Outline

- Probabilistic models for unsupervised and semi-supervised category learning
- Nonparametric models for categorization: exemplars, neural networks

EM algorithm

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

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

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

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

What EM is really about

- Want to maximize log $p(\mathbf{X}|\theta)$, e.g.

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

What EM is really about

- Want to maximize $\log p(\mathbf{X}|\theta)$, e.g.

$$\log p(\mathbf{X} \mid \theta) = \sum_k \log \sum_j p(\mathbf{x}^{(k)} \mid c_j;\theta) p(c_j;\theta)$$

What EM is really about

- Want to maximize log $p(\mathbf{X}|\theta)$, e.g.

$$\log p(\mathbf{X} \mid \theta) = \sum_k \log \sum_j p(\mathbf{x}^{(k)} \mid c_j; \theta) p(c_j; \theta)$$

- Instead, maximize expected value of the "complete data" loglikelihood, log $p(\mathbf{X}, \mathbf{Z}|\theta)$:

$$\log p(\mathbf{X}, \mathbf{Z} \mid \theta) = \sum_k \sum_j z_j^{(k)} \log p(\mathbf{x}^{(k)} \mid c_j; \theta) + \log p(c_j; \theta)$$

  – **E-step:** Compute expectation

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

  – **M-step:** Maximize $\theta^{(t+1)} = \arg\max_{\theta} Q(\theta \mid \theta^{(t)})$

Good features of EM

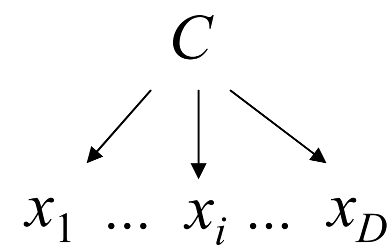
- Convergence
  - Guaranteed to converge to at least a local maximum of the likelihood.
  - Likelihood is non-decreasing across iterations (useful for debugging).
- Efficiency
  - Convergence usually occurs within a few iterations (super-linear).
- Generality
  - Can be defined for many simple probabilistic models.

Limitations of EM

- Local minima
  - E.g., one component poorly fits two clusters, while two components split up a single cluster.
- Degeneracies
  - Two components may merge.
  - A component may lock onto just one data point, with variance going to zero.
- How do you choose number of clusters?
- May be intractable for complex models.

Mixture models for binary data

- Data: $\mathbf{x}^{(k)} = \{x_1^{(k)}, \ldots, x_D^{(k)}\}$, $x_i^{(k)} \in \{0, 1\}$
- Probabilistic model: mixture of Bernoulli distributions (coin flips).



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

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

$$= \prod_k \sum_j p(c_j) \prod_i \mu_{ij}^{x_i^{(k)}} (1 - \mu_{ij})^{(1 - x_i^{(k)})}$$

EM algorithm

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

$$h_j^{(k)} = p(c_j \mid \mathbf{x}^{(k)};\theta) \propto p(c_j)\prod_i \mu_{ij}{}^{x_i^{(k)}}(1-\mu_{ij})^{(1-x_i^{(k)})}$$

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

$$\mu_{ij} = \frac{\sum_k h_j^{(k)} x_i^{(k)}}{\sum_k h_j^{(k)}} \qquad p(c_j) = \sum_k h_j^{(k)}$$

Applications of EM to human learning

- Chicken and egg problems
  - Categories, prototypes
  - Categories, similarity metric (feature weights)

Additive clustering for the integers 0-9:

$$s_{ij} = \sum_{k} w_k f_{ik} f_{jk}$$

Rank	**Weight**	**Stimuli in cluster**										**Interpretation**
		0	1	2	3	4	5	6	7	8	9	
1	.444			*		*				*		powers of two
2	.345	*	*	*								small numbers
3	.331				*			*			*	multiples of three
4	.291							*	*	*	*	large numbers
5	.255			*	*	*	*	*				middle numbers
6	.216		*		*		*		*		*	odd numbers
7	.214		*	*	*	*						smallish numbers
8	.172					*	*	*	*	*		largish numbers

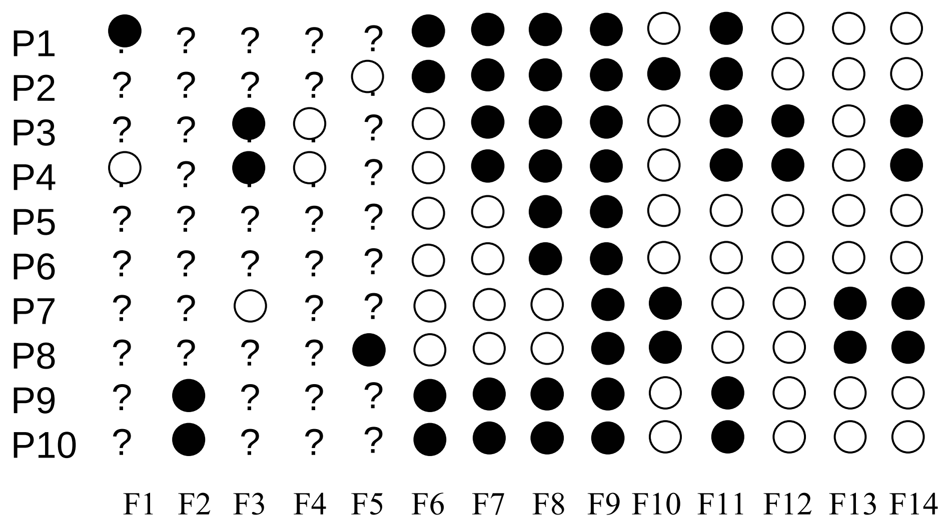
Applications of EM to human learning

- Chicken and egg problems
  - Categories, prototypes
  - Categories, similarity metric (feature weights)
  - Categories, outliers
  - Categories, unobserved features

