Introduction to Pattern Recognition

Human Perception

• Humans have developed highly sophisticated skills for sensing their environment and taking actions according to what they observe, e.g., recognizing a face, understanding spoken words, reading handwriting, distinguishing fresh food from its smell. We would like to give similar capabilities to machines.

What is Pattern Recognition?

A Pattern is an entity, e.g. object process or event that could be given a name.
Examples are Fingerprint image, Handwritten word, Human face, Speech signal, DNA sequence textures, crystals, biology patterns, constellations, weather pattern, speech waveform etc.
Pattern recognition is the study of how machines can observe the environment, learn to distinguish patterns of interest, make sound and reasonable decisions about the categories of the patterns. In brief, Recognition is the ability to classify patterns.

In other words, a **pattern** is an object, process or event that can be given a name. A **pattern class** (or category) is a set of patterns sharing common attributes and usually originating from the same source. During **recognition** (or **classification**) given objects are assigned to prescribed classes.

Human and Machine Perception

• We are often influenced by the knowledge of how patterns are modeled and recognized in nature when we develop pattern recognition algorithms.
• Research on machine perception also helps us gain deeper understanding and appreciation for pattern recognition systems in nature.
• Yet, we also apply many techniques that are purely numerical and do not have any correspondence in natural systems.

PATTERN RECOGNITION AREAS

1. **Optical Character Recognition (OCR)**: Today's OCR systems are very useful to Reconstruct text from printed materials e.g. newspaper, textbooks, letters etc. Especial applications are in Sorting letters by postal code and reading aid for blind people.

2. **Analysis and identification of human patterns**: Human Speech and voice recognition finds growing application in person identification. Also it begins to apply in text to speech and speech to text system. Person identification also uses Fingerprints, face, iris and DNA mapping to control access in a secured system. Another area is dating services, where pattern includes age, sex, race, hobbies, income, etc.

3. **Banking and insurance applications**: In banking area, Credit cards applicants classified by income, credit worthiness, mortgage amount, Number of dependents, etc.
Also Car insurance uses pattern including make of car, Number of accidents, age, sex, driving habits, location, etc.

4. **Diagnosis systems**: Medical diagnosis systems include disease vs. symptoms classification, X-Ray, EKG and tests analysis, etc. It also includes diagnosis of automotive malfunctioning.

5. **Prediction systems**: This area includes Weather forecasting from satellite data, Analysis of seismic patterns etc.

6. **Security and military applications**: Computer security where threats are identified through analysis of network traffic patterns.

By Image segmentation and analysis military systems recognizes missile silos and terrorist camps from aerial photographs. Automatic target recognition systems use Optical or infrared image to determine types of target. Radar signal classification produces necessary precautions against various types of air attacks.

7. **Industrial Area:** In Industrial automation, the pattern of Intensity or range image is used to identify defective/non-defective products e.g. Printed circuit board inspection to determine its validity, Fruit sorting on a conveyor belt to determine the Grade of quality etc.

TERMINOLOGY

Feature: The relevant and intrinsic trait or characteristic of a pattern that makes it apart from another. It is the data extractable through measurement and/or processing from a pattern.
Examples are color, age, weight, aspect ratio, etc.

Pattern class: It is the set of patterns that sharing a set of common attributes or features and usually originating from the same source associated with the generalization or abstraction of patterns.

Classification: It means to assign patterns into classes based on features. If the pattern is recognized, known or understood then it is called **Correct classification** and if the pattern is not known, misunderstood or missed then it is called **Wrong classification.**

Errors: If a recognition process results in a Confusion matrix, from where no decision can be drawn, then it is an error.

Noise: These are the distortions associated with pattern processing e.g. errors in feature extraction and/or training samples that impact the classification or recognition abilities of the system.

PR APPROACHES

There are two main PR approaches. One is **Sub-Symbolic** and other is **Symbolic.**
Sub-Symbolic PR Approaches:
Machine appears as a network of cells modeling the neurons of the human brain.
Process only numeric symbols.
Reasoning model: Connectionist reasoning

Symbolic Level: Reasoning Systems
Not modeled on the human brain
Machine manipulate numeric symbols (0 to 9) as well as non numeric symbols encoding domain specific information.
Symbols are processed with the mathematical logic of formal reasoning.
Two basic reasoning models:
Case-based reasoning: problems to match and to adapt to new problem contexts
Rule-based reasoning (expert systems): difficult to have set of rules covering every possible input (and systems often breaking down: “brittle” systems)

1. Sub-Symbolic: Artificial Neural Network Classifiers
2. Symbolic: Traditional Classifiers:
Statistical: assume that underlying model is a set of probabilities “learned from examples” (while internal structural model is ignored).
Structural or Syntactic: assume that interrelations between parts (features) are more important and finding them (though not easy) may require prior knowledge about internal structure.

