Semi-Supervised Learning using Gaussian Mixture Models

Peter L Green

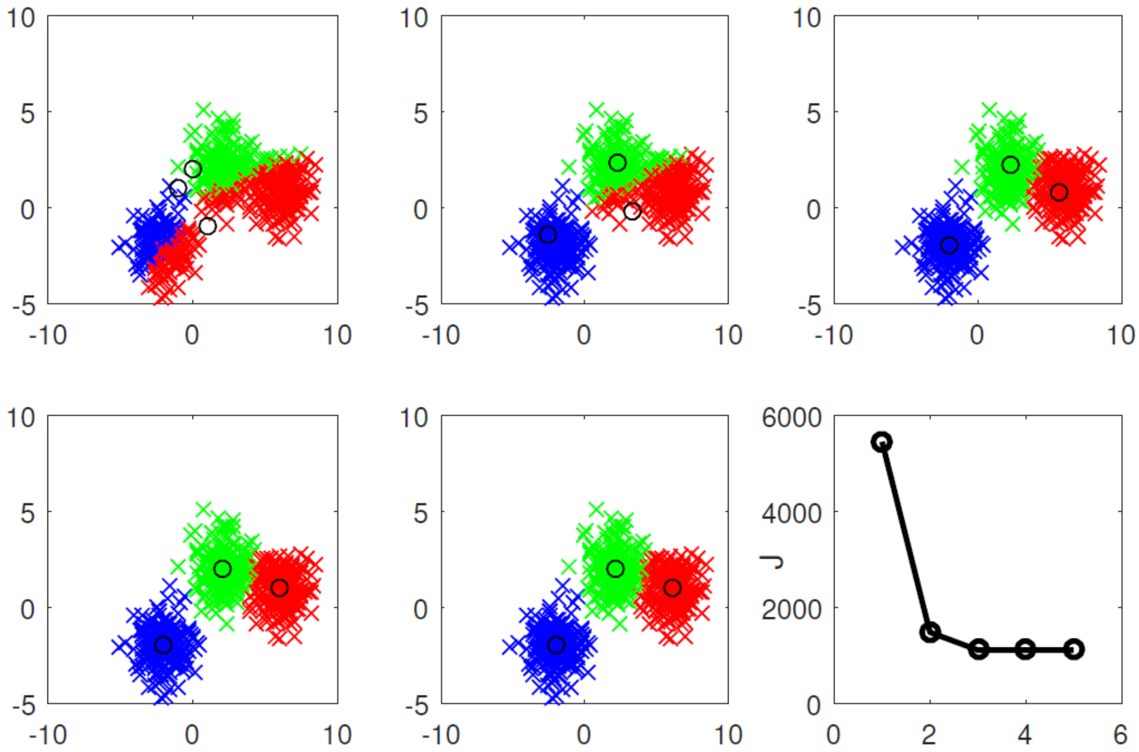
p.l.green@Liverpool.ac.uk

Institute for Risk and Uncertainty

University of Liverpool

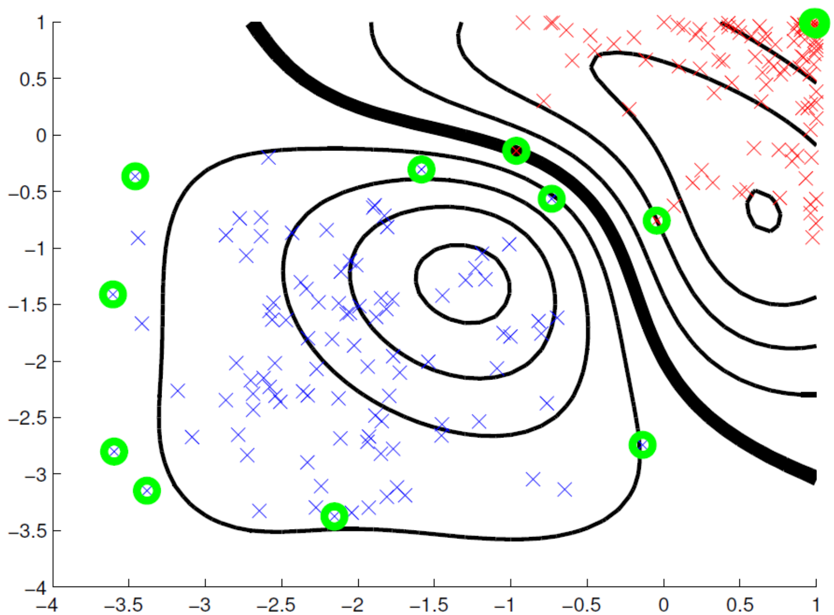
Introduction

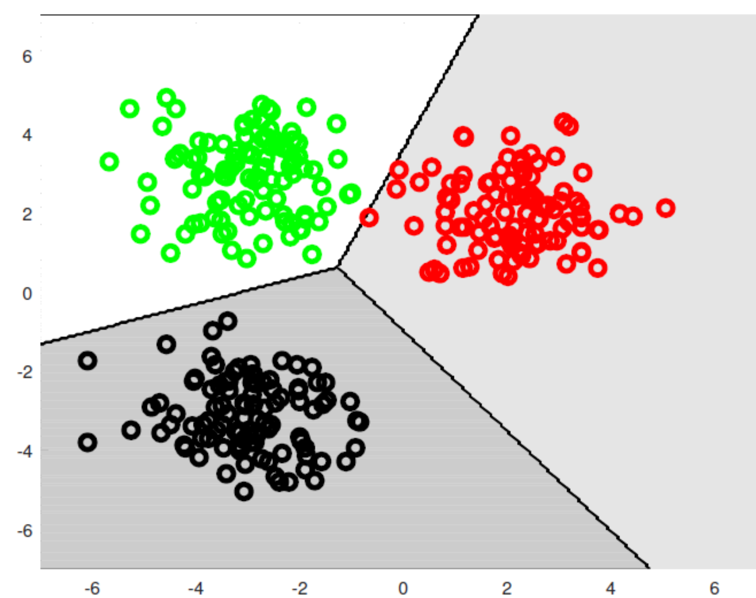
Unsupervised Learning: each data point belongs to a ‘cluster’, but this information is hidden. The aim is to identify the clusters:



Introduction

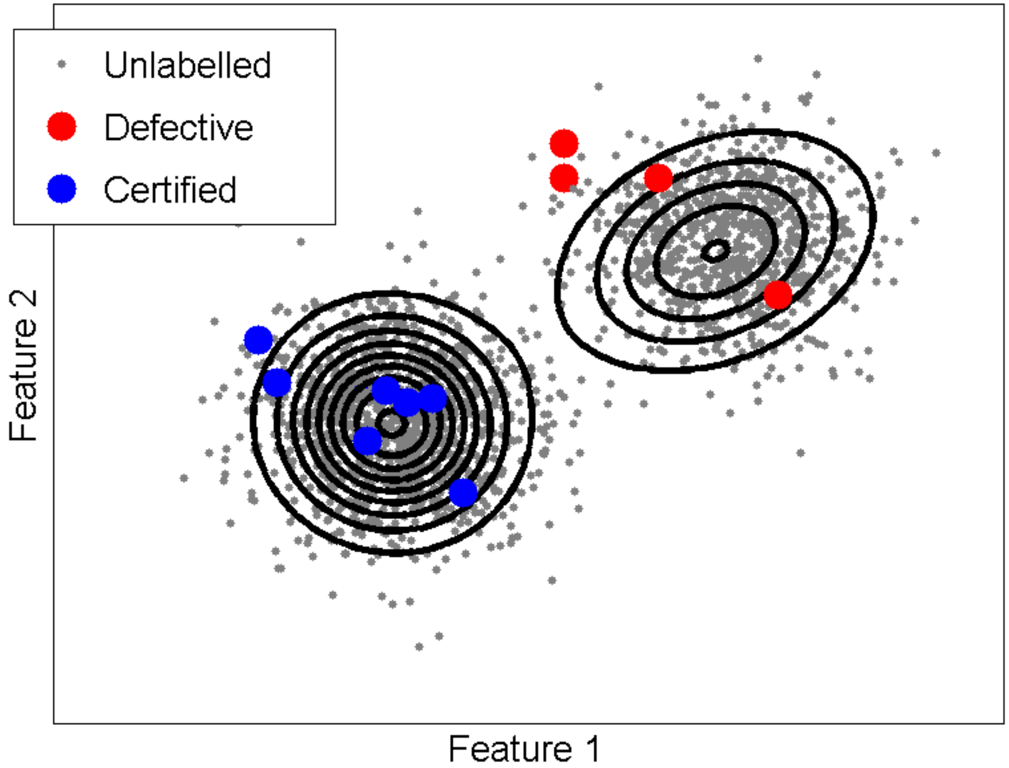
Supervised Learning: the data arrives with ‘labels’, which tell us which cluster it belongs to. The aim is to classify future data using this information:





Introduction

Semi-Supervised Learning: some data is labelled, some is not. It is useful when labelling data is expensive.



Introduction

Disclaimer…

- Semi-supervised learning is not very new (in Machine Learning anyway).

- It hasn’t found its way into some disciplines and it looks useful – potential for collaboration.

- Here we look at one form (Gaussian Mixture Models) because I think they are easier to understand.

- Extra maths is in the appendix (slides will be sent round later).

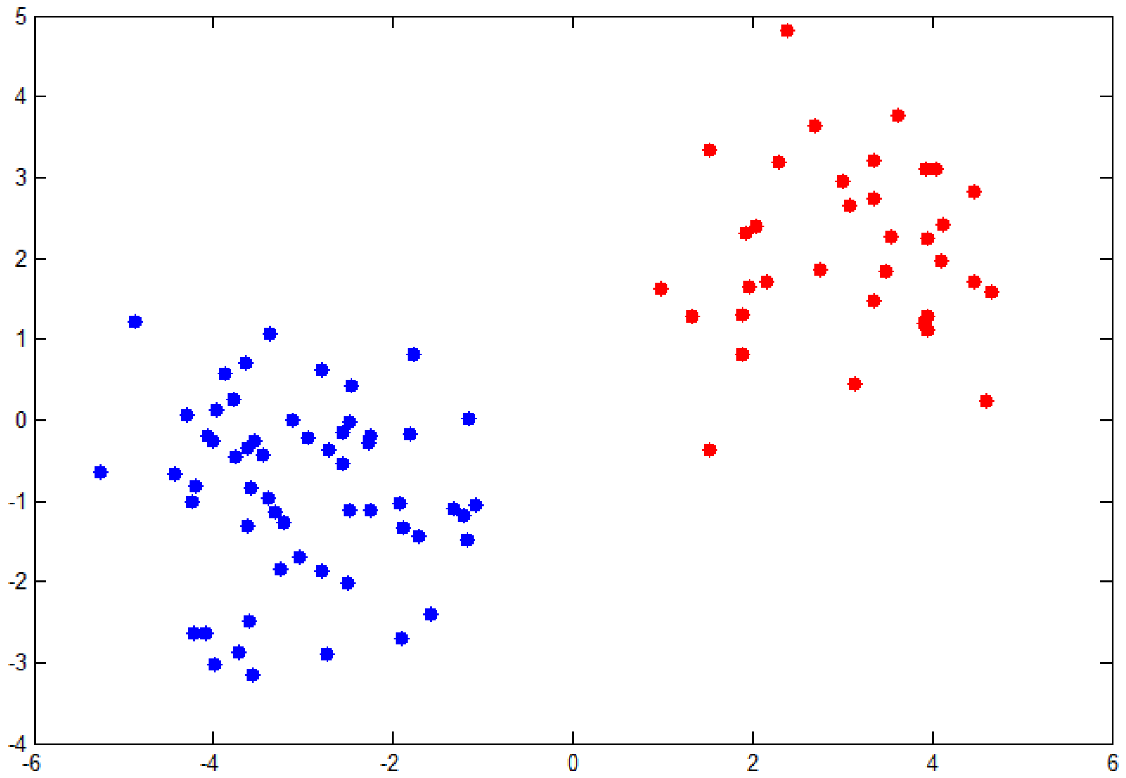
Introduction

In this talk:

- Supervised Learning
- Unsupervised Learning
- The Expectation Maximisation algorithm
- Semi-Supervised Learning

Gaussian mixture (supervised learning)

Say we collect some labelled data:



We hypothesise that each of these samples comes from a mixture of Gaussian distributions:

$$p(\boldsymbol{x}) = \sum_{k=1}^{K} \pi_k N(\boldsymbol{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

$\boldsymbol{\mu} = \{\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_K\}$: Means of Gaussians

$\boldsymbol{\Sigma} = \{\boldsymbol{\Sigma}_1, \dots, \boldsymbol{\Sigma}_K\}$: Covariance matrices of Gaussians

$\boldsymbol{\pi} = \{\pi_1, \dots, \pi_K\}$: Mixture proportions (sum to 1)

$\boldsymbol{\theta} = \{\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\pi}\}$: **Parameters to be estimated**

- In this *supervised learning* problem, the data is already labelled (we already know which Gaussian was used to generate each point).

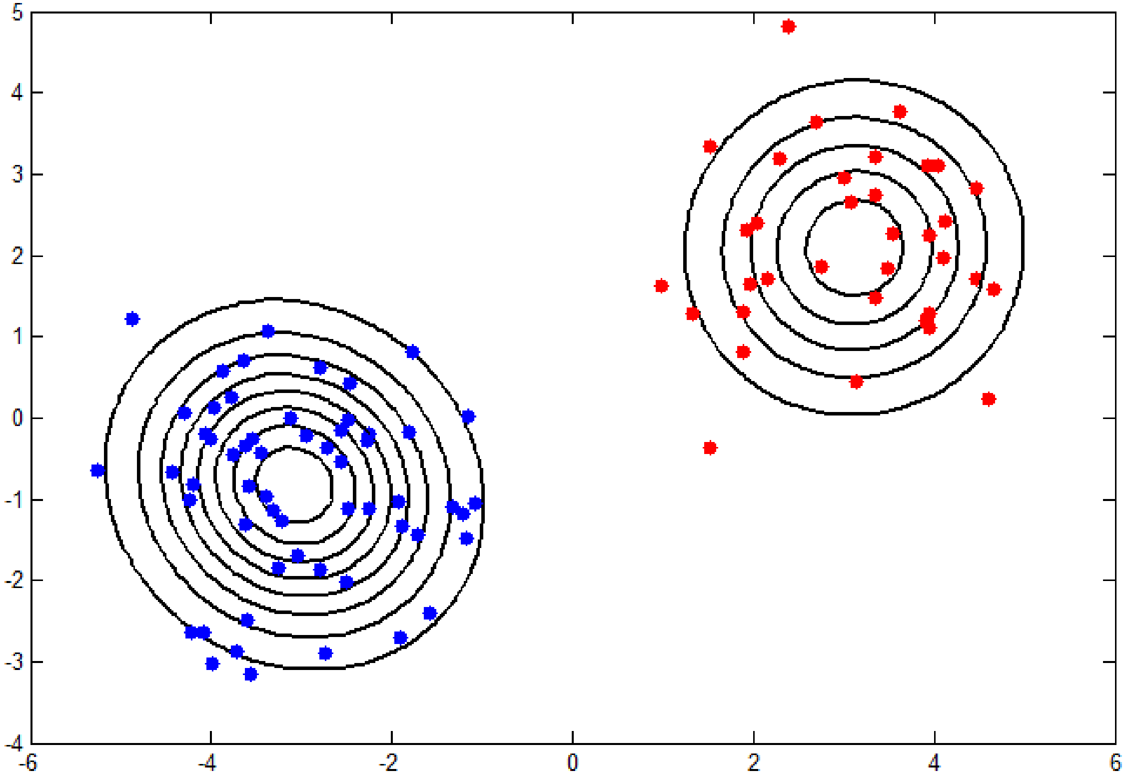
- Say that X_k is the set of N_k samples that were generated from the kth Gaussian. Intuitively we set:

$$\boldsymbol{\mu}_k = \frac{1}{N_k} \sum_{x \in X_k} \boldsymbol{x}$$

$$\boldsymbol{\Sigma}_k = \frac{1}{N_k} \sum_{x \in X_k} (\boldsymbol{x} - \boldsymbol{\mu}_k)(\boldsymbol{x} - \boldsymbol{\mu}_k)^T$$

$$\pi_k = \frac{N_k}{N}$$

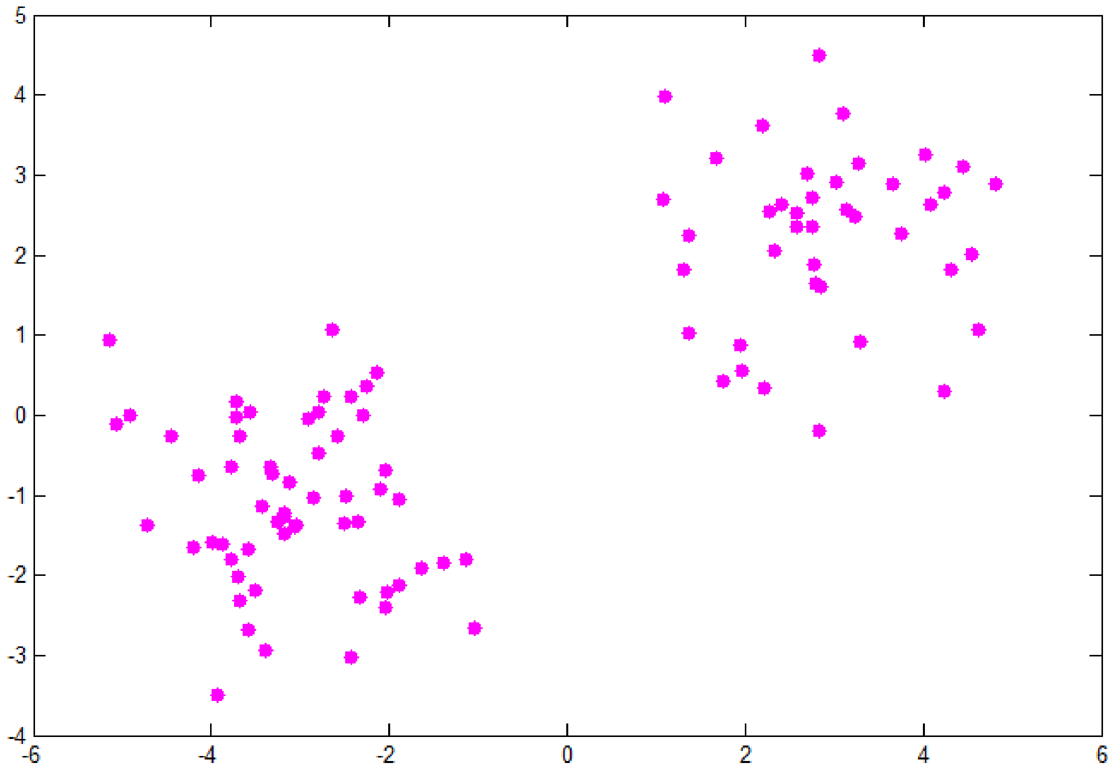
Results:



(You can also show that these parameter estimates are equal to the estimates realised through maximum likelihood.)

Gaussian mixture (unsupervised learning)

Now the data arrives unlabelled:



Say we have 2 clusters and N data points, as in the previous example. We introduce 'indicators', $\boldsymbol{Z}$:

$$\boldsymbol{Z} = \{\boldsymbol{z}_1, \dots, \boldsymbol{z}_N\}$$

$$\boxed{\boldsymbol{z}_n = \begin{Bmatrix} 1 \\ 0 \end{Bmatrix}} \longleftarrow \text{n th point is from cluster 1}$$

$$\boxed{\boldsymbol{z}_n = \begin{Bmatrix} 0 \\ 1 \end{Bmatrix}} \longleftarrow \text{n th point is from cluster 2}$$

We call these *latent variables,* as they are hidden from us.

We also define:

$$\pi_k = \Pr(z_{nk} = 1)$$

This formulation allows us to write the likelihood of each point as:

$$p(\boldsymbol{x}_n, \boldsymbol{z}_n | \boldsymbol{\theta}) = p(\boldsymbol{x}_n | \boldsymbol{z}_n, \boldsymbol{\theta}) p(\boldsymbol{z}_n | \boldsymbol{\theta}) = \prod_{k=1}^{K} N(\boldsymbol{x}_n; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)^{z_{nk}} \pi_k^{z_{nk}}$$

$$= \prod_{k=1}^{K} [N(\boldsymbol{x}_n; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \pi_k]^{z_{nk}}$$

Assuming uncorrelated samples:

Likelihood

$$\boxed{p(\boldsymbol{X}, \boldsymbol{Z} | \boldsymbol{\theta}) = \prod_{n=1}^{N} \prod_{k=1}^{K} [\pi_k N(\boldsymbol{x}_n; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)]^{z_{nk}}}$$

What else can we do?

We already have expressions for $p(\boldsymbol{X}|\boldsymbol{Z},\boldsymbol{\theta})$ and $p(\boldsymbol{Z}|\boldsymbol{\theta})$ so we can write the posterior probability of $\boldsymbol{Z}$ using Bayes' theorem:

$$p(\boldsymbol{Z}|\boldsymbol{X},\boldsymbol{\theta}) = \frac{p(\boldsymbol{X}|\boldsymbol{Z},\boldsymbol{\theta})p(\boldsymbol{Z}|\boldsymbol{\theta})}{\sum_Z p(\boldsymbol{X}|\boldsymbol{Z},\boldsymbol{\theta})p(\boldsymbol{Z}|\boldsymbol{\theta})}$$

(Maths is in the appendix.)

This unsupervised learning problem is harder because we have to estimate both $\boldsymbol{Z}$ *and* $\boldsymbol{\theta}$. They cannot be estimated independently.

