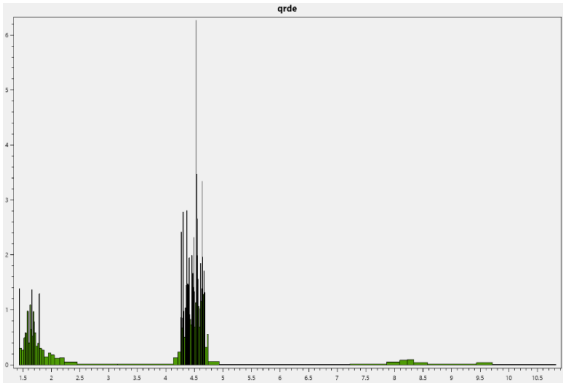

Figure 5 – QRDE for dataset A

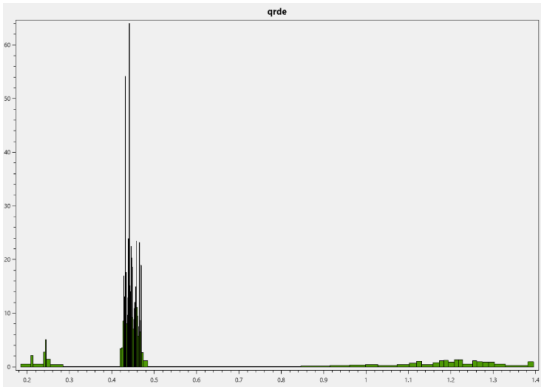

Figure 6 – QRDE for dataset B

Upon completion of identifying the modes, the tri-modal centroids for data set A are (**1.654, 4.533, 8.280**) and the tri-modal centroids for data set B are (**0.244, 0.441, 1.213**). What is unobvious is the connection between the data.

As is typically done, histograms are sorted as shown previously in Figure 2 and Figure 3. However, this sorting approach leads one to believe the order of A's centroids are the same as those for B's centroids, i.e., A(1.654) correlates with B(0.244), A(4.533) correlates with B(0.441), and A(8.280) correlates with B(1.213). But in this case, they don't go together as one may be led to believe. This is resolved through indexing.

Indexing

Recall correlating the modes according to Figure 1. If the data is indexed, then one simply looks for an index corresponding to a particular centroid (from A's data set) and then uses that same index to locate the other centroid (from B's data set). For example, data set A has one of many indexes that match the value 4.533 (within a few values of the second decimal place), one of which happens to be the index 115. Looking at data set B, index 155 leads one to find a corresponding value of 0.4412. Recognizing 0.4412 is near 0.441, one concludes that one of the centroid clusters (A, B) is the pair **(4.533, 0.441)**. It turns out that this just happens the be the same as the second elements in the histogram order for A and B.

Repeating the methodology, data set A has one of many indexes that match the value 8.280 (within a few values of the second decimal place), one of which happens to be the index 566. Looking at data set B, index 566 leads one to find a corresponding value of 0.240, which just happens to coincide with the centroid. Thus, one concludes that another of the centroid clusters (A, B) is the pair **(8.280, 0.244)**. The astute reader will recognize this is <u>not</u> in order of the histogram data. The *third* centroid value of A corresponds to the *first* centroid values of B.

One can repeat the methodology for the last pair (or deduce by elimination) that is it must be **(1.654, 1.213)**. Again, this is <u>not</u> in the order of the histogram data. The *first* centroid value of A corresponds to the *third* centroid values of B.

The aggregated set of three <u>correlated</u> centroids are **(1.654, 1.213)**, **(4.533, 0.441)**, and **(8.280, 0.244)**. This indicates to the practitioner that there are, effectively, 3 outcomes of the training syllabus and we can see the values of the training metrics that correspond to each of those outcomes.

RESULTS

To evaluate the performance of correlated histograms, we will look at other well-known algorithms on the cluster datasets from sklearn, a popular machine learning toolkit (scikit-learn, 2022). Labels are not available for these datasets (for most of them they are not really needed), these are mostly to demonstrate how each of the algorithms behave for different scenarios.

