Australian National University

COMP4680/8650 Advanced Topics in Statistical Machine Learning

Week 5: Learning with Missing Data

Stephen Gould

Review: Maximum Likelihood Learning

The *maximum likelihood principle* says that we wish to choose the parameters of the model that maximizes the probability of us observing our training data,

$$L(\theta; D) = \prod_{m} P(x^{(m)}; \theta)$$

MLE Worked Example

Suppose we have a biased coin. Let $\theta \in [0,1]$ be the probability of heads, so

$$P(x;\theta) = \begin{cases} \theta & \text{if heads} \\ 1-\theta & \text{if tails} \end{cases}$$

Then given a sequence of coin flips,

$$L(\theta; D) = \theta^H (1-\theta)^T$$

where *H* is the number of heads and *T* is the number of tails observed. **Note it is often easier to maximize the log-likelihood.**

The maximum likelihood parameters are then $\hat{\theta} = \frac{H}{H+T}$

Learning with Latent Variables

- Suppose we wish to estimate parameters θ of the model $p(x, z; \theta)$
- But we are only given $\mathrm{D} = \{x^{(1)}, x^{(2)}, \dots, x^{(m)}\}$
- z is called a **latent** (or hidden) variable

Maximum Likelihood Principle

- By maximum log-likelihood we have

$$l(\theta; D) = \sum_{i=1}^{m} \log p(x^{(i)}; \theta)$$

- But our distribution is over (x, z) so we need to marginalize out z

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

Difficulties with Latent Variables

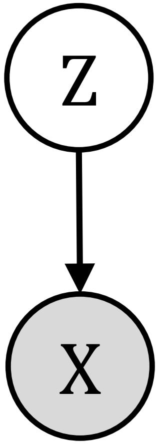
- There are a few difficulties when learning models with latent variables
  - We need to marginalize them out, which could be expensive (depending on the distribution)
  - The resulting likelihood function is nonconvex
  - Parameters become unidentifiable

Latent Variable Example

- Assume I have two biased coins. I repeatedly pick a coin at random, flip it ten times, and tell you the outcomes, but not which coin I used.

- Let $Z \in \{1,2\}$ be the coin and $X \in \{H, T\}$

- Let θ_1 and θ_2 be the probability that the first and second coin land heads, respectively.



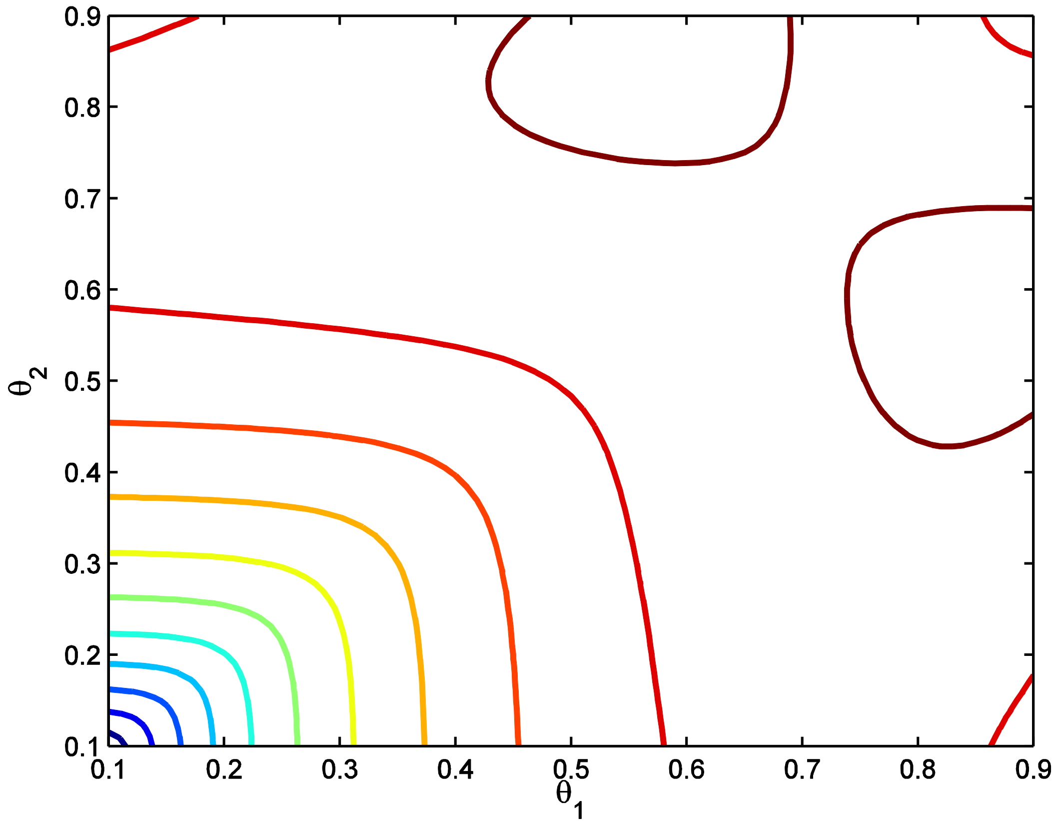
Latent Variable Example

- Let H_t be the number of heads in round t. The log-likelihood function is then

$$\sum_t \log\left(\frac{1}{2}\theta_1^{H_t}(1-\theta_1)^{10-H_t} + \frac{1}{2}\theta_2^{H_t}(1-\theta_2)^{10-H_t}\right)$$

- Note the symmetry between θ_1 and θ_2. **What experiment modification would break the symmetry?**

