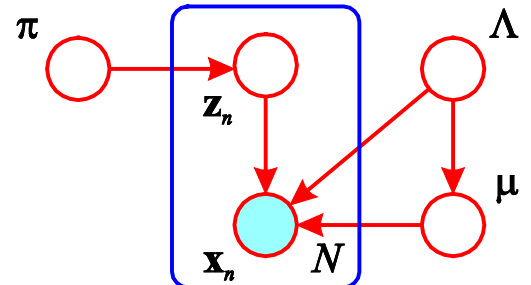
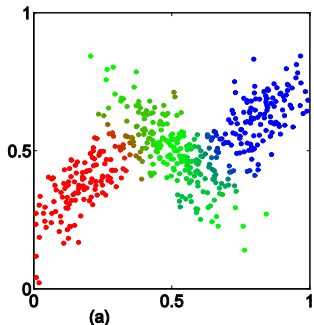


# Latent Variables, Mixture Models and EM

Christopher M. Bishop

Microsoft Research, Cambridge

BCS Summer School  
Exeter, 2003



# Overview.

---

- K-means clustering
- Gaussian mixtures
- Maximum likelihood and EM
- Latent variables: EM revisited
- Bayesian Mixtures of Gaussians

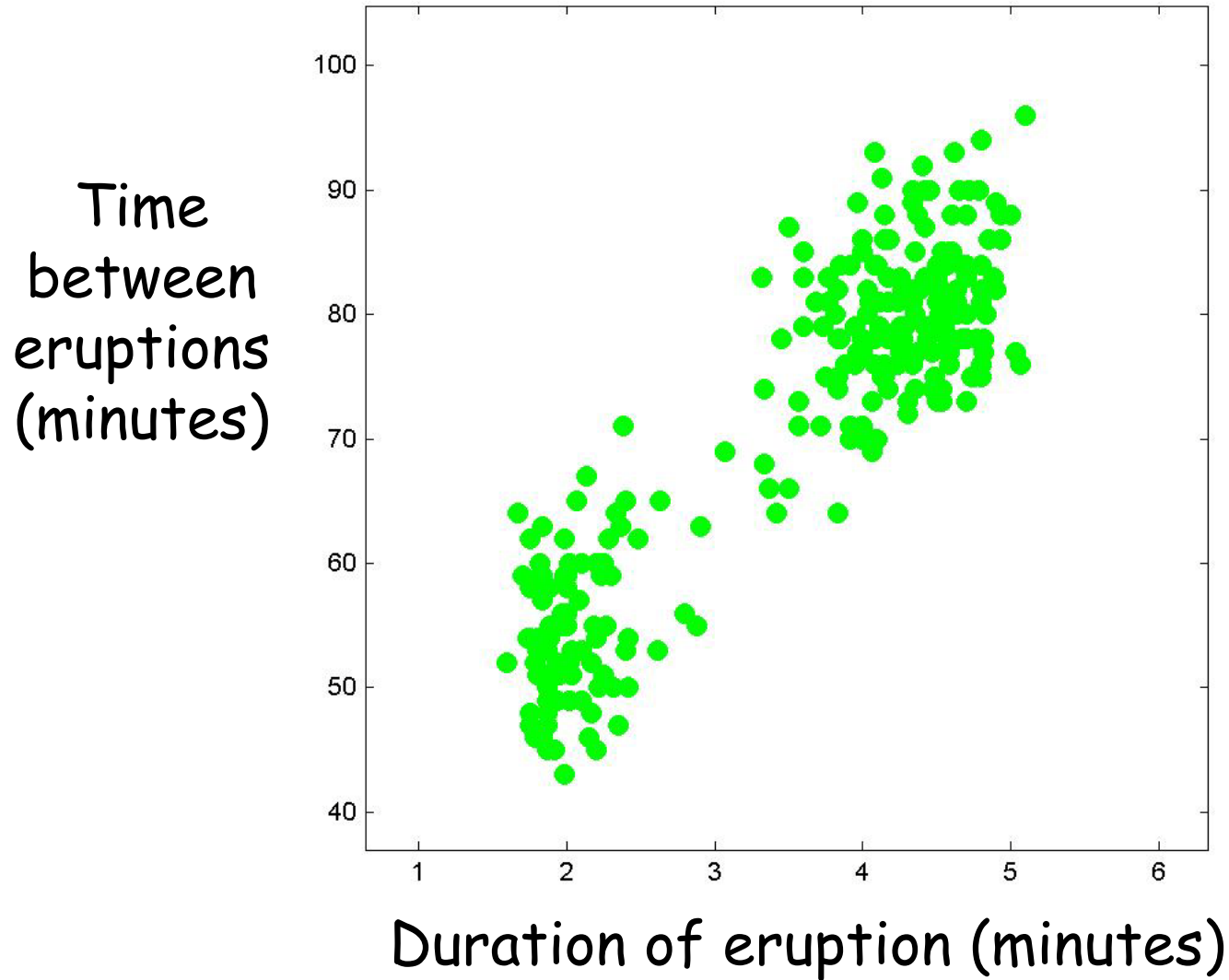
# Old Faithful

---



# Old Faithful Data Set

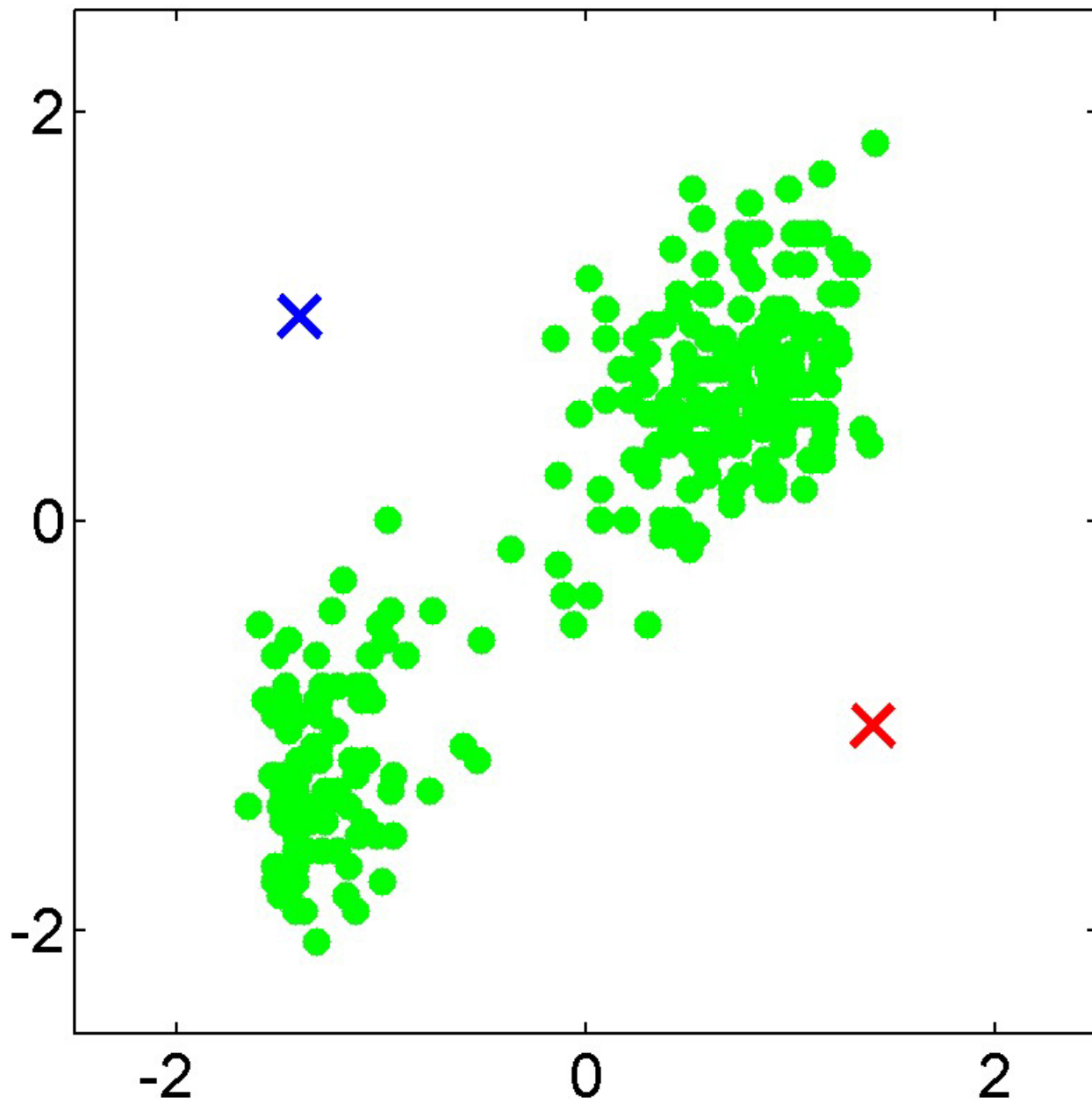
---



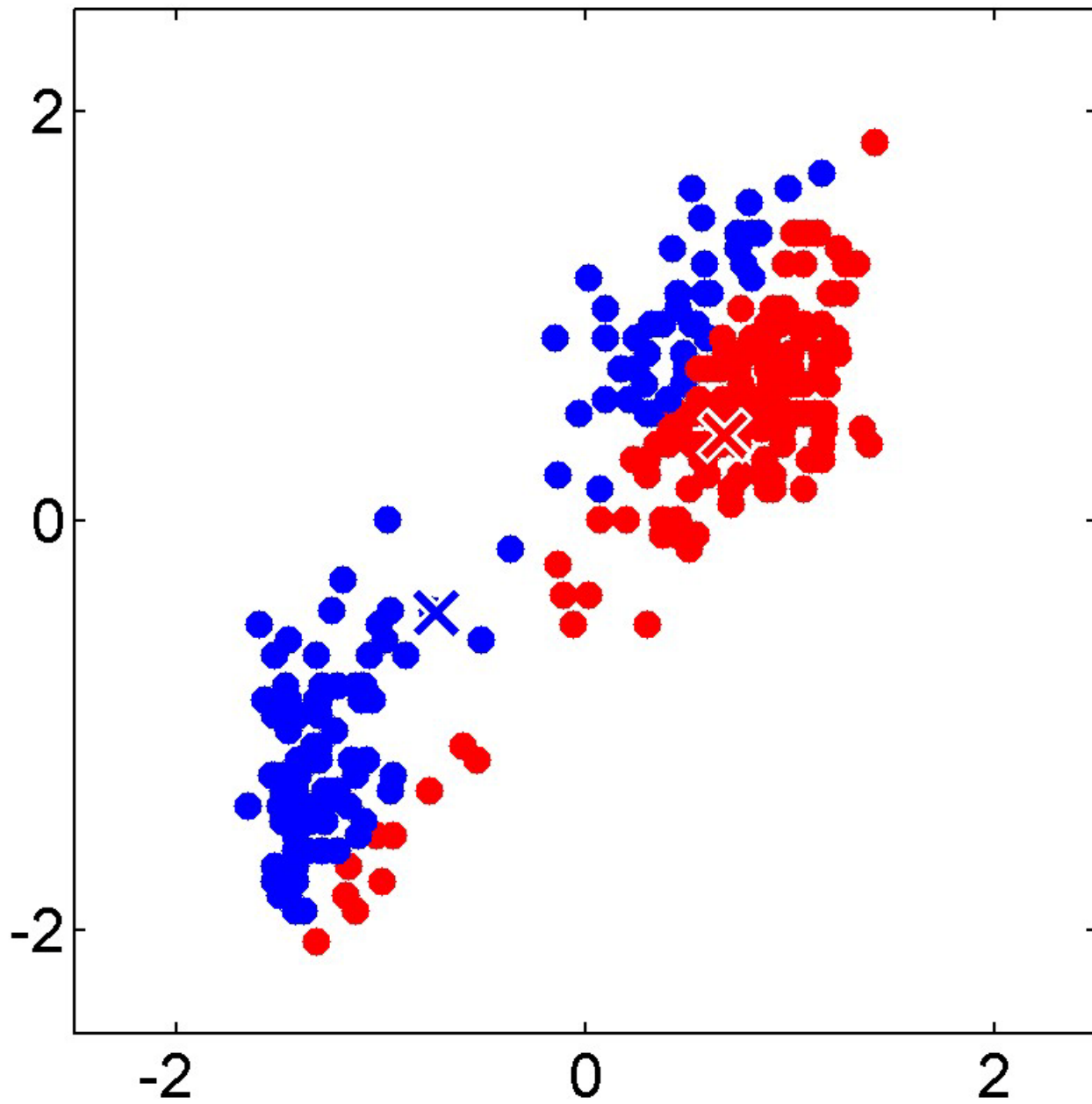
# K-means Algorithm

---

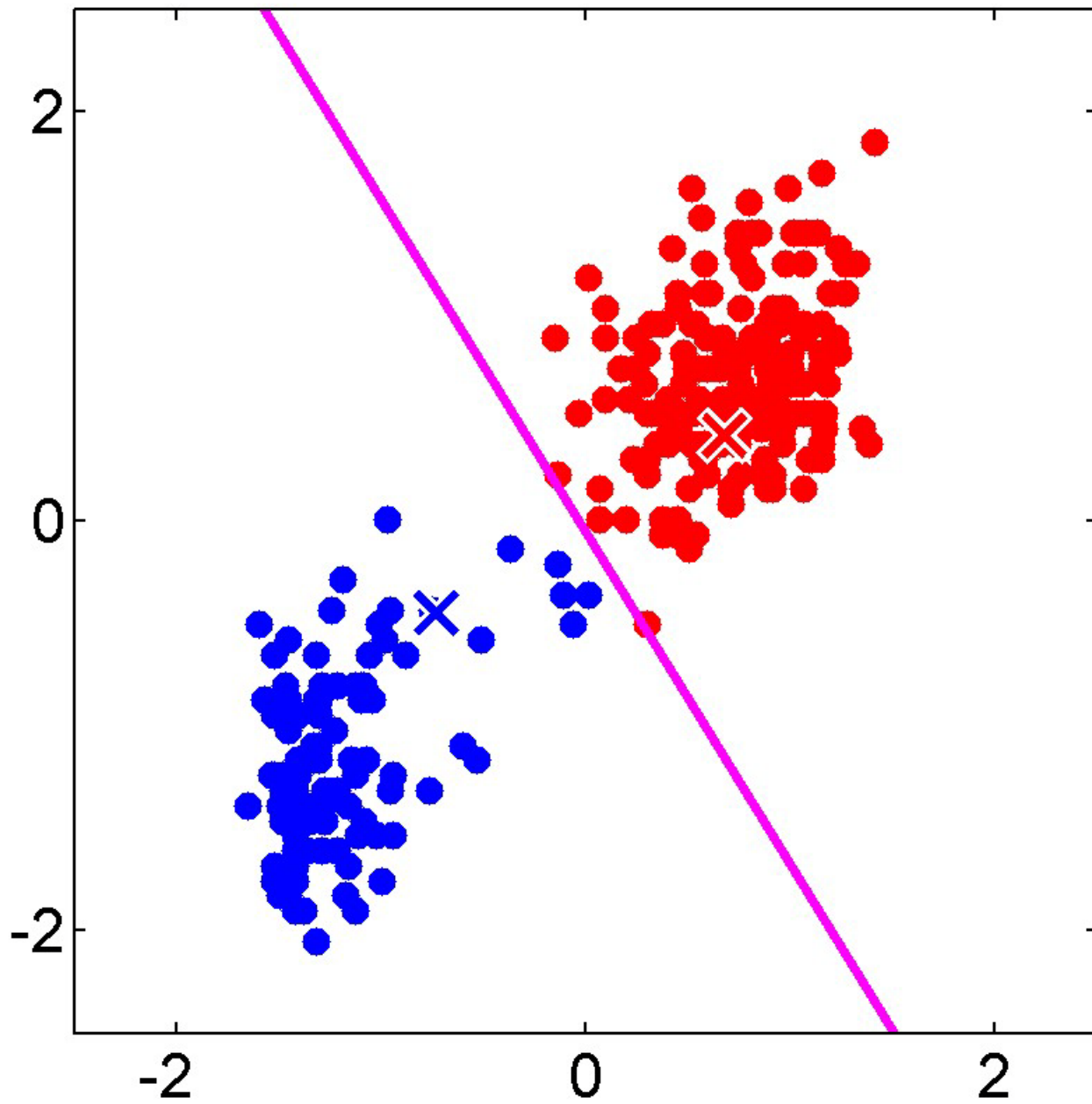
- Goal: represent a data set in terms of  $K$  clusters each of which is summarized by a prototype  $\mu_k$
- Initialize prototypes, then iterate between two phases:
  - E-step: assign each data point to nearest prototype
  - M-step: update prototypes to be the cluster means
- Simplest version is based on Euclidean distance
  - re-scale Old Faithful data

