COMPARISON OF DISTANCES FOR MULTI-LABEL CLASSIFICATION WITH PCTs

Valentin Gjorgjioski, Dragi Kocev, Sašo Džeroski
Jozef Stefan Institute
Jamova cesta 39, 1000 Ljubljana, Slovenia
e-mail: valentin.gjorgjioski@ijs.si

ABSTRACT

Multi-label classification has received significant attention in the research community over the past few years: this has resulted in the development of a variety of multi-label classification methods. These methods either transform the multi-label dataset to several simpler datasets or adapt the learning algorithm so it can handle the multiple labels. In this paper, we consider the latter approach. Namely, we use predictive clustering trees to perform multi-label classification. Furthermore, we perform an experimental comparison of four distance measures used to select the splits in the nodes of the trees. The experimental evaluation was conducted on 6 benchmark datasets using 6 different evaluation measures. The results show that, averaged overall, the Euclidean distance and the Hamming loss yield the best predictive performance.

1 INTRODUCTION

Traditionally, binary classification is concerned with deciding whether a given example has (or doesn't have) a single given target property/class. Multi-class classification involves the labeling of a given example with a single label/class λ_i from a finite set of disjoint labels $L = \{ \lambda_1, \lambda_2,\ldots, \lambda_Q\}$, Q>2. In contrast, multi-label classification learns a mapping from an example in the input space ($x \in X$) to a set of labels ($Y \subseteq L$) from the output space L. Note that, unlike in multi-class classification, in multi-label classification the labels are not mutually exclusive, i.e., a single example can be labeled with multiple labels. The labels that belong to the output Y are called relevant labels, while those from $L \setminus Y$ are called irrelevant for a given example.

The machine learning task of multi-label classification data has lately received significant attention from the research community [1], which has resulted in development of many methods that tackle this task. The developed methods can be generally divided into two categories: problem transformation and algorithm adaptation. Problem transformation methods transform problem into one or more single-label classification problems. These problems are then solved using a commonly used method for single-label classification and, afterwards, the output is transformed back into a multi-label representation. Algorithm adaptation methods adapt the learning algorithms to handle the multi-label data directly. In this work, we focus on algorithm adaptation methods. Specifically, we use predictive clustering trees (PCTs) [2] as classifiers and extend the distance function used when learning the tree. PCTs are a generalization of decision trees that are capable of predicting structured outputs. Namely, PCTs can handle multiple continuous targets, multiple discrete targets, time-series [3] and hierarchies of classes [4]. In the context of multi-label classification, we employ the PCTs for multiple discrete targets where a weighted Euclidean distance is used to generate the tests in the internal nodes of the tree. Here, we extend the PCTs with three distance measures: Hamming distance, Jaccard distance and a matching distance. These distances will provide additional flexibility for the users when they apply PCTs to different domains.

We compare the predictive performances of the PCTs obtained using different distance measures. The predictive performance was assessed on several benchmark datasets from multi-label classification. The predictive performance was measured with six evaluation measures: Hamming loss, accuracy, precision, recall, F1 score and subset accuracy.

The remainder of this paper is organized as follows. In Section 2, we present the predictive clustering trees for multiple discrete targets. We define the distances that we use in Section 3. We give the experimental design and in Section 4 and the results in Section 5. Section 6 concludes.

2 PREDICTIVE CLUSTERING TREES

The Predictive Clustering Trees (PCTs) framework sees a decision tree as a hierarchy of clusters: the top-node corresponds to one cluster containing all data, which is recursively partitioned into smaller clusters while moving down the tree. The PCT framework is implemented in the CLUS system, which is available for download at http://www.cs.kuleuven.be/~dtai/clus.

PCTs can be induced with a standard top-down induction of decision trees (TDIDT) algorithm. The algorithm takes as input a set of examples and outputs a tree. The heuristic that is used for selecting the tests is the reduction in variance caused by partitioning the instances. By maximizing the variance reduction the cluster homogeneity is maximized and it improves the predictive

performance. If no acceptable test can be found, that is, if the test does not significantly reduces the variance, then the algorithm creates a leaf and computes the prototype of the instances belonging to that leaf. The main difference between the algorithm for learning PCTs and a standard decision tree learner is that the former considers the variance function and the prototype function, that computes a label for each leaf, as parameters that can be instantiated for a given learning task. So far, the PCTs have been instantiated for the following tasks: multiple targets prediction [5], hierarchical-multi label classification [4] and prediction of time-series [3].

