Learning Cluster Analysis: A SAS Tutorial Using PROC CLUSTER

Authored by

Mohammed looti

November 14, 2025

Cluster analysis is recognized as a foundational technique in both modern statistical analysis and **machine learning**. Its core purpose is to uncover intrinsic patterns and latent structures hidden within complex datasets by grouping similar items together. This powerful methodology, frequently termed **clustering**, seeks to transform a collection of heterogeneous observations into meaningful, internally homogeneous groups.

The effectiveness of any clustering methodology rests on a critical balance: maximizing internal group cohesion while ensuring maximum separation between groups. This dictates that every observation placed within a cluster must exhibit strong shared characteristics, distinguishing them clearly from observations belonging to other clusters. Successfully achieving this balance is essential for defining natural segmentation boundaries that expose underlying differences in the population, differences that often remain obscured when relying only on basic descriptive statistics.

For analysts and researchers working within the **SAS** (Statistical Analysis System) environment, the primary utility for performing hierarchical clustering is the **PROC CLUSTER** procedure. This robust procedure offers a flexible framework for investigating complex data structures through diverse algorithmic approaches, making it indispensable for advanced data analysis, research, and projects that demand rigorous empirical segmentation.

This guide provides a detailed, step-by-step walkthrough focusing on the effective utilization of **PROC CLUSTER** within the **SAS** ecosystem. We will cover essential data preparation steps, thoroughly explain the critical procedural options, and demonstrate how to interpret the complex resulting output. All concepts will be anchored by a practical, real-world example of data segmentation, ensuring clarity and immediate applicability.

Understanding Cluster Analysis and Its Place in Unsupervised Learning

Cluster analysis, often synonymous with segmentation analysis, is fundamentally categorized as a method of **unsupervised learning**--a major branch of statistical and **machine learning** fields. The defining characteristic that separates unsupervised techniques from supervised counterparts is the absence of a predefined target or dependent **variable**. Rather than attempting to predict a known outcome, clustering algorithms explore the inherent characteristics of the data points themselves, striving to establish structure where no explicit labels exist. This exploratory nature makes clustering invaluable for initial data assessment and generating novel hypotheses.

The utility of clustering extends far beyond theoretical study, impacting numerous critical domains globally. In commerce, it forms the basis of robust **customer segmentation**, allowing businesses to tailor marketing campaigns, product development, and service protocols to distinct consumer groups. Across scientific disciplines, particularly **genomics** and biology, clustering assists in classifying species, identifying co-expressed genes, or grouping patients based on similar disease manifestations. Furthermore, in specialized areas like image processing and **anomaly detection**,

clustering helps rapidly pinpoint deviations from established patterns or norms.

Ultimately, a successful clustering effort is measured by its ability to simplify the complexity inherent in a large dataset. By drastically reducing the internal variability within each resultant group while maximizing the difference between groups, the process delivers a clean, actionable representation of the data's underlying structure. This simplification transforms raw, disparate data into understandable segments, providing decision-makers with powerful, interpretable insights necessary for strategic categorization.

The Mechanics of Hierarchical Clustering with PROC CLUSTER

The **PROC CLUSTER** statement in **SAS** is specifically designed to perform **hierarchical clustering**. Unlike partitioning methods, such as K-means, which yield a single, fixed set of partitions, hierarchical clustering constructs an entire hierarchy--a nested sequence of cluster formations. **PROC CLUSTER** predominantly employs **agglomerative techniques**. This process begins by treating every individual observation as its own cluster. It then iteratively merges the two closest clusters, step by step, until all observations are eventually consolidated into a single, comprehensive cluster.

A critical decision when executing **PROC CLUSTER** is the selection of the **linkage method**. This method is the mathematical rule that defines how the distance or dissimilarity between two existing clusters is calculated. Different **linkage methods**--including average linkage, Ward's method, complete linkage, and single linkage--can dramatically influence the final cluster structure. For example, Ward's method is optimized to minimize the increase in the total within-cluster sum of squares, typically producing compact, roughly spherical clusters. Conversely, single linkage, which uses the minimum distance between points, is notorious for creating elongated clusters due to the "chaining" effect. The analyst must carefully choose the most suitable method based on the assumed geometry of the underlying data.

Beyond the core clustering output, **PROC CLUSTER** generates indispensable diagnostic information essential for solution evaluation. This includes a detailed cluster history, which itemizes the sequence of mergers and the distance metrics recorded at each stage. Crucially, it also prepares the necessary data structure required for visualization tools, most notably the **dendrogram**. These outputs empower researchers not only to trace the formation of the clusters but also to make an informed, empirical decision regarding the optimal number of clusters to extract from the resulting hierarchy.