Latent Variable Example



Aside: Jensen's Inequality

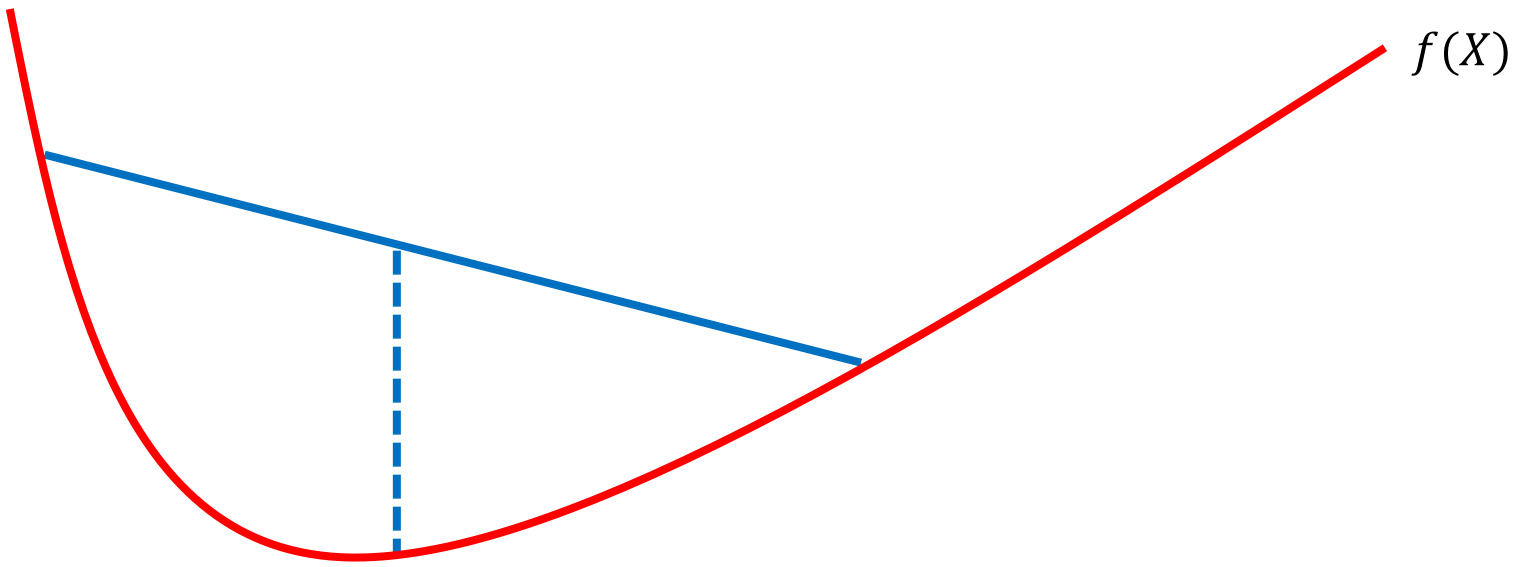
Let f be a convex function and let X be a random variable. Then

$$E[f(X)] \geq f(E[X])$$

If f is strictly convex, then $E[f(X)] = f(E[X])$ if and only if $X = E[X]$ (with probability 1).

Australian National University

Aside: Jensen's Inequality



A Lower Bound on the Log-Likelihood

- For each training example let Q_i be some distribution over $z^{(i)}$
  - So $\sum_z Q_i(z) = 1$ and $Q_i(z) \geq 0$

- Then…

A Lower Bound on the Log-Likelihood

$$\sum_{i=1}^{m} \log p\left(x^{(i)}; \theta\right) = \sum_{i=1}^{m} \log\left(\sum_{z} p\left(x^{(i)}, z; \theta\right)\right)$$

$$= \sum_{i=1}^{m} \log\left(\sum_{z} Q_i(z) \frac{p\left(x^{(i)}, z; \theta\right)}{Q_i(z)}\right)$$

$$\geq \sum_{i=1}^{m} \sum_{z} Q_i(z) \log\left(\frac{p\left(x^{(i)}, z; \theta\right)}{Q_i(z)}\right)$$

