


Multimedia Information Systems



Samson Cheung

EE 639, Fall 2004

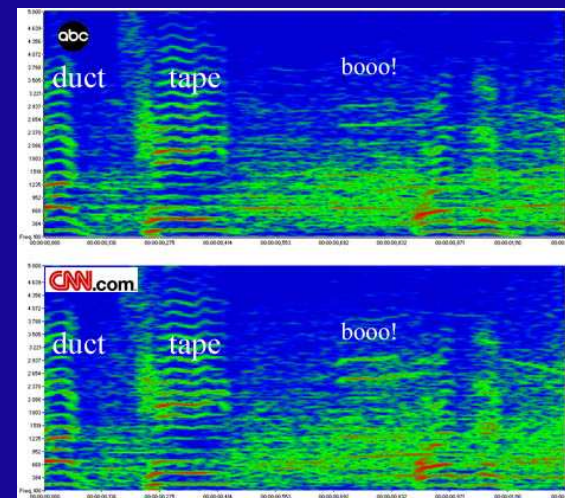
Lecture 15: Statistical Pattern Recognition

Why Pattern Recognition?

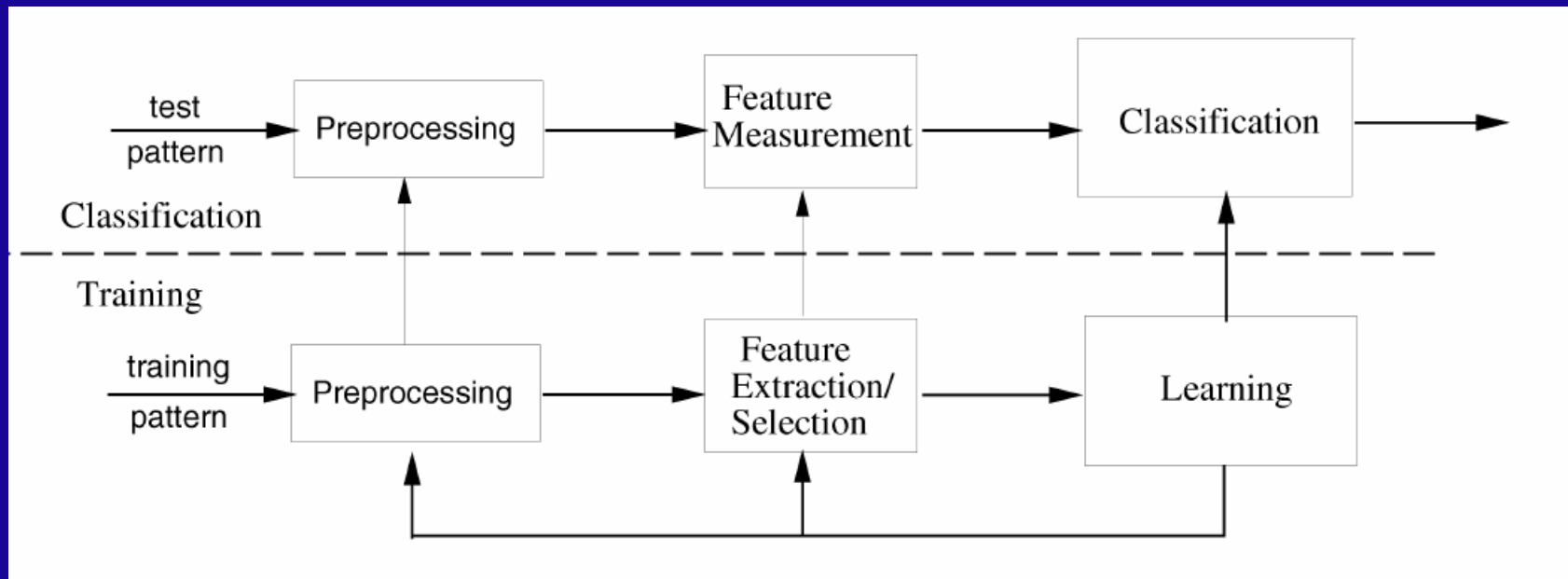
- Retrieve all news stories that show Kerry and Bush in the same shot
- Identify in the surveillance video all groups with more than three individuals
- Separate all the instruments in a symphony recording
- Find all the goal shots in a football match video
- Block out a certain individual in a surveillance tape for privacy reason
- ...

