Yixiao Zhang
31353282
yz3a19@soton.ac.uk

Assignment

COMP6245 Lab 6

1 Labs 1 - 5

I have completed all the five assignments, uploaded reports and have taken on board any feedback provided. The feedback of first three labs are ”complete work, good”, ”complete work, good”, and ”good work” respectively.

2 K-Means Clustering

2.1 Implementation of K-Means Clustering

My K-means algorithm is developed in Python. The pseudocode of algorithm is shown below:

```
Pseudocode of K-Means Clustering
Input: Data = d1, d2, ... dn , K = numbers of cluster
Output: C = K centres
Algorithm:
C = choose random K centres from Data
repeat
    assign each item di in Data to the cluster Ci using its distance to cluster centre
    calculate new centre(mean) Ci for each cluster
until new centre == old centre
```

2.2 Applying K-Means Clustering on mixture of Gaussian probability density

Three mixture of Gaussian probability densities is generated by formula $p\left(x\right) = \sum_{K}^{j=1} \lambda_j N\left(m_j, C_j\right)$, where K is numbers of Gaussian probability density, m_j and C_j are mean and covariance matrix respectively, λ_j is weight of each Gaussian probability density.

The 3 mixture of Gaussian probability densities and 1000 samples from each of mixture densities are shown as Figure 1, Figure 2, and Figure 3. It could be noticed that the data points are mainly distribute like 3 Gaussian probability densities but it is weighted by parameter λ. For example, in Figure 2, the Gaussian probability density which centre at (4,4) is much less weighted than density which centre at (0,3), so that points surround (4,4) is much less than points surround (0,3).

K-Means clustering algorithm is applied on three 1000-point dataset which from three mixture Gaussian density respectively. The clustering result are shown as Figure 1, Figure 2, and Figure 3. In Figure 4, Figure 5, and Figure 6, the initial centres, centres during iterations, and final centres are marked in cross, dot, and star respectively.

