International Journal of Approximate Reasoning 68 (2016) 164–178

Contents lists available at ScienceDirect

International Journal of Approximate Reasoning

www.elsevier.com/locate/ijar

Decision functions for chain classifiers based on Bayesian networks for multi-label classification

Gherardo Varando *, Concha Bielza, Pedro Larrañaga

Dept. of Artificial Intelligence, Universidad Politécnica de Madrid, Campus de Montegancedo, Madrid, Spain

ARTICLE INFO

Article history:
Received 16 December 2014
Received in revised form 17 April 2015
Accepted 11 June 2015
Available online 23 June 2015

Keywords:
Bayesian network classifier
Multi-label classification
Expressive power
Chain classifier
Binary relevance
Decision functions

ABSTRACT

Multi-label classification problems require each instance to be assigned a subset of a defined set of labels. This problem is equivalent to finding a multi-valued decision function that predicts a vector of binary classes. In this paper we study the decision boundaries of two widely used approaches for building multi-label classifiers, when Bayesian network-augmented naive Bayes classifiers are used as base models: *Binary relevance method* and *chain classifiers*. In particular extending previous single-label results to multi-label chain classifiers, we find polynomial expressions for the multi-valued decision functions associated with these methods. We prove upper boundings on the expressive power of both methods and we prove that chain classifiers provide a more expressive model than the binary relevance method.

1. Introduction

We consider a multi-label classification problem [24,20] over categorical predictors, that is, mapping every instance $\mathbf{x} = (x_1, \ldots, x_n)$ to a subset of h labels:

$$\mathbf{\Omega} = \Omega_1 \times \cdots \times \Omega_n \to Y \subseteq \mathcal{Y} = \{y_1, \ldots, y_h\},$$

where $\Omega_i \subset \mathbb{R}$, $|\Omega_i| = m_i < \infty$. As usual the problem could be transformed into a multi-dimensional binary classification problem, that is, finding an h-valued decision function $\mathbf{f}$ that maps every instance of n predictor variables $\mathbf{x}$ to a vector of h binary values $\mathbf{c} = (c_1, \ldots, c_h) \in \{-1, +1\}^h$:

$$\begin{aligned} \mathbf{f}: \quad \mathbf{\Omega} = \Omega_1 \times \cdots \times \Omega_n \quad &\to \quad \{-1, +1\}^h \\ (x_1, \ldots, x_n) \quad &\mapsto \quad (c_1, \ldots, c_h), \end{aligned}$$

where $c_i = +1$ (-1) means that the ith label is present (absent) in the predicted label subset Y. We consider the predictor variables $X_1, \ldots, X_n$ and the binary classes $C_i \in \{-1, +1\}$ as categorical random variables. Real examples include classification of texts into different categories [8], diagnosis of multiple diseases from common symptoms and identification of multiple biological gene functions [3,23].

The easiest way to approach a multi-label classification problem is to divide it into a set of single-label classification problems (equivalent to binary classification problems). Each binary problem is then solved independently and thus h binary

* Corresponding author.
E-mail addresses: gherardo.varando@upm.es (G. Varando), mcbielza@fi.upm.es (C. Bielza), pedro.larranaga@fi.upm.es (P. Larrañaga).

http://dx.doi.org/10.1016/j.ijar.2015.06.006
0888-613X/

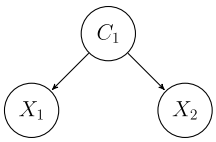

Fig. 1. Naive Bayes classifier structure in Example 1.

classifiers, one for each class variable C_i, are built. Each binary classifier is learned from predictor variables and C_i data only. At the end the results are combined to form multi-label prediction. Known as *binary relevance*, this method is easily implementable, has low computational complexity and is fully parallelizable. Therefore it is scalable to a large number of classes. However, it completely ignores dependencies among labels and generally does not represent the most likely set of labels.

