Pattern Recognition

CSE 6510

Dept. of Computer Science & Engineering
Chittagong University of Engineering & Technology

Kaushik Deb, Ph.D.

About Myself

- Instructor name : Kaushik Deb
- E-mail : debkaushik99@gmail.com
- Phone : 031-714920-21 Ext:2306(office)
  019-1474-5508(only in emergency)

Administration

- Class time : Tuesday (14:30 – 17:00)
- Classroom : Multimedia Lab.

Prerequisites

- Courses on Digital image proces sing, Computer vision
- Probability and Statistics
- Strong programming and math ski lls

Required text

- Pattern classification by Duda, Hart, and Stork, 2nd edition, John Wiley Interscience, 2001
- **Optional text**
  - Statistical Pattern Recognition, by K. Fukunaga. 2nd edition, John Wiley Interscience, 2001.
  - *Information Theory, Inference, and Learning Algorithms*, by D. MacKay, Cambridge Univ Press, 2003.

Course Policies

- Grading will based on (tentative)
  - Midterm exam (25%)
  - Final exam (35%)
  - Paper presentation (20%)
  - Home work assignments or class performanc e (10%)
  - Attendance (10%)

Note: Late home work and paper presentation will not be accepted

Important Dates

- Paper presentation will be start from **after 3rd class** **(i.e. Jan. 03 , 2012)**
- Midterm will be held on **after 7th class**

Objectives

- **Recognize Patterns.** Make decisions about pa tterns
- Process the sensed data to eliminate noise
  - Visual example: is this person happy or sad?
  - Speech example: did the speaker say “Yes” or “No”?
- This course will introduce the ***fundamentals of statistical pattern recognition*** with exa mples from several application areas ***with emphasis on digital image processing and co mputer vision***
- **Techniques for handling multidimensional da ta *of various types and scales*** along with algorithms for clustering and classifying data will be explained

Cont.

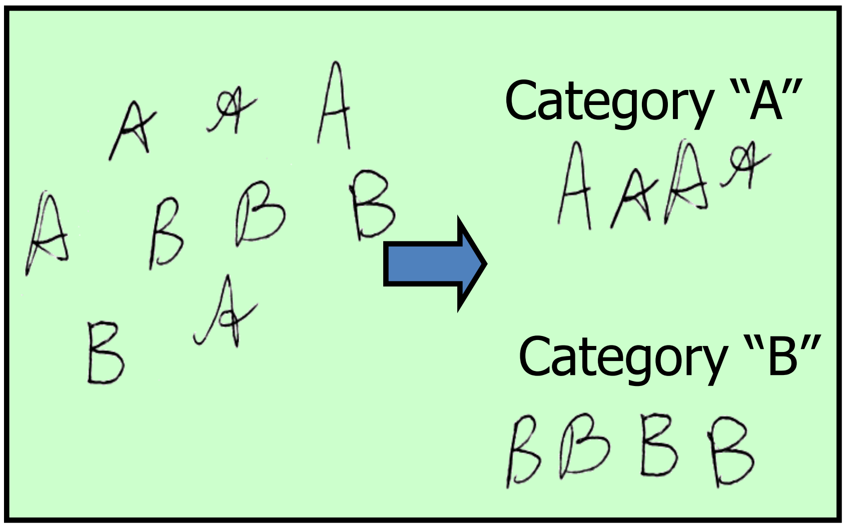

This is a horse
http://www.flickr.com/photos/kimbar/2027234083/


This is a horse
http://www.flickr.com/photos/genewolf/2031802050/

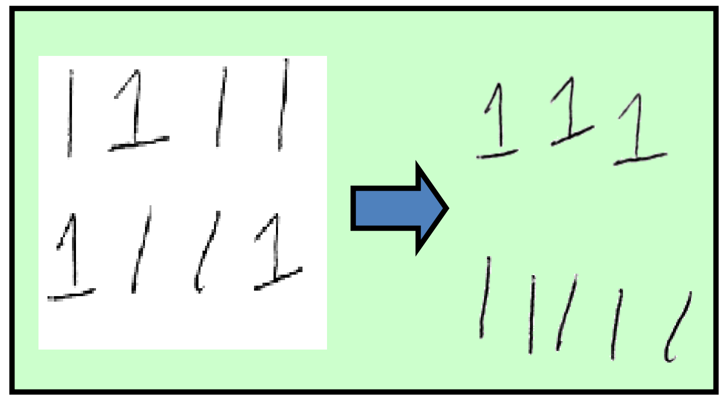

This is a... Horse?
http://www.flickr.com/photos/masheeebanshee/413465808/

Classification vs. Clustering

– Classification (known categories)
– Clustering (creation of new categories)



Classification (Recognition)
(Supervised Classification)



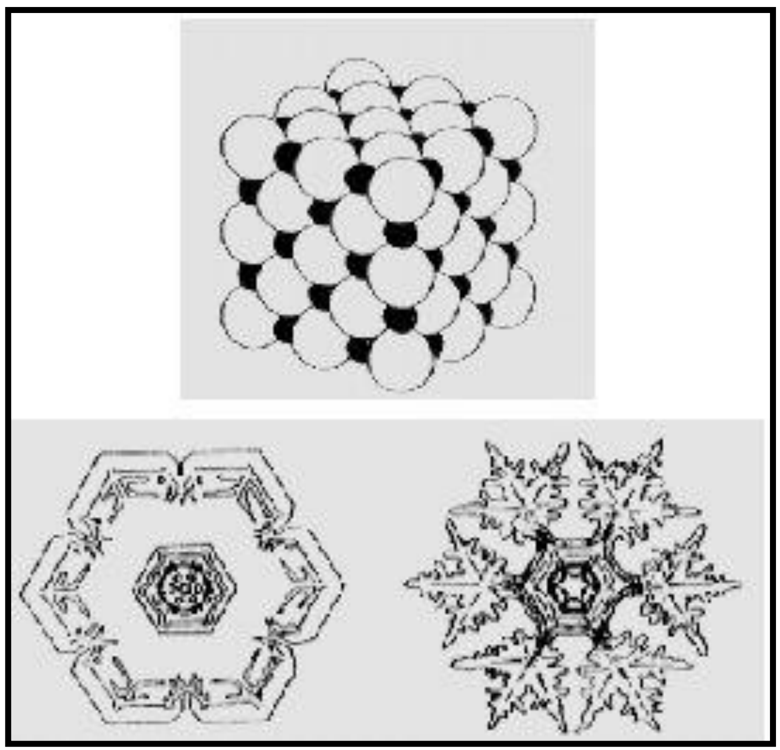
Clustering
(Unsupervised Classification)

What is Pattern Recognition (PR)?

- It is the study of how machines can
  - observe the environment
  - learn to distinguish patterns of interest from their background
  - make sound and reasonable decisions about the categories of the patterns.

What are patterns?