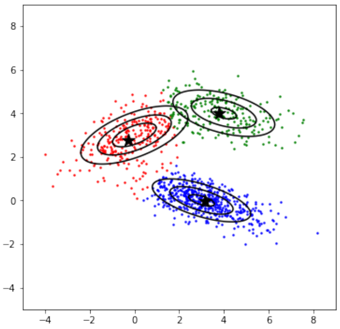

Figure 1: Clustering Case 1

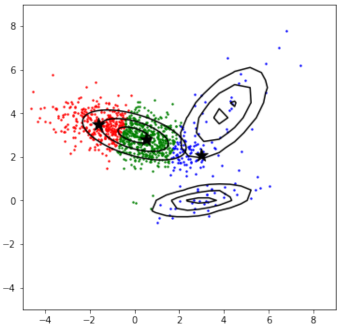

Figure 2: Clustering Case 2

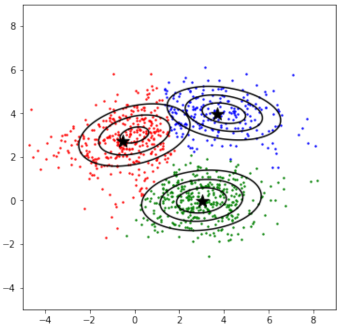

Figure 3: Clustering Case 3

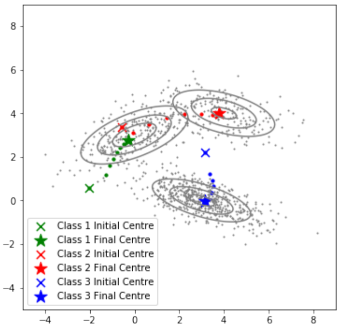

Figure 4: Cluster Centres 1

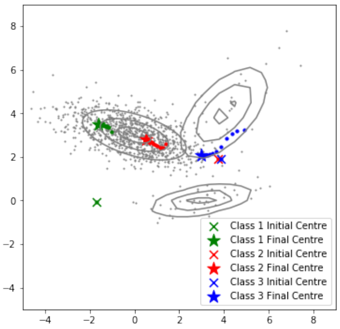

Figure 5: Cluster Centres 2

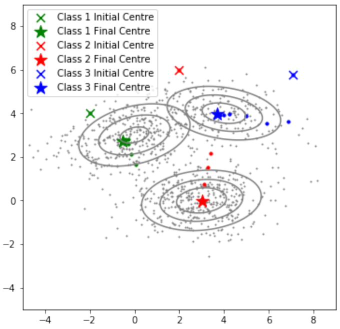

Figure 6: Cluster Centres 3

2.3 Comparing My Implementation with Sklearn Implementation

Another implement of K-Means clustering algorithm ***KMeans()*** is import from sklearn package ***sklearn.cluster***. The function ***Kmeans.fit(data)*** is used to run the K-means algorithm and the parameter ***Kmeans.labels_*** is used to export the centre of the clusters.

In order to compare the sklearn implement with my implement, the sklearn K-Means clustering algorithm is applied on three densities which are used in section above. Figure 7, Figure 8, and Figure 9 show the clustering result of sklearn K-Means algorithm which could be compare with my implement shown in Figure 1, Figure 2, and Figure 3.

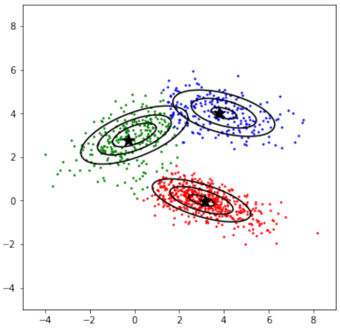

Figure 7: Sklearn Result 1

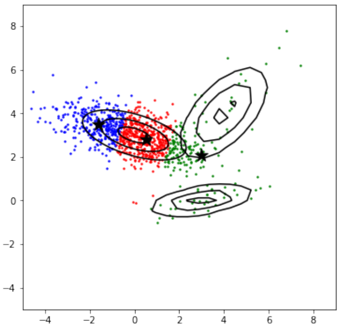

Figure 8: Sklearn Result 2

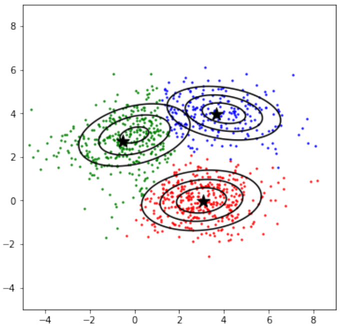

Figure 9: Sklearn Result 3

The accuracy of these three case in two implement are shown in Table 1, the accuracy of these two implement is really similar. In Case 1 and Case 3, my implement and sklearn implement got same accuracy, whereas my implement is slightly better than sklean implement in Case 2.

As clustering often used in unsupervised learning which do not have data label and cannot calculate accuracy directly, some methods may used to evaluate the effective of clustering. Three evaluation methods are also used in the experiment: Firstly, sum of squares method is used. It calculate the sum of the square of distance from each data to the nearest centre. The lower sum indicate the better performance. Secondly, silhouette analysis which is a method of interpretation and validation of consistency within clusters is used. Silhouette coefficient is from -1 to 1, higher coefficient means better performance. Thirdly, Calinski-Harabasz index method which is based on the concept of dense and well-separated clusters is used. It is much quickly than silhouette analysis and the higher score means better performance. From Table 1, no matter use which method, sum of squares, silhouette analysis, and Calinski-Harabasz index, the performance of my implement and sklearn implement is very similar.

