CSE/ISyE 6740
Computational Data Analysis

Gaussian Mixture Models (GMMs)

09/03/2025

Kai Wang, Assistant Professor in Computational Science and Engineering
kwang692@gatech.edu

Georgia Tech

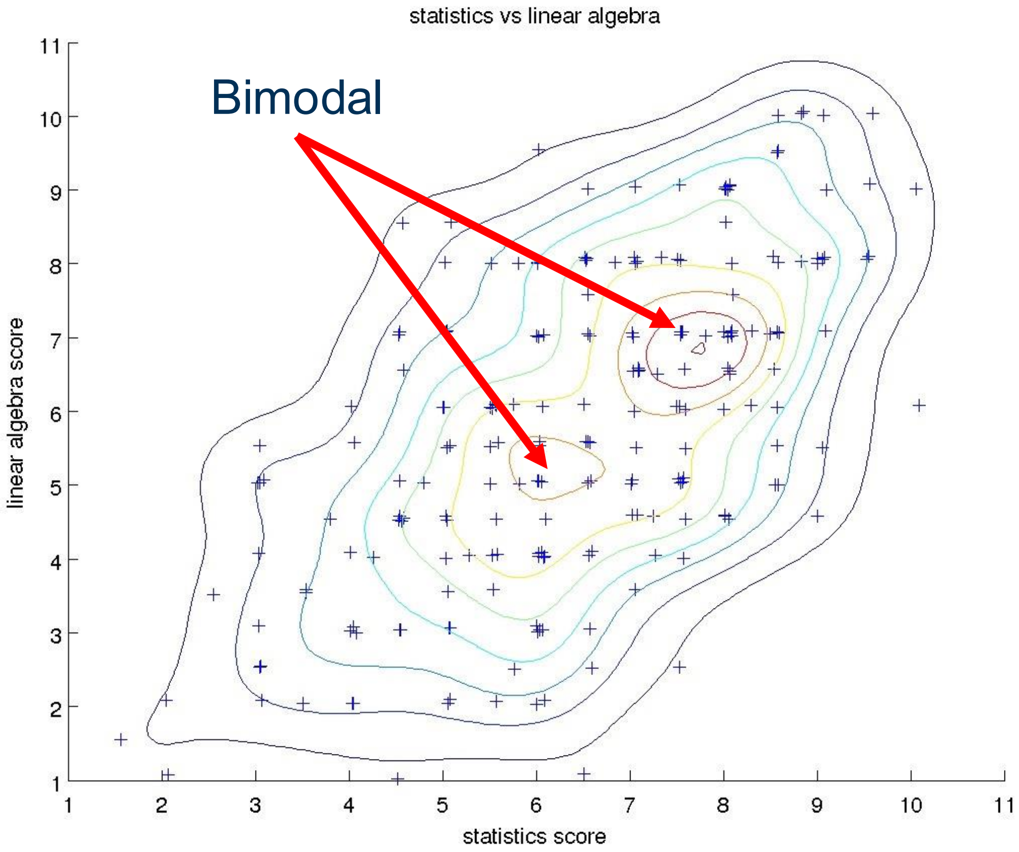
Outline

- **Unsupervised Learning**
  - Density estimation
    - Gaussian mixture models (GMMs)
      - Probability density function of mixture of Gaussians
      - Expectation-Maximization (EM) algorithm
      - Mathematical meaning of the EM algorithm

