A KDDSE-independent PMML Visualizer

Dietrich Wettschereck[1]

University of Applied Sciences, Bonn-Rhein-Sieg, Grantham Allee 20, 53757 Sankt Augustin, Germany,
dietrich.wettschereck@fh-bonn-rhein-sieg.de

Abstract. Several knowledge discovery support engines (KDDSE) feature the export and in a few cases even the import of data mining models in the Predictive Modeling Markup Language (PMML) standard. A visualization tool for PMML models that is independent of a specific KDDSE is presented in this paper. An extension of the PMML model for association rules that allows the definition of propositional and first order rules is also presented in its document type description form (DTD).

1 Introduction

The emerging standard for the platform and system independent representation of data mining models PMML (*Predictive Markup Modeling Language* [1]) is currently supported by a number of commercial and non-commercial knowledge discovery support engines (KDDSE). Most of these systems can export one or several model types in PMML, some can even import models generated by other KDDSEs. The primary purpose of the PMML standard is to separate model generation from model storage in order to enable users to view, post-process, and utilize data mining models independently of the KDDSE that generated the model.

This paper makes two contributions that are only related through the fact that they both employ PMML: it proposes an extension to the PMML model for association rules (AssociationModel) for the definition of (first-order) classification and regression rules (Section 2) and it describes a KDDSE-independent PMML visualizer (Section 3). A short discussion of the utility of such a visualizer in Section 4 is followed by a description of planed extensions to the tool (Section 5).

2 PMML DTD for (first order) rules and subgroups

This section describes a DTD (document type description) for propositional and first order rules. Subgroups [8] can also be represented by this PMML model type. The proposed DTD closely follows the AssociationModel DTD that is part of the PMML standard.

The rule models in PMML allow for defining either a classification or prediction structure. Each Rule holds a logical predicate expression that defines the conditions under which a rule will fire and a similar expression for the conclusions that can be drawn from the rule.

```
<!ELEMENT RuleModel (Extension*, MiningSchema,
                ModelStats?, GeneralRuleItem*,
                Itemset*, Rule+,
                Extension*)>
```

The *RuleModel* element starts the definition of a rule model. It has a few optional slots (denoted by '*' and '?' and one required slot, the element *Rule*. *Extension*, *MiningSchema*, and *ModelStats* are standard PMML. *GeneralRuleItem*, *Itemset*, and *Rule* are extensions that are described in detail below. The attributes of the *RuleModel* are not shown as they are identical to the attributes of the *AssociationModel*.

The *Rule* element is an encapsulation for a propositional or a first order rule. Every rule contains an antecedent and a consequent. The antecedent is either a simple predicate, a compound predicate or a reference to an *Itemset*. A compound predicate combines simple predicates and *Itemsets*. An *Itemset* is a generalization of an association rule *Itemset*. It represents a literal in first order logic terminology. The consequent is either a simple predicate or an *Itemset*.

The element *GeneralRuleItem* is an element of an *Itemset* and denotes a generalization of *Item* as defined in *AssociationModel* or a *RuleItem* as defined in this model:

```
<!ELEMENT GeneralRuleItem (Extension*, (RuleItem|Item))>
<!ATTLIST GeneralRuleItem EMPTY>
```

RuleItems are contained in *Itemsets* and represent a field (variable).

```
<!ELEMENT RuleItem EMPTY>
<!ATTLIST RuleItem
     id                %ELEMENT-ID;      #REQUIRED
     field             CDATA             #REQUIRED
     mappedValue       CDATA             #IMPLIED   >
```

Attribute description:

id: An identification to uniquely identify an item.

field: This must point to a field that was previously defined in the *DataDictionary*

mappedValue: Optional, a value to which the internal field value is mapped. This should be kept empty, since this information is redundant to the information given in the *DataDictionary* (it is only included here for compatibility with *Item* in *AssociationModel*).

Itemsets are contained in compound predicates of rules or directly in the antecedent or consequent. They are a generalization of *AssociationModel Itemsets*:

```
<!ELEMENT Itemset (Extension*, ItemRef+, DisplayTerm?)>
<!ATTLIST Itemset
        id                %ELEMENT-ID;     #REQUIRED
        predicate         CDATA            #REQUIRED
        support           %PROB-NUMBER;    #IMPLIED
        numberOfItems     %INT-NUMBER;     #IMPLIED   >
```

Attribute description:

id: An identification to uniquely identify an *Itemset*

predicate: the name of the predicate that will be used to combine the arguments. This can be a simple predicate (equals, greaterThan, ...), but shouldn't, since a *SimplePredicate* is better in this case. The predicate is the name of a functor and the items are its arguments. Example: predicate="father_of" ... itemRef="1" .. itemRef="2" indicates that the person indicated by item #1 is the father of the person indicated by item #2.