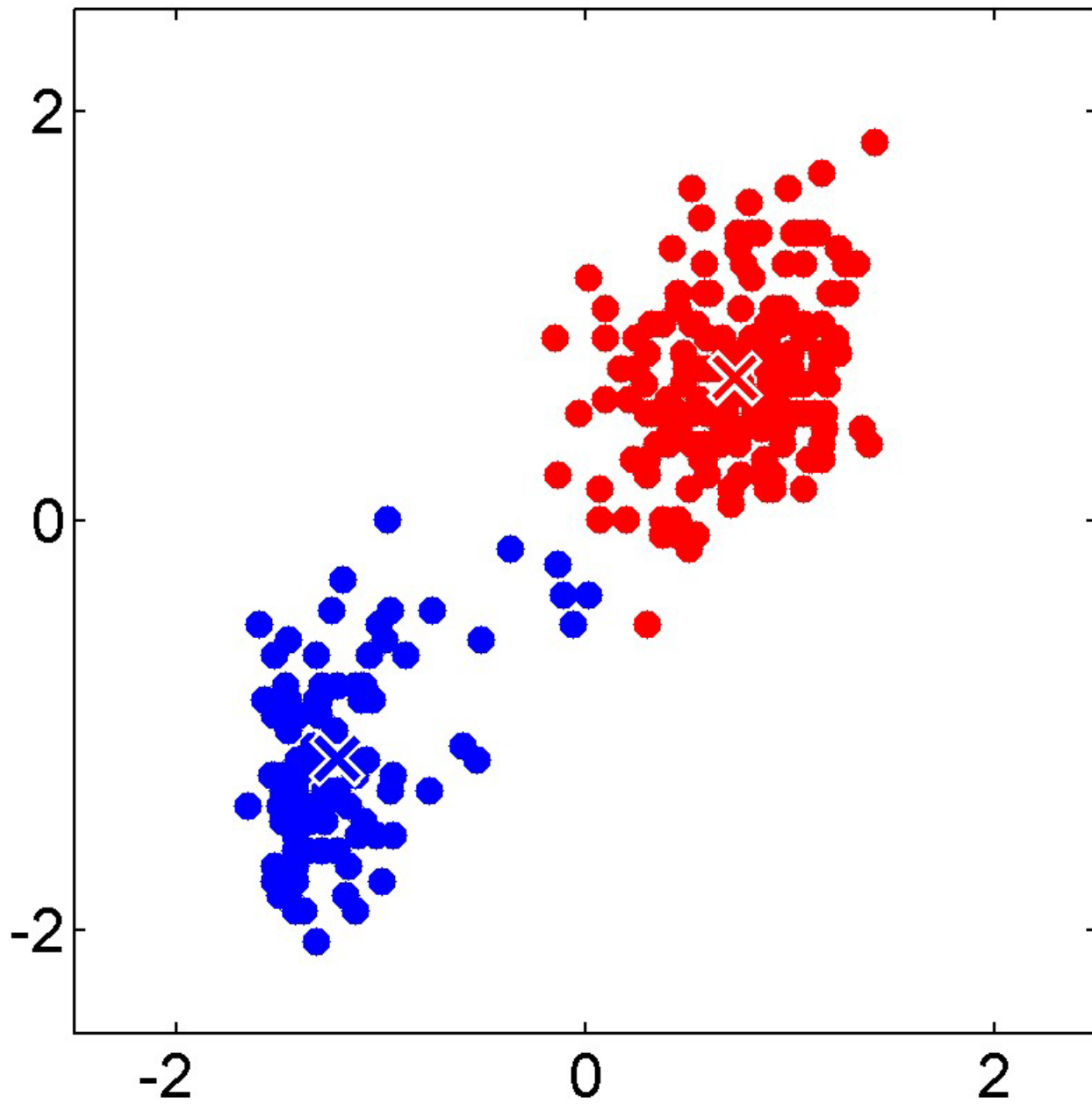


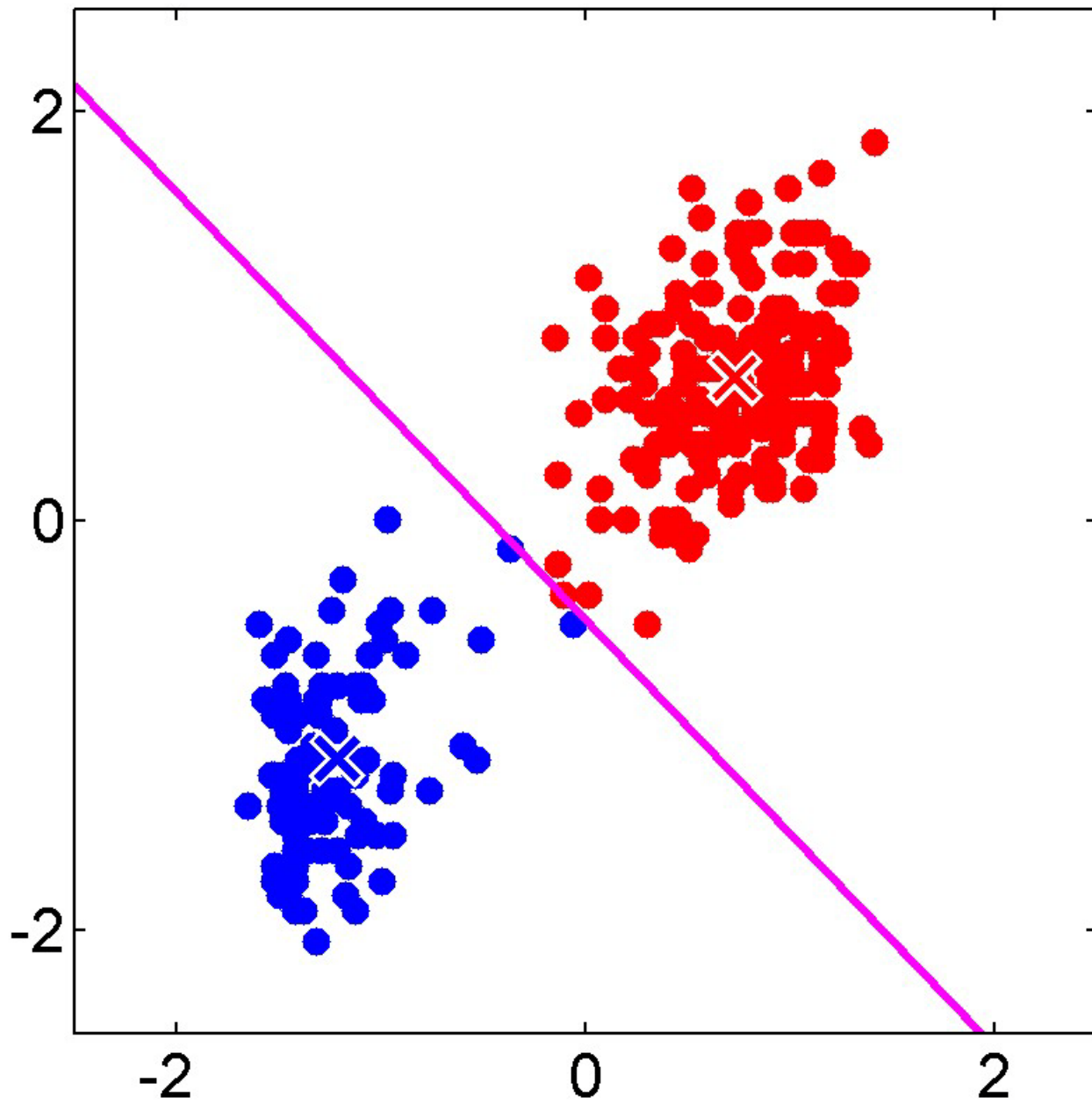


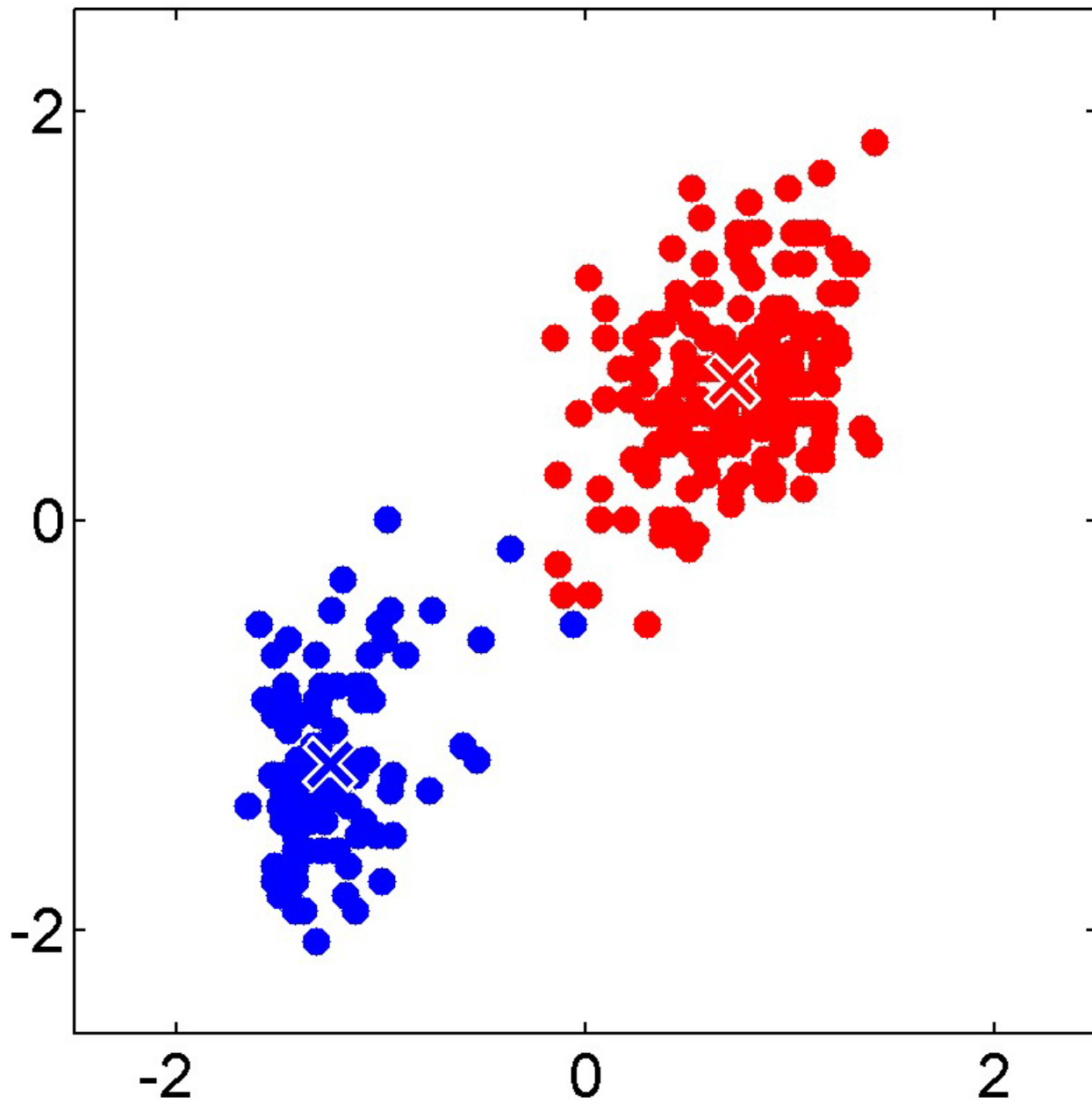


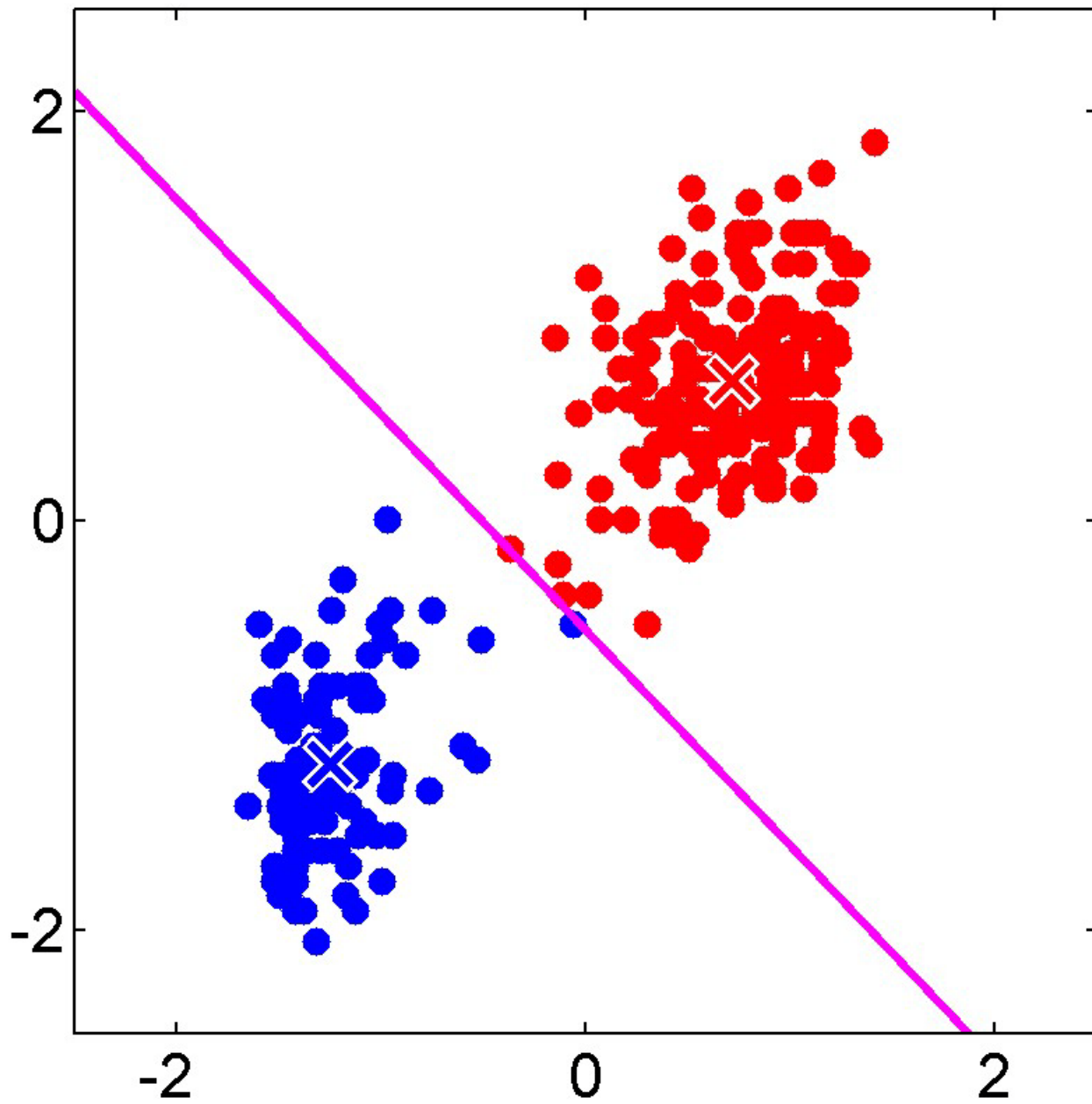


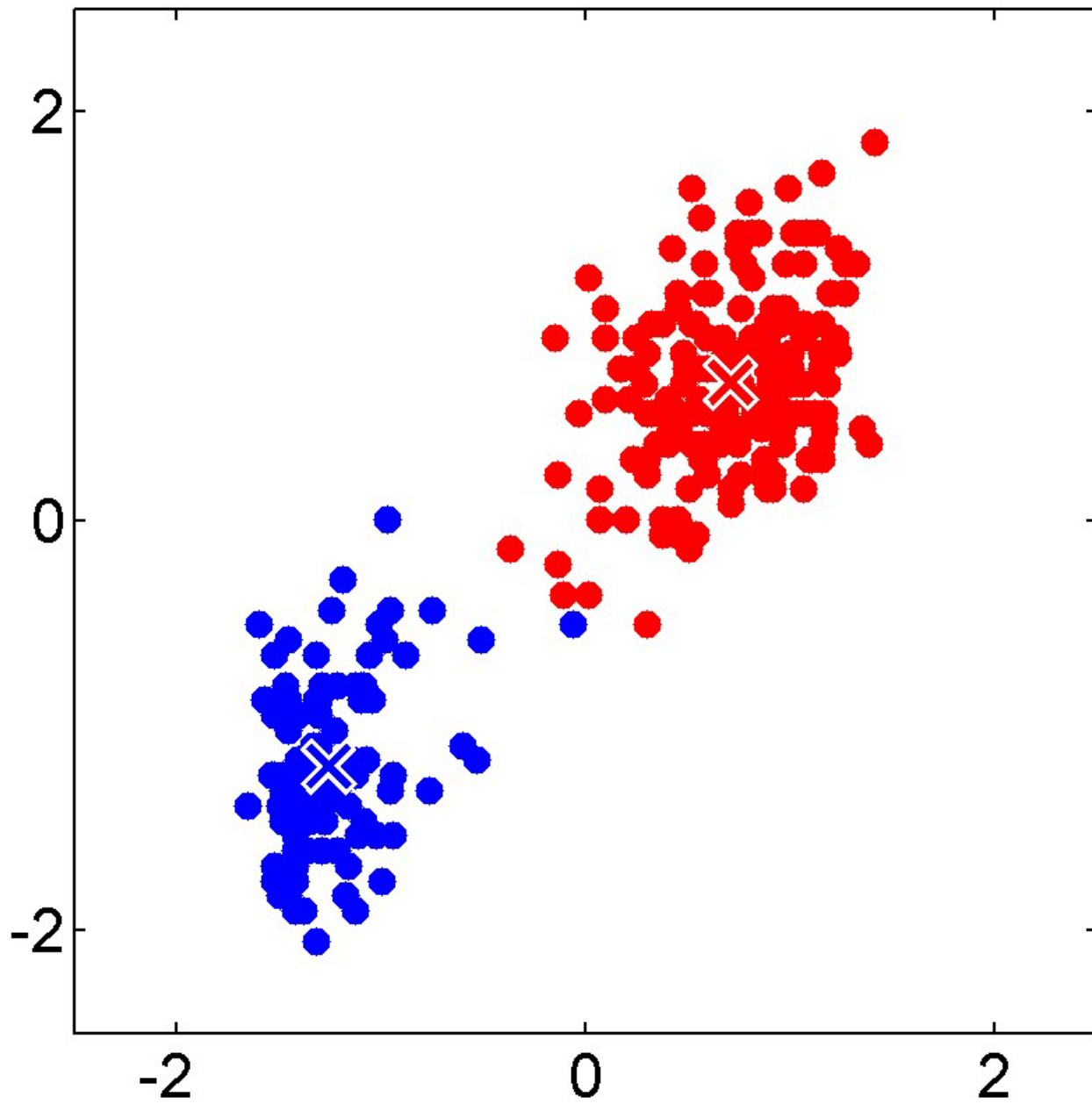












# Responsibilities

---

- *Responsibilities* assign data points to clusters

$$r_{nk} \in \{0, 1\}$$

such that

$$\sum_k r_{nk} = 1$$

- Example: 5 data points and 3 clusters

