南方科技大学

MAT8034: Machine Learning

EM Algorithms

Fang Kong

https://fangkongx.github.io/Teaching/MAT8034/Spring2025/index.html

Part of slide credit: Stanford CS229

Outline

- EM for the mixture of Gaussians
- Jensen's inequality
- General EM algorithms

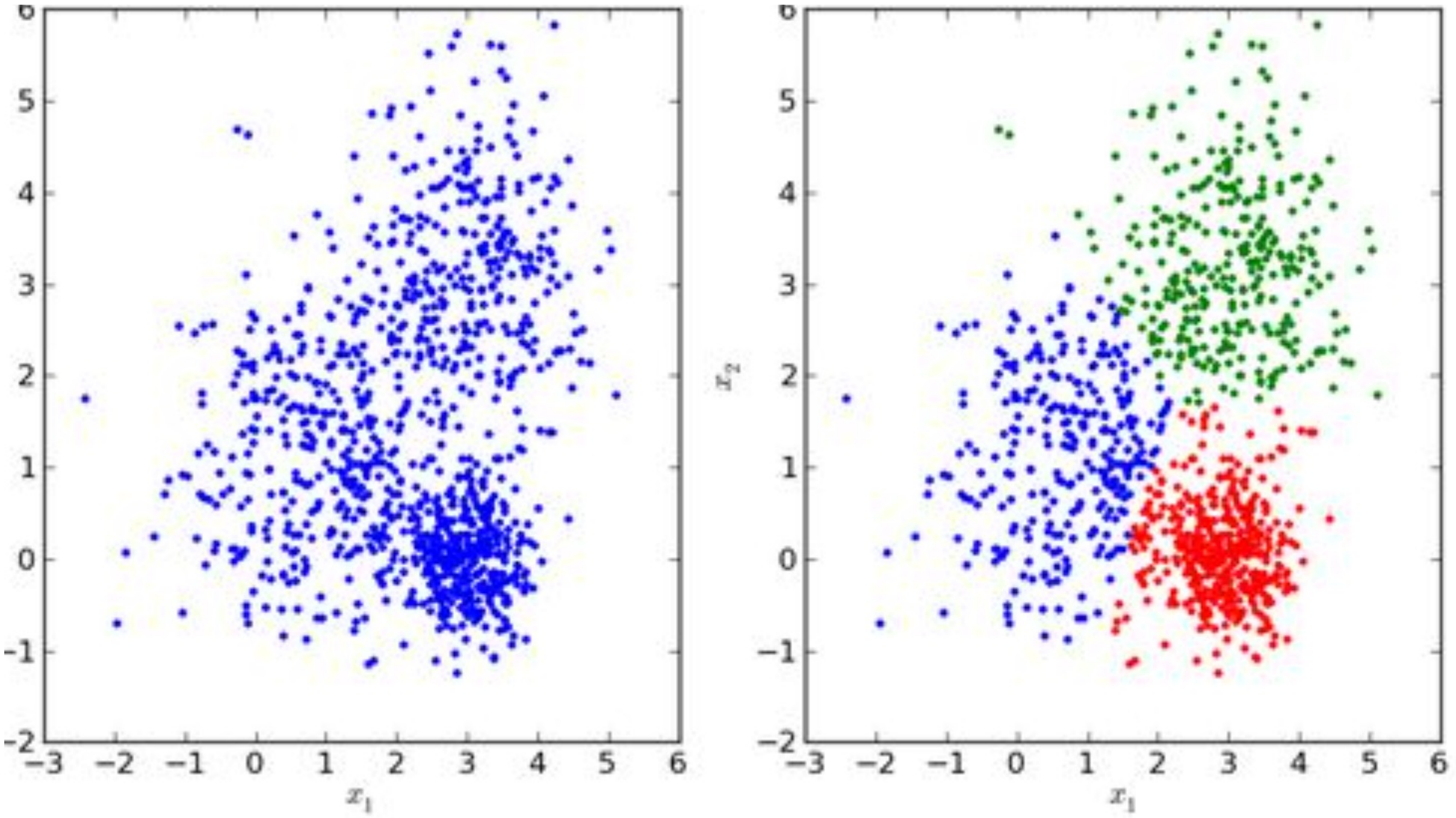
Intuition

- Recall that in supervised learning, we are given the training set without labels

$$\{x^{(1)}, \ldots, x^{(n)}\}$$

- We can assume these data are from different underlying classes $j = 1,2, \ldots, k$
- Each class is modeled by a Gaussian $\mathcal{N}(\mu_j, \Sigma_j)$
- The class label follows a multinomial distribution
  - Each data can only belong to one of these classes
  - Distribution parameter ϕ with $\phi_j \geq 0$ and $\sum_j \phi_j = 1$

