Machine Learning (CSC 246): Project 3 Writeup

Catherine Giugno

10 February 2022

1 How To Run Program

As with any python project, the first step is to open the Command Prompt/Terminal, navigate to the project folder, and run the program in python, using the command:

```
python em.py
```

The terminal will then print a welcome message and allow you to choose which dataset (A, B, C, or D) that you desire to use.

```
WELCOME TO THE TERMINAL!
Please select the dataset you wish to examine:
1. Dataset A (Enter 1)
2. Dataset B (Enter 2)
3. Dataset C (enter 3)
4. Dataset Z (enter 4
Enter a number from 1 to 4:
```

If the input '1', '2' '3', or '4' is not entered, then the program will quit. Otherwise: The terminal will print another welcome message and allow you to choose what you wish to do with the given dataset:

```
WELCOME TO THE DATASET TESTER!
Please select a functionality:
1. Read in and graph the parameters for the best models of this
dataset (enter 1)
2. Train a new model
Please enter 1 or 2:
```

If the input '1' or '2' is not entered, then the program will quit.
Otherwise:
1. If 1 is entered, the parameters for the best model of the dataset will be read in. The points in the dataset will be plotted and categorized according to the model, then shown to you in a graph.
2. You will be prompted for the number of clusters you wish to model for and the number of times you wish the EM algorithm to run.

> Please enter the number of clusters you wish to model:
> Please enter the number of iterations you wish the EM algorithm to run:

Once, you have entered these numbers, the program will run the EM algorithm on the dataset for the specified model and number of iterations and graph the log likelihood over the number of iterations. If the dataset dimensionality is equal to 2, the points in the dataset will be plotted and categorized according to the final model, then shown to you in a graph.

2 Overall Approach to Model Determination

2.1 Determining the Number of Clusters

In order to determine the number of clusters, while reducing the likelihood of overfitting, I compared the Bayesian Information Criterion for each model with a cluster number of k = 2, 3,... 19. The Bayesian Information Criterion is a well-known heuristic for determining the most robust out of a set of models, using the likelihood function, the number of free parameters (which for Expectation-Maximization with multivariate Gaussians include the values in μ, σ and π. The formula for the Bayesian Information Criterion is shown below:

$$BIC = -2\,ln(l) + k\,ln(n) \tag{1}$$

where l is the likelihood function computed by Expectation-Maximization, k is the number of free parameters, and n is the number of data points.
The Bayesian Information Criterion is a good fit for this particular problem because it works best when the number of samples, n, is far higher than the number of clusters, k. This is true of all the datasets my program is attempting to fit a model to. I also chose this particular metric over AIC (Akaike's Information Criterion), as it penalizes model complexity more strongly and I wanted to favor simpler models.

The Bayesian Information Criterion for models with cluster number, k = 2, 3, ... 10 fitted to dataset A by the EM algorithm for 100 iterations is shown in Figure 1.

As can be seen from the graph, the point at which the BIC is minimized

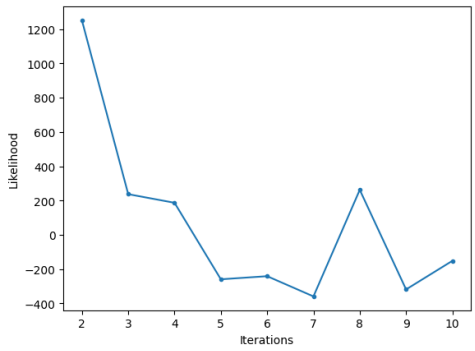

Figure 1: Bayesian Information Criterion for models with cluster number k = 2, 3, ... 10 fitted to dataset A

is at k = 7, that is, a model with 7 clusters. That would suggest that the model that best categorizes the points in the graph is one with 7 clusters. However, let us examine the corresponding graph of the points, sorted into n=7 clusters for this model. This is shown in Figure 2.

From this graph, it in fact appears as though there are functionally 6 groups

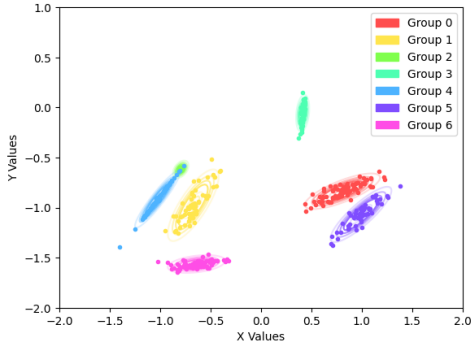

Figure 2: Dataset A, fitted to a model with 7 clusters, after 100 steps of Expectation Maximization

and that the 2nd group has converged upon a single point. But why is it, then, that the EM algorithm run with k=6 produces a model with a higher

BIC score? Again, let us examine the corresponding graph of the points, sorted into n=6 clusters. This is shown in Figure 3.

