Validity of the objective function

We set the equation of the objective function to $NLL+\lambda*C$. NLL stands for 'Negative Log-Likelihood', λ is a constant like a weight of C , and C is an approximation constraint. We use the equation (16) on the paper as NLL , and the constraint equation on the middle of the paper on page 4 as C .

The equation of probability is that

$$\sum_{i}^{N}\sum_{k}^{K} P(p_i,k|\alpha_k,\theta_k)=1\Rightarrow\alpha_1^2\sum_i G_{i,1}^2+\alpha_2^2\sum_i G_{i,2}^2+2\alpha_1\alpha_2\cos\phi\sum_i G_{i,1}G_{i,2}$$

Thus, the approximation constraint is that

$$\alpha_1^2+\alpha_2^2+2\alpha_1\alpha_2\cos\phi\sum_i G_{i,1}G_{i,2}-1$$

Therefore, in this experiment, we'll check if the objective function works correctly.

Aim

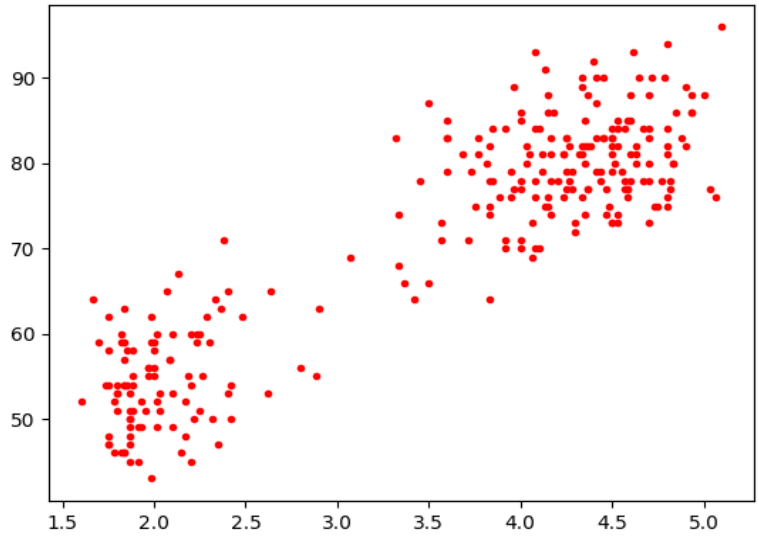
The aim of this experiment is to check if when we optimize the objective function, QGMM can train the parameters correctly.

Dataset

Dataset: *Data/Dataset/faithful.csv*

For dataset, we selected 'Old faithful' dataset because it is 2 dimensions.

Below is the graph for Old faithful.



[Fig 0. Old faithful]

Parameters and Data

These are the records of the parameters in each experiment, so each experiment can be reproduced later.

- **Optimizer**
  Adam

- **Common variables (t1 ~ t5)**
  lambda = 1

  learning rate = 0.01

  alphas = [0.5, 0.5]

  covs1 = [[0.08, 0.1],
  [0.1, 3.3]]

  covs2 = [[0.08, 0.1],
  [0.1, 3.3]]

- **t1** - Good
  mean1 = [2.6585135519388348, 54.66062219876824]
  mean2 = [3.1085745233652995, 77.99698134521407]

- **t2** - Good
  mean1 = [3.427976229515216, 61.46413393088303]
  mean2 = [4.5517041217554945, 51.756595162050985]

- **t3** - Not good
  mean1 = [2.756031811312966, 76.62447648112042]
  mean2 = [2.9226572802266397, 88.3509418943818]

- **t4** - Good
  mean1 = [4.893025788130122, 59.46713813379837]
  mean2 = [2.080000263954121, 78.15976694366192]

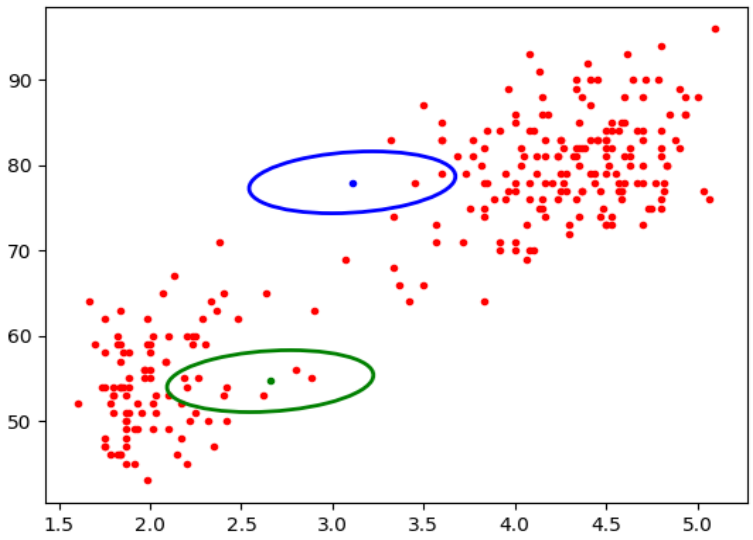
- **t5** - Bad
  mean1 = [4.171021823127277, 83.66322004888708]
  mean2 = [1.781079954983019, 95.411542531776]

All the data that were used in this experiment are in this directories.

1. Videos: *Data/Videos/Validity of the objective function*
2. CSVs: *Data/Csvs/Validity of the objective function*
3. Images: *Data/Images/Validity of the objective function*
4. Graphs: *Data/Graphs/Validity of the objective function*

Results

[t1]

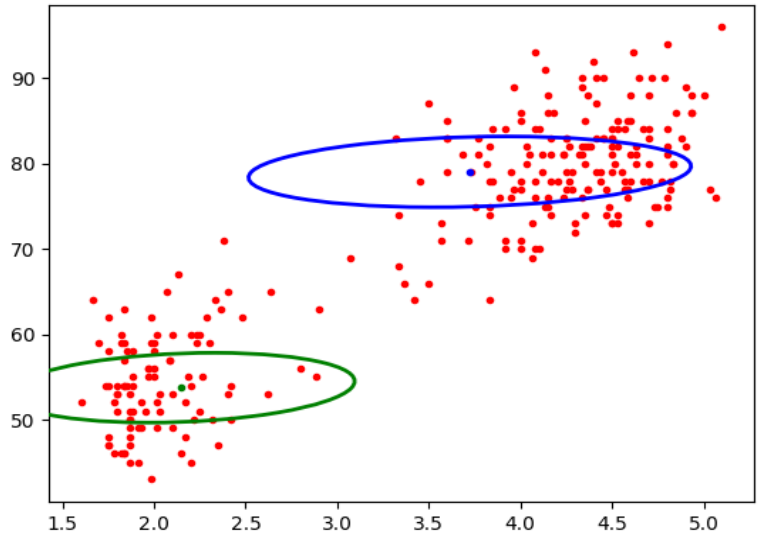


[Fig 1-1. Initial state (t1)]

[Green, Blue]

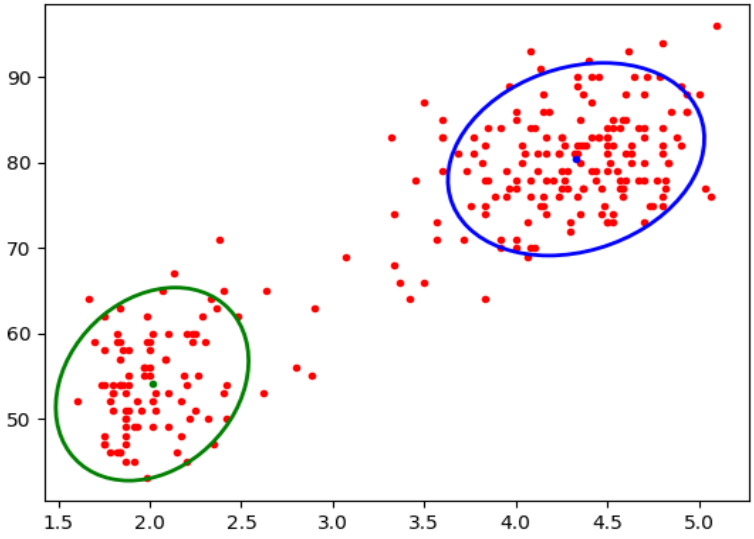
- **Alphas**
  : 0.5, 0.5
- **Unnormalized Gaussians**
  : 6.6263933, 2.5418115
- **Cosine**
  : 1534546200.0
- **Sum of probabilities**
  : 0.9930843

Seeing that the unnormalized Gaussians, the green cluster has more observations than the blue one. The shapes of the initial covariances are wide horizontally. The cosine should be reduced to between -1 and 1 theoretically. From the alphas, we can see that the two clusters have the same scale initially.



[Green, Blue]

- **Alphas**
  : 0.5521862, 0.7609049
- **Unnormalized Gaussians**
  : 15.295971, 20.916563
- **Cosine**
  : 26186176.0
- **Sum of probabilities**
  : 3.9756699



At this state, from the unnormalized cosine should be reduced more. From

[Green, Blue]

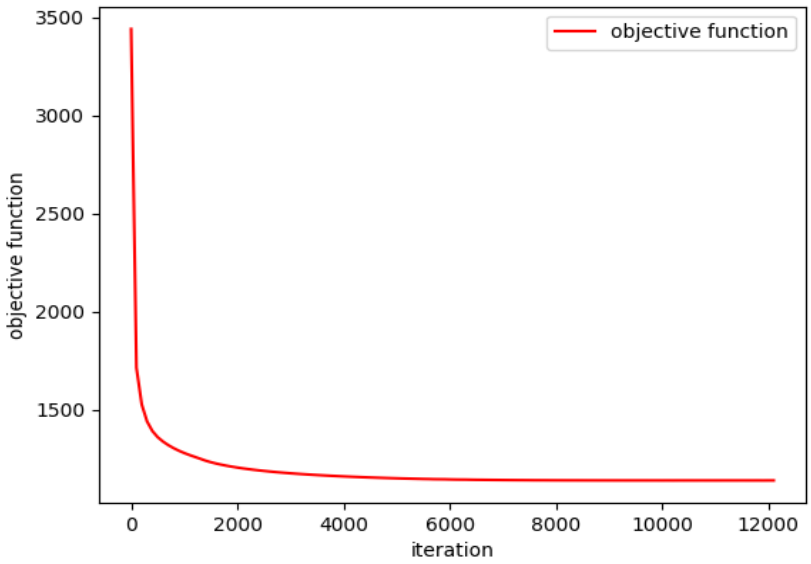
- **Alphas**
  : 0.578844, 0.7878746

[Fig 1-3. Iteration 12100 (t1)]

- **Unnormalized Gaussians**
  : 21.563248, 31.67363
- **Cosine**
  : 305.7096
- **Sum of probabilities**
  : 6.0042257

