Software Requirements Specification for Guassian Mixture Model based on EM algorithm

WONG, Kim Ying

April 18, 2024

Contents

Revision History

Date	Version	Notes
1 Feb 2024	1.0	SRS First Draft
10 April 2024	2.0	SRS Second Draft
16 April 2024	3.0	SRS Final Version

1 Reference Material

This section records information for easy reference.

1.1 Table of Units

Throughout this document, we would consider all values to be unit-free. The rationale behind is that all units used in this document or model depend on the input dataset, while there is no general constraint on the unit of input. Therefore the input and the derived the statistical relation could be any units.

1.2 Table of Symbols

The table that follows summarizes the symbols used in this document along with their units (if exists). The choice of symbols was made to be consistent with the statistics literature and with existing documentation for the Gaussian Mixture Model. The symbols are listed in alphabetical order.

symbol	Description
$\mathbf{X}$	Dataset $\in \mathbb{R}^{M \times N}$
N	number of data-points $\in \mathbb{N}$
M	number of predictor variables $\in \mathbb{N}$
K	number of clusters needed $\in \mathbb{N}$
z	latent variable in GMM
l	labels for each data points
k	index for the corresponding cluster
n	index for the corresponding data points
$\mathbf{\Sigma}$	covariance
μ	mean
π	mixing coefficient
γ	responsibility
$\mathbf{\Sigma}_k$	covariance matrix for k^{th} cluster
μ_k	mean vector for k^{th} cluster
π_k	mixing coefficient for k^{th} cluster
γ	responsibility

1.3 Abbreviations and Acronyms

symbol	description
A	Assumption
DD	Data Definition
GD	General Definition
GS	Goal Statement
IM	Instance Model
LC	Likely Change
PS	Physical System Description
R	Requirement
SRS	Software Requirements Specification
TM	Theoretical Model
GMM	Gaussian Mixture Model
EM	Expectation Maximization
GMM-EM	Software Name: Gaussian Mixture Model based on EM Algorithm

1.4 Mathematical Notation

We borrow the convention for mathematical notation from the book Pattern Recognition and Machine Learning. Bishop (2006)
Set notation to represent the number system:

symbol	Description
$\mathbb{R}$	Real numbers
$\mathbb{R}_0^+$	non-negative real numbers
$\mathbb{Z}^+$	Positive integers

Notation to represent the statistics:

Notation	Description
$\mathcal{N}(\mu, \Sigma)$	Gaussian distribution with mean μ and covariance Σ
$p(x)$	Probability density function
$p(x\|z)$	Condition probability for x with z
$\mathcal{N}(x\|\mu, \Sigma)$	Conditional distribution for data point x under a given Gaussian distribution

These notation could be generalized to higher dimension by replacing real-valued numbers with vectors or matrices.
Typographic Conventions:

Notation	**Description**
x	a single data point
$\mathbf{x}$	Lowercase bold roman letters to represent vector
$\mathbf{X}$	Matrices are denoted by uppercase bold roman letters
$\mathbf{x}_n$	n^{th} column vector in a matrix $\mathbf{X}$

2 Introduction

Mixture model is a probabilistic model that describes the datasets or measurement in a linear combination of some basic distributions. Gaussian mixture model falls into a subset of the mixture model, where Gaussian distribution is used as a basis. In general, almost all continuous density could be approximated as sufficient number of Gaussian mixtures with appropriate mean and covariance.
Therefore, it can be used in various cases in machine learning, such as clustering and density estimation. The GMM-EM aims at implementation of the Gaussian mixture model for clustering with the Expectation-Maximization algorithm (EM Algorithm). This section is divided into 4 parts:

2.1 Purpose of Document

The document serves as a outline for the requirement specification for the GMM-EM. The reader could refer to this document for the general and specific system description. Theoretical model and their assumption will be detailed in this document.

2.2 Scope of Requirements

The GMM-EM would be a general library that implements the Gaussian mixture model based on the EM algorithm with a fundamental focus on the clustering problem. We will only deal with datasets that solely contain real numbers. The software would guarantee a convergence in solution but not optimality.

