Gaussian Mixture Model

Mathematical Foundation for Artificial Intelligence
Lecture Notes

Ganguk Hwang

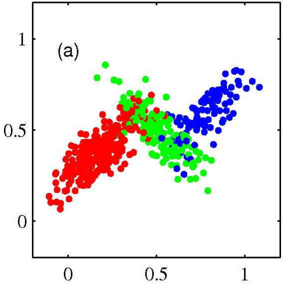
Department of Mathematical Sciences
KAIST

Gaussian Mixture Model

Reference: Christopher M. Bishop, Pattern Recognition and Machine Learning, Springer 2006.

Hard Assignement vs. Soft Assignment

- k-means clustering is a sort of a hard assignment of observations to clusters.
- However, for observations near the decision boundaries, hard assignment of obserations may not be a good idea.
- Instead, we could think about making a soft assignment of observations to clusters



Gaussian Mixutre Model

Gaussian Mixture Model (GMM) assumes that data points $\{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N\}$ are generated by different Gaussian distributions as

$$p(\mathbf{x}_i) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}_i | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

where $\sum_{k=1}^{K} \pi_k = 1$.

We use a latent random variable $\mathbf{z}_i = (z_{i1}, z_{i2}, \cdots, z_{iK})$ with $z_{ik} \in \{0, 1\}$, $\sum_{k=1}^{K} z_{ik} = 1$, and

$$p(z_{ik} = 1) = \pi_k, \quad 1 \leq k \leq K.$$

$$p(\mathbf{z}_i) = \prod_{k=1}^{K} \pi_k^{z_{ik}}$$

Note that

$$p(\mathbf{x}_i|z_{ik} = 1) = \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k),$$

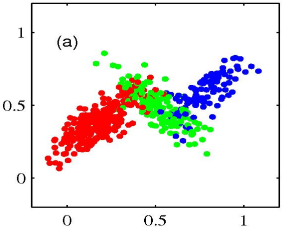
$$p(\mathbf{x}_i|\mathbf{z}_i) = \prod_{k=1}^{K} \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)^{z_{ik}}$$

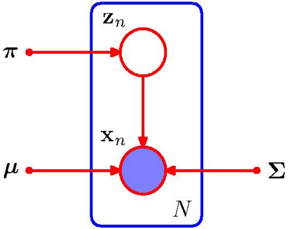
where $\mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ denotes the pdf of a normal distribution. Moreover,

$$\begin{aligned} p(\mathbf{x}_i, \mathbf{z}_i) &= p(\mathbf{x}_i|\mathbf{z}_i)p(\mathbf{z}_i) \\ &= \prod_{k=1}^{K} \pi_k^{z_{ik}} \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)^{z_{ik}}, \\ p(\mathbf{x}_i) &= \sum_{\mathbf{z}_i} p(\mathbf{x}_i, \mathbf{z}_i) \\ &= \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k). \end{aligned}$$

We now consider the joint probability of all data points $\mathbf{x} := \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N\}$ and $\mathbf{z} = \{\mathbf{z}_1, \mathbf{z}_2, \cdots, \mathbf{z}_N\}$.

$$
\begin{aligned}
p(\mathbf{x}, \mathbf{z}) &= \prod_{i=1}^{N} p(\mathbf{x}|\mathbf{z})p(\mathbf{z}) \\
&= \prod_{i=1}^{N} \prod_{k=1}^{K} \pi_k^{z_{ik}} \mathcal{N}(\mathbf{x}_i | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)^{z_{ik}}.
\end{aligned}
$$





It is important to get $p(z_{ik} = 1|\mathbf{x}_i)$. Observe that

$$\begin{aligned}
p(z_{ik} = 1|\mathbf{x}_i) &= \frac{p(z_{ik} = 1, \mathbf{x}_i)}{p(\mathbf{x}_i)} \\
&= \frac{p(z_{ik} = 1, \mathbf{x}_i)}{\sum_{l=1}^{K} p(\mathbf{x}_i, z_{il} = 1)} \\
&= \frac{p(\mathbf{x}_i|z_{ik} = 1)p(z_{ik} = 1)}{\sum_{l=1}^{K} p(\mathbf{x}_i|z_{il} = 1)p(z_{il} = 1)} \\
&= \frac{\pi_k \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{l=1}^{K} \pi_l \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_l, \boldsymbol{\Sigma}_l)}.
\end{aligned}$$

