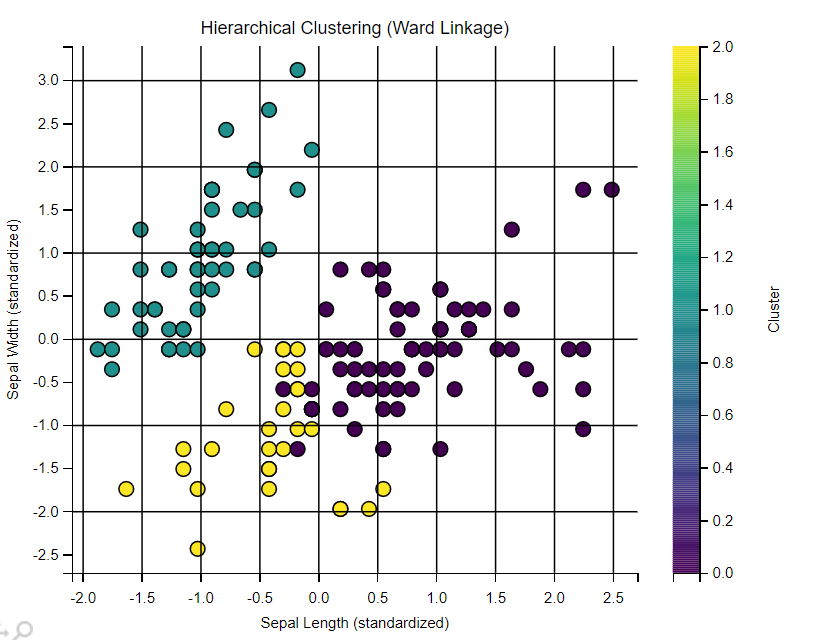
## Cluster Algorithm Analysis

## Iris.py

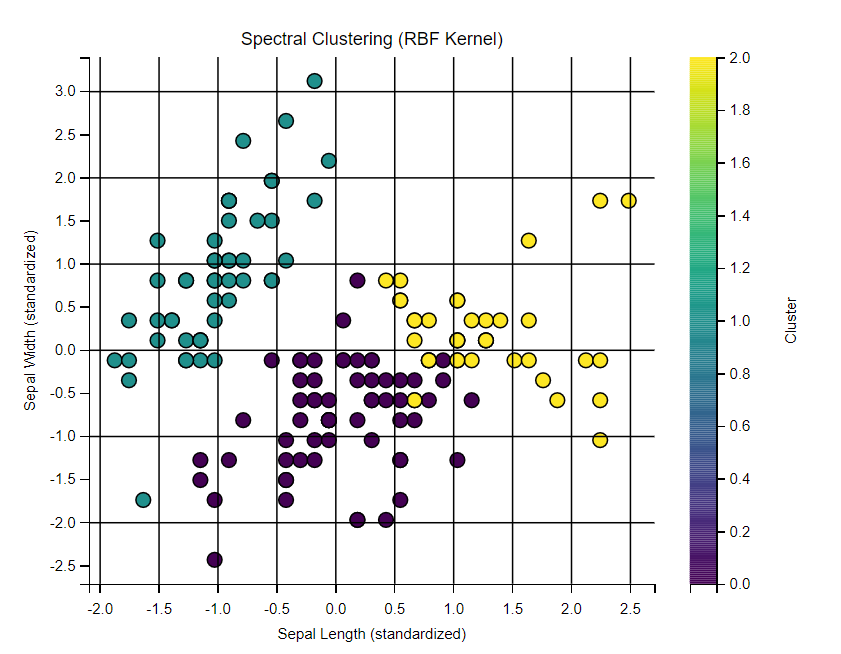
**Code Overview:**  
This Python script performs clustering analysis on the Iris dataset using five different algorithms and visualizes the results.

