Lesson 30: Hierarchical Clustering

Data Science with Python – BSc Course

Data Science Program

BSc Course

45 Minutes

Previously on L29: K-Means Clustering

What We Learned:

- K-Means partitions data into K groups by minimizing variance
- Elbow method and silhouette score help choose K
- Works well for spherical, well-separated clusters

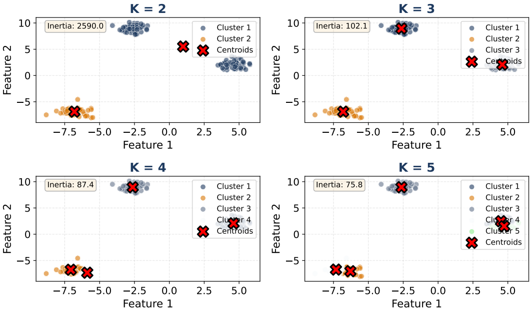
K-Means Groups → **Elbow + Silhouette** → **But must choose K!**

K-Means requires specifying K upfront – what if we don't know the right number?

The K Problem

What If K=3 and K=5 Both Look Reasonable?

- K-Means gives one partition – no relationships between clusters
- Elbow and silhouette sometimes disagree on the best K



We need a method that shows the FULL hierarchy – not just one fixed partition

Learning Objectives

The Problem: K-Means requires choosing K upfront. What if we want to see the full hierarchy of cluster relationships at all levels?

After this lesson, you will be able to:

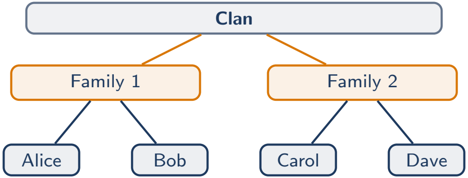
- Build and interpret dendrograms (cluster family trees)
- Choose between linkage methods (single, complete, Ward)
- Cut dendrograms at any height to obtain flat clusters
- Apply hierarchical clustering to portfolio construction (HRP)

Finance Application: Hierarchical Risk Parity (HRP) portfolio optimization

The Family Tree Idea

Analogy: Building a Family Tree Bottom-Up

- Start with individuals (data points)
- Merge closest pairs into families, families into clans, clans into nations
- The tree records every merge – you can cut at any level



Bottom-up: merge closest pairs

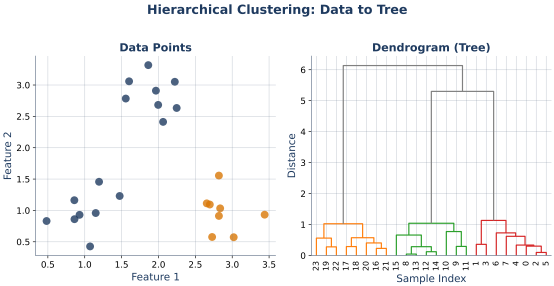
This is hierarchical clustering: build the full tree, then cut where you want.

Dendrogram = the family tree of your data. Cut at any level to get clusters.

Hierarchical Clustering Concept

Building a Tree of Clusters

- Agglomerative: start with N clusters, merge closest pairs
- Each point begins as its own cluster (N clusters initially)
- Result: nested hierarchy showing relationships at all scales

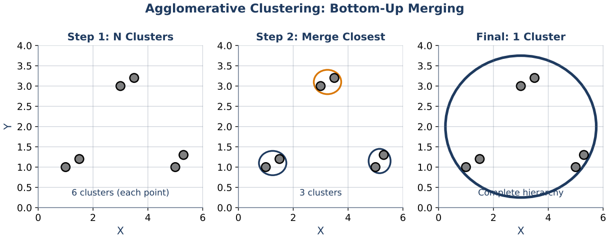


Advantage over K-Means: no need to specify K in advance

Agglomerative Steps

Bottom-Up Merging Process

- **Step 1:** Each point = 1 cluster (N clusters initially)
- **Step 2:** Find 2 closest clusters and merge (N-1 remain)
- **Step 3:** Repeat: find next closest pair and merge
- Continue until only 1 cluster remains



Agglomerative = "bottom-up": starts with N clusters, ends with 1

How to Measure Distance Between Clusters?

