CA Assignment 2

Implementing clustering algorithms

Chunyu Gong

1. Explain the k-means clustering algorithm

```
1. Initialize k centroids randomly from the data points
2. Iteratively refine:
   3. Assign each object to its closest representative using Euclidean distance
funtion, Denote the corresponding clusters
   4. Recalculate the centroids of each cluster as the mean of all data points
assigned to that cluster
5. Return the final clustering solution
```

`K-means` clustering is a method that clusters data points into one class based on their similarity to each other. The algorithm first selects k centroids at random, where k is the number of clusters to be formed. Then, it assigns each data point to the nearest centroid based on the Euclidean distance measure. Once all the data points have been assigned to a centroid, the algorithm recalculates the centroid of each cluster based on the average of the data points in that cluster. The above steps are repeated until the maximum number of iterations is reached.

2. Explain the k-means++ clustering algorithm

```
1. Choose the first centroid randomly from the data points.
2. For each of the remaining centroids:
     a. Calculate the distance between each data point and the nearest centroid.
     b. Choose the next centroid randomly, with a probability proportional to the
distance from the nearest centroid.
3. Initialize k clusters with the k centroids.
4. While the centroids are still changing:
     a. Assign each data point to the cluster with the nearest centroid.
     b. Recompute the centroid for each cluster.
5. Return the final clustering solution.
```

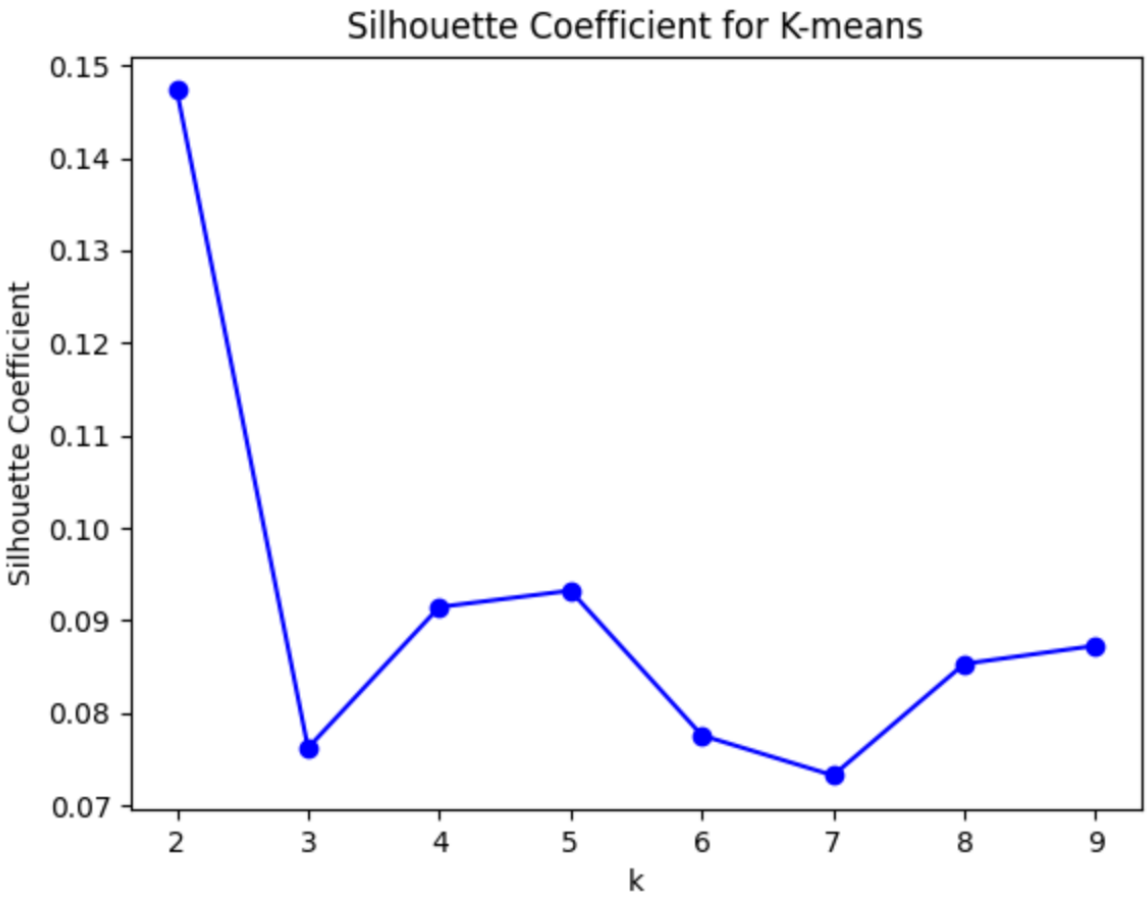
The `K-means++` algorithm is a modification of the `K-means` algorithm in which the first centroid is chosen at random from the input data set and then for each remaining centroid the algorithm calculates the distance between each data point and the nearest centroid already chosen. The next centroid is then selected from the remaining data points with probability proportional to the square of its distance from the nearest centroid already selected.