2.3 Characteristics of Intended Reader

The intended readers should acquire basic understanding on linear algebra, calculus and statistics. Take McMaster University courses as examples, MATH 1B03 (Linear Algebra 1), MATH 1AA3 (Calculus II) and STATS 1LL3 (Introduction to Probability and Statistics) should fulfill the requirements. Readers preferably have exposure on the basic statistical learning algorithm. A graduate introduction level course , for example , STATS 790 (Statistical Learning) should satisfy the requirement.

2.4 Organization of Document

The document consist of 5 parts:

- Section 3 General System Description
- Section 4 Specific System Description
- Section 5 Requirements
- Section 6 Likely Changes and Section 7 Unlikely Changes
- Section 8 Traceability Matrices and Graphs

3 General System Description

This section provides general information about the software GMM-EM. It consists of two parts:

- 3.1 System Context
- 3.2 User Characteristics

3.1 System Context

The software follows the structure shown in Figure 1. The user is supposed to input a real-valued dataset as a matrix $\mathbf{X} \in \mathbb{R}^{M \times N}$ to the GMM-EM. $\mathbf{X}$ has N datapoints, and the column vector (each datapoint) can be represents as x_n. M represent the number of predictor variable which could be understood as dimension for the column vector x_n. The GMM-EM will give an label array $\mathbf{l} \in \mathbb{Z}^{+N}$ which contains the predicted label for each data points ($\mathbf{x}_n$) in the dataset. Hyper-parameters in the model will be initiated by the GMM-EM based on the model algorithm and statistics of dataset.

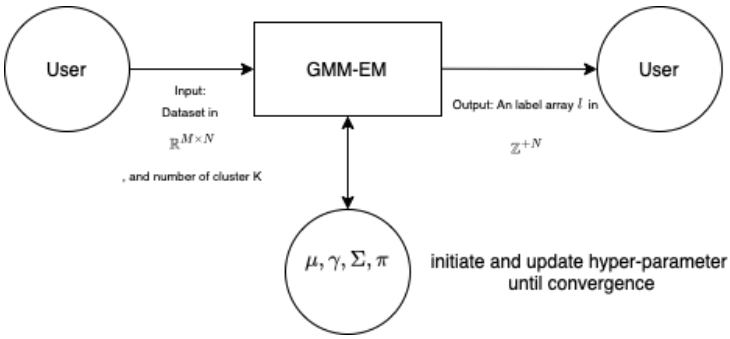

Figure 1: System Context

- User Responsibilities:
  - Provides a real-valued number of dataset without missing value in a matrix form
  - Specify the number of clusters needed
- GMM-EM Responsibilities:
  - Detect any wrong data type in input like containing string
  - Return an array of label for each data point to which cluster it should belong.

3.2 User Characteristics

The user should acquire basic programming skill and be able to perform data preprocessing for a dataset. It is also preferable that users know some common libraries in handling different data formats such as csv or json.

4 Specific System Description

- Section 4.1 Problem Description
- Section 4.2 Solution Characteristics Specification

This section first presents the problem description, which gives a high-level view of the problem to be solved. This is followed by the solution characteristics specification, which presents the assumptions, theories, definitions and finally the instance models.

4.1 Problem Description

GMM-EM is intended to implement a library on Gaussian Mixture model which is specifically used in clustering. The software tackle a general clustering problem for different datasets

4.1.1 Terminology and Definitions

In this section, we would explain the terminology used in our theoretical model.

- Gaussian distribution: A probabilistic distribution with bell-shaped.
- EM Algorithm: An optimization algorithm to find the optimized hyper-parameters for the model: start with a raw guess and adjust it based on the feedback iteratively.
- Clustering: Group the data points based on the similarity to each other
- Likelihood function: The degree of similarity of the observed dataset similar to what we guess about the data points
- Mixture of Gaussian: Data that could be separated into different Gaussian distributions.

4.1.2 Goal Statements

Given the a dataset that solely contains real numbers, which perferably designed for clustering or classification, and number of cluster stated , the goal statements are:

GS1: Predict which cluster should each data point belongs to

4.2 Solution Characteristics Specification

This section illustrate the theory which GMM-EM fundamentally relied on. We start from statistical assumption, to statistical theorem and definitions and the provide more details on the mathematical foundation about the training process for GMM-EM The instance models that governed the GMM-EM are presented in Subsection 4.3.4. The information to understand the meaning of the instance models and their derivation is also presented, so that the instance models can be verified.