In the GMM, we have the following parameters to learn:

$$\boldsymbol{\theta} := \{\pi_k, \boldsymbol{\mu}_k, \Sigma_k,\ 1 \le k \le K\}.$$

To this end, we consider the log likelihood function of $\mathbf{x}$ and find

$$\begin{aligned}\boldsymbol{\theta}^* &= \arg\max_{\boldsymbol{\theta}} \log p(\mathbf{x}|\boldsymbol{\theta}) \\ &= \arg\max_{\boldsymbol{\theta}} \sum_{i=1}^{N} \log \left(\sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right)\end{aligned}$$

Recall that, for $\mathbf{x}_i \in \mathbb{R}^d$

$$\mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) = \frac{1}{\sqrt{(2\pi)^d \text{det}(\boldsymbol{\Sigma}_k)}} e^{-\frac{1}{2}(\mathbf{x}_i - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1}(\mathbf{x}_i - \boldsymbol{\mu}_k)}.$$

We are now ready to learn the model. The learning process of the GMM is very similar to that of k-means clustering. That is,

- If we know latent variables $\mathbf{z}_i$, then we learn the Gaussian parameters $\boldsymbol{\mu}_k$ and Σ_k.
- If we know the Gaussian parameters $\boldsymbol{\mu}_k$ and Σ_k, then we learn the latent variables $\mathbf{z}_i$.

Let

$$J(\boldsymbol{\theta}) := \sum_{i=1}^{N} \log \left(\sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}_i | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right).$$

We then have

$$\begin{aligned}
\nabla_{\boldsymbol{\mu}_k} J(\boldsymbol{\theta}) &= -\sum_{i=1}^{N} \frac{\pi_k \mathcal{N}(\mathbf{x}_i | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{l=1}^{K} \pi_l \mathcal{N}(\mathbf{x}_i | \boldsymbol{\mu}_l, \boldsymbol{\Sigma}_l)} \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k) \\
&= -\sum_{i=1}^{N} p(z_{ik} = 1 | \mathbf{x}_i) \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k).
\end{aligned}$$

Letting $\nabla_{\boldsymbol{\mu}_k} J(\boldsymbol{\theta}) = \mathbf{0}$ yields

$$\sum_{i=1}^{N} p(z_{ik} = 1 | \mathbf{x}_i) \boldsymbol{\Sigma}_k^{-1} \mathbf{x}_i = \sum_{i=1}^{N} p(z_{ik} = 1 | \mathbf{x}_i) \boldsymbol{\Sigma}_k^{-1} \boldsymbol{\mu}_k.$$

It then follows that

$$\boldsymbol{\mu}_k = \sum_{i=1}^{N} \frac{p(z_{ik} = 1|\mathbf{x}_i)}{\sum_{i=1}^{N} p(z_{ik} = 1|\mathbf{x}_i)} \mathbf{x}_i.$$

Note that $\sum_{i=1}^{N} p(z_{ik} = 1|\mathbf{x}_i)$ is the *effective* number of data points in cluster k.

Similarly, from

$$
\begin{aligned}
\frac{\partial}{\partial \boldsymbol{\Sigma}_k} J(\boldsymbol{\theta}) =& -\frac{1}{2}\sum_{i=1}^{N} \frac{\pi_k \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{l=1}^{K} \pi_l \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_l, \boldsymbol{\Sigma}_l)} \boldsymbol{\Sigma}_k^{-1} \\
&+ \frac{1}{2}\sum_{i=1}^{N} \frac{\pi_k \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{l=1}^{K} \pi_l \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_l, \boldsymbol{\Sigma}_l)} \boldsymbol{\Sigma}_k^{-1}(\mathbf{x}_i - \boldsymbol{\mu}_k)(\mathbf{x}_i - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1} \\
=& -\frac{1}{2}\sum_{i=1}^{N} p(z_{ik} = 1|\mathbf{x}_i)\boldsymbol{\Sigma}_k^{-1} \\
&+ \frac{1}{2}\sum_{i=1}^{N} p(z_{ik} = 1|\mathbf{x}_i)\boldsymbol{\Sigma}_k^{-1}(\mathbf{x}_i - \boldsymbol{\mu}_k)(\mathbf{x}_i - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1} = \mathbf{0},
\end{aligned}
$$