Learning as interpolation of missing data

- Interpolating a sparse binary matrix:

Objects/ People/ Entities



Features/Concepts/Attributes

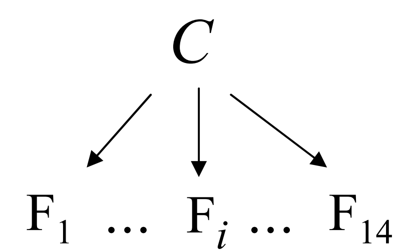
- Interpolating a sparse binary matrix:

Objects/ People/ Entities

	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12	F13	F14
P1	●	?	?	?	?	●	●	●	●	○	●	○	○	○
P2	?	?	?	?	○	●	●	●	●	●	●	○	○	○
P3	?	?	●	○	?	○	●	●	●	○	●	●	○	●
P4	○	?	●	○	?	○	●	●	●	○	●	●	○	●
P5	?	?	?	?	?	○	○	●	●	○	○	○	○	○
P6	?	?	?	?	?	○	○	●	●	○	○	○	○	○
P7	?	?	○	?	?	○	○	○	●	●	○	○	●	●
P8	?	?	?	?	●	○	○	○	●	●	○	○	●	●
P9	?	●	?	?	?	●	●	●	●	○	●	○	○	○
P10	?	●	?	?	?	●	●	●	●	○	●	○	○	○

Features/Concepts/Attributes

- Assume mixture of Bernoulli distributions for objects P_k:

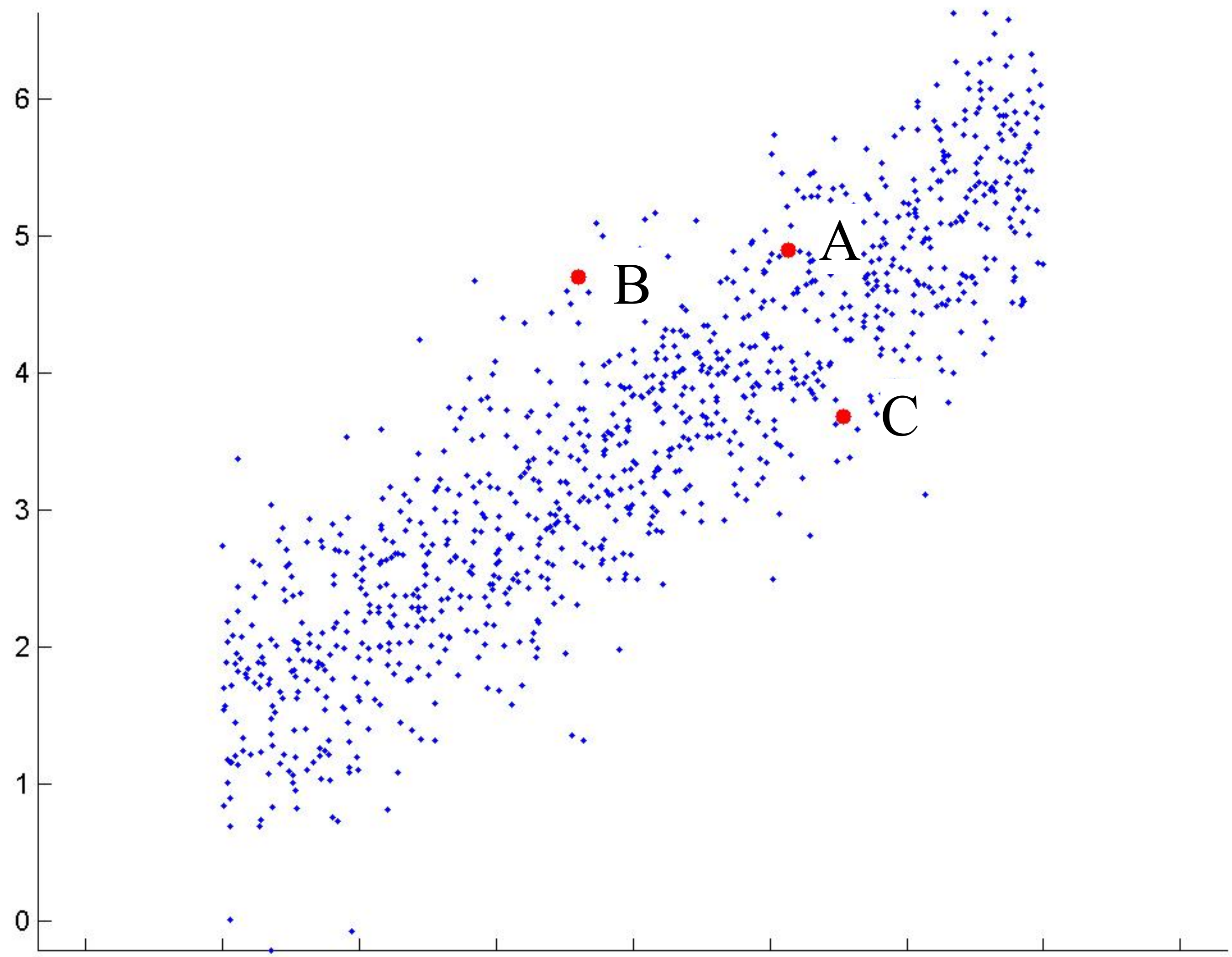


- Learn with EM, treating both class labels and unobserved features as missing data.

