Outline

- Bayesian concept learning: Discussion
- Probabilistic models for unsupervised and semi-supervised category learning

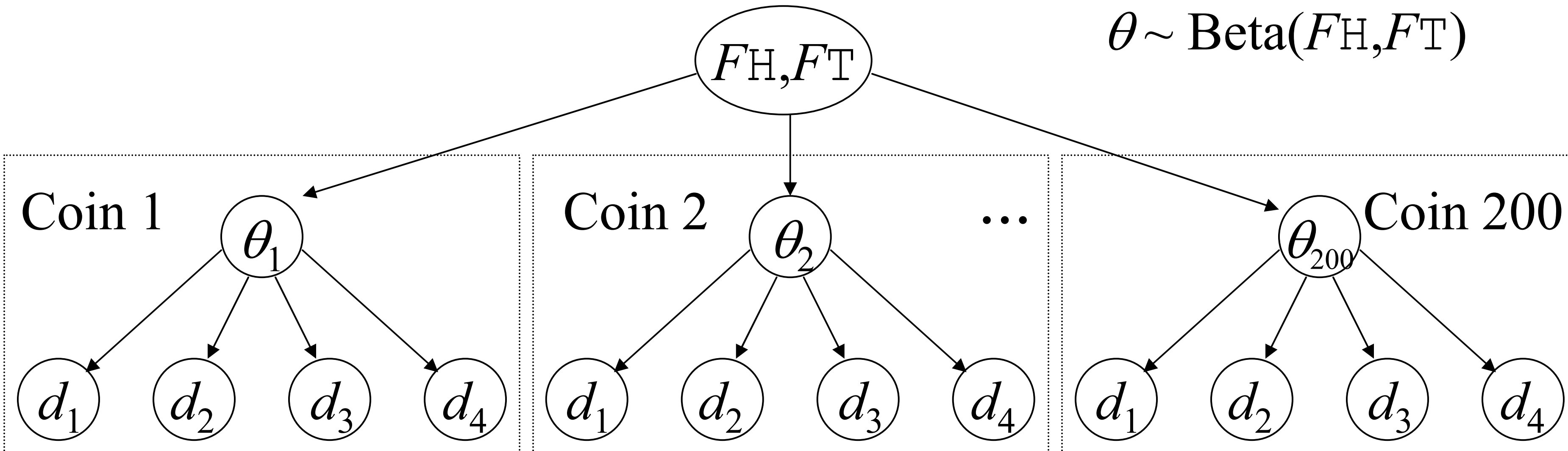
Discussion points

- Relation to “Bayesian classification”?
- Relation to debate between rules / logic / symbols and similarity / connections / statistics?
- Where do the hypothesis space and prior probability distribution come from?

Discussion points

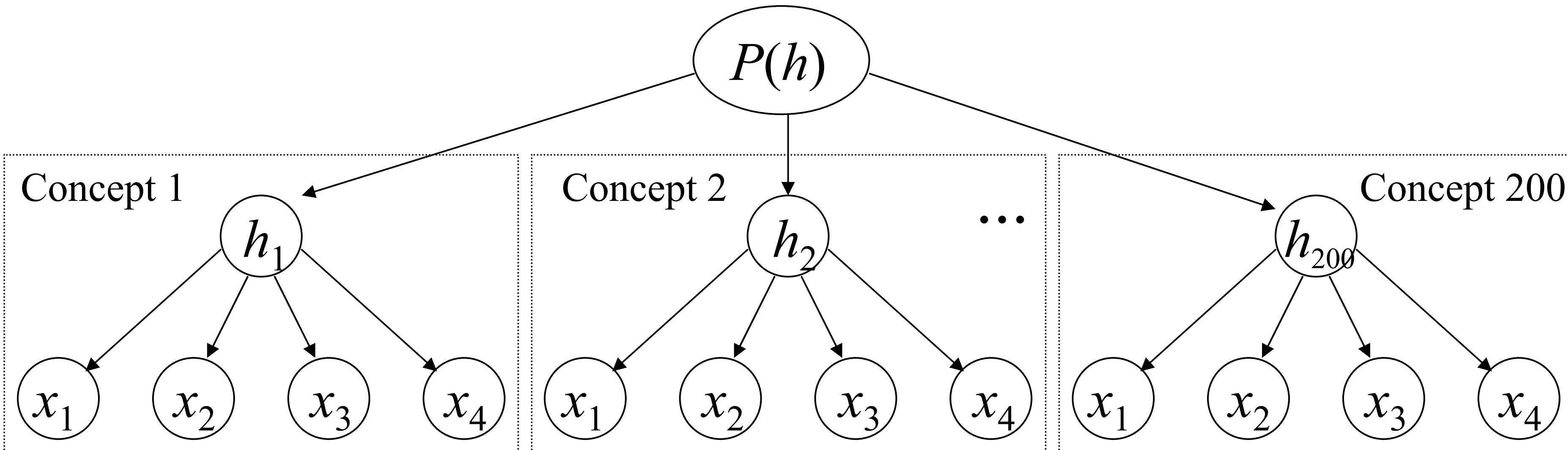
- Relation to "Bayesian classification"?
  - Causal attribution versus referential inference.
  - Which is more suited to natural concept learning?
- Relation to debate between rules / logic / symbols and similarity / connections / statistics?
- Where do the hypothesis space and prior probability distribution come from?

