1. **Clustering Fundamentals & Evaluation:**
   - **Why:** Clustering is a cornerstone of unsupervised learning. Understanding the core concepts (distance metrics, cluster validity, etc.) is essential.
   - **Scikit-learn:** `sklearn.cluster` (general concepts), metrics in `sklearn.metrics` (e.g., `silhouette_score`, `calinski_harabasz_score`, `davies_bouldin_score`). Learn *how* to evaluate the quality of your clusters.
   - **Key Concepts:** Choosing appropriate distance metrics (Euclidean, Manhattan, cosine, etc.), understanding internal vs. external evaluation methods.

2. **K-Means Clustering:**
   - **Why:** The most widely used and understood clustering algorithm. A solid starting point.
   - **Scikit-learn:** `sklearn.cluster.KMeans`
   - **Key Concepts:** Centroids, inertia, the elbow method (for choosing *k*), initialization methods (K-Means++), limitations of K-Means (spherical clusters, sensitivity to outliers).

3. **Hierarchical Clustering:**
   - **Why:** Offers a different perspective on clustering, creating a hierarchy of clusters (dendrogram). Useful for understanding relationships at different granularities.
   - **Scikit-learn:** `sklearn.cluster.AgglomerativeClustering`
   - **Key Concepts:** Linkage criteria (single, complete, average, ward), dendrogram interpretation, cutting the dendrogram to form clusters.

4. **DBSCAN (Density-Based Spatial Clustering of Applications with Noise):**
   - **Why:** Handles clusters of arbitrary shapes and identifies noise points (outliers). Excellent for data where K-Means struggles.
   - **Scikit-learn:** `sklearn.cluster.DBSCAN`
   - **Key Concepts:** Core points, border points, noise points, `eps` (radius) and `min_samples` parameters, advantages over K-Means for non-spherical clusters.

5. **Gaussian Mixture Models (GMMs):**
   - **Why:** A probabilistic approach to clustering. Assumes data points are generated from a mixture of Gaussian distributions. Provides "soft" cluster assignments (probabilities).

- **Scikit-learn:** `sklearn.mixture.GaussianMixture`
- **Key Concepts:** Probability distributions, expectation-maximization (EM) algorithm, covariance types (full, tied, diag, spherical), model selection (BIC, AIC).

6. **Principal Component Analysis (PCA):**
    - **Why:** The *fundamental* dimensionality reduction technique. Finds the directions of greatest variance in the data.
    - **Scikit-learn:** `sklearn.decomposition.PCA`
    - **Key Concepts:** Eigenvectors, eigenvalues, explained variance ratio, choosing the number of components, data scaling (often crucial before PCA).

7. **t-distributed Stochastic Neighbor Embedding (t-SNE):**
    - **Why:** Excellent for visualizing high-dimensional data in 2D or 3D. Preserves local structure well.
    - **Scikit-learn:** `sklearn.manifold.TSNE`
    - **Key Concepts:** Perplexity parameter, understanding that t-SNE is primarily for visualization (not for transforming data for other algorithms), non-deterministic nature.

8. **Manifold Learning (Beyond t-SNE):**
    - **Why:** This is important to know how data can lie on a lower-dimensional *manifold* within the higher-dimensional space.
    - **Scikit-learn:** `sklearn.manifold` (includes Isomap, LocallyLinearEmbedding, etc.)
    - **Key Concepts:** understanding when linear techniques like PCA will be insufficient.

9. **Anomaly Detection with Isolation Forest:**
    - **Why:** A powerful, tree-based method for identifying outliers. Efficient and works well in high-dimensional spaces.
    - **Scikit-learn:** `sklearn.ensemble.IsolationForest`
    - **Key Concepts:** Isolation, path length, anomaly score, contamination parameter.

10. **Preprocessing for Unsupervised Learning:**

- **Why:** Data preparation is *critical* for unsupervised learning. Scaling, normalization, and handling missing values can drastically impact results.
- **Scikit-learn:** `sklearn.preprocessing` (e.g., `StandardScaler`, `MinMaxScaler`, `RobustScaler`, `Normalizer`).
- **Key Concepts:** Understanding when to use different scaling methods, handling outliers appropriately, impact of preprocessing on distance calculations.