DSA Project Report

Ashwin Kumar (CE21BTECH11008)—Aakash Loyar (EP21BTECH11001)

May 1, 2024

Abstract

The onset of COVID-19, stemming from a novel coronavirus, has precipitated widespread disruption across nations globally. Comparable to historical pandemics such as the Black Death and the Spanish Flu, its profound impact has been felt on a global scale. Characterized by its high infectivity, COVID-19 precipitated stringent containment measures during its zenith, compelling individuals to reside within restricted environments. Medical professionals encountered arduous challenges as they endeavored to preserve a vast populace amidst the crisis, while the global economy bore the brunt of its ramifications.

Our predictive model demonstrates notable efficacy in forecasting forthcoming waves of COVID-19 based on historical data. Additionally, we have undertaken a comprehensive analysis of various parameters to facilitate optimal medical resource allocation in critical scenarios. Leveraging diverse mathematical models including Gaussian Mixture Models (GMM) and classification techniques such as Latent Semantic Analysis (LSA) and Expectation-Maximization (EM), our methodologies offer insights crucial for strategic decision-making and crisis management.

1 Introduction

COVID-19 is a viral disease quickly spread in the world.The first case appeared in china then it spread in the whole world. It has symptoms include fever, difficulty in breathing, cough. This virus has infected millions of people globally. Many people lost their life, livelihoods and they struggle to survive for basic requirements. It can survive for a week on a surface. COVID-19 disease was highly contagious and can spread by coming in contact with an infected person or even by touching any common surfaces. The model is in the form of mathematical series with different parameters to account for various physical phenomena dictating the count of people getting infected by the virus. The model estimation is done mainly on India. Prediction is performed using the Gaussian mixture model curve-fitting approach to predict the total number of cases and the end-dates (occurrence of 99% of the total expected cases) for the disease in various parts of the world. It also compares with the actual cases that were seen on the predicted days. After that, the least squares approach of classification has been used to show the data can partitioned on the basis of different factors and at last used Expectation Maximization on generated dataset that can be applied similarly to the COVID dataset.

2 Methodology

2.1 Gaussian Mixture Models (GMMs)

A mixture of various Gaussians can be indexed using $k \in \{1, 2, \ldots, K\}$ where K is the number of clusters, or in this case, the number of Gaussians. This very simply relates to the K-Means algorithm; however, it differs from the K-Means Algorithm in the manner that it is a soft clustering problem, i.e., each point is assigned a probability of belonging to a certain cluster. Moreover, we can show that K-means is a special case of GMMs. Each Gaussian k in the GMM consists of the following:

1. μ_k: the mean of the Gaussian

2. Σ_k: the covariance of the Gaussian

3. π_k: the contribution of the Gaussian to the mixture. Note that $\pi_k \in [0, 1]$ and $\sum_{k=1}^{K} \pi_k = 1$.

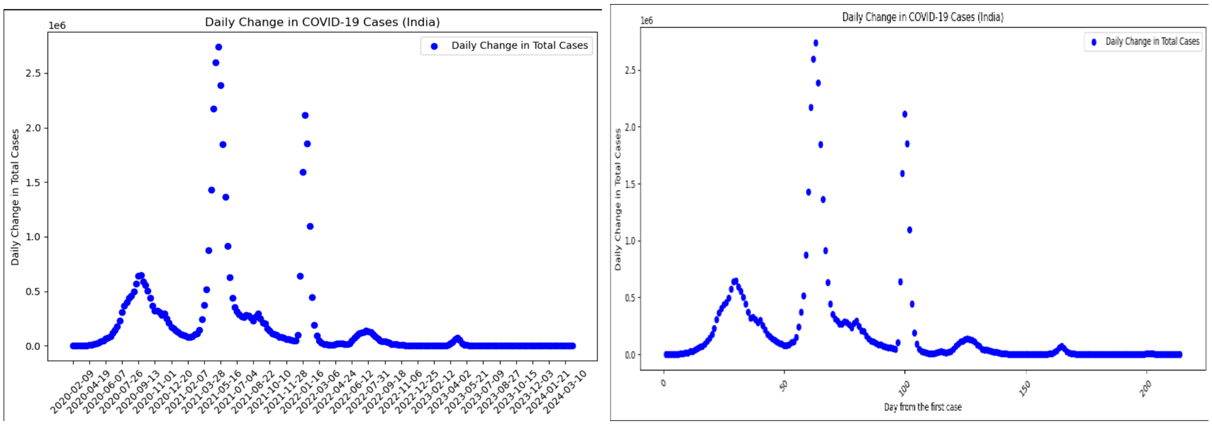
Thus, our Gaussian Mixture Model shall look as follows:

$$Pr(x|\theta) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x|\mu_k, \Sigma_k) \tag{1}$$

where $\theta = \{\mu_1, \Sigma_1, \pi_1, \ldots, \mu_K, \Sigma_K, \pi_K\}$ are all the parameters. Note that the number of clusters K is not a learnable parameter and is given as an input (i.e., hyperparameter).

The data given was for all the country and there were many unnecessary column that we have not used due to complexity.We have extracted two column location and total active cases for location. Then for given country differentiate the data. In this project mainly focus on the data of India.

Then found the daily changes from the data by simple array substraction.This will essentially give number of cases per day.There were many data that were not present so removed those column that was containing "nan". Also removed the value equal to 0 as it may possible hinder the graph.Then simply plotted the COVID cases with the date.Plot is clearly showing three peak that signifies three peak of COVID came in India with second peak was more brutal when daily cases increased more than 10 lakhs.



(a) Daily Changes Vs Date

(b) Daily Changes VS Day Number

Figure 1: GMM fit

This graph show 1st wave came around june 2020 , 2nd wave came around march 2021 and 3rd wave around december 2021.

2.2 Maximum Likelihood Estimation (MLE)

If we are simply modeling the data to a single Gaussian, then we can use the Maximum Likelihood Estimator. Let the Gaussian's mean and standard deviation be μ and σ, respectively. Likelihood is defined as the probability of seeing the given data, provided we assume the model parameters are correct. By the product rule of probability, the total likelihood is the product of the individual likelihoods of all N data points, given by:

$$L = \prod_{i=1}^{N} Pr(x_i|\mu, \sigma) = \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x_i-\mu)^2}{2\sigma^2}} \tag{2}$$

Maximizing the likelihood is equivalent to minimizing the negative log-likelihood, which means our optimal parameters μ^*, σ^* are given by:

$$\mu^*, \sigma^* = \arg\min_{\mu,\sigma} \sum_{i=1}^{N} \frac{1}{2}\left(\ln 2\pi\sigma^2 + \frac{(x_i-\mu)^2}{2\sigma^2}\right) \tag{3}$$

Simply setting the partial derivatives to 0, we get:

$$\frac{\partial \ln L}{\partial \mu} = 0 \Rightarrow \mu^*_{MLE} = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{4}$$

$$\frac{\partial \ln L}{\partial \sigma} = 0 \Rightarrow \sigma^*_{MLE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2} \tag{5}$$

The graph have idea that 3 peaks are there so GMM with 3 component will fit better.So, for this converted the date into days taking first date as reference and then generating frequency of each day same as number of daily cases corresponding to that day and then applied the one dimensional Gaussian Mixture Model with 3 component. 3 GMM looks fitting the changes perfectly.