Hierarchical priors

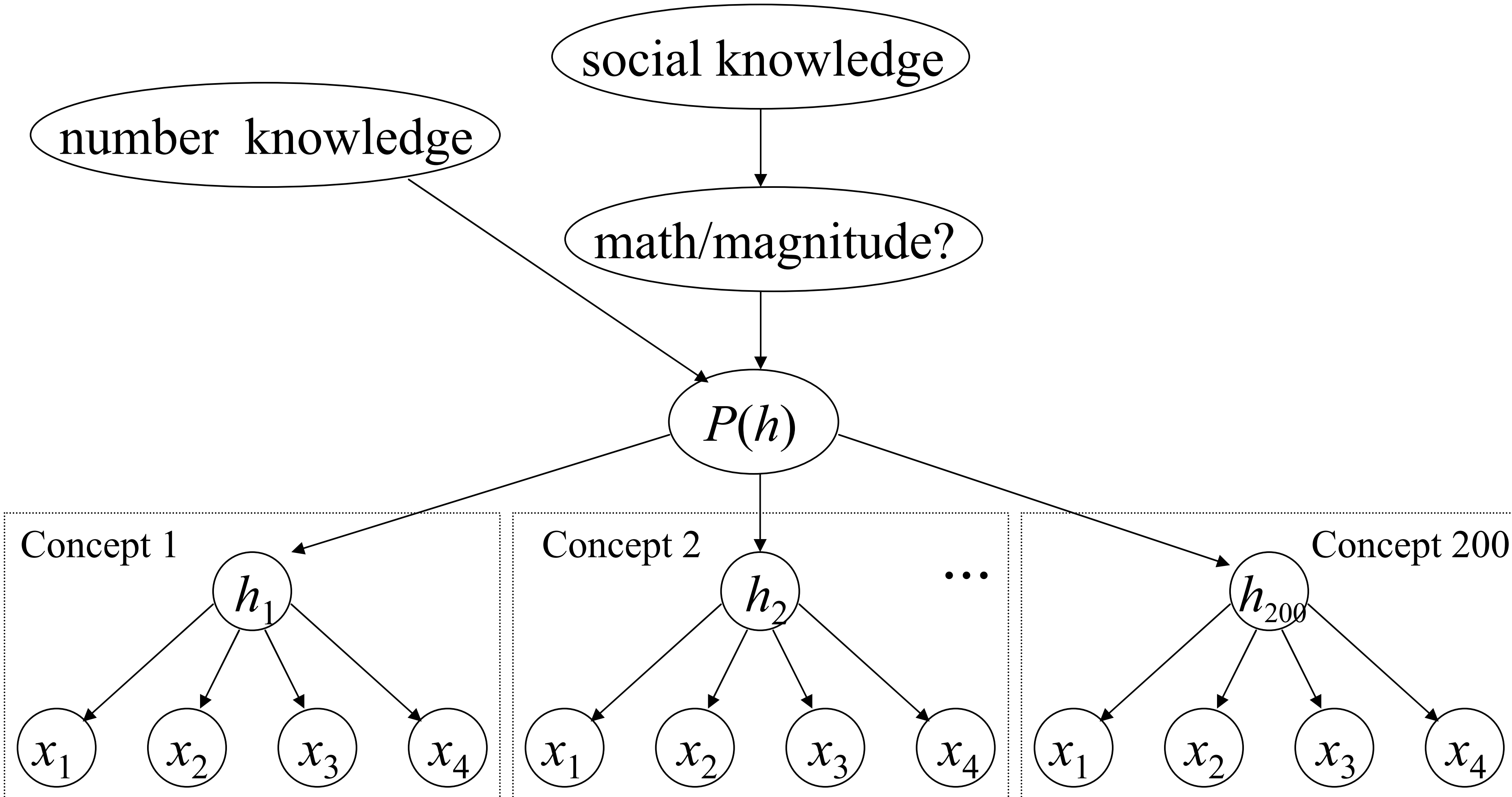


- Latent structure captures what is common to all coins, and also their individual variability

Hierarchical priors



- Latent structure captures what is common to all concepts, and also their individual variability
- *Is this all we need?*

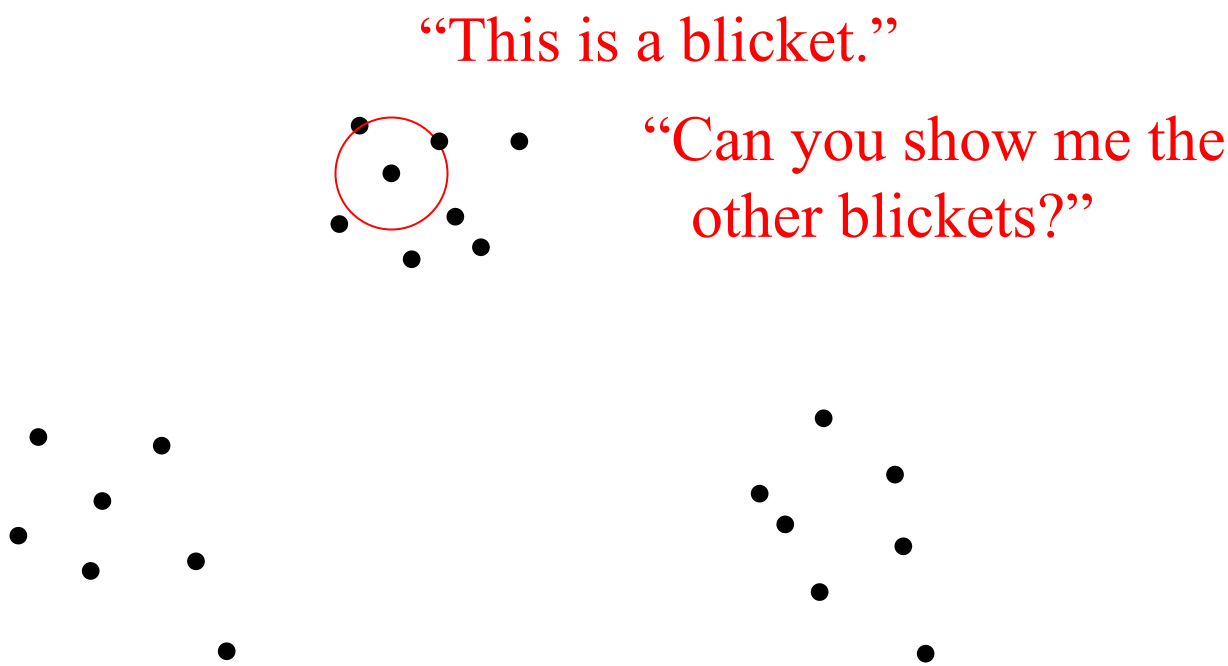


- Hypothesis space is not just an arbitrary collection of hypotheses, but a principled system.
- Far more structured than our experience with specific number concepts.