Case Study: Segmenting Basketball Player Performance Data

To provide a practical demonstration of **PROC CLUSTER**'s syntax and utility, we will analyze a hypothetical dataset comprising key performance indicators for 20 professional basketball players.

These indicators--average points, assists, and rebounds per game--will serve as the input variables for our analysis. Our goal is to apply hierarchical clustering to automatically segment these players into distinct statistical groups, allowing us to potentially identify and label different player archetypes, such as primary scorers versus specialized facilitators.

The initial phase requires defining and populating the sample data within the **SAS** environment. The following code snippet demonstrates the creation of the `my_data` dataset using the `DATALINES` statement. It is absolutely vital during this preparatory stage to ensure data integrity and correct input **variable** assignment, as the clustering procedure is highly sensitive to the scale and values of the metrics used. This setup ensures that the three numerical performance metrics are appropriately structured for the subsequent statistical analysis.

```
/* Create the input dataset containing player statistics */
data my_data;
input points assists rebounds;
datalines;
18 3 15
20 3 14
19 4 14
14 5 10
14 4 8
15 7 14
20 8 13
28 7 9
30 6 5
31 9 4
35 12 11
33 14 6
29 9 5
25 9 5
25 4 3
27 3 8
29 4 12
30 12 7
19 5 6
23 11 5
;
run;

/* Verify dataset structure */
```

```
proc print data=my_data;
run;
```

Following the initial data definition, the subsequent `PROC PRINT` step serves to confirm that the data has been correctly loaded into the **SAS** session, displaying the 20 observations and their associated statistical metrics. This verification step is fundamental before initiating the computationally intensive clustering process. The resulting table, visually represented below, confirms the integrity of the raw input data that will be processed by the clustering algorithm to determine the underlying player groupings.

Obs	player_ID	points	assists	rebounds
1	1	18	3	15
2	2	20	3	14
3	3	19	4	14
4	4	14	5	10
5	5	14	4	8
6	6	15	7	14
7	7	20	8	13
8	8	28	7	9
9	9	30	6	5
10	10	31	9	4
11	11	35	12	11
12	12	33	14	6
13	13	29	9	5
14	14	25	9	5
15	15	25	4	3
16	16	27	3	8
17	17	29	4	12
18	18	30	12	7
19	19	19	5	6
20	20	23	11	5

With the data successfully prepared, we proceed to execute the **PROC CLUSTER** statement. In this example, we employ the `METHOD=AVERAGE` option, which explicitly mandates the average **linkage method**. This method computes the distance between two clusters as the arithmetic mean of the distances between all possible pairs of observations, one drawn from each cluster. The `VAR` statement clearly specifies which variables--points, assists, and rebounds--should be utilized by

SAS to calculate the similarity metrics used in the clustering process.

```
/* Perform hierarchical clustering using the Average Linkage method */
proc cluster data=my_data method=average;
var points assists rebounds;
run;
```

Interpreting Cluster History and Visualizing the Dendrogram

The output generated by **PROC CLUSTER** provides critical data that must be scrutinized to validate the clustering solution. The primary tabular output, typically titled "Cluster History," is paramount for understanding the process. This detailed table meticulously tracks the agglomerative procedure step-by-step, recording the mergers from 20 initial individual clusters down to the final single cluster. For each step, it identifies the two clusters that were combined and lists the distance value (or similarity measure) at which the merge took place. Scrutinizing this history helps the analyst map the metric space of the dataset and quickly identify large increases in the merging distance, which often serve as strong indicators of natural boundaries between distinct cluster groups.

The image below provides a visual excerpt of this crucial tabular output. Observe how the table systematically documents the merging sequence. If two single observations merge at a very small distance, it signifies that they are statistically highly similar. Conversely, a merger occurring later in the process at a significantly larger distance suggests that the two groups being joined are fundamentally dissimilar, thus demarcating a strong cluster boundary.