A sensible looking option... Start from an initial guess, denoted $\boldsymbol{\theta}^{old}$ and:

1. Hold $\boldsymbol{\theta}$ fixed and find the expected values of $\boldsymbol{Z}$ (latent variables):

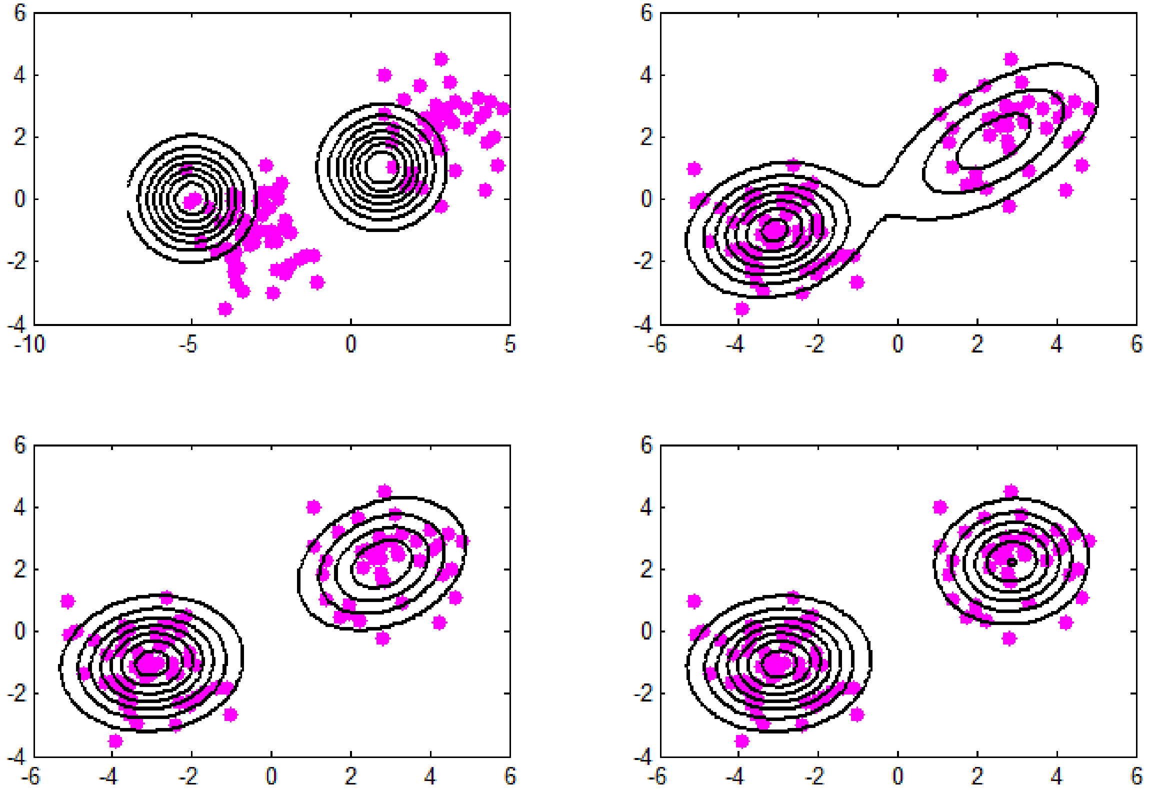
$$\mathrm{E}[z_{nk}] = \sum_{k=1}^{K} z_{nk}\, p(z_{nk}|x_n, \boldsymbol{\theta}^{old})$$

2. Hold $\boldsymbol{Z}$ equal to their expected values and find new theta estimate (maximum likelihood):

$$\boldsymbol{\theta}^{new} = \mathrm{argmax}_{\boldsymbol{\theta}}\, p(\boldsymbol{X}, \mathrm{E}[\boldsymbol{Z}]|\boldsymbol{\theta})$$

Set $\boldsymbol{\theta}^{old} = \boldsymbol{\theta}^{new}$ and repeat...

Example results:



This is actually a very good strategy, and is described generally by the **Expectation-Maximisation (EM) algorithm**.

The EM algorithm

Remember that $\boldsymbol{Z}$ are the latent (hidden) variables. We write:

$$\log p(\boldsymbol{X}|\boldsymbol{\theta}) = \sum_{\boldsymbol{Z}} q(\boldsymbol{Z}) \log p(\boldsymbol{X}|\boldsymbol{\theta})$$

Arbitrary probability distribution

The EM algorithm

Remember that $\boldsymbol{Z}$ are the latent (hidden) variables. We write:

$$\log p(\boldsymbol{X}|\boldsymbol{\theta}) = \sum_{\boldsymbol{Z}} q(\boldsymbol{Z}) \log p(\boldsymbol{X}|\boldsymbol{\theta})$$

$$= \sum_{\boldsymbol{Z}} q(\boldsymbol{Z}) \log\left(\frac{p(\boldsymbol{X},\boldsymbol{Z}|\boldsymbol{\theta})}{q(\boldsymbol{Z})}\right) - \sum_{\boldsymbol{Z}} q(\boldsymbol{Z}) \log\left(\frac{p(\boldsymbol{Z}|\boldsymbol{X},\boldsymbol{\theta})}{q(\boldsymbol{Z})}\right)$$

$$= L(q,\boldsymbol{\theta}) + KL(q||p)$$

The EM algorithm

$$\log p(\boldsymbol{X}|\boldsymbol{\theta}) = \sum_{\boldsymbol{Z}} q(\boldsymbol{Z}) \log\left(\frac{p(\boldsymbol{X},\boldsymbol{Z}|\boldsymbol{\theta})}{q(\boldsymbol{Z})}\right) - \sum_{\boldsymbol{Z}} q(\boldsymbol{Z}) \log\left(\frac{p(\boldsymbol{Z}|\boldsymbol{X},\boldsymbol{\theta})}{q(\boldsymbol{Z})}\right)$$

$$= L(q,\boldsymbol{\theta}) + KL(q||p)$$

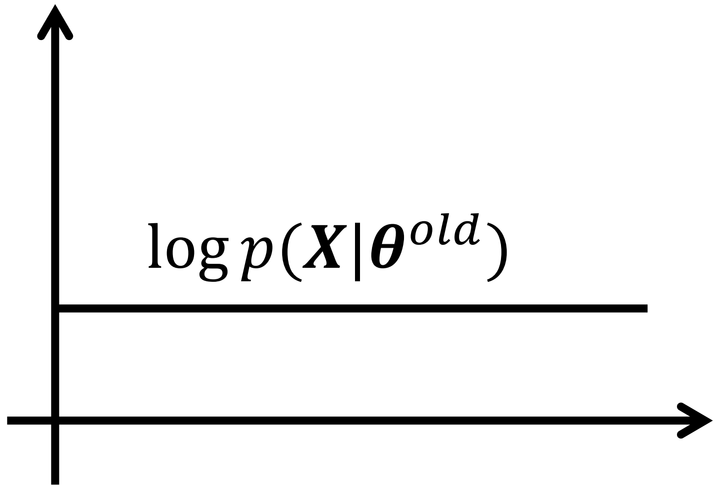
$KL(q||p) \geq 0$ where $KL(q||p) = 0$ iff $q = p$

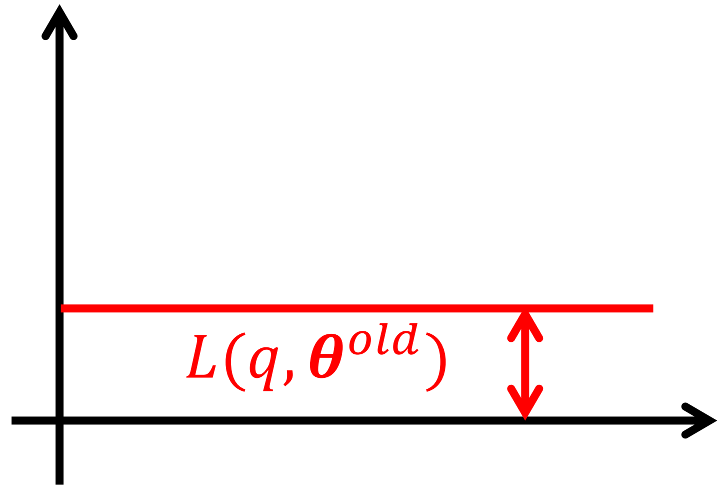
If follows that $L(q,\boldsymbol{\theta}) \leq \log p(\boldsymbol{X}|\boldsymbol{\theta})$. L is therefore a lower bound on $\log p(\boldsymbol{X}|\boldsymbol{\theta})$.

