Name : 李權桓(Jack Li)

Student ID: D10546004

Subject Name: Nonlinear Programming

Date: 06 June 2024

**Gaussian Mixture Model with Variational Inference**

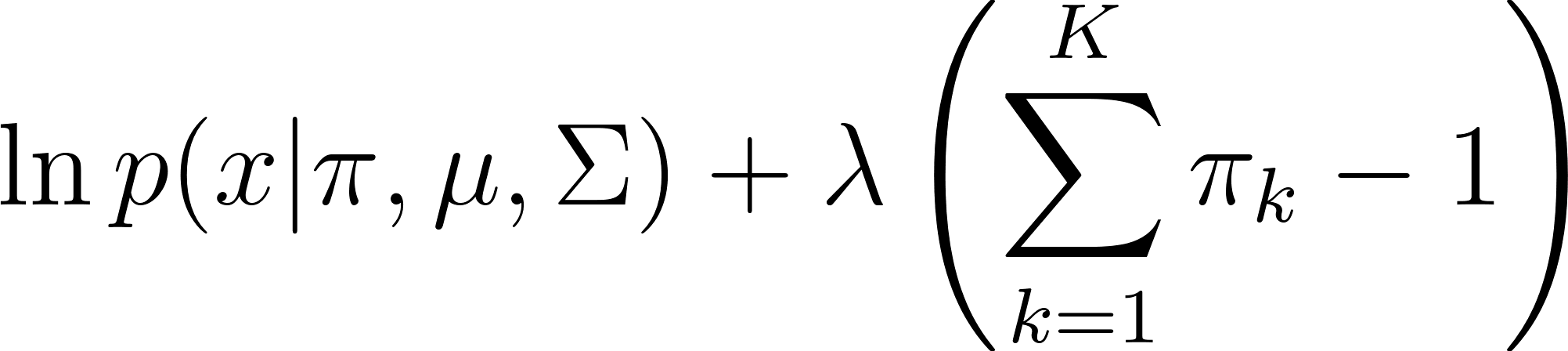
# 1 Introduction

A Gaussian Mixture Model (GMM) is a probabilistic model that assumes that the data is generated from a mixture of several Gaussian distributions with unknown parameters. It is a generalization of the Gaussian distribution to multiple distributions. The GMM represents a dataset as a mixture of several Gaussian distributions, each defined by its mean, covariance, and mixture weight. The GMM model is widely used for clustering and density estimation and it shows many GMM variants solved for specific purposes successfully.

# 2 Motivation

In our research, we delve into the GMM model which offers a powerful framework despite its constraints. GMM assumes that the underlying data distribution is a mixture of Gaussian components and (Reynolds, 2009)[[1]](https://www.zotero.org/google-docs/?rhVKl5) provides a foundational concept of GMM. If the data doesn’t follow this assumption (e.g., if it has heavy tails or non-Gaussian shapes), GMM may not perform well. It is noteworthy that choosing too few components may oversimplify the model, while too many components can lead to overfitting. In our research, we also investigate whether the distributions are closed. If they are, this insight allows us to classify the data into a smaller number of Gaussian distributions.

The main approach to solve GMM is by using Expectation Maximization(EM) is the maximization function derivative with respect to [](https://www.codecogs.com/eqnedit.php?latex=%5Cmu%2C%20%5CSigma%2C%20%5Cpi#0), the Lagrange multiplier function as follow,

[](https://www.codecogs.com/eqnedit.php?latex=%5Cln%20p(x%7C%5Cpi%2C%20%5Cmu%2C%20%5CSigma)%20%2B%20%5Clambda%20%5Cleft(%5Csum_%7Bk%3D1%7D%5EK%20%5Cpi_k%20-%201%5Cright)%20#0)

the pseudocode as follow:

1. Initialization:

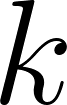
- Initialize model parameters (e.g., means, covariances, mixing coefficients).

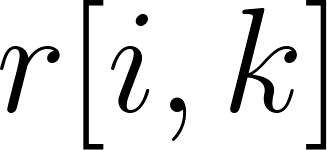
- Set convergence criteria (e.g., maximum number of iterations or tolerance).

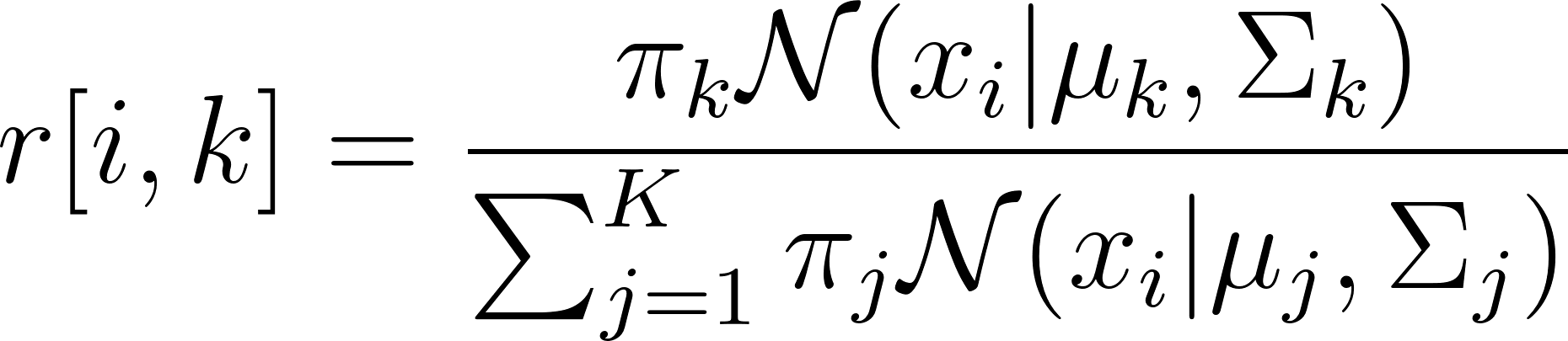
2. Repeat until convergence:

- E-step (Expectation):

- Compute the responsibilities (posterior probabilities) for each data point:

- For each data point [](https://www.codecogs.com/eqnedit.php?latex=i#0) and component [](https://www.codecogs.com/eqnedit.php?latex=k#0):

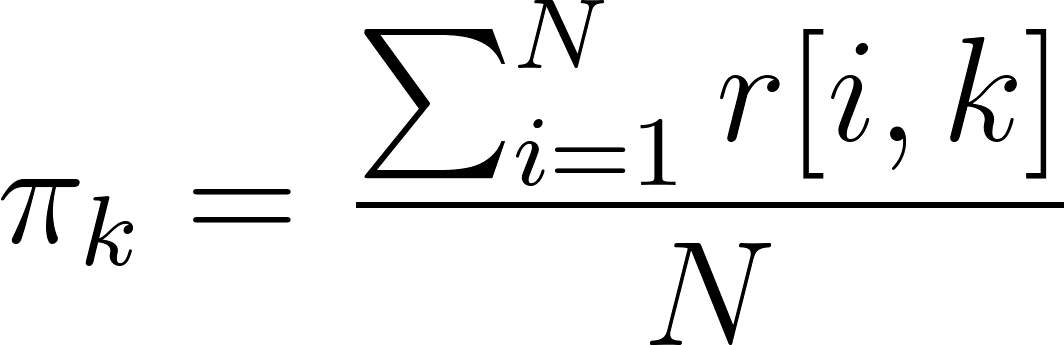
- Compute the responsibility [](https://www.codecogs.com/eqnedit.php?latex=r%5Bi%2C%20k%5D#0):

[](https://www.codecogs.com/eqnedit.php?latex=%20r%5Bi%2C%20k%5D%20%3D%20%5Cfrac%7B%5Cpi_k%20%5Cmathcal%7BN%7D(x_i%20%7C%20%5Cmu_k%2C%20%5CSigma_k)%7D%7B%5Csum_%7Bj%3D1%7D%5EK%20%5Cpi_j%20%5Cmathcal%7BN%7D(x_i%20%7C%20%5Cmu_j%2C%20%5CSigma_j)%7D%20#0)