$$(r_{nk}) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}$$

# K-means Cost Function

---

$$J = \sum_{n=1}^N \sum_{k=1}^K r_{nk} \|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2$$

data

responsibilities

prototypes



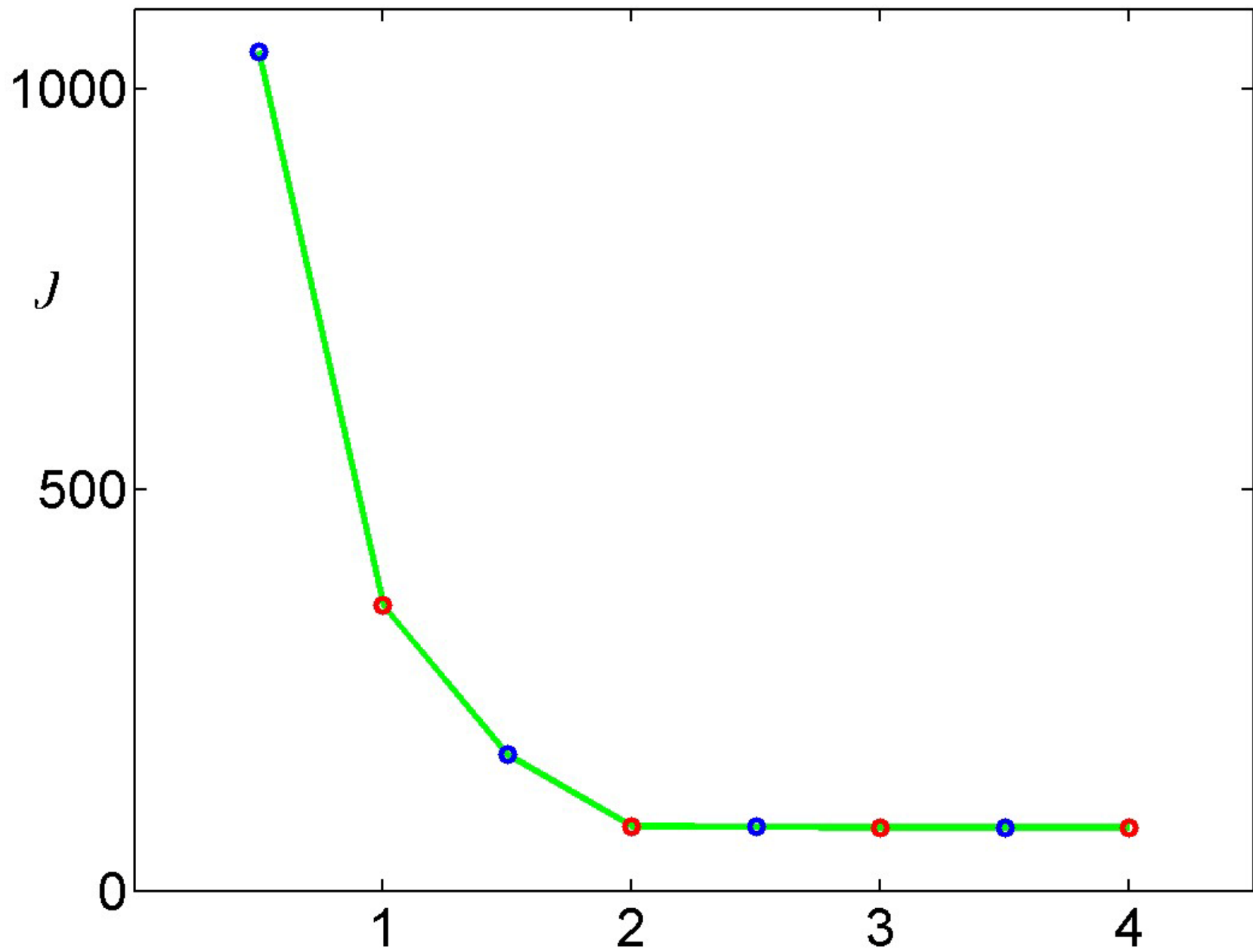
# Minimizing the Cost Function

---

- E-step: minimize  $J$  w.r.t.  $r_{nk}$ 
  - assigns each data point to nearest prototype
- M-step: minimize  $J$  w.r.t.  $\mu_k$ 
  - gives

