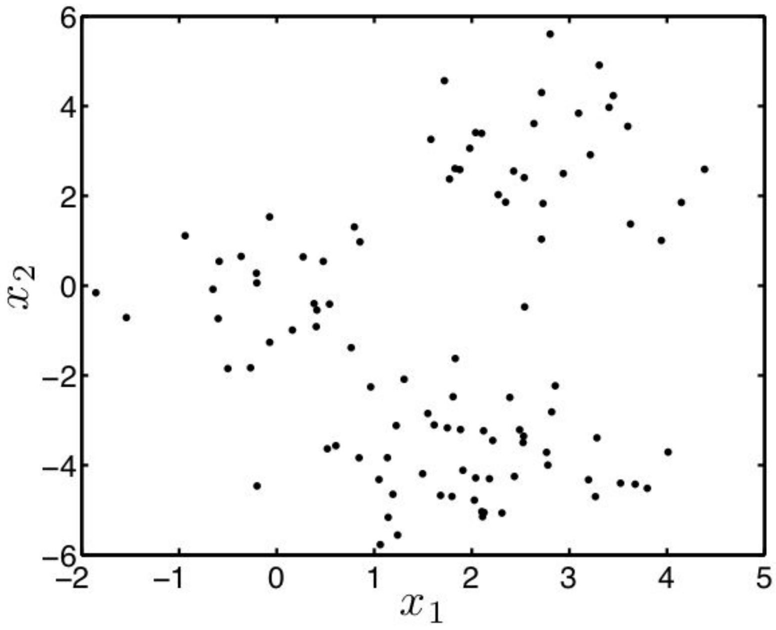
# Machine Learning & Artificial Intelligence for Data Scientists: Clustering (Part 2)

Ke Yuan
School of Computing Science

# Mixture models – thinking generatively




- Could we hypothesis a model that could have created this data?
- Each $\mathbf{x}_n$ seems to have come from one of three distributions.

# A generative model

- Assumption:Each $\mathbf{x}_n$ comes from one of different $K$ distributions.
- To generate $\mathbf{X}$:
- For each $n$:
  1. Pick one of the $K$ components.
  2. Sample $\mathbf{x}_n$ from this distribution.

- We already have $\mathbf{X}$
- Define parameters of all these distributions as $\Delta$.
- We'd like to reverse-engineer this process learn $\Delta$ which we can then use to find which component each point came from.
- Maximise the likelihood!

## Mixture model likelihood

- Let the $k$th distribution have pdf:

$$p(\mathbf{x}_n | z_{nk} = 1, \Delta_k)$$

- We want the likelihood:

$$p(\mathbf{X}|\Delta)$$

- First, factorise:

$$p(\mathbf{X}|\Delta) = \prod_{i=1}^{N} p(\mathbf{x}_n|\Delta)$$

- Then, un-marginalise $k$:

$$\begin{aligned} p(\mathbf{X}|\Delta) &= \prod_{i=1}^{N} \sum_{k=1}^{K} p(\mathbf{x}_n, z_{nk} = 1|\Delta) \\ &= \prod_{i=1}^{N} \sum_{k=1}^{K} p(\mathbf{x}_n|z_{nk} = 1, \Delta_k) p(z_{nk} = 1|\Delta) \end{aligned}$$

# Mixture model likelihood

- So, we have a likelihood:

$$p(\mathbf{X}|\Delta) = \prod_{i=1}^{N} \sum_{k=1}^{K} p(\mathbf{x}_n | z_{nk} = 1, \Delta_k) p(z_{nk} = 1 | \Delta)$$

- And we want to find $\Delta$.
- So:

$$\underset{\Delta}{\operatorname{argmax}} \prod_{i=1}^{N} \sum_{k=1}^{K} p(\mathbf{x}_n | z_{nk} = 1, \Delta_k) p(z_{nk} = 1 | \Delta)$$

- Logging made this easier before, so let's try it:

$$\underset{\Delta}{\operatorname{argmax}} \sum_{n=1}^{N} \log \sum_{k=1}^{K} p(\mathbf{x}_n | z_{nk} = 1, \Delta_k) p(z_{nk} = 1 | \Delta)$$

# Gaussian mixture model

- Assume component distributions are Gaussians with diagonal covariance:

$$p(\mathbf{x}_n|z_{nk}=1,\boldsymbol{\mu}_k,\sigma_k^2)=\mathcal{N}(\boldsymbol{\mu}_k,\sigma_k^2\mathbf{I})$$

- We need to be able to estimate the prior of assignment. Let

$$\pi_k=p(z_{nk}=1|\Delta)$$

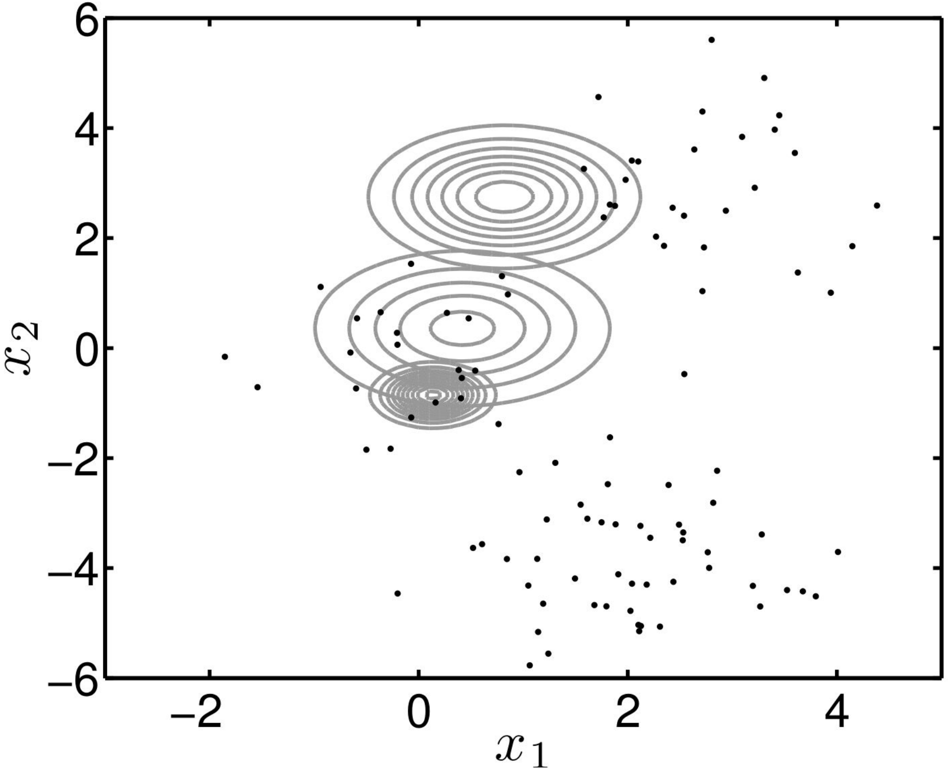
- We also want to estimate the probability to assign data to each component

$$q_{nk}=\frac{\pi_k p(\mathbf{x}_n|z_{nk}=1,\boldsymbol{\mu}_k,\sigma_k^2)}{\sum_{j=1}^{K}\pi_j p(\mathbf{x}_n|z_{nj}=1,\boldsymbol{\mu}_j,\sigma_j^2)}$$