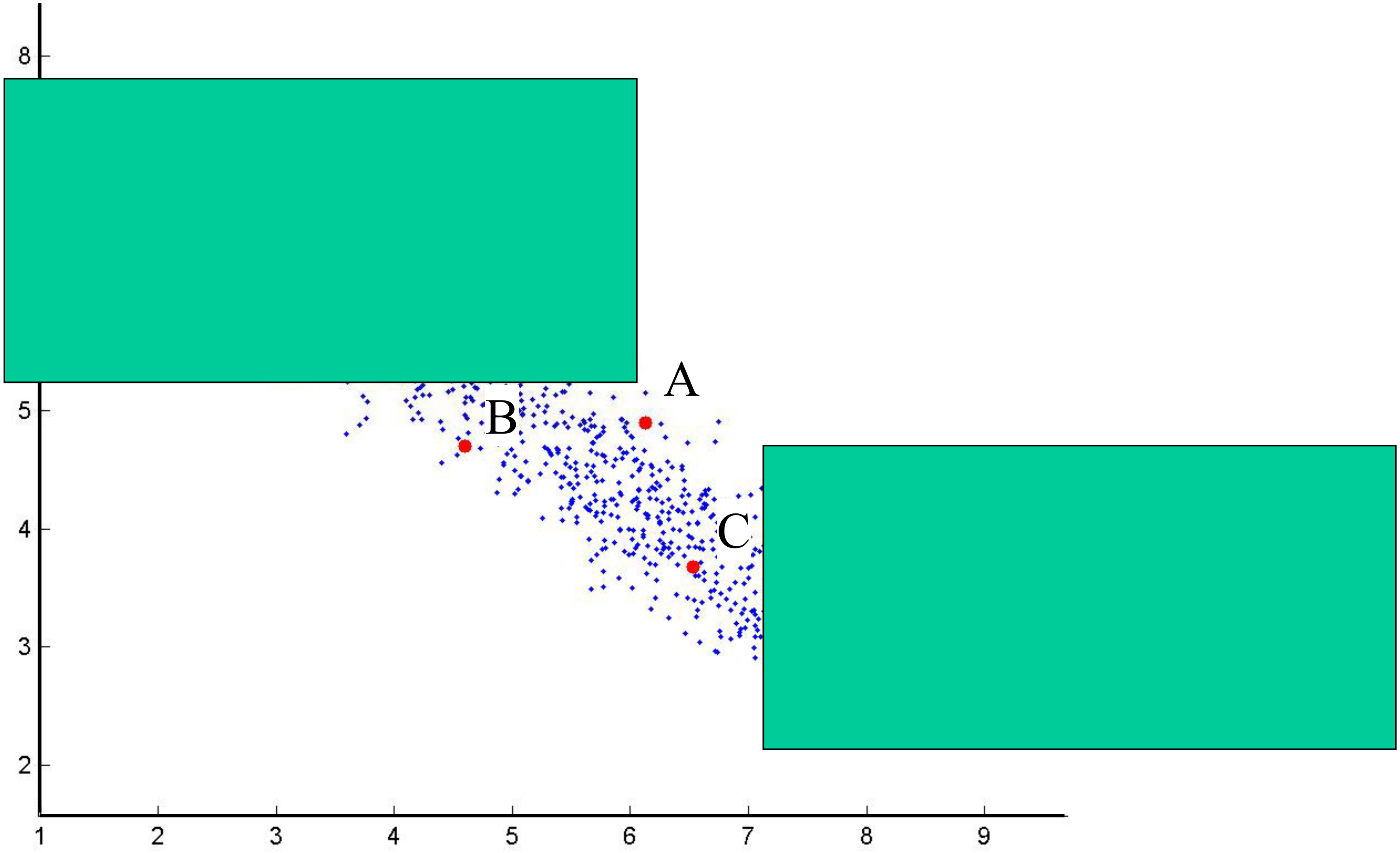
Applications of EM to human learning

- Chicken and egg problems
  - Categories, prototypes
  - Categories, similarity metric (feature weights)
  - Categories, outliers
  - Categories, unobserved features
  - Theories, similarity metric (feature weights)

Is B or C more “similar” to A?



Is B or C more “similar” to A?



EM for factor analysis

- A simple causal theory
  – Generate points at random positions z on a line segment. (*Unobserved "latent" data*)
  – Linearly embed these points (with slope a, intercept b) in two dimensions. (*Observed data*)
  – Add Gaussian noise to one of the two observed dimensions (x-dim or y-dim).
- Examples:
  – Sensory integration
  – Weighing the advice of experts

EM for factor analysis

- A simple causal theory
  - Generate points at random positions z on a line segment. (*Unobserved "latent" data*)
  - Linearly embed these points (with slope a, intercept b) in two dimensions. (*Observed data*)
  - Add Gaussian noise to one of the two observed dimensions (x-dim or y-dim).
- Goal of learning:
  - Estimate parameters: a, b, dimension of noise (x-dim or y-dim)
  - Infer unobserved data: z

Applications of EM to human learning

- Chicken and egg problems
  - Categories, prototypes
  - Categories, similarity metric (feature weights)
  - Categories, unobserved features
  - Categories, outliers
  - Theories, similarity metric (feature weights)
  - Learning in Bayes nets with hidden variables
  - Others?

Fried and Holyoak (1984)

- Can people learn probabilistic categories without labels?
- How does learning with labels differ from learning without labels?
- What kind of concept is learned?
  - Prototype (mean)
  - Prototype + variability (mean + variances)
- Is categorization close to ideal* of a Gaussian mixture model?

Fried and Holyoak stimuli

Image removed due to copyright considerations. Please see:

Fried, L. S., and K. J. Holyoak. “Induction of Category Distributions: A Framework for Classification Learning.” *Journal of Experimental Psychology: Learning, Memory and Cognition* 10 (1984): 234-257.

Fried and Holyoak, Exp. 4

Image removed due to copyright considerations. Please see:

Fried, L. S., and K. J. Holyoak. “Induction of Category Distributions: A Framework for Classification Learning.” *Journal of Experimental Psychology: Learning, Memory and Cognition* 10 (1984): 234-257.