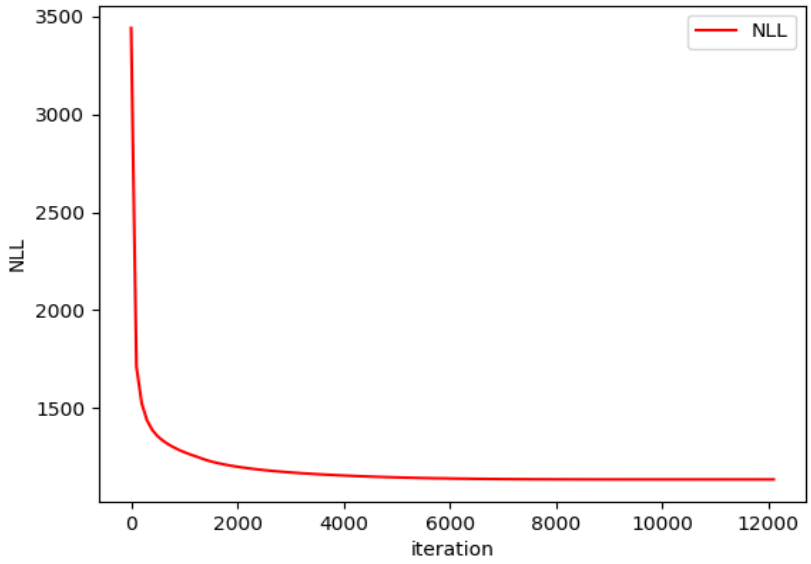
Fig 1-3 is the last state of the training. From the unnormalized Gaussians, we can see that the blue cluster had more observations than the green one. Although the cosine has be decreased, it should be decreased more. I think it is associated with the unconstrained optimization. From the alphas, we also see that the blue one has bigger scale than another. The sum of probabilities was converged to about 6 while arrived at the center of the observations.

Now, let's look into some graphs that are represented as indicates of the training process.



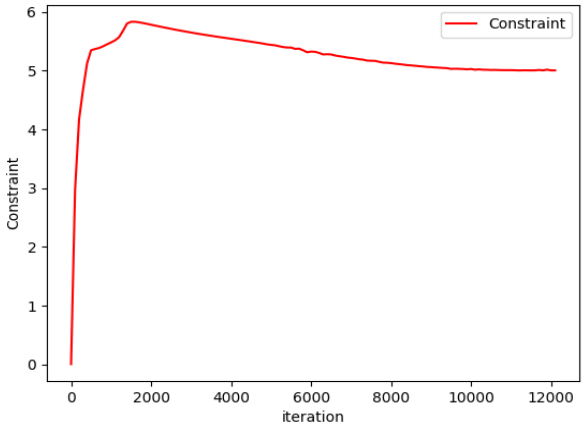
[Fig 1-4. Objective function (t1)]

The objective function has been decreased, and at the end it converged to 1140.5731. Therefore, we can judge our optimizer minimized the objective function properly. However, it doesn't guarantee a good result, so we should look into other graphs as well.

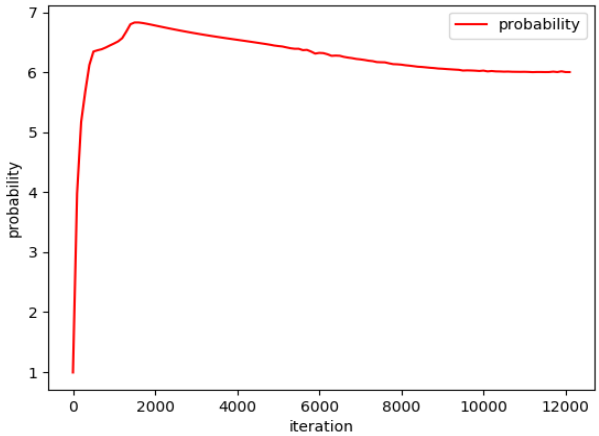


[Fig 1-5. Negative log-likelihood (t1)]

From the above graph, we can see that NLL has been decreased until iteration 4000 sharply, and after that, it became gently. The shape of line is very similar to the Fig 1-4. From that, we also figure out that the constraint hasn't much impact on the objective functions.

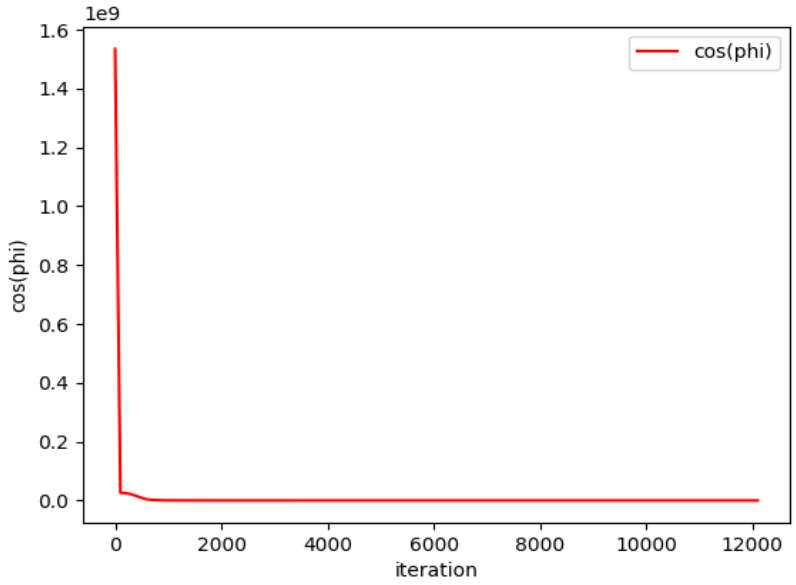


[Fig 1-6. Constraint (t1)]



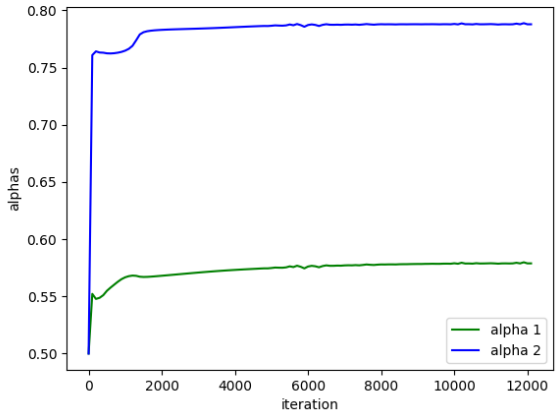
[Fig 1-7. Sum of probabilities (t1)]

The constraint has been increased to 6 until iteration 1900, but it converged to about 5 at the end. Also, we can find an interesting feature. The constraint graph's shape is similar to the graph's of the sum of probabilities and the value's difference between them is about 1. Therefore, it means that if we control the constraint, it is possible to restrict the sum of probabilities.

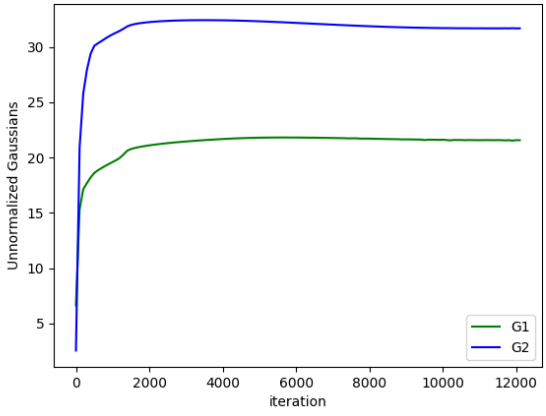


[Fig 1-8. Cosine (t1)]

The cosine has converged to 305.7096. Theoretically, it should be ranged from -1 to 1. I think it is because of the unconstrained optimization.



[Fig 1-9. Alphas (t1)]

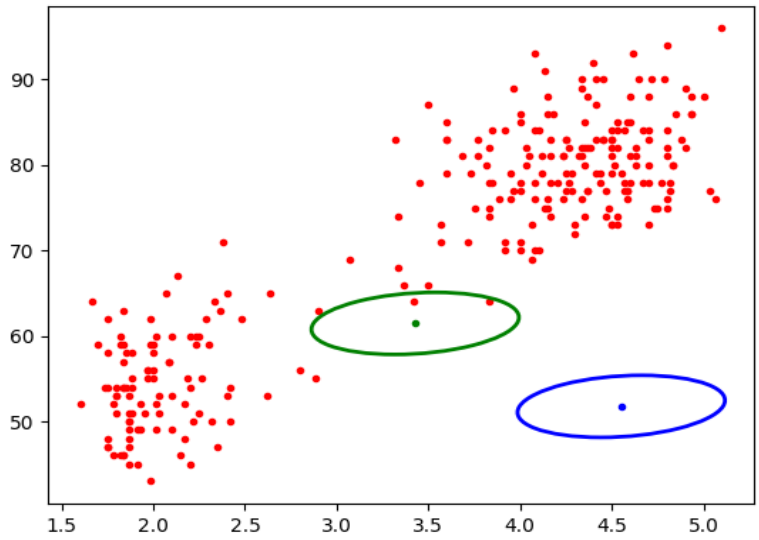


[Fig 1-10. Unnormalized Gaussians (t1)

In Fig 1-9, the alphas has been increased together, but the blue increased more sharply. Especially, we can see the initial degree of change was the greatest because while the clusters was going to the pile of the observations, the shapes were changed rapidly. After that, they have been increased gently. From the gentle lines in Fig 1-9, we can figure out that at about iteration 1800, the clusters were positioned to the center of the observations.

In Fig 1-10, initially the green one has more points than the blue one, but later the blue one has more points. Also, in iteration range from 2,000 and 12,000 in Fig 1-10, because the two lines are gentle we can see that the means and covariances were changed little by little.

[t2]

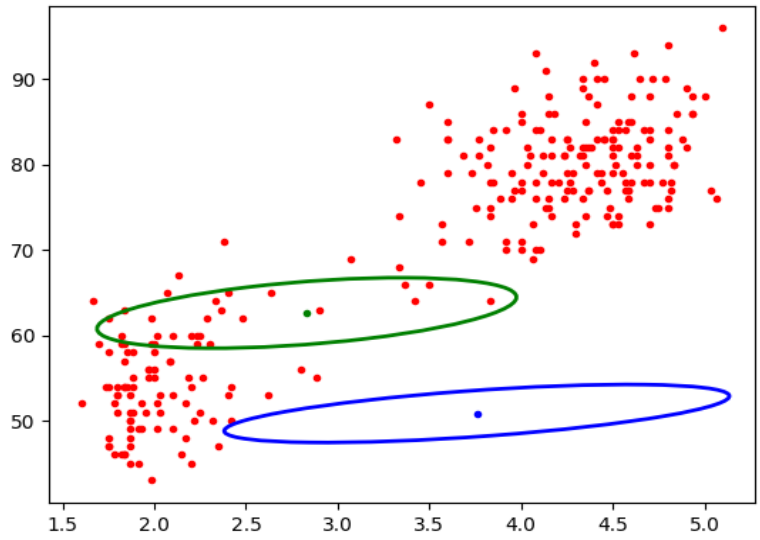


[Fig 2-1. Initial state (t2)]

[Green, Blue]

- **Alphas**
  : 0.5, 0.5
- **Unnormalized Gaussians**
  : 1.1552413, 1.3998942e-05
- **Cosine**
  : 4177031.5
- **Sum of probabilities**
  : 0.5564297

From the initial unnormalized Gaussians, we figure out that the blue cluster has a few observations. The cosine should be reduced to between -1 and 1 theoretically. The alphas are the same, so their scales are equal. Besides, the shapes of covariance are wide horizontally.