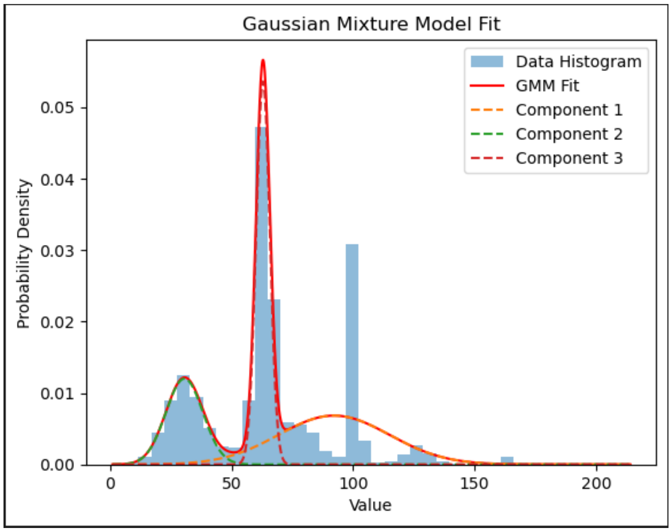


Figure 2: 3 component Gaussian Mixture Model Fit to COVID Data

2.3 Least square approach classification

Least squares classification is a technique used to classify data points into different classes based on their features. It is often employed in machine learning and statistics. The goal is to find the decision boundaries that best separate the classes by minimizing the sum of squared errors between the actual class labels and the predicted class labels.

The basic idea behind least squares classification is to find the coefficients of a linear equation that best fits the data points to their respective classes. This linear equation can be represented as:

$$y = \mathbf{w}^T\mathbf{x} + b$$

Where: - y is the predicted class label. - $\mathbf{w}$ is the weight vector. - $\mathbf{x}$ is the feature vector of the data point. - b is the bias term.

The decision boundary is defined by the hyperplane where $y = 0$. Data points on one side of the hyperplane are classified as belonging to one class, while data points on the other side are classified as belonging to the other class.

The coefficients $\mathbf{w}$ and b are typically found by minimizing the sum of squared errors between the actual class labels and the predicted class labels. This can be formulated as an optimization problem and solved using techniques such as gradient descent or the normal equation.

Once the coefficients are found, new data points can be classified by plugging their feature vectors into the linear equation and checking which side of the decision boundary they fall on.

Here's the least squares classification equation in matrix form:

$$\mathbf{Y} = \mathbf{X}\mathbf{W} + \mathbf{B}$$

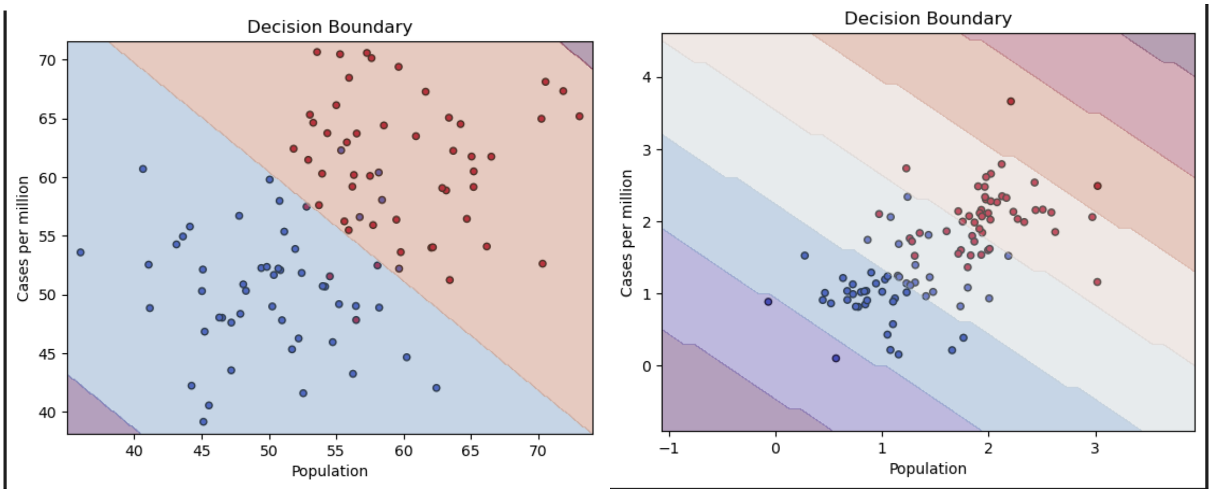
Where: - $\mathbf{Y}$ is a vector of predicted class labels. - $\mathbf{X}$ is a matrix of feature vectors, with each row representing a data point. - $\mathbf{W}$ is the weight matrix. - $\mathbf{B}$ is a vector of bias terms.