If we choose $q = p$ then the lower bound will be equal to $\log p(\boldsymbol{X}|\boldsymbol{\theta})$

Start from current guess: $\boldsymbol{\theta}^{old}$.

Set $q(\boldsymbol{Z}) = p(\boldsymbol{Z}|\boldsymbol{X}, \boldsymbol{\theta}^{old})$. KL goes to 0 and so L is maximised:





Next, hold $q(\boldsymbol{Z}) = p(\boldsymbol{Z}|\boldsymbol{X}, \boldsymbol{\theta}^{old})$ and maximise $L(q, \boldsymbol{\theta})$ w.r.t to $\boldsymbol{\theta}$ (find $\boldsymbol{\theta}^{new}$).

Why? Because

$$L(q, \boldsymbol{\theta}^{new}) > L(q, \boldsymbol{\theta}^{old})$$

$$KL(q||p) = \cdots \frac{p(\boldsymbol{Z}|\boldsymbol{X}, \boldsymbol{\theta}^{new})}{p\left(\boldsymbol{Z}\middle|\boldsymbol{X}, \boldsymbol{\theta}^{old}\right)} > 0$$

$$\therefore \log p(\boldsymbol{X}|\boldsymbol{\theta}^{new}) > \log p(\boldsymbol{X}|\boldsymbol{\theta}^{old})$$

In pictures …

Next, hold $q(\boldsymbol{Z}) = p(\boldsymbol{Z}|\boldsymbol{X}, \boldsymbol{\theta}^{old})$ and maximise $L(q, \boldsymbol{\theta})$ w.r.t to $\boldsymbol{\theta}$ (find $\boldsymbol{\theta}^{new}$).

More than zero because now $p \neq q$

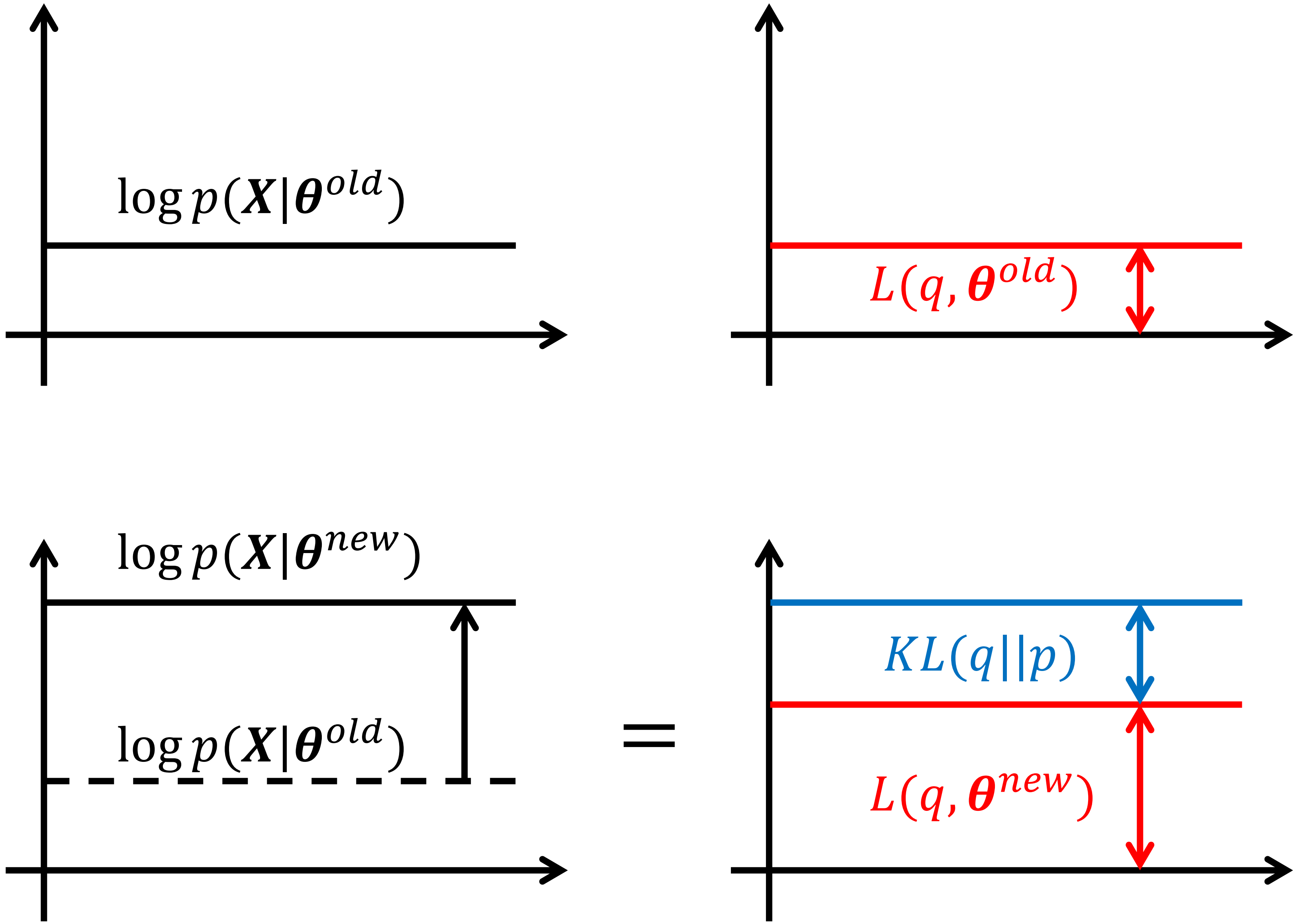
Why? Because

$$L(q, \boldsymbol{\theta}^{new}) > L(q, \boldsymbol{\theta}^{old})$$

$$\boxed{KL(q||p) = \cdots \frac{p(\boldsymbol{Z}|\boldsymbol{X}, \boldsymbol{\theta}^{new})}{p(\boldsymbol{Z}|\boldsymbol{X}, \boldsymbol{\theta}^{old})} > 0}$$

$$\therefore \log p(\boldsymbol{X}|\boldsymbol{\theta}^{new}) > \log p(\boldsymbol{X}|\boldsymbol{\theta}^{old})$$

In pictures …



Set $\boldsymbol{\theta}^{old} = \boldsymbol{\theta}^{new}$ and repeat …

Finally, note that to maximise L w.r.t $\boldsymbol{\theta}$ we are finding the maximum of:

$$\sum_{Z} q(\boldsymbol{Z}) \log\left(\frac{p(\boldsymbol{X},\boldsymbol{Z}|\boldsymbol{\theta})}{q(\boldsymbol{Z})}\right) = \sum_{\boldsymbol{Z}} q(\boldsymbol{Z}) \log p(\boldsymbol{X},\boldsymbol{Z}|\boldsymbol{\theta}) + \text{const.}$$

$$= \sum_{Z} p\left(\boldsymbol{Z}\middle|\boldsymbol{X},\boldsymbol{\theta}^{old}\right) \log p(\boldsymbol{X},\boldsymbol{Z}|\boldsymbol{\theta}) + \text{const.}$$

And so we have the Expectation Maximisation (EM) algorithm:

Expectation: find $Q\left(\boldsymbol{\theta}, \boldsymbol{\theta}^{old}\right) = \sum_{\boldsymbol{Z}} p\left(\boldsymbol{Z} \middle| \boldsymbol{X}, \boldsymbol{\theta}^{old}\right) \log p(\boldsymbol{X}, \boldsymbol{Z} | \boldsymbol{\theta})$

Maximisation: $\boldsymbol{\theta}^{new} = \mathrm{argmax}_{\boldsymbol{\theta}}\, Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{old})$

For a Gaussian Mixture Model it can be shown that this is identical to what 'looked sensible' earlier (**maths is in the appendix**).