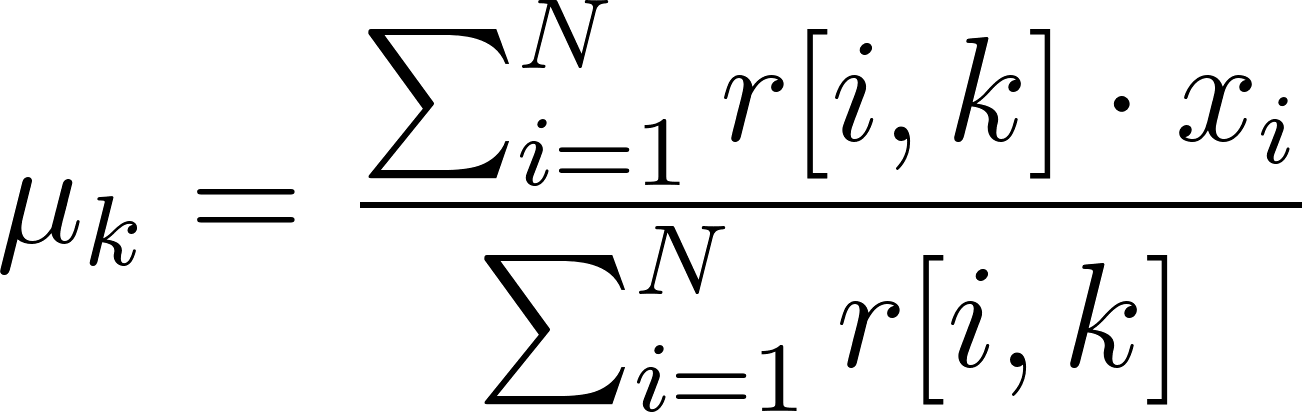
- M-step (Maximization):

- Update model parameters:

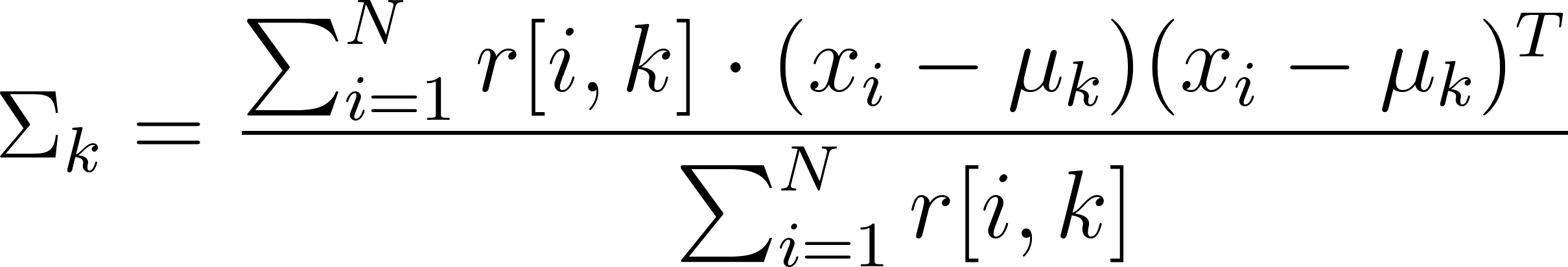
- Update mixing coefficients:

[](https://www.codecogs.com/eqnedit.php?latex=%20%5Cpi_k%20%3D%20%5Cfrac%7B%5Csum_%7Bi%3D1%7D%5EN%20r%5Bi%2C%20k%5D%7D%7BN%7D%20#0)

- Update means:

[](https://www.codecogs.com/eqnedit.php?latex=%20%5Cmu_k%20%3D%20%5Cfrac%7B%5Csum_%7Bi%3D1%7D%5EN%20r%5Bi%2C%20k%5D%20%5Ccdot%20x_i%7D%7B%5Csum_%7Bi%3D1%7D%5EN%20r%5Bi%2C%20k%5D%7D%20#0)

- Update covariances:

[](https://www.codecogs.com/eqnedit.php?latex=%20%5CSigma_k%20%3D%20%5Cfrac%7B%5Csum_%7Bi%3D1%7D%5EN%20r%5Bi%2C%20k%5D%20%5Ccdot%20(x_i%20-%20%5Cmu_k)(x_i%20-%20%5Cmu_k)%5ET%7D%7B%5Csum_%7Bi%3D1%7D%5EN%20r%5Bi%2C%20k%5D%7D%20#0)

3. Check for convergence:

- If the change in model parameters is below a predefined threshold, stop.

- Otherwise, go back to step 2.

## 2.1 Problem scenario

Despite the promising GMM work, (Coretto, 2022)[[2]](https://www.zotero.org/google-docs/?g2r6Is) also addresses practical challenges in GMM. We would like to use this example to illustrate the limitations of the GMM model. GMM training involves maximizing the likelihood function, which is non-convex. Multiple local optima exist, making it challenging to find the global maximum. The analytic solution is meant to have gradients with respect to parameters equal to 0. However, in practice, achieving this is difficult, especially when dealing with high-dimensional data.

Also, GMMs are sensitive to outliers because they aim to fit Gaussian distributions to the data. Outliers can significantly affect the estimated means and covariances.

In Figure 1, it has 2 closed distributions(Component 1 and 2) with the outlier(Component 3) distribution. The black dash line shows the three distributions multiplying corresponding weights. If we fit with GMM model with 3 distributions, results converge. But it will have impact as we mention in the above context in terms of outlier and optimality issue. When double the components to 6, it still shows inconsistent results compared to the original model(black line).