Problem: We merge closest clusters – but how do we define "closest"?

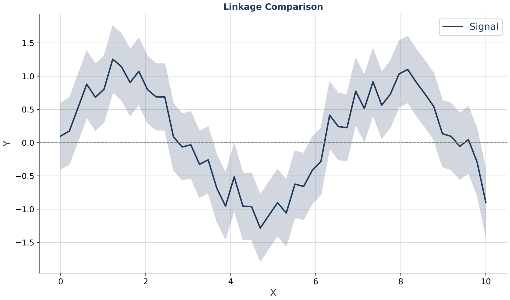
Four Common Linkage Methods:

1. **Single:** Min distance between any pair – finds chains
2. **Complete:** Max distance between any pair – compact clusters
3. **Average:** Mean distance between all pairs – compromise
4. **Ward:** Minimize variance increase when merging – balanced, most popular

---

Ward is recommended as default – produces balanced, intuitive clusters

Linkage Methods Compared

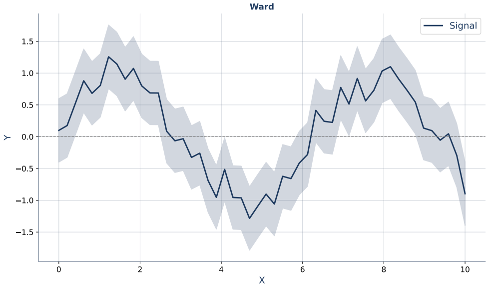


Single: chains. Complete: compact. Ward: balanced. Average: compromise.

Ward Linkage in Detail

How Ward Measures Cluster Distance

- Merges the pair that increases total variance least
- Produces compact, similarly-sized clusters
- Default choice for most applications

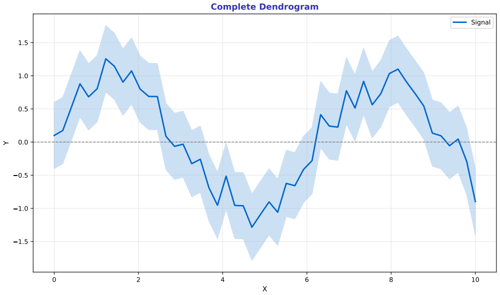


Ward minimizes within-cluster variance – the default choice for most problems

The Dendrogram: Reading the Family Tree

Dendrogram = Tree Diagram of Cluster Merges

- Y-axis: distance (height) at which clusters merge
- X-axis: individual observations (leaves)
- Low merge = very similar; high merge = very different

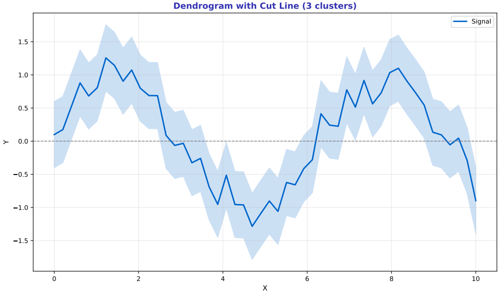


Read bottom-up: similar items merge early (low), different items merge late (high)

Cutting the Tree

From Hierarchy to Flat Clusters

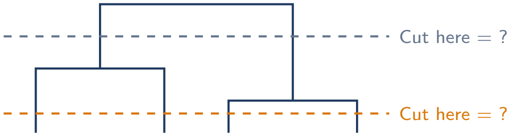
- Draw a horizontal line at height h
- Count branches crossing the line = number of clusters
- High cut = few large clusters; low cut = many small clusters



Look for large vertical gaps – they suggest natural cluster boundaries

Checkpoint: Can You Read a Dendrogram?

Quick Quiz: Look at this dendrogram sketch.



