UvA-DARE (Digital Academic Repository)

Formal explorations of knowledge intensive tasks

Jansen, M.G.

_Citation for published version (APA):_

**General rights**
It is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), other than for strictly personal, individual use, unless the work is under an open content license (like Creative Commons).

**Disclaimer/Complaints regulations**
If you believe that digital publication of certain material infringes any of your rights or (privacy) interests, please let the Library know, stating your reasons. In case of a legitimate complaint, the Library will make the material inaccessible and/or remove it from the website. Please Ask the Library: https://uba.uva.nl/en/contact, or a letter to: Library of the University of Amsterdam, Secretariat, Singel 425, 1012 WP Amsterdam, The Netherlands. You will be contacted as soon as possible.
A Spectrum of Classification Methods

This chapter is submitted for publication.
We present a systematic survey of criteria for knowledge intensive classification problems. Criteria are build up from some simple definitions and combined into more complex ones. Orderings will allow the description of some basic preferential criteria. By relaxing some fundamental constraints other preferential criteria are obtained. It will be shown that all such criteria are interrelated and can be ordered by set-inclusion of their solution sets.
Such a systematic overview may be helpful as an indexing scheme for Problem Solving methods (PSMs) for classification. It can facilitate finding a criterion on the basis of a goal description.

4.1 Introduction

The goal of a classification task is to identify an object as belonging to a certain class. The object is described in terms of a (possible incomplete) set of observations. Identifying a bird, flower, rock or an object of art on the basis of some (possible incomplete) set of observations are all examples of classification problems.

An overview of several classification criteria and associated methods is given by Stefik [87]. By a classification criteria we mean a predicate which is true of a class when the class is a solution to the problem at hand, and false otherwise. The choice of a Problem Solving Method (PSM) depends on the criteria one chooses, and these may differ between different instances of classification problems. With each criterion one can associate a set of classes which are regarded as solutions. These can be ordered by set-inclusion. The resulting ordering makes it possible to compare criteria and the functionality of their associated PSMs.

Recently, attempts have been made to capture PSMs in libraries and facilitate their automatic retrieval on the basis of a problem specification [9] [65]. As a test-case classification was chosen as a task to be made operational. We feel that a systematic description of criteria which we offer here, may add to the construction of such libraries.

In this chapter we present a systematic overview of possible criteria for classification problems. We do this by describing simple criteria from basic definitions and use these in the construction of more complex ones.

A similar overview of criteria for diagnostic criteria was given by Console and Tarasso [21], and TenTeije and van Harmelen [89]. At least intuitively there seems to be a relation, and even a
possible overlap between diagnosis and some forms of classification. However, classification tasks are often regarded as much easier to define, and making use of much less sophisticated methods of problem solving.

4.2 Classification in terms of attributes

In classification tasks classes as well as observations are typically described in terms of attribute-value pairs (av-pairs).

**Definition 4.1** : Let $A$ be a finite set $^1$ of attributes. Each attribute $a \in A$ has a set $V_a$ (a domain) of possible values associated to it. An attribute-value pair is an ordered pair $(a, v)$ where $a \in A$ is an attribute and $v$ a value from its associated domain $V_a$.

Observations form a set of av-pairs and they represent an object that should be classified. Each av-pair in the set of observations will be referred to as an observation. Classes are here also represented as non-empty, finite sets of av-pairs, where the name of a set represents the name of the class. An alternative definition, used in the previous chapter and encountered in the literature (see e.g. Wieilinga [97]) is that a class-name implies the conjunction of av-pairs. The relation between such a representation and the one used here is quite straightforward.

Within each class and within the set of observations each attribute can have only a single value. When this is the case then the class, or the observations are said to be **internally consistent**. In the following internal consistency is assumed for all classes and the set of observations.

Not all attributes have to occur as part of some av-pair in a class or the observations. If an attribute does not occur in a class or the observations it is called **missing** (with respect to the class or observations).

Instead of allowing attributes to be missing, an alternative representation [87] is to give the value **unknown** to an attribute. In this case all attributes have to be mentioned in all class definitions and the observations.