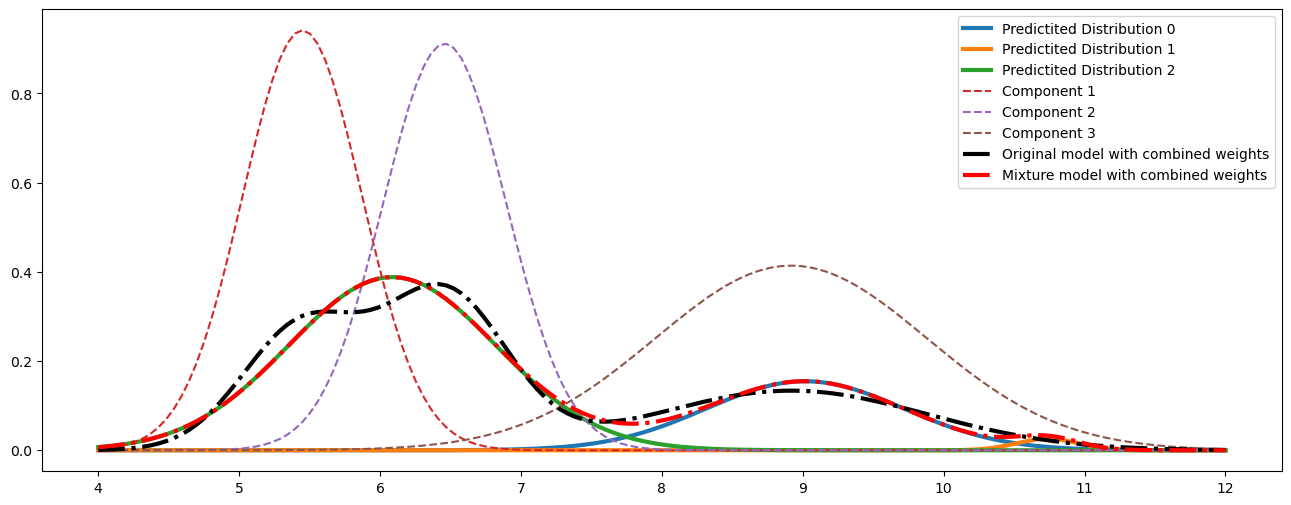


Figure 1: Original distribution fit with vanilla GMM model 3 components

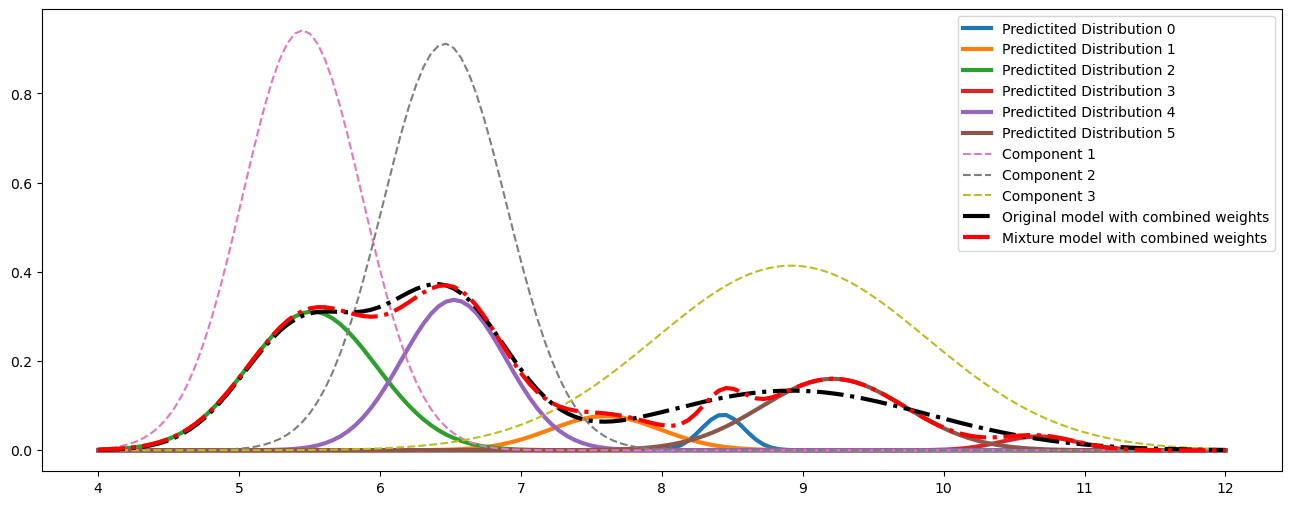
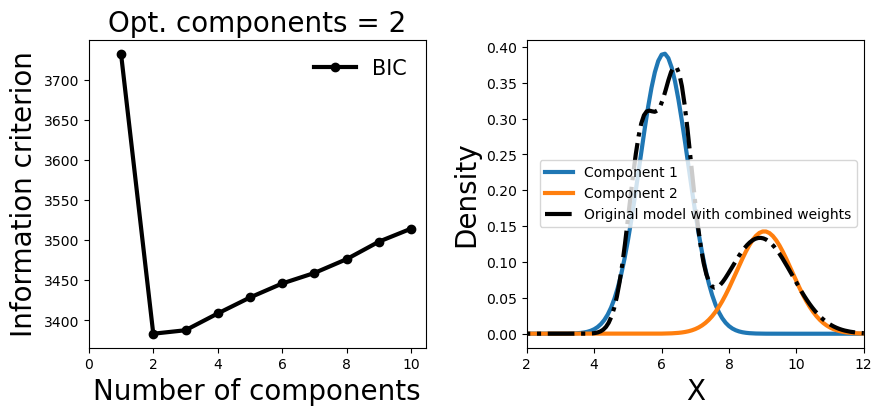


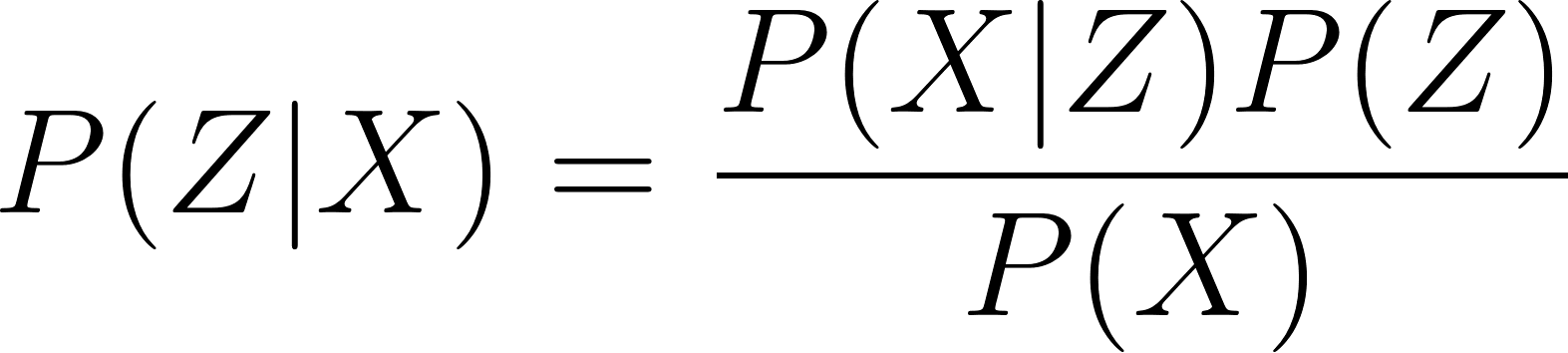
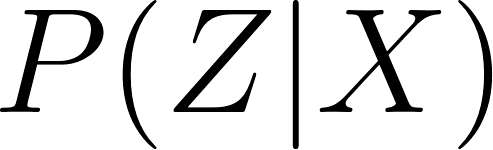
Figure 2: Original distribution fit with vanilla GMM model 6 components

There are also some techniques to decide optimal components like Bayesian Information Criterion (BIC) and Akaike Information Criterion (AIC). The AIC penalizes models with more parameters, encouraging simplicity while accounting for how well the model fits the data. Lower AIC values indicate a better-performing model. The BIC favors simpler models more aggressively than the AIC. It is particularly useful when dealing with small sample sizes. Like the AIC, lower BIC values indicate better models. We simply use BIC and the results show better components are 2 in Figure 4.



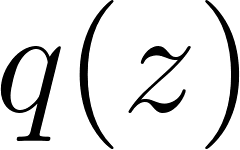
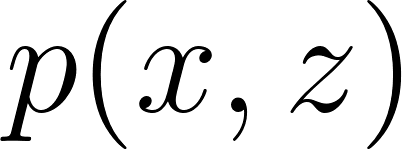
## 2.2 Variational inference

(Blei et al., 2017)[[3]](https://www.zotero.org/google-docs/?Hr6XxO)Variational Inference is a powerful technique for approximating complex probability distributions. It allows us to approximate an intractable posterior distribution with a simpler, tractable distribution. The core idea is to find the best approximation (usually from a specific family of distributions) that minimizes the divergence between the true posterior and the approximating distribution.

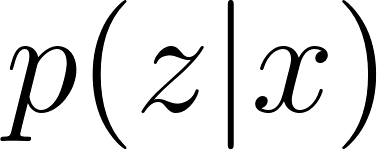
we observe [](https://www.codecogs.com/eqnedit.php?latex=X#0) as dataset and latent valuable is [](https://www.codecogs.com/eqnedit.php?latex=Z#0). From Bayes' Rule: [](https://www.codecogs.com/eqnedit.php?latex=P(Z%7CX)%3D%5Cfrac%7BP(X%7CZ)P(Z)%7D%7BP(X)%7D#0) the goal is to find posterior probability [](https://www.codecogs.com/eqnedit.php?latex=P(Z%7CX)#0) given prior [](https://www.codecogs.com/eqnedit.php?latex=P(Z)#0).

In order to solve [](https://www.codecogs.com/eqnedit.php?latex=P(Z%2CX)#0) and [](https://www.codecogs.com/eqnedit.php?latex=P(X)#0), the key concepts and equations related to VI:

1. Evidence Lower Bound (ELBO):

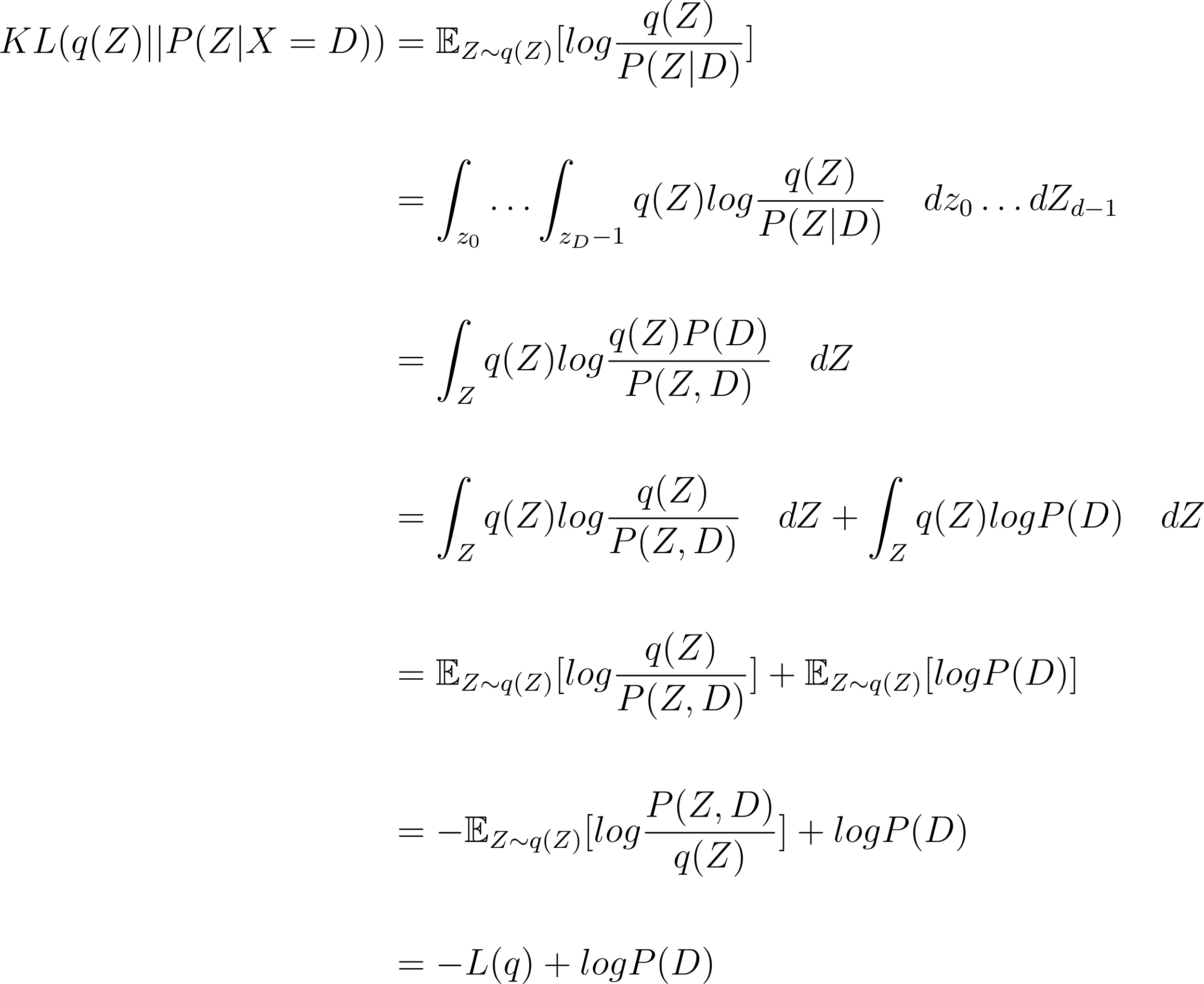
The ELBO is a fundamental quantity in VI. It provides a lower bound on the log-likelihood of the data.Here, The surrogate posterior [](https://www.codecogs.com/eqnedit.php?latex=q(z)#0) is the variational distribution (our approximation), and [](https://www.codecogs.com/eqnedit.php?latex=p(x%2C%20z)#0) is the joint distribution of the model.