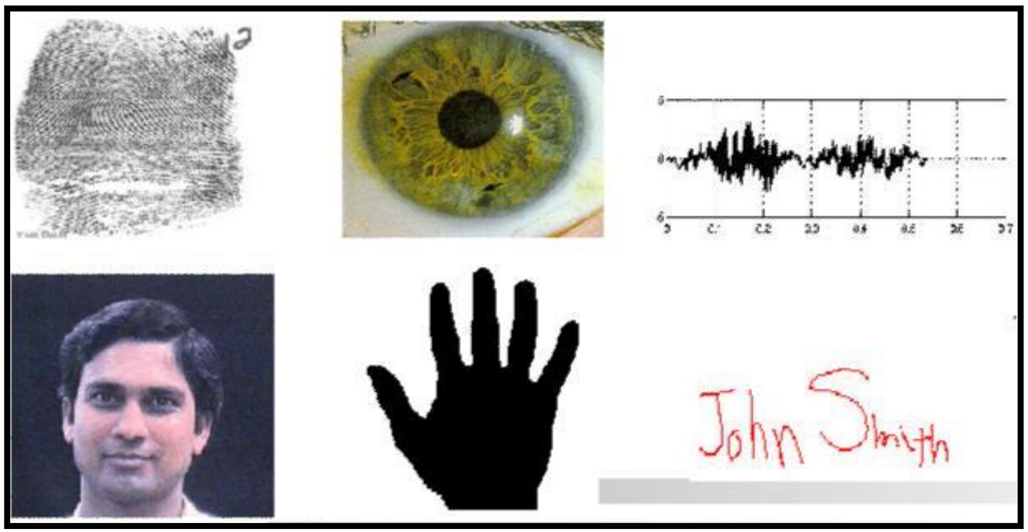
- Laws of Physics and Chemistry genera
  te patterns



Cont.

- **Watanable** defines a pattern as

  "*the opposite of a chaos; it is an entity, vaguely defined, that could be given a name*"



Variations of Patterns

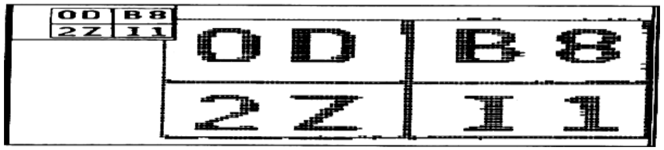
- Patterns are vary with expression, lighting, occlusions

Pattern Class

- A collection of “similar” (not necessarily identical) objects
  - **Inter-class** variability

The letter “T” in different typefaces

  - **Intra-class** variability



Characters that look similar

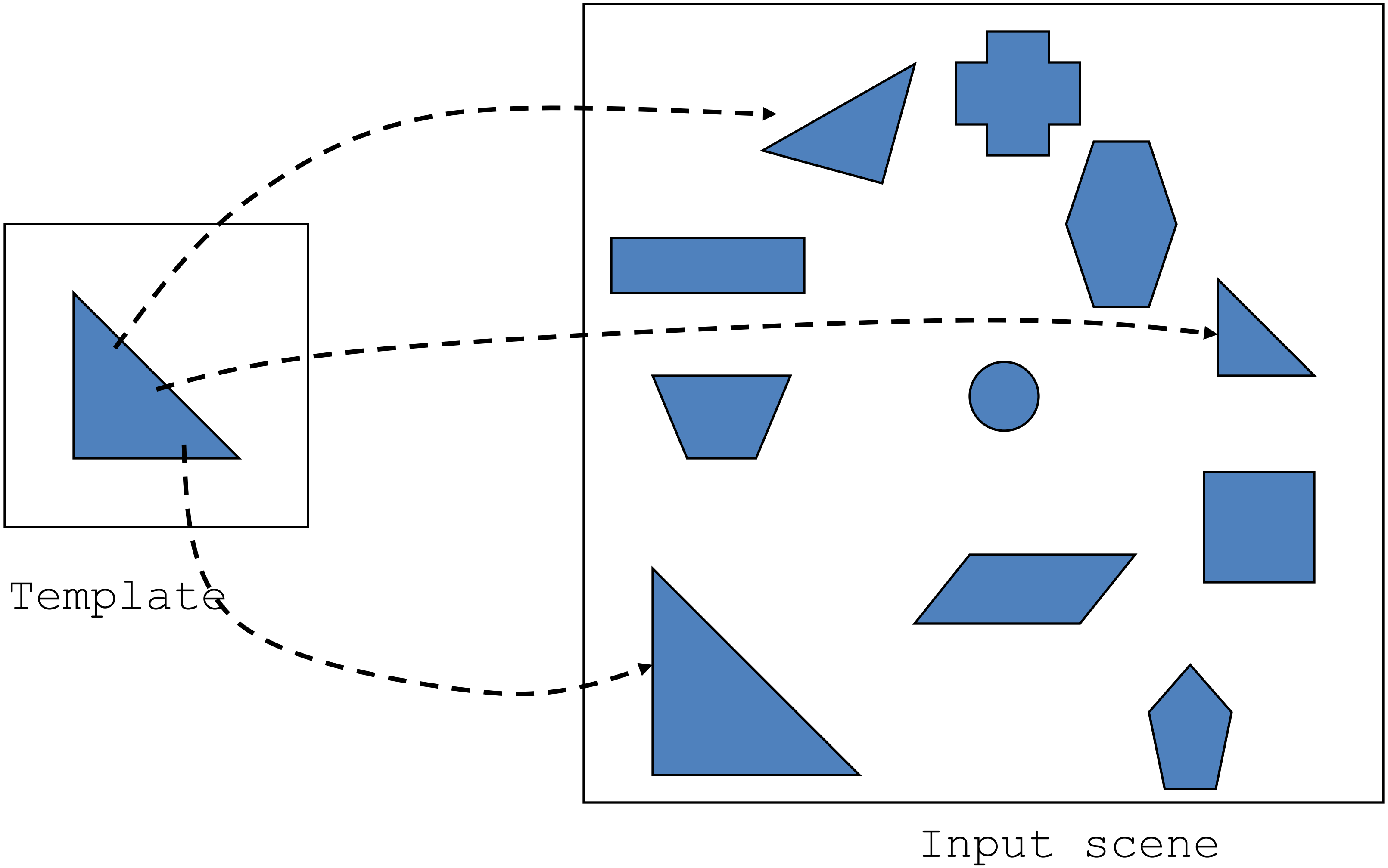
Applications

Problem	Input	Output
Speech recognition	Speech waveforms	Spoken words, speaker identity
Manufacturing	3-D images (structured light, laser, stereo)	Identify objects, pose, as sembly
Identification and counting of cells	Slides of blood samples, mic ro-sections of tissues	Type of cells
Detection and diagnosis of disease	ECG, EEG waveforms	Types of cardiac conditio ns, classes of brain condi tions
Aerial re-con-nais-sance	Visual, infrared, radar images	Tanks, airfields
Character recognition (page readers, zip code, license pl ate)	Optical scanned image	Alphanumeric characters