The CLUSTER Procedure
Average Linkage Cluster Analysis

Eigenvalues of the Covariance Matrix				
	Eigenvalue	Difference	Proportion	Cumulative
1	51.0645891	40.0401485	0.7416	0.7416
2	11.0244406	4.2529440	0.1601	0.9017
3	6.7714966		0.0983	1.0000

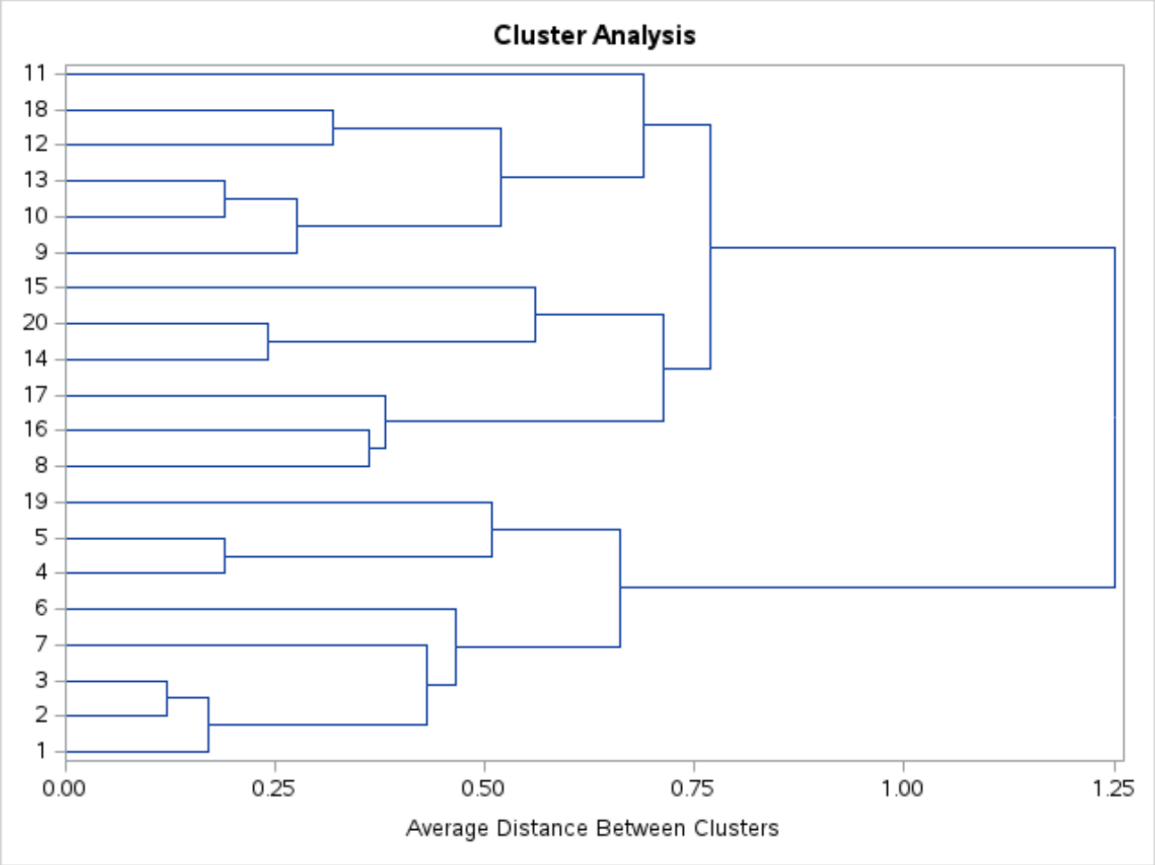
Root-Mean-Square Total-Sample Standard Deviation	4.790982

