Suppose we wish to cluster this dataset:
The data points cluster naturally into 3 groups. Each cluster is well modeled by a bivariate Gaussian density. The ellipses are "90% contours" based on a fitted Gaussian mixture model (GMM).

In these notes we'll develop a general method for clustering data when clusters are elliptical, based on maximum likelihood estimation of GMMs.
Gaussian Mixture Models

Recall the multivariate Gaussian density

\[ \phi(x; \mu, \Sigma) = \]

where

\[ x \in \mathbb{R}^d \]
\[ \mu \in \mathbb{R}^d \]
\[ \Sigma \in \mathbb{R}^{d \times d}, \Sigma > 0 \]

A random variable \( X \) follows a Gaussian mixture model if its density function has the form

\[ f(x) = \sum_{k=1}^{K} w_k \phi(x; \mu_k, \Sigma_k) \]

where

\[ 0 < w_k < 1, \sum_{k=1}^{K} w_k = 1 \]
\[ \mu_k \in \mathbb{R}^d \]
\[ \Sigma_k \in \mathbb{R}^{d \times d}, \Sigma_k > 0 \]
$X_1 \sim \mathcal{N}(-2, 1)$  \hspace{0.5cm} $X_2 \sim \mathcal{N}(2, 1)$

\[ \frac{1}{2} f_{X_1} + \frac{1}{2} f_{X_2} \]

\[ f_{X_1 + X_2} \]

mixture  \hspace{0.5cm} \text{not a mixture}
Simulating a GMM

Suppose

$$\theta = (w_1, \ldots, w_K, \mu_1, \ldots, \mu_K, \Sigma_1, \ldots, \Sigma_K)$$

is known. How can we simulate a realization of the GMM?

Basic idea:

1: Choose a "component" at random, weighted according to $w_k$

2: Draw a realization from $N(\mu_k, \Sigma_k)$

Why does this work?
Let $S \in \{1, 2, \ldots, K\}$ be a discrete RV such that

$$\Pr \{ S = k \} = w_k.$$  

Generate $X$ as follows:

1. Generate a realization $s$ of $S$.
2. Generate $X \sim \mathcal{N}(\mu_s, \Sigma_s)$.

Then the density of $X$ generated in this way is

$$f(x) = \sum_{k=1}^{K} f(x | S = k) \cdot \Pr \{ S = k \} = \sum_{k=1}^{K} w_k \phi(x; \mu_k, \Sigma_k)$$

as desired.

The variable $S$ is called a (hidden) state variable. We will imagine that every realization from a GMM is associated with a specific realization of a state variable.
In clustering, our objective is to estimate \( \Theta = (\{w_k\}, \{\mu_k\}, \{\Sigma_k\}) \), and to define clusters in terms of the estimated GMM.

That is, we assume

\[ X_1, \ldots, X_n \overset{iid}{\sim} f(x; \Theta) \]

and reduce clustering to parameter estimation.

**Maximum Likelihood Estimation**

Suppose \( f(x; \Theta) \) is an arbitrary density parametrized by \( \Theta \), and consider

\[ X_1, \ldots, X_n \overset{iid}{\sim} f(x; \Theta). \]

Denote a fixed realization of \( X_1, \ldots, X_n \) by

\[ \mathbf{x} = (x_1, \ldots, x_n) \]

The **likelihood function** of \( \Theta \) is

\[ l(\Theta; \mathbf{x}) := f(\mathbf{x}; \Theta) \]
The maximum likelihood estimator (MLE) is

$$\hat{\theta} = \hat{\theta}(x) := \arg \max_\theta l(\theta; x)$$

**Example** Suppose

$$X_1, \ldots, X_n \overset{iid}{\sim} N(\mu, \Sigma)$$

so that $\theta = (\mu, \Sigma)$. Then

$$l(\theta; x) = \prod_{i=1}^n \frac{1}{\sqrt{(2\pi)^d |\Sigma|}} \exp \left\{ -\frac{1}{2} (x_i - \mu)^T \Sigma^{-1} (x_i - \mu) \right\}$$

Computationally, it is convenient to maximize the log-likelihood

$$\log l(\theta; x) = \sum_{i=1}^n \left[ -\frac{d}{2} \log (2\pi) - \frac{1}{2} \log |\Sigma| ight. \\
\left. - \frac{1}{2} (x_i - \mu)^T \Sigma^{-1} (x_i - \mu) \right]$$

The MLE is

$$\hat{\mu} =$$

$$\hat{\Sigma} =$$
Now consider a GMM. Assume $K$ is known.
The likelihood function is
\[ l(\theta; x) = \prod_{i=1}^{n} f(x_i; \theta) \]

and the log-likelihood is
\[ \log l(\theta; x) = \]

There is unfortunately no known closed-form maximizer.

