Hidden Variables, the EM Algorithm, and Mixtures of Gaussians

Computer Vision

CS 543 / ECE 549

University of Illinois

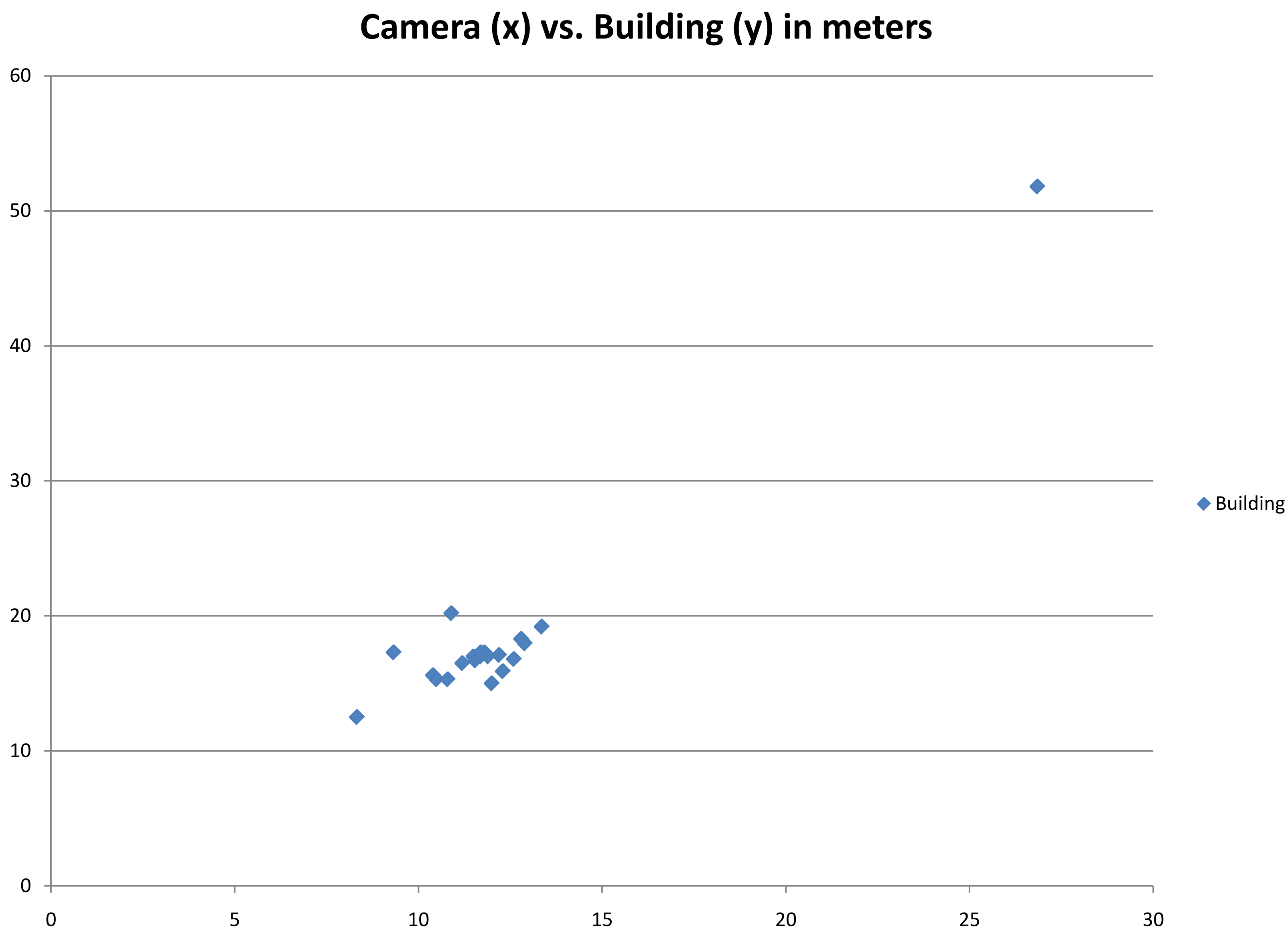
Derek Hoiem

HW 1 is graded

- Mean = 93, Median = 98

- A few comments
  - Diffuse component for estimating light color
  - Make sure to choose an appropriate size filter
  - Comparing frequencies for images at multiple scales
  - Wide variety of interesting apps
    - Maybe some would make good final project?