What is Pattern Recognition?

- **Two main tasks:**
 - **Supervised classification**
 - the input pattern is classified into a number of predefined classes
 - ex. Speech recognition
 - **Unsupervised classification**
 - the pattern is assigned to a hitherto unknown class
 - ex. Image segmentation



Model for PR



- Not every PR problem has the training part.
- Feature Extraction/Selection refers to the process of identifying the most important features for the PR tasks (later)
- Focus on classification and learning this week

Classification

Classification:

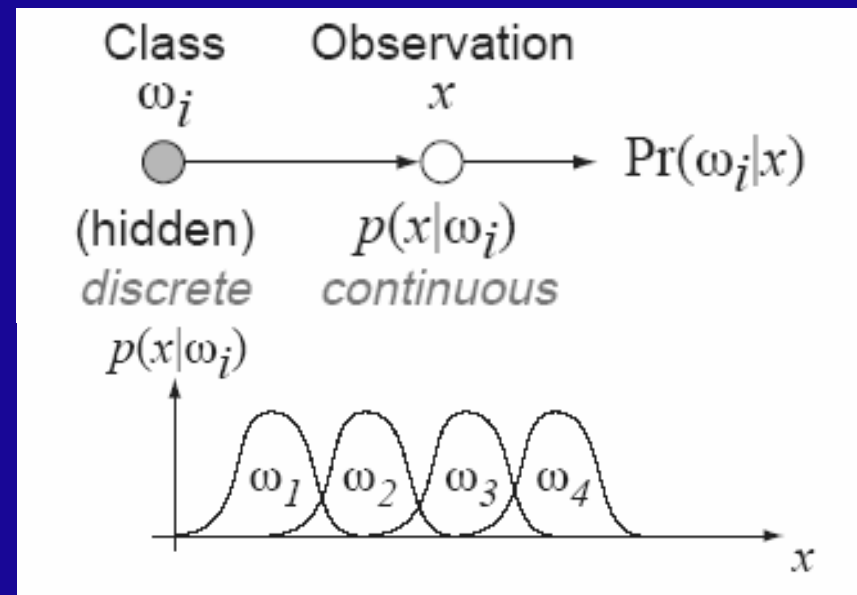
Given a measurement x , find corresponding class label y

Modeling:

Measurements are **RANDOM VARIABLES** whose **DISTRIBUTION** depends on the class

Class Conditional distributions:

- Reflect variability in feature
- Reflect noise



Random Variables review

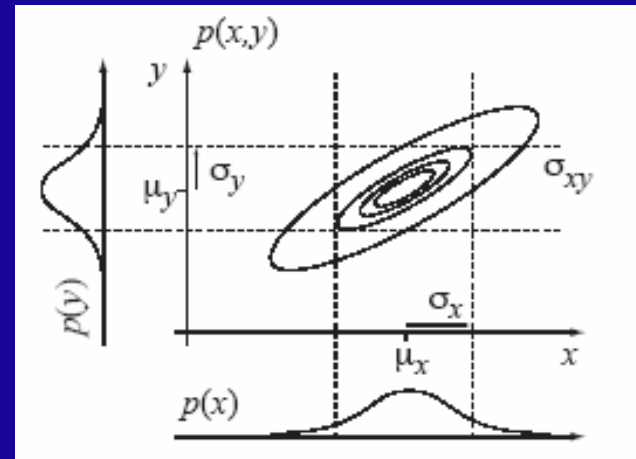
- **Joint PDF**

- **Marginal**

$$p(x) = \int p(x, y) dy$$

- **Covariance**

$$\Sigma = E[(x - \mu)(x - \mu)^T] = \begin{bmatrix} \sigma_x^2 & \sigma_{xy} \\ \sigma_{xy} & \sigma_y^2 \end{bmatrix}$$



