Chapter 1
INTRODUCTION

Wei-Yang Lin

Department of Computer Science

& Information Engineering

mailto:wylin@cs.ccu.edu.tw

1.1 **Machine Perception**

1.2 An Example

1.3 Pattern Recognition Systems

1.4 The Design Cycle

1.5 Learning and Adaption

Summary by Chapters

1.1 Machine Perception

- It is natural that we would seek to build machines that can recognize patterns, such as speech, fingerprint, DNA sequence, and much more.
- It is clear that reliable pattern recognition by machine would be immensely useful.
- Moreover, in solving the problems required to build such systems, we gain deeper appreciation for pattern recognition systems in natural world, particularly in humans.

1.1 Machine Perception

1.2 **An Example**

1.3 Pattern Recognition Systems

1.4 The Design Cycle

1.5 Learning and Adaption

Summary by Chapters

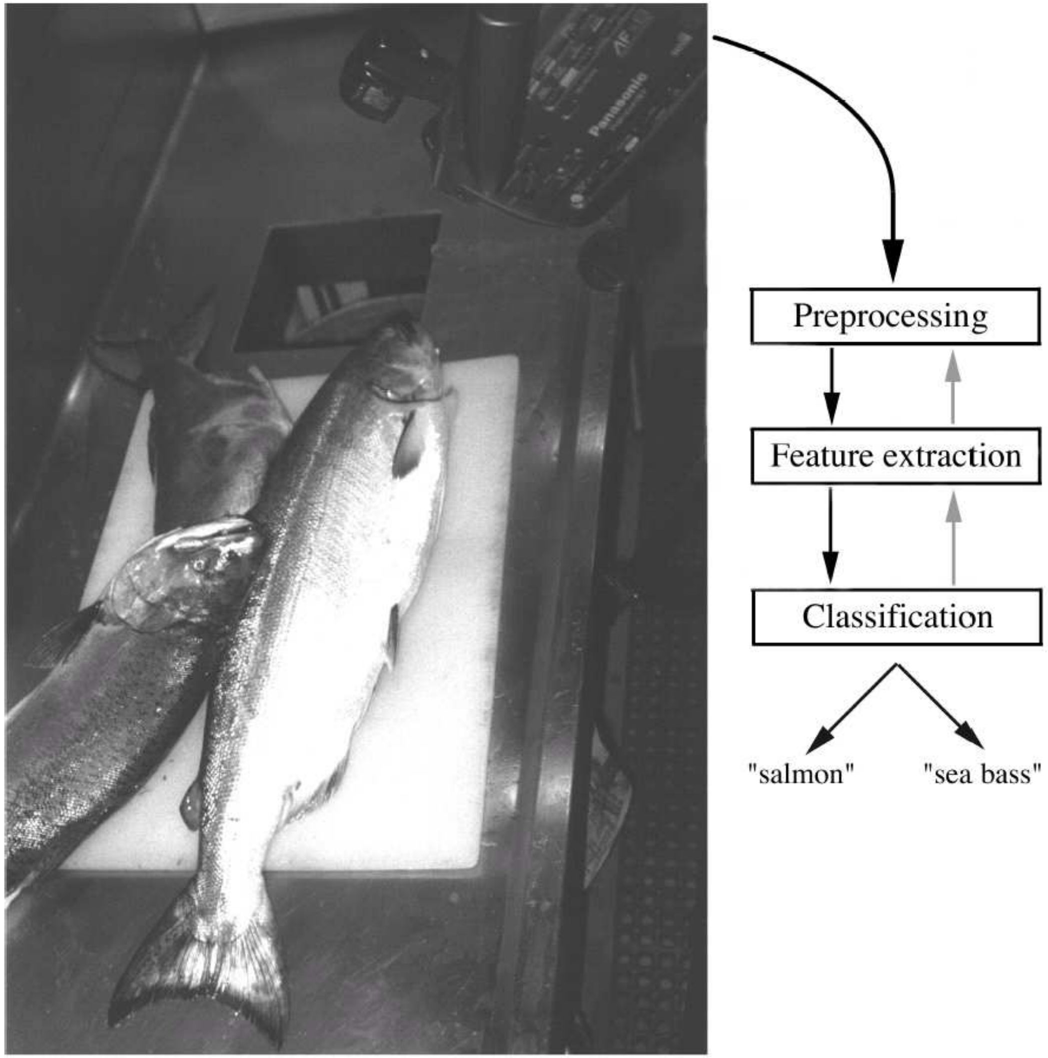
1.2 An Example

- To illustrate the types of problems involved, let us consider the following example.
- Suppose that a fish-packing plant wants to automate the process of sorting fish.
- As a pilot project it is decided to try to separate sea bass from salmon using optical sensing.

