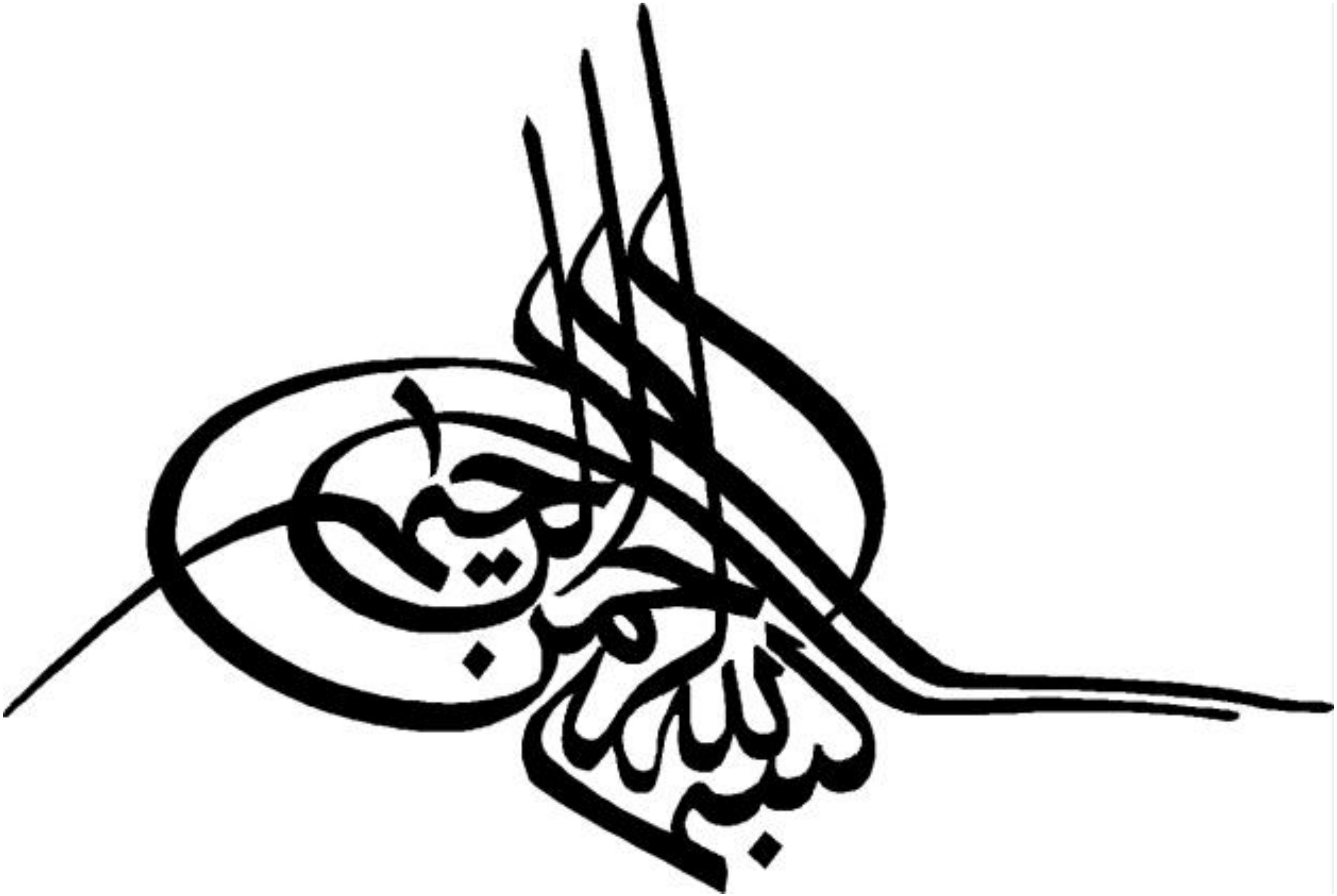

بسم الله الرحمن الرحیم

Statistical Pattern Recognition

- Introduction
- Bayesian decision theory
- Maximum likelihood and Bayesian parameter estimation
- Nonparametric techniques
- Linear Discriminant Functions
- Stochastic Methods
- Algorithm-independent machine learning
- Unsupervised Learning and Clustering

Textbooks

- **Pattern Classification (2nd ed.) by Richard O. Duda, Peter E. Hart and David G. Stork**
- **Pattern Recognition, 4th Ed., Theodoridis and Koutroumbas**
- Statistical Pattern Recognition, 3rd Ed. Andrew R.Webb And Keith D. Copsey
- Pattern Recognition and Machine Learning, Bishop
- Introduction to Statistical Pattern Recognition, 2nd Ed., Fukunaga
- A Statistical Approach to Neural Networks for Pattern Recognition, R. A. Dunne.
- Pattern Recognition and Image Analysis, Gose and Johansonbaugh

Grading Criteria

- ❖ Midterm Exam ≈ 25%
- ❖ HW, Comp. Assignments and projects: ≈ 30%
- ❖ Final exam ≈ 45%

- ❖ **Course Website:**
- ❖ http://ivut.iut.ac.ir ثبت نام در سامانه الزامی است
- ❖ Ebooks…

Chapter 1: Introduction to Pattern Recognition

- Machine Perception
- An example
- Pattern Recognition Systems
- The Design Cycle
- Learning and Adaptation
- Conclusion

All materials used in this course were taken from the textbook *"Pattern Classification"* by Duda et al., John Wiley & Sons, 2001 (Djamel Bouchaffra)

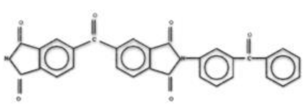
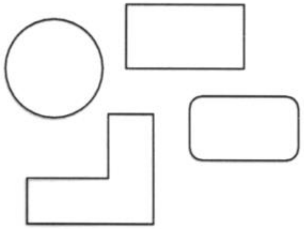
Pattern Recognition

“The real power of human thinking is based on recognizing patterns. The better computers get at pattern recognition, the more humanlike they will become”.

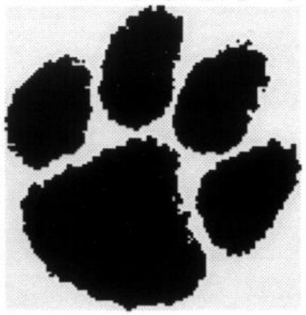
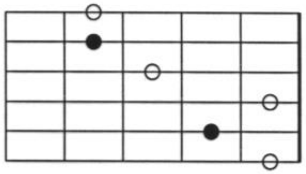
Ray Kurzweil, NY Times, Nov 24, 2003

What is a Pattern?

- "A pattern is the opposite of a chaos; it is an entity vaguely defined, that could be given a name."

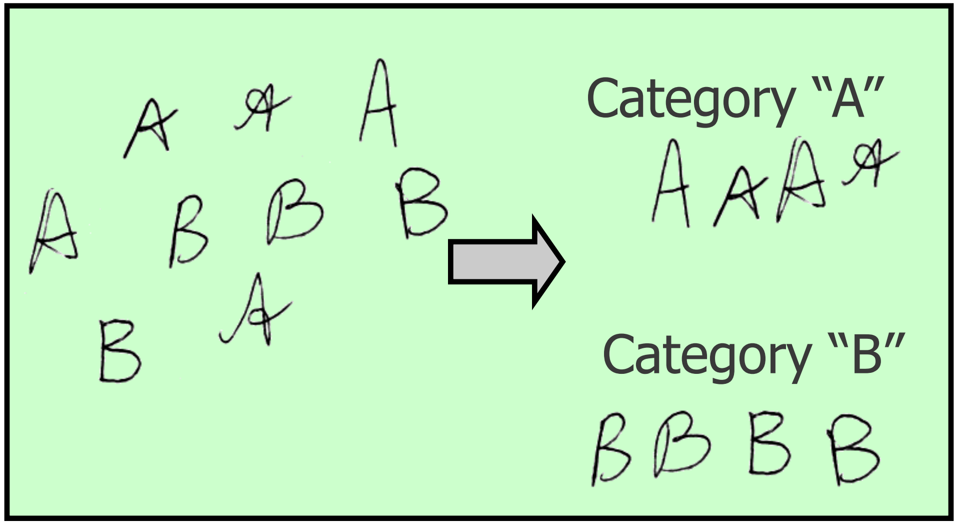





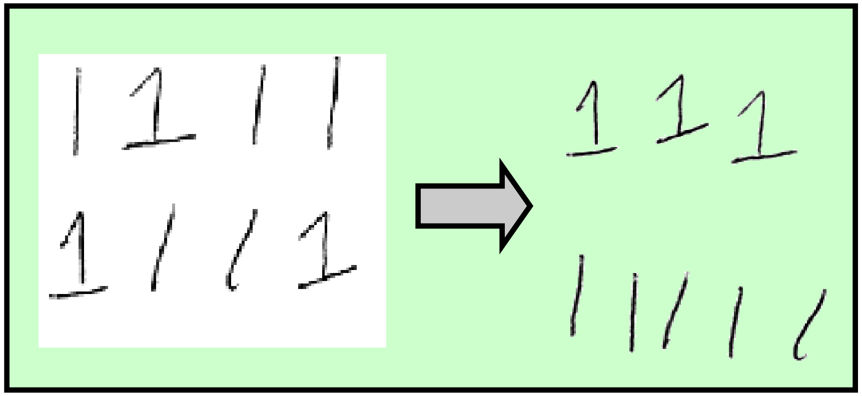




Recognition

- Identification of a pattern as a member of a category we already know, or we are familiar with
  - Classification (known categories)
  - Clustering (creation of new categories)



Classification



Clustering

Pattern Recognition

- Given an input pattern, make a decision about the "category" or "class" of the pattern

- Pattern recognition is a very broad subject with many applications

- In this course we will study a variety of techniques to solve P.R. problems and discuss their relative strengths and weaknesses

Pattern Class

- A collection of “similar” (not necessarily identical) objects
- A class is defined by class samples (paradigms, exemplars, prototypes)
- Inter-class variability
- Intra-class variability

Pattern Class Model

- ❖ Different descriptions, which are typically mathematical in form for each class/population

- ❖ Given a pattern, choose the best-fitting model for it and then assign it to class associated with the model

Intra-class and Inter-class Variability

The letter "T" in different typefaces

Same face under different expression, pose….

