FACULTY WORKING PAPER 92-0145

New Dimensions of Inductive Learning for Credit Risk Analysis

Antoinette Canart Tessmer
Department of Finance
and
The Beckman Institute for Advanced Science and Technology

Bureau of Economic and Business Research
College of Commerce and Business Administration
University of Illinois at Urbana-Champaign

BEBR

FACULTY WORKING PAPER NO. 92-0145

College of Commerce and Business Administration

University of Illinois at Urbana-Champaign

August 1992

New Dimensions of Inductive Learning for Credit Risk Analysis

Antoinette Canart Tessmer

Department of Finance
College of Commerce and Business Administration
and
The Beckman Institute for Advanced Science and Technology

NEW DIMENSIONS OF INDUCTIVE LEARNING FOR CREDIT RISK ANALYSIS

Antoinette Canart Tessmer

Visiting Assistant Professor

Department of Finance
College of Commerce and Business Administration
and
The Beckman Institute for Advanced Science and Technology
University of Illinois at Urbana-Champaign

108 Commerce West
1206 South 6th Street
Champaign, IL 61820
(217) 333-3522
acanart@ux1.cso.uiuc.edu

NEW DIMENSIONS OF INDUCTIVE LEARNING FOR CREDIT RISK ANALYSIS

Abstract

The paper presents two new dimensions of inductive learning for credit risk analysis. The first new dimension points out the specific impact of type I and type II errors on the accuracy of the inductive learning process. A Dynamic Updating Process is proposed to refine the credit granting decision over time and therefore improve the accuracy of the learning process. The second new dimension takes advantage of the decision tree representation to solve the problem of instability in classification results. A global tree interpretation method (GTip) is proposed which globalizes the relevant content of a set of original trees while reducing noise and overfitting effects. Both new dimensions are aimed at improving the inductive learning approach when applied in credit risk analysis.

NEW DIMENSIONS OF INDUCTIVE LEARNING FOR CREDIT RISK ANALYSIS[1]

Introduction

Evaluating credit risk is one of the most important activities of a commercial bank. When assessing an applicant's creditworthiness, the credit officer must take into consideration the credit terms, the applicant's characteristics and the lending bank characteristics in order to meet the bank's risk-return objectives [14]. For that purpose, the credit investigation process is aimed at acquiring enough information to determine the applicant's ability and willingness to service the requested credit. However, a balance has to be struck between credit investigation costs and return probabilities. Cohen, et al. observe that "*the loan evaluation process generally seems to be handled by particular sets of heuristics. These heuristics lead to a 'satisfying' behavior, i.e., choosing the best alternative which has been found after a limited period of search*" [10].

Several decision support tools are currently used in credit management in order to reduce the time and costs spent during investigation and assessment. For several years, the ability of computers to process numerical data has been used to rapidly search for financial information in large data bases and to process this information in a concise and easy to assess format. However, as non-financial and often qualitative information prevail in the assessment of small size credits, common credit scoring tools appear less adapted to process this type of data. Research conducted in the field of artificial intelligence propose an alternative approach, called *expert systems*. These "intelligent" tools have already been used to support several business or management problems.[2] A specific knowledge acquisition method, called *inductive learning*, have already given positive results which suggest the need to further investigate how this artificial intelligence approach may help to design a decision support tool in credit analysis

This paper briefly reviews the inductive learning methodology that has already been compared with success to well known statistical classification tools.[2] Without innovating the methodology itself, a promising dimension is proposed, which emphasizes the use of type I and type II errors to improve the inductive process. As they are located very close to the limit between accepted and rejected credits, type I and type II errors may nudge the inductive process towards a more accurate definition of the concept to be learned. A Dynamic Updating Process is proposed to refine the credit granting decision over time and, therefore, improve the accuracy of the learning process. The paper also focuses on the symbolic representation of the results obtained by induction. The decision tree is presented as a highly explicit representation that offers a new dimension to the interpretation of the classification results. This new dimension extends a sole evaluation based on accuracy, which was usually used in the literature.[2] In the framework of credit risk analysis, the results obtained by induction may give relevant insights about the underlying structure of the data [7] as well as about a financial theory justifying the credit analysis. However, experiments show how inductive results may be negatively influenced by noise and overfitting effects, and, therefore, may be highly unstable. This paper proposes and illustrates a new global interpretation process that reduces noise and overfitting effects and helps to discover a stable and relevant underlying structure of the data

Part I briefly presents the inductive learning approach. Starting with a presentation of the research environment in which the inductive learning methodology has emerged, the historical evolution and the terminology of the inductive approach are reviewed. Part II proposes the new dimensions of the methodology in the framework of a credit risk analysis: the impact of specific credit decisions in induction and a specific knowledge representation, the *decision tree*. Part III develops the impact of specific credit decisions and proposes an innovative process to update the credit decision over time. Finally, Part IV discusses a way to interpret the results obtained by induction and proposes a new method to interpret the decision trees and to globalize the *insights about the predictive structure of the data* [7].

I Review of the Inductive Learning Methodology

Artificial Intelligence and Expert Systems

"*Artificial Intelligence is the part of Computer Science concerned with designing intelligent computer systems, that is, systems that exhibit the characteristics we associate with intelligence in human behavior*" [2]. There are four distinct research areas regarding the typical human behavior they address: robotics, language, vision, and reasoning. Artificial intelligence research in *reasoning* is the field that is addressed in this paper. Reasoning involves several knowledge-related tasks such as problem solving and learning, which artificial intelligence tries to emulate by computer. The main application of those techniques are the *expert systems*. They are "intelligent" programs that interact with the user in a "consultation dialogue," just as a human with some type of expertise, explaining the problem, performing suggested tests, and justifying the solutions [3].

Typically, an *expert system* is composed of three main modules: the *knowledge base*, the *inference engine*, and the *user interface*, as shown in Exhibit 1. The main characteristic resides in the physical separation between the data or knowledge that the program processes (the *knowledge base*) and the logic it follows to solve a problem (the *inference engine*). Artificial intelligence research has been working on more and more sophisticated *inference engines* able to deal with the most tricky problems and to process any kind of knowledge. The knowledge, contained in the *knowledge base*, gathers the expertise related to a given domain of application. The completion of a relevant knowledge base remains closely related to the domain in which the expert system will be used. It also forms the main stage in the design of a successful system. The expertise contained in the knowledge base may be expressed in several formats. The most common format is the decision rule which expresses a condition-action relation in a very natural language. For example, the following rule expresses a piece of expertise in a very simple way :

> if the applicant asks a credit to take over a restaurant, and
> if the previous owner decided to retire, and
> if the restaurant is very popular, and
> if the applicant has an experience of 10 years in a similar activity,
> then the credit committee decides to accept the credit.

Several *knowledge acquisition* strategies have been applied to the financial field. For example, Bouwman proposes a *handcrafting* acquisition method to try to identify the decision making behavior of a financial analyst [4,5,6]. The strategy consists of asking decision makers to verbalize their reasoning during their financial analysis, tape-recording their verbalizations, and using the resulting transcripts, called *concurrent* or *thinking-aloud protocols*, as input data for a protocol analysis. Bouwman's research forms one of the most advanced knowledge acquisition strategies by handcrafting. As a result, the author presents a detailed, descriptive analysis of the decision making processes involved in screening companies for potential investment.

Another knowledge acquisition approach, which has been developed by the artificial intelligence community as well, is proposed in some recent research in finance or management. This method, called *machine learning*, does not require the long and tedious interviews conducted in Bouwman's research. Instead, it relies on examples of previous decisions to learn the decision maker's reasoning process. Carter [8], Chandler [9], Currim [11], Han [13], Messier [18] and Shaw [21,22] have successfully applied the machine learning approach in the fields of consumer choice, accountancy, bankruptcy prediction, and credit risk assessment. The following sections review the historical evolution of the inductive learning approach and introduce its terminology.

Inductive Learning: Historical Evolution

According to the dictionary [1]: "*in.duc.tion, n., the act or process of deriving general principles from particular facts or instances*". According to a computer scientist: "*The ability of people to make accurate generalizations from a few scattered facts or to discover patterns in a seemingly choatic collection of observations*" [19] is achieved by a process called *inductive learning.*

In 1933, Kenneth Smoke published the results of a famous conceptual learning study which attempted to understand the learning process of a human being [23]. Smoke tried to determine the contributions of "positives instances", where the required characteristics of the concept are included in the stimulus, and "negative instances", where one or more of the required characteristics are absent, in the acquisition of concepts. Smoke concluded that negative instances do little to facilitate learning, which raised a controversy largely discussed in the psychology literature [15,16] and started a continuous wave of research focusing on inductive concept learning.