Table 1: Clustering Evaluation (Case 1 to Case 3)

	Accuracy		Error Sum of Squares		Silhouette Coefficient		Calinski-Harabaz Index	
	My impl	Sklearn	My impl	Sklearn	My impl	Sklearn	My impl	Sklearn
Case 1	94.29%	94.29%	2.15×10^3	2.15×10^3	5.45×10^{-1}	5.45×10^{-1}	1.30×10^3	1.30×10^3
Case 2	57.06%	55.06%	1.46×10^3	1.46×10^3	3.97×10^{-1}	3.82×10^{-1}	8.88×10^2	8.88×10^2
Case 3	90.28%	90.28%	2.78×10^3	2.78×10^3	4.84×10^{-1}	4.84×10^{-1}	1.09×10^3	1.09×10^3

In conclusion, as three cases are evaluated by accuracy, sum of squares, silhouette analysis, and Calinski-Harabasz index, my implement of K-Means clustering algorithm shows the same performance with sklearn implement.

2.4 Importance of Initial Cluster Centres

As it is said that the K-means algorithm is sensitive to the initial centres, a group of experiment is designed to demonstrate this feature. A mixture of Gaussian probability density is used in Case 4 and Case 5. Two group of initial centres are generated and the clustering results are shown in Figure 10 and Figure 11, as well as the initial centres, centres during iterations, and final centres are marked in cross, dot, and star respectively in Figure 12 and Figure 13.

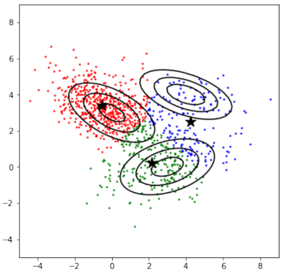

Figure 10: Case 4

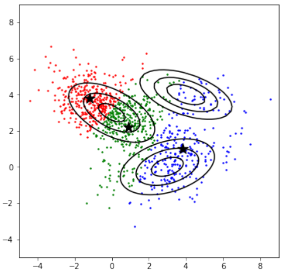

Figure 11: Case 5

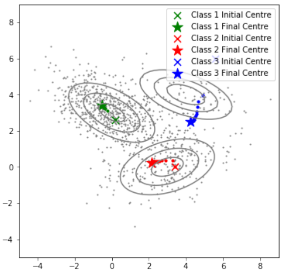

Figure 12: Case 4

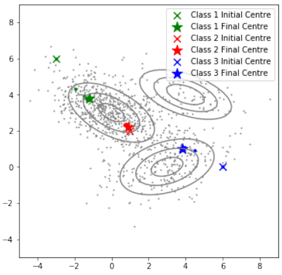

Figure 13: Case 5

Table 2 shows the locations of initial centres and the clustering accuracy. The result shows that location of initial centres could dramatically influence the clustering result. On the other words, K-means algorithm is sensitive to the initial centres.

Table 2: Clustering Evaluation (Case 4 and Case 5)

	Initial Centre 1	Initial Centre 2	Initial Centre 3	Accuracy
Case 4	$(0.2, 2.6)$	$(3.4, 0.0)$	$(5.6, 6.0)$	$\mathbf{80.38\%}$
Case 5	$(-3.0, 6.0)$	$(1.0, 2.0)$	$(6.0, 0.0)$	59.66%

2.5 Importance of Parameter K

The K-means algorithm is also sensitive to the parameter K, another group of experiment is designed to demonstrate this feature. A mixture of Gaussian probability density is used in Case 6 and Case 7. Case 6 has 3 initial centres, whereas Case 7 has 5 centres. The initial centres are generated and the clustering results are shown in Figure 14 and Figure 15, as well as the initial centres, centres during iterations, and final centres are marked in cross, dot, and star respectively in Figure 16 and Figure 17. As the data only has 3 classes, red cluster and pink cluster, blue cluster and green cluster is combined to one cluster in order to calculate accuracy.

