

Chapter 1

Introduction

Thus all human knowledge begins with intuitions, then goes to concepts, and is completed in ideas.

IMMANUEL KANT

1.1 Recognizing the pattern

We recognize many patterns¹ while observing the world. Even in a country never visited before, we recognize buildings, streets, trees, flowers or animals. There are pattern characteristics, learned before, that can be applied in a new environment. Sometimes, we encounter a place with objects that are alien to us, e.g. a garden with an unknown flower species or a market place with strange types of fish. How do we learn these patterns so that this place will look more familiar on our next visit?

If we take the time, we are able to learn some patterns by ourselves. If somebody shows us around, points out and explains what is what, we may learn faster and group the observations according to the underlying concepts. What is the first step in this categorization process? Which principle is used in the observations to constitute the first grouping? Are these descriptive features like color, shape or weight? Or is it our basic perception that some objects are somehow different and others are similar? The ability to observe the differences and the similarities between objects seems to be very basic. Discriminating features can be found once we are familiar with similarities.

This book is written from the perspective that the most primary observation we can make when studying a group of objects or phenomena is that some are dissimilar and others are similar. From this starting point, we aim to define a theory for learning and recognizing patterns by automatic means: sensors and computers that try to imitate the human ability

¹We will use the word ‘pattern’ exclusively to refer to quantitative/qualitative characteristics between objects. In the literature, however, ‘pattern’ is also used to refer to a single object for which such characteristics are studied. We will avoid this usage here.

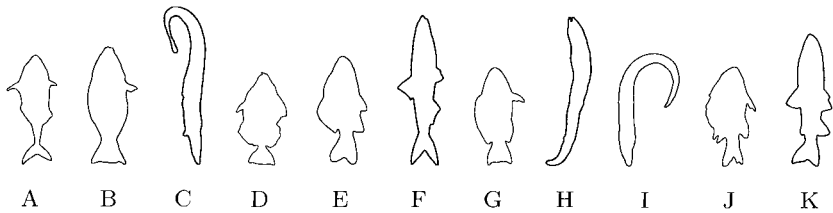


Figure 1.1 Fish contours.

of pattern recognition. We will develop a framework in which the initial representation of objects is based on dissimilarities, assuming that a human expert can make explicit how to measure them from sensor data. We will develop techniques to generalize from dissimilarity-based representations of sets of objects to the concepts of the groups or classes that can be distinguished.

This is in contrast to the traditional paradigm in automatic pattern recognition that starts from a set of numerical features. As stated above, features are defined after dissimilarities have been observed. In a feature-based approach, more human expertise may be included. Consequently, if this is done properly, a feature description should be preferred. If, however, this expertise is not available, then dissimilarities have to be preferred over arbitrarily selected features.

There are already many applied studies in this area based on dissimilarities. They lack a foundation and, consequently, consistent ways for building a generalization. This book will contribute to these two. In the first part, Chapters 2 to 5, concepts and theory are developed for dissimilarity-based pattern recognition. In the second part, Chapters 6 to 10, they are used for analyzing dissimilarity data and for finding and classifying patterns. In this chapter, we will first introduce our concepts in an intuitive way.

1.2 Dissimilarities for representation

Human perception and inference skills allow us to recognize the common characteristics of a collection of objects. It is, however, difficult to formalize such observations. Imagine, for instance, the set of fish shape contours [Fish contours, site] as presented in Fig. 1.1. Is it possible to define a simple rule that divides them into two or three groups? If we look at the contours, we find that some of the fish are rather long without characteristic fins (shape C, H and I), whereas others have distinctive tails as well as fins, say a group

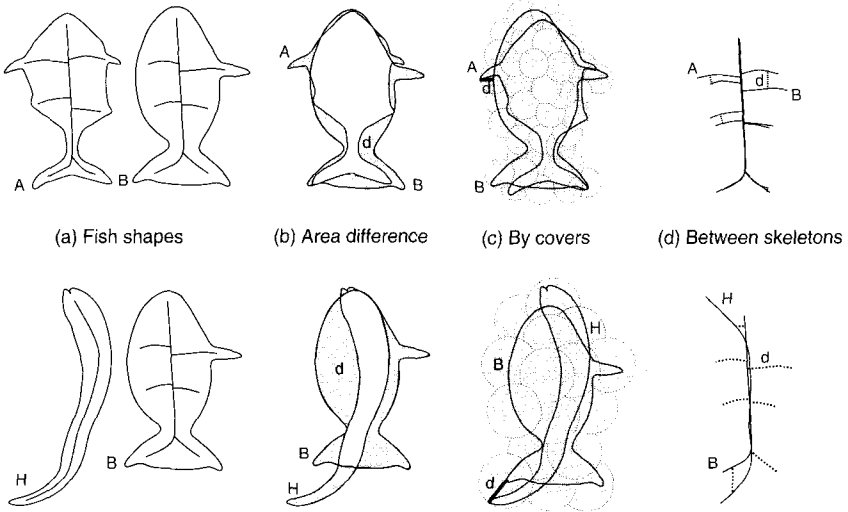


Figure 1.2 Various dissimilarity measures can be constructed for matching two fish shapes. (b) Area difference: the area of non-overlapping parts is computed. To avoid scale dependency, the measured difference can be expressed relative to the sum of the areas of the shapes. (c) Measure by covers: one shape is covered by identical balls (such that the ball centers belong to it), taking care that the other shape is covered as well. The shapes are exchanged and the radius of the minimal ball is the sought distance. In both cases above, B is covered such that either A or H are also covered. (d) Measure between skeletons: two shape skeletons are compared by summing up the differences between corresponding parts, weighting missing correspondences more heavily.