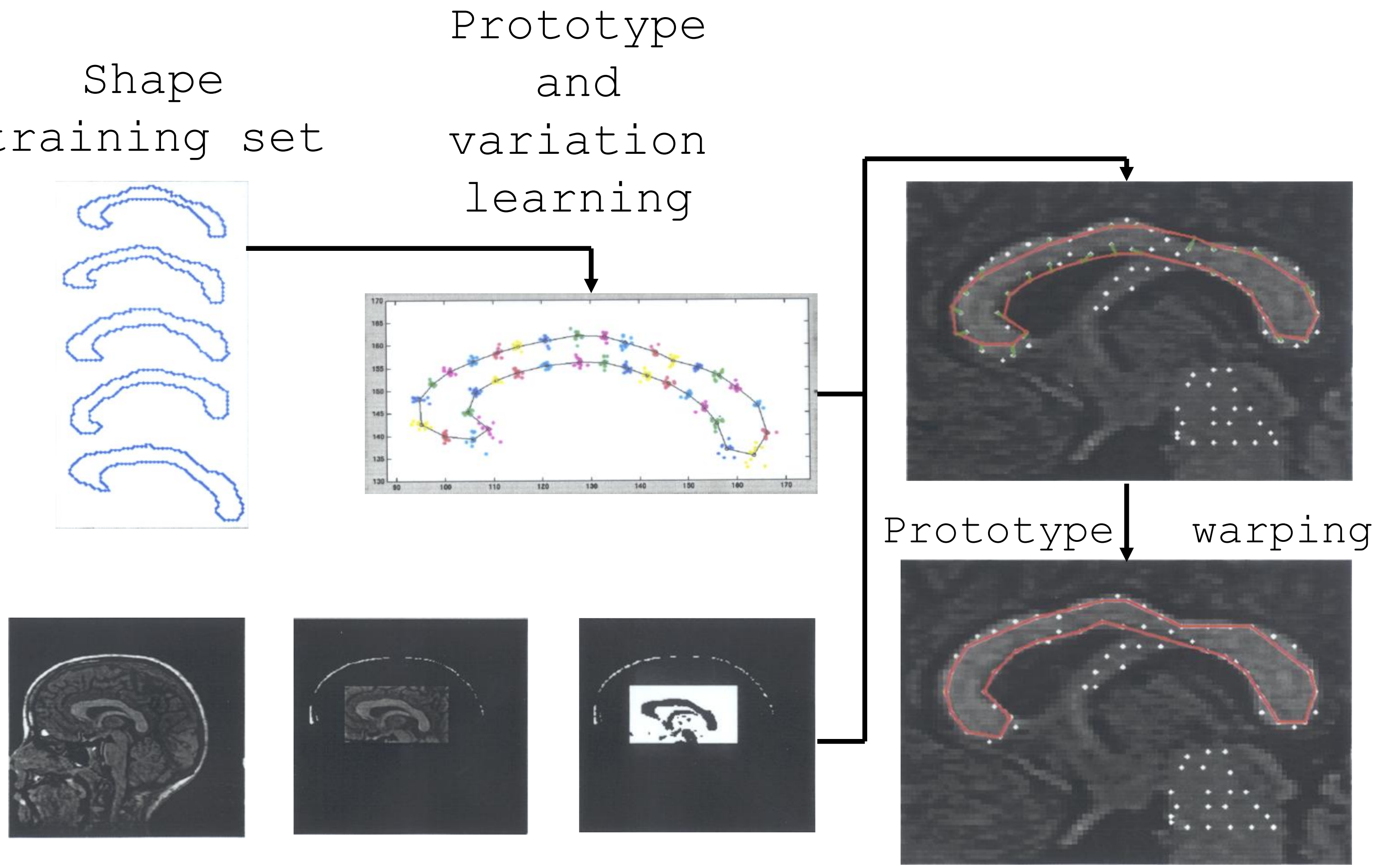
Main PR Areas

- *Template matching*
  - The pattern to be recognized is matched against a stored template while taking into account all allowable pose (translation and rotation) and scale changes
- *Statistical pattern recognition*
  - Focuses on the statistical properties of the patterns (i.e., probability densities)
- *Structural Pattern Recognition*
  - Describe complicated objects in terms of simple primitives and structural relationships
- *Syntactic pattern recognition*
  - Decisions consist of logical rules or grammars
- *Artificial Neural Networks*
  - Inspired by biological neural network models

Template Matching

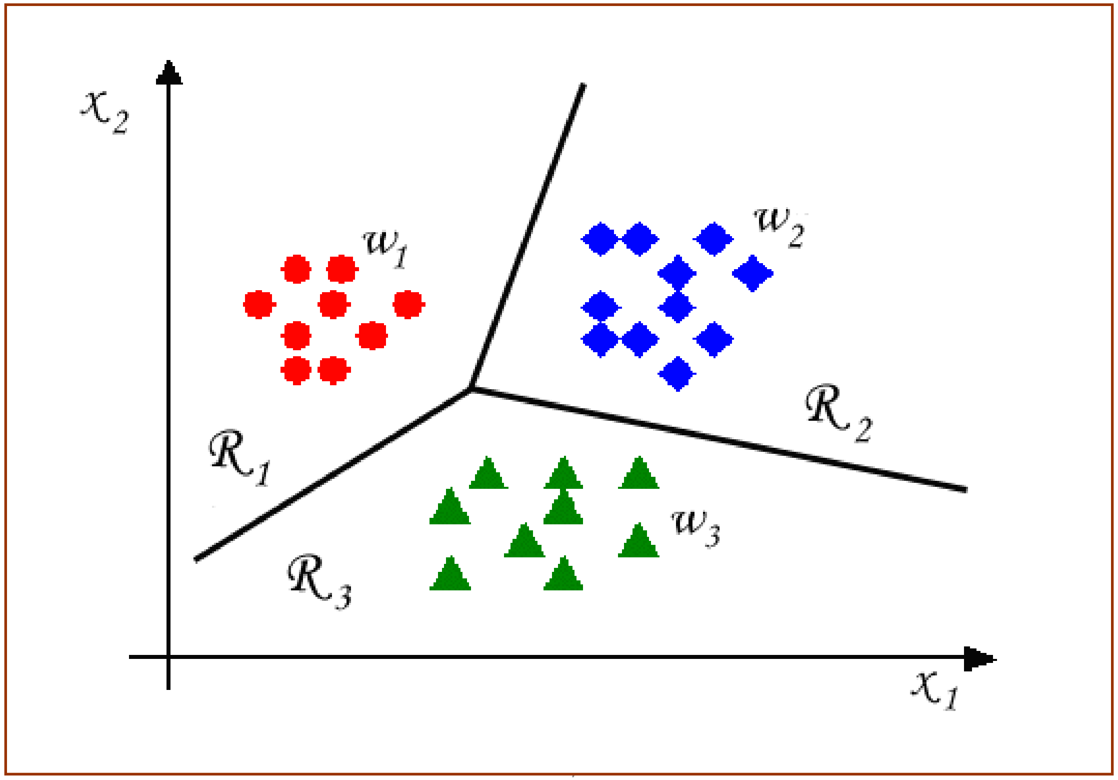


Deformable Template: Corpus Callosum Segmentation



Statistical Pattern Recognition (SPR)

- Patterns represented in a feature space
- Statistical model for pattern generation in feature space

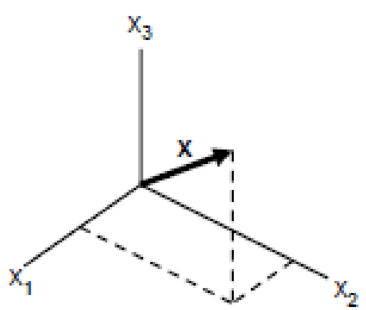


Cont.