Illustration



https://roboticsknowledgebase.com/wiki/math/gaussian-process-gaussian-mixture-model/

Mixture of gaussian models

- Each data x^i corresponds to a **(latent)** class label z^i
- z^i~Multinomial(ϕ), with $\phi_j \geq 0$ and $\sum_j \phi_j = 1$
    - $\mathbb{P}(z^i = j) = \phi_j$
- $x^i \mid z^i = j \sim \mathcal{N}(\mu_j, \Sigma_j)$

Maximum likelihood

- Log-likelihood

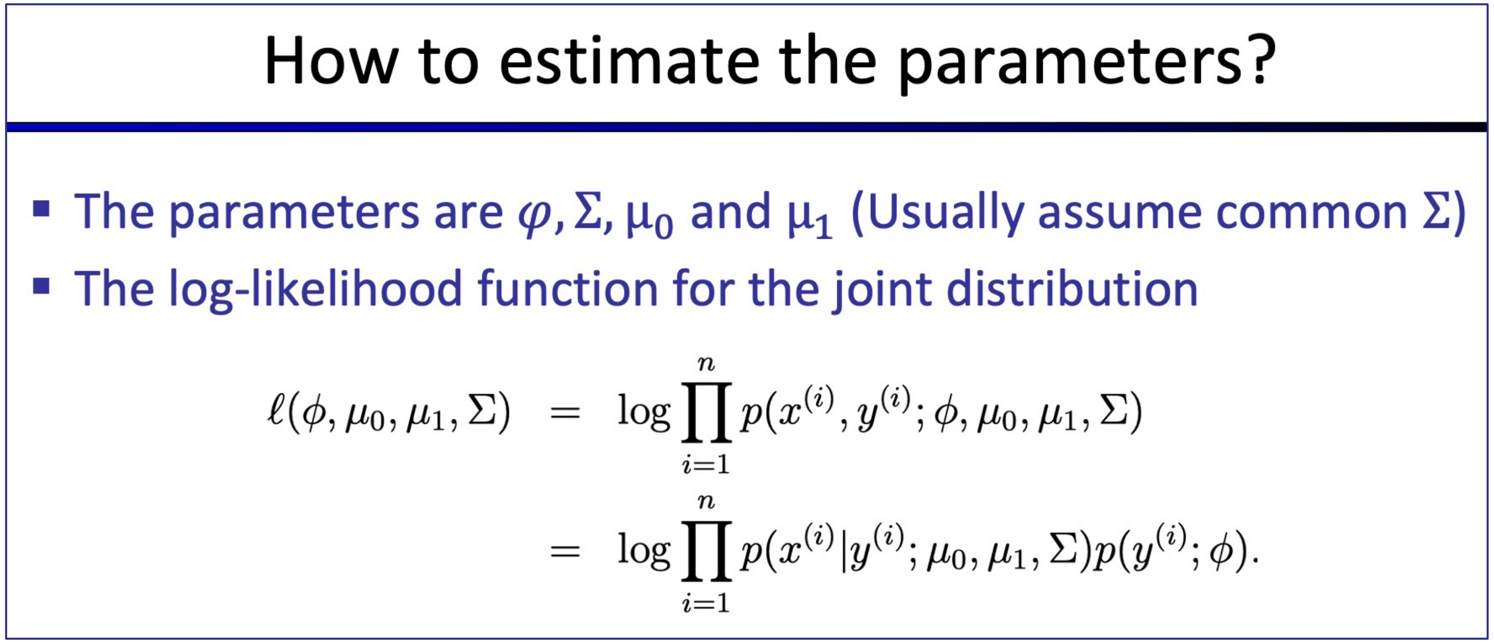
$$\begin{aligned}\ell(\phi,\mu,\Sigma) &= \sum_{i=1}^{n} \log p(x^{(i)};\phi,\mu,\Sigma) \\ &= \sum_{i=1}^{n} \log \sum_{z^{(i)}=1}^{k} p(x^{(i)}|z^{(i)};\mu,\Sigma)p(z^{(i)};\phi)\end{aligned}$$