we get

$$
\boldsymbol{\Sigma}_k = \sum_{i=1}^{N} \frac{p(z_{ik} = 1|\mathbf{x}_i)}{\sum_{l=1}^{N} p(z_{lk} = 1|\mathbf{x}_l)} (\mathbf{x}_i - \boldsymbol{\mu}_k)(\mathbf{x}_i - \boldsymbol{\mu}_k)^\top.
$$

Remark 1.

Let $\det(\mathbf{A})$ be the determinant of matrix $\mathbf{A} = (a_{ij})$ of size n, and $\mathbf{C} = (c_{ij})$ be the cofactor matrix of $\mathbf{A}$. We know that

$$\begin{aligned}
\det(\mathbf{A}) &= \sum_{i=1}^{n} a_{ij} c_{ij} \text{ for any } j \\
&= \sum_{j=1}^{n} a_{ij} c_{ij} \text{ for any } i, \\
\mathbf{A}^{-1} &= \frac{1}{\det(\mathbf{A})} \operatorname{adj}(\mathbf{A})
\end{aligned}$$

where $\operatorname{adj}(\mathbf{A}) = \mathbf{C}^\top$ is the adjoint matrix of $\mathbf{A}$, that is, the transpose of the cofactor matrix $\mathbf{C}$. Then, the derivatives of the log determinant are given by

$$\frac{\partial}{\partial a_{ij}} \det(\mathbf{A}) = c_{ij} = \det(\mathbf{A})(\mathbf{A}^{-1})_{ji}.$$

Remark 2.

From $\frac{d\mathbf{A}^{-1}}{d\theta} = -\mathbf{A}^{-1}\frac{d\mathbf{A}}{d\theta}\mathbf{A}^{-1}$, we see that

$$\frac{\partial}{\partial a_{mn}}(\mathbf{A}^{-1})_{ij} = -(\mathbf{A}^{-1})_{im}(\mathbf{A}^{-1})_{nj}.$$

Remark 3.

Let $f(\mathbf{A}) = \mathbf{y}^\top \mathbf{A}^{-1}\mathbf{y} = \sum_i \sum_j y_i y_j (\mathbf{A}^{-1})_{ij}$. Then,

$$\begin{aligned}
\frac{\partial f(\mathbf{A})}{\partial a_{mn}} &= \sum_i \sum_j y_i y_j \frac{\partial}{\partial a_{mn}}(\mathbf{A}^{-1})_{ij} \\
&= -\sum_i \sum_j y_i y_j (\mathbf{A}^{-1})_{im}(\mathbf{A}^{-1})_{nj} \\
&= -(\mathbf{y}^\top \mathbf{A}^{-1})_m (\mathbf{A}^{-1}\mathbf{y})_n,
\end{aligned}$$

i.e.,

$$\frac{\partial f(\mathbf{A})}{\partial \mathbf{A}} = \left(\frac{\partial}{\partial a_{mn}} f(\mathbf{A})\right) = -(\mathbf{A}^{-1})^\top \mathbf{y}\mathbf{y}^\top (\mathbf{A}^{-1})^\top.$$

Considering $\sum_{k=1}^{K} \pi_k = 1$ and a Lagrange multiplier λ, we formulate

$$\mathcal{L}(\boldsymbol{\theta}) := \log p(\mathbf{x}|\boldsymbol{\theta}) + \lambda(\sum_{k=1}^{K} \pi_k - 1) = J(\boldsymbol{\theta}) + \lambda(\sum_{k=1}^{K} \pi_k - 1).$$