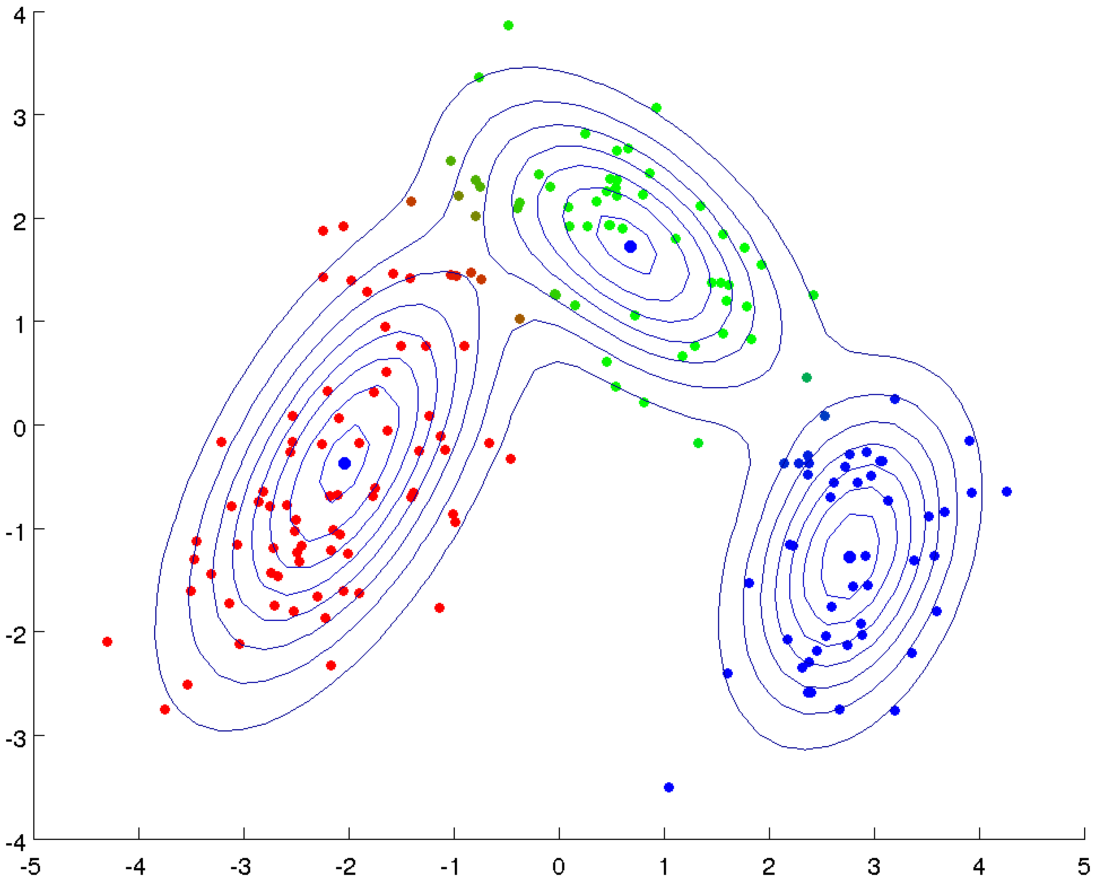
Gaussian Mixture Models

Density Estimation Revisited

- Can we have a more flexible model than Gaussian, but less parameter than KDE?



Demo: test_wine.py



Georgia Tech

Gaussian Mixture Model

- A density model $p(X)$ may be multi-modal: model it as a mixture of uni-modal distributions (e.g., Gaussians)

$$\mathcal{N}(X|\mu_k, \Sigma_k) \coloneqq \frac{1}{|\Sigma_k|^{\frac{1}{2}}(2\pi)^{\frac{d}{2}}} \exp\left(-\frac{1}{2}(X-\mu_k)^\top \Sigma^{-1}(X-\mu_k)\right)$$

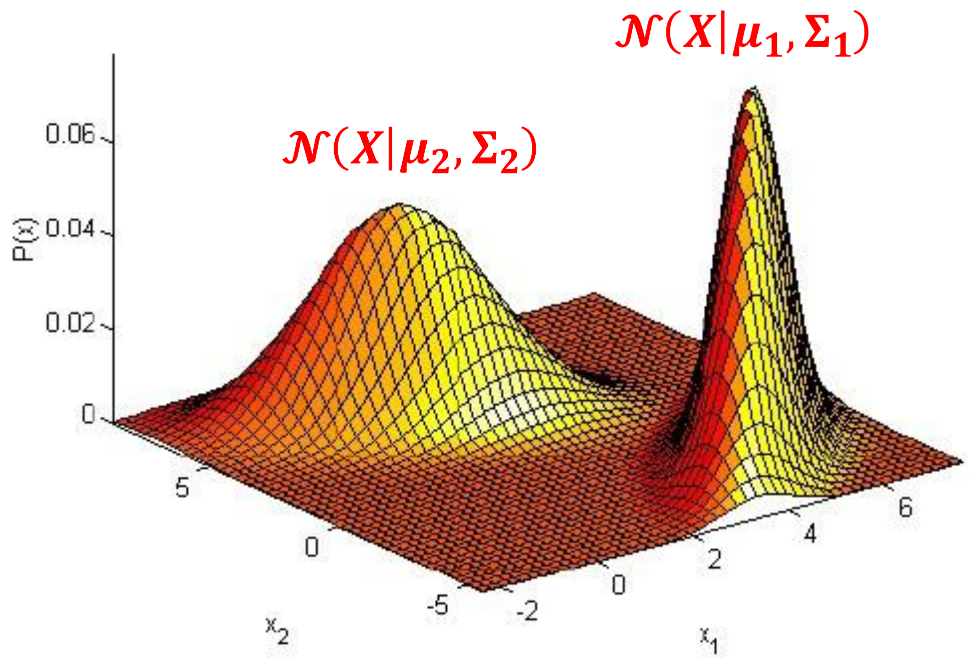
- Consider a mixture of K Gaussians
  - $p(X) \coloneqq \sum_{k=1}^{K} \pi_k \, \mathcal{N}(X|\mu_k, \Sigma_k)$

Mixing proportion

Mixing components

- **Question**: parametric or nonparametric?

