Unsupervised Learning II:

Soft Clustering
with Gaussian Mixture Models
K-means assigns each data point to exactly one cluster.

But:
Soft Clustering with Gaussian mixture models

- A “soft” version of $K$-means clustering

- **Given:** Training set $S = \{x_1, \ldots, x_N\}$, and $K$.

- **Assumption:** Data is generated by sampling probabilistically from a “mixture” (linear combination) of $K$ Gaussians.
Soft Clustering with Gaussian mixture models

- A “soft” version of $K$-means clustering

- **Given:** Training set $S = \{x_1, ..., x_N\}$, and $K$.

- **Assumption:** Data is generated by sampling probabilistically from a “mixture” (linear combination) of $K$ Gaussians.
Gaussian Mixture Models Assumptions
Gaussian Mixture Models Assumptions

- $K$ clusters
Gaussian Mixture Models Assumptions

• $K$ clusters

• Each cluster is modeled by a Gaussian distribution with a certain mean and standard deviation (or covariance).
Gaussian Mixture Models Assumptions

• $K$ clusters

• Each cluster is modeled by a Gaussian distribution with a certain \textit{mean} and \textit{standard deviation} (or \textit{covariance}). [This contrasts with K-means, in which each cluster is modeled only by a \textit{mean}.]
Gaussian Mixture Models Assumptions

- \( K \) clusters

- Each cluster is modeled by a Gaussian distribution with a certain mean and standard deviation (or covariance). [This contrasts with K-means, in which each cluster is modeled only by a mean.]

- Assume that each data instance we have was generated by the following procedure:
Gaussian Mixture Models Assumptions

• $K$ clusters

• Each cluster is modeled by a Gaussian distribution with a certain mean and standard deviation (or covariance). [This contrasts with K-means, in which each cluster is modeled only by a mean.]

• Assume that each data instance we have was generated by the following procedure:

  1. Select cluster $c_i$ with probability $P(c_i) = \pi_i$
Gaussian Mixture Models Assumptions

- \( K \) clusters

- Each cluster is modeled by a Gaussian distribution with a certain mean and standard deviation (or covariance). [This contrasts with K-means, in which each cluster is modeled only by a mean.]

- Assume that each data instance we have was generated by the following procedure:

  1. Select cluster \( c_i \) with probability \( P(c_i) = \pi_i \)

  2. Sample point from \( c_i \)'s Gaussian distribution
Mixture of three Gaussians
(one dimensional data)

\[ p(x) = \pi_1 \mathcal{N}(x \mid \mu_1, \sigma_1) + \pi_2 \mathcal{N}(x \mid \mu_2, \sigma_2) + \pi_3 \mathcal{N}(x \mid \mu_3, \sigma_3) \]

where \( \pi_1 + \pi_2 + \pi_3 = 1 \)
Clustering via Gaussian mixture models

- **Clusters**: Each cluster will correspond to a single Gaussian. Each point $x \in S$ will have some probability distribution over the $K$ clusters.
Clustering via Gaussian mixture models

- **Clusters:** Each cluster will correspond to a single Gaussian. Each point \( x \in S \) will have some probability distribution over the \( K \) clusters.

- **Goal:** Given the data \( S \), find the Gaussians! (And their probabilities \( \pi_i \).)
Clustering via Gaussian mixture models

- **Clusters:** Each cluster will correspond to a single Gaussian. Each point $x \in S$ will have some probability distribution over the $K$ clusters.

- **Goal:** Given the data $S$, find the Gaussians! (And their probabilities $\pi_i$.)
  
  i.e., Find parameters $\{\theta_K\}$ of these $K$ Gaussians such $P(S | \{\theta_K\})$ is maximized.
**Clusters:** Each cluster will correspond to a single Gaussian. Each point \( x \in S \) will have some probability distribution over the \( K \) clusters.

**Goal:** Given the data \( S \), find the Gaussians! (And their probabilities \( \pi_i \)).

i.e., Find parameters \( \{\theta_K\} \) of these \( K \) Gaussians such \( P(S \mid \{\theta_K\}) \) is maximized.

This is called a **Maximum Likelihood** method.

- \( S \) is the data
- \( \{\theta_K\} \) is the “hypothesis” or “model”
- \( P(S \mid \{\theta_K\}) \) is the “likelihood”.
General form of one-dimensional (univariate) Gaussian Mixture Model

\[ p(x) = \sum_{i=1}^{K} \pi_i \mathcal{N}(x \mid \mu_i, \sigma_i) \]

where \( \sum_{i=1}^{K} \pi_i = 1 \)
Learning a GMM

Simplest Case:
Maximum Likelihood for Single Univariate Gaussian
Learning a GMM

Simplest Case:
Maximum Likelihood for Single Univariate Gaussian

• Assume training set $S$ has $N$ values generated by a univariate Gaussian distribution:

$$S = \{x_1, \ldots, x_N\}, \text{ where}$$

$$\mathcal{N}(x | \mu, \sigma) = \frac{1}{\sqrt{2\pi \sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$
Assume training set $S$ has $N$ values generated by a univariate Gaussian distribution:

$$S = \{x_1, \ldots, x_N\}, \text{ where}$$

$$\mathcal{N}(x \mid \mu, \sigma) = \frac{1}{\sqrt{2\pi \sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

Likelihood function: probability of the data given the model:

$$p(S \mid \mu, \sigma) = ?$$
Learning a GMM
Simplest Case:
Maximum Likelihood for Single Univariate Gaussian

- Assume training set S has $N$ values generated by a univariate Gaussian distribution:

$$S = \{x_1, \ldots, x_N\}, \text{ where }$$

$$\mathcal{N}(x \mid \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

- Likelihood function: probability of the data given the model:

$$p(S \mid \mu, \sigma) = \prod_{i=1}^{N} \mathcal{N}(x_i \mid \mu, \sigma)$$
How to estimate parameters $\mu$ and $\sigma$ from $S$?
• How to estimate parameters \( \mu \) and \( \sigma \) from \( S \)?