In order to check whether the object, which is described by the observations falls under a particular class, the class must be compared to the observations. It is assumed here that in both the descriptions of observations and classes the same attributes and values are being used. However, this assumption may easily be given up. A **match** between two attributes can be defined as:

**Definition 4.2** : Two attributes (one from a class and one from the set of observations) match if there is a predefined mapping between the two values.

Matching two av-pairs can be as simple as a test for syntactical identity. Class and observation attributes may be distinct and a mapping might be needed to compare their values. Another option is the use of an interval as a value for an attribute. Hence, there is a certain degree of freedom in the implementation of matching.

The important notion of consistency between an av-pair from a class and one from the observations can be defined as follows:

**Definition 4.3** : An attribute $a_c$ of a class $c$ is consistent with the observation $o$ iff it does not occur in $o$ or it matches an attribute in $o$. This relation is symmetric in the following sense: an

---

$^1$In what follows all sets are finite.
attribute \( a_0 \) of an observation is consistent with a class \( c \) iff it does not occur in \( c \) or it matches an attribute in \( c \).

One could also allow disjunctions of attribute-value pairs. For example one could state that the value for the attribute colour for the class blackbird is either brown or black. In a representation in terms of sets of av-pairs this would mean that some attributes will have multiple values, which should be interpreted as disjunctions. This means that the notion of internal consistency of a set of attribute value-pairs can no longer be computed.

Note that the notion of consistency between class and observation attributes is in line with the use of disjunctive attributes.

4.2.1 The attribute level

Given a set of observations (as a set of av-pairs) we must determine which class can be regarded as a solution. In order to formulate a criterion we explore what possibilities arise when an av-pair belonging to a class is compared to one belonging to the observations. We limit our focus to the attribute level and focus on attributes without considering their values. There are four basic options:

1. The attribute occurs in both the class and the observation.
2. The attribute does not occur in the class but does in the observation.
3. The attribute occurs in the class but not in the observation.
4. The attribute occurs in neither the class nor the observation.

Next, we consider each of the above options and take the values of the attributes into consideration when needed. In the first case, when an attribute occurs in both the class and observations, there are two possibilities:

1. The two values of the attribute do not match. In that case the two av-pairs are inconsistent. (For short we say that the attribute itself (with respect to the class and observations) is inconsistent.)
2. The two values of the attribute match. Then the av-pair in the observation is said to be explained by the matching av-pair in the class. Similarly, the av-pair in the class is said to explain the matching av-pair in the observation.

In the second case the attribute is present in the observation but not in the class. In this case there is no inconsistency, but there is no matching and the attribute in the observation is said to be unexplained.

In the third case, the attribute present in the class is said to be an absent explanation of the absent attribute in the observation. Figure 4.1 illustrates these terms.

The fourth case represents an empty statement. The attribute is neither observed nor mentioned in the class-definition.
4.2.2 Universal criteria for classification

These options on the attribute level can be generalized to class-level by universally quantifying over av-pairs. Referring to this quantification we call the resulting criteria universal criteria for classification. The well-known criteria for weak and strong classification are among them [87].

Definition 4.4: Weak classification: A class \( c \) is a weak solution for an observation \( o \) iff all the av-pairs in \( o \) are consistent with all the av-pairs in \( c \).

Note that this definition merely involves a generalization of the attribute level criterion of consistency to class-level.

Definition 4.5: Strong classification: A class \( c \) is a strong solution for an observation \( o \) iff all the av-pairs in \( o \) are explained by all the av-pairs in \( c \).

Weak and strong classification are perhaps the best known criteria. However, they are not the only ones. Following up on the notions described earlier, another criterion would be:

Definition 4.6: Explanative classification: A class \( c \) is an explanatory solution for an observation \( o \) iff all the av-pairs in \( c \) are explanations for all the av-pairs in \( o \).

In addition, the last two criteria can be combined by the use of conjunction:

Definition 4.7: Strong explanatory classification: A class \( c \) is a strong explanatory solution for an observation \( o \) iff all the av-pairs in \( o \) are explained by all the av-pairs in \( c \), and all the av-pairs in \( c \) are explanations of an av-pair in \( o \).

Likewise disjunction can be used to produce yet another criterion:

Definition 4.8: Covered classification: A class \( c \) is a covered solution for an observation \( o \) iff all the av-pairs in \( o \) are explained by all the av-pairs in \( c \), or all the av-pairs in \( c \) are explanations of an av-pair in \( o \).