- Learn $\pi_k \in (0,1), \mu_k, \Sigma_k$



A Generative Process for Data Points

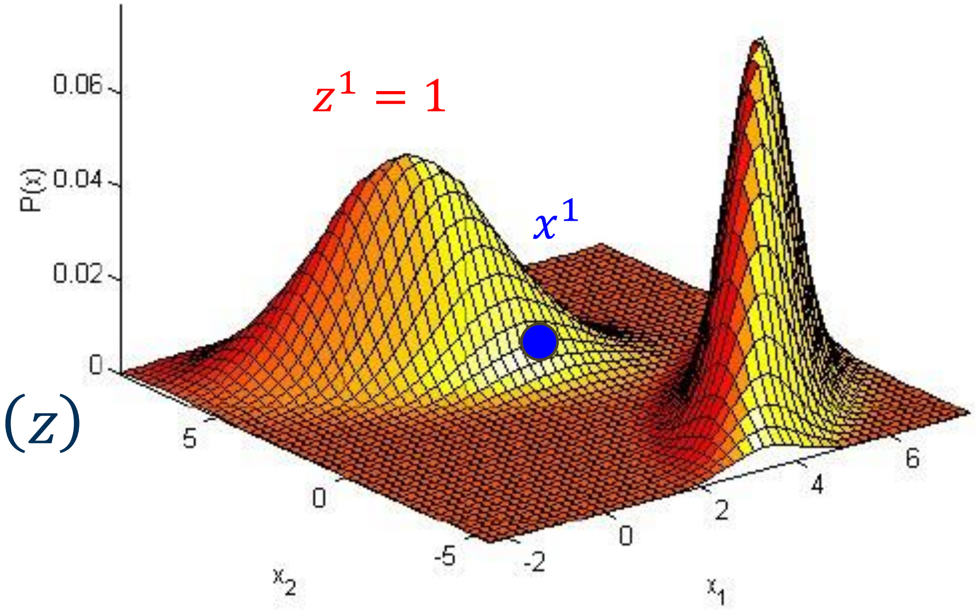
- Given $p(X) \coloneqq \sum_{k=1}^{K} \pi_k \, \mathcal{N}(X|\mu_k, \Sigma_k)$, how to generate a data point $x^i \sim p$?
  - Randomly choose a mixture component, $z^i \in \{1,2,\ldots,K\}$, with probability π_{z^i}
  - Sample x^i from the corresponding Gaussian distribution $\mathcal{N}(X|\mu_{z^i}, \Sigma_{z^i})$

- Joint distribution $p(x,z)$ over x and z:

$$p(x,z) = \pi_z \mathcal{N}(X|\mu_z, \Sigma_z)$$

- Marginal distribution $p(x)$

$$p(x) = \sum_{z=1}^{K} p(x,z) = \sum_{z=1}^{K} p(x|z)p(z)$$



Learning the Parameters

- We know how to sample. But how to learn the parameters?

- **Maximum likelihood estimation (MLE)** by letting $\theta = (\pi_k, \mu_k, \Sigma_k)_{k=1,2,\ldots,K}$:

$$\theta^* = argmax_\theta \, l(\theta; D) \coloneqq \log P(D|\theta)$$

- Use our generative process

$$l(\theta; D) = \log \prod_{i=1}^{n} p(x^i|\theta)$$

$$= \log \prod_{i=1}^{n} \left(\sum_{z^i=1}^{K} p(x^i, z^i|\theta) \right)$$

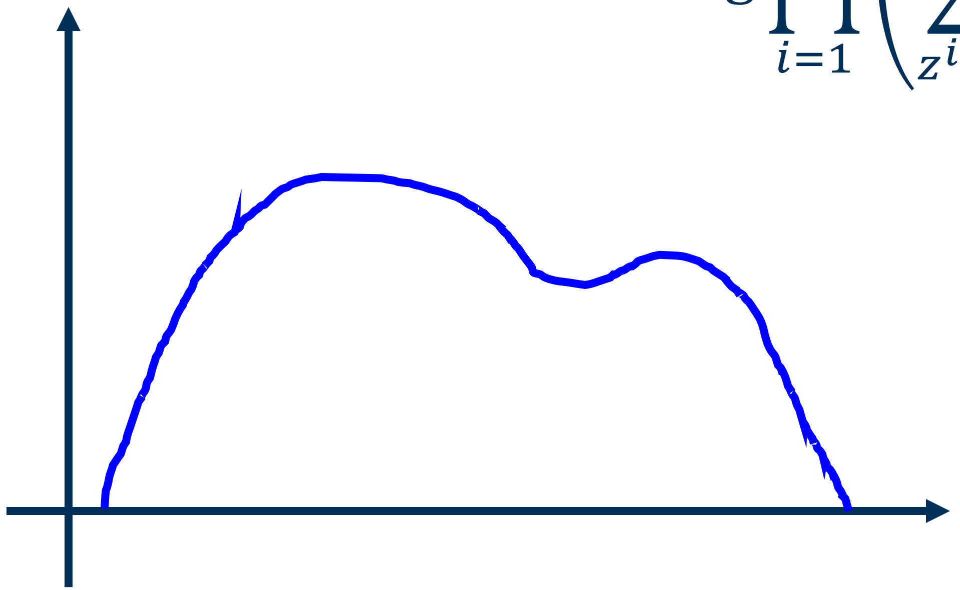
$$= \log \prod_{i=1}^{n} \left(\sum_{z^i=1}^{K} p(x|\mu_{z^i}, \Sigma_{z^i}) p(z^i|\pi) \right)$$

Why is Learning Hard?

- With latent variables z, likelihood of the data becomes:

$$l(\theta; D) = \log \prod_{i=1}^{n} \left(\sum_{z^i=1}^{K} p(x|\mu_{z^i}, \Sigma_{z^i}) p(z^i|\pi) \right)$$

$$= \log \prod_{i=1}^{n} \left(\sum_{z^i=1}^{K} \pi_{z^i} \, \mathcal{N}(x|\mu_{z^i}, \Sigma_{z^i}) \right)$$



Nonconvex!
Difficult!