HW 1: Estimating Camera/Building Height



Today’s Class

- Examples of Missing Data Problems
  - Detecting outliers
  - Latent topic models (HW 2, problem 3)
  - Segmentation (HW 2, problem 4)

- Background
  - Maximum Likelihood Estimation
  - Probabilistic Inference

- Dealing with “Hidden” Variables
  - EM algorithm, Mixture of Gaussians
  - Hard EM

Missing Data Problems: Outliers

You want to train an algorithm to predict whether a photograph is attractive. You collect annotations from Mechanical Turk. Some annotators try to give accurate ratings, but others answer randomly.

Challenge: Determine which people to trust and the average rating by accurate annotators.

Annotator Ratings

10
8
9
2
8

Photo: Jam343 (Flickr)

Missing Data Problems: Object Discovery

You have a collection of images and have extracted regions from them. Each is represented by a histogram of "visual words".

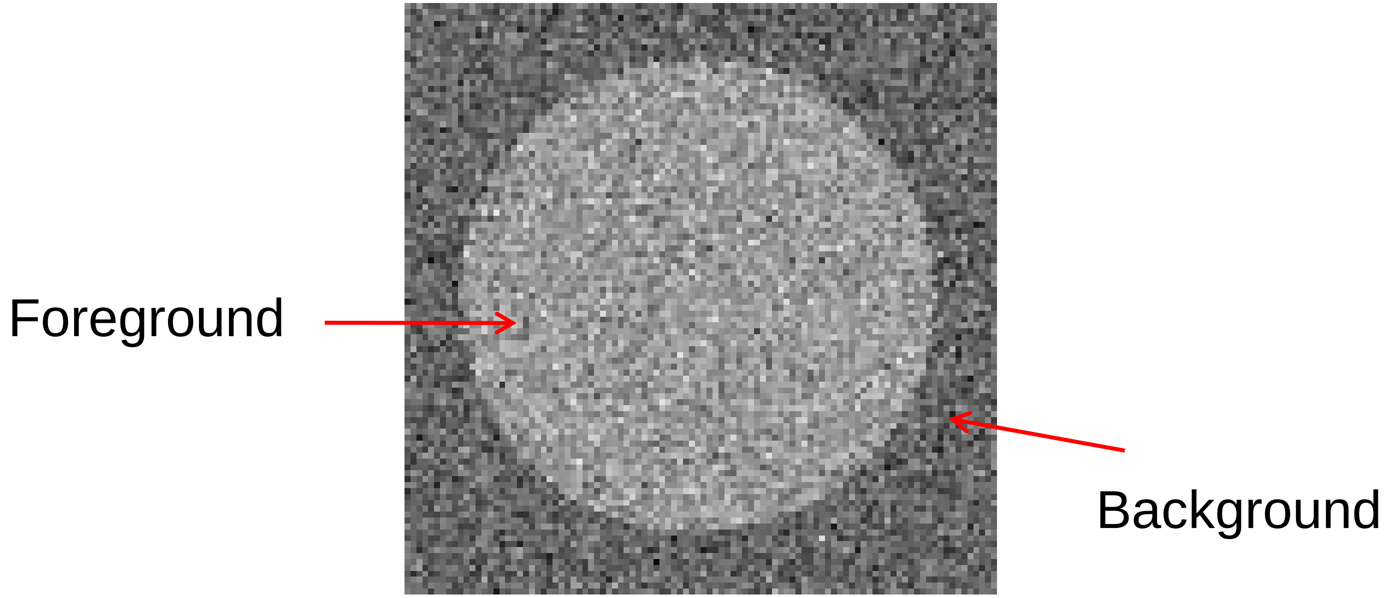
Challenge: Discover frequently occurring object categories, without pre-trained appearance models.

http://www.robots.ox.ac.uk/~vgg/publications/papers/russell06.pdf

Missing Data Problems: Segmentation

You are given an image and want to assign foreground/background pixels.