• Maximize the likelihood function with respect to \( \mu \) and \( \sigma \).
• How to estimate parameters $\mu$ and $\sigma$ from $S$?

• Maximize the likelihood function with respect to $\mu$ and $\sigma$.

• We want the $\mu$ and $\sigma$ that maximize the probability of the data.

\[
\text{Find } \text{argmax } p(S \mid \mu, \sigma) = \prod_{i=1}^{N} \mathcal{N}(x_i \mid \mu, \sigma) \\
= \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x_i-\mu)^2}{2\sigma^2}}
\]
• How to estimate parameters $\mu$ and $\sigma$ from $S$?

• Maximize the likelihood function with respect to $\mu$ and $\sigma$.

• We want the $\mu$ and $\sigma$ that maximize the probability of the data.

$$\text{Find } \arg\max_{\mu, \sigma} p(S \mid \mu, \sigma) = \prod_{i=1}^{N} \mathcal{N}(x_i \mid \mu, \sigma)$$

$$= \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi}\sigma^2} e^{-\frac{(x_i - \mu)^2}{2\sigma^2}}$$

• Use log of likelihood to make this tractable.
Log-Likelihood version:

\[
\ln p(S \mid \mu, \sigma) = \ln \left( \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x_i-\mu)^2}{2\sigma^2}} \right)
\]
Log-Likelihood version:

\[
\ln p(S \mid \mu, \sigma) = \ln \left( \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi \sigma^2}} e^{-\frac{(x_i - \mu)^2}{2\sigma^2}} \right)
\]

Find a simplified expression for this.
\[
\ln p(S \mid \mu, \sigma) = \ln \left( \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x_i - \mu)^2}{2\sigma^2}} \right)
\]

\[
= \sum_{i=1}^{N} \ln \left( \frac{e^{-\frac{(x_i - \mu)^2}{2\sigma^2}}}{\sqrt{2\pi\sigma^2}} \right) = \sum_{i=1}^{N} \left( \ln \left( e^{-\frac{(x_i - \mu)^2}{2\sigma^2}} \right) - \ln \left( \sqrt{2\pi\sigma^2} \right) \right)
\]
\[
\ln p(S \mid \mu, \sigma) = \ln \left( \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x_i - \mu)^2}{2\sigma^2}} \right)
= \sum_{i=1}^{N} \ln \left( e^{\frac{-(x_i - \mu)^2}{2\sigma^2}} \right) = \sum_{i=1}^{N} \left( \ln \left( e^{\frac{-(x_i - \mu)^2}{2\sigma^2}} \right) - \ln \left( \sqrt{2\pi\sigma^2} \right) \right)
= -\sum_{i=1}^{N} \left( \frac{(x_i - \mu)^2}{2\sigma^2} + \frac{1}{2} \ln (2\pi) + \frac{1}{2} \ln (\sigma^2) \right)
\]
\[
\ln p(S \mid \mu, \sigma) = \ln \left( \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x_i-\mu)^2}{2\sigma^2}} \right)
\]

\[
= \sum_{i=1}^{N} \ln \left( e^{-\frac{(x_i-\mu)^2}{2\sigma^2}} \right) = \sum_{i=1}^{N} \left( \ln e^{-\frac{(x_i-\mu)^2}{2\sigma^2}} - \ln \left( \sqrt{2\pi\sigma^2} \right) \right)
\]

\[
= -\sum_{i=1}^{N} \left( \frac{(x_i-\mu)^2}{2\sigma^2} + \frac{1}{2} \ln (2\pi) + \frac{1}{2} \ln (\sigma^2) \right)
\]

\[
= -\frac{1}{2\sigma^2} \sum_{i=1}^{N} (x_i - \mu)^2 - \frac{N}{2} \ln \sigma^2 - \frac{N}{2} \ln(2\pi)
\]
Now, given simplified expression for $\ln p(S \mid \mu, \sigma)$, find the maximum likelihood parameters, $\mu$ and $\sigma$. 
Now, given simplified expression for \( \ln p(S \mid \mu, \sigma) \), find the maximum likelihood parameters, \( \mu \) and \( \sigma \).

First, maximize

\[
\ln p(S \mid \mu, \sigma) = -\frac{1}{2\sigma^2} \sum_{i=1}^{N} (x_i - \mu)^2 - \frac{N}{2} \ln \sigma^2 - \frac{N}{2} \ln(2\pi)
\]

with respect to \( \mu \).
Now, given simplified expression for $\ln p(S \mid \mu, \sigma)$, find the maximum likelihood parameters, $\mu$ and $\sigma$.