In this paper, we focus on the first task. PCTs that are able to predict a tuple of discrete variables are called multi-target classification trees (MTCTs). An example of a MTCT is shown in Figure 1. This MTCT presents a habitat model for 14 bioindicator species [6]. The internal nodes of the tree contain tests on the descriptive variables (in this case, chemical parameters of the water samples) and the leaves store the predictions (in this case, which species are encountered and which not in a given water sample).

The variance is calculated as the sum of the squared pairwise distances between the instances, i.e.,

$$Var(E) = \frac{1}{2|E|^2} \sum_{X \in E} \sum_{Y \in E} d^2(X, Y)$$

The function used to calculate the prototype is then $m = \arg\min_q \sum_{X \in E} d^2(X, q)$. In this case, the prototype is an instance from the dataset and is called medoid. Different distances can be used depending on the application domain. By default, PCTs use the Euclidean distance.

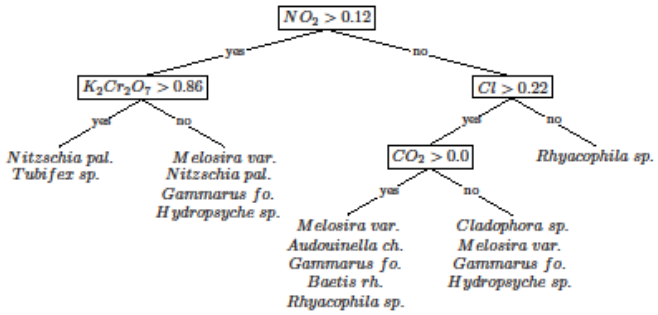

Figure 1: An example of a predictive clustering tree for predicting multiple discrete targets. The leaves predict the presence or absence for each bioindicator species.

3 DISTANCES FOR MULTI-LABEL LEARNING

In a multi-label learning setup, the target variable is a set of labels. Therefore, we can readily use distances over sets. Another approach to the problem is to see the multi-label classification problem as a predicting tuples of discrete targets and use distances over tuples. PCTs (and decision trees) have been previously used in the later context [2]. The focus of this study is the former approach to multi-label classification. In the remainder of this section, we present the distances over sets that can be used for extension of PCTs for multi-label classification.

3.1 Euclidean distance

The target in multi-label classification can be represented as a tuple of 0/1 values. The length of the target tuple is the number of all labels in the dataset. In this case, the Euclidean distance between two sets of labels C_i and C_j is defined as the Euclidean distance between their vector representations.

3.2 Hamming distance

The Hamming distance between two strings (i.e., bit-vectors) of equal length is the number of positions at which the corresponding symbols are different. In other words, it measures the minimum number of substitutions required to change the first string into the second. In terms of sets, the Hamming distance between two sets C_i and C_j is defined as:

$$h(C_i, C_j) = |C_i \cup C_j| - |C_i \cap C_j|$$

3.3 Jaccard distance

The Jaccard distance measures the dissimilarity between two sets by dividing the difference of the sizes of the union and the intersection of the two sets with the size of the union. The Jaccard distance can be calculated as follows.

$$j(C_i, C_j) = \frac{|C_i \cup C_j| - |C_i \cap C_j|}{|C_i \cup C_j|}$$

3.4 Matching distance (MD)

Motivated by a recently introduced distance on sets of structured objects, this distance is based on the matching between object from the sets. The matched objects do not contribute to the distance, which has the value of the unmatched part of the larger dataset, as defined below

$$md(C_i, C_j) = \max(|C_i|, |C_j|) - |C_i \cap C_j|$$

4 EXPERIMENTAL DESIGN

We begin by describing the benchmark datasets used in this study. Next, we present the most typically used evaluation measures for multi-label classification. We then give the experimental setup for the data analysis.

4.1 Datasets

We use 6 multi-label classification benchmark problems. Parts of the selected problems were used in various studies and evaluations of methods for multi-label learning. In the process of selection of problems, we opted to include benchmark datasets with different scale and from various application domains. Table 1 presents the basic statistics of the datasets. The datasets vary in size: from 391 up to 5318 training examples, from 202 up to 2635 testing examples, from 16 up to 1449 features, from 5 to 53 labels, and from 1.20 to 6.34 average number of labels per example.

	domain	N/T	D	Q	l_c
water quality	ecology	721/339	16	14	5.07
emotions	music	391/202	72	6	1.87
mediana	text	5318/2635	79	5	1.20
soil quality	ecology	1308/636	54	39	6.34
medical	text	645/333	1449	45	1.25
enron	text	1123/579	1001	53	3.38

Table 1. Description of the datasets in terms of application *domain*, number of training (*N*) and test (*T*) examples, the number of features (*D*), the total number of labels (*Q*) and label cardinality (l_c). The problems are ordered by their overall complexity roughly calculated as *N* x *D* x *Q*.