As can be inferred from Figure 3, by some quirk of initialization, Group

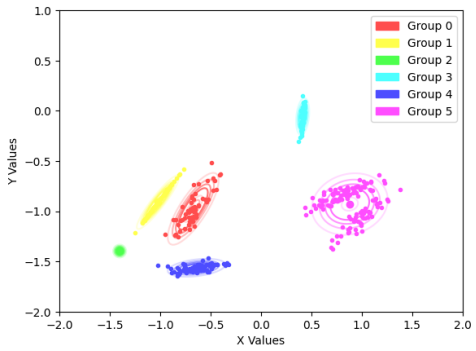

Figure 3: Dataset A, fitted to a model with 6 clusters, after 100 steps of Expectation Maximization

2 ended up far to the left side of the graph where it was crowded out by the other Groups and converged on a single point. Meanwhile, Group 5 expanded to cover the collection of neighboring points that its unlucky counterpart would have otherwise labelled. So can there be a *better* categorization of dataset A than what is seen in this Figure 3?

The answer is yes.

By restarting the program with random initialization, I managed to find a model with n = 6 with graph shown in Figure 4.

As can be seen, this particular model for dataset A appears to cover all

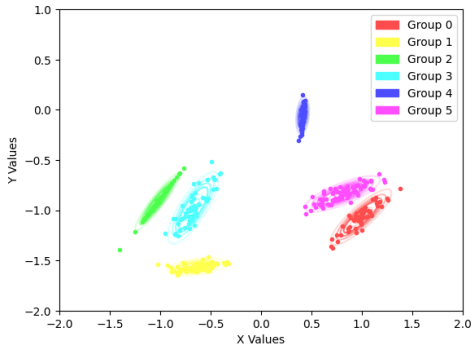

Figure 4: Dataset A, fitted to a model with 6 clusters, after 100 steps of Expectation Maximization

the points in a similar matter as the model with seven clusters in Figure 2. *However*—It does not contain the Group that has converged upon a single point. The BIC value for this six-cluster model is -383.2, which is comparable to that of the seven-cluster model. As mentioned above, however, I have chosen to favor simpler models of the dataset, and thus find the six-cluster model to be the most compelling. Its values for μ, σ and π are found in the file "BestA.txt." A graph of the log likelihood for 100 iterations of the Expecation-Maximization function on dataset A with this six-cluster model is shown in Figure 5 below.

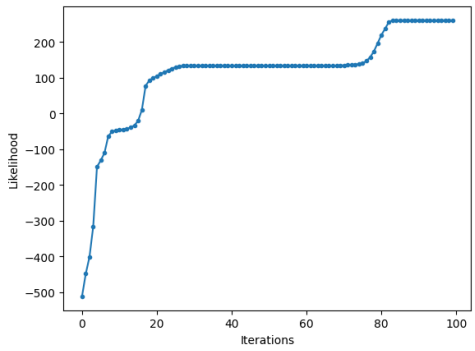

Figure 5: Log Likelihood plotted over 100 iterations of the EM algorithm for dataset A, fitted to a model with 6 clusters

As can be seen, the log likelihood value is consistently increasing throughout every iteration. It converges at around i = 85 iterations to a value just above 200.

The Bayesian Information Criterion for models with cluster number k = 2, 3, ... 10 fitted to dataset B by the EM algorithm for 100 iterations is shown in Figure 6.

As can be clearly seen from the graph, the point at which the BIC is minimized is at k = 3, that is, a model with three clusters. Therefore, we know that the best model of the data in set B has three clusters.
A graph of the log likelihood for 100 iterations of the Expectation-Maximization function on dataset B with a model that has k=3 clusters is shown in Figure 7 below.

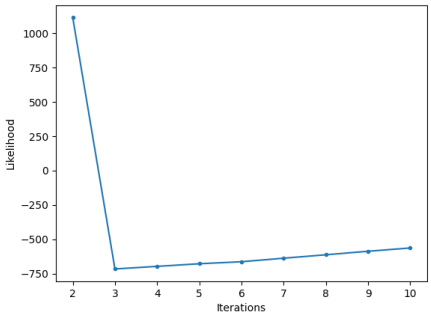

Figure 6: Bayesian Information Criterion for models with cluster number k = 2, 3, ... 10 fitted to dataset B

