Cluster Methods on Vertebral Column Data

Kimberly Glock  
 MSDS 5163-99  
 Lipscomb University  
Nashville, TN USA  
 [kmglock@mail.lipscomb.edu](mailto:kmglock@mail.lipscomb.edu)

Andre Louie  
 MSDS 5163-99  
 Lipscomb University  
 Nashville, TN USA  
 alouie@mail.lipscomb.edu

Charlie Napier  
 MSDS 5163-99  
 Lipscomb University  
 Nashville, TN USA  
 charlie.d.napier@gmail.com

ABSTRACT

The goal of this paper is to utilize a variety of clustering techniques to analyze a set of vertebral column data. Clustering is the process of partitioning data into groups (called “clusters”) based on their similarity. There are two broad approaches to clustering: compactness and connectivity. Compactness is an approach characterized by the “closeness” of data points that fall within the same cluster; compact around a cluster center. An example of this approach is a method called partitioning around medoids (PAM). Connectivity is an approach where data points that are connected or are immediately near each other are partitioned into the same cluster. If the data points are not connected, then they are not clustered together. Examples of this approach are hierarchical and spectral clustering. Our research team will utilize all three of these clustering methods on vertebral column data, observing the clustering statistics and visualizations to determine which methods best partition our data and what these partitions infer about the data itself.

KEYWORDS

cluster, PAM, hierarchical, spectral, data science, machine learning, unsupervised, analysis, vertebral

1. Introduction

As previously stated, clustering is defined as the process of partitioning data into distinct subgroups (or “clusters”) for the purpose of classification or pattern observation. In order to cluster efficiently and validate the results, the following processes must be performed: detecting structures present in data, known as “tendency”; determining the optimal number of quality clusters, known as “evaluation”; and observing the sensitivity of the clusters given minor changes in the data or algorithm, known as “stability”. Clustering techniques that do not require the pre-labeling of classes are referred to as unsupervised machine learning. Our team’s unsupervised clustering algorithm is intended to find hidden structures in the data, similar data points, and ultimately group them together. This will provide insight into these cluster attributes and patterns. The team will be using three cluster methods to determine which one best detects structures in our dataset: hierarchical, PAM, and spectral. These methods will each be described in further detail below.

1.1 Hierarchical Clustering Method

Hierarchical clustering is a connectivity-based clustering method which uses the distances between objects of the entire dataset to determine clusters. There are two types of hierarchical clustering: agglomerative and divisive. Agglomerative, known as AGNES, is the process where each individual data point starts in its own cluster, called a “leaf,” and eventually all clusters are merged into one cluster called the “root”. Divisive, known as DIANA, is the opposite; it is a process where are data points start off together in one cluster, the root, and are separated until all data points end up in their own cluster. For this paper, our team will use the agglomerative method. Hierarchical clustering generates a type of tree called a dendogram which shows the initial structure of the data. The process of how the algorithm defines clusters can be understood with the following steps. First, initial clusters are observed and the distance between these clusters is approximated. Next, the most proximate clusters are merged into a single cluster. The distances between the nearest pairwise clusters are then recalculated, and they continue to merge until only a single cluster remains, or we stop at the chosen number of clusters. This clustering method is deterministic, meaning that the same results will be obtained every time to process is run. Some drawbacks to hierarchical clustering include high sensitivity to noise and outliers, and a tendency to provide suboptimal results (known as a “greedy algorithm”). Finally, it is also inapplicable on datasets with little to no hierarchy.

1.2 PAM Clustering Method

The partitioning around medoids (PAM) method is a compactness-based clustering method which minimizes the distance between the non-medoid data points and the medoid which acts as the center of the cluster. A medoid is defined as the most centrally located point in a cluster where average dissimilarity between it and all other members of the cluster is minimal. PAM is considered “hard” clustering where each observation belongs to one cluster only. This method is closely related to K-means clustering, but it uses a centrally located medoid instead of the mean of the data points to act as the center of a cluster. The process of how the algorithm defines clusters can be understood with the following steps. First, the user determines the number of medoids to use (k), and the algorithm calculates the dissimilarity matrix. Then, clusters are formed by assigning each observation to the nearest medoid. The SWAP step attempts to improve clustering quality by exchanging selected medoids and non-selected medoids. If the objective function can be reduced by this interchange, the SWAP is carried out. The objective function corresponds to the sum of dissimilarities of all objects to their nearest medoid. If at least one medoid has changed in this SWAP step, then it again assigns every datapoint to its closest medoid, repeating the SWAP step again. If nothing changes, the algorithm ends. This method is less sensitive to outliers and noise because it uses a centrally located medoid instead of the mean. It is not suitable for clustering non-spherical data. You may obtain different results for different runs on the same dataset due to the typically random starting points.