To find $\mathbf{W}$ and $\mathbf{B}$, the following optimization problem is typically solved:

$$\min_{\mathbf{W},\mathbf{B}} ||\mathbf{Y} - \mathbf{X}\mathbf{W} - \mathbf{B}||_2^2$$

Where $|| \cdot ||_2$ denotes the L2 norm.

Once $\mathbf{W}$ and $\mathbf{B}$ are found, new data points can be classified using the equation $\mathbf{Y} = \mathbf{X}\mathbf{W} + \mathbf{B}$, where the sign of the elements of $\mathbf{Y}$ determine the class labels.



(a) Decision Boundary (magnified)

(b) Decision Boundary all

Figure 3: Division of Data in Different Decision Boundary

2.4 Expectation Maximization Algorithm (EM Algorithm)

The Expectation-Maximization Algorithm or the EM Algorithm is an iterative solution to finding the optimal parameters. It consists of two steps: the E step or the Expectation step and the M step or the Maximization step. Both these steps are repeated iteratively till there is some convergence in the parameter space.

Let us define a latent variable $\mathbf{z} = (z_1, z_2, \ldots, z_K)$, where $z_k = 1$ if data point belongs to cluster k and 0 otherwise. Here, $\mathbf{z}$ is a K-dimensional binary random vector. We can also state the following:

$$\pi_k = Pr(z_k = 1) \tag{6}$$

which means the probability of obtaining a data point from a particular Gaussian is equivalent to its contribution to the mixture. We have:

$$Pr(\mathbf{z}) = \prod_{k=1}^{K} \pi_k^{z_k} \tag{7}$$

Also, the probability of obtaining a data sample x_i from a cluster k is given by:

$$Pr(x_i|z_k) = \begin{cases} 0 & \text{if } z_k = 0 \\ \mathcal{N}(x_i; \mu_k, \Sigma_k) & \text{if } z_k = 1 \end{cases} \tag{8}$$

We can compute the joint probability as:

$$Pr(x_i, \mathbf{z}) = \prod_{k=1}^{K} (\pi_k N(x_i; \mu_k, \Sigma_k))^{z_k} \tag{9}$$

Thus, given the latent variable $\mathbf{z} = (z_1, z_2, \ldots, z_K)$, we have:

$$Pr(x_i|\mathbf{z}) = \prod_{k=1}^{K} N(x_i; \mu_k, \Sigma_k)^{z_k} \tag{10}$$

The likelihood of the data point can be found by summing over all possible states of $\mathbf{z}$. We shall find the likelihood of the total data X as the probability of seeing the data point over all possible clusters:

$$Pr(X) = \prod_{i=1}^{N} Pr(x_i) = \prod_{i=1}^{N} \sum_{k=1}^{K} \pi_k N(x_i; \mu_k, \Sigma_k) \tag{11}$$

which is precisely the same as Equation 8. We have one latent vector $\mathbf{z}_i$ per data point x_i. Let Z be all the latent vectors. If we knew Z, we could use MLE as we would know which data point belongs to which cluster and then estimate the parameters of each cluster. However, in this case, we only know X.

Note that we also have some additional information now. We can determine $Pr(z_k = 1|x_i)$, which is the probability of a data point x_i belonging to a certain cluster k given by:

$$Pr(z_k = 1|x_i) = \frac{Pr(x_i|z_k = 1)Pr(z_k)}{Pr(x_i)} \tag{12}$$

$$= \frac{\pi_k \mathcal{N}(x_i; \mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(x_i; \mu_j, \Sigma_j)} \tag{13}$$

$$= \gamma(z_{ik}) \tag{14}$$

where $\gamma(z_{ik})$ is the responsibility of cluster k for data point x_i.