- Zero the derivatives of this formula, but challenging to find the closed-form solution

Relaxation: If we know the class label

- The log-likelihood becomes

$$\ell(\phi, \mu, \Sigma) = \sum_{i=1}^{n} \log p(x^{(i)}|z^{(i)}; \mu, \Sigma) + \log p(z^{(i)}; \phi)$$



How to estimate the parameters?

- The parameters are φ, Σ, μ_0 and μ_1 (Usually assume common Σ)
- The log-likelihood function for the joint distribution

$$\begin{aligned}\ell(\phi, \mu_0, \mu_1, \Sigma) &= \log \prod_{i=1}^{n} p(x^{(i)}, y^{(i)}; \phi, \mu_0, \mu_1, \Sigma) \\ &= \log \prod_{i=1}^{n} p(x^{(i)}|y^{(i)}; \mu_0, \mu_1, \Sigma) p(y^{(i)}; \phi).\end{aligned}$$

Relaxation: If we know the class label (cont'd)

- The log-likelihood becomes

$$\ell(\phi,\mu,\Sigma) = \sum_{i=1}^{n} \log p(x^{(i)}|z^{(i)};\mu,\Sigma) + \log p(z^{(i)};\phi)$$

- Zero the derivatives and get

$$\begin{aligned}
\phi_j &= \frac{1}{n}\sum_{i=1}^{n} 1\{z^{(i)}=j\}, \\
\mu_j &= \frac{\sum_{i=1}^{n} 1\{z^{(i)}=j\}x^{(i)}}{\sum_{i=1}^{n} 1\{z^{(i)}=j\}}, \\
\Sigma_j &= \frac{\sum_{i=1}^{n} 1\{z^{(i)}=j\}(x^{(i)}-\mu_j)(x^{(i)}-\mu_j)^T}{\sum_{i=1}^{n} 1\{z^{(i)}=j\}}
\end{aligned}$$

How to solve with unknown z^i?

Iterative algorithm to update z^i

- Repeat until converge
  - Guess the value of z^i: compute the posterior probability

$$w_j^{(i)} := p(z^{(i)} = j | x^{(i)}; \phi, \mu, \Sigma) = \frac{p(x^{(i)} | z^{(i)} = j; \mu, \Sigma) p(z^{(i)} = j; \phi)}{\sum_{l=1}^{k} p(x^{(i)} | z^{(i)} = l; \mu, \Sigma) p(z^{(i)} = l; \phi)}$$

  - Based on z^i, use maximum likelihood to estimate parameters

Iterative algorithm to update z^i

- Repeat until converge
  - Guess the value of z^i: compute the posterior probability
  - Based on z^i, use maximum likelihood to estimate parameters

$$
\begin{aligned}
\phi_j &:= \frac{1}{n}\sum_{i=1}^{n} w_j^{(i)}, \\
\mu_j &:= \frac{\sum_{i=1}^{n} w_j^{(i)} x^{(i)}}{\sum_{i=1}^{n} w_j^{(i)}}, \\
\Sigma_j &:= \frac{\sum_{i=1}^{n} w_j^{(i)} (x^{(i)} - \mu_j)(x^{(i)} - \mu_j)^T}{\sum_{i=1}^{n} w_j^{(i)}}
\end{aligned}
$$

Comparison with existing forms?

