8. Mixture Models and Expectation-Maximization
Motivation

• Often the introduction of latent (unobserved) random variables into a model can help to express complex (marginal) distributions

• A very common example are mixture models, in particular Gaussian mixture models (GMM)

• Mixture models can be used for clustering (unsupervised learning) and to express more complex probability distributions

• As we will see, the parameters of mixture models can be estimated using maximum-likelihood estimation such as expectation-maximization
K-means Clustering (Rep.)

• Given: data set \( \{x_1, \ldots, x_N\} \), number of clusters \( K \)

• Goal: find cluster centers \( \{\mu_1, \ldots, \mu_K\} \) so that

\[
J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \|x_n - \mu_k\|
\]

is minimal, where \( r_{nk} = 1 \) if \( x_n \) is assigned to \( \mu_k \)

• Idea: compute \( r_{nk} \) and \( \mu_k \) iteratively

• Start with some values for the cluster centers

• Find optimal assignments \( r_{nk} \)

• Update cluster centers using these assignments

• Repeat until assignments or centers don’t change
K-means Clustering

Initialize cluster means: \( \{ \mu_1, \ldots, \mu_K \} \)
K-means Clustering

Find optimal assignments:

\[ r_{nk} = \begin{cases} 
1 & \text{if } k = \arg \min_j \| x_n - \mu_j \| \\
0 & \text{otherwise} 
\end{cases} \]
K-means Clustering

Find new optimal means:

\[
\frac{\partial J}{\partial \mu_k} = 2 \sum_{n=1}^{N} r_{nk} (x_n - \mu_k) = 0
\]

\[\Rightarrow \mu_k = \frac{\sum_{n=1}^{N} r_{nk} x_n}{\sum_{n=1}^{N} r_{nk}}\]
K-means Clustering

Find new optimal assignments:

\[ r_{nk} = \begin{cases} 
1 & \text{if } k = \arg\min_j \|x_n - \mu_j\| \\
0 & \text{otherwise}
\end{cases} \]
K-means Clustering

Iterate these steps until means and assignments do not change any more
2D Example

- Real data set
- Random initialization

- Magenta line is “decision boundary”
• After every step the cost function $J$ is minimized
• Blue steps: update assignments
• Red steps: update means
• Convergence after 4 rounds
K-means for Segmentation

$K = 2$

$K = 3$

$K = 10$

Original image
K-Means: Additional Remarks

• There is an online version of $K$-means (vs. batch until now)
  • After each addition of $x_i$, the nearest center $c_k$ is updated:
    \[
    \mu_k^{\text{new}} = \mu_k^{\text{old}} + \eta_n (x_n - \mu_k^{\text{old}})
    \]

• The $K$-medoid variant (see also lecture on kernels):
  • Replace the Euclidean distance by a general measure $V$.
    \[
    \tilde{J} = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} V(x_n, \mu_k)
    \]
Mixtures of Gaussians

• Assume that the data consists of $K$ clusters
• The data within each cluster is Gaussian
• For any data point $x$ we introduce a $K$-dimensional binary random variable $z$ so that:

$$p(x) = \sum_{k=1}^{K} \underbrace{p(z_k = 1)}_{=: \pi_k} \mathcal{N}(x \mid \mu_k, \Sigma_k)$$

where

$$z_k \in \{0, 1\}, \quad \sum_{k=1}^{K} z_k = 1$$
A Simple Example

- Mixture of three Gaussians with mixing coefficients
- Left: all three Gaussians as contour plot
- Right: samples from the mixture model, the red component has the most samples
Parameter Estimation

• From a given set of training data \(\{x_1, \ldots, x_N\}\) we want to find parameters \((\pi_1, \ldots, K, \mu_1, \ldots, K, \Sigma_1, \ldots, K)\) so that the likelihood is maximized (MLE):

\[
p(x_1, \ldots, x_N \mid \pi_1, \ldots, K, \mu_1, \ldots, K, \Sigma_1, \ldots, K) = \prod_{n=1}^{N} \sum_{k=1}^{K} \pi_k \mathcal{N}(x_n \mid \mu_k, \Sigma_k)
\]

or, applying the logarithm:

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

• However: this is not as easy as maximum-likelihood for single Gaussians!
Problems with MLE for Gaussian Mixtures