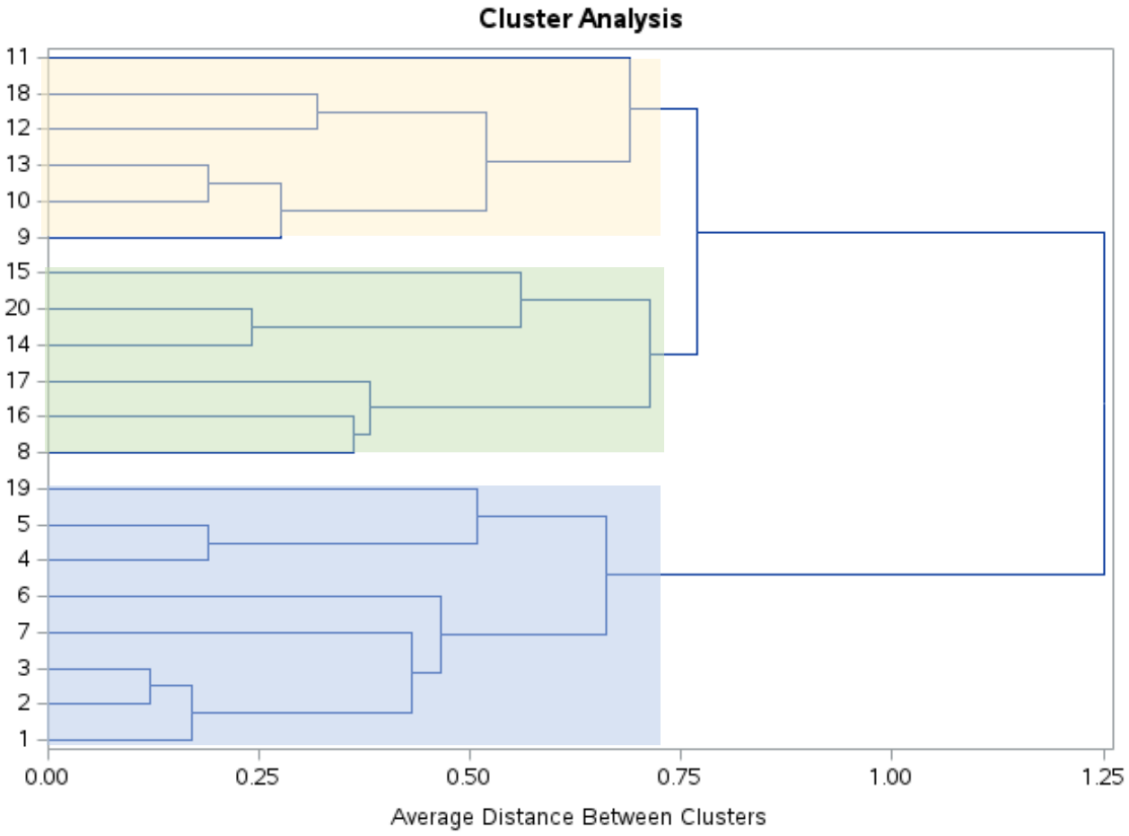
Root-Mean-Square Distance Between Observations	11.73546

Cluster History					
Number of Clusters	Clusters Joined		Freq	Norm RMS Distance	Tie
19	OB2	OB3	2	0.1205	
18	OB1	CL19	3	0.1704	
17	OB4	OB5	2	0.1905	T
16	OB10	OB13	2	0.1905	
15	OB14	OB20	2	0.241	
14	OB9	CL16	3	0.2761	
13	OB12	OB18	2	0.3188	T
12	OB8	OB16	2	0.3615	
11	CL12	OB17	3	0.3811	
10	CL18	OB7	4	0.4317	
9	CL10	OB6	5	0.4648	
8	CL17	OB19	3	0.5077	
7	CL14	CL13	5	0.5183	T
6	CL15	OB15	3	0.5588	
5	CL9	CL8	8	0.6615	
4	CL7	OB11	6	0.6881	
3	CL11	CL6	6	0.7124	
2	CL3	CL4	12	0.7677	
1	CL5	CL2	20	1.251	

The most intuitive and informative output for hierarchical clustering is the **dendrogram**. This specialized tree diagram visually summarizes the entire cluster formation process. The individual observations are displayed along one axis (often vertical), while the horizontal axis represents the distance or dissimilarity measure at which the merges occurred. The length of the vertical lines within the **dendrogram** is directly proportional to the distance between the clusters being joined; longer vertical lines denote greater dissimilarity and, consequently, stronger cluster boundaries.

Visual inspection of the **dendrogram** is the standard method for determining the optimal number of clusters for segmentation. In our basketball player example, by identifying substantial gaps between sequential merging points, we can pinpoint where the data structure naturally separates. The visual evidence provided in the subsequent images strongly suggests that the 20 players naturally coalesce into three distinct, internally consistent groups. Relying on this visual guidance is often more effective and intuitive than using purely numerical metrics for establishing the optimal cut-off point.





Finalizing Segmentation: Assigning Membership with PROC TREE

Once the optimal number of clusters (which we determined to be three) has been confidently confirmed via the visual analysis of the dendrogram, the next crucial step is the formal assignment of each original observation to its designated group. This is achieved in **SAS** using the **PROC TREE** procedure. **PROC TREE** takes the hierarchical output generated by **PROC CLUSTER** and allows the user to "cut" the tree at a specified level, thereby creating a fixed, non-hierarchical set of final clusters.

The following code snippet illustrates the syntax required for implementing **PROC TREE**. The core instruction is the `NCL=3` option, which formally partitions the player data into the three groups identified during our visual inspection. The `OUT=clusts` option is essential, as it instructs **SAS** to generate a new dataset named `clusts`. This output dataset includes the original performance statistics supplemented by a new categorical **variable** that clearly indicates the cluster membership for every player.

