Lecture 16: 24 March, 2022

Madhavan Mukund

https://www.cmi.ac.in/~madhavan

Data Mining and Machine Learning
January–May 2022

Mixture models

- Probabilistic process — parameters Θ
  - Tossing a coin with $\Theta = \{Pr(H)\} = \{p\}$

Mixture models

- Probabilistic process — parameters Θ
  - Tossing a coin with $\Theta = \{Pr(H)\} = \{p\}$
- Perform an experiment
  - Toss the coin N times, $H\ T\ H\ H\ \cdots\ T$

Mixture models

- Probabilistic process — parameters Θ
  - Tossing a coin with $\Theta = \{Pr(H)\} = \{p\}$
- Perform an experiment
  - Toss the coin N times, $H\ T\ H\ H\ \cdots\ T$
- Estimate parameters from observations
  - From h heads, estimate $p = h/N$
  - Maximum Likelihood Estimator (MLE)

Mixture models

- Probabilistic process — parameters Θ
  - Tossing a coin with $\Theta = \{Pr(H)\} = \{p\}$
- Perform an experiment
  - Toss the coin N times, $H\ T\ H\ H\ \cdots\ T$
- Estimate parameters from observations
  - From h heads, estimate $p = h/N$
  - Maximum Likelihood Estimator (MLE)
- What if we have a mixture of two random processes

Mixture models

- Probabilistic process — parameters Θ
  - Tossing a coin with $\Theta = \{Pr(H)\} = \{p\}$
- Perform an experiment
  - Toss the coin N times, $H\ T\ H\ H\ \cdots\ T$
- Estimate parameters from observations
  - From h heads, estimate $p = h/N$
  - Maximum Likelihood Estimator (MLE)
- What if we have a mixture of two random processes
  - Two coins, c_1 and c_2, with $Pr(H) = p_1$ and p_2, respectively

Mixture models

- Probabilistic process — parameters Θ
  - Tossing a coin with $\Theta = \{Pr(H)\} = \{p\}$
- Perform an experiment
  - Toss the coin N times, $H\ T\ H\ H\ \cdots\ T$
- Estimate parameters from observations
  - From h heads, estimate $p = h/N$
  - Maximum Likelihood Estimator (MLE)
- What if we have a mixture of two random processes
  - Two coins, c_1 and c_2, with $Pr(H) = p_1$ and p_2, respectively
  - Repeat N times: choose c_i with probability $1/2$ and toss it

Mixture models

- Probabilistic process — parameters Θ
  - Tossing a coin with $\Theta = \{Pr(H)\} = \{p\}$
- Perform an experiment
  - Toss the coin N times, $H\ T\ H\ H\ \cdots\ T$
- Estimate parameters from observations
  - From h heads, estimate $p = h/N$
  - Maximum Likelihood Estimator (MLE)
- What if we have a mixture of two random processes
  - Two coins, c_1 and c_2, with $Pr(H) = p_1$ and p_2, respectively
  - Repeat N times: choose c_i with probability $1/2$ and toss it
  - Outcome: N_1 tosses of c_1 interleaved with N_2 tosses of c_2, $N_1 + N_2 = N$

Mixture models

- Probabilistic process — parameters Θ
  - Tossing a coin with $\Theta = \{Pr(H)\} = \{p\}$
- Perform an experiment
  - Toss the coin N times, $H\ T\ H\ H\ \cdots\ T$
- Estimate parameters from observations
  - From h heads, estimate $p = h/N$
  - Maximum Likelihood Estimator (MLE)
- What if we have a mixture of two random processes
  - Two coins, c_1 and c_2, with $Pr(H) = p_1$ and p_2, respectively
  - Repeat N times: choose c_i with probability $1/2$ and toss it
  - Outcome: N_1 tosses of c_1 interleaved with N_2 tosses of c_2, $N_1 + N_2 = N$
  - Can we estimate p_1 and p_2?

Mixture models . . .

- Two coins, c_1 and c_2, with $Pr(H) = p_1$ and p_2, respectively
- Sequence of N interleaved coin tosses $H\ T\ H\ H\ \cdots\ H\ H\ T$

Mixture models . . .

- Two coins, c_1 and c_2, with $Pr(H) = p_1$ and p_2, respectively
- Sequence of N interleaved coin tosses $H\ T\ H\ H\ \cdots\ H\ H\ T$
- If the sequence is labelled, we can estimate p_1, p_2 separately
  - $H\ T\ T\ H\ H\ T\ H\ T\ H\ H\ T\ H\ T\ H\ T\ H\ H\ T\ H\ T$
  - $p_1 = 8/12 = 2/3$, $p_2 = 3/8$

Mixture models . . .

- Two coins, c_1 and c_2, with $Pr(H) = p_1$ and p_2, respectively
- Sequence of N interleaved coin tosses $H\ T\ H\ H\ \cdots\ H\ H\ T$
- If the sequence is labelled, we can estimate p_1, p_2 separately
  - $H\ T\ T\ H\ H\ T\ H\ T\ H\ H\ T\ H\ T\ H\ T\ H\ H\ T\ H\ T$
  - $p_1 = 8/12 = 2/3$, $p_2 = 3/8$
- What the observation is unlabelled?
  - $H\ T\ T\ H\ H\ T\ H\ T\ H\ H\ T\ H\ T\ H\ T\ H\ H\ T\ H\ T$

Mixture models . . .

- Two coins, c_1 and c_2, with $Pr(H) = p_1$ and p_2, respectively
- Sequence of N interleaved coin tosses $H\ T\ H\ H\ \cdots\ H\ H\ T$
- If the sequence is labelled, we can estimate p_1, p_2 separately
  - $H\ T\ T\ H\ H\ T\ H\ T\ H\ H\ T\ H\ T\ H\ T\ H\ H\ T\ H\ T$
  - $p_1 = 8/12 = 2/3$, $p_2 = 3/8$
- What the observation is unlabelled?
  - $H\ T\ T\ H\ H\ T\ H\ T\ H\ H\ T\ H\ T\ H\ T\ H\ H\ T\ H\ T$
- Iterative algorithm to estimate the parameters
  - Make an initial guess for the parameters
  - Compute a (fractional) labelling of the outcomes
  - Re-estimate the parameters

Expectation Maximization (EM)

- Iterative algorithm to estimate the parameters
  - Make an initial guess for the parameters
  - Compute a (fractional) labelling of the outcomes
  - Re-estimate the parameters

Expectation Maximization (EM)

- Iterative algorithm to estimate the parameters
  - Make an initial guess for the parameters
  - Compute a (fractional) labelling of the outcomes
  - Re-estimate the parameters
- *H T T H H T H T H H T H T H T H H T H T*
  - Initial guess: $p_1 = 1/2$, $p_2 = 1/4$

P(H in c_1)? $\frac{1}{2}$

P(H in c_2)? $\frac{1}{4}$

$\frac{\text{count heads}}{\text{total}}$

divide: c_1 count, c_2 count, c_1 total, c_2 total

$2 : 1$

$\frac{\frac{1}{2}}{\frac{1}{2}+\frac{1}{4}} = \frac{2}{3}$ $\quad$ $\frac{\frac{1}{4}}{\frac{1}{2}+\frac{1}{4}} = \frac{1}{3}$

