

Unsupervised Learning and Patient Segmentation

Study Guide

by Richard J. Young, Ph.D.

richard@deepneuro.ai

Chapter 9 (enrichment reading) of "Healthcare Analytics and AI: Building Systems That Actually Work"
The essentials: what to know, why it matters at the bedside, and how models fail.

One-sentence thesis. Unsupervised learning does not discover truth automatically; it discovers structure, and your job is deciding whether that structure is biological, operational, or bias.

PART 1: Seeing High-Dimensional Patients

Dimensionality Reduction: PCA, t-SNE, UMAP

A single patient may carry a feature vector of 2,000-plus dimensions. No one can see that space, so we compress it while trying to preserve meaningful structure. **PCA** rotates the data to the directions of greatest variance and is linear, deterministic, and interpretable; it reveals correlation structure (liver enzymes load onto one "liver health" component) and is reproducible, which matters for audit and regulation. **t-SNE** and **UMAP** are nonlinear methods that preserve local neighborhoods and produce the compelling cluster plots you see in papers, and UMAP is the visualization standard in single-cell genomics and EHR phenotyping.

ANALOGY

A t-SNE plot is a subway map, not a road atlas: it tells you which stations are neighbors, but the distances between clusters are not to scale and should not be read literally. UMAP is closer to a simplified city map, still stylized but more faithful to the large-scale layout.

Watch out. t-SNE is stochastic (different seeds, different pictures) and its perplexity setting can fragment or merge real clusters, so only report patterns stable across settings. The recommended workflow is PCA first for variance and noise reduction, UMAP for the eyes only, and clustering in the PCA-reduced space, never in the UMAP embedding.

Method	Linear?	Preserves	Reproducible	Typical EHR use
PCA	Linear	Global variance	Yes	Variance analysis; audit-ready pipelines
t-SNE	Nonlinear	Local neighborhoods	No	Visualization only; distances not meaningful
UMAP	Nonlinear	Local + some global	Mostly (fixed seed)	Phenotyping visualization standard

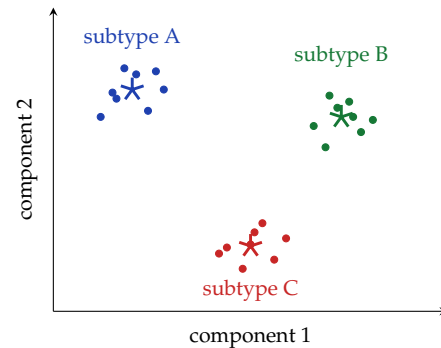
K-Means, Hierarchical, DBSCAN, and GMMs

Clustering partitions patients into groups that are more similar within than between; in healthcare these groups are read as **phenotypes**. The algorithms differ mainly in the shape of cluster they look for and how firmly they commit each patient to one. **K-Means** minimizes within-cluster squared distance to k centroids, like placing k magnets on a table of metal filings until the pull stabilizes; in symbols it seeks the assignment that minimizes that total distance,

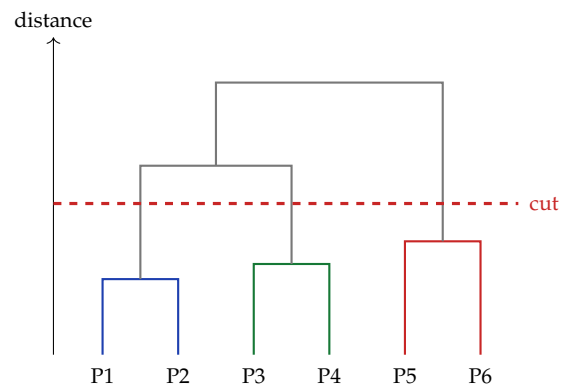
$$\arg \min_C \sum_{k=1}^K \sum_{\mathbf{x} \in C_k} \|\mathbf{x} - \mu_k\|^2$$

