Introduction to Pattern Recognition
Part I

Selim Aksoy
Bilkent University
Department of Computer Engineering
saksoy@cs.bilkent.edu.tr

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, vaguely defined, that could be given a name, e.g.,
  - fingerprint image
  - handwritten word
  - human face
  - speech signal
  - DNA sequence
  - ...
- *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

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.

An Example

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

Figure 1: Picture taken from a camera.

An Example: Decision Process

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

An Example: 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

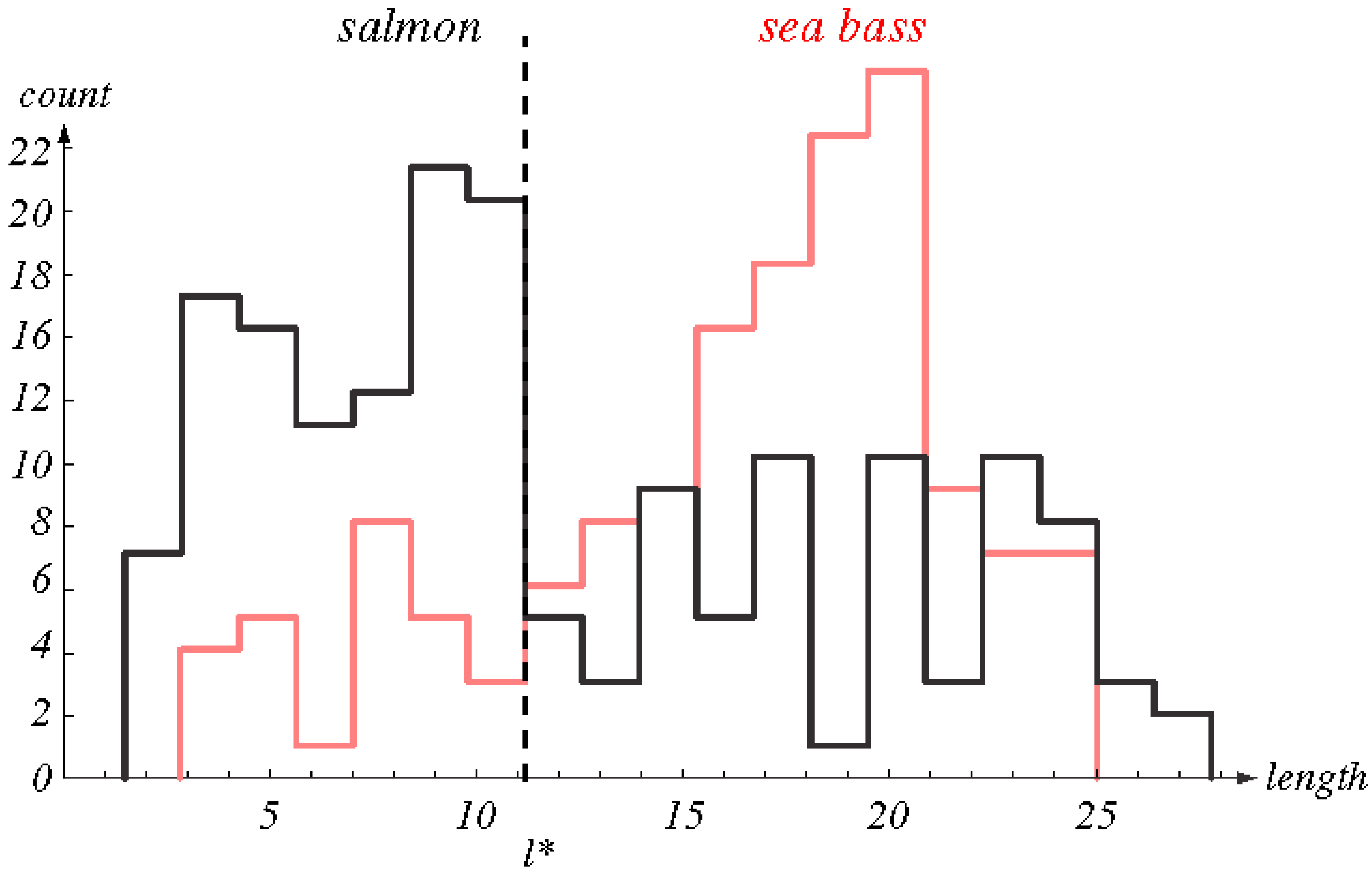

Figure 2: *Histograms* of the length feature for two types of fish in *training samples*. How can we choose the threshold l^* to make a reliable decision?