When the Latent Variable z^i is Given

- For the case $z^i = k$ is given, we can simplify the log-likelihood by:

$$\log p\left(x^i, z^i = k \middle| \theta\right) = \log\left(p(x|\mu_k, \Sigma_k)p\left(z^i = k|\pi\right)\right)$$ ⟵ Because $z^i = k$ is known

$$= \log(\mathcal{N}(x|\mu_k, \Sigma_k)\pi_k)$$

$$= \sum_{k=1}^{K} \tau_k^i \log(\mathcal{N}(x|\mu_k, \Sigma_k)\pi_k)$$ ⟵ Simplify the expression by using $\tau_k^i = I(z^i = k)$ to denote if $z^i = k$

$$= \sum_{k=1}^{K} \tau_k^i \left[\log \pi_k - \frac{1}{2}\left(x^i - \mu_k\right)^{\top} \Sigma_k \left(x^i - \mu_k\right) - \frac{1}{2}\log|\Sigma_k| - c\right]$$

When the Latent Variable z^i is Given

- Introduce a binary variable $\tau_k^i \in \{0,1\}$ to denote $z^i = k$ as the **Gaussian assignment**

$$l(\theta; D, \tau) = \sum_{i=1}^{n} \sum_{k=1}^{K} \tau_k^i \left[\log \pi_k - \frac{1}{2}(x^i - \mu_k)^\top \Sigma_k^{-1}(x^i - \mu_k) - \frac{1}{2}\log|\Sigma_k| - c \right]$$

- It is now "easy" to show that (by using Lagrangian multiplier and setting $\frac{\partial l(\theta;D,\tau)}{\partial \theta} = 0$), the maximum likelihood estimation of the parameters are:

$$\pi_k = \frac{\sum_i \tau_k^i}{n}, \qquad \mu_k = \frac{\sum_i \tau_k^i x^i}{\sum_i \tau_k^i}$$

$$\Sigma_k = \frac{\sum_i \tau_k^i (x^i - \mu_k)(x^i - \mu_k)^\top}{\sum_i \tau_k^i}$$

Please work this out yourself!

For $k \in \{1,2, \dots, K\}$

When z is NOT Given

- We don't know τ_k^i, but we can guess which Gaussian x^i comes from by computing the posterior probability

$$p(z^i = k \mid x^i, \mu, \Sigma) = \frac{\pi_k \, \mathcal{N}(x^i | \mu_k, \Sigma_k)}{\sum_{k'=1}^{K} \pi_{k'} \, \mathcal{N}(x^i | \mu_{k'}, \Sigma_{k'})} \qquad \forall k, i$$

- Bayes rule

$$\underbrace{P(z|x)}_{\text{Posterior}} = \frac{\overbrace{P(x|z)}^{\text{Likelihood}}\overbrace{P(z)}^{\text{Prior}}}{\underbrace{P(x)}_{\text{Normalization constant}}} = \frac{P(x, z)}{\sum_{z'} P(x, z')}$$

Prior: $p(z) = \pi_z$

Likelihood: $p(x|z) = \mathcal{N}(x|\mu_z, \Sigma_z)$

When z is NOT Given

- We don't know τ_k^i, but we can guess which Gaussian x^i comes from by computing the posterior probability

$$p(z^i = k \mid x^i, \mu, \Sigma) = \frac{\pi_k \, \mathcal{N}(x^i | \mu_k, \Sigma_k)}{\sum_{k'=1}^{K} \pi_{k'} \, \mathcal{N}(x^i | \mu_{k'}, \Sigma_{k'})} \qquad \forall k, i$$

- Now we pretend $p(z^i = k \mid D, \mu, \Sigma)$ as our unknown **Gaussian assignment** τ_k^i, but now is regarded as a "soft" assignment:
  - Probability of assigning x^i to k-th component (Gaussian)

EM (Expectation-Maximization) Algorithm

- Associate each data and each component with a τ_k^i
- Initialized $(\pi_k, \mu_k, \Sigma_k), k = 1,2, \dots, K$
- Iterate the following two steps until convergence:
  - **Expectation step (E-step, Gaussian assignment)**: update τ_k^i given the current (π_k, μ_k, Σ_k)

$$\tau_k^i = p\left(z^i = k \mid D, \mu, \Sigma\right) = \frac{\pi_k \, \mathcal{N}(x|\mu_k, \Sigma_k)}{\sum_{k'=1}^{K} \pi_{k'} \, \mathcal{N}(x|\mu_{k'}, \Sigma_{k'})}, \qquad \forall k \in \{1,2, \dots, K\}; i \in \{1,2, \dots, n\}$$

  - **Maximization step (M-step, Gaussian adjustment)**: update (π_k, μ_k, Σ_k) given τ_k^i

$$\pi_k = \frac{\sum_i \tau_k^i}{n}, \qquad \mu_k = \frac{\sum_i \tau_k^i x^i}{\sum_i \tau_k^i}$$

$$\Sigma_k = \frac{\sum_i \tau_k^i \left(x^i - \mu_k\right)\left(x^i - \mu_k\right)^\top}{\sum_i \tau_k^i}, \qquad \forall k \in \{1,2, \dots, K\}$$

Details of EM

- We intend to learn the parameters that maximizes the log-likelihood of the data

$$l(\theta; D) = \log \prod_{i=1}^{n} \left(\sum_{z^i=1}^{K} p(x^i, z^i | \theta) \right)$$

