Gaussian Mixture Models and EM Algorithm

CS 145: Introduction to Data Mining (Spring 2024)

Yanqiao Zhu

UCLA

May 3, 2024

Announcements

- HW2 scores have been released
- HW3 solution has been released
- HW4 due next Wednesday (May 8)
- Project dummy submission due this Sunday
- Project proposal due on May 13

Midterm Exam

Date
- May 20, 12:00PM—1:45PM
- In-person only; no online or make-up exams offered

Format
- Close-book but two letter-sized cheat sheets allowed
- Simple calculators allowed
- Internet access strictly prohibited

Scope
- Supervised Learning: Linear Regression, Logistic Regression, MLP, Gradient Descent, Backpropagation, Regularization, Batch/Layer Norm, Decision Trees, Random Forests, Mixture-of-Expert, Ensemble (Bagging, Boosting, Adaboost), K-Nearst Neighbor
- Unsupervised Learning: K-Means, Gaussian Mixture Models, EM Algorithm, (Variational) Auto Encoders
- Graphs and Networks: Random Walks, Spectral Clustering, Graph Representation Learning

Midterm Exam

Structure

- Part A. True/False Questions ($5 * 2 = 10$ points)
- Part B. Multiple-Choice Questions ($5 * 2 = 10$ points)
- Part C. Fill-in-the-Blank Questions ($5 * 1 = 5$ points)
- Part D. Open-Answer Questions (75 points)
  - Problem 16. Linear Regression with Regularization (15 points)
  - Problem 17. Multilayer Perceptron and Backpropagation (15 points)
  - Problem 18. Decision Trees and Bagging (15 points)
  - Problem 19. K-Means and Gaussian Mixture Models (15 points)
  - Problem 20. Random Walks and Spectral Clustering (15 points)

Overview

1. Generative Models
2. Latent Variable Models
3. Learning with Observed Data
4. Evidence Lower Bound (ELBO)
5. Expectation-Maximization (EM) Algorithm

Generative Models

- Learn the probability distribution $p(\boldsymbol{x})$
- Assign small values to inputs that are "unreasonable"
- Useful for:
  - Generation: Sampling $\boldsymbol{x}_{\text{new}} \sim p(\boldsymbol{x})$ should produce realistic samples
  - Density estimation: $p(\boldsymbol{x})$ should be high if $\boldsymbol{x}$ is similar to training data
  - Unsupervised representation learning: Learn meaningful features from unlabeled data

Latent Variable Models

- Variables that are always unobserved are called latent or hidden variables
- Latent variables $\boldsymbol{z}$ correspond to high-level features
- If $\boldsymbol{z}$ is chosen properly, $p(\boldsymbol{x} \mid \boldsymbol{z})$ could be much simpler than $p(\boldsymbol{x})$

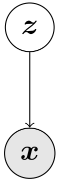

Figure: Bayesian network representation of a latent variable model

Mixture of Gaussians

- A mixture model where the identity of the component that generated a datapoint is a latent variable
- Generative process:
  1. Pick a mixture component k by sampling $\boldsymbol{z} \sim \mathrm{Categorical}(\pi_1, \ldots, \pi_K)$
  2. Generate a data point by sampling from that Gaussian $p(\boldsymbol{x} \mid z_k = 1) = \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$
- The posterior $p(\boldsymbol{z} \mid \boldsymbol{x})$ identifies the mixture component (clustering)

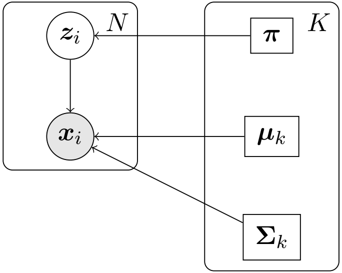

Figure: A graphical plate diagram of GMM

Learning with Observed Data

- Let $\boldsymbol{X}$ denote observed random variables, and $\boldsymbol{Z}$ the unobserved ones
- We have a model for the joint distribution $p(\boldsymbol{X}, \boldsymbol{Z}; \boldsymbol{\theta})$
- To compute the probability of observing a datapoint $\boldsymbol{x}$:

$$p(\boldsymbol{x}; \boldsymbol{\theta}) = \sum_{\boldsymbol{z}} p(\boldsymbol{z}, \boldsymbol{x}; \boldsymbol{\theta})$$