Traditional Pattern Recognition Techniques:
1. Parametric Methods: *Bayesian*

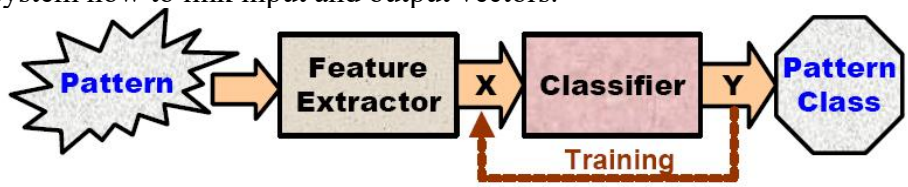
Each class is described along pre-defined parameters and it is identified by the parametric distribution it exhibits. Statistical distribution of parameters is learned through training.
2. Non-Parametric Methods: *Nearest-Neighbor, Parzen Windows* Pattern classes are not associated with parametric distributions, but with density functions and/or decision regions within feature space.
A crude classification model (usually based on intuition) is iteratively refined through training.

Approaches

Statistical PR: based on underlying statistical model of patterns and pattern classes.
Structural (or syntactic) PR: pattern classes represented by means of formal structures as grammars, automata, strings, etc.
Neural networks: classifier is represented as a network of cells modeling neurons of the human brain (connectionist approach).

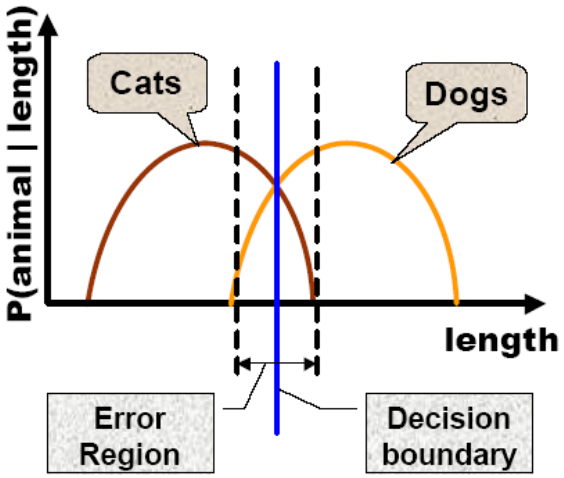
GENERAL PROBLEM

Given an input set of features (or input vector) extracted from a sample pattern:
X = {x1, x2, …, xn}
find the associated set of characteristics (output vector) matching the input according to a pre-defined criteria:
Y = {y1, y2, …, ym}
It is assumed that a set of training samples was previously used to "teach" the recognition system how to link input and output vectors.



DECISION BOUNDARY

How to distinguish a cat from a dog? Suppose we choose to make the classification based on the length of the animal (from tail to snout). Then, the length is our feature for classification.



Comp 473/6731

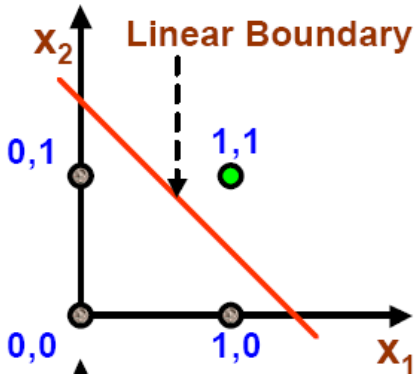
CLASSIFIERS

A classifier partitions the feature space into class-labeled decision regions. The border of each region is a decision boundary.
Ideally, an unique class assignment is achieved if, and only if, the decision regions cover the entire feature space and they are disjoint (i.e. the decision boundaries do not create error or confusion areas).
Classification strategy:
Extract the feature vector of the pattern.

Compare the feature vector with the characteristics of existing decision regions (i.e. with features of each pattern class).
Assign pattern to the class (associated with decision region) matching the feature vector extracted.