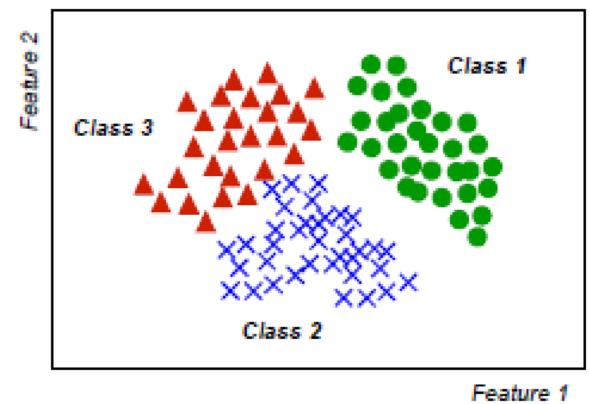
- **Feature**
  - Feature is any distinctive aspect, quality or characteristic
    - Features may be symbolic (i.e., color) or numeric (i.e., height)
  - Definitions
    - The combination of d features is represented as a d-dimensional column vector called a **feature vector**
    - The d-dimensional space defined by the feature vector is called the **feature space**
    - Objects are represented as points in feature space. This representation is called a **scatter plot**

$$X = \begin{bmatrix} x_1 \\ x_2 \\ \\ x_d \end{bmatrix}$$

Feature vector

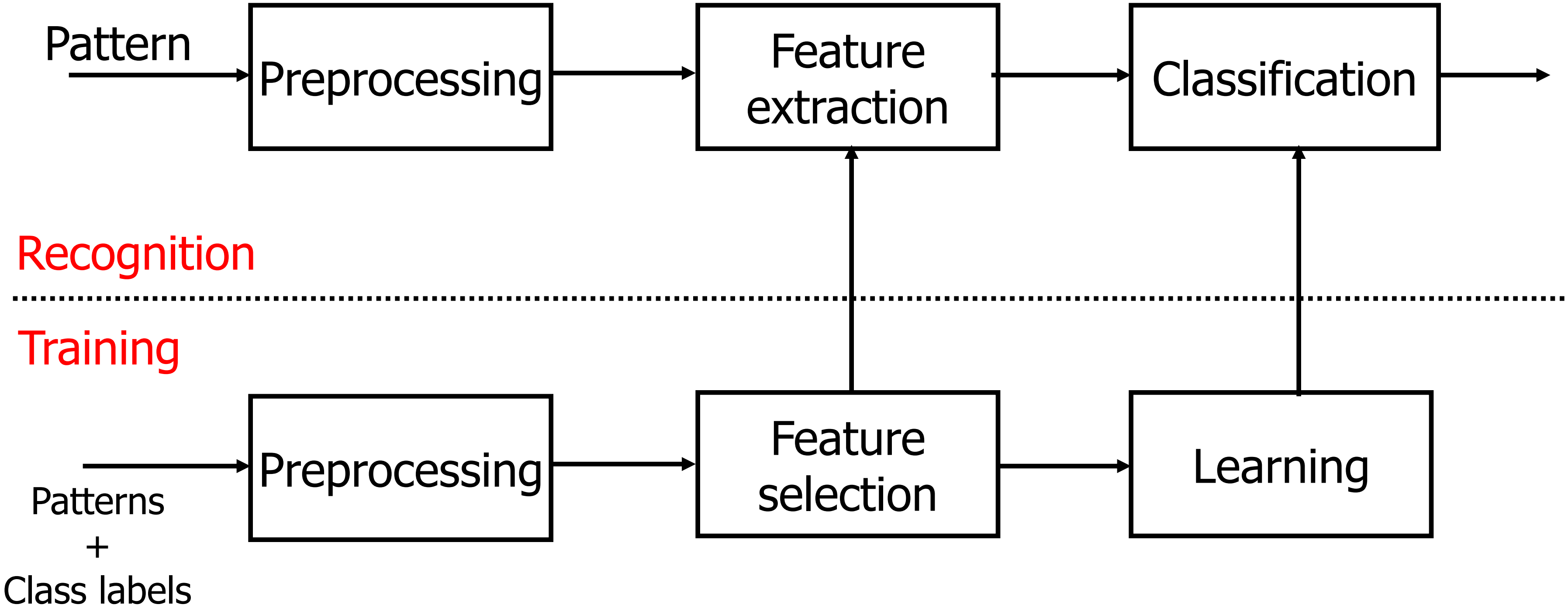


Feature space (3D)



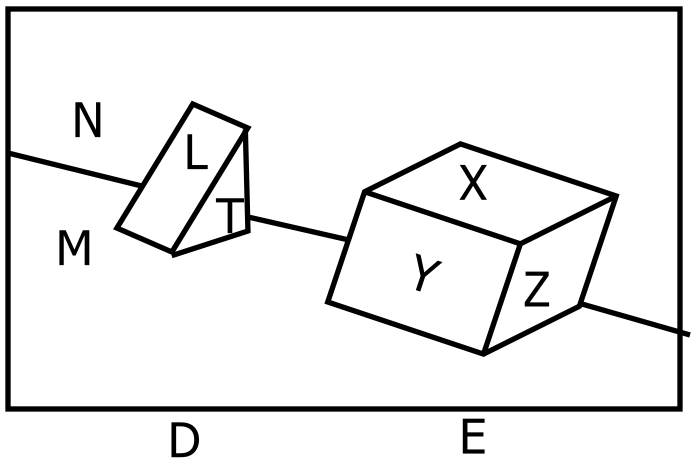
Scatter plot (2D)

