Ensembles of Multi-Objective Decision Trees

Dragi Kocev[1], Celine Vens[2], Jan Struyf[2], and Sašo Džeroski[1]

[1] Department of Knowledge Technologies, Jožef Stefan Institute,
Jamova 39, 1000 Ljubljana, Slovenia
{dragi.kocev,saso.dzeroski}@ijs.si
[2] Department of Computer Science, Katholieke Universiteit Leuven,
Celestijnenlaan 200A, 3001 Leuven, Belgium
{celine.vens,jan.struyf}@cs.kuleuven.be

Abstract. Ensemble methods are able to improve the predictive performance of many base classifiers. Up till now, they have been applied to classifiers that predict a single target attribute. Given the non-trivial interactions that may occur among the different targets in multi-objective prediction tasks, it is unclear whether ensemble methods also improve the performance in this setting. In this paper, we consider two ensemble learning techniques, bagging and random forests, and apply them to multi-objective decision trees (MODTs), which are decision trees that predict multiple target attributes at once. We empirically investigate the performance of ensembles of MODTs. Our most important conclusions are: (1) ensembles of MODTs yield better predictive performance than MODTs, and (2) ensembles of MODTs are equally good, or better than ensembles of single-objective decision trees, i.e., a set of ensembles for each target. Moreover, ensembles of MODTs have smaller model size and are faster to learn than ensembles of single-objective decision trees.

1 Introduction

In this work, we concentrate on the task of predicting multiple attributes. Examples thus take the form $(\mathbf{x_i}, \mathbf{y_i})$ where $\mathbf{x_i} = (x_{i1}, \ldots, x_{ik})$ is a vector of k input attributes and $\mathbf{y_i} = (y_{i1}, \ldots, y_{it})$ is a vector of t target attributes. This task is known under the name of multi-objective prediction. Existing learning techniques have been extended to address this task by learning to predict all target attributes at once [1–4]. This has two main advantages over building a separate model for each target: first, a multi-objective model is usually much smaller than the total size of the individual models for all target attributes, and second, such a multi-objective model explicates dependencies between the different target attributes. Moreover, the cited literature reports similar or slightly improved predictive performance results for the multi-objective models.

The goal of this paper is to investigate whether ensemble methods [5] can be applied to multi-objective prediction problems in order to achieve better performance. Ensemble methods construct a set of classifiers for a given prediction task and classify new data instances by taking a vote over their predictions. Ensemble methods typically improve the predictive performance of their base classifier [6].

Up till now, they have only been applied to single-objective prediction, i.e., predicting one target attribute. Given the non-trivial interactions between the target attributes in a multi-objective domain, it is unclear whether ensembles are also able to improve predictive performance in this non-standard setting. A positive answer would stimulate further research towards multi-objective problems, which are present in many real world applications [1, 7–12].

In this paper, we use decision trees as base classifiers. The ensemble methods that we investigate are bagging [6] and random forests [13]. More precisely, the main questions we want to answer are (1) does building ensembles of multi-objective decision trees improve predictive performance, and (2) how do ensembles of multi-objective decision trees compare to ensembles of single-objective decision trees, i.e., a set of separate ensembles for each target attribute. The last comparison is made along three dimensions: predictive performance, model size, and running times.

The paper is organized as follows. In Section 2, we briefly discuss ensemble methods. Section 3 explains multi-objective decision trees in more detail. Section 4 presents a detailed experimental evaluation. Conclusions and some ideas for further work are presented in Section 5.

2 Ensemble Methods

An ensemble is a set of classifiers constructed with a given algorithm. Each new example is classified by combining the predictions of every classifier from the ensemble. These predictions can be combined by taking the average (for regression tasks) or the majority vote (for classification tasks), as described by Breiman [6], or by taking more complex combinations [17, 18].

A necessary condition for an ensemble to be more accurate than any of its individual members, is that the classifiers are accurate and diverse [14]. An accurate classifier does better than random guessing on new examples. Two classifiers are diverse if they make different errors on new examples. There are several ways to introduce diversity: by manipulating the training set (by changing the weight of the examples [6, 15] or by changing the attribute values of the examples [16]), or by manipulating the learning algorithm itself [5].

In this paper, we consider two ensemble learning techniques that have primarily been used in the context of decision trees: bagging and random forests.