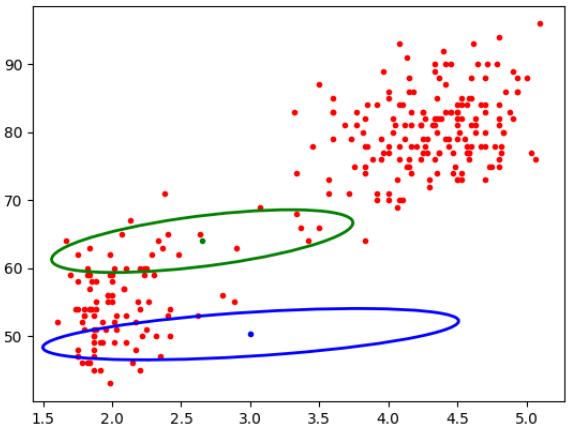


[Fig 2-2. Iteration 100 (t2)]

[Green, Blue]

- **Alphas**
  : 0.7367879, 0.36889336
- **Unnormalized Gaussians**
  : 6.871545, 1.7412033
- **Cosine**
  : 3143.8923
- **Sum of probabilities**
  : 1.0912439

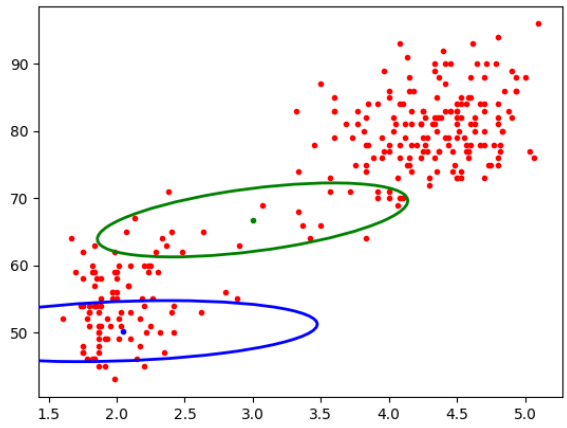
The clusters move to the left bottom pile together. All the values were changed largely compared with the initial state. Seeing the values, the training process was stable.



[Fig 2-3. Iteration 200 (t2)]

[Green, Blue]

- **Alphas**
  : 0.78161997, 0.44530806
- **Unnormalized Gaussians**
  : 7.3448105, 6.193829
- **Cosine**
  : 3143.8923
- **Sum of probabilities**
  : 1.3390088

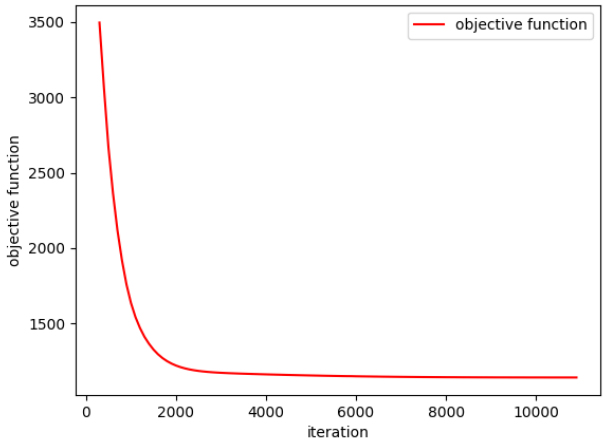


[Fig 2-4. Iteration 400 (t2)]

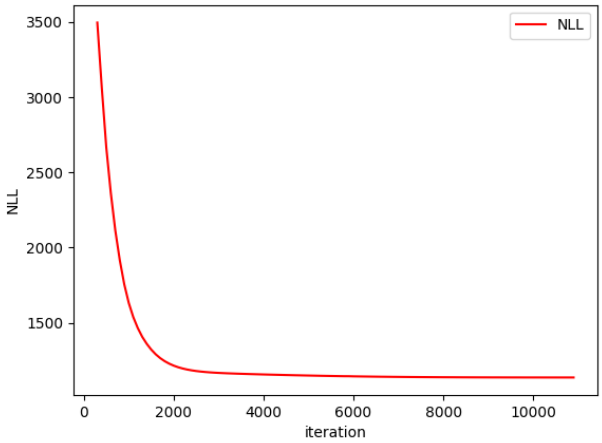
- **Alphas**
  : 0.8247638, 0.47997424
- **Unnormalized Gaussians**
  : 7.3361406, 12.233006
- **Cosine**
  : 9.686769
- **Sum of probabilities**
  : 1.4794102

I think this is the most interesting process in the experiment t2. Please look at the above two states together. It is interesting for the green cluster to move away from the a pile of the observations and the blue cluster took the place.

Also, while the unnormalized Gaussians were decreased, the cosine was decreased sharply.

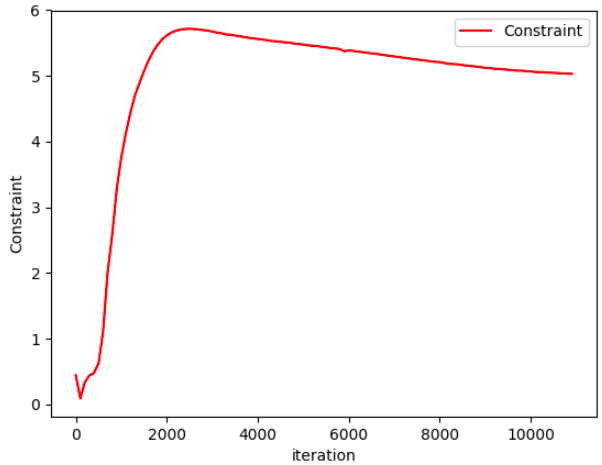


[Fig 2-5. Objective function (t2)

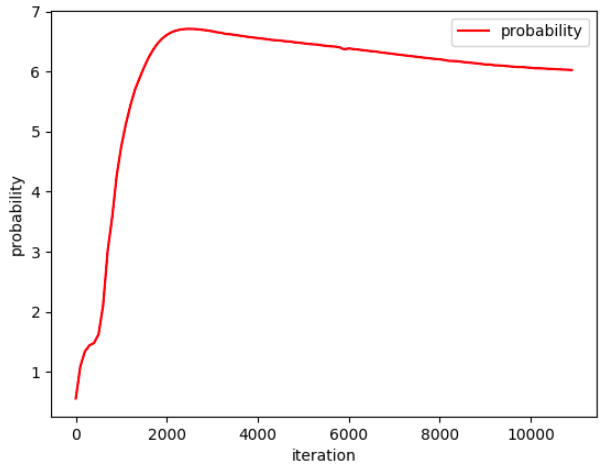


[Fig 2-6. Negative log-likelihood (t2)]

From the above graphs, we can figure out that the constrain hasn't much impact on the objective function.

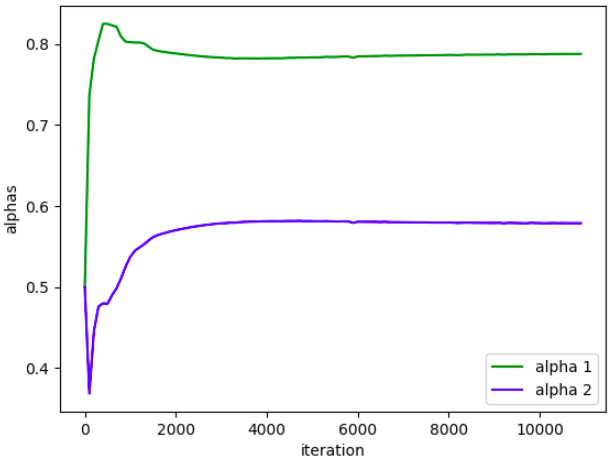


[Fig 2-7. Constraint (t2)

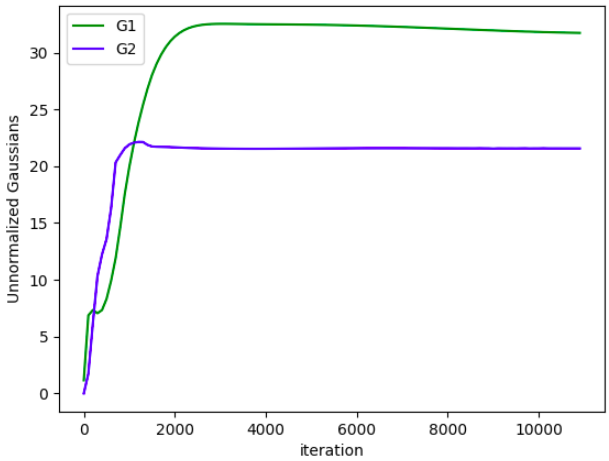


[Fig 2-8. Sum of Probabilities (t2)]

Except for early iterations, the shape of graphs are similar and actually the value difference is about 1.

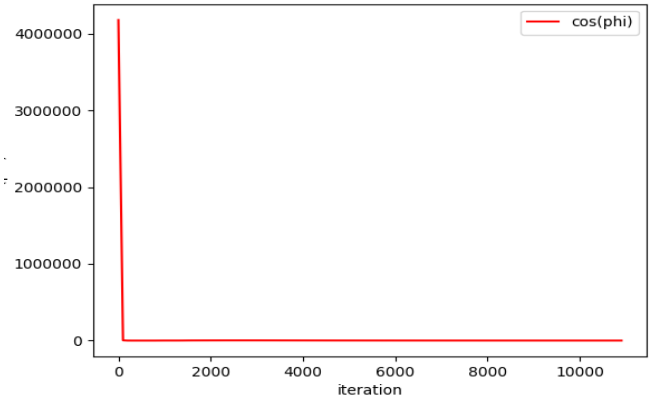


[Fig 2-9. Alphas (t2)



[Fig 2-10. Unnormalized Gaussians (t2)]

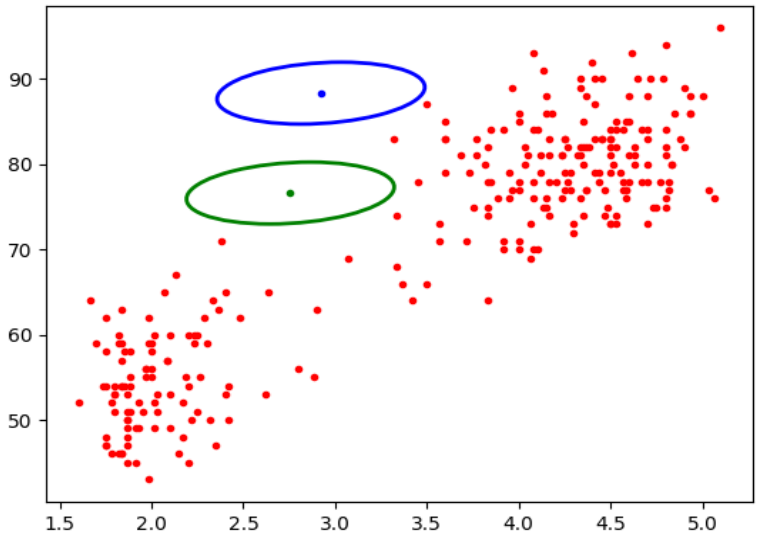
Likewise, from the above two graphs, although we can see the training process wasn't stable during the early stages, after that, the training process became stable.



