Decision Making by a Multiple-Rule Classifier: The Role of Rule Qualities

Ivan Bruha
McMaster University, Canada

INTRODUCTION

A rule-inducing learning algorithm yields a set of decision rules that depict knowledge discovered from a (usually large) dataset; therefore, this topic is often known as knowledge discovery from databases (KDD). Any classifier (or, expect system) then can utilize this decision set to derive a decision about given problems, observations, or diagnostics. The decision set (induced by a learning algorithm) may be either of the form of an ordered or unordered set of rules. The latter seems to be more understandable by humans and directly applicable in most expert systems, or generally, any decision-supporting one. However, classification utilizing the unordered-mode decision set may be accompanied by some conflict situations, particularly when several rules belonging to different classes match (are satisfied by, "fire" for) an input to-be-classified (unseen) object. One of the possible solutions to this conflict is to associate each decision rule induced by a learning algorithm with a numerical factor, which is commonly called the *rule quality* (An & Cercone, 2001; Bergadano et al., 1988; Bruha, 1997; Kononenko, 1992; Mingers, 1989; Tkadlec & Bruha, 2003).

This article first briefly introduces the underlying principles for defining rules qualities, including statistical tools such as contingency tables and then surveys empirical and statistical formulas of the rule quality and compares their characteristics. Afterwards, it presents an application of a machine learning algorithm utilizing various formulas of the rule qualities in medical area.

BACKGROUND

There are a few paradigms of extracting knowledge from databases. One commonly used paradigm is called *divide-and-conquer*. It is widely utilized by the family of top-down induction of decision trees (TDIDT) learning algorithms that induce decision trees. One of its first pioneers is the ID3 algorithm (Quinlan, 1986), but currently mostly used and well-known members are C4.5 and C5.0 algorithms (Quinlan, 1994).

Another widely used strategy in learning uses the *covering* paradigm that generates sets of decision rules; the AQx and CNx families as well as C4.5Rules or Ripper are the well-known algorithms generating knowledge bases as sets of decision rules. A decision rule has the following general form:

R: if *Cond* then class is *C*

Here *R* is the name of the rule, *Cond* represents the condition under which the rule is satisfied (fires), and *C* is the class of the rule, that is, an unseen object satisfying this condition is classified to the class C.

As we already stated a *decision set* of (decision) rules may be either ordered or unordered. To understand the situation better, here is an example of a very simple ordered decision set:

if outlook=overcast then class is +; quality=0.889
else if windy=false && humidity=normal then class is +; quality=0.867
else if humidity=high && outlook=sunny then class is -; quality=0.920
else if windy=true && outlook=rain then class is -; quality=0.880
else if true then class is +; quality=0.844

This case uses the well-known "weather" problem introduced by Quinlan (1986). Here windy, humidity, outlook, and temperature are attributes with two classes + and -. As we can see the order character of this decision set is arranged by if .. else if statement; notice that the last statement else if true then in fact represents else.

Corresponding unordered decision set induced by the same covering learning algorithm looks as follows:

if humidity=high && outlook=sunny
then class is -; quality=0.920
if outlook=overcast
then class is +; quality=0.889
if windy=false && humidity=normal
then class is +; quality=0.889
if windy=true && outlook=rain
then class is -; quality=0.880
if windy=false && outlook=rain
then class is +; quality=0.867
if humidity=normal && temperature=mild
then class is +; quality=0.844
if true
then class is +; quality=0.714

As we see the unordered decision set uses only the if statements. If the decision set is *ordered*, then classifying an input unseen (to-be-classified) object is quite straightforward: the classifier goes through the ordered list of rules from its beginning and looks for the first rule that matches (is satisfied by, "fires" for) the given input object; it is then categorized into the class attached to the rule.

However, the important and natural way seems to be the *unordered* mode that is utilized in various expert and decision-making systems. The classification utilizing an unordered set of decision rules exhibits a significant deficiency, not immediately apparent. When classifying an unseen object by a decision-making system consisting of an unordered set of decision rules, the system has to go through the entire decision set of rules; some rules will be satisfied (will fire), some not. Three cases are then possible:

1. If the unseen object satisfies one or more rules of the same class, then the object is categorized to the class assigned to the rule(s).
2. If the unseen object is not covered by any rule, then either the classifier informs the user about its inability to decide ("I do not know"), or the object is assigned by default to the majority class in the training set, or some similar techniques are invoked.
3. Difficulty arises if the object satisfies more rules assigned to different classes. Then some schemes have to be applied to assign the unseen input object to the most appropriate class.

One possibility to clarify the conflict situation (case 3) is to associate each rule in the decision scheme of the classifier with a numerical factor that can express its properties and characterize a measure of belief in the rule, its power, predictability, reliability, likelihood, and so forth. A collection of these properties is symbolized by a function commonly called the *rule quality*. This characteristic has been studied and applied in many research projects. One of the first studies was done by Bergadano et al. (1988). A similar approach can be found in Kononenko and Bratko (1992) and Kononenko (1992). A systematic survey of formulas of rule qualities is presented in Bruha (1997). A methodological and theoretical approach to this characteristic is delivered by Tkadlec and Bruha (2003).

Afterwards, when we choose a formula for the rule quality, we also have to select a *scheme for combining* these qualities. To solve the aforementioned conflict case, the qualities of fired rules of the same class have to be combined using a certain scheme (formula). Consequently, the rule-quality combination with the maximum value will determine the class of the unseen object. A survey of methods for the rule combination can be found, for example, in Kononenko (1992) and Bruha and Tkadlec (2003).

RULE QUALITY: CHARACTERISTICS, FORMULAS, AND COMBINATION SCHEMES

As we already stated, one eventuality in how to solve the multiple-rule decision-making problem is to associate each rule in the decision scheme (knowledge base) of a classifier with the *rule quality*. Formulas for the rule quality have been studied and tested in several aritcles (An & Cercone, 2001; Bergadano et al., 1992; Brazdil & Torgo, 1990; Bruha, 1997; Torgo, 1993). The previous conflict is then actually worked out by combining the qualities of rules that fire for (are satisfied, match) a given input object; the object is then assigned to the class for which the *quality combination* reaches the maximum.

We now discuss the general characteristics of any formula of the rule quality. The first feature required for the rule quality is its *monotony* (or, more precisely, *nondecreasibility*) towards its arguments. Its common arguments are the *consistency* and *completeness* factors of decision rules. Consistency of a decision rule