# Mixture model optimisation – the Expectation-Maximization (EM) algorithm

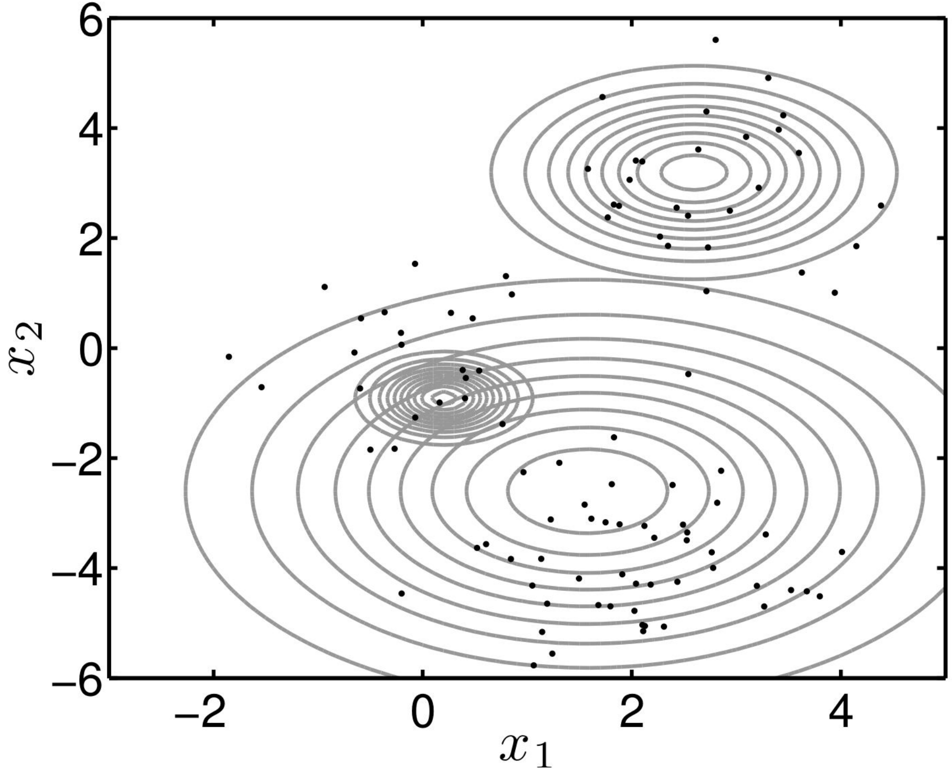
- Following optimisation algorithm:
  1. Guess $\boldsymbol{\mu}_k, \sigma_k^2, \pi_k$
  2. **(E)**xpectation-step: Compute $q_{nk}$
  3. **(M)**aximization-step: Update $\boldsymbol{\mu}_k, \sigma_k^2, \pi_k$
  4. Return to 2 unless parameters are unchanged.
- Guaranteed to converge to a local maximum of the lower bound.
- Note the similarity with kmeans.

# Algorithm in operation



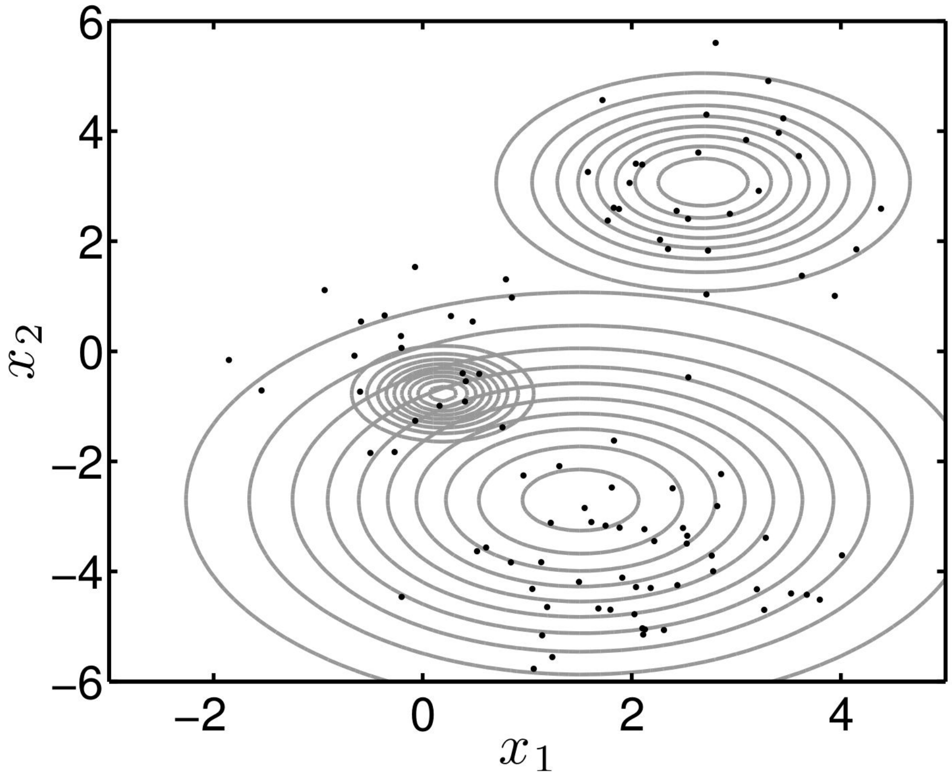

Initial guess

# Algorithm in operation



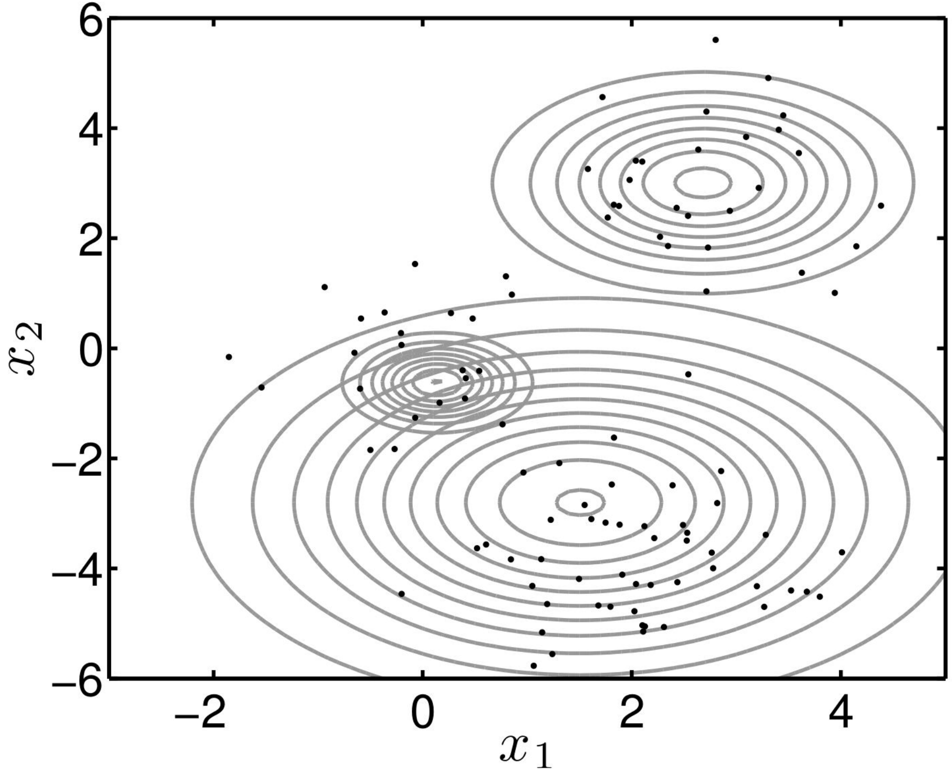

Update q_nk and then other parameters.