$$\mu_k = \frac{\sum_n r_{kn} \mathbf{x}_n}{\sum_n r_{kn}}$$

- each prototype set to the mean of points in that cluster
- Convergence guaranteed since there is a finite number of possible settings for the responsibilities



# Limitations of K-means

---

- Hard assignments of data points to clusters – small shift of a data point can flip it to a different cluster
- Not clear how to choose the value of K
- Solution: replace ‘hard’ clustering of K-means with ‘soft’ probabilistic assignments
- Represents the probability distribution of the data as a *Gaussian mixture model*

# The Gaussian Distribution

---

- Multivariate Gaussian

$$\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi|\boldsymbol{\Sigma}|)^{1/2}} \exp \left\{ -\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x} - \boldsymbol{\mu}) \right\}$$

mean                      covariance

- Define precision to be the inverse of the covariance

$$\boldsymbol{\Lambda} = \boldsymbol{\Sigma}^{-1}$$

- In 1-dimension

$$\tau = \frac{1}{\sigma^2}$$

# Likelihood Function

---

- Data set

$$D = \{\mathbf{x}_n\} \quad n = 1, \dots, N$$

- Assume observed data points generated independently

$$p(D|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \prod_{n=1}^N \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}, \boldsymbol{\Sigma})$$

- Viewed as a function of the parameters, this is known as the *likelihood function*

# Maximum Likelihood

---

- Set the parameters by maximizing the likelihood function
- Equivalently maximize the log likelihood

$$\begin{aligned} \ln p(D|\boldsymbol{\mu}, \boldsymbol{\Sigma}) &= -\frac{N}{2} \ln |\boldsymbol{\Sigma}| - \frac{N}{2} \ln(2\pi) \\ &\quad - \frac{1}{2} \sum_{n=1}^N (\mathbf{x}_n - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\mathbf{x}_n - \boldsymbol{\mu}) \end{aligned}$$

# Maximum Likelihood Solution

---

- Maximizing w.r.t. the mean gives the *sample mean*

$$\boldsymbol{\mu}_{\text{ML}} = \frac{1}{N} \sum_{n=1}^N \mathbf{x}_n$$

- Maximizing w.r.t covariance gives the *sample covariance*

$$\boldsymbol{\Sigma}_{\text{ML}} = \frac{1}{N} \sum_{n=1}^N (\mathbf{x}_n - \boldsymbol{\mu}_{\text{ML}})(\mathbf{x}_n - \boldsymbol{\mu}_{\text{ML}})^{\text{T}}$$

# Gaussian Mixtures

---

- Linear super-position of Gaussians

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

- Normalization and positivity require

$$\sum_{k=1}^K \pi_k = 1 \quad 0 \leq \pi_k \leq 1$$

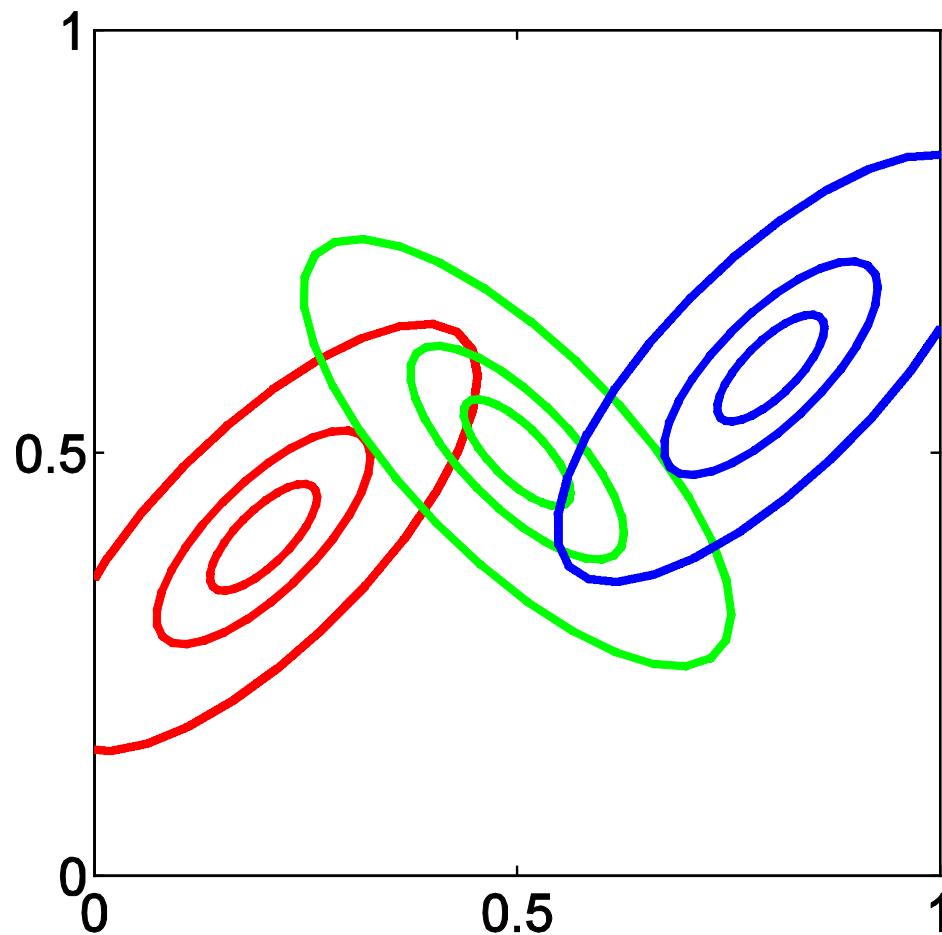
- Can interpret the mixing coefficients as prior probabilities