/* Assign each observation to one of three final clusters */
proc tree data=clustd noprint ncl=3 out=clusts;
copy points assists rebounds;

```
id player_ID;
run;
proc sort;
by cluster;
run;

/* View the final cluster assignments */
proc print data=clusts;
id player_ID;
run;
```

The `COPY` statement within **PROC TREE** ensures that the original numerical attributes are carried forward alongside the new cluster designation, which is absolutely vital for subsequent profiling of the segments. The subsequent `PROC SORT` organizes the output based on the new `cluster` variable, dramatically improving readability and facilitating the comparative analysis of the resulting segments. The final `PROC PRINT` displays the clustered dataset, providing the tangible result of the segmentation effort.

player_ID	points	assists	rebounds	CLUSTER	CLUSNAME
2	20	3	14	1	CL5
3	19	4	14	1	CL5
1	18	3	15	1	CL5
4	14	5	10	1	CL5
5	14	4	8	1	CL5
7	20	8	13	1	CL5
6	15	7	14	1	CL5
19	19	5	6	1	CL5
10	31	9	4	2	CL4
13	29	9	5	2	CL4
9	30	6	5	2	CL4
12	33	14	6	2	CL4
18	30	12	7	2	CL4
11	35	12	11	2	CL4
14	25	9	5	3	CL3
20	23	11	5	3	CL3
8	28	7	9	3	CL3
16	27	3	8	3	CL3
17	29	4	12	3	CL3
15	25	4	3	3	CL3

As clearly illustrated in the final output table, the players have been successfully categorized into three distinct groups. For instance, players 2, 3, 1, 4, 5, 7, 6, and 19 are all unified within Cluster 1. This grouping strongly suggests that these eight players share a statistically similar profile across their performance metrics (points, assists, and rebounds), enabling the analyst to profile Cluster 1 as a specific player archetype within the league, distinct in characteristics from players categorized in Cluster 2 or Cluster 3.

Post-Clustering Analysis and Assessing Solution Robustness

While obtaining cluster assignments via **PROC TREE** completes the mechanical execution of the hierarchical analysis, the most crucial phase of the workflow is yet to come: profiling the derived groups to understand their defining characteristics. Utilizing other analytical procedures in **SAS**, such as `PROC MEANS`, `PROC FREQ`, or `PROC UNIVARIATE`--with the new cluster assignment serving as the grouping variable--allows the analyst to calculate descriptive statistics (e.g., mean points, median assists) for each cluster. This statistical summary is what transforms abstract cluster numbers into interpretable segments, facilitating meaningful labels like "High

Rebounders/Moderate Scorers" for Cluster 1 and "Elite Scorers/High Facilitators" for Cluster 3.

A vital element of sound hierarchical clustering practice is evaluating the sensitivity of the solution to the chosen algorithmic parameters, specifically the **linkage method**. Although we successfully employed the `AVERAGE` **linkage method** for the basketball data, **PROC CLUSTER** supports numerous distinct alternatives, including `WARD`, `SINGLE`, `COMPLETE`, and `CENTROID`. Because these methods rely on fundamentally different mathematical criteria for calculating inter-cluster distance, a solution derived using Ward's method (which minimizes total variance) will likely differ significantly from one derived using Single Linkage (which prioritizes minimum proximity).

To establish the robustness and stability of the findings, advanced practitioners routinely perform sensitivity analysis by running the clustering procedure using two or three distinct **linkage methods**. If the core structural elements and the primary cluster boundaries remain largely consistent across these different methodological approaches, confidence in the derived cluster solution is substantially reinforced. Conversely, a highly unstable solution--where cluster membership shifts wildly between methods--suggests that the inherent structure of the dataset may not be strongly defined or that the underlying groupings are too subtle to be resolved unambiguously by hierarchical techniques.

Conclusion and Next Steps for SAS Data Analysis

Proficiency in utilizing **PROC CLUSTER** and its indispensable companion procedure, **PROC TREE**, equips data analysts with essential tools for conducting rigorous unsupervised segmentation within the **SAS** environment. By diligently selecting the appropriate linkage method, critically interpreting the cluster history, and leveraging the powerful visual guidance of the dendrogram, analysts can successfully extract meaningful, actionable segments from complex raw data. This crucial capability to discover inherent structure is invaluable across a wide range of disciplines, from optimizing corporate business strategy to facilitating scientific research breakthroughs.

For those seeking to delve deeper into data analysis techniques and expand their expertise within the **SAS** software environment, the following tutorials offer guidance on other common and advanced statistical tasks:

A comprehensive tutorial on performing regression analysis in SAS.
An in-depth guide to conducting Analysis of Variance (ANOVA) in SAS.
Detailed instructions for creating custom reports and formatted tables using specialized SAS procedures.
An introductory guide to time series analysis techniques with SAS.