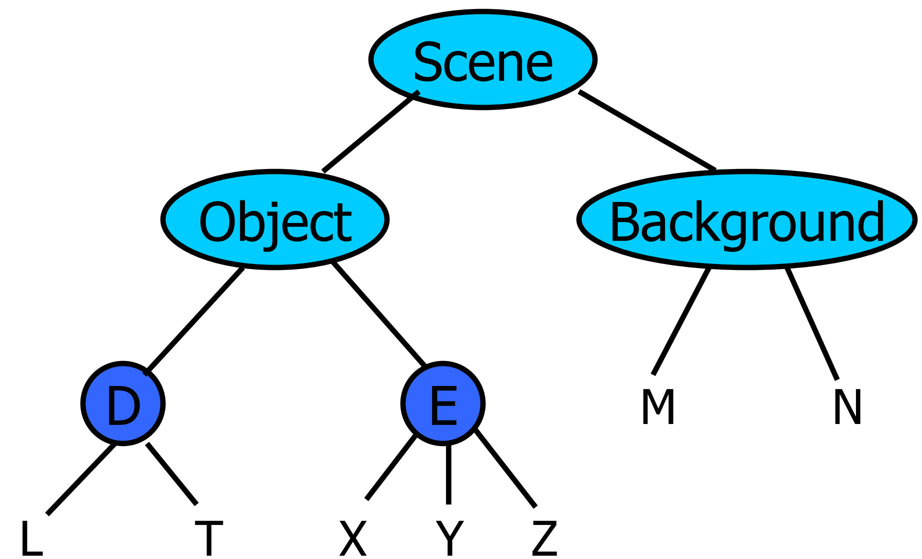
Statistical Pattern Recognition



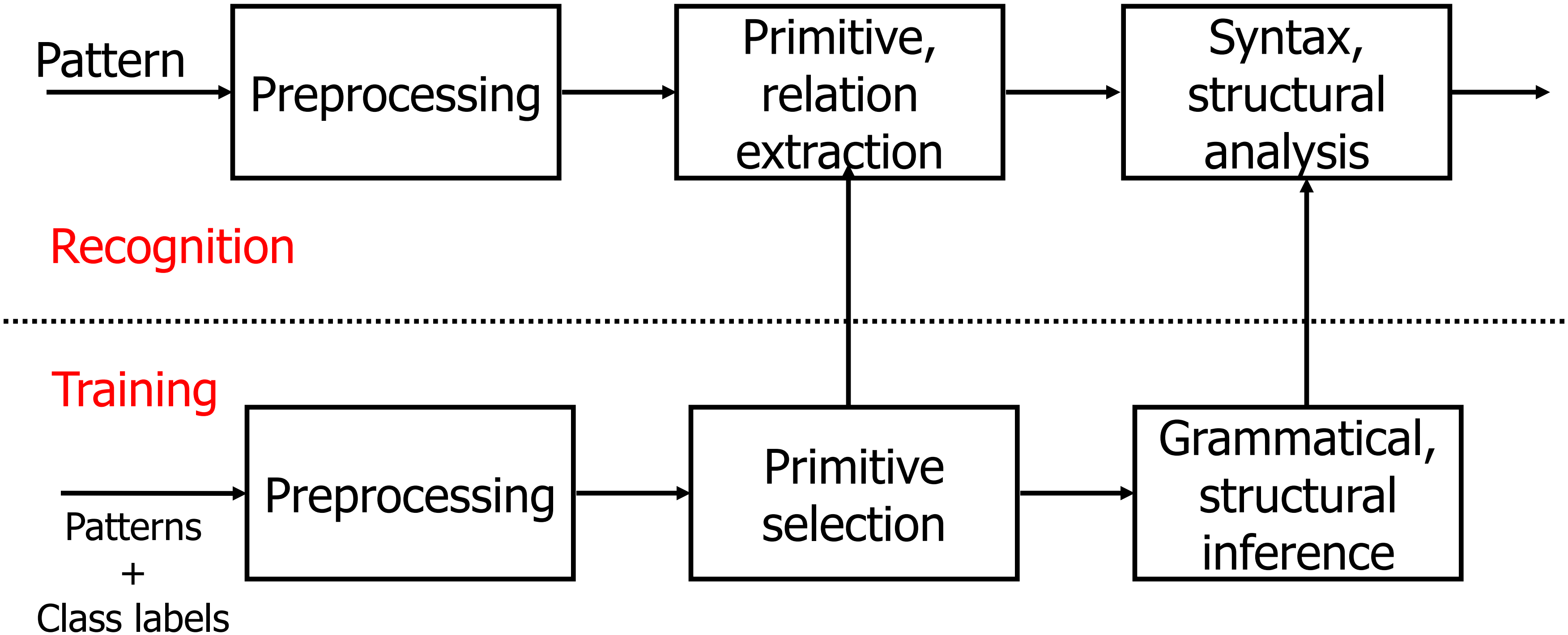
Structural Pattern Recognition

- Describe complicated objects in terms of simple primitives and structural relationships.
- Decision-making when features are non-numeric or structural





Syntactic Pattern Recognition



Describe patterns using deterministic grammars or formal languages

Chromosome Grammars

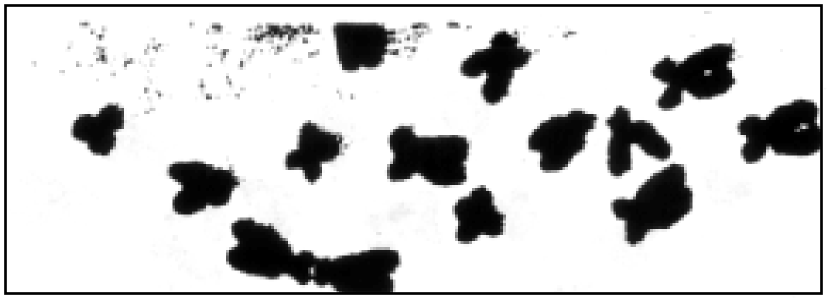
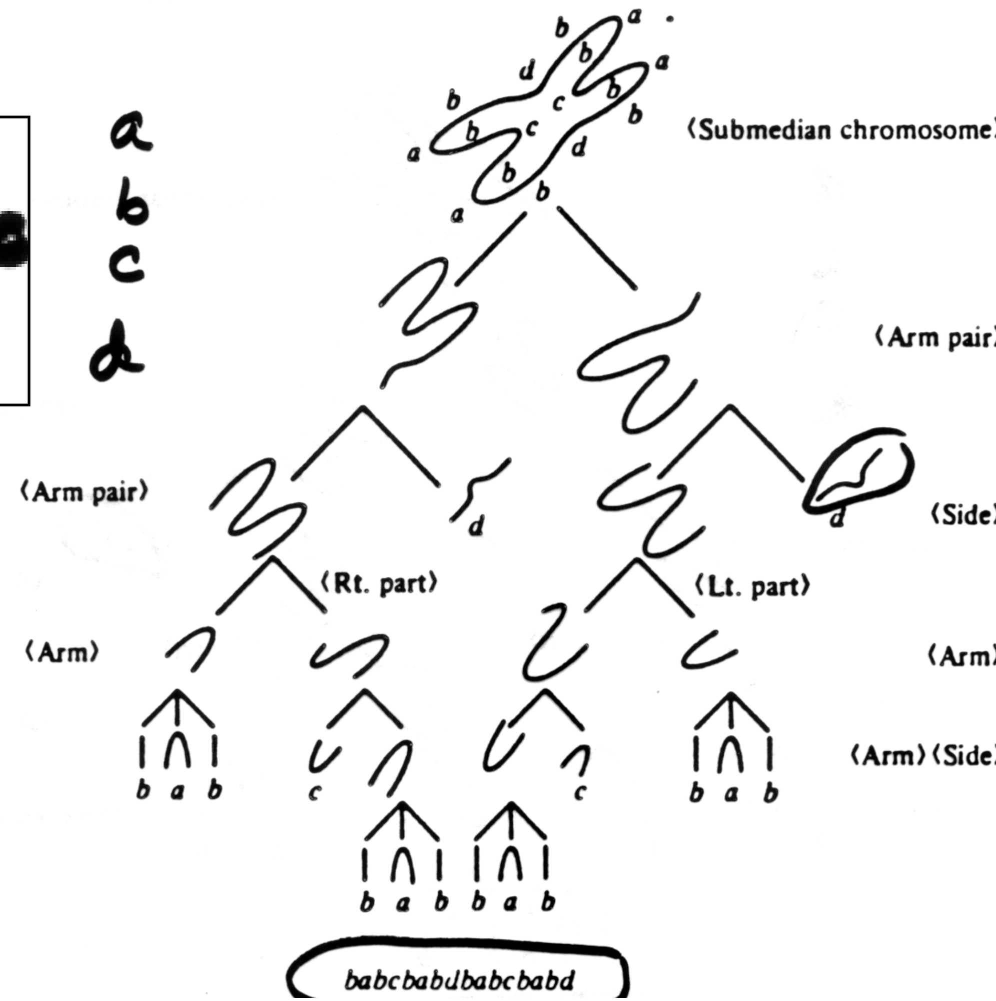