- We setup a camera, take some pictures, and begin to notice some physical differences between the two type of fish
  - Length
  - Lightness
  - Width
  - Number and shape of fins, etc.
- We also notice variations in the images – variations in lighting and position of fish.

- Main procedures
  - **Preprocessing – segmentation** from each other and background
  - **Feature extraction**
  - Classification and decision
- Tentative model
  - Sea bass have a typical length longer than that for salmon
- Design (Training) samples
  - For feature measurement and model identification

- Cost
  - Cost of actions (e.g., false positive and false negative)
  - Symmetry in the cost is often assumed, but not invariably
- Decision theory
  - Make a decision rule (i.e., set a **decision boundary**) to minimize a cost


Preprocessing
Feature extraction
Classification
"salmon"
"sea bass"

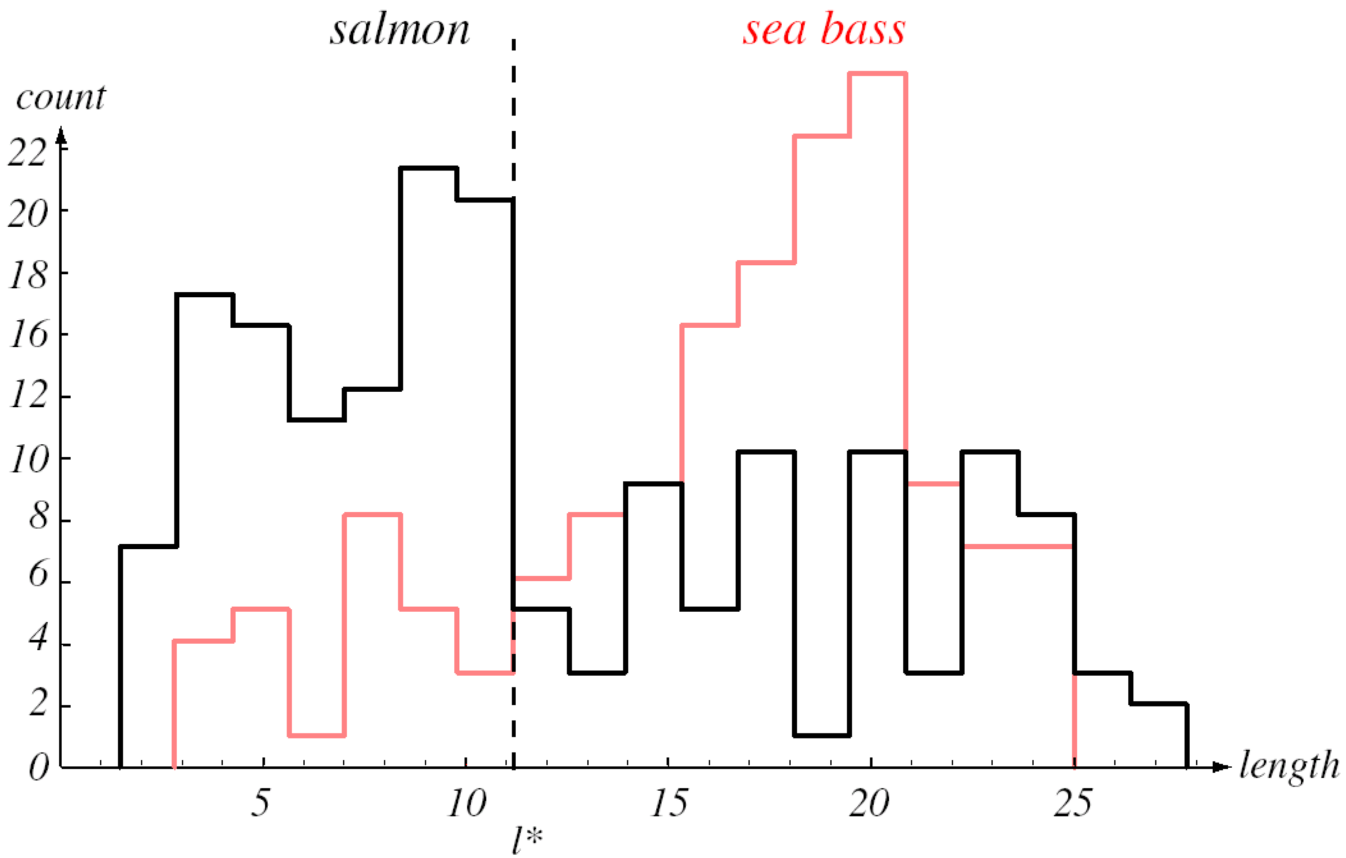


FIGURE 1.2. Histograms for the length feature for the two categories. No single threshold value of the length will serve to unambiguously discriminate between the two categories; using length alone, we will have some errors. The value marked *l** will lead to the smallest number of errors, on average.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