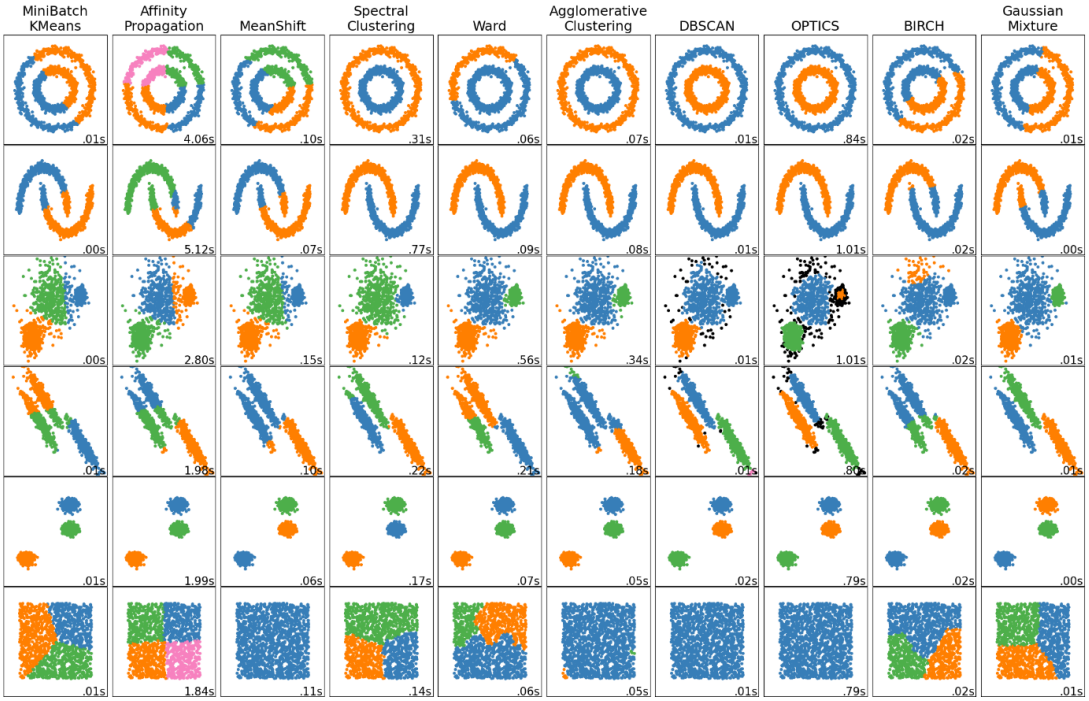

Figure 7 – Cluster Performance from Sklearn on the datasets (top to bottom respectively) noisy circles, noisy moons, varied, aniso, blobs, and unstructured.

Below is Correlated Histograms applied to a selection of the above datasets that embody the pros and cons of the method (unstructured, at the bottom of Figure 7, is left out).

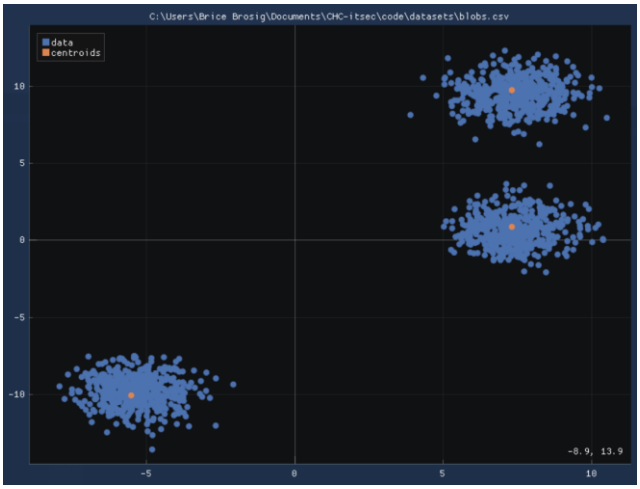

Figure 8 – Correlated Histogram clustering applied to blobs dataset. Number of QRDE bins = 45, 43 for x, y respectively; sensitivity = .5.

Figure 8 shows Correlated histograms identifying centroids near (-5, -10), (7, 0), and (7, 10). These are all near the center of the gaussian blobs.

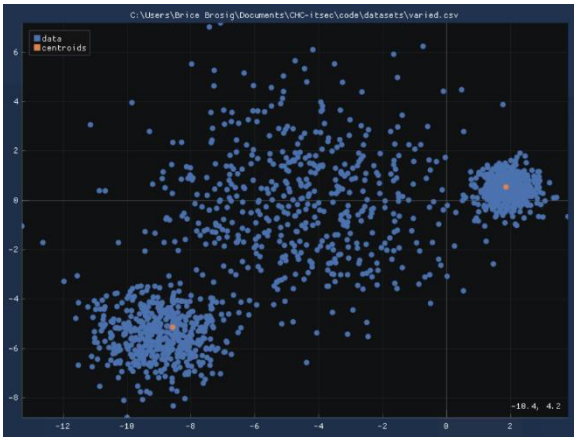

Figure 9 – Correlated Histograms applied to varied. Number of QRDE bins = 42, 32 for x, y respectively, sensitivity = .5.