Specifically, we find that:

$$\mathrm{E}[z_{nk}] = \frac{N(\boldsymbol{x}_n; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\pi_k}{\sum_{j=1}^{K} N\left(\boldsymbol{x}_n; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j\right)\pi_j}$$

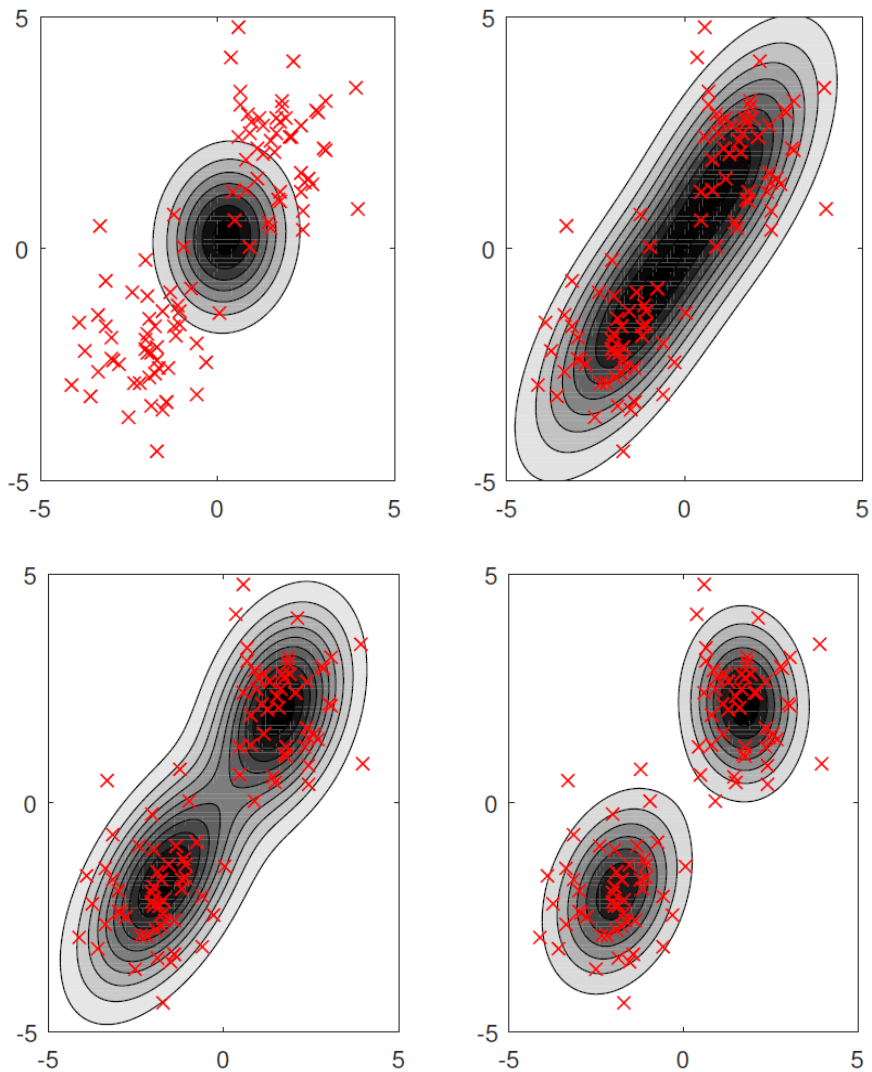
$$\boldsymbol{\mu}_k = \frac{\sum_{n=1}^{N} \mathrm{E}[z_{nk}]\boldsymbol{x}_n}{N_k}$$

$$\boldsymbol{\Sigma}_k = \frac{\sum_{n=1}^{N} \mathrm{E}[z_{nk}](\boldsymbol{x}_n - \boldsymbol{\mu}_k)(\boldsymbol{x}_n - \boldsymbol{\mu}_k)^T}{N_k}$$

$$\pi_k = \frac{N_k}{N}, \qquad N_k = \sum_{n=1}^{N} \mathrm{E}[z_{nk}]$$

where we have used $\mathrm{E}[z_{nk}] \equiv \mathrm{E}_{p(z_{nk}|x_{nk},\boldsymbol{\theta})}[z_{nk}]$ (**all derived in the appendix**).

Example convergence:



Recap

- Unsupervised learning: we have to estimate the parameters of our Gaussians *and* the labels of our data at the same time.

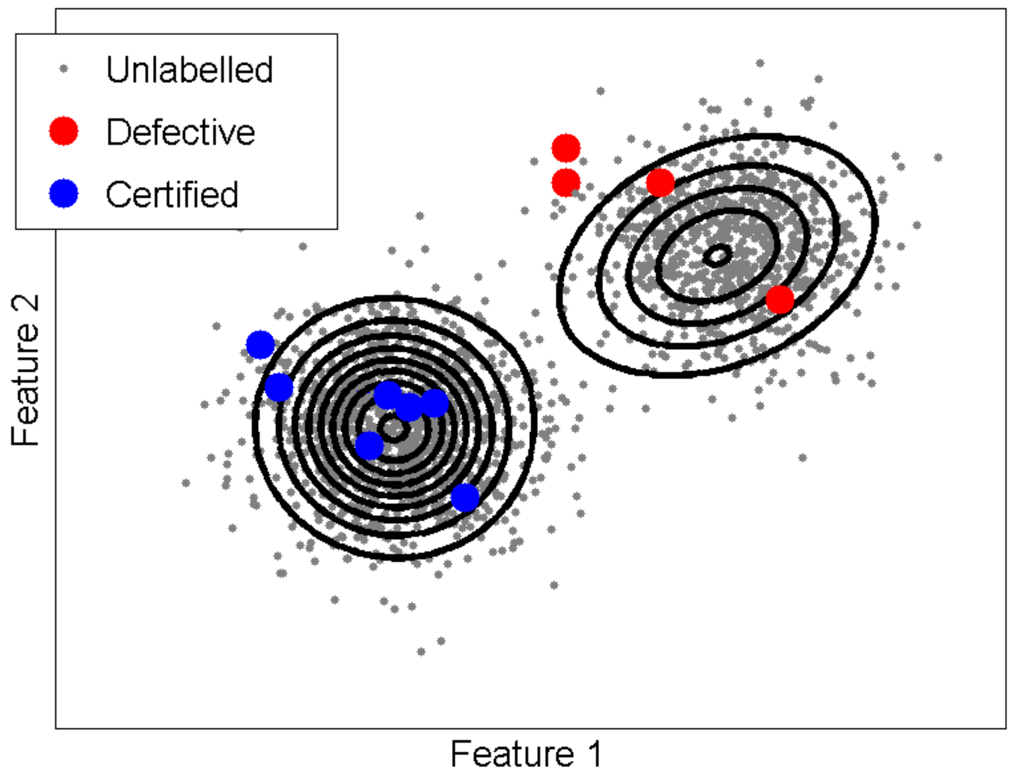
- This is hard, because our best estimate of the parameters will depend on our best estimate of the data labels (and *vice versa*).

