Extracting Classification Variable Precision Rules
From Induced Decision-Exception Trees

Nabil M. Hewahi
Computer Science Department
Islamic University of Gaza-Palestine
nnewahi@iugaza.edu

Abstract - Classification by decision trees induction is one of
the very well known methods for classification in data mining.
In this paper we present an algorithm for inducing a decision
tree and extracting Censored Production Rules (CPR) from it.
CPR is based on Variable Precision Logic (VPL) in which
certainty varies, while specificity stays constant. Systems that
are based on Censored Production Rules have numerous
applications in situations where decisions must be taken in real
time with the presence of uncertain information. The extracted
CPRs are not only used in prediction but also in decision
support systems as well as systems which need real time
response. The proposed algorithm is based on choosing the
attribute with the most frequently occurrence of the attribute
value as a distribution attribute (branching attribute). Some of
the induced rules do not give any specific classification because
they refer to undecided cases (the classification is called
'unknown'). Such rules are good to indicate that more
information and cases are needed. The extracted rule must
cover a number of cases which exceed any other cases for the
class attribute. Those other cases will be as censors for the
extracted rule.

Keywords: Decision trees, variable precision logic,
classification rules, censored production rules.

1 Introduction

In this section, we shall give a brief background on
variable precision rules and decision trees induction.

1.1 Variable precision Rules

The standard rule structure is very well known in the
area of expert systems and data mining. The structure of
standard rule is (<IF condition THEN action>). An
extension of standard production rule, Michalski and
Winston [16] proposed the censored production rule
(CPR) of the form (<IF condition THEN action UNLESS
censor>) as an underlying representational and
computational mechanism to enable logic based systems
to exhibit variable precision logic (VPL) in which
certainty varies, while specificity stays constant. The form
of CPR is P → D ∣ C, where P is the premise, D is the
decision, and C is the censor. The premise is a
conjunct of literals; the decision is a single literal; and
the censor is a disjunction of literals. CPRs embody both
object level and control level information. The object
definition is false most of the time; we have certain
expectations concerning the character of inferences made
with such rules. These expectations may be used to
control the inferences. To understand the implication of
CPR, Michalski and Winston [16] presented a
quantitative definition for it, where two parameters γ and
δ have been introduced. A CPR is then written:

P → D | C (γ ∨ UNK) : γ, δ,

where γ = prob(P→D), certainty of P → D. When it is
not known either C1 or UNK holds (UNK means not yet
known censor conditions). The implication P → D is
with certainty 1, when (C1 v UNK) is known to be
false. When δ = prob(D/P& C1), it is the certainty
that P implies D, when C1 is true. Obviously the a priori
certainty of (C1 v UNK) must be equal to or smaller
than the a priori certainty that UNK. Therefore, γ ≤ δ.
Note that γ = 1 if it is certain that there are no conditions
in the censor other than C1. An example of CPR is as
below:

IF Working_Day → John_in_office
|, John_is_sick, John_on_leave

Extensive research have been performed on VPL systems
[1-3][7-8][15].

1.2 Decision trees and related algorithms

A decision tree is a flow-chart-like tree structure,
where each internal node denotes a test on an attribute,
each branch represents an outcome of the test, and leaf
nodes represent classes or class distributions. The top
most move in a tree is the root node. A very well known
algorithm for decision tree induction is ID3 [18][19].
The basic decision tree induction algorithm requires all
attributes to be categorical. Many enhancements to the
algorithm have been done and incorporated into C4.5
algorithm [20]. ID3 and C4.5 algorithms have been well
established for relatively small data sets. Efficiency and
scalability become issues of concern when these
algorithms are applied to the mining of very large real-
world databases. Since the training millions of samples
should reside in the memory, this restriction limits the
scalability of such algorithms, where the decision tree
construction can become inefficient due to swapping of
the training samples in and out of main and cache
memories. A new version of C4.5 algorithm was
developed and called C5.0 [23]. C5.0 algorithm proved to