support: The relative support of the *Itemset*

numberOfItems: The number of items contained in this *Itemset*

DisplayTerm: The *Itemset* (Literal)described in natural language. Placeholders within the *DisplayTerm* allow for insertion of actual values.

```
<!ELEMENT DisplayTerm EMPTY >
<!ATTLIST DisplayTerm value CDATA  #REQUIRED>
```

Attribute description:

value: The ItemSet described in natural language. A visualization of this model should use this term instead of the standard predicate(arg1, arg2, ..., argn) representation. Placeholders in this value are denoted by %0, %1, ... where %0 is replaced by the actual value of the TermRef at position one, %1 by the TermRef at position two, and so on. The order of the placeholders is arbitrary, and not all TermRefs must be listed.

Each *Rule* consists of:

```
<!ELEMENT Rule ( Extension*, ScoreDistribution*,
                 Antecedent, Consequent )>
<!ATTLIST Rule
     support          %PROB-NUMBER;    #REQUIRED
     confidence       %PROB-NUMBER;    #REQUIRED
     ruleId           CDATA     #IMPLIED         >
```

Attribute description:

support: The relative support of the rule

confidence: The confidence of the rule

ruleId: The id value of the rule

The standard PMML definition of *PREDICATE* is extended here by *ItemSetRef* indicating that a predicate can also be a more complicated operator as foreseen by the standard:

```
<!ENTITY % PREDICATE "( SimplePredicate | SimpleSetPredicate |
                        CompoundPredicate | ItemSetRef |
                        True | False  ) " >
```

Each *Antecedent* consists of:

```
<!ELEMENT Antecedent ( Extension*, (%PREDICATE;) )>
```

The antecedant has an empty attribute list. The definition of a *Consequent* is identical to the definition of an antecedent.

3 Visualization of PMML models

Data visualization methods have been part of statistics and data analysis research for many years. This research concentrated primarily on plotting one or more independent variables against a dependent variable in support of explorative data analysis [4, 6]. The visualization of analysis results, however, only recently gained some attention with the proliferation of data mining[2]. This recent interest was spawned by the often overwhelming number and complexity of data mining results.

The visualization of analysis results primarily serves four purposes: (1) to better illustrate the model to the end user, (2) to utilize comparison of models, (3) to increase model acceptance, and (4) to enable model editing and provide support for ”what-if questions”.

The tool presented in this section was designed by the author to address these four issues. It is a Java implementation that can be run as an application or as an Applet. [1]

[1] The software is available upon request from the author. See also: http://soleunet.ijs.si/website/other/pmml.html.

It allows for viewing of models by users that either do not have access to the actual KDDSE, want to avoid the overhead of starting a KDDSE or want to present their results in the internet. This visualization wizzard currently supports the following PMML models: decision and regression trees (Figure 1), association rules (Figure 2), propositional and first order rules (non-standard PMML, Figure 3), and subgroups (non-standard PMML, Figure 4).

Figure 1 shows a decision tree for the well known Iris domain. The tree is fully expanded and is normally shown in color, where different colors in the nodes denote the number of instances from each class contained in that node. The user can browse through the tree and open or close subtrees as needed.

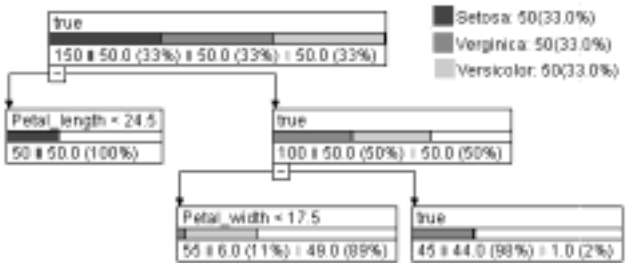

Fig. 1. Visualization of a decision tree for the Iris data set (courtesy of G. Meyer, IBM, visualized from PMML model exported from Intelligent Miner)

Figure 2 shows an interactive visualization for association rules. For each rule, confidence and support are displayed. The bar below each rule display graphically these two numbers where the length of the bar shows the support and the color of the bar its confidence (from red denoting low confidence to green denoting high confidence). The two sliders at the bottom of the display allow the restriction of the rules to be displayed to those that satisfy selected minimal confidence and support values.

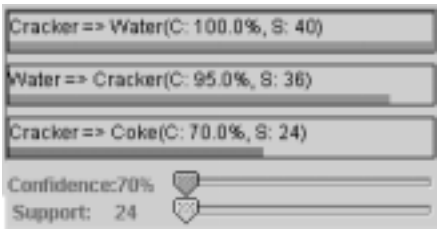

Fig. 2. Visualization of three association rules (courtesy DMG, slightly modified example for AssociationModel).