4.3 Scope Decisions

The GMM-EM will fundamentally focus on the real-valued dataset which is designed for clustering or classification. Other GMM features or application such as density estimation may not be provided in our library.

4.3.1 Assumptions

A1: We assume the non-negativity of probability density function.

A2: We assume the probability density function is normalizable.

A3: We assume the dataset could be approximated by mixtures of Gaussian.

A4: We assume the dataset is real-valued without any missing value.

4.3.2 Theoretical Models

In this section, we will introduce the theoretical foundation for the GMM-EM. It relied on the assumption that the data could always approximated by a mixture of Gaussian (A3). The mathematical formulation is listed below.

RefName: **TM:GMM**

Label: mixture of Gaussian

Equation: $p(\mathbf{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}|\mu_\mathbf{k}, \mathbf{\Sigma_k})$ with $0 \leq \pi_k \leq 1$ and $\sum_{k=1}^{K} \pi_k = 1$

Description: The above equation gives the mathematical formulation on GMM. It states that for a Gaussian mixture distribution, it could be written as the linear combination of Gaussian distribution with a certain covariance $\mathbf{\Sigma_k}$ and mean $\mu_\mathbf{k}$. From the normalization and non-negativity of the probability density function, the condition for mixing coefficient is imposed as above. It is a direct result from our basic assumption (A1 and A2).

Notes: None.

Source: Pattern Recognition and Machine Learning ,Bishop (2006)

Ref. By: GD2, GD1

Preconditions for TM:GMM: None

Derivation for TM:GMM: Not Applicable

RefName: **TM:BT**

Label: Bayes' Theorem

Equation: $P(Y|X) = \frac{P(X|Y) \cdot P(Y)}{P(X)}$, $f(x|y) = \frac{f(x|y) \cdot f(y)}{f(x)}$ (form in probability density function)

Description: The above equation gives the mathematical foundation on Bayesian Statistics. This define the Log likelihood function which will be maximized in parameter training process. X and Y is two events. $P(X|Y)$ is a conditional probability : the probability of event X occurring given that Y is true. $P(Y|X)$ is also conditional probability : the probability of event Y occurring given that X is true.

Notes: None.

Source: Pattern Recognition and Machine Learning Bishop (2006)

Ref. By: GD2, GD1

Preconditions for TM:BT: None

Derivation for TM:BT: Not Applicable

4.3.3 General Definitions

We will define the Log Likelihood function and responsibilities in this section. The maximization in Log likelihood function will be key for training the GMM-EM.

Number	GD1
Label	**Responsibility**
Equation	$\gamma(z_{nk}) = \frac{\pi_k \mathcal{N}(\mathbf{x}_n \mid \mu_\mathbf{k}, \mathbf{\Sigma_k})}{\sum_{j=1}^{K} \pi_k \mathcal{N}(\mathbf{x} \mid \mu_\mathbf{j}, \mathbf{\Sigma_j})}$
Description	$\gamma(z_{nk})$: responsibility $\in [0,1]^{N \times K}$: it shows for each x_n the probability belongs to k^{th} cluster, for example $\gamma(z_{11})$ means the probability that x_1 belongs to the 1^{st} cluster. Each element for $\gamma(z_{nk})$ should lie on the close interval of $[0,1]$. z_{nk}: a latent variable or dummy variable to represent the x_n in k^{th} cluster. π_k : mixing coefficient for k^{th} cluster. It can be understood as the proportion of proportion of k^{th} cluster in the dataset. Therefore, $0 \leq \pi_k \leq 1$, since the entire proportion is 1. $\mathbf{x}_n$: n^{th} column vector of $\mathbf{X}$, represent the n^th datapoints in the dataset $\mathbf{X}$. $\mathcal{N}(\mathbf{x}_n \mid \mu_\mathbf{k}, \mathbf{\Sigma_k})$: evaluate the Gaussian distribution density function for datapoint $\mathbf{x}_n$ under the $\mu_\mathbf{k}$,$\mathbf{\Sigma_k}$ for the k^{th} cluster. This could be understood as a result from Bayes'theorem, a conditional probability which given the information of k^{th} cluster, the probabilty that x_n is classified as the member of k^{th} cluster.
Source	Pattern Recognition and Machine Learning Bishop (2006)
Ref. By	IM2

