Clustering using K-Means and EM and Gaussian mixture models(GMM)

Menaka Kollu Nishanth Solomon Sushilkumar Muralikumar
Arizona State University Arizona State University Arizona State
University
mkollu@asu.edu nsolomo2@asu.edu smural32@asu.edu

***Abstract*-The Assignment focuses on Clustering Algorithms based on unsupervised learning. The aim is to create three different models with K-means and EM and Gaussian mixture models(GMM) each, these models create 3, 5 and 7 clusters of the data. This report serves as an all in resource, where we introduce the concepts of the clustering algorithms followed by the results of the problem that we worked on as part of the assignment.**

I. INTRODUCTION

Unsupervised learning is a type of machine learning algorithm used to draw inferences from datasets consisting of input data without labeled responses[1]. The most common unsupervised learning method is cluster analysis, which is used for exploratory data analysis to find hidden patterns or groupings in data[1]. The clusters are modeled using a measure of similarity which is defined upon metrics such as Euclidean or probabilistic distance[1]. Common clustering algorithms include:

- **Hierarchical clustering**: builds a multilevel hierarchy of clusters by creating a cluster tree[1].
- **K-Means clustering**: partitions data into k distinct clusters based on the distance to the centroid of a cluster[1].
- **Gaussian mixture models**: models clusters as a mixture of multivariate normal density components[1].
- **Self-organizing maps**: uses neural networks that learn the topology and distribution of the data[1].
- **Hidden Markov models**: uses observed data to recover the sequence of states[1].

Unsupervised learning methods are used in bioinformatics for sequence analysis and genetic clustering; in data mining for sequence and pattern mining, in medical imaging for image segmentation; and in computer vision for object recognition[1]. The scope of this report is limited to the K-Means and the Gaussian Mixture Models.

1.1. K-Means Clustering:

K-means clustering is a type of unsupervised learning, which is used when you have unlabeled data (i.e., data without defined categories or groups)[5]. The goal of this algorithm is to find groups in the data, with the number of groups represented by the variable *K*[5]. The algorithm works iteratively to assign each data point to one of the *K* groups based on the features that are provided[5]. Data points are clustered based on feature similarity[5]. The results of the *K*-means clustering algorithm are:

A. The centroids of the *K* clusters, which can be used to label new data[5]
B. Labels for the training data (each data point is assigned to a single cluster)[5]

Rather than defining groups before looking at the data, clustering allows you to find and analyze the groups that have formed organically[5]. Each centroid of a cluster is a collection of feature values that define the resulting groups[5]. Examining the centroid feature weights can be used to qualitatively interpret what kind of group each cluster represents[5].

1.2. Gaussian Mixture Model(GMM) Clustering Algorithm:

GMM is a type of clustering algorithm where each cluster is modeled according to a different Gaussian distribution[7]. This flexible and probabilistic approach to modeling the data means that rather than having hard assignments into clusters like k-means, we have soft assignments[7]. This means that each data point could have been generated by any of the distributions with a corresponding probability[7]. In effect, each distribution has some ‘responsibility’ for generating a particular data point[7].

In order to estimate this model, one plausible way is to introduce a latent variable γ (gamma) for each data point[7]. This assumes that each data point was generated by using some information about the latent variable γ[7]. In other words, it tells us which Gaussian generated a particular data point[7]. In practice, however, we do not observe these latent variables so we need to estimate them[7]. Thus in order to practically implement the GMM, we use a technique known as the Expectation-Maximization(EM)[7]. The EM Technique is discussed in the following section.

1.3. Expectation-Maximization(EM) Clustering Technique:

The EM technique is similar to the K-Means technique[6]. The basic operation of K-Means clustering algorithms is relatively simple: Given a fixed number of *k* clusters, assign observations to those clusters so that the means across clusters (for all variables) are as different from each other as possible[6]. The EM algorithm extends this basic approach to clustering in two important ways[6]:

- Instead of assigning examples to clusters to maximize the differences in means for continuous variables, the EM clustering algorithm computes the probabilities of cluster memberships based on one or more probability distributions[6]. The goal of the clustering algorithm then is to maximize the overall probability or likelihood of the data, given the (final) clusters[6].