1.3 Spectral Clustering Method

Spectral clustering is a connectivity-based clustering method which utilizes the distances between objects of the entire dataset to find clusters. In spectral clustering, data points are treated as nodes of a graph and clustering segments the graph. Initially, a similarity graph is computed. The algorithm then uses eigenvectors computed from the Laplacian matrix, which is computed from the similarity matrix. The data points (or “nodes”) are then mapped to a Euclidean low dimensional space to form clusters. This method is not affected by noise and outliers and the data can be of any shape. It utilizes K-means clustering in the final step, so clustering results are not always the same when repeated.

2.  Dataset

This data set, used for classifying vertebral column disorders, is taken from the UCI machine learning database. The set contains biomechanical data on 310 orthopedic patients, with each patient classified into one of three distinct classes: normal (NO; 100 samples), disc hernia patients (DH; 60 samples), and spondylolisthesis (SL; 150 samples). Ignoring the class identifiers, all attributes in this dataset are numerical. In total there are six numerical attributes: pelvic incidence, pelvic tilt, lumbar lordosis angle, sacral slope, pelvic radius, and grade of spondylolisthesis. For purposes of performing unsupervised learning on this dataset, the classifier attribute (ie: NO, DH, or SL) was removed in the initial phase of the analysis

3. Analysis and Validation

The overall analysis, along with each of the three clustering methods, were performed within an R Studio environment. Each clustering method was run using k=2, 3, and 4 clusters, though for research purposes, the team ran an initial function to determine the optimal number of clusters for a dataset of this type (in this case, that result was 2 clusters). To evaluate the performance of each method, a confusion matrix was generated for each scenario. In short, a confusion matrix is a table that summarizes the results of a classification or cluster model when making predictions, displaying the number of observations that were properly and improperly classified for each class. From this, a generalized accuracy score can be calculated, which will help determine the best clustering method.

4. Hypothesis

Provided that this is a clustering problem, we define the null hypothesis that the data is uniform, and there are no measurable clusters that exist within this dataset. We also define the alternative hypothesis, that there are clusters that do exist in this dataset. Additionally, given that this dataset contains three different classes of patients, our research team hypothesizes that the data points can be clearly partitioned into three distinct clusters, synonymous with the classes themselves. Alternatively, we hypothesize that, under a k = 2 clusters scenario, the normal patients would group into a one cluster while the abnormal patients (DH and SL) would group into the other cluster.

5. Results

Appendix II summarizes the statistics obtained from the confusion matrices of each clustering method. For k=2, the hierarchical, PAM, and spectral models achieved overall accuracies of 28%, 19%, and 31%, respectively. For k=3, they achieved accuracies of 50%, 48.7%, and 48%, respectively. Furthermore, each scenario produced fairly large p-values (including values of 1.0 for certain scenarios), implying that we cannot reject the null hypothesis: that there are measurable clusters that exist within this dataset. However, the plots shown in Appendix III appear to suggest otherwise. Here, across all scenarios, clusters can be clearly seen despite some moderate overlap. This visual contradiction, along with fact that certain scenarios yielded a p-value of 1.0, suggest that a bug is likely present within the clustering code, causing the confusion matrices to calculate inaccurately. Further research will be required to debug and ultimately rerun these confusion matrices.

Despite some potential inaccuracies present within the confusion matrices, there are still inferences that can be made based on other output results. Appendix I shows the labeled tables generated after fitting each clustering model to the data. Across these tables, our team identified two patterns in the data. First, for the k=2 and k=3 scenarios in every method, the majority of the SL patients (spondylolisthesis) were grouped within a single, clearly defined cluster. Second, again for both the k=2 and k=3 scenarios, the NO and DH patients (normal and disc hernia, respectively) possessed a tendency to be grouped together in the same cluster, separate from the SL class. Both of these observations can also be seen in the k=4 scenarios, albeit to a lesser extent.

Additionally, the team has begun to research one of the three classes in further detail. Initial research has shown that patients from the SL class alone can be further grouped into two “sub-clusters,” with roughly a 60%/40% split across the two clusters. Further research will be required to confirm the cause of this result, but our hypotheses will be described in the Conclusions section below.