Machine Perception

- Build a machine that can recognize patterns:
  - Speech recognition
  - Fingerprint identification
  - OCR (Optical Character Recognition)
  - DNA sequence identification

Pattern Recognition Applications

Problem	Input	Output
Speech recognition	Speech waveforms	Spoken words, speaker identity
Non-destructive testing	Ultrasound, eddy current, acoustic emission waveforms	Presence/absence of flaw, type of flaw
Detection and diagnosis of disease	EKG, EEG waveforms	Types of cardiac conditions, classes of brain conditions
Natural resource identification	Multispectral images	Terrain forms, vegetation cover
Aerial reconnaissance	Visual, infrared, radar images	Tanks, airfields
Character recognition (page readers, zip code, license plate)	Optical scanned image	Alphanumeric characters

Pattern Recognition Applications

Problem	Input	Output
Identification and counting of cells	Slides of blood samples, micro-sections of tissues	Type of cells
Inspection (PC boards, IC masks, textiles)	Scanned image (visible, infrared)	Acceptable/unacceptable
Manufacturing	3-D images (structured light, laser, stereo)	Identify objects, pose, assembly
Web search	Key words specified by a user	Text relevant to the user
Fingerprint identification	Input image from fingerprint sensors	Owner of the fingerprint, fingerprint classes
Online handwriting retrieval	Query word written by a user	Occurrence of the word in the database

An Example

"Sorting incoming Fish on a conveyor according to species using optical sensing"

Species

- Sea bass گرگ دریایی- ماهی خاردار
- Salmon ماهی آزاد

❖ Problem Analysis

- ❖ Set up a camera and take some sample images to extract features
  - ❖ Length
  - ❖ Lightness
  - ❖ Width
  - ❖ Number and shape of fins
  - ❖ Position of the mouth, etc…

This is the set of all suggested features to explore for use in our classifier!

- Preprocessing

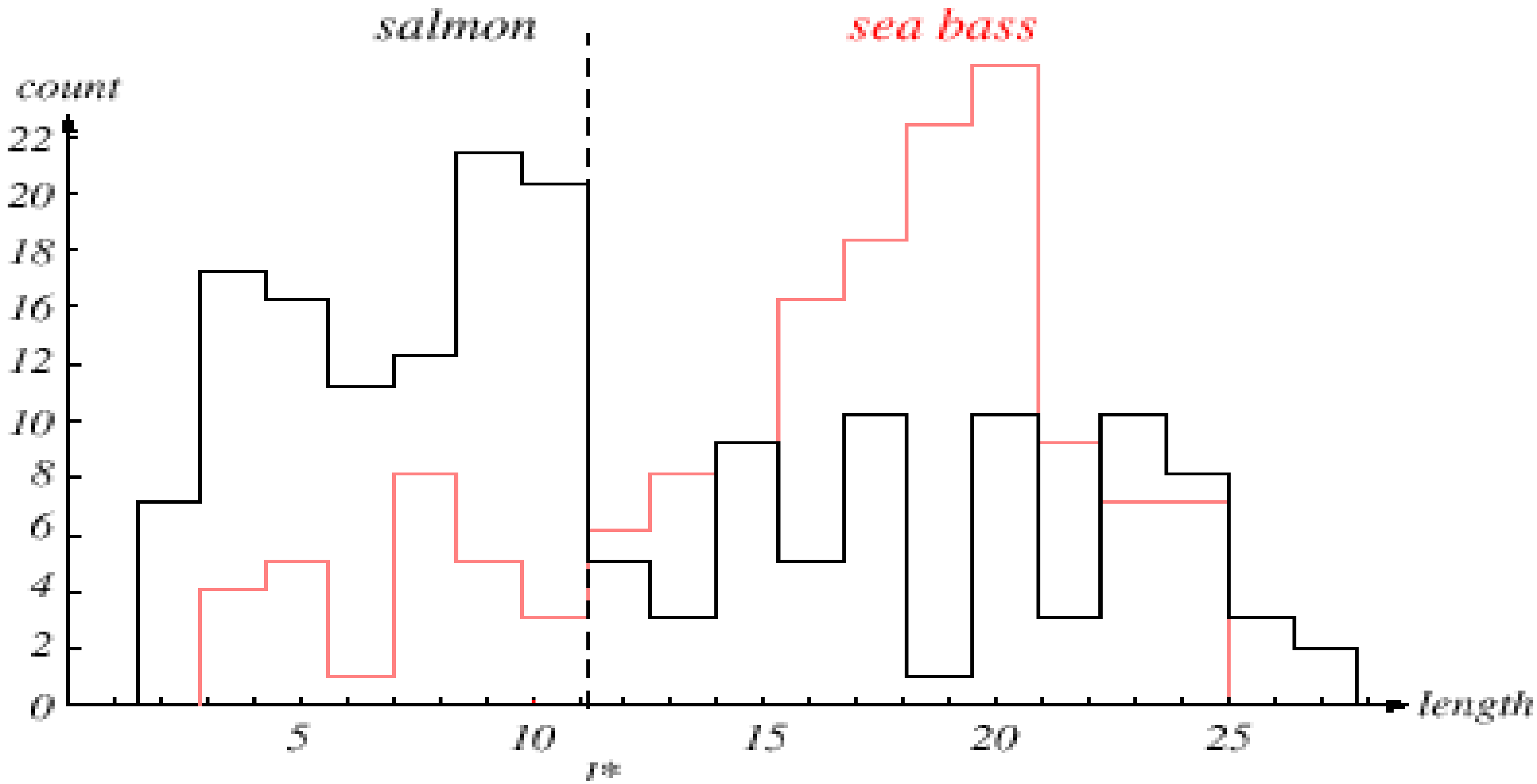
  - Use a segmentation operation to isolate fishes from one another and from the background

- Information from a single fish is sent to a feature extractor whose purpose is to reduce the data by measuring certain features

- The features are passed to a classifier

Figure 1.1: The objects to be classified are first sensed by a transducer (camera), whose signals are preprocessed, then the features extracted and finally the classification emitted (here either “salmon” or “sea bass”). Although the information flow is often chosen to be from the source to the classifier (“bottom-up”), some systems employ “top-down” flow as well, 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.

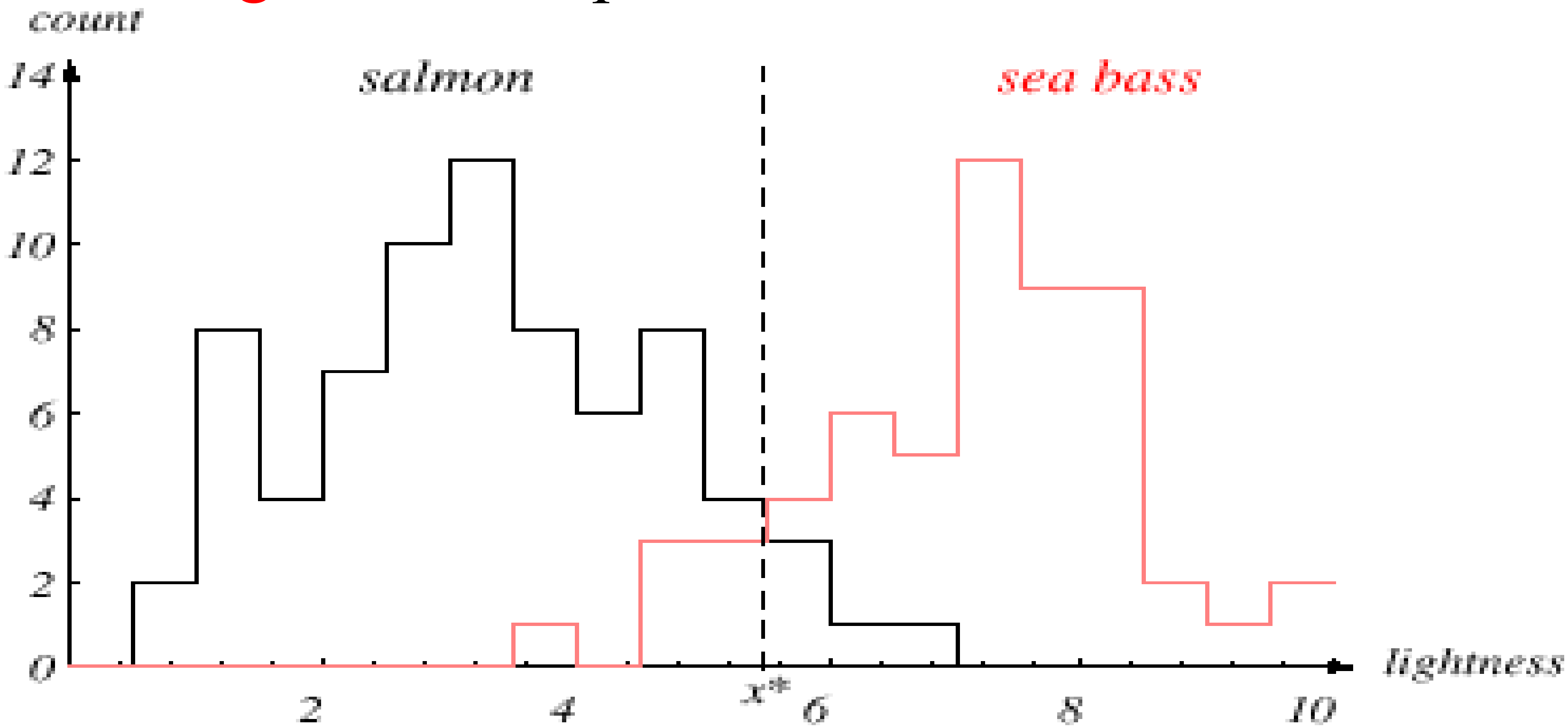
Classification: Select the length of the fish as a possible feature for discrimination



Histograms for the length feature for the two categories. No single threshold value l^* (decision boundary) will serve to unambiguously discriminate between the two categories; using length alone, we will have some errors. The value l^* marked will lead to the smallest number of errors, on average.

The length is a poor feature alone!

Select the lightness as a possible feature.



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.

- Threshold decision boundary and cost relationship

  - Move our decision boundary toward smaller values of lightness in order to minimize the cost (reduce the number of sea bass that are classified salmon!)