**Key Components:**

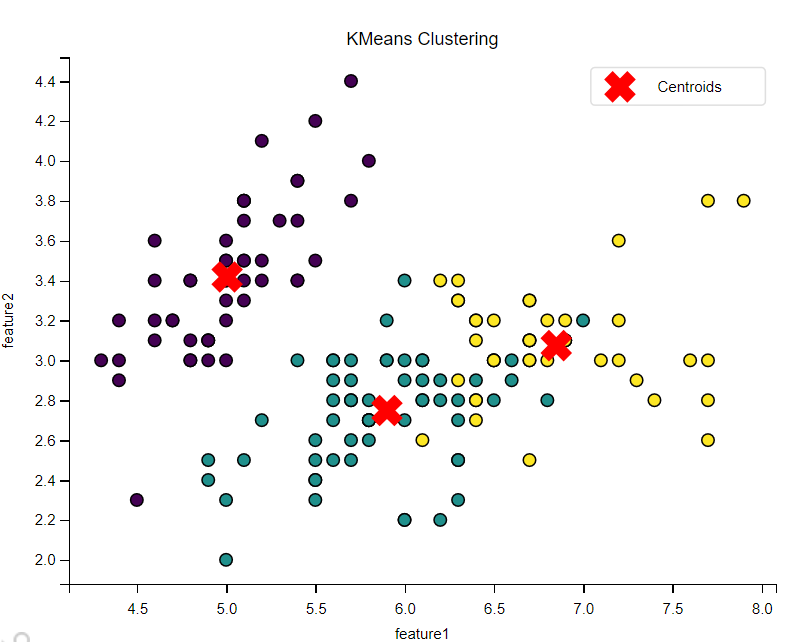
1. **Data Preparation:**
   * Imports the Iris dataset (ID 53) from UCI repository
   * Standardizes features using StandardScaler
2. **Clustering Algorithms:**
   * **Hierarchical Clustering:**
     + Uses Ward linkage method
     + Visualizes with dendrogram and scatter plot



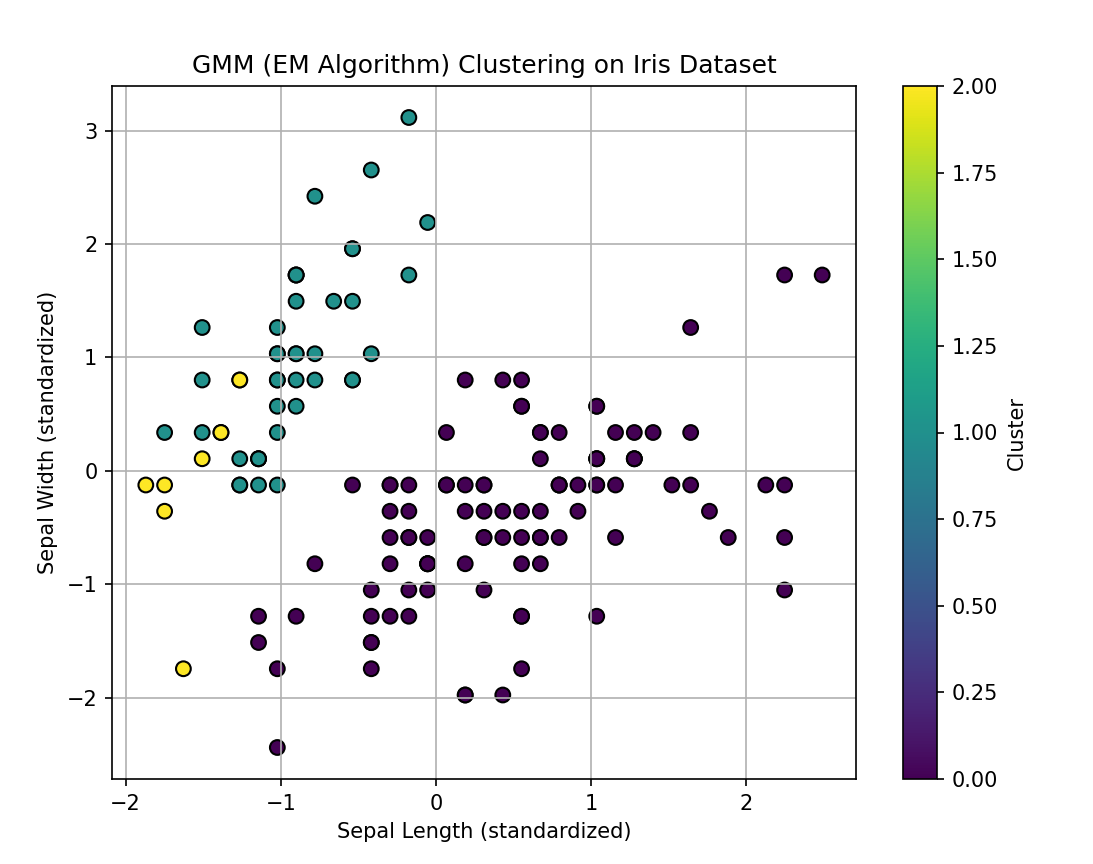
* + **Spectral Clustering:**
    - Uses RBF kernel with gamma=1.0
    - Shows results in scatter plot