Number	GD2
Label	**Log likelihood function**
Equation	$\ln p(\mathbf{X} \mid \pi, \mu, \mathbf{\Sigma}) = \sum_{n=1}^{N} ln \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x} \mid \mu_\mathbf{k}, \mathbf{\Sigma_k})$
Description	Likelihood function is a direct result from given by the expression of Gaussian mixture with the Bayes' Theorem. We impose the log scale to the likelihood and obtain the log likelihood function. The mathematical symbols are defined same as the GD1.
Source	Pattern Recognition and Machine Learning Bishop (2006)
Ref. By	IM2

4.3.4 Instance Models

We will outline the EM part in GMM-EM. Instead of giving details in implementation, we will provide a rather abstract but mathematically non-rigorous treatment in this section.

Number	IM1
Label	**KMeans Algorithm**
Input	$\mathbf{X}$ and number of cluster K
Output	$\pi_k, \mu_\mathbf{k}, \mathbf{\Sigma_k}, \gamma(z_n k)$
Description	A clustering algorithm to initiate the parameters by grouping the datapoints in the K number of cluster following S represents the set of clusters : $S : S_1, S_2, ...S_k$ Idea: to find the assign $\mathbf{x_n}$ (the datapoint or column vector from $\mathbf{X}$) to the cluster that could minimize the distance from that cluster $argmin \sum_{i=1}^{K} \sum_{x \in S} ||\mathbf{x} - \mu_\mathbf{i}||^2$. The $\mu_\mathbf{k}$ will be the $\mu_\mathbf{k}$ found above (centroid of the cluster) π_k : the portion of S_k $\gamma(z_n k)$: 1 if $x_n \in S_k$, 0 if not Σ_k : Identity matrix (For simplicity)
Sources	Pattern Recognition and Machine Learning Bishop (2006)
Ref. By	IM2

Number	IM2
Label	**EM Algorithm**
Input	Initialized $\pi_\mathbf{k}, \mu_\mathbf{k}, \mathbf{\Sigma_k}, \gamma(z_{nk})$ from IM1
Output	Best parameter in $\pi_\mathbf{k}, \mu_\mathbf{k}, \mathbf{\Sigma_k}, \gamma(z_n k)$ and from this we could obtain the label $l \in \mathbb{Z}^{+\mathbb{M}}$ from responsibility $\gamma(z_{nk})$.
Description	EM algorithm aims at maximize the likelihood function with respect to parameters. Re-estimate the parameters followed until reaching maximum iteration or the hyper-parameter converge ($||\mu_k^{new} - \mu|| \leq tol$, or μ could be replaced by $\mathbf{\Sigma}, \pi$) : $\gamma(z_{nk}) = \pi_k N(x_n \vert \mu_k, \mathbf{\Sigma_k}) / \sum_{j=1}^{k} \pi_j N(x_n \vert \mu_j, \mathbf{\Sigma_j})$ $\mu_k^{new} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk}) \mathbf{x}_n$ $\mathbf{\Sigma}_k^{new} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk})(\mathbf{x}_n - \mu_k^{new})(\mathbf{x}_n - \mu_k^{new})^T$ $\pi_k^{new} = \frac{N_k}{N}$ The algorithm Iterative update on $\pi_\mathbf{k}, \mu_\mathbf{k}, \mathbf{\Sigma_k}$ based on the responsibility and log likelihood calculated until convergence in $\pi_\mathbf{k}, \mu_\mathbf{k}, \mathbf{\Sigma_k}$ or log likelihood.
Sources	Pattern Recognition and Machine Learning by Christopher M. Bishop

A rough derivation of EM Algorithm

To avoid over-complexity, instead of giving the rigorous derivation, we outline the idea here. The idea is to guess the parameters and update to the new set of parameters which maximize the log likelihood. The implementation of update detail is ignored here but the idea is basically taken the derivative for likelihood function and simplify the expression with the concept of responsibility.