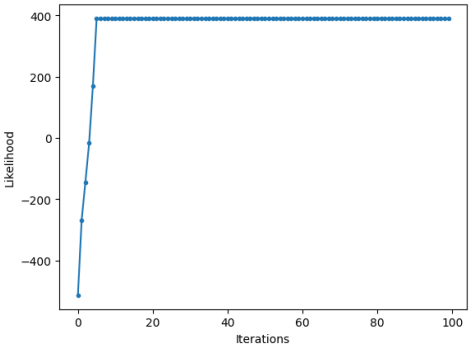

Figure 7: Log Likelihood plotted over 100 iterations of the EM algorithm for dataset B, fitted to a model with 3 clusters

As can be seen, the log likelihood value is consistently increasing throughout every iteration. It converges at around i = 10 iterations to a value just below 400.
The corresponding graph of the points, sorted into the n=3 clusters for this model, is shown in Figure 8. Points are labeled by color according to what group they're in. The Multivariate Gaussians representing the clusters are shown using groups of rings of contour lines, centered at μ, the mean.

The Bayesian Information Criterion for models with cluster number, k = 2, 3, ... 10 fitted to dataset C by the EM algorithm for 100 iterations is shown in Figure 9.

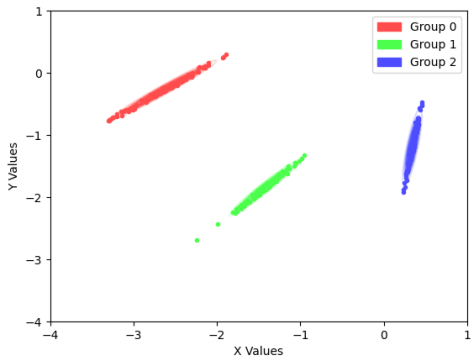

Figure 8: Dataset B, fitted to a model with 3 clusters, after 100 steps of Expectation Maximization

As can be seen from the graph, the point at which the BIC is minimized

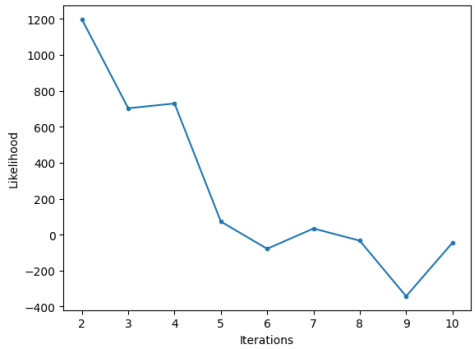

Figure 9: Bayesian Information Criterion for models with cluster number k = 2, 3, ... 10 fitted to dataset C

is at k = 9, that is, a model with 9 clusters. That would suggest that the model that best categorizes the points in the graph is one with, of course, 9 clusters. However, let us examine the corresponding graph of the points, sorted into n=9 clusters for this model, just as we did with the seven-cluster model in dataset A. This is shown in Figure 10.

As before, when we examine this graph, it appears as though there are functionally 8 groups and that the Group 4 has converged onto an area similar to that of the Group 7. But why is it, then, that the EM algorithm run with k=8 produces a model with a higher BIC score? Again, let us examine the corresponding graph of the points, sorted into n=8 clusters. This is

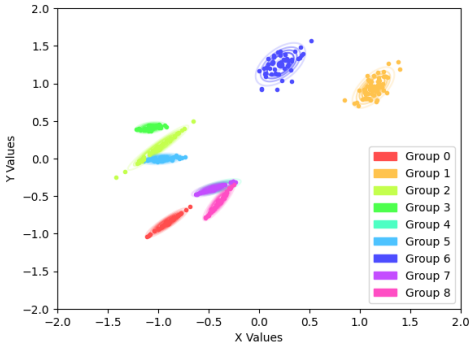

Figure 10: Dataset C, fitted to a model with 9 clusters, after 100 steps of Expectation Maximization

shown in Figure 11.

As can be inferred from Figure 11, by some quirk of initialization, Group

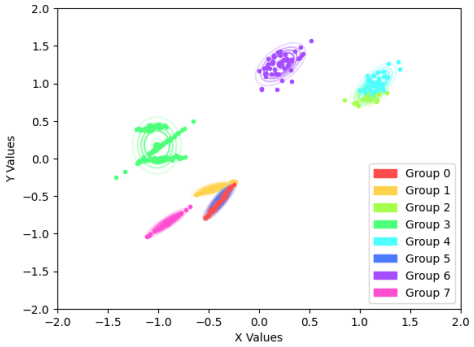

Figure 11: Dataset C, fitted to a model with 8 clusters, after 100 steps of Expectation Maximization

5 has converged to the same area as covered by Group 0, and Groups 2 and 4 are crowded into the same cluster of points, as Group 3 has spread out to cover 2 extra clusters of points. Even to the human eye, this is clearly not the best categorization of dataset C. So can there be a *better* categorization? The answer is yes.