Image of human chromosomes

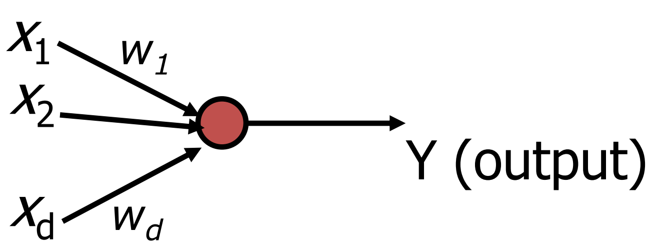
a
b
c
d



Hierarchical-structure description of a submedian chromosome

Artificial Neural Networks

- Massive parallelism is essential for complex pattern recognition tasks (e.g., speech and image recognition)
  - Human take only a few hundred ms for most cognitive tasks; suggests parallel computation
- Nodes in neural networks are nonlinear, typically analog



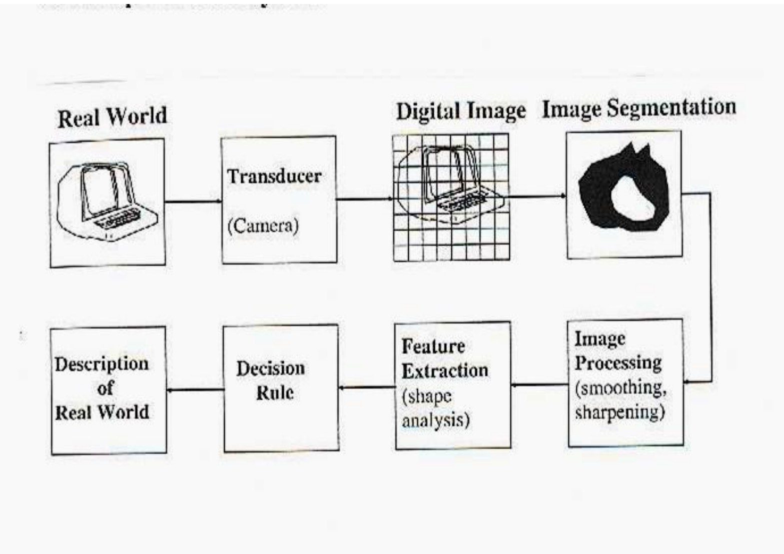
$$Y = f(\sum_{i=1}^{d} w_i x_i - \theta)$$

Where θ is an internal threshold

Comparing Pattern Recognition Models

- Template Matching
  - Assumes very small intra-class variability
  - Learning is difficult for deformable templa tes
- Structural / Syntactic
  - Primitive extraction is sensitive to noise
  - Describing a pattern in terms of primitives is difficult
- Statistical
  - Assumption of density model for each class
- Artificial Neural Network
  - Parameter tuning and local minima in learni ng

Main Components of a PR system



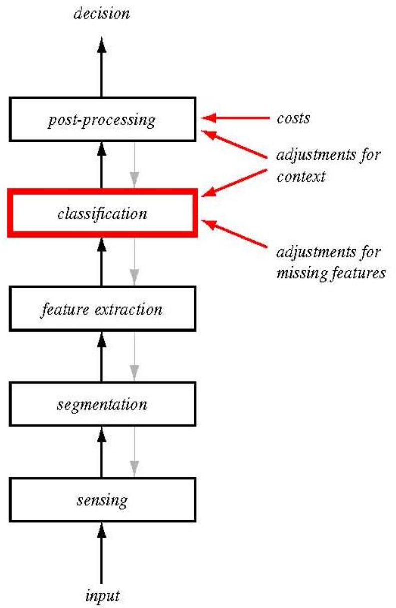


FIGURE 1.7. Many pattern recognition systems can be partitioned into components such as the ones shown here. A sensor converts images or sounds or other physical inputs into signal data. The segmentor isolates sensed objects from the background or from other objects. A feature extractor measures object properties that are useful for classification. The classifier uses these features to assign the sensed object to a category. Finally, a post processor can take account of other considerations, such as the effects of context and the costs of errors, to decide on the appropriate action. Although this description stresses a one-way or "bottom-up" flow of data, some systems employ feedback from higher levels back down to lower levels (gray arrows). From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

Complexity of PR – An Example

<u>Preprocessing involves:</u>

(1) Image enhancement

(2) Separating touching or occluding fish

(3) Finding the boundary of the fish

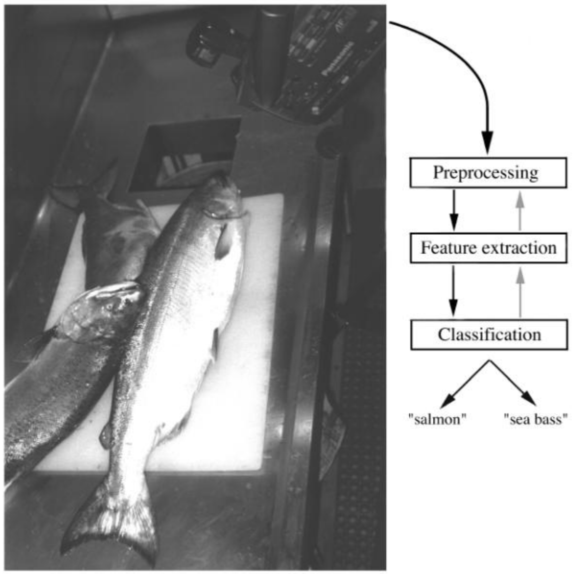


FIGURE 1.1. The objects to be classified are first sensed by a transducer (camera), whose signals are preprocessed. Next the features are extracted and finally the classification is emitted, here either "salmon" or "sea bass." Although the information flow is often chosen to be from the source to the classifier, some systems employ information flow in which earlier levels of processing can be altered based on the tentative or preliminary response in later levels (gray arrows). Yet others combine two or more stages into a unified step, such as simultaneous segmentation and feature extraction. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

How to separate sea bass from salmon?