Hierarchical clustering builds a dendrogram you cut at a chosen height. **DBSCAN** finds dense regions and explicitly labels sparse points as noise, asking of each point “are you standing in a crowd or by yourself?” **Gaussian Mixture Models (GMMs)** give each patient a soft membership probability across clusters. That soft membership matters because patients rarely belong cleanly to one phenotype: a GMM can tell a clinician “this patient is 55 percent consistent with the insulin-resistant subtype and 40 percent with the obesity-related one, so monitor for both complication profiles,” preserving ambiguity that a hard label destroys. Remember that K-Means forces spherical, equal-size clusters and is dragged by outliers, and hierarchical clustering is memory-bound to roughly 20,000-50,000 patients, so run several algorithms and triangulate: agreement raises confidence,

and disagreement flags genuinely ambiguous boundary patients you should not paper over.



Three patient clusters after dimensionality reduction, each with its centroid (the large *). K-Means assigns every point to its nearest centroid; the stars are the means the algorithm iterates toward.



cutting the tree at this height yields 3 phenotype clusters

A hierarchical-clustering dendrogram. Merges join the closest groups at increasing distance; the height of a horizontal cut sets the number of clusters — here three.

Algorithm	Shape assumed	Handles noise	Assignment
K-Means	Spherical, equal size	No	Hard
Hierarchical (Ward)	None	No	Hard; multi-resolution
DBSCAN	Arbitrary	Yes (noise label)	Hard + noise
GMM	Ellipsoidal	Soft membership absorbs it	Soft (probabilities)

Computational Phenotyping

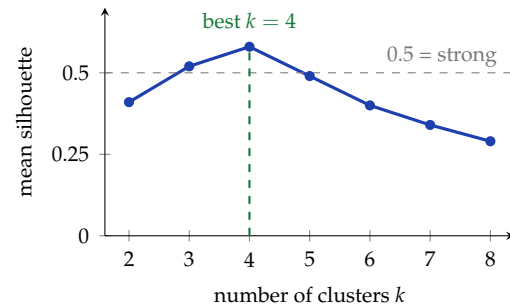
Feeding hundreds of features to a clustering algorithm can reveal disease subtypes no human specified. A randomized trial reports that a drug “works on average”; clustering is the magnifying glass that shows it works brilliantly for one subtype and not at all for another, unmasking heterogeneity the average masks. The Ahlqvist diabetes study clustered six variables into five subtypes (SAID, SIDD, SIRD, MORD, MARD) with distinct complication profiles and treatment responses, replicated across Sweden, China, India, and Ghana and confirmed in a 2025 long-term follow-up. The clinical payoff is concrete: in the HARP-2 trial, simvastatin improved survival for hyper-inflammatory ARDS patients but did nothing for hypo-inflammatory ones, and phenotype-aware treatment is now moving into prospective trial design (the PRECISE sepsis trial, for example, assigns phenotype in real time). Handle these subtypes with care: in stigmatized conditions like addiction and mental health, algorithmic subtypes can harden into labels driven by demographics rather than clinical need, so always ask who benefits from a classification and who is harmed by it.

Phenotype Stability: Is the Cluster Clinically Real?

Clustering algorithms *always* return clusters, even on pure noise, so the burden of proof is on the analyst, and three tests carry it. The first is geometric: does each patient actually sit closer to its own cluster than to any other? The **silhouette score** captures exactly that, comparing how close a patient sits to its own cluster, $a(i)$, against its distance to the nearest rival cluster, $b(i)$:

$$s(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))}$$

A score near 1 means clean separation, near 0 means the patient sits on a boundary. But geometry is not clinical truth, so two harder tests follow. **Bootstrap re-sampling** measures whether clusters reappear under perturbation via the Adjusted Rand Index (ARI); a real phenotype should survive being jostled, so shake the dataset, recluster, and see if the same groups fall back out. **External validation** asks whether cluster membership predicts an outcome that was never a clustering input, and this is the definitive test, because it is clinical, not geometric: if clusters built from labs and utilization predict mortality or readmission they never saw, the structure is real, whereas a clustering that merely separates patients by age and sex tells clinicians nothing new. **Watch out.** Consensus clustering minimizes overfitting but does not eliminate it; it can carve stable-looking clusters out of random unimodal data, so external clinical validation remains indispensable.