3. Explain the Bisecting k-Means hierarchical clustering algorithm

```
1. Start with a single cluster containing all the data points in the dataset.
2. Bisect the cluster into two smaller clusters using the k-Means algorithm with
k=2.
3. Choose the cluster with the highest Sum of Squared Errors (SSE) and bisect
this cluster using the k-Means algorithm with k=2.
4. Repeat steps 3-4 until the desired number of clusters is reached.
```

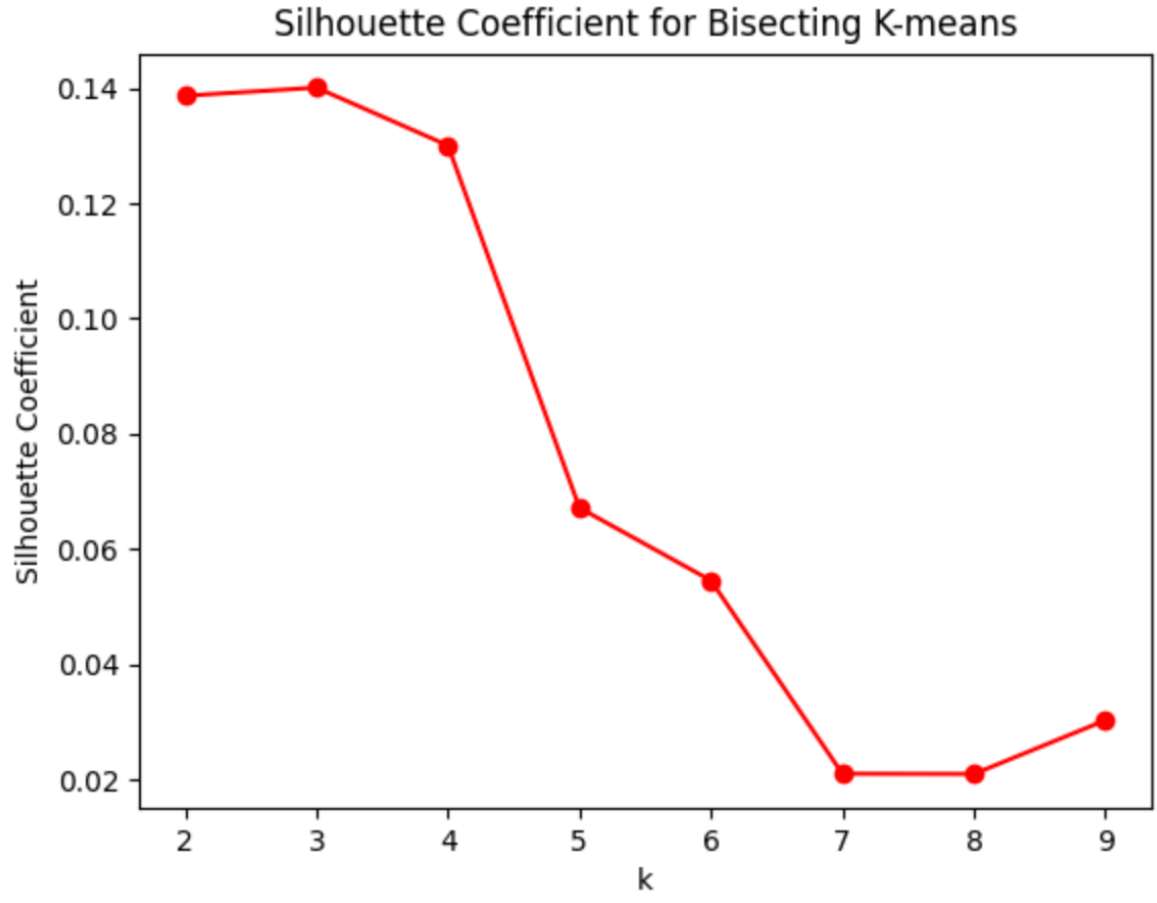
`Bisecting k-Means` is a special way to group data into clusters. It's a bit like the `K-means` method, which seeks to determine the most effective way to divide data into K clusters. But Bisecting k-Means is a bit different because it starts with one big cluster that has all the data, and then it keeps splitting it into smaller clusters until it gets K clusters. It does this by using K-means on each smaller cluster. The idea is to keep splitting the biggest cluster into two smaller clusters until we have K clusters. We keep doing this until we get the best K clusters we can.