An Example: 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

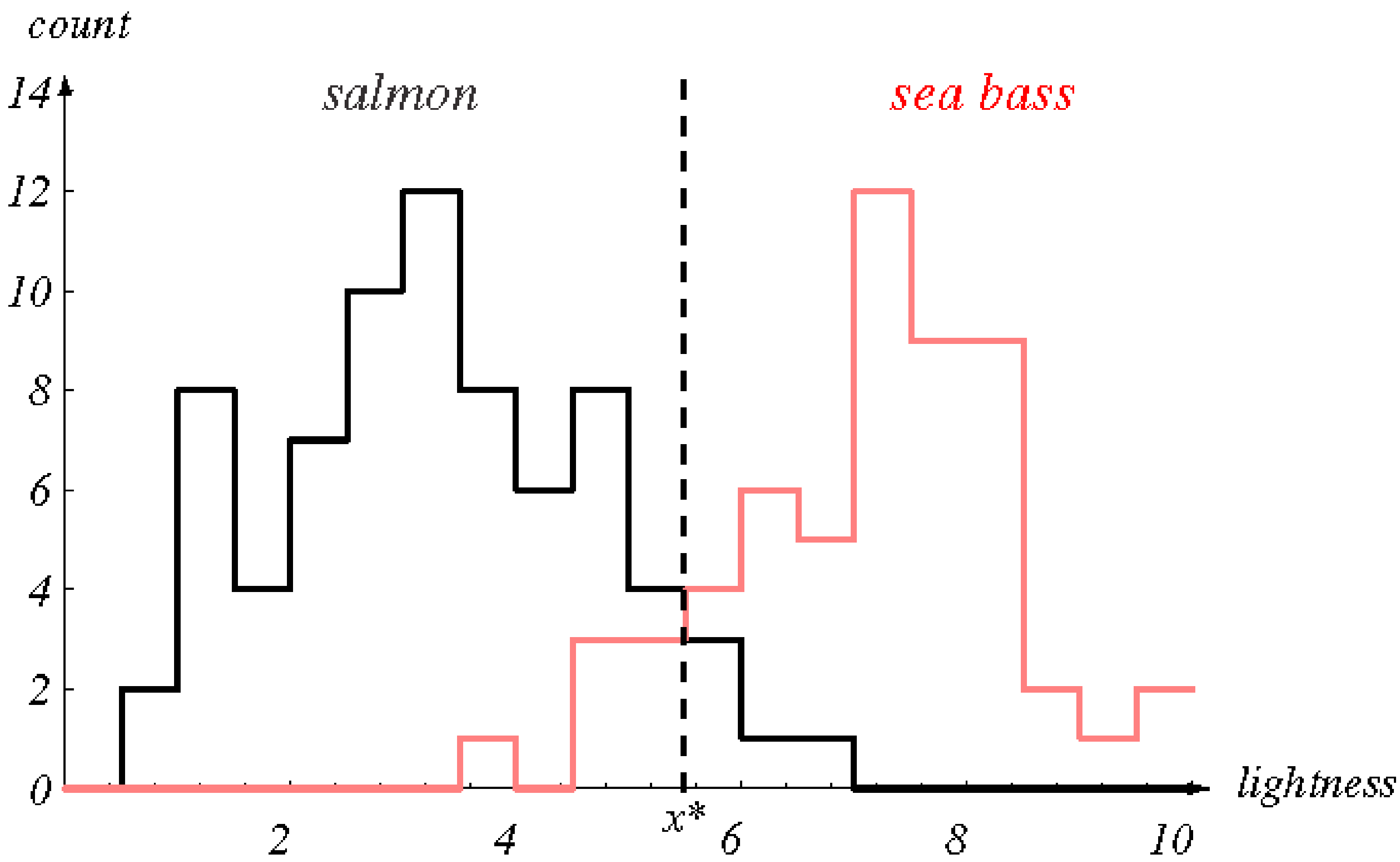

Figure 3: Histograms of the lightness feature for two types of fish in training samples. It looks easier to choose the threshold x^* but we still cannot make a perfect decision.