Suppose, just for kicks, that we also had access to the realization of state variables
\[ z = (s_1, \ldots, s_n) \]

associated with $x$.

**Notation**
\[ I_k = \{ i : s_i = k \} \]
\[ n_k = |I_k| \]
Then the likelihood is

\[ l(\theta; x, \Sigma) = \prod_{i=1}^{n} f(x_i; \theta) \]

\[ = \prod_{i=1}^{n} f(x_i | s_i; \theta) \cdot \Pr(s_i = s_i; \theta) \]

(5)

and the log-likelihood is

\[ \log l(\theta; x, \Sigma) = \]

\[ \Rightarrow \text{can maximize w.r.t. } (\mu_k, \Sigma_k) \text{ independently from other components} \]

\[ \Rightarrow \quad \hat{\mu}_k = \]

\[ \hat{\Sigma}_k = \]
To solve
\[
\max \sum_{k=1}^{K} \eta_k \log w_k
\]
\[\text{s.t. } \sum w_k = 1\]
we can use **Lagrange multipliers**.

The Lagrangian is
\[
L(w_1, \ldots, w_K, \lambda) = \sum_{k=1}^{K} \eta_k \log w_k + \lambda (\sum_{k=1}^{K} w_k - 1)
\]

\[
\frac{\partial L}{\partial w_k} = \frac{\eta_k}{w_k} + \lambda = 0
\]

\[\Rightarrow w_k = -\frac{\eta_k}{\lambda}\]

\[\Rightarrow \sum \left(-\frac{\eta_k}{\lambda}\right) = 1\]

\[\Rightarrow \lambda = -\sum \eta_k = -\hat{\eta}\]

\[\Rightarrow \hat{w}_k = \frac{\eta_k}{\hat{\eta}}\]
The combined data

\[ \Xi = (x, s) \]

is called the complete data. More generally, the complete data is a combination of observed and unobserved data that makes the MLE tractable.

The Expectation-Maximization Algorithm

Define the "indicator" variable

\[ \Delta_{i,k} = \begin{cases} 1 & \text{if } s_i = k \\ 0 & \text{if } s_i \neq k \end{cases} \]

The complete data log-likelihood can be written

\[
\log l(\theta; x, s) = \sum_{i=1}^{n} \log \left( \sum_{k=1}^{K} \Delta_{i,k} \cdot w_k \phi(x_i; \mu_k, \Sigma_k) \right)
= \sum_{i=1}^{n} \sum_{k=1}^{K} \Delta_{i,k} \left[ \log w_k + \log \phi(x_i; \mu_k, \Sigma_k) \right]
\]

The EM algorithm is an iterative algorithm that produces a sequence \( \theta^{(1)}, \theta^{(2)}, \ldots \) of estimates.
E-Step: Given $\theta^{(j)}$, compute the expected complete-data log likelihood

$$A(\theta, \theta^{(i)}) = \mathbb{E} \left[ \log l(\theta; \mathbf{x}, \mathbf{S}) \middle| \mathbf{x}; \theta^{(j)} \right]$$

w.r.t. $\mathbf{x} | \mathbf{x}$

In our case,

$$\mathbb{E} \left[ \log l(\theta; \mathbf{x}, \mathbf{S}) \middle| \mathbf{x}; \theta^{(j)} \right]$$

$$= \sum_{i=1}^{n} \sum_{k=1}^{K} \gamma_{i,k}(\theta^{(j)}) \cdot \left[ \log w_k + \log f(x_i; m_k, \Sigma_k) \right]$$

where

$$\gamma_{i,k}(\theta^{(j)})$$

M-Step

$$\theta^{(j+1)} =$$
**EM Algorithm for GMMs**

Initialize $\theta^{(0)}$

Repeat

**E-Step:** Compute

$$\gamma_{i,k}^{(j)} = \gamma_{i,k}(\theta^{(j)}) = E[\Delta_{i,k} | x; \theta^{(j)}]$$