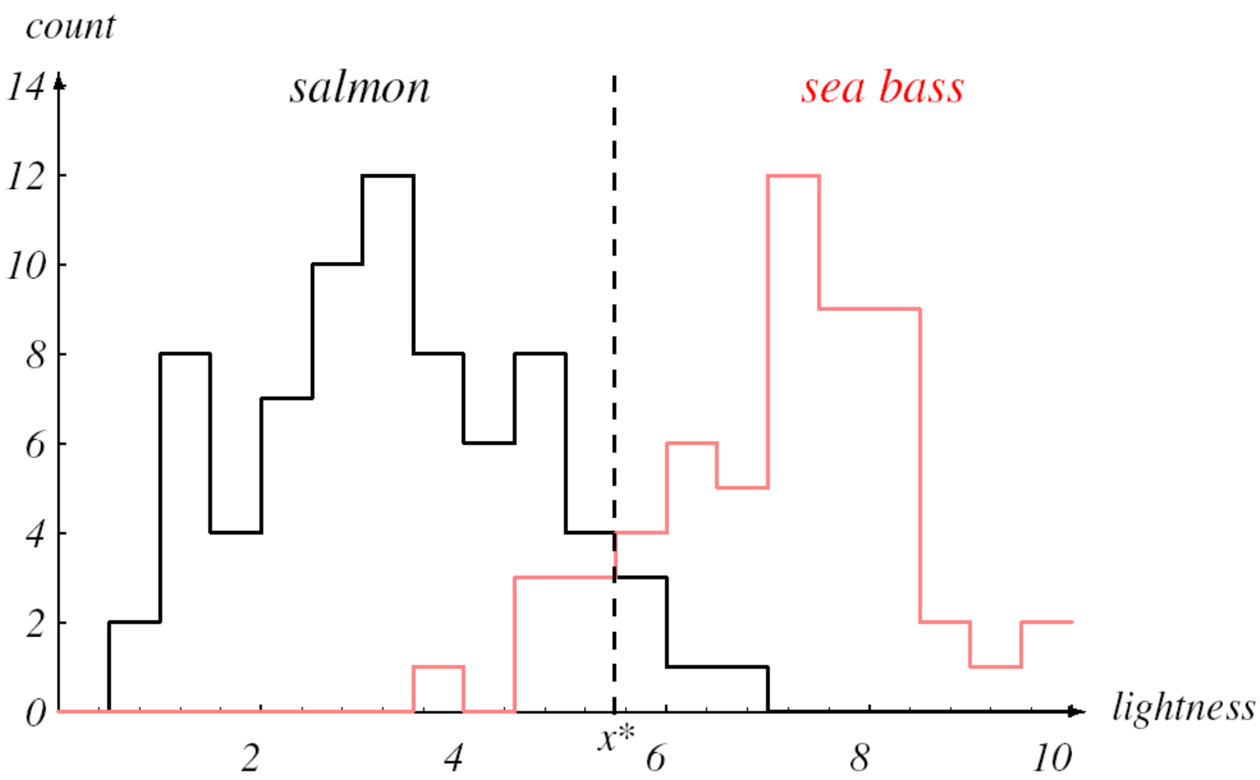


FIGURE 1.3. Histograms for the lightness feature for the two categories. No single threshold value x^* (decision boundary) will serve to unambiguously discriminate between the two categories; using lightness alone, we will have some errors. The value x^* marked will lead to the smallest number of errors,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

Classifier design

- Feature space
  - Feature vector

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

- Scattering plot for training samples
- Classifier: design of **decision boundary** on scattering plot
  - Partition the feature space into several regions.