Now, let's take the partial derivatives of the Likelihood with respect to μ_k, which shall be exactly equal to Equation 11:

$$\sum_{i=1}^{N} \left[\frac{\pi_k \mathcal{N}(x_i; \mu_k, \Sigma_k)}{\sum_{k=1}^{K} \pi_k \mathcal{N}(x_i; \mu_k, \Sigma_k)} \Sigma_k^{-1}(x_i - \mu_k) \right] = 0 \tag{15}$$

$$\Rightarrow \sum_{i=1}^{N} \gamma(z_{ik}) \Sigma_k^{-1}(x_i - \mu_k) = 0 \tag{16}$$

$$\Rightarrow \mu_k^* = \frac{\sum_{i=1}^{N} \gamma(z_{ik}) x_i}{\sum_{i=1}^{N} \gamma(z_{ik})} \tag{17}$$

where we assume the covariance matrix Σ_k is non-singular.

Similarly, we can obtain:

$$\Sigma_k^* = \frac{\sum_{i=1}^{N} \gamma(z_{ik})(x_i - \mu_k)(x_i - \mu_k)^T}{\sum_{i=1}^{N} \gamma(z_{ik})} \tag{18}$$

We also wish to obtain π_k. We can simply differentiate with respect to π_k. However, we need to keep the constraint that $\sum_{k=1}^{K} \pi_k = 1$. This can be done using Lagrange Multipliers and maximizing the following quantity:

$$\ln L + \lambda \left(\sum_{k=1}^{K} \pi_k - 1 \right) \tag{19}$$

which gives:

$$\sum_{i=1}^{N}\left[\frac{\mathcal{N}(x_i;\mu_k,\Sigma_k)}{\sum_{k=1}^{K}\pi_k\mathcal{N}(x_i;\mu_k,\Sigma_k)}+\lambda\right]=0 \tag{20}$$

$$\Rightarrow \pi_k\lambda=-\sum_{i=1}^{N}\gamma(z_{ik}) \tag{21}$$

$$\Rightarrow \sum_{k=1}^{K}\pi_k\lambda=-\sum_{k=1}^{K}\sum_{i=1}^{N}\gamma(z_{ik}) \tag{22}$$

$$\Rightarrow \lambda=\sum_{k=1}^{K}\sum_{i=1}^{N}\gamma(z_{ik}) \tag{23}$$

$$\Rightarrow \pi_k^*=\frac{\sum_{i=1}^{N}\gamma(z_{ik})}{\sum_{k=1}^{K}\sum_{i=1}^{N}\gamma(z_{ik})} \tag{24}$$

Thus, our Expectation-Maximization Algorithm is as given in Algorithm 1:

In multivariate statistics, covariance matrices describe the relationships between variables. Full covariance represents all pairwise covariances, given by

$$\operatorname{cov}(X_i,X_j)=\frac{1}{N}\sum_{k=1}^{N}(x_{ik}-\bar{x_i})(x_{jk}-\bar{x_j})$$

, forming an $n\times n$ matrix. Spherical covariance assumes equal variances (σ^2) and zero covariances, resulting in a diagonal matrix $\Sigma=\sigma^2 I_n$, where I_n is the identity matrix. Diagonal covariance allows individual variances (σ_i^2) with no covariances, creating a diagonal matrix $\Sigma=\operatorname{diag}(\sigma_1^2,\sigma_2^2,\ldots,\sigma_n^2)$. These structures balance complexity and parsimony in modeling multivariate data, offering insights into the dependencies and variations among variables while facilitating efficient computations.