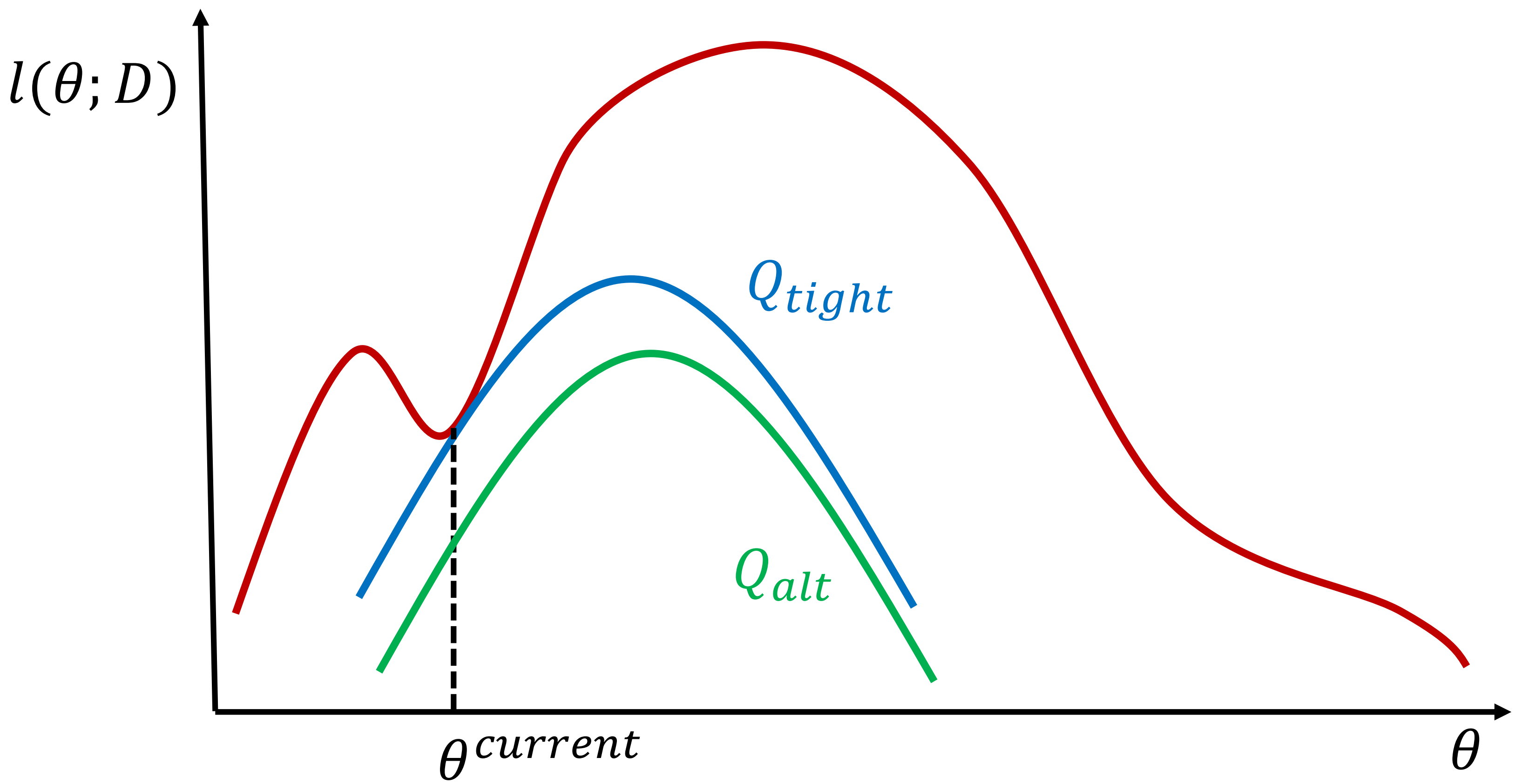
$$= \sum_{i=1}^{m} E_{z \sim Q_i}\left[\log\left(\frac{p\left(x^{(i)}, z; \theta\right)}{Q_i(z)}\right)\right]$$

A Lower Bound on the Log-Likelihood

For any set of distributions Q_i

$$\sum_{i=1}^{m} E_{z\sim Q_i}\left[\log\left(\frac{p\left(x^{(i)}, z; \theta\right)}{Q_i(z)}\right)\right]$$

gives a lower bound on $l(\theta; D)$. It seems natural to choose Q_i to be tight at the current estimate.


$l(\theta; D)$
Q_{tight}
Q_{alt}
$\theta^{current}$
θ

Making the Lower Bound Tight

For Q_i to be tight we must have equality at $\theta_{current}$.

By Jensen's inequality

$$\frac{p\left(x^{(i)}, z; \theta\right)}{Q_i(z)} = const.$$

and since $\sum_z Q_i(z) = 1$ we have

$$Q_i(z) = \frac{p(x^{(i)}, z; \theta)}{\sum_z p(x^{(i)}, z; \theta)} = p(z \mid x^{(i)}; \theta)$$