First, maximize

$$\ln p(S \mid \mu, \sigma) = -\frac{1}{2\sigma^2} \sum_{i=1}^{N} (x_i - \mu)^2 - \frac{N}{2} \ln \sigma^2 - \frac{N}{2} \ln(2\pi)$$

with respect to $\mu$.

$$\frac{d}{d\mu} \left[ -\frac{1}{2\sigma^2} \sum_{i=1}^{N} (x_i - \mu)^2 - \frac{N}{2} \ln \sigma^2 - \frac{N}{2} \ln(2\pi) \right] = 0$$
Now, given simplified expression for $\ln p(S \mid \mu, \sigma)$, find the maximum likelihood parameters, $\mu$ and $\sigma$.

First, maximize

$$\ln p(S \mid \mu, \sigma) = -\frac{1}{2\sigma^2} \sum_{i=1}^{N} (x_i - \mu)^2 - \frac{N}{2} \ln \sigma^2 - \frac{N}{2} \ln(2\pi)$$

with respect to $\mu$.

$$\frac{d}{d\mu} \left[ -\frac{1}{2\sigma^2} \sum_{i=1}^{N} (x_i - \mu)^2 - \frac{N}{2} \ln \sigma^2 - \frac{N}{2} \ln(2\pi) \right]$$

$$= \frac{d}{d\mu} \left[ -\frac{1}{2\sigma^2} \sum_{i=1}^{N} (x_i - \mu)^2 \right]$$
Now, given simplified expression for $\ln p(S \mid \mu, \sigma)$, find the maximum likelihood parameters, $\mu$ and $\sigma$.

First, maximize

$$\ln p(S \mid \mu, \sigma) = -\frac{1}{2\sigma^2} \sum_{i=1}^{N} (x_i - \mu)^2 - \frac{N}{2} \ln \sigma^2 - \frac{N}{2} \ln(2\pi)$$

with respect to $\mu$.

$$\frac{d}{d\mu} \left[ -\frac{1}{2\sigma^2} \sum_{i=1}^{N} (x_i - \mu)^2 - \frac{N}{2} \ln \sigma^2 - \frac{N}{2} \ln(2\pi) \right]$$

$$= \frac{d}{d\mu} \left[ -\frac{1}{2\sigma^2} \sum_{i=1}^{N} (x_i - \mu)^2 \right]$$

$$= -\frac{1}{2\sigma^2} \left[ \sum_{i=1}^{N} -2(x_i - \mu) \right]$$
Now, given simplified expression for \( \ln p(S \mid \mu, \sigma) \), find the maximum likelihood parameters, \( \mu \) and \( \sigma \).

First, maximize

\[
\ln p(S \mid \mu, \sigma) = -\frac{1}{2\sigma^2} \sum_{i=1}^{N} (x_i - \mu)^2 - \frac{N}{2} \ln \sigma^2 - \frac{N}{2} \ln(2\pi)
\]

with respect to \( \mu \).

\[
\frac{d}{d\mu} \left[ -\frac{1}{2\sigma^2} \sum_{i=1}^{N} (x_i - \mu)^2 - \frac{N}{2} \ln \sigma^2 - \frac{N}{2} \ln(2\pi) \right] = \frac{d}{d\mu} \left[ -\frac{1}{2\sigma^2} \sum_{i=1}^{N} (x_i - \mu)^2 \right]
\]

\[
= -\frac{1}{2\sigma^2} \sum_{i=1}^{N} -2(x_i - \mu)
\]

\[
= \frac{1}{\sigma^2} \left[ \left( \sum_{i=1}^{N} x_i \right) - N\mu \right] = 0
\]
Now, given simplified expression for $\ln p(S \mid \mu, \sigma)$, find the maximum likelihood parameters, $\mu$ and $\sigma$.

First, maximize

$$\ln p(S \mid \mu, \sigma) = -\frac{1}{2\sigma^2} \sum_{i=1}^{N} (x_i - \mu)^2 - \frac{N}{2} \ln \sigma^2 - \frac{N}{2} \ln(2\pi)$$

with respect to $\mu$.

$$\frac{d}{d\mu} \left[ -\frac{1}{2\sigma^2} \sum_{i=1}^{N} (x_i - \mu)^2 - \frac{N}{2} \ln \sigma^2 - \frac{N}{2} \ln(2\pi) \right]$$

$$= \frac{d}{d\mu} \left[ -\frac{1}{2\sigma^2} \sum_{i=1}^{N} (x_i - \mu)^2 \right]$$

$$= -\frac{1}{2\sigma^2} \left[ \sum_{i=1}^{N} -2(x_i - \mu) \right]$$

$$= -\frac{1}{\sigma^2} \left[ \sum_{i=1}^{N} -2(x_i - \mu) \right]$$

$$= -\frac{1}{\sigma^2} \left[ \sum_{i=1}^{N} x_i - N\mu \right] = 0$$

Result:

$$\mu_{ML} = \frac{1}{N} \sum_{i=n}^{N} x_n$$

(ML = “Maximum Likelihood”)
Now, maximize

$$\ln p(S \mid \mu, \sigma) = -\frac{1}{2\sigma^2} \sum_{i=1}^{N} (x_i - \mu)^2 - \frac{N}{2} \ln \sigma^2 - \frac{N}{2} \ln(2\pi)$$

with respect to $\sigma$. It’s actually easier to maximize with respect to $\sigma^2$. 