$$p(\mathbf{x}) = \sum_{k=1}^K p(k) p(\mathbf{x} | k)$$



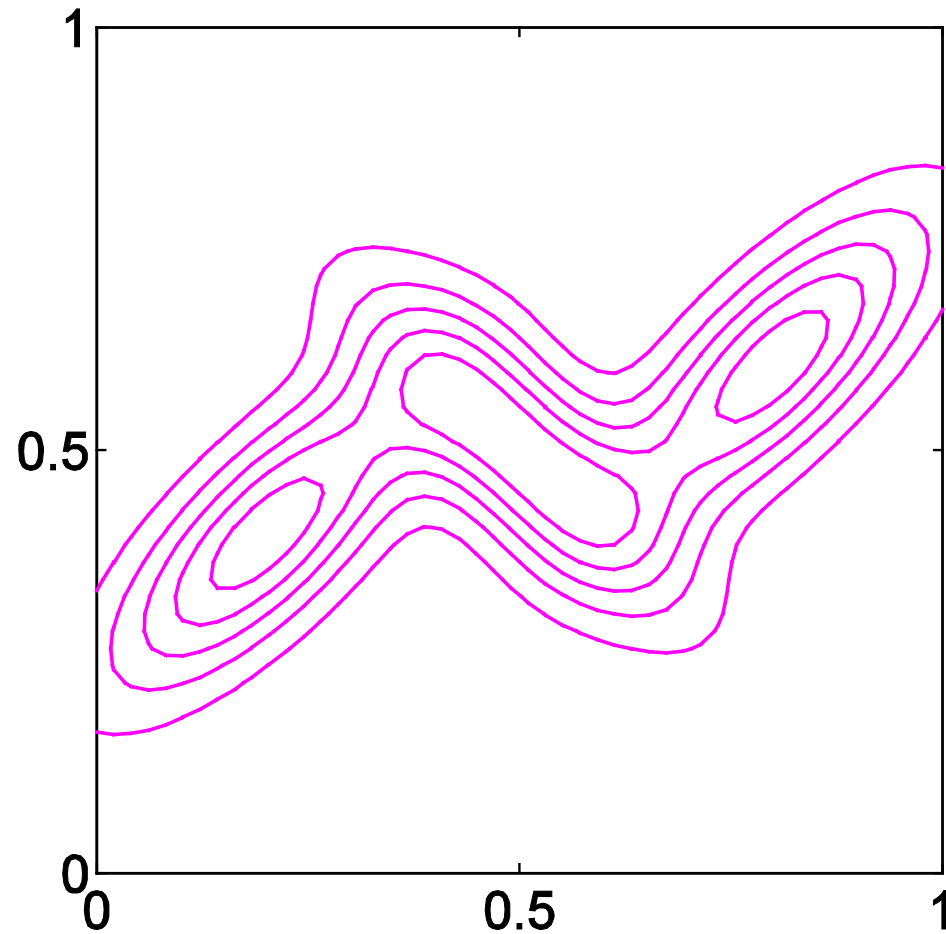
# Example: Mixture of 3 Gaussians

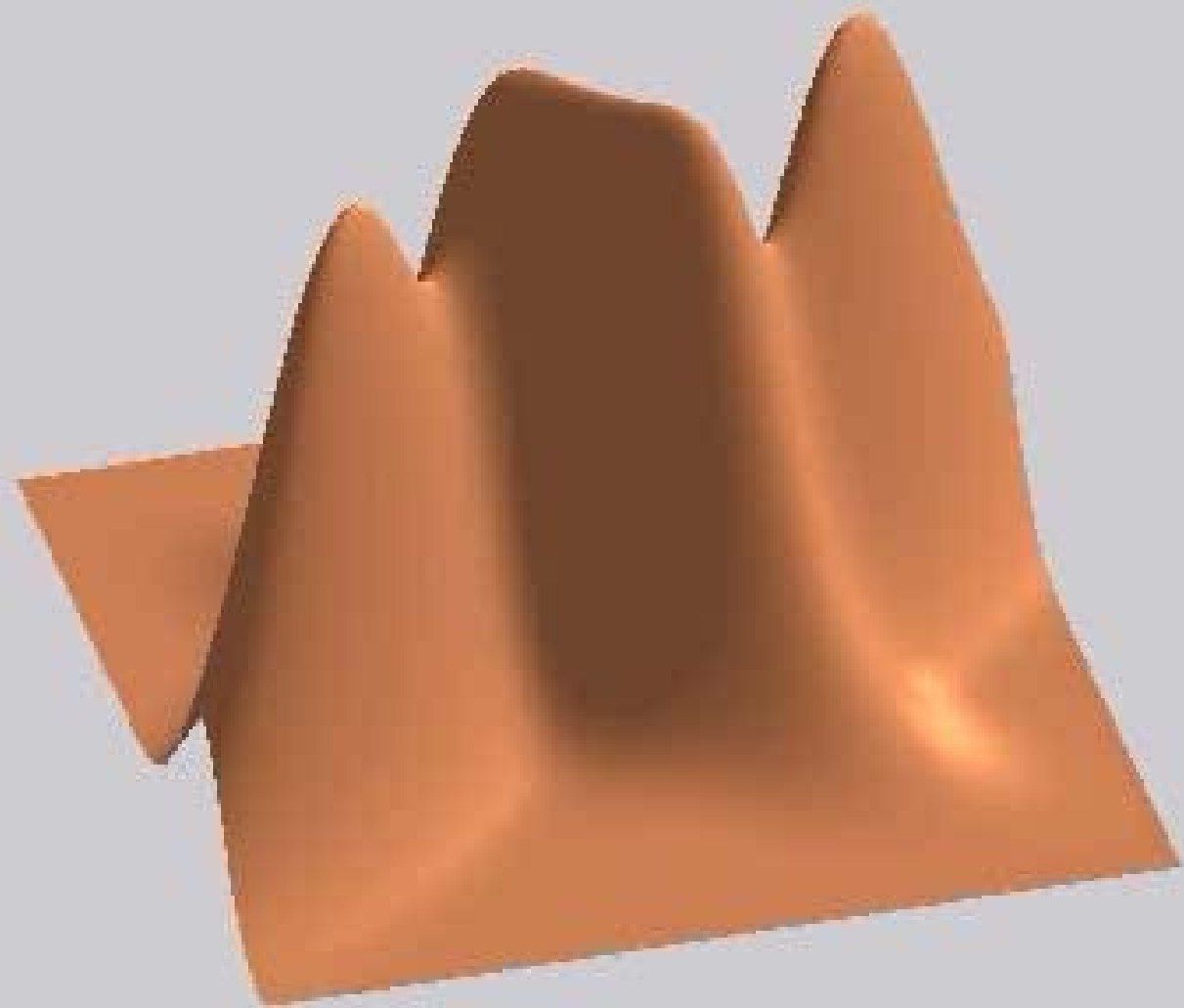
---



# Contours of Probability Distribution

---





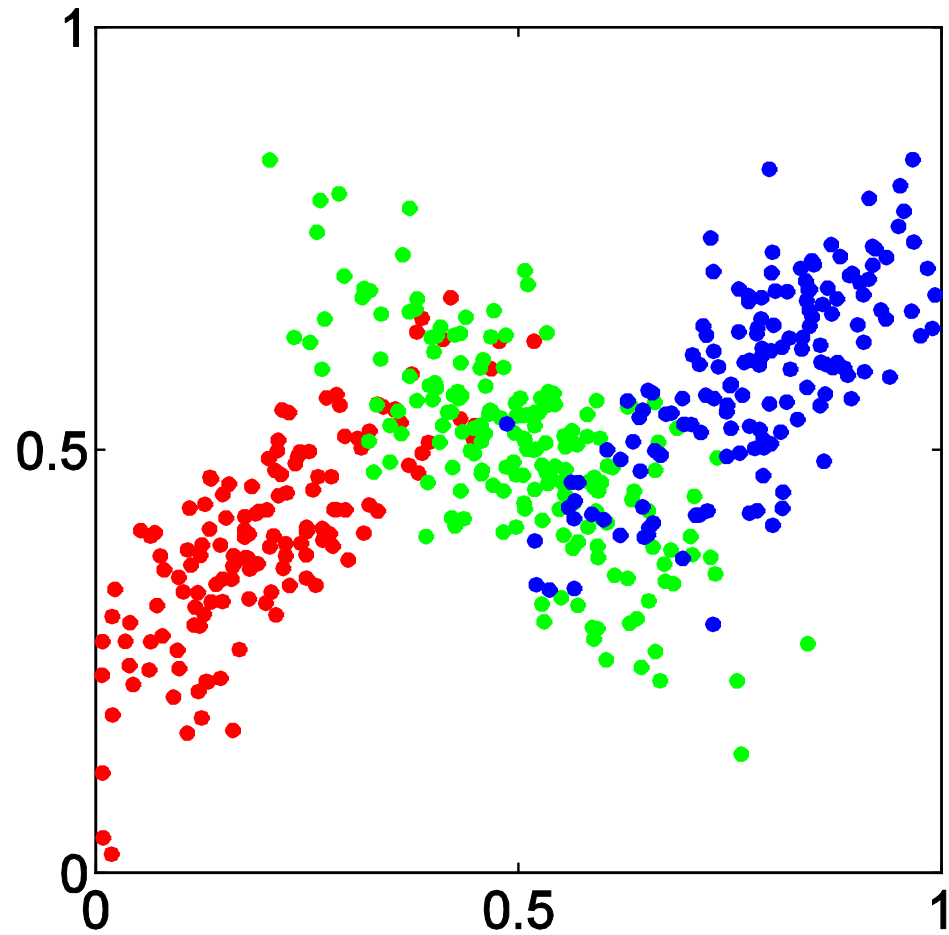
# Sampling from the Gaussian

---

- To generate a data point:
  - first pick one of the components with probability  $\pi_k$
  - then draw a sample  $\mathbf{x}_n$  from that component
- Repeat these two steps for each new data point

# Synthetic Data Set

---



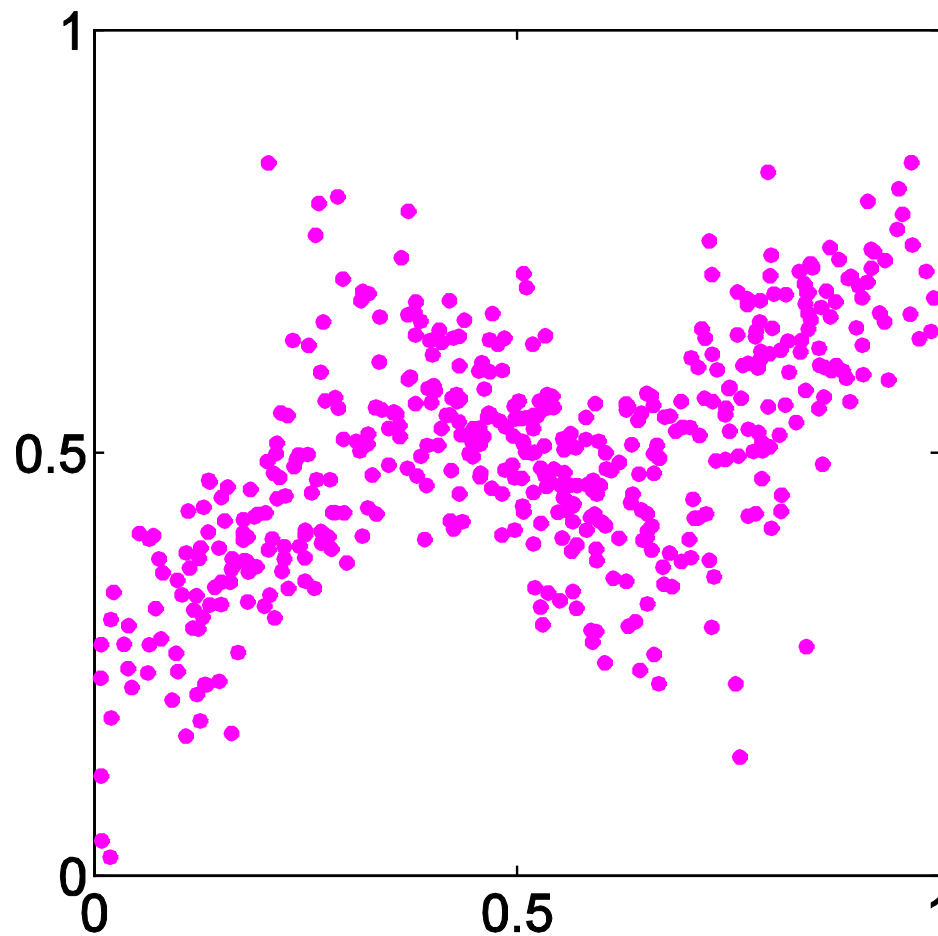
# Fitting the Gaussian Mixture

---

- We wish to invert this process – given the data set, find the corresponding parameters:
  - mixing coefficients
  - means
  - covariances
- If we knew which component generated each data point, the maximum likelihood solution would involve fitting each component to the corresponding cluster
- Problem: the data set is unlabelled
- We shall refer to the labels as *latent* (= hidden) variables

# Synthetic Data Set Without Labels

---



# Posterior Probabilities

---

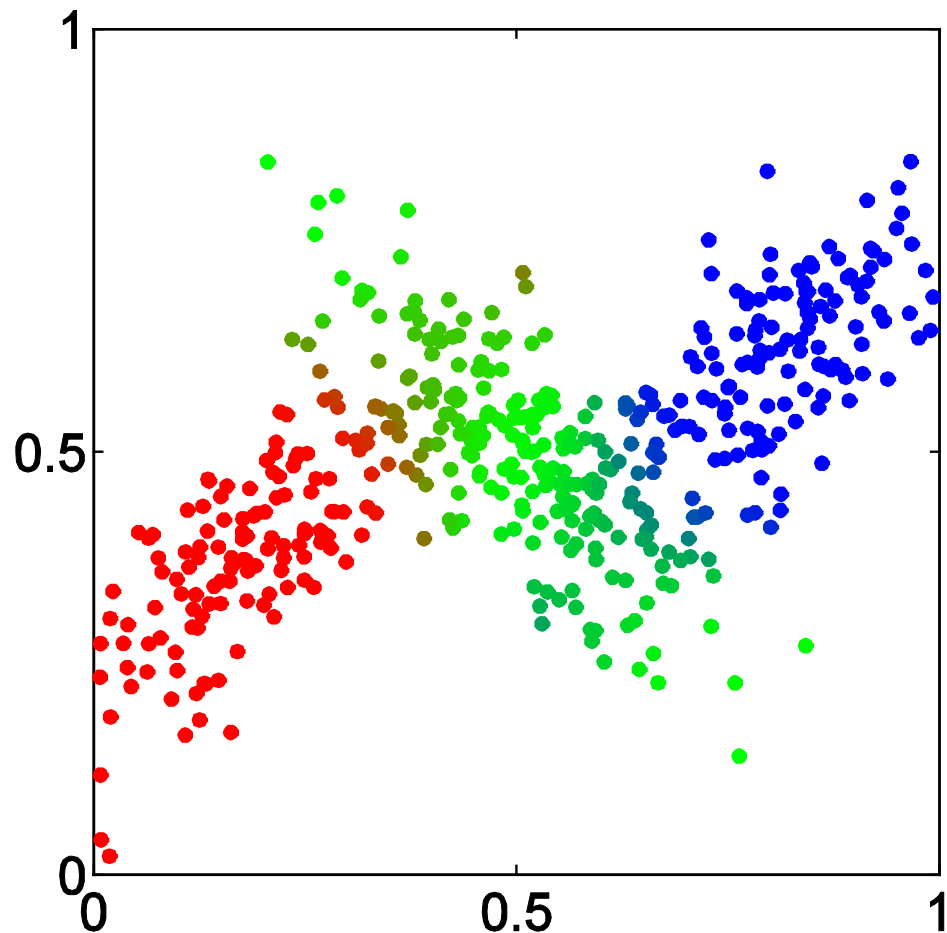
- We can think of the mixing coefficients as prior probabilities for the components
- For a given value of  $\mathbf{x}$  we can evaluate the corresponding posterior probabilities, called *responsibilities*
- These are given from Bayes' theorem by

$$\begin{aligned}\gamma_k(\mathbf{x}) \equiv p(k|\mathbf{x}) &= \frac{p(k)p(\mathbf{x}|k)}{p(\mathbf{x})} \\ &= \frac{\pi_k \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^K \pi_j \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}\end{aligned}$$



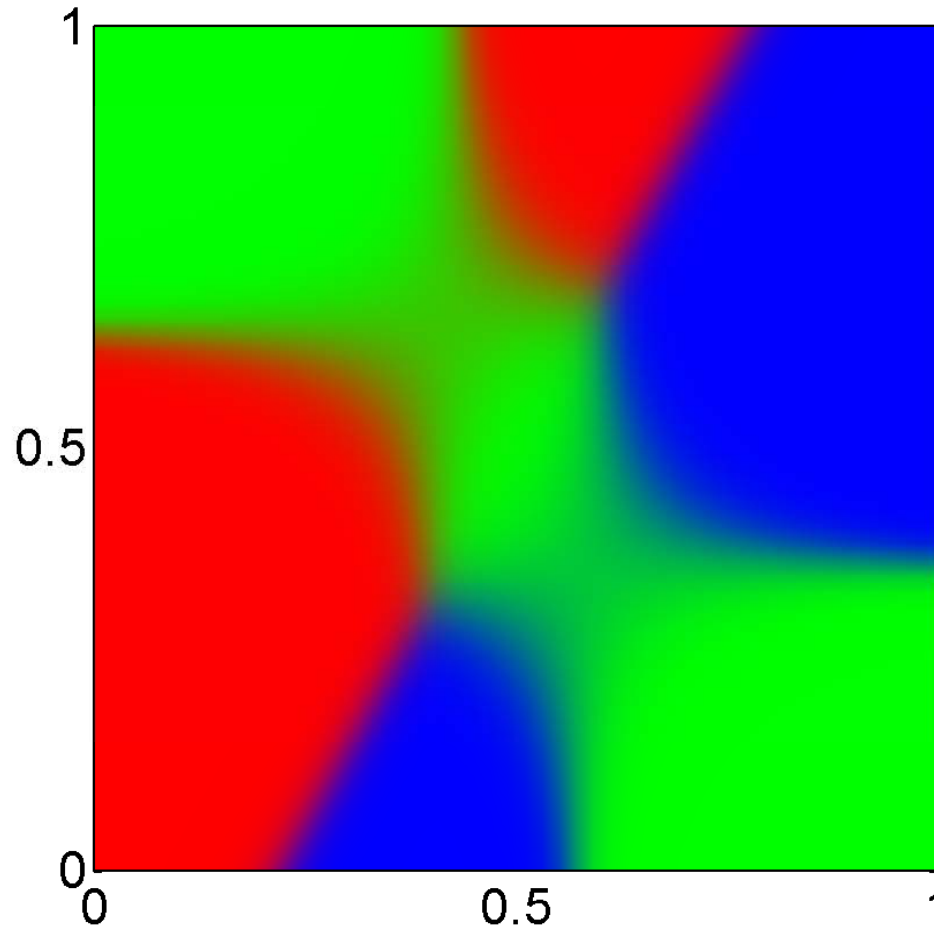
# Posterior Probabilities (colour coded)