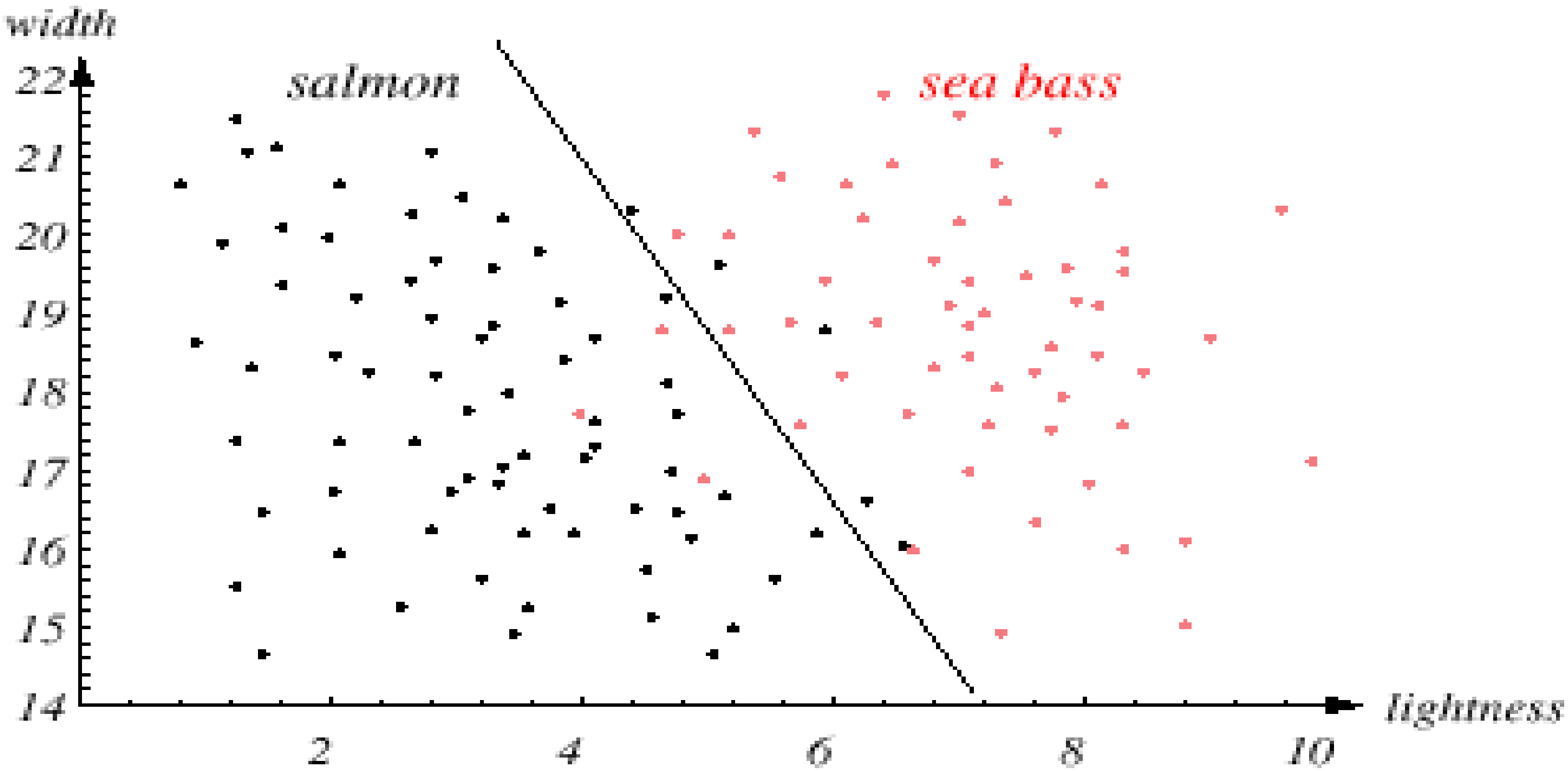
Task of decision theory

- Adopt the lightness and add the width of the fish

Fish $\Longrightarrow$ $\mathbf{x} = [x_1, x_2]$

Lightness **Width**

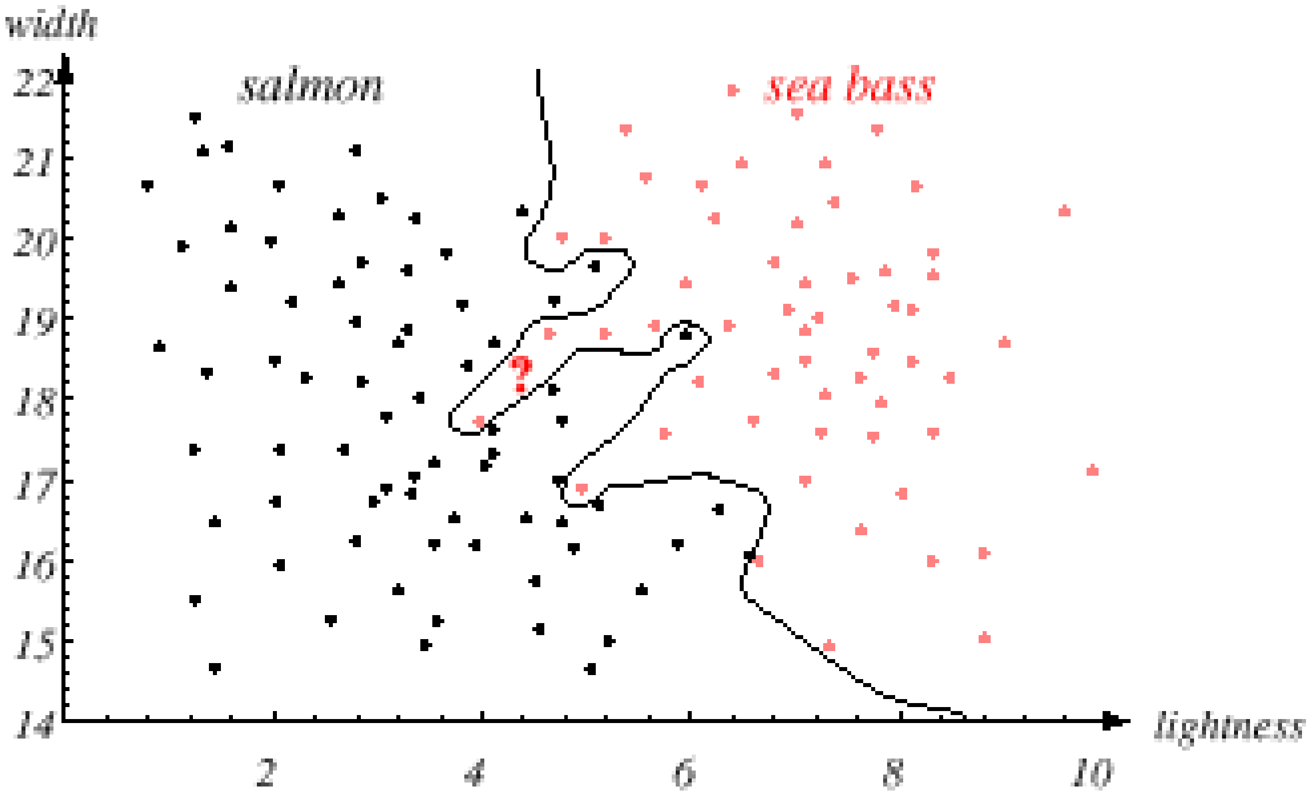
We realize that the feature extractor has thus reduced the image of each fish to a point or *feature vector* $\mathbf{x}$ in a two-dimensional *feature space*.



The two features of lightness and width for sea bass and salmon. The dark line might 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.

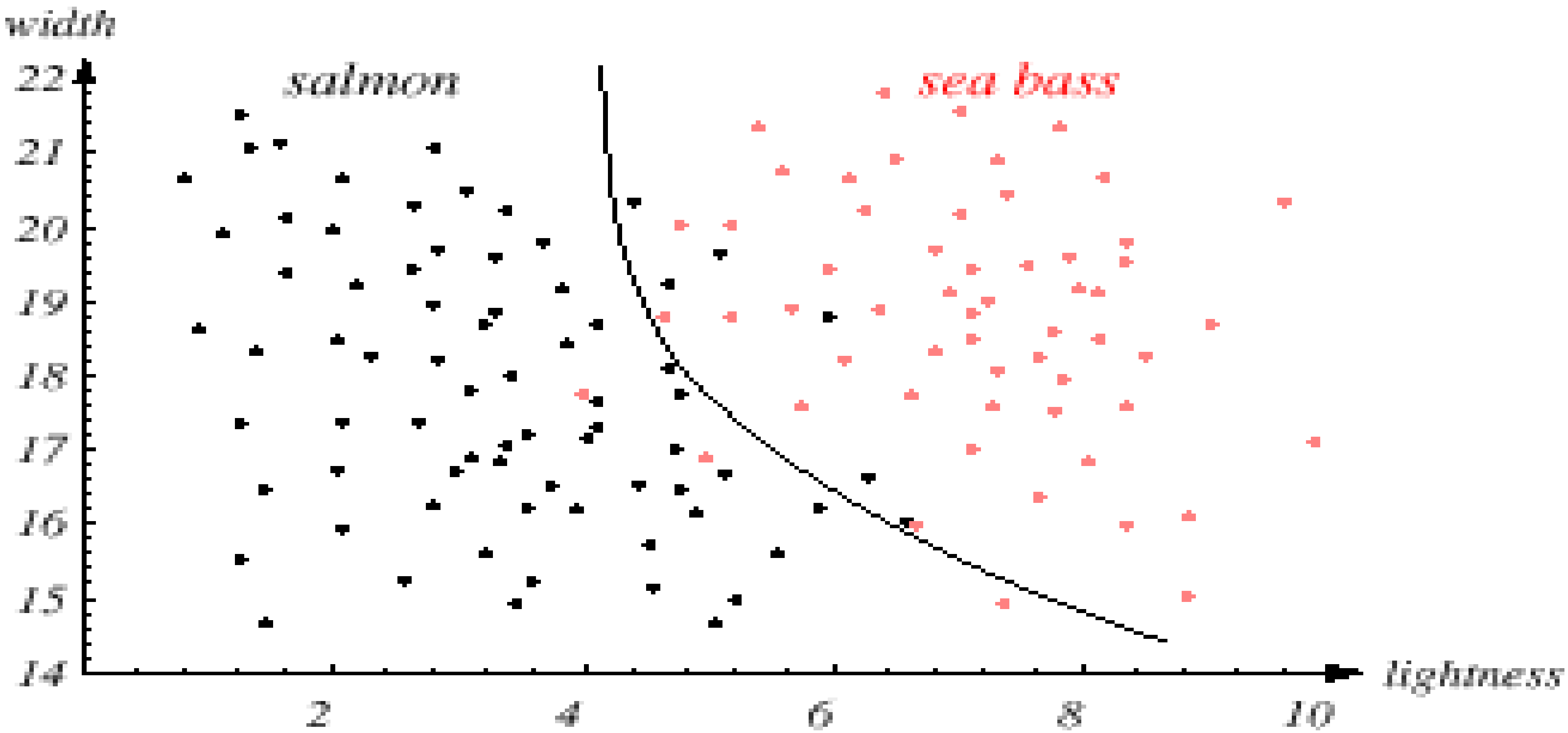
- We might add other features that are not correlated with the ones we already have. A precaution should be taken not to reduce the performance by adding such “noisy features”

- Ideally, the best decision boundary should be the one which provides an optimal performance such as in the following figure:


width
salmon
sea bass
?
lightness
22
21
20
19
18
17
16
15
14
2
4
6
8
10

- However, our satisfaction is premature because the central aim of designing a classifier is to correctly classify novel input

Issue of generalization!