Challenge: Segment the image into figure and ground without knowing what the foreground looks like in advance.

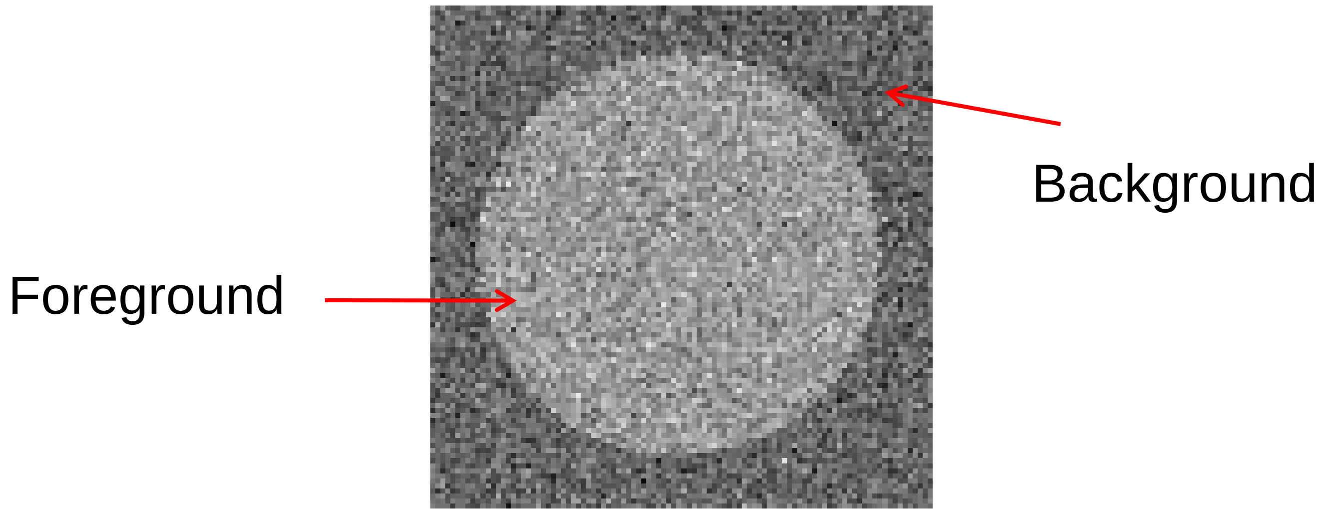


Missing Data Problems: Segmentation

Challenge: Segment the image into figure and ground without knowing what the foreground looks like in advance.

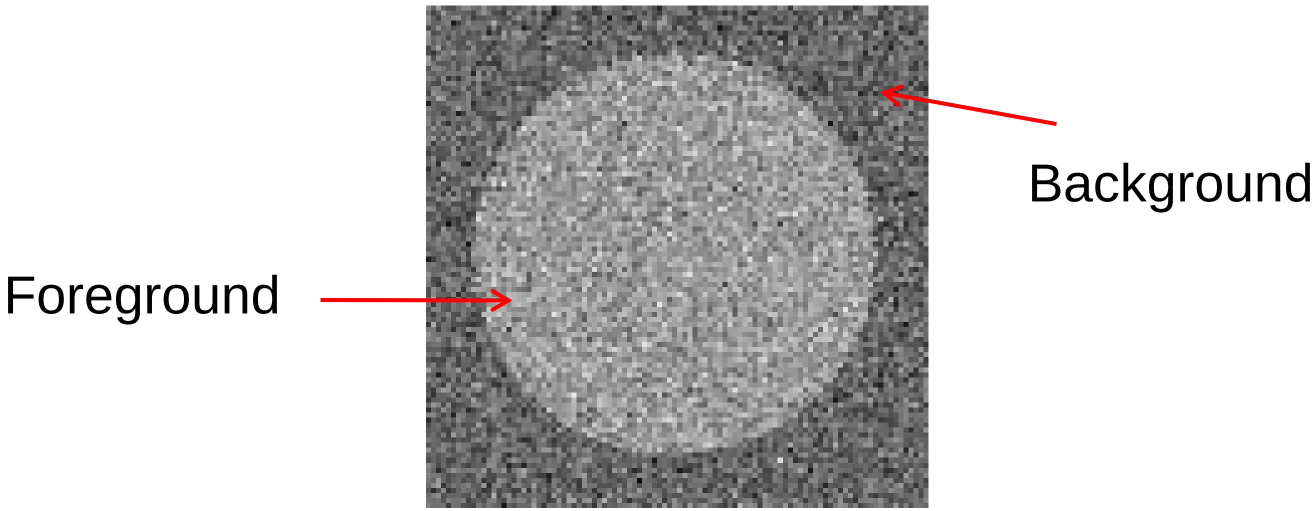
Three steps:

1. If we had labels, how could we model the appearance of foreground and background?
2. Once we have modeled the fg/bg appearance, how do we compute the likelihood that a pixel is foreground?
3. How can we get both labels and appearance models at once?



Maximum Likelihood Estimation

1. If we had labels, how could we model the appearance of foreground and background?



Maximum Likelihood Estimation

data

$$\mathbf{x} = \{x_1 .. x_N\}$$

parameters

$$\hat{\theta} = \underset{\theta}{\operatorname{argmax}}\; p(\mathbf{x} \mid \theta)$$

$$\hat{\theta} = \underset{\theta}{\operatorname{argmax}} \prod_n p(x_n \mid \theta)$$

Maximum Likelihood Estimation

$$\mathbf{x} = \{x_1..x_N\}$$

$$\hat{\theta} = \underset{\theta}{\operatorname{argmax}}\, p(\mathbf{x} \mid \theta)$$

$$\hat{\theta} = \underset{\theta}{\operatorname{argmax}} \prod_n p(x_n \mid \theta)$$

Gaussian Distribution

$$p(x_n \mid \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(- \frac{(x_n - \mu)^2}{2\sigma^2} \right)$$

Maximum Likelihood Estimation

$$\mathbf{x} = \left\{x_1 .. x_N\right\}$$

$$\hat{\theta} = \underset{\theta}{\operatorname{argmax}}\, p(\mathbf{x} \mid \theta)$$

$$\hat{\theta} = \underset{\theta}{\operatorname{argmax}} \prod_n p(x_n \mid \theta)$$

Gaussian Distribution

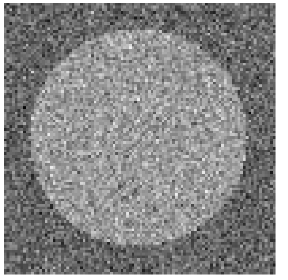
$$p(x_n \mid \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{\left(x_n - \mu\right)^2}{2\sigma^2} \right)$$

$$\hat{\mu} = \frac{1}{N} \sum_n x_n \qquad \hat{\sigma}^2 = \frac{1}{N} \sum_n \left(x_n - \hat{\mu}\right)^2$$

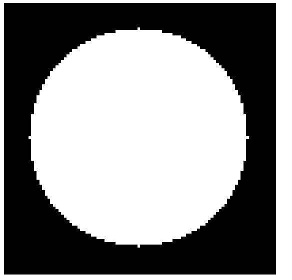
Example: MLE

Parameters used to Generate

```
fg: mu=0.6, sigma=0.1
bg: mu=0.4, sigma=0.1
```



im

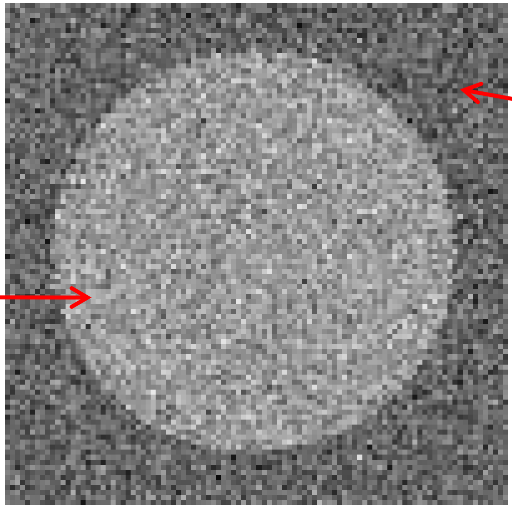


labels

```
>> mu_fg = mean(im(labels))
      mu_fg = 0.6012

>> sigma_fg = sqrt(mean((im(labels)-mu_fg).^2))
      sigma_fg = 0.1007

>> mu_bg = mean(im(~labels))
      mu_bg = 0.4007

>> sigma_bg = sqrt(mean((im(~labels)-mu_bg).^2))
      sigma_bg = 0.1007

>> pfg = mean(labels(:));
```