of fin-type fish. Judging shapes F and K in the context of all fish shapes presented here, they could be found similar to other fin-type fish: A, B, D, E, G and J. By visual inspection, they do not really appear to be alike, as they seem to be thinner and somewhat larger. If the examples of C, H and I had been absent, the differences between F and K and other fin-type fish would have been more pronounced. Furthermore, shape A could be considered similar to F and K, but also different due to the position and shape of its tail and fins.

This simple example shows that without any extra knowledge or a clear context, one cannot claim that the identification of two groups is better than the identification of three groups. This decision relies on a free interpretation of what makes objects similar to be considered as a group.

For the purpose of automatic grouping or identification, it is difficult to determine proper features, i.e. mathematically encoded particular properties of the shapes that would precisely discriminate between different fish

and at the same time emphasize the similarity between resembling examples. An alternative is to compare the shapes by matching them as well as possible and determining the remaining differences. Such a match is found with respect to a specified measure of dissimilarity. This measure should take on small values for objects that are alike and large values for distinct objects.

There are many ways of comparing two objects, and hence there are many dissimilarity measures. In general, the suitability of a measure depends on the problem at hand and should rely on additional knowledge one has about this particular problem. Example measures are presented in Fig. 1.2, where two fish shapes are compared. Here, the dissimilarity between two similar fish, A and B, is much smaller than between two different fish, B and H. Which to choose depends on expert knowledge or problem characteristics. If there is no clear preference for one measure over the other, a number of measures can be studied and combined. This may be beneficial, especially when different measures focus on different aspects of patterns.

1.3 Learning from examples

The question how to extract essential knowledge and represent it in a formal way such that a machine can 'learn' a concept of a class, identify objects or discriminate between them, has intrigued and provoked many researchers. The growing interest inherently led to the establishment of the areas of pattern recognition, machine learning and artificial intelligence. Researchers in these disciplines try to find ways to mimic the human capacity of using knowledge in an intelligent way. In particular, they try to provide mathematical foundations and develop models and methods that automate the process of recognition by learning from a set of examples. This attempt is inspired by the human ability to recognize for example what a tree is, given just a few examples of trees. The idea is that a few examples of objects (and possible relations between them) might be sufficient for extracting suitable knowledge to characterize their class.

After years of research, some practical problems can now be successfully treated in industrial processing tasks such as automatic recognition of damaged products on a conveyor belt, or to speed up data-handling procedures, or the automatic person identification by fingerprints. The algorithms developed so far are very task specific and, in general, they are

still far from reaching the human recognition performance. Although the models designed are becoming more and more complex, it seems that to take them a step further, one will need to analyze their basic underlying assumptions. An understanding of the recognition process is needed; not only the learning approaches (inductive or deductive principles) must be understood, but mainly the basic notions of class, measurement process and the representation of objects derived from these. The formalized representation of objects (usually in mathematical terms) and the definition of classes determine how the act of learning should be modeled. While many researchers are concerned with various algorithmic procedures, we would like to focus on *the issue of representation*. This work is devoted to particular representations, namely dissimilarity representations. Below and in the subsequent sections, we will give some insight into the nature of basic problems in pattern recognition and machine learning and motivate the use of dissimilarity representations.

While dealing with entities to be compared, we will always refer to them as to objects, elements or instances, regardless of whether they are real or abstract. For instance, images, textures and shapes are called objects in the same way as apples and chairs. An appropriate representation of objects is based on data. These are usually obtained by a measurement device and encoded in a numerical way or given by a set of observations or dependencies, presented in a structural form, e.g. a relational graph. It is assumed that objects can, in general, be grouped together. Our aim then is to identify a number of groups (clusters) whose existence supports an understanding of not only the data, but also the problem itself. Such a process is often used to order information and to find suitable or efficient descriptions of the data.

The challenge of automatic object recognition is to develop computer methods which learn to identify whether an object belongs to a specific class or learn to distinguish between a number of classes. Typically, the system is first presented with a set of labeled objects, the training set, in some convenient representation. Learning consists of finding the class descriptions such that the system can correctly classify novel examples. In practice, the entire system is trained such that the given examples are (mostly) assigned to the correct class. The underlying assumption is that the training examples are representative and sufficient for the problem at hand. This implies that the system can extrapolate well to previously unseen examples, that is, it can *generalize* well.

There are two principal directions in pattern recognition, statistical and

Table 1.1 Basic differences between statistical and structural Pattern Recognition [Nadler and Smith, 1993]. Distances are a common factor used for discrimination in both approaches.