---



# Posterior Probability Map

---



# Maximum Likelihood for the GMM

---

- The log likelihood function takes the form

$$\ln p(D|\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \sum_{n=1}^N \ln \left\{ \sum_{k=1}^K \pi_k \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right\}$$

- Note: sum over components appears *inside* the log
- There is no closed form solution for maximum likelihood

# Problems and Solutions

---

- How to maximize the log likelihood
  - solved by expectation-maximization (EM) algorithm
- How to avoid singularities in the likelihood function
  - solved by a Bayesian treatment
- How to choose number  $K$  of components
  - also solved by a Bayesian treatment

# EM Algorithm – Informal Derivation

---

- Let us proceed by simply differentiating the log likelihood
- Setting derivative with respect to  $\mu_j$  equal to zero gives

$$- \sum_{n=1}^N \frac{\pi_j \mathcal{N}(\mathbf{x}_n | \mu_j, \Sigma_j)}{\underbrace{\sum_k \pi_k \mathcal{N}(\mathbf{x}_n | \mu_k, \Sigma_k)}_{\gamma_j(\mathbf{x}_n)}} \Sigma_j^{-1} (\mathbf{x}_n - \mu_j) = 0$$

giving

$$\mu_j = \frac{\sum_{n=1}^N \gamma_j(\mathbf{x}_n) \mathbf{x}_n}{\sum_{n=1}^N \gamma_j(\mathbf{x}_n)}$$

which is simply the weighted mean of the data

# EM Algorithm – Informal Derivation

---

- Similarly for the covariances

$$\Sigma_j = \frac{\sum_{n=1}^N \gamma_j(\mathbf{x}_n) (\mathbf{x}_n - \boldsymbol{\mu}_j) (\mathbf{x}_n - \boldsymbol{\mu}_j)^\top}{\sum_{n=1}^N \gamma_j(\mathbf{x}_n)}$$

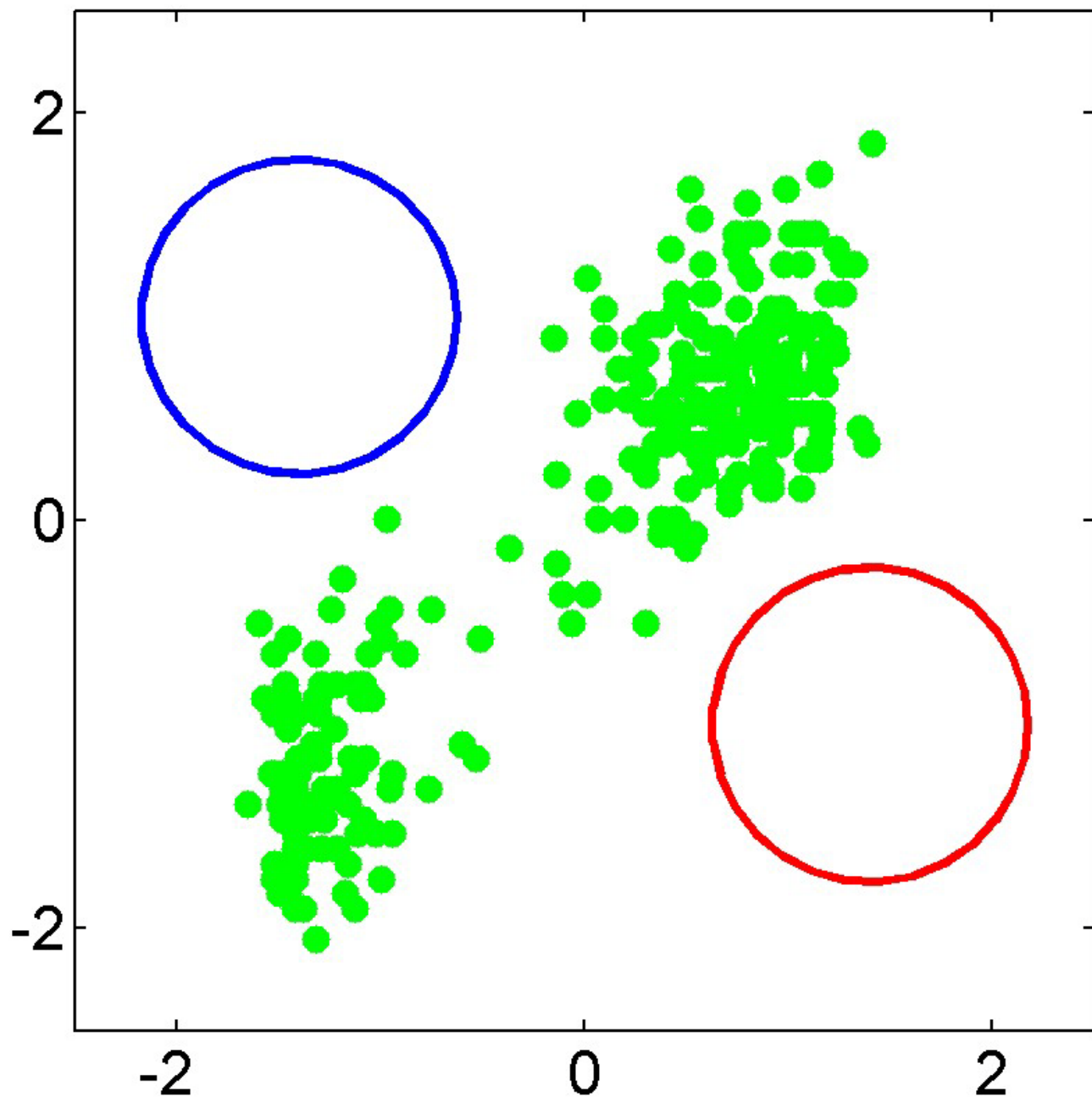
- For mixing coefficients use a Lagrange multiplier to give