# Algorithm in operation



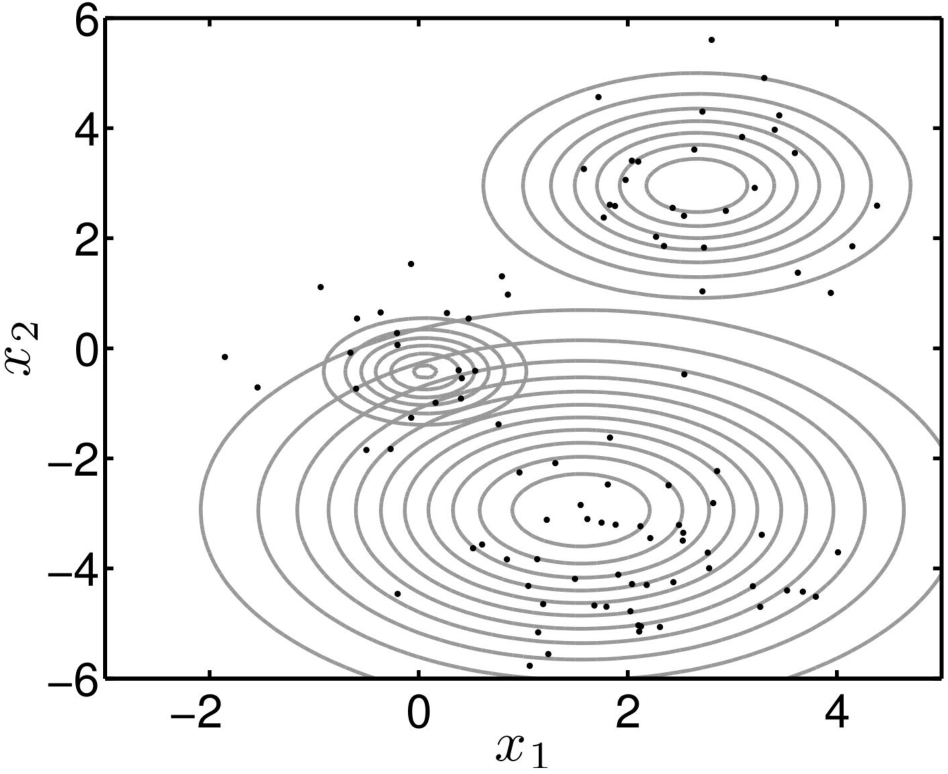

Update q_nk and then other parameters.

# Algorithm in operation



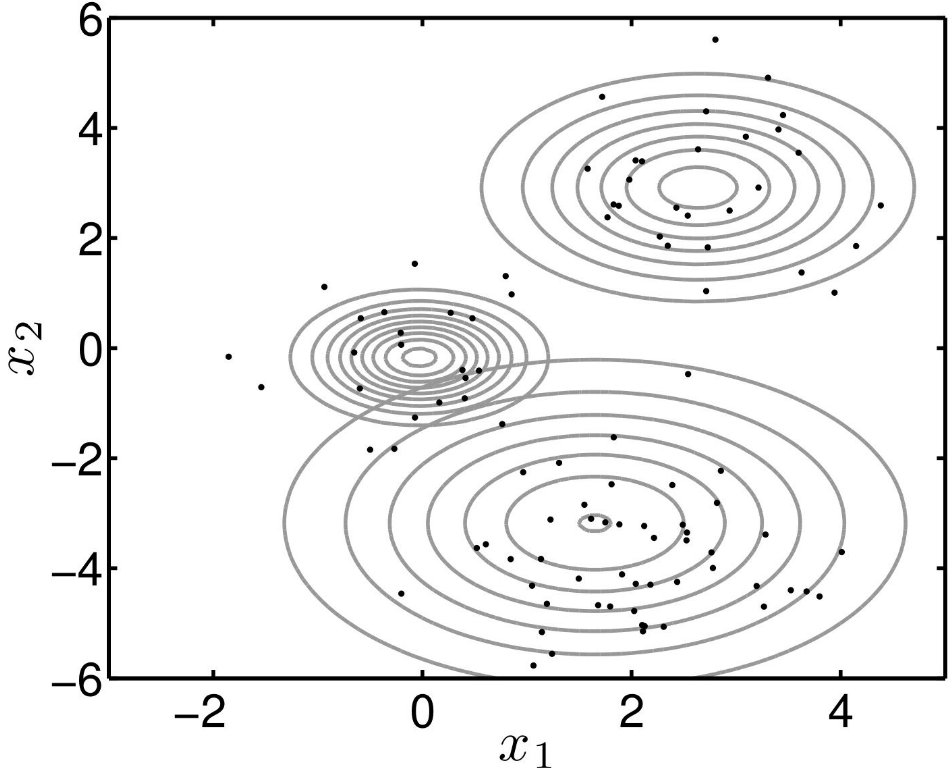

Update q_nk and then other parameters.

# Algorithm in operation



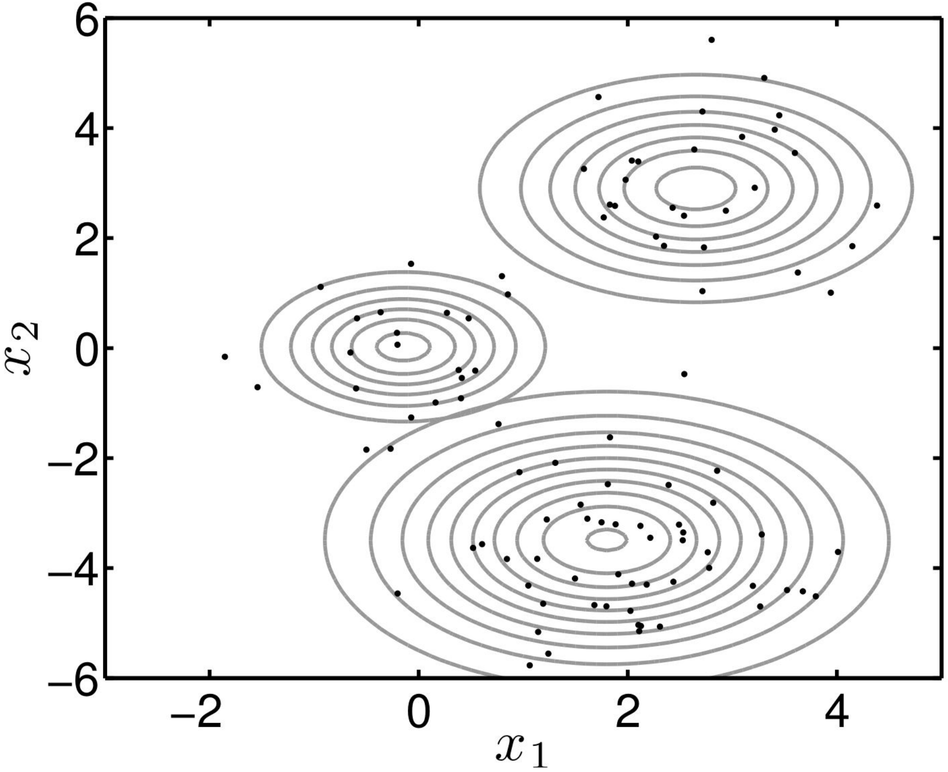

Update q_nk and then other parameters.