This last criterion does not add anything extra as far as individual classes are concerned, since a class which is a covered solution is either a strong or an explanatory solution. However the criterion is relevant when we associate with each criterion a set of solutions:

\[
\begin{align*}
&W E A K = \{ c \mid c \text{ is a weak solution} \} \\
&S T R O N G = \{ c \mid c \text{ is a strong solution} \} \\
&E X P L = \{ c \mid c \text{ is an explanatory solution} \}
\end{align*}
\]

\footnote{The name covered is chosen because either all attributes in the class or in the observation are explanations or explained respectively.}
\(\text{COVERED} = \{c \mid c \text{ is an explanative or a strong solution}\}\)

\(\text{STR-EXPL} = \{c \mid c \text{ is a strong explanative solution}\}\)

These sets can be partially ordered by set-inclusion as is shown in figure 4.2.

\[
\begin{align*}
\text{WEAK} \
\downarrow \\
\text{COVERED} \
\downarrow \\
\text{STRONG} & \quad \text{EXPL} \\
\downarrow \\
\text{STR-EXPL}
\end{align*}
\]

**Figure 4.2**
The ordering of classification criteria by set-inclusion. An upward line represents set inclusion: e.g. \(\text{STR-EXPL} \subseteq \text{STRONG}\).

From this partial order it follows that every strong explanative solution is also a strong solution. Every strong solution is also a weak solution. And both explanative and strong solutions are covered solutions. Every strong explanative solution is also an explanative solution, which in turn is also a weak solution.

### 4.2.3 An analysis in terms of attribute sets

The ordering of classification criteria can be further analyzed by looking at the attributes occurring in class and observations. Again, we do not consider the values of the attributes.

**Definition 4.9:** Let \(c\) be a class and \(A_c\) be the set containing all attributes having a value in \(c\):
\[
A_c = \{a \mid (a, v) \in c\}.
\]
Similarly let \(A_o\) be the set containing all attributes having a value in the observation \(o\):
\[
A_o = \{a \mid (a, v) \in o\}.
\]
When these two sets are compared the following cases may occur:

1. \(A_c = A_o\)
2. \(A_o \subseteq A_c\)
3. \(A_c \subseteq A_o\)
4. \(A_o \cap A_c = \emptyset\)
5. \(A_o \cap A_c \neq \emptyset\) (This subsumes 1-3).

These possible relations between the attributes in the observation and class can be used to get some insight into the nature of the above mentioned criteria.
Theorem 4.1: Suppose \( A_c = A_o \) for some class \( c \) and some set of observations \( o \). Then the five criteria coincide for \( c \): If, and only if \( c \) is a solution according to one of the criteria, it is a solution according to all others.

Example 4.1: The following (simplified) class definition was taken from a system built to classify igneous rocks [46],
\[
\text{granite} = \{ \langle \text{Grain size, Coarse grained} \rangle, \langle \text{Colour, Light} \rangle, \langle \text{Quartz, 20} \rangle, \langle \text{Quartz, 80} \rangle \}.
\]
The attribute Quartz is an interval attribute. Any observation of Quartz with a value within the interval 20 – 80 matches it.

The set \( A_{\text{granite}} \) contains all attributes occurring in the class granite: \( A_{\text{granite}} = \{ \text{Grain size, Colour, Quartz} \} \). If the set of observations defines values for all (and no more) of these attributes, than granite is either a solution according to all the criteria defined thus far, or to none at all.

Theorem 4.2: If for a class \( c \) \( A_0 \subset A_c \) holds, then the notions of weak and strong classification coincide for \( c \). In addition, \( c \) cannot be an explanatory solution and hence not a strong explanatory solution either.

Theorem 4.3: Suppose that for a class \( c \) \( A_c \subset A_o \) holds, then the notions of explanatory and weak classification coincide for \( c \). Also, \( c \) cannot be a strong solution and hence not a strong explanatory solution either.

Theorem 4.4: Suppose \( A_o \cap A_r = \emptyset \) for class \( c \). This means that the class \( c \) and the observations have no attributes in common. \( c \) is a weak solution with respect to \( o \) but evidently not a covered, strong, explanatory or strong explanatory solution.