Properties	Statistical	Structural
Foundation	Well-developed mathematical theory of vector spaces	Intuitively appealing: human cognition or perception
Approach	Quantitative	Qualitative: structural/syntactic
Descriptors	Numerical features: vectors of a fixed length	Morphological primitives of a variable size
Syntax	Element position in a vector	Encoding process of primitives
Noise	Easily encoded	Needs regular structures
Learning	Vector-based methods	Graphs, decisions trees, grammars
Dissimilarity	Metric, often Euclidean	Defined in a matching process
Discrimination	Relies on distances or inner products in a vector space	Grammars recognize valid objects; distances often used
Class overlap	Due to improper features and probabilistic models	Due to improper primitives leading to ambiguity in the description

structural (or syntactic) pattern recognition [Jain *et al.*, 2000; Nadler and Smith, 1993; Bunke and Sanfeliu, 1990]. The basic differences are summarized in Table 1.1. Both approaches use features to describe objects, but these features are defined differently. In general, features are functions of (possibly preprocessed) measurements performed on objects, e.g. particular groups of bits in a binary image summarizing it in a discriminative way. The statistical, decision-theoretical approach is (usually) metric and quantitative, while the structural approach is qualitative [Bunke and Sanfeliu, 1990; Nadler and Smith, 1993]. This means that in the statistical approach, features are encoded as purely numerical variables. Together, they constitute a feature vector space, usually Euclidean, in which each object is represented as a point² of feature values. Learning is then inherently restricted to the mathematical methods that one can apply in a vector space, equipped with additional algebraic structures of an inner product, norm and the distance. In contrast, the structural approach tries to describe the structure of objects that intuitively reflects the human perception [Edelman *et al.*, 1998; Edelman, 1999]. The features become primitives (subpatterns), fundamental structural elements, like strokes, corners or other morphological elements.

²In this book, the words ‘points’ and ‘vectors’ are used interchangeably. In the rigorous mathematical sense, points and vectors are not the same, as points are defined by fixed sets of coordinates in a vector space, while vectors are defined by differences between points. In statistical pattern recognition, objects are represented as points in a vector space, but for the sake of convenience, they are also treated as vectors, as only then they define the operations of vector addition and multiplication by a scalar.

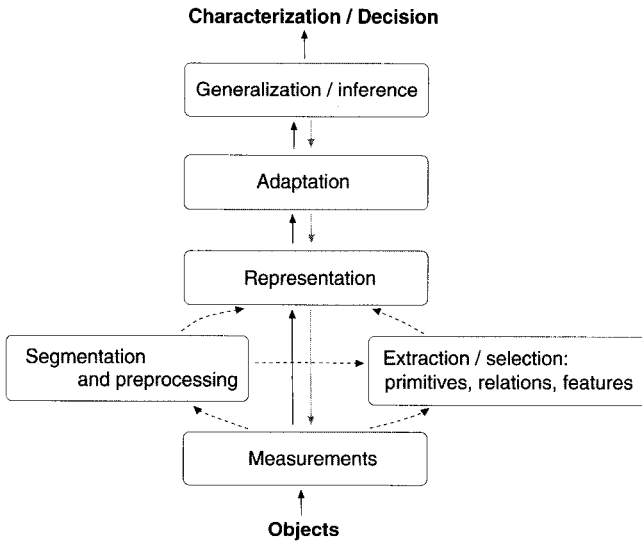


Figure 1.3 Components of a general pattern recognition system. A representation is either a numerical description of objects and/or their relations (statistical pattern recognition) or their syntactical encoding by set of primitives together with a set of operations on objects (structural pattern recognition). Adaptation relies on a suitable change (simplification or enrichment) of a representation, e.g. by a reduction of the number of features, relations or primitives describing objects, or some nonlinear transformation of the features, to enhance the class or cluster descriptions. Generalization is a process of determining a statistical function which finds clusters, builds a class descriptor or constructs a classifier (decision function). Inference describes the process of a syntax analysis, resulting in a (stochastic) grammar. Characterization reflects the final decision (class label) or the data description (determined clusters). Arrows illustrate that a building of the complete system may not be sequential.

Next, the primitives are encoded as syntactic units from which objects are constructed. As a result, objects are represented by a set of primitives with specified syntactic operations. For instance, if the operation of concatenation is used, objects are described by strings of (concatenated) primitives. The strength of the statistical approach relies on well-developed concepts and learning techniques, while in the structural approach, it is much easier to encode existing knowledge on the objects.

A general description of a pattern recognition system is illustrated in Fig. 1.3; see also [Duin *et al.*, 2002] for a more elaborate discussion and [Nadler and Smith, 1993] for an engineering approach. The description starts from a set of measurements performed on a set of objects. These measurements may be subjected to various operations in order to extract the

essential information (e.g. to segment an object from the image background and identify a number of characteristic subpatterns), leading to some numerical or structural representation. Such a representation has evolved from an initial description, derived from the original measurements. Usually, it is not directly the most appropriate one for realizing the task, such as identification or classification. It may be adapted by suitable transformations, e.g. a (nonlinear) rescaling of numerical features or an extension and redefinition of primitives. Then, in the generalization/inference stage, a classifier/identifier is trained, or a grammar³ is determined. These processes should include a careful treatment of unbalanced classes, non-representative data, handling of missing values, a rejection option, combining of information and combining of classifiers and a final evaluation. In the last stage, a class is assigned or the data are characterized (e.g. in terms of clusters and their relations). The design of a complete pattern recognition system may require repetition of some stages to find a satisfactory trade-off between the final recognition accuracy or data description and the computational and storage resources required.