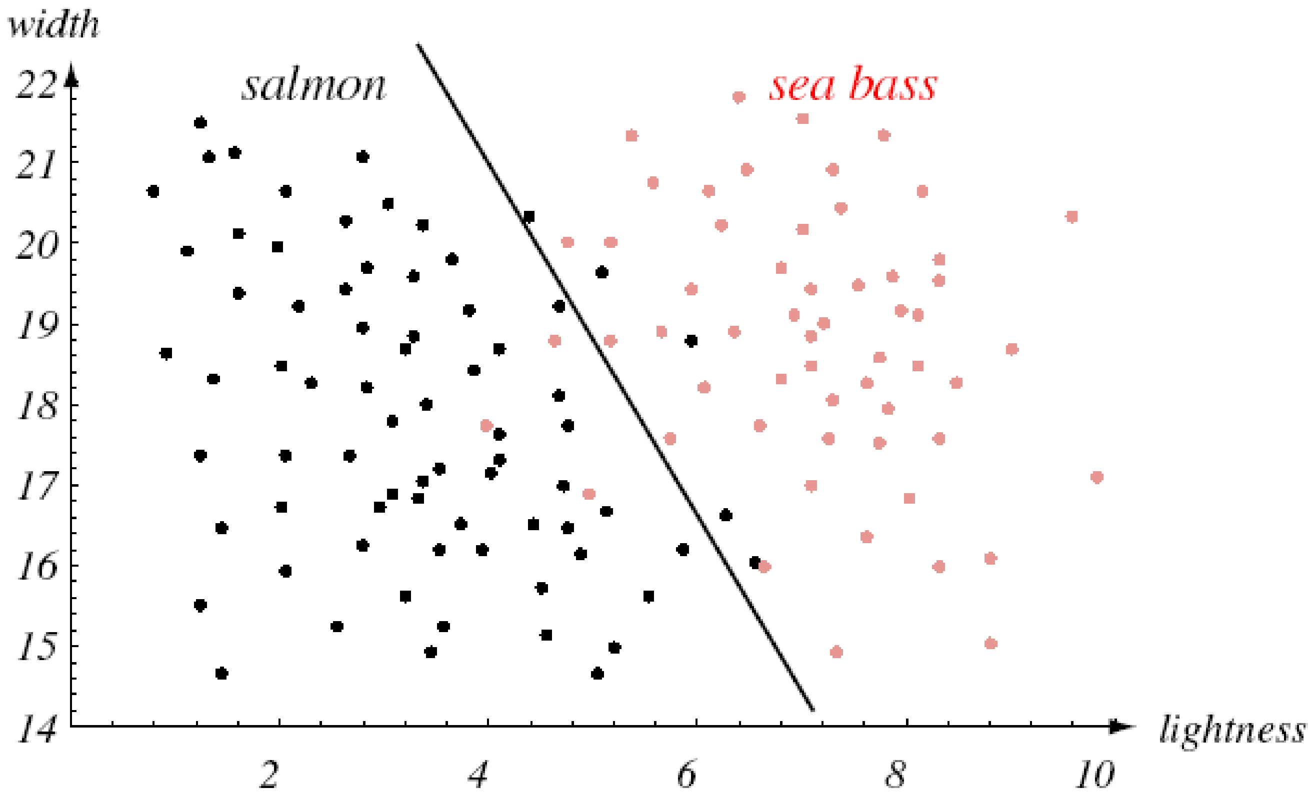


FIGURE 1.4. The two features of lightness and width for sea bass and salmon. The dark line could serve as a decision boundary of our classifier. Overall classification error on the data shown is lower than if we use only one feature as in Fig. 1.3, but there will still be some errors. From: Richard O.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

Generalization of a classifier

- Classifier that can classify novel patterns (or true model) very well, not just tuned to particular **training samples**.
  - Novel patterns: patterns not yet seen
- It would be better to estimate the true underlying characteristics (e.g., probability distributions) of each category by getting more samples.
- Better performance on novel patterns while poorer performance on the **training samples** V simpler **decision boundary** is preferred.

- Different decision tasks may require different features and yield different boundaries.
- A single general purpose artificial pattern recognition device is difficult to create.