II. METHODS

2.1. K-Means Algorithm:

K-means algorithm is an iterative algorithm that tries to partition the dataset into *K* pre-defined distinct non-overlapping subgroups (clusters) where each data point belongs to only one group[3]. It tries to make the inter-cluster data points as similar as possible while also keeping the clusters as different (far) as possible[3]. It assigns data points to a cluster such that the sum of the squared distance between the data points and the cluster’s centroid (arithmetic mean of all the data points that belong to that cluster) is at the minimum[3]. The less variation we have within clusters, the more homogeneous (similar) the data points are within the same cluster[3].

The K-Means algorithm is as follows:

1. Specify the number of clusters *K*[3].
2. Initialize centroids by first shuffling the dataset and then randomly selecting *K* data points for the centroids without replacement[3].
3. Keep iterating until there is no change to the centroids. i.e assignment of data points to clusters isn't changing[3].
4. Compute the sum of the squared distance between data points and all centroids[3].
5. Assign each data point to the closest cluster (centroid)[3].
6. Compute the centroids for the clusters by taking the average of all data points that belong to each cluster[3].

Mathematically the process can be represented as follows:
Given a set of observations ($\mathbf{x}_1$, $\mathbf{x}_2$, ..., $\mathbf{x}_n$), where each observation is a *d*-dimensional real vector, *k*-means clustering aims to partition the *n* observations into k ($\leq$ n) sets $\mathbf{S} = \{S_1, S_2, ..., S_k\}$ so as to minimize the within-cluster sum of squares (WCSS) (i.e. variance)[2]. Formally, the objective is to find[2]:

$$\underset{\mathbf{S}}{\arg\min} \sum_{i=1}^{k} \sum_{\mathbf{x} \in S_i} \|\mathbf{x} - \boldsymbol{\mu}_i\|^2 = \underset{\mathbf{S}}{\arg\min} \sum_{i=1}^{k} |S_i| \operatorname{Var} S_i$$

where $\boldsymbol{\mu}_i$ is the mean of points in S_i. This is equivalent to minimizing the pairwise squared deviations of points in the same cluster[2]:

$$\underset{\mathbf{S}}{\arg\min} \sum_{i=1}^{k} \frac{1}{2|S_i|} \sum_{\mathbf{x},\mathbf{y} \in S_i} \|\mathbf{x} - \mathbf{y}\|^2$$

The equivalence can be deduced from identity[2]:

$$\sum_{\mathbf{x} \in S_i} \|\mathbf{x} - \boldsymbol{\mu}_i\|^2 = \sum_{\mathbf{x} \neq \mathbf{y} \in S_i} (\mathbf{x} - \boldsymbol{\mu}_i)(\boldsymbol{\mu}_i - \mathbf{y}).$$

Because the total variance is constant, this is equivalent to maximizing the sum of squared deviations between points in *different* clusters (between-cluster sum of squares, BCSS), which follows from the law of total variance[2].

2.2. EM Algorithm:

The EM algorithm consists of two steps, an E-step or an Expectation step and M-step or Maximisation step[7]. Given the statical model which generates a set **X** of observed data, a set of unobserved latent data or missing values **Z** and a vector of unknown parameters $\boldsymbol{\theta}$, along with a Likelihood function, $L(\boldsymbol{\theta}; \mathbf{X}, \mathbf{Z}) = p(\mathbf{X}, \mathbf{Z} \mid \boldsymbol{\theta})$
the maximum likelihood estimate (MLE) of the unknown parameters is determined by maximizing the Marginal Likelihood of the observed data[8]:

$$L(\boldsymbol{\theta}; \mathbf{X}) = p(\mathbf{X} \mid \boldsymbol{\theta}) = \int p(\mathbf{X}, \mathbf{Z} \mid \boldsymbol{\theta})\, d\mathbf{Z}$$

The EM algorithm seeks to find the MLE of the marginal likelihood by iteratively applying these two steps[8]:

Expectation step (E step):
Define $Q(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(t)})$ as the expected value of the log-likelihood function of $\boldsymbol{\theta}$, with respect to the current conditional distribution of **Z** given **X** and current estimates of the parameter $\boldsymbol{\theta}^{(t)}$[8]:

$$Q(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(t)}) = \operatorname{E}_{\mathbf{Z}|\mathbf{X},\boldsymbol{\theta}^{(t)}}[\log L(\boldsymbol{\theta}; \mathbf{X}, \mathbf{Z})]$$