Expectation-Maximization

- Repeat until converge
  - Guess the value of z^i: compute the posterior probability

    Step E

  - Based on z^i, use maximum likelihood to estimate parameters

    Step M

Tool: Jensen’s inequality

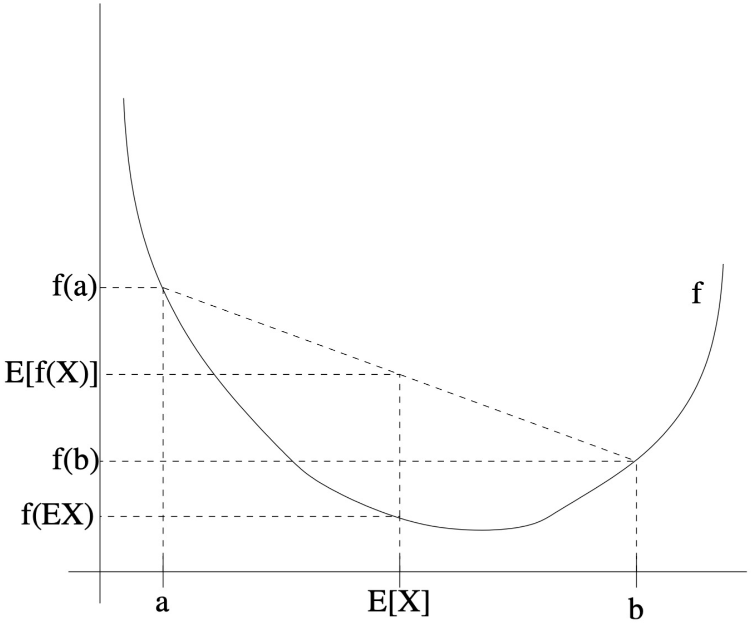
Convex functions

- Definition (convex functions)
  - f is a convex function if f’’(x) ≥ 0 (for all x ∈ R)
  - f is a strictly convex function if f’’(x) > 0 (for all x ∈ R)

  - If taking vector-valued inputs, f is a convex function if its hessian H is positive semi-definite

Jensen's inequality

- **Theorem.** Let f be a convex function, and let X be a random variable. Then:
$$\mathrm{E}[f(X)] \geq f(\mathrm{E}X).$$
Moreover, if f is strictly convex, then $\mathrm{E}[f(X)] = f(\mathrm{E}X)$ holds true if and only if $X = \mathrm{E}[X]$ with probability 1 (i.e., if X is a constant).



Concave functions

- Definition (concave functions)
  - f is [strictly] concave if and only if –f is [strictly] convex (i.e., f''(x) ≤ 0 or H ≤ 0).
  - Jensen's inequality also holds for concave functions f with E[f(X)] ≤ f(EX)

General EM algorithms

Setting

- Recall we have the training set $\{x^{(1)}, \ldots, x^{(n)}\}$
- We have a latent variable model $p(x, z; \theta)$

- Hope to maximize the likelihood

$$\begin{aligned} \ell(\theta) &= \sum_{i=1}^{n} \log p(x^{(i)}; \theta) \\ &= \sum_{i=1}^{n} \log \sum_{z^{(i)}} p(x^{(i)}, z^{(i)}; \theta) \longleftarrow \boxed{p(x; \theta) = \sum_{z} p(x, z; \theta)} \end{aligned}$$

Intuition

- Directly optimizing the likelihood is infeasible

- How about optimizing the **lower bound** of the likelihood?
  - Construct a lower bound – Step E
  - Optimizing the lower bound – Step M

Lower bound of the likelihood

- Hope to derive the **lower bound** for

$$\log p(x;\theta) = \log \sum_z p(x, z; \theta)$$

-

$$\begin{aligned} \log p(x;\theta) &= \log \sum_z p(x, z; \theta) \\ &= \log \sum_z Q(z) \frac{p(x, z; \theta)}{Q(z)} \end{aligned}$$

Q is any distribution on z with $Q(z) \geq 0$ and $\sum_z Q(z) = 1$

Jensen's inequality

$$\geq \sum_z Q(z) \log \frac{p(x, z; \theta)}{Q(z)}$$

Choice of Q

- For any distribution Q, we have the lower bound

$$\log p(x;\theta) \;\geq\; \sum_z Q(z)\log\frac{p(x,z;\theta)}{Q(z)}$$

- How to choose Q?
  - Try to make the lower-bound tight at that value of θ
  - Hope the inequality hold with equality

How?