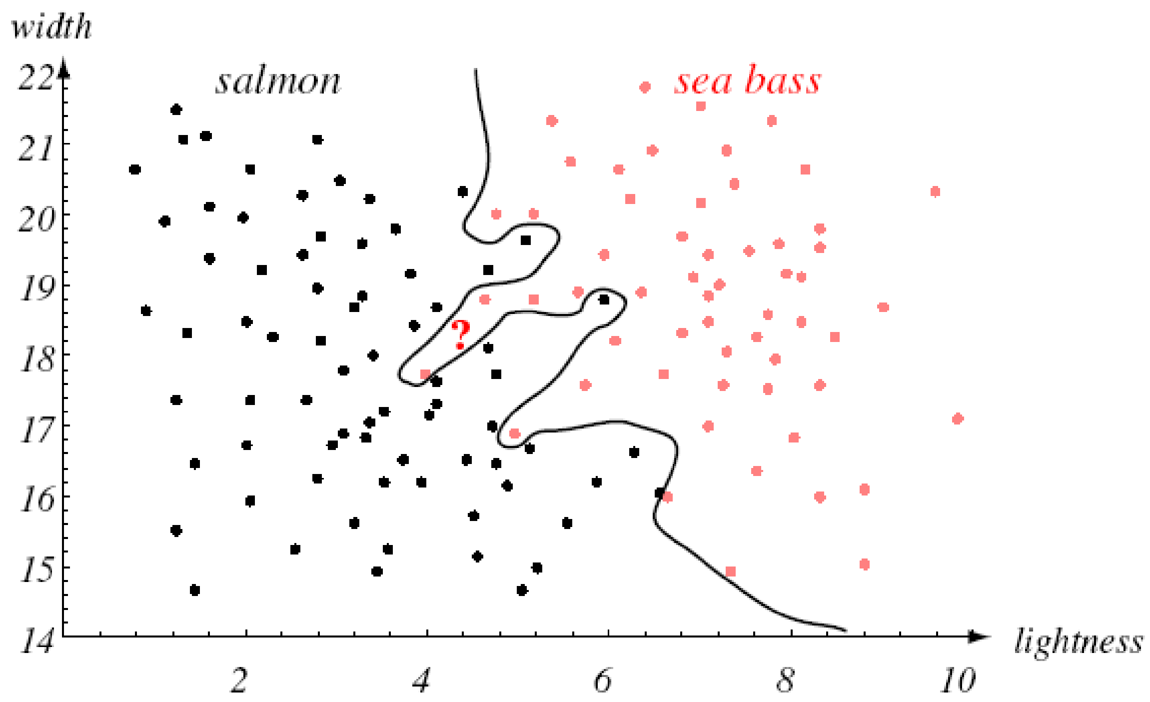


FIGURE 1.5. Overly complex models for the fish will lead to decision boundaries that are complicated. While such a decision may lead to perfect classification of our training samples, it would lead to poor performance on future patterns. The novel test point marked **?** is evidently most likely a salmon, 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.

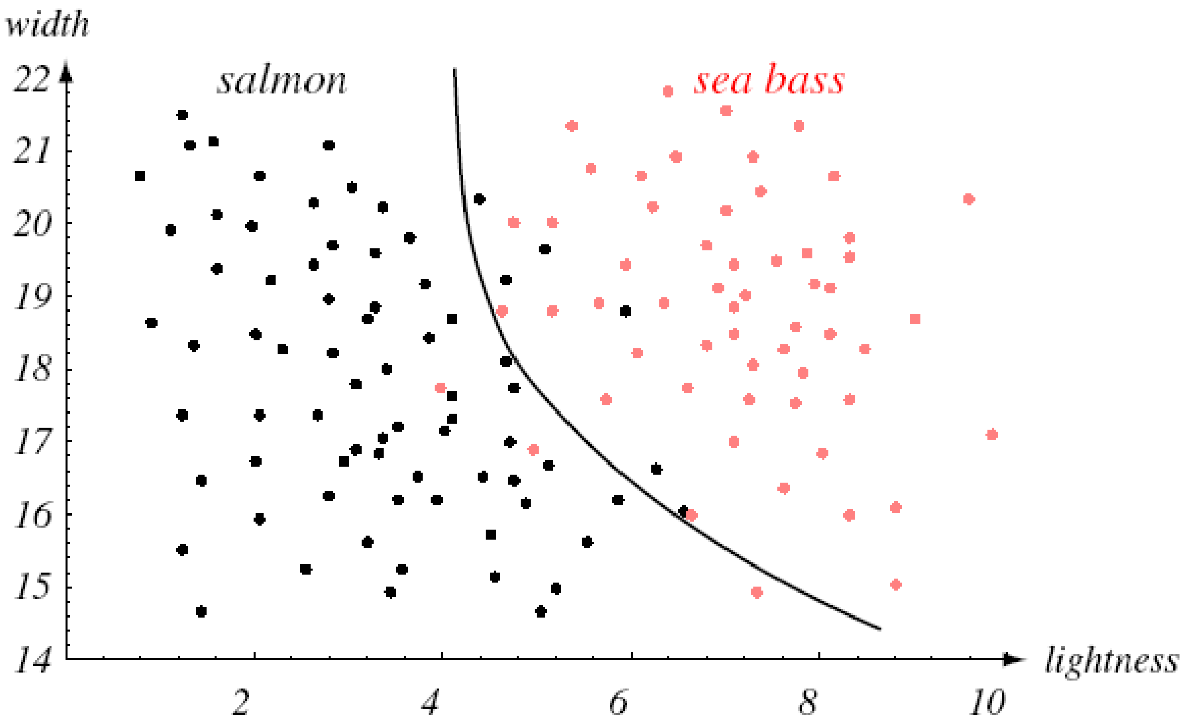


FIGURE 1.6. The decision boundary shown might represent the optimal tradeoff between performance on the training set and simplicity of classifier, thereby giving the highest accuracy on new patterns. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001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

Classification techniques

- Classification is to recover the model that generates the patterns.