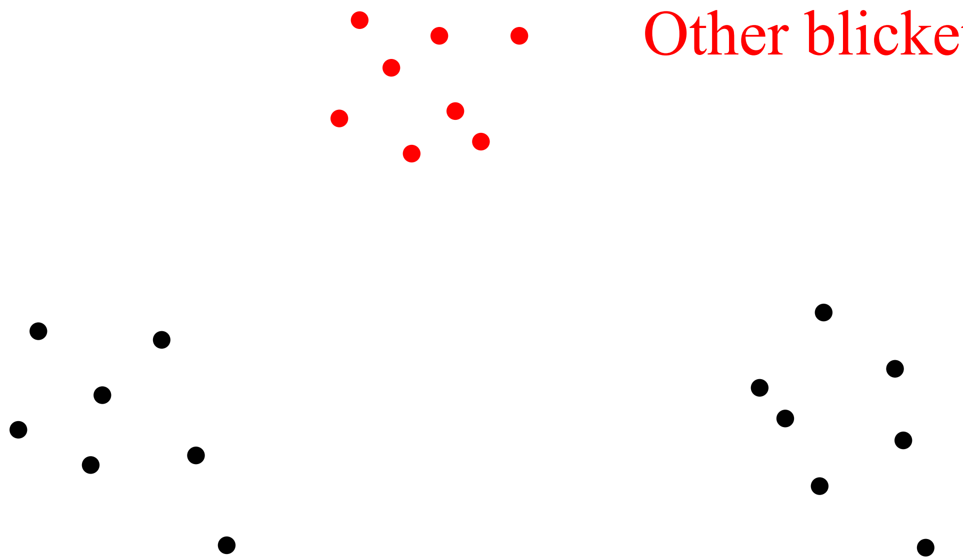
Outline

- Bayesian concept learning: Discussion
- Probabilistic models for unsupervised and semi-supervised category learning

Simple model of concept learning



Simple model of concept learning

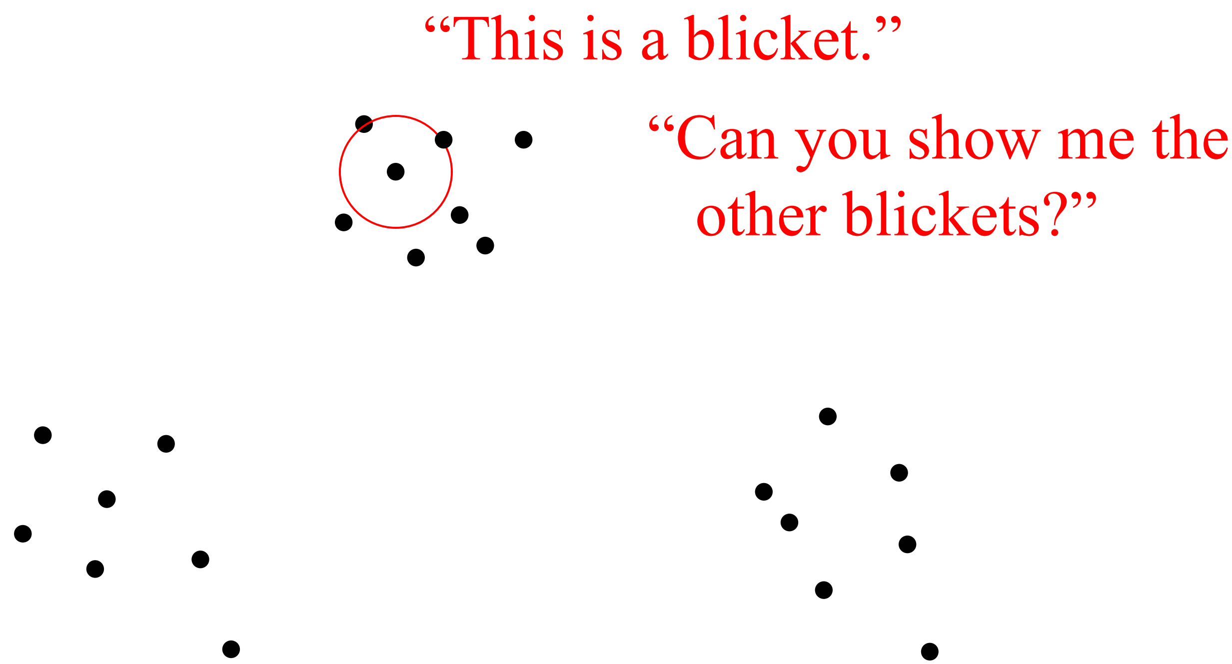


Other blickets.

The objects of planet Gazoob

Image removed due to copyright considerations.

