2025

Machine Learning Assignment

DSCI8950

JOELL ERCHUL

Table of Contents

Theory of Hierarchical Clustering

Hierarchical Clustering is one of the many different types of machine learning algorithms. It is an unsupervised algorithm meaning that these algorithms do not need a human to be in the process in order to discover patterns and groupings. Typically, these unsupervised algorithms are good for clustering, association, and dimensionality reduction tasks. Like the name alludes, Hierarchical Clustering is one of the algorithms that fit into the clustering category for unsupervised algorithms and we can use this to study the relationship between all other data points in the system as well as estimate the relation between each sample in a quantitative way (Pai, 2021).

A dendrogram, or tree-like diagram as shown in Figure 1, is typically used to visualize the algorithm because it is easy to display the order of cluster division or merging. Dendrograms are also good for displaying what makes up a cluster, so you can answer what makes the clustering similar, and it is easy to see the distance between the data points. As described in Figure 2, the use of a dendrogram provides more information than just showing the data points in their relative clusters. "The sole concept of hierarchical clustering lies in just the construction and analysis of a dendrogram" (Pai, 2021). Because of the production of the dendrogram, we are able to use it to visually show relationships between clusters, which make Hierarchical Clustering more interpretable than some of the other clustering methods within unsupervised learning (J, 2023).

Hierarchical clustering has three unique milestones in which the dendrogram is created from. Step one is to calculate the distance matrix by computing the distance between every pair. Something like Euclidean distance can be used to calculate this distance metric. Step two is to merge the closest clusters together. Finally step three is to update the distance matric by recalculating the distances between the new cluster and the rest of the clusters. Steps two and three are repeated until there is only one cluster remaining, or a different stopping criterion has been reached (Geeks for Geeks, 2024).

There are two categories used to describe different approaches to Hierarchical Clustering, the first being Agglomerative Clustering, and the second being Agglomerative Clustering. Agglomerative is considered to be a bottom-up approach because all of the data points are continuously merged together until there is only one cluster at the top. Divisive is considered to be a top-down approach because it starts with the single cluster and continues to divide until all clusters are single points (IBM, 2021).

Hierarchical Clustering Parameters

Simply enough, the only parameters needed for a basic Hierarchical Clustering algorithm is your data set (Geeks for Geeks, 2024). Datasets can be categorical, binary, or continuous in nature to work with these models (Pushkar, 2025). Once you'd like to visualize your clusters in a way other than a dendrogram, you can provide the algorithm the number of clusters or max distance for cluster formation you would like to assess your data set at, depending on what libraries you use. There are also opportunities to specify which linkage method you think is best for your data. Experimenting with various linkage methods and analyzing the constructed dendrogram to determine a good amount of clusters are two of the best ways to start optimizing a Hierarchical Clustering algorithm (ML Journey, 2024).

Applications for Hierarchical Clustering

This clustering algorithm is a good consideration to use when you have a clear hierarchy in your data set or are looking for exploration with levels differing in granularity instead of having to commit to a specific number of clusters. It is important to understand however, that this can be very computationally expensive when it comes to large datasets due to pairwise distance computations between each observation (J, 2023). Some areas that you would want to avoid implementing Hierarchical Clustering for are with data sets that have not been cleaned of outliers, and again data size as it both increases the computational power needed and the complexity of the resulting dendrogram to analyze. Specific examples of good use cases for Hierarchical Clustering could be customer segmentation, bioinformatics, document clustering, image segmentation and social network analysis. In all of these cases, you are trying to categorize isolated data points into related groups, such as grouping customers based on their purchasing behavior (ML Journey, 2024).

Theory of Prophet

Prophet is generally used for the automatic forecasting of univariate time series data (Brownlee, 2020). It is good at what it is designed to do but does not have very much flexibility in using it for other cases outside of univariate time series. While limited in use cases, its appeal is being simple to understand and implement if you're only looking for forecasting this specific format of data. The model is able to incorporate desirable components for time forecasting of trends such as seasons and holidays (ryh4n, 2024). Seasonality is different from holidays as seasonality in the context of time series pertains to variations occurring over short periods of time that can't necessarily be considered trends (Geeks for Geeks, 2024). In addition to being easy to use with its simple syntax, Prophet is also good for handling data sets that have missing values and outliers or extreme changes, as long as the data being forecasted is univariate of course (ryh4n, 2024).