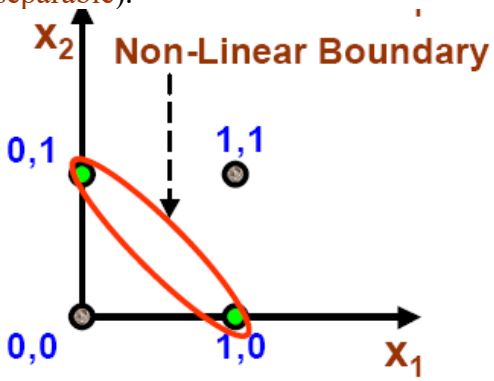
DECISION SPACE

AND Gate: patterns can be separated by a single line (they are said to be linearly separable). The same logic is applicable to an OR-gate.



X_1X_2	AND	OR	XOR
00	0	0	0
01	0	1	1
10	0	1	1
11	1	1	0

XOR Gate: patterns cannot be separated by a single line (they are said to be non-linearly separable).



LEARNING

Problem: we fit a model based on past experience (past examples seen), but we are ultimately interested in learning the mapping that performs well on the entire population of examples.

Training data is used to adjust (iteratively) the parameters of the model until it fits; training is considered accomplished when overall error falls below a pre-define threshold.

Training error:

S(yi - f(xi))2

Generalized error (over the whole population):

Ex,y = (y - f(x))2

□□*Minimizing the training error does not imply a low generalized error*.
Comp 473/6731 15
TRAINING & LEARNING
□□Learning models:
□Supervised learning: the classifier system is "taught" to associate patterns with their respective classification labels (provided along with each training pattern).
□Unsupervised learning or clustering: the system is directed (based on some pre-defined hypothesis) to form clusters (or "natural groupings") of the training patterns (that are not labeled).
□□Recommended when labeling is very difficult or very expensive.
□□It requires insight into the problem domain.
□□Implies data mining and pattern discovery.
□Reinforced learning: training is assisted by a "critic" that provides a binary feedback by informing the system whether the classification associated with a given training sample is either right or wrong.
□□Issues:
□Over-fitting the training data.
□Over-generalizing the training data.
Comp 473/6731 16

Example#2

• Problem: Sorting incoming fish on a conveyor belt according to species
• Assume that we have only two kinds of fish: sea bass and salmon



Figure 1: Picture taken from a camera.

Decision Process

• What kind of information can distinguish one species from the other?
I length, width, weight, number and shape of fins, tail shape, etc.
• What can cause problems during sensing?
I lighting conditions, position of fish on the conveyor belt, camera noise, etc.
• What are the steps in the process?
I capture image ! isolate fish ! take measurements
! make decision

Selecting Features

• Assume a fisherman told us that a sea bass is generally longer than a salmon.
• We can use length as a feature and decide between sea bass and salmon according to a threshold on length.
• How can we choose this threshold?

An Example: Selecting Features

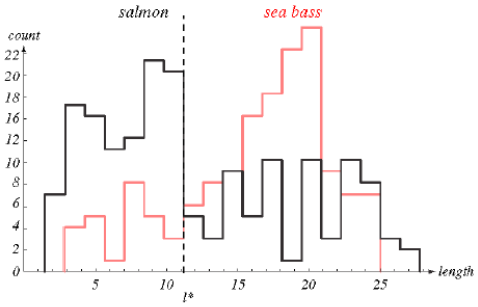


Figure 2: *Histograms* of the length feature for two types of fish in *training samples*. How can we choose the threshold l^* to make a reliable decision?

Selecting Features

• Even though sea bass is longer than salmon on the average, there are many examples of fish where this observation does not hold.
• Try another feature: average lightness of the fish scales.

An Example: Selecting Features

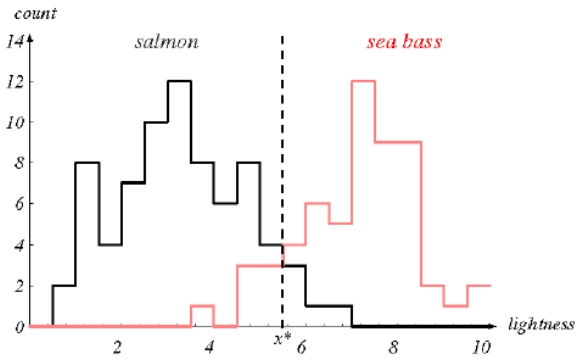


Figure 3: Histograms of the lightness feature for two types of fish in training samples. It looks easier to choose the threshold x^* but we still cannot make a perfect decision.