Although this research is grounded in statistical pattern recognition, we recognize the necessity of combining numerical and structural information. Dissimilarity measures as the common factor used for discrimination, Table 1.1, seems to be the natural bridge between these two types of information. The integration is realized by a representation. A general discussion on the issue of representation can be found in [Duin *et al.*, 2004a].

1.4 Motivation of the use of dissimilarity representations

The notion of similarity plays a pivotal role in class formation, since it might be seen as a natural link between observations on objects on the one hand and a judgment on their shared properties on the other. In essence, similar objects can be grouped together to form a class, and consequently *a class is a set of similar objects*. However, there is no such thing as a general object similarity that can be universally measured or applied. A comparison of two objects is always with respect to a frame of reference, i.e. a particular point of view, a context, basic characteristics, a type of domain, or attributes considered (see also Fig. 1.1). This means that background information, or

³Primitives are interpreted as syntactic units or symbols. A grammar is a set of rules of syntax that enables the generation of sentences (structures) from the given symbols (units).

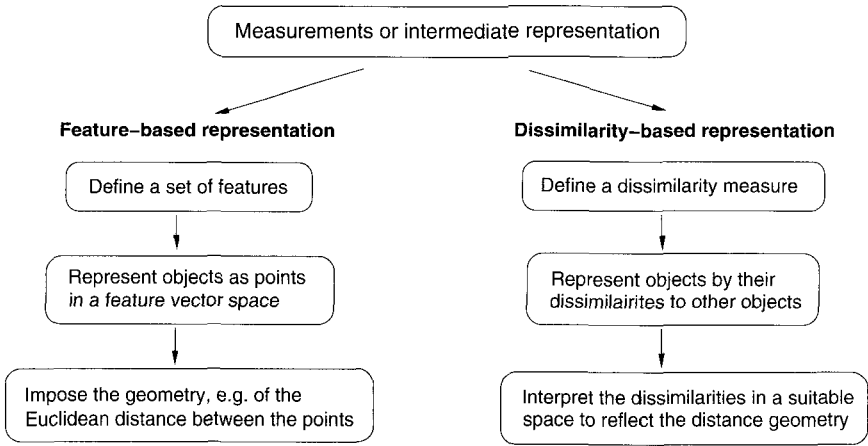


Figure 1.4 The difference with respect to the geometry between the traditional feature-based (absolute) representations and dissimilarity-based (relative) representations.

the existence of other classes, will influence the way objects are compared. For instance, two brothers may not appear to resemble each other. However, they may appear much more alike if compared in the presence of their parents. The degree of similarity between two objects should be determined *relative* to a given context or a procedure.

Any measurement of similarity of objects will be based on certain assumptions concerning the properties of their relation. Such assumptions come from some model. Similarity can be modeled by a measure of similarity or dissimilarity. These are intimately connected; a small dissimilarity and a large similarity both imply a close resemblance of objects. There exist ways of changing a similarity value into a dissimilarity value and vice versa, but the interpretation of the measure might be affected. In this work, we mostly concentrate on dissimilarities, which by their construction, focus on the class and object differences. The choice for dissimilarities is supported by the fact that they can be interpreted as distances in suitable vector spaces, and in many cases, they may be more intuitively appealing.

In statistical pattern recognition, objects are usually encoded by feature values. A feature is a conjunction of measured values for a particular attribute. For instance, if weight is an attribute for the class of apples, then a feature consists of the measured weights for a number of apples.

For a set T of N objects, a feature-based representation relying on a set \mathcal{F} of m features is then encoded as an $N \times m$ matrix $A(T, \mathcal{F})$, where each

row is a vector describing the feature values for a particular object. Features \mathcal{F} are usually interpreted in a Euclidean vector space equipped with the Euclidean metric. This is motivated by the algebraic structure (defined by operations on vectors) being consistent with the geometric (topological) structure defined by the Euclidean distance (which is then defined by the norm). Then all traditional mathematical concepts and methods, such as continuity, convergence or differentiation are applicable. The continuity of algebraic operations ensures that the local geometry (defined by the Euclidean distance) is preserved throughout the space [Munkres, 2000; Köthe, 1969]. Discrimination techniques operating in vector spaces make use of their homogeneity and other properties. Consequently, such spaces require that up to scaling all the features are treated in the same way. Moreover, there is no possibility to relate the learning to the geometry defined between the raw representations of the training examples. The geometry is simply *imposed* beforehand by the nature of the Euclidean distance between (reduced) descriptions of objects, i.e. between vectors in a Euclidean space; see also Fig. 1.4. The existence of a well-established theory for Euclidean metric spaces made researchers place the learning paradigm in that context. However, the severe restrictions of such spaces simply do not allow discovery of structures richer than affine subspaces. From this point of view, the act of learning is very limited.

