ISHAAN JAIN

Q1. Supervised V/S Unsupervised

Supervised learning uses labeled data where each input has a corresponding output, and the model is trained to map inputs to outputs (examples: Classification, regression).

Unsupervised learning works with unlabeled data and aims to discover patterns or groupings with data itself; common tasks include clustering and dimensionality reduction (examples: k-means clustering, PCA).

Aspect	Supervised Learning	Unsupervised learning
Data Labelling	labelled data	Unlabelled Data
Purpose	Predict Outcomes	Find hidden pattern
Examples	SVM, decision Trees	K-Means, DBSCAN

Q2. DBSCAN V/S Hierarchical

DBSCAN is a density-based clustering algorithm that groups together closely packed points with many neighbors; it can identify clusters of arbitrary shape and automatically discard noise/outliers.

Difference from Hierarchical: Hierarchical clustering builds nested clusters by repeatedly ~~clustering~~ merging or splitting clusters based on distance. It is sensitive to outliers and often requires a set number of clusters unlike DBSCAN, which determines the number of clusters, ~~[illegible]~~ from data density.

- DBSCAN handles outliers better
- Hierarchical creates a tree-like structure called dendrogram.

Q3. Variational Bayesian Gausian Mixture Model

The variational Bayesian Gausian Mixture Model is an extension of the standard GMM that uses variational inference to estimate parameters. It is generally preferred over GMM due to its automatic regularizations, ability to determine the effective number of clusters, and robustness to overfitting and unstable solutions.

Working of Variational Bayesian GMM-

The VBGMM employs variational inference, an advanced optimization technique instead of traditional Expectation-Maximization used in standard GMMs.

- It incorporates prior distributions on model parameters, allowing the algorithm to maximize a lower bound on model evidence (not just raw data likelihood).
- The VBGMM iteratively updates both cluster assignment probabilities and cluster parameters by combining observed data with prior information, providing a Bayesian approach that captures uncertainity in model estimates.

Why VBGMM preferred over GMM?

- Automatic Cluster Number Selection: VBGMM reliably identifiers a suitable number of clusters - even if a larger upper bound is supplied - by naturally pushing unnecesarry cluster weights to near zero.

- **Regularization and Stability** - The inclusion of prior distributions act as a regularization, making the model less prone to unstable solutions and reducing issues like singularities in parameter estimates.
- **Handles Uncertainity**: VBGMM model parameter uncertainity offering more robust results in the presence of outliers or noisy data.
- **less sensitivity to Tuning**: Solutions do not change dramatically with the number of components or initializations, making VBGMM more stable and easier to use in exploratory data analysis.

Feature	Standard GMM	Variational Bayesian GMM
Cluster Number Selection	User-defined	Automatic via prior
Regularization	Explicit, manual	Bayesian prior, built-in
Robustness to outliers	lower	higher
Model Stability	May be unstable	Usualy stable
Parameter uncertainity	No	Yes (Bayesian)

Q4. Isomap Manifold Learning Technique

Applying the isomap manifold learning technique to the digits dataset (such as the one in Sklearn) reduces the original 64-dimensional data of handwritten digit images to 2D while preserving important non-linear structures. The result is a visualization where similar ~~str~~ digits form distinct clusters in the lower-dimensional space.

Steps to apply Isomap to digits Dataset

1. Load and preprocess Data:
   The digits dataset contains handwritten images each of vector of 64 features.

2. Isomap Transformation:
   - Initialize Isomap (common: n-neighbors = 30, n-components = 2)
   - fit and transform the data to reduce to two dimensions

```
from sklearn.dataset import load_digits
from sklearn.manifold import Isomap

digits = load_digits()
Isomap = Isomap (n_neighbors = 30, n_components = 2)
digits_isomap = isomap.fit_transform ( digits.data )
```

3. Visualization:
   - Plot the result - each point colored by its digit class
   - Distinct clusters emerge, showing how Isomap has unfolded the high-dimensional data along its intrinsic ~~method~~. manifold.
   - Example output plot shows digits of the same class grouped together in 2D.

Explaination of Results

- Cluster formation - Digits of the same class (eg all 3's) generally appear closer together, forming visible clusters, indicating that Isomap suceeded at revealing the geometric structure of data.

-) Non-linear Relationships -
  Isomap preserves geodesic (manifold) distances rather than straight Euclidean distances, capturing curved underlying patterns that linear methods like PCA miss.

- Interpretation:
  This technique helps with visualization, exploratory analysis, and reveals hidden structure in image datasets.