- The log-likelihood of the observed data involves a summation over the latent variables:

$$\log p(\boldsymbol{x}; \boldsymbol{\theta}) = \log \left(\sum_{\boldsymbol{z}} p(\boldsymbol{z}, \boldsymbol{x}; \boldsymbol{\theta}) \right)$$

- This summation can be intractable, making it difficult to optimize the log-likelihood directly

Evidence Lower Bound (ELBO)

- We introduce auxiliary distributions $q(\boldsymbol{z})$ to construct a lower bound on the log-likelihood:

$$\log p(\boldsymbol{x};\boldsymbol{\theta}) = \log\left(\sum_{\boldsymbol{z}} {\color{red}q(\boldsymbol{z})}\frac{p(\boldsymbol{z},\boldsymbol{x};\boldsymbol{\theta})}{{\color{red}q(\boldsymbol{z})}}\right)$$

- They provide a way to approximate the intractable summation and enable optimization
- Apply Jensen's inequality:

$$\begin{aligned}\log p(\boldsymbol{x};\boldsymbol{\theta}) &= \log\left(\sum_{\boldsymbol{z}} q(\boldsymbol{z})\frac{p(\boldsymbol{z},\boldsymbol{x};\boldsymbol{\theta})}{q(\boldsymbol{z})}\right)\\ &= \log \mathbb{E}_{\boldsymbol{z}\sim q(\boldsymbol{z})}\left[\frac{p(\boldsymbol{z},\boldsymbol{x};\boldsymbol{\theta})}{q(\boldsymbol{z})}\right]\\ &\geq \mathbb{E}_{\boldsymbol{z}\sim q(\boldsymbol{z})}\left[\log\left(\frac{p(\boldsymbol{z},\boldsymbol{x};\boldsymbol{\theta})}{q(\boldsymbol{z})}\right)\right]\end{aligned}$$

- The last line is the Evidence Lower Bound (ELBO), denoted as $\mathcal{L}(q,\boldsymbol{\theta})$

Jensen's Inequality

- Jensen's inequality states that for a concave function f and a random variable X:

$$f(\mathbb{E}[X]) \geq \mathbb{E}[f(X)]$$

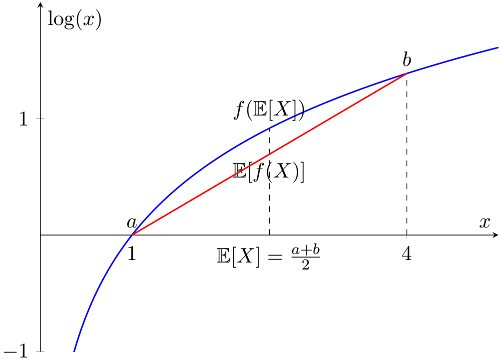

Figure: Illustration of Jensen's inequality for a convex function

Evidence Lower Bound (ELBO)

- Suppose $q(\boldsymbol{z})$ is any probability distribution over the hidden variables
- ELBO holds for any q