The decision boundary shown might represent the optimal trade off between performance on the training set and simplicity of classifier.

Generalization

- One approach would be to get more training samples for obtaining a better estimate of the true underlying characteristics, for instance the probability distributions of the categories.
- We might be satisfied with the slightly poorer performance on the training samples if it means that our classifier will have better performance on novel patterns (very complex recognizer or simpler classifiers?).

- Decisions are fundamentally task or cost specific.
- Creating a single *general purpose* artificial pattern recognition device is a profoundly difficult challenge.
- Classification is, the task of recovering the model that generated the patterns, different classification techniques are useful depending on the type of candidate models themselves.
- In statistical pattern recognition we focus on the statistical properties of the patterns (generally expressed in probability densities).

- Neural network pattern classification although can be considered its own discipline, because of its somewhat different intellectual pedigree, we will consider it a close descendant of statistical pattern recognition.
- If instead, the model consists of some set of crisp logical rules, then we employ the methods of *syntactic* pattern recognition, where rules or grammars describe our decision.

Representation

- A central aspect in virtually every pattern recognition problem is that of achieving such a “good” representation.
- In some cases patterns should be represented as vectors of real-valued numbers, in others ordered lists of attributes, in yet others descriptions of parts and their relations, and so forth.
- We seek a representation in which the patterns that lead to the same action are somehow “close” to one another, yet “far” from those that demand a different action.

- We might wish to favor a small number of features, which might lead to simpler decision regions, and a classifier easier to train.
- We might also wish to have features that are robust, i.e., relatively insensitive to noise or other errors.
- In practical applications we may need the classifier to act *quickly*, or use few electronic components, memory or processing steps.

Difficulties of Representation

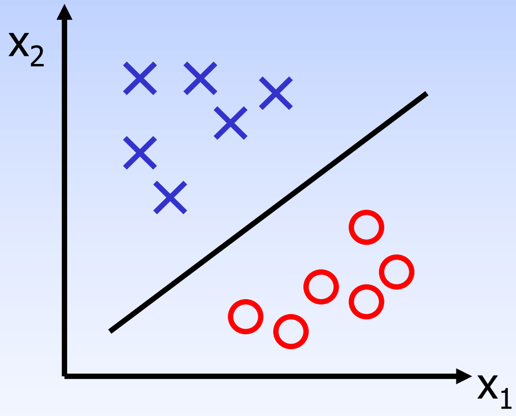
- “How do you instruct someone (or some computer) to recognize caricatures in a magazine, let alone find a human figure in a misshapen piece of work?”
- “A program that could distinguish between male and female faces in a random snapshot would probably earn its author a Ph.D. in computer science.” (Penzias 1989)
- A representation could consist of a vector of real-valued numbers, ordered list of attributes, parts and their relations.

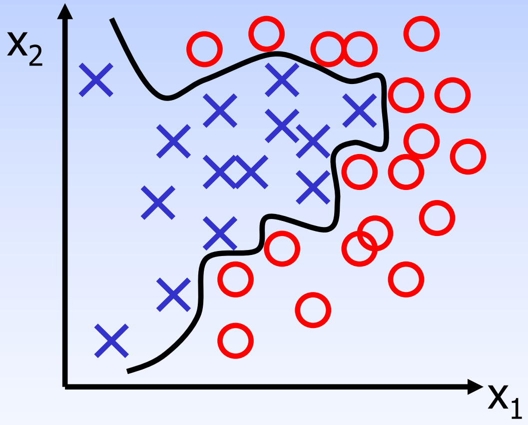
Good Representation

- Should have some invariant properties (e.g., w.r.t. rotation, translation, scale…)
- Account for intra-class variations
- Ability to discriminate pattern classes of interest
- Robustness to noise/occlusion
- Lead to simple decision making (e.g., decision boundary)

Representation

- Each pattern is represented as a point in the *d*-dimensional feature space
- Features are domain-specific and be *invariant* to translation, rotation and scale





Good representation ≡ small intraclass variation, large interclass separation, simple decision rule

Analysis by Synthesis

- A central technique, when we have insufficient training data, is to incorporate knowledge of the problem domain.
- In the ideal case one has a model of how each pattern is generated.
- In speech recognition, for example, the possible "dee"s that might be uttered by different people

- So a “physiological” model (or so-called “motor” model) for production of the utterances is appropriate, and different (say) from that for “doo” and indeed all other utterances.
- If this underlying model of production can be determined from the sound (and that is a very big *if*), then we can classify the utterance by how it was produced. That is to say, the production representation may be the “best” representation for classification.

Related fields

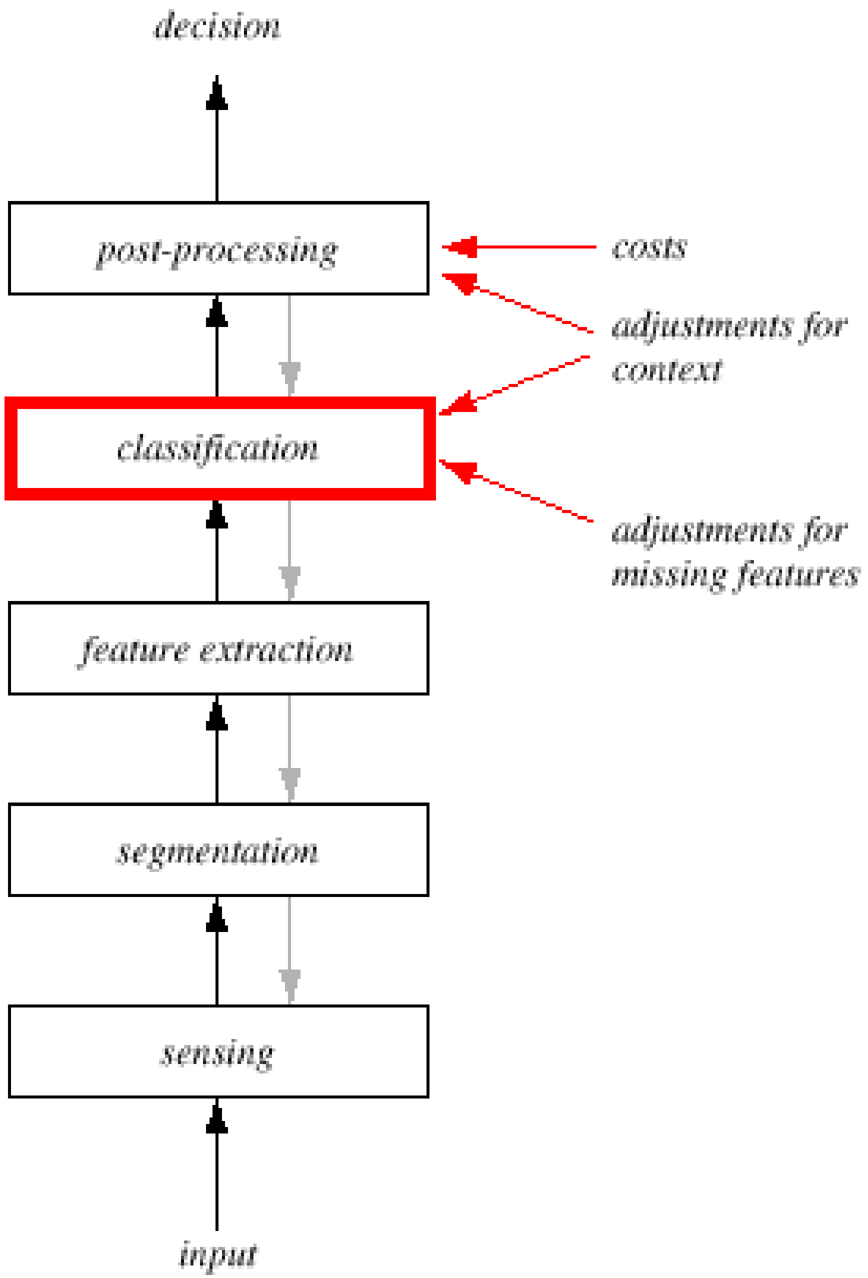
- Pattern classification differs from classical statistical *hypothesis testing*, wherein the sensed data are used to decide whether or not to reject a *null hypothesis* in favor of some alternative hypothesis.
- Pattern classification differs, too, from *image processing*.

Pattern Recognition Systems

- Sensing
  - Use of a transducer (camera or microphone)
  - PR system depends of the bandwidth, the resolution sensitivity distortion of the transducer

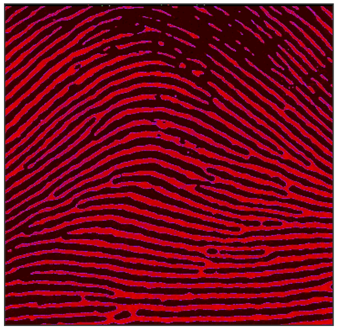
- Segmentation and grouping
  - Patterns should be well separated and should not overlap

- Feature extraction
  - Discriminative features
  - Invariant features with respect to translation, rotation and scale.
- Classification
  - Use a feature vector provided by a feature extractor to assign the object to a category
- Post Processing
  - Exploit context input dependent information other than from the target pattern itself to improve performance


decision
post-processing
costs
adjustments for context
classification
adjustments for missing features
feature extraction
segmentation
sensing
input

Fingerprint Classification

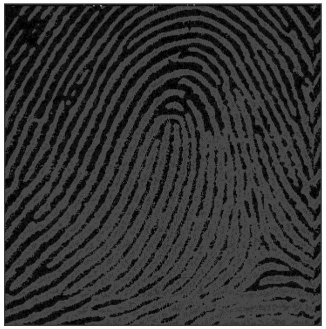
- Assign fingerprints into one of pre-specified types