Upon closer inspection we can define the aforementioned criteria, shown in figure 4.2, in terms of the attribute sets \( A_o \) and \( A_r \). By doing this it becomes clear why the above theorems are true.

Consider a class \( c \) which is regarded as a solution according to weak classification. When \( A_o \cap A_r \neq \emptyset \) holds, then all attributes which the class and observations have in common should match, otherwise an inconsistency occurs. Attributes which are missing in either \( A_o \) or \( A_r \) are allowed: they do not disturb consistency. Even if the class and the observations have no attributes in common (\( A_o \cap A_r = \emptyset \)) is \( c \) a consistent solution with respect to \( o \).

Formally we can express this as:
\[
c \text{ is a weak solution } \iff \forall a : (a \in A_o \cap A_r \land \text{match}(a)) \lor (a \notin A_o \cap A_r)
\]

The other criteria can be defined by a set of attributes as follows: We define a set of attributes whose elements are required to match. This set is constructed by taking the intersection of the set of attributes \( A_r \) occurring in the class \( c \) which is regarded as a solution, and the set of attributes occurring in the observations \( A_o \).

1. Strong classification requires that a class \( c \) is regarded as a solution when \( A_o \subseteq A_r \) holds, and all elements of the intersection \( A_r \cap A_o = A_o \) match.

2. Explanatory classification requires that \( A_r \subseteq A_o \) holds and all elements of the intersection \( A_r \cap A_o = A_c \) match.
3. Strong-explanative classification requires that $A_o = A_c$ holds and that all elements in the intersection (either $A_o$ or $A_c$) match.

4. Covered classification requires that either $A_o \subseteq A_c$ or $A_c \subseteq A_o$ holds and that all attributes in the subsumed set match.

In between covered and weak classification another criterion can be defined. To the definition of weak classification we can add the constraint that $A_o \cap A_c \neq \emptyset$ holds. We call this criterion **fortified classification**. It excludes the possibility that a class is solution if it has no attributes in common with the observations. This possibility can be seen as an (albeit absurd) criterion on its own and we refer to as **NIL**. It states that a class $c$ is a solution if $A_c \cap A_o = \emptyset$, and it’s use is obviously futile.

The partial ordering of universal criteria can be extended by these two criteria in the way depicted in figure 4.3.

```
WEAK
$A_o \cap A_c \neq \emptyset \lor A_o \cap A_c = \emptyset$

FORTIFIED
$A_o \cap A_c \neq \emptyset$
NIL
$A_o \cap A_c = \emptyset$

COVERED
$A_o \subseteq A_c \lor A_c \subseteq A_o$

STRONG
$A_o \subseteq A_c$
EXPL
$A_c \subseteq A_o$

STR-EXPL
$A_c = A_o$
```

**Figure 4.3**
The extended ordering for universal classification criteria.

From this ordering the aforementioned theorems are easily validated. For example the first theorem that the criteria (except NIL) coincide when $A_o = A_c$ can be easily proved as follows: suppose $o$ is a set of observations and $c$ a candidate solution. In addition assume that $c$ and $o$ have the same attributes, so $A_o = A_c$. If all attributes match, then $c$ is a strong-explanative solution. All other criteria (except NIL) subsume strong-explanative classification, so $c$ is a solution according to these as well. If not all attributes match there is an inconsistency and $c$ is not a solution according to any of the criteria. In a similar way the other theorems can be proven.
4.2.4 Preferential criteria

The ordering by set-inclusion of universal classification criteria can also be seen as a preferential structure for possible solutions. Instead of selecting a criterion for a particular classification problem, one could determine for each class the strongest criterion which it satisfies. Some classes may be only weak solutions, others strong (and hence weak) but not strong-explanatory etc. All classes will then be partially ordered along the several criteria and the most preferred can then be selected. The result is a universal preferential criterion for classification.

Note that the decision of what to consider as the most preferred class can still be a matter of choice. Whether to prefer strong to explanatory solutions, or vice versa is not determined by the ordering.

Such preferential criteria for classification can be extended further. One could allow some attributes to be inconsistent and prefer the class(es) with the least inconsistencies as a solution. One could count the number of explained attributes and give a preference to the class which has the most. Such criteria are not universal: they are existential, in the sense that they can be described with the help of an existential quantifier ranging over the attributes.