We argue here that the notion of proximity (similarity or dissimilarity) is more fundamental than that of a feature or a class. According to an intuitive definition of a class as a set of similar objects, proximity plays a crucial role for its constitution, and not features, which may (or may not) come later. From this point of view, features might be a superfluous step in the description of a class. Surely, proximity can be specified by features, such as their weighted linear combination, but the features should be meaningful with respect to the proximity. In other words, the chosen combination of features should reflect the (natural) proximity between the objects. On the other hand, proximity can be directly derived from raw or pre-processed measurements like images or spectra. Moreover, in the case of symbolic objects, graphs or grammars, the determination of numerical features might be an intractable problem, while proximity may be easier to define. This emphasizes that a class of objects is represented by individual examples which are judged to be similar according to a specified measure. A dissimilarity representation of objects is then based on pairwise comparisons and is expressed e.g. as an $N \times N$ dissimilarity matrix $D(T, T)$. Each entry of D is a dissimilarity value computed between pairs of objects; see

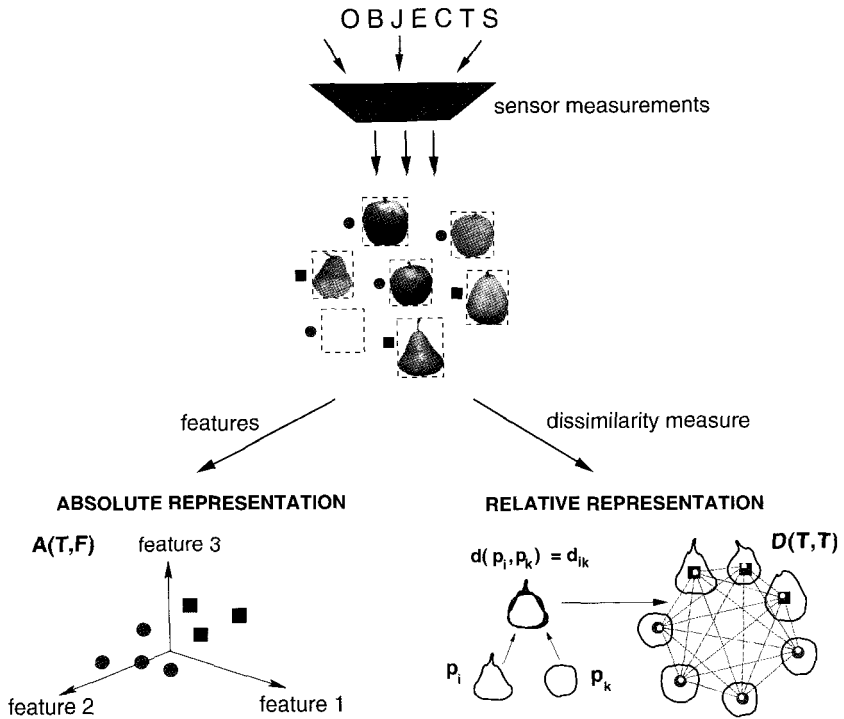


Figure 1.5 Feature-based (absolute) representation vs. dissimilarity-based (relative) representation. In the former description, objects are represented as points in a feature vector space, while in the latter description, objects are represented by a set of dissimilarity values.

also Fig. 1.5. Hence, each object x is represented by a vector of proximities $D(x, T)$ to the objects of T (precise definitions will be given in Chapter 4).

For a number of years, Goldfarb and colleagues have been trying to establish a new mathematical formalism allowing one to describe objects from a metaphysical point of view, that is, to learn their structure and characteristics from the process of their construction. This aims at unifying the geometric learning models (statistical approach with the geometry imposed by a feature space) and symbolic ones (structural approach) using dissimilarity as a natural bridge. A dissimilarity measure is determined in a process of inductive learning realized by so-called evolving transformation systems [Goldfarb, 1990; Goldfarb and Deshpande, 1997; Goldfarb and Golubitsky, 2001]. Loosely speaking, such a system is composed of a set of primitive structures, basic operations that transform one object

into another (or which generate a particular object) and some composition rules which permit the construction of new operations from existing ones [Goldfarb *et al.*, 1995, 1992, 2004; Goldfarb and Deshpande, 1997; Goldfarb and Golubitsky, 2001]. This is the symbolic component of the integrated model. The geometric component is defined by means of a dissimilarity. Since there is a cost associated with each operation, the dissimilarity is determined by the minimal sum of the costs of operations transforming one object into another (or generating this particular object). In this sense, the operations play the role of features, and the dissimilarity - dynamically learned in the training process - combines the objects into a class.

In this book, the study of dissimilarity representations has mainly an epistemological character. It focuses on *how* we decide (how we make a model to decide) that an entity belongs to a particular class. Since such a decision builds on the dissimilarities, we come closer to the nature of *what* a class is, as we believe that it is proximity which defines the class. This approach is much more flexible than the one based on features, since now, the geometry and the structure of a class are defined by the dissimilarity measure, which can reflect the structure of the objects in some space. Note that the reverse holds in a feature space, that is, a feature space determines the (Euclidean) distance measure, and hence the geometry; see also Fig. 1.4. Although, dissimilarity information is further treated in a numerical way, the development of statistical methods dealing with general dissimilarities is the first necessary step towards a unified learning model, as the dissimilarity measure may be developed in a structural approach.