Expectation Maximization (EM)

- Iterative algorithm to estimate the parameters
  - Make an initial guess for the parameters
  - Compute a (fractional) labelling of the outcomes
  - Re-estimate the parameters
- $H\ T\ T\ H\ H\ T\ H\ T\ H\ H\ T\ H\ T\ H\ T\ H\ H\ T\ H\ T$
  - Initial guess: $p_1 = 1/2$, $p_2 = 1/4$
  - $Pr(c_1 = T) = q_1 = 1/2$, $Pr(c_2 = T) = q_2 = 3/4$,

Heads split as

2 : 1

$\frac{2}{3}$ $\frac{1}{3}$

$\frac{2}{5}$ — $\frac{3}{5}$

Tail

$\frac{2}{4}$ · $\frac{3}{4}$

$\frac{1/2}{1/2+3/4}$ $\frac{3/4}{1/2+3/4}$

Expectation Maximization (EM)

- Iterative algorithm to estimate the parameters
  - Make an initial guess for the parameters
  - Compute a (fractional) labelling of the outcomes
  - Re-estimate the parameters
- $H\ T\ T\ H\ H\ T\ H\ T\ H\ H\ T\ H\ T\ H\ T\ H\ H\ T\ H\ T$
  - Initial guess: $p_1 = 1/2$, $p_2 = 1/4$
  - $Pr(c_1 = T) = q_1 = 1/2$, $Pr(c_2 = T) = q_2 = 3/4$,
  - For each H, likelihood it was c_i, $Pr(c_i \mid H)$, is $p_i/(p_1 + p_2)$

Expectation Maximization (EM)

- Iterative algorithm to estimate the parameters
  - Make an initial guess for the parameters
  - Compute a (fractional) labelling of the outcomes
  - Re-estimate the parameters
- $H\ T\ T\ H\ H\ T\ H\ T\ H\ H\ T\ H\ T\ H\ T\ H\ H\ T\ H\ T$
  - Initial guess: $p_1 = 1/2$, $p_2 = 1/4$
  - $Pr(c_1 = T) = q_1 = 1/2$, $Pr(c_2 = T) = q_2 = 3/4$,
  - For each H, likelihood it was c_i, $Pr(c_i \mid H)$, is $p_i/(p_1 + p_2)$
  - For each T, likelihood it was c_i, $Pr(c_i \mid T)$, is $q_i/(q_1 + q_2)$

Expectation Maximization (EM)

- Iterative algorithm to estimate the parameters
  - Make an initial guess for the parameters
  - Compute a (fractional) labelling of the outcomes
  - Re-estimate the parameters
- $H\ T\ T\ H\ H\ T\ H\ T\ H\ H\ T\ H\ T\ H\ T\ H\ H\ T\ H\ T$
  - Initial guess: $p_1 = 1/2$, $p_2 = 1/4$
  - $Pr(c_1 = T) = q_1 = 1/2$, $Pr(c_2 = T) = q_2 = 3/4$,
  - For each H, likelihood it was c_i, $Pr(c_i \mid H)$, is $p_i/(p_1 + p_2)$
  - For each T, likelihood it was c_i, $Pr(c_i \mid T)$, is $q_i/(q_1 + q_2)$
  - Assign fractional count $Pr(c_i \mid H)$ to each H: $2/3 \times c_1$, $1/3 \times c_2$

Expectation Maximization (EM)

- Iterative algorithm to estimate the parameters
  - Make an initial guess for the parameters
  - Compute a (fractional) labelling of the outcomes
  - Re-estimate the parameters
- $H\ T\ T\ H\ H\ T\ H\ T\ H\ H\ T\ H\ T\ H\ T\ H\ H\ T\ H\ T$
  - Initial guess: $p_1 = 1/2$, $p_2 = 1/4$
  - $Pr(c_1 = T) = q_1 = 1/2$, $Pr(c_2 = T) = q_2 = 3/4$,
  - For each H, likelihood it was c_i, $Pr(c_i \mid H)$, is $p_i/(p_1 + p_2)$
  - For each T, likelihood it was c_i, $Pr(c_i \mid T)$, is $q_i/(q_1 + q_2)$
  - Assign fractional count $Pr(c_i \mid H)$ to each H: $2/3 \times c_1$, $1/3 \times c_2$
  - Likewise, assign fractional count $Pr(c_i \mid T)$ to each T: $2/5 \times c_1$, $3/5 \times c_2$

Expectation Maximization (EM)

- $H\ T\ T\ H\ H\ T\ H\ T\ H\ H\ T\ H\ T\ H\ T\ H\ H\ T\ H\ T$
- Initial guess: $p_1 = 1/2$, $p_2 = 1/4$
- Fractional counts: each H is $2/3 \times c_1$, $1/3 \times c_2$, each T: $2/5 \times c_1$, $3/5 \times c_2$

Expectation Maximization (EM)

- $H\ T\ T\ H\ H\ T\ H\ T\ H\ H\ T\ H\ T\ H\ T\ H\ H\ T\ H\ T$
- Initial guess: $p_1 = 1/2$, $p_2 = 1/4$
- Fractional counts: each H is $2/3 \times c_1$, $1/3 \times c_2$, each T: $2/5 \times c_1$, $3/5 \times c_2$
- Add up the fractional counts
  - c_1: $11 \cdot (2/3) = 22/3$ heads, $9 \cdot (2/5) = 18/5$ tails
  - c_2: $11 \cdot (1/3) = 11/3$ heads, $9 \cdot (3/5) = 27/5$ tails

11 H
9 T ⇒ 20

Expectation Maximization (EM)

- $H\ T\ T\ H\ H\ T\ H\ T\ H\ H\ T\ H\ T\ H\ T\ H\ H\ T\ H\ T$
- Initial guess: $p_1 = 1/2$, $p_2 = 1/4$
- Fractional counts: each H is $2/3 \times c_1$, $1/3 \times c_2$, each T: $2/5 \times c_1$, $3/5 \times c_2$
- Add up the fractional counts
  - c_1: $11 \cdot (2/3) = 22/3$ heads, $9 \cdot (2/5) = 18/5$ tails
  - c_2: $11 \cdot (1/3) = 11/3$ heads, $9 \cdot (3/5) = 27/5$ tails
- Re-estimate the parameters
  - $p_1 = \dfrac{22/3}{22/3 + 18/5} = 110/164 = 0.67$, $q_1 = 1 - p_1 = 0.33$
  - $p_2 = \dfrac{11/3}{11/3 + 27/5} = 55/136 = 0.40$, $q_2 = 1 - p_2 = 0.60$

Expectation Maximization (EM)