Choice of Q (cont'd)

- Hope the inequality hold with equality

How?

- Recall that in the Jensen's inequality, the equality holds when X is a constant
  - To make $\frac{p(x,z;\theta)}{Q(z)}$ be a constant, let $Q(z) \propto p(x,z;\theta)$.
  - Since $\sum_z Q(z) = 1$, it follows that

$$\begin{aligned} Q(z) &= \frac{p(x,z;\theta)}{\sum_z p(x,z;\theta)} \\ &= \frac{p(x,z;\theta)}{p(x;\theta)} \\ &= p(z|x;\theta) \end{aligned}$$

Verify the equality with $Q(z) = p(z|x;\theta)$

- $$\begin{aligned}\sum_z Q(z) \log \frac{p(x,z;\theta)}{Q(z)} &= \sum_z p(z|x;\theta) \log \frac{p(x,z;\theta)}{p(z|x;\theta)} \\ &= \sum_z p(z|x;\theta) \log \frac{p(z|x;\theta)p(x;\theta)}{p(z|x;\theta)} \\ &= \sum_z p(z|x;\theta) \log p(x;\theta) \\ &= \log p(x;\theta) \sum_z p(z|x;\theta) \\ &= \log p(x;\theta) \qquad\qquad (\text{because } \textstyle\sum_z p(z|x;\theta) = 1)\end{aligned}$$

Evidence lower bound (ELBO)

EM algorithm procedure

- Foundation

$$\forall Q, \theta, x, \qquad \log p(x;\theta) \geq \mathrm{ELBO}(x;Q,\theta)$$

- Procedure of EM
  - Setting Q(z) = p(z|x; θ) so that ELBO(x; Q, θ) = log p(x; θ)
  - Maximizing ELBO(x; Q, θ) w.r.t θ while fixing the choice of Q

Generalization to multiple training data

- $$\begin{aligned}\ell(\theta) &\geq \sum_i \mathrm{ELBO}(x^{(i)}; Q_i, \theta) \\ &= \sum_i \sum_{z^{(i)}} Q_i(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)}; \theta)}{Q_i(z^{(i)})}\end{aligned}$$

- The equality holds with $Q_i(z^{(i)}) \quad = \quad p(z^{(i)} | x^{(i)}; \theta)$

Formal procedure of EM

- Repeat until convergence {

  (E-step) For each i, set

$$Q_i(z^{(i)}) := p(z^{(i)}|x^{(i)};\theta).$$

  (M-step) Set

$$\begin{aligned}\theta &:= \arg\max_\theta \sum_{i=1}^{n} \mathrm{ELBO}(x^{(i)};Q_i,\theta) \\ &= \arg\max_\theta \sum_i \sum_{z^{(i)}} Q_i(z^{(i)}) \log \frac{p(x^{(i)},z^{(i)};\theta)}{Q_i(z^{(i)})}\end{aligned}$$

  }

Convergence analysis

- Objective: prove $\ell(\theta^{(t)}) \;\leq\; \ell(\theta^{(t+1)})$

- Proof

$$\ell(\theta^{(t+1)}) \geq \sum_{i=1}^{n} \mathrm{ELBO}(x^{(i)}; Q_i^{(t)}, \theta^{(t+1)})$$

Jensen's inequality

$$\geq \sum_{i=1}^{n} \mathrm{ELBO}(x^{(i)}; Q_i^{(t)}, \theta^{(t)})$$

Updating rule

$$= \ell(\theta^{(t)})$$

Selection of Q

Formal procedure of EM (cont'd)

- Repeat until convergence {

When the change between θ^{t+1} and θ^{t} is small enough

(E-step) For each i, set

$$Q_i(z^{(i)}) := p(z^{(i)}|x^{(i)};\theta).$$

(M-step) Set

$$\begin{aligned}\theta &:= \arg\max_{\theta} \sum_{i=1}^{n} \mathrm{ELBO}(x^{(i)};Q_i,\theta) \\ &= \arg\max_{\theta} \sum_{i} \sum_{z^{(i)}} Q_i(z^{(i)}) \log \frac{p(x^{(i)},z^{(i)};\theta)}{Q_i(z^{(i)})}\end{aligned}$$