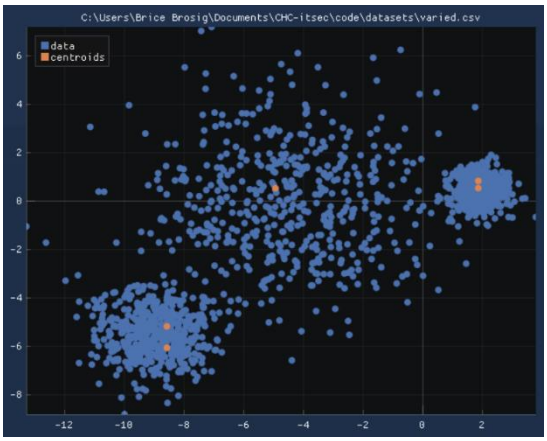

Figure 10 – Correlated Histograms applied to varied. Number of QRDE bins = 42, 32 for x, y respectively, sensitivity = .9.

In Figures 9 and 10, we see how Correlated Histograms responds to noisy datasets and how it can be adjusted using the sensitivity parameter to respond differently. In Figure 9, CHC identifies only two cluster centroids: one near (-8.5, -5) and another near (2, .5). In Figure 10, with sensitivity set to .9, the large, noisy blob in the center of the image is identified as being a cluster with centroid near (-5, 0).

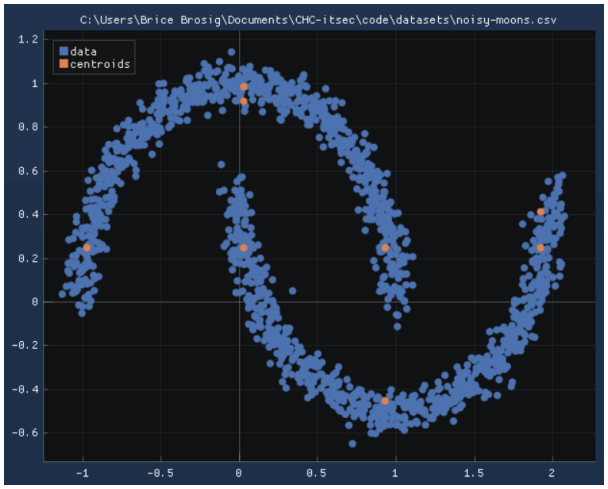

Figure 11 – CHC applied to the noisy-moons dataset with sensitivity = .90 and bin count for QRDE set to 100 for both x and y.

In Figure 11, CHC identifies 4 centroids for each moon shape. Note that in both clusters, the centroids are at the tips and vertex of the paraboloid. For the top moon shape, CHC identifies a centroid near (-1, 0.2), 2 centroids near (0, 1), and another near (1, 0.2). For the bottom moon shape, CHC identifies a centroid near (0, 0.2), a centroid near (1, -0.4), and two centroids near (2, 0.3).

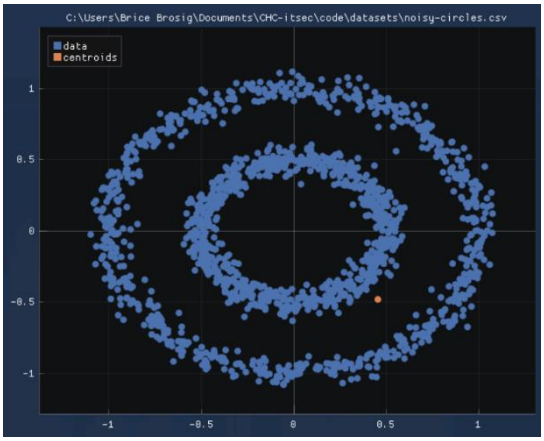

Figure 12 – CHC applied to the noisy-circles dataset with sensitivity = .5.

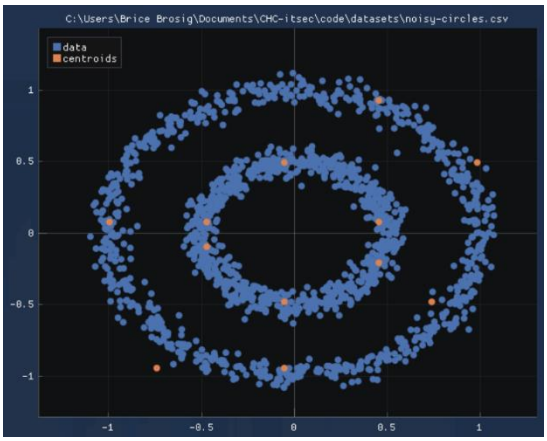

Figure 13 – CHC applied to the noisy-circles dataset with sensitivity = .9.