Now, maximize

\[
\ln p(S \mid \mu, \sigma) = -\frac{1}{2\sigma^2} \sum_{i=1}^{N} (x_i - \mu)^2 - \frac{N}{2} \ln \sigma^2 - \frac{N}{2} \ln(2\pi)
\]

with respect to \( \sigma \). It’s actually easier to maximize with respect to \( \sigma^2 \).

\[
\frac{d}{d\sigma^2} \left[ -\frac{1}{2\sigma^2} \sum_{i=1}^{N} (x_i - \mu)^2 - \frac{N}{2} \ln \sigma^2 - \frac{N}{2} \ln(2\pi) \right] = 0
\]
Now, maximize

\[
\ln p(S \mid \mu, \sigma) = -\frac{1}{2\sigma^2} \sum_{i=1}^{N} (x_i - \mu)^2 - \frac{N}{2} \ln \sigma^2 - \frac{N}{2} \ln(2\pi)
\]

with respect to \( \sigma \). It’s actually easier to maximize with respect to \( \sigma^2 \).

\[
\frac{d}{d\sigma^2} \left[ -\frac{1}{2\sigma^2} \sum_{i=1}^{N} (x_i - \mu)^2 - \frac{N}{2} \ln \sigma^2 - \frac{N}{2} \ln(2\pi) \right] = \frac{d}{d\sigma^2} \left[ -\frac{1}{2}(\sigma^2)^{-1} \sum_{i=1}^{N} (x_i - \mu)^2 - \frac{N}{2} \ln \sigma^2 \right]
\]
Now, maximize

\[ \ln p(S \mid \mu, \sigma) = -\frac{1}{2\sigma^2} \sum_{i=1}^{N} (x_i - \mu)^2 - \frac{N}{2} \ln \sigma^2 - \frac{N}{2} \ln(2\pi) \]

with respect to \( \sigma \). It’s actually easier to maximize with respect to \( \sigma^2 \).

\[
\frac{d}{d\sigma^2} \left[ -\frac{1}{2\sigma^2} \sum_{i=1}^{N} (x_i - \mu)^2 - \frac{N}{2} \ln \sigma^2 - \frac{N}{2} \ln(2\pi) \right]
\]

\[
= \frac{d}{d\sigma^2} \left[ -\frac{1}{2} (\sigma^2)^{-1} \sum_{i=1}^{N} (x_i - \mu)^2 - \frac{N}{2} \ln \sigma^2 \right]
\]

\[
= \frac{1}{2} (\sigma^2)^{-2} \sum_{i=1}^{N} (x_i - \mu)^2 - \frac{N}{2\sigma^2} = \frac{1}{(\sigma^2)^2} \sum_{i=1}^{N} (x_i - \mu)^2 - \frac{N}{\sigma^2} = 0
\]
Now, maximize

$$\ln p(S \mid \mu, \sigma) = -\frac{1}{2\sigma^2} \sum_{i=1}^{N} (x_i - \mu)^2 - \frac{N}{2} \ln \sigma^2 - \frac{N}{2} \ln(2\pi)$$

with respect to $\sigma$. It’s actually easier to maximize with respect to $\sigma^2$.

$$\frac{d}{d\sigma^2} \left[ -\frac{1}{2\sigma^2} \sum_{i=1}^{N} (x_i - \mu)^2 - \frac{N}{2} \ln \sigma^2 - \frac{N}{2} \ln(2\pi) \right]$$

$$= \frac{d}{d\sigma^2} \left[ -\frac{1}{2} \left( \sigma^2 \right)^{-1} \sum_{i=1}^{N} (x_i - \mu)^2 - \frac{N}{2} \ln \sigma^2 \right]$$

$$= \frac{1}{2} \left( \sigma^2 \right)^{-2} \sum_{i=1}^{N} (x_i - \mu)^2 - \frac{N}{2\sigma^2} = \frac{1}{\left( \sigma^2 \right)^2} \sum_{i=1}^{N} (x_i - \mu)^2 - \frac{N}{\sigma^2} = 0$$

$\Rightarrow \sigma^2_{ML} = \frac{1}{N} \sum_{n=1}^{N} (x_n - \mu_{ML})^2$
• The resulting distribution is called a **generative model** because it can generate new data values.

\[
\mathcal{N}(x \mid \mu_{ML}, \sigma_{ML}) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu_{ML})^2}{2\sigma_{ML}^2}}
\]
The resulting distribution is called a **generative model** because it can generate new data values.

\[
\mathcal{N}(x \mid \mu_{ML}, \sigma_{ML}) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu_{ML})^2}{2\sigma_{ML}^2}}
\]

- We say that

\[
\theta_{ML} = \{\mu_{ML}, \sigma_{ML}\}
\]

parameterizes the model.
• The resulting distribution is called a **generative model** because it can generate new data values.

\[
\mathcal{N}(x | \mu_{ML}, \sigma_{ML}) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu_{ML})^2}{2\sigma_{ML}^2}}
\]

• We say that

\[
\theta_{ML} = \{\mu_{ML}, \sigma_{ML}\}
\]

parameterizes the model.