- $H\ T\ T\ H\ H\ T\ H\ T\ H\ H\ T\ H\ T\ H\ T\ H\ H\ T\ H\ T$
- Initial guess: $p_1 = 1/2$, $p_2 = 1/4$
- Fractional counts: each H is $2/3 \times c_1$, $1/3 \times c_2$, each T: $2/5 \times c_1$, $3/5 \times c_2$
- Add up the fractional counts
  - c_1: $11 \cdot (2/3) = 22/3$ heads, $9 \cdot (2/5) = 18/5$ tails
  - c_2: $11 \cdot (1/3) = 11/3$ heads, $9 \cdot (3/5) = 27/5$ tails
- Re-estimate the parameters
  - $p_1 = \dfrac{22/3}{22/3 + 18/5} = 110/164 = 0.67$, $q_1 = 1 - p_1 = 0.33$
  - $p_2 = \dfrac{11/3}{11/3 + 27/5} = 55/136 = 0.40$, $q_2 = 1 - p_2 = 0.60$
- Repeat until convergence

Expectation Maximization (EM)

- Mixture of probabilistic models $(M_1, M_2, \ldots, M_k)$ with parameters $\Theta = (\theta_1, \theta_2, \ldots, \theta_k)$

Expectation Maximization (EM)

- Mixture of probabilistic models $(M_1, M_2, \ldots, M_k)$ with parameters $\Theta = (\theta_1, \theta_2, \ldots, \theta_k)$
- Observation $O = o_1 o_2 \ldots o_N$

Expectation Maximization (EM)

- Mixture of probabilistic models $(M_1, M_2, \ldots, M_k)$ with parameters $\Theta = (\theta_1, \theta_2, \ldots, \theta_k)$
- Observation $O = o_1 o_2 \ldots o_N$
- Expectation step
  - Compute likelihoods $Pr(M_i|o_j)$ for each M_i, o_j

Expectation Maximization (EM)

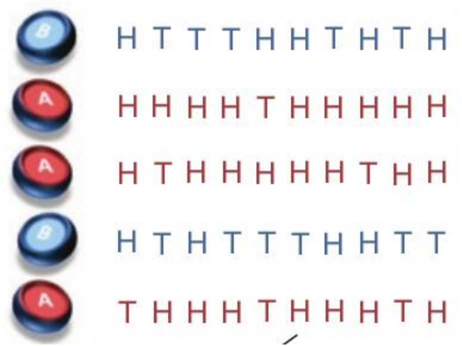
- Mixture of probabilistic models $(M_1, M_2, \ldots, M_k)$ with parameters $\Theta = (\theta_1, \theta_2, \ldots, \theta_k)$
- Observation $O = o_1 o_2 \ldots o_N$
- Expectation step
  - Compute likelihoods $Pr(M_i | o_j)$ for each M_i, o_j
- Maximization step
  - Recompute MLE for each M_i using fraction of O assigned using likelihood

Expectation Maximization (EM)

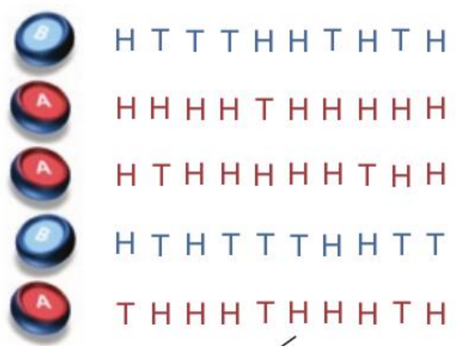
- Mixture of probabilistic models $(M_1, M_2, \ldots, M_k)$ with parameters $\Theta = (\theta_1, \theta_2, \ldots, \theta_k)$
- Observation $O = o_1 o_2 \ldots o_N$
- Expectation step
  - Compute likelihoods $Pr(M_i | o_j)$ for each M_i, o_j
- Maximization step
  - Recompute MLE for each M_i using fraction of O assigned using likelihood
- Repeat until convergence
  - Why should it converge?
  - If the value converges, what have we computed?

EM — another example

- Two biased coins, choose a coin and toss 10 times, repeat 5 times



- Two biased coins, choose a coin and toss 10 times, repeat 5 times



- If we know the breakup, we can separately compute MLE for each coin

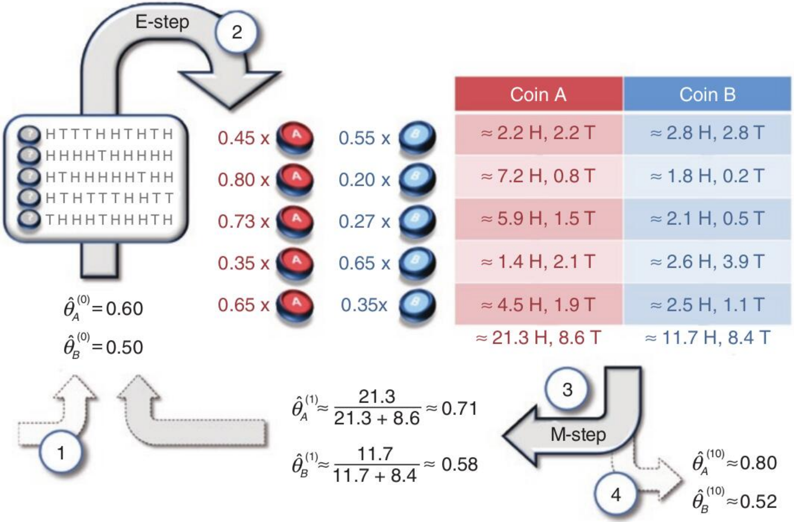
Coin A	Coin B
	5 H, 5 T
9 H, 1 T	
8 H, 2 T	
	4 H, 6 T
7 H, 3 T	
24 H, 6 T	9 H, 11 T