[Fig 2-11. Cosine (t2)

The value of cosine converged to 325.88364. It should be from -1 to 1. we'll test it by changing the lambda to adjust the impact of the approximation constraint.

[t3]

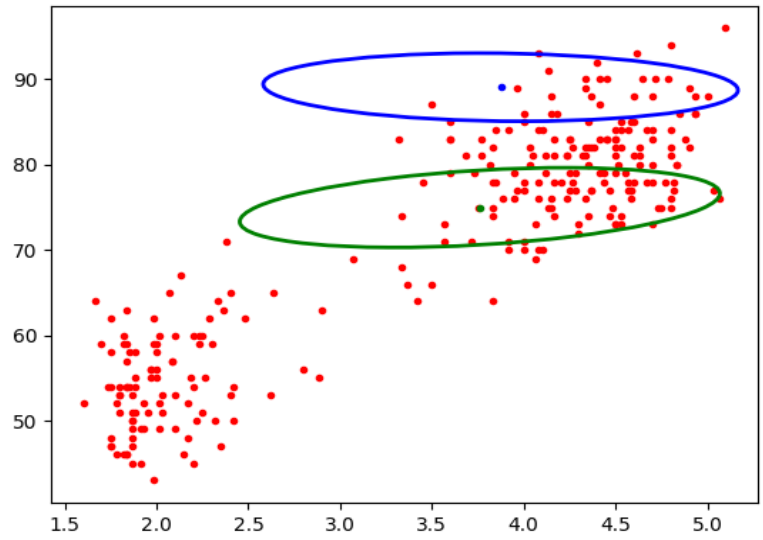


[Fig 3-1. Initial state (t3)

[Green, Blue]

- **Alphas**
  : 0.5, 0.5
- **Unnormalized Gaussians**
  : 0.5354648, 0.26814005
- **Cosine**
  : 1828.4346
- **Sum of probabilities**
  : 0.51283795

The alphas mean that the two clusters' scale are the same. From the values unnormalized Gaussians, the two clusters have a few observations at the initial state. The initial shapes of clusters are wide horizontally.

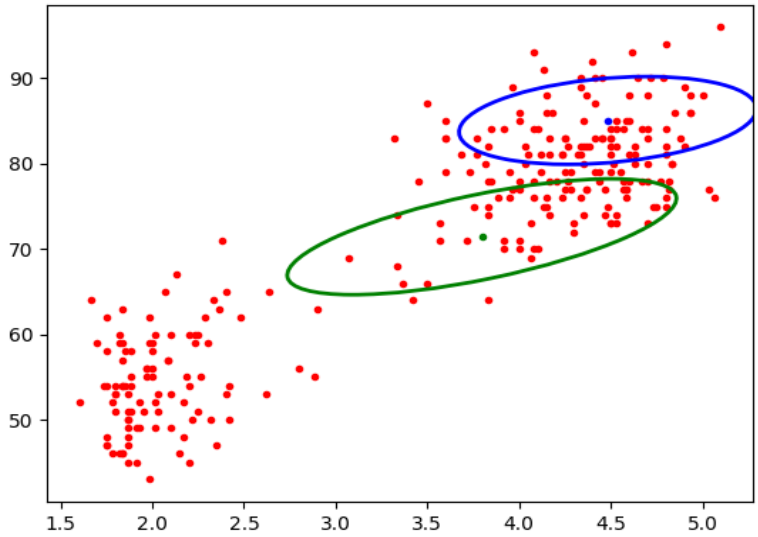


[Fig 3-2. Iteration 200 (t3)

[Green, Blue]

- **Alphas**
  : 0.75765216, 0.4059952
- **Unnormalized Gaussians**
  : 17.75705, 7.7367334
- **Cosine**
  : 16.371122
- **Sum of probabilities**
  : 2.5415406

The green cluster's scale is bigger than the blue one. From the unnormalized Gaussians, we can get the information that the green cluster has more observations.

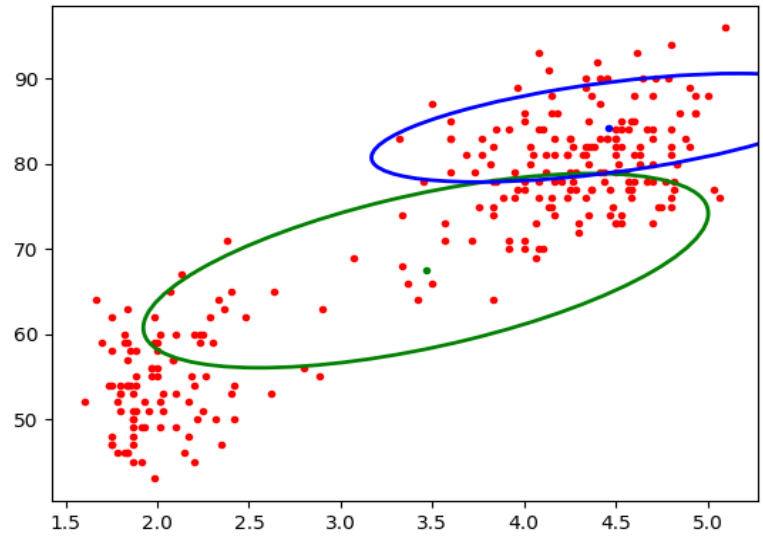


[Fig 3-3. Iteration 700 (t3)

[Green, Blue]

- **Alphas**
  : 3.9526238, 2.7919555
- **Unnormalized Gaussians**
  : 16.087805, 22.437603
- **Cosine**
  : -2.0724096
- **Sum of probabilities**
  : 67.03735

Although the size of the green cluster is bigger than the blue one, the blue cluster has more observations. Also, the sum of probabilities is very large, so we can see that the training process is very unstable.

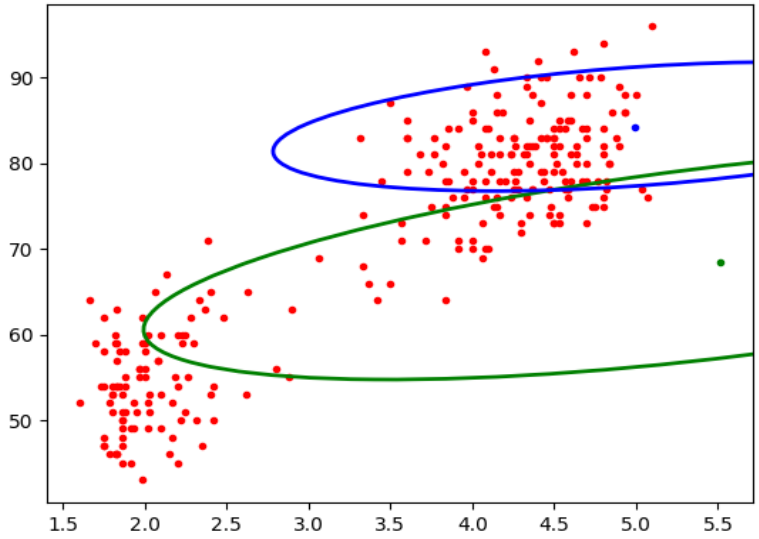


[Fig 3-4. Iteration 4200 (t3)

[Green, Blue]

- **Alphas**
  : 14.649853, 7.92133
- **Unnormalized Gaussians**
  : 17.0784, 22.488148
- **Cosine**
  : -1.3869946
- **Sum of probabilities**
  : 398.17413

Likewise, from the sum of probabilities, we can figure out that the current training is very unstable. Thus, the possibility to be diverged is very high.

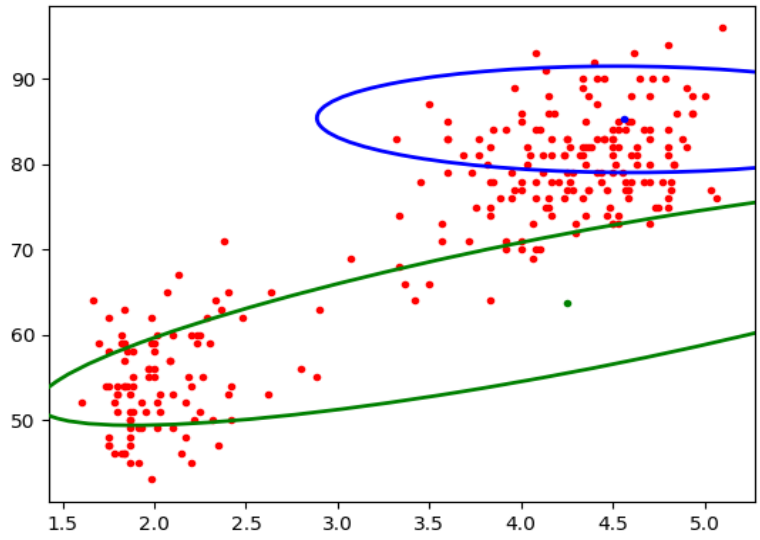


[Fig 3-5. Iteration 7100 (t3)

[Green, Blue]

- **Alphas**
  : 28.042501, 16.819433
- **Unnormalized Gaussians**
  : 8.447334, 18.521732
- **Cosine**
  : -3.2187788
- **Sum of probabilities**
  : -2.846777

We should finish this training because the sum of probabilities become negative. It seems that there were some problems seriously.