# Algorithm in operation



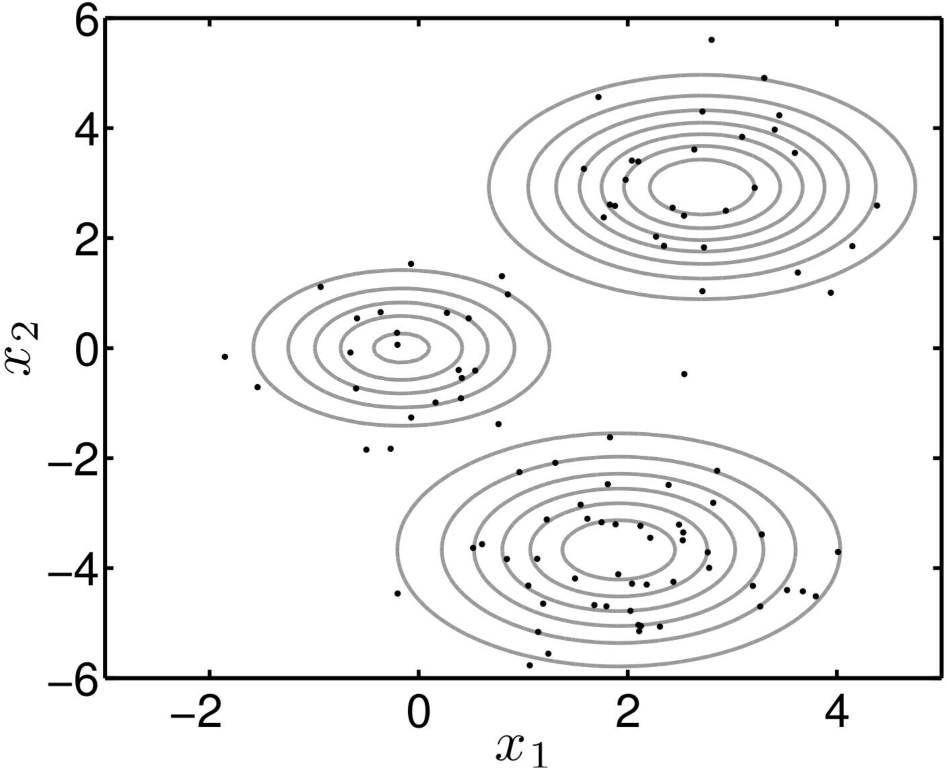

Update q_nk and then other parameters.

# Algorithm in operation



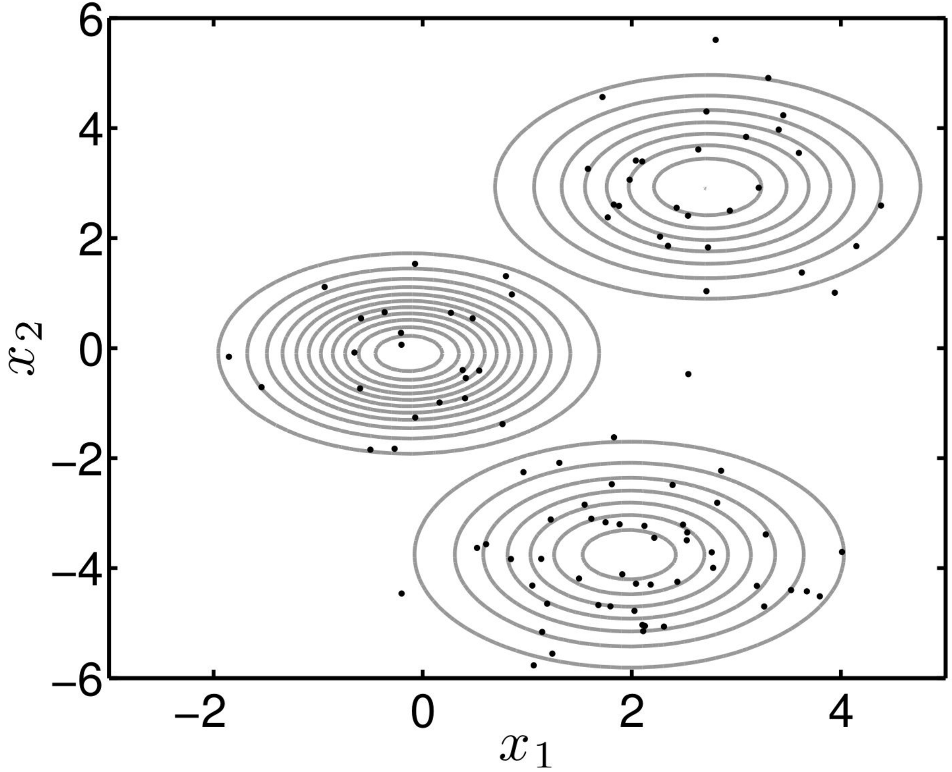

Update q_nk and then other parameters.

# Algorithm in operation



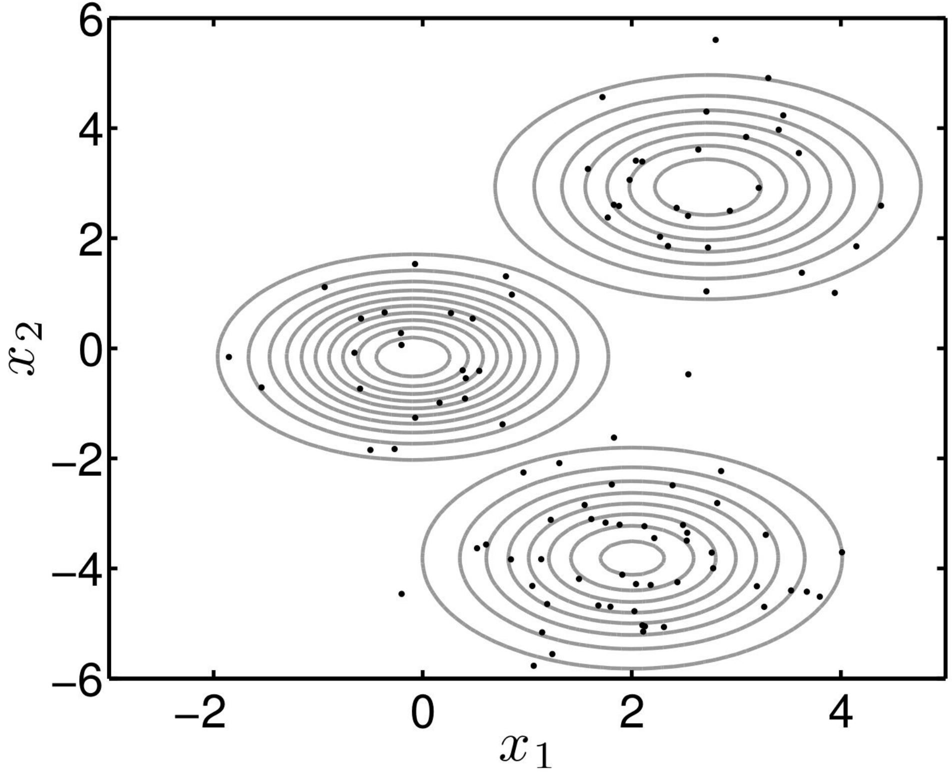

Update q_nk and then other parameters.