Conditional Probability

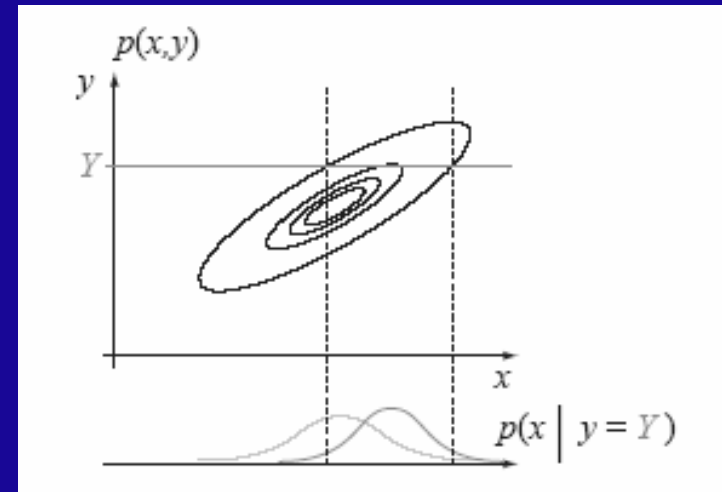
- Knowing one value in a joint distribution constrains the remainder

- Bayes' rule:

$$p(x, y) = p(x | y) \cdot p(y)$$

$$\Rightarrow p(y | x) = \frac{p(x | y) \cdot p(y)}{p(x)}$$

- Can reverse conditioning given priors/marginal
 - Either term can be discrete or continuous



Priors and Posterior

- Bayesian inference can be interpreted as updating *prior beliefs* with new measurements, x :

Bayes' Rule:

$$\underbrace{Pr(\omega_i)}_{\text{Prior probability}} \cdot \frac{\overbrace{p(x|\omega_i)}^{\text{Likelihood}}}{\underbrace{\sum_j p(x|\omega_j) \cdot Pr(\omega_j)}_{\text{'Evidence' = } p(x)}} = \underbrace{Pr(\omega_i|x)}_{\text{Posterior probability}}$$

- *Posterior* is *prior* scaled by *likelihood* & normalized by *evidence* (so $\Sigma(\text{posteriors}) = 1$)
- **Objection: priors are often unknown**
 - but omitting them amounts to assuming they are all equal

Classification criterion

1. Maximum Likelihood (ML):

$$y = \arg \max_{\omega} P(X=x | Y=\omega)$$

2. Maximum a Posterior (MAP):

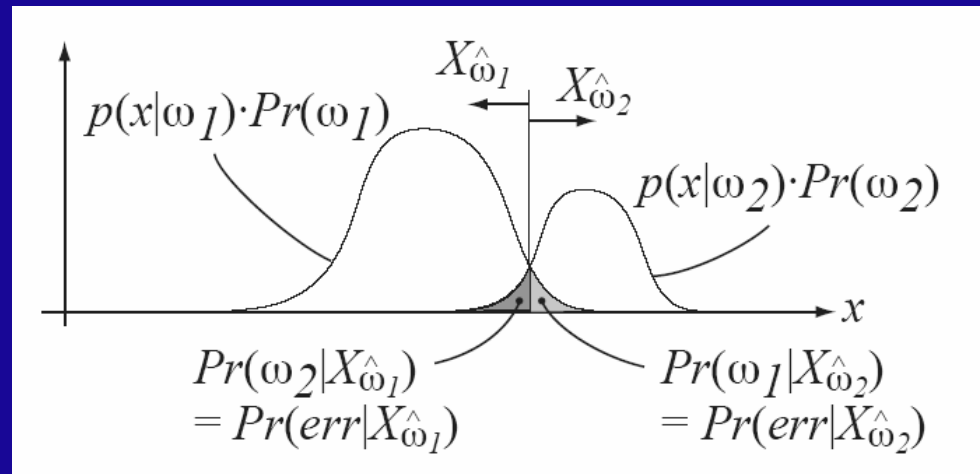
$$\begin{aligned} y &= \arg \max_{\omega} P(Y=\omega | X=x) \\ &= \arg \max_{\omega} P(X=x | Y=\omega) \cdot P(Y=\omega) \end{aligned}$$