Fried and Holyoak, Exp. 2

Image removed due to copyright considerations. Please see:

Fried, L. S., and K. J. Holyoak. "Induction of Category Distributions: A Framework for Classification Learning." *Journal of Experimental Psychology: Learning, Memory and Cognition* 10 (1984): 234-257.

Fried and Holyoak (1984)

- Can people learn probabilistic categories without labels? Yes.
- How does learning with labels differ from learning without labels? It's better.
- What kind of concept is learned?
  - Prototype (mean)
  - Prototype + variability (mean + variances)
- Is categorization close to ideal* of a Gaussian mixture model? Yes.

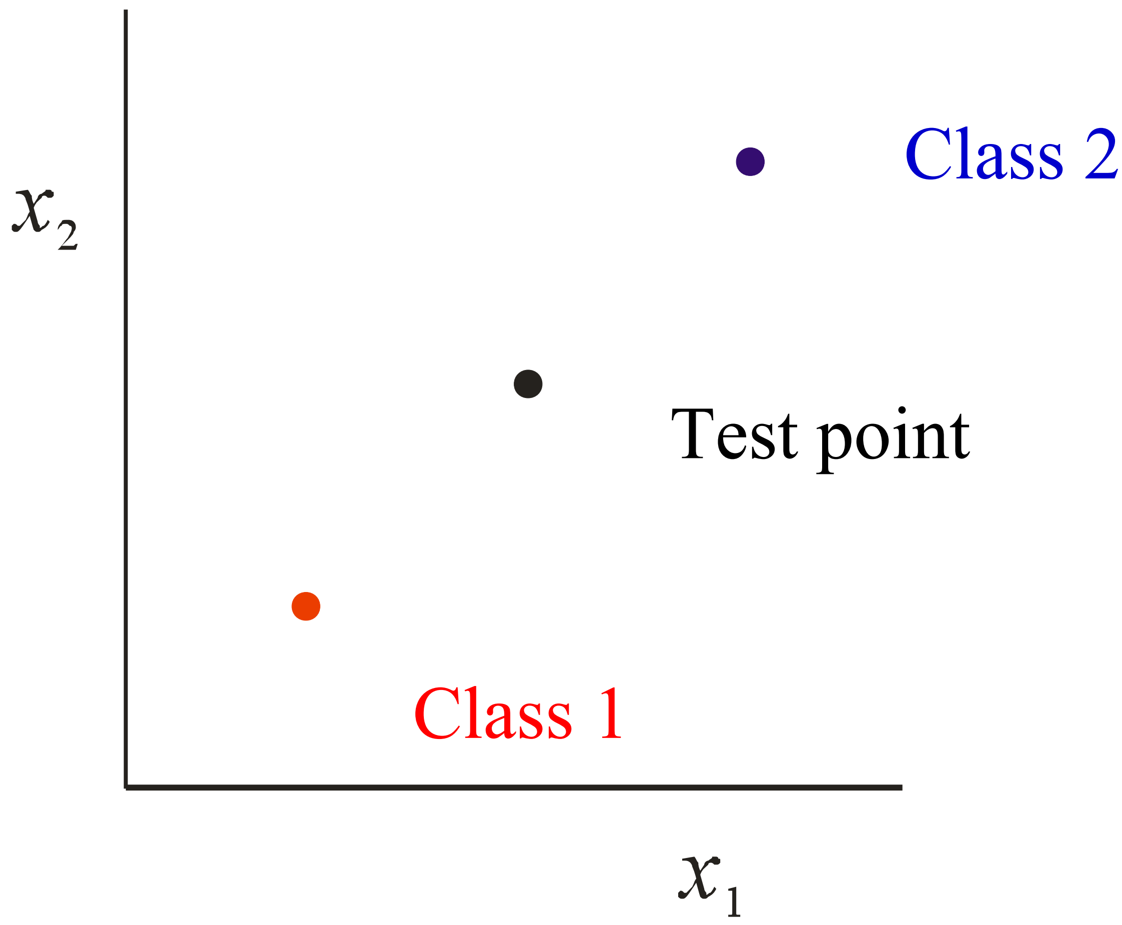
Relevance for human cognition

- How important are these three paradigms for human category learning?
  - Labeled examples
  - Unlabeled examples
  - Unlabeled examples but known # of classes
- Other ways of combining labeled and unlabeled examples that are worth pursuing?

