Deep Unsupervised Learning (Overview)

Naeemullah Khan
naeemullah.khan@kaust.edu.sa

KAUST Academy
King Abdullah University of Science and Technology

May 21, 2025

Table of Contents

Unsupervised Learning - Definition

- We have a dataset without labels. Out goal is to learn something interesting about the underlying structure of the data:
  - Clusters hidden in the dataset.
  - Outliers: particularly unusual and/or interesting data points.
  - Useful signals hidden in the noise, e.g., human speech over a noisy background.

Components of Unsupervised Learning

- **Data**: Unlabeled data, e.g., images, text, or sensor readings.
- **Model**: A mathematical representation of the data, e.g., a mixture model or a neural network.
- **Objective function**: A measure of how well the model fits the data, e.g., likelihood or reconstruction error.
- **Optimization algorithm**: An algorithm to minimize the objective function, e.g., gradient descent or expectation-maximization.
- **Evaluation metrics**: Measures to assess the quality of the learned model, e.g., silhouette score or clustering accuracy.
- **Applications**: Use cases for unsupervised learning, e.g., clustering, dimensionality reduction, or anomaly detection.

Supervised vs Unsupervised Learning

Aspect	**Supervised Learning**	**Unsupervised Learning**
Objective	Learn a function f from labeled input–output pairs.	Discover structure or representations in unlabeled data.
Evaluation	Accuracy, precision/recall on held-out labels.	Clustering validity indices (e.g. silhouette), reconstruction error.
Cost	Methods range from $\mathcal{O}(n)$ to $\mathcal{O}(n^3)$ per fit.	k-means $\mathcal{O}(nkd)$, hierarchical $\mathcal{O}(n^2)$, PCA $\mathcal{O}(nd^2)$.
Labels/Clusters	Fixed, known set of classes.	Number of clusters unknown; must be chosen or inferred.
Output	Classifier or regressor for new inputs.	Cluster assignments, embeddings, density models, or generative samples.

Table 1: Key differences between Supervised and Unsupervised Learning

Unsupervised Learning - Applications

Unsupervised learning is used in various fields and applications, including:

- **Visualisation**: Identifying and making accessiblge useful hidden structures in the data.
- **Anomaly Detection**: Identifying factory components that are likely to break soon.
- **Signal denoising**: Extracting human speech from a noisy recording.
- **Generative Models**: Learning to generate new data points similar to the training data.
- **Feature Learning**: Automatically discovering useful representations of the data.
- **Data Preprocessing**: Cleaning and transforming data for better performance in supervised learning tasks.

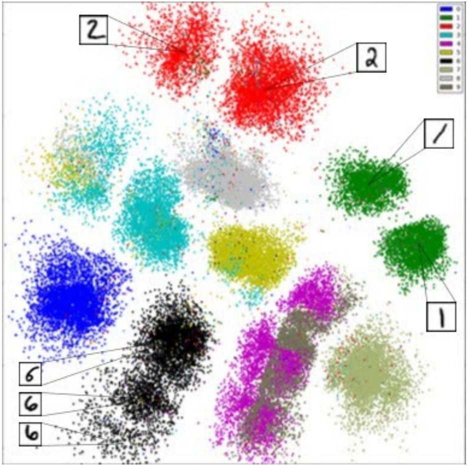

Figure 2: Unsupervised learning can discover structure in digits without any labels.

Application: DNA Analysis

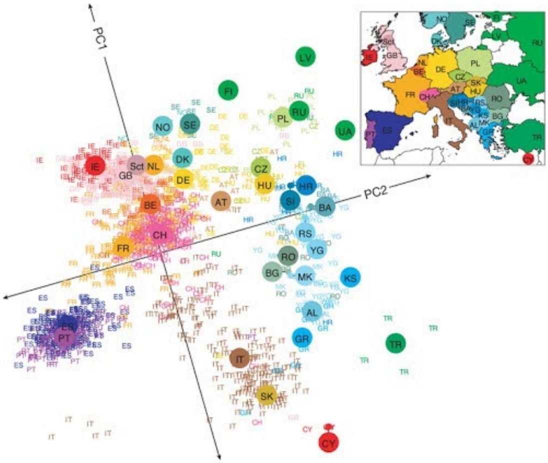

Figure 3: Dimensionality reduction applied to DNA reveal the geography of European countries.

What is Deep Unsupervised Learning?

What is Deep Unsupervised Learning? (cont.)

- Capturing rich patterns in raw data with deep networks in a label-free way.

What is Deep Unsupervised Learning? (cont.)

- Capturing rich patterns in raw data with deep networks in a label-free way.
  - **Generative Models**: Recreate raw data distribution.

Unsupervised Learning - Challenges

Why is unsupervised learning challenging?