Cost of Error

• We should also consider costs of different errors we make in our decisions.
• For example, if the fish packing company knows that:
I Customers who buy salmon will object vigorously if they see sea bass in their cans.
I Customers who buy sea bass will not be unhappy if they occasionally see some expensive salmon in their cans.
• How does this knowledge affect our decision?

Multiple Features

• Assume we also observed that sea bass are typically wider than salmon.
• We can use two features in our decision:
I lightness: x1
I width: x2
• Each fish image is now represented as a point (feature vector)
x = (x1, x2) in a two-dimensional feature space.

An Example: Multiple Features

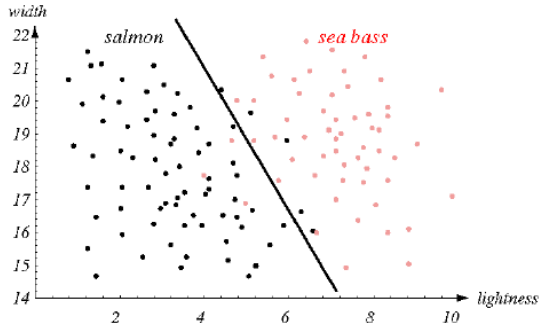


Figure 4: Scatter plot of lightness and width features for training samples. We can draw a *decision boundary* to divide the feature space into two regions. Does it look better than using only lightness?

- Does adding more features always improve the results?
  - Avoid unreliable features.
  - Be careful about correlations with existing features.
  - Be careful about measurement costs.
  - Be careful about noise in the measurements.
- Is there some curse for working in very high dimensions?

Decision Boundaries

- Can we do better with another decision rule?
- More complex models result in more complex boundaries.

An Example: Decision Boundaries

- Can we do better with another decision rule?
- More complex models result in more complex boundaries.

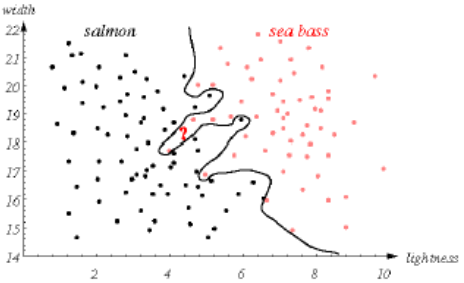


Figure 5: We may distinguish training samples perfectly but how can we predict how well we can *generalize* to unknown samples?

- How can we manage the tradeoff between complexity of decision rules and their performance to unknown samples?

An Example: Decision Boundaries

- How can we manage the *tradeoff* between complexity of decision rules and their performance to unknown samples?

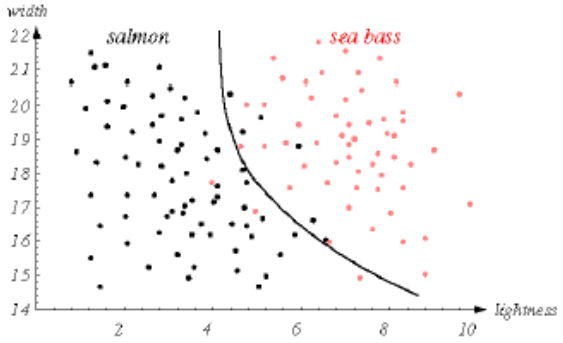


Figure 6: Different criteria lead to different decision boundaries.

Pattern Recognition Systems

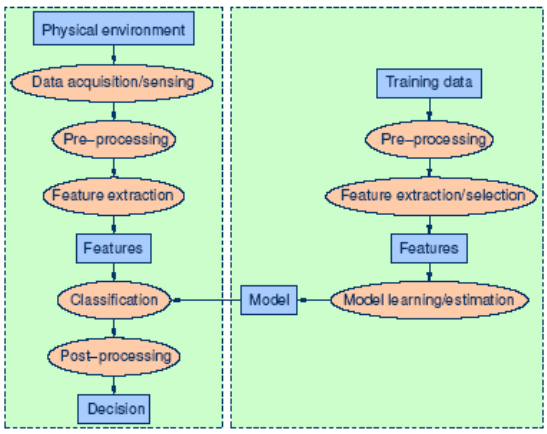


Figure 7: Object/process diagram of a pattern recognition system.