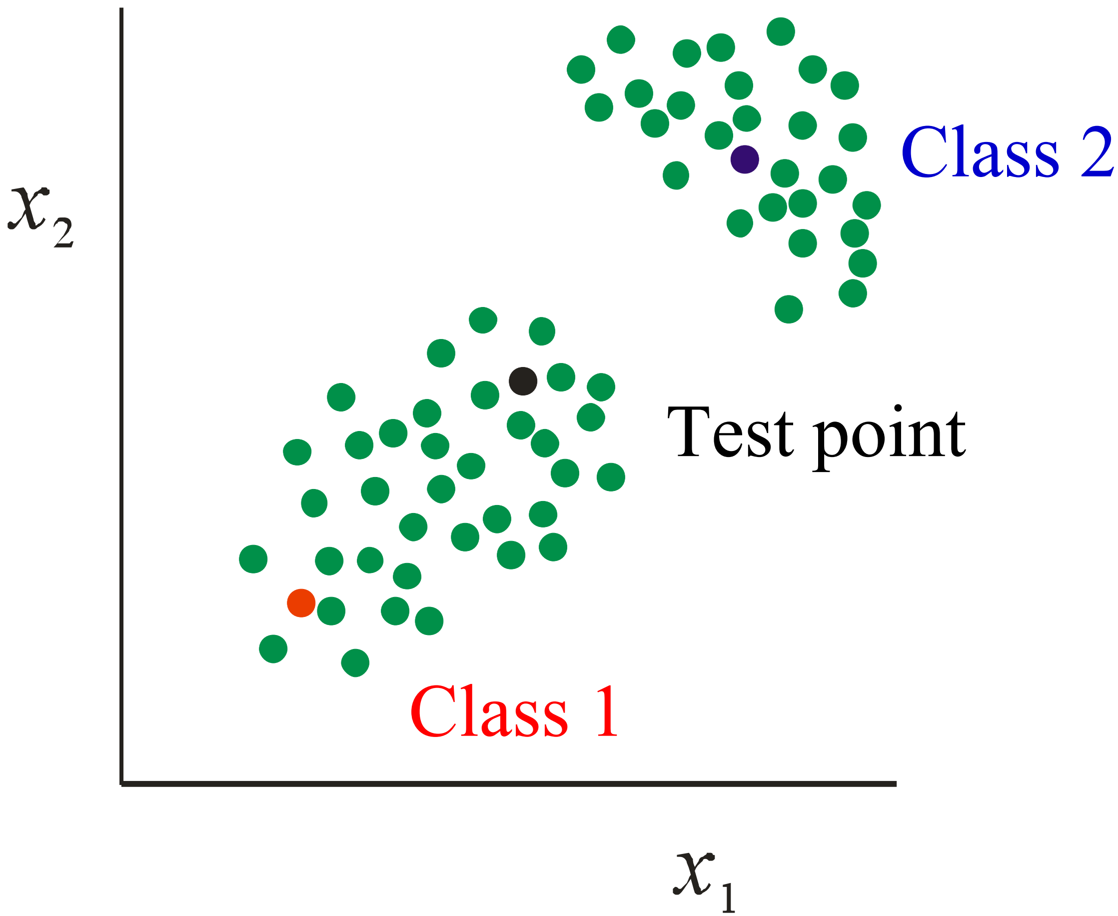
Semi-supervised learning

- Learning with many unlabeled examples and a small number of labeled examples.
- Important area of current work in machine learning.
  - E.g., learning about the web (or any large corpus)
- Natural situation in human learning.
  - E.g., word learning
  - Not much research here though….

The benefit of unlabelled data



The benefit of unlabelled data

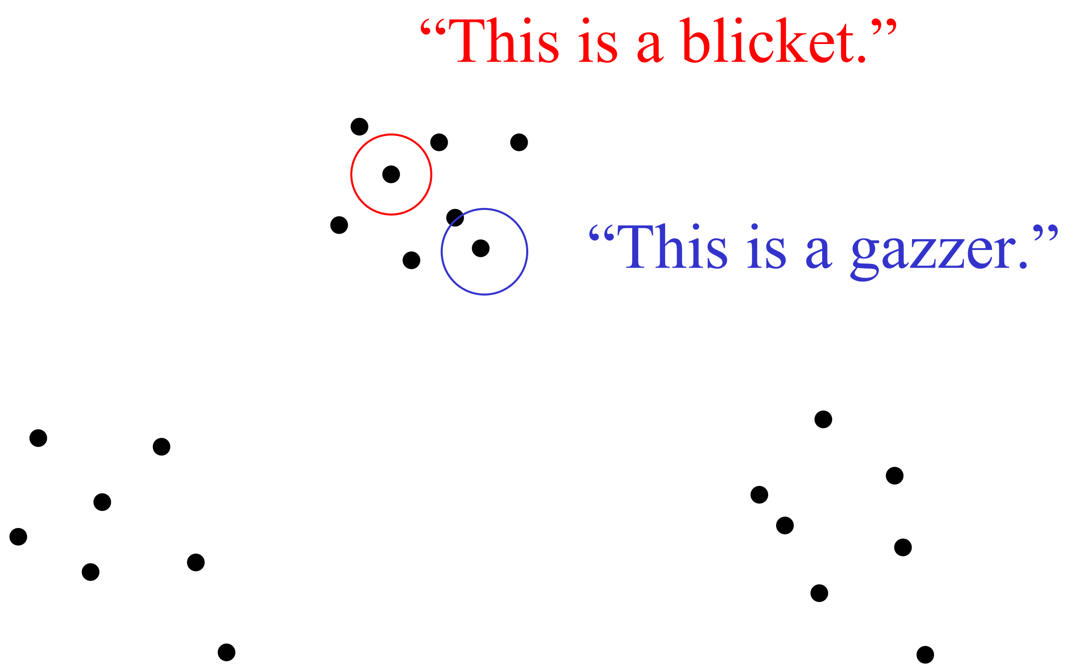


Is this really a new problem?

- Why not just do unsupervised clustering first and then label the clusters?