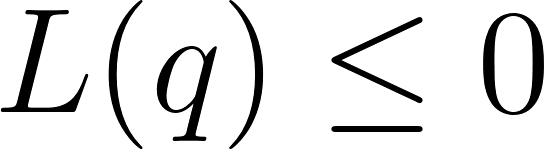
1. KL Divergence:

The Kullback-Leibler (KL) divergence measures the difference between two probability distributions.The goal in VI is to minimize the KL divergence between the true posterior [](https://www.codecogs.com/eqnedit.php?latex=p(z%7Cx)#0) and the variational distribution. To proof :

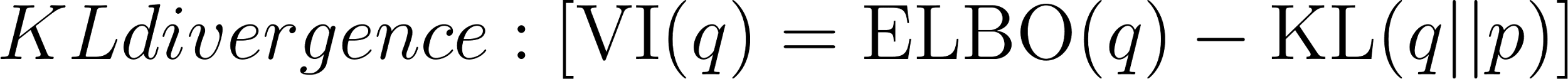
[](https://www.codecogs.com/eqnedit.php?latex=q%5E*%20(Z)%3D%5Cunderset%7Bq(Z)%5Cin%20Q%20%7D%7B%5Cmathrm%7Barg%5Cmin%7D%7D%5C%20KL(q(Z)%7C%7CP(Z%7CX%3DD))#0)

Steps:

[](https://www.codecogs.com/eqnedit.php?latex=%5Cbegin%7Balign*%7D%5C%5C%5C%5C%20KL(q(Z)%7C%7CP(Z%7CX%3DD))%20%26%3D%5Cmathbb%7BE%7D_%7BZ%5Csim%20q(Z)%7D%20%5Blog%20%5Cfrac%7Bq(Z)%7D%7BP(Z%7CD)%7D%5D%20%5C%5C%5C%5C%5C%5C%20%26%3D%20%5Cint_%7Bz_0%7D%5Cdots%5Cint_%7Bz_D-1%7D%20q(Z)%20log%20%5Cfrac%7Bq(Z)%7D%7BP(Z%7CD)%7D%20%5Cquad%20dz_0%20%5Cdots%20dZ_%7Bd-1%7D%20%5C%5C%5C%5C%5C%5C%20%26%3D%20%5Cint_%7BZ%7D%20q(Z)%20log%20%5Cfrac%7Bq(Z)P(D)%7D%7BP(Z%2CD)%7D%20%5Cquad%20dZ%20%5C%5C%5C%5C%5C%5C%20%26%3D%20%5Cint_%7BZ%7D%20q(Z)%20log%20%5Cfrac%7Bq(Z)%7D%7BP(Z%2CD)%7D%20%5Cquad%20dZ%20%2B%20%5Cint_%7BZ%7D%20q(Z)%20log%20P(D)%20%5Cquad%20dZ%20%5C%5C%5C%5C%5C%5C%20%26%3D%20%5Cmathbb%7BE%7D_%7BZ%5Csim%20q(Z)%7D%20%5Blog%20%5Cfrac%7Bq(Z)%7D%7BP(Z%2CD)%7D%5D%20%2B%20%5Cmathbb%7BE%7D_%7BZ%5Csim%20q(Z)%7D%20%5Blog%20P(D)%5D%20%5C%5C%5C%5C%5C%5C%20%26%3D%20-%5Cmathbb%7BE%7D_%7BZ%5Csim%20q(Z)%7D%20%5Blog%20%5Cfrac%7BP(Z%2CD)%7D%7Bq(Z)%7D%5D%20%2B%20log%20P(D)%20%5C%5C%5C%5C%5C%5C%20%26%3D%20-L(q)%20%2B%20log%20P(D)%20%20%5Cend%7Balign*%7D#0)

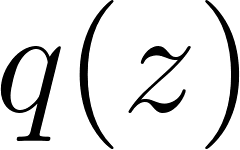
Because KL divergence is greater than 0, and the ELBO [](https://www.codecogs.com/eqnedit.php?latex=L(q)%20%5Cleq%200#0) must be smaller than evidence, because of [](https://www.codecogs.com/eqnedit.php?latex=KL%20%5Cgeq%200#0). ELBO optimal is here for [](https://www.codecogs.com/eqnedit.php?latex=KL%3D0#0).

1. Variational Objective:

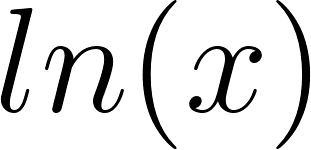
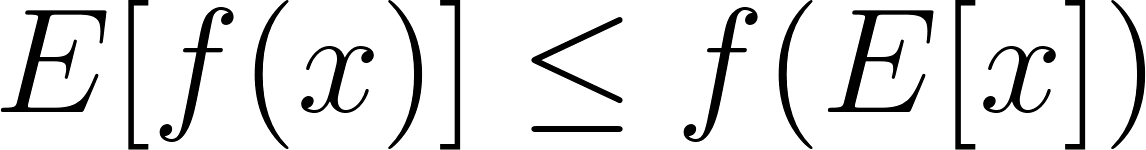
The variational objective (also known as the evidence lower bound) combines the ELBO and the [](https://www.codecogs.com/eqnedit.php?latex=KL%20divergence%3A%20%5B%20%5Ctext%7BVI%7D(q)%20%3D%20%5Ctext%7BELBO%7D(q)%20-%20%5Ctext%7BKL%7D(q%20%7C%7C%20p)%20%5D#0)

Maximizing the VI corresponds to minimizing the KL divergence.

1. Optimization:

We optimize the variational objective by adjusting the parameters of the variational distribution [](https://www.codecogs.com/eqnedit.php?latex=q(z)#0).Given a probabilistic model with latent variables (z) and observed data (x), the ELBO is defined as:

[](https://www.codecogs.com/eqnedit.php?latex=%5Ctext%7BELBO%7D(q)%20%3D%20%5Cmathbb%7BE%7D_%7Bq(z)%7D%5B%5Clog%20p(x%2C%20z)%20-%20%5Clog%20q(z)%5D%20#0)

It can derive by using [](https://www.codecogs.com/eqnedit.php?latex=ln(x)#0) concave function and Jesen’s inequality [](https://www.codecogs.com/eqnedit.php?latex=E%5Bf(x)%5D%20%5Cleq%20f(E%5Bx%5D)#0) and results to find an approximation that balances accuracy and tractability.