$$\hat{\theta}_A = \frac{24}{24+6} = 0.80$$

$$\hat{\theta}_B = \frac{9}{9+11} = 0.45$$

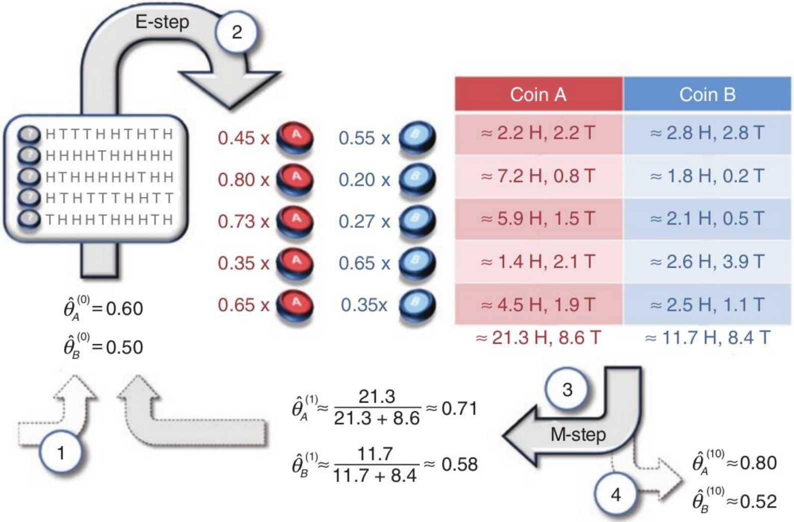
EM — another example

- Expectation-Maximization



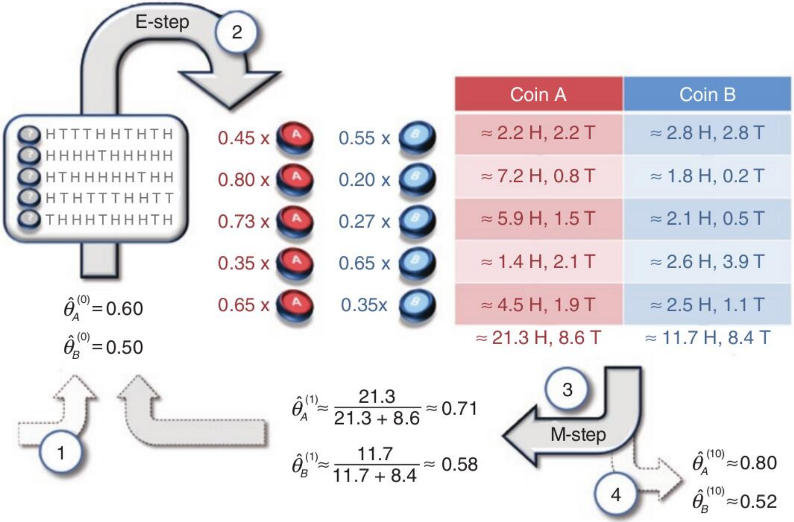
EM — another example

- Expectation-Maximization
- Initial estimates, $\theta_A = 0.6$, $\theta_B = 0.5$



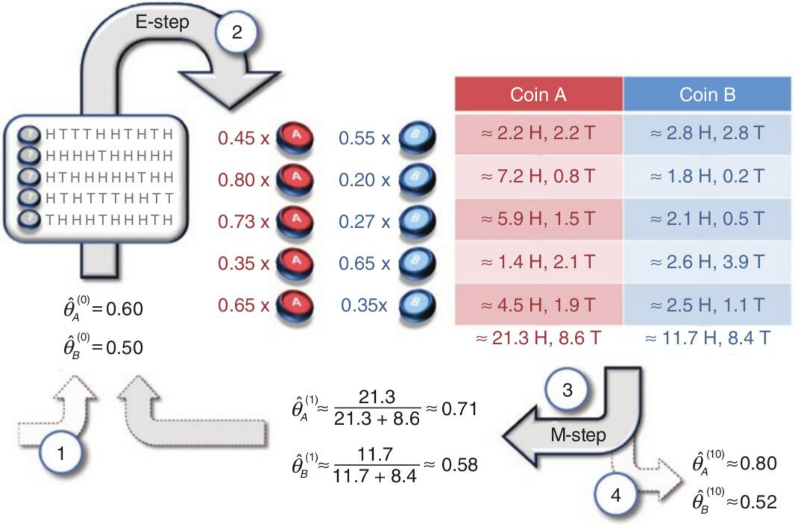
EM — another example

- Expectation-Maximization
- Initial estimates, $\theta_A = 0.6$, $\theta_B = 0.5$
- Compute likelihood of each sequence: $\theta^{n_H}(1-\theta)^{n_T}$



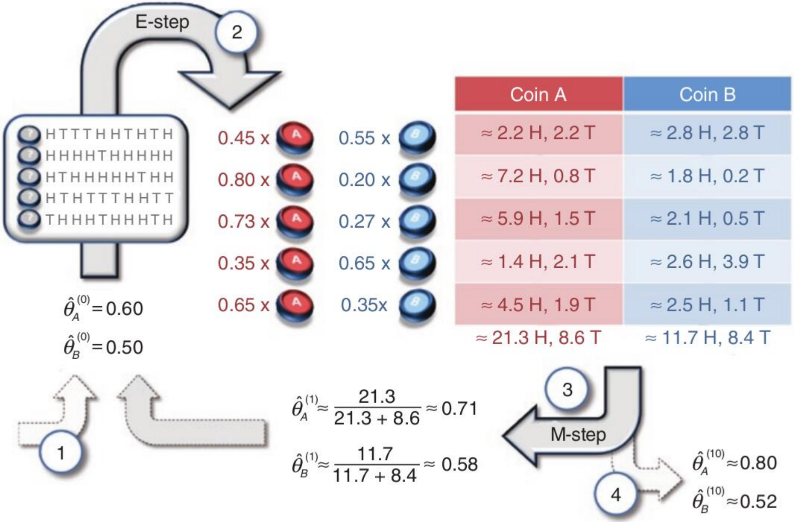
EM — another example

- Expectation-Maximization
- Initial estimates, $\theta_A = 0.6$, $\theta_B = 0.5$
- Compute likelihood of each sequence: $\theta^{n_H}(1-\theta)^{n_T}$
- Assign each sequence proportionately



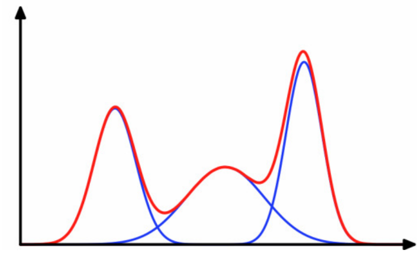
EM — another example

- Expectation-Maximization
- Initial estimates, $\theta_A = 0.6$, $\theta_B = 0.5$
- Compute likelihood of each sequence: $\theta^{n_H}(1-\theta)^{n_T}$
- Assign each sequence proportionately
- Converge to $\theta_A = 0.8$, $\theta_B = 0.52$



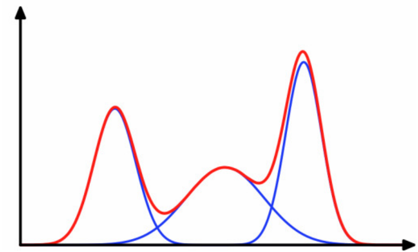
EM — mixture of Gaussians

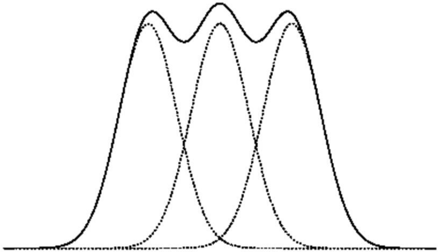
- Sample uniformly from multiple Gaussians, $\mathcal{N}(\mu_i, \sigma_i)$



EM — mixture of Gaussians

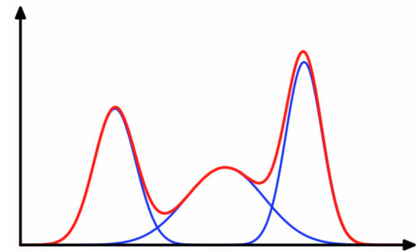
- Sample uniformly from multiple Gaussians, $\mathcal{N}(\mu_i, \sigma_i)$
- For simplicity, assume all $\sigma_i = \sigma$

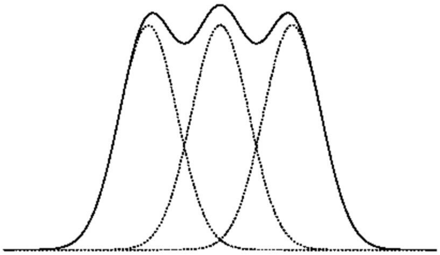




EM — mixture of Gaussians

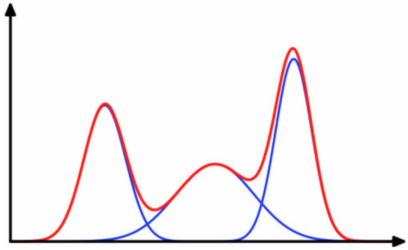
- Sample uniformly from multiple Gaussians, $\mathcal{N}(\mu_i, \sigma_i)$
- For simplicity, assume all $\sigma_i = \sigma$
- N sample points $z_1, z_2, \ldots, z_N$