By restarting the program with random initialization, I managed to find a model with n = 8 with graph shown in Figure 12.

As can be seen, this particular model for dataset C appears to cover all the points in a similar matter as the model with nine clusters in Figure 10.

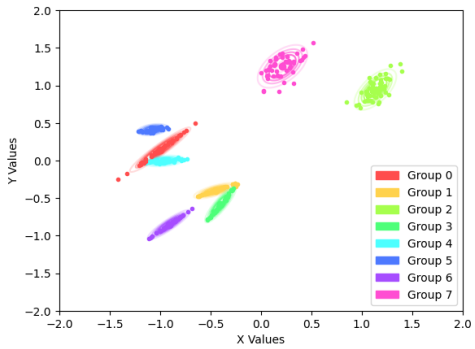

Figure 12: Dataset C, fitted to a model with 8 clusters, after 100 steps of Expectation Maximization

However—It does not contain the Group from the nine-cluster model that converged upon the area of another group. The BIC value for this eight-cluster model is -365.0, which is comparable to that of the nine-cluster model. As mentioned above, however, I have chosen to favor simpler models of the dataset, and thus find the eight-cluster model to be the most compelling. Its values for μ, σ and π are found in the file "BestC.txt." A graph of the log likelihood for 100 iterations of the Expecation-Maximization function on dataset C with this eight-cluster model is shown in Figure 13 below.

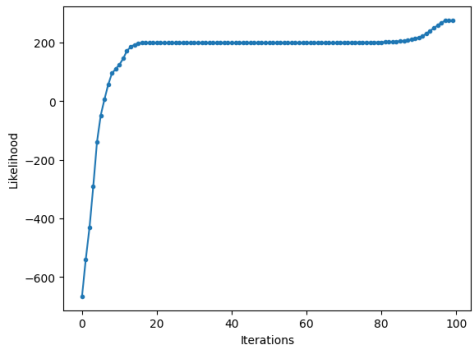

Figure 13: Log Likelihood plotted over 100 iterations of the EM algorithm for dataset C, fitted to a model with 8 clusters

As can be seen, the log likelihood value is consistently increasing throughout every iteration. It converges at around i = 85 iterations to a value just

above 200.

The Bayesian Information Criterion for models with cluster number k = 2, 3, ... 10 fitted to dataset Z by the EM algorithm for 100 iterations is shown in Figure 14.

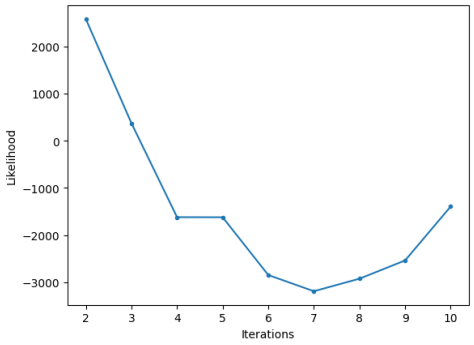

Figure 14: Bayesian Information Criterion for models with cluster number k = 2, 3, ... 10 fitted to dataset Z

As can be clearly seen from the graph, the point at which the BIC is minimized is at k = 7, that is, a model with seven clusters. Though we can't say for *certain* that the best model of the data in set Z has 7 clusters (because of the clear issues with the BIC in datasets A and C), we can make an educated guess that 7 clusters is a good fit for the data in set Z.
A graph of the likelihood for 100 iterations of the Expecation-Maximization function on dataset Z with a model that has k=7 clusters is shown in Figure 15 below.

As can be seen, the log likelihood value is consistently increasing throughout every iteration. It converges, like in dataset B, at around i = 10 iterations to a value slightly below 2500.
Unfortunately, due to the fact that the data is 8-dimensional, no corresponding graph of the sorted points is able to be shown.

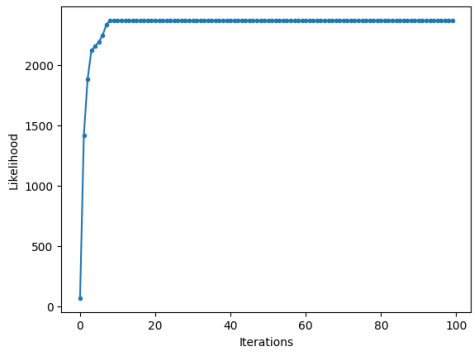

Figure 15: Log Likelihood plotted over 100 iterations of the EM algorithm for dataset Z, fitted to a model with 7 clusters