Which cut gives 3 clusters?

Think About:

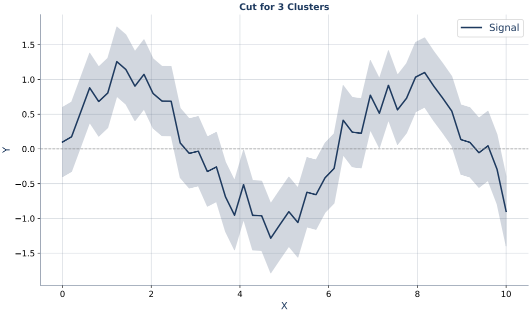
- How many branches does the lower (amber) line cross?
- How many branches does the upper (gray) line cross?

Answer: lower cut crosses 3 branches = 3 clusters; upper cut crosses 2 = 2 clusters

Different Cuts, Different Clusterings

One Dendrogram, Many Possible Partitions

- Unlike K-Means, you build the tree once and explore multiple K values
- Cut high: broad groupings (sectors). Cut low: fine-grained (sub-sectors)



The dendrogram encodes ALL possible clusterings – just move the cut line

K-Means vs Hierarchical Clustering

Key Difference: Single Cut vs Full Tree

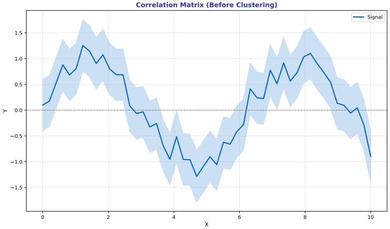
- **K-Means:** Specify K, get one flat partition
- **Hierarchical:** Build full tree, cut at any level
- Hierarchical shows relationships BETWEEN clusters (K-Means doesn't)

Hierarchical reveals structure at all scales; K-Means gives one fixed partition

Correlation-Based Clustering

Finance: Cluster Assets by Return Correlation

- Convert correlation to distance: $d = 1 - \rho$ (or $\sqrt{2(1-\rho)}$)
- Highly correlated assets ($\rho \approx 1$) have small distance
- Raw correlation matrix – can you spot the clusters?

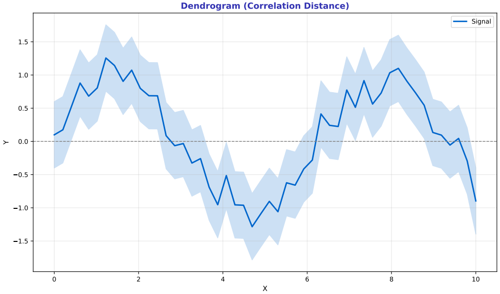


Raw correlation matrix shows pairwise relationships – hard to see structure

Dendrogram from Correlation Distances

Hierarchical Clustering Reveals Hidden Structure

- Apply linkage to the correlation distance matrix
- Dendrogram groups correlated assets into clusters
- The tree tells us which assets move together

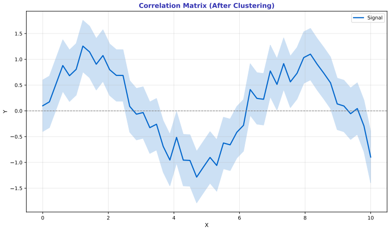


Hierarchical clustering groups correlated assets together

Reordered Correlation Matrix

Same Data, Reordered by Dendrogram

- Rows and columns reordered to match cluster hierarchy
- Block-diagonal structure reveals natural asset groups

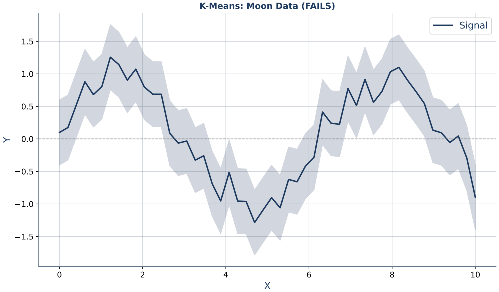


Reordered by dendrogram – blocks along diagonal reveal cluster structure

When K-Means Fails: Non-Spherical Data

Moon-Shaped Clusters

- K-Means assumes spherical clusters – fails on crescents
- Centroids land in wrong positions, splitting each crescent
- Hierarchical (single linkage) can follow elongated shapes