Time series analysis in general has some important components and concepts to consider and understand when working with these types of models. First is that the main part of output for time series models are temporal relationships, which means that the models will take preceding values to affect what the current value its working on results to be. Secondly, trends are very important to these models and recognizing what direction the trend is; however, trends do not always have to be in a direction, they can also be stable. Thirdly are seasonal patterns, which can vary in length, such as being daily occurrences, or even as long as yearly. Lastly, it is still important to know when data needs to be cleaned especially with time series data as these often contain random fluctuations that don't relate to specific trends or patterns, so these could create false areas to look at (Geeks for Geeks, 2024).

Prophet Parameters

At the very least, Prophet requires a data set that is univariate in nature and has date or timestamp data. For easier results, the format of date values should be in the format of YYYY-MM-DD and the format of timestamp values should be in the format of YYYY-MM-DD HH:MM:SS. To start making future predictions with the model, you'll have to specify how many periods you'd like the model to forecast out to (Facebook, 2024).

Applications for Prophet

Prophet is great for scenarios such as business forecasting. The business industry is really where you'd start gaining value from forecasting out what values will be at specific timeframes (Seko, 2024). An example could be a shop in a popular tourism location would want to use historical data on sales to analyze trends of when certain items sold more of so that they could ensure they have enough stock to support common timeframes of increased tourist visits, such as a surf shop in California catering to college students visiting on spring break. Another example would be to analyze flight trends since many people fly around certain points in the year.

References

Brownlee, J. (2020, May 8). *Time Series Forecasting With Prophet in Python*. Retrieved from Machine Learning Mastery: https://machinelearningmastery.com/time-series-forecasting-with-prophet-in-python/

Facebook. (2024, October 20). *Quick Start*. Retrieved from Prophet: https://facebook.github.io/prophet/docs/quick_start.html

Geeks for Geeks. (2024, June 12). *Hierarchical Clustering with Scikit-Learn*. Retrieved from Geeks for Geeks: https://www.geeksforgeeks.org/hierarchical-clustering-with-scikit-learn/

Geeks for Geeks. (2024, November 21). *Time Series Analysis using Facebook Prophet*. Retrieved from Geeks for Geeks: https://www.geeksforgeeks.org/time-series-analysis-using-facebook-prophet/

IBM. (2021, September 23). *What is unsupervised learning?* Retrieved from IBM: https://www.ibm.com/think/topics/unsupervised-learning

J, E. (2023, August 31). *When to Use Hierarchical Clustering: A Guide for Data Analysts*. Retrieved from Data Rundown: https://datarundown.com/hierarchical-clustering/

ML Journey. (2024, November 30). *Hierarchical Clustering in Python: A Comprehensive Guidev*. Retrieved from ML Journey: https://mljourney.com/hierarchical-clustering-in-python-a-comprehensive-guide/

Noble, J. (2024, August 05). *What is hierarchical clustering?* Retrieved from IBM: https://www.ibm.com/think/topics/hierarchical-clustering

Pai, P. (2021, May 07). *Hierarchical clustering explained*. Retrieved from Torwards Data Science: https://towardsdatascience.com/hierarchical-clustering-explained-e59b13846da8/

Pushkar, A. (2025, January 21). *What is Hierarchical Clustering? An Introduction*. Retrieved from IntelliPaat: https://intellipaat.com/blog/what-is-hierarchical-clustering/

ryh4n. (2024, January 3). *Understanding Forecasting Model Using Prophet — A Comprehensive Guide*. Retrieved from Medium:

https://reyhannananta.medium.com/understanding-forecasting-model-using-prophet-a-comprehensive-guide-e05a76ecc5bb

Seko, B. (2024, November 21). *Pros and Cons of Using Facebook's Prophet for Time Series Forecasting*. Retrieved from Easy Data Does It: https://easydatadoesit.org/pros-and-cons-of-using-facebooks-prophet-for-time-series-forecasting/