$$
\begin{aligned}
\log p(\boldsymbol{x};\boldsymbol{\theta}) = \log\left(\mathbb{E}_{\boldsymbol{z}\sim q(\boldsymbol{z})}\left[\frac{p(\boldsymbol{x},\boldsymbol{z};\boldsymbol{\theta})}{q(\boldsymbol{z})}\right]\right) &\geq \mathbb{E}_{\boldsymbol{z}\sim q(\boldsymbol{z})}\left[\log\left(\frac{p(\boldsymbol{x},\boldsymbol{z};\boldsymbol{\theta})}{q(\boldsymbol{z})}\right)\right] \\
&= \sum_{\boldsymbol{z}} q(\boldsymbol{z}) \log\left(\frac{p(\boldsymbol{x},\boldsymbol{z};\boldsymbol{\theta})}{q(\boldsymbol{z})}\right) \\
&= \sum_{\boldsymbol{z}} q(\boldsymbol{z}) \log p(\boldsymbol{x},\boldsymbol{z};\boldsymbol{\theta}) \underbrace{- \sum_{\boldsymbol{z}} q(\boldsymbol{z}) \log q(\boldsymbol{z})}_{\text{Entropy } H(q) \text{ of } q} \\
&= \sum_{\boldsymbol{z}} q(\boldsymbol{z}) \log p(\boldsymbol{x},\boldsymbol{z};\boldsymbol{\theta}) + H(q)
\end{aligned}
$$

Choosing the Auxiliary Distribution

- We want to choose $q(\boldsymbol{z})$ to make the ELBO as tight as possible
- The ELBO becomes tight when the auxiliary distribution $q(\boldsymbol{z})$ is equal to the posterior distribution of the latent variables given the observed data $p(\boldsymbol{z} \mid \boldsymbol{x}; \boldsymbol{\theta})$
- To show this, consider the case where:

$$q(\boldsymbol{z}) = p(\boldsymbol{z} \mid \boldsymbol{x}; \boldsymbol{\theta}) = \frac{p(\boldsymbol{z}, \boldsymbol{x}; \boldsymbol{\theta})}{p(\boldsymbol{x}; \boldsymbol{\theta})}$$

- Substituting this into the ELBO:

$$\begin{aligned}
\mathbb{E}_{\boldsymbol{z} \sim q(\boldsymbol{z})}\left[\log\left(\frac{p(\boldsymbol{x}, \boldsymbol{z}; \boldsymbol{\theta})}{q(\boldsymbol{z})}\right)\right] &= \mathbb{E}_{\boldsymbol{z} \sim p(\boldsymbol{z}|\boldsymbol{x};\boldsymbol{\theta})}\left[\log\left(\frac{p(\boldsymbol{x}, \boldsymbol{z}; \boldsymbol{\theta})}{p(\boldsymbol{z} \mid \boldsymbol{x}; \boldsymbol{\theta})}\right)\right] \\
&= \mathbb{E}_{\boldsymbol{z} \sim p(\boldsymbol{z}|\boldsymbol{x};\boldsymbol{\theta})}\left[\log\left(\frac{p(\boldsymbol{x}, \boldsymbol{z}; \boldsymbol{\theta})}{\frac{p(\boldsymbol{z},\boldsymbol{x};\boldsymbol{\theta})}{p(\boldsymbol{x};\boldsymbol{\theta})}}\right)\right] \\
&= \mathbb{E}_{\boldsymbol{z} \sim p(\boldsymbol{z}|\boldsymbol{x};\boldsymbol{\theta})}[\log p(\boldsymbol{x}; \boldsymbol{\theta})] \\
&= \log p(\boldsymbol{x}; \boldsymbol{\theta})
\end{aligned}$$

EM Algorithm

- Alternates between making the ELBO tight and optimizing it
- Let the current parameters be $\boldsymbol{\theta}_{\text{old}} = \{\boldsymbol{\mu}_k^{\text{old}}, \pi_k^{\text{old}}, \boldsymbol{\Sigma}_k^{\text{old}}\}_{k=1}^K$
- E-step: Set $q(\boldsymbol{z}) = p(\boldsymbol{z} \mid \boldsymbol{x}; \boldsymbol{\theta}_{\text{old}})$ for the current parameters $\boldsymbol{\theta}_{\text{old}}$
- M-step: Optimize the ELBO with respect to $\boldsymbol{\theta}$:

$$\boldsymbol{\theta}_{\text{new}} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \, \mathcal{L}(q, \boldsymbol{\theta})$$