- The EM algorithm is a maximum likelihood method for doing this. It is nice because it must increase the likelihood after every iteration.

- So far we have looked at supervised and unsupervised learning. We can now combine these together, in a semi-supervised approach.

Gaussian mixture (semi-supervised learning)

Some data is labelled... But some is not...



Labelled data and **unlabelled data**:

$$\{\boldsymbol{x}_n, \boldsymbol{y}_n\}_{n=1}^{L} \equiv \{\boldsymbol{X}_l, \boldsymbol{Y}\}$$

$$\{\boldsymbol{x}_n, \boldsymbol{z}_n\}_{n=L+1}^{N} \equiv \{\boldsymbol{X}_u, \boldsymbol{Z}\}$$

Say we have 2 clusters again so $\boldsymbol{y}_n = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ or $\boldsymbol{y}_n = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$

Likelihood:

$$p(\boldsymbol{X}_l, \boldsymbol{Y}, \boldsymbol{X}_u, \boldsymbol{Z} | \boldsymbol{\theta}) = p(\boldsymbol{X}_l, \boldsymbol{Y} | \boldsymbol{\theta}) p(\boldsymbol{X}_u, \boldsymbol{Z} | \boldsymbol{\theta})$$

As before, we need to estimate $\boldsymbol{\theta}$ and some latent variables – we will use the EM algorithm again.

Start from an initial parameter estimate, $\boldsymbol{\theta}^{old}$. **Expectation step:**

$$Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{old}) = \sum_{Z} p(\boldsymbol{Z}|\boldsymbol{X}_l, \boldsymbol{Y}, \boldsymbol{X}_u, \boldsymbol{\theta}^{old}) \log p(\boldsymbol{X}_l, \boldsymbol{Y}, \boldsymbol{X}_u, \boldsymbol{Z}|\boldsymbol{\theta})$$

$$= \sum_{Z} p(\boldsymbol{Z}|\boldsymbol{X}_l, \boldsymbol{Y}, \boldsymbol{X}_u, \boldsymbol{\theta}^{old})[\log p(\boldsymbol{X}_l, \boldsymbol{Y}|\boldsymbol{\theta}) + \log p(\boldsymbol{X}_u, \boldsymbol{Z}|\boldsymbol{\theta})]$$

$$= \boxed{\log p(\boldsymbol{X}_l, \boldsymbol{Y}|\boldsymbol{\theta})} + \boxed{\mathrm{E}_Z[\log p(\boldsymbol{X}_u, \boldsymbol{Z}|\boldsymbol{\theta})]}$$

Labelled data **Unlabelled data**

$$Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{old}) = \boxed{\log p(\boldsymbol{X}_l, \boldsymbol{Y}|\boldsymbol{\theta})} + \boxed{\mathrm{E}_Z[\log p(\boldsymbol{X}_u, \boldsymbol{Z}|\boldsymbol{\theta})]}$$

$$\boxed{\log p(\boldsymbol{X}_l, \boldsymbol{Y}|\boldsymbol{\theta}) = \sum_{n=1}^{L} \sum_{k=1}^{K} y_{nk} \log(N(\boldsymbol{x}_n; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\pi_k)}$$

$$\boxed{\mathrm{E}_Z[\log p(\boldsymbol{X}_u, \boldsymbol{Z}|\boldsymbol{\theta})] \sum_{n=l+1}^{N} \sum_{k=1}^{K} \mathrm{E}_Z[z_{nk}] \log(N(\boldsymbol{x}_n; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\pi_k)}$$

Now we have an expression for $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{old})$, we maximise it w.r.t $\boldsymbol{\theta}$ (the Maximisation step).

This is really just a combination of the expressions we derived in the supervised and unsupervised sections of the talk:

$$N_k = \left(\sum_{n=1}^{L} y_{nk} \right) + \left(\sum_{n=L+1}^{N} \mathrm{E}[z_{nk}] \right)$$

$$\boldsymbol{\mu}_k = \frac{1}{N_k} \left(\sum_{n=1}^{L} y_{nk} \boldsymbol{x}_n + \sum_{n=L+1}^{N} \mathrm{E}[z_{nk}] \boldsymbol{x}_n \right)$$

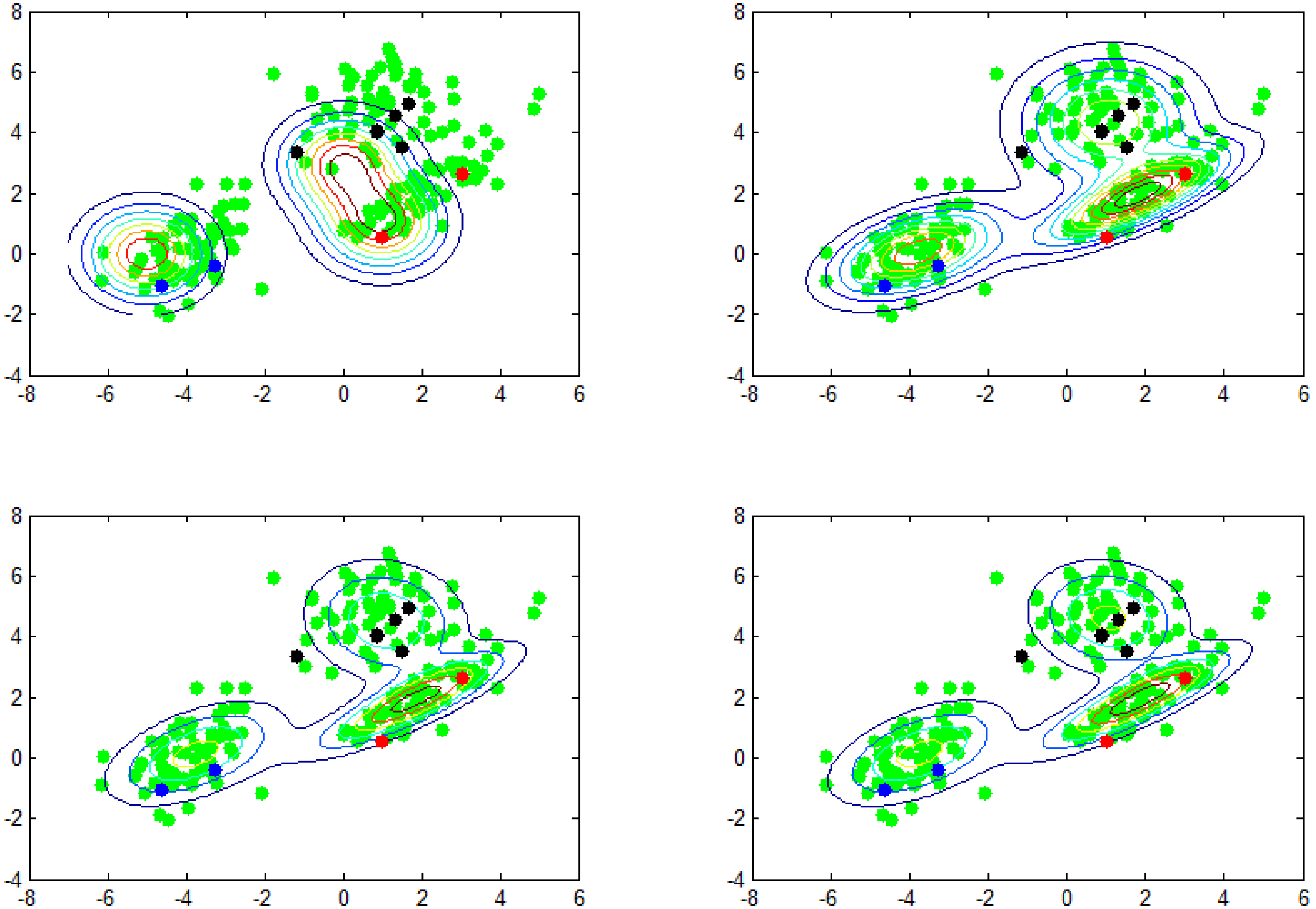
$$\boldsymbol{\Sigma}_k = \frac{1}{N_k}\left(\sum_{n=1}^{L} y_{nk}(\boldsymbol{x}_n - \boldsymbol{\mu}_k)(\boldsymbol{x}_n - \boldsymbol{\mu}_k)^T + \sum_{n=L+1}^{N} \mathrm{E}[z_{nk}](\boldsymbol{x}_n - \boldsymbol{\mu}_k)(\boldsymbol{x}_n - \boldsymbol{\mu}_k)^T\right)$$