2.1 Bagging

Bagging [6] is an ensemble method that constructs the different classifiers by making bootstrap replicates of the training set and using each of these replicates to construct one classifier. Each bootstrap sample is obtained by randomly sampling training instances, with replacement, from the original training set, until an equal number of instances is obtained.

Breiman [6] has shown that bagging can give substantial gains in predictive performance, when applied to an unstable learner (i.e., a learner for which small

Table 1. The top-down induction algorithm for PCTs.

procedure PCT(E) **returns** tree	**procedure** BestTest(E)
1: $(t^*, h^*, \mathcal{P}^*) = \text{BestTest}(E)$	1: $(t^*, h^*, \mathcal{P}^*) = (none, 0, \emptyset)$
2: **if** $t^* \neq none$ **then**	2: **for each** possible test t **do**
3: **for each** $E_k \in \mathcal{P}^*$ **do**	3: $\mathcal{P} =$ partition induced by t on E
4: $tree_k = \text{PCT}(E_k)$	4: $h = Var(E) - \sum_{E_k \in \mathcal{P}} \frac{\lvert E_k \rvert}{\lvert E \rvert} Var(E_k)$
5: **return** node$(t^*, \bigcup_k \{tree_k\})$	5: **if** $(h > h^*) \wedge \text{Acceptable}(t, \mathcal{P})$ **then**
6: **else**	6: $(t^*, h^*, \mathcal{P}^*) = (t, h, \mathcal{P})$
7: **return** leaf(Prototype(E))	7: **return** $(t^*, h^*, \mathcal{P}^*)$

changes in the training set result in large changes in the predictions), such as classification and regression tree learners.

2.2 Random Forests

A random forest [13] is an ensemble of trees, where diversity among the predictors is obtained by using bagging, and additionally by changing the feature set during learning. More precisely, at each node in the decision trees, a random subset of the input attributes is taken, and the best feature is selected from this subset. The number of attributes that are retained is given by a function f of the total number of input attributes x (e.g., $f(x) = 1$, $f(x) = \sqrt{x}$, $f(x) = \lfloor log_2(x) + 1 \rfloor \ldots$). By setting $f(x) = x$, we obtain the bagging procedure.

3 Multi-Objective Decision Trees

Multi-objective decision trees (MODTs) [2] are decision trees capable of predicting multiple target attributes at once. They are an instantiation of predictive clustering trees (PCTs) [2] that are used for multi-objective prediction. In the PCT framework, a tree is viewed 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.

PCTs can be constructed with a standard "top-down induction of decision trees" (TDIDT) algorithm [19]. The algorithm is shown in Table 1. The heuristic that is used for selecting the tests is the reduction in variance caused by partitioning the instances (see line 4 of BestTest). Maximizing the variance reduction maximizes cluster homogeneity and improves predictive performance.

The main difference between the algorithm for learning PCTs and a standard decision tree learner is that the former treats 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. In order to construct MODTs, these functions have been instantiated towards multiple target attributes [2, 20]. For the classification case, the variance function is computed as the sum of the entropies of class variables, i.e., $Var(E) = \sum_{i=1}^{t} Entropy(E, y_i)$ (this definition has also been used in the context of multi-label prediction [21]), and the prototype

function returns a vector containing the majority class for each target attribute. For multi-objective regression trees, the sum of the variances of the targets is used, i.e., $Var(E) = \sum_{i=1}^{t} Var(y_i)$, and each leaf's prototype is the vector mean of the target vectors of it's training examples.

The PCT framework is implemented in the TILDE [2] and CLUS [22, 4] systems. In this work we use CLUS. More information about PCTs and CLUS can be found at `http://www.cs.kuleuven.be/~dtai/clus`.

4 Experimental Evaluation

In this section, we empirically evaluate the application of bagging and random forests to multi-objective decision trees. We describe the experimental methodology, the datasets, and the obtained results.

4.1 Ensembles for Multi-Objective Decision Trees

In order to apply bagging to MODTs, the procedure $PCT(E_i)$ (Table 1) is used as a base classifier. For applying random forests, the same approach is followed, changing the procedure BestTest (Table 1, right) to take a random subset of size $f(x)$ of all possible attributes.