Chain classifiers [18,6] relax the independence assumption by iteratively adding class dependencies in the binary relevance scheme. The kth classifier in the chain predicts class C_k from $X_1, \ldots, X_n, C_1, \ldots, C_{k-1}$. Sucar et al. [19] employed naive Bayes within chain classifiers.

In this paper, we study differences in the *expressive power* of these two methods when Bayesian network (BN) classifiers [1] are used. Expressive power of a classifier over categorical variables could be seen simply as the number of distinct decision functions that a given type of classifier induces.

In Varando et al. [22] the expressive power of one-dimensional binary, or one-label classifiers has been studied. In particular, the results of Minsky [11] and Peot [14] about the decision boundary of naive Bayes have been extended to a broader class of Bayesian network classifiers. A polynomial representation of the decision functions induced by Bayesian network-augmented naive Bayes classifier is described, and in absence of V-structures a stronger characterization is shown to hold. In this paper, we extend these results to multi-label classifiers. Moreover, we suggest some theoretical reasons why the simple binary relevance method can perform poorly when relationships among labels exist, and we prove that chain classifiers provide more expressive models. A broader chain classifiers class than in Varando et al. [21] is considered and studied extensively and a bounding on the expressive power of those models is proved. Moreover we present novel illustrative examples both about the one-dimensional results and about multi-label ones.

In Section 2 we review previous work on one-dimensional binary classifiers. We describe the binary relevance method and compute its expressive power in Section 3. We analyse chain classifiers in Section 4. In Section 5 we compare the two methods, proving that actually chain classifiers are more expressive than binary relevance and in Section 6 we present our conclusions and some ideas for future research.

2. Expressive power of one-dimensional BN classifiers

We report here previous results on the decision boundary and expressive power of one-label, or equivalently one-dimensional binary, BN classifiers [22]. We restrict to binary classifier and we can assume that the class variables takes its values on $\{-1, +1\}$. Classifiers where the class variable takes more than two values are more complex to study, the associated decision functions could be seen as combinations of binary decision functions and thus some of the results of this section could probably be extended. In the present work we prefer to remain in the binary case. Moreover binary classes are the variables needed to define multi-label classification problems.

In particular, we look at Bayesian network-augmented naive Bayes (BAN) classifiers [7].

BAN classifiers are Bayesian network classifiers where the class variable C is assumed to be a parent of every predictor and the predictor sub-graph $\mathcal{G}$ can be a general BN. We observe that every BAN classifier is determined by the predictor sub-graph $\mathcal{G}$, because the class variable C is superposed as parent of every variable of $\mathcal{G}$. As we focus only on Bayesian network, we will use the word graph to refer only to a directed acyclic graph, the structure of a Bayesian network (For general notations see Table 2).

For every BAN classifier, the induced decision function is

$$f_{\mathcal{G}}^{BAN}(x_1, \ldots, x_n) = \arg \max_{c \in \{-1,+1\}} P(C = c, X_1 = x_1, \ldots, X_n = x_n), \tag{1}$$

and $P(C = c, X_1 = x_1, \ldots, X_n = x_n)$ is factorized according to BN theory [13] as

$$P(C = c) \prod_{i=1}^{n} P\left(X_i = x_i | C = c, \mathbf{X}_{\mathbf{pa}(i)} = \mathbf{x}_{\mathbf{pa}(i)}\right),$$

where $\mathbf{X}_{\mathbf{pa}(i)}$ are the parents of X_i in the predictor sub-graph $\mathcal{G}$. Moreover, $\mathbf{pa}(i)$ denotes the set of indexes defining the parents of X_i that are not C and $\mathbb{M}_i = \times_{s \in \mathbf{pa}(i)} \{1, \ldots, m_s\}$, the set of possible configurations of $\mathbf{X}_{\mathbf{pa}(i)}$.

Example 1. Consider a naive Bayes classifier (structure in Fig. 1), that is, the simplest BAN, over predictor variables $X_1 \in \{0, 1, 2\}$, $X_2 \in \{0, 1\}$. In this case the joint probability over (C, X_1, X_2) is factorized as