# 3 GMM with VI

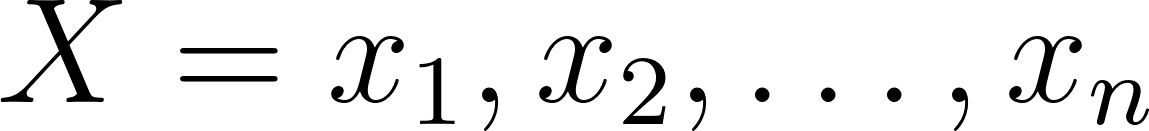
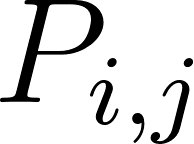
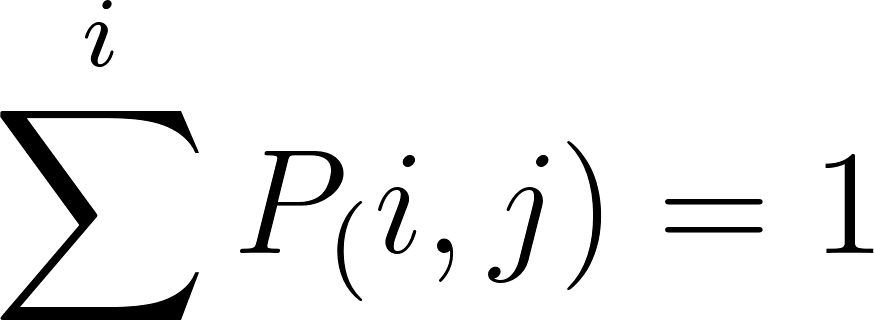
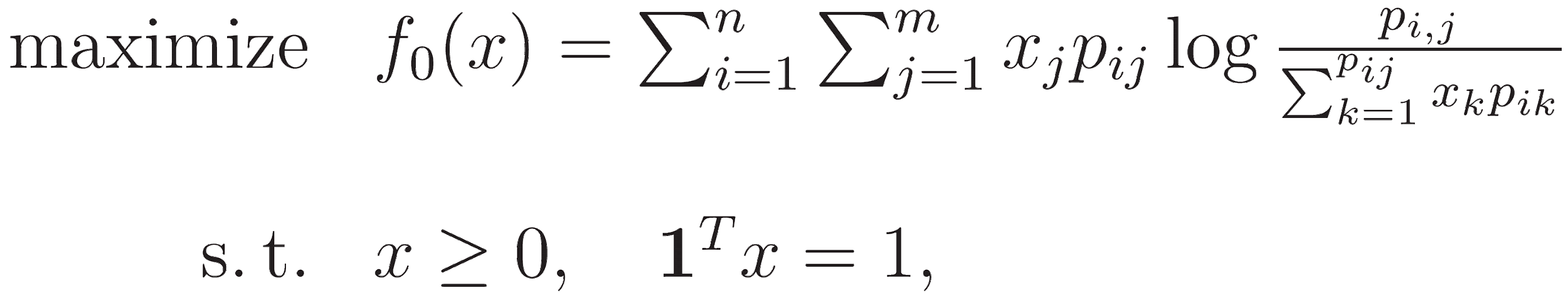
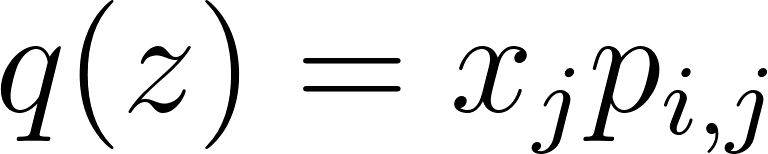
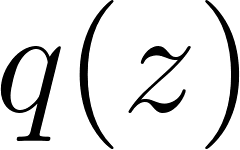
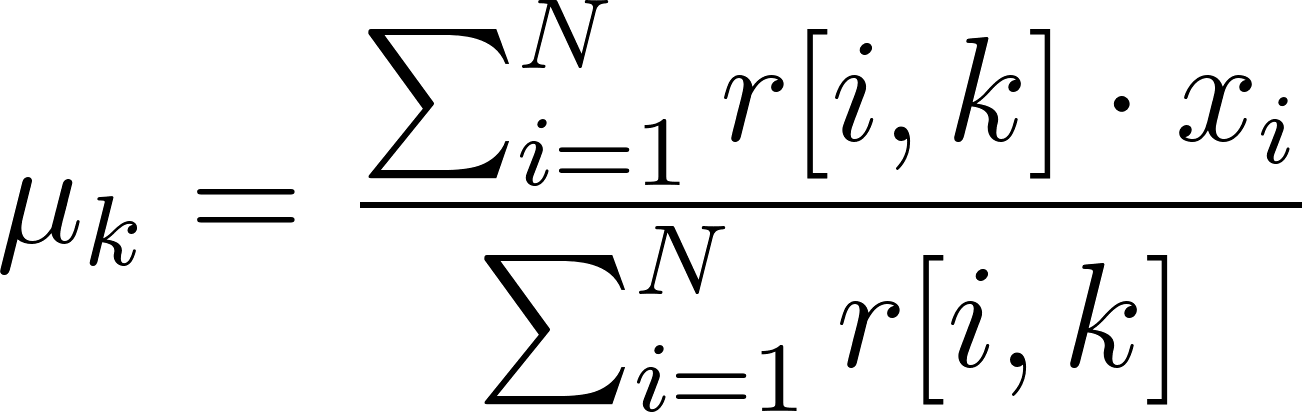
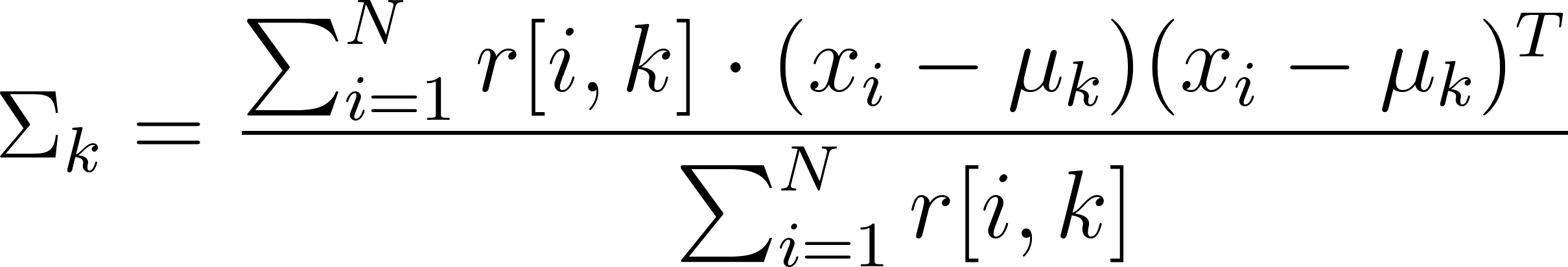
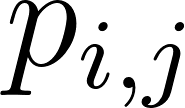
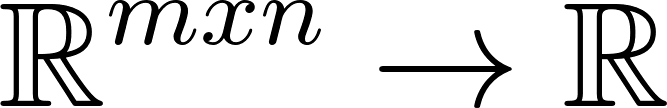
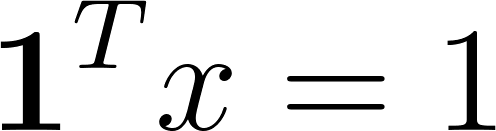
From GMM, there are each data points [](https://www.codecogs.com/eqnedit.php?latex=X%3D%7Bx_1%2C%20x_2%2C%20%5Cdots%20%2C%20x_n%7D#0) contribute to each Gaussian distribution. The probability of from [](https://www.codecogs.com/eqnedit.php?latex=P_%7Bi%2Cj%7D#0) is matrix of [](https://www.codecogs.com/eqnedit.php?latex=%5Cmathbb%7BR%7D%5E%7Bnxm%7D#0) . In Channel of Capacity, the probability [](https://www.codecogs.com/eqnedit.php?latex=%5Csum%5Ei%20P_(i%2Cj)%20%3D%201#0) is a constraint along with each transition probability greater than 0.



Figure 3: Illustration of GMM in directed graphical model

## 3.1 Reformulation of nonlinear problem

[](http://www.sciweavers.org/tex2img.php?bc=Transparent&fc=Black&im=jpg&fs=100&ff=modern&edit=0&eq=%5Cbegin%7Barray%7D%7Br%20l%7D%7B%5Coperatorname*%7Bmaximize%7D%7D%26%7B%7B%7Df_%7B0%7D(x)%3D%5Csum_%7Bi%3D1%7D%5E%7Bn%7D%5Csum_%7Bj%3D1%7D%5E%7Bm%7Dx_%7Bj%7Dp_%7Bi%20j%7D%5Clog%20%5Cfrac%7Bp_%7Bi%2Cj%7D%7D%20%7B%20%5Csum_%7Bk%3D1%7D%5E%7Bp_%7Bi%20j%7D%7Dx_%7Bk%7Dp_%7Bi%20k%7D%7D%7D%20%5C%5C%20%5C%5C%20%20%7B%5Coperatorname*%7Bs.t.%7D%7D%26%7Bx%5Cgeq%200%2C%5Cquad%5Cmathbf%7B1%7D%5E%7BT%7Dx%3D1%2C%7D%20%5Ctag%7B2%7D%20%5Cend%7Barray%7D%20#0)