Appendix A: Figures

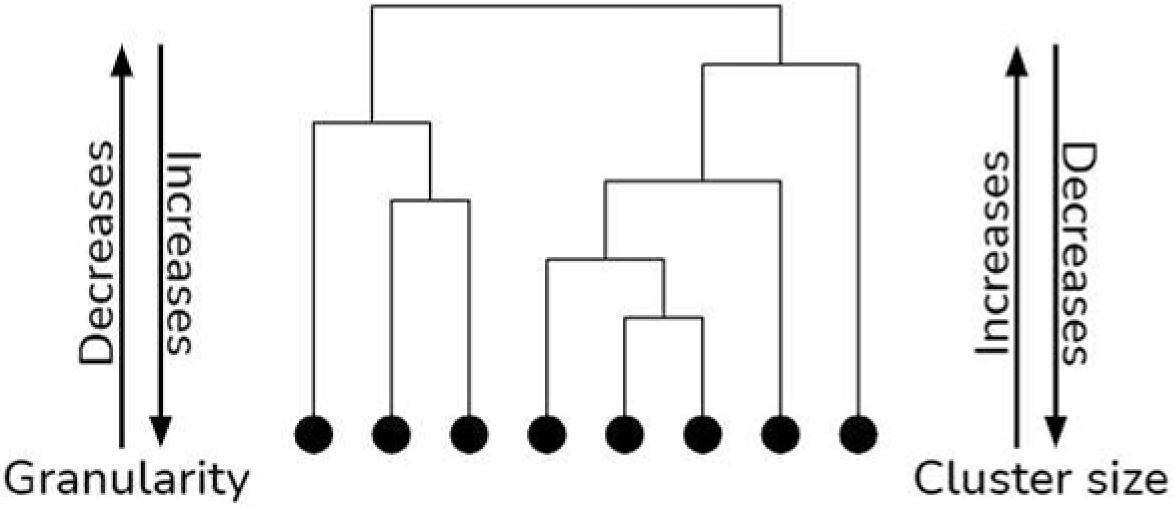

Figure 1: Example of a Dendrogram (Pai, 2021).

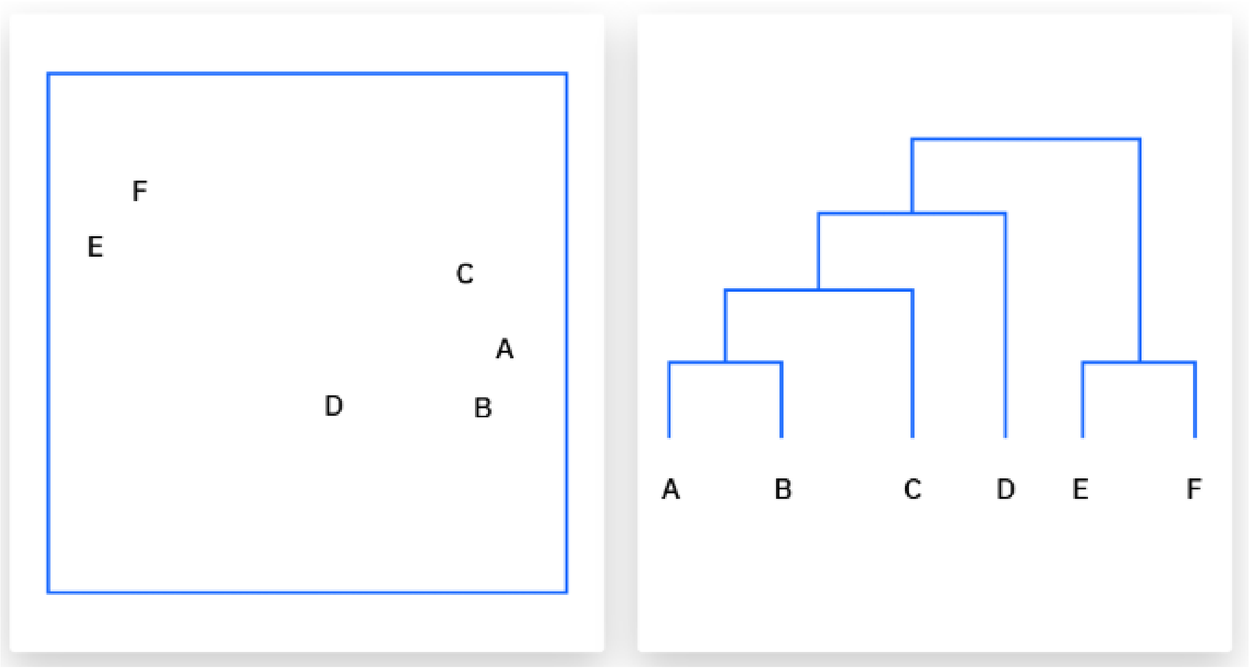

Figure 2: Visualizing Clustering with a Dendrogram (Noble, 2024).