Notwithstanding the fact that integrated model may be constructed for objects containing an inherent, identifiable structure or organization, like apples, shapes, spectra, text excerpts etc., current research is far from being generally applicable [Korkin and Goldfarb, 2002; Goldfarb and Golubitsky, 2001; Goldfarb *et al.*, 2000b, 2004]. On the other hand, there are a number of instances or events which are mainly characterized by discontinuous numerical or categorical information, e.g. gender, or number of children, etc. Therefore, we may have to consider heterogeneous types of information to support decisions in medicine, finance, etc. In such cases, the symbolic learning model cannot be directly utilized, but a dissimilarity can be defined. This emphasizes the importance of techniques operating on general dissimilarities. The study of proximity representations is the necessary starting point from which to depart on a journey into alternative inductive learning methodologies. These will learn the proximity measure, and hence a class description, from examples.

1.5 Relation to kernels

Kernel methods have become popular in statistical learning [Cristianini and Shawe-Taylor, 2000; Schölkopf and Smola, 2002]. Kernels are (conditionally) positive definite (cpd) functions of two variables, which can be thought to encode similarities between pairs of objects. They are originally defined in vector spaces, e.g. based on a feature representation of objects, and interpreted as generalized inner products in a reproducing kernel Hilbert space (RKHS). They offer a way to construct non-linear decision functions. In 1995, Vapnik proposed an elegant formulation of the largest margin classifier [Vapnik, 1998]. This support vector machine (SVM) is based on the reproducing property of kernels. Since then, many variants of the SVM have been applied to a wide range of learning problems.

Before the start of our research project [Duin *et al.*, 1997, 1998, 1999] it was already recognized that the class of cpd functions is restricted. It does not accommodate a number of useful proximity measures already developed in pattern recognition and computer vision. Many existing similarity measures are not positive definite and many existing dissimilarity measures are not Euclidean⁴ or even not metric. Examples are pairwise structural alignments of proteins, variants of the Hausdorff distance, and normalized edit-distances; see Chapter 5. The major limitation of using such kernels is that the original formulation of the SVM relies on a quadratic optimization. This problem is guaranteed to be convex for cpd kernels, and therefore uniquely solvable by standard algorithms. Kernel matrices disobeying these requirements are usually somehow regularized, e.g. by adding a suitable constant to their diagonal. Whether this is a beneficial strategy is an open question.

Although our research was inspired by the concept of kernel, the line we followed heavily deviates from the usage of kernels in machine learning [Shawe-Taylor and Cristianini, 2004]. This is caused by the pattern-recognition background of the problems we aim to solve. Our starting point is a given set of dissimilarities, observed or determined during the development of a pattern recognition system. It is defined by a human expert and his/her insight into the problem. This set is, thereby, an alternative to the definition of features (which also have to originate from such expertise).

A given Euclidean distance matrix may be transformed into a kernel and interpreted as a generalized Gram matrix in a proper Hilbert space.

⁴The dissimilarity measure being Euclidean is inherently related to the corresponding kernel being positive definite; this is explained in Chapter 3.

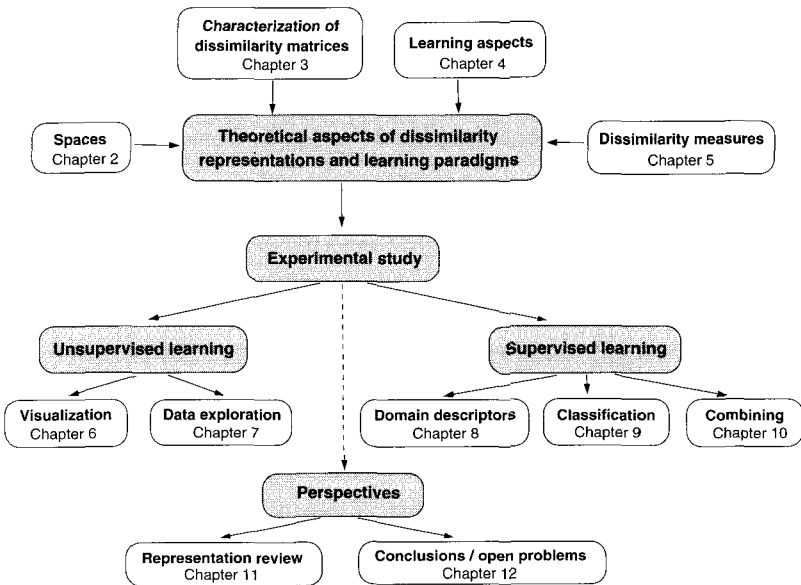


Figure 1.6 Conceptual outline of the book.

However, many general dissimilarity measures used in pattern recognition give rise to indefinite kernels, which have only recently become of interest [Haasdonk, 2005; Laub and Müller, 2004; Ong *et al.*, 2004], although we had already identified their importance before [Pekalska *et al.*, 2002b]. How to handle these is an important issue in this book.

1.6 Outline of the book

Dissimilarities play a key role in the quest for an integrated statistical-structural learning model, since they are a natural bridge between these two approaches, as explained in the previous sections. This is supported by the theory that (dis)similarity can be considered as a link between perception and higher-level knowledge, a crucial factor in the process of human recognition and categorization [Goldstone, 1999; Edelman *et al.*, 1998; Wharton *et al.*, 1992]. Throughout this book, the investigations are dedicated to dissimilarity (or similarity) representations. The goal is to study both methodology and approaches to learning from such representations. An outline of the book is presented in Fig. 1.6.