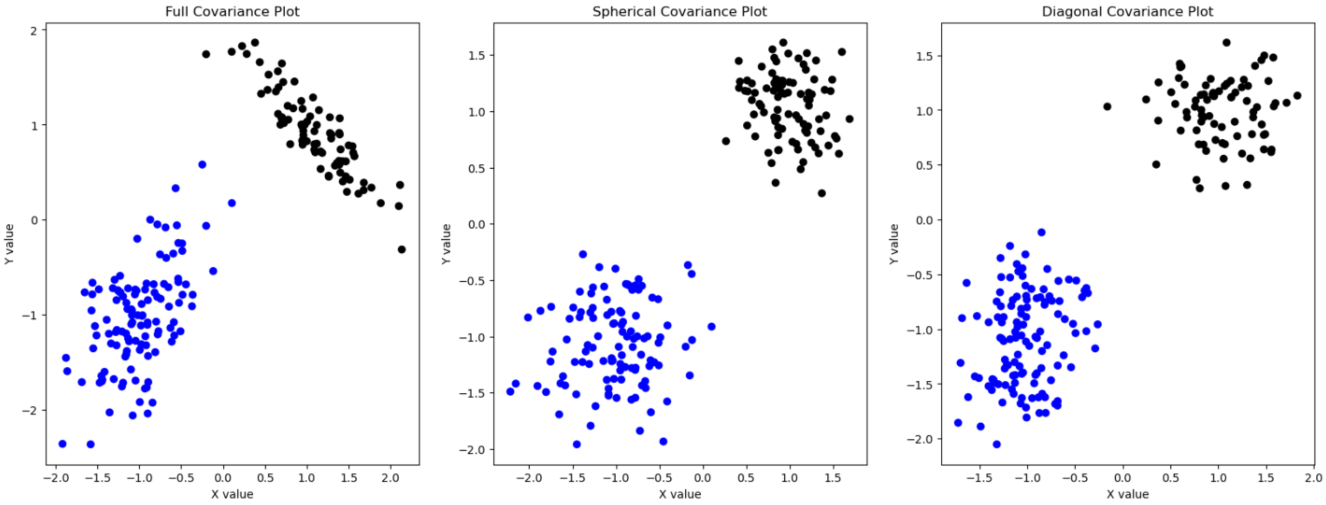


Figure 4: Showing Various Covariance Plot

2.5 Algorithm 1: Expectation Maximization Algorithm

1. Initialize the means μ_k, the covariance matrix Σ_k, and the contributions π_k.

2. While convergence is not attained, do

   (a) Calculate $\gamma(z_{ik})=\pi_k\mathcal{N}(x_i;\mu_k,\Sigma_k)/\sum_{j=1}^{K}\pi_j\mathcal{N}(x_i;\mu_j,\Sigma_j)$.

(b) Re-evaluate μ_k, Σ_k, and π_k as follows:

$$\mu_k = \frac{\sum_{i=1}^{N} \gamma(z_{ik}) x_i}{\sum_{i=1}^{N} \gamma(z_{ik})} \tag{25}$$

$$\Sigma_k = \frac{\sum_{i=1}^{N} \gamma(z_{ik})(x_i - \mu_k)(x_i - \mu_k)^T}{\sum_{i=1}^{N} \gamma(z_{ik})} \tag{26}$$

$$\pi_k = \frac{\sum_{i=1}^{N} \gamma(z_{ik})}{\sum_{k=1}^{K} \sum_{i=1}^{N} \gamma(z_{ik})} \tag{27}$$

(c) Compute the log-likelihood and check for convergence.

There is an alternative view of the EM algorithm. Since we only have the data X, we cannot compute the complete log-likelihood $Pr(X, Z)$; however, we can compute the expectation of this log-likelihood under the posterior distribution of Z. This refers to the E-Step or the Expectation Step. Under the M-Step or the Maximization Step, we try to maximize this expectation. We define:

$$Q(\theta^*, \theta) = E[\ln Pr(X, Z|\theta^*)] \tag{28}$$

$$= \sum_{Z} Pr(Z|X, \theta) \ln Pr(X, Z|\theta^*) \tag{29}$$

shown to be the lower bound on the gain in Likelihood over successive iterations. However, we shall skip the proof here. We can find the joint probability using Equation 15 as:

$$Pr(X, Z) = \prod_{i=1}^{N} \prod_{k=1}^{K} \pi_{zik} \mathcal{N}(x_i; \mu_k, \Sigma_k)^{zik} \tag{30}$$

$$\text{Taking the logarithm, we get:} \tag{31}$$

$$\ln Pr(X, Z) = \sum_{i=1}^{N} \sum_{k=1}^{K} \ln z_{ik} + \ln(\pi_k \mathcal{N}(x_i; \mu_k, \Sigma_k)) \tag{32}$$