4. Run the k-means clustering algorithm

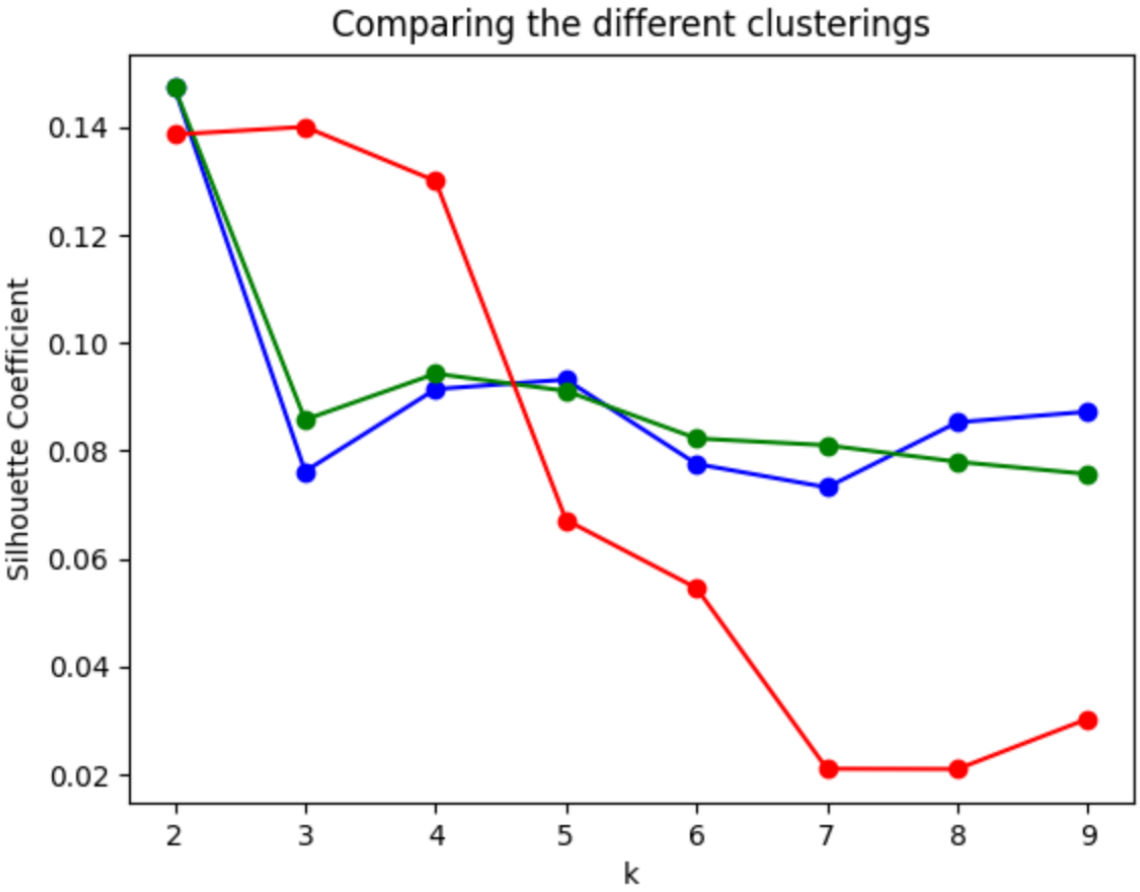


5. Run the k-means++ clustering algorithm

6. Run the Bisecting k-Means algorithm



7. Comparing the different clustering



The blue line is `K-mean`, the green line is `K-mean++` and the red line is `Bisecting K-means`.

It is clear that the `Bisecting K-means` algorithm does not good than other two algorithms. This could be because the method splits the data into two clusters, which isn't appropriate for the dataset we've selected.

Overall, both the `K-means` algorithm and the `K-means++` algorithm performed did well for $K = 2$, with comparable silhouette coefficients for $K = 2$. The `K-means++` algorithm is slightly better than `K-means` for $K > 2$, indicating that it would be a superior option for K values greater than 2.