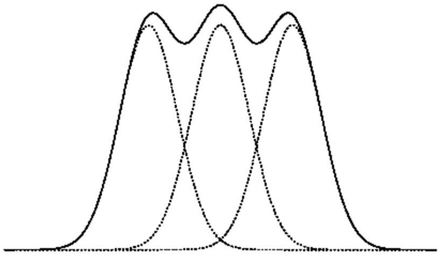




EM — mixture of Gaussians

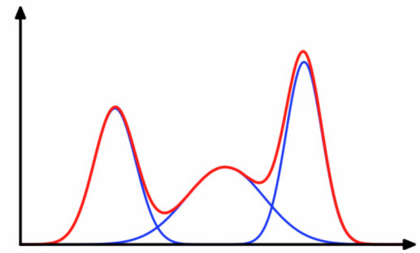
- Sample uniformly from multiple Gaussians, $\mathcal{N}(\mu_i, \sigma_i)$
- For simplicity, assume all $\sigma_i = \sigma$
- N sample points $z_1, z_2, \ldots, z_N$
- Make an initial guess for each μ_j

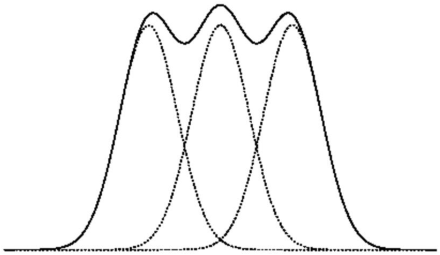




EM — mixture of Gaussians

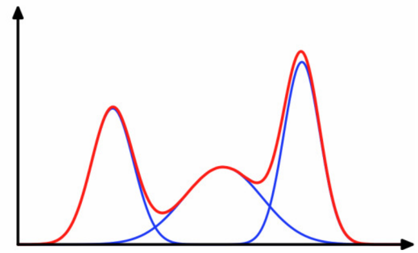
- Sample uniformly from multiple Gaussians, $\mathcal{N}(\mu_i, \sigma_i)$
- For simplicity, assume all $\sigma_i = \sigma$
- N sample points $z_1, z_2, \ldots, z_N$
- Make an initial guess for each μ_j
- $Pr(z_i \mid \mu_j) = exp(-\frac{1}{2\sigma^2}(z_i - \mu_j)^2)$

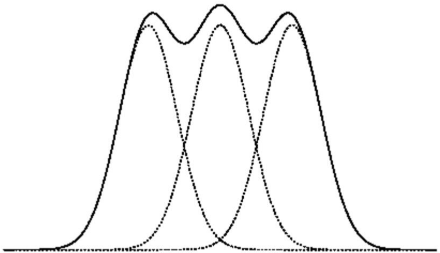




EM — mixture of Gaussians

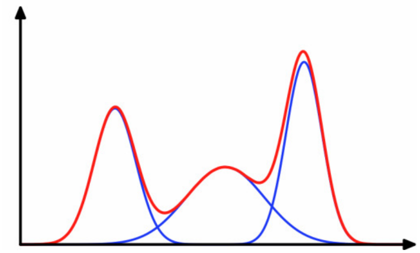
- Sample uniformly from multiple Gaussians, $\mathcal{N}(\mu_i, \sigma_i)$
- For simplicity, assume all $\sigma_i = \sigma$
- N sample points $z_1, z_2, \ldots, z_N$
- Make an initial guess for each μ_j
- $Pr(z_i \mid \mu_j) = exp(-\frac{1}{2\sigma^2}(z_i - \mu_j)^2)$
- $Pr(\mu_j \mid z_i) = c_{ij} = \dfrac{Pr(z_i \mid \mu_j)}{\sum_k Pr(z_i \mid \mu_k)}$

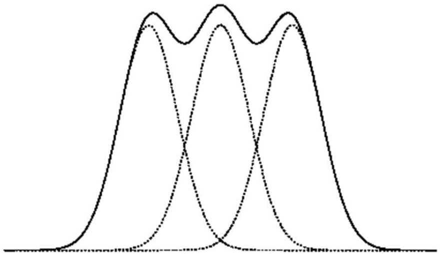




EM — mixture of Gaussians

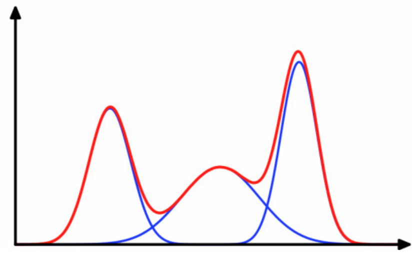
- Sample uniformly from multiple Gaussians, $\mathcal{N}(\mu_i, \sigma_i)$
- For simplicity, assume all $\sigma_i = \sigma$
- N sample points $z_1, z_2, \ldots, z_N$
- Make an initial guess for each μ_j
- $Pr(z_i \mid \mu_j) = exp(-\frac{1}{2\sigma^2}(z_i - \mu_j)^2)$
- $Pr(\mu_j \mid z_i) = c_{ij} = \dfrac{Pr(z_i \mid \mu_j)}{\sum_k Pr(z_i \mid \mu_k)}$
- MLE of μ_j is sample mean, $\dfrac{\sum_i c_{ij} z_i}{\sum_i c_{ij}}$

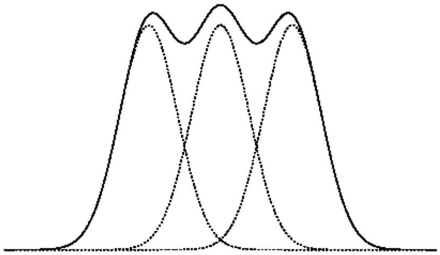




EM — mixture of Gaussians

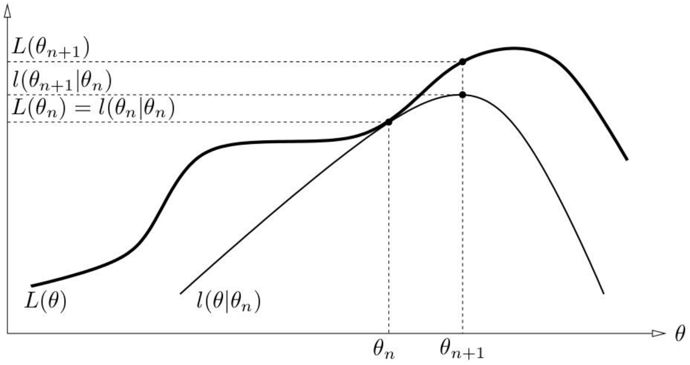
- Sample uniformly from multiple Gaussians, $\mathcal{N}(\mu_i, \sigma_i)$
- For simplicity, assume all $\sigma_i = \sigma$
- N sample points $z_1, z_2, \ldots, z_N$
- Make an initial guess for each μ_j
- $Pr(z_i \mid \mu_j) = exp(-\frac{1}{2\sigma^2}(z_i - \mu_j)^2)$
- $Pr(\mu_j \mid z_i) = c_{ij} = \dfrac{Pr(z_i \mid \mu_j)}{\sum_k Pr(z_i \mid \mu_k)}$
- MLE of μ_j is sample mean, $\dfrac{\sum_i c_{ij} z_i}{\sum_i c_{ij}}$
- Update estimates for μ_j and repeat