# Algorithm in operation



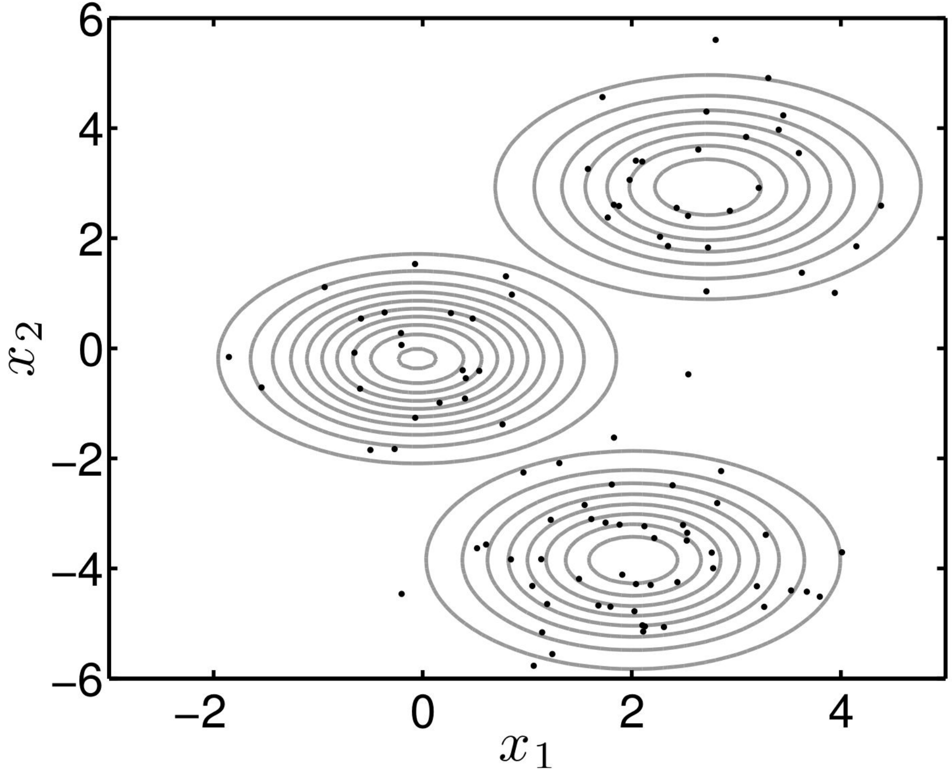

Update q_nk and then other parameters.

# Algorithm in operation



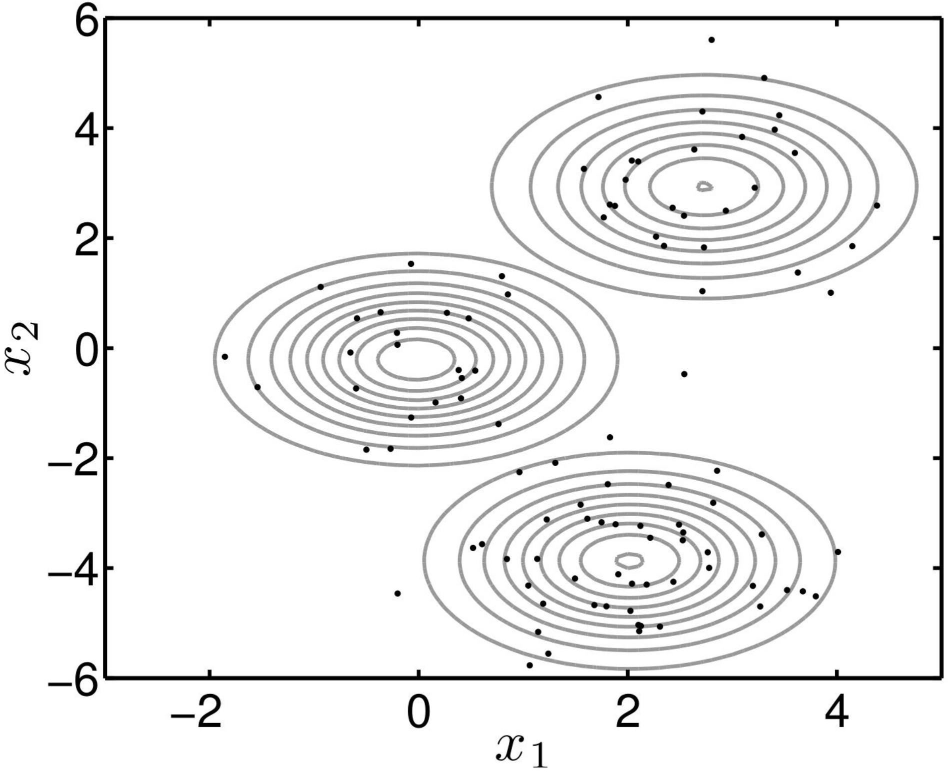

Update q_nk and then other parameters.

# Algorithm in operation



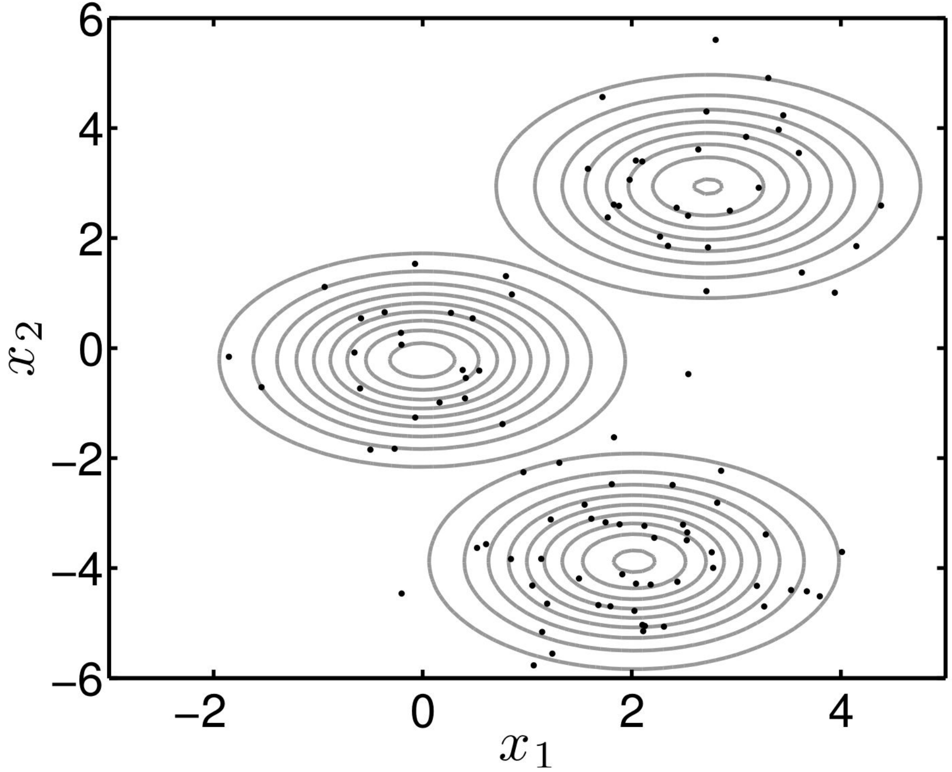

Update q_nk and then other parameters.