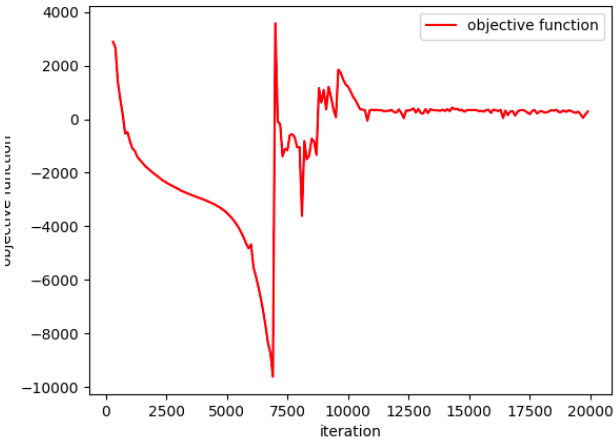


[Fig 3-6. Iteration 19900 (t3)

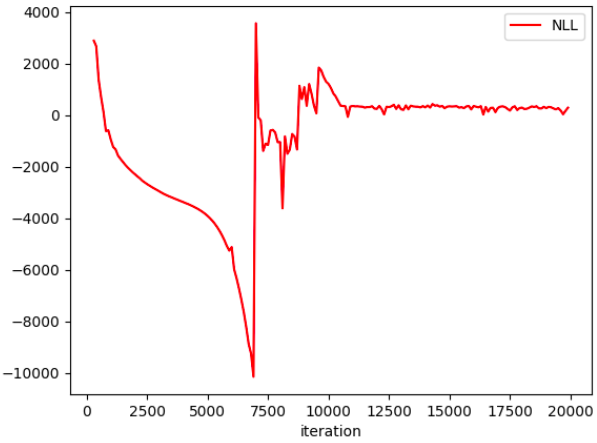
[Green, Blue]

- **Alphas**
  : 58.56975, 25.336065
- **Unnormalized Gaussians**
  : 8.825818, 18.543152
- **Cosine**
  : -26.35208
- **Sum of probabilities**
  : 4.638092

This is the last iteration of the experiment t3. Although the sum of probabilities became positive, it seems to be hard to be trained properly. Of course, the result was bad.

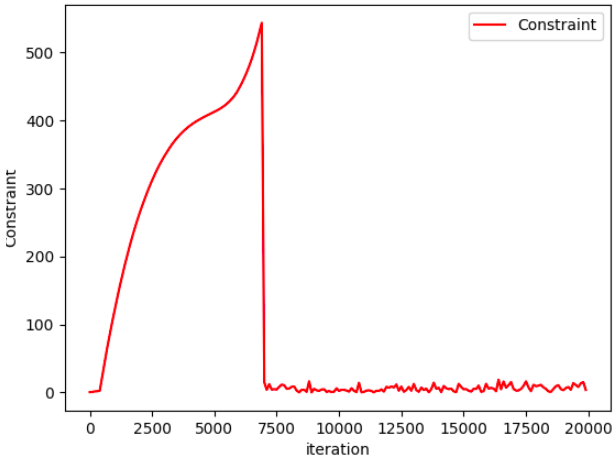


[Fig 3-7. Objective function (t3)

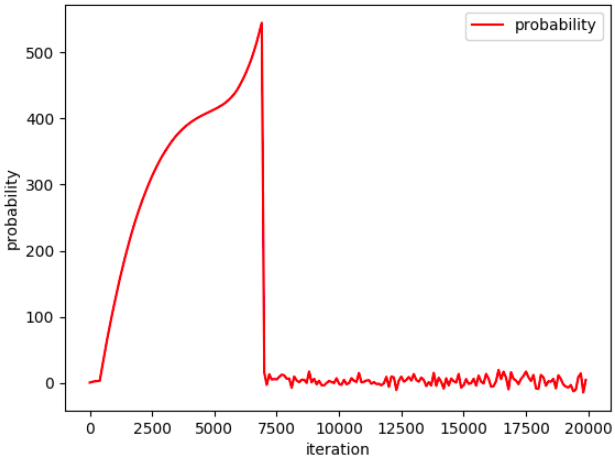


[Fig 3-8. Negative log-likelihood (t3)]

The graphs Fig 3-7 and 3-8 are so jagged. Thus, we can see the training process was so unstable. Also, judging from the shapes of them are equal, the constraint hasn't much impact on the training. Later, we'll dig into it by increasing or decreasing the lambda in the constraint.

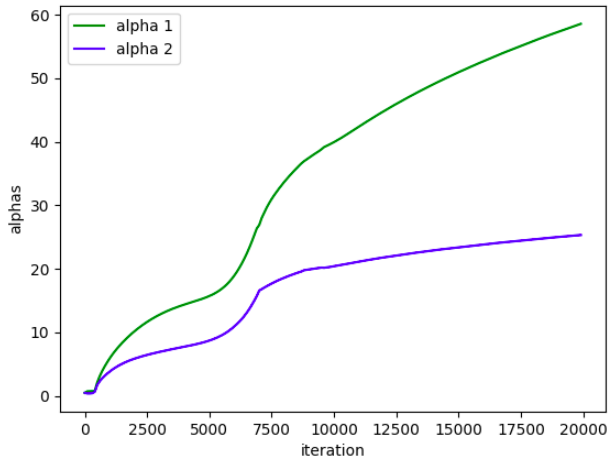


[Fig 3-9. Constraint (t3)

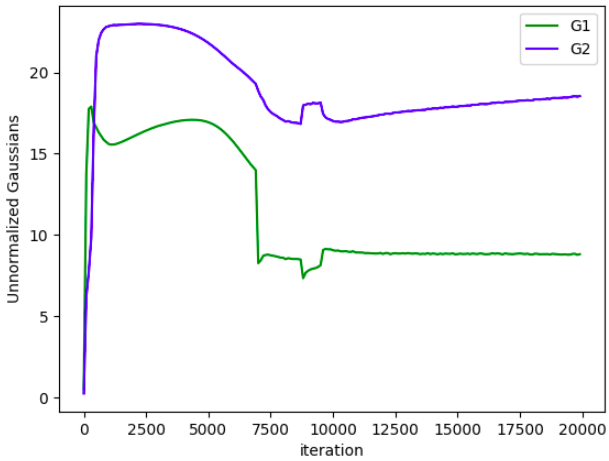


[Fig 3-10. Sum of Probabilities (t3)]

We can see that the shapes of them are equal. Therefore, it means that we can control the sum of probabilities by adjusting the lambda constant in the constraint.

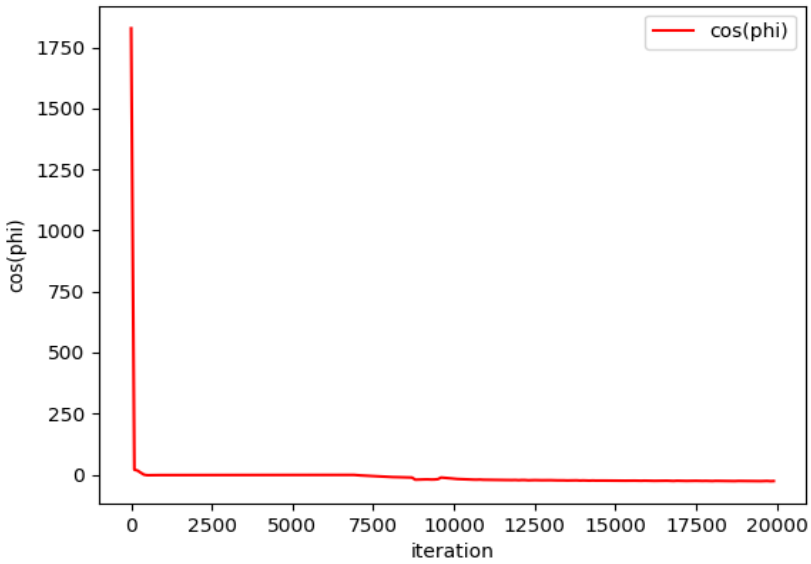


[Fig 3-11. Alphas (t3)



[Fig 3-12. Unnormalized Gaussians (t3)]

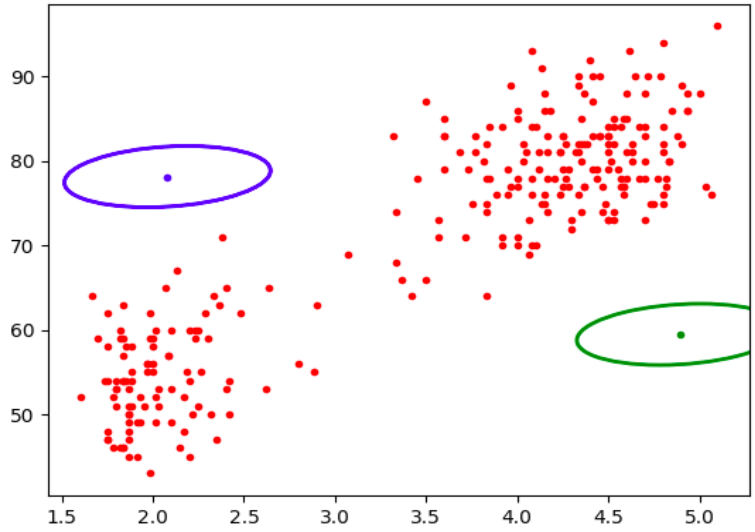
In Fig 3-11, the alphas has been increased continuously, so we should constraint them for preventing the parameters from diverging. Besides, the shapes of the unnormalized Gaussians are so strange. I think it is because of the unconstrained optimization.



[Fig 3-13. Cosine (t3)]

At iterations from 7500 to 10000, the line was twisted a bit like the unnormalized Gaussians. Thus, we can figure out that the training process was unstable during that period.

[t4]

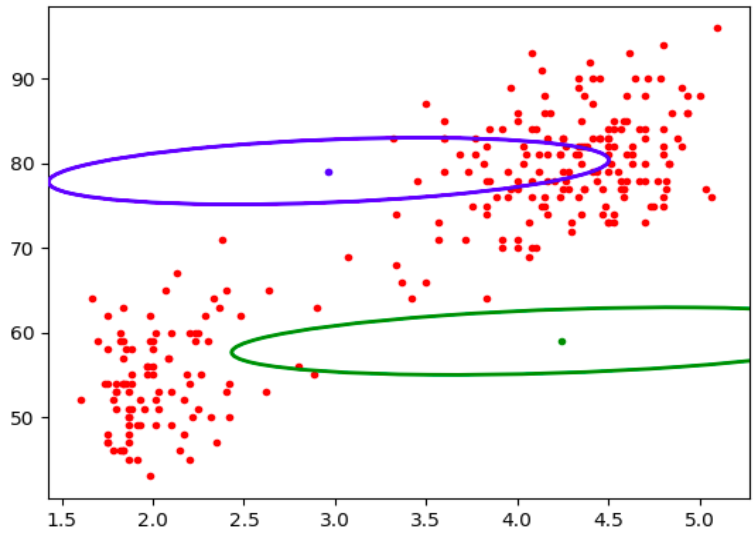


[Fig 4-1. Initial state (t4)

[Green, Blue]

- **Alphas**
  : 0.5, 0.5
- **Unnormalized Gaussians**
  : 0.0011769242, 0.009839533
- **Cosine**
  : 51669333000000.0
- **Sum of probabilities**
  : 0.50000757

Although the experiment t4 has low unnormalized Gaussians initially, it is one of the cases that the training is performed well as the initial clusters are scattered without interfering with each other. Also, we can see the cosine is very high, because the Gaussians are low.