Maximization step (M step):
Find the parameters that maximize this quantity[8]:

$$\boldsymbol{\theta}^{(t+1)} = \underset{\boldsymbol{\theta}}{\arg\max}\, Q(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(t)})$$

2.3. EM Algorithm for GMM:

Expectation step (E step):
We can write the Gaussian Mixture distribution as a combination of Gaussians with weights equal to π as below, where K is the number of Gaussians we want to model[7]:

$$p(x) = \Sigma_{k=1}^{K} \pi_k N(x \mid \mu_k, \Sigma_k)$$

Taking the above results we can calculate the posterior distribution of the responsibilities that each Gaussian has for each data point using the formula below[7]. This equation is just Bayes rule where π is the prior weights and the likelihood is normal[7].

$$\gamma(z_{nk}) = \frac{\pi_k N(x_n|\mu_k, \Sigma_k)}{\Sigma_{j=1}^{K} \pi_j N(x_n|\mu_j, \Sigma_j)}$$

Maximization step (M step):
After calculating our posterior all we need to do is get an estimate of the parameters of each Gaussian defined by the equations below and then evaluate the log-likelihood. These two steps are then repeated until convergence[7]. The mean of the Gaussians Equations is as follows[7]:

$$\mu_k^{new} = \frac{1}{N_k} \Sigma_{n=1}^{N} \gamma(z_{nk}) x_n$$

The equation for the covariance of the Gaussians is as follows[7]:

$$\Sigma_k^{new} = \frac{1}{N_k} \Sigma_{n=1}^{N} \gamma(z_{nk})(x_n - \mu_k^{new})(x_n - \mu_k^{new})^T$$

The new weights are given as follows[7]:

$$\pi_k^{new} = \frac{N_k}{N}$$

The sum of Responsibilities in each Gaussian k is as follows[7]:

$$N_k = \Sigma_{n=1}^{N} \gamma(z_{nk})$$

The Marginal Likelihood function which is to be maximized is as follows[7]:

$$ln\, p(X|\mu, \Sigma, \pi) = \Sigma_{n=1}^{N} ln \left\{ \Sigma_{k=1}^{K} \pi_k N(x_n|\mu_k, \Sigma_k) \right\}$$

2.4. Principal Component Analysis(PCA):

Principal component analysis is a technique for feature extraction — so it combines our input variables in a specific way, then we can drop the "least important" variables while still retaining the most valuable parts of all of the variables![9].In this project , we have used PCA to reduce the 5 dimensional data to 2 dimensional data for visualizations purposes. We have considered Principal component 1 (PC1) and Principal Component 2 (PC2).

III. Results

The given dataset contained 5 columns of data(5D dataset). We were required to perform K-Means clustering for K=3,5 and 7 and GMM clustering using the EM technique for 3,5 and 7 clusters respectively. The results are as follows:

3.1.K-Means Results:

3.1.1. K=3

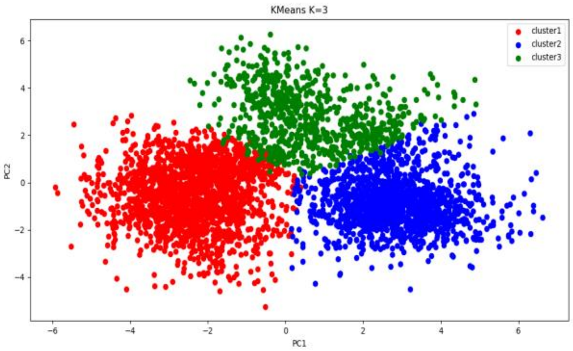

Figure 1: Plot of PC1 vs PC2 for K=3

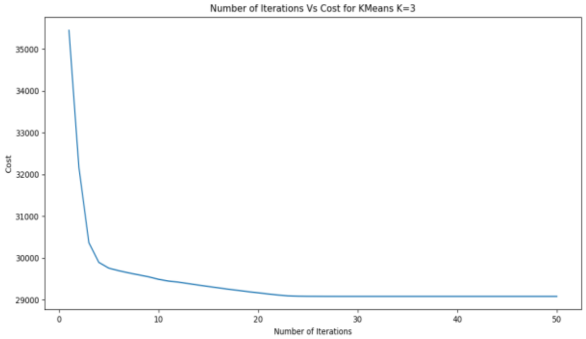

Figure 2: Plot ofthe number of iterations vs cost for K=3

3.1.2. K=5

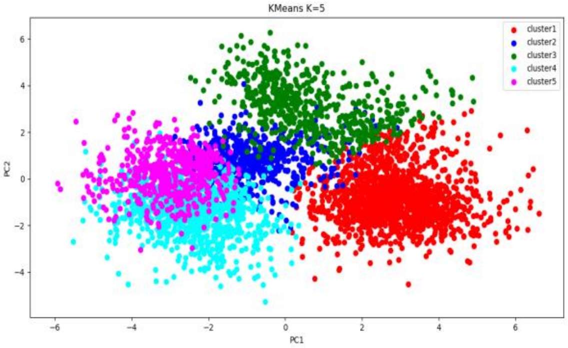

Figure 3: Plot of PC1 vs PC2 for K=5

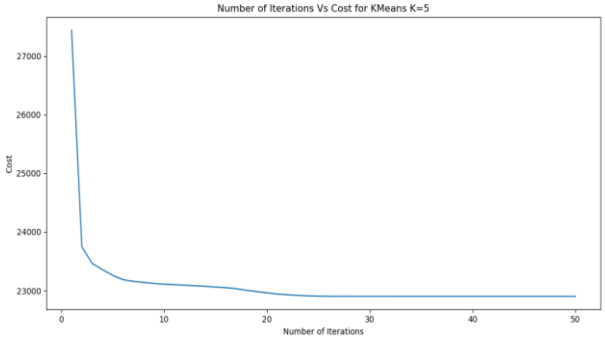

Figure 4: Plot of the number of iterations vs cost for K=5

3.1.3. K=7

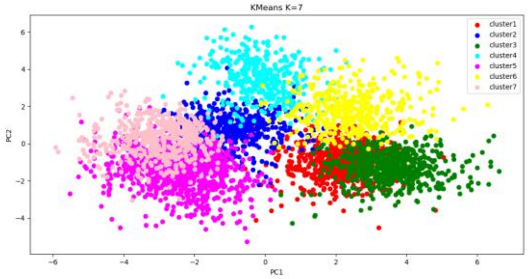

Figure 5: Plot of PC1 vs PC2 for K=7

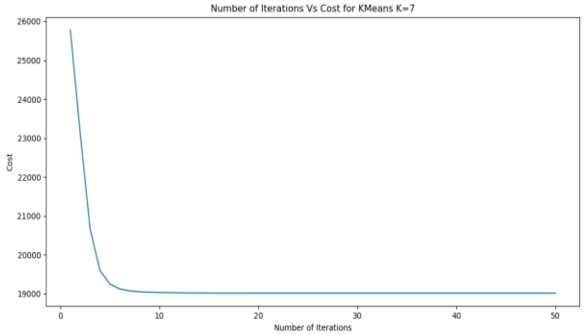

Figure 6: Plot of number of iterations vs cost for K=7

3.2.GMM Results:

3.1.1. K=3

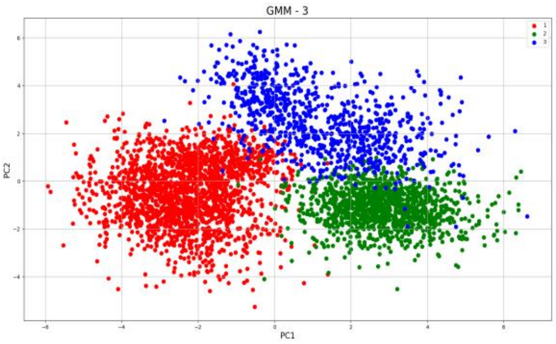

Figure 1: Plot of PC1 vs PC2 for K=3

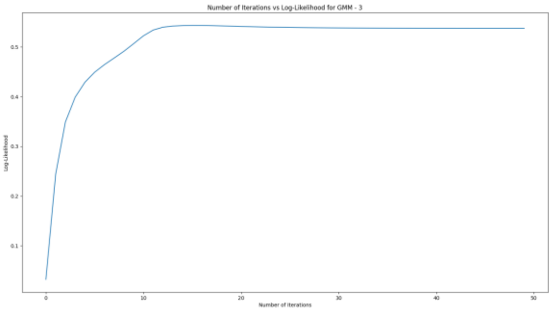

Figure 2: Plot of the number of iterations vs log-likelihood for K=3

3.1.2. K=5

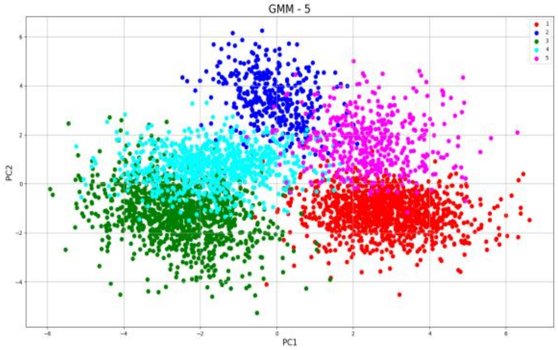

Figure 3: Plot of PC1 vs PC2 for K=5

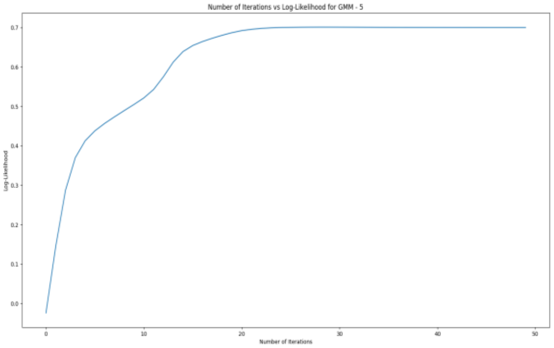

Figure 4: Plot of the number of iterations vs log-likelihood for K=5

3.1.3. K=7

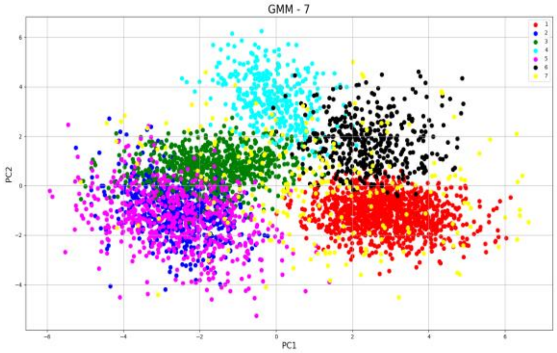

Figure 5: Plot of PC1 vs PC2 for K=7

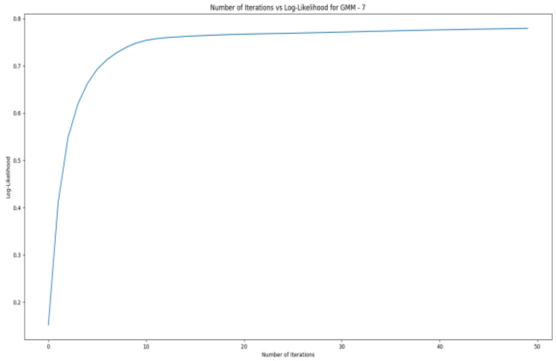

Figure 6: Plot of the number of iterations vs log-likelihood for K=7

IV. CONCLUSIONS

This assignment has focussed on using the following two Clustering Algorithms: 1. K-Means 2. EM Technique applied for GMM, for classifying the given dataset into clusters. The algorithms were used to classify the given dataset into 3,5 and 7 clusters respectively. Principal Component Analysis(PCA) was performed on the clustered dataset in order to visualize the results. The Clustered dataset was converted from five dimensions to Two dimensions and the results are as shown in the previous section.

REFERENCES

[1]https://www.mathworks.com/discovery/unsupervised-learning.html
[2]https://en.wikipedia.org/wiki/K-means_clustering
[3https://towardsdatascience.com/k-means-clustering-algorithm-applications-evaluation-methods-and-drawbacks-aa03e644b48a
[4]https://stanford.edu/~cpiech/cs221/handouts/kmeans.html
[5]https://blogs.oracle.com/datascience/introduction-to-k-means-clustering
[6]https://docs.rapidminer.com/latest/studio/operators/modeling/segmentation/expectation_maximization_clustering.html
[7]https://towardsdatascience.com/gaussian-mixture-modelling-gmm-833c88587c7f
[8]https://en.wikipedia.org/wiki/Expectation%E2%80%93maximization_algorithm
[9]https://towardsdatascience.com/a-one-stop-shop-for-principal-component-analysis-5582fb7e0a9c