In order to combine the predictions output by the base classifiers, we take the average for regression, and apply a probability distribution vote instead of a simple majority vote for classification, as suggested by Bauer and Kohavi [23]. These combining functions generalize trivially to the multi-objective case. Each ensemble consists of 100 trees, which are unpruned [23]. For building random forests, the parameter $f(x)$ was set to $\lfloor log_2(x) + 1 \rfloor$ as in Breiman [13].

4.2 Datasets

Table 2 lists the datasets that we use, together with their properties. Most datasets are of ecological nature. Each dataset represents a multi-objective prediction problem. Of the 13 listed datasets, 8 are used both for multi-objective regression and for multi-objective classification (after discretizing the target attributes), resulting in 21 datasets in total.

4.3 Results

We assess the predictive performance of the algorithms comparing the accuracy for classification, and RRMSE (relative root mean squared error) for regression. The results are obtained by a 10-fold cross validation procedure[1], using the same folds for all experiments.

[1] When using bagging or random forests, one could also use the out-of-bag error measure [13]. In order to obtain a fairer comparison with the (non-ensemble) decision tree methods, we instead used 10-fold cross validation.

Table 2. Dataset properties: domain name, number of instances (N), number of input attributes ($Attr$), number of target attributes (T), and whether used as multi-objective classification ($Class$) or regression ($Regr$) dataset.

	Domain	Task	N	$Attr$	T	$Class$	$Regr$
E_1	Bridges [24]		85	7	5	√	
E_2	EDM - 1 [7]		154	16	2	√	√
E_3	Monks [24]		432	6	3	√	
E_4	Sigmea real [8]	with coordinates	817	6	2	√	√
E_5		without coordinates	817	4	2	√	√
E_6	Sigmea simulated [9]		10368	11	2	√	√
E_7	Soil quality 1 [10]	Acari/Coll./Biodiv.	1944	142	3		√
E_8	Solar-flare 1 [24]		323	10	3	√	√
E_9	Solar-flare 2 [24]		1066	10	3		√
E_{10}	Thyroid [24]		9172	29	7	√	
E_{11}	Water quality [11, 12]	Plants	1060	16	7	√	√
E_{12}		Animals	1060	16	7	√	√
E_{13}		Plants & Animals	1060	16	14	√	√

Table 3. Wilcoxon test outcomes (**SO** Single-Objective, **MO** Multi-Objective; **DT** Decision Tree, **Bag** Bagging, **RF** Random Forest).

Classification	Regression
MOBag > MODT	MOBag > MODT
$p = 5.14 * 10^{-6}$	$p = 2.44 * 10^{-3}$
MORF > MODT	MORF > MODT
$p = 6.61 * 10^{-7}$	$p = 2.03 * 10^{-5}$

Classification	Regression
MOBag > SOBag	MOBag > SOBag
$p = 0.301$	$p = 1.28 * 10^{-6}$
MORF > SORF	MORF > SORF
$p = 0.451$	$p = 0.094$

Here, we discuss the results along two dimensions of interest: comparing ensembles of MODTs to single multi-objective decision trees, and to ensembles of single-objective decision trees. Afterwards, we investigate ensembles of MODTs in more detail, and compare bagging and random forests in the multi-objective setting. For testing whether the difference in predictive performance between different methods is statistically significant over all datasets and all targets, we use the Wilcoxon test [25]. The results are summarized in Table 3. In the results, $A > B$ means that method A has a better predictive performance than method B. The significance is reported by the corresponding p-value.

Ensembles of Multi-Objective Decision Trees versus Multi-Objective Decision Trees. The left part of Table 3 shows the outcome of the Wilcoxon test comparing ensembles of MODTs to MODTs. The results show that the predictive performance of ensembles of MODTs is better than MODTs, which is the same as for ensembles in the single-objective setting.

A preliminary empirical evaluation of boosting of multi-objective regression trees has been performed by Sain and Carmack [26]. Experimental results on a single dataset yielded the same conclusion.

Table 4. Total model size (number of nodes) for the different methods (**SO** Single-Objective, **MO** Multi-Objective; **Bag** Bagging, **RF** Random Forest).