- Different classification techniques are required for different types of models.

Categories of classification techniques

- Statistical pattern recognition
  - Assume or get statistical properties of the patterns (generally expressed in probability densities)
  - The model is often expressed as a single specific set of features.
- Neural pattern recognition
  - By neural network approach
  - Models are represented by weightings between neurons.

- Syntactic pattern recognition
  - Description of parts and their relations
  - A model consists of some set of crisp logical rules
  - Rules or grammars describe the decision

Analysis by synthesis

- The model of how each pattern is generated is available.
- Determine the underlying model of production from the input. That is, recover the generating parameters from the sensed patterns.
- The production representation may be the best representation for classification.
- For examples, stroke representation for OCR and vocal representation for speech recognition.

Related fields

- Image processing
- Speech processing
- Artificial intelligence
- Associate memory
- Neural and fuzzy

- Probability and Statistics (Statistical)
  - **Regression** (find functional description of data for new input prediction)
  - **Interpolation** (infer the function for intermediate ranges of input)
  - **Density estimation** (for ML and MAP classification)
- Formal language (Syntactic)
- Neural network architecture design and training (Neural)

1.1 Machine Perception

1.2 An Example

1.3 **Pattern Recognition Systems**

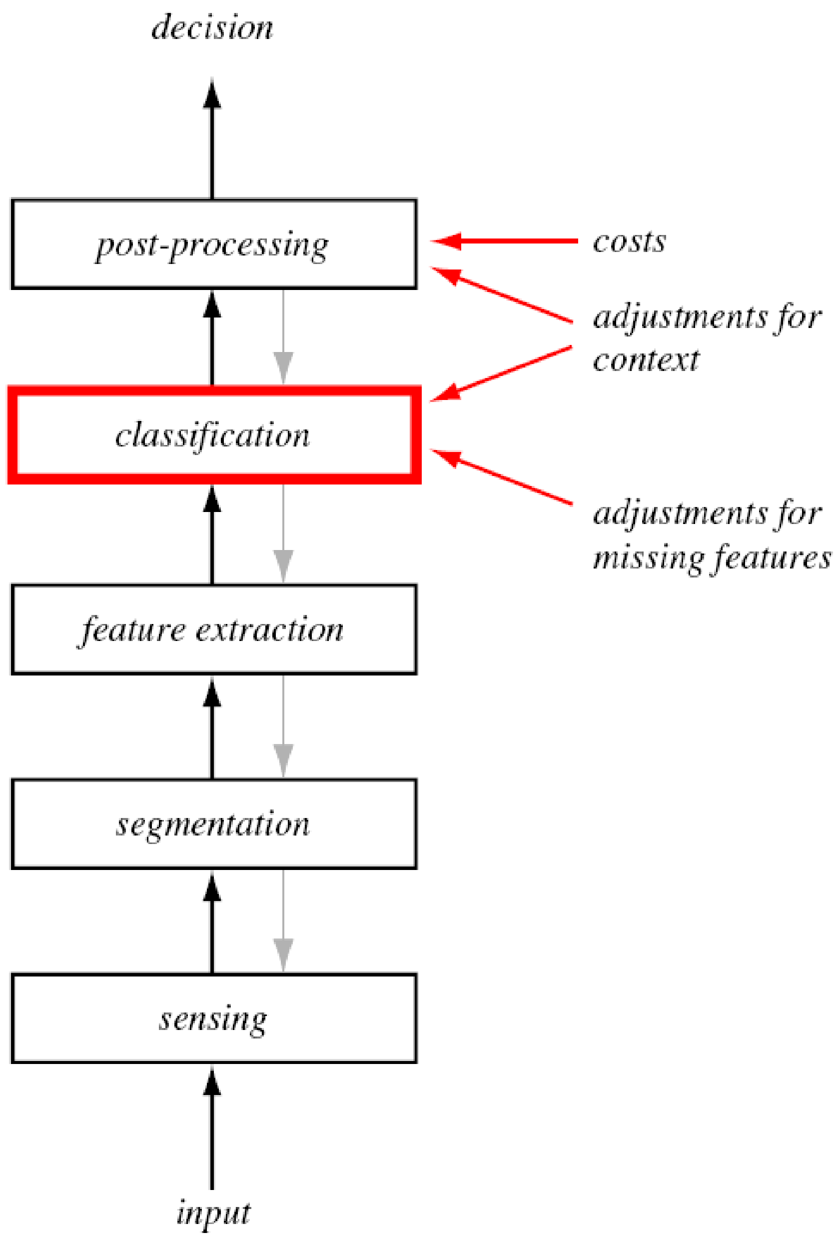
1.4 The Design Cycle

1.5 Learning and Adaption

Summary by Chapters

1.3 Pattern Recognition Systems

- Main components
  - Sensing (design of transducers)
  - Segmentation and grouping (into a composite object)
  - Feature extraction
  - Classification
  - Post-processing (considering the effects of context and the cost of errors)


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

Feature extraction

- Seek **distinguishing features** that are **invariant** to irrelevant transformations.
  - Distinguishing features