$$\pi_j = \frac{1}{N} \sum_{n=1}^N \gamma_j(\mathbf{x}_n)$$

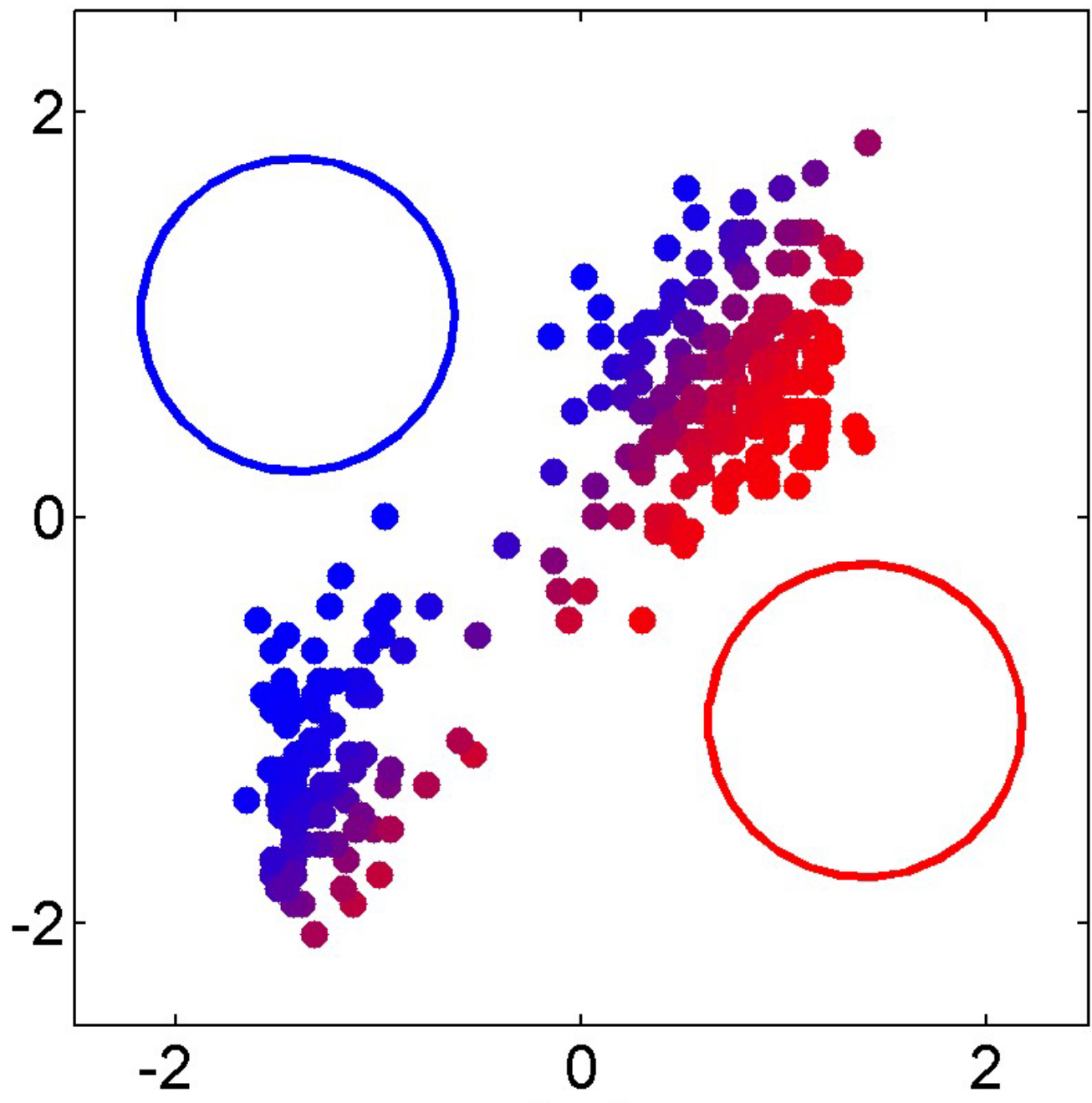
# EM Algorithm – Informal Derivation

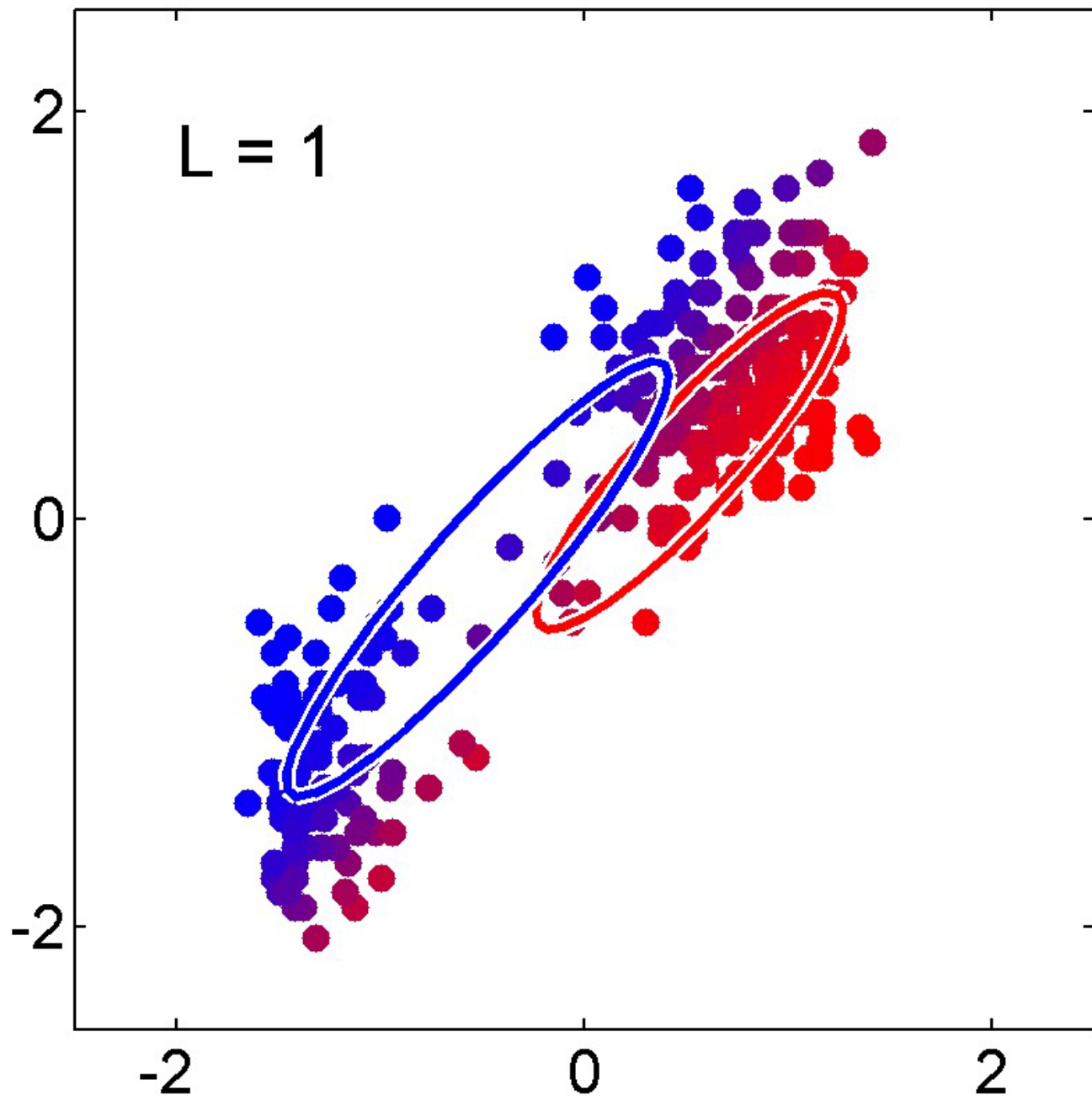
---

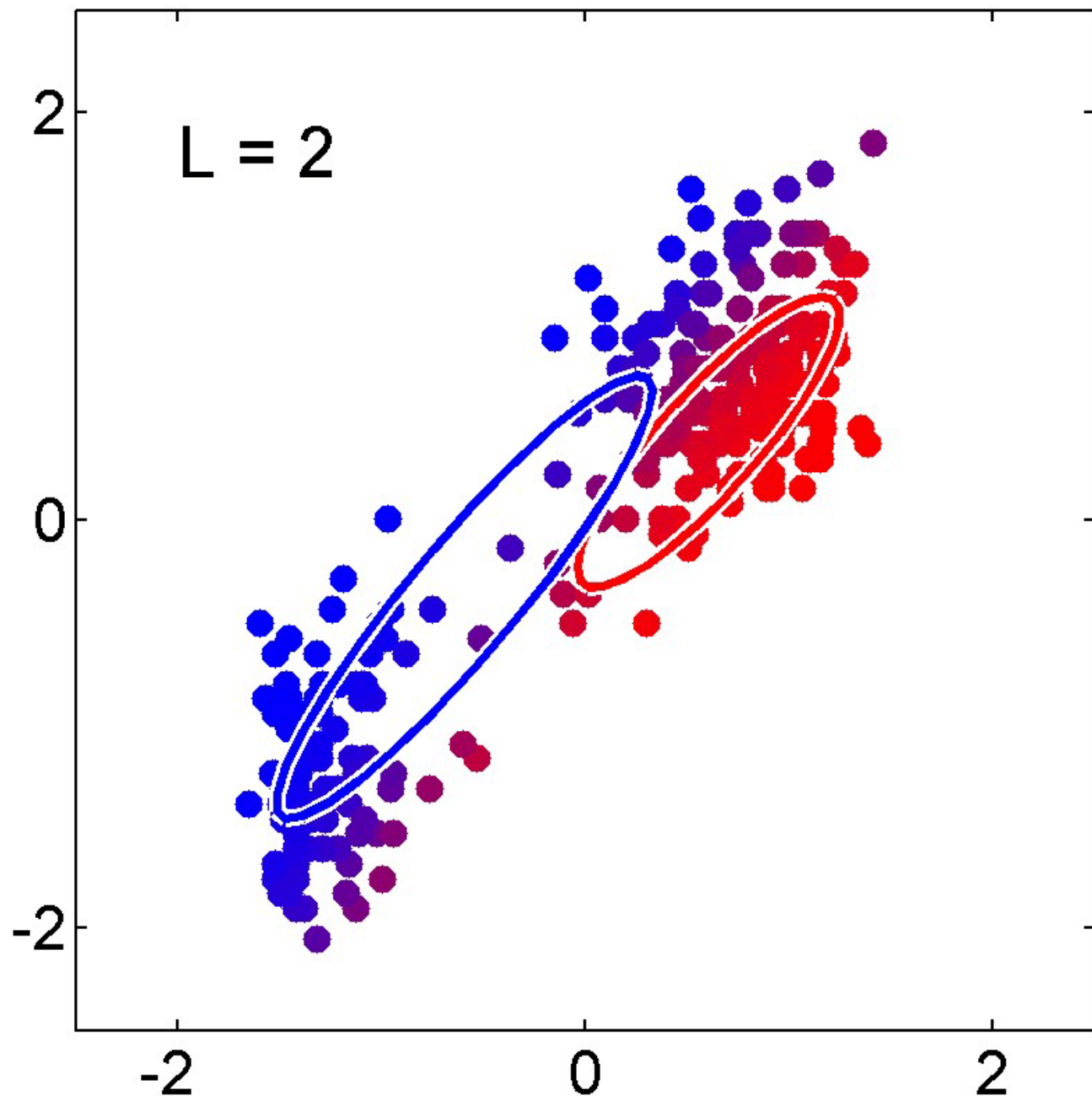
- The solutions are not closed form since they are coupled
- Suggests an iterative scheme for solving them:
  - Make initial guesses for the parameters
  - Alternate between the following two stages:
    1. E-step: evaluate responsibilities
    2. M-step: update parameters using ML results

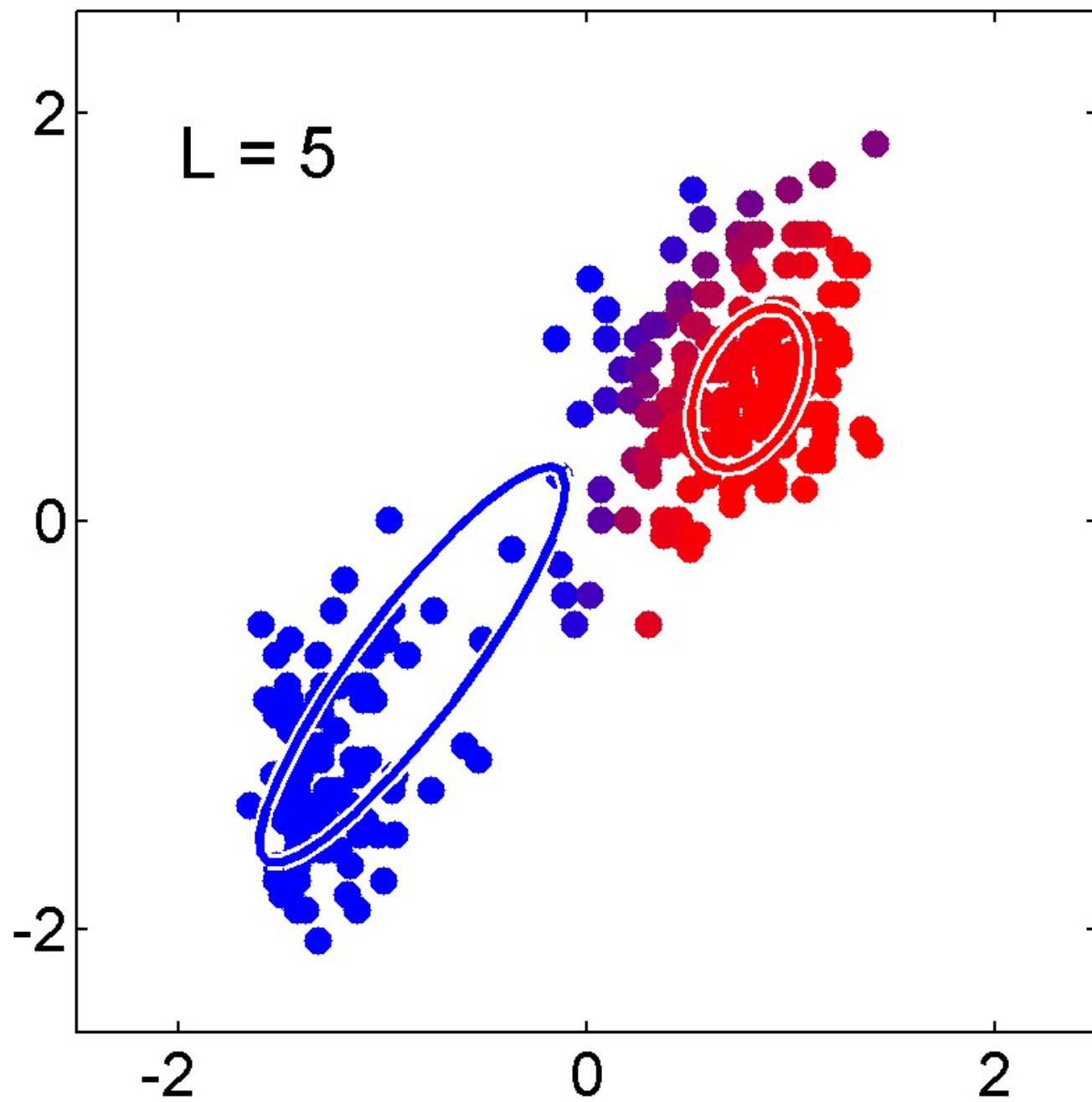


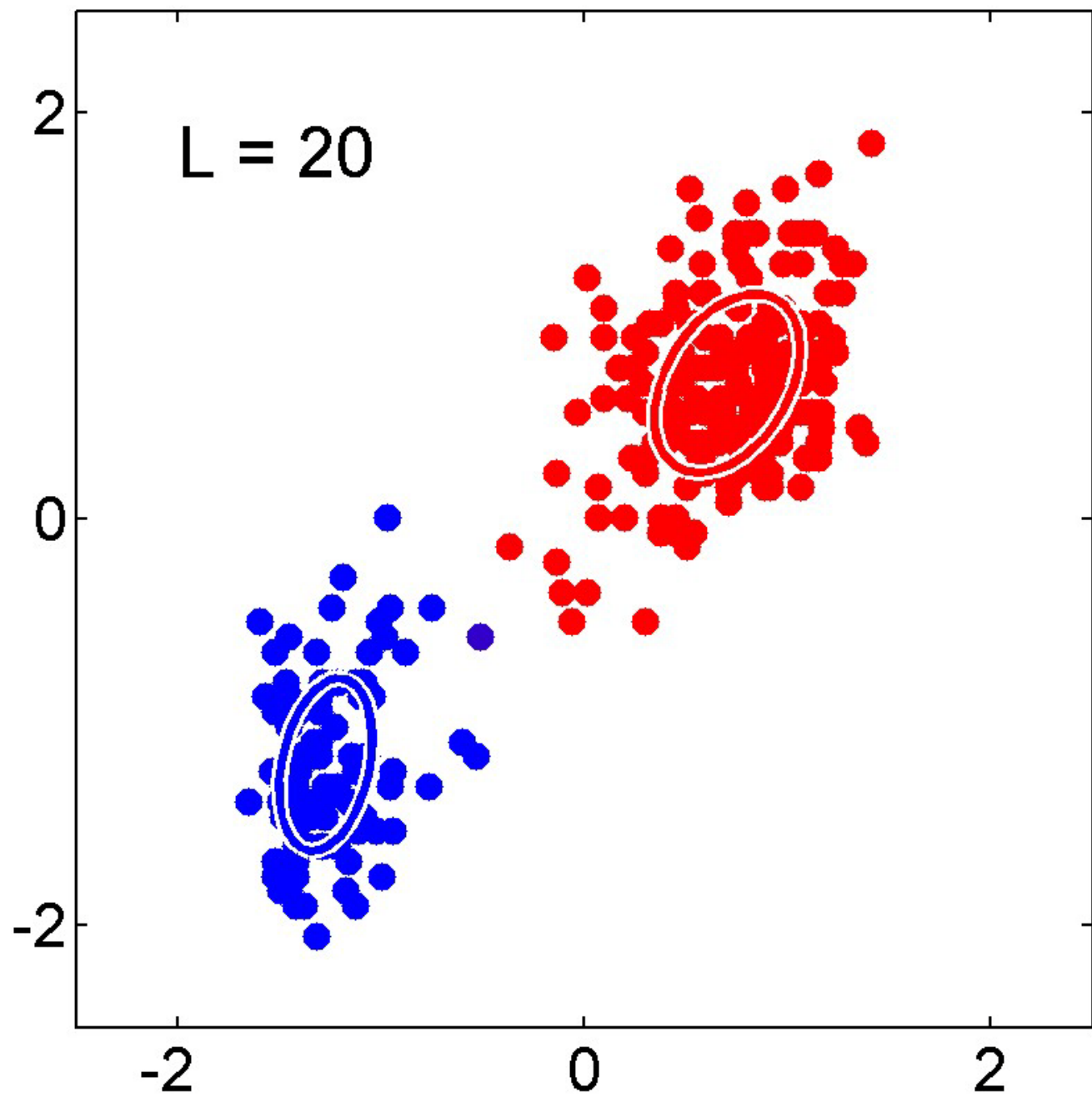












# EM – Latent Variable Viewpoint

---

- Binary latent variables  $\mathbf{z} = \{z_{kn}\}$  describing which component generated each data point
- Conditional distribution of observed variable