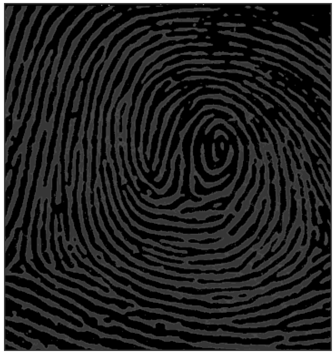
Plain Arch

Tented Arch

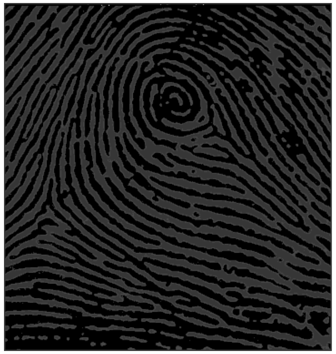
Right Loop



Left Loop

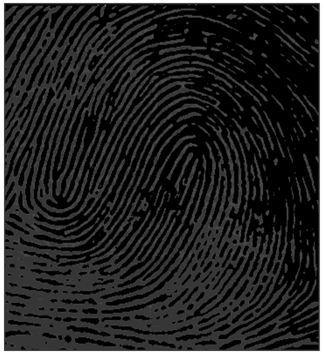


Accidental



Pocket Whorl

Plain Whorl

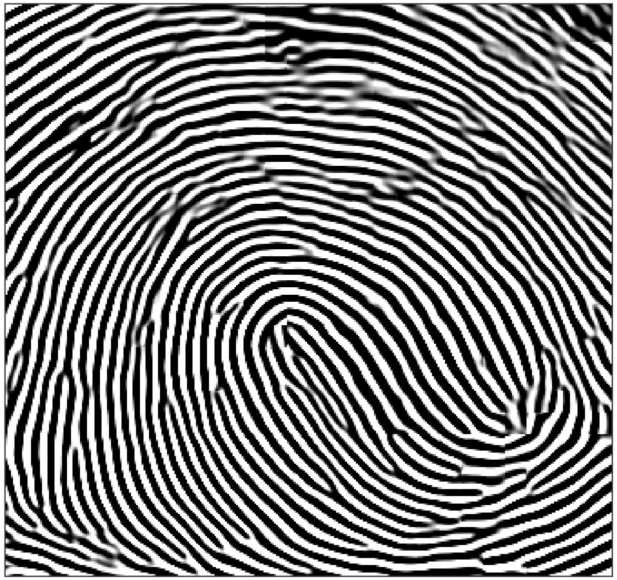


Double Loop

Fingerprint Enhancement

- To address the problem of poor quality fingerprints

Noisy image



Enhanced image

Pattern Recognition System Performance

- Error rate (Prob. of misclassification) on independent test samples
- Speed
- Cost
- Robustness
- Reject option

The Sub-problems of Pattern Classification

- Noise
  - We define *noise* very general terms: any property of the sensed pattern due not to the true underlying model but instead to randomness in the world or the sensors.
- Overfitting
  - While an overly complex model may allow perfect classification of the training samples, it is unlikely to give good classification of novel patterns — a situation known as *overfitting*.

- Model Selection
  - How do we know when a hypothesized model differs significantly from the true model underlying our patterns, and thus a new model is needed?
- Prior Knowledge
  - Information about the production of the patterns
  - The form of the underlying categories
- Missing Features
  - Like *occlusion* by another object

- Context
  - We might be able to use *context* — input-dependent information other than from the target pattern itself— to improve our recognizer.

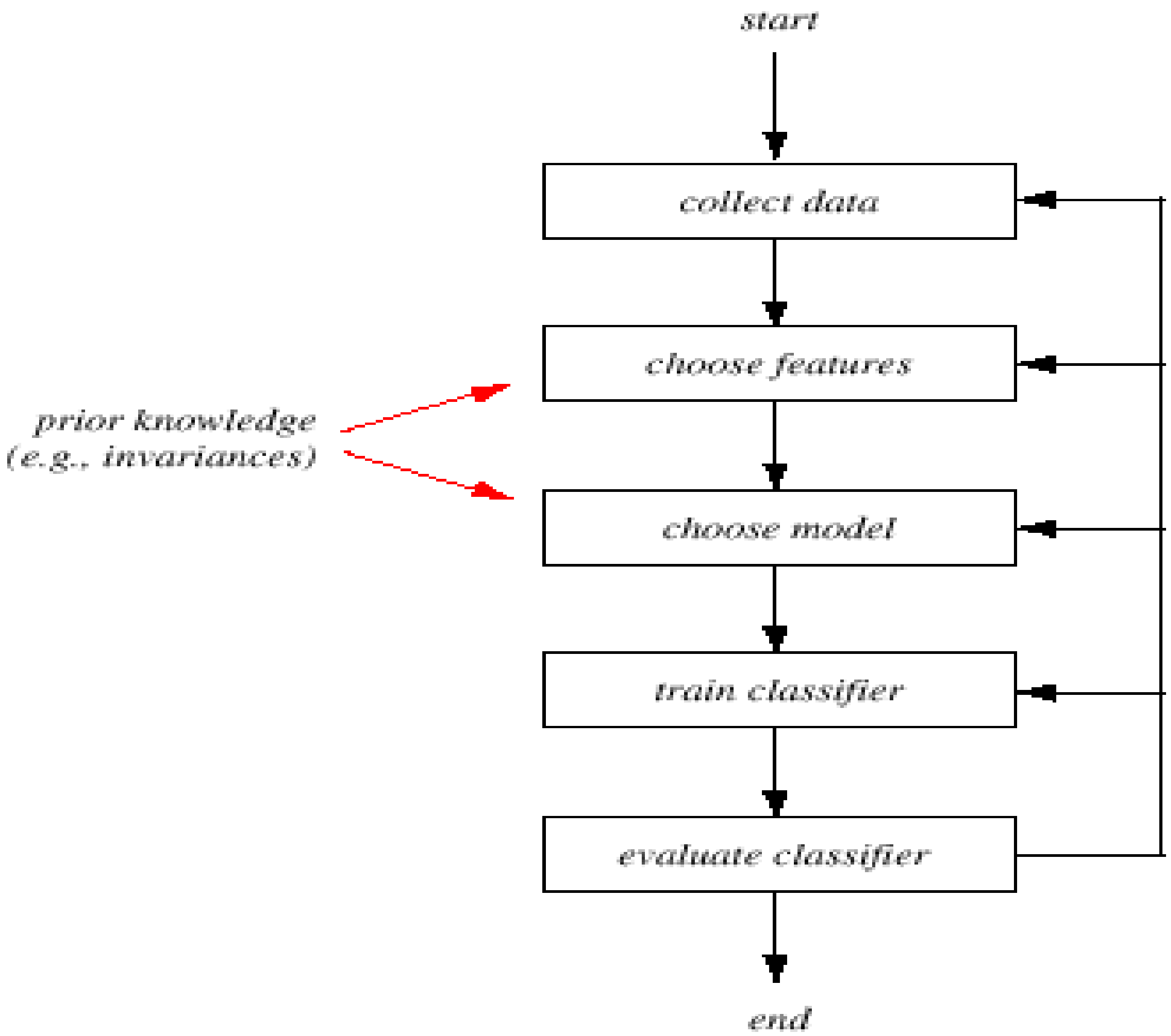
How m ch
info mation are
y u mi sing

- Invariances
  - Invariant to the transformation of *translation* and transformations (like orientation, size, the *rate* at which the pattern evolves, *deformations, discrete* symmetries).
  - How do we determine whether an invariance is present? How do we efficiently incorporate such knowledge into our recognizer?
- Evidence Pooling
  - If we have several component *classifiers,* but suppose they disagree. How should a “super” classifier *pool the evidence* from the component recognizers to achieve the best decision?

- Costs and Risks
  - We often design our classifier to recommend actions that minimize some total expected cost or risk.
  - The simplest risk is the classification error, however the notion of risk is far more general.
  - How do we incorporate knowledge about such risks and how will they affect our classification decision?
- Computational Complexity
  - The computational complexity of different algorithms is of importance, especially for practical applications.

The Design Cycle

- Data collection
- Feature Choice
- Model Choice
- Training
- Evaluation
- Computational Complexity


start
collect data
choose features
choose model
train classifier
evaluate classifier
end
prior knowledge
(e.g., invariances)

- Data Collection

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

- Feature Choice

  Depends on the characteristics of the problem domain. Simple to extract, invariant to irrelevant transformation insensitive to noise.

- Model Choice

  Unsatisfied with the performance of our fish classifier and want to jump to another class of model

- Training

    Use data to determine the classifier. Many different procedures for training classifiers and choosing models

- Evaluation

    Measure the error rate (or performance) and switch from one set of features to another one

- Computational Complexity

    What is the trade off between computational ease and performance?

    (How an algorithm scales as a function of the number of features, patterns or categories?)

Learning and Adaptation

- Supervised learning
  - A teacher provides a category label or cost for each pattern in the training set
- Unsupervised learning
  - The system forms clusters or “natural groupings” of the input patterns
- Reinforcement Learning
  - In *reinforcement learning* or *learning with a critic*, no desired category signal is given; critic instead, the only teaching feedback is that the tentative category is right or wrong.
  - This is analogous to a critic who merely states that something is right or wrong, but does not say specifically *how* it is wrong.