From $\frac{\partial}{\partial \boldsymbol{\pi}} \mathcal{L}(\boldsymbol{\theta}) = \mathbf{0}$, i.e.,

$$\sum_{i=1}^{N} \frac{\mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{l=1}^{K} \pi_l \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_l, \boldsymbol{\Sigma}_l)} + \lambda = 0, \;\; 1 \leq k \leq K,$$

we get

$$\frac{1}{\pi_k} \sum_{i=1}^{N} \frac{\pi_k \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{l=1}^{K} \pi_l \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_l, \boldsymbol{\Sigma}_l)} + \lambda = 0$$

Recall that

$$p(z_{ik} = 1|\mathbf{x}_i) = \frac{\pi_k \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{l=1}^{K} \pi_l \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_l, \boldsymbol{\Sigma}_l)}.$$

So it follows that

$$\pi_k = -\frac{1}{\lambda} \sum_{i=1}^{N} p(z_{ik} = 1|\mathbf{x}_i).$$

From $\sum_{k=1}^{K} \pi_k = 1$, λ satisfies

$$\begin{aligned} \lambda &= -\sum_{k=1}^{K} \sum_{i=1}^{N} p(z_{ik} = 1|\mathbf{x}_i) \\ &= -\sum_{i=1}^{N} \sum_{k=1}^{K} p(z_{ik} = 1|\mathbf{x}_i) = -N. \end{aligned}$$

Therefore, we get

$$\pi_k = \frac{1}{N} \sum_{i=1}^{N} p(z_{ik} = 1|\mathbf{x}_i).$$

In summary,

$$\boldsymbol{\mu}_k = \sum_{i=1}^{N} \frac{p(z_{ik}=1|\mathbf{x}_i)}{\sum_{i=1}^{N} p(z_{ik}=1|\mathbf{x}_i)} \mathbf{x}_i,$$

$$\boldsymbol{\Sigma}_k = \sum_{i=1}^{N} \frac{p(z_{ik}=1|\mathbf{x}_i)}{\sum_{l=1}^{N} p(z_{lk}=1|\mathbf{x}_l)} (\mathbf{x}_i - \boldsymbol{\mu}_k)(\mathbf{x}_i - \boldsymbol{\mu}_k)^\top,$$

$$\pi_k = \frac{1}{N} \sum_{i=1}^{N} p(z_{ik}=1|\mathbf{x}_i).$$

Note that all the above solutions depend on $p(z_{ik}=1|\mathbf{x}_i)$.

From our derivations we have the following EM algorithm.

- Initialization
  (K-means clustering is often used to initialize the EM algorithm)
- (E step) Using the current parameters $\boldsymbol{\theta}$ compute

$$p(z_{ik}=1|\mathbf{x}_i)=\frac{\pi_k\mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k)}{\sum_{l=1}^{K}\pi_l\mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_l,\boldsymbol{\Sigma}_l)}.$$

- (M step) Update all parameters $\boldsymbol{\theta}$

$$\boldsymbol{\mu}_k=\sum_{i=1}^{N}\frac{p(z_{ik}=1|\mathbf{x}_i)}{\sum_{i=1}^{N}p(z_{ik}=1|\mathbf{x}_i)}\mathbf{x}_i,$$

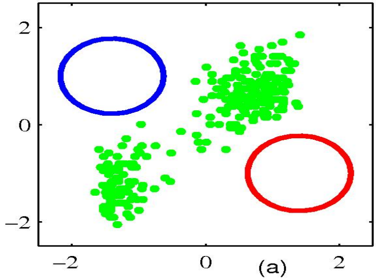
$$\boldsymbol{\Sigma}_k=\sum_{i=1}^{N}\frac{p(z_{ik}=1|\mathbf{x}_i)}{\sum_{l=1}^{N}p(z_{lk}=1|\mathbf{x}_l)}(\mathbf{x}_i-\boldsymbol{\mu}_k)(\mathbf{x}_i-\boldsymbol{\mu}_k)^\top,$$

$$\pi_k=\frac{1}{N}\sum_{i=1}^{N}p(z_{ik}=1|\mathbf{x}_i).$$

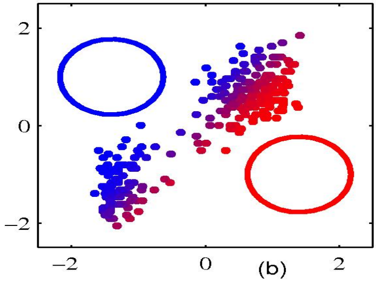
- Repeat E step and M step until convergence.