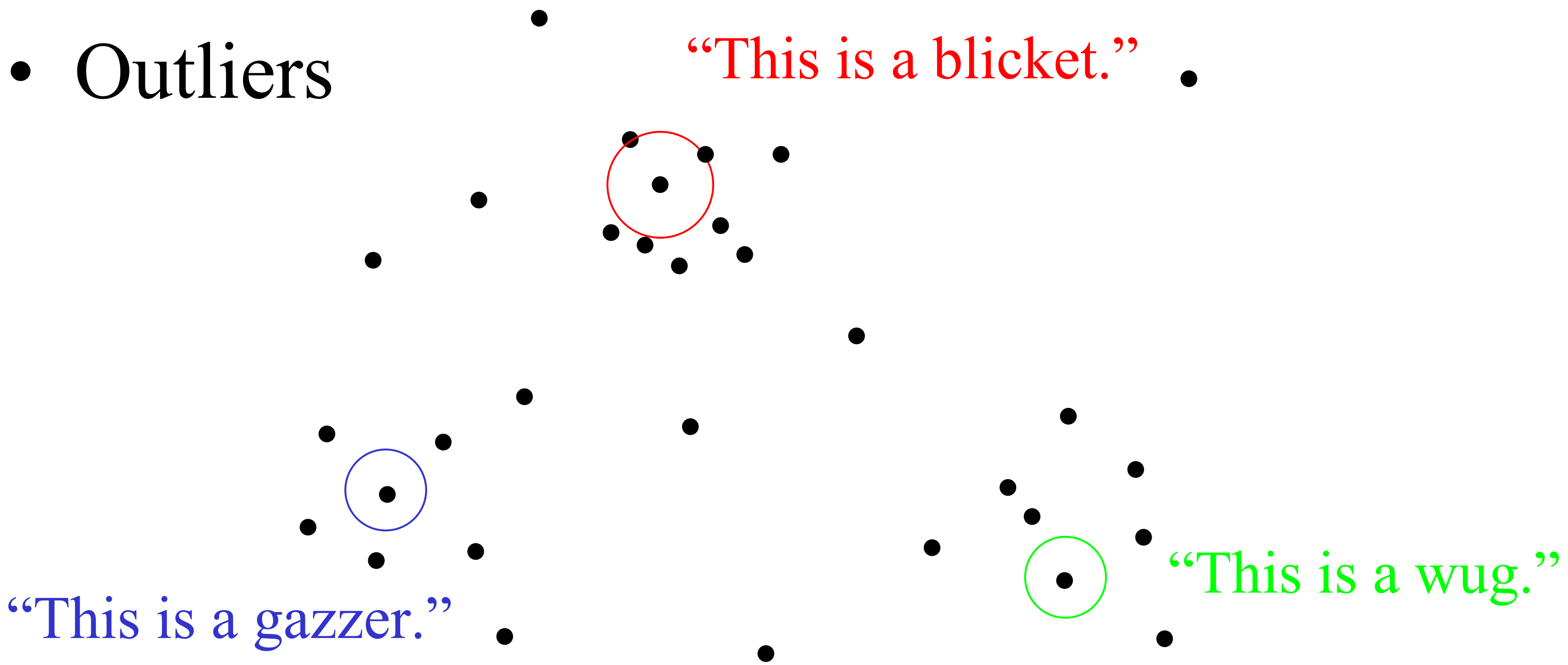
Complications

- Concept labels inconsistent with clusters



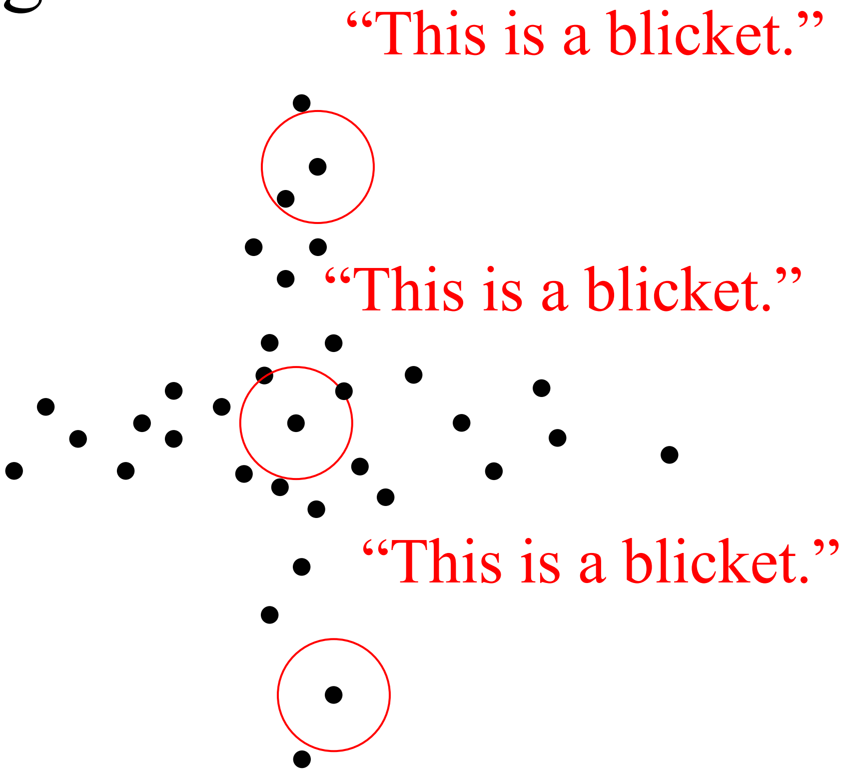
Complications

- Outliers



Complications

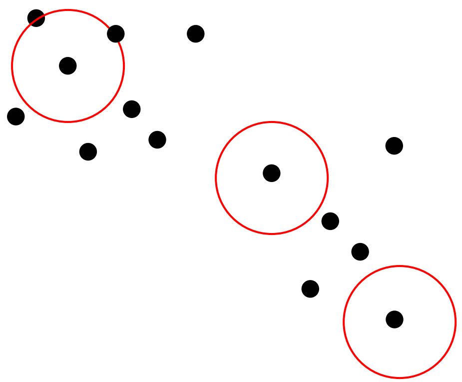
- Overlapping clusters



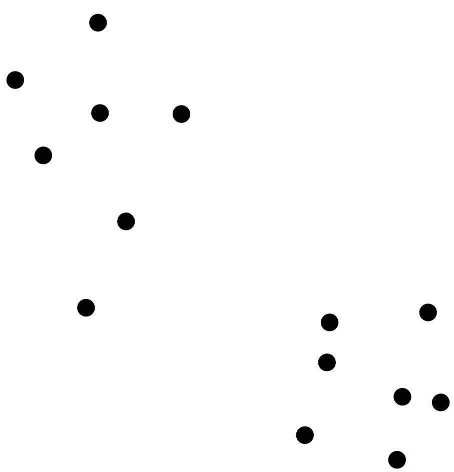
Complications

- How many clusters?

“This is a blicket.”