In Figure 12, CHC identifies a cluster that does not lie on the visually recognizable clusters. In Figure 13, with sensitivity set higher than in figure 12, it identifies many clusters, all of which lie on the visually recognizable clusters. Specifically, 6 different centroids around the outer circle and 6 more around the inner circle.

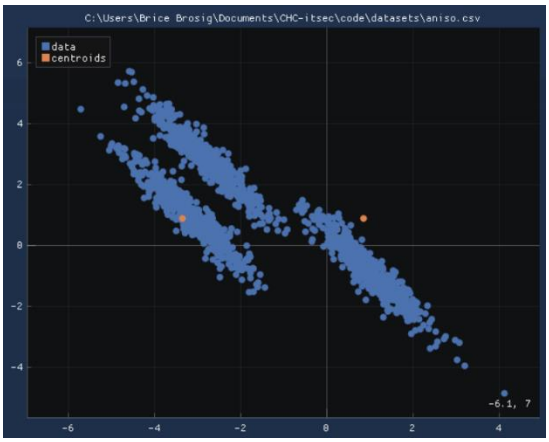

Figure 14 – CHC applied to the aniso dataset with sensitivity = .50.

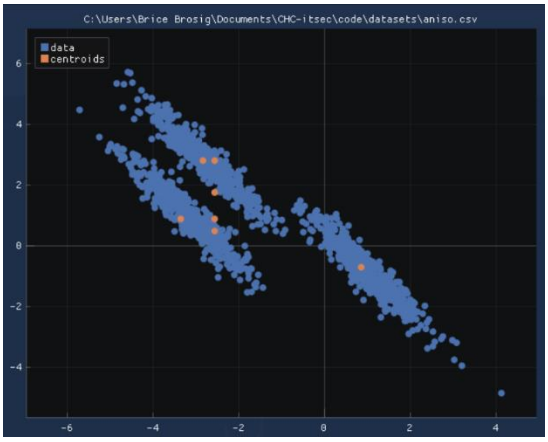

Figure 15 – CHC applied to the aniso dataset with sensitivity set to .90.

In Figure 14, CHC identifies only two clusters – one near (-3, 1) where the true centroid of the left most gaussian would be and another near (1, 1) that is outside of any cluster. Figure 15 identifies many cluster centroids: 3 near (-3, 1), 3 near (-3, 3) and then one near (1, -1). All the identified cluster centroids in figure 15 lie inside of the actual clusters and are near the true centroids)

CONCLUSIONS

Correlated Histograms Clustering is easily able to identify clusters in Figure 8 and datasets like it expect the same results. Moreover, CHC is effective at finding cluster centroids amongst *noisy* datasets since it looks at the modality of the data rather than looking at distance metrics and relying on thresholds relative to Euclidean distance between points. This is illustrated well in Figure 9 where amongst the noise, CHC identifies only two clusters. Compare this to DBSCAN, a popular choice for clustering, in Figure 7 (row 3, column 7) and see that it fails to discern noise from the cluster on the right.

CHC starts to underperform on datasets whose underlying statistics in each dimension do not necessarily identify the shape of distinct clusters – note Figures 12, 13, and 15. These results are *expected* from Correlated Histograms Clustering; that is, one can look at the statistics along each dimension of the data and make sense of the decisions the algorithm made by considering where peaks would lie and intersect with other dimensions. Methods for using histogram data in such scenarios is discussed in the future work sections. In some cases, setting the sensitivity higher can give a better idea of the where centroids in the data lie. Figure 16 has all the centroids "covered" – a practitioner would need further analysis on the relationship of centroids identified to find those that are close to one another and correspond to the same cluster.

FUTURE WORK

There are two areas where CHC can be improved: its ability to handle datasets where the clusters of the data are not obvious to the statistics of each dimension of data and its tendency to overestimate the number of clusters, specifically, outputting many clusters next to one another when the sensitivity is high.

Dealing with "Odd" Shaped Data

CHC effectively projects the data on the x and y axis, or in the case of n-dimensional datasets, projects it onto the orthogonal basis vectors and considers the frequency / density estimates on those lines. Perhaps there are other projection techniques where the histogram or density estimate of a dataset onto some other basis vector *or* some function yields information about the cluster.