and form

$$Q(\theta, \theta^{(j)}) := E[\log l(\theta; x, \xi) | x; \theta^{(j)}]$$

**M-Step:**

$$\theta^{(j+1)} = \arg\max_{\theta} Q(\theta, \theta^{(j)})$$

Until termination criterion satisfied
M-Step for GMM

Maximizing

\[ Q(\theta, \theta^{(s)}) = \sum_{i=1}^{n} \sum_{k=1}^{K} \delta_{i,k}^{(s)} \left[ \log w_k - \frac{d}{2} \log 2\pi - \frac{1}{2} \log |\Sigma_k| - \frac{1}{2} (x_i - \mu_k)^T \Sigma_k^{-1} (x_i - \mu_k) \right] \]

w.r.t. \( \theta = (\forall \omega_k, \forall \mu_k, \forall \Sigma_k) \) yields

\[ M_k^{(j+1)} = \frac{\sum_{i=1}^{n} \delta_{i,k}^{(s)} x_i}{\sum_{i=1}^{n} \delta_{i,k}^{(s)}} \]

\[ \Sigma_k^{(j+1)} = \frac{\sum_{i=1}^{n} \delta_{i,k}^{(s)} (x_i - \mu_k^{(j+1)})(x_i - \mu_k^{(j+1)})^T}{\sum_{i=1}^{n} \delta_{i,k}^{(s)}} \]

\[ w_k^{(j+1)} = \frac{1}{n} \sum_{i=1}^{n} \delta_{i,j,k}^{(s)} \]
Exercise 1. Express \( \gamma_{i,k}^{(j)} \) as a function of \( x_i, \theta^{(j)} \).

Hint: Apply Bayes' rule.
\[ \delta_{i,k}^{(j)} = E[\Delta_{i,k} \mid x; \theta^{(j)}] \]
\[ = \Pr \{ \Delta_{i,k} = 1 \mid x; \theta^{(j)} \} \]
\[ = \Pr \{ S_i = k \mid x; \theta^{(j)} \} \]
\[ = \Pr \{ S_i = k; \theta^{(j)} \} \cdot f(x_i \mid S_i = k; \theta^{(j)}) \]
\[ = W_{k}^{(j)} \phi(x_i \mid \mu_{k}^{(j)}, \Sigma_{k}^{(j)}) \]
\[ \sum_{k}^{K} W_{k}^{(j)} \phi(x_i \mid \mu_{k}^{(j)}, \Sigma_{k}^{(j)}) \]

**Termination**

One possible termination criterion is to stop iterating when

\[ |l(\theta^{(j)}; x) - l(\theta^{(j)}; x)| \leq \varepsilon \]

or when

\[ |Q(\theta^{(j+1)}, \theta^{(j)}) - Q(\theta^{(j)}, \theta^{(j)})| \leq \varepsilon \]

for some pre-chosen tolerance \( \varepsilon \).
Initialization

In general, the likelihood has several local maxima. Therefore initialization of the EM algorithm is critical.

In fact, a global maximum is obtained by putting \( \mu_i = x_i \) for some \( i \), \( \Sigma_i = 0 \), \( \omega_i = 1 \), which is not a useful solution. Hence, we are actually seeking a local maximum.

A good initialization for the GMM is

\[
\mu_k^{(0)} = \text{random } x_i \quad \text{(distinct)}
\]

\[
\Sigma_k^{(0)} = \text{sample covariance}
\]

\[
\omega_k^{(0)} = \frac{1}{K}
\]

In practice, it may be beneficial to initialize the algorithm and run it several times, and select the final estimate with largest likelihood.
Defining Clusters

Recall

\[ \gamma_{i,k}(\theta) = \Pr \{ S_i = k \mid x_i ; \theta \} \]

Therefore a reasonable "hard" assignment of points to clusters is given by

\[ x_i \rightarrow \arg \max_{k=1,...,K} \gamma_{i,k}(\hat{\theta}) \]

Alternatively, for "soft" assignments, we may view \( \gamma_{i,k}(\hat{\theta}) \) as the affinity of \( x_i \) for cluster \( k \).

The EM Algorithm in General

The EM is not specific to GMMs, but rather a general class of algorithms for computing ML/MAP estimators.