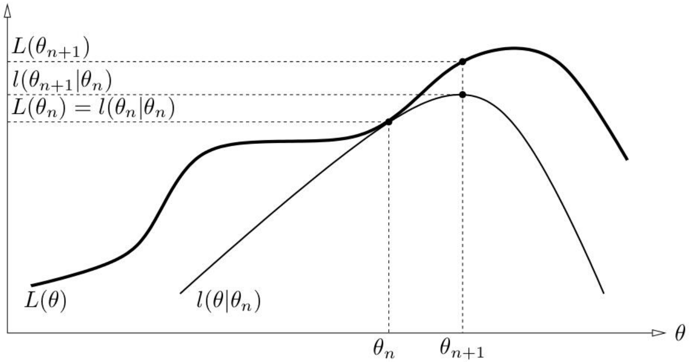
Theoretical foundations of EM

- Mixture of probabilistic models $(M_1, M_2, \ldots, M_k)$ with parameters $\Theta = (\theta_1, \theta_2, \ldots, \theta_k)$



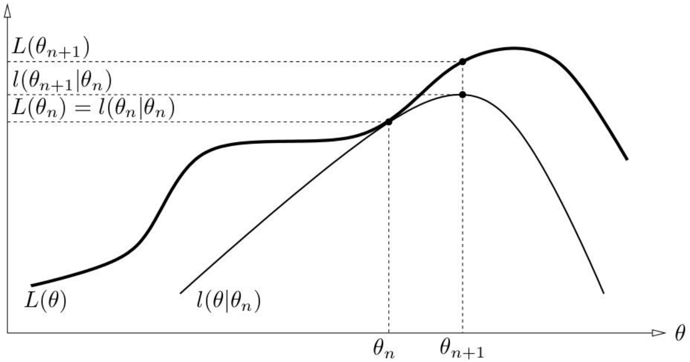
Theoretical foundations of EM

- Mixture of probabilistic models $(M_1, M_2, \ldots, M_k)$ with parameters $\Theta = (\theta_1, \theta_2, \ldots, \theta_k)$
- Observation $O = o_1 o_2 \ldots o_N$



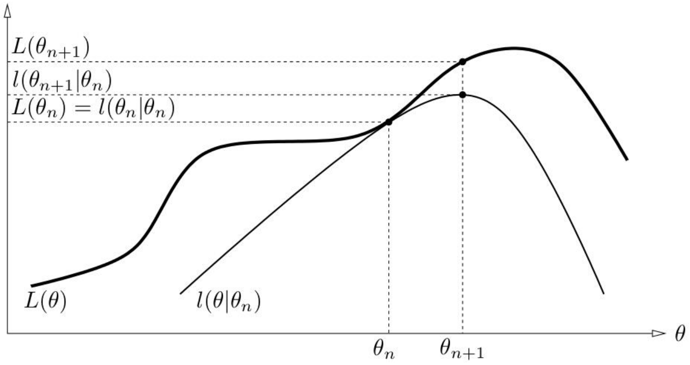
Theoretical foundations of EM

- Mixture of probabilistic models $(M_1, M_2, \ldots, M_k)$ with parameters $\Theta = (\theta_1, \theta_2, \ldots, \theta_k)$
- Observation $O = o_1 o_2 \ldots o_N$
- EM builds a sequence of estimates $\Theta_1, \Theta_2, \ldots, \Theta_n$



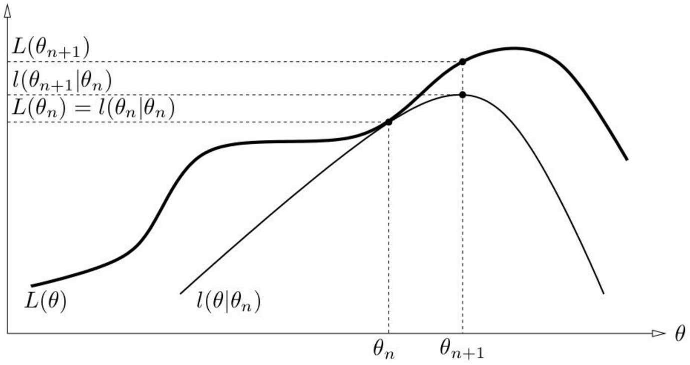
Theoretical foundations of EM

- Mixture of probabilistic models $(M_1, M_2, \ldots, M_k)$ with parameters $\Theta = (\theta_1, \theta_2, \ldots, \theta_k)$
- Observation $O = o_1 o_2 \ldots o_N$
- EM builds a sequence of estimates $\Theta_1, \Theta_2, \ldots, \Theta_n$
- $L(\Theta_j)$ — log-likelihood function, $\ln Pr(O \mid \Theta_j)$



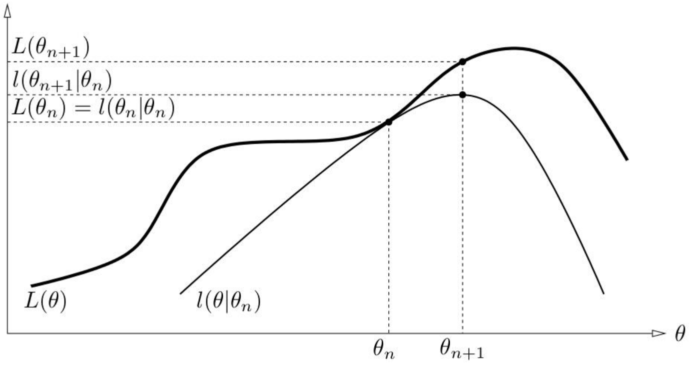
Theoretical foundations of EM

- Mixture of probabilistic models $(M_1, M_2, \ldots, M_k)$ with parameters $\Theta = (\theta_1, \theta_2, \ldots, \theta_k)$
- Observation $O = o_1 o_2 \ldots o_N$
- EM builds a sequence of estimates $\Theta_1, \Theta_2, \ldots, \Theta_n$
- $L(\Theta_j)$ — log-likelihood function, $\ln Pr(O \mid \Theta_j)$
- Want to extend the sequence with Θ_{n+1} such that $L(\Theta_{n+1}) > L(\Theta_n)$