• In general, \(\theta\) is used to denote the (learnable) parameters of a probabilistic model.
Discriminative vs. generative models in machine learning

• **Discriminative model:** Creates a discrimination curve to separate the data into classes

• **Generative model:** Gives the parameters $\theta_{ML}$ of a probability distribution that gives maximum likelihood $p(S | \theta)$ for the data.
Learning a GMM

More general case: Multivariate Gaussian Distribution

Multivariate \((D\text{-dimensional})\) Gaussian:

\[
\mathcal{N}(x \mid \mu, \Sigma) = \frac{1}{(2\pi)^{D/2} |\Sigma|^{1/2}} e^{-\frac{(x-\mu)^T \Sigma^{-1} (x-\mu)}{2}}
\]

where \(\mu\) is a \(D\)-dimensional mean vector, \(\Sigma\) is a \(D \times D\) covariance matrix, and \(|\Sigma|\) is the determinant of \(\Sigma\).
Covariance: 
\[
\text{cov}(x, y) = \frac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{(n - 1)}
\]

Variance: 
\[
\text{cov}(x, x) = \frac{\sum_{i=1}^{n} (x_i - \bar{x})(x_i - \bar{x})}{n}
\]
\[
= \frac{\sum_{i=1}^{n} (x_i - \bar{x})^2}{n}
\]
\[
= \text{var}(x) = \sigma^2(x)
\]

Covariance Matrix \( \Sigma \): 
\[
\Sigma_{i,j} = \text{cov} (x_i, x_j)
\]
• Let $S$ be a set of multivariate data points (vectors):
  \[ S = \{x_1, \ldots, x_m\}. \]
• Let $S$ be a set of multivariate data points (vectors):
  \[ S = \{x_1, \ldots, x_m\}. \]

• General expression for finite Gaussian mixture model:
  \[
p(x) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x | \mu_k, \Sigma_k)
  \]
Let $S$ be a set of multivariate data points (vectors):

$$S = \{x_1, \ldots, x_m\}.$$

**General expression for finite Gaussian mixture model:**

$$p(x) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x | \mu_k, \Sigma_k)$$

That is, $x$ has probability of "membership" in multiple clusters/classes.
Goal: Given $S = \{x_1, ..., x_N\}$, and given $K$, find the Gaussian mixture model (with $K$ multivariate Gaussians) for which $S$ has maximum log-likelihood.
Maximum Likelihood for Multivariate Gaussian Mixture Model

- Goal: Given \( S = \{x_1, \ldots, x_N\} \), and given \( K \), find the Gaussian mixture model (with \( K \) multivariate Gaussians) for which \( S \) has maximum log-likelihood.

- Log likelihood function:

\[
\ln P(S \mid \pi, \mu, \Sigma) = \sum_{n=1}^{N} \ln \left( \sum_{k=1}^{K} \pi_k \mathcal{N}(x_n \mid \mu_k, \Sigma_k) \right)
\]
Maximum Likelihood for Multivariate Gaussian Mixture Model

• Goal: Given $S = \{x_1, ..., x_N\}$, and given $K$, find the Gaussian mixture model (with $K$ multivariate Gaussians) for which $S$ has maximum log-likelihood.

• Log likelihood function:

$$\ln P(S | \pi, \mu, \Sigma) = \sum_{n=1}^{N} \ln \left( \sum_{k=1}^{K} \pi_k \mathcal{N}(x_n | \mu_k, \Sigma_k) \right)$$

• Given $S$, we can maximize this function to find

$$\{\pi, \mu, \Sigma\}_{ML}$$
Maximum Likelihood for Multivariate Gaussian Mixture Model

- Goal: Given \( S = \{x_1, \ldots, x_N\} \), and given \( K \), find the Gaussian mixture model (with \( K \) multivariate Gaussians) for which \( S \) has maximum log-likelihood.

- Log likelihood function:

\[
\ln P(S | \pi, \mu, \Sigma) = \sum_{n=1}^{N} \ln \left( \sum_{k=1}^{K} \pi_k \mathcal{N}(x_n | \mu_k, \Sigma_k) \right)
\]

- Given \( S \), we can maximize this function to find

\[
\{\pi, \mu, \Sigma\}_{ML}
\]

- But no closed form solution (unlike simple case in previous slides)
Maximum Likelihood for Multivariate Gaussian Mixture Model

• Goal: Given $S = \{x_1, ..., x_N\}$, and given $K$, find the Gaussian mixture model (with $K$ multivariate Gaussians) for which $S$ has maximum log-likelihood.

• Log likelihood function:

$$\ln P(S | \pi, \mu, \Sigma) = \sum_{n=1}^{N} \ln \left( \sum_{k=1}^{K} \pi_k \mathcal{N}(x_n | \mu_k, \Sigma_k) \right)$$

• Given $S$, we can maximize this function to find

$$\{\pi, \mu, \Sigma\}_{ML}$$

• But no closed form solution (unlike simple case in previous slides)

• In this multivariate case, we can efficiently maximize this function using the “Expectation / Maximization” (EM) algorithm.
Expectation-Maximization (EM) algorithm

\[ p(x) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x | \mu_k, \Sigma_k) \]
Expectation-Maximization (EM) algorithm

\[ p(x) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x | \mu_k, \Sigma_k) \]

- **Goal:** Learn \( \pi_k, \mu_k, \Sigma_k \) for \( k = 1, \ldots, K \)
Expectation-Maximization (EM) algorithm

\[ p(x) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x | \mu_k, \Sigma_k) \]