Example:

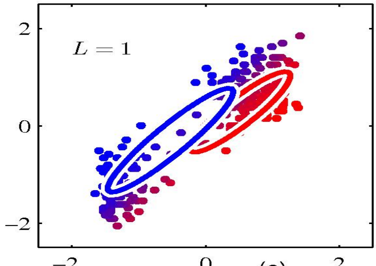
Initialization



Compute $p(z_{ik} = 1|\mathbf{x}_i)$

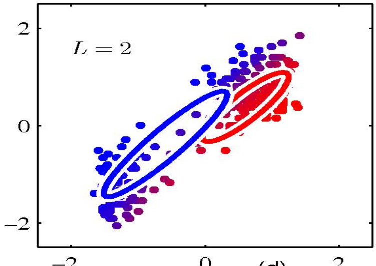


Update $\boldsymbol{\theta}$



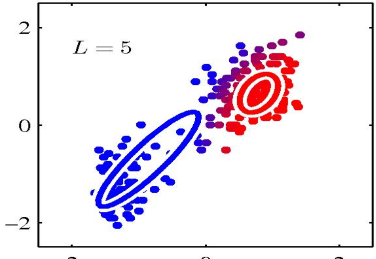
(c)

No. of interations: 2



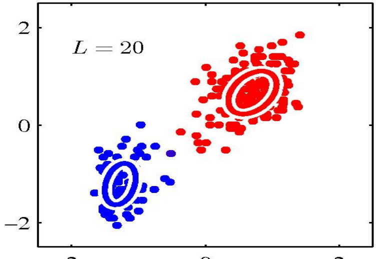
(d)

No. of interations: 5



(e)

No. of interations: 20



(f)

Examples: Clustering

We generate a data set by using the `make_blobs` function. It generates isotropic Gaussian blobs for clustering. Here we make 4 blobs. We will compare k-means clustering and Gaussian mixture model for this data.

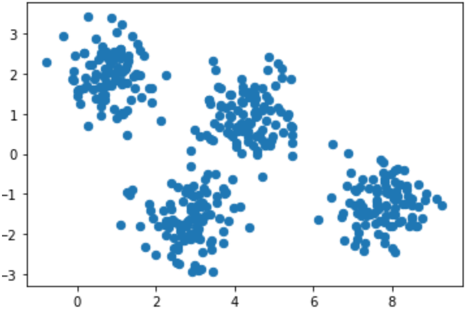


Figure: Data set ($N = 400$)

If we use k-means clustering, the result is given below. However, if we transform the data, the new decision boundaries do not look well.

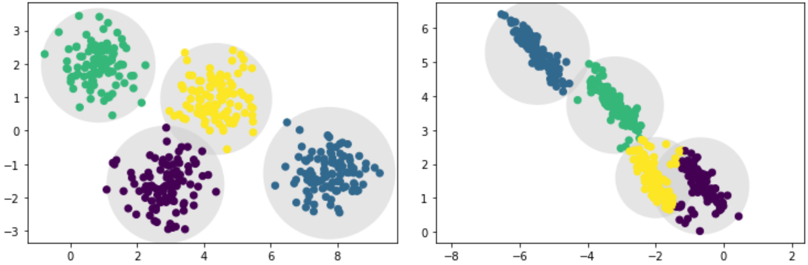


Figure: k-means clustering for two different data sets

When we use the GMM, it works well for both data sets.

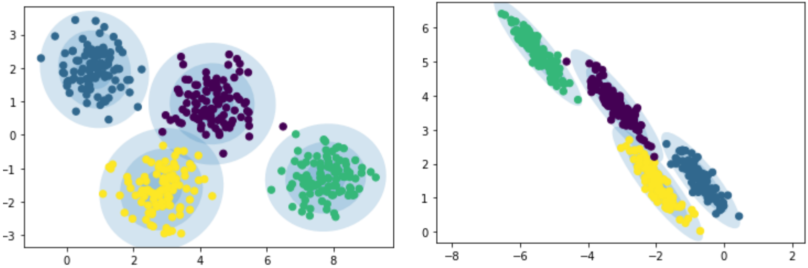


Figure: GMM for two different data sets

Density Estimation

The GMM can be also used for density estimation. We are going to use a synthetic data set generated by the `make_blobs` function.