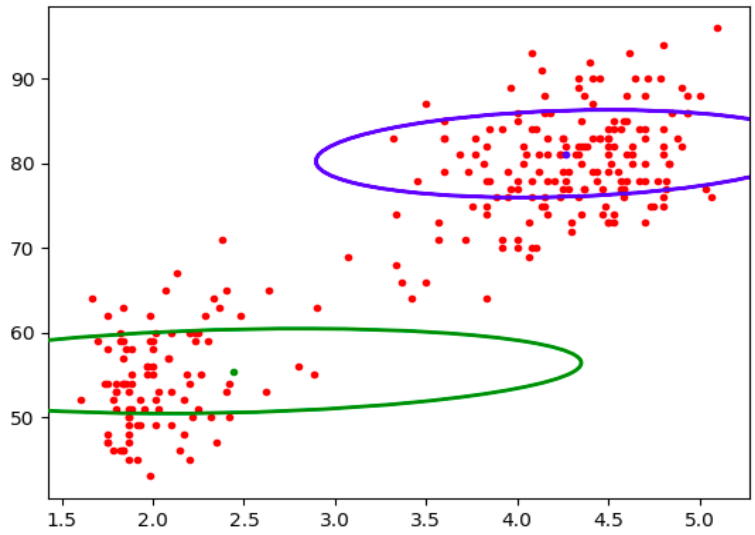
[Fig 4-2. Iteration 100 (t4)

[Green, Blue]

- **Alphas**
  : 0.5581918, 0.72645646
- **Unnormalized Gaussians**
  : 2.377851, 11.409248
- **Cosine**
  : 2002988.6
- **Sum of probabilities**
  : 0.98332864

The two clusters moved to the horizontal direction. it seems to be easy to move to the horizontal direction because of the scales between x-axis and y-axis.

When moved to the piles, we can see that the unnormalized Gaussians increased a lot, and due to that, the cosine became low compared with the iteration 100.



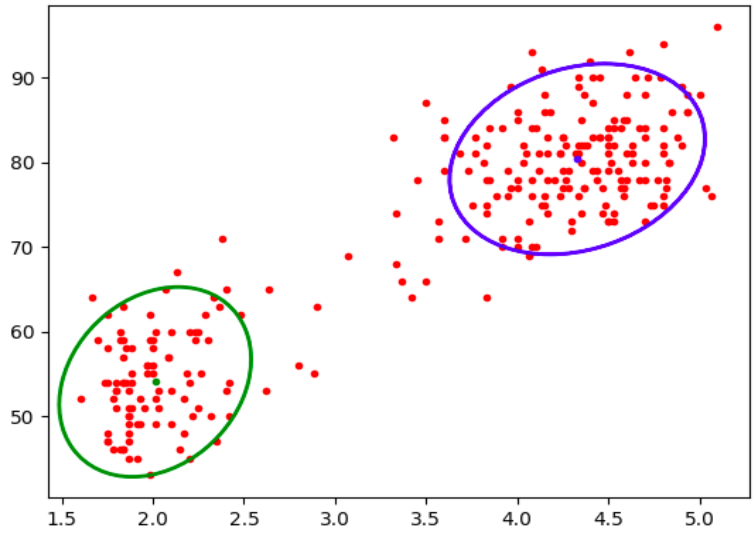
[Green, Blue]

- **Alphas**
  : 0.5650098, 0.74101704
- **Unnormalized Gaussians**
  : 11.36011, 25.701958
- **Cosine**
  : 56862.453
- **Sum of probabilities**
  : 3.8548665

[Fig 4-3. Iteration 500 (t4)

While the unnormalized Gaussians increased a lot, the cosine was decreased compared with the iteration 100. The unnormalized Gaussians means that how much each cluster has observations.

Seeing that the sum of probabilities wasn't 1, this training was performed with the unconstrained optimization.



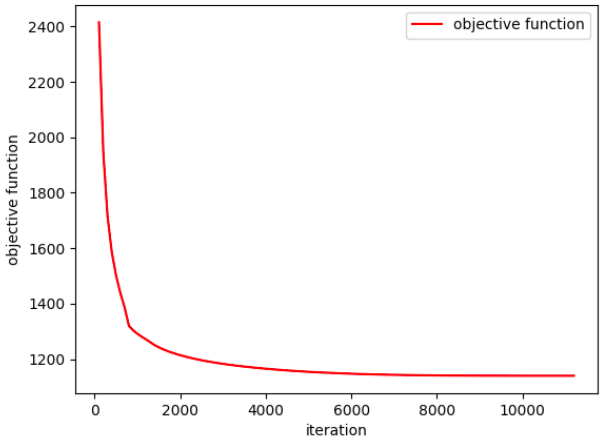
[Green, Blue]

- **Alphas**
  : 0.5786815,0.7879414
- **Unnormalized Gaussians**
  : 21.58389, 31.67678
- **Cosine**
  : 311.33807
- **Sum of probabilities**
  : 6.011079

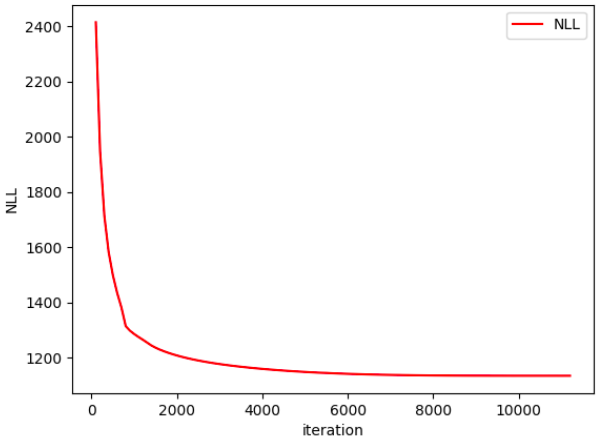
[Fig 4-4. Iteration 11200 (t4)

Fig is the last state in t4. Seeing that the alphas and the unnormalized Gaussians, the blue one is bigger and has more observations than the green one.

Also, the cosine and the sum of probabilities were converged to about 311 and 6 respectively. From those values, we can figure out that our training was performed with the unconstrained optimization.

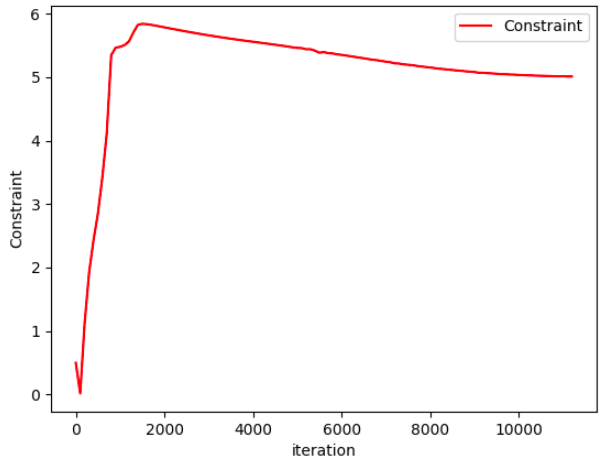


[Fig 4-5. Objective function (t4)

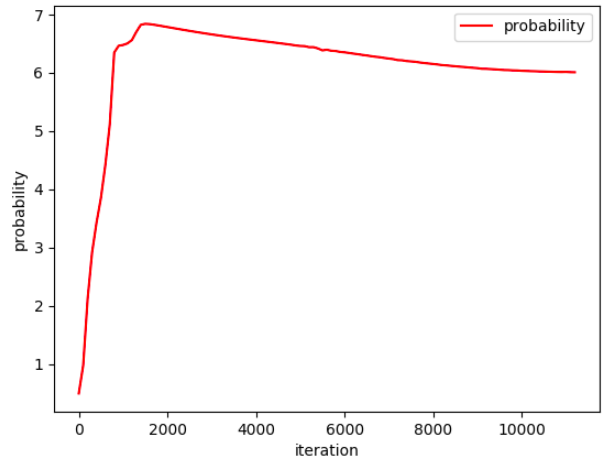


[Fig 4-6. Negative log-likelihood (t4)]

Seeing that the two graphs are similar, our constraint doesn't much impact on the training process. Thus, we can figure out the training was performed with the unconstrained or weak-constrained optimization. Besides, the value of the objective function has been decreased gently,

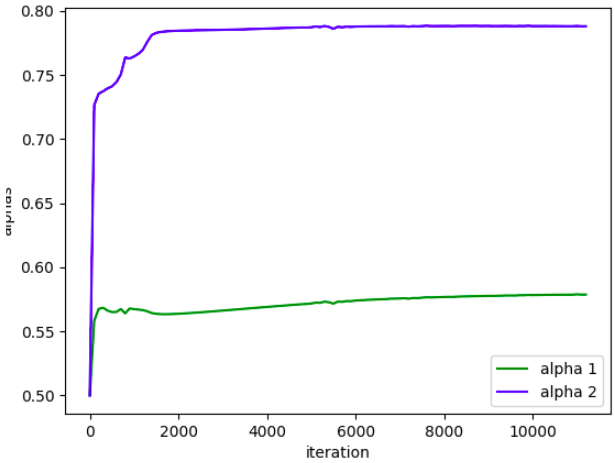


[Fig 4-7. Constraint (t4)

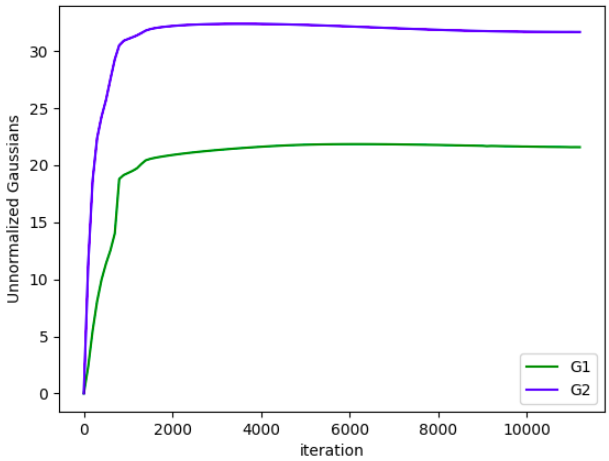


[Fig 4-8. Sum of Probabilities (t4)]

From that the shape of the above two graphs are similar, we can figure out that if we control the constraint, we can restrict the sum of probabilities as well. Thus, we can try this experiment with the more constrained optimization than now. Maybe, it would be useful if we failed to train with the unconstrained optimization.