Probabilistic Inference

2. Once we have modeled the fg/bg appearance, how do we compute the likelihood that a pixel is foreground?



Probabilistic Inference

Compute the likelihood that a particular model generated a sample

component or label

$$p(z_n = m \mid x_n, \theta)$$

Probabilistic Inference

Compute the likelihood that a particular model generated a sample

component or label

$$p(z_n = m \mid x_n, \theta) = \frac{p(z_n = m, x_n \mid \theta_m)}{p(x_n \mid \theta)}$$

Probabilistic Inference

Compute the likelihood that a particular model generated a sample

component or label

$$p(z_n = m \mid x_n, \theta) = \frac{p\left(z_n = m, x_n \mid \theta_m\right)}{p\left(x_n \mid \theta\right)}$$

$$= \frac{p\left(z_n = m, x_n \mid \theta_m\right)}{\sum_k p\left(z_n = k, x_n \mid \theta_k\right)}$$

Probabilistic Inference

Compute the likelihood that a particular model generated a sample

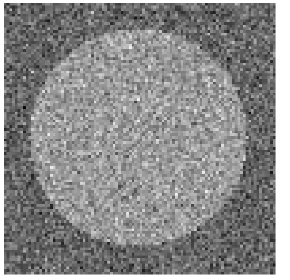
component or label

$$p(z_n = m \mid x_n, \theta) = \frac{p(z_n = m, x_n \mid \theta_m)}{p(x_n \mid \theta)}$$

$$= \frac{p(z_n = m, x_n \mid \theta_m)}{\sum_k p(z_n = k, x_n \mid \theta_k)}$$

$$= \frac{p(x_n \mid z_n = m, \theta_m)\, p(z_n = m \mid \theta_m)}{\sum_k p(x_n \mid z_n = k, \theta_k)\, p(z_n = k \mid \theta_k)}$$