An Example: 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:
  - Customers who buy salmon will object vigorously if they see sea bass in their cans.
  - 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?

An Example: Multiple Features

- Assume we also observed that sea bass are typically wider than salmon.
- We can use two features in our decision:
  - lightness: x_1
  - width: x_2
- Each fish image is now represented as a point (*feature vector*)

$$\mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

in a two-dimensional *feature space*.

An Example: Multiple Features

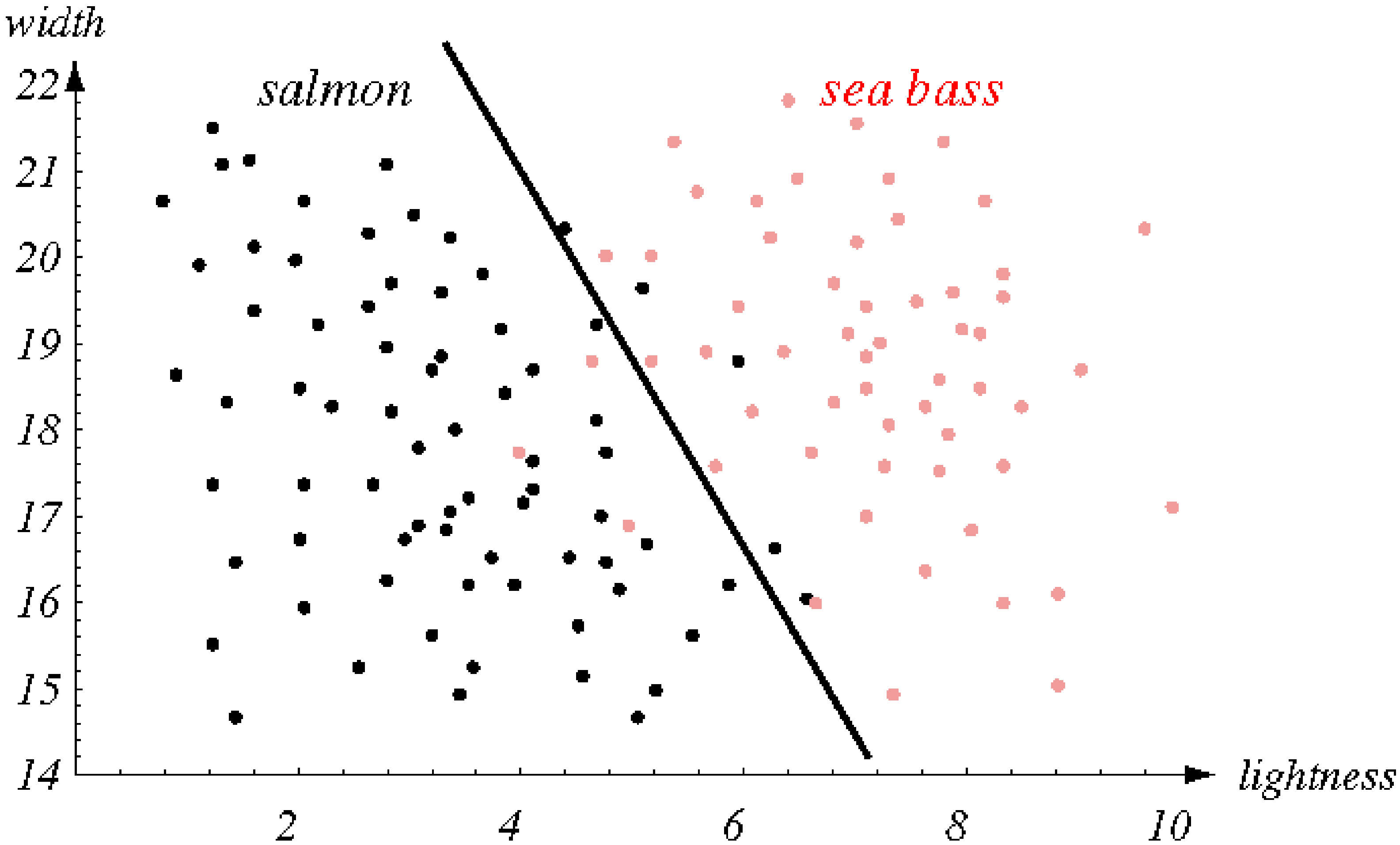

Figure 4: *Scatter plot* of lightness and width features for training samples. We can draw a *decision boundary* to divide the feature space into two regions. Does it look better than using only lightness?