Where [](https://www.codecogs.com/eqnedit.php?latex=q(z)%20%3D%20x_j%20p_%7Bi%2Cj%7D#0) like previous VI method, it make the distribution closed to [](https://www.codecogs.com/eqnedit.php?latex=q(z)#0) when maximizing the ELBO. The [](https://www.codecogs.com/eqnedit.php?latex=%20%5Cmu_k%20%3D%20%5Cfrac%7B%5Csum_%7Bi%3D1%7D%5EN%20r%5Bi%2C%20k%5D%20%5Ccdot%20x_i%7D%7B%5Csum_%7Bi%3D1%7D%5EN%20r%5Bi%2C%20k%5D%7D%20#0) and [](https://www.codecogs.com/eqnedit.php?latex=%20%5CSigma_k%20%3D%20%5Cfrac%7B%5Csum_%7Bi%3D1%7D%5EN%20r%5Bi%2C%20k%5D%20%5Ccdot%20(x_i%20-%20%5Cmu_k)(x_i%20-%20%5Cmu_k)%5ET%7D%7B%5Csum_%7Bi%3D1%7D%5EN%20r%5Bi%2C%20k%5D%7D%20#0) used to obtain [](https://www.codecogs.com/eqnedit.php?latex=%20p_%7Bi%2Cj%7D#0). The objective function is [](https://www.codecogs.com/eqnedit.php?latex=%5Cmathbb%7BR%7D%5E%7Bmxn%7D%20%5Crightarrow%20%5Cmathbb%7BR%7D#0). The [](https://www.codecogs.com/eqnedit.php?latex=%5Cmathbf%7B1%7D%5E%7BT%7Dx%3D1#0) is to ensure the transition probability for each gaussian distribution.

## 3.2 Experiments results

### 3.2.1 1-D case:

The same scheme, we use 3 components and the Figure 4 and 5 indicate when VI has strong generative ability and is not limited by outlier and initial star points. In my case, Iteration also converges fast, 839 vs 147, which is around more than 5 times improvement than vanilla GMM and shortest time. Because the logarithm of a convex function is not necessarily convex, we use cvxpy package with ECOS solver for my scenario.

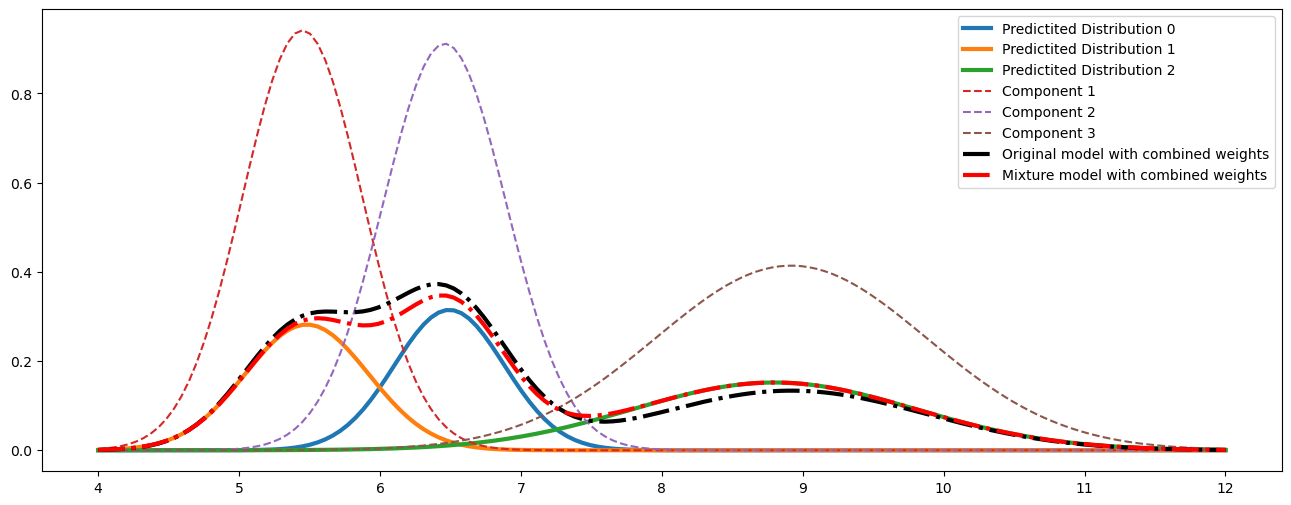


Figure 4: Original distribution fit with VI GMM model 3 components

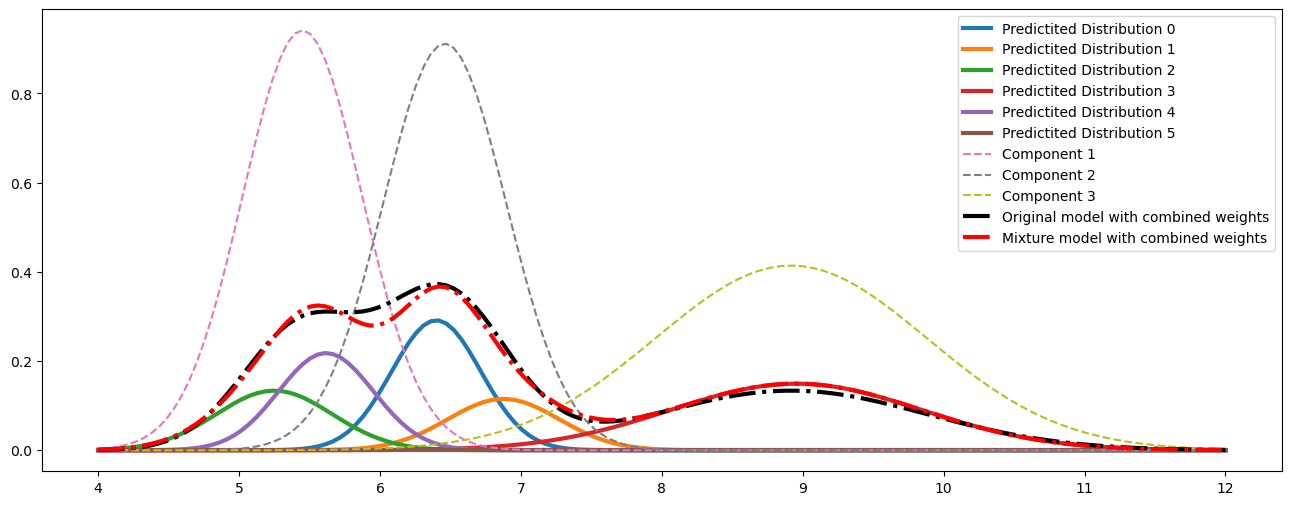


Figure 5: Original distribution fit with VI GMM model 6 components

### 3.2.1 2-D case:

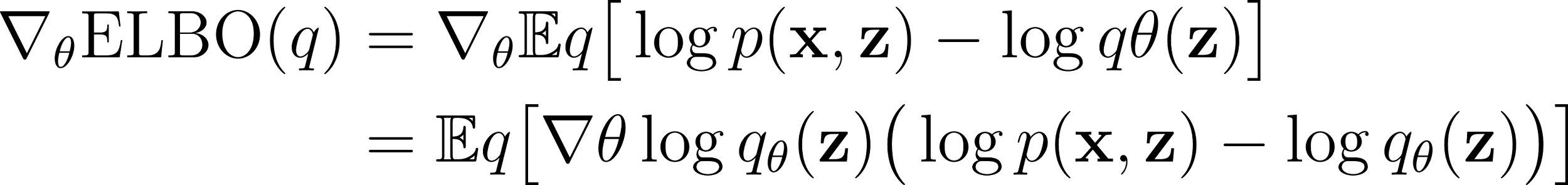
We managed to dive into multivariate normal distribution. From 100 iterations we could conclude VI GMM model has promising results in terms of classification and convergence. The drawback is each covariance can not be positive semi-definite(PSD) which might face multivariate normal distribution not to be valid. It will fail to have all eigenvalues be non-negative. We use a small trick which is to add a small multiple of the identity matrix to my original matrix. There is also Higham’s algorithm to find the nearest positive-definite matrix which might work in some scenarios as well. My observation is that the VI GMM might need more refinement in order to reach a more robust and consistent process.