Example: Inference

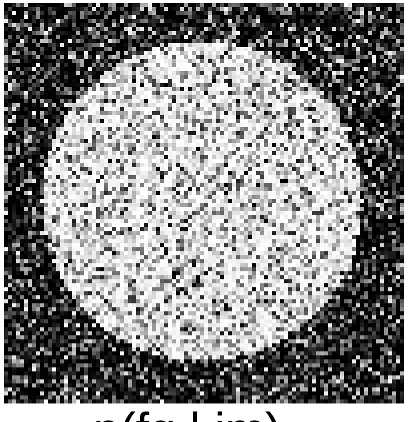


im

Learned Parameters

```
fg: mu=0.6, sigma=0.1
bg: mu=0.4, sigma=0.1
```

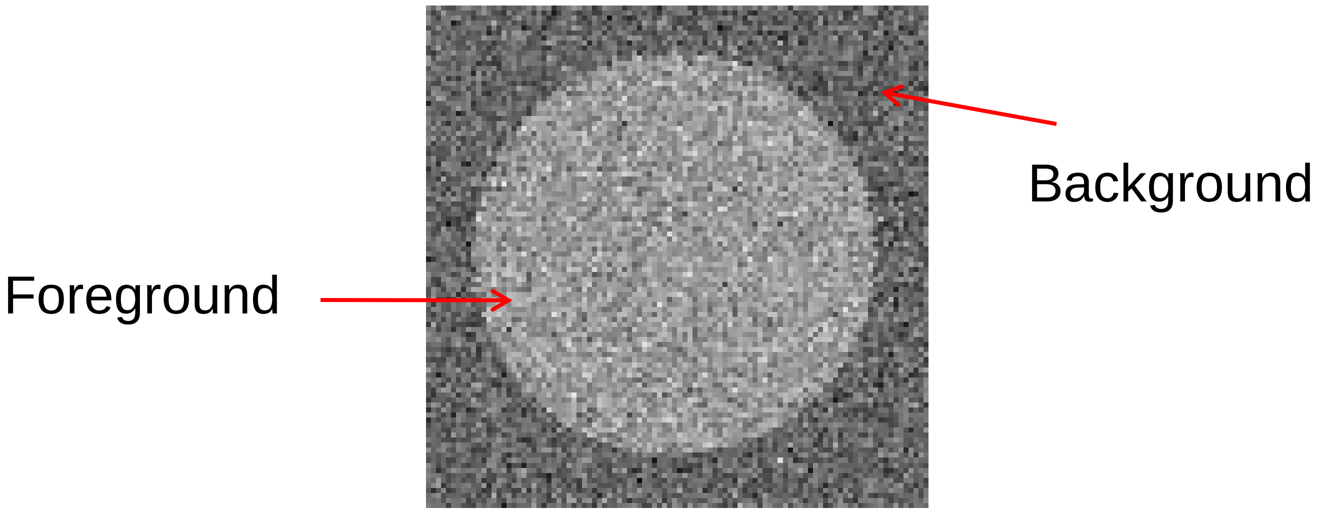
```
>> pfg = 0.5;
>> px_fg = normpdf(im, mu_fg, sigma_fg);
>> px_bg = normpdf(im, mu_bg, sigma_bg);
>> pfg_x = px_fg*pfg ./ (px_fg*pfg + px_bg*(1-pfg));
```



p(fg | im)

Dealing with Hidden Variables

3. How can we get both labels and appearance models at once?



Mixture of Gaussians

mixture component

component model parameters

component prior

$$p\left(x_n \mid \boldsymbol{\mu}, \boldsymbol{\sigma}^2, \boldsymbol{\pi}\right) = \sum_m p\left(x_n, z_n = m \mid \mu_m, \sigma_m{}^2, \pi_m\right)$$

$$p\left(x_n, z_n = m \mid \boldsymbol{\mu}, \boldsymbol{\sigma}^2, \boldsymbol{\pi}\right) = p\left(x_n, z_n = m \mid \mu_m, \sigma_m{}^2, \pi_m\right)$$

$$= p\left(x_n \mid \mu_m, \sigma_m{}^2\right) p\left(z_n = m \mid \pi_m\right)$$

$$= \frac{1}{\sqrt{2\pi\sigma_m{}^2}} \exp\left(-\frac{\left(x_n - \mu_m\right)^2}{2\sigma_m{}^2}\right) \cdot \pi_m$$

Mixture of Gaussians

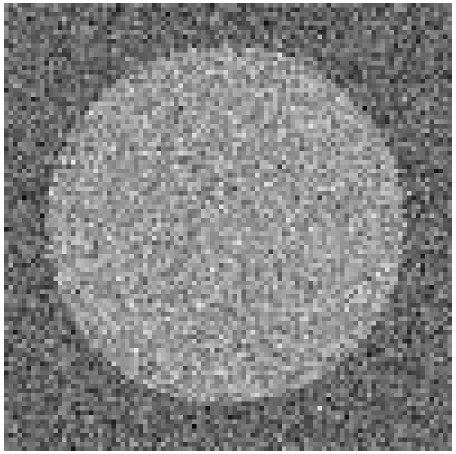
With enough components, can represent any probability density function

– Widely used as general purpose pdf estimator

Segmentation with Mixture of Gaussians

Pixels come from one of several Gaussian components

- We don’t know which pixels come from which components
- We don’t know the parameters for the components



Simple solution

1. Initialize parameters

2. Compute the probability of each hidden variable given the current parameters

3. Compute new parameters for each model, weighted by likelihood of hidden variables

4. Repeat 2-3 until convergence

Mixture of Gaussians: Simple Solution

1. Initialize parameters

2. Compute likelihood of hidden variables for current parameters

$$\alpha_{nm} = p(z_n = m \mid x_n, \boldsymbol{\mu}^{(t)}, \boldsymbol{\sigma}^{2^{(t)}}, \boldsymbol{\pi}^{(t)})$$