An Example: Multiple Features

- 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?

An Example: Decision Boundaries

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

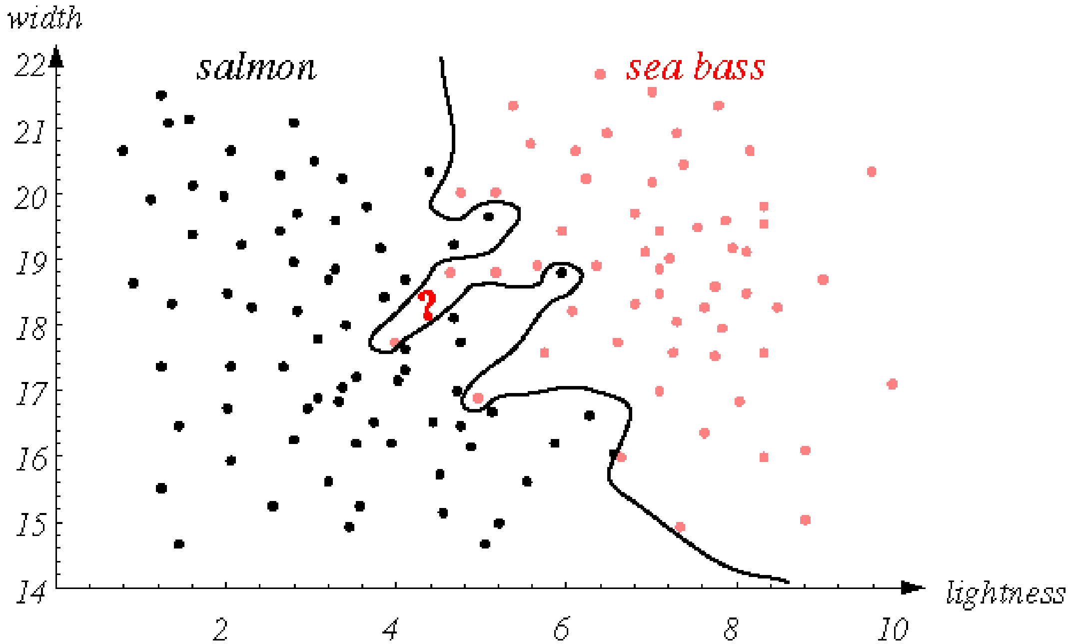

Figure 5: We may distinguish training samples perfectly but how can we predict how well we can *generalize* to unknown samples?