- **Expectation step (E-step)**: set $\tau_k^i = p(z^i = k \mid D, \mu, \Sigma)$
  - What does this step mean?
  - Find the best lower bound $l(\theta; D, q)$ to estimate the expectation $l(\theta; D)$

$$l(\theta; D) \geq l(\theta; D, q) \coloneqq \mathbb{E}_{z^1, z^2, \ldots, z^n \sim q} \left[\log \prod_{i=1}^{n} p(x^i, z^i | \theta) \right] + H(q)$$

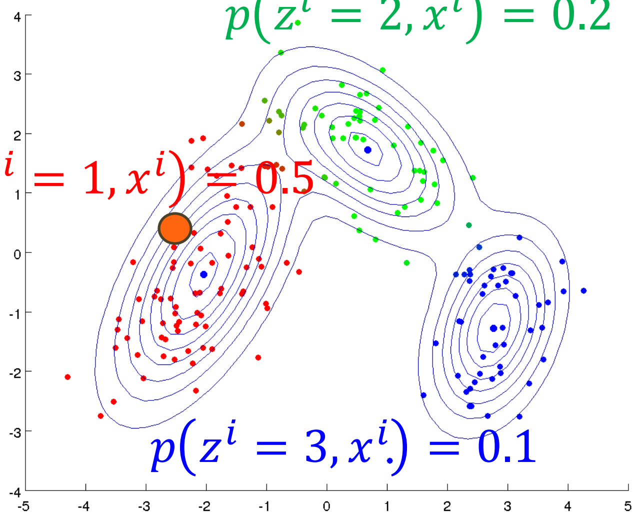
- Maximization step (M-step)

E-step: What is $\tau(z^1, z^2, \dots, z^n)$

- $q(z^1, z^2, \dots, z^n)$: any distribution of the latent variables

$$q(z^1, z^2, \dots, z^n) = \prod_{i=1}^{n} prob(z^i | x^i, \theta)$$

- Specifically, we choose $q(z^i) = \tau_k^i$ with



Therefore, we can find

$$\tau_{k=1}^i = \frac{0.5}{0.5 + 0.2 + 0.1}$$

$$\boxed{\tau_k^i = p(z^i = k |\, x^i)} = \frac{p(z^i = k, x^i)}{\sum_{k'=1,2,\dots,K} p(z^i = k', x^i)} = \frac{\pi_k\, \mathcal{N}(x | \mu_k, \Sigma_k)}{\sum_{k'=1}^{K} \pi_{k'}\, \mathcal{N}(x | \mu_{k'}, \Sigma_{k'})}$$

Conditioned on seeing x^i, what is the probability that x^i is sampled from $z^i = k$-th Gaussian distribution?

Can also be interpreted as a "soft" assignment

E-step: Compute the Expectation

$$
\begin{aligned}
l(\theta; D, q) &= \mathbb{E}_{z^1, z^2, \ldots, z^n \sim q}\left[\log \prod_{i=1}^{n} p\left(x^i, z^i \mid \theta\right)\right] + H(q) \\
&= \mathbb{E}_{z^1, z^2, \ldots, z^n \sim q}\left[\sum_{i=1}^{n} \left[\log p\left(x^i, z^i \mid \theta\right)\right]\right] + H(q) \\
&= \mathbb{E}_{z^i \sim p\left(z^i \mid x^i\right)\ \forall i} \sum_{i=1}^{n} \left[\log p\left(x^i, z^i \mid \theta\right)\right] + H(q) \\
&= \sum_{i=1}^{n} \mathbb{E}_{z^i \sim p\left(z^i \mid x^i\right)}\left[\log\left(\pi_{z^i}\, \mathcal{N}\left(x \middle| \mu_{z^i}, \Sigma_{z^i}\right)\right)\right] + H(q) \\
&= \sum_{i=1}^{n} \sum_{k=1}^{K} \tau_k^i \log\left(\pi_k\, \mathcal{N}\left(x \mid \mu_k, \Sigma_k\right)\right) + H(q)
\end{aligned}
$$

- Expand log of Gaussian $\log \mathcal{N}\left(x \middle| \mu_{z^i}, \Sigma_{z^i}\right)$

$$
l(\theta; D, \tau) = \sum_{i=1}^{n} \sum_{k=1}^{K} \tau_k^i \left[\log \pi_k - \frac{1}{2}\left(x^i - \mu_k\right)^{\top} \Sigma_k^{-1}\left(x^i - \mu_k\right) - \frac{1}{2}\log\left|\Sigma_k\right| - c\right] + H(q)
$$

Details of EM

- We intend to learn the parameters that maximizes the log-likelihood of the data

$$l(\theta; D) = \log \prod_{i=1}^{n} \left(\sum_{z^i=1}^{K} p\left(x^i, z^i | \theta\right) \right)$$

- **Expectation step (E-step)**: set $\tau_k^i = p\left(z^i = k \mid D, \mu, \Sigma\right)$
  - What does this step mean?
  - Find the best lower bound $l(\theta; D, q)$ to estimate the expectation $l(\theta; D)$

$$l(\theta; D) \geq l(\theta; D, q) := \mathbb{E}_{z^1, z^2, \ldots, z^n \sim q} \left[\log \prod_{i=1}^{n} p\left(x^i, z^i | \theta\right) \right] + H(q)$$