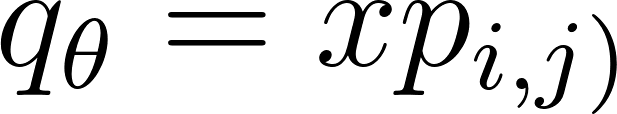
Table 1: 2-D clusters fit with VI GMM model 6 components

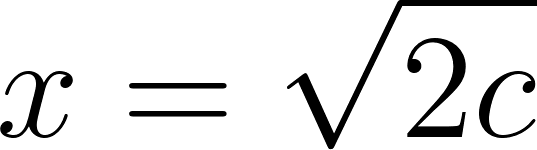
|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
| weights:  [0.26594647 0.23062394 0.22338953 0.22630703 0.01443529 0.03927455],  means:  [[6.32213808 2.35767698]  [8.75275074 8.2155947 ]  [2.67778884 5.43553413]  [1.71303902 1.46485518]  [5.3869087 9.19282174]  [1.4555381 4.10617346]],  cov:  [[[ 2.38092292 -0.27707638]  [-0.27707638 0.70244211]]  [[ 1.77909436 0.16122733]  [ 0.16122733 0.49937488]]  [[ 2.80589061 -0.11374297]  [-0.11374297 0.24995371]]  [[ 1.88645364 0.07396465]  [ 0.07396465 0.52119153]]  [[ 0.26950374 0.08554552]  [ 0.08554552 0.05829775]]  [[ 6.17115731 4.81799474]  [ 4.81799474 3.84442762]]] | weights:  [3.57306088e-11 2.32240414e-01 3.57306088e-11 2.95132962e-01  1.65161887e-01 3.07464738e-01],  means:  [[0. 0. ]  [9.39217567 8.38590059]  [0. 0. ]  [4.03528407 1.9113075 ]  [7.32249954 8.10698907]  [2.74723794 5.33736873]],  cov:  [[[ 1.00000000e-04 0.00000000e+00]  [ 0.00000000e+00 1.00000000e-04]]  [[ 1.10218465e+00 9.28470182e-02]  [ 9.28470182e-02 4.74811277e-01]]  [[ 1.00000000e-04 0.00000000e+00]  [ 0.00000000e+00 1.00000000e-04]]  [[ 8.12813397e+00 8.62306637e-01]  [ 8.62306637e-01 7.40959782e-01]]  [[ 1.65220708e+00 -4.78582925e-01]  [-4.78582925e-01 5.48463200e-01]]  [[ 2.88096356e+00 -8.83487528e-02]  [-8.83487528e-02 4.05972940e-01]]] |

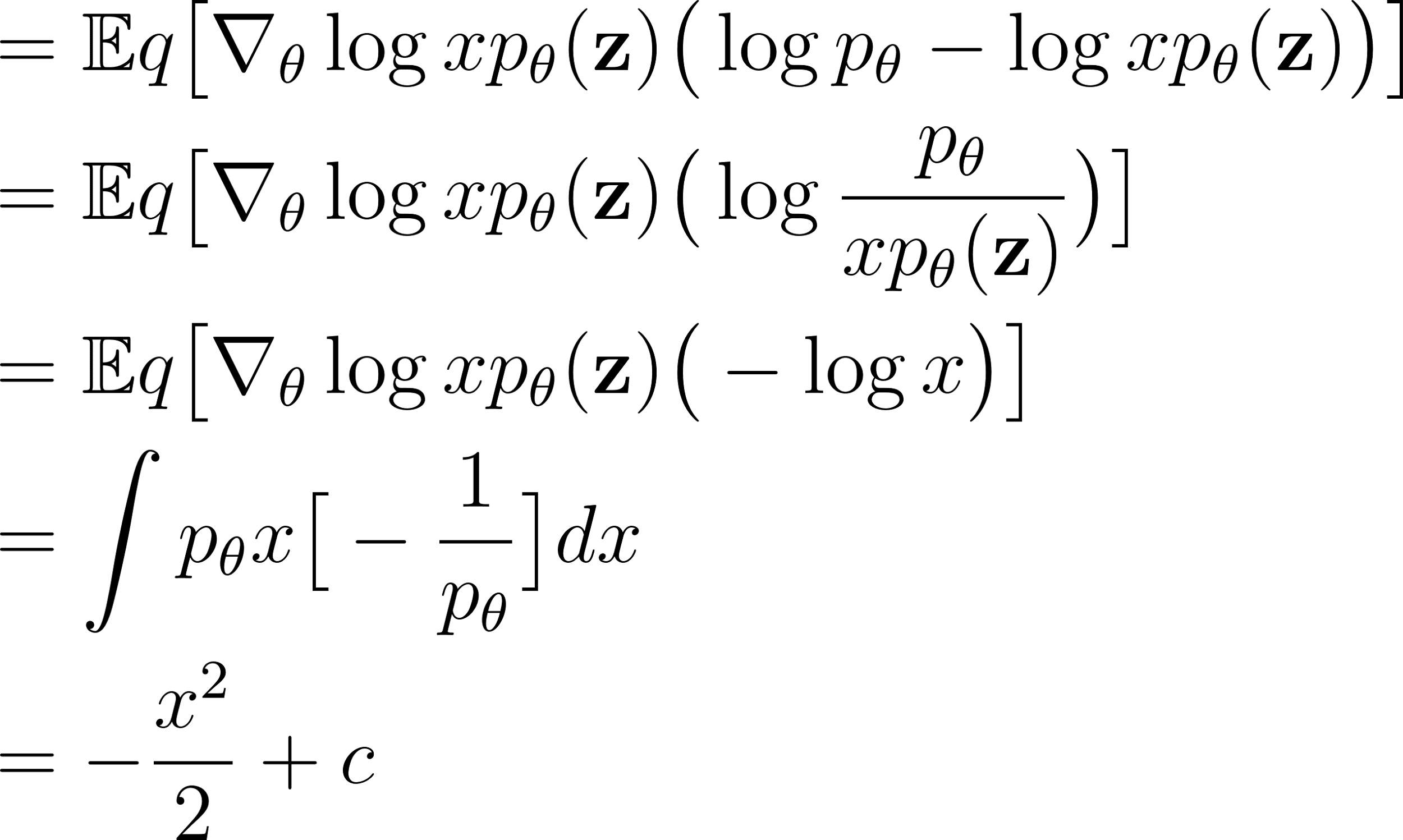
# 5 Observations

The estimation of ELBO gradient by using Score Function Estimator method:[](https://www.codecogs.com/eqnedit.php?latex=%5Cbegin%7Balign*%7D%5Cnabla_%7B%5Ctheta%7D%5Cmathrm%7BELBO%7D(q)%20%26%3D%5Cnabla_%7B%5Ctheta%7D%5Cmathbb%7BE%7D_%7Bq%7D%7B%5Cbig%5B%7D%5Clog%20p(%5Cmathbf%7Bx%7D%2C%5Cmathbf%7Bz%7D)-%5Clog%20q_%7B%5Ctheta%7D(%5Cmathbf%7Bz%7D)%7B%5Cbig%5D%7D%20%5C%5C#0)

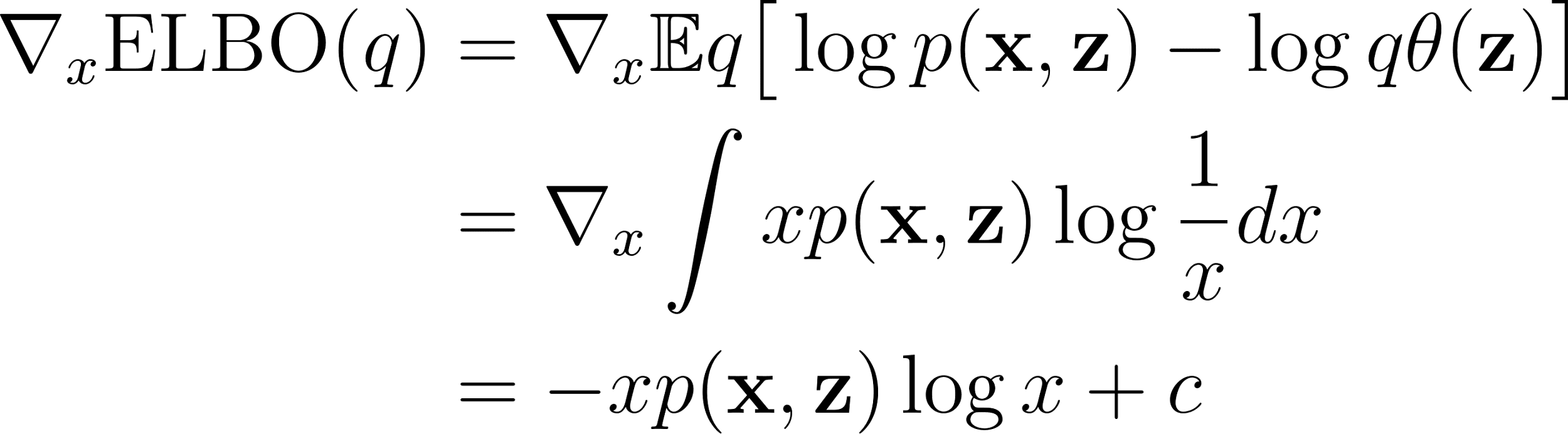
[](https://www.codecogs.com/eqnedit.php?latex=%5Cbegin%7Balign*%7D%5Cnabla_%7B%5Ctheta%7D%5Cmathrm%7BELBO%7D(q)%20%26%3D%5Cnabla_%7B%5Ctheta%7D%5Cmathbb%7BE%7D%7Bq%7D%7B%5Cbig%5B%7D%5Clog%20p(%5Cmathbf%7Bx%7D%2C%5Cmathbf%7Bz%7D)-%5Clog%20q%7B%5Ctheta%7D(%5Cmathbf%7Bz%7D)%7B%5Cbig%5D%7D%20%5C%5C%20%26%3D%5Cmathbb%7BE%7D%7Bq%7D%7B%5Cbig%5B%7D%5Cnabla%7B%5Ctheta%7D%5Clog%20q_%7B%5Ctheta%7D(%5Cmathbf%7Bz%7D)%7B%5Cbig(%7D%5Clog%20p(%5Cmathbf%7Bx%7D%2C%5Cmathbf%7Bz%7D)-%5Clog%20q_%7B%5Ctheta%7D(%5Cmathbf%7Bz%7D)%7B%5Cbig)%7D%7B%5Cbig%5D%7D%5Cend%7Balign*%7D#0)