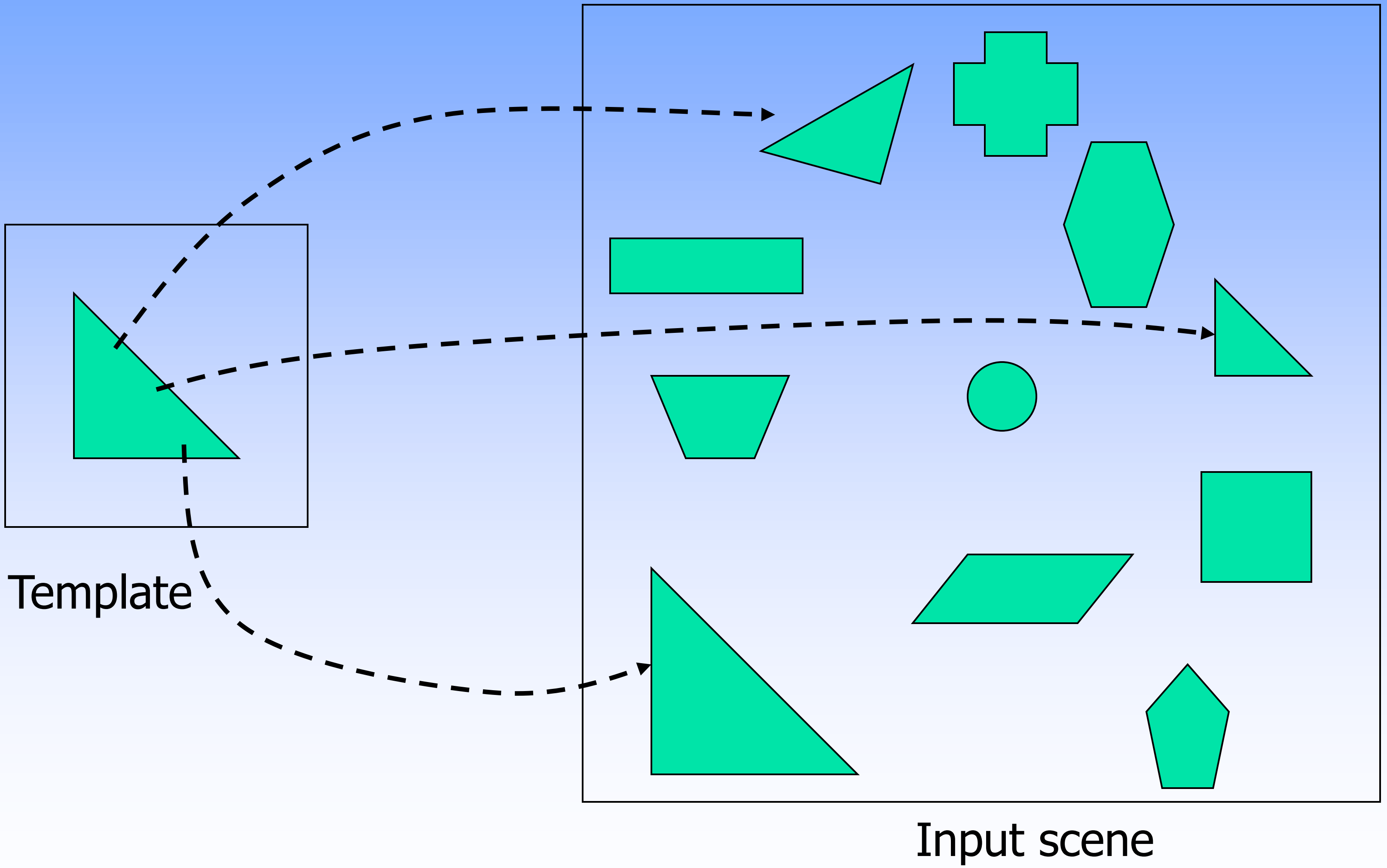
Supervised Classification

Unsupervised Classification

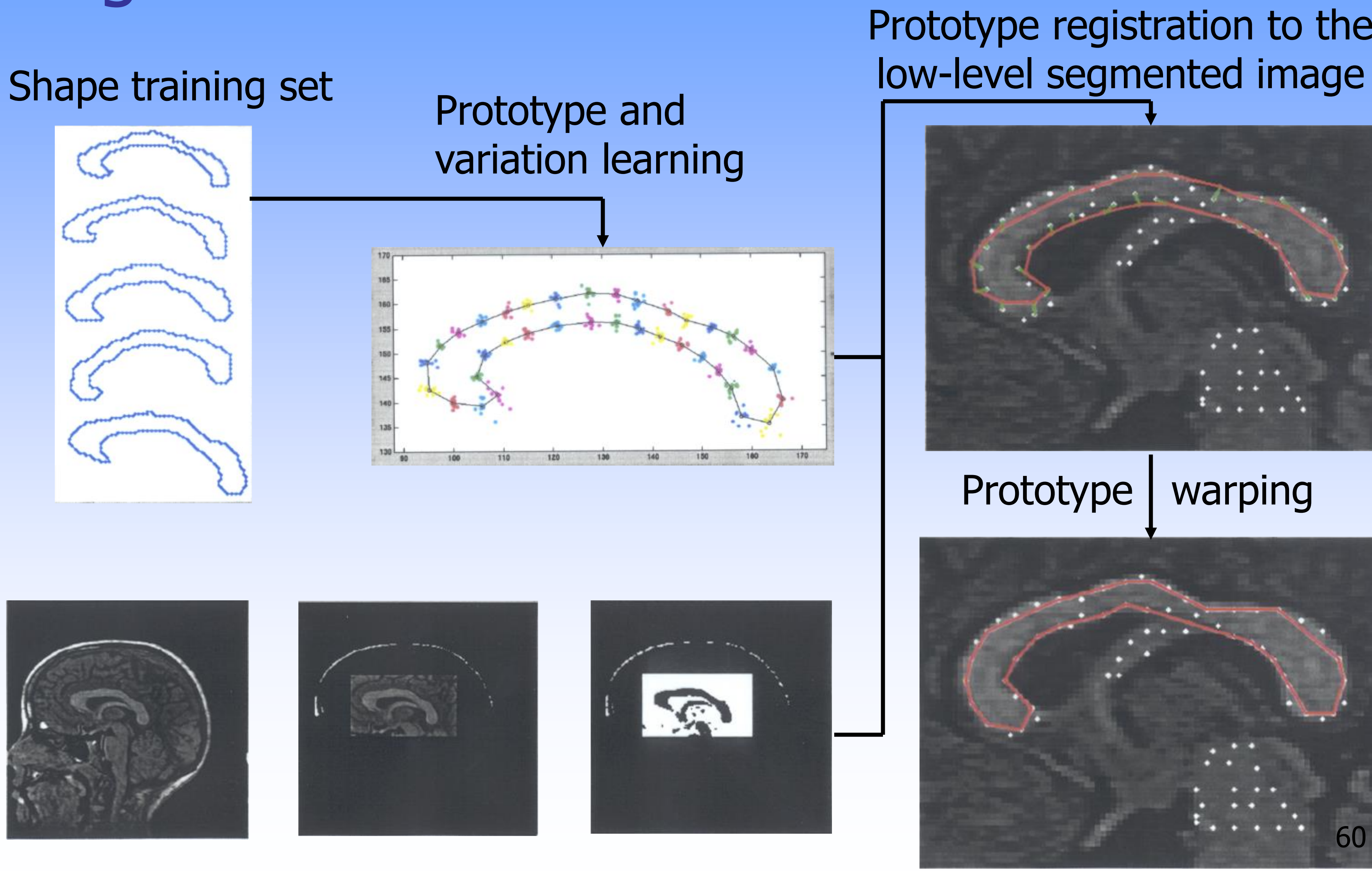
Models for Pattern Recognition

- Template matching
- Statistical (geometric)
- Syntactic (structural)
- Artificial neural network (biologically motivated?)
- Hybrid approach

Template Matching

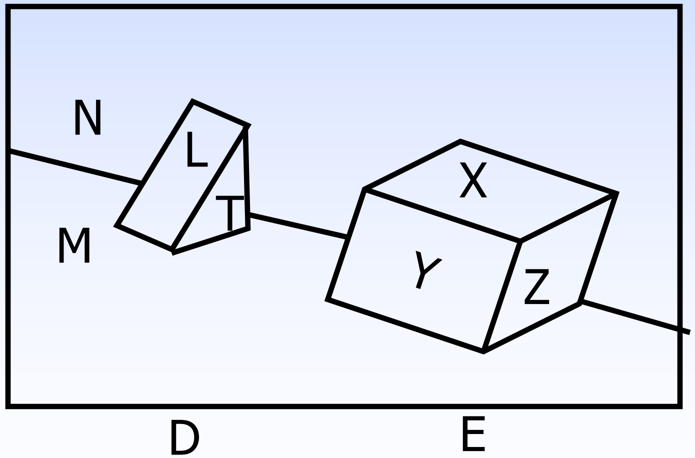


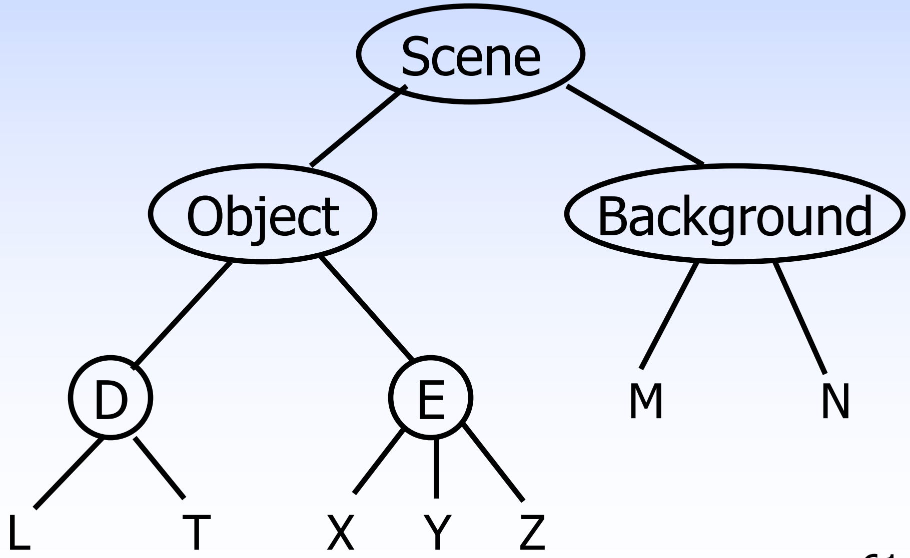
Deformable Template: Corpus Callosum Segmentation