It is applicable when having knowledge of certain hidden variables renders the MLE tractable.
\[ k \text{ = observed data} \]
\[ z \text{ = unobserved/hidden data} \]

**EM Algorithm**

Initialize \( \theta^{(0)} \)

Repeat

**E-Step:**

Form

\[
Q(\theta, \theta^{(j)}) := \mathbb{E} \left[ \log l(\theta ; x, z) \mid x ; \theta^{(j)} \right]
\]

**M-Step:**

Compute

\[
\theta^{(j+1)} = \arg \max_{\theta} Q(\theta, \theta^{(j)})
\]

Until termination criterion satisfied

---

**Theorem**

For each \( j = 1, 2, ... \)

\[
l(\theta^{(j+1)} ; x) \geq l(\theta^{(j)} ; x)
\]

A proof based on Jensen's inequality may be found in Hastie, Tibshirani & Friedman
Example 3 component GMM shown at start of notes.
Connection to K-means

Consider a Gaussian mixture model (GMM)

\[ f(x) = \sum_{k=1}^{K} w_k \phi(x; \mu_k, \sigma^2 I) \]

where \( \sigma^2 \) is fixed. The EM algorithm for ML estimation of \( \sum_{k=1}^{K} w_k, \sum_{k=1}^{K} \mu_k \) is to iterate

\[ \mu_k = \frac{\sum_{i=1}^{n} \delta_{i,k} x_i}{\sum_{i=1}^{n} \delta_{i,k}} \]

\[ w_k = \frac{1}{n} \sum_{i=1}^{n} \delta_{i,k} \]

\[ \delta_{i,k} = \frac{w_k \phi(x_i; \mu_k, \sigma^2 I)}{\sum_{l=1}^{K} w_l \phi(x_i; \mu_l, \sigma^2 I)} \]

When \( \sigma^2 \to 0 \), \( \delta_{i,k} \to \begin{cases} 1 & \text{if } k = \arg \min_{l} \|x_i - \mu_l\| \\ 0 & \text{otherwise} \end{cases} \)

so the algorithm reduces to K-means.
A. \( \phi(\mathbf{x}; \mu, \Sigma) = (2\pi)^{-d/2} |\Sigma|^{-1/2} \exp\left\{ -\frac{1}{2} (\mathbf{x} - \mu)^T \Sigma^{-1} (\mathbf{x} - \mu) \right\} \)

B. \( N(0, 2) \), assuming \( X_1, X_2 \) are uncorrelated.

C. \( \prod_{i=1}^{n} f(x_i; \theta) \)

D. \( \hat{\mu} = \frac{1}{n} \sum x_i \), \( \hat{\Sigma} = \frac{1}{n} \sum (x_i - \hat{\mu})(x_i - \hat{\mu})^T \)

E. \( l(\theta; x) = \prod_{i=1}^{n} \left( \frac{1}{\sum_{k=1}^{K} \mathbf{w}_k} \phi(x_i; \mu_k, \Sigma_k) \right) \)

\[ \log l(\theta; x) = \sum_{i=1}^{n} \log \left( \frac{1}{\sum_{k=1}^{K} \mathbf{w}_k} \phi(x_i; \mu_k, \Sigma_k) \right) \]

F. \( l(\theta; x, \varepsilon) = \prod_{i=1}^{n} \mathbf{w}_{s_i} \phi(x_i; \mu_{s_i}, \Sigma_{s_i}) \)

\[ = \prod_{k=1}^{K} \mathbf{w}_{k_{le}}^{n_{k_{le}}} \cdot \prod_{k=1}^{K} \prod_{i \in I_{k_{le}}} \phi(x_i; \mu_k, \Sigma_k) \]

\[ \log l(\theta; x, \varepsilon) = \sum_{k=1}^{K} n_{k_{le}} \log \mathbf{w}_{k_{le}} \]

\[ + \sum_{k=1}^{K} \sum_{i \in I_{k_{le}}} \log \phi(x_i; \mu_k, \Sigma_k) \]

\[ \hat{\mu}_{k_{le}} = \frac{1}{n_{k_{le}}} \sum_{i \in I_{k_{le}}} x_i \]

\[ \hat{\Sigma}_{k_{le}} = \frac{1}{n_{k_{le}}} \sum_{i \in I_{k_{le}}} (x_i - \hat{\mu}_{k_{le}})(x_i - \hat{\mu}_{k_{le}})^T \]
\[ \gamma_{i,k}(\theta^{(j)}) := \mathbb{E}[\Delta_{i,k} \mid \mathbf{x} ; \theta^{(j)}] = \Pr \left\{ S_i = k \mid \mathbf{x} ; \theta^{(j)} \right\} \]

\[ \theta^{(j+1)} = \arg \max_{\theta} Q(\theta, \theta^{(j)}) \]