In a first interdisciplinary research project, dating of 1966, researchers in data processing, psychology, and social sciences together tried "*to learn more about how people solve complex learning problems and to observe the performance of different varieties of a general learning automaton designed to solve inductive problems which require some learning on the part of the problem solver*" [17]. This research project introduced the first paradigm of Concept Learning System. Hunt defines a Concept Learning System as "*a device for creating a concept corresponding to some partition of a sample of objects which have been categorized by a pre-established rule for using a name. It is assumed that the Concept Learning System forms its concept by observing examples of the use of the name, i.e., by observing a subset of objects of the universe and being informed of whether or not the name is applicable to them*" [17].

Subsequent research on the same paradigm has been conducted in several fields, e.g., machine learning, pattern detection and recognition, statistics, or behavioral sciences, and led to an implementation of Hunt's optimistic view of a Concept Learning System as a "*complement to Factor Analysis for nominal data*" [17]. This concern has also been expressed by Michalski, in 1983: "*the widely used traditional mathematical and statistical data analysis techniques [...] are not sufficiently powerful for [the detection of conceptual patterns]. Methods for conceptual data analysis are needed, that generate not merely mathematical formulas but logic-style descriptions, characterizing data in terms of high-level human-oriented concepts and relationships*" [19].

Terminology

This section briefly presents the vocabulary frequently used in inductive learning. As the terminology may be new for some readers, new terms are related to their counterparts in credit risk analysis to facilitate the comprehension.

Inductive learning, also called *induction*, is a classification procedure that accurately models the classes of known objects so that, when an object of an unknown class is encountered, a plausible class label can be affixed to it. In its general form, an inductive process is structured into three distinct elements: an *instance space*, an *algorithm* performing induction, and an *output* describing a classification *concept.* Rendell [20] illustrates it as in Exhibit 2.

The *instance space* is a k-dimensional space where each point, or example, is described by a vector (x) composed of k independent variables called *attributes*, and a discrete or continuous classification (u). A typical discrete classification function has a value 0 to represent an example that does not belong to the concept, also called a *negative* example. A value of 1 represents an example that belongs to the concept, also called a *positive* example. If continuous, the classification function gives the probability with which a given example represents the concept. In the framework of a given run of the algorithm, the *instance space* is defined in the input *training sample.* Some examples of instance space are given in Exhibit 3 where the examples are defined with two attributes, att_1 and att_2. A binary classification function is represented as a '+' for a positive example and a '-' for a negative example.

The *output concept* represents the knowledge acquired by induction. Rendell [20] expresses it as a function, $u(x)$, mapping a k-dimensional vector into a class membership discrete value (0 or 1), when the concept is *binary-valued*, or into a class membership continuous value (in [0..1]) for a *graded* or *probabilistic* concept. The concept is usually tested on a *testing sample*, independent of the input *training sample.* In Exhibit 3, the concept to be learned corresponds to the boundaries which separate the positive examples from the negative examples.

The outlook of the concept in the instance space is defined by its *size* and its *concentration* [20]. The *size* refers to the relative frequency of positive examples in the training sample. The *concentration* is determined by the number of boundaries appearing in the space or, in other words, by the degree of localization of the concept in the instance space. Exhibit 3 illustrates four possible outlooks: a large concept (3a), a scattered concept (3b), a concentrated concept (3c), and a widespread concept (3d). In each situation, the inductive process tries to discover the most accurate boundary between positive and negative examples and gives the characteristics of both classes of points.

In the framework of a credit risk assessment process, a positive point (+) corresponds to a credit that has been previously accepted by the credit committee. A negative point (-) corresponds to a credit that has been previously rejected by the credit committee. The instance space is thus a set of past credit decisions on which the induction is performed. Each past decision is described by a set of quantitative or qualitative pieces of information, which are the attributes. The "credit granting" concept is positive or negative, no intermediate value is possible. Therefore, the concept to be learned is binary (1 or 0). The objective of the inductive process is to discover the boundary separating the accepted and rejected credits and to give the characteristics of both groups of points.[3]

II New Dimensions Offered by the Methodology

As explained in the previous chapter, the knowledge acquisition by induction relies on past credit decisions, i.e., positive and negative points of the instance space, and generates a description of the intra-group similarities and inter-group dissimilarities of those past decisions. The presence of positive or negative points being very close to the boundary between groups, i.e., the presence of positive and negative past credit decisions being very similar, may be important for the induction process. That similarity is the first new dimension explained in the following section. Moreover, the quality of the knowledge description generated by induction determines how understandable and usable a knowledge base containing this description will be. The advantages of a concept representation that is called the *decision tree* forms the second new dimension that is discussed in the next two sections.

Specific Examples or Near Misses

Suppose that the concept that we are trying to learn consists of only one boundary in the instance space and that a large number of examples are located close to that boundary, as shown in Exhibit 4. It looks obvious that the discovery of the correct boundary, and therefore the definition of the correct characteristics of positive and negative examples, will be facilitated in a hypothesis space as illustrated in Exhibit 4. As a matter of fact, the positive and negative examples very close to the boundary are nudging the inductive process toward the correct location of the boundary.

In the literature, those specific examples, either correctly or incorrectly classified, that are very close to the boundary are referred to as *near misses* , which are "*examples [..] quite like the concept to be learned but which differ from that concept in only a small number of significant points*" [24]. The near miss is an example that is very close to the concept, but some elements make it incorrect to be considered as a member of the concept. For example, in Exhibit 4, they are the negative examples close to the boundary. Alternatively, a near miss is an example that is very close to a negative example, but some elements make it correct to be considered as a member of the concept. For example, in Exhibit 4, they are the positive examples close to the boundary. The small but significant differences given by near misses allow the inductive algorithm to localize some parts of its current position about a concept and to improve it. Important qualities of the concept to be learned can be suggested by carefully selecting representative near misses. The near misses thus offer the possibility of conveying quite directly some particular ideas to the algorithm. The detection and a relevant definition of near-misses is not a straightforward task and has to be found in relation with the field of application. The relevance of this definition is primordial as the literature maintains that the presence of near-misses in the training sample may have a dramatic impact on the accuracy of the output concept [20].

In the framework of a credit risk assessment process, the near-misses can be given an interesting interpretation. As those specific examples are very close to the boundary, their description, in terms of the chosen attributes, is very close to the attribute values of the examples belonging to the opposite class. In other words, some previously accepted credits (positive examples) may show most of the characteristics of a rejected credit (negative example). As a result, the decision to grant or not to grant the credit may have been uncertain and confused. Therefore, it may be supposed that that sub-set of positive credits contains *credit errors*, i.e., type I errors, which will be called *positive near-misses*. In the same way, some previously rejected credits (negative examples) may actually show most of the characteristics of an accepted credit (positive example) and therefore may contain *commercial errors*, i.e., type II errors, which will be called *negative near-misses.*

The detection of near-misses corresponds therefore to the detection of type I and type II errors. Type I errors are possible to trace, as the evolution of an accepted credit will determine its positive (well-running credit) or negative (failed credit) outcome. Unfortunately, type II errors are not so obvious to discover. Supposing however that the credit department is able to provide the necessary information concerning type I and type II errors, the instance space of the corresponding concept looks as in Exhibit 5, in which type I errors (+') and type II errors (-') remain very close to the boundary.

Such a specific impact of type I and type II errors on the output results is unique and has never been approached by any statistical classification tool commonly used in credit risk analysis. Therefore, the use of type I and type II errors as near-missed examples presents a promising contribution to the problem of designing a decision support tool in credit risk analysis. The contribution of near-misses is further discussed in part III.

Explicit Representation of the Acquired Knowledge

The second new dimension concerns the format in which a concept may be expressed. There exist several formats [20], each of them offering a more or less detailed and explicit way of expression to define the characteristics of the acquired knowledge. For example, the *boundary* representation, which is typically encountered in the statistical tools, represents the acquired knowledge in the equation of a straight line separating the instance space into sub-spaces. The illustration given in Exhibit 6 shows the similarity with a linear regression.

In contrast with the statistical tools, research in machine learning is oriented towards other representations that better describe the acquired knowledge. The boundary representation, as shown in Exhibit 6, gives the equation of a line which splits positive and negative examples. No detail is given on the characteristics of the points below or above the line. A representation able to clearly define the characteristics of positive and negative examples offers far more information about the groups of points and therefore allows a more relevant interpretation of the concept. For example, in the *logic or classical view*, the classification function u is described in terms of conditions about the attributes. A new example belongs to the concept if its attribute values satisfy those conditions. As shown in Exhibit 7, the classification function is described in terms of conditions about the attributes *att1* and *att2*. The logic or classical view tries to characterize the points of the instance space present in the boundaries in terms of *att1* and *att2*. As illustrated in Exhibit 7a, an example belongs to the concept if its *att1* value is between a and b and its *att2* value is between c and d.