The concept of a vector space is fundamental to dissimilarity representations. The dissimilarity value captures the notion of closeness between two objects, which can be interpreted as a *distance* in a suitable space, or which can be used to build other spaces. Chapter 2 focuses on mathematical characteristics of various spaces, among others (generalized) metric spaces, norm spaces and inner product spaces. These spaces will later become the context in which the dissimilarities are interpreted and learning algorithms are designed. Familiarity with such spaces, their properties and their interrelations is needed for further understanding of learning processes.

Chapter 3 discusses fundamental issues of dissimilarity measures and generalized metric spaces. Since a metric distance, particularly the Euclidean distance, is mainly used in statistical learning, its special role is explained and related theorems are given. The properties of dissimilarity matrices are studied, together with some embeddings, i.e. spatial representations (vectors in a vector space found such that the dissimilarities are preserved) of symmetric dissimilarity matrices. This supports the analysis of pairwise dissimilarity data $D(T, T)$ based on a set of examples T .

Chapter 4 starts with a brief introduction into traditional statistical learning, followed by a more detailed description of dissimilarity representations. Three different approaches to building classifiers for such representations are considered. The first one uses dissimilarity values directly by interpreting them as neighborhood relations. The second one interprets them in a space where each dimension is a dissimilarity to a particular object. Finally, the third approach relies on a distance-preserving embedding to a vector space, in which classifiers are built.

In Chapter 5, various types of similarity and dissimilarity measures are described, together with their basic properties. The chapter ends with a brief overview of dissimilarity measures arising from various applications.

Chapters 6 and 7 start from fundamental questions related to exploratory data analysis on dissimilarity data. Data visualization is one of the most basic ways to get insight into relations between data instances. This is discussed in Chapter 6. Other issues related to data exploration and understanding are presented in Chapter 7. They focus on methods of unsupervised learning by reflecting upon the intrinsic dimension of the dissimilarity data, the complexity of the description and data structure in terms of clusters.

A possible approach to outlier detection is analyzed in Chapter 8 by constructing one-class classifiers. These methods are designed to solve problems, where mainly one of the classes, called the target class, is present.

Objects of the other, outlier, class occur rarely, cannot be well sampled, e.g. due the measurement costs or are untrustworthy. We introduce the problem and study a few one-class classifier methods built on dissimilarity representations.

Chapter 9 deals with classification. It practically examines three approaches to learning. For recognition, a so-called representation set is used instead of a complete training set. This chapter explains how to select such a set out of a training set and discusses the advantages and drawbacks of the studied techniques.

Chapter 10 investigates combining approaches. These either combine different dissimilarity representations or different types of classifiers. Additionally, it briefly discusses issues concerning meta-learning, i.e. conceptual dissimilarity representations resulting from combining classifiers, one-class classifiers or weak models, in general.

Chapter 11 discusses the issue of representation in pattern recognition and provides practical recommendations for the use of dissimilarity representations. Overall conclusions are given in Chapter 12.

Appendices A–D provide additional information on algebra, probability and statistics. Appendix E describes the data sets used in the experiments.

1.7 In summary

Dissimilarity representations are advantageous for identification and recognition, especially in the following cases:

- sensory data, such as spectra, digital or hyperspectral images
- data represented by histograms, contours or shapes,
- phenomena that can be described by probability density functions,
- binary files,
- text-related problems,
- when objects are encoded in a structural way by trees, graphs or strings,
- when objects are represented as vectors in a high-dimensional space,
- when the features describing objects are of mixed types,
- as a way of constructing nonlinear classifiers in given vector spaces.

Mathematical foundations for dissimilarity representations rely on:

(1) topology and general topology

[Sierpiński, 1952; Čech, 1966; Köthe, 1969; Willard, 1970; Munkres, 2000; Stadler *et al.*, 2001; Stadler and Stadler, 2001b],

- (2) linear algebra
[Greub, 1975; Białyński-Birula, 1976; Noble and Daniel, 1988; Leon, 1998; Lang, 2004],
- (3) operator theory
[Dunford and Schwarz, 1958; Sadovnichij, 1991],
- (4) functional analysis
[Kreyszig, 1978; Kurcyusz, 1982; Conway, 1990; Rudin, 1986, 1991],
- (5) indefinite inner product spaces
[Bognár, 1974; Alpay *et al.*, 1997; Iohvidov *et al.*, 1982; Dritschel and Rovnyak, 1996; Constantinescu and Gheondea, 2001],
- (6) probability theory
[Feller, 1968, 1971; Billingsley, 1995; Chung, 2001],
- (7) statistical pattern recognition
[Devişver and Kittler, 1982; Fukunaga, 1990; Webb, 1995; Devroye *et al.*, 1996; Duda *et al.*, 2001],
- (8) statistical learning
[Vapnik, 1998; Cherkassky and Mulier, 1998; Hastie *et al.*, 2001],
- (9) the work of Schölkopf and colleagues
[Schölkopf, 1997, 2000; Schölkopf *et al.*, 1999b, 1997a, 1999a, 2000b],
- (10) the results of Goldfarb
[Goldfarb, 1984, 1985, 1992],