Simple model of concept learning



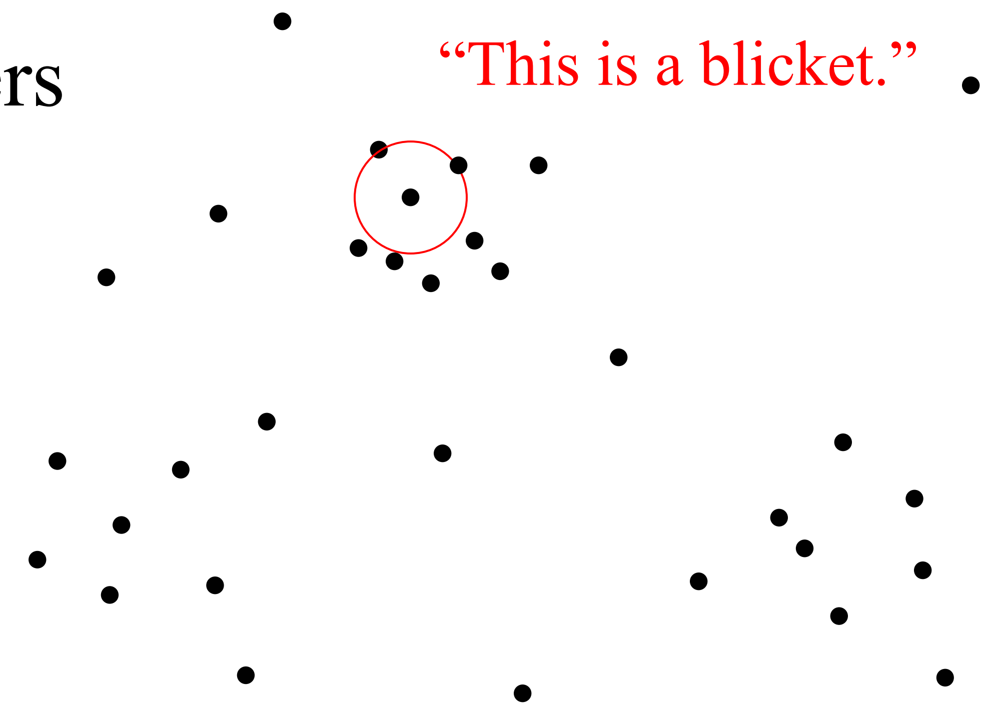
Learning from just one positive example is possible if:

- Assume concepts refer to clusters in the world.
- Observe enough unlabeled data to identify clear clusters.