$$\pi_k = \frac{N_k}{N}$$

Sanity check: setting $L = 0$ we can recover the expressions for the unsupervised learning case etc.

Note that using only the labelled data to get your first parameter estimates can help convergence.

Easily generalisable to more than 2 clusters:



Conclusions

- Semi-supervised learning is useful when generating unlabelled data is easy and generating labelled data is difficult / expensive.
- Here we have exploited the EM algorithm's ability to estimate parameters + latent variables.
- We just looked at Gaussian Mixture Models – there are many other methods for semi-supervised learning (SVMs etc.)

Conclusions

- Slides will be sent around
- Presentation will (hopefully) be made available on StreamCapture

Thank you for listening!

p.l.green@liverpool.ac.uk

Appendix: EM algorithm applied to the unsupervised Gaussian Mixture Model

Expectation step:

$$\log p(\boldsymbol{X}, \boldsymbol{Z} | \boldsymbol{\theta}) = \sum_{n=1}^{N} \sum_{k=1}^{K} z_{nk} \log(N(\boldsymbol{x}_n; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\pi_k)$$

so we need to find:

$$Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{old}) = \sum_{n=1}^{N} \sum_{k=1}^{K} \mathrm{E}[z_{nk}] \log(N(\boldsymbol{x}_n; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\pi_k)$$

$$\boxed{\mathrm{E}[z_{nk}] \equiv \mathrm{E}_{p(\boldsymbol{z}_{nk} | \boldsymbol{x}_n, \boldsymbol{\theta}^{old})}[z_{nk}]}$$

Noting that $p(\boldsymbol{z}_n|\boldsymbol{x}_n,\boldsymbol{\theta}) \equiv p(z_{nk}=1|\boldsymbol{x}_n,\boldsymbol{\theta})$ we find that:

$$p(\boldsymbol{z}_n|\boldsymbol{x}_n,\boldsymbol{\theta}) = \frac{p(\boldsymbol{x}_n|\boldsymbol{z}_{nk}=1,\boldsymbol{\theta})p(z_{nk}=1|\boldsymbol{\theta})}{\sum_{j=1}^{K} p\left(\boldsymbol{x}_n\middle|z_{nj}=1,\boldsymbol{\theta}\right)p(z_{nj}=1|\boldsymbol{\theta})}$$

$$= \frac{N(\boldsymbol{x}_n;\boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k)\pi_k}{\sum_{j=1}^{K} N\left(\boldsymbol{x}_n;\boldsymbol{\mu}_j,\boldsymbol{\Sigma}_j\right)\pi_j}$$

therefore:

$$\mathrm{E}[z_{nk}] = \sum_{k=1}^{K} z_{nk}p\left(z_{nk}\middle|\boldsymbol{x}_n,\boldsymbol{\theta}^{old}\right) = p(z_{nk}|\boldsymbol{x}_n,\boldsymbol{\theta}^{old})$$

Maximisation step:

$$Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{old}) = \sum_{n=1}^{N} \sum_{k=1}^{K} \mathrm{E}[z_{nk}] \log(N(\boldsymbol{x}_n; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\pi_k)$$

To find a new estimate for $\boldsymbol{\mu}_k$ we set:

$$\frac{\partial Q}{\partial \boldsymbol{\mu}_k} = \sum_{n=1}^{N} \mathrm{E}[z_{nk}] \boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x}_n - \boldsymbol{\mu}_k) = 0$$

$$\Rightarrow \boldsymbol{\mu}_k = \frac{\sum_{n=1}^{N} \mathrm{E}[z_{nk}] \boldsymbol{x}_n}{\sum_{n=1}^{N} \mathrm{E}[z_{nk}]}$$

To find a new estimate of $\boldsymbol{\Sigma}_k$ we set:

$$\frac{\partial Q}{\partial \boldsymbol{\Sigma}_k} = \sum_{n=1}^{N} \mathrm{E}\left[z_{nk}\right] \mathrm{Tr}\left(\boldsymbol{\Sigma}_k^{-1}\left(-\boldsymbol{I} + \boldsymbol{\Sigma}_k^{-1}\boldsymbol{S}_{nk}\right)\right) = 0$$

where $\boldsymbol{S}_{nk} = (\boldsymbol{x}_n - \boldsymbol{\mu}_k)(\boldsymbol{x}_n - \boldsymbol{\mu}_k)^T$. This implies that

$$\sum_{n=1}^{N} \mathrm{E}[z_{nk}]\left(-\boldsymbol{I} + \boldsymbol{\Sigma}_k^{-1}\boldsymbol{S}_{nk}\right) = \mathbf{0}$$

$$\therefore \boldsymbol{\Sigma}_k = \frac{\sum_{n=1}^{N} \mathrm{E}[z_{nk}](\boldsymbol{x}_n - \boldsymbol{\mu}_k)(\boldsymbol{x}_n - \boldsymbol{\mu}_k)^T}{\sum_{n=1}^{N} \mathrm{E}[z_{nk}]}$$

To find a new estimate of π_k we want to maximise Q subject to the constraint that $\sum_k \pi_k = 1$. This problem can be tackled neatly using *Lagrange multipliers*.

In this case the Lagrangian is given by:

$$L = Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{old}) - \lambda \left(\sum_{k=1}^{K} \pi_k - 1 \right)$$

where λ is the Lagrangian multiplier.

Differentiating the Lagrangian w.r.t λ and setting the resulting expression equal to zero gives:

$$\frac{1}{\pi_k}\sum_{n=1}^{N} \mathrm{E}[z_{nk}] - \lambda = 0$$

$$\therefore \sum_{k=1}^{K}\left(\sum_{n=1}^{N} \mathrm{E}[z_{nk}] - \pi_k\lambda\right) = 0 \Rightarrow \lambda = N$$

From this we can show that:

$$\pi_k = \frac{1}{N}\sum_{n=1}^{N} \mathrm{E}[z_{nk}]$$