The *decision tree representation* may be considered as an extension of the *logic or classical view*. Exhibit 8 expresses on a decision tree the same concept as described in Exhibit 7b. The decision tree representation offers the advantage of showing a hierarchical structure among the attribute values. For example, the splitting of *att2* on a (top of the tree) appears decisive for the subsequent tests of the tree, whereas the splitting on m of this same attribute (bottom of the tree) is less likely to be considered in the classification of a new example. The decision tree representation has largely contributed to the success of the inductive learning method in numerous fields of application.[2] The next section is devoted to decision trees in order to explain and illustrate this concept representation as it relates to credit analysis.

Representation of the Acquired Knowledge in a Decision Tree

Consider the tree presented on Exhibit 9. This tree is composed of 6 nodes, each of them concerning an attribute, and 10 leaves or final nodes giving a final decision concerning the granting (+) or non-granting (-) of a credit. Each branch gives the attribute value to be considered in order to follow the correct path towards a final decision. A final leaf can be reached by one and only one path. In other words, a node has at most one parent node, but as many children as required by the final decision.

A decision tree is read from the top to the bottom and each *path* may be expressed in the form of a decision rule. For example, the final node marked '*' on Exhibit 9 corresponds to the following decision:

if there has been no appraisal of the project to be financed (att1=0) and
there is no information about the marketing prospect (att2=0) and
the credit applicant is active in the hotel-restaurant industry (att3=1) and
the applicant is male (att4=1),

then the credit is accepted (final leaf=+).

Each path represents an alternate line of reasoning and may be related to the rectangles in Exhibit 7. Exhibit 7a presents a widespread concept where each path of the tree represents one boundary in the instance space (1 rectangle of the instance space). Whereas Exhibit 7b illustrates only one boundary expressed into several paths (5 rectangles of the instance space). Each path of the decision tree represents a conjunction that approximates a portion of the total concept (one rectangle). The complete concept expressed in the tree corresponds to the set of all the decision paths leading to an alternate final classification.

Such a symbolic representation of the acquired knowledge in a decision tree provides the designer of a knowledge base with an explicit and intelligible tool. Each node symbolizes a test on a given attribute,[4] whether this attribute is quantitative or qualitative (i.e., symbolic). Each branch subsequent to a non-final node symbolizes the outcome of the test and symbolically leads the interpretation, according to the appropriate outcome. Each path symbolizes a complete line of reasoning and leads to a final decision. The entire tree symbolizes alternate lines of reasoning with their corresponding final decision. The position of an attribute is fixed on the tree and symbolizes a hierarchy among the set of attributes. Moreover, the final tree may be used directly as a decision support tool for classifying new unknown examples. The interpretation of a decision tree learned by induction is further discussed in part IV.

III A New Dimension for a Dynamic Updating of the Credit Granting Decision

The first new dimension emphasized in the previous chapter points out the specific impact of type I errors and type II errors as near-missed examples, on the accuracy of the learning process. As near-misses are located very close to the concept boundary, they nudge the inductive process towards the correct location of the boundary, as shown in Exhibit 5. It may, however, be argued that the credits that were initially accepted by the Credit Committee but failed in the aftermath (type I errors) should not have been accepted, or should not have been contained in the concept. In the same way, it may be argued that the credits that were initially rejected by the Credit Committee but eventually happened to be profitable for a competing financial institution (type II errors) should not have been rejected, or should have been contained in the concept. Exhibit 10 illustrates how, over time, the outcome of an initially accepted/rejected credit may change. An accepted credit at time t_1 may turn failed at time t_{1+i} (type I error). Similarly, a rejected credit at time t_1 may become a profitable credit in a competing financial institution at time t_{1+i} (type II error).

Therefore, the unique impact of type I and type II errors on the learning process results in their informative content that allows an updating of the credit decisions over time and therefore, an updating of the initial concept boundary. Exhibit 11 illustrates this updating process by showing the position of the updated concept boundary among near-misses. The updated boundary contains correctly accepted credits (+), as well as type II errors (-'), i.e., the shaded groups of Exhibit 10. Type I errors (+') are now outside the boundary, together with the correctly rejected credits (-). The updated boundary represents a more informed granting decision which does not repeat the errors initially made by the Credit Committee. The type I and type II errors still remain very close to the updated boundary and keep playing their role of near-misses, i.e., they nudge the learning process towards a more informed and therefore more accurate concept definition.

Dynamic Updating of the Boundary Position over Time

The repositioning of the near-misses according to an updated credit granting decision over time requires the detection of type I and type II errors.

Type I errors are observable as they represent previously accepted credits that finally failed. The initial decision, as illustrated in Exhibit 5, positions these errors as positive examples very close to the boundary (+'), while the Dynamic Updating Process excludes them from the boundary and keeps them among the negative examples, as shown in Exhibit 11. The detection of type II errors is more problematic as such credits are almost impossible to trace. However, the Dynamic Updating Process should follow the same method as for type I errors, but in the opposite sense: during updating, they should be included in the boundary, i.e., among the positive examples, as shown in Exhibit 11.

As a matter of fact, the characteristics that exclude the type I errors from the boundary may be learned by induction in order to discover what differentiate these failed credits from the other positive examples. In this context, the *conceptual* difference between credits is considered, which is different from a *mathematical* difference between points in a multidimensional space. The same difference can then be assessed on the rejected credits which will be included in the boundary if they are recognized as similar enough to be positive examples. Exhibit 12 illustrates that learning and shows the minimum difference (d) that should exclude an example from the positive group. The difference d is then used as a cut-off point to make a decision concerning the rejected credits. A rejected credit that is very similar to the positive examples (difference $< d$) is considered as a type II error and is contained in the boundary. As a result, a new boundary is defined according to which an updated decision may be modeled.

As the new boundary represents a more informed credit granting decision, the resulting concept learned by induction should be more accurate and more stable. Further experiments should be conducted on real data to confirm that hypothesis.

IV A New Dimension for a Global Interpretation of the Results

The tree-based inductive learning approach appears as an alternative in the design of a decision support tool in credit risk analysis. The literature concerning the application of the inductive approach in various management classification problems successfully compares the accuracy of a decision tree with the accuracy of common statistical tools, such as logit, probit, or MDA.[3] Accuracy evaluation appears particularly important when researchers try to compare the performance of various classification models. Although it is considered as a relevant and necessary evaluation parameter, it is not the only criterion to take into consideration. As Breiman points out, "*an important criterion for a good classification procedure is that it not only produces accurate classifiers (within the limit of the data) but that it also provides insight and understanding into the predictive structure of the data* "[7]. As a matter of fact, the learning approach describes the knowledge related to the credit granting decision in a comprehensive and complete formalism that offers a unique advantage over the results commonly provided by a statistical classification tool. An interpretation of the output decision tree in terms of credit risk elements provide insights about an *underlying predictive structure of the data* [7] as well as about a financial theory justifying the credit assessment process.

The instability of the classification results is, however, a major problem when applying the inductive learning approach to real data. Experiments prove that decision trees generated by induction are greatly influenced by the characteristics of the training sample. A slightly different training sample may generate a markedly different decision tree. The differences concern not only the occurrence of specific attributes in the trees, but also the positions of the attributes in the trees. In other words, a decision tree generated by induction quickly overfits the training sample and is influenced by noisy input data.

These observations raise the problem of selecting a single decision tree which would offer the most relevant insights about the underlying predictive structure of the data. A selection based on accuracy does not assure that the chosen tree will reflect the global financial theory justifying the credit assessment process. Moreover, one of the unique observations of our experiments shows that one can never be certain that the most accurate tree is contained in a given set of induced trees. Therefore, it becomes apparent that a global interpretation process be developed in order to summarize and concentrate the relevant information provided by a set of original trees, while reducing noisy and overfitting negative effects. The *global tree interpretation* proposed in the following section is used for that purpose.

The Global Tree Interpretation: Insights about the Predictive Structure of the Data

As illustrated in Exhibit 13, the *Global Tree Interpretation Process* (GTip) is performed on a set of *i original* trees ($tree_1$, $tree_2$, ..., $tree_i$), each of them generated from a different training sample randomly selected from the same original data set. The *original* trees are unstable not only regarding their accuracy (error rate on testing sample), but also regarding their general outlook (position of the attributes on the tree). GTip summarizes and concentrates the relevant information provided in the original trees. As a result, a *final global tree* is generated which reduces noise and overfitting negative effects present in the original trees.

At first, GTip retains the most frequently appearing attributes in the original trees. The attributes appearing with a frequency higher than 50% are called *primary*; the attributes appearing with a frequency lower than 50% but higher than 25% are called *secondary*; an attribute appearing with a frequency lower than 25% is considered as a noise effect. The 50% and 25% cutoff values were determined through Jackknife experimentation.