3. Bayesian decision rule:

$$y = \arg \min_{\omega} E[L(\omega) | X=x]$$

where $L(\omega)$ measures the loss when misclassifying the class as ω

Sources of error



- **Suboptimal threshold / regions (bias error)**
 - use a Bayes classifier!
- **Incorrect distributions (model error)**
 - better distribution models/more training data
- **Misleading features ('Bayes error')**
 - *irreducible* for a given feature set regardless of classification scheme

Training

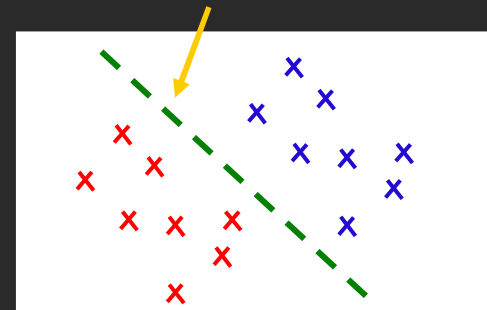
Given labeled training data: $\{(x_1', y_1'), (x_2', y_2'), \dots, (x_N', y_N')\}$,
derive the classification rule

Density-based approach

- **Estimate $P(X=x | Y=\omega)$ directly**
 - **Parametric form**
 - assume a specific PDF
 - ML estimate : find parameters to maximize $P(X,Y)$
 - **Non-parametric**
 - Histogram: how many bins?
- **Problematic when dimension is high**
 - **Need lots of data**
 - **Overfitting** : a histogram with too many bins is a poor estimate of the underlying pdf.

