S21CSE471: Statistical Methods in AI **Spring 2021**

Gaussian Mixture Model

Prepared by: Ankit Parashar, Kajal Sanklecha, Madhav Agarwal

1 Anomaly Detection with Single Gaussian

Anomaly detection is the process of identifying data points that deviates from the dataset's normal behaviour. It has several use-cases such as credit card fraud detection, identifying cyber attack etc. The traditional method to identify data anomaly is to fit a gaussian model. A Gaussian Model is given by:

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}}.e^{\frac{-(x-\mu)^2}{2\sigma^2}} \tag{1}$$

For a gaussian model to work properly, the classes in data must not be overlapping i.e. clusters are easily separable. However even in this scenario, using a single gaussian model for anomaly detection suffers from a major drawback.

Let us illustrate by an example.
Suppose we try to model a data distribution of user behaviour as shown in Figure: 1. The data consists of two distributions given in Red and Blue colour describing two groups of users. Suppose a new type of user behaviour is noticed and we mark it as green point. If we fit a single gaussian, then it will try to classify the green point as non-anomalous. Hence, we notice that single gaussian is not enough to detect anomaly. So, what should we do in this case? And, what if classes are not separable? The obvious answer is to use multi gaussian model or Gaussian Mixture Model. In the subsequent sections, we will discuss about GMM in more detail.

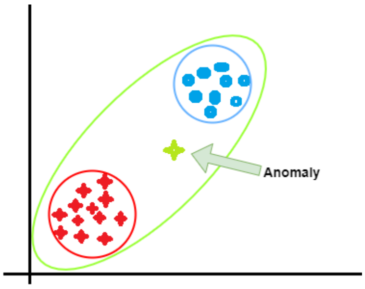

Figure 1: Single Gaussian

2 Gaussian Mixture Model

Gaussian Mixture Model is a soft-clustering algorithm. Unlike K-Means, which is a hard-clustering algorithm, GMM can works well even when the clusters are overlapping. GMM works better because it not only uses the cluster mean but also their covariance.

Gaussian Mixture is nothing but the weighted sum of gaussians as shown in Figure2. For a gaussian 'N' having μ_c (define the centre of gaussian) and σ_c (define the spread of a gaussian), GMM is given by:

$$f(x) = \sum_c \pi_c N(x_i; \mu_c, \sigma_c) \tag{2}$$

where π_c is the weight of N_c gaussian. It defines how big or small a gaussian function is. The weight coffecient are probabilistic and hold:

$$\sum \pi_c = 1 \tag{3}$$

For three gaussians, a GMM can be illustrated as:

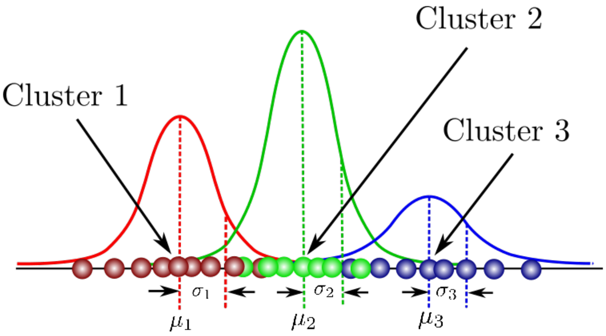

Figure 2: Gaussian Mixture Model[1]

Let us examine the idle scenarios in which we can use Gaussian mixture models.

Scenario 1: When source is given.
If we have data points with their cluster label, we can estimate the respective gaussians. It can be done by calculating the mean and variance of each cluster and fit the gaussian using Equation 1.

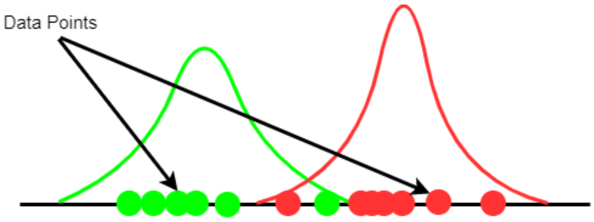

Figure 3: Data Points with labels are given

Scenario 2: When Gaussians are given.
The gaussians are given i.e. we have mean (μ) and standard deviation (σ) of each gaussian as shown in Figure 4.

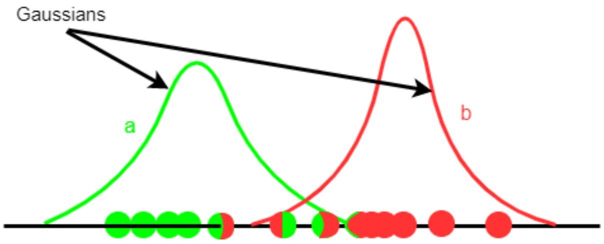

Figure 4: Gaussians are given

Suppose we have two gaussians, a(green) and b(red). For each data point x_i, We can estimate the source or label by using:

$$P(x_i|b) = \frac{1}{\sqrt{2\pi\sigma_b^2}}.e^{\dfrac{-(x_i-\mu_b)^2}{2\sigma_b^2}} \tag{4}$$

Now, using P(x_i|b), we can estimate P(b|x_i) i.e. the probability that point x_i came from the distribution 'b'.

$$b_i = P(b|x_i) = \frac{P(x_i|b).P(b)}{P(x_i|b).P(b) + P(x_i|a).P(a)} \tag{5}$$

where P(a) and P(b) are known as prior given by: P(a) = $\sum a_i$/N and P(b)= $\sum b_i$/N.
Similarly we can estimate P(x_i|a) followed by P(a|x_i) or a_i. These can now be use to estimate the class of a given data point.