    * Feature values are similar in the same category and very different in different categories.
  - Irrelevant transformations
    * **Translation**, **rotation**, and **scale** (RST invariance, major concern)
    * **Occlusion**
    * **Projective distortion**

* **Nonrigid deformations**

- **Feature selection** (those are most effective)

Classification

- Assign an object to a category by using the feature vector.
- Difficulty of classification depends on the variability of the feature values in the same category relative to the difference between feature values in different categories.
- The variability of feature values in the same category may come from noise.

Post-processing

- Consider the cost of action
  - Minimize classification **error rate**
  - Minimize **risk** (total expected cost)
- Exploit **context** (input-dependent information) to improve system performance
  - E.g., use the context information for OCR or speech recognition

- **Multiple classifier** (different from multiple features)
  - Each classifier operates on different aspects of the input (e.g., speech recognition = acoustic recognition + lip reading)
  - Decision fusion

1.4 The Design Cycle

- The design of a pattern classification system usually consists of following steps: data collection, feature choice, model choice, training, and evaluation.
- In the following subsections, we provides an overview of this design cycle.

1.4.1 Data Collection

- Data collection can account for surprising large part of the cost of developing a pattern recognition system.
- It may be possible to perform a preliminary feasibility study with a small set of data.
- Much more data will usually be needed to ensure good performance in real world applications.

1.4.2 Feature Choice

- The choice of features is a critical step and depends on the characteristics of the problem domain.
- Here, **prior knowledge** plays a major role.
- Incorporating prior knowledge can be subtle and difficult.
- We would like to find features that are simple to extract, invariant to irrelevant transformations, insensitive to noise, and useful for discriminating patterns from different categories.