An Example: Decision Boundaries

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

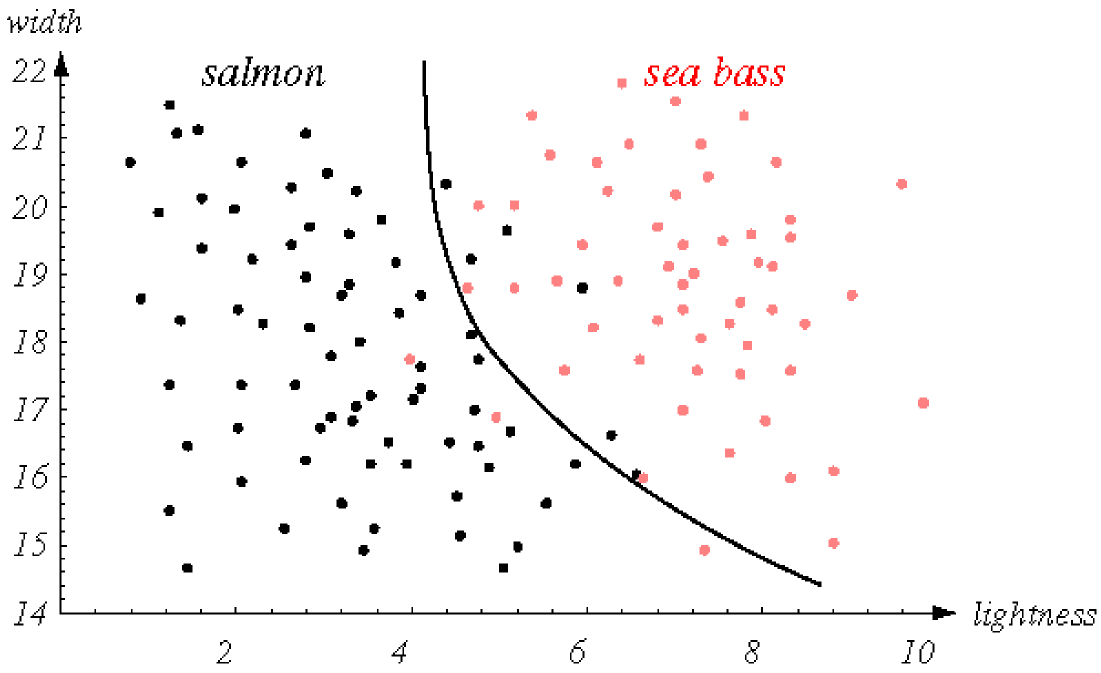

Figure 6: Different criteria lead to different decision boundaries.

