1. **What is the difference between supervised and unsupervised learning? Give some examples to illustrate your point.**

**- Supervised Learning:**

In supervised learning, the algorithm is trained on a labeled dataset where the correct output is provided.

It learns to map input data to the correct output and makes predictions based on this learned mapping.

Examples:

Image classification (labeling images as "cat" or "dog").

Spam email detection (classifying emails as "spam" or "not spam").

**Unsupervised Learning:**

In unsupervised learning, the algorithm is given data without explicit labels or outputs.

It identifies patterns, structures, or relationships in the data without predefined categories.

Examples:

Clustering: Grouping similar customers for targeted marketing.

Dimensionality reduction: Reducing the number of features for visualization or data compression.

1. **Mention a few unsupervised learning applications.**

**-** Clustering: Identifying customer segments for marketing.

Dimensionality Reduction: Reducing features for data visualization.

Anomaly Detection: Identifying unusual patterns in data (e.g., fraud detection).

Topic Modeling: Discovering themes in text data (e.g., document clustering).

Generative Modeling: Creating new data (e.g., image or text generation).

1. **Explain how the k-means algorithm determines the consistency of clustering.**

* The k-means algorithm determines the consistency of clustering by minimizing the within-cluster variance. It calculates the distance between data points and the centroid of their assigned cluster and aims to make this distance as small as possible. When the distances are minimized, it indicates that the data points within a cluster are closer to each other and form a more consistent group. K-means iteratively refines the cluster assignments and centroid positions to achieve this consistency. The overall consistency is assessed by the sum of squared distances within clusters, and a lower value indicates more consistent clustering.

1. **With a simple illustration, explain the key difference between the k-means and k-medoids algorithms.**

* The key difference between the k-means and k-medoids algorithms is how they choose cluster representatives:
* K-Means: It uses the mean (average) of data points within a cluster as the cluster center, represented by a centroid. The centroid may not necessarily be one of the data points.
* K-Medoids: It selects one of the actual data points within the cluster as the cluster center (medoid). The medoid is the most central and representative data point in terms of minimizing dissimilarity to other points in the cluster.

1. **What is a dendrogram, and how does it work? Explain how to do it.**

* A dendrogram is a tree-like diagram used in hierarchical clustering to visualize the arrangement of data points into clusters. Here's how it works:
* Data Points: Begin with a set of data points, each representing an object or observation.
* Pairwise Distances: Calculate the pairwise distances between all data points. These distances can be based on various metrics, such as Euclidean distance or correlation.
* Agglomerative Clustering: Start with each data point as its own cluster. Then, iteratively merge the two closest clusters into a single cluster based on the computed distances. This process continues until all data points belong to a single cluster, and the hierarchy is formed.
* Dendrogram: As clusters merge, the dendrogram is constructed. It's a tree-like structure where the height of each fusion (the vertical lines connecting clusters) represents the dissimilarity between the merged clusters. Taller fusions indicate greater dissimilarity.
* Cutting the Dendrogram: To form clusters at different levels of granularity, you can cut the dendrogram horizontally at a chosen height. Clusters are then identified based on the branches cut, with the height of the cut influencing cluster size.

1. **What exactly is SSE? What role does it play in the k-means algorithm?**

* SSE (Sum of Squared Errors): SSE is a measure of the within-cluster variance or the sum of the squared distances between data points within each cluster and their respective cluster center (centroid). In the context of k-means:
* SSE quantifies how much data points within a cluster deviate from their cluster's center.
* Lower SSE indicates that data points in each cluster are closer to their respective centroids, indicating tighter and more consistent clustering.