• Assume that for one $k$ the mean $\mu_k$ is exactly at a data point $x_n$
  • For simplicity: assume that $\Sigma_k = \sigma_k^2 I$
  • Then: $\mathcal{N}(x_n | x_n, \sigma_k^2 I) = \frac{1}{\sqrt{2\pi\sigma_k^D}}$

• This means that the overall log-likelihood can be maximized arbitrarily by letting $\sigma_k \to 0$ (overfitting)

• Another problem is the identifiability:
  • The order of the Gaussians is not fixed, therefore:
  • There are $K!$ equivalent solutions to the MLE problem
Overfitting with MLE for Gaussian Mixtures

- One Gaussian fits exactly to one data point
- It has a very small variance, i.e. contributes strongly to the overall likelihood
- In standard MLE, there is no way to avoid this!
Expectation-Maximization

- EM is an elegant and powerful method for MLE problems with latent variables
- Main idea: model parameters and latent variables are estimated iteratively, where average over the latent variables (expectation)
- A typical example application of EM is the Gaussian Mixture model (GMM)
- However, EM has many other applications
- First, we consider EM for GMMs
Expectation-Maximization for GMM

• First, we define the **responsibilities**:

\[ \gamma(z_{nk}) = p(z_{nk} = 1 \mid x_n) \]
Expectation-Maximization for GMM

- First, we define the responsibilities:

\[ \gamma(z_{nk}) = p(z_{nk} = 1 \mid x_n) \]

\[ = \frac{\pi_k \mathcal{N}(x_n \mid \mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(x_n \mid \mu_j, \Sigma_j)} \]

- Next, we derive the log-likelihood wrt. to \( \mu_k \):

\[ \frac{\partial \log p(X \mid \pi, \mu, \Sigma)}{\partial \mu_k} = 0 \]
Expectation-Maximization for GMM

• First, we define the responsibilities:

\[
\gamma(z_{nk}) = p(z_{nk} = 1 \mid x_n)
\]

\[
= \frac{\pi_k \mathcal{N}(x_n \mid \mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(x_n \mid \mu_j, \Sigma_j)}
\]

• Next, we derive the log-likelihood wrt. to \(\mu_k\):

\[
\frac{\partial \log p(X \mid \pi, \mu, \Sigma)}{\partial \mu_k} = 0
\]

and we obtain:

\[
\mu_k = \frac{\sum_{n=1}^{N} \gamma(z_{nk}) x_n}{\sum_{n=1}^{N} \gamma(z_{nk})}
\]
**Expectation-Maximization for GMM**

- We can do the same for the covariances:

\[
\frac{\partial \log p(X \mid \pi, \mu, \Sigma)}{\partial \Sigma_k} = 0
\]

and we obtain:

\[
\Sigma_k = \frac{\sum_{n=1}^{N} \gamma(z_{nk})(x_n - \mu_k)(x_n - \mu_k)^T}{\sum_{n=1}^{N} \gamma(z_{nk})}
\]

- Finally, we derive wrt. the mixing coefficients \( \pi_k \):

\[
\frac{\partial \log p(X \mid \pi, \mu, \Sigma)}{\partial \pi_k} = 0 \quad \text{where:} \quad \sum_{k=1}^{K} \pi_k = 1
\]
Expectation-Maximization for GMM

- We can do the same for the covariances:

\[
\frac{\partial \log p(X \mid \pi, \mu, \Sigma)}{\partial \Sigma_k} = 0
\]

and we obtain:

\[
\Sigma_k = \frac{\sum_{n=1}^{N} \gamma(z_{nk})(x_n - \mu_k)(x_n - \mu_k)^T}{\sum_{n=1}^{N} \gamma(z_{nk})}
\]

- Finally, we derive wrt. the mixing coefficients \( \pi_k \):

\[
\frac{\partial \log p(X \mid \pi, \mu, \Sigma)}{\partial \pi_k} = 0 \quad \text{where:} \quad \sum_{k=1}^{K} \pi_k = 1
\]

and the result is:

\[
\pi_k = \frac{1}{N} \sum_{n=1}^{N} \gamma(z_{nk})
\]
Algorithm Summary

1. Initialize means $\mu_k$, covariance matrices $\Sigma_k$ and mixing coefficients $\pi_k$
2. Compute the initial log-likelihood $\log p(X \mid \pi, \mu, \Sigma)$
3. E-Step. Compute the responsibilities:
   \[ \gamma(z_{nk}) = \frac{\pi_k \mathcal{N}(x_n \mid \mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(x_n \mid \mu_j, \Sigma_j)} \]
4. M-Step. Update the parameters:
   \[ \mu_k^{\text{new}} = \frac{\sum_{n=1}^{N} \gamma(z_{nk}) x_n}{\sum_{n=1}^{N} \gamma(z_{nk})} \quad \Sigma_k^{\text{new}} = \frac{\sum_{n=1}^{N} \gamma(z_{nk}) (x_n - \mu_k^{\text{new}})(x_n - \mu_k^{\text{new}})^T}{\sum_{n=1}^{N} \gamma(z_{nk})} \quad \pi_k^{\text{new}} = \frac{1}{N} \sum_{n=1}^{N} \gamma(z_{nk}) \]
5. Compute log-likelihood; if not converged go to 3.
The Same Example Again

$L = 1$

$L = 2$

$L = 5$

$L = 20$
Observations

- Compared to K-means, points can now belong to both clusters (soft assignment)
- In addition to the cluster center, a covariance is estimated by EM
- Initialization is the same as used for K-means
- Number of iterations needed for EM is much higher
- Also: each cycle requires much more computation
- Therefore: start with K-means and run EM on the result of K-means (covariances can be initialized to the sample covariances of K-means)
- EM only finds a local maximum of the likelihood!
A More General View of EM

• Assume for a moment that we observe $X$ and the binary latent variables $Z$. The likelihood is then:

$$p(X, Z \mid \pi, \mu, \Sigma) = \prod_{n=1}^{N} p(z_n \mid \pi) p(x_n \mid z_n, \mu, \Sigma)$$

where

$$p(z_n \mid \pi) = \prod_{k=1}^{K} \pi_k^{z_{nk}}$$

and

$$p(x_n \mid z_n, \mu, \Sigma) = \prod_{k=1}^{K} \mathcal{N}(x_n \mid \mu_k, \Sigma_k)^{z_{nk}}$$

which leads to the log-formulation:

$$\log p(X, Z \mid \pi, \mu, \Sigma) = \sum_{n=1}^{N} \sum_{k=1}^{K} z_{nk} (\log \pi_k + \log \mathcal{N}(x_n \mid \mu_k, \Sigma_k))$$

Remember:

$$z_{nk} \in \{0, 1\}, \quad \sum_{k=1}^{K} z_{nk} = 1$$
The Complete-Data Log-Likelihood

\[
\log p(X, Z \mid \pi, \mu, \Sigma) = \sum_{n=1}^{N} \sum_{k=1}^{K} z_{nk} \left( \log \pi_k + \log \mathcal{N}(x_n \mid \mu_k, \Sigma_k) \right)
\]

- This is called the **complete-data log-likelihood**
- Advantage: solving for the parameters \((\pi_k, \mu_k, \Sigma_k)\) is much simpler, as the log is inside the sum!
- We could switch the sums and then for every mixture component \(k\) only look at the points that are associated with that component.
- This leads to simple closed-form solutions for the parameters
- However: the latent variables \(Z\) are not observed!
The Main Idea of EM

• Instead of maximizing the joint log-likelihood, we maximize its **expectation** under the latent variable distribution:

\[
\mathbb{E}_Z[\log p(X, Z | \pi, \mu, \Sigma)] = \sum_{n=1}^{N} \sum_{k=1}^{K} \mathbb{E}_Z[z_{nk}](\log \pi_k + \log \mathcal{N}(x_n | \mu_k, \Sigma_k))
\]

where the latent variable distribution per point is:

\[
p(z_n | x_n, \theta) = \frac{p(x_n | z_n, \theta)p(z_n | \theta)}{p(x_n | \theta)} = \frac{\prod_{l=1}^{K} (\pi_l \mathcal{N}(x_n | \mu_l, \Sigma_l))^{z_{nl}}}{\sum_{j=1}^{K} \pi_j \mathcal{N}(x_n | \mu_j, \Sigma_j)}
\]

\[
\theta = (\pi, \mu, \Sigma)
\]
The Main Idea of EM

The expected value of the latent variables is:

$$\mathbb{E}[z_{nk}] = \gamma(z_{nk})$$

plugging in we obtain:

$$\mathbb{E}_Z[\log p(X, Z \mid \pi, \mu, \Sigma)] = \sum_{n=1}^{N} \sum_{k=1}^{K} \gamma(z_{nk})(\log \pi_k + \log \mathcal{N}(x_n \mid \mu_k, \Sigma_k))$$