# Algorithm in operation



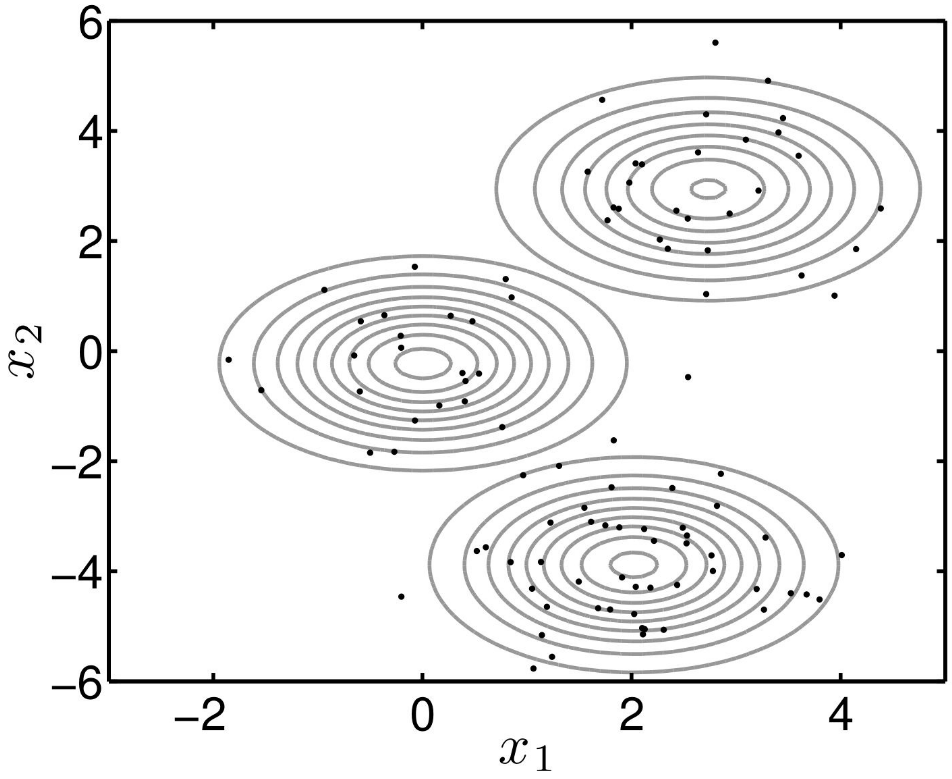

Update q_nk and then other parameters.

# Algorithm in operation



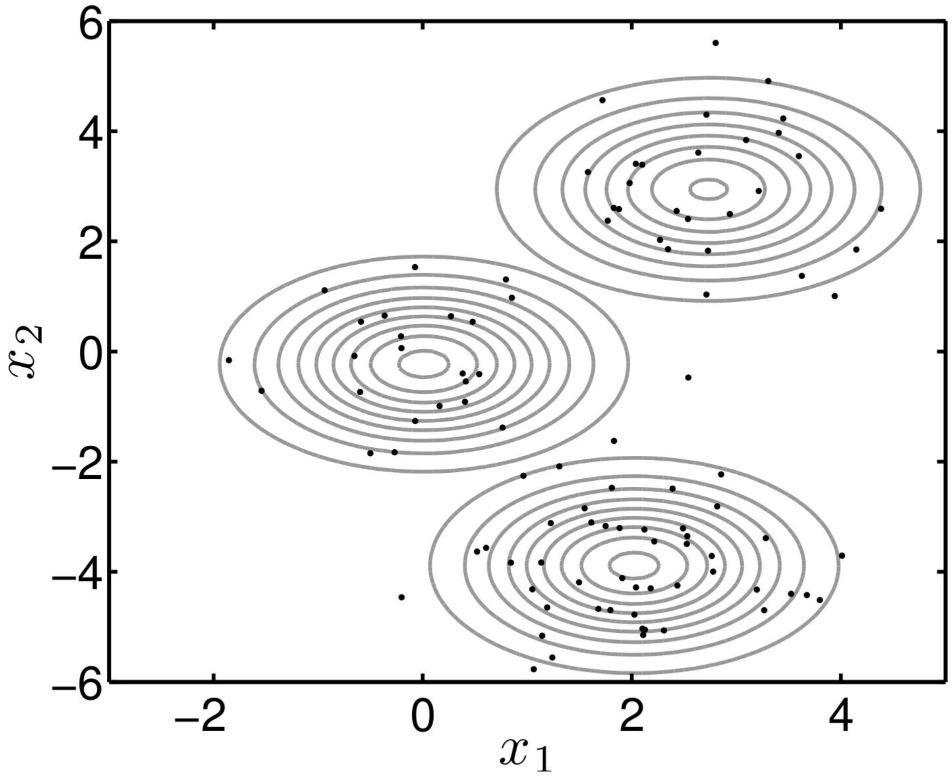

Update q_nk and then other parameters.