4.3 Relaxing constraints

The universal criteria defined above impose constraints on all attributes which occur in class and observations. There are situations in which one would like to relax these constraints. This can be done in two ways: One could relax the constraint concerning the presence of attributes, and one could allow (some) inconsistencies. We first discuss the latter option.

4.3.1 Allowing inconsistencies

Consider a case where all classes fail the constraint that all their attributes match the observations. In this case one could still prefer the class with the least inconsistencies.

Allowing inconsistencies seems particularly useful in preferential criteria: the class with the least inconsistencies is to be preferred. By relaxing the consistency constraint for each universal criterion a preferential structure emerges. To illustrate this construction we show this for strong-explanatory classification.

Strong-explanative (SE) classification is the strongest criteria in the sense that it puts the most constraints on the set of attributes that should match. Notice that if a class is a solution according to the strong-explanatory criterion there are no absent explanations or unexplained attributes and hence $A_n = A_e$ holds for any solution class $c$ with respect to the observations $o$. All attributes in either of these sets are required to match and hence no inconsistencies are allowed.

By gradually relaxing this last constraint we can construct a preferential structure for SE. Meanwhile we do not weaken the constraint that no absent explanations and unexplained attributes are allowed. Therefore we demand that $A_o = A_e$ holds as before, and relax the constraint that all the elements from one of these sets should match.

Let $A_o = \{a_1 \ldots a_n\}$ be the set of attributes whose elements have a definite value in a given observation set $O = \{a_1 = v_1 \ldots a_n = v_n\}$. For a class $c$ we define the set of attributes whose elements each match an av-pair in the observation set $O$: $M_c = \{a | matches(a) \text{ and } a \in A_o\}$.

We call such a set the matching-attribute set of class $c$. 
We assume that attributes occurring in the observations and the class are the same (their values may be different). Note that under this assumption a matching attribute set is always a subset of $A_o$. Therefore all possible matching-attribute sets (for all possible classes $c$) are given by the powerset of $A_o$. We can order all these sets by set-inclusion. The result is a lattice. An example lattice is shown in figure 4.4.

![Figure 4.4](image)

**Figure 4.4**
The lattice of possible matching attributes sets. The figure shows all possible subsets of the set $\{a_1 \ldots a_n\}$ ordered by set-inclusion. An upward line indicates subset inclusion. Not all subsets are shown. Dashed lines and dots indicate the presence of sets not shown.

This lattice can be interpreted in two ways. First, as a preferential structure: for each class $c$, the set of matching attributes $M_c$ can be computed (given an observation) and its place in the lattice be determined. To decide which class contains the fewest inconsistent attributes, can now be decided by an answer to the question whose class’s matching-attribute set is closest to the top of the lattice. At the top of the lattice is the set $A_o = \{a_1 \ldots a_n\}$. If a class has $A_o$ as it’s matching-attribute set then there are no inconsistencies. On the other hand if it’s attribute-matching set is $\emptyset$ then there are $n$ inconsistencies. All other possible matching attribute sets are in between these extremes.

Another view on the same structure is to see every node in the lattice as a demand that the mentioned attributes are required to match. For example, the node $\{a_1 \ldots a_i\}$ can be interpreted as follows: $c$ is a solution if (and only if) all the attributes in $\{a_1 \ldots a_i\}$ match. Then the lattice is no longer viewed as a preferential structure. Every node represents an absolute criterion and the lattice as a whole represents the space of possible criteria relative to strong-explanative classification.

### 4.3.2 Absolute criteria

The lattice of matching attributes under the assumption that the attribute-sets of class and observations are the same, can be seen in two ways. First, as a preference structure for strong-explanative classification with possible inconsistencies. Given an observation with attributes $A_o$ we can determine for each class $c$ with attributes $A_c$ and $A_c = A_o$ the subset of attributes $M_c$, whose elements
each match an attribute occurring in the observations. The place in the lattice of the matching-attribute set determines the preference order of the class.

Second, with each node of this lattice we can also associate a new absolute criterion for classification. Each set of attributes $M_i \subset A_o$ can be taken as a constraint saying that these attributes should match. For example the set at the top of the lattice contains all attributes by definition. Demanding that these attributes should match is equal to the criterion of strong-explanative classification.