1.4.3 Model Choice

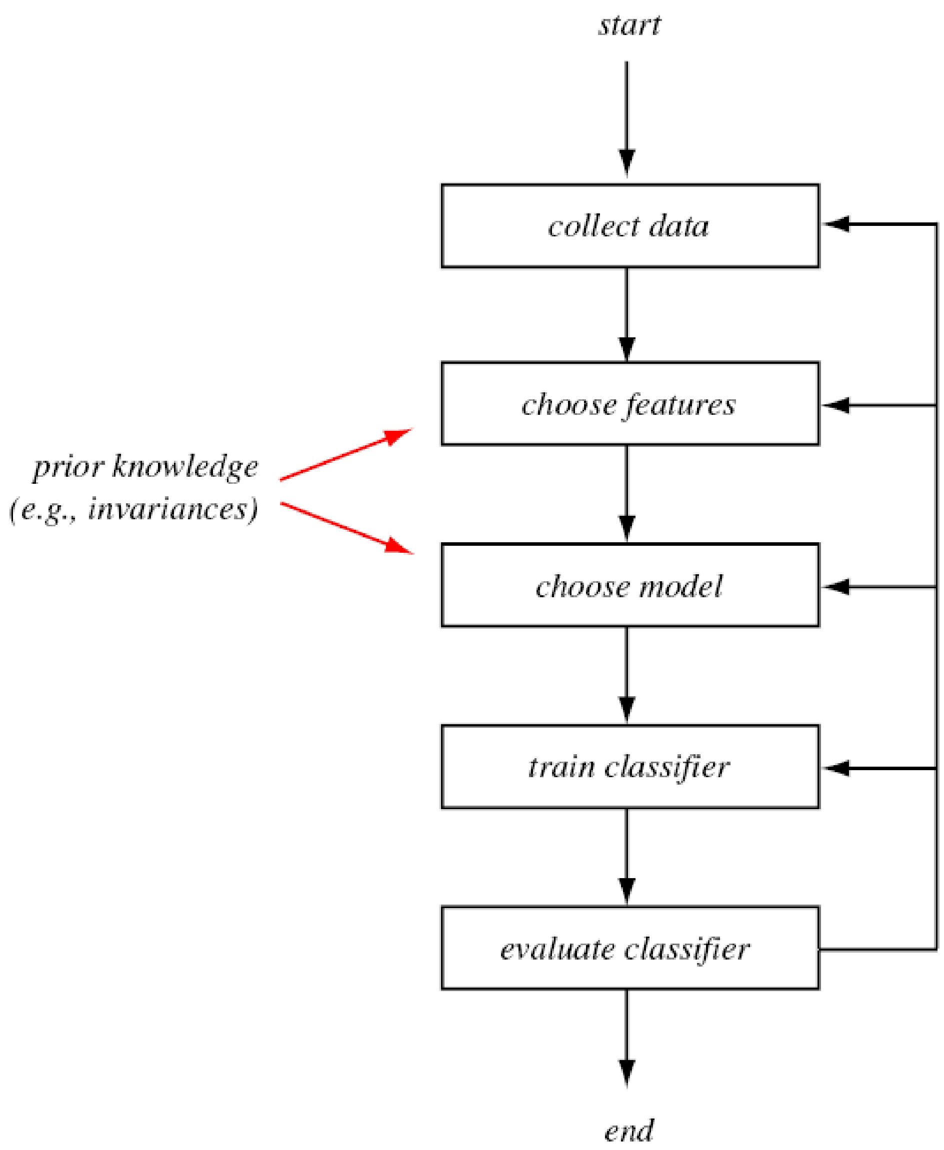
- We might have been unsatisfied with the performance of our fish classifier.
- One could choose an entirely different class of model, for instance one based on the number and position of fins, the weight, shape of mouth, and so on.

1.4.4 Training

- The process of using data to determine a classifier is referred to as training a classifier.
- Much of this book is concerned with many different procedures for training classifiers.
- Most of the effective methods for developing classifiers involve learning from sample patterns.

1.4.5 Evaluation

- Evaluation is important both to measure the performance and to identify the needs for improvements.
- While an overly complex system may allow perfect classification of the training samples, it is unlikely to perform well on new patterns.
- This situation is known as **over-fitting**.


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

1.5 Learning and Adaptation

- Because practical pattern classification problems are so hard that we cannot have the best classifier ahead of time, we shall spend the great majority of time here considering learning.
- Learning refers to some form of algorithm for reducing the error on a set of training samples.
- Learning comes in several general forms.

1.5.1 Supervised Learning

- A teacher provides a label for each training sample.

1.5.2 Unsupervised Learning

- There is no explicit teacher. A system forms clusters of training samples.

1.5.3 Reinforcement Learning

- The most typical way to train a classifier is to present an input , compute its tentative category label, and use the known target category label to improve the classifier.
- For example, in optical characteristic recognition, the input is an image of a character. The tentative output of the classifier might be “R” and the desired output is “B”.

1.1 Machine Perception

1.2 An Example

1.3 Pattern Recognition Systems

1.4 The Design Cycle

1.5 Learning and Adaption

Summary by Chapters

Summary by Chapters

- (Ch 1) Introduction
- (Ch 2) Bayesian Decision Theory
  - The full probability structure underlying the categories is known.
- (Ch 3) Maximum-Likelihood and Bayesian Parameter Estimation
  - The general-form probability structure underlying categories is known, but certain parameters are not known.
  - Seek to determine these parameters to attain the best categorization

- (Ch 4) Nonparametric Techniques
  - There is no prior parameterized knowledge about the underlying probability structure.
  - Classification is based on the training samples alone.
  - E.g., nearest-neighbor (NN) algorithm
- (Ch 5) Linear Discriminant Functions
  - The simplest class of training rules for discriminating two categories.

- (Ch 6) Multilayer Neural Networks
  - Neural network is basically a nonlinear classifier
  - Extend some of the ideas of linear discriminants for training multilayer neural networks
- (Ch 7) Stochastic Methods (Option)
  - e.g., simulated annealing, Boltzmann learning algorithm
- (Ch 8) Nonmetric Methods
  - moves beyond models that are statistical in nature to the ones described by logical rules (syntactic-based methods based on grammars)

- (Ch 9) Algorithm-Independent Machine Learning (Option)
  - essential but difficult to understand
- (Ch 10) Unsupervised Learning and Clustering
  - Input training patterns are not labeled
  - Determine the cluster structure