4.3.5 Input Data Constraints

Each value x in the dataset should satisfied $x \in \mathbb{R}$.

4.3.6 Properties of a Correct Solution

We could not verify the solution analytically, since algorithm is an optimization approach, which does not guarantee the analytical correctness. Therefore, in theory, if the model converge to a solution , it exists a certain possibility that it is a correct solution. In practice, since this is a clustering algorithm to classify each data-point to a given cluster. We could verify the correctness of the solution with the true label provided from the original dataset. We could measure the correctness in terms of similarity to our solution (predicted label) and the true label. If it is close enough, then it could be said as a correct solution in practice. The exact verfication of the solution will state in the VnV part.

5 Requirements

The function requirements and nonfunctional requirements for GMM-EM are defined in this section

5.1 Functional Requirements

R1: Missing values or illegal values will be detected by the software.

R2: Convergence of solution is guaranteed (IM2)

R3: We verified the model with existing library with some simple dataset as test case.

R4: The output will be an array specific the which cluster data points should belong to. (IM2)

5.2 Nonfunctional Requirements

NFR1: **Accuracy** We do not guarantee an optimal clustering for our GMM-EM, specially when user input a dataset which is not intended to perform clustering or classification. The model still provides output but may not be applicable in this context. In the general clustering or classification case, other existing library of GMM will be used as a benchmark for the accuracy.

NFR2: **Understandibility** We will give a detailed description of function call, class in the library, which should include the at least function/class attributes, class method and function/class arguments taken to facilitate the user.

NFR3: **Maintainability** Our library should perform code maintainability for any developers to update or add functions to current library. A documentation for the library will be created to describe the design of the program, including algorithms, function call, class and other possible implementation details. Future development could be facilitated based by the documentation.

NFR4: **Performance** The GMM-EM aims at performance comparable to the library in python package scikit-learn or some simlilar library . It should be able to handle large dataset, e.g. dataset with more than 100000 datapoints.

NFR5: **Portability** The GMM-EM should be a cross-platform library. The library should be able to operated in Windows 7+ and MacOS 10+ with the corresponding compiler or interpreter.

5.3 Rationale

To reduce the complexity and retain the software to a specific domain in machine learning tasks, the scope of GMM-EM would be focused on the clustering in real-valued dataset.

6 Likely Changes

LC1: The actual data structure implemented depends on the programming languages and programmer's choice. For example, pandas dataframe will be used in python or vector will be used in C++.

7 Unlikely Changes

LC2: EM algorithm as the training algorithm is unlikely changed as it is widely adapted in GMM.

8 Traceability Matrices and Graphs

The purpose of the traceability graphs is also to provide easy references on what has to be additionally modified if a certain component is changed. The arrows in the graphs represent dependencies. The component at the tail of an arrow is depended on by the component at the head of that arrow. Therefore, if a component is changed, the components that it points to should also be changed.

	R1	R2	R3	R4
IM1		X		
IM2		X		X

Table 1: Traceability Matrix Showing the Connections Between Requirements and Instance Models

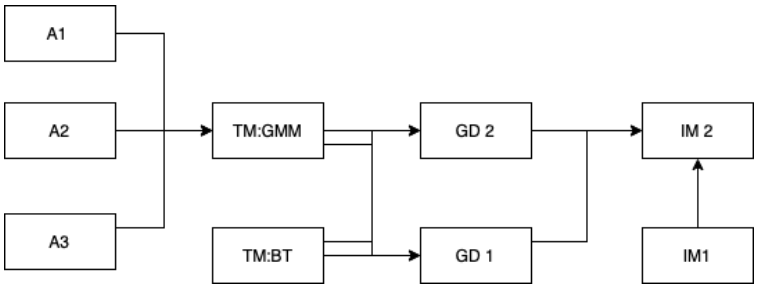

Figure 2: Traceability Graph

References

Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, Berlin, Heidelberg, 2006. ISBN 0387310738.