Secondly, the *horizontal position of an attribute* is considered. Consider Exhibit 14, where two situations are proposed: a concept widespread over the hypothesis space (14a) and a concentrated concept (14b). In both cases, one may assume that the shaded surface is the easiest portion of the concept that will be discovered by the inductive process. As a matter of fact, the points of the instance space are most likely concentrated on those surfaces. As explained earlier concerning Exhibit 7, one path of the tree represents one boundary as in Exhibit 14a, or one rectangle approximating a portion of the total concept, as in Exhibit 14b. The shaded surfaces will thus correspond to the path of the tree followed by the largest portion of the training examples. We call such a path, the *main path* of the tree. The other surfaces of the hypothesis space not covered by the main path (non-shaded surfaces in Exhibit 14a and 14b) will be considered as *alternate paths*. Attributes appearing on the main path are called *major* attributes, as they help in discriminating the largest part of the examples. Attributes belonging to an alternate path are called *minor* attributes. The root node implicitly belongs to the main path.

In the framework of a credit risk assessment process, the main path may be considered as the usual analysis procedure followed for the majority of the credit applications. The attributes taken into consideration for those routine credit applications would cover common financial criteria used in credit analysis. Therefore, these attributes are recognized as *major* elements in a normal credit analysis. The alternate paths would involve analytical processes used for unusual credit applications. The attributes taken into consideration for this minority of credits may be less common and more dependent on case-by-case analyses. Therefore, these attributes are recognized as *minor* elements in a routine credit analysis.

Finally, GTip calculates the *average level* on which primary and secondary attributes appear among the original trees.

By compiling, for each attribute, its positions in each tree according to the criteria mentioned above, a global pattern emerges. The final *global tree* retains this global pattern, while avoiding as much noise and overfitting effects as possible. The resulting global tree may reveal, therefore, relevant insights about the global underlying predictive structure of the data. The algorithm followed by GTip is detailed in Appendix A.

For instance, consider a set of 10 trees induced from the same original set of examples. Each example is defined in terms of 23 attributes. Those attributes are classified by family (F1 to F7), which reflects their conceptual relationships. The position of each attribute in each tree is given in the first 10 rows of Exhibit 15: a value 1 informs that the attribute is the top node in the corresponding tree; a bold value corresponds to a main path position.

The row entitled *occurrence*, which gives the number of times an attribute appears in the original trees, allows the compilation of the attribute status as *primary* (*P* if *occurrence* >5) or *secondary* (*S* if 2<*occurrence*<5). The attributes appearing 2 times or less are considered as noise and disregarded (*). The next row, entitled *main path*, gives the percentage of time an attribute appears on a main path which, in turn, allows the definition of the attribute status as *major* (*M* if *main path* >= 50%) or *minor* (*m* if *main path* < 50%). Finally, the row entitled *aver. level* informs about the attribute's average position on the trees.

As a result, the last row (*status*) is considered in order to build a final global tree, as shown in Exhibit 16. Four major attributes are used to build the main path (F17, F42, F61, F72), with the *primary* attributes prevailing over the *secondary* attributes. The root position is occupied by attribute F61. Attribute F42 stands on the second level of the main path. Attributes F16, and F72 are in conflict for the third level of the tree: as attribute F72 is primary, it is kept on the main path; an alternate path, on which F16 appears, is built from attribute F42. Attribute F16 is followed by its "conceptual fellow," attribute F11, while attribute F17 completes the fourth level of the main path. The final global tree has a size comparable to the average of the original trees (length = 4 and width = 2), as shown in Exhibit 16.

The final global tree given in Exhibit 16 concerns the credit risk analysis of small Belgian businesses. Although the purpose of this paper does not concern the details of an application of the GTip method to small Belgian businesses, Exhibit 16 gives a final global tree generated from real data and shows an interesting underlying structure of the data. The main path, which is followed in the majority of the cases, reveals a hierarchical structure of attributes related to the applicant's capacity to repay the credit (F61), the guarantees (F42), the marketing prospect that the bank may expect from the credit (F72), and the applicant's industry (F17). The alternate path considers the same first two attributes (F61 and F42), then concentrates on the previous activities (F16) and the type of the applicant (F11) . This underlying structure of the data was presented to the bank's credit officers, who recognized it as a relevant representation of their risk analysis process. As financial information provided by small size applicants is almost inexistant, the credit analysis process relies on other qualitative information which were discovered by the GTip method, such as the guarantees, the applicant's industry, or the type of the applicant.

Conclusion

The tree-based inductive learning approach has been presented as an alternative to statistical methods for designing a decision support tool in credit risk analysis. A brief review of the learning methodology allowed the emphasis of two new dimensions of the approach that have not been previously considered in the literature.

The first new dimension points out the specific impact of the type I and type II errors, as near-misses, on the accuracy of the inductive learning process. The literature contends that near-misses nudge the learning process towards a more accurate definition of the boundary between positive and negative examples. Such a specific impact of type I and type II errors is unique and has not been examined in credit analysis. A Dynamic Updating Process is proposed which relocates the boundary between type I and type II errors in order to define a more informed credit granting decision and learn a more accurate concept.

The second new dimension takes advantage of the representation of the results in a decision tree in order to solve the problem of instability in classification results. The decision tree representation used by the inductive learning approach offers a unique hierarchical structure of the attributes taken into consideration in credit analysis. However, unstable results raise the problem of selecting one decision tree from a set of different trees. The global tree interpretation that is proposed globalizes the relevant content of a set of original trees while reducing sources of instability, i.e., noise and overfitting effects. Thus the global interpretation process allows a better understanding of the underlying structure of the data.

These new dimensions, which have been defined in theory, present some promising contributions to the problem of designing a decision support tool in credit risk analysis. However, experimental results are necessary to further define the exact impact of type I and type II errors and to further improve the efficiency of the Dynamic Updating Process and the GTip method. Several applications on real data are currently under process to investigate this challenging subject [12]. Moreover, an automated version of the GTip method will allow the application on large data sets. Following the results of these experiments, an improvement and completion of the basic principles defined in this paper will be reported in a forthcoming publication.

Footnotes

1 Research sponsored by a Doctoral Fellowship of the Intercollegiate Center for Management Science, Brussels, Belgium

2 See 4, 5, 6, 8, 9, 11, 13, 18, 21, 22

3 Several measures are used to select the attributes that classify positive and negative examples in the best way. For a review of these measures, see [7]. The most commonly used measure, called *entropy*, was introduced by Quinlan in 1983 [Quinlan, J.R., "Learning Efficient Classification Procedures and their Application to Chess End Games," *Machine Learning. An Artificial Intelligence Approach*, R.S. Michalski, J.G. Carbonell, and T.M. Mitchell (Eds.), Morgan Kaufmann Publishers, Inc., Palo Alto, CA, 1983].

4 Advanced learning algorithms are now able to relate a node of the tree to a higher relation among several attributes. The "constructive" learning approach is currently being investigated in the framework of credit risk assessment and bankruptcy prediction. Quite unexpected multivariate relations are learned which give relevant insights about uncommon ways to understand some aspects of a firm's financial risk.

References

[1] The American Heritage Dictionary (Second College Edition), 1985.

[2] Barr, A. and E.A. Feigenbaum, *The Handbook of Artificial Intelligence - Volume 1*, William Kaufmann, Inc., Los Altos, CA, 1981.

[3] Barr, A. and E.A. Feigenbaum, *The Handbook of Artificial Intelligence - Volume 2*, Addison-Wesley, Readings, MS, 1982.

[4] Bouwman, M.J., "Expert vs Novice Decision Making in Accounting: A Summary," *Accounting Organization and Society*, Vol. 9, No. 3/4, 1984, pp. 325-327.

[5] Bouwman, M.J., "Towards Expert Systems: The Analysis of Expert Financial Behavior," Working Paper, College of Business Administration, University of Oregon, August 1985.

[6] Bouwman, M.J., "How Do Financial Analysts Make Decisions? A Process Model of the Investment Screening Decision," *Accounting Organizations and Society*, Vol. 12, No. 1, 1987, pp. 1-29.

[7] Breiman, L., J. H. Friedman, R. A. Olshen, and C. J. Stone, *Classification and Regression Trees*, Wadsworth International Group, Belmont, 1984.

[8] Carter, C. and J. Catlett, "Assessing Credit Card Applications Using Machine Learning," *IEEE Expert*, Fall 1987, pp. 71-79.

[9] Chandler, J.S., T. Liang, and I. Han, "An Empirical Comparison of Probit and ID3 Methods for Accounting Classification Research," Dpt of Accountancy, College of Commerce and Business Administration, UIUC, August 1989.

[10] Cohen K.J., T.C. Gilmore, and F.A. Singer, "Bank Procedures for Analyzing Business Loan Applications," *Analytical Methods in Banking*, K.J. Cohen and F.S. Hammer (Eds), Richard D. Irwin, Homewood, IL, 1966, pp. 219-249.

[11] Currim, I.S., R.J. Meyer, and N.T. Le, "Disaggregate Tree-structured Modeling of Consumer Choice Data," Graduate School of Management, University of California, Los Angeles, CA, August 1986.

[12] Gentry, J. A., M. J. Shaw, A. C. Tessmer, and D. T. Whitford, "Predicting Loan Risk with an Inductive Learning Approach," Working Paper, College of Commerce and Business Administration, University of Illinois, 1992, 47 pages.