We compute this iteratively:
1. Initialize $i = 0$, $(\pi_i^k, \mu_i^k, \Sigma_i^k)$
2. Compute $\mathbb{E}[z_{nk}] = \gamma(z_{nk})$
3. Find parameters $(\pi_i^{k+1}, \mu_i^{k+1}, \Sigma_i^{k+1})$ that maximize this
4. Increase $i$; if not converged, goto 2.
The Theory Behind EM

• We have seen that EM maximizes the expected complete-data log-likelihood, but:

• Actually, we need to maximize the log-marginal

\[
\log p(X \mid \theta) = \log \sum_Z p(X, Z \mid \theta)
\]

• It turns out that the log-marginal is maximized implicitly!
The Theory Behind EM

- We have seen that EM maximizes the **expected complete-data log-likelihood**, but:
- Actually, we need to maximize the log-marginal
  \[
  \log p(X | \theta) = \log \sum_Z p(X, Z | \theta)
  \]
- It turns out that the log-marginal is maximized implicitly!
  \[
  \log p(X | \theta) = \mathcal{L}(q, \theta) + \text{KL}(q \| p)
  \]

\[
\mathcal{L}(q, \theta) = \sum_Z q(Z) \log \frac{p(X, Z | \theta)}{q(Z)} \quad \text{KL}(q \| p) = -\sum_Z q(Z) \log \frac{p(Z | X, \theta)}{q(Z)}
\]
• The KL-divergence is positive or 0
• Thus, the log-likelihood is at least as large as $L$ or:
• $L$ is a lower bound of the log-likelihood
What Happens in the E-Step?

• The log-likelihood is independent of \( q \)
• Thus: \( \mathcal{L} \) is maximized iff KL is minimal
• This is the case iff \( q(Z) = p(Z \mid X, \theta) \)
What Happens in the M-Step?

- In the M-step we keep $q$ fixed and find new $\theta$

$$
\mathcal{L}(q, \theta) = \sum_Z p(Z | X, \theta^{\text{old}}) \log p(X, Z | \theta) - \sum_Z q(Z) \log q(Z)
$$

- We maximize the first term, the second is indep.
- This implicitly makes KL non-zero
- The log-likelihood is maximized even more!
Visualization in Parameter-Space

- In the E-step we compute the concave lower bound for given old parameters \( \theta^{\text{old}} \) (blue curve).
- In the M-step, we maximize this lower bound and obtain new parameters \( \theta^{\text{new}} \).
- This is repeated (green curve) until convergence.
Variants of EM

• Instead of maximizing the log-likelihood, we can use EM to maximize a posterior when a prior is given (MAP instead of MLE) ⇒ less overfitting

• In Generalized EM, the M-step only increases the lower bound instead of maximization (useful if standard M-step is intractable)

• Similarly, the E-step can be generalized in that the optimization wrt. $q$ is not complete

• Furthermore, there are incremental versions of EM, where data points are given sequentially and the parameters are updated after each data point.
Example 1: Learn a Sensor Model

- A Radar range finder on a metallic target will return 3 types of measurement:
  - The distance to target
  - The distance to the wall behind the target
  - A completely random value
Example 1: Learn a Sensor Model

- Which point corresponds to from which model?
- What are the different model parameters?
- Solution: Expectation-Maximization
Example 2: Environment Classification

- From each image, the robot extracts features: $\Rightarrow$ points in nD space
- K-means only finds the cluster centers, not their extent and shape
- The centers and covariances can be obtained with EM
Example 3: Plane Fitting in 3D

- Has been done in this paper
- Given a set of 3D points, fit planes into the data
- Idea: Model parameters $\theta$ are normal vectors and distance to origin for a set of planes
- Gaussian noise model: $p(z \mid \theta) = \mathcal{N}(d(z, \theta) \mid 0, \sigma)$

- Introduce latent correspondence variables $C_{ij}$ and maximize the expected log-lik.:
  $$\mathbb{E}[\log p(Z, C \mid \theta)]$$
- Maximization can be done in closed form
Example 3: Plane Fitting in 3D
Summary

• K-means is an iterative method for clustering
• Mixture models can be formalized using latent (unobserved) variables
• A very common example are Gaussian mixture models (GMMs)
• To estimate the parameters of a GMM we can use expectation-maximization (EM)
• In general EM can be interpreted as maximizing a lower bound to the complete-data loglikelihood
• EM is guaranteed to converge, but it may run into local maxima