• **Goal:** Learn \( \pi_k, \mu_k, \Sigma_k \) for \( k = 1, \ldots, K \)

• **General idea:**
  – Choose random initial values for means, covariances and mixing coefficients.  
  (Analogous to choosing random initial cluster centers in \( K \)-means.)
Expectation-Maximization (EM) algorithm

\[ p(x) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x | \mu_k, \Sigma_k) \]

- **Goal:** Learn \( \pi_k, \mu_k, \Sigma_k \) for \( k = 1, \ldots, K \)

- **General idea:**
  - Choose random initial values for means, covariances and mixing coefficients. *(Analogous to choosing random initial cluster centers in K-means.)*
  
  - Alternate between E (expectation) and M (maximization) step:
Expectation-Maximization (EM) algorithm

\[ p(x) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x \mid \mu_k, \Sigma_k) \]

• **Goal:** Learn \( \pi_k, \mu_k, \Sigma_k \) for \( k = 1, \ldots, K \)

• **General idea:**
  – Choose random initial values for means, covariances and mixing coefficients. *(Analogous to choosing random initial cluster centers in K-means.)*

  – Alternate between E (expectation) and M (maximization) step:
    
    • **E step:** use current values for parameters to evaluate posterior probabilities, or “responsibilities”, for each data point. *(Analogous to determining which cluster a point belongs to, in K-means.)*
Expectation-Maximization (EM) algorithm

\[ p(x) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x \mid \mu_k, \Sigma_k) \]

- **Goal:** Learn \( \pi_k, \mu_k, \Sigma_k \) for \( k = 1, \ldots, K \)
- **General idea:**
  - Choose random initial values for means, covariances and mixing coefficients. *(Analogous to choosing random initial cluster centers in K-means.)*
  - Alternate between E (expectation) and M (maximization) step:
    - **E step:** use current values for parameters to evaluate posterior probabilities, or “responsibilities”, for each data point. *(Analogous to determining which cluster a point belongs to, in K-means.)*
    - **M step:** Use these probabilities to re-estimate means, covariances, and mixing coefficients. *(Analogous to moving the cluster centers to the means of their members, in K-means.)*
Expectation-Maximization (EM) algorithm

\[ p(x) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x | \mu_k, \Sigma_k) \]

- **Goal:** Learn \( \pi_k, \mu_k, \Sigma_k \) for \( k = 1, \ldots, K \)

- **General idea:**
  - Choose random initial values for means, covariances and mixing coefficients. (Analogous to choosing random initial cluster centers in K-means.)
  - Alternate between E (expectation) and M (maximization) step:
    - **E step:** use current values for parameters to evaluate posterior probabilities, or “responsibilities”, for each data point. (Analogous to determining which cluster a point belongs to, in K-means.)
    - **M step:** Use these probabilities to re-estimate means, covariances, and mixing coefficients. (Analogous to moving the cluster centers to the means of their members, in K-means.)

Repeat until the log-likelihood or the parameters \( \theta \) do not change significantly.
More detailed version of EM algorithm
More detailed version of EM algorithm

1. **Initialization:** Let $\mathbf{X}$ be the set of training data. Initialize the means $\mu_k$, covariances $\Sigma_k$, and mixing coefficients $\pi_k$, and evaluate initial value of log likelihood.

$$\ln p(\mathbf{X} \mid \pi, \mu, \Sigma) = \sum_{n=1}^{N} \ln \left( \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}_n \mid \mu_k, \Sigma_k) \right)$$
More detailed version of EM algorithm

1. **Initialization:** Let $X$ be the set of training data. Initialize the means $\mu_k$, covariances $\Sigma_k$, and mixing coefficients $\pi_k$, and evaluate initial value of log likelihood.

$$
\ln p(X | \pi, \mu, \Sigma) = \sum_{n=1}^{N} \ln \left( \sum_{k=1}^{K} \pi_k \mathcal{N}(x_n | \mu_k, \Sigma_k) \right)
$$

2. **E step.** Evaluate the “responsibilities” using the current parameter values

$$
r_{n,k} = \frac{\pi_k \mathcal{N}(x_n | \mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(x_n | \mu_j, \Sigma_j)}
$$

where $r_{n,k}$ denotes the “responsibilities” of the $k$th cluster for the $n$th data point.
3. **M step.** Re-estimate the parameters $\theta$ using the current responsibilities.

\[
\mu_k^{new} = \frac{1}{N} \sum_{n=1}^{N} \left( r_{n,k} x_n \right)
\]

\[
\Sigma_k^{new} = \frac{1}{N} \sum_{n=1}^{N} \left( r_{n,k} (x_n - \mu_k)(x_n - \mu_k)^T \right)
\]

\[
\pi_k^{new} = \frac{\sum_{n=1}^{N} r_{n,k}}{N}
\]
4. **Recompute log-likelihood:** Evaluate the log-likelihood with the new parameters

\[
\ln p(X | \pi, \mu, \Sigma) = \sum_{n=1}^{N} \ln \left( \sum_{k=1}^{K} \pi_k \mathcal{N}(x_n | \mu_k, \Sigma_k) \right)
\]

and check for convergence of either the parameters or the log likelihood. If not converged, return to step 2.
• EM much more computationally expensive than \textit{K-means}

• \textbf{Common practice:} Use \textit{K}-means to set initial parameters, then improve with EM.
  