4.2 Evaluation measures

The evaluation of the predictive performance for multi-label learning systems differs from that of classical single-label learning systems. In any multi-label experiment, it is essential to include multiple and contrasting measures because of the additional degrees of freedom that the multi-label setting introduces. In our experiments, we used various evaluation measures that have been suggested by Tsoumakas et al [1]. In particular, we used six example-based evaluation measures: Hamming loss, accuracy, precision, recall, F1 score and subset accuracy.

In the definitions below, *Yi* denotes the set of true labels of example xi and $h(x_i)$ denotes the set of predicted labels for the same examples. All definitions refer to the multi-label setting.

Hamming loss evaluates how many times an example-label pair is misclassified, i.e., label not belonging to the example is predicted or a label belonging to the example is not predicted. The smaller the value of *hamming_loss*(*h*), the better the performance. The performance is perfect when *hamming_loss*(*h*) = 0. This metric is defined as:

$$hamming\_loss(h) = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{Q}\left|h(x_i)\Delta y_i\right|$$

where Δ stands for the symmetric difference between the two sets, *N* is the number of examples and *Q* is the total number of possible class labels.

Accuracy for a single example x_i is defined by the Jaccard similarity coefficients between the label sets $h(x_i)$ and y_i. Accuracy is micro-averaged across all examples.

$$accuracy(h) = \frac{1}{N}\sum_{i=1}^{N}\frac{\left|h(x_i)\cap y_i\right|}{\left|h(x_i)\cup y_i\right|}$$

Precision is defined as:

$$precision(h) = \frac{1}{N}\sum_{i=1}^{N}\frac{\left|h(x_i)\cap y_i\right|}{\left|y_i\right|}$$

Recall is defined as:

$$recall(h) = \frac{1}{N}\sum_{i=1}^{N}\frac{\left|h(x_i)\cap y_i\right|}{\left|h(x_i)\right|}$$

F1 score is the harmonic mean between precision and recall and is defined as:

$$F_1(h) = \frac{1}{N}\sum_{i=1}^{N}\frac{2\times\left|h(x_i)\cap y_i\right|}{\left|h(x_i)\right|+\left|y_i\right|}$$

F_1 score is an example based metric and its value is an average over all examples in the dataset. F_1 score reaches its best value at 1 and worst at 0.

Subset Accuracy is defined as follows:

$$subset\_accuracy(h) = \frac{1}{N}\sum_{i=1}^{N} I\left(h(x_i) = y_i\right)$$

where *I*(*true*) = 1 and *I*(*false*) = 0. This is a very strict evaluation measure as it requires the predicted set of labels to be an exact match of the true set of labels.

4.3 Experimental setup

We used the predictive clustering framework implemented in the CLUS system to investigate the performance of the different distance measures. To this end, we constructed single PCTs.

The PCTs were pruned with the F-test pruning method. This method checks whether a given test statistically significantly reduces the intra-cluster variance at a given significance level. An optimal significance level was selected by using internal 3-fold cross validation, from the following values: 0.01, 0.02, 0.03, 0.04 and 0.05.

5 RESULTS

Tables 2, 3, 4, 5, 6 and 7 show the results from the experimental evaluation of the distance measures. In the following, we briefly discuss the results for each evaluation measure. The Hamming distance has best predictive performance according to the Hamming loss measure. This is expected, since the trees with this distance are set to optimize that measure. Furthermore, since the Euclidean and Hamming distance are quite similar for vectors with 1/0 values, the Euclidean distance also has good predictive performance. On average, the Jaccard distance has the lowest predictive performance.

	Euc.	Ham.	Jac.	MD
water quality	0.314	**0.309**	0.528	0.312
emotions	**0.249**	0.272	0.274	0.253
mediana	**0.157**	0.165	0.355	0.203
soil quality	0.106	**0.099**	0.169	0.100
medical	**0.013**	**0.013**	0.014	**0.013**
enron	0.058	**0.055**	0.062	0.057

Table 2. The Hamming loss measure for different distances

In terms of accuracy, the Euclidean, Hamming and MD distance have similar predictive performance on average, while the Euclidean distance has the best performance on three datasets. The Jaccard distance, on the other hand, has the worst performance on average.