- **Maximization step (M-step)**: how to maximize$l(\theta; D, q)$?

$$\theta^{t+1} = argmax_{\theta} l(\theta; D, q)$$

M-step: Maximize $l(\theta; D, q)$

- $l(\theta; D, q) = \sum_{i=1}^{n} \sum_{k=1}^{K} \tau_k^i \left[\log \pi_k - \frac{1}{2}(x^i - \mu_k)^\top \Sigma_k^{-1}(x^i - \mu_k) - \frac{1}{2}\log|\Sigma_k| - c\right] + H(q)$

- For instance, we want to find π_k, and $\sum_{k=1}^{K} \pi_k = 1$
  - Form Lagrangian

$$L = \sum_{i=1}^{n} \sum_{k=1}^{K} \tau_k^i [\log \pi_k - \text{other terms}] + \lambda \left(1 - \sum_{i=1}^{K} \pi_k\right)$$

  - Take partial derivative and set to 0

$$\frac{\partial L}{\partial \pi_k} = \left(\sum_{i=1}^{n} \frac{\tau_k^i}{\pi_k}\right) - \lambda = 0, \qquad \Rightarrow \pi_k = \frac{1}{\lambda} \sum_{i=1}^{n} \tau_k^i, \qquad \Rightarrow \lambda = n$$

EM (Expectation-Maximization) Algorithm

- Associate each data and each component with a τ_k^i
- Initialized $(\pi_k, \mu_k, \Sigma_k), k = 1,2, \dots, K$
- Iterate the following two steps until convergence:
  - **Expectation step (E-step)**: update τ_k^i given the current (π_k, μ_k, Σ_k)

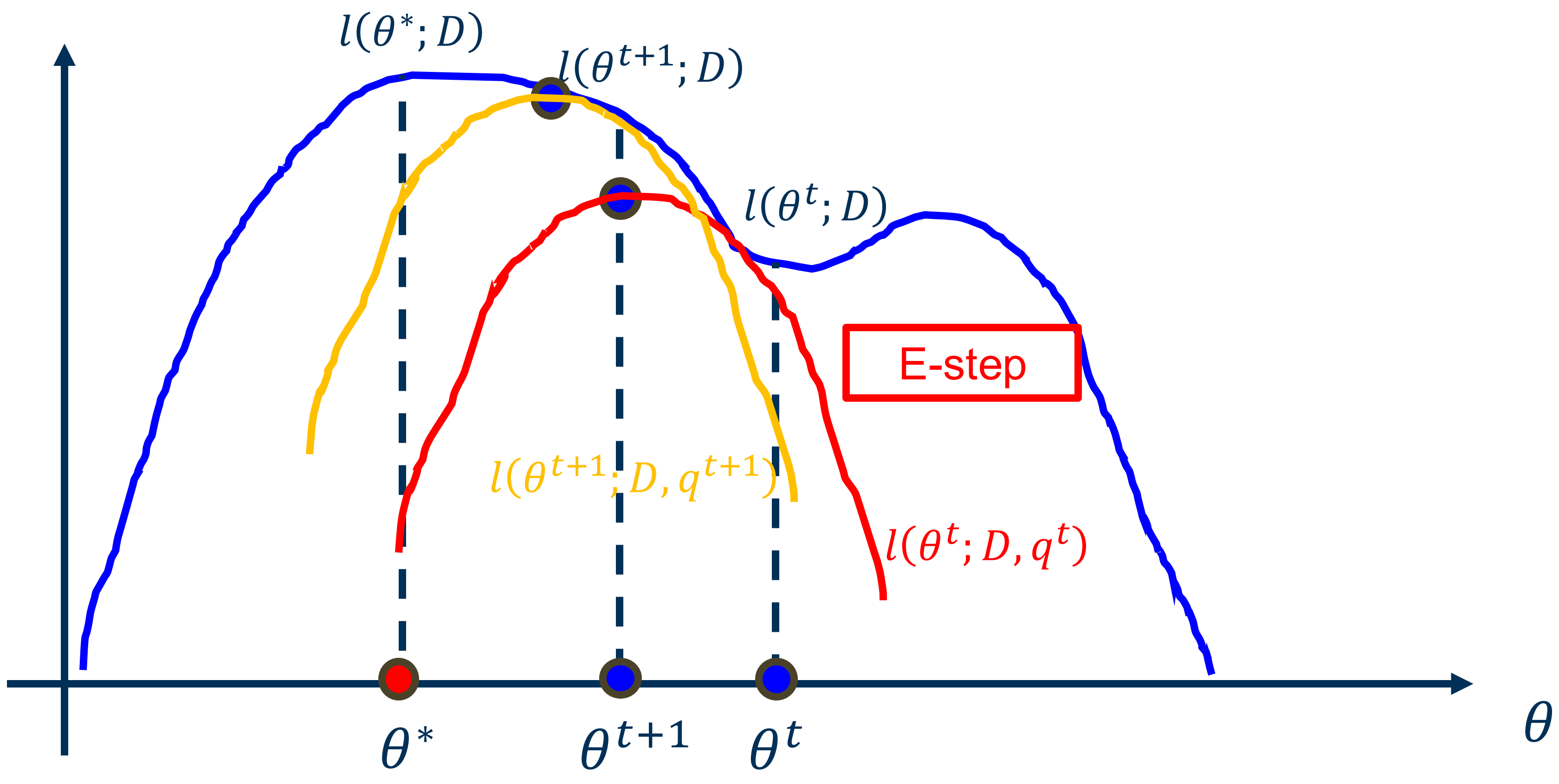
$$\tau_k^i = p(z^i = k \mid D, \mu, \Sigma) = \frac{\pi_k \, \mathcal{N}(x|\mu_k, \Sigma_k)}{\sum_{k'=1}^{K} \pi_{k'} \, \mathcal{N}(x|\mu_{k'}, \Sigma_{k'})}, \qquad \forall k \in \{1,2, \dots, K\}; i \in \{1,2, \dots, n\}$$