[13] Han, I., *The impact of Measurement Scale on Classification Performance of Inductive Learning and Statistical Approaches*, doctoral dissertation, Dept of Accountancy, UIUC, August 1990

[14] Hester, D.D., "An Empirical Examination of a Commercial Bank Loan Offer Function," *Analytical Methods in Banking*, K.J. Cohen and F.S. Hammer (Eds), Richard D. Irwin, Homewood, IL, 1966, pp. 178-215.

[15] Hovland, C.I., "A 'Communication Analysis' of Concept Learning," *Psychology Review*, No. 59, 1952, pp. 461-472.

[16] Hovland, C.I. and W. Weiss, "Transmission of Information Concerning Concepts through Positive and Negative Instances," *Journal of Experimental Psychology*, Vol. 45, No. 3, 1953, pp. 175-182.

[17] Hunt, E. B., J. Marin, and P. J. Stone, *Experiments in Induction*, Academic Press, New York, 1966.

[18] Messier, W.F. and J.V. Hansen, "Inducing rules for Expert System Development: An Example Using Default and Bankruptcy Data," *Management Science*, Vol. 34, No. 12, December 1988, pp. 1403-1415.

[19] Michalski, R.S., "A Theory and Methodology of Inductive Learning," *Machine Learning: An Artificial Intelligence Approach*, R.S. Michalski, J.G. Carbonell, and T.M. Mitchell (Eds.), Symbolic Computation, Springer-Verlag, Berlin, 1983, pp. 83-134.

[20] Rendell, L. and H. Cho, "Empirical Concept Learning as a Function of Data Sampling and Concept Character," Technical Report R-88-1410, Dpt of Computer Science, UIUC, May 1988.

[21] Shaw, M.J. and J.A. Gentry, "Using an Expert System with Inductive Learning to Evaluate Business Loans," *Financial Management*, Autumn 1988, pp. 45-56.

[22] Shaw, M. J. and J. A. Gentry, "Inductive Learning for Risk Classification," IEEE Expert, February 1990.

[23] Smoke, K.L., "Negative Instances in Concept Learning," *Journal of Experimental Psychology*, No. 16, 1933, pp. 583-588.

[24] Winston. P.H., "Learning Structural Description from Examples," *The Psychology of Computer Vision*, Mc Graw-Hill, N-Y, 1975, pp. 157-209.

Appendix A

The Global Tree Interpretation Process Algorithm

Given

- a set of examples described with a list of n attributes
- a set of m trees generated by induction from a subset of examples selected at random
- $i = 1,...,n$ and $j = 1,...,m$

Step 1. For each attribute$_i$, compile

a. the level on which it appears on each tree$_j$

b. its presence on a main path or on an alternate path of each tree$_j$

Step 2. Define the final status of attribute$_i$ as

a. a primary (secondary) attribute if it appears in more than 50% (between 50% and 25%) of the original trees

b. a major (minor) attribute if it appears in more (less) than 50% of the main paths of the original trees

and calculate the average level on which the attribute appears.

Step 3. The resulting underlying structure of the data is built per level, starting from the root node of the tree, and according to the following rules:

- primary attributes prevail on secondary attributes;
- major attributes appear on the main path;
- minor attributes appear on alternate paths;
- an alternate path is started from a given node, as soon as several attributes are in conflict for the subsequent level.

The final global tree should match the average width (number of final nodes) and length (number of nodes on the longest path) of the original trees.

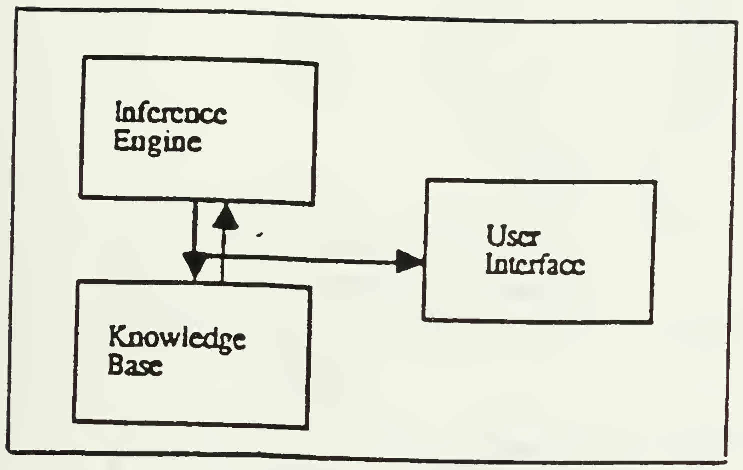


Exhibit 1.: Main modules of an expert system

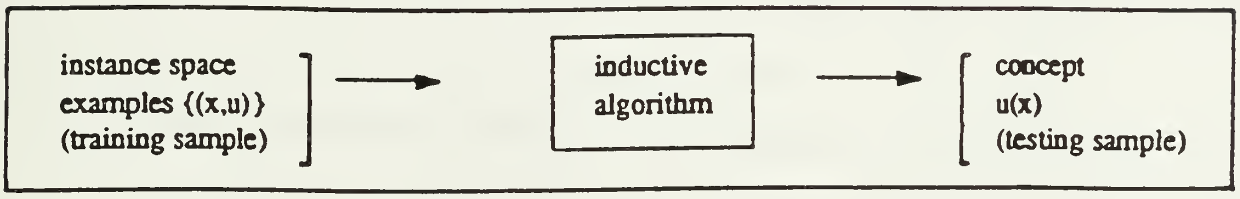
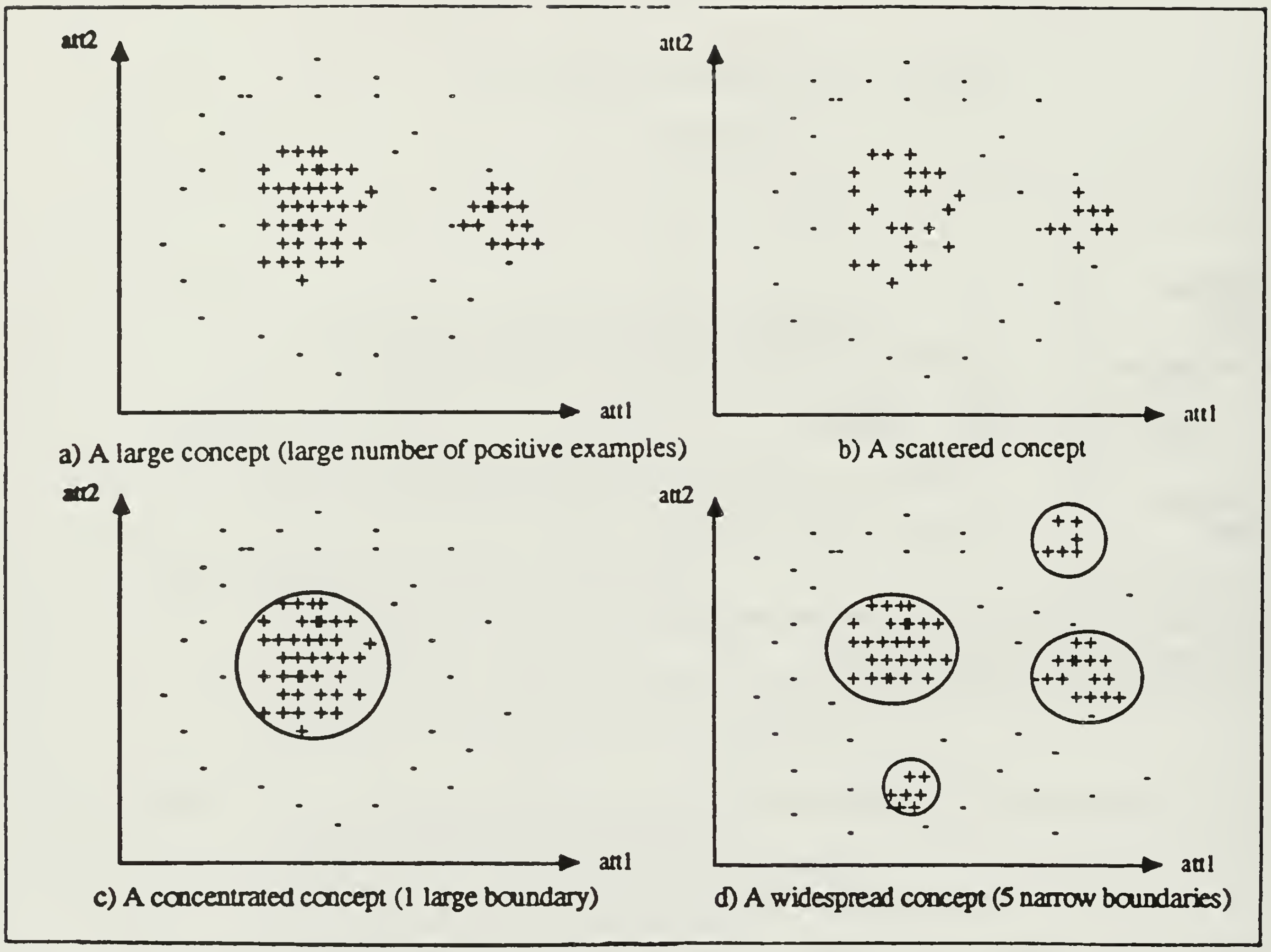