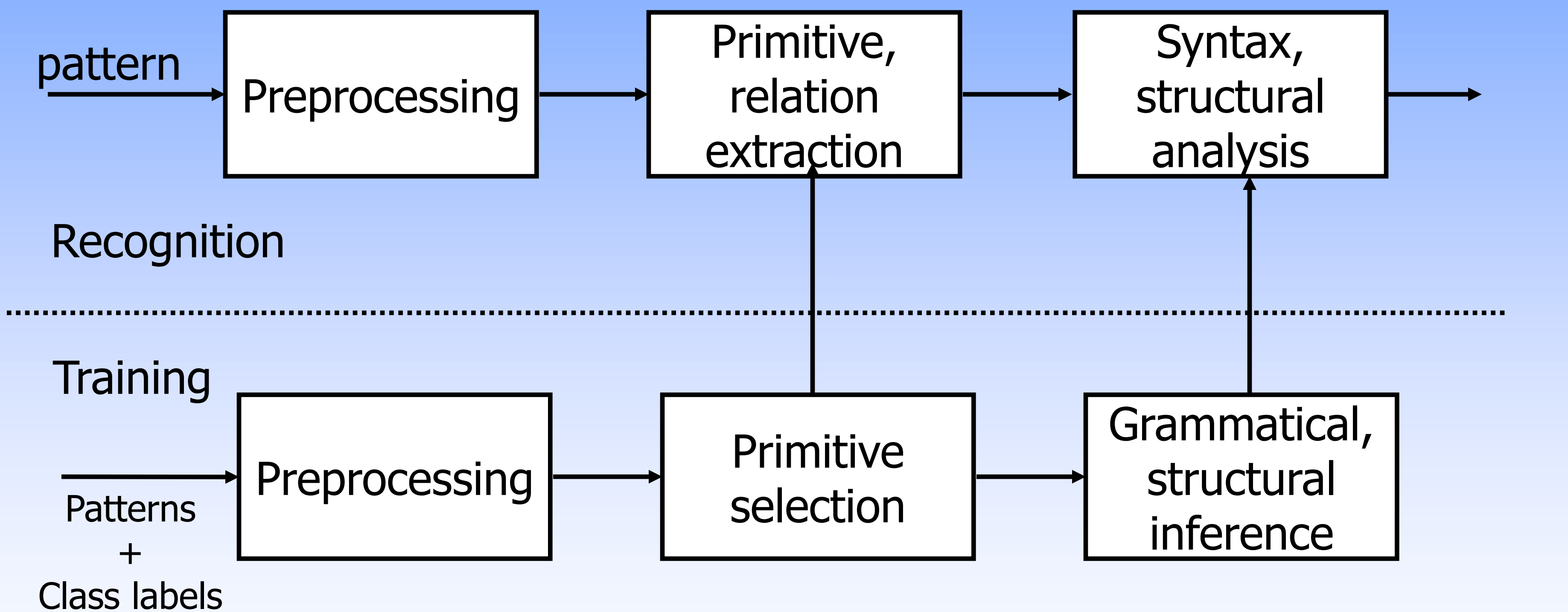
Structural Patten Recognition

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



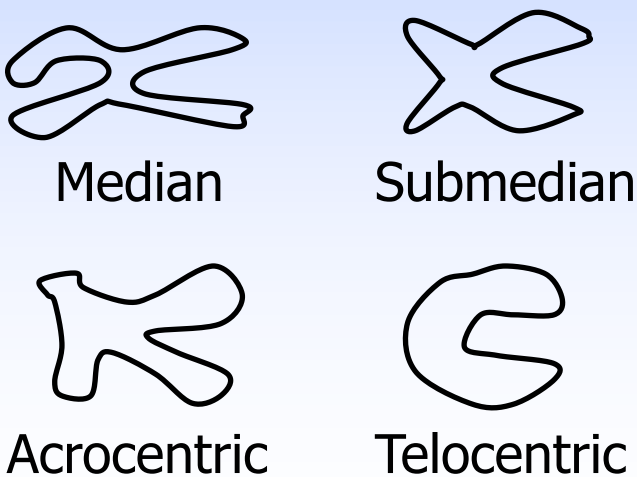


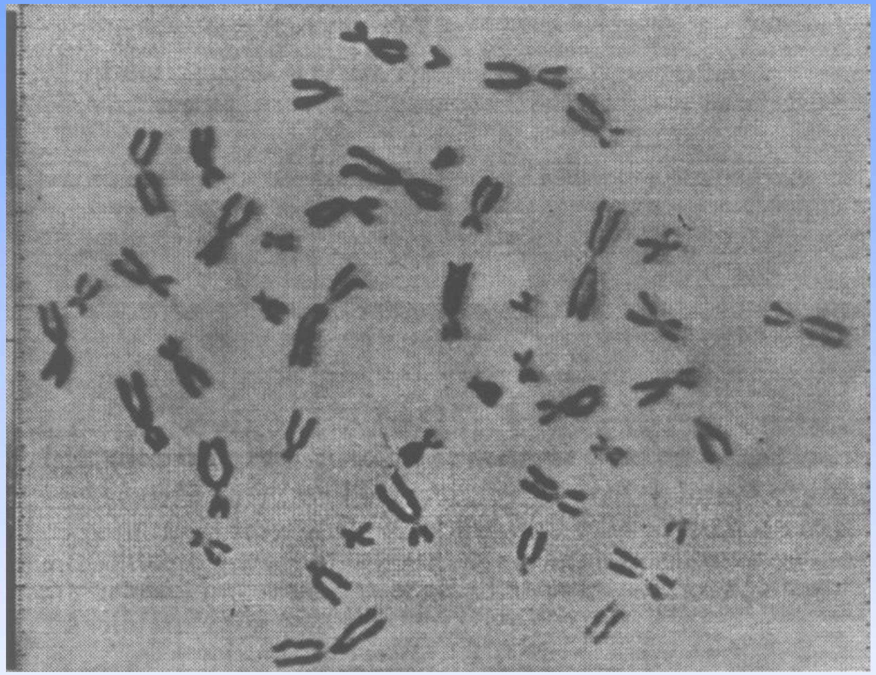
Syntactic Pattern Recognition



Chromosome Grammars

- Terminals:
  $V_T=\{\cap, |, \cup, \wr, \smile\}$
- Non-terminals:
  $V_N=\{A,B,C,D,E,F\}$
- Pattern Classes:





Chromosome Grammars

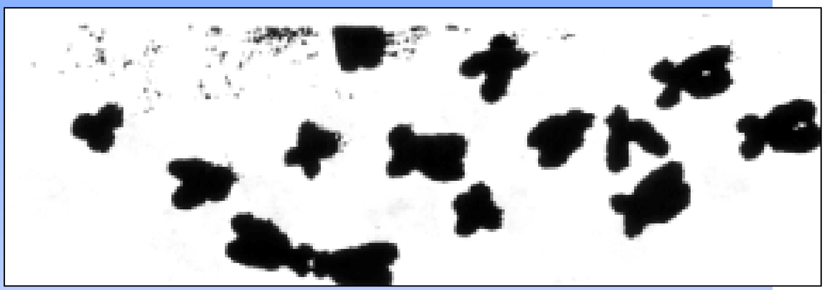
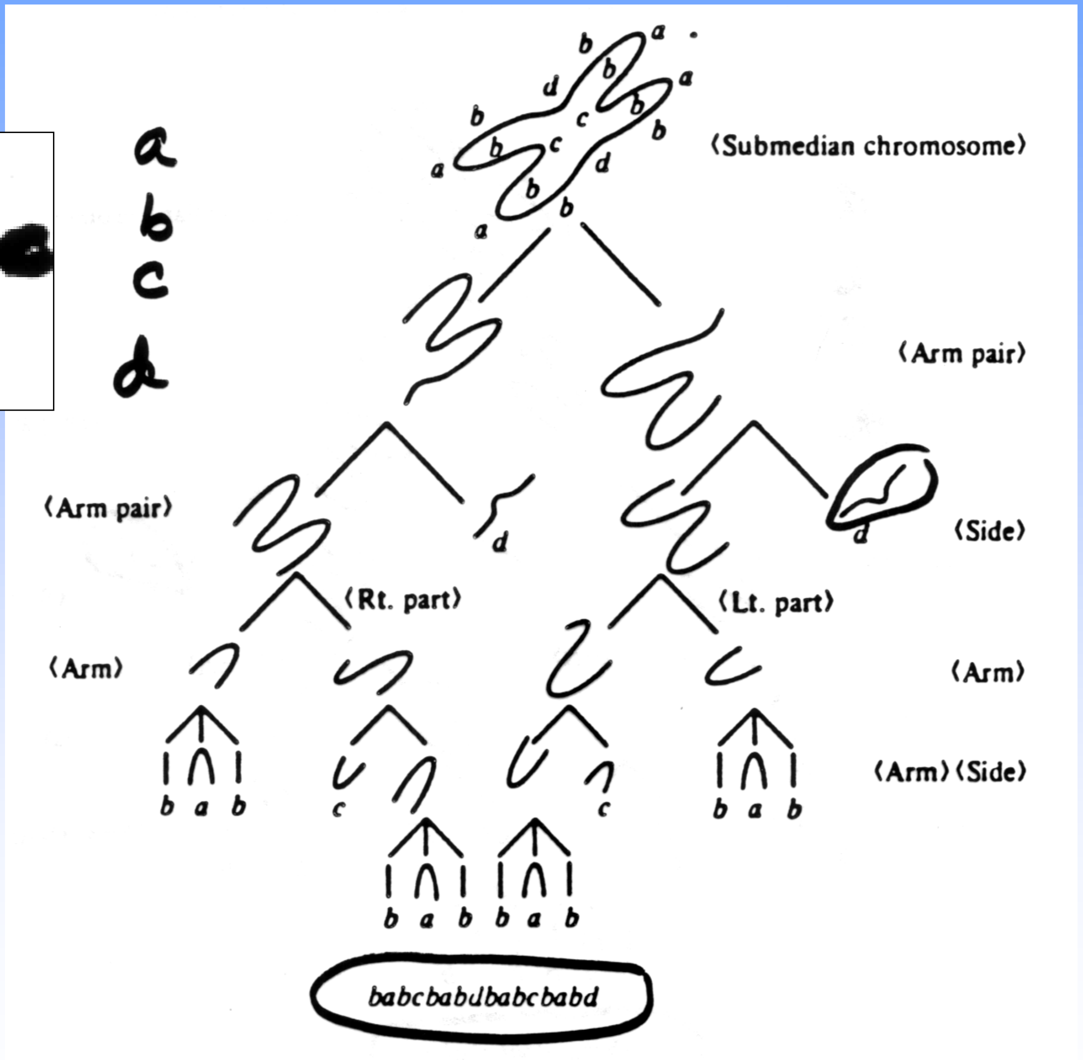


Image of human chromosomes



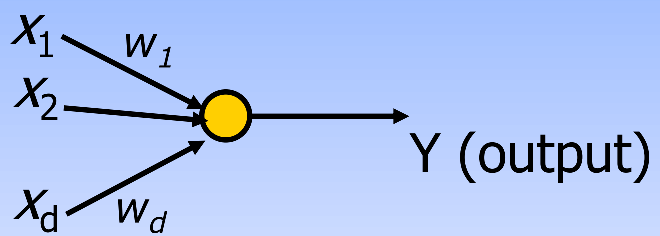
Hierarchical-structure description of a submedian chromosome

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
- Biological networks attempt to achieve good performance via dense interconnection of simple computational elements (neurons)
  - Number of neurons $\approx 10^{10} - 10^{12}$
  - Number of interconnections/neuron $\approx 10^{3} - 10^{4}$
  - Total number of interconnections $\approx 10^{14}$

Artificial Neural Networks

- Nodes in neural networks are nonlinear, typically analog

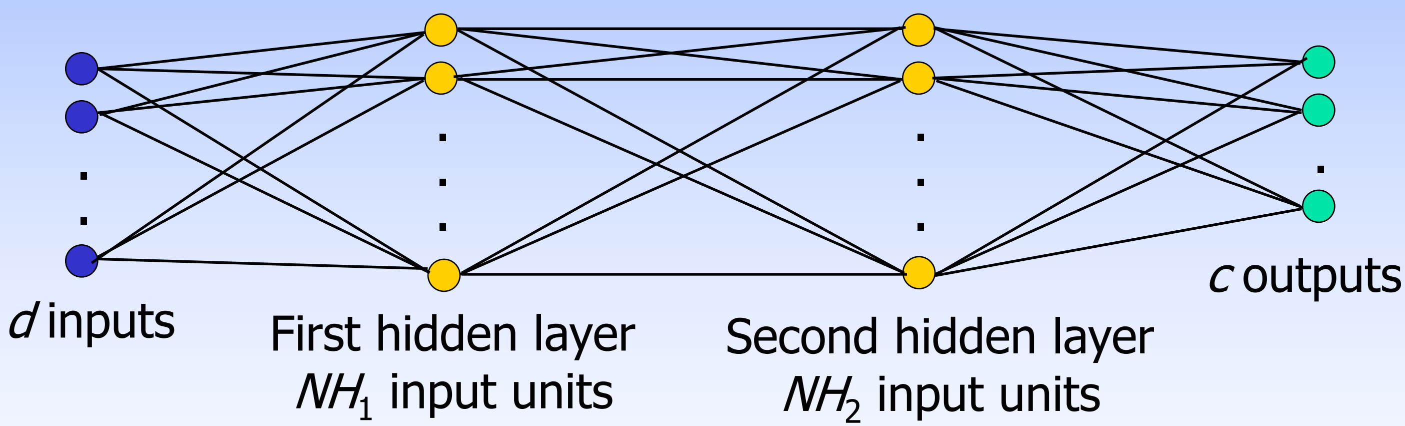


$$Y = f\left(\sum_{i=1}^{d} w_i x_i - \theta\right)$$

where θ is internal threshold or offset

Multilayer Perceptron

- Feed-forward nets with one or more layers (hidden) between the input and output nodes
- A three-layer net can generate arbitrary complex decision regions