  - **Maximization step (M-step)**: update (π_k, μ_k, Σ_k) given τ_k^i

$$\pi_k = \frac{\sum_i \tau_k^i}{n}, \qquad \mu_k = \frac{\sum_i \tau_k^i x^i}{\sum_i \tau_k^i}$$

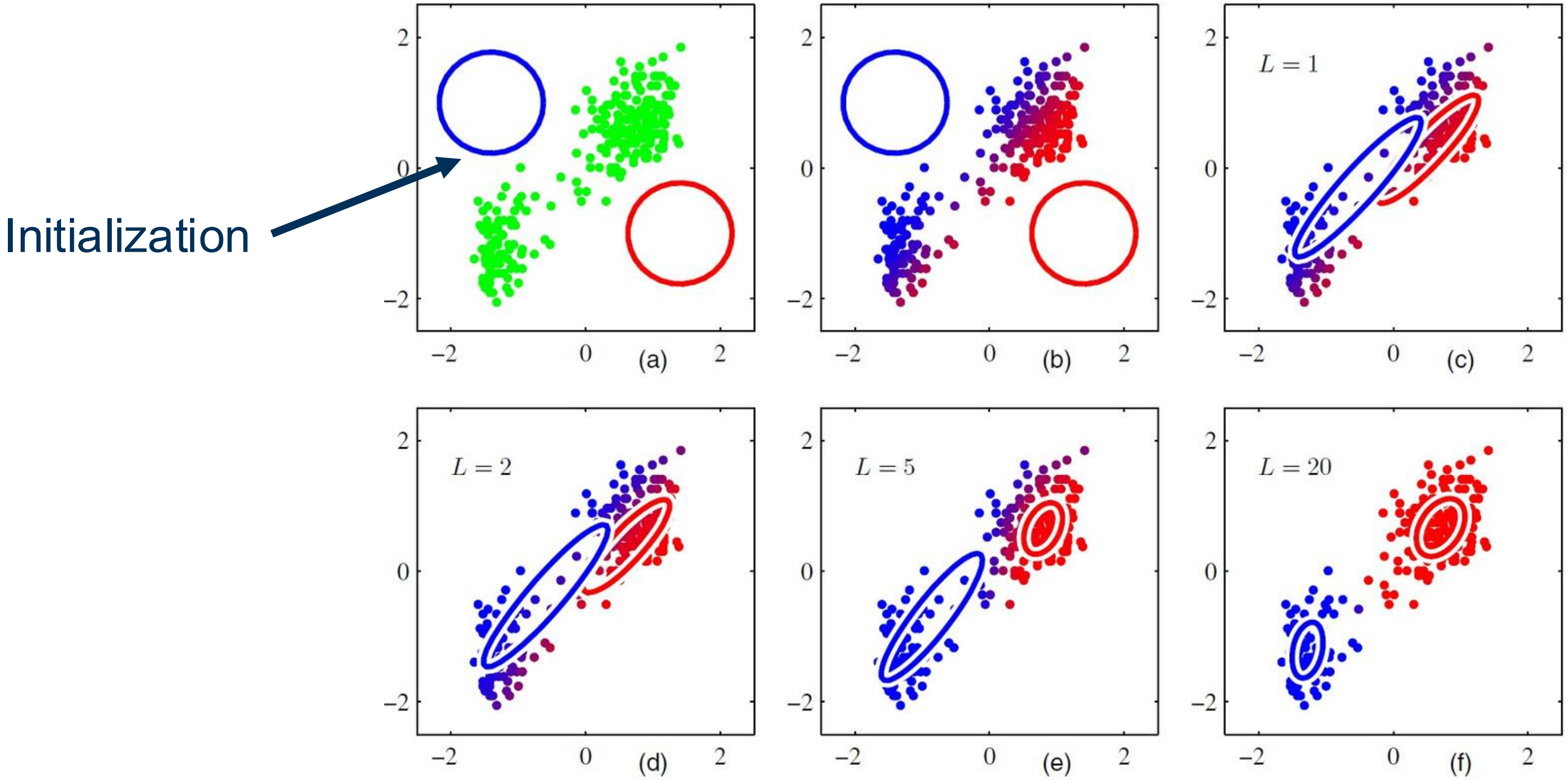
$$\Sigma_k = \frac{\sum_i \tau_k^i (x^i - \mu_k)(x^i - \mu_k)^\top}{\sum_i \tau_k^i}, \qquad \forall k \in \{1,2, \dots, K\}$$