and inspiration from many other researchers. We will present a systematic approach to study dissimilarity representations and discuss some novel procedures to learning. These are inevitably compared to the nearest neighbor rule (NN) [Cover and Hart, 1967], the method traditionally applied in this context. Although many researchers have thoroughly studied the NN method and its variants together with design of perfect dissimilarity measures (appropriate to the character of the NN rule), to our knowledge little attention was dedicated to alternative approaches. An exception are the support vector machines. These rely on a relatively narrow class of (conditionally) positive definite kernels, which, in turn, are special cases of similarity representations [Duin *et al.*, 1997, 1998]. Only recently the interest has arisen in indefinite kernels [Haasdonk, 2005; Laub and Müller, 2004; Ong *et al.*, 2004]. The methods presented here are applicable to general (dis)similarity representations, and this is where our main contribution lies. A more detailed description of the overall contributions is presented below.

Representation of objects. A proximity representation quantitatively encodes the proximity between pairs of objects. It relies on the representa-

tion set R , a relatively small collection of objects capturing the variability in the data. Each object is described by a vector of proximities to R . In the beginning, the representation set may consist of all training examples as it is reduced later in the process of instance selection. Here, a number of selection criteria are proposed and experimentally investigated for different learning frameworks. In this way, we extend the notion of a kernel to that of a proximity representation. If R is chosen to be the set of training examples, then this representation becomes a generalized kernel. When a suitable similarity measure is selected, a cpd kernel is obtained as a special case. Using a proximity representation, learning can be addressed in a more general way than by using the support vector machine. As such, we develop proximity representations as a first step towards bridging the statistical and structural approaches to pattern recognition. They are successfully used for solving object recognition problems.

Data understanding. Understanding data is a difficult task. The main consideration is whether the data sampling is sufficient to describe the problem domain well. Other important questions refer to intrinsic dimension, data structure, e.g. in terms of possible clusters and the means of data visualization. Since there exist many algorithms for unsupervised learning, our primary interest lies in the former questions.

In this book, three distinct approaches to learning from dissimilarity representations are proposed. The first one addresses the given dissimilarities directly. The second addresses a dissimilarity representation as a mapping based on the representation set R . As a result, the so-called dissimilarity space is considered, where each dimension corresponds to a dissimilarity to a particular object from R . The third one relies on an approximate embedding of dissimilarities into a (pseudo-)Euclidean space. The approaches are introduced, studied and applied in various situations.

Domain description. The problem of describing a class has gained a lot of attention, since it can be identified in many applications. The area of interest covers all problems where specified targets have to be recognized and anomalies or outlier situations have to be detected. These might be examples of any type of fault detection, abnormal behavior, or rare diseases. The basic assumption that an object belongs to a class is based on the idea that it is similar to other examples within this class. The identification procedure can be realized by a proximity function equipped with a threshold, determining whether or not an instance is a class member. This proximity function can be e.g. a distance to a set of selected prototypes.

Therefore, the data represented by proximities is more natural for building concept descriptors, since the proximity function can directly be built on these proximities.

To study this problem, we have not only adopted known algorithms for dissimilarity representations, but have also implemented and investigated new methods. Both in terms of efficiency and performance issues, our methods were found to perform well.

Classification. We propose new methodologies to deal with dissimilarity/similarity data. These rely either on approximate embedding in a pseudo-Euclidean space and construction of the classifiers there, or on building of the decision rules in a dissimilarity space, or on designing of neighborhood-based classifiers, e.g. the NN rule. In all cases, foundations are established, that allow us to handle general dissimilarity measures. Our methods do not require metric constraints, so their applicability is quite universal.

Combining. The possibility to combine various types of information has proved to be useful in practical applications; see e.g. [MCS00, 2000; MCS02, 2002]. We argue that combining either significantly different dissimilarity representations or classifiers different in nature on the same representation can be beneficial for learning. This may be useful when there is a lack of expertise of how a well-discrimination dissimilarity measure should be designed. A few measures can be considered, taking into account different characteristics of the data. For instance, when scanned digits should be compared, one measure focuses on the contour information, while others on the area or on statistical properties.

Applications. The proximity measure plays an important role in many research problems. Proximity representations are widely used in many areas, although often indirectly. They are used for text or image retrieval, data visualization, the process of learning from partially labeled sets, etc. A number of applications is discussed where such measures are found to be advantageous.

In essence. The study on dissimilarity representations applies to all dissimilarities, independently of the way they have been derived, e.g. from raw data or from an initial representation by features, strings or graphs. Expert knowledge on the application can be used to formulate this initial representation and in the definition of the proximity measure. This makes the dissimilarity representations developed natural candidates for combining

the strengths of structural and statistical approaches in pattern recognition and machine learning. The advantage of the structural approach lies in encoding both domain knowledge and the structure of an object. The benefit of the statistical approach lies in a well-developed mathematical theory of vector spaces. First, a description of objects in the structural framework can be found. This can then be quantized to capture the dissimilarity relations between the objects. If necessary, other structurally and statistically derived measures can be designed and combined. The final dissimilarity representation is then used in statistical learning. The results in this work justify the use and further exploration of dissimilarity information for pattern recognition and machine learning.