Complications

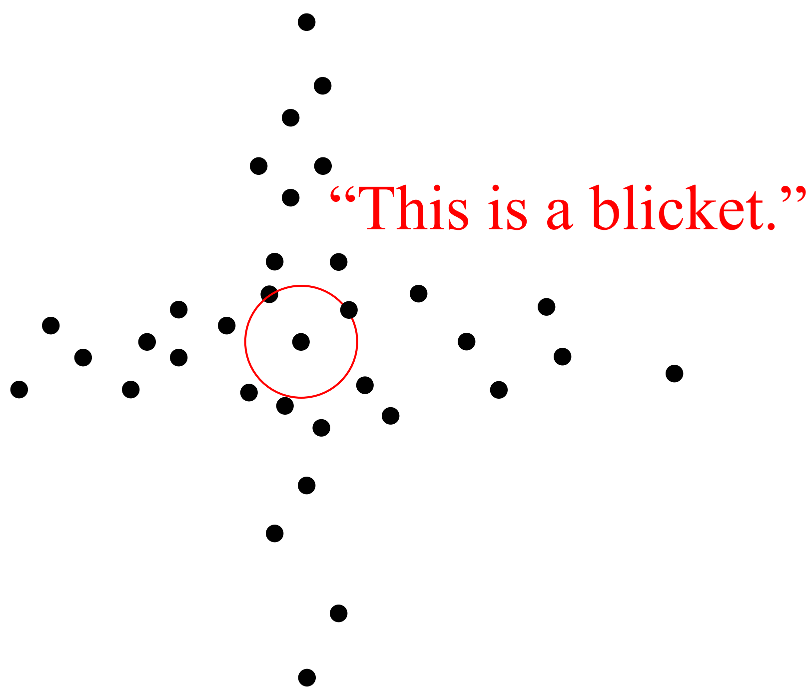
Complications

- Outliers



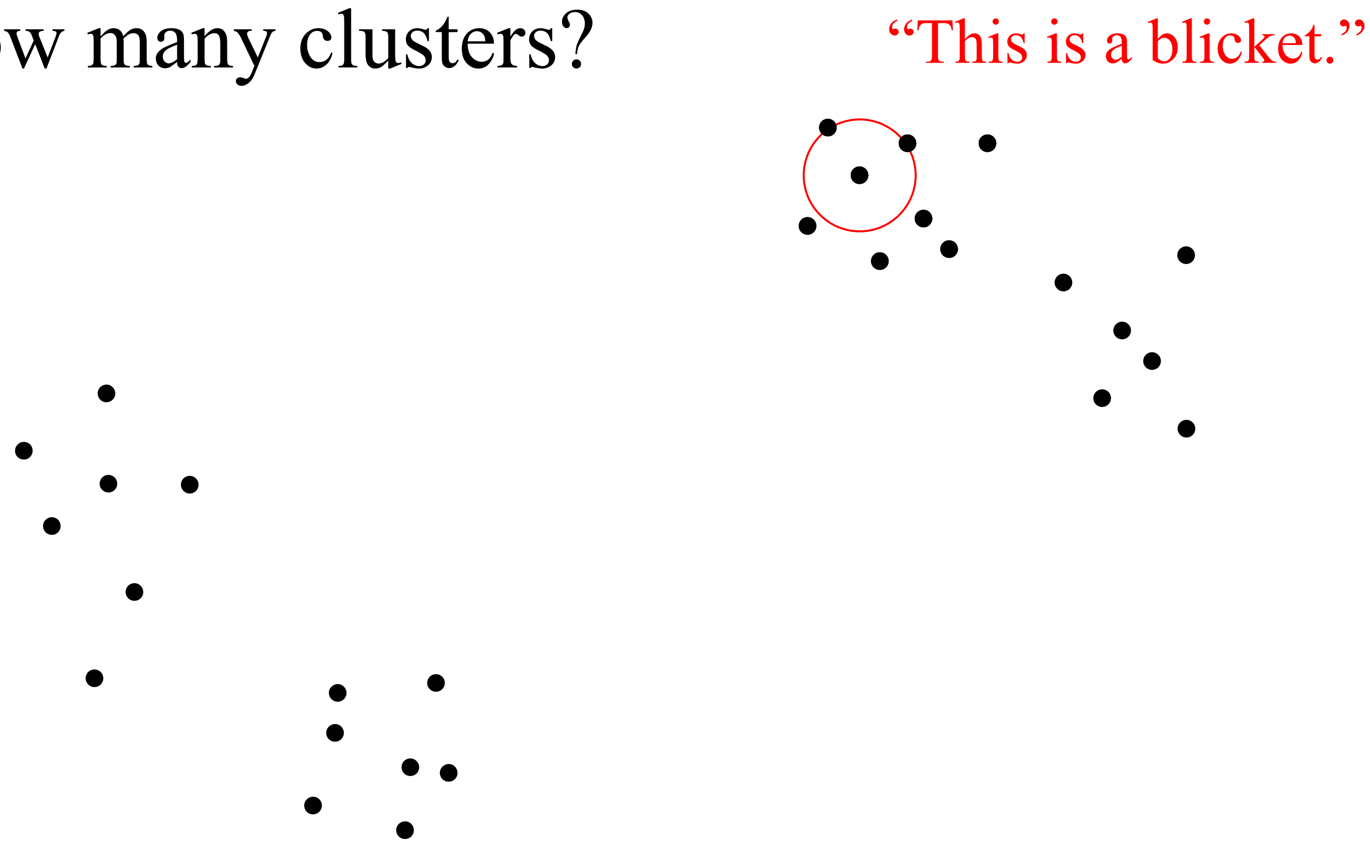
Complications

- Overlapping clusters



Complications

- How many clusters?

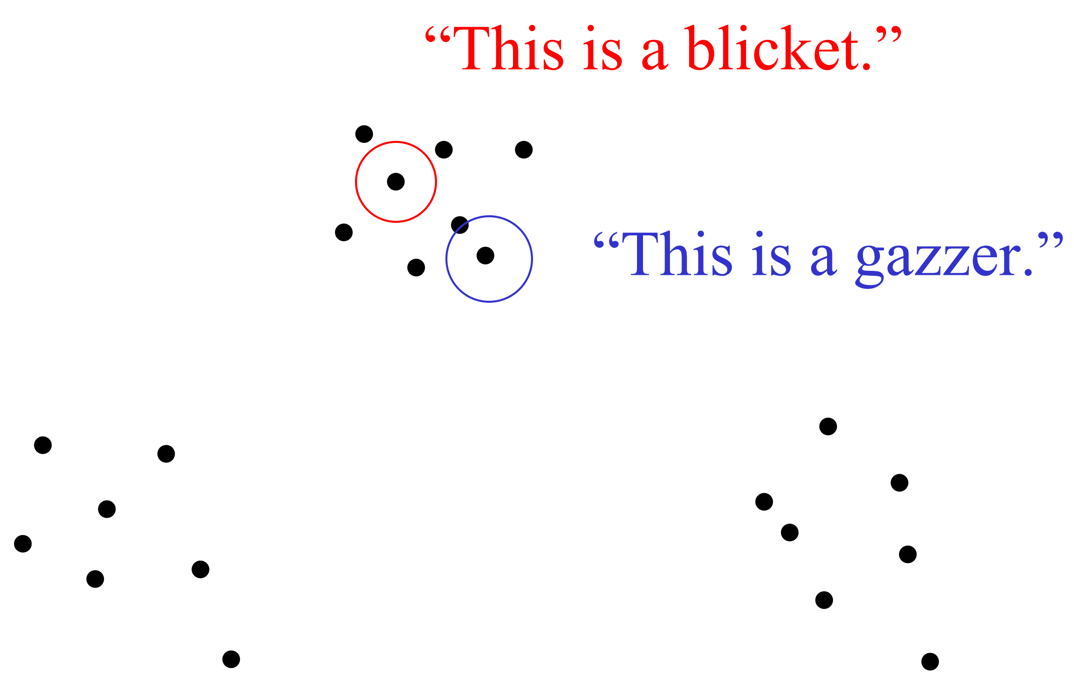


Complications

- Clusters that are not simple blobs

Complications

- Concept labels inconsistent with clusters

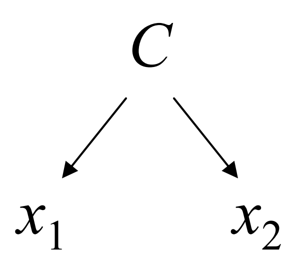


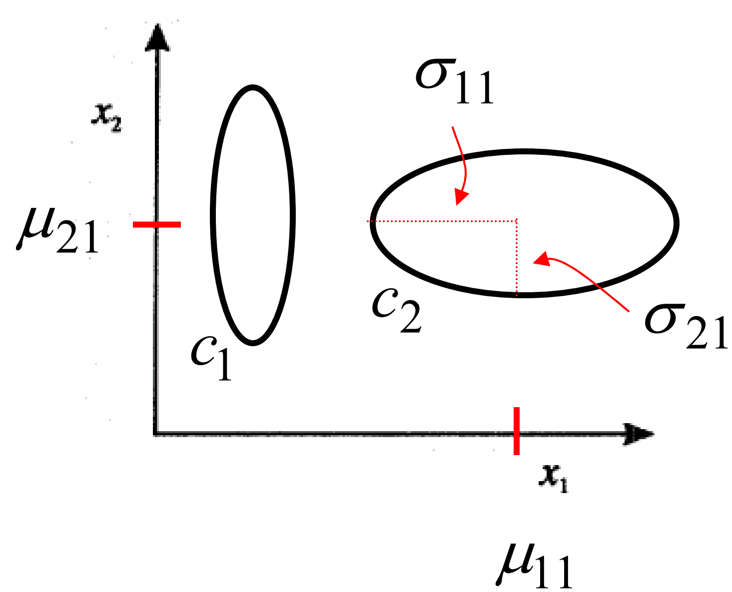
Simple model of concept learning

- Can infer a concept from just one positive example if:
  - Assume concepts refer to clusters in the world.
  - Observe lots of unlabeled data, in order to identify clusters.
- How do we identify the clusters?
  - With no labeled data ("unsupervised learning")
  - With sparsely labeled data ("semi-supervised learning")

Unsupervised clustering with probabilistic models

- Assume a simple parametric probabilistic model for clusters, e.g., Gaussian.





$$p(x \mid c_j) \quad p(x_1 \mid c_j) \times p(x_2 \mid c_j)$$

$$p(x_i \mid c_j) \propto e^{-(x_i - \mu_{ij})^2 / (2\sigma_{ij}^2)}$$

Unsupervised clustering with probabilistic models

- Assume a simple parametric probabilistic model for clusters, e.g., Gaussian.