Pattern Recognition Systems
• Data acquisition and sensing:
I Measurements of physical variables
I Important issues: bandwidth, resolution, sensitivity, distortion, SNR, latency, etc.
• Pre-processing:
I Removal of noise in data

I Isolation of patterns of interest from the background
• Feature extraction:
I Finding a new representation in terms of features
CS 551, Spring 2005 17/25
Pattern Recognition Systems
• Model learning and estimation:
I Learning a mapping between features and pattern groups and categories
• Classification:
I Using features and learned models to assign a pattern to a category
• Post-processing:
I Evaluation of confidence in decisions
I Exploitation of context to improve performance
I Combination of experts
CS 551, Spring 2005 18/25

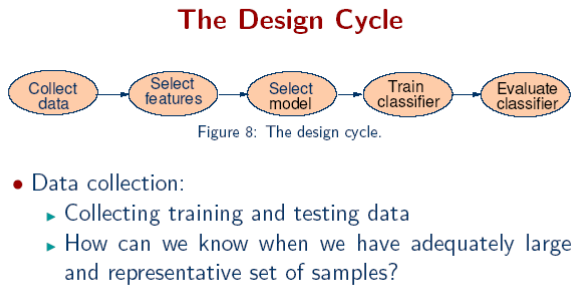
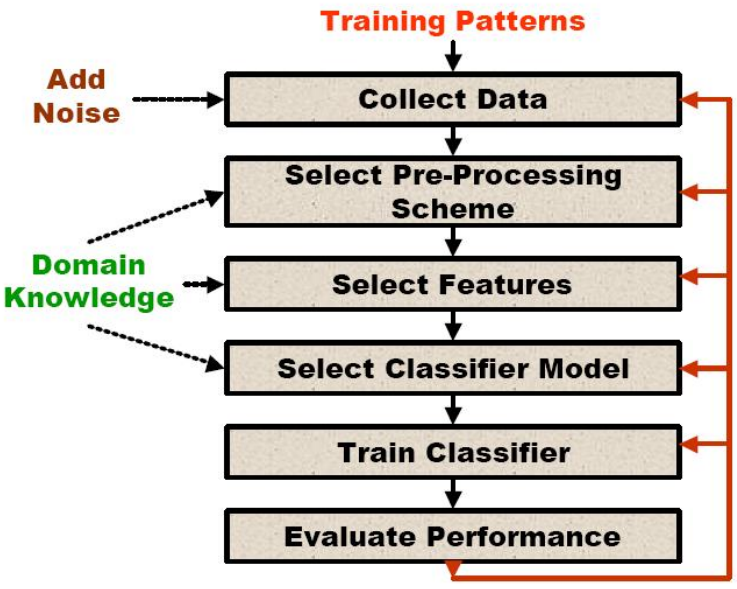


Figure 8: The design cycle.

SYSTEM DESIGN CYCLE



• Data collection:

I Collecting training and testing data
I How can we know when we have adequately large and representative set of samples?

The Design Cycle

• Feature selection:
I Domain dependence and prior information
I Computational cost and feasibility
I Discriminative features
– Similar values for similar patterns
– Different values for different patterns
I Invariant features with respect to translation, rotation and scale
I Robust features with respect to occlusion, distortion, deformation, and variations in environment

The Design Cycle

• Model selection:
I Domain dependence and prior information
I Definition of design criteria
I Parametric vs. non-parametric models
I Handling of missing features
I Computational complexity
I Types of models: templates, decision-theoretic or statistical, syntactic or structural, neural, and hybrid
I How can we know how close we are to the true model underlying the patterns?

The Design Cycle

• Training:
I How can we learn the rule from data?
I Supervised learning: a teacher provides a category label or cost for each pattern in the training set
I Unsupervised learning: the system forms clusters or natural groupings of the input patterns
I Reinforcement learning: no desired category is given but the teacher provides feedback to the system such as the decision is right or wrong

The Design Cycle

• Evaluation:
I How can we estimate the performance with training samples?
I How can we predict the performance with future data?
I Problems of overfitting and generalization