# Algorithm in operation



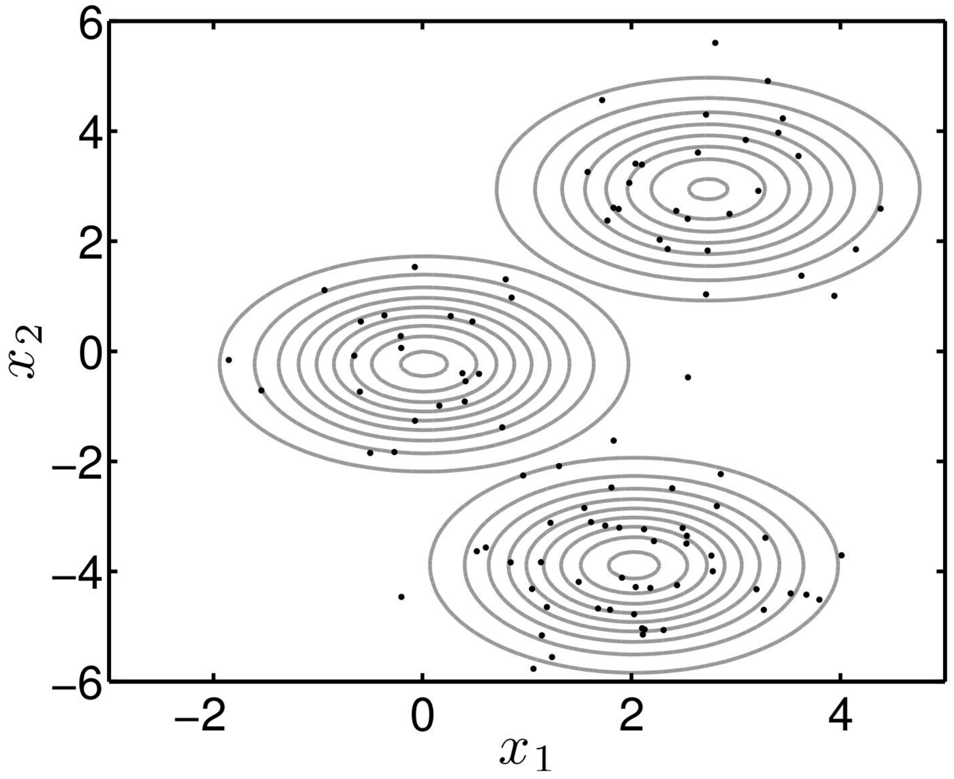

Update q_nk and then other parameters.

# Algorithm in operation




Update q_nk and then other parameters.

# Algorithm in operation




Solution at convergence

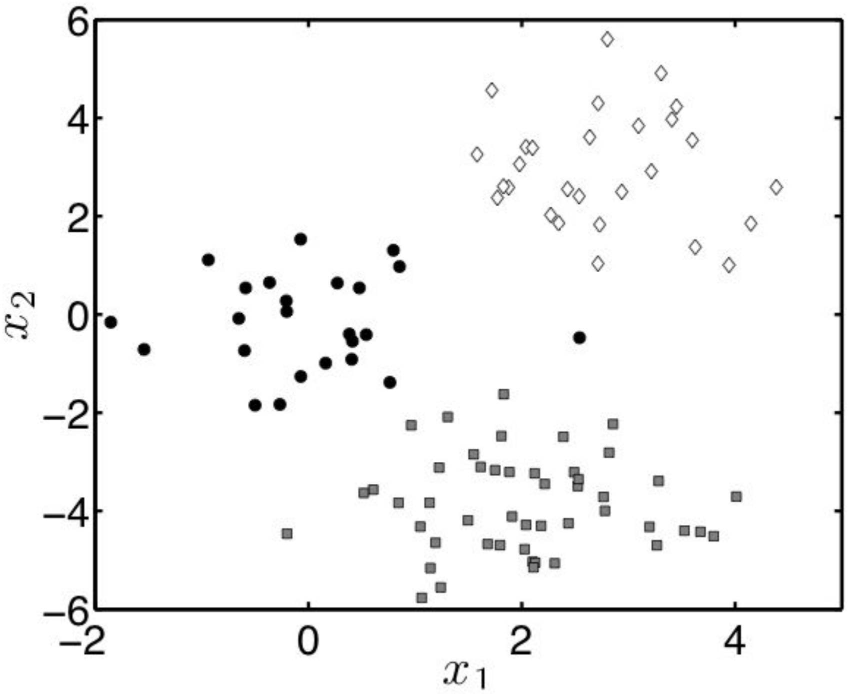
# Mixture model clustering

- So, we've got the parameters, but what about the assignments?
- Which points came from which distributions?
- $q_{nk}$ is the probability that $\mathbf{x}_n$ came from distribution $k$.

$$q_{nk} = P(z_{nk} = 1 | \mathbf{x}_n, \mathbf{X}, \mathbf{t})$$

- Can stick with probabilities or assign each $\mathbf{x}_n$ to it's most likely component.

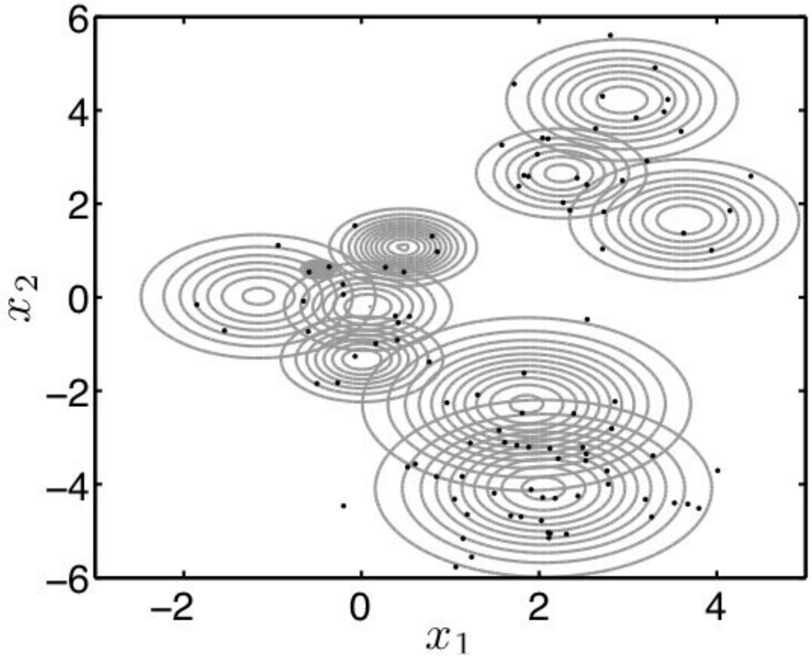
# Mixture model clustering



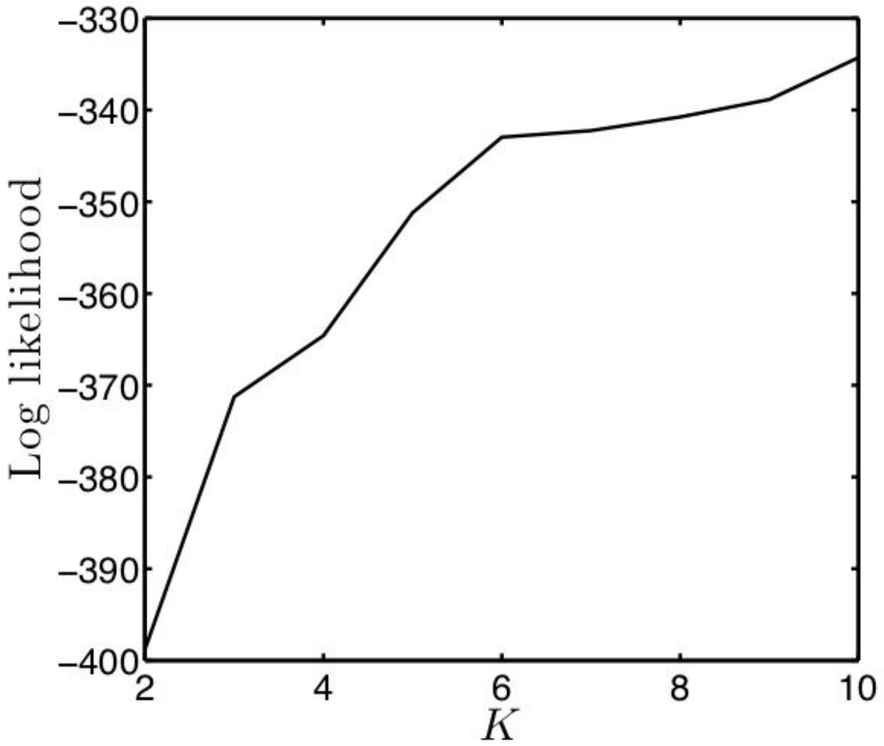

- Points assigned to the cluster with the highest $q_{nk}$ value.