The set $M_c = \{a_1 \ldots a_{n-1}\}$ then corresponds to the criterion in which all the attributes except $a_n$ should match. In this case the attribute $a_n$ is given a special status: it is regarded as unnecessary.

Similar to the construction of the universal criteria, we can define a set of classes which are covered by these new criteria. (Remember we have assumed $A_c = A_o$). With each node of the lattice of matching-attribute sets we associate a set of classes. Each class $c$ is a member of the associated set if its matching-attribute set $M_c$ is equal to the set mentioned at the node:

Let $\phi(A_o) = \{A_o, A_1, \ldots, A_n, \emptyset\}$ be the powerset of $A_o$.

- $E0_{se} = STR-EXP = \{c \mid M_c = A_o\}$
- $E1_{se} = \{c \mid M_c = A_1\}$
- $\cdots$
- $En_{se} = \{c \mid M_c = A_n\}$
- $E\emptyset_{se} = \{c \mid M_c = \emptyset\}$

Hence with every node in the lattice in figure 4.4 we associate a set $Ei_{se}$ which contains the classes whose matching attribute sets are equal to the set which forms the node. As said before each node can be interpreted as a criterion that the given list of attributes should match, and the associated set of classes contains classes which satisfy this criterion. Moving up the lattice (by following the lines) means moving to stronger criteria. The classes associated with each node also satisfy the criteria with lower, subsumed nodes.

These sets of classes, or criteria are closed under conjunction. For example, when one wants the set of classes who match attributes $a_i \ldots a_j$ and $a_k \ldots a_l$ then the answer can be found as follows. Identify the nodes corresponding to both sets and follow lines upward in the lattice to the first node at which they meet. The result is the union $a_i \ldots a_j \cup a_k \ldots a_l$. This set is part of the lattice and so is the associated set of classes.

However, the disjunction of criteria is not yet defined. In fact the matching attribute lattice is defined on the attribute level, and when generalizing to sets of classes the disjunctive closure should be added. For example, consider the criterion where at most one inconsistent attribute is allowed. This can be defined by a disjunction of all the criteria which allow one particular element to be inconsistent. This criterion is not an element of the sets of classes defined in the above construction.

By adding all sets associated with the disjunctions of criteria, the disjunctive closure of all the criteria in the matching attribute lattice is obtained. This results in a structure which is closed under conjunction and disjunction. Again this forms a lattice of criteria whose elements are all variants of strong-explanative classification.
We refer to these criteria as existential criteria because they allow for some attributes to be inconsistent. Universal strong-explanative classification can now be extended by a lattice of existential strong-explanative criteria, of which the universal criterion is the bottom element.

### 4.3.3 Comparing existential criteria

The same construction which produced new existential criteria for strong-explanative classification can be used to relax the other universal classification criteria.

To summarize the construction procedure: for each universal criterion list all the possible sets of attributes which are required to match the observations. In fact, all these possible sets are given by the powerset of the attribute set defined by the universal criterion. Ordering all elements of this powerset by set-inclusion results in a lattice. The top element of this lattice will allow no inconsistencies (it's the universal criterion itself), the bottom element will allow all elements to be inconsistent (the empty set), and all the other sets express that some particular elements are allowed to be inconsistent.

Next, associate with each node a set of classes in such a way that the matching attributes of each class are exactly those in the node. Of all these sets take the disjunctive closure. All these sets of classes are now closed under disjunction and conjunction and form another lattice. This lattice is the inverse of the one it is constructed from. The top element contains all classes that may have no consistent attributes, the bottom element is the universal criterion that one has chosen at the beginning of the construction process.

Hence, instead of strong-explanative classification we can relax the criterion of strong classification by assuming \( A_0 \subset A_c \) instead of \( A_0 = A_c \). The line of reasoning is completely similar to the above construction. The only difference between all the existential strong criteria, as opposed to the existential strong-explanative criteria is that the former allows some attributes in the class to be absent explanations, which is expressed by \( A_0 \subset A_c \). Under all strong classification criteria, whether we allow inconsistencies or not, the difference of the sets \( A_c \) and \( A_o \) (written as \( A_c \sim A_o \)) is the set of absent explanations. Note that there are no unexplained attributes under all strong classification criteria. Similarly, all explanatory criteria allow unexplained attributes, characterized by the set \( A_o \sim A_c \). There are no absent explanations.