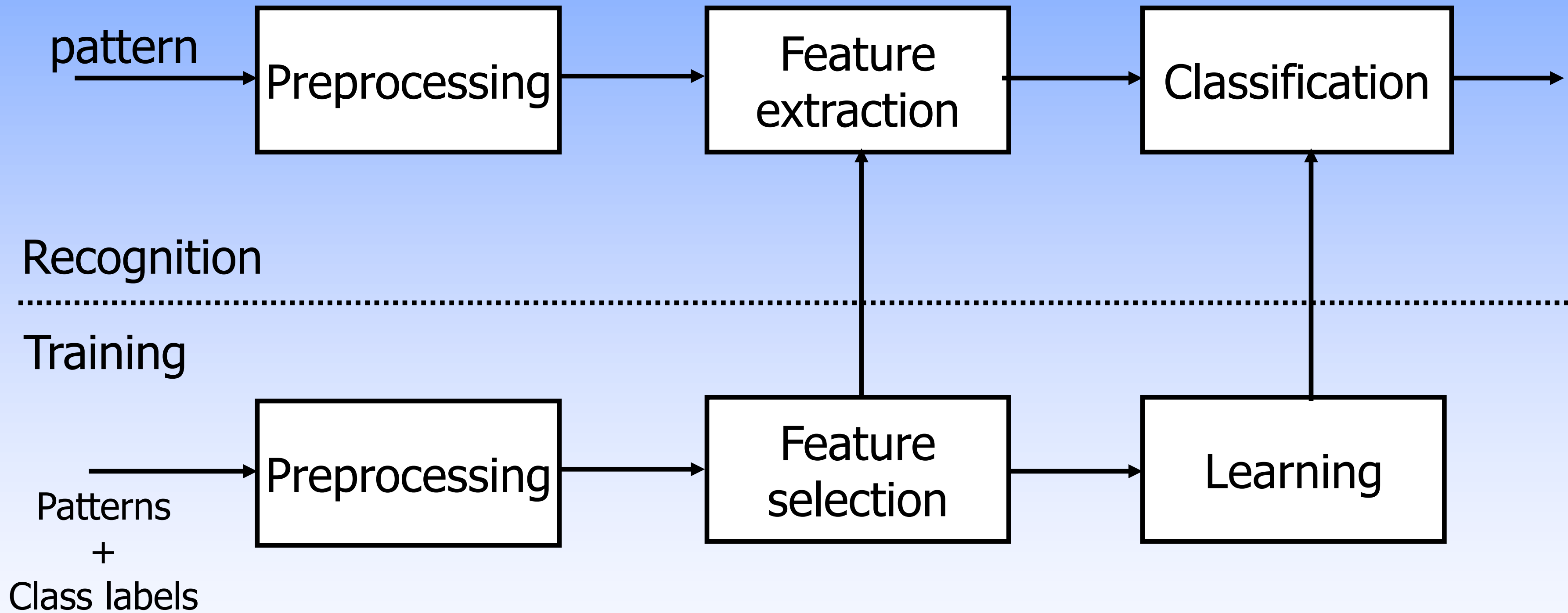


- These nets can be trained by *back-propagation* training algorithm

“Super Classifier”

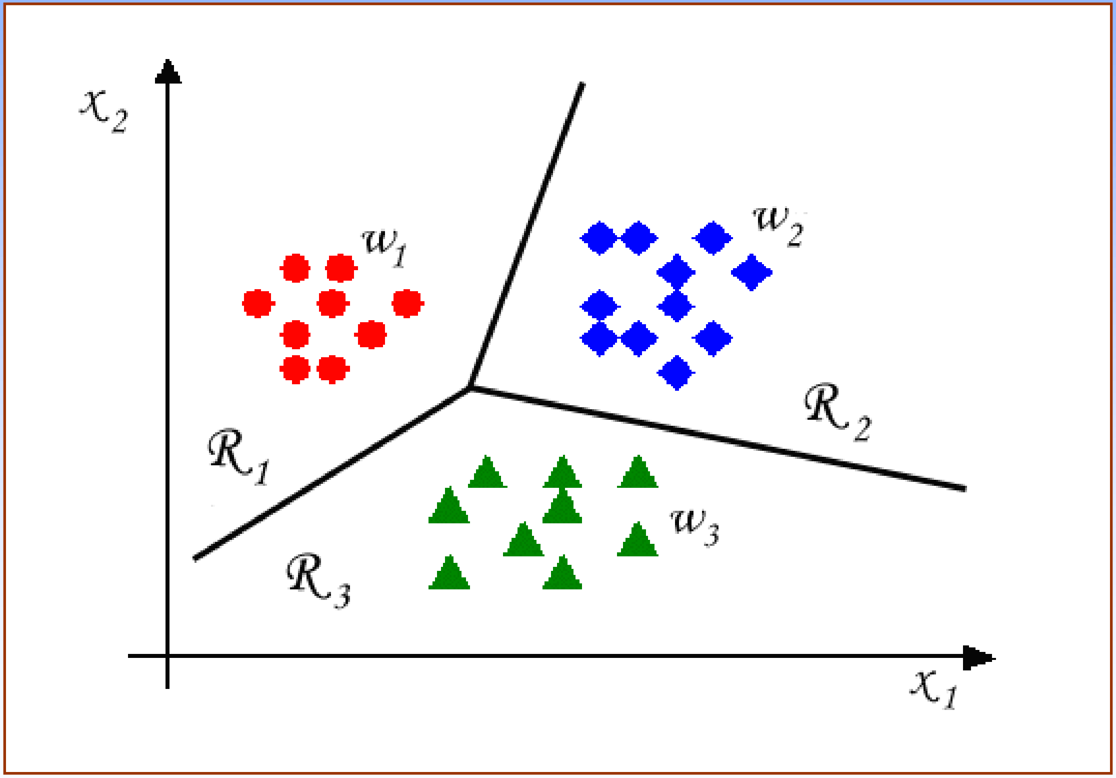
Pool the evidence from component recognizers (classifier combination, mixture of experts, evidence accumulation)

Statistical Pattern Recognition



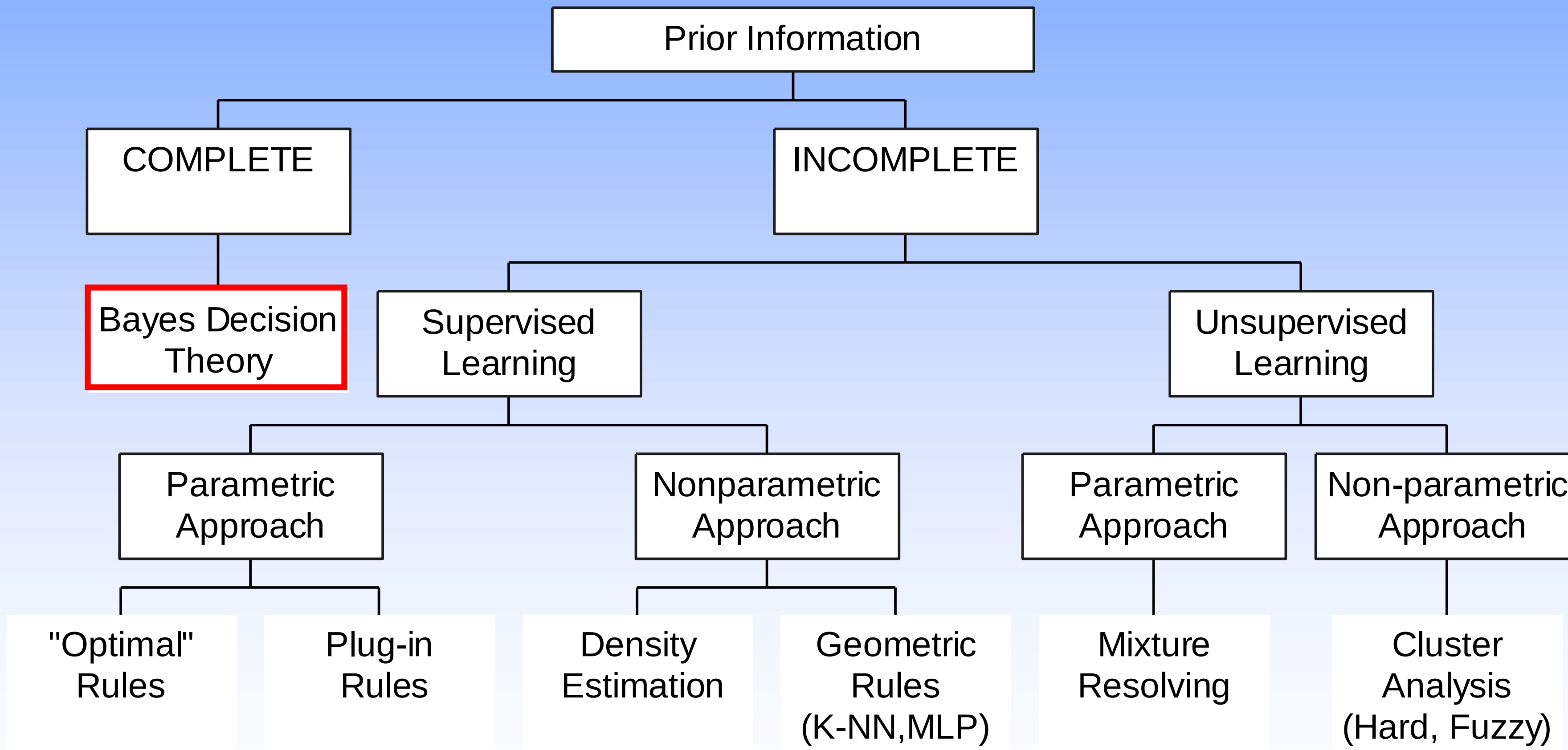
Statistical Pattern Recognition

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



- Given training patterns from each class, goal is to partition the feature space.

Approaches to Statistical Pattern Recognition



Comparing Pattern Recognition Models

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

In practice, statistical and neural network approaches work well