* + **K-Means:**
    - Creates 3 clusters
    - Plots data points and centroids

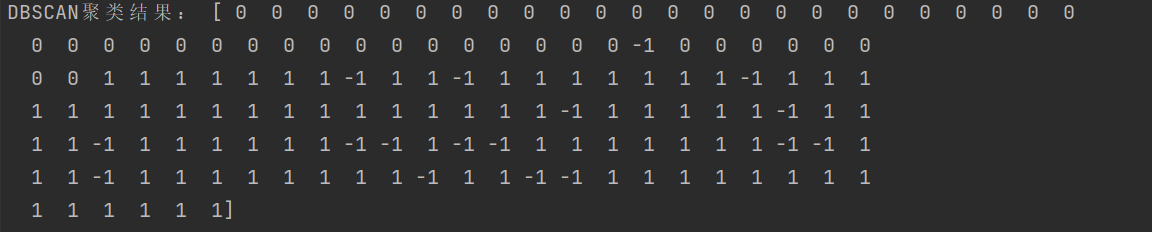


* + **Gaussian Mixture Model (EM):**
    - Implements Expectation-Maximization
    - Displays probabilistic clustering



* + **DBSCAN:**
    - Uses eps=0.5 and min\_samples=5
    - Prints clustering labels

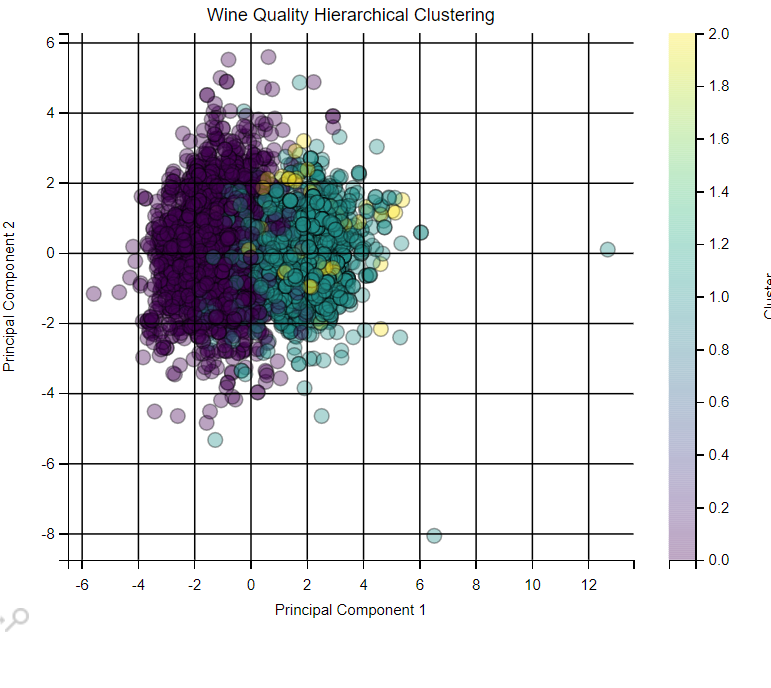
1. **Visualization:**
   * Each algorithm's results are shown in separate matplotlib figures
   * All plots use the viridis colormap and standardized axes