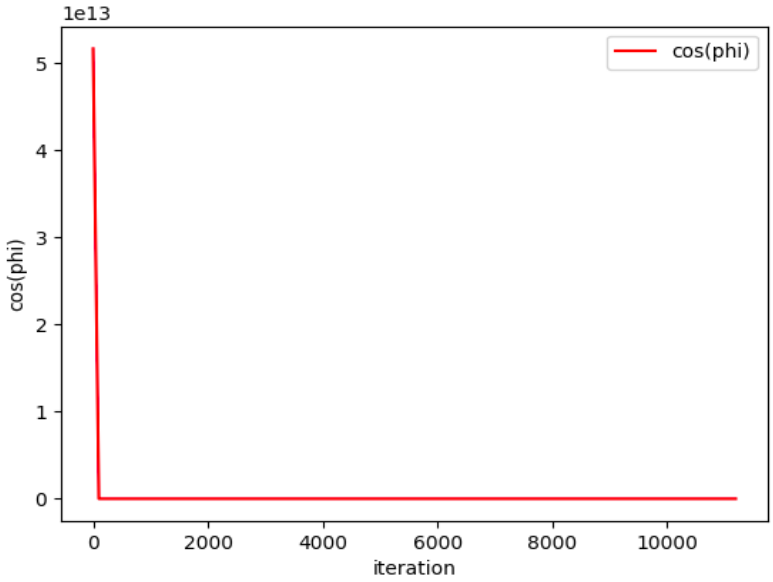


[Fig 4-9. Alphas (t4)



[Fig 4-10. Unnormalized Gaussians (t4)]

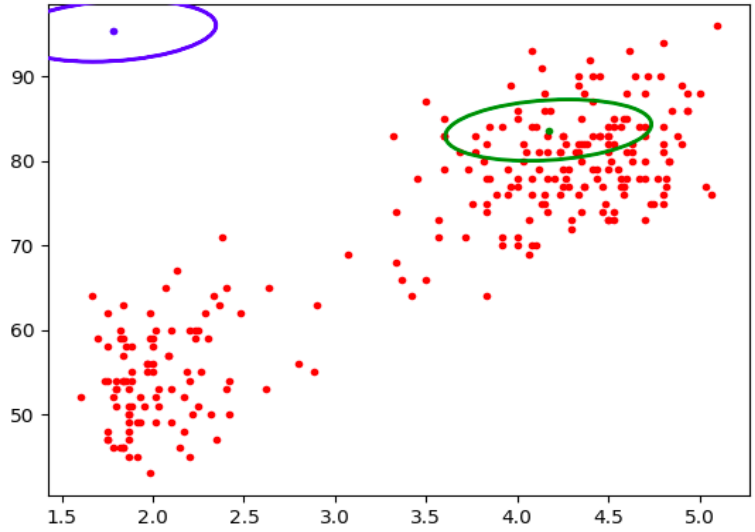
Seeing that the shapes of the above graphs are twisted, at the iterations between 0 ~ 2,000, there were many changes for the parameters in clusters. After that, from the shapes became gently, the parameters of the clusters were updated little by little.



[Fig 4-11. Cosine (t4)]

The cosine has been converged to about 300, and from the shape of the cosine, we can guess the alphas and Gaussians were changed smoothly.

[t5]

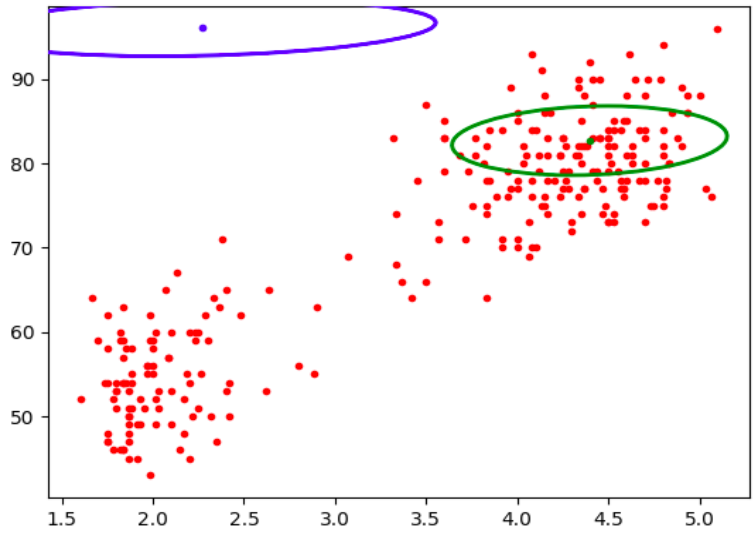


[Fig 5-1. Initial state (t5)

[Green, Blue]

- **Alphas**
  : 0.5, 0.5
- **Unnormalized Gaussians**
  : 21.59848, 1.3045009e-08
- **Cosine**
  : 3122437400.0
- **Sum of probabilities**
  : 2.2268722

Seeing that the alphas and the unnormalized Gaussians, they have the same scale and the green cluster has much more observations initially.

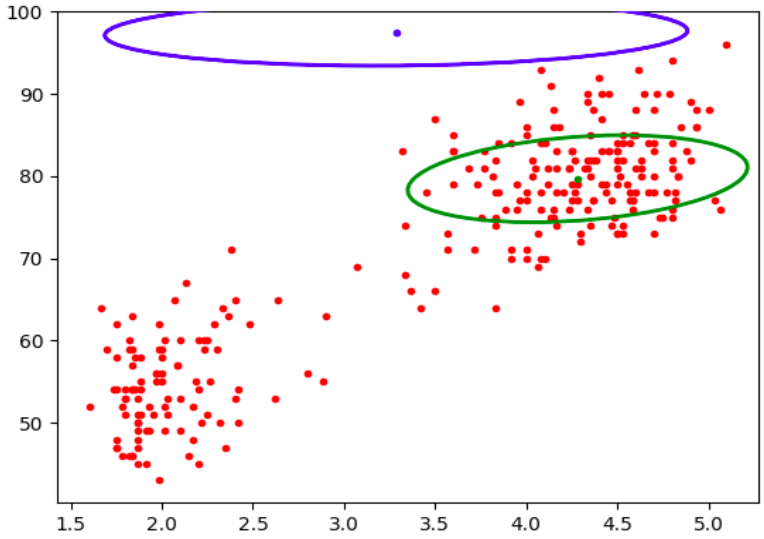


[Fig 5-2. Iteration 100 (t5)

[Green, Blue]

- **Alphas**
  : 0.95344657, -0.00036870688
- **Unnormalized Gaussians**
  : 26.654049,0.024954103
- **Cosine**
  : -2510764.0
- **Sum of probabilities**
  : 7.4304276

Both of them moved to the left. The alphas and the unnormalized Gaussians were almost zero.

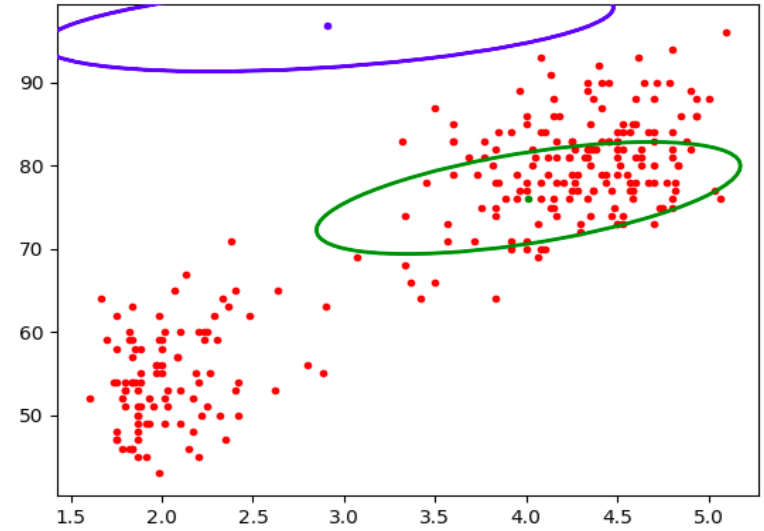


[Fig 5-3. Iteration 600 (t5)

[Green, Blue]

- **Alphas**
  : 0.9343704, -3.2536795e-05
- **Unnormalized Gaussians**
  : 29.61023, 0.3102831
- **Cosine**
  : -4323375.0
- **Sum of probabilities**
  : 6.602552

The unnormalized Gaussian of the blue cluster increased a bit compared with the iteration 100. It seems to find a pile of the observations that the blue one should position to.

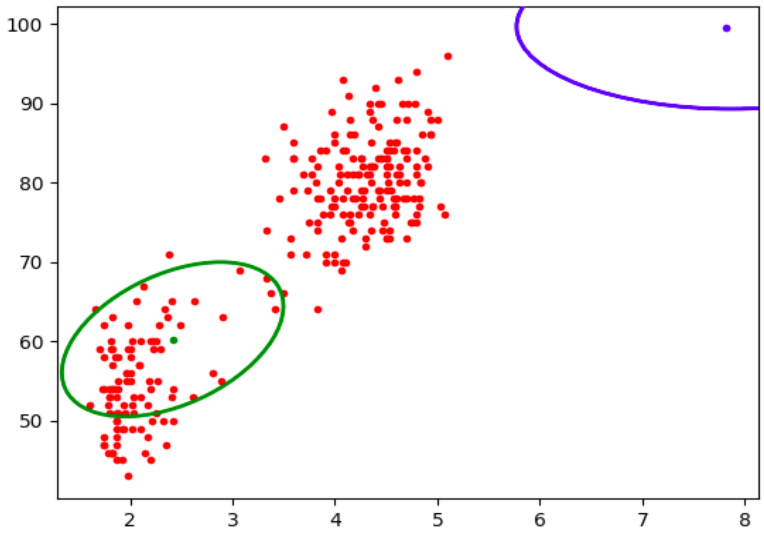


[Fig 5-4. Iteration 1200 (t5)

[Green, Blue]

- **Alphas**
  : 0.90868163, 3.136229e-05
- **Unnormalized Gaussians**
  : 24.625912, 0.18130603
- **Cosine**
  : 18495126.0
- **Sum of probabilities**
  : 4.388392

Suddenly, the blue cluster moved back. The unnormalized Gaussians decreased a bit compared with the iteration 600.