- Two ways to characterize *j*th cluster:
  - Parameters: μ_{ij}, σ_{ij}
  - Assignments: $z_j^{(k)} = 1$ if *k*th point belongs to cluster *j*, else 0.

Unsupervised clustering with probabilistic models

- Chicken-and-egg problem:
  - Given assignments we could solve for maximum likelihood parameters:

$$\mu_{ij} \quad \frac{\sum_k z_j^{(k)} x_i^{(k)}}{\sum_k z_j^{(k)}} \qquad\qquad \sigma^2_{ij} \quad \frac{\sum_k z_j^{(k)} \left(x_i^{(k)} - \mu_{ij}\right)^2}{\sum_k z_j^{(k)}}$$

Unsupervised clustering with probabilistic models

- Chicken-and-egg problem:
  - Given parameters we could solve for assignments $z_j^{(k)}$:

$$z_j^{(k)} \quad \begin{array}{l} 1,\ j \quad \arg\max\limits_{j'} p(c_{j'} \mid \mathbf{x}^{(k)}) \\ 0,\ \text{else} \end{array}$$

$$p(c_j \mid \mathbf{x}^{(k)}) \propto p(\mathbf{x}^{(k)} \mid c_j)\, p(c_j)$$

$$\prod_i \frac{1}{\sqrt{2\pi}\sigma_{ij}} e^{-(x_i^{(k)} - \mu_{ij})^2 / (2\sigma_{ij}^2)}\, p(c_j)$$

Solve for "base rate" parameters:

$$p(c_j) \quad \sum_k z_j^{(k)}$$

Alternating optimization algorithm

0. Guess initial parameter values.

1. Given parameter estimates, solve for maximum a posteriori assignments $z_j^{(k)}$:

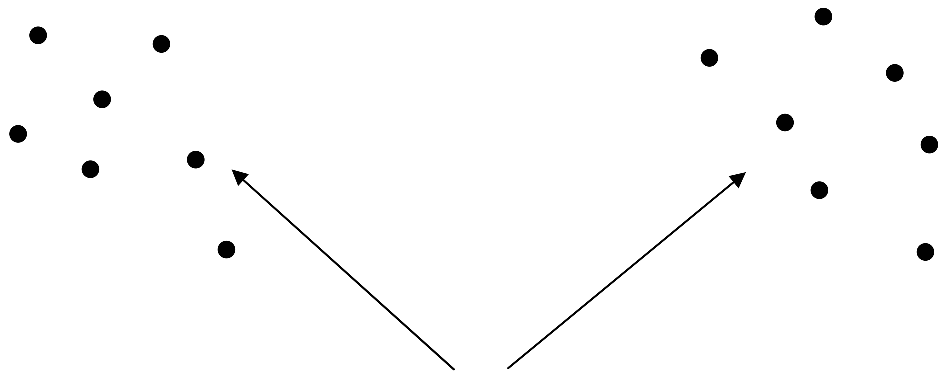
$$p(c_j \mid \mathbf{x}^{(k)}) \propto \prod_i \frac{1}{\sqrt{2\pi}\sigma_{ij}} e^{-(x_i^{(k)} - \mu_{ij})^2 / (2\sigma_{ij}^2)} p(c_j) \qquad z_j^{(k)} \quad \begin{matrix} 1, \ j \quad \arg\max_{j'} \ p(c_{j'} \mid \mathbf{x}^{(k)}) \\ 0, \text{ else} \end{matrix}$$

2. Given assignments $z_j^{(k)}$, solve for maximum likelihood parameter estimates:

$$\mu_{ij} \quad \frac{\sum_k z_j^{(k)} x_i^{(k)}}{\sum_k z_j^{(k)}} \qquad \sigma^2_{ij} \quad \frac{\sum_k z_j^{(k)} \left(x_i^{(k)} - \mu_{ij}\right)^2}{\sum_k z_j^{(k)}} \qquad p(c_j) \quad \sum_k z_j^{(k)}$$