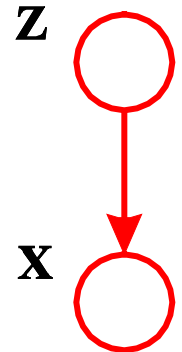
$$p(\mathbf{x}|\mathbf{z}) = \prod_{k=1}^K \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Lambda}_k)^{z_k}$$

- Prior distribution of latent variables

$$p(\mathbf{z}) = \prod_{k=1}^K \pi_k^{z_k}$$

- Marginalizing over the latent variables we obtain

$$p(\mathbf{x}) = \sum_{k=1}^K \pi_k \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Lambda}_k)$$



# Expected Value of Latent Variable

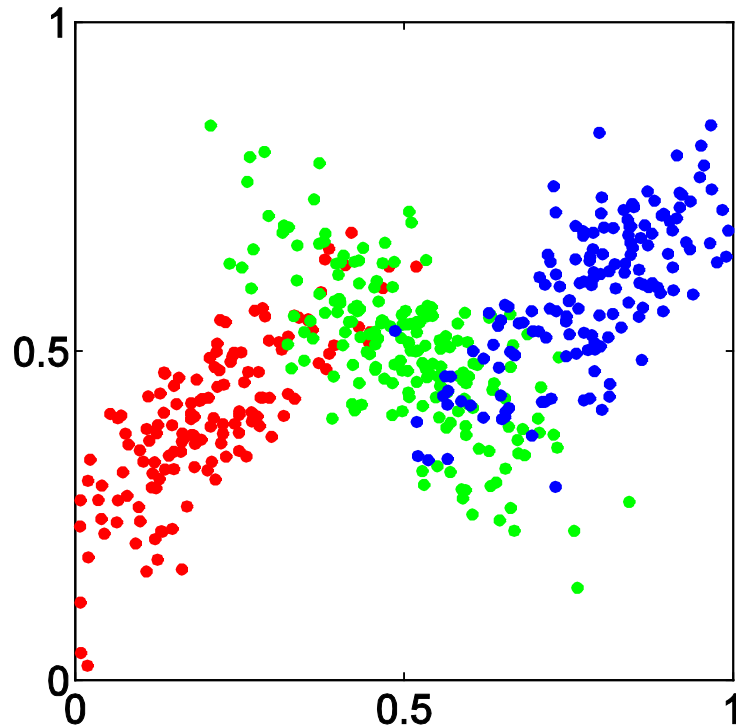
---

- From Bayes' theorem

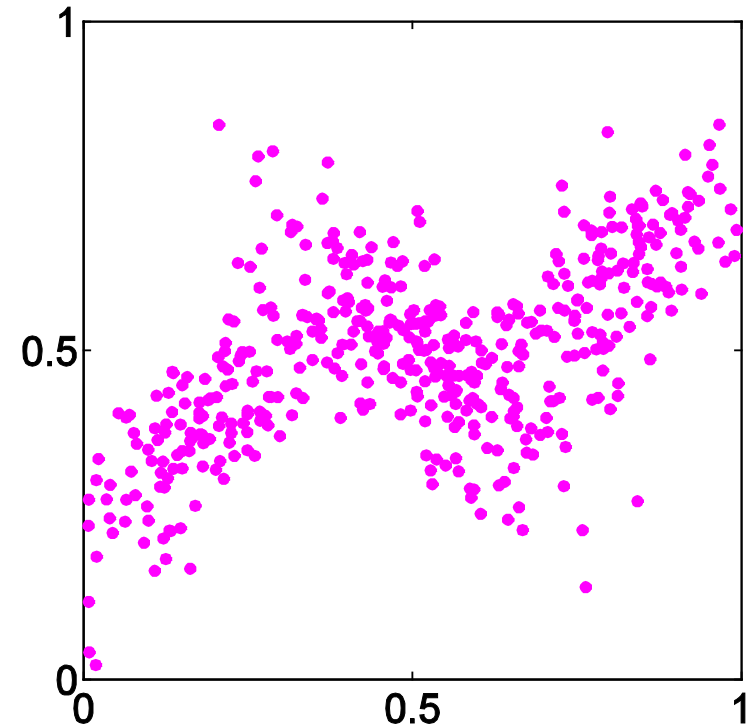
$$\begin{aligned} E[z_{ni}] &= \frac{\sum_{z_{ni}} z_{ni} [\pi_i p(\mathbf{x}_n | \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)]^{z_{ni}}}{\sum_{z_{ni}} [\pi_i p(\mathbf{x}_n | \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)]^{z_{ni}}} \\ &= \frac{\pi_i \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)}{\sum_{j=1}^K \pi_j \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)} \\ &= \gamma_i(\mathbf{x}_n) \end{aligned}$$

# Complete and Incomplete Data

---



complete



incomplete



# Latent Variable View of EM

---

- If we knew the values for the latent variables, we would maximize the complete-data log likelihood

$$\ln p(\mathbf{x}, \mathbf{z} | \boldsymbol{\theta}) = \sum_{n=1}^N \sum_{k=1}^K z_{nk} \{ \ln \pi_k + \ln \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \}$$

which gives a trivial closed-form solution (fit each component to the corresponding set of data points)

- We don't know the values of the latent variables
- However, for given parameter values we can compute the expected values of the latent variables

# Expected Complete-Data Log Likelihood

---

- Suppose we make a guess  $\theta_{\text{old}}$  for the parameter values (means, covariances and mixing coefficients)
- Use these to evaluate the responsibilities
- Consider expected complete-data log likelihood

$$E_{\mathbf{z}}[\ln p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta})] = \sum_{n=1}^N \sum_{i=1}^K \gamma_i(\mathbf{x}_n) \{ \ln \pi_i + \ln \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i) \}$$

where responsibilities are computed using  $\theta_{\text{old}}$

- We are implicitly ‘filling in’ latent variables with best guess
- Keeping the responsibilities fixed and maximizing with respect to the parameters give the previous results

# EM in General

---

- Consider arbitrary distribution  $q(\mathbf{z})$  over the latent variables
- The following decomposition always holds

$$\ln p(\mathbf{x}|\boldsymbol{\theta}) = \mathcal{L}(q, \boldsymbol{\theta}) + \text{KL}(q||p)$$

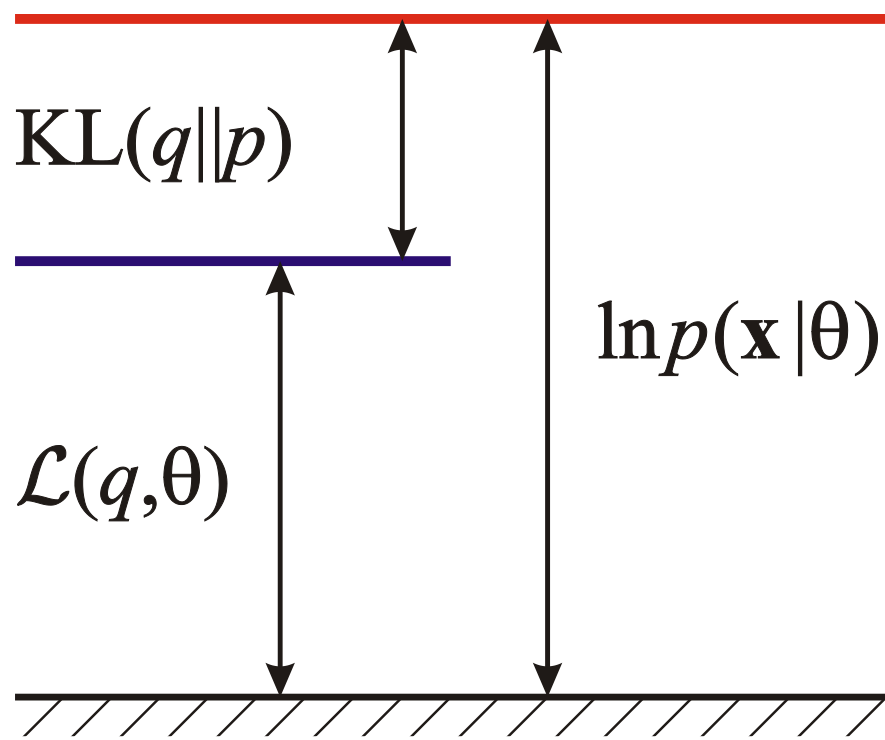
where

$$\mathcal{L}(q, \boldsymbol{\theta}) = \sum_{\mathbf{z}} q(\mathbf{z}) \ln \left\{ \frac{p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta})}{q(\mathbf{z})} \right\}$$

$$\text{KL}(q||p) = - \sum_{\mathbf{z}} q(\mathbf{z}) \ln \left\{ \frac{p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta})}{q(\mathbf{z})} \right\}$$

# Decomposition

---



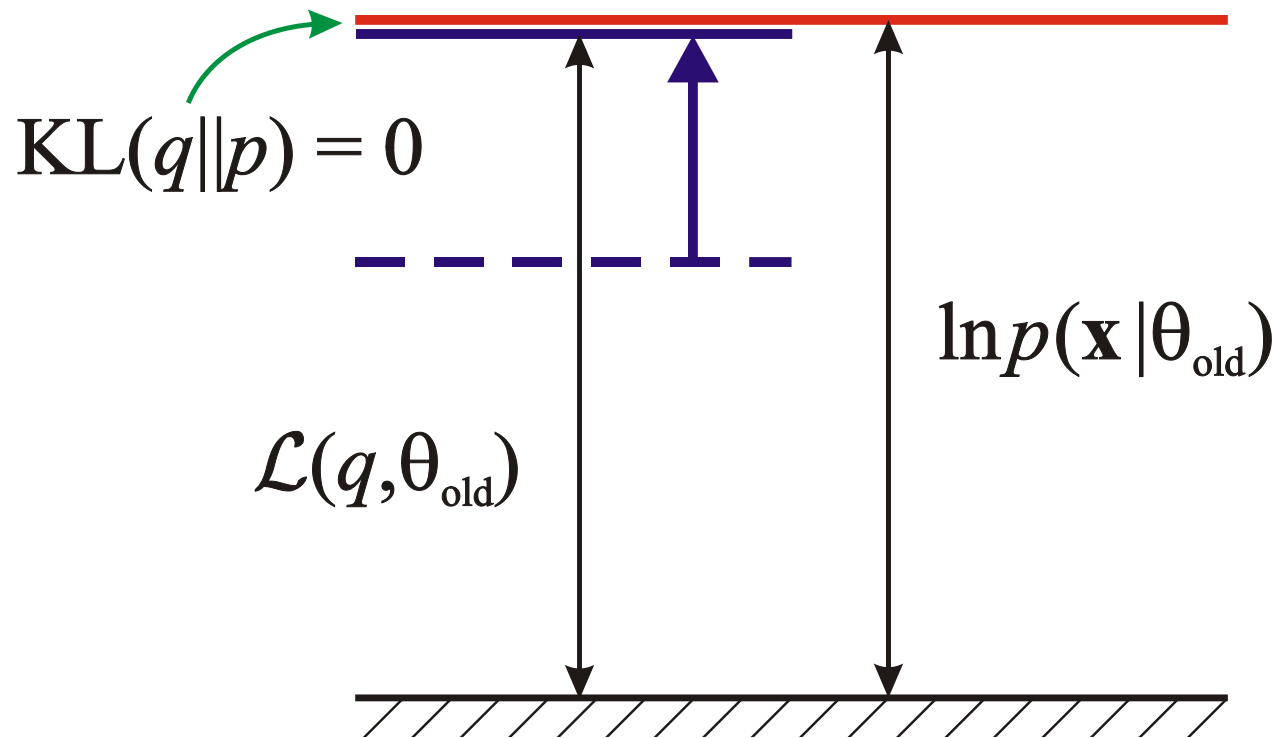
# Optimizing the Bound

---

- E-step: maximize  $\mathcal{L}$  with respect to  $q(\mathbf{z})$ 
  - equivalent to minimizing KL divergence
  - sets  $q(\mathbf{z})$  equal to the posterior distribution  $p(\mathbf{z}|\mathbf{x}, \theta)$
- M-step: maximize bound with respect to  $\theta$ 
  - equivalent to maximizing expected complete-data log likelihood
- Each EM cycle must increase incomplete-data likelihood unless already at a (local) maximum

# E-step

---



# M-step

---