EM Graphically



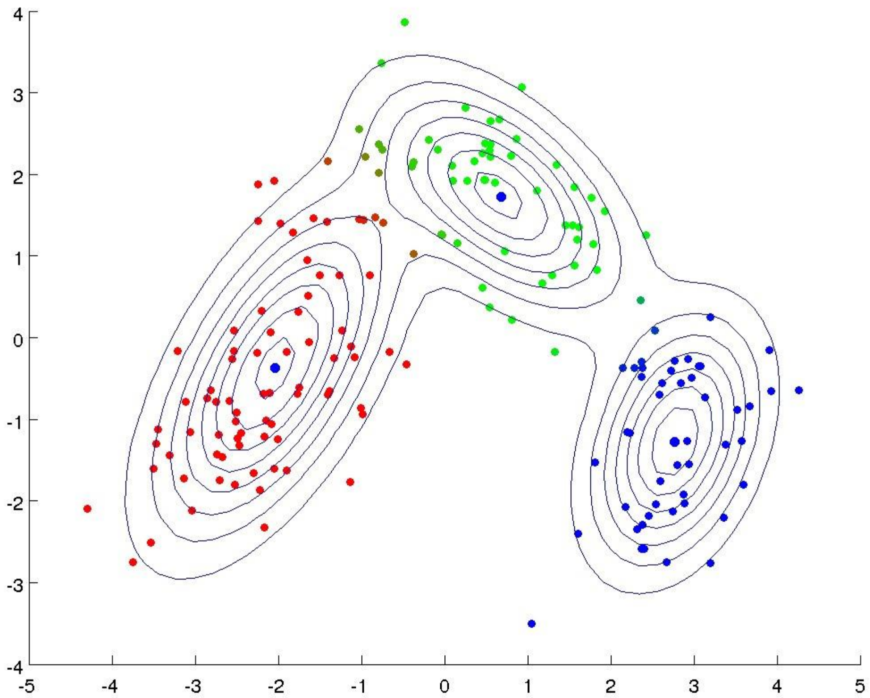
Expectation-Maximization Iterations

- $k = 1$ or 2
- Use τ_1^i as the proportion of red, and τ_2^i as the proportion of blue
- Draw only one contour for each Gaussian component



Mixture of 3 Gaussians

- First run PCA to reduce the dimension to 2
- $k = 1$ or 2 or 3
- Use τ_1^i as the proportion of red, τ_2^i as the proportion of blue, and τ_3^i as the proportion of green



EM v.s. Modified K-means

- The EM algorithm for mixture of Gaussian is like of soft clustering algorithm

- K-means:
  - ”E-step”: we do hard assignment
    - $z^i = argmax_k \left(x^i - \mu_k\right)^\top \Sigma_k^{-1}\left(x^i - \mu_k\right)$
  - “M-step”: we update the means and covariance of cluster suing maximum likelihood estimate:
    - $\mu_k = \frac{\sum_i \tau_k^i x^i}{\sum_i \tau_k^i}$
    - $\Sigma_k = \frac{\sum_i \tau_k^i (x^i - \mu_k)(x^i - \mu_k)^\top}{\sum_i \tau_k^i}$
    - where $\tau_k^i = 1$ if $z^i = k$; otherwise 0

General Applicability of EM Algorithm

- Applicable to other models with latent (or missing) variables

- Expectation maximization applied to a coin toss example (python example)
  - Assume you have a sequence of coin flip observations from two coins, but you don’t know from which coin each of the observations is from

  - The EM algorithm starts by initializing a random prior

  - Then it calculates the expected log probability distribution over the observations, and based on the log probability updates the prior

```
In [1]:
# N.B. each coin label in `labels` corresponds to the sequence

labels = ['B', 'A', 'A', 'B', 'A']

flip_seqs = ['HTHHHTTHHHHTHHHTTHHHHHTHHHHH',
             'HHTHHHHHTTTTTTTHHTT',
             'HTHHHHTTHHTTTTTTTHTTTTHHTTT',
             'HTHTTHHTTHHHHHTHHHHHHHTHHHHHTHHHHHHTTHHHHH',
             'THHHTHHHTHTTTTTTTTTT']
```

```
In [12]: weight_A

Out[12]: [0.9993617897653697,
          0.04041543659201761,
          0.0001453833718729461,
          0.9999258022267 5,
          0.0007623605427559703]

In [13]: weight_B

Out[13]: [0.0006382102346302874,
          0.9595845634079825,
          0.9998546166281271,
          7.419777324936436e-06,
          0.9992376394572 44]
```

Next Week (Sep 8th) Preview

- We intend to learn the parameters that maximizes the log-likelihood of the data

$$l(\theta; D) = \log \prod_{i=1}^{n} \left(\sum_{z^i=1}^{K} p(x^i, z^i | \theta) \right)$$

- **Expectation step (E-step)**: what do we take expectation over?

$$\boxed{l(\theta; D) \geq l(\theta; D, q)} := \mathbb{E}_{z^1, z^2, \ldots, z^n \sim q} \left[\log \prod_{i=1}^{n} p(x^i, z^i | \theta) \right] + H(q)$$

Why?

- **Maximization step (M-step)**: how to maximize?

$$\theta^{t+1} = argmax_\theta \, f(\theta)$$