Replacing QRDE

The Harrel-Davis Quantile Respective Density Estimate, for certain datasets, can be overly sensitive to false modes and the sensitivity parameter in the lowland modality can be hard to tune to account for this. The use of a traditional histogram seems like a reasonable candidate as they are less likely to have such jagged peaks as the density estimate. Even better *could* be the use of the Adaptive Histogram (Akinshin, 2020) that claims to express modality in histograms better without using a density estimate.

Other Modality Techniques

One of the difficult parts of the algorithm is finding the modality of a dataset. One can crudely do it by looks at just peaks in some histogram but a histogram that accurately depicts the modality is required. A potential solution is the m-value technique that sums up the differentials of frequencies for a histogram (Gregg, 2015).

REFERENCES

1. Akinshin, A., (2020), Lowland Multimodality Detection, https://aakinshin.net/posts/lowland-multimodality-detection/
2. Akinshin, A., (2020), Quantile-respectful Density Estimation Based on the Harrell-Davis Quantile Estimator, https://aakinshin.net/posts/qrde-hd/
3. Alcorn, M. A., Li, Q., Gong, Z., Wang, C., Mai, L., Ku, W.-S. et al. (2018). Strike (with) a Pose: Neural Networks Are Easily Fooled by Strange Poses of Familiar Objects. arXiv, 1811.11553v3.
4. Allen, R. (2019). Adaptive Nonconvex Optimization for AI and Machine Learning, I/ITSEC paper 19109.
5. Allen, R. (2019). Evolved AI - Interpretable Network Architectures for Machine Learning, I/ITSEC paper 19149.
6. Allen, R. (2020), Evolved Artificial Intelligence for Stochastic Clustering Unsupervised Learning, I/ITSEC paper 20258.

7. Allen, R., Engel, Z., Haney, E., (2021), Evolved AI for Model-Based Reinforcement Learning, I/ITSEC paper 21199.
8. Allen, R., Engel, Z., Volpi, M., (2021), Evolved AI for the Neural Net Enthusiast, I/ITSEC workshop ID9.
9. Gregg, B., (2015), Frequency Trails: Modes and Modality. https://www.brendangregg.com/FrequencyTrails/modes.html
10. DARPA, (2018), https://www.darpa.mil/news-events/2018-07-20a.
11. Hao, K., (2020), A debate Between AI Experts Shows a Battle Over the Technology's Future, MIT Technology Review, https://www.technologyreview.com/s/615416/ai-debate-gary-marcus-danny-lange
12. Heaton, J. (2013), Artificial Intelligence for Humans Volume 1: Fundamental Algorithms, p.155
13. Hume K. and Taylor, M. (2021), Why AI That Teaches Itself to Achieve a Goal Is the Next Big Thing, Harvard Business Review, https://hbr.org/2021/04/why-ai-that-teaches-itself-to-achieve-a-goal-is-the-next-big-thingKaplan A., and Haenlein, K., (2018), Siri, Siri in my Hand, who's the Fairest in the Land?, https://doi.org/10.1016/j.bushor.2018.08.004.
14. Knight, W., (2017), The U.S. Military Wants Its Autonomous Machines to Explain Themselves, https://www.technologyreview.com/s/603795/the-us-military-wants-its-autonomous-machines-to-explain-themselves.
15. Lamberth, M., (2019), The White House and Defense Department unveiled AI strategies. Now what?, https://www.c4isrnet.com/opinion/2019/02/27/the-white-house-and-defense-department-unveiled-ai-strategies-now-what.
16. Marcus, G. and Davis, E., (2019), Rebooting AI: Building Artificial Intelligence We Can Trust, Pantheon.
17. Marcus, G., (2020), The Next Decade in AI: Four Steps Towards Robust Artificial Intelligence, https://arxiv.org/pdf/2002.06177
18. Tadjdeh, Y., Interview with NDIA's Senior Fellow for AI, National Defense Magazine, 10 Jan 2020 https://www.nationaldefensemagazine.org/articles/2020/1/10/interview-with-ndias-senior-fellow-for-ai.
19. Tadjdeh, Y., Marines Lack Trust in Artificial Intelligence, National Defense Magazine, April 2021.
20. Zhou, H., et al., (2019), Deconstructing Lottery Tickets: Zeros, Signs, and the Supermask, https://eng.uber.com/deconstructing-lottery-tickets