	Classification				Regression			
	MOBag	SOBag	MORF	SORF	MOBag	SOBag	MORF	SORF
E_1	4344	6996	4614	8910				
E_2	4102	4916	5014	5930	4780	5900	5746	7390
E_3	18580	20360	17222	22362				
E_4	29586	37906	30360	38988	46482	70896	46866	71842
E_5	29936	38082	29422	38312	46816	71660	44896	69928
E_6	6184	6544	13104	13990	153994	192038	164814	203416
E_7					53586	160506	24722	73356
E_8	5158	7742	4364	6588	9330	15562	7840	13842
E_9					23196	33264	15248	24018
E_{10}	55454	77244	83506	126916				
E_{11}	68560	137860	71258	163948	78310	221832	79606	257122
E_{12}	69484	137514	72590	164484	80034	229990	81122	267364
E_{13}	80804	275374	81568	328432	82842	451822	83036	524486

Ensembles of Multi-Objective Decision Trees versus Ensembles of Single-Objective Decision Trees. Ensembles of single-objective decision trees are ensembles that predict one target attribute. Results of the Wilcoxon test comparing a ensemble of MODTs to building ensembles for each target attribute separately are presented in the right part of Table 3. For regression, ensembles of MODTs are significantly better than ensembles of single-objective decision trees in case of bagging, and, to a lesser extent, in the case of random forests. For classification, the two methods perform comparably.

In addition, we have compared the total sizes of ensembles of multi-objective and single-objective decision trees. While the number of trees will be smaller for ensembles of MODTs (with a factor equal to the number of targets), the effect on the total number of nodes of all trees is less obvious. Table 4 presents the results. We see that ensembles of MODTs yield smaller models, with an increased difference in the presence of many target attributes.

We have also compared the running times of the different methods. Except for dataset E_1, the multi-objective ensemble method is always faster to learn than its single-objective counterpart, with an average speed-up ratio of 2.03.

Multi-Objective Bagging versus Multi-Objective Random Forests. We compared the performance of the two multi-objective ensemble methods. The test concludes that multi-objective random forests have a better predictive performance than multi-objective bagging (p-values of 0.025 for classification and 0.060 for regression). Note that, also in terms of efficiency, random forests are to be preferred, since they are faster to learn.

The obtained results are similar to results obtained in single-objective setting. In their experimental comparison, Banfield et al. [27] obtain significantly better results for random forests on 8 of 57 datasets. Also for our datasets, random

forests perform better than bagging in the single-objective case (p-values of 0.047 for classification and $2.37 * 10^{-5}$ for regression).

5 Conclusions and Further Work

In this paper, an empirical study is presented on applying ensemble methods to multi-objective decision trees. As such, the interaction between two dimensions (multi-objective learning and ensemble learning) was investigated. The results can be summarized as follows. First, the performance of a multi-objective tree learner is significantly improved by learning an ensemble (using bagging or random forests) of multi-objective trees. This suggests that the non-trivial relations that may be present between the different target attributes are preserved when combining predictions of several classifiers or when injecting some source of randomness in the learning algorithm. Second, ensembles of MODTs perform equally good as or significantly better than single-objective ones. In addition, ensembles of MODTs are faster to learn and reduce the total model size. Third, multi-objective random forests are significantly better than multi-objective bagging, which is consistent with results in the single-objective context.

As future work, we plan to extend the empirical evaluation along two dimensions: (a) to other ensemble methods, such as boosting; one research question here is how to adapt boosting's reweightening scheme to the multi-objective case; and (b) to multi-objective datasets with mixed nominal and numeric targets.

A different line of work that we consider is to develop methods for directly controlling the model diversity of predictive clustering trees. Model diversity improves the predictive performance of ensemble methods [14]. In particular, Kocev et al. [28] show that beam search with a heuristic that explicitly incorporates the diversity of the trees can be used to this end. We plan to investigate if beam search can yield more accurate ensembles than bagging or random forests.

Acknowledgements This work was supported by the EU FET IST project "Inductive Querying", contract number FP6-516169. Jan Struyf is a post-doctoral fellow of the Research Foundation - Flanders (FWO-Vlaanderen). The authors would like to thank Hendrik Blockeel for providing valuable suggestions.

References

1. Caruana, R.: Multitask learning. Machine Learning **28** (1997) 41–75
2. Blockeel, H., De Raedt, L., Ramon, J.: Top-down induction of clustering trees. In: Proc. of the 15th ICML. (1998) 55–63
3. Suzuki, E., Gotoh, M., Choki, Y.: Bloomy decision trees for multi-objective classification. In : Proc. of