**6. Conclusion**

In conclusion, I cannot reject or accept the null hypothesis that our dataset does not cluster. Further investigation is needed in order to accurately calculate our p-value so that I can make a statistically based decision. Interestingly our clustering visualizations do suggest that our dataset does cluster. The results for each clustering method do imply a pattern in how our dataset clusters. We found that our data mostly wants to cluster into two partitions. NO and DH cluster together overall and SL clusters to itself. I hypothesize this happens because the attributes in our dataset are bone measurements, excluding the attribute, grade of SL. Since DH is a disk issue and most often does not affect bone measurements nor does it misplace bones, DH clusters with NO and is not a distinguishable diagnosis given these attributes and our chosen clustering methods. The SL class clusters to itself and furthermore into two classes. I hypothesize this happens because the data attributes of bone measurements do capture the SL diagnosis as SL does involve broken and misplaced bones. Clustering may be able to capture this diagnosis. I hypothesize SL clusters into two clusters based on the grade of SL with one cluster being low grade and the other cluster being high grade. In theory, clustering may be able to capture and diagnose patients with SL versus NO and perhaps even distinguish patients with low grade SL versus high grade SL. Further investigation and research is needed.

**References**

A complete guide to K-means clustering algorithm. (n.d.). Retrieved October 28, 2019, from <https://www.kdnuggets.com/2019/05/guide-k-means-clustering-algorithm.html?fbclid=IwAR13m4XKJHdGY7bN4MHlv_ubBygITwu5YtzThi04xw4YYXXnk1PDFiWDydY>.

Brooks-Bartlett, J. (2018, September 7). Probability concepts explained: Bayesian inference for parameter estimation. Retrieved October 28, 2019, from <https://towardsdatascience.com/probability-concepts-explained-bayesian-inference-for-parameter-estimation-90e8930e5348>.

Choosing the Right Clustering Algorithm for your Dataset. (n.d.). Retrieved October 28, 2019, from <https://www.kdnuggets.com/2019/10/right-clustering-algorithm.html?fbclid=IwAR0aHnVHUZ2FtAlSxB_fZUMQ2dOGBmjLha92TGPWYwkd49_iTufBhv1L-YU>.

Clustering Distance Measures. (n.d.). Retrieved October 28, 2019, from <https://www.datanovia.com/en/lessons/clustering-distance-measures/?fbclid=IwAR0ZEjpadFKOMYaL-dCdZKEBgpUYRXqqcUe8IoHs8B7Yyvzvlqve3Eriv0c>.

Clustering Metrics Better Than the Elbow Method. (n.d.). Retrieved October 28, 2019, from<https://www.kdnuggets.com/2019/10/clustering-metrics-better-elbow-method.html?fbclid=IwAR1IO6FmWhFb9lMkcm7UE5n99xb1TXi4-Nd6W6WD6iWyLxjAElFlu4_Uiac>.

DataFlair Team (2019, July 13). Clustering in R - A Survival Guide on Cluster Analysis in R for Beginners! Retrieved from <https://data-flair.training/blogs/clustering-in-r-tutorial/>.

Doshi, N. (2019, May 2). Spectral clustering. Retrieved October 28, 2019, from <https://towardsdatascience.com/spectral-clustering-82d3cff3d3b7>.

Frank, Manohar, & Jignesh. (2017, February 14). How does k-means clustering work ? - an example. Retrieved October 28, 2019, from <http://www.machinelearningtutorial.net/2017/02/14/k-means-clustering-example/?fbclid=IwAR1HAdndtIxFFGSXluMKT2G_MlXiSyhNbQF7ytyU1-ypC85A143iTc2uTTk>.

Kassambara. (2018, November 12). 5 Amazing Types of Clustering Methods You Should Know. Retrieved October 28, 2019, from <https://www.datanovia.com/en/blog/types-of-clustering-methods-overview-and-quick-start-r-code/>.

Modi, M. (1970, January 1). Filter Bubble. Retrieved October 28, 2019, from <https://filterbubblecn.blogspot.com/2016/03/hands-on-spectral-clustering-in-r.html?fbclid=IwAR1OfPaWMlD-z2CH4-8s5TDFMdYMbSonQXXhpLuQ7UA_53rxtlPXO3QFrzM>.