	Euc.	Ham.	Jac.	MD
water quality	0.298	0.315	**0.370**	0.317
emotions	**0.496**	0.469	0.488	0.493
mediana	**0.589**	0.588	0.302	0.505
soil quality	0.481	0.502	0.347	**0.504**
medical	**0.733**	0.731	0.718	0.727
enron	0.413	**0.435**	0.427	0.425

Table 3. The accuracy for the different distances

The precision and recall have inverted values. In the case of precision, Jaccard distance is the best performing, while for recall it is the worst performing. The distance to the other methods is large in the both cases. This means that the labels produced with Jaccard distance are reliable (low false positive rate); however, they do not cover all relevant labels for a given example (high false negative rate). The other three distances have similar performances to each other.

	Euc.	Ham.	Jac.	MD
water quality	0.352	0.382	**0.860**	0.390
emotions	0.583	0.561	**0.635**	0.580
mediana	0.605	**0.641**	0.465	0.602
soil quality	0.595	0.606	0.556	**0.618**
medical	0.755	**0.761**	0.746	0.755
enron	0.502	0.524	**0.558**	0.523

Table 4. The precision for the different distances

	Euc.	Ham.	Jac.	MD
water quality	**0.625**	0.623	0.397	0.614
emotions	**0.613**	0.592	0.571	0.600
mediana	**0.722**	0.704	0.359	0.595
soil quality	0.719	**0.730**	0.492	0.712
medical	0.779	**0.787**	0.771	0.776
enron	0.568	**0.600**	0.552	0.572

Table 5. The recall for the different distances

The F_1 score balances the performance measured by the precision and the recall. On average, the Jaccard distance has the lowest performance (because of the weak results for recall). The Hamming distance is slightly better than the remaining two distances.

	Euc.	Ham.	Jac.	MD
water quality	0.423	0.441	**0.523**	0.444
emotions	0.574	0.551	**0.575**	0.568
mediana	0.634	**0.642**	0.385	0.567
soil quality	0.617	0.634	0.491	**0.635**
medical	**0.757**	0.760	0.746	0.753
enron	0.515	**0.543**	0.535	0.530

Table 6. The F_1 scores for the different distances

The subset accuracy measures the fraction of the complete and accurate predictions. In this regard, the Euclidean distance has the best average performance, while MD is the best performing distance on four datasets. The worst performing distance is the Jaccard distance.

	Euc.	Ham.	Jac.	MD
water quality	0.009	0.012	0.000	**0.018**
emotions	0.262	0.233	0.223	**0.272**
mediana	**0.468**	0.440	0.063	0.327
soil quality	0.036	0.041	0.003	**0.044**
medical	**0.661**	0.640	0.631	0.646
enron	0.145	0.149	0.149	**0.150**

Table 7. The subset accuracy for the different distances

6 CONCLUSIONS

In this paper, we have presented an experimental evaluation of four distance measures for multi-label classification. The evaluation was performed on 6 benchmark datasets using 6 evaluation measures.

The results show that there is no overall best distance measure. The best choice for a distance measure is the one that optimizes a selected evaluation measure. For example, the Hamming distance works the best when optimizing the Hamming loss, while the best according to precision is the Jaccard distance (since there is a strong connection between precision and the Jaccard coefficient). All in all, the Euclidean distance and Hamming loss perform the best averaged across all evaluation measures.

References

1. G. Tsoumakas, I. Katakis. Multi Label Classification: An Overview. International Journal of Data Warehouse and Mining 3(3). 2007. pp. 1–13.
2. H. Blockeel, L. D. Raedt, J. Ramon. Top-down induction of clustering trees. In Proceedings of the 15th International Conference on Machine Learning. 1998. pp. 55–63.
3. I. Slavkov, V. Gjorgjioski, J. Struyf, and S. Dzeroski. Finding explained groups of time-course gene expression profiles with predictive clustering trees. Molecular Biosystems 6. 2010. pp. 729-740.
4. C. Vens, J. Struyf, L. Schietgat, S. Dˇzeroski, H. Blockeel. Decision trees for hierarchical multi-label classification. Machine Learning 73 (2). 2008. pp. 185–214.
5. J. Struyf and S. Dzeroski. Constraint based induction of multi-objective regression trees. In proceedings of the 4th International Workshop on Knowledge Discovery in Inductive Databases. 2005. pp. 110-121.
6. H. Blockeel, S. Dzeroski, J. Grbovic, Simultaneous prediction of multiple chemical parameters of river water quality with TILDE. In Proceedings of the 3rd European Conference on PKDD - LNAI 1704. 1999. pp. 32-40. Springer.