}

Other interpretation of EM/ELBO

EM=alternating maximization on ELBO(Q, θ)

- Define ELBO(Q, θ)

$$\mathrm{ELBO}(Q,\theta) = \sum_{i=1}^{n} \mathrm{ELBO}(x^{(i)}; Q_i, \theta) = \sum_{i} \sum_{z^{(i)}} Q_i(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)}; \theta)}{Q_i(z^{(i)})}$$

  - E step: maximizes ELBO(Q, θ) with respect to Q
  - M step: maximizes ELBO(Q, θ) with respect to θ

Hint: show that

$$\begin{aligned}\mathrm{ELBO}(x; Q, \theta) &= \textstyle\sum_z Q(z) \log \frac{p(x,z;\theta)}{Q(z)} \\ &= \log p(x) - D_{KL}(Q \| p_{z|x})\end{aligned}$$

KL-divergence form of ELBO

- Rewrite ELBO:

$$\begin{aligned}\mathrm{ELBO}(x;Q,\theta) &= \textstyle\sum_z Q(z)\log\frac{p(x,z;\theta)}{Q(z)} \\ &= \mathrm{E}_{z\sim Q}[\log p(x,z;\theta)] - \mathrm{E}_{z\sim Q}[\log Q(z)] \\ &= \mathrm{E}_{z\sim Q}[\log p(x|z;\theta)] - D_{KL}(Q\|p_z)\end{aligned}$$

$$\boxed{D_{KL}(Q\|p_z) = \sum_z Q(z)\log\frac{Q(z)}{p(z)}}$$

  - The second term does not depend on θ, so maximizing ELBO over θ is equivalent to maximizing the first term
  - Corresponds to maximizing the conditional likelihood of x conditioned on z

Back to Mixture of Gaussians

Mixture of Gaussians

- Recall the iterative optimization algorithm for Mixture of Gaussians

- Repeat until converge
  - Guess the value of z^i: compute the posterior probability

$$w_j^{(i)} := p(z^{(i)} = j | x^{(i)}; \phi, \mu, \Sigma) = \frac{p(x^{(i)} | z^{(i)} = j; \mu, \Sigma) p(z^{(i)} = j; \phi)}{\sum_{l=1}^{k} p(x^{(i)} | z^{(i)} = l; \mu, \Sigma) p(z^{(i)} = l; \phi)}$$

  - Based on z^i, use maximum likelihood to estimate parameters

Applying general EM to Mixture of Gaussians

- Step E: compute the posterior probability

$$w_j^{(i)} = Q_i(z^{(i)} = j) = P(z^{(i)} = j | x^{(i)}; \phi, \mu, \Sigma)$$

- Step M: maximize

$$\begin{aligned}
&\sum_{i=1}^{n} \sum_{z^{(i)}} Q_i(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)}; \phi, \mu, \Sigma)}{Q_i(z^{(i)})} \\
&\quad = \sum_{i=1}^{n} \sum_{j=1}^{k} Q_i(z^{(i)} = j) \log \frac{p(x^{(i)} | z^{(i)} = j; \mu, \Sigma) p(z^{(i)} = j; \phi)}{Q_i(z^{(i)} = j)} \\
&\quad = \sum_{i=1}^{n} \sum_{j=1}^{k} w_j^{(i)} \log \frac{\frac{1}{(2\pi)^{d/2} |\Sigma_j|^{1/2}} \exp\left(-\frac{1}{2}(x^{(i)} - \mu_j)^T \Sigma_j^{-1} (x^{(i)} - \mu_j)\right) \cdot \phi_j}{w_j^{(i)}}
\end{aligned}$$