Exhibit 2.: General description of a learning algorithm



a) A large concept (large number of positive examples)

b) A scattered concept

c) A concentrated concept (1 large boundary)

d) A widespread concept (5 narrow boundaries)

Exhibit 3.: Outlook of a concept in the instance space

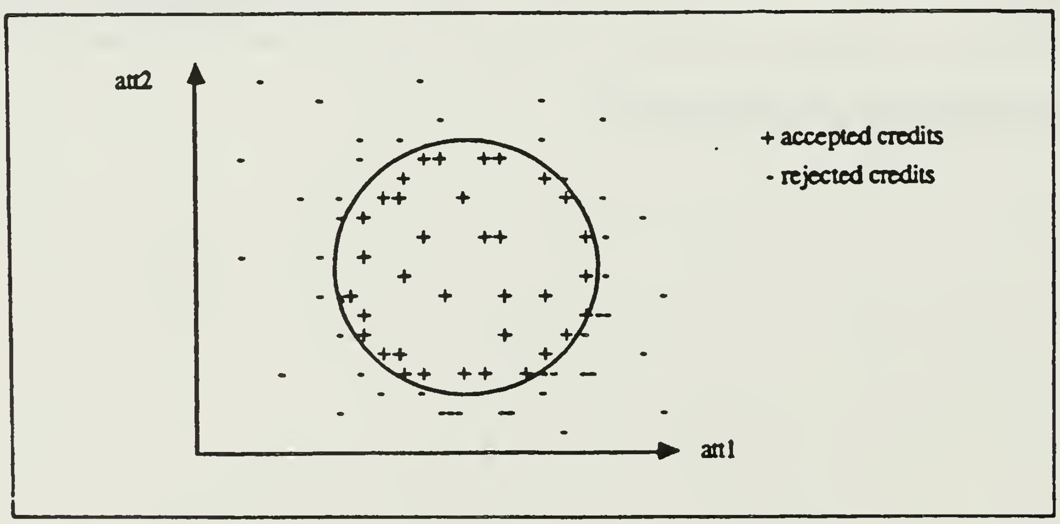


Exhibit 4.: Boundary concentration of positive and negative examples

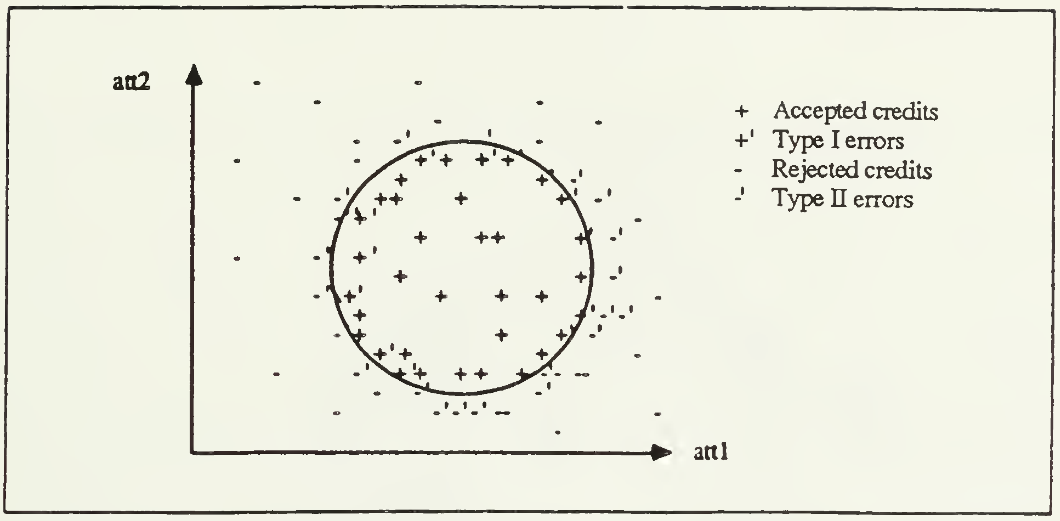


Exhibit 5.: Type I errors and type II errors as near-misses

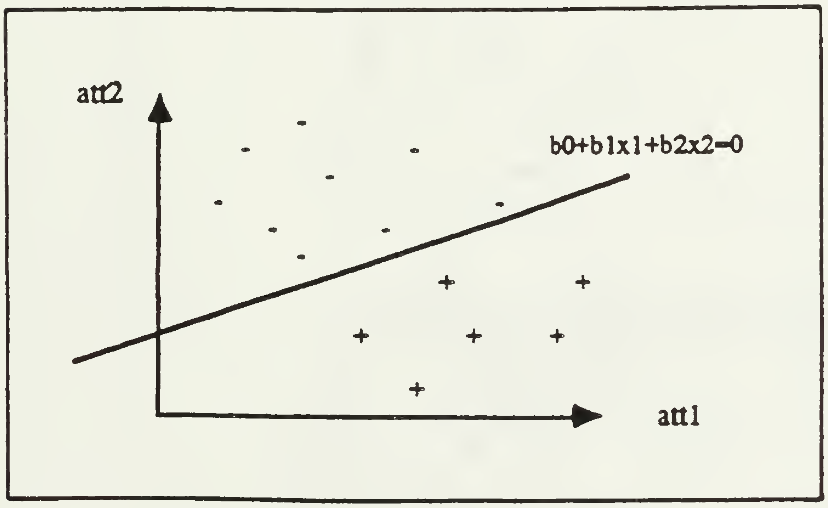


Exhibit 6.: Boundary representation

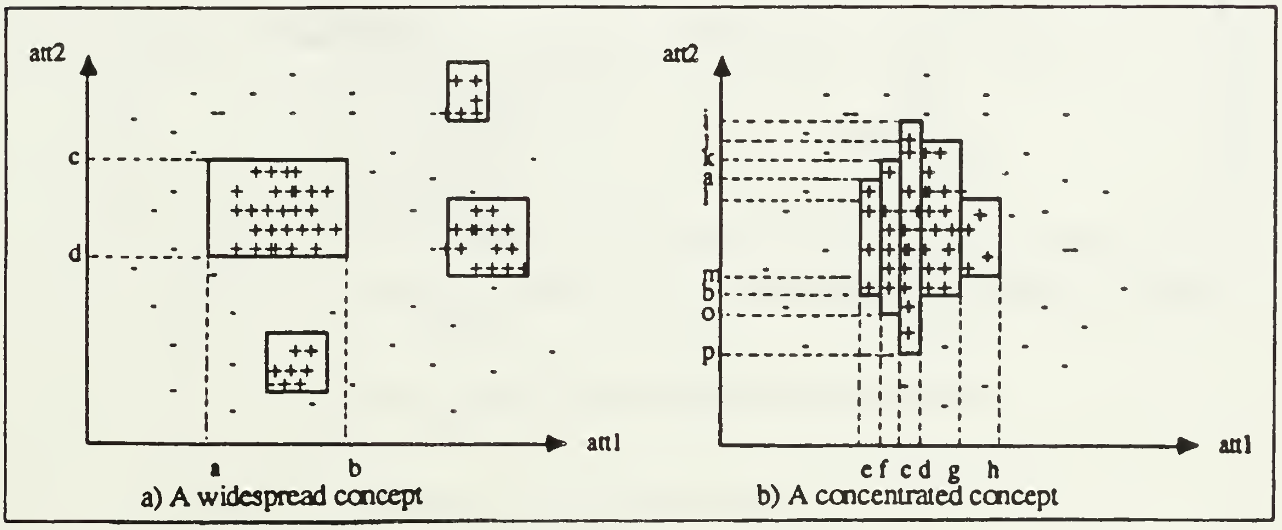


Exhibit 7.: Logic or classical view representation

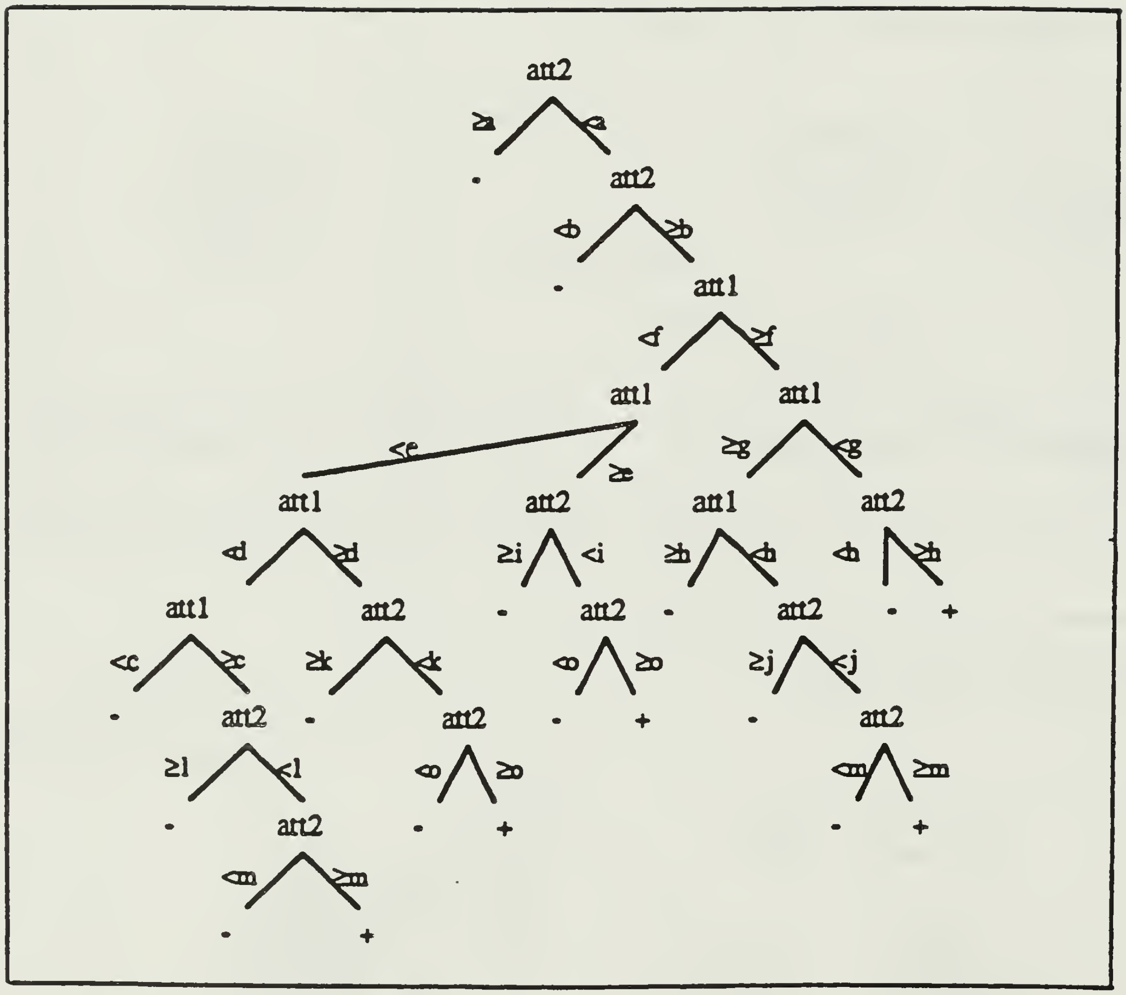
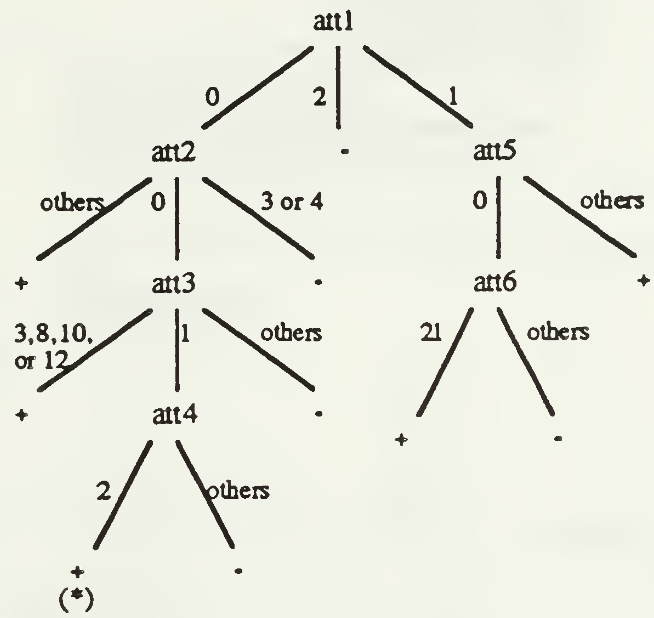


Exhibit 8.: Decision tree representation



where

att1 = Result of the appraisal of the project to be financed
- 0 - Without appraisal
- 1 - Positive result
- 2 - Negative result

att2 = Marketing prospect
- 0 - Without prospect
- 3 - Customer coming from a competitor
- 4 - Not obvious

att3 = Industry
- 1 - Hotel -restaurant
- 3 - Building
- 8 - Health
- 10 - Services
- 12 - Media

att4 = type of the applicant
- 2 - Male

att5 = Does the applicant have an education in management ?
- 0 - No

att6 = Previous activities of the applicant
- 21 - Related education

Exhibit 9.: Example of a decision tree

t1 \ t1-i	Correct decisions	Incorrect decisions
Accepted credits	positive examples +	Type I errors +'
Rejected credits	negative examples -	Type II errors -'

positive examples

negative examples

→ *Initial Decision* (Exhibit 5)