Figure 3 displays a modified set of rules learned by Aleph in the animal domain. The rules for each class are summarized by the bars at the right of the figure. The left bar shows the number of instances correctly covered, and the right bar the number of exceptions covered by the rule.

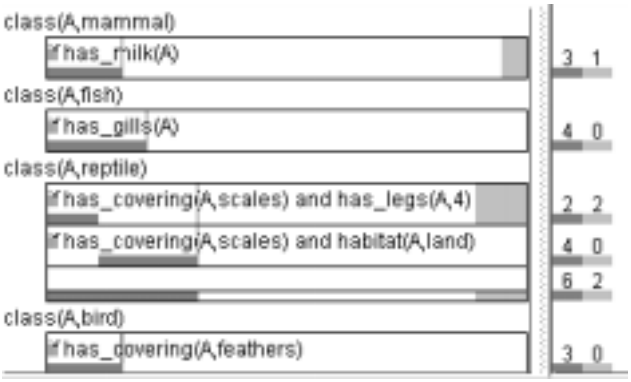

Fig. 3. Visualization of first order rules for the animals task (courtesy S. Moyle, Oxford University)

Figure 4 displays a set of subgroups discovered by Midos [8] in the Cleveland Heart Disease domain. Shown is the size of each subgroup, how it compares to the entire population and the distribution of the target values within each subgroup. Experience gained from working with non-technical end users has shown that a pie chart visualization is more appealing to these users because they more closely resemble business charts. Pie charts, however, often mislead the perception of the user due to difficulties with relating the size of pie slices to actual values. Hence, alternative visualizations are possible (see, for example [3]).

4 Discussion

The visualization tool presented is a simple, yet powerful tool that can function as a dissemination tool for data mining results. Its simplicity ensures that non-KDD users can operate the tool and interpret the results obtained by a data mining expert. Java technology ensures that platform issues are secondary and that results could even be part of online content management or workgroup support systems.

The proposed extension to the PMML AssociationModel should be seen as a first proposal of a first order PMML rule model. It does not at this time utilize the extension mechanism that can be used in PMML models. The reason for this deviation from the proposed extension procedure is that first order rules models are sufficiently generic data mining models to justify the existence of a distinct model type. However, significant effort has been extended to stay as close as possible to the terminology employed by the AssociationModel.

5 Future Work

The tool presented should be enhanced in three directions: (1) Addition of visualization methods for the other PMML models that are supported by the current standard.

Fig. 4. Visualization of selected subgroups for the Cleveland Heart Disease domain (generated by Midos, exported from Kepler)

(2) Addition of functionality that enables the user to edit PMML models. Straight forward editing operations are the deletion of entire rules or subgroups or of conditions within these. More complex editing operations are the modification of existing rules or the addition of entirely new rules. Likewise, the editing of decision and regression trees will be supported. (3) Addition of a model evaluator. A common request voiced by current users of the visualizer is to be able to evaluate single records or entire tables on the model that is displayed (and possibly modified by the user). However, in order to avoid a significant increase in tool complexity, it is envisioned to realize the PMML model evaluator as a separate tool.

ACKNOWLEDGMENT

This work has been supported in part by the EU funded project Data Mining and Decision Support for Business Competitiveness: A European Virtual Enterprize (IST-1999-11495). I am grateful to G. Meyer and the members of the data mining group for providing sample PMML models. S. Moyle generated the first order rules for the Animals domain. The visualizations presented here were developed by A. and G. Andrienko, AIS, FhG, Sankt Augustin, Germany.

References

1. Data Mining Group, see www.dmg.org
2. U.M. Fayyad, G.G. Grinstein, and A. Wierse, Information visualization in data mining and knowledge discovery. Morgan Kaufmann, (2002).
3. D. Gamberger, N. Lavrač, D. Wettschereck, Subgroup Visualization: A Method and Application in Population Screening. *ECAI 2002 Workshop on INTELLIGENT DATA ANALYSIS IN MEDICINE AND PHARMACOLOGY*, (2002).
4. H.Y. Lee, H.L. Ong, and L.H. Quek, Exploiting visualization in knowledge discovery. In *Proc. of the First Inter. Conference on Knowledge Discovery and Data Mining*, pp. 198-203, (1995).
5. S. Müller, diplom thesis, University of Magdeburg, (2000).
6. Workshop on visual data mining, PKDD 2001, Freiburg, Germany, (2001). http://www-staff.it.uts.edu.au/˜simeon/vdm_pkdd2001/
7. A. Unwin, Visualisation for data mining, (2000). http://www1.math.uni-augsburg.de/ unwin/
8. S. Wrobel, An algorithm for multi-relational discovery of subgroups. *Proc. First European Symposium on Principles of Data Mining and Knowledge Discovery*, 78–87, Springer, (1997).