Nss. (2019, June 11). Bayesian Statistics Explained in Simple English For Beginners. Retrieved October 28, 2019, from <https://www.analyticsvidhya.com/blog/2016/06/bayesian-statistics-beginners-simple-english/?fbclid=IwAR0PIaQRZBrEa-7_10dyMKr56f__aw0mtJ7JYz88M0UHuR_Qk5pwBVJ1bH8>

**Appendix I: Cluster Tables**

**Hierarchical**

**A screenshot of a computer

Description automatically generated**

**PAM**

**A screenshot of a computer

Description automatically generated**

**Spectral**

A screenshot of a computer screen

Description automatically generated

**Appendix II: Cluster Statistics**

**Hierarchical**

|  |  |  |  |
| --- | --- | --- | --- |
| **STATS** | **K=2** | **K=3** | **K=4** |
| AVG BETWEEN | 69.789 | 70.105 | 70.781 |
| AVG WITHIN | 44.426 | 38.340 | 36.263 |
| DUNN | 0.007 | 0.015 | 0.041 |
| RAND | 0.431 | 0.267 | 0.263 |
| ACCURACY | 0.28 | 0.50 | na |
| P VALUE | 1.0 | 0.23 | na |
| SENSITIVITY | 0.167/0.77/0.0 | 0.167/0.77/0.467 | na |
| SPECIFICITY | 0.348/0.714/1.00 | 0.636/0.714/0.988 | na |
| AVG SIL WID | 0.329 | 0.287 | 0.296 |

**PAM**

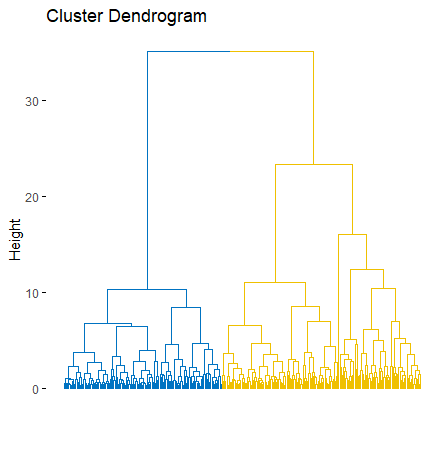
|  |  |  |  |
| --- | --- | --- | --- |
| **STATS** | **K=2** | **K=3** | **K=4** |
| AVG BETWEEN | 79.418 | 71.131 | 66.974 |
| AVG WITHIN | 42.065 | 36.736 | 34.166 |
| DUNN | 0.027 | 0.013 | 0.014 |
| RAND | 0.299 | 0.271 | 0.312 |
| ACCURACY | 0.19 | 0.487 | na |
| P VALUE | 1.0 | 0.477 | na |
| SENSITIVITY | 1.00/0.010/0.00 | 0.117/0.76/0.453 | na |
| SPECIFICITY | 0.440/0.481/1.00 | 0.616/0.700/1.00 | na |
| AVG SIL WID | 0.449 | 0.325 | 0.241 |

**Spectral**

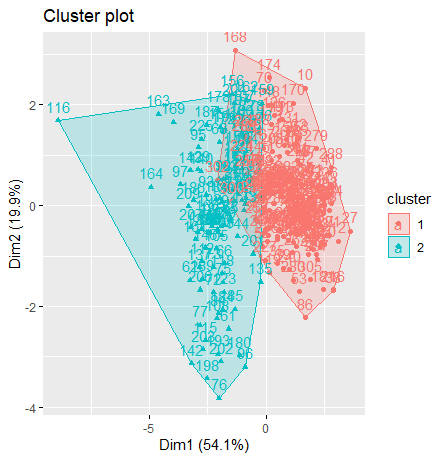
|  |  |  |  |
| --- | --- | --- | --- |
| **STATS** | **K=2** | **K=3** | **K=4** |
| AVG BETWEEN | 76.705 | 73.853 | 73.014 |
| AVG WITHIN | 41.855 | 37.146 | 35.075 |
| DUNN | 0.010 | 0.025 | 0.029 |
| RAND | 0.423 | 0.310 | 0.351 |
| ACCURACY | 0.31 | 0.48 | na |
| P VALUE | 1.0 | 0.617 | na |
| SENSITIVITY | 0.0/0.96/0.0 | 0.950/0.00/0.607 | na |
| SPECIFICITY | 0.488/0.591/1.00 | 0.592/0.786/0.906 | na |
| AVG SIL WID | 0.432 | 0.367 | 0.339 |

**Appendix III: Plots**

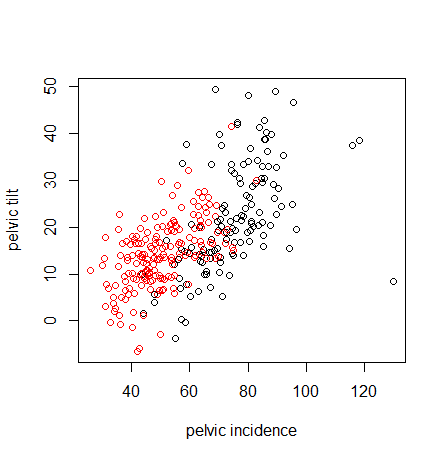
**Hierarchical k=2**



**PAM k=2**



**Spectral k=2**

****