- The M-step corresponds to maximizing the expected complete data log-likelihood

EM for Gaussian Mixture Models

- E-step: Set $q_n(z_k^{(n)} = 1)$ to the posterior probabilities (responsibilities) of each data point belonging to each mixture component $p(z_k^{(n)} = 1 \mid \boldsymbol{x}^{(n)}; \boldsymbol{\theta}_{\text{old}})$

$$\begin{aligned} q_n(z_k^{(n)} = 1) &= p(z_k^{(n)} = 1 \mid \boldsymbol{x}^{(n)}; \boldsymbol{\theta}_{\text{old}}) \\ &= \frac{p(z_k^{(n)} = 1, \boldsymbol{x}^{(n)}; \boldsymbol{\theta}_{\text{old}})}{p(\boldsymbol{x}^{(n)}; \boldsymbol{\theta}_{old})} \\ &= \frac{\pi_k^{\text{old}} \mathcal{N}(\boldsymbol{x}^{(n)}; \boldsymbol{\mu}_k^{\text{old}}, \boldsymbol{\Sigma}_k^{\text{old}})}{\sum_{j=1}^{K} \pi_j^{\text{old}} \mathcal{N}(\boldsymbol{x}^{(n)}; \boldsymbol{\mu}_j^{\text{old}}, \boldsymbol{\Sigma}_j^{\text{old}})} \end{aligned}$$

EM for Gaussian Mixture Models

- M-step: Maximize the expected complete data log-likelihood:

$$\begin{aligned}\boldsymbol{\theta}_{\text{new}} &= \operatorname*{argmax}_{\boldsymbol{\theta}} \sum_{n=1}^{N} \sum_{k=1}^{K} q_n(z_k^{(n)} = 1) \log p(z_k^{(n)} = 1, \boldsymbol{x}^{(n)}; \boldsymbol{\theta}) \\ &= \operatorname*{argmax}_{\boldsymbol{\theta}} \sum_{n=1}^{N} \sum_{k=1}^{K} q_n(z_k^{(n)} = 1)(\log \pi_k + \log \mathcal{N}(\boldsymbol{x}^{(n)}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k))\end{aligned}$$

- The last equality is because:

$$\begin{aligned}\log p(z_k^{(n)} = 1, \boldsymbol{x}^{(n)}; \boldsymbol{\theta}) &= \log(p(z_k^{(n)} = 1; \boldsymbol{\theta}) \cdot p(\boldsymbol{x}^{(n)} \mid z_k^{(n)} = 1; \boldsymbol{\theta})) \\ &= \log p(z_k^{(n)} = 1; \boldsymbol{\theta}) + \log p(\boldsymbol{x}^{(n)} \mid z_k^{(n)} = 1; \boldsymbol{\theta}) \\ &= \log \pi_k + \log \mathcal{N}(\boldsymbol{x}^{(n)}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\end{aligned}$$

- This follows from the GMM model assumptions:
  - $p(z_k^{(n)} = 1; \boldsymbol{\theta}) = \pi_k$ (mixing coefficient)
  - $p(\boldsymbol{x}^{(n)} \mid z_k^{(n)} = 1; \boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{x}^{(n)}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ (Gaussian distribution)

EM for Gaussian Mixture Models

- M-step: Update the parameters:

$$\pi_k^{\text{new}} = \frac{1}{N} \sum_{n=1}^{N} q_n(z_k^{(n)} = 1)$$

$$\boldsymbol{\mu}_k^{\text{new}} = \frac{\sum_{n=1}^{N} q_n(z_k^{(n)} = 1)\boldsymbol{x}^{(n)}}{\sum_{n=1}^{N} q_n(z_k^{(n)} = 1)}$$

$$\boldsymbol{\Sigma}_k^{\text{new}} = \frac{\sum_{n=1}^{N} q_n(z_k^{(n)} = 1)(\boldsymbol{x}^{(n)} - \boldsymbol{\mu}_k^{\text{new}})(\boldsymbol{x}^{(n)} - \boldsymbol{\mu}_k^{\text{new}})^{\mathsf{T}}}{\sum_{n=1}^{N} q_n(z_k^{(n)} = 1)}$$