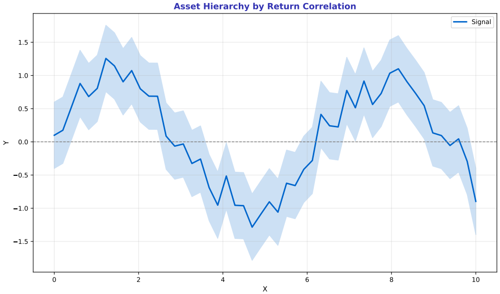


K-Means fails on non-spherical shapes – use hierarchical or DBSCAN instead

Finance: Hierarchical Risk Parity (HRP)

Modern Portfolio Construction Using Clustering

- Cluster assets by correlation hierarchy
- Allocate risk inversely to cluster variance
- More stable weights than mean-variance optimization



HRP (Lopez de Prado, 2016): more robust portfolio construction using hierarchical clustering

When to Use Which Method?

Decision Framework

Criterion	K-Means	Hierarchical
Need to choose K?	Yes, upfront	No – cut tree later
Speed	Fast ($O(nKt)$)	Slow ($O(n^3)$)
Cluster shape	Spherical only	Any shape (linkage)
Interpretability	Centroids	Dendrogram (tree)
Relationships	None	Full hierarchy
Large datasets	Good ($>$10K)	Impractical ($>$10K)

Rules of Thumb:

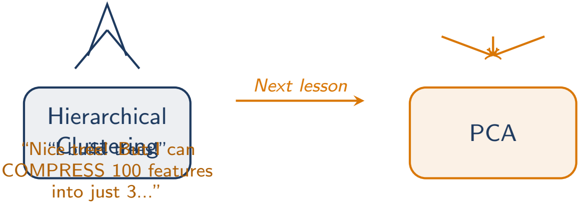
- Small data + need hierarchy → hierarchical (Ward)
- Large data + known K → K-Means
- Non-spherical shapes → hierarchical (single) or DBSCAN

Finance: Use hierarchical for asset clustering (N ¡ 500), K-Means for customer segmentation

What's Next: Compression with PCA

From Grouping to Compression

We can now group similar items (K-Means, hierarchical). But what about **reducing dimensions**?



L31 Preview: PCA finds directions of maximum variance to reduce dimensions while preserving as much information as possible.

Next lesson: PCA – reducing 100 features to 3 while keeping 95% of the information

Hands-On Exercise (25 min)

Task: Build an Asset Hierarchy

1. Calculate correlation matrix for 20 stocks (1 year daily returns)
2. Convert to distance matrix: $d = \sqrt{2(1-\rho)}$
3. Build dendrogram with Ward linkage
4. Cut at 2–3 different heights – compare resulting clusters
5. Label clusters by dominant sector

Starter Code:

- `Z = linkage(squareform(dist), method='ward')`
- `dendrogram(Z, labels=ticker_names)`
- `labels = fcluster(Z, t=3, criterion='maxclust')`

Deliverable: Dendrogram with cluster cut lines annotated.

Extension: Implement simple HRP allocation based on your clusters

Lesson Summary

Problem Solved: We can now discover hierarchical relationships and create clusters at any granularity.

Key Takeaways:

- **Dendrogram** = family tree of your data (bottom-up merging)
- **Linkage** method matters: Ward for balanced, single for chains
- **Cut** the dendrogram at desired height for flat clusters
- **Finance:** correlation-based clustering powers HRP portfolios

Next Lesson: PCA (L31) – reducing dimensions while preserving information

---

Memory: Dendrogram = tree. Cut horizontally to get clusters. Ward = balanced.