This means that if we allow inconsistencies, we can still order criteria on the basis of explanatory attributes and explanations. If we relax weak and fortified classification by allowing inconsistencies, similar lattices are obtained. The only difference between all the pairs of new criteria is that the fortified ones drop the possibility that \( A_o = A_c \).

Hence, every universal criterion can be extended by a lattice of existential criteria of which it is the bottom element. Furthermore, every existential criterion which allows an attribute to be inconsistent is related to the other existential criteria which expresses the same fact, but differs under the assumption of the relation between \( A_o \) and \( A_c \). For example: Strong classification which allows attribute \( a_1 \) to be inconsistent subsumes strong-explanative classification which allows \( a_1 \) to be inconsistent. This existential strong-explanative criterion is still stronger than its strong counterpart in that it does not allow attributes failing to be explanations. All these inter-relationships lead to the ordering of the criteria by set-inclusion shown in figure 4.5.
4.4 Allowing missing attributes

Instead of relaxing the constraint that all attributes from a certain set should match by allowing some of them to be inconsistent, one could also allow certain attributes to be missing from either the class or set of observations.

For each universal criterion a similar construction procedure as was used for allowing inconsistencies can be used to allow missing attributes. Instead of listing the possible sets of allowed inconsistent attributes for each criterion, we list the sets of allowed missing attributes.

For example, for strong classification enumerating all possible sets of attributes which are allowed to be missing in the observation, can be done as follows. Again, let \( A_o \) be the set of attributes in the observations: \( A_o = \{a_1 \ldots a_n\} \). Let \( \wp(A_o) = \{\emptyset, A_1, \ldots, A_m, A_o\} \) be the powerset of \( A_o \). \( A_c \) denotes the set of attributes occurring in the class \( c \). For strong classification \( A_o \subset A_c \) holds and hence \( A_o \cap A_c = A_o \). We enumerate all subsets of the powerset of this set.

\[
\begin{align*}
A_o \cap A_c &= A_o \\
A_o \cap \{a_1\} \cap A_c &= A_o \cap \{a_1\} \\
& \ldots \\
a_o \cap \{a_1 \ldots a_i\} \cap A_c &= A_o \cap \{a_1 \ldots a_i\} \ (0 \leq i \leq n) \\
& \ldots \\
A_o \cap A_c &= A_o \cap A_o \Leftarrow \emptyset \cap A_c = \emptyset
\end{align*}
\]

The same can be done for explanatory criteria. Here we \( A_c \subset A_o \) holds and hence \( A_o \cap A_c = A_c \). Hence, by reading \( A_o \) for \( A_c \) and vice versa in the above enumeration, one obtains the possible sets of missing attributes for explanatory classification. For strong-explanative criteria a similar
construction $A_0 = A_c$ holds and the following enumeration can be performed:

\[
A_0 \sim \emptyset = A_o \\
A_0 \sim \{a_1\} \\
\ldots \\
A_0 \sim \{a_1 \ldots a_n\} = \emptyset
\]

Similar to the construction of inconsistent criteria, with each of these sets a set of classes which fulfill the associated constraints can be associated. Each set of classes then represents a criterion. If we take the disjunctive closure of these criteria, we end up with all possible criteria for missing attributes.

There are a few things to note here. Allowing all attributes to be missing, comes down to the universal criterion NIL, as defined in section 2. Allowing no attributes missing in either the class or observations is of course universal strong-explanative classification.

Allowing no unexplained attributes equals strong classification, demanding all attributes to be explained equals covered classification. Expressing that at most two attributes may be unexplained, comes down to the disjunction of all the constraints which mention two particular attributes to be unexplained.

These missing-attribute criteria do form a lattice with as top the criterion NIL. This is not in line with the ordering of universal criteria. For example, we can relax the constraint of strong classification that each attribute in $A_o \cap A_c$ should match, further and further until $A_o \cap A_c = \emptyset$, which is the criterion NIL. However, in the definition of the universal criteria this option was explicitly forbidden, and NIL does therefore not subsume any universal criterion.