M-step: Update Rules for Gaussian Mixture Models

- Update rule for mixing coefficients π_k:
  - Maximize the expected complete data log-likelihood with respect to π_k subject to the constraint $\sum_{k=1}^{K} \pi_k = 1$ using a Lagrange multiplier λ:

$$\mathcal{L}(\boldsymbol{\pi}, \lambda) = \sum_{n=1}^{N} \sum_{k=1}^{K} q_n(z_k^{(n)} = 1) \log \pi_k + \lambda \left(\sum_{k=1}^{K} \pi_k - 1 \right)$$

  - Setting the derivative with respect to π_k to zero and solving:

$$\frac{\partial \mathcal{L}}{\partial \pi_k} = \sum_{n=1}^{N} \frac{q_n(z_k^{(n)} = 1)}{\pi_k} + \lambda = 0$$

$$\pi_k = -\frac{1}{\lambda} \sum_{n=1}^{N} q_n(z_k^{(n)} = 1)$$

  - Using the constraint $\sum_{k=1}^{K} \pi_k = 1$, we can solve $\lambda = -N$ and thus we obtain:

$$\pi_k^{\text{new}} = \frac{1}{N} \sum_{n=1}^{N} q_n(z_k^{(n)} = 1)$$

M-step: Update Rules for Gaussian Mixture Models

- Update rule for means $\boldsymbol{\mu}_k$:
  - Maximize the expected complete data log-likelihood with respect to $\boldsymbol{\mu}_k$:

$$\boldsymbol{\mu}_k^{\text{new}} = \underset{\boldsymbol{\mu}_k}{\operatorname{argmax}} \sum_{n=1}^{N} q_n(z_k^{(n)} = 1) \log \mathcal{N}(\boldsymbol{x}^{(n)}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

  - Setting the derivative with respect to $\boldsymbol{\mu}_k$ to zero and solving:

$$\frac{\partial}{\partial \boldsymbol{\mu}_k} \sum_{n=1}^{N} q_n(z_k^{(n)} = 1) \log \mathcal{N}(\boldsymbol{x}^{(n)}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) = \sum_{n=1}^{N} q_n(z_k^{(n)} = 1) \boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x}^{(n)} - \boldsymbol{\mu}_k) = 0$$

    - Note 1: The log-likelihood of a multivariate Gaussian distribution is:

$$\log \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = -\frac{1}{2} \log \det(\boldsymbol{\Sigma}) - \frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu})^{\mathsf{T}} \boldsymbol{\Sigma}^{-1}(\boldsymbol{x} - \boldsymbol{\mu}) + \text{const}$$

    - Note 2:

$$\frac{\partial}{\partial \boldsymbol{s}}(\boldsymbol{x} - \boldsymbol{s})^{\mathsf{T}} \boldsymbol{W}(\boldsymbol{x} - \boldsymbol{s}) = -2\boldsymbol{W}^{-1}(\boldsymbol{x} - \boldsymbol{s})$$

  - We obtain:

$$\boldsymbol{\mu}_k^{\text{new}} = \frac{\sum_{n=1}^{N} q_n(z_k^{(n)} = 1)\boldsymbol{x}^{(n)}}{\sum_{n=1}^{N} q_n(z_k^{(n)} = 1)}$$

M-step: Update Rules for Gaussian Mixture Models

- Update rule for covariances $\boldsymbol{\Sigma}_k$:
  - Maximize the expected complete data log-likelihood with respect to $\boldsymbol{\Sigma}_k$:

$$\boldsymbol{\Sigma}_k^{\text{new}} = \operatorname*{argmax}_{\boldsymbol{\Sigma}_k} \sum_{n=1}^{N} q_n(z_k^{(n)} = 1) \log \mathcal{N}(\boldsymbol{x}^{(n)}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