- Possible features to be used:
  - Length
  - Lightness
  - Width
  - Number and shape of fins
  - Position of the mouth
  - Etc …

Decision Using Length

- Choose the optimal threshold using a number of *training examples*.

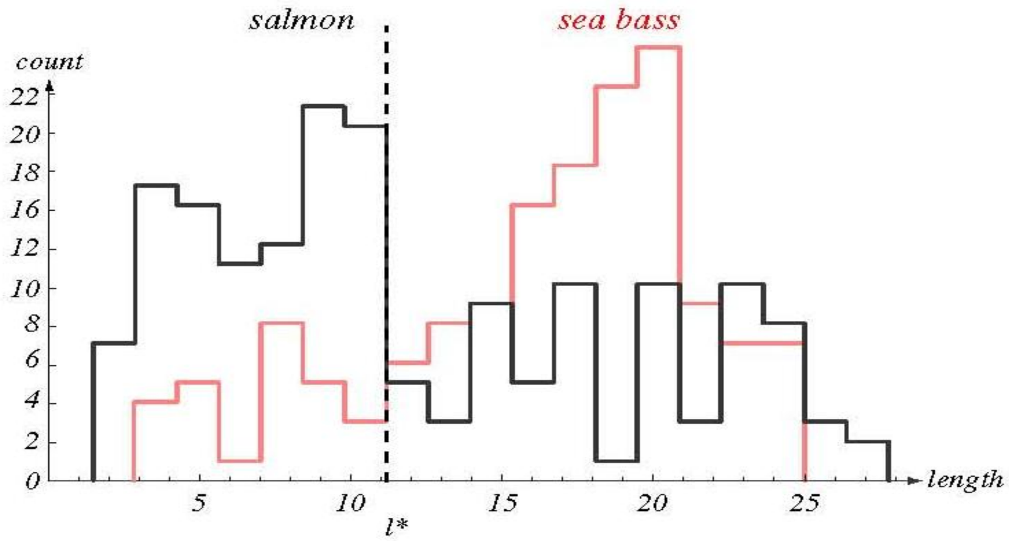


FIGURE 1.2. Histograms for the length feature for the two categories. No single threshold value of the length will serve to unambiguously discriminate between the two categories; using length alone, we will have some errors. The value marked l^* will lead to the smallest number of errors, on average. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

Decision Using Average Lightness

- Choose the optimal threshold using a number of *training examples*

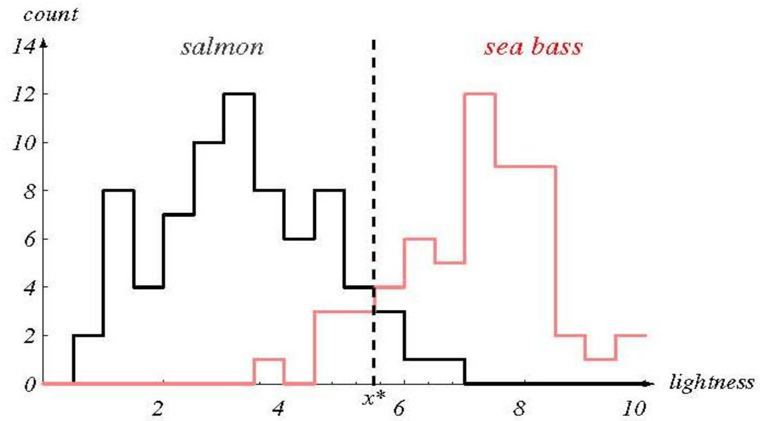


FIGURE 1.3. Histograms for the lightness feature for the two categories. No single threshold value x^* (decision boundary) will serve to unambiguously discriminate between the two categories; using lightness alone, we will have some errors. The value x^* marked will lead to the smallest number of errors, on average. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

Overlap in the histograms is small compared to length feature

Cost of Miss-classification

- There are two possible classification errors

  (1) deciding the fish was a sea bass when it was a salmon

  (2) deciding the fish was a salmon when it was a sea bass

- Which error is more important ?

Decision Using Multiple Features

- To improve recognition, we might have to use more than one feature at a time
  - Single features might not yield the best performance
  - Combinations of features might yield better performance

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad \begin{matrix} x_1 : lightness \\ x_2 : width \end{matrix}$$

Decision Boundary

- Partition the *feature space* into two regions by finding the *decision boundary* that minimizes the error

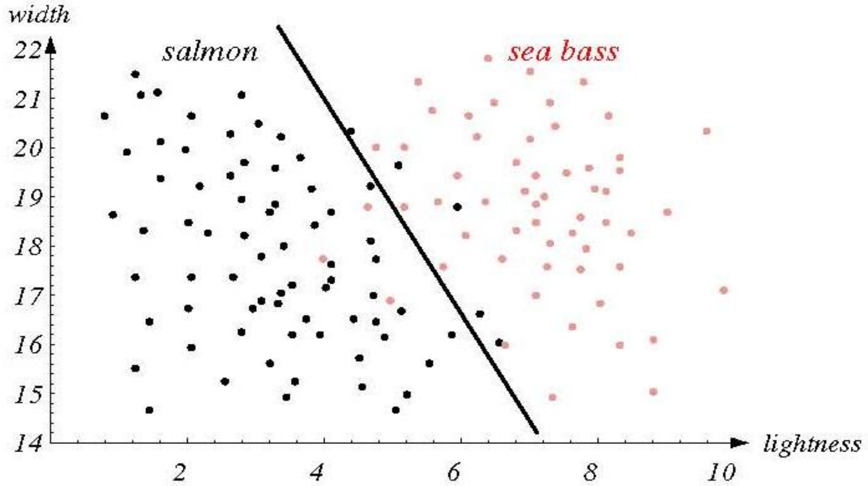


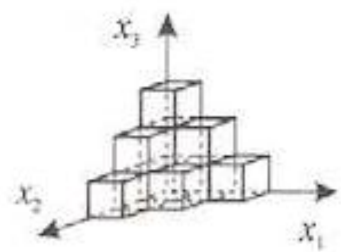
FIGURE 1.4. The two features of lightness and width for sea bass and salmon. The dark line could serve as a decision boundary of our classifier. Overall classification error on the data shown is lower than if we use only one feature as in Fig. 1.3, but there will still be some errors. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

How Many Features and Which?

- Issues with feature extraction:
  - Correlated features do not improve performance
  - It might be difficult to extract certain features
  - It might be computationally expensive to extract many features
  - “Curse” of dimensionality …

Curse of Dimensionality

- Adding too many features can, paradoxically, lead to a worsening of performance.
  - Divide each of the input features into a number of intervals, so that the value of a feature can be specified approximately by saying in which interval it lies.



  - If each input feature is divided into **M** divisions, then the total number of cells is $\mathbf{M}^{\mathbf{d}}$ (***d:*** # of features) which grows exponentially with ***d***.
  - Since each cell must contain at least one point, <u>the number of training data grows exponentially</u> !!