- **Exploratory data analysis**: Unsupervised learning is often used for exploratory data analysis, where the goal is to discover patterns or structures in the data without any prior knowledge of the labels.
- **Difficult to assess performance**: Evaluating the performance of unsupervised learning algorithms can be challenging, as there are no ground truth labels to compare against ("right answer" unknown).
- **Sensitivity to noise**: Unsupervised learning algorithms can be sensitive to noise and outliers in the data, which can lead to misleading results.
- **Curse of dimensionality**: As the number of features increases, the data becomes sparse, making it difficult to find meaningful patterns.

Unsupervised Learning - Types

- **Cluster Analysis**:
  - For identifying homogenous subgroups of samples.
  - **Examples**: K-means, hierarchical clustering, DBSCAN.
- **Dimensionality Reduction**:
  - For finding a low-dimensional representation to characterize and visualize the data.
  - Reducing the number of features in a dataset while preserving important information.
  - **Examples**: PCA, t-SNE, UMAP.
- **Anomaly Detection**:
  - **Finding outliers in the dataset**: Identifying unusual (rare items, events, or observations) data points that do not conform to expected patterns.
  - **Examples**: Isolation Forest, One-Class SVM, Autoencoders.

Clustering

A set of methods for finding subgroups within the dataset.

- Observations should share common characteristics within the same group, but differ across groups.
- Groupings are determined from attributes of the data itself — differs from classification.

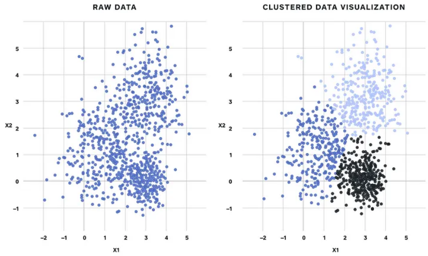

Figure 4: Taking a 2 dimensional dataset and separating it into 3 distinct clusters. [Source]

Clustering (cont.)

Input: Dataset $D = \{x_1, x_2, \ldots, x_n\}$, number of clusters k
Output: Cluster assignments for each data point

Initialization: Randomly initialize k cluster centroids or seeds;

repeat

> **Assignment Step:** Assign each data point x_i to the nearest cluster based on a distance metric;
>
> **Update Step:** Recompute cluster centroids using current assignments;

until *convergence or maximum iterations reached*;

return Final cluster assignments;

Algorithm: Generic Clustering Algorithm

Clustering Vs Classification

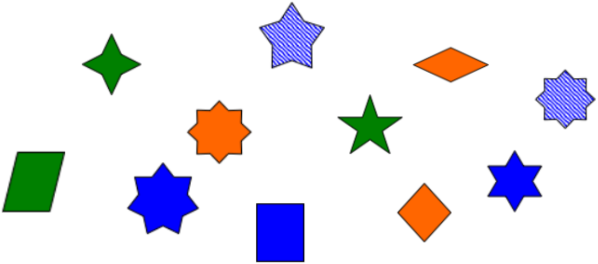

Figure 5: Sample data points.

Classification
- Labels available
- Assigning to known classes
- Supervised

Clustering
- No labels
- Grouping based on similarity
- Unsupervised

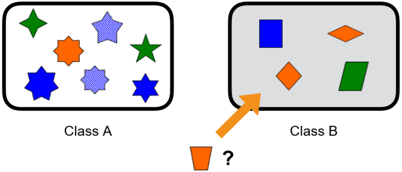

Figure 6: Classification result.

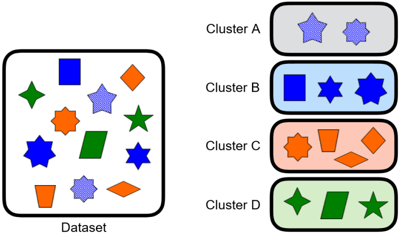

Figure 7: Clustering result.

- **Centroid-Based Clustering**: Groups data points based on their proximity to a central point, such as K-means or K-medoids.
- **Hierarchical Clustering**: Builds a hierarchy of clusters using either agglomerative (bottom-up) or divisive (top-down) approaches.
- **Model-Based Clustering**:
  - Each cluster is represented by a parametric distribution.
  - Dataset is a mixture of distributions.
  - Assumes a probabilistic model for the data and uses statistical methods to identify clusters, such as Gaussian Mixture Models (GMM).
- **Hard Clustering**:
  - Each data point is assigned exclusively to exactly one cluster.
  - **Example algorithms**: K-means, Hierarchical clustering.

- **interpretation**: No ambiguity — clusters are crisp and non-overlapping.

- **Soft/Fuzzy Clustering**:
  - Each data point can belong to multiple clusters simultaneously with varying degrees of membership (probabilities or weights).
  - **Example algorithms**: Gaussian Mixture Models (GMM), Fuzzy C-means.
  - **interpretation**: Reflects uncertainty or mixed membership — clusters can overlap.