Substituting Equation 19 and Equation 33 into Equation 31, we get:

$$Q(\theta^*, \theta) = \sum_{i=1}^{N} \sum_{k=1}^{K} \gamma(z_{ik}) \ln(\pi_k \mathcal{N}(x_i; \mu_k, \Sigma_k)) \tag{33}$$

where we can get rid of z_{nk} using the fact that for an entire z, it is equal to 1 only once.For maximization step,simply set

$$\theta^* = \arg\max_{\theta} Q(\theta^*, \theta) \tag{34}$$

On taking partial derivatives of Equation 31, we shall obtain the same results as Equation 22, Equation 23, and Equation 29.

2.6 Algorithm 2: Expectation Maximization Algorithm

1. Initialize the initial parameters θ_{old} which consist of means μ_k, the covariance matrix Σ_k, and the contributions π_k.

2. While convergence is not attained, do

   (a) Calculate $Pr(Z|X, \theta_{\text{old}})$.

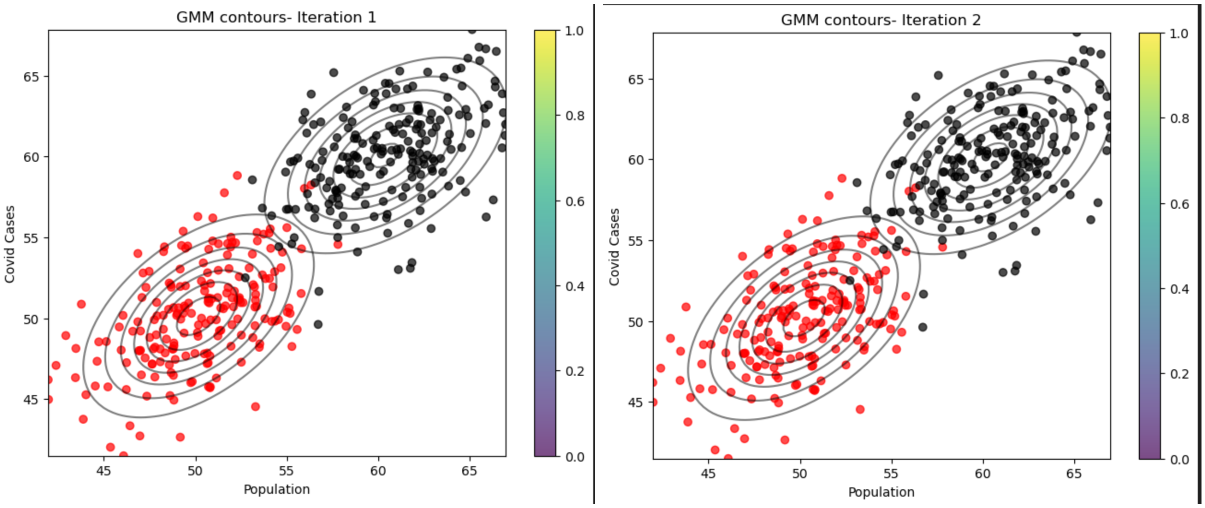
(b) Evaluate $\theta_{\text{new}} = \arg\max_\theta Q(\theta, \theta_{\text{old}})$ where $Q(\theta, \theta_{\text{old}}) = \sum_Z Pr(Z|X, \theta) \ln Pr(X, Z|\theta_{\text{old}})$ using the following:

$$\mu_k = \frac{\sum_{i=1}^{N} \gamma(z_{ik}) x_i}{\sum_{i=1}^{N} \gamma(z_{ik})} \tag{35}$$

$$\Sigma_k = \frac{\sum_{i=1}^{N} \gamma(z_{ik})(x_i - \mu_k)(x_i - \mu_k)^T}{\sum_{i=1}^{N} \gamma(z_{ik})} \tag{36}$$

$$\pi_k = \frac{\sum_{i=1}^{N} \gamma(z_{ik})}{\sum_{k=1}^{K} \sum_{i=1}^{N} \gamma(z_{ik})} \tag{37}$$

(c) Compute the log-likelihood and check for any convergence of parameters. Assign $\theta_{\text{old}} = \theta_{\text{new}}$.



(a) First iteration fit

(b) Second Iteration Fit

Figure 5: Expectation Maximization with confidence interval show with contour

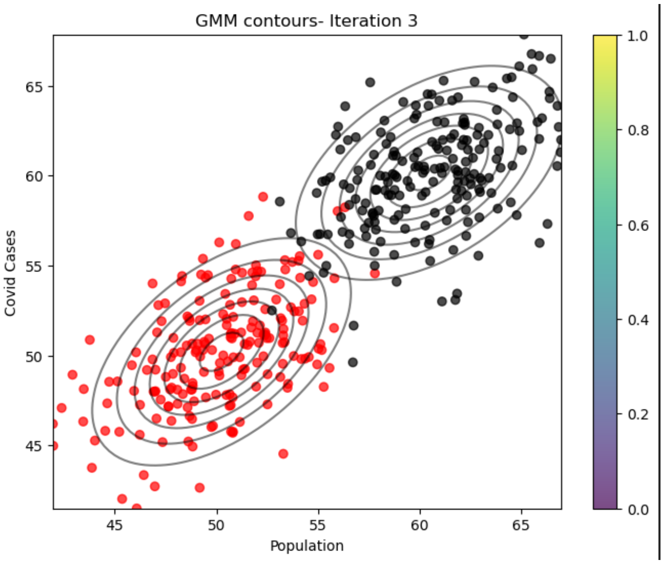


Figure 6: Third Iteration fit

CONCLUSION

In conclusion, our DSA Project Report offers a comprehensive analysis and predictive modeling of the COVID-19 pandemic, providing essential insights for strategic decision-making and crisis management. The profound impact of COVID-19 has necessitated innovative approaches akin to historical pandemics like the Black Death and the Spanish Flu.

Utilizing mathematical models such as Gaussian Mixture Models (GMMs), and classification techniques like Maximum Likelihood Estimation (MLE) and Expectation Maximization (EM), we effectively forecasted upcoming waves of COVID-19 based on historical data. These methodologies not only facilitate prediction but also enable a thorough analysis of critical parameters for optimal medical resource allocation.The project initiated with a detailed introduction to COVID-19, highlighting its rapid spread and global impact. Our predictive model, primarily focused on India, accurately forecasted total case numbers and end-dates of outbreaks worldwide, employing Gaussian mixture model curve-fitting.We identified three distinct waves of COVID-19 in India, with the second wave being notably severe. Through Maximum Likelihood Estimation (MLE), we demonstrated the effectiveness of a single Gaussian model in capturing case distributions, providing valuable insights into pandemic dynamics. Furthermore, we explored the application of the least squares approach classification to identify decision boundaries crucial for understanding COVID-19 spread. Additionally, the Expectation Maximization Algorithm facilitated accurate parameter estimation, contributing to our understanding of COVID-19 data structure.

Our project, through diverse methodologies, offers valuable insights into COVID-19 dynamics. By leveraging mathematical models and statistical techniques, we contribute to crisis management and strategic decision-making, paving the way for effective response and mitigation strategies against future pandemics.

Self Implementation

Please refer to GitHub

References

1. WORLDOMETER COVID-19 DATA .

2. Text classification: A least square support vector machine approach.