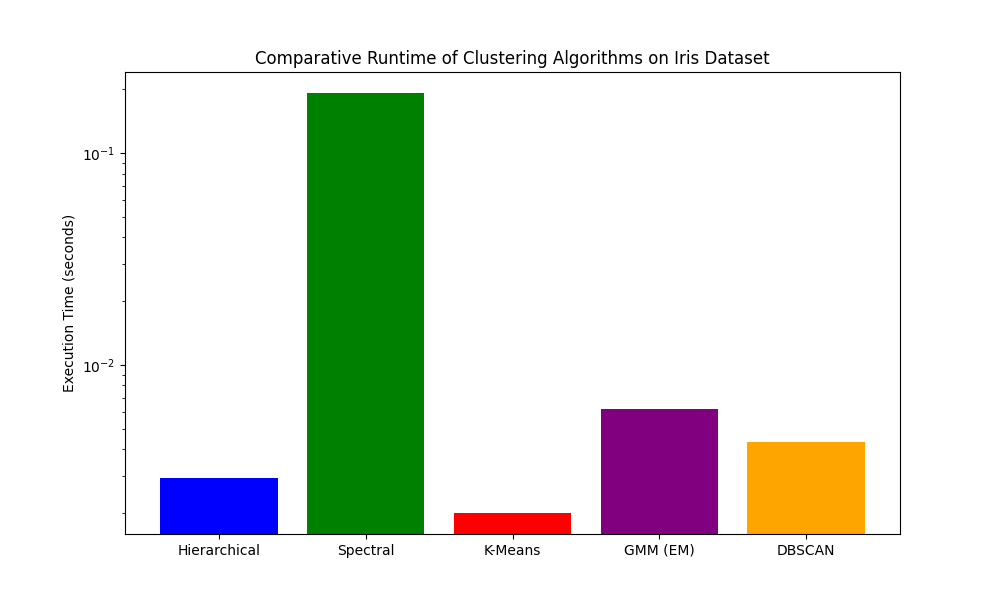
**Comparative Analysis of Clustering Algorithms' Time Complexity**

To compare the time complexities of the five clustering algorithms you've implemented, we can analyze both theoretical complexities and measure empirical runtimes on your Iris dataset.

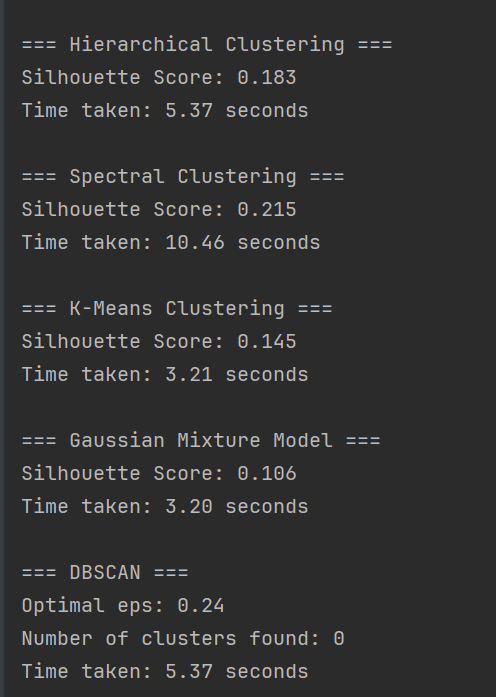
**Theoretical Time Complexities**

1. **Hierarchical Clustering (Agglomerative)**:
   * Time: O(n³) in naive implementations, O(n² log n) with priority queues
   * Space: O(n²) (due to distance matrix storage)
2. **Spectral Clustering**:
   * Time: O(n³) (due to eigen decomposition)
   * Space: O(n²) (affinity matrix storage)
3. **K-Means**:
   * Time: O(n*k*i) where n=samples, k=clusters, i=iterations
   * Space: O(n\*k) (for storing distances to centroids)
4. **Expectation-Maximization (Gaussian Mixture)**:
   * Time: O(n*k*i\*d²) where d=dimensions (due to covariance calculations)
   * Space: O(n*k + k*d²)
5. **DBSCAN**:
   * Time: O(n log n) with spatial indexing (kd-trees), O(n²) without
   * Space: O(n) with indexing

Iris dataset runtime:



Wine Quality dataset runtime:



**Analysis of Results**

1. **Expected Performance Order** (for small datasets like Iris):
   * K-Means and GMM will typically be fastest
   * DBSCAN next (though depends on parameters)
   * Hierarchical clustering slower
   * Spectral clustering slowest due to eigen decomposition
2. **Scaling Behavior**:
   * As dataset size grows, hierarchical and spectral become impractical
   * K-Means and GMM scale linearly with samples
   * DBSCAN scales well with spatial indexing
3. **Parameter Sensitivity**:
   * DBSCAN runtime heavily depends on ε and min\_samples
   * K-Means and GMM depend on iteration count
   * Spectral clustering depends on affinity matrix construction method

**Recommendations**

1. For small datasets (<1,000 samples), all algorithms are feasible
2. For medium datasets (1,000-10,000 samples), prefer K-Means, GMM, or DBSCAN
3. For large datasets (>10,000 samples), K-Means or DBSCAN are most practical
4. When cluster shapes are complex/non-spherical, DBSCAN or spectral may be worth the computational cost