3. Go to step 1.

Alternating optimization algorithm



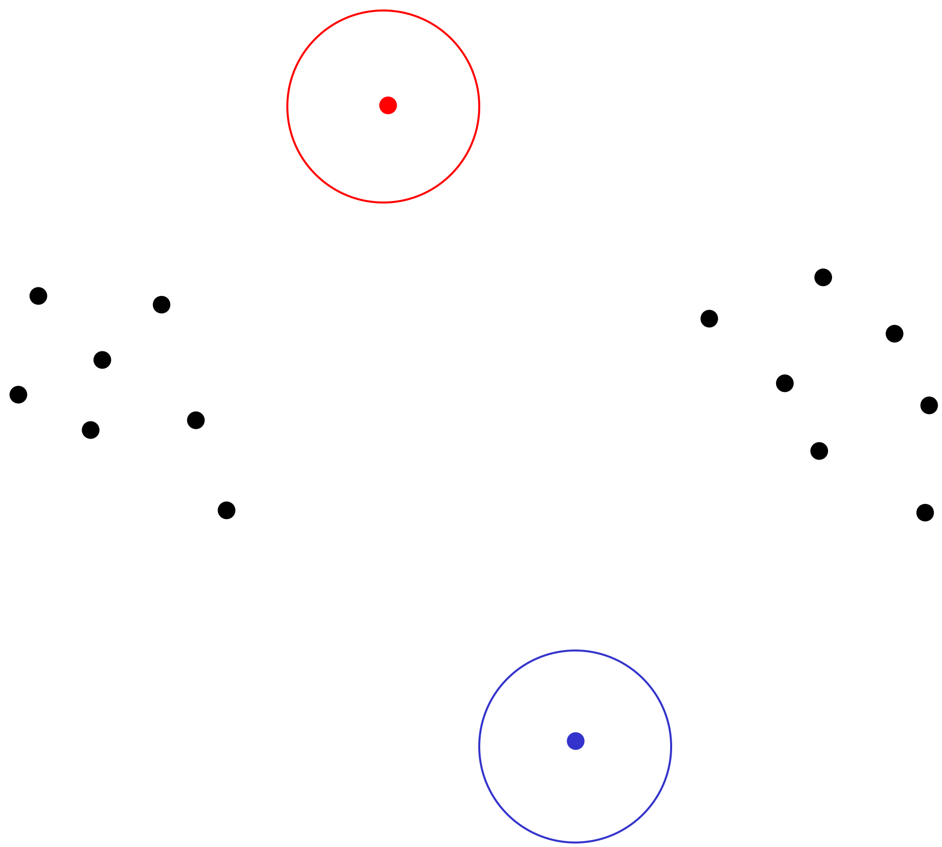
x

z: assignments to cluster

μ, σ, $p(c_j)$: cluster parameters

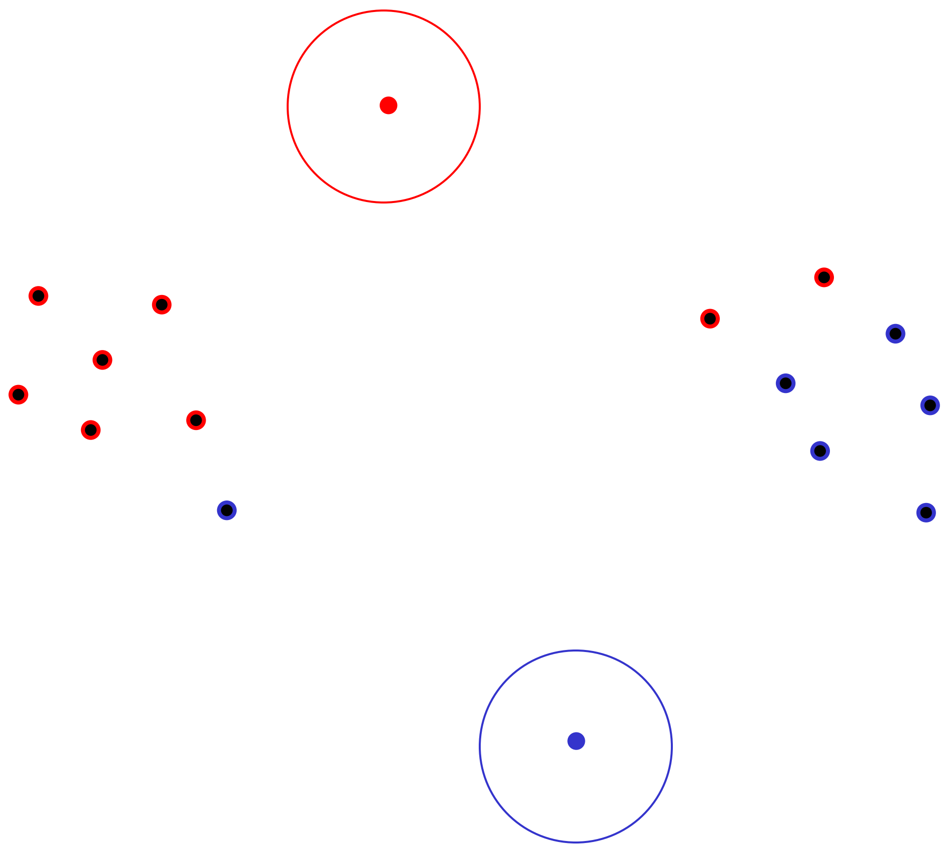
[For simplicity, assume σ, $p(c_j)$ fixed.]