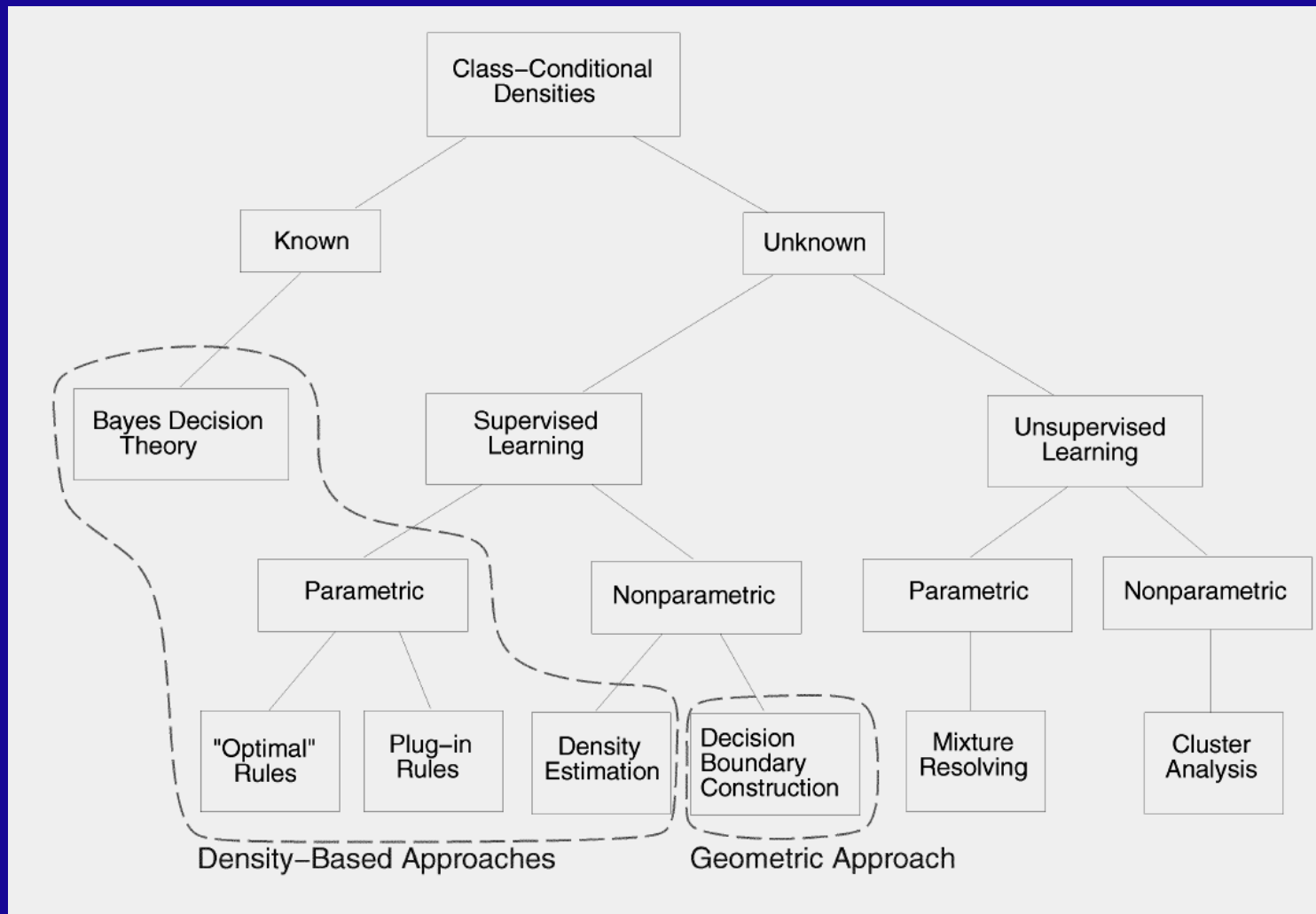
Boundary-based approach

- **Identify the class decision boundary of region directly**



- **Does not overfit easily**
- **Difficult to interpret**

Ontology for PR



Our focus

- **Parametric Models**
 - Gaussian Mixture Models
 - Hidden Markov Models
- **Non-parametric Models**
 - Neural Networks
 - Support Vector Machine
 - Boosting
- **Dimension Reduction**
 - Feature selection
 - PCA and Randomized projection
 - Applications to fast similarity search

Gaussian Distribution

- Easiest way to model distributions is via parametric model
 - assume known form, estimate a few parameters
- Gaussian model is simple & useful:

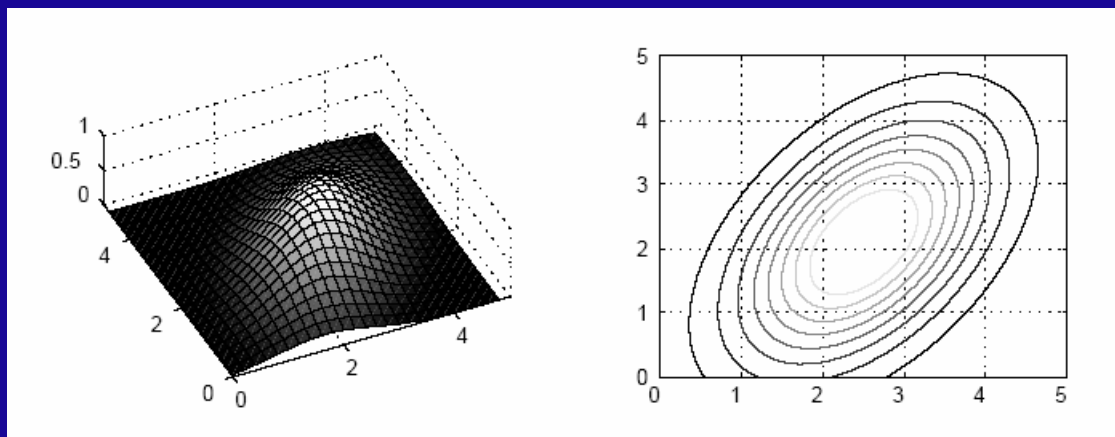
$$\text{in 1-D: } p(x|\omega_i) = \frac{1}{\sqrt{2\pi}\sigma_i} \cdot \exp\left[-\frac{1}{2}\left(\frac{x - \mu_i}{\sigma_i}\right)^2\right]$$