Australian National University

The EM Algorithm

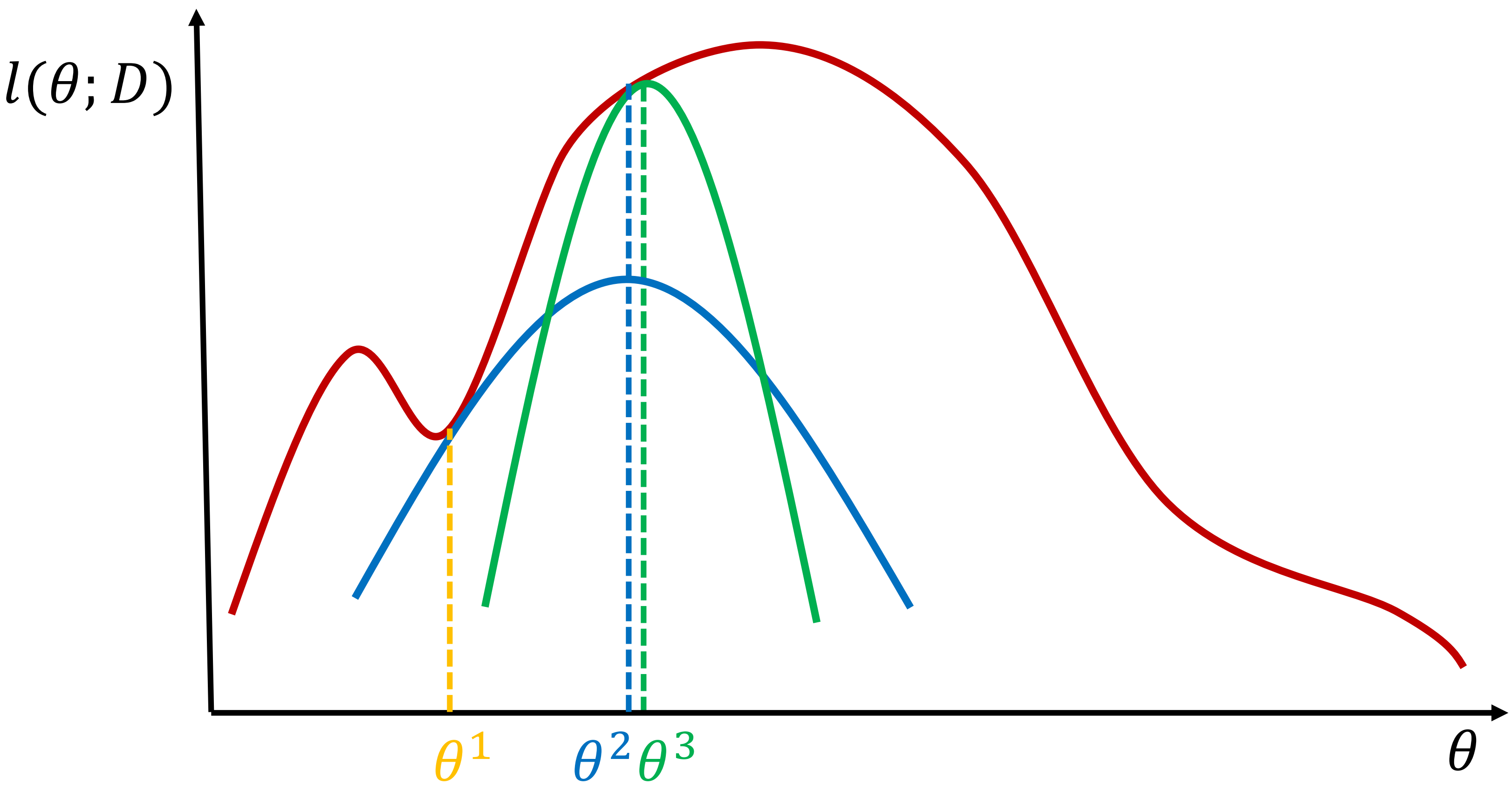
E-Step:

$$\text{set } Q_i(z) = p\left(z \mid x^{(i)}; \theta \right)$$

M-Step:

$$\text{set } \theta = \underset{\theta}{\operatorname{argmax}} \sum_{i=1}^{m} E_{z\sim Q_i}\left[\log\left(\frac{p\left(x^{(i)},z;\theta\right)}{Q_i(z)}\right)\right]$$

EM Illustration



Convergence

Theorem. EM algorithm will converge to a local maximum of the log-likelihood function.

Proof.

$$\begin{aligned} l\left(\theta^{(t+1)}\right) &\geq \sum_{i=1}^{m} E_{z\sim Q_i}\left[\log\left(\frac{p\left(x^{(i)}, z; \theta^{(t+1)}\right)}{Q_i(z)}\right)\right] \\ &\geq \sum_{i=1}^{m} E_{z\sim Q_i}\left[\log\left(\frac{p\left(x^{(i)}, z; \theta^{(t)}\right)}{Q_i(z)}\right)\right] \\ &= l\left(\theta^{(t)}\right) \end{aligned}$$

EM Variants

- **Generalised EM:** It is not necessary to perform exact maximization during the M-step. It is sufficient to improve over the current estimate.

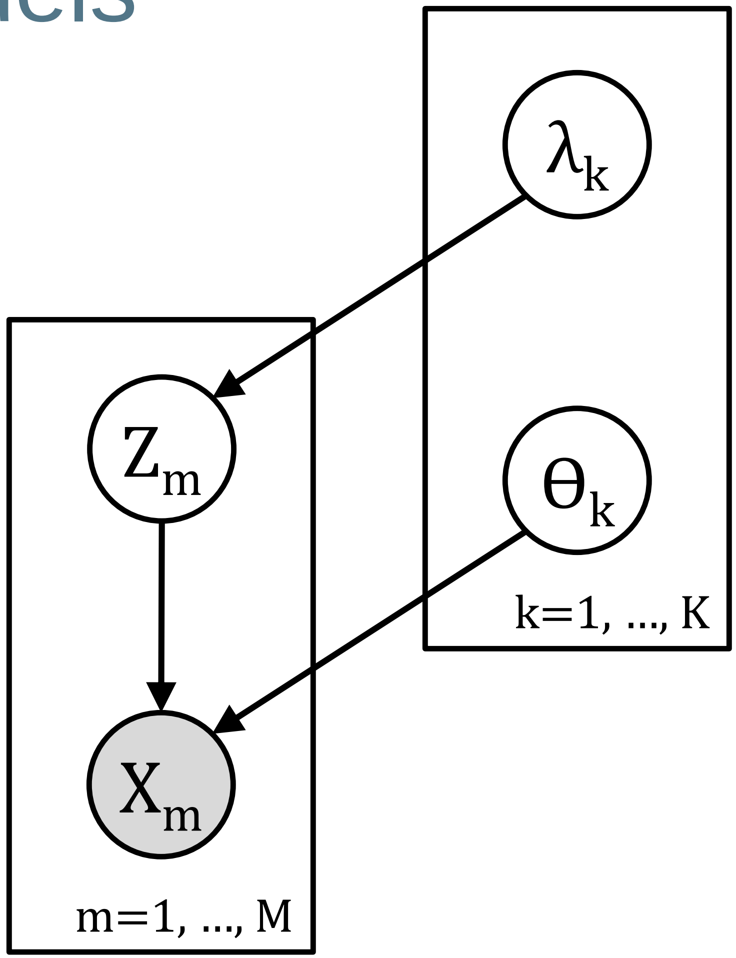
- **Hard assignment EM:** “Complete the data” by choosing the $z^{(m)}$ that maximizes $p(z \mid x^{(m)})$ for the current parameters.