Theoretical foundations of EM

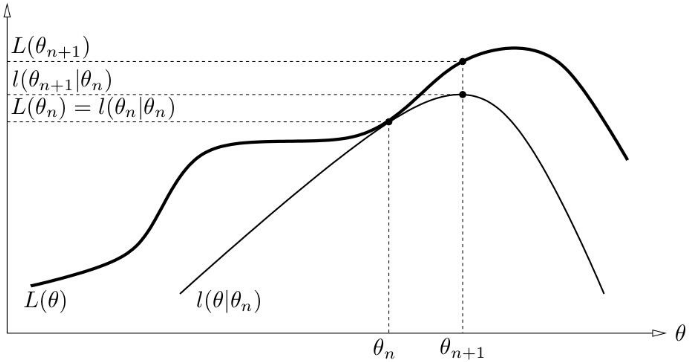
- Mixture of probabilistic models $(M_1, M_2, \ldots, M_k)$ with parameters $\Theta = (\theta_1, \theta_2, \ldots, \theta_k)$
- Observation $O = o_1 o_2 \ldots o_N$
- EM builds a sequence of estimates $\Theta_1, \Theta_2, \ldots, \Theta_n$
- $L(\Theta_j)$ — log-likelihood function, $\ln Pr(O \mid \Theta_j)$
- Want to extend the sequence with Θ_{n+1} such that $L(\Theta_{n+1}) > L(\Theta_n)$



- EM performs a form of gradient descent

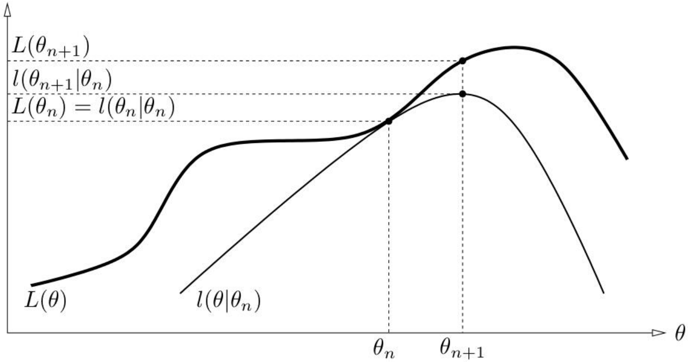
Theoretical foundations of EM

- Mixture of probabilistic models $(M_1, M_2, \ldots, M_k)$ with parameters $\Theta = (\theta_1, \theta_2, \ldots, \theta_k)$
- Observation $O = o_1 o_2 \ldots o_N$
- EM builds a sequence of estimates $\Theta_1, \Theta_2, \ldots, \Theta_n$
- $L(\Theta_j)$ — log-likelihood function, $\ln Pr(O \mid \Theta_j)$
- Want to extend the sequence with Θ_{n+1} such that $L(\Theta_{n+1}) > L(\Theta_n)$



- EM performs a form of gradient descent
- If we update Θ_n to Θ' we get an new likelihood $L(\Theta_n) + \Delta(\Theta', \Theta_n)$ which we call $\ell(\Theta' \mid \Theta_n)$

- Mixture of probabilistic models $(M_1, M_2, \ldots, M_k)$ with parameters $\Theta = (\theta_1, \theta_2, \ldots, \theta_k)$
- Observation $O = o_1 o_2 \ldots o_N$
- EM builds a sequence of estimates $\Theta_1, \Theta_2, \ldots, \Theta_n$
- $L(\Theta_j)$ — log-likelihood function, $\ln Pr(O \mid \Theta_j)$
- Want to extend the sequence with Θ_{n+1} such that $L(\Theta_{n+1}) > L(\Theta_n)$



- EM performs a form of gradient descent
- If we update Θ_n to Θ' we get an new likelihood $L(\Theta_n) + \Delta(\Theta', \Theta_n)$ which we call $\ell(\Theta' \mid \Theta_n)$
- Choose Θ_{n+1} to maximize $\ell(\Theta' \mid \Theta_n)$

Semi-supervised learning

- Supervised learning requires labelled training data

Semi-supervised learning

- Supervised learning requires labelled training data
- What if we don't have enough labelled data?

Semi-supervised learning

- Supervised learning requires labelled training data
- What if we don't have enough labelled data?
- For a probabilistic classifier we can apply EM

Semi-supervised learning

- Supervised learning requires labelled training data
- What if we don't have enough labelled data?
- For a probabilistic classifier we can apply EM
  - Use available training data to assign initial probabilities

Semi-supervised learning

- Supervised learning requires labelled training data
- What if we don't have enough labelled data?
- For a probabilistic classifier we can apply EM
  - Use available training data to assign initial probabilities
  - Label the rest of the data using this model — fractional labels

Semi-supervised learning

- Supervised learning requires labelled training data
- What if we don't have enough labelled data?
- For a probabilistic classifier we can apply EM
  - Use available training data to assign initial probabilities
  - Label the rest of the data using this model — fractional labels
  - Add up counts and re-estimate the parameters

Semi-supervised topic classification

- Each document is a multiset or bag of words over a vocabulary
  $V = \{w_1, w_2, \ldots, w_m\}$

$n_1 \quad n_2 \quad - \quad n_m$

Set of words

Bag/multiset of words

Semi-supervised topic classification

- Each document is a multiset or bag of words over a vocabulary $V = \{w_1, w_2, \ldots, w_m\}$

- Each topic c has probability $Pr(c)$

Semi-supervised topic classification

- Each document is a multiset or bag of words over a vocabulary $V = \{w_1, w_2, \ldots, w_m\}$
- Each topic c has probability $Pr(c)$
- Each word $w_i \in V$ has conditional probability $Pr(w_i \mid c_j)$, for $c_j \in C$
  - Note that $\sum_{i=1}^{m} Pr(w_i \mid c_j) = 1$

Semi-supervised topic classification

- Each document is a multiset or bag of words over a vocabulary $V = \{w_1, w_2, \ldots, w_m\}$
- Each topic c has probability $Pr(c)$
- Each word $w_i \in V$ has conditional probability $Pr(w_i \mid c_j)$, for $c_j \in C$
  - Note that $\sum_{i=1}^{m} Pr(w_i \mid c_j) = 1$
- Assume document length is independent of the class

Semi-supervised topic classification

- Each document is a multiset or bag of words over a vocabulary $V = \{w_1, w_2, \ldots, w_m\}$
- Each topic c has probability $Pr(c)$
- Each word $w_i \in V$ has conditional probability $Pr(w_i \mid c_j)$, for $c_j \in C$
  - Note that $\sum_{i=1}^{m} Pr(w_i \mid c_j) = 1$
- Assume document length is independent of the class
- Only a small subset of documents is labelled
  - Use this subset for initial estimate of $Pr(c)$, $Pr(w_i \mid c_j)$

Generate one word
Second word
⋮
last word

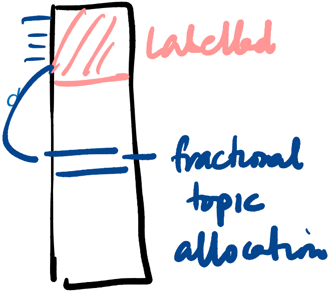
Semi-supervised topic classification

- Current model $Pr(c)$, $Pr(w_i \mid c_j)$

Semi-supervised topic classification

- Current model $Pr(c)$, $Pr(w_i \mid c_j)$
- Compute $Pr(c_j \mid d)$ for each unlabelled document d
  - Normally we assign the maximum among these as the class for d
  - Here we keep fractional values

- Current model $Pr(c)$, $Pr(w_i \mid c_j)$
- Compute $Pr(c_j \mid d)$ for each unlabelled document d
  - Normally we assign the maximum among these as the class for d
  - Here we keep fractional values
- Recompute $Pr(c_j) = \dfrac{\sum_{d \in D} Pr(c_j \mid D)}{|D|}$
  - For labelled d, $Pr(c_j \mid d) \in \{0, 1\}$
  - For unlabelled d, $Pr(c_j \mid d)$ is fractional value computed from current parameters



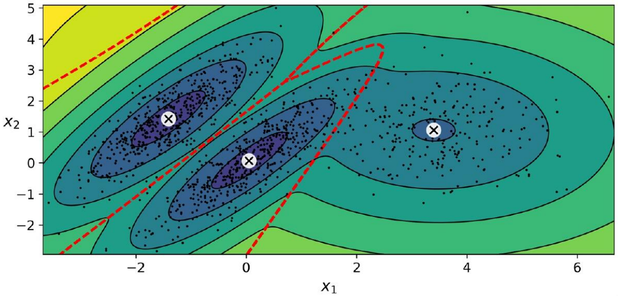
Semi-supervised topic classification

- Current model $Pr(c)$, $Pr(w_i \mid c_j)$
- Compute $Pr(c_j \mid d)$ for each unlabelled document d
  - Normally we assign the maximum among these as the class for d
  - Here we keep fractional values
- Recompute $Pr(c_j) = \dfrac{\sum_{d \in D} Pr(c_j \mid D)}{|D|}$
  - For labelled d, $Pr(c_j \mid d) \in \{0, 1\}$
  - For unlabelled d, $Pr(c_j \mid d)$ is fractional value computed from current parameters
- Recompute $Pr(w_i \mid c_j)$ — fraction of occurrences of w_i in documents labelled c_j
  - n_{id} — occurrences of w_i in d
  - $Pr(w_i \mid c_j) = \dfrac{\sum_{d \in D} n_{id}\, Pr(c_j \mid d)}{\sum_{t=1}^{m} \sum_{d \in D} n_{td}\, Pr(c_j \mid d)}$

$$\frac{\sum_{d \in D_j} n_{id}}{\sum_t \sum_{d \in D_j} n_{td}}$$

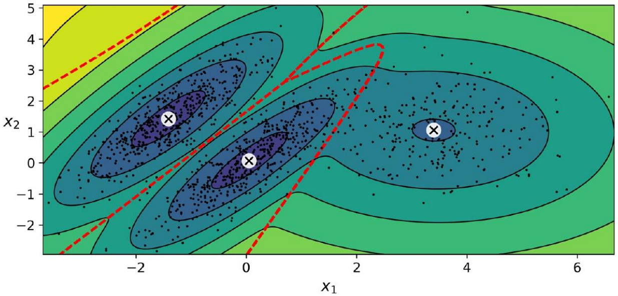
Clustering

- Data points from a mixture of Gaussian distributions



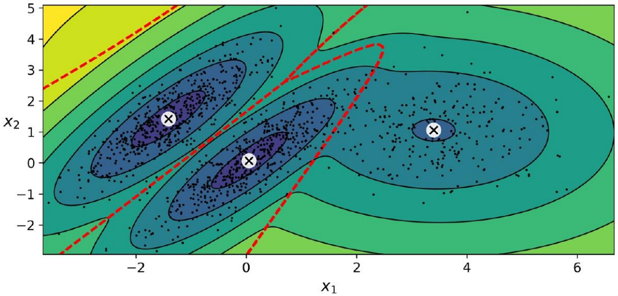
Clustering

- Data points from a mixture of Gaussian distributions
- Use EM to estimate the parameters of each Gaussian distribution



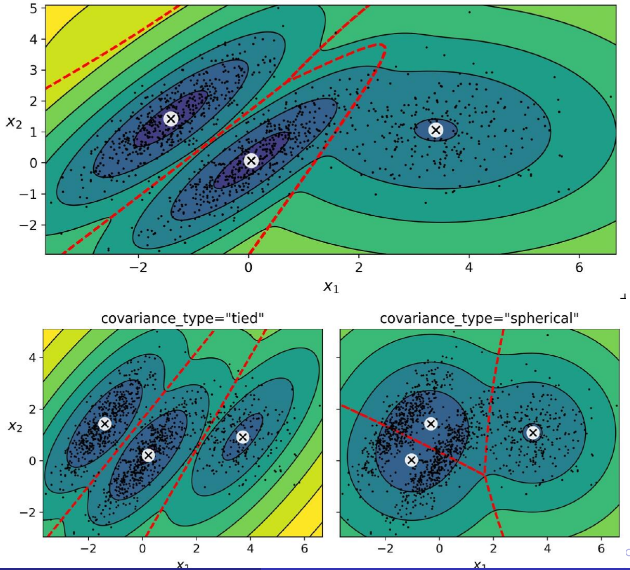
Clustering

- Data points from a mixture of Gaussian distributions
- Use EM to estimate the parameters of each Gaussian distribution
- Assign each point to "best" Gaussian



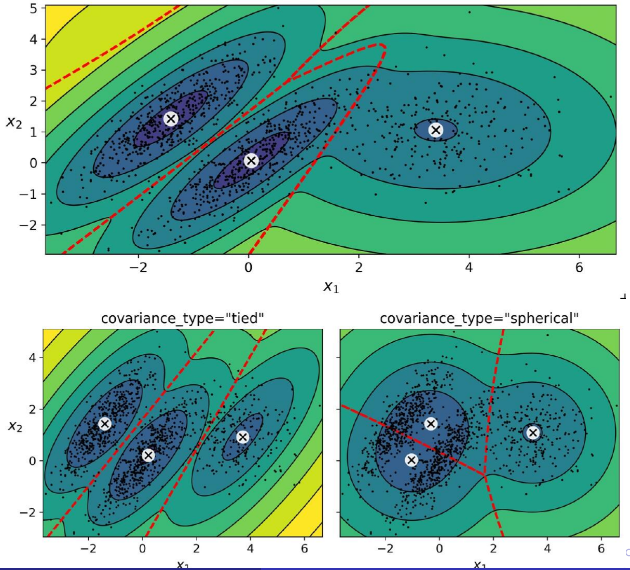
Clustering

- Data points from a mixture of Gaussian distributions
- Use EM to estimate the parameters of each Gaussian distribution
- Assign each point to "best" Gaussian
- Can tweak the shape of the clusters by constraining the covariance matrix



Clustering

- Data points from a mixture of Gaussian distributions
- Use EM to estimate the parameters of each Gaussian distribution
- Assign each point to "best" Gaussian
- Can tweak the shape of the clusters by constraining the covariance matrix
- Outliers are those that are outside $k\sigma$ for all the Gaussians



Clustering

- Data points from a mixture of Gaussian distributions
- Use EM to estimate the parameters of each Gaussian distribution
- Assign each point to "best" Gaussian
- Can tweak the shape of the clusters by constraining the covariance matrix
- Outliers are those that are outside $k\sigma$ for all the Gaussians