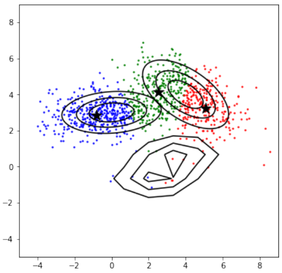

Figure 14: Case 6

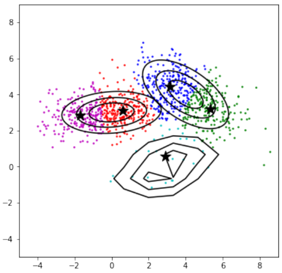

Figure 15: Case 7

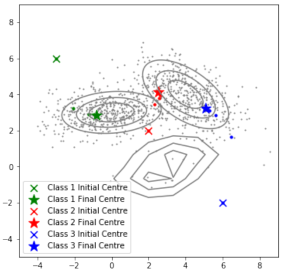

Figure 16: Case 6

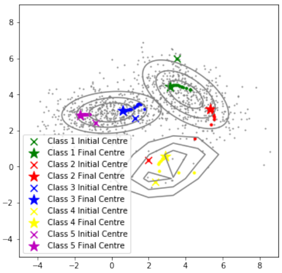

Figure 17: Case 7

Table 3 shows the k value, locations of initial centres and the clustering accuracy. The result shows that parameter K which is a parameter to describe numbers of cluster could dramatically influence the clustering performance.

Table 3: Clustering Evaluation (Case 6 and Case 7)

	Value of K	Initial Centres	Accuracy
Case 6	3	$(-1.1, 3.0)$, $(2.2, 4.1)$, $(4.5, 2.6)$	66.47%
Case 7	5	$(-1.9, 2.3)$, $(0.7, 3.1)$, $(2.4, 0.6)$, $(2.5, 4.3)$, $(5.2, 3.1)$	$\mathbf{96.75}\%$

2.6 Application on Iris Dataset

Iris dataset from the UCI repository is used in this section. The Iris dataset has 3 classes and 4 attributes, so that K-means clustering algorithm is applied on a 3-means 4 detentions data. The clustering result is shown in Table 4. It can be noticed that the K-means algorithm achieves 89.33% accuracy.

Table 4: Comparison of accuracy of two clustering

	Number of attributes	Iris Setosa	Iris Versicolour	Iris Virginica	Overall
Before data selection	4	100.00%	98.00%	72.00%	89.33%
After data selection	2	100.00%	98.00%	92.00%	**96.67%**

As not every attribute is strongly related to class(label), a data pre-processing which includes data selection is developed. A famous data mining software, WEKA, is used to choose two most important attributes. The data selection result shows that petal length and petal width are two most important attributes which should be choose. After data selection, the accuracy improved around 7% to 96.67% as Table 4 shows.

The data after data selection which only has two attributes, petal length and petal width is shown as Figure 18. In Figure 19, the initial centres, centres during iterations, and final centres of K-means are marked in cross, dot, and star respectively. In Figure 20, the true clustering is marked in dot and false clustering is marked in cross. There are 2 blue cross, 3 green cross, and 0 red cross, so that the accuracy of Iris Versicolour, Iris Virginica, and Iris Setosa are 98%, 92%, and 100% respectively which is also shown in Table 4.

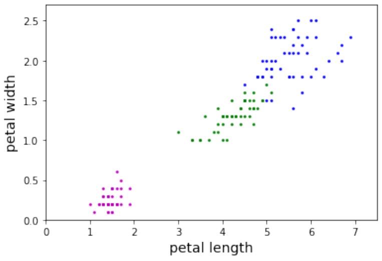

Figure 18: Original Data

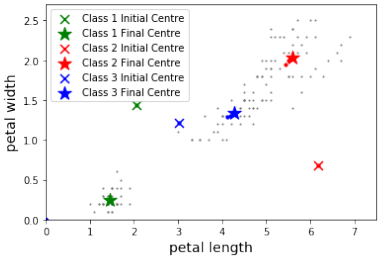

Figure 19: Clustering Centres

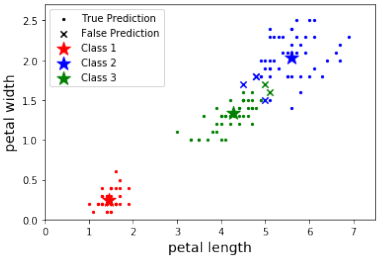

Figure 20: Clustering Result