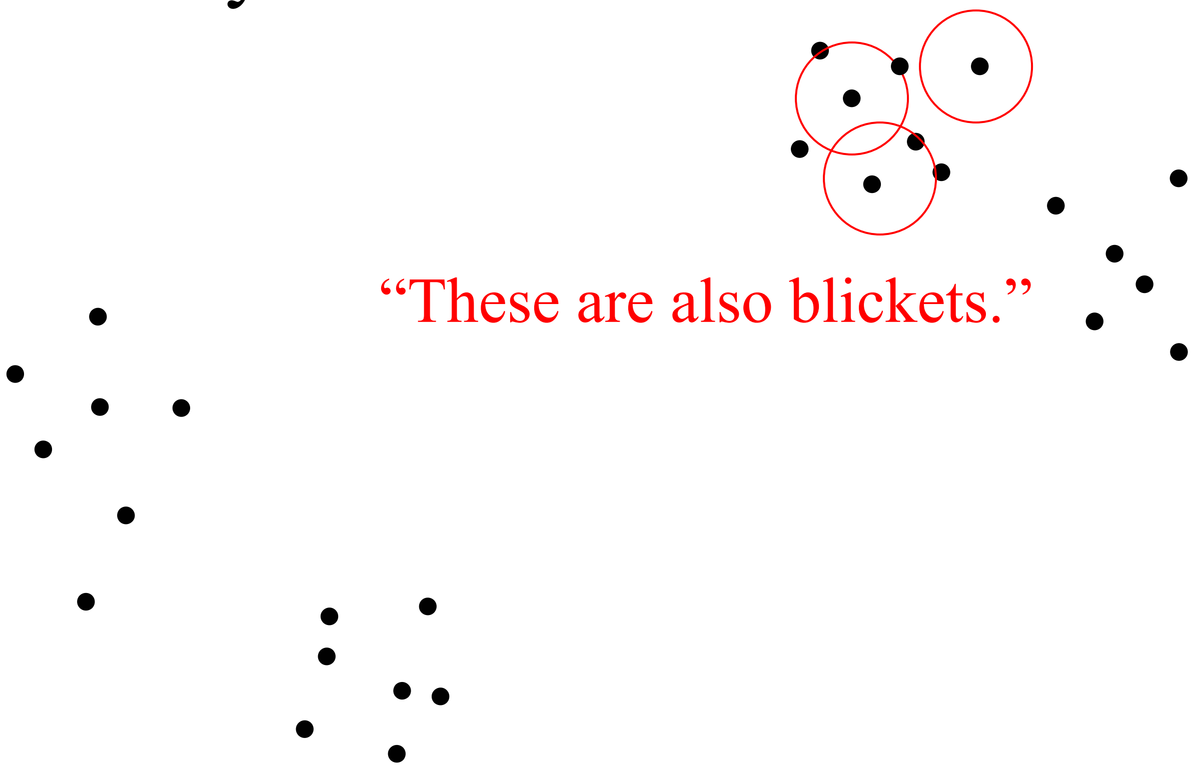
“These are also blickets.”



Complications

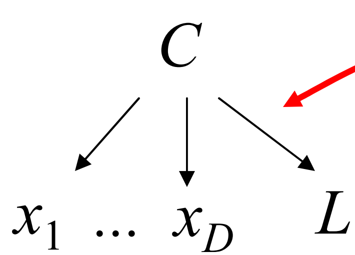
- How many clusters?

“This is a blicket.”



Semi-supervised learning

- Learning with many unlabeled examples and a small number of labeled examples.
- Approaches based on EM with mixtures
  - Identify each concept with one mixture component.
    - Labels serve to anchor class assignments in E-step.

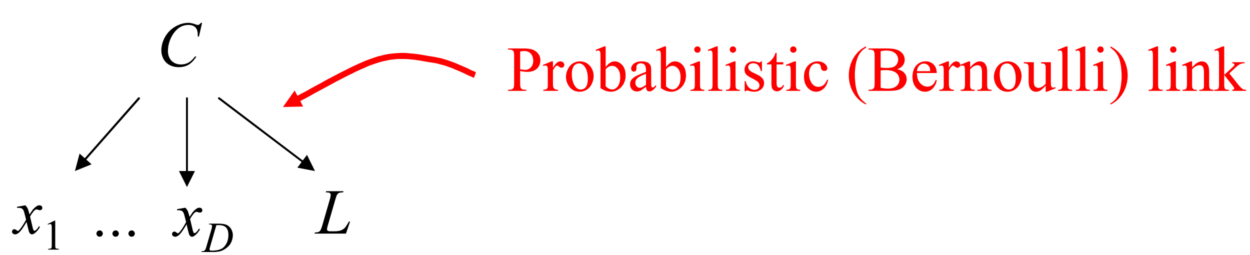


Deterministic (1-to-1) link

(Could also be many-to-1.)

Semi-supervised learning

- Learning with many unlabeled examples and a small number of labeled examples.
- Approaches based on EM with mixtures
  - Treat concept labels as separate features, conditionally independent of observed features given classes.
    - e.g., Ghahramani and Jordan (cf. Anderson).



Other approaches to semi-supervised learning

- Graph-based
  - Szummer & Jaakkola
  - Zhu, Ghahramani & Lafferty
  - Belkin & Niyogi
  - Blum & Chawla
- Tree-based
  - Tenenbaum and Xu; Kemp, Tenenbaum, et al.

Graph-based semi-supervised learning

E.g., Class labeling function is smooth over a graph of *k*-nearest neighbors:

Image removed due to copyright considerations.