normalization to make it sum to 1

High-dimensional Gaussian

- Described by d dimensional mean vector $\boldsymbol{\mu}_i$ and $d \times d$ covariance matrix $\boldsymbol{\Sigma}_i$

$$p(\mathbf{x}|\omega_i) = \frac{1}{(\sqrt{2\pi})^d |\boldsymbol{\Sigma}_i|^{1/2}} \cdot \exp\left[-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_i)^T \boldsymbol{\Sigma}_i^{-1} (\mathbf{x} - \boldsymbol{\mu}_i)\right]$$



Gaussian Mixture Model (GMM)

- **Single Gaussians cannot model**

- distributions with multiple modes
- distributions with nonlinear correlation

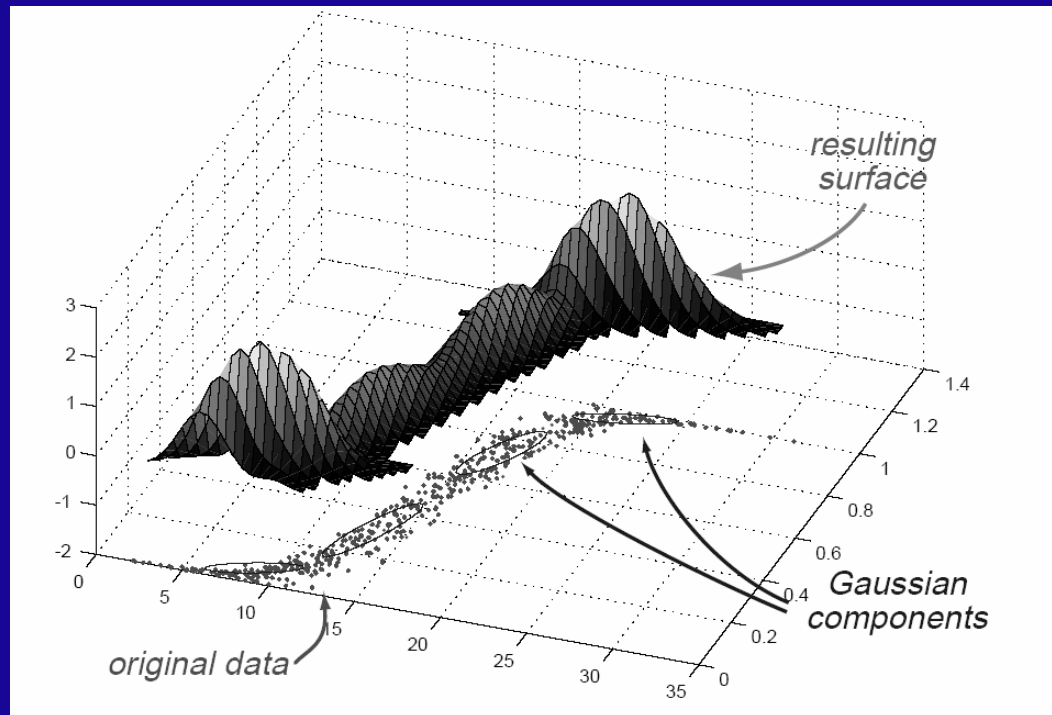
- **What about a weighted sum? i.e.**

$$p(x) \approx \sum_k c_k p(x|m_k)$$

- where $\{c_k\}$ is a set of weights and $\{p(x|m_k)\}$ is a set of Gaussian components
- can fit anything given enough components
- **Interpretation: Each observation is generated by one of the Gaussians, chosen at random, with priors:**

$$c_k = Pr(m_k)$$

Example



- **Problems: need to find the parameters for each component and the prior**

Two cases

- **Supervised classification**
- **Unsupervised clustering**
- **See notes**