  - Taking the derivative with respect to $\boldsymbol{\Sigma}_k$:

$$\begin{aligned}
&\frac{\partial}{\partial \boldsymbol{\Sigma}_k} \sum_{n=1}^{N} q_n(z_k^{(n)} = 1) \log \mathcal{N}(\boldsymbol{x}^{(n)}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \\
&= \sum_{n=1}^{N} q_n(z_k^{(n)} = 1) \frac{\partial}{\partial \boldsymbol{\Sigma}_k} \left(-\frac{1}{2} \log \det(\boldsymbol{\Sigma}_k) - \frac{1}{2} (\boldsymbol{x}^{(n)} - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1} (\boldsymbol{x}^{(n)} - \boldsymbol{\mu}_k) \right) \\
&= \sum_{n=1}^{N} q_n(z_k^{(n)} = 1) \left(-\frac{1}{2} \boldsymbol{\Sigma}_k^{-1} + \frac{1}{2} \boldsymbol{\Sigma}_k^{-1} (\boldsymbol{x}^{(n)} - \boldsymbol{\mu}_k)(\boldsymbol{x}^{(n)} - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1} \right)
\end{aligned}$$

  - Note: We used the following matrix calculus identities:
    - $\frac{\partial}{\partial \boldsymbol{A}} \log \det(\boldsymbol{A}) = \boldsymbol{A}^{-1}$
    - $\frac{\partial}{\partial \boldsymbol{A}} \boldsymbol{x}^\top \boldsymbol{A}^{-1} \boldsymbol{x} = -\boldsymbol{A}^{-1} \boldsymbol{x} \boldsymbol{x}^\top \boldsymbol{A}^{-1}$

M-step: Update Rules for Gaussian Mixture Models

- Update rule for covariances $\boldsymbol{\Sigma}_k$ (cont.):
  - Setting the derivative to zero and solving for $\boldsymbol{\Sigma}_k$:

$$\sum_{n=1}^{N} q_n(z_k^{(n)} = 1)\left(-\frac{1}{2}\boldsymbol{\Sigma}_k^{-1} + \frac{1}{2}\boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x}^{(n)} - \boldsymbol{\mu}_k)(\boldsymbol{x}^{(n)} - \boldsymbol{\mu}_k)^{\mathsf{T}}\boldsymbol{\Sigma}_k^{-1}\right) = 0$$

$$-\frac{1}{2}\sum_{n=1}^{N} q_n(z_k^{(n)} = 1)\boldsymbol{\Sigma}_k^{-1} + \frac{1}{2}\sum_{n=1}^{N} q_n(z_k^{(n)} = 1)\boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x}^{(n)} - \boldsymbol{\mu}_k)(\boldsymbol{x}^{(n)} - \boldsymbol{\mu}_k)^{\mathsf{T}}\boldsymbol{\Sigma}_k^{-1} = 0$$

$$\sum_{n=1}^{N} q_n(z_k^{(n)} = 1)\boldsymbol{\Sigma}_k = \sum_{n=1}^{N} q_n(z_k^{(n)} = 1)(\boldsymbol{x}^{(n)} - \boldsymbol{\mu}_k)(\boldsymbol{x}^{(n)} - \boldsymbol{\mu}_k)^{\mathsf{T}}$$

$$\boldsymbol{\Sigma}_k^{\text{new}} = \frac{\sum_{n=1}^{N} q_n(z_k^{(n)} = 1)(\boldsymbol{x}^{(n)} - \boldsymbol{\mu}_k^{\text{new}})(\boldsymbol{x}^{(n)} - \boldsymbol{\mu}_k^{\text{new}})^{\mathsf{T}}}{\sum_{n=1}^{N} q_n(z_k^{(n)} = 1)}$$

  - Note: We used $\boldsymbol{\mu}_k^{\text{new}}$ instead of $\boldsymbol{\mu}_k$ in the last step to ensure that the updated covariance matrix is consistent with the updated mean

Summary

- Latent variable models introduce unobserved variables to simplify the modeling of complex data
- The EM algorithm is a general method for optimizing latent variable models
- EM alternates between making the ELBO tight (E-step) and optimizing it (M-step)
- The E-step involves computing the posterior distribution of the latent variables given the observed data
- The M-step maximizes the expected complete data log-likelihood