# Mixture model – issues

- How do we choose $K$?
- What happens when we increase it?
- $K = 10$



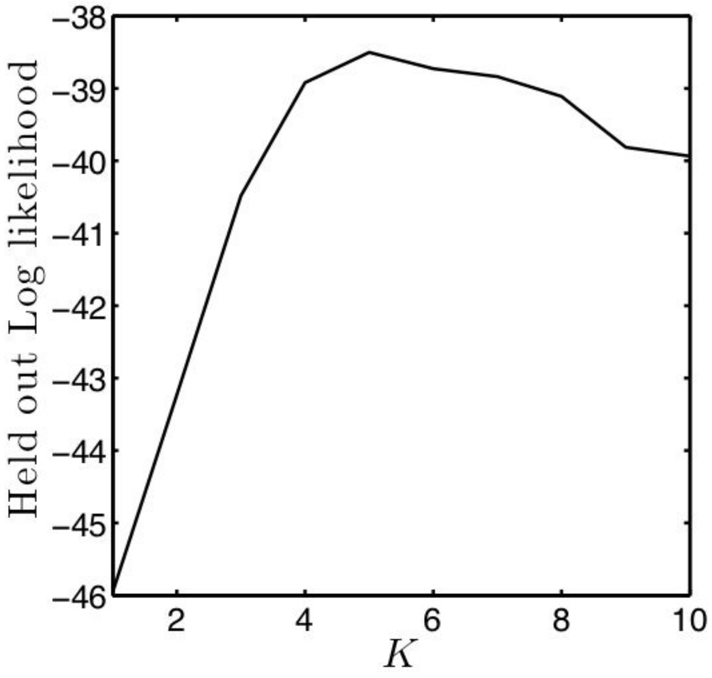
# Likelihood increase




- Likelihood always increases as $\sigma_k^2$ decreases.

# What can we do?

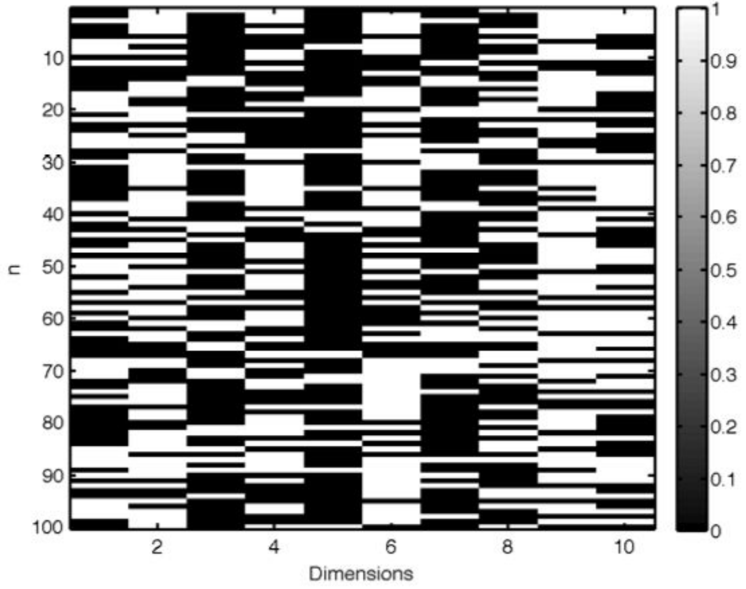
- What can we do?
- Cross-validation...




- 10-fold CV. Maximum is close to true value (3)
- 5 might be better for this data....

# Mixture models – other distributions

- We've seen Gaussian distributions.
- Can actually use anything....
- As long as we can define $p(\mathbf{x}_n | z_{nk} = 1, \Delta_k)$
- e.g. Binary data:



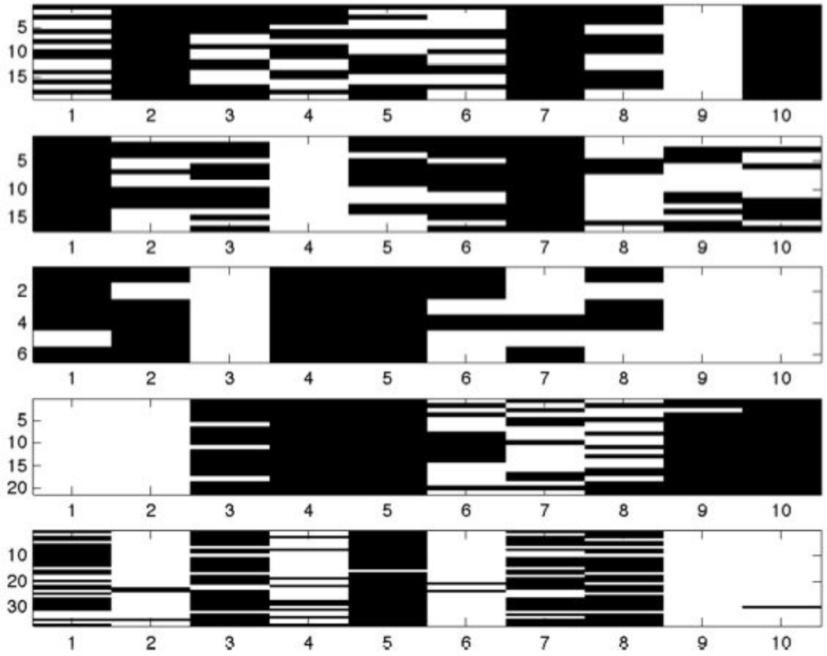
# Binary example

- $\mathbf{x}_n = [0, 1, 0, 1, 1, \ldots, 0, 1]^\mathsf{T}$ ($D$ dimensions)
- $p(\mathbf{x}_n | z_{nk} = 1, \Delta_k) = \prod_{d=1}^{D} p_{kd}^{x_{nd}} (1 - p_{kd})^{1-x_{nd}}$
- Updates for $p_{kd}$ are:

$$p_{kd} = \frac{\sum_n q_{nk} x_{nd}}{\sum_n q_{nk}}$$

- $q_{nk}$ and $\pi_k$ are the same as before...
- Initialise with random $p_{kd}$ $(0 \leq p_{kd} \leq 1)$

# Binary example




- $K = 5$ clusters.
- Clear structure present.

# Summary

- Introduced two clustering methods.
- K-means
  - Very simple.
  - Iterative scheme.
  - Can be kernelised.
  - Need to choose $K$.
- Mixture models
  - Create a model of each class (similar to Bayes classifier)
  - Iterative sceme (EM)
  - Can use any distribution for the components.
  - Can set $K$ by cross-validation (held-out likelihood)
  - State-of-the-art: Don't need to set $K$ – treat as a variable in a Bayesian sampling scheme.