Pattern Recognition Systems

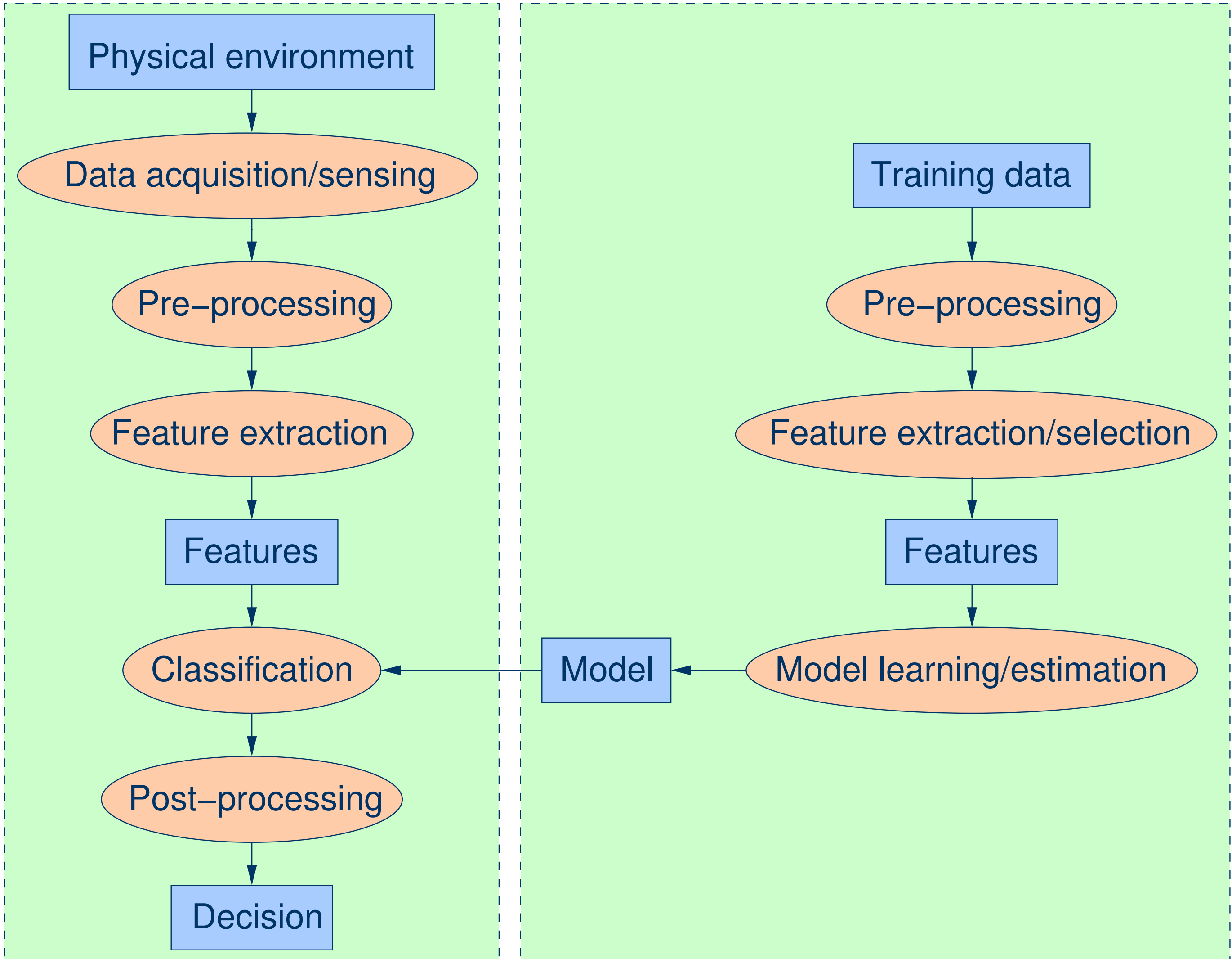

Figure 7: Object/process diagram of a pattern recognition system.

Pattern Recognition Systems

- Data acquisition and sensing:
  - Measurements of physical variables
  - Important issues: bandwidth, resolution, sensitivity, distortion, SNR, latency, etc.
- Pre-processing:
  - Removal of noise in data
  - Isolation of patterns of interest from the background
- Feature extraction:
  - Finding a new representation in terms of features

Pattern Recognition Systems

- Model learning and estimation:
  - Learning a mapping between features and pattern groups and categories
- Classification:
  - Using features and learned models to assign a pattern to a category
- Post-processing:
  - Evaluation of confidence in decisions
  - Exploitation of context to improve performance
  - Combination of experts

Pattern Recognition Applications

Table 1: Example pattern recognition applications.

Problem Domain	**Application**	**Input Pattern**	**Pattern Classes**
Document image analysis	Optical character recognition	Document image	Characters, words
Document classification	Internet search	Text document	Semantic categories
Document classification	Junk mail filtering	Email	Junk/non-junk
Multimedia database retrieval	Internet search	Video clip	Video genres
Speech recognition	Telephone directory assistance	Speech waveform	Spoken words
Natural language processing	Information extraction	Sentences	Parts of speech
Biometric recognition	Personal identification	Face, iris, fingerprint	Authorized users for access control
Medical	Diagnosis	Microscopic image	Cancerous/healthy cell
Military	Automatic target recognition	Optical or infrared image	Target type
Industrial automation	Printed circuit board inspection	Intensity or range image	Defective/non-defective product
Industrial automation	Fruit sorting	Images taken on a conveyor belt	Grade of quality
Remote sensing	Forecasting crop yield	Multispectral image	Land use categories
Bioinformatics	Sequence analysis	DNA sequence	Known types of genes
Data mining	Searching for meaningful patterns	Points in multidimensional space	Compact and well-separated clusters

The Design Cycle

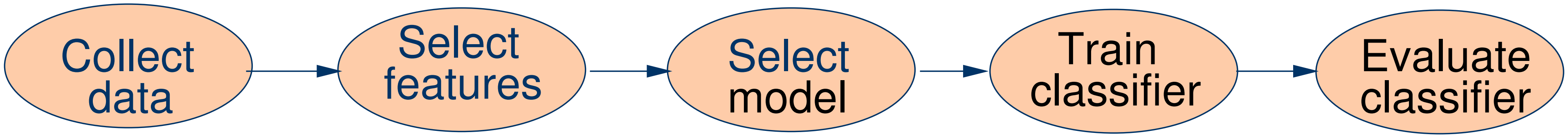

Figure 8: The design cycle.

- Data collection:
  - Collecting training and testing data
  - How can we know when we have adequately large and representative set of samples?

The Design Cycle

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

The Design Cycle

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

The Design Cycle

- Training:
  - How can we learn the rule from data?
  - 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: 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:
  - How can we estimate the performance with training samples?
  - How can we predict the performance with future data?
  - Problems of overfitting and generalization

Summary

- Pattern recognition techniques find applications in many areas: machine learning, statistics, mathematics, computer science, biology, etc.
- There are many sub-problems in the design process.
- Many of these problems can indeed be solved.
- More complex learning, searching and optimization algorithms are developed with advances in computer technology.
- There remain many fascinating unsolved problems.