Model Complexity

- We can get perfect classification performance on the training data by choosing complex models.
- Complex models are *tuned* to the particular training samples, rather than on the characteristics of the true model.

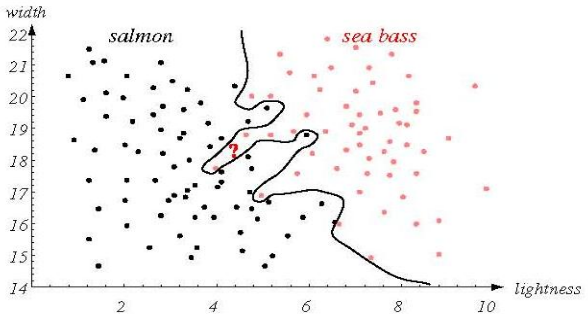


FIGURE 1.5. Overly complex models for the fish will lead to decision boundaries that are complicated. While such a decision may lead to perfect classification of our training samples, it would lead to poor performance on future patterns. The novel test point marked ? is evidently most likely a salmon, whereas the complex decision boundary shown leads it to be classified as a sea bass. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

Issue of generalization

Generalization

- The ability of the classifier to produce correct results on *novel* patterns.
- How can we improve generalization performance ?
  - More training examples (i.e., better pdf estimates)
  - Simpler models (i.e., simpler classification boundaries) usually yield better performance

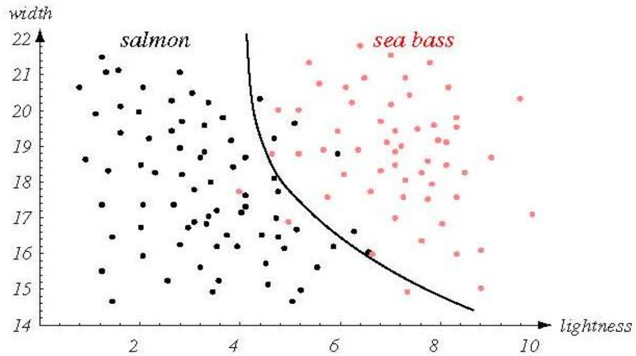


FIGURE 1.6. The decision boundary shown might represent the optimal tradeoff between performance on the training set and simplicity of classifier, thereby giving the highest accuracy on new patterns. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

Simplify the decision boundary!

Key Questions in PR

- How should we quantify and favor simpler classifiers ?
- Can we *predict* how well the system will generalize to novel patterns ?

The Design Cycle

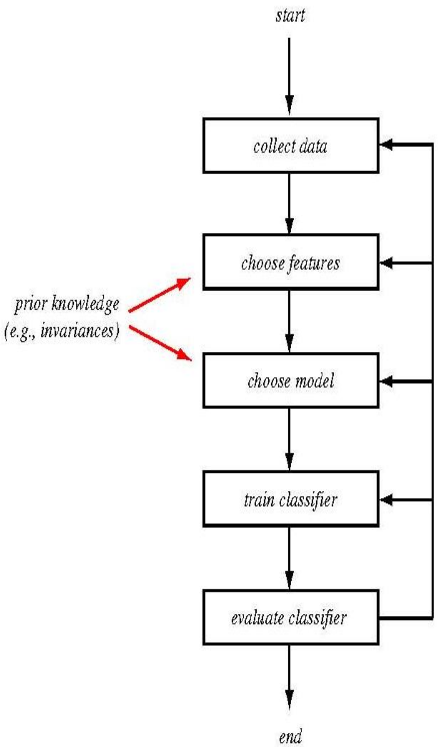


FIGURE 1.8. The design of a pattern recognition system involves a design cycle similar to the one shown here. Data must be collected, both to train and to test the system. The characteristics of the data impact both the choice of appropriate discriminating features and the choice of models for the different categories. The training process uses some or all of the data to determine the system parameters. The results of evaluation may call for repetition of various steps in this process in order to obtain satisfactory results. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

Overview of Important Issues

- Noise / Segmentation
- Data Collection / Feature Extraction
- Pattern Representation / Invariance/Missing Features
- Model Selection / Over fitting
- Prior Knowledge / Context
- Classifier Combination
- Costs and Risks
- Computational Complexity

Issue: Noise

- Various types of noise (e.g., shadows, conveyor belt might shake, etc.)
- Noise can reduce the reliability of the feature values measured.
- Knowledge of the noise process can help improve performance.

Issue: Segmentation

- Individual patterns have to be segmented
  - How can we segment without having categorized them first ?
  - How can we categorize them without having segmented them first ?
- How do we "group" together the proper number of elements ?

Issue: Data Collection

- How do we know that we have collected an adequately large and representative set of examples for training/testing the system?

Issue: Feature Extraction

- It is a domain-specific problem which infl uences classifier's performance.
- Which features are most promising ?
- Are there ways to automatically learn whic h features are best ?
- How many should we use ?
- Choose features that are robust to noise.
- Favor features that lead to simpler decision regions.

Issue: Pattern Representation

- Similar patterns should have similar representations.
- Patterns from different classes should have dissimilar representations.
- Pattern representations should be invariant to transformations such as:
  - translations, rotations, size, reflections, non-rigid deformations
- Small intra-class variation, large inter-class variation.

Issue: Missing Features

- Certain features might be missing (e.g., due to occlusion).
- How should the classifier make the best decision with missing features ?
- How should we train the classifier with missing features ?

Issue: Model Selection

- How do we know when to reject a class of models and try another one ?
- Is the model selection process just a trial and error process ?
- Can we automate this process ?

Issue: Over fitting

- Models complex than necessary lead to *over fitting* (i.e., good performance on the training data but poor performance on novel data).
- How can we adjust the complexity of the model ? (not very complex or simple).
- Are there principled methods for finding the best complexity ?

Issue: Domain Knowledge

- When there is not sufficient training data, incorporate domain knowledge:
  - Model how each pattern in generated (*analysis by synthesis*) - this is difficult !! (e.g., recognize all types of chairs).
  - Incorporate *some* knowledge about the pattern generation method. (e.g., optical character recognition (OCR) assuming characters are sequences of strokes)

Issue: Classifier Combination

- Performance can be improved using a "pool" of classifiers.
- How should we combine multiple classifiers ?

Issue: Costs and Risks

- Each classification is associated with a cost or risk (e.g., classification error).
- How can we incorporate knowledge about such risks ?
- Can we estimate the *lowest* possible risk of *any* classifier ?

Issue: Computational Complexity

- How does an algorithm *scale* with
  - the number of feature dimensions
  - number of patterns
  - number of categories
- Brute-force approaches might lead to perfect classifications results but usually have impractical time and memory requirements.
- What is the tradeoff between computational ease and performance ?

General Purpose PR Systems?

- Humans have the ability to switch rapidly and seamlessly between different pattern recognition tasks
- It is very difficult to design a device that is capable of performing a variety of classification tasks
  - Different decision tasks may require different features.
  - Different features might yield different solutions.
  - Different tradeoffs (e.g., classification error) exist for different tasks.