3. Estimate new parameters for each model, weighted by likelihood

$$\hat{\mu}_m^{(t+1)} = \frac{1}{\sum_n \alpha_{nm}} \sum_n \alpha_{nm} x_n \qquad \hat{\sigma}_m^{2^{(t+1)}} = \frac{1}{\sum_n \alpha_{nm}} \sum_n \alpha_{nm} \left(x_n - \hat{\mu}_m\right)^2 \qquad \hat{\pi}_m^{(t+1)} = \frac{\sum_n \alpha_{nm}}{N}$$

Expectation Maximization (EM) Algorithm

Goal: $\hat{\theta} = \underset{\theta}{\operatorname{argmax}} \log\left(\sum_{\mathbf{z}} p(\mathbf{x}, \mathbf{z} \mid \theta) \right)$

Log of sums is intractable

Jensen's Inequality

$$f\left(\mathrm{E}[X]\right) \geq \mathrm{E}\left[f(X)\right]$$

for concave funcions, such as f(x)=log(x)

See here for proof: www.stanford.edu/class/cs229/notes/cs229-notes8.ps

Expectation Maximization (EM) Algorithm

Goal: $\hat{\theta} = \underset{\theta}{\operatorname{argmax}} \log\left(\sum_{\mathbf{z}} p(\mathbf{x}, \mathbf{z} \mid \theta)\right)$

1. E-step: compute

$$\mathrm{E}_{z|x,\theta^{(t)}}\left[\log\left(p(\mathbf{x}, \mathbf{z} \mid \theta)\right)\right] = \sum_{\mathbf{z}} \log\left(p(\mathbf{x}, \mathbf{z} \mid \theta)\right) p\left(\mathbf{z} \mid \mathbf{x}, \theta^{(t)}\right)$$

2. M-step: solve

$$\theta^{(t+1)} = \underset{\theta}{\operatorname{argmax}} \sum_{\mathbf{z}} \log\left(p(\mathbf{x}, \mathbf{z} \mid \theta)\right) p\left(\mathbf{z} \mid \mathbf{x}, \theta^{(t)}\right)$$

EM for Mixture of Gaussians (on board)

$$p(x_n \mid \boldsymbol{\mu}, \boldsymbol{\sigma}^2, \boldsymbol{\pi}) = \sum_m p(x_n, z_n = m \mid \mu_m, \sigma_m{}^2, \pi_m) = \sum_m \frac{1}{\sqrt{2\pi\sigma_m{}^2}} \exp\left(-\frac{(x_n - \mu_m)^2}{\sigma_m{}^2}\right) \cdot \pi_m$$

1. E-step: $\mathrm{E}_{z|x,\theta^{(t)}}\left[\log(p(\mathbf{x}, \mathbf{z} \mid \theta))\right] = \sum_{\mathbf{z}} \log(p(\mathbf{x}, \mathbf{z} \mid \theta))\, p(\mathbf{z} \mid \mathbf{x}, \theta^{(t)})$

2. M-step: $\theta^{(t+1)} = \underset{\theta}{\operatorname{argmax}} \sum_{\mathbf{z}} \log(p(\mathbf{x}, \mathbf{z} \mid \theta))\, p(\mathbf{z} \mid \mathbf{x}, \theta^{(t)})$

EM for Mixture of Gaussians (on board)

$$p\left(x_n \mid \boldsymbol{\mu}, \boldsymbol{\sigma}^2, \boldsymbol{\pi}\right) = \sum_m p\left(x_n, z_n = m \mid \mu_m, {\sigma_m}^2, \pi_m\right) = \sum_m \frac{1}{\sqrt{2\pi{\sigma_m}^2}} \exp\left(-\frac{\left(x_n - \mu_m\right)^2}{{\sigma_m}^2}\right) \cdot \pi_m$$