### 4.5 Combining inconsistencies and missing attributes

The last type of criteria we discuss are those that allow a combination of missing and inconsistent attributes. If the attributes that are allowed to be missing according to one criterion are disjoint from those of the one which allows them to be missing, the criteria can be combined by both conjunction and disjunction.

However, the situation is different for those criteria which share attributes. First, note that an attribute cannot be missing and inconsistent at the same time. We will show that inconsistency is the stronger notion of the two.

Intuitively, if a class is allowed to have one particular attribute inconsistent then it should also be allowed to have this attribute missing. Missing criteria seem to subsume inconsistent ones, defined for the same attributes.

One could argue one step further: if an attribute is missing in either the observations or the class, we do not know its value. This means that it could be inconsistent. If we allow a class with one particular attribute to be missing, we also allow the possibility that it is inconsistent. Therefore if a class $c$ with one particular attribute missing is regarded as a solution, so must a class $c'$ with the same inconsistent attribute. This view would entail that inconsistent and missing attributes amount to the same. We do not agree with this view and argue against it:
It may be true that if we allow a class with one particular attribute to be missing, we also allow the possibility that it is inconsistent. But we only disqualify a class as a solution if we are certain of an inconsistency. So if a class is allowed to have an attribute missing, we trust it will not be inconsistent.

Hence, missing criteria subsume inconsistent ones (for the same attributes), but not *vice versa*. This means that instead of saying that the set of classes which satisfy the constraint that an attribute is allowed to be inconsistent or missing, equals the set of classes which satisfy the constraint that it is allowed to be inconsistent, which is the stronger notion.

In summary, combining missing and inconsistent criteria for the same attributes can only be performed by disjunction and yields the inconsistent ones. If attributes of both criteria overlap, the ones who are allowed to be missing should be retained. The other attributes can be combined by disjunction and conjunction into new criteria.

### 4.6 Discussion

The previous treatment of classification criteria is not meant to be an exhaustive description of the task. The abstraction of features, the hierarchical ordering of classes and the refinement of solutions, have not been treated here. Also, we have not indicated how to solve any of the classification problems associated with a criterion.

We have shown how classification criteria can be described and be compared to each other in a systematic way. Starting with some simple criteria, one could build up more complex ones by simple constructions and weakening constraints. Any partial ordering of criteria can be viewed as a preference structure which in itself can be used as another classification criterion.

Describing a task like classification this way may facilitate the construction of libraries of problem solving methods for the task. Attempts are being made for the automated selection of a problem solving method after some initial goal specification by a user [7]. In order to select such a method one has to have some systematic account of how to describe the competence of the PSMs and their relationships.

Describing criteria for a task also allows for more complex ways of problem solving. Instead of identifying a criterion and try to establish a solution accordingly, one could go one step further. If some idea of what a solution may look like is available, one may search for the criterion which approaches this idea in an optimal way. This involves moving through the ordering of criteria, until the constraints for a solution set are optimally realized.

As an example, consider classification criteria which produce too many, or too few solutions. Strengthening, or weakening may lead to a more appropriate solution set. Other examples involve the search for criteria that should at least produce certain predefined classes as solutions, or should meet some predefined constraints. It is this kind of 'meta problem solving' which is also involved in the selection of a PSM on the basis of a goal description.

Another point for discussion involves the representation of the ontological elements like classes, attributes, values etc. We have chosen here for a set-based representation instead of one in (predicate)logic. Although one could easily translate one of these representations into the other, the logical one seems to be an interesting candidate for linking criteria to PSMs.

An an example we mention the criteria for classification with the least number of inconsistent attributes. At least intuitively there seems some correspondence between these criteria and the
'approximate reasoning' logic of Schaerf and Cadoli [79]. Ten Teije [88] uses this logic to describe diagnostic criteria.

This is an example of how one can associate with each criterion a consequence relation. Problem solving can then be described as performing deduction or abduction according to this consequence relation. The relation between criteria and different styles of logical reasoning will be a topic of future research.

4.7 Acknowledgement

This chapter originated after a discussion of the main ideas in [65] which can be found at http://kmi.open.ac.uk/projects/ibrow/documents.htm. We thank Enrico Motta for reviewing an earlier draft of this chapter. We also would like to thank F. van Harmelen en A. ten Teije from the Free Univeristy of Amsterdam for discussing the main ideas.