Tree-based semi-supervised learning

- A motivating problem: learning words for kinds of objects

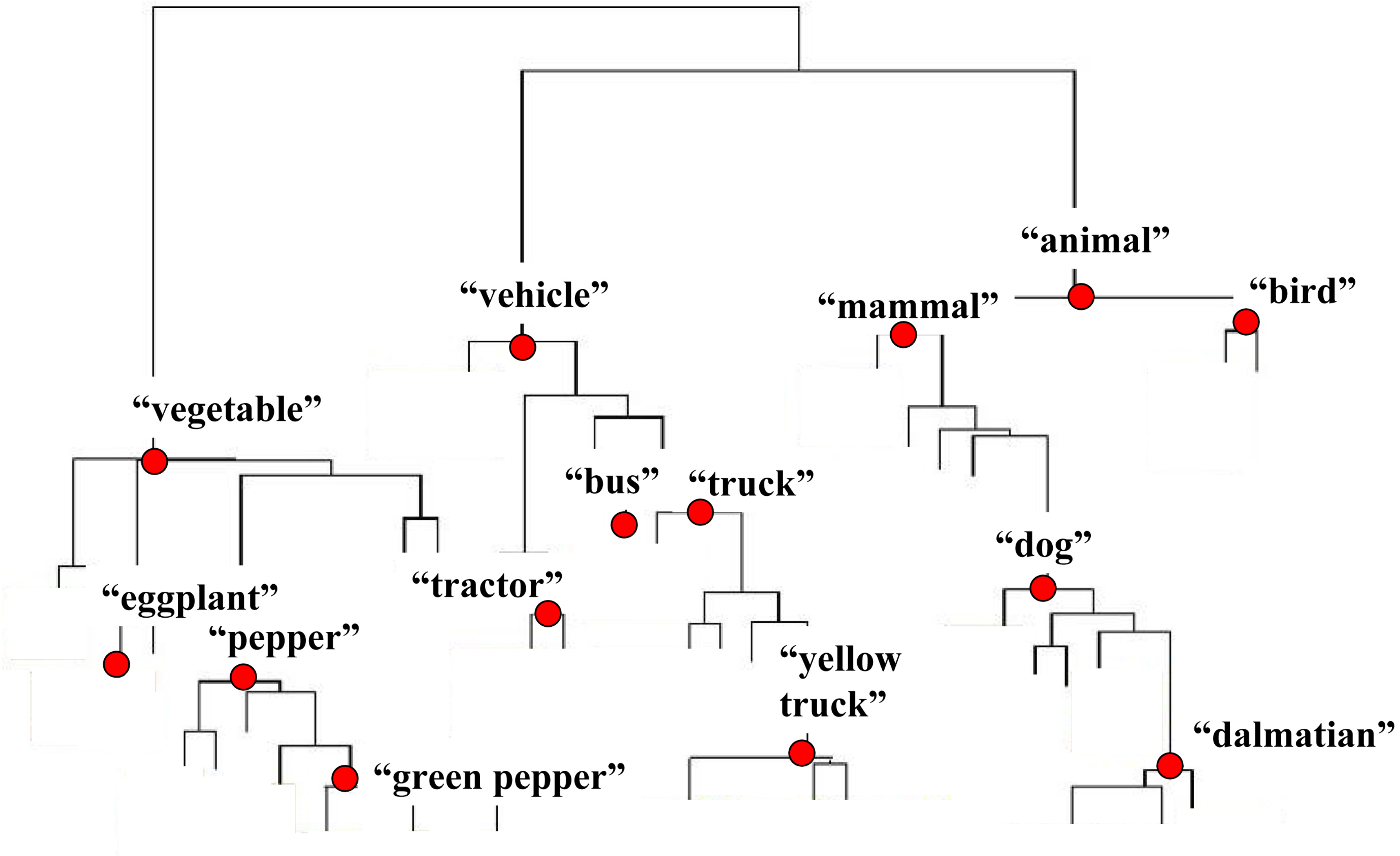
Image removed due to copyright considerations.

What does the word “dog” refer to?

- All (and only) dogs?
- All mammals?
- All animals?
- All labradors?
- All yellow labradors?
- Undetached dog parts?
- All dogs plus Silver?
- All yellow things?
- All running things?
- . . .

Image removed due to copyright considerations.

Similarity → Clusters → Hypotheses



Images removed due to copyright considerations.

Bayesian model of word learning

- H: Hypotheses correspond to taxonomic clusters
  - h_1 = "all (and only) dogs"
  - h_2 = "all mammals"
  - h_3 = "all animals"
  - h_4 = "all labradors"
  - . . .
- Same model as for learning number concepts, but with two new features specific to this task:
  - Prior favors more distinctive taxonomic clusters.
  - Prior favors naming categories at a privileged (basic) level.

Image removed due to copyright considerations.

Bayes (with basic-level bias)

Bayes (*without* basic-level bias)

Image removed due to copyright considerations.

The objects of planet Gazoob

Image removed due to copyright considerations.

Image removed due to copyright considerations.

Adults:

Image removed due to copyright considerations.

Bayes: (with basic-level bias)

Image removed due to copyright considerations.

Semi-supervised learning?

- Interpolating a sparse binary matrix:

Objects

	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12	F13	F14
P1	●	?	?	?	?	●	●	●	●	○	●	○	○	○
P2	?	?	?	?	?	●	●	●	●	●	●	○	○	○
P3	?	?	●	?	?	○	●	●	●	○	●	●	○	●
P4	?	?	●	●	?	○	●	●	●	○	●	●	○	●
P5	?	?	?	?	?	○	○	●	●	○	○	○	○	○
P6	?	?	?	?	?	○	○	●	●	○	○	○	○	○
P7	?	?	?	?	?	○	○	○	●	●	○	○	●	●
P8	?	?	?	?	●	○	○	○	●	●	○	○	●	●
P9	?	●	●	?	?	●	●	●	●	○	●	○	○	○
P10	?	●	?	?	?	●	●	●	●	○	●	○	○	○

Words | **Features**

- Use features to infer tree or graph over objects.
- Use tree or graph to generate priors for the extensions of words.