1. E-step: $\mathrm{E}_{z|x,\theta^{(t)}}\left[\log\left(p\left(\mathbf{x}, \mathbf{z} \mid \theta\right)\right)\right] = \sum_{\mathbf{z}} \log\left(p\left(\mathbf{x}, \mathbf{z} \mid \theta\right)\right) p\left(\mathbf{z} \mid \mathbf{x}, \theta^{(t)}\right)$

2. M-step: $\theta^{(t+1)} = \operatorname*{argmax}_{\theta} \sum_{\mathbf{z}} \log\left(p\left(\mathbf{x}, \mathbf{z} \mid \theta\right)\right) p\left(\mathbf{z} \mid \mathbf{x}, \theta^{(t)}\right)$

$$\alpha_{nm} = p(z_n = m \mid x_n, \boldsymbol{\mu}^{(t)}, {\boldsymbol{\sigma}^2}^{(t)}, \boldsymbol{\pi}^{(t)})$$

$$\hat{\mu}_m^{(t+1)} = \frac{1}{\sum_n \alpha_{nm}} \sum_n \alpha_{nm} x_n \qquad \hat{\sigma}_m^{2(t+1)} = \frac{1}{\sum_n \alpha_{nm}} \sum_n \alpha_{nm} \left(x_n - \hat{\mu}_m\right)^2 \qquad \hat{\pi}_m^{(t+1)} = \frac{\sum_n \alpha_{nm}}{N}$$

EM Algorithm

- Maximizes a lower bound on the data likelihood at each iteration

- Each step increases the data likelihood
  - Converges to *local maximum*

- Common tricks to derivation
  - Find terms that sum or integrate to 1
  - Lagrange multiplier to deal with constraints

EM Demos

- Mixture of Gaussian demo
- Simple segmentation demo

“Hard EM”

- Same as EM except compute z* as most likely values for hidden variables
- K-means is an example
- Advantages
  - Simpler: can be applied when cannot derive EM
  - Sometimes works better if you want to make hard predictions at the end
- But
  - Generally, pdf parameters are not as accurate as EM

Missing Data Problems: Outliers

You want to train an algorithm to predict whether a photograph is attractive. You collect annotations from Mechanical Turk. Some annotators try to give accurate ratings, but others answer randomly.

Challenge: Determine which people to trust and the average rating by accurate annotators.

Annotator Ratings

10
8
9
2
8

Photo: Jam343 (Flickr)

Missing Data Problems: Object Discovery

You have a collection of images and have extracted regions from them. Each is represented by a histogram of "visual words".

Challenge: Discover frequently occurring object categories, without pre-trained appearance models.

http://www.robots.ox.ac.uk/~vgg/publications/papers/russell06.pdf

3 EM - Mixture of Multinomials (15%)

Probabilistic mixture models are useful in a variety of applications, such as gaussian mixture models for segmentation (see problem 4). Multinomial distributions are another useful distribution for mixture models, and can be used to model the bag-of-words representation seen in the previous problem: for a given texture i, codeword j occurs with probability θ_{ij}.

A mixture of multinomials would allow modeling images that are composed of multiple textures, each defined by Θ_i, where texture i occurs with probability τ_i. More sophisticated methods such as pLSA and LDA replace τ with a per-image distribution over textures classes, which must be inferred. Again, note that this was originally introduced for representing documents composed of multiple "topics."

Derive the EM algorithm for the following multinomial mixture model for n examples $\{x_i\}$:

$$P(\mathbf{x}|\{\Theta_i\},\{\pi_i\}) = \sum_i \pi_i P(\mathbf{x}|\Theta_i), \text{ s.t. } \sum_i \pi_i = 1,\ 0 \le \pi_i \le 1$$

$$P(\mathbf{x}|\Theta_i) = \frac{n!}{\prod_j x_j!} \prod_j \theta_{ij}^{x_j}, \text{ s.t. } \sum_j \theta_{ij} = 1,\ 0 \le \theta_{ij} \le 1$$

Show the Expectation step (7 pts) and give the EM update formulae for π_i (3 pts) and Θ_i (5 pts). Show all steps including application of Bayes rule and computation of derivatives. Lagrange multipliers can be helpful for keeping π_i and Θ_i on the probability simplex.

n is number of elements in histogram

Next class

- MRFs and Graph-cut Segmentation