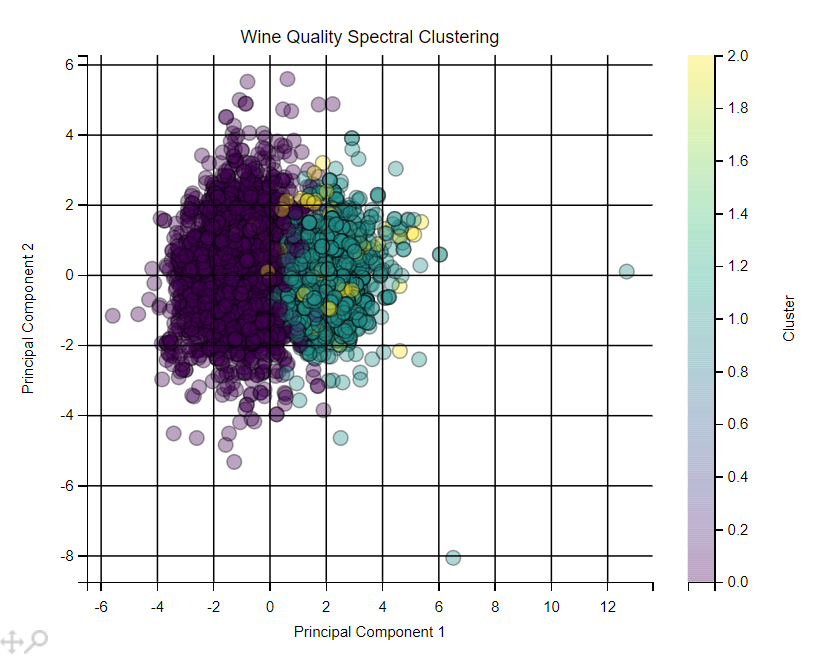
## Wine\_Quality.py

This python script is much like the former iris.py, performs comparative clustering analysis on the Wine Quality dataset using five algorithms:

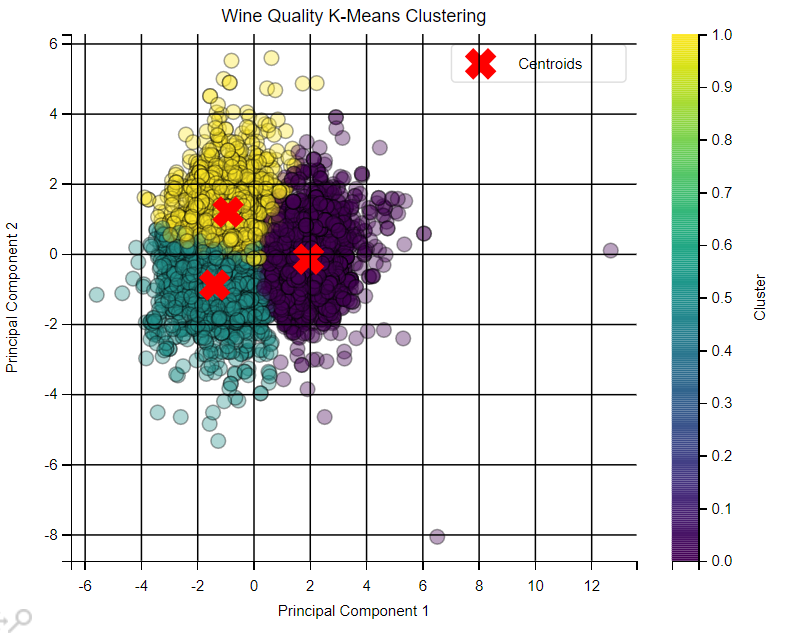
1. Hierarchical Clustering



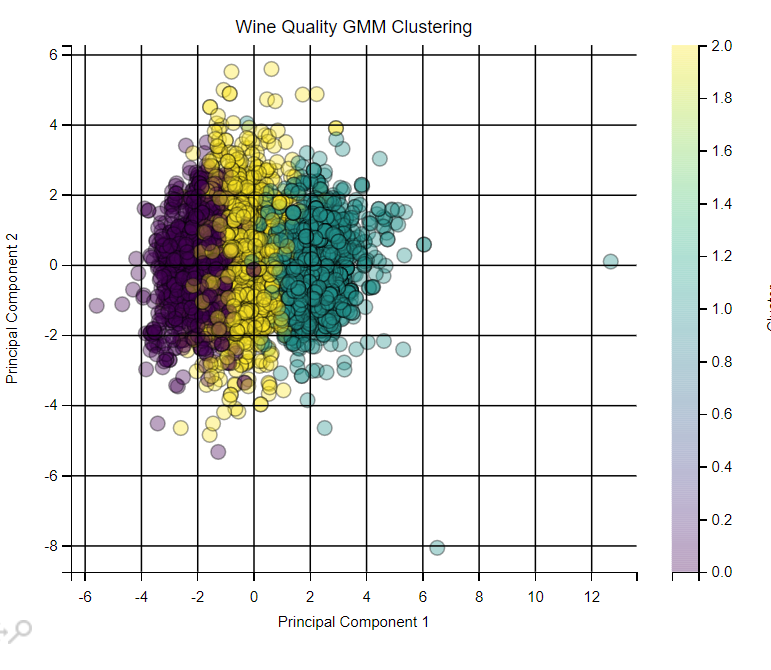
1. Spectral Clustering



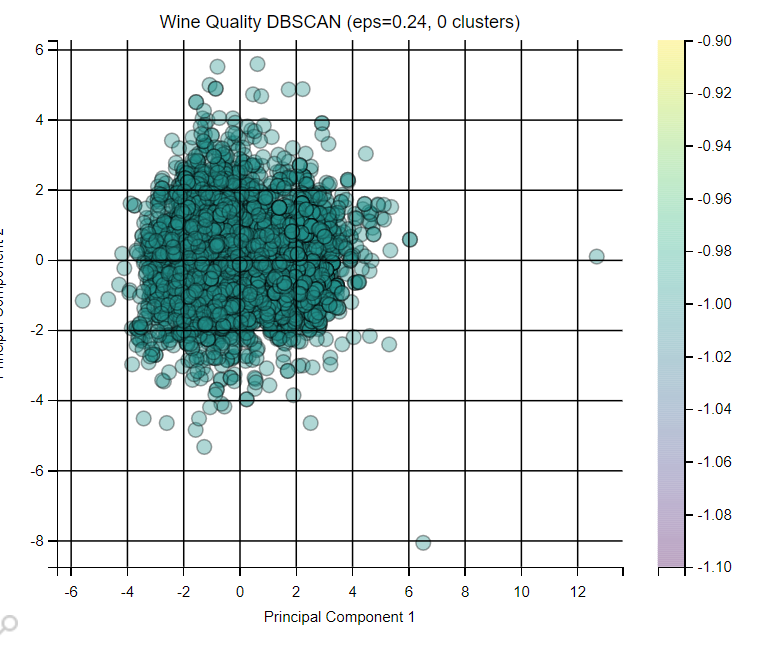
1. K-Means

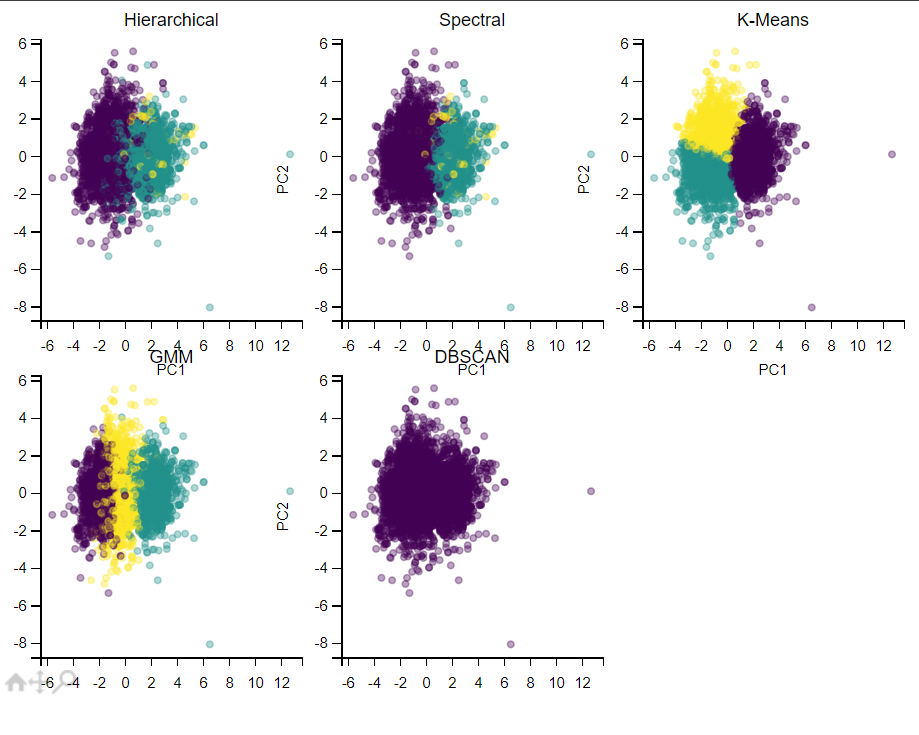


1. Gaussian Mixture Model (EM)



1. DBSCAN





**Key Features**

* **Data Preprocessing**: Automatic feature standardization and PCA visualization
* **Algorithm Optimizations**:
  + Memory-efficient hierarchical clustering
  + Nearest-neighbor spectral clustering
  + Automated DBSCAN parameter tuning
* **Evaluation**: Silhouette scores and runtime metrics
* **Visualization**: Consistent 2D plots with cluster highlighting