Alternating optimization algorithm



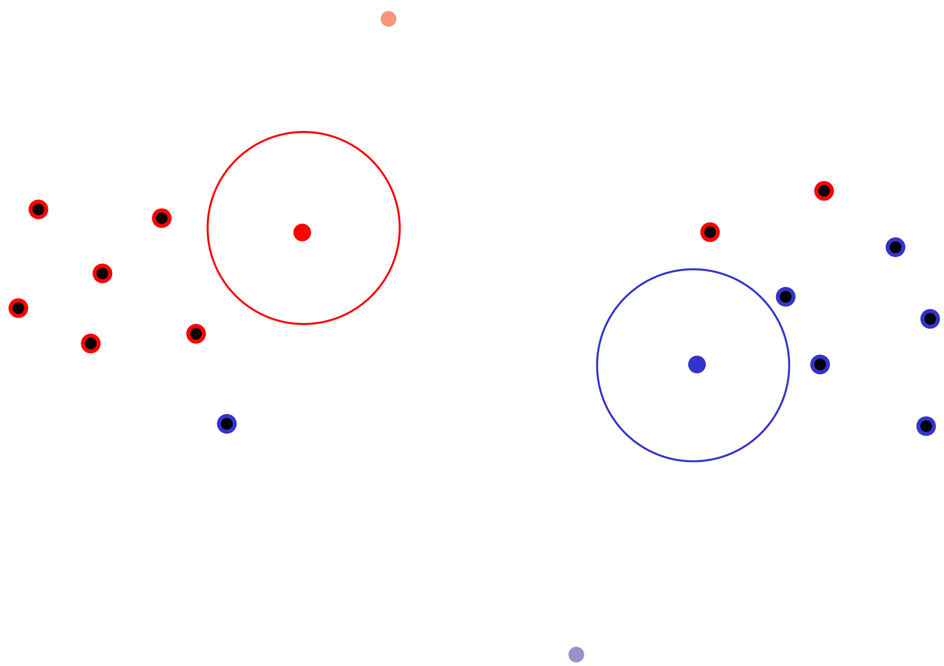
Step 0: initial parameter values

Alternating optimization algorithm



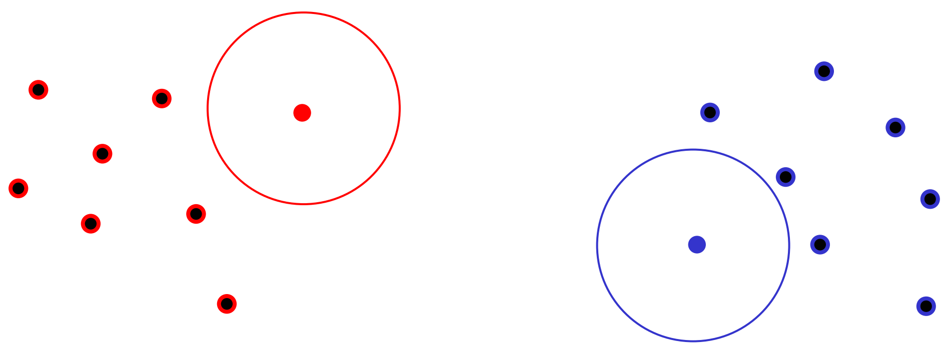
Step 1: update assignments

Alternating optimization algorithm



Step 2: update parameters

Alternating optimization algorithm



Step 1: update assignments

Alternating optimization algorithm

Step 2: update parameters

Alternating optimization algorithm

0. Guess initial parameter values.
1. Given parameter estimates, solve for maximum a posteriori assignments $z_j^{(k)}$:

$$p(c_j \mid \mathbf{x}^{(k)}) \propto \prod_i \frac{1}{\sqrt{2\pi}\sigma_{ij}} e^{-(x_i^{(k)}-\mu_{ij})^2/(2\sigma_{ij}^2)} p(c_j)$$

$$z_j^{(k)} \quad \begin{matrix} 1,\ j \quad \arg\max_{j'} p(c_{j'} \mid \mathbf{x}^{(k)}) \\ 0, \text{ else} \end{matrix}$$

2. Given assignments $z_j^{(k)}$, solve for maximum likelihood parameter estimates:

Why hard assignments?

$$\mu_{ij} \quad \frac{\sum_k z_j^{(k)} x_i^{(k)}}{\sum_k z_j^{(k)}} \qquad \sigma^2_{ij} \quad \frac{\sum_k z_j^{(k)} \left(x_i^{(k)} - \mu_{ij}\right)^2}{\sum_k z_j^{(k)}} \qquad p(c_j) \quad \sum_k z_j^{(k)}$$

3. Go to step 1.

EM algorithm

0. Guess initial parameter values $\theta = \{\mu, \sigma, p(c_j)\}$.
1. **"Expectation" step:** Given parameter estimates, compute expected values of assignments $z_j^{(k)}$

$$h_j^{(k)} \quad p(c_j \mid \mathbf{x}^{(k)}; \theta) \propto \prod_i \frac{1}{\sqrt{2\pi}\sigma_{ij}} e^{-(x_i^{(k)} - \mu_{ij})^2 / (2\sigma_{ij}^2)} p(c_j)$$

2. "**Maximization" step:** Given expected assignments, solve for maximum likelihood parameter estimates:

$$\mu_{ij} \quad \frac{\sum_k h_j^{(k)} x_i^{(k)}}{\sum_k h_j^{(k)}} \qquad \sigma^2{}_{ij} \quad \frac{\sum_k h_j^{(k)} \left(x_i^{(k)} - \mu_{ij}\right)^2}{\sum_k h_j^{(k)}} \qquad p(c_j) \quad \sum_k h_j^{(k)}$$

What EM is really about

- Define a single probabilistic model for the whole data set:

$$p(\mathbf{X} \mid \theta) \quad \prod_k p(\mathbf{x}^{(k)} \mid \theta)$$

$$\prod_k \sum_j p(\mathbf{x}^{(k)} \mid c_j;\theta)\, p(c_j;\theta)$$

"mixture model"

$$\prod_k \sum_j \prod_i \frac{1}{\sqrt{2\pi}\sigma_{ij}} e^{-(x_i^{(k)}-\mu_{ij})^2/(2\sigma_{ij}^2)}\, p(c_j)$$

What EM is really about

- Define a single probabilistic model for the whole data set:

$$p(\mathbf{X} \mid \theta) \quad \prod_k p(\mathbf{x}^{(k)} \mid \theta)$$

$$\prod_k \sum_j p(\mathbf{x}^{(k)} \mid c_j;\theta)\, p(c_j;\theta)$$

"mixture model"

$$\prod_k \sum_j \prod_i \frac{1}{\sqrt{2\pi}\sigma_{ij}} e^{-(x_i^{(k)}-\mu_{ij})^2/(2\sigma_{ij}^2)}\, p(c_j)$$

- How do we maximize w.r.t. θ?

What EM is really about

- Maximization would be simpler if we introduced new labeling variables $\mathbf{Z} = \{z_j^{(k)}\}$:

$$p(\mathbf{X},\mathbf{Z} \mid \theta) \quad \prod_k \prod_j \left(p(\mathbf{x}^{(k)} \mid c_j;\theta)\, p(c_j;\theta) \right)^{z_j^{(k)}}$$

$$\log p(\mathbf{X},\mathbf{Z} \mid \theta) \quad \sum_k \sum_j z_j^{(k)} \sum_i \log\left(p(x_i^{(k)} \mid c_j;\theta)\, p(c_j;\theta) \right)$$

$$-\sum_k \sum_j z_j^{(k)} \sum_i (x_i^{(k)} - \mu_{ij})^2 / (2\sigma_{ij}^2) + \log p(c_j)$$

- Problem: we don’t know $\mathbf{Z} = \{z_j^{(k)}\}$!

What EM is really about

- Maximization expected value of the "complete data" loglikelihood, log $p(\mathbf{X}, \mathbf{Z}|\theta)$:
  - **E-step:** Compute expectation

$$Q(\theta \mid \theta^{(t)}) \quad \sum_{\mathbf{Z}} p(\mathbf{Z} \mid \mathbf{X}, \theta^{(t)}) \log p(\mathbf{X}, \mathbf{Z} \mid \theta)$$

  - **M-step:** Maximize

$$\theta^{(t+1)} \quad \arg\max_{\theta} Q(\theta \mid \theta^{(t)})$$