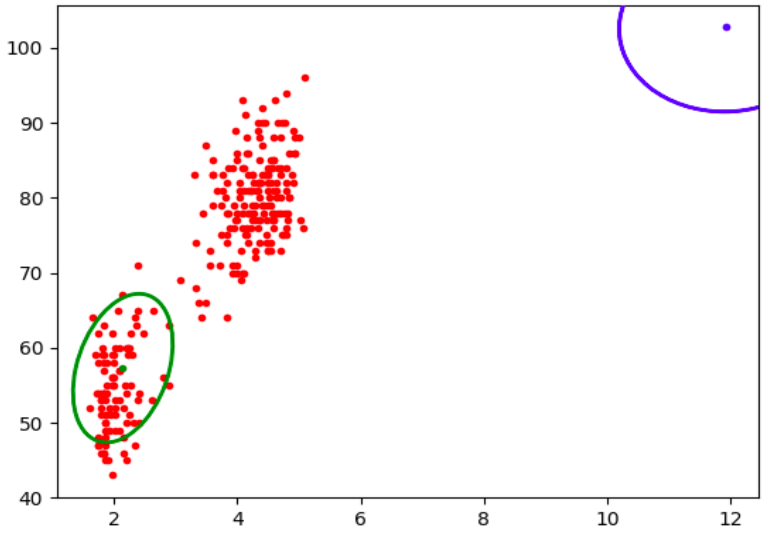


[Fig 5-5. Iteration 3300 (t5)]

[Green, Blue]

- **Alphas**
  : 0.62676644, 0.00012729265
- **Unnormalized Gaussians**
  : 14.612187, 0.19712558
- **Cosine**
  : 114669470.0
- **Sum of probabilities**
  : 1.5372882

Strangely, the blue cluster has been moved to the left more and more. The alpha of the blue was still almost zero.

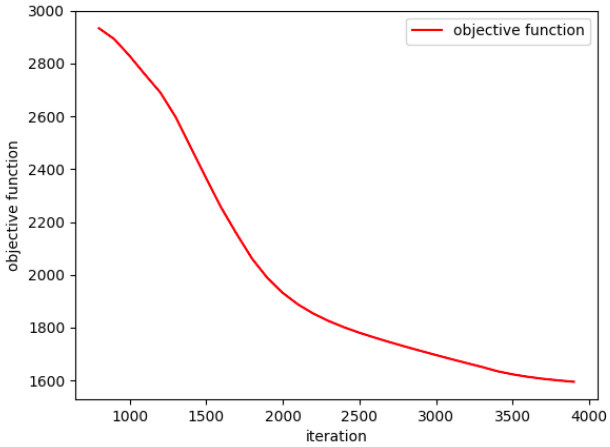


[Fig 5-6. Iteration 3900 (t5)]

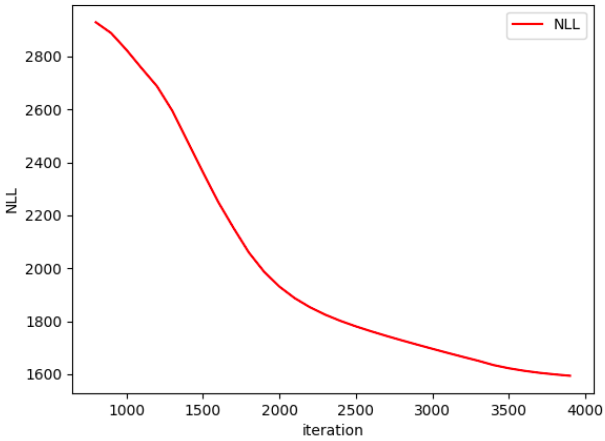
[Green, Blue]

- **Alphas**
  : 0.6055952,0.00057936425
- **Unnormalized Gaussians**
  : 18.111656, 7.9755985e-08
- **Cosine**
  : 1.945357e+16
- **Sum of probabilities**
  : 2.0234442

While the green cluster converged, the blue one diverged. The alpha of the blue was still about zero.



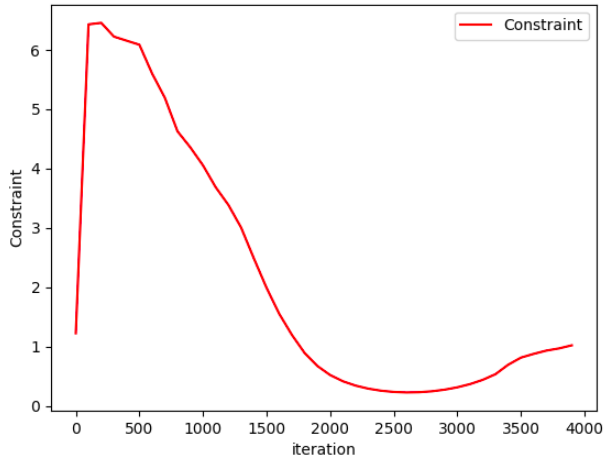
[Fig 5-7. Objective function (t5)



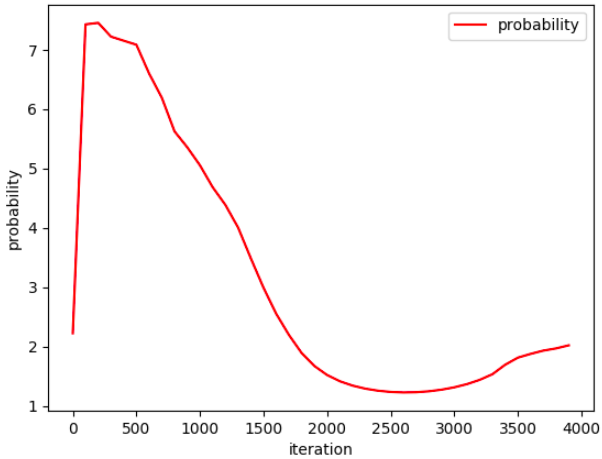
[Fig 5-8. Negative log-likelihood (t5)]

Seeing that the two graphs are similar, our constraint doesn't much impact on the training process. I can guess that our optimization is unconstrained or weak-constrained.

Although the value of the objective function has decreased continuously, because it is doesn't guarantee a good result, we should look at other graphs that indicate the training process.

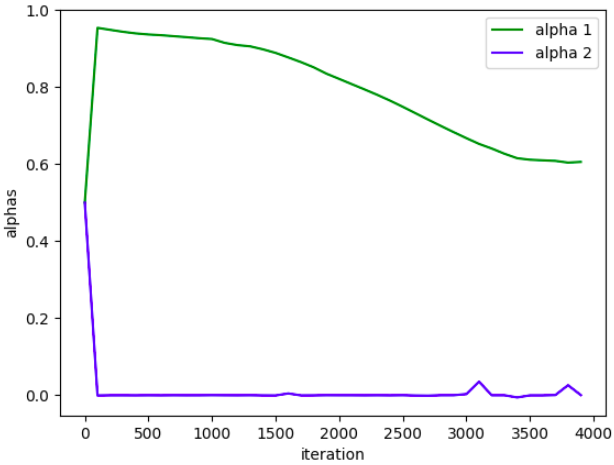


[Fig 5-9. Constraint (t5)

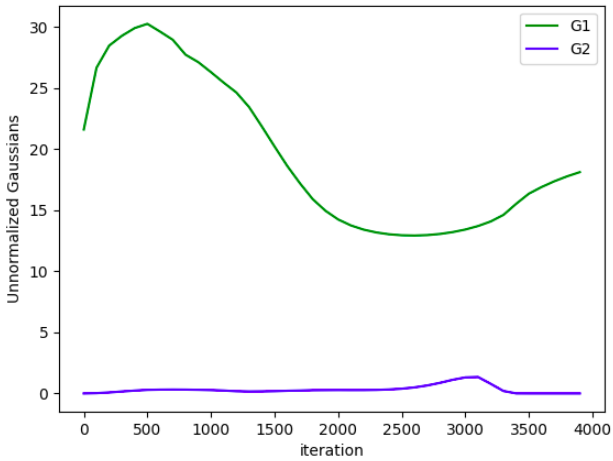


[Fig 5-10. Sum of Probabilities (t5)]

The both lines are similar and the value difference is about 1, therefore, we can figure out that if we control the constraint, we can restrict the sum of probabilities as well.



[Fig 5-11. Alphas (t5)]

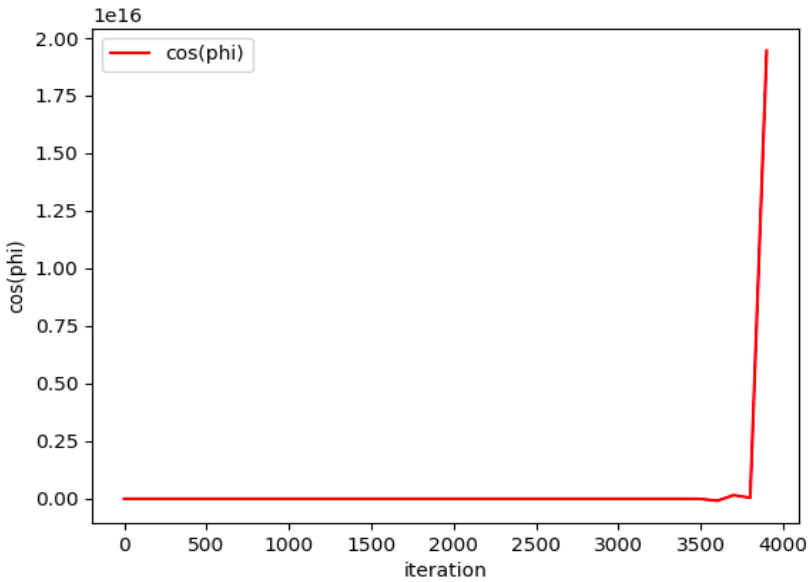


[Fig 5-12. Unnormalized Gaussians (t5)]

I think that from the above graphs, we can figure out that why the result was bad. According to the graphs, the alpha and the unnormalized Gaussian of the blue one were almost zero while the green's were very high. Consequently, I think this training was performed excluding the blue cluster.

Also, the reason that the blue cluster moves to the left continuously would be because of the tf.clip_by_value() that is set to avoid log(0) operation in the Python implementation.

Therefore, we can see that this training was performed excluding the blue cluster because the green cluster had much impact initially.



[Fig 5-13. Cosine (t5)]

Moreover, the cosine was so strange. Theoretically, while training the parameters, the cosine should be between -1 and 1. Because the unnormalized Gaussian of the blue cluster has been to zero while training, at the end, the cosine became very high.

Conclusions

From several experiments, we found some valuable findings that effect on the training performance. First, the training performance is sensitive to the initial values. Therefore, it is important to scatter the initial clusters with a gap to avoid moving to the same pile of the observations. Also, it would be good to set the initial means and covariances according to the scales that axes of the data set have for better training.

Second, we figured out that there are some cases that are sensitive to the unconstrained or weak-constrained optimization like the experiments t3 and t5. The two cases showed a bad result in clustering. Especially, the experiment t3 showed the unconstrained shapes on the graphs of alphas. While the alphas increased continuously, the optimizer couldn't minimize the objective function very well. In the experiment t5, the blue one was excluded from the optimization. Although the green cluster has been trained well, the blue cluster hasn't. Seeing that the shapes of graphs were totally different from the nice cases, we can try it with more constrained optimization again.

Lastly, we could see the possibility of the constraint from some graphs because the experiment t3 and t5 seem to be associated with the unconstrained optimization. Thus, the next research is to check the impact of the lambda in training that was trick in this experiments.