  – \textbf{Initial means:} Means of clusters found by \textit{k}-means
  
  – \textbf{Initial covariances:} Sample covariances of the clusters found by \textit{K}-means algorithm.

  – \textbf{Initial mixture coefficients:} Fractions of data points assigned to the respective clusters.
• Can prove that EM finds local maxima of log-likelihood function.

• EM is very general technique for finding maximum-likelihood solutions for probabilistic models
Using GMM for Classification

Assume each cluster corresponds to one of the classes.

A new test example $x$ is classified according to

$$
\text{class} = \arg\max_{\text{class}_i} P(y = \text{class}_i)P(x | \theta_i)
$$

where

$$
P(x | \theta_i) = \sum_{i=1}^{K} \pi_i \mathcal{N}(x, \mu_i, \Sigma_i)
$$
Case Study:
Text classification from labeled and unlabeled documents using EM

- Big problem with text classification: need labeled data.

- What we have: lots of unlabeled data.

- Question of this paper: Can unlabeled data be used to increase classification accuracy?

- **In other words:** Any information implicit in unlabeled data? Any way to take advantage of this implicit information?
General idea: A version of EM algorithm

- Train a classifier with small set of available labeled documents.

- Use this classifier to assign probabilistically-weighted class labels to unlabeled documents.

- Then train a new classifier using all the documents, both originally labeled and formerly unlabeled.

- Iterate.
A document \( \mathbf{x} \) is represented by a vector of word frequencies \( x_i \):

\[
\mathbf{x} = (x_1, x_2, \ldots, x_N)
\]

for the \( N \) words in the vocabulary.

Assume there are \( K \) classes, \( c_1, \ldots, c_K \)

(E.g., classes might correspond to different “topics”)
Probabilistic framework

• Assumes data are generated with Gaussian mixture model with \( K \) components

• Assumes one-to-one correspondence between mixture components and classes.

• “These assumptions rarely hold in real-world text data”
Let \( \theta = \{ \mu_1, ..., \mu_K \} \cup \{ \Sigma_1, ..., \Sigma_K \} \cup \{ \pi_1, ..., \pi_K \} \) be the mixture parameters.
Probabilistic framework

Let $\theta = \{\mu_1, ..., \mu_K\} \cup \{\Sigma_1, ..., \Sigma_K\} \cup \{\pi_1, ..., \pi_K\}$ be the mixture parameters.

Assumptions:
Document $x$ is created by:
Probabilistic framework

Let $\theta = \{\mu_1, ..., \mu_K\} \cup \{\Sigma_1, ..., \Sigma_K\} \cup \{\pi_1, ..., \pi_K\}$ be the mixture parameters.

**Assumptions:**

Document $x$ is created by:

1. selecting mixture component $k$ according to the mixture weights $\pi_j$
Probabilistic framework

Let $\theta = \{\mu_1, ..., \mu_K\} \cup \{\Sigma_1, ..., \Sigma_K\} \cup \{\pi_1, ..., \pi_K\}$ be the mixture parameters.

Assumptions:

Document $x$ is created by:

(1) selecting mixture component $k$ according to the mixture weights $\pi_j$

(2) this selected mixture component generates document $x$, with distribution $N(x | \mu_k, \Sigma_k)$
Probabilistic framework

Let \( \theta = \{ \mu_1, ..., \mu_K \} \cup \{ \Sigma_1, ..., \Sigma_K \} \cup \{ \pi_1, ..., \pi_K \} \) be the mixture parameters.

Assumptions:

Document \( x \) is created by:

1. selecting mixture component \( k \) according to the mixture weights \( \pi_j \)

2. this selected mixture component generates document \( x \), with distribution \( N(x | \mu_k, \Sigma_k) \)

Likelihood of document \( x \) under model \( \theta \):

\[
p(x | \theta) = \sum_{k=1}^{K} \pi_k N(x | \mu_k, \Sigma_k)
\]
• Now, we will apply EM to a Naive Bayes Classifier

Let $\mathbf{x} = (x_1, x_2, \ldots, x_N)$
Now, we will apply EM to a Naive Bayes Classifier

Let $\mathbf{x} = (x_1, x_2, \ldots, x_N)$

Recall Naive Bayes classifier: Assume each feature $x_i$ is conditionally independent, given $c_j$. 
• Now, we will apply EM to a Naive Bayes Classifier

Let \( \mathbf{x} = (x_1, x_2, \ldots, x_N) \)

Recall Naive Bayes classifier: Assume each feature \( x_i \) is conditionally independent, given \( c_j \).

We have:

\[
P(x_1, x_2, \ldots, x_N | c_j) \approx P(x_1 | c_j)P(x_2 | c_j) \cdots P(x_N | c_j)
\]
Now, we will apply EM to a Naive Bayes Classifier

Let \( \mathbf{x} = (x_1, x_2, \ldots, x_N) \)

Recall Naive Bayes classifier: Assume each feature \( x_i \) is conditionally independent, given \( c_j \).