However, In real world, we don't have either Scenario1 or Scenario2. We have only points. To find GMM in such case, we use Expectation–Maximization (EM) algorithm.

3 Expectation–Maximization (EM) algorithm

EM algorithm is used to estimate the parameters in GMMs in an iterative manner. Let the parameters of the model be

$$\Theta = \{\pi, \mu, \Sigma\} \tag{6}$$

Following are the steps which an EM algorithm follows:

Step 1: Initialise the parameters randomly. For instance, we can use the results obtained by a previous K-Means run as a good starting point for our algorithm.

Step 2(Expectation): For each point x_i, compute

$$P(x_i|a) = \frac{1}{\sqrt{2\pi\sigma_a^2}}.e^{\dfrac{-(x_i-\mu_a)^2}{2\sigma_a^2}} \tag{7}$$

$$a_i = P(a|x_i) = \frac{P(x_i|a).P(a)}{P(x_i|a).P(a) + P(x_i|b).P(b)} \tag{8}$$

$$a_i = 1 - b_i \tag{9}$$

The above equations can also be generalized for k more than 2 where all the probabilities will sum up to 1.

Step 3 (Maximization): Find the revised parameters using

$$\Theta * = argmax_{\Theta} Q(\Theta *, \Theta) \tag{10}$$

which adjusts the parameters for the Gaussians to fit the points assigned to them.

The revised mean and variance for any Gaussian is calculated as follows:

$$\mu_b = \frac{b_1 * x_1 + b_2 * x_2 + ... + b_N * x_N}{b_1 + b_2 + ... + b_N} \tag{11}$$

$$\sigma_b^2 = \frac{b_1 * (x_1 - \mu_b)^2 + b_2 * (x_2 - \mu_b)^2 + ... + b_N * (x_N - \mu_b)^2}{b_1 + b_2 + ... + b_N} \tag{12}$$

$$P(a) = \pi_a = \frac{a_1 + a_2 + ... + a_N}{N} \tag{13}$$

$$P(b) = \pi_b = \frac{b_1 + b_2 + ... + b_N}{N} \tag{14}$$

EM Algorithm: Working in 1D
The working of EM algorithm on a dataset having two classes is illustrated in the figure below. It shows how repeated iteration of EM algorithm fits two gaussians and provide a soft-clustering.

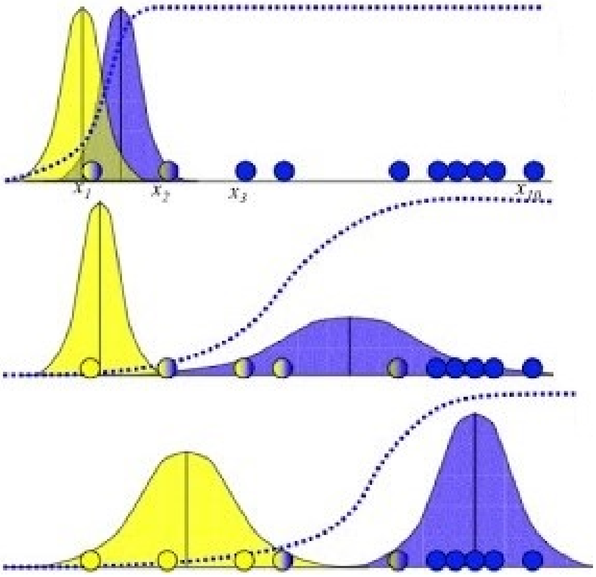

Figure 5: EM Algorithm in 1D [2]

Points to Consider:

- Is there a better way to initialize rather than random initialization? We can use the results obtained in k-means algorithm.
- How many Gaussians to start with the EM algorithm / What are the optimal number of clusters? The best method of get optimal number of clusters is using Silhouette score[3]. It helps in checking how much clusters are compact and well separated. The EM algorithm is fitted for each of the k

clusters (generally k=2 to k=20). The fit with best Silhouette score is considered for the optimal number of clusters. Another commonly used method is Bayesian information criterion [4]. It prevents over-fitting by penalizing large clusters. The lowest BIC value is considered for selecting optimal number of clusters.

- Point of convergence? EM algorithm is not guaranteed to converge to a local minimum. It is only guaranteed to converge to a point with zero gradient with respect to the parameters. So it can indeed get stuck at saddle points.

Sampling: The final function is a single function which is a convex combination of set of Gaussians. the function is given by:

$$f(x) = \sum \pi_c N(x_i, \mu_c, \sigma_c) \tag{15}$$

where π_c are weights of the corresponding Gaussians.

Let there are c Gaussians where π_1, π_2, π_3, ... , π_c are the corresponding weights of the Gaussians. They represent how likely the point is generated from the c_{th} Gaussian, and holds:

$$\sum \pi_c = 1 \tag{16}$$

To perform sampling, first we select a Gaussian and find the weight π corresponding to that Gaussian. Then we sample a point from that Gaussian. So, it is a two level sampling process.

References

[1] Gaussian Mixture Model Explained
`https://towardsdatascience.com/gaussian-mixture-models-explained-6986aaf5a95`

[2] Understanding Expectation Maximization and Soft Clustering `https://medium.com/@thiagoricieri/understanding-expectation-maximization-and-soft-clustering-4645e997cdb6`

[3] Silhouette Clustering
`https://en.wikipedia.org/wiki/Silhouette_(clustering)`

[4] Bayesian information criterion
`https://en.wikipedia.org/wiki/Bayesian_information_criterion`