Choosing k by silhouette. The score peaks at $k = 4$ and sits above the 0.5 “strong structure” line; more clusters past the peak only fragment real groups.

Metric	Threshold	Interpretation
Silhouette	> 0.5	Strong structure
Silhouette	$0.25\text{--}0.5$	Overlapping but distinguishable
Silhouette	< 0.25	Likely artifacts; caution
Bootstrap ARI	> 0.8	Highly stable
Bootstrap ARI	< 0.6	Not robust; do not deploy

The Proxy Trap (The Optum Algorithm)

A commercial algorithm serving roughly 200 million patients a year decided who qualified for high-risk care management. Its architects chose an elegant-looking target: predict future healthcare **cost**, on the theory that costlier patients need more care. The model never saw race and was, by convention, “race-blind.” It still produced a racist outcome. Because Black patients in the US receive about **\$1,800** less in annual spending than white patients at equal illness (from coverage gaps, access barriers, and referral bias), the algorithm read lower spending as lower need; at any given risk score, Black patients were actually sicker.

ANALOGY

Using cost as a proxy for need is like judging thirst by how much water someone was given, in a system that hands some people a smaller cup.

Correcting the target would have raised the share of Black patients flagged from 17.7 percent to **46.5 percent**, affecting an estimated 47,000 patients a year. The lesson is that “race-blind” is not race-neutral: any feature correlated with race (zip code, insurance type, prior spending, visit counts) becomes a proxy, and retraining on direct clinical indicators (chronic condition counts, lab values, validated comorbidity in-

dices) cut the disparity by **84 percent**. **Watch out.** Risk stratification creates a feedback loop: patients flagged high-risk get more care, generate more data, and are confirmed high-need next cycle, while patients wrongly flagged low-risk fade from the data. Audit cluster demographics at every pipeline stage, use SHAP (Chapter 8) to expose which features drive assignments, and bring clinicians and community representatives into defining what the clusters mean.

Should You Build This?

Build it when the population is genuinely heterogeneous and a phenotype would change what you do, as when clustering 82,577 opioid-related emergency department visits surfaced six phenotypes with distinct treatment patterns, and when you are prepared to prove stability with bootstrap resampling, consensus clustering, and external clinical validation before anyone acts on a cluster label. **Think twice when** the target or the features are proxies contaminated by unequal access to care: Optum’s race-blind, cost-as-need algorithm systematically under-flagged Black patients, and K-Means will happily partition pure noise into confident clusters. **Who bears the risk:** patients on the wrong side of a cluster boundary, since those misclassified as low-risk lose care management and then generate the very data that confirms the error, while patients in stigmatized clusters carry algorithmic labels that resist nuance.

Quick Self-Test

1. Why should you cluster in the PCA-reduced space rather than in the UMAP embedding, and what is UMAP actually for?
2. A GMM reports a patient is 55 percent cluster A and 40 percent cluster B. Why can that be more clinically useful than a hard K-Means label?
3. Your clusters have a silhouette of 0.6 but do not predict any outcome they were not trained on. Are they clinically meaningful? Explain.
4. What does bootstrap ARI test that a silhouette score does not?
5. The Optum algorithm never used race. Why did it still produce a racist outcome, and what single design change fixed most of it?
6. In one sentence, why does the phrase “the algorithm found clusters” prove almost nothing?

If You Remember One Thing

Running the clustering is the easy part; deciding whether the clusters are real, useful, and fair is the whole job. A phenotype earns a place in a clinical workflow only after it survives stability testing and predicts outcomes it never trained on, and only after you have asked who the segmentation excludes. The Optum model was race-blind, technically clean, and profoundly biased, because it optimized a proxy no one audited.

Richard J. Young, Ph.D. · richard@deepneuro.ai · Senior AI Research Scientist, UnitedHealth Group · UNLV Lee Business School · ORCID 0000-0002-1109-7552