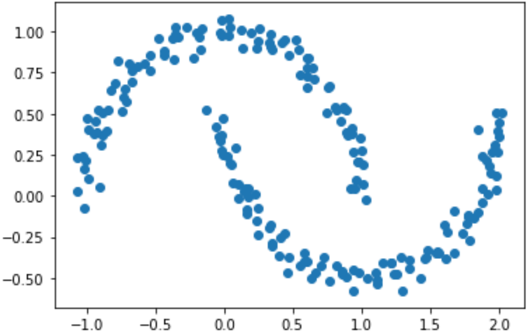
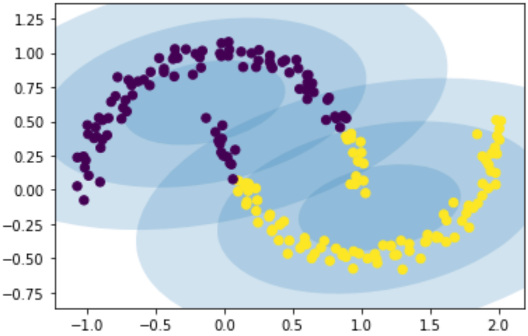


Figure: Data set ($N = 200$)

If we use a 2-components GMM for clustering this data set, we get the following result which doesn't look useful.



Instead, we ignore labels and approximate the input distributions by introducing more components.

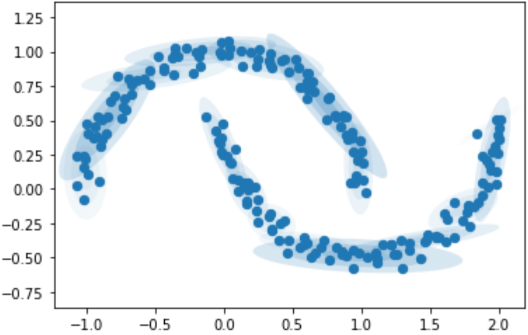


Figure: GMM with 16 components

- However, we now have an important question regarding how to choose the proper number of components.
- Obviously, too many components may occur over-fitting.

To this end, some criteria such as Akaike information criterion(AIC) and Bayesian information criterion(BIC) are proposed.

$$AIC = 2k - 2\log(\hat{L}),$$

$$BIC = \log(n)k - 2\log(\hat{L}),$$

where $\hat{L}$ is the maximum value of the likelihood function, n is the number of data points, and k is the number of parameters.

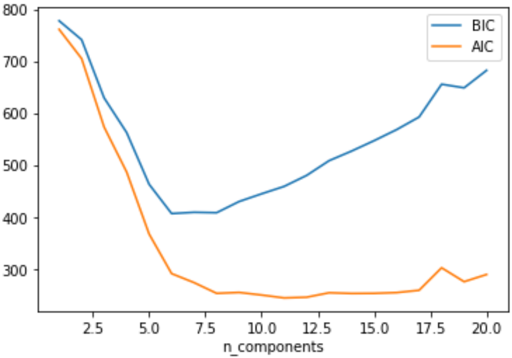


Figure: BIC and AIC

According to the above graph, choosing 8 components seems reasonable.

Given below is our final result with 8 components.

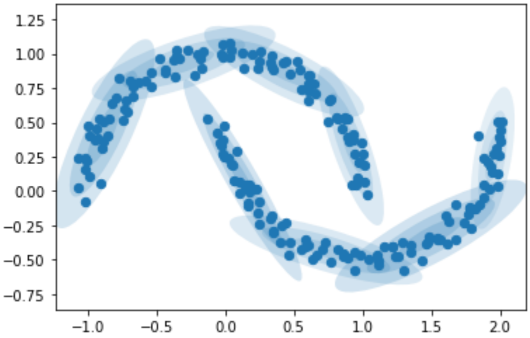


Figure: GMM with 8 components