Gradient Ascent versus EM

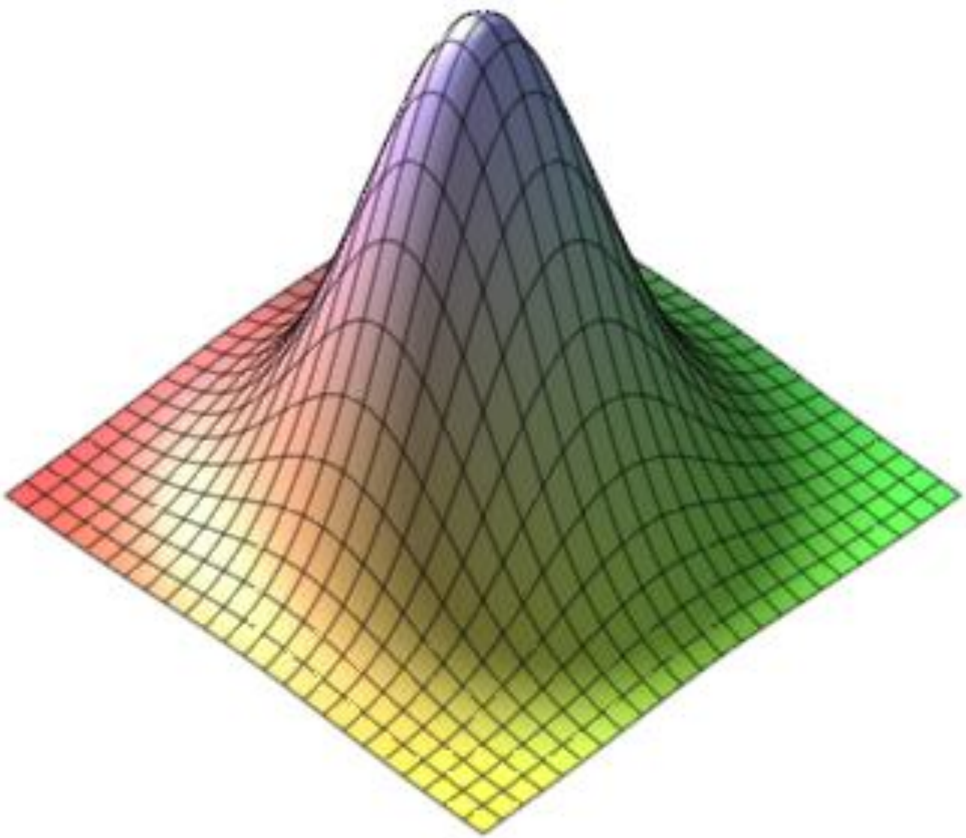
- The EM algorithm often makes good progress during the first few iterations and then slows down.

- Gradient methods usually show the opposite behaviour. They are initially slow, but speed up when close to a local maximum.

Mixture Models



Multivariate Gaussian Distribution



Multivariate Gaussian Distribution

The multivariate Gaussian distribution is given by

$$p(x;\mu,\Sigma) = \frac{1}{(2\pi)^{\frac{n}{2}}|\Sigma|^{\frac{1}{2}}}\exp\left(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right)$$

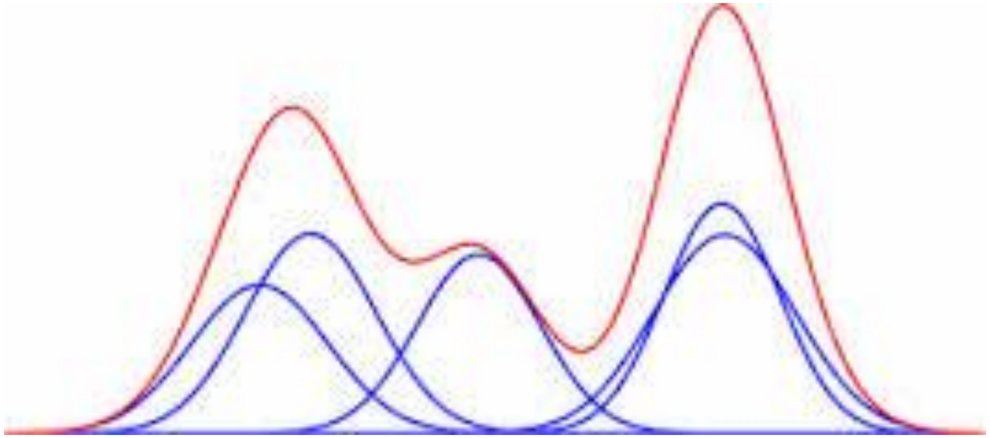
Often written as $\mathcal{N}(\mu,\Sigma)$ where μ is the mean and Σ is the covariance matrix.

Gaussian Mixture Models

Many distributions can be approximated by a mixture of Gaussians

$$p(x) = \sum_k \lambda_k \mathcal{N}(x; \mu_k, \Sigma_k)$$

where $\sum_k \lambda_k = 1$.



Australian National University

EM for Mixture of Gaussians

E-Step:

$$
\begin{aligned}
Q_i(z=k) &= p\left(z=k \mid x^{(i)}\right) \\
&\propto p\left(x^{(i)} \mid z=k\right) p(z=k) \\
&= \lambda_k N\left(x^{(i)}; \mu_k, \Sigma_k\right)
\end{aligned}
$$

$$
\therefore Q_i(z=k) = \frac{\lambda_k N\left(x^{(i)}; \mu_k, \Sigma_k\right)}{\sum_{k'=1}^{K} \lambda_{k'} N\left(x^{(i)}; \mu_{k'}, \Sigma_{k'}\right)}
$$

EM for Mixture of Gaussians

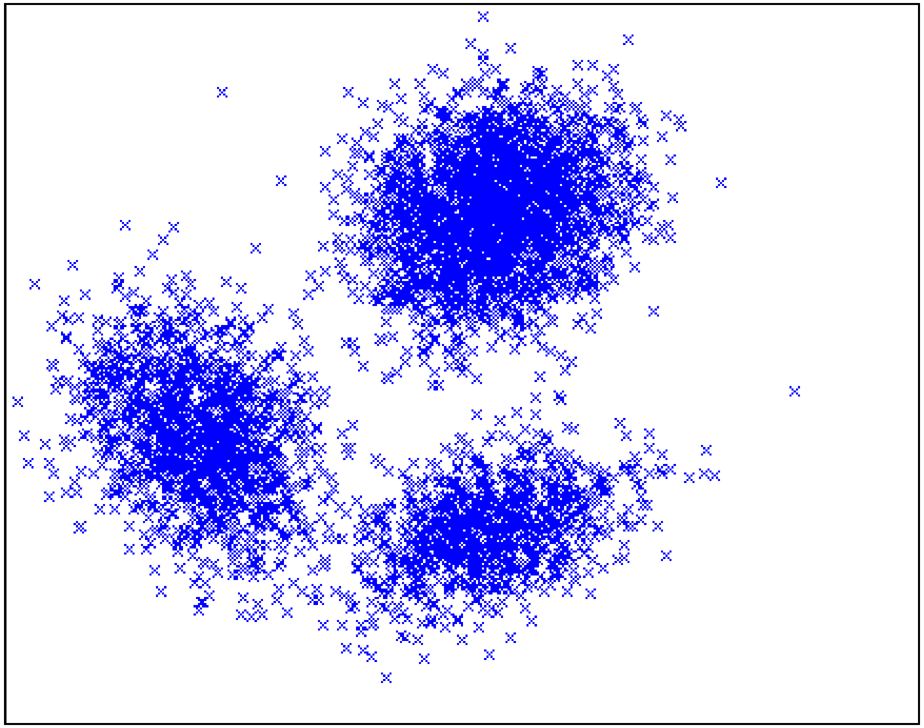
M-Step:

$$\lambda_k = \frac{1}{m}\sum_i Q_i(k)$$

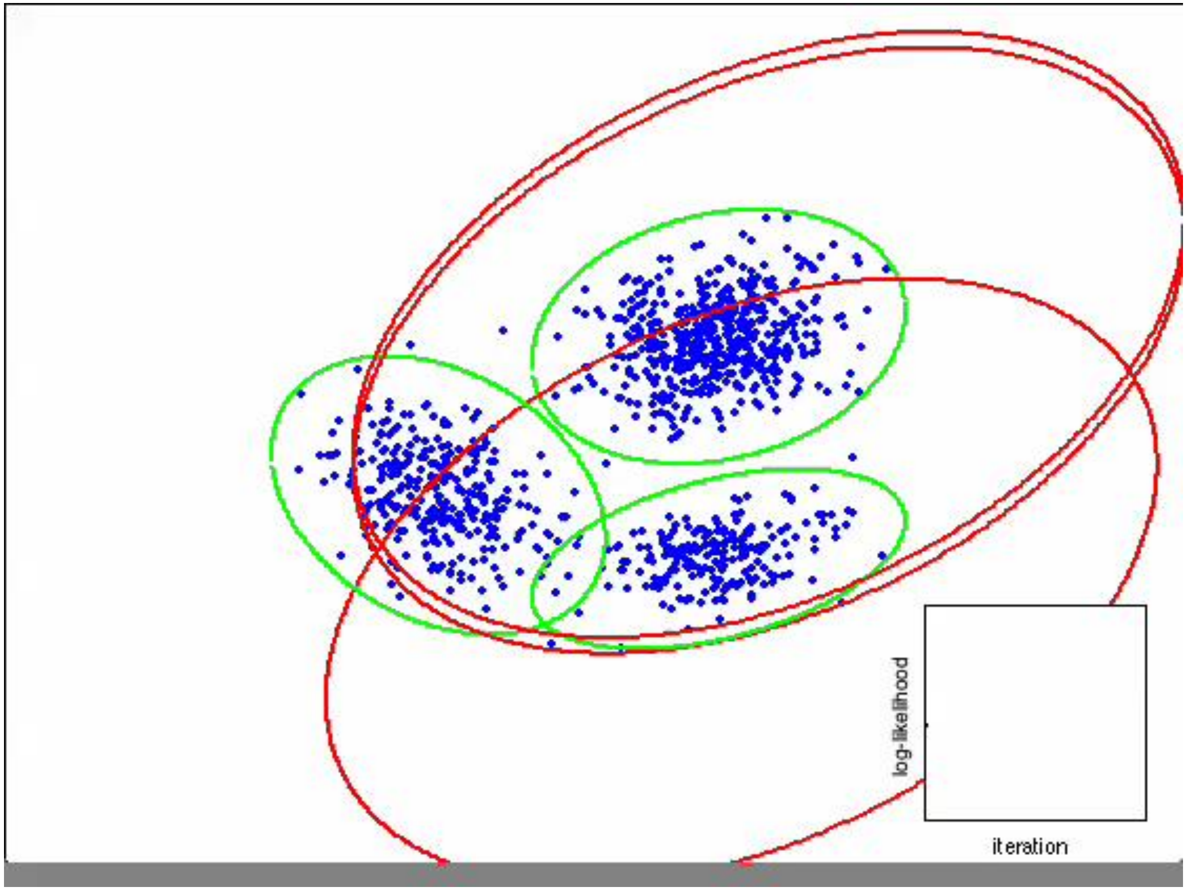
$$\mu_k = \frac{\sum_i Q_i(k)\, x^{(i)}}{\sum_i Q_i(k)}$$

$$\Sigma_k = \frac{\sum_i Q_i(k)\left(x^{(i)} - \mu_k\right)\left(x^{(i)} - \mu_k\right)^T}{\sum_i Q_i(k)}$$

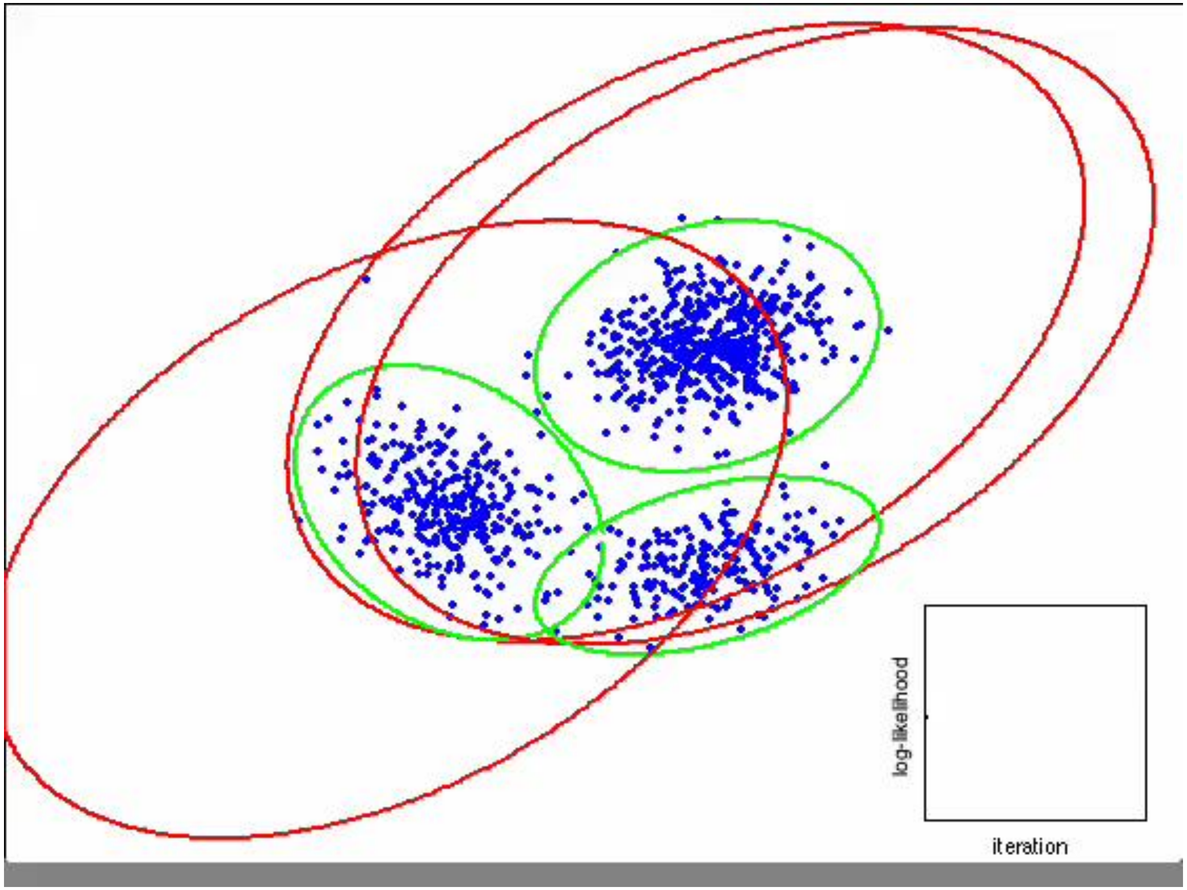
GMM Demonstration



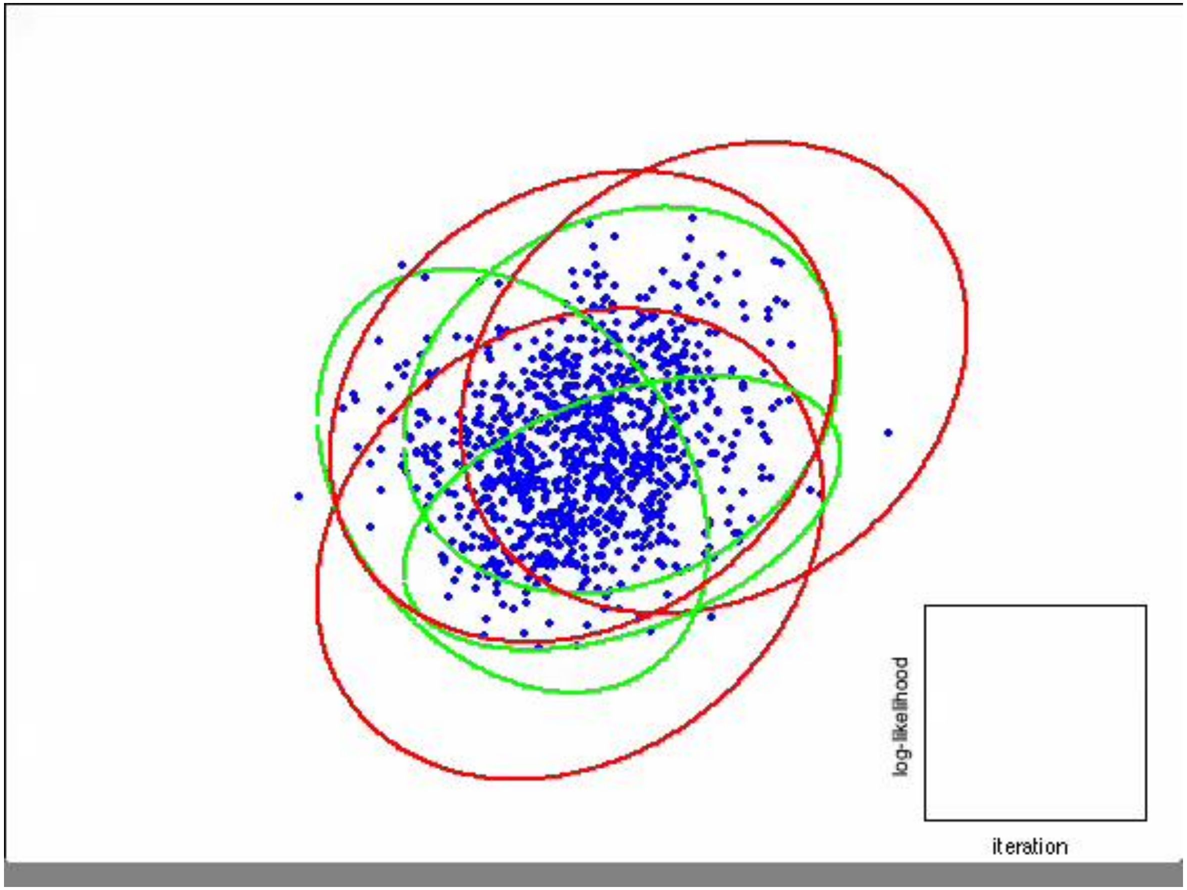
GMM Demonstration



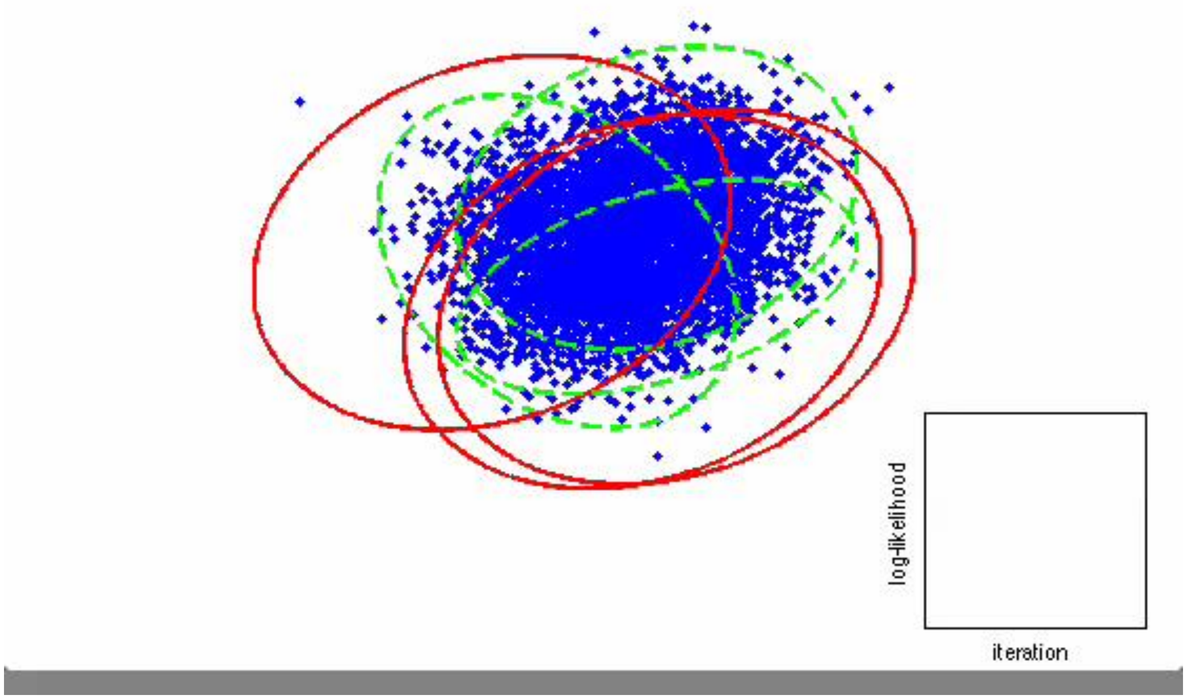
GMM Demonstration



GMM Demonstration



GMM Demonstration



Australian National University

Practical Considerations

- **Random initialization:** multiple random restarts may find better local maxima.
- **Numerical stability:** computing in log-space often helps with numerical stability (especially when dealing with small probabilities).
- **Regularization:** the maximum likelihood parameters are not always the ones you want!