We have:

\[
P(x_1, x_2, \ldots, x_N \mid c_j) \approx P(x_1 \mid c_j)P(x_2 \mid c_j) \cdots P(x_N \mid c_j)
\]

\[
p(c_j \mid \mathbf{x}) = p(c_j) \prod_{i} p(x_i \mid c_j), \quad i = 1, \ldots, N; \quad j = 1, \ldots, K
\]
To train naïve Bayes from labeled data, estimate:

\[ p(c_j) \text{ and } p(x_i \mid c_j), \quad j = 1, \ldots, K; \quad i = 1, \ldots, N \]

Note that under our assumption of a Gaussian mixture model, these are estimates of the parameters \( \theta \). Call these values \( \hat{\theta} \).

Thus, Naïve Bayes gives us an initial estimate \( \hat{\theta} \) of our GMM. We're going to use unlabeled data to improve this estimate.
To train naïve Bayes from labeled data, estimate:

\[ p(c_j) \text{ and } p(x_i \mid c_j), \quad j = 1, \ldots, K; \quad i = 1, \ldots, N \]

Note that under our assumption of a Gaussian mixture model,

\[ p(c_j) \approx \pi_j \]
To train naïve Bayes from labeled data, estimate:

\[ p(c_j) \text{ and } p(x_i | c_j), \ j = 1, ..., K; \ i = 1, ..., N \]

Note that under our assumption of a Gaussian mixture model,

\[ p(c_j) \approx \pi_j \]

and

\[ \prod_{i=1}^{N} p(x_i | c_j) \approx N(x | \mu_j, \Sigma_j) \]
To train naïve Bayes from labeled data, estimate:

\[ p(c_j) \text{ and } p(x_i \mid c_j), \; j = 1, \ldots, K; \; i = 1, \ldots, N \]

Note that under our assumption of a Gaussian mixture model,

\[ p(c_j) \approx \pi_j \]

and

\[ \prod_{i=1}^{N} p(x_i \mid c_j) \approx N(x \mid \mu_j, \Sigma_j) \]

These are estimates of the parameters \( \theta \). Call these values \( \hat{\theta} \).

Thus, Naïve Bayes gives us an initial estimate (\( \hat{\theta} \)) of our GMM.

We’re going to use unlabeled data to improve this estimate.
Applying EM to Naive Bayes

- We have a small number of labeled documents $S_{\text{labeled}}$, and a large number of unlabeled documents, $S_{\text{unlabeled}}$. 

- Expectation step: The resulting classifier is used to assign probabilistically-weighted class labels to each unlabeled document $x \in S_{\text{unlabeled}}$. 

- Maximization step: Re-estimate using values for $x \in S_{\text{labeled}} \cup S_{\text{unlabeled}}$. 

- Repeat until $\theta$ or $\theta^*$ has converged.
Applying EM to Naive Bayes

- We have a small number of labeled documents $S_{\text{labeled}}$, and a large number of unlabeled documents, $S_{\text{unlabeled}}$.

- The initial parameters $\hat{\theta}$ are estimated from the labeled documents $S_{\text{labeled}}$.

Applying EM to Naive Bayes

- We have a small number of labeled documents $S_{labeled}$, and a large number of unlabeled documents, $S_{unlabeled}$.

- The initial parameters $\hat{\theta}$ are estimated from the labeled documents $S_{labeled}$.

- **Expectation step:** The resulting classifier is used to assign probabilistically-weighted class labels $p(c_j \mid x)$ to each unlabeled document $x \in S_{unlabeled}$. 
Applying EM to Naive Bayes

- We have a small number of labeled documents \( S_{\text{labeled}} \), and a large number of unlabeled documents, \( S_{\text{unlabeled}} \).

- The initial parameters \( \hat{\theta} \) are estimated from the labeled documents \( S_{\text{labeled}} \).

- **Expectation step:** The resulting classifier is used to assign probabilistically-weighted class labels \( p(c_j \mid x) \) to each unlabeled document \( x \in S_{\text{unlabeled}} \).

- **Maximization step:** Re-estimate \( \hat{\theta} \) using \( p(c_j \mid x) \) values for \( x \in S_{\text{unlabeled}} \cup S_{\text{unlabeled}} \) until \( \hat{\theta} \) or \( \theta \) has converged.
Applying EM to Naive Bayes

- We have a small number of labeled documents $S_{\text{labeled}}$, and a large number of unlabeled documents, $S_{\text{unlabeled}}$.

- The initial parameters $\hat{\theta}$ are estimated from the labeled documents $S_{\text{labeled}}$.

- **Expectation step:** The resulting classifier is used to assign probabilistically-weighted class labels $p(c_j | x)$ to each unlabeled document $x \in S_{\text{unlabeled}}$.

- **Maximization step:** Re-estimate $\hat{\theta}$ using $p(c_j | x)$ values for $x \in S_{\text{unlabeled}} \cup S_{\text{unlabeled}}$.

- Repeat until $p(c_j | x)$ or $\hat{\theta}$ has converged.
Data

- 20 UseNet newsgroups
- Web pages (WebKB)
- Newswire articles (Reuters)
Figure 2. Classification accuracy on the 20 Newsgroups data set, both with and without 10,000 unlabeled documents. With small amounts of training data, using EM yields more accurate classifiers. With large amounts of labeled training data, accurate parameter estimates can be obtained without the use of unlabeled data, and the two methods begin to converge.
Figure 4. Classification accuracy on the WebKB data set, both with and without 2500 unlabeled documents. When there are small numbers of labeled documents, EM improves accuracy. When there are many labeled documents, however, EM degrades performance slightly—indicating a misfit between the data and the assumed generative model.