Solve μ

- Zero the derivative

$$\nabla_{\mu_l} \sum_{i=1}^{n} \sum_{j=1}^{k} w_j^{(i)} \log \frac{\frac{1}{(2\pi)^{d/2}|\Sigma_j|^{1/2}} \exp\left(-\frac{1}{2}(x^{(i)} - \mu_j)^T \Sigma_j^{-1} (x^{(i)} - \mu_j)\right) \cdot \phi_j}{w_j^{(i)}}$$

$$= \quad -\nabla_{\mu_l} \sum_{i=1}^{n} \sum_{j=1}^{k} w_j^{(i)} \frac{1}{2} (x^{(i)} - \mu_j)^T \Sigma_j^{-1} (x^{(i)} - \mu_j)$$

$$= \quad \frac{1}{2} \sum_{i=1}^{n} w_l^{(i)} \nabla_{\mu_l} 2\mu_l^T \Sigma_l^{-1} x^{(i)} - \mu_l^T \Sigma_l^{-1} \mu_l$$

$$= \quad \sum_{i=1}^{n} w_l^{(i)} \left(\Sigma_l^{-1} x^{(i)} - \Sigma_l^{-1} \mu_l \right)$$

$$\mu_l := \frac{\sum_{i=1}^{n} w_l^{(i)} x^{(i)}}{\sum_{i=1}^{n} w_l^{(i)}}$$

Solve ϕ

- Terms related to ϕ: $\sum_{i=1}^{n}\sum_{j=1}^{k} w_j^{(i)} \log \phi_j$
- Additional constraint: $\sum_j \phi_j = 1$
- Construct the Lagrangian $\mathcal{L}(\phi) = \sum_{i=1}^{n}\sum_{j=1}^{k} w_j^{(i)} \log \phi_j + \beta(\sum_{j=1}^{k} \phi_j - 1)$
- Zero the derivatives $\frac{\partial}{\partial \phi_j}\mathcal{L}(\phi) = \sum_{i=1}^{n} \frac{w_j^{(i)}}{\phi_j} + \beta$ and get $\phi_j = \frac{\sum_{i=1}^{n} w_j^{(i)}}{-\beta}$
- Using the constraint and get $\phi_j := \frac{1}{n}\sum_{i=1}^{n} w_j^{(i)}$

Summary

- EM for the mixture of Gaussians
- Jensen’s inequality
- General EM algorithms
  - ELBO
  - Different interpretations

Extension to high dimensional latent variables

- Variational auto-encoder (VAE)
  - A widely-known generative model
  - Foundations for GAN and diffusion models

- Different from Gaussian mixtures, now consider that

$$z \sim \mathcal{N}(0, I_{k\times k})$$
$$x|z \sim \mathcal{N}(g(z;\theta), \sigma^2 I_{d\times d})$$

  - θ is the collection of the weights of a neural network
  - g(z; θ) maps z ∈ R^k to R^d
  - Challenging to compute the exact posterior distribution

Solution: find the approximation

Extension to high dimensional latent variables

- Optimizing ELBO over a pre-defined class Q

$$\boxed{\max_{Q\in\mathcal{Q}}}\max_{\theta}\mathrm{ELBO}(Q,\theta)$$

- Common assumption over Q: mean field assumption
  - $Q_i(z)$ gives a distribution with independent coordinates

$$Q_i = \mathcal{N}(q(x^{(i)};\phi), \mathrm{diag}(v(x^{(i)};\psi))^2)$$

Chosen as neural networks
Referred to as the encoder: encodes the data into latent code

What is the decoder?

Optimize ELBO

- Evaluate ELBO:

$$\mathrm{ELBO}(\phi, \psi, \theta) = \sum_{i=1}^{n} \mathrm{E}_{z^{(i)} \sim Q_i} \left[\log \frac{p(x^{(i)}, z^{(i)}; \theta)}{Q_i(z^{(i)})} \right],$$

$$\text{where } Q_i = \mathcal{N}(q(x^{(i)}; \phi), \mathrm{diag}(v(x^{(i)}; \psi))^2)$$

Sample multiple data to approximate

- Optimizing ELBO:
  - Run gradient ascent over ϕ, ψ, θ

re-parameterization trick to solve

$$\theta := \theta + \eta \nabla_\theta \mathrm{ELBO}(\phi, \psi, \theta)$$

$$\phi := \phi + \eta \nabla_\phi \mathrm{ELBO}(\phi, \psi, \theta)$$

$$\psi := \psi + \eta \nabla_\psi \mathrm{ELBO}(\phi, \psi, \theta)$$