Exhibit 10.: Evolution of the credit granting decision over time

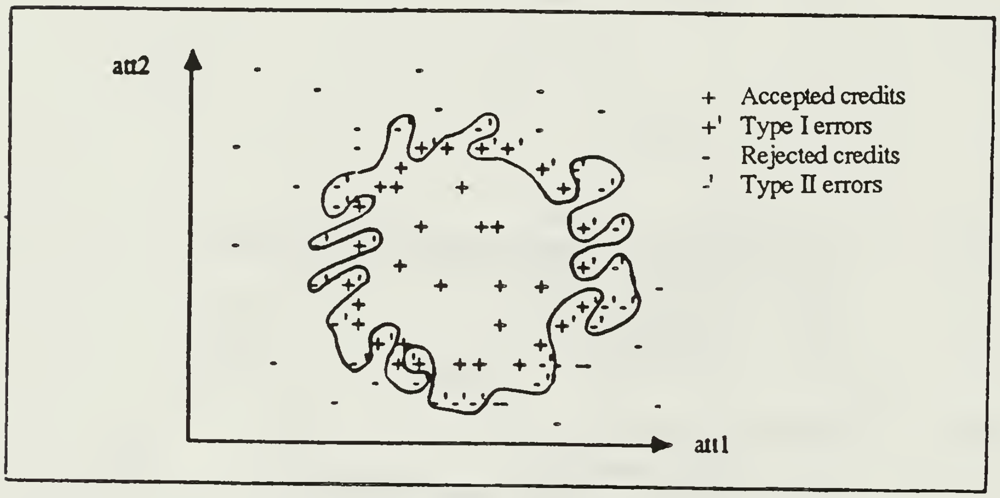


Exhibit 11.: Updated concept boundary

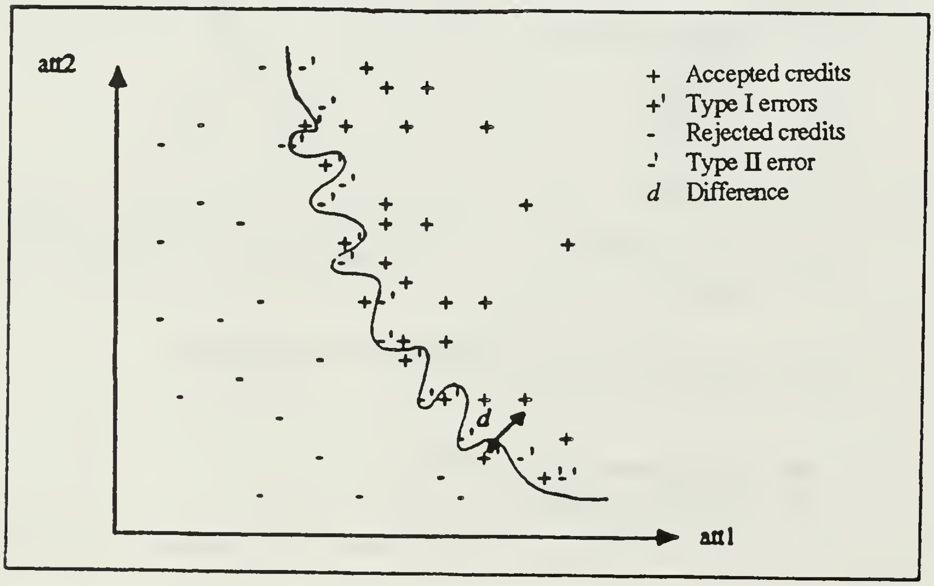


Exhibit 12.: Minimum distance between positive and negative examples

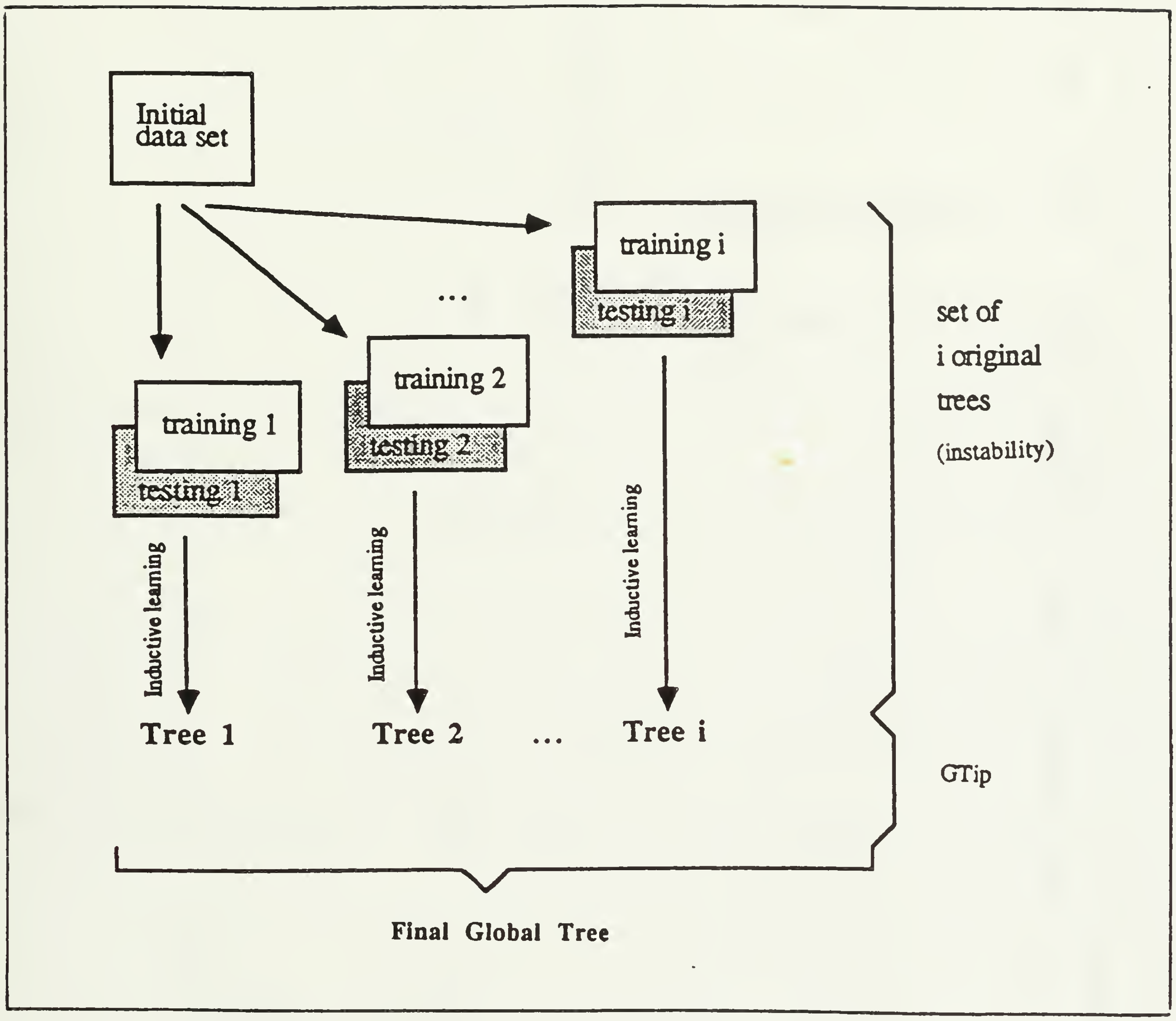


Exhibit 13.: Global tree interpretation

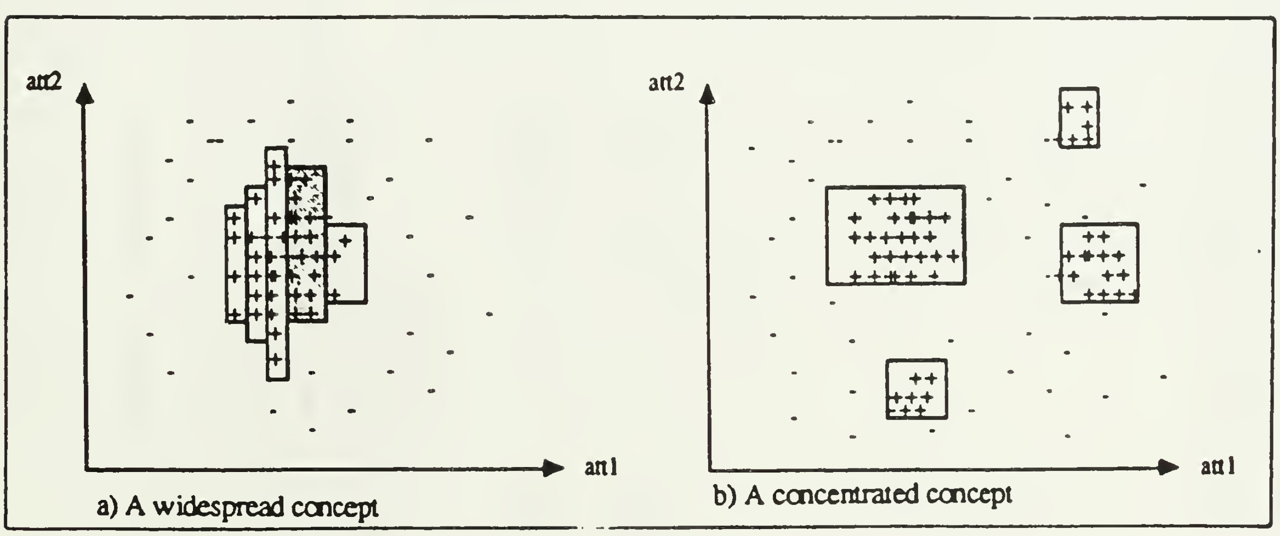


Exhibit 14.: Main path of a tree

	F_{11}	F_{12}	F_{13}	F_{14}	F_{15}	F_{16}	F_{17}	F_{18}	F_{21}	F_{31}	F_{32}	F_{33}	F_{41}	F_{42}	F_{43}	F_{44}	F_{45}	F_{51}	F_{61}	F_{62}	F_{63}	F_{71}	F_{72}	% error	length	width
tree 1						3													1				2	29	3	1
tree 2					2														1	4	3		3	26	4	3
tree 3	3						5												1		2		4	31	5	1
tree 4	4													3			1							35	4	1
tree 5	4					3	3							1		2							2	45	4	2
tree 6		2				4	2							3					1				4	30	5	3
tree 7		2											3	1						4			3	25	4	2
tree 8							5						3	2					1			4	3	21	5	2
tree 9	4					2								3				2	1			3	2	30	4	4
tree 10							3							1									2	17	3	2
Occurence	4	2	-	-	1	4	5	-	-	-	-	-	2	7	-	1	1	1	6	2	2	2	9	aver. error	aver. size	
main path	25	*	-	-	*	25	80	-	-	-	-	-	*	86	-	*	*	*	100	*	*	*	89	30	4.1	2.1
aver. level	3.8	*	-	-	*	3	3.6	-	-	-	-	-	*	2	-	*	*	*	1	*	*	*	2.8			
status	mS	*	*	*	*	mS	MP	-	*	*	*	*	*	MP	*	*	*	*	MS	*	*	*	MP			

where
m = minor attribute
M = major attribute
S = secondary discriminant attribute
P = primary discriminant attribute
BP = before pruning
AP = after pruning

Exhibit 15.: Example of a global tree interpretation

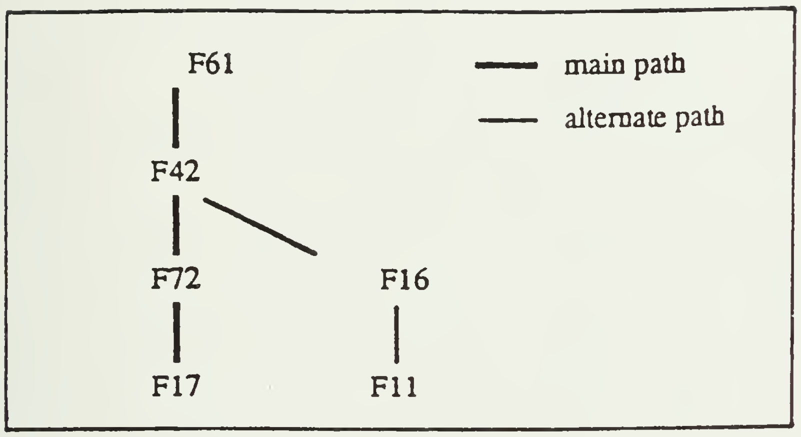


Exhibit 16.: Example of a final global tree