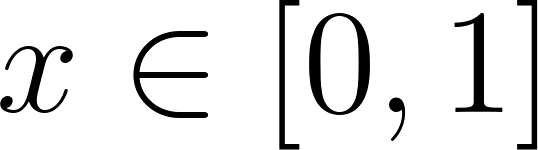
In my case [](https://www.codecogs.com/eqnedit.php?latex=q_%7B%5Ctheta%7D%20%3D%20x%20p_%7Bi%2Cj)#0):

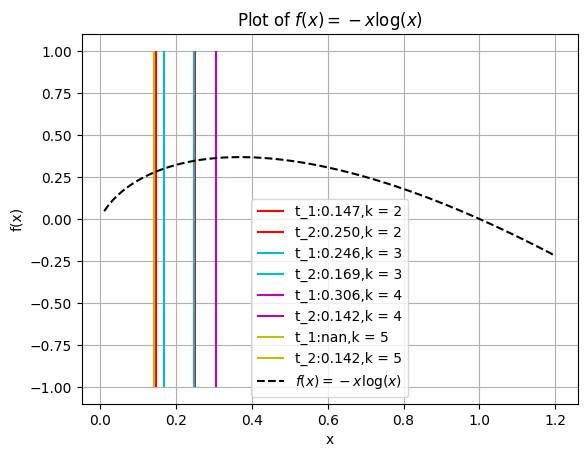
The derivative ELBO with respect to [](https://www.codecogs.com/eqnedit.php?latex=%5Ctheta#0) which is the normal distribution [](https://www.codecogs.com/eqnedit.php?latex=%5Cmu#0) and [](https://www.codecogs.com/eqnedit.php?latex=%5Csigma#0) which lead in order to have optimal [](https://www.codecogs.com/eqnedit.php?latex=%5Ctheta#0) the [](https://www.codecogs.com/eqnedit.php?latex=x%20%3D%20%5Csqrt%20%7B2c%7D#0).

[](https://www.codecogs.com/eqnedit.php?latex=%5Cbegin%7Balign*%7D%20%26%3D%5Cmathbb%7BE%7D%7Bq%7D%7B%5Cbig%5B%7D%5Cnabla_%7B%5Ctheta%7D%5Clog%20x%20p_%7B%5Ctheta%7D(%5Cmathbf%7Bz%7D)%7B%5Cbig(%7D%5Clog%20p_%7B%5Ctheta%7D-%5Clog%20x%20p_%7B%5Ctheta%7D(%5Cmathbf%7Bz%7D)%7B%5Cbig)%7D%7B%5Cbig%5D%7D%20%5C%5C%5C%5C%20%26%3D%20%5Cmathbb%7BE%7D%7Bq%7D%7B%5Cbig%5B%7D%5Cnabla_%7B%5Ctheta%7D%5Clog%20x%20p_%7B%5Ctheta%7D(%5Cmathbf%7Bz%7D)%7B%5Cbig(%7D%20%5Clog%20%5Cfrac%20%7B%20p_%7B%5Ctheta%7D%7D%20%7Bx%20p_%7B%5Ctheta%7D(%5Cmathbf%7Bz%7D)%7D%20%7B%5Cbig)%7D%7B%5Cbig%5D%7D%20%5C%5C%5C%5C%20%20%26%3D%20%5Cmathbb%7BE%7D%7Bq%7D%7B%5Cbig%5B%7D%5Cnabla_%7B%5Ctheta%7D%5Clog%20x%20p_%7B%5Ctheta%7D(%5Cmathbf%7Bz%7D)%7B%5Cbig(%7D%20-%5Clog%20x%20%7B%5Cbig)%7D%7B%5Cbig%5D%20%7D%5C%5C%5C%5C%20%20%26%3D%20%5Cint%20p_%7B%5Ctheta%7D%20x%20%7B%5Cbig%5B%7D-%5Cfrac%7B1%7D%7B%20p_%7B%5Ctheta%7D%7D%7B%5Cbig%5D%20%7Ddx%20%5C%5C%5C%5C%20%20%26%3D%20-%5Cfrac%7Bx%5E2%7D%7B%202%7D%20%2Bc%20%5Cend%7Balign*%7D%20#0)

On the other hand, derivative with respect to [](https://www.codecogs.com/eqnedit.php?latex=x#0), this have the results as follow:

[](https://www.codecogs.com/eqnedit.php?latex=%5Cbegin%7Balign*%7D%5Cnabla_%7Bx%7D%5Cmathrm%7BELBO%7D(q)%20%26%3D%5Cnabla_%7Bx%7D%5Cmathbb%7BE%7D%7Bq%7D%7B%5Cbig%5B%7D%5Clog%20p(%5Cmathbf%7Bx%7D%2C%5Cmathbf%7Bz%7D)-%5Clog%20q%7B%5Ctheta%7D(%5Cmathbf%7Bz%7D)%7B%5Cbig%5D%7D%20%5C%5C%5C%5C%20%26%3D%20%5Cnabla_%7Bx%7D%20%5Cint%20x%20p(%5Cmathbf%7Bx%7D%2C%5Cmathbf%7Bz%7D)%20%5Clog%5Cfrac%7B1%7D%7B%20x%7Ddx%20%5C%5C%5C%5C%20%26%3D%20-x%20p(%5Cmathbf%7Bx%7D%2C%5Cmathbf%7Bz%7D)%5Clog%20x%20%2Bc%20%5Cend%7Balign*%7D#0)

If coefficient is equal to 0, the [](https://www.codecogs.com/eqnedit.php?latex=x%20%5Cin%20%5B0%2C1%5D#0), which is the results coverage to this range.



When we run the VI GMM model, it shows substantially small negative weights during training. My suggestion is that the solver,cp.EMOS, took the slightly infeasible path to approach optimal values.

# 6 CONCLUSION

In our experiments, we could conclude when VI GMM models have better performance despite solver dependency and matrix operations issues. It might not work in some particular scenarios. The python CVXPY provides an optimization problem solution. The future research is that we find in what scenario and how to deal with matrices that are not PSD. The alternative distribution uses bivariate beta distribution (Olkin and Trikalinos, 2014)[[4]](https://www.zotero.org/google-docs/?cByuDX) to mitigate the multivariate normal distribution constraints.

The Variational method is a powerful technique to estimate latent variables. There are generative methods like the Hidden Markov Chain, normalizing flow can find latent space with different approaches. This generative method can help tackle the underlying distribution analytically.

# Reference

## [[1] D. Reynolds, “Gaussian Mixture Models,” in *Encyclopedia of Biometrics*, S. Z. Li and A. Jain, Eds., Boston, MA: Springer US, 2009, pp. 659–663. doi: 10.1007/978-0-387-73003-5\_196.](https://www.zotero.org/google-docs/?twzlLw)

## [[2] P. Coretto, “Estimation and computations for Gaussian mixtures with uniform noise under separation constraints,” *Stat. Methods Appl.*, vol. 31, no. 2, pp. 427–458, Jun. 2022, doi: 10.1007/s10260-021-00578-2.](https://www.zotero.org/google-docs/?twzlLw)

## [[3] D. M. Blei, A. Kucukelbir, and J. D. McAuliffe, “Variational Inference: A Review for Statisticians,” *J. Am. Stat. Assoc.*, vol. 112, no. 518, pp. 859–877, Apr. 2017, doi: 10.1080/01621459.2017.1285773.](https://www.zotero.org/google-docs/?twzlLw)

## [[4] I. Olkin and T. A. Trikalinos, “Constructions for a bivariate beta distribution.” arXiv, Sep. 16, 2014. Accessed: Jun. 02, 2024. [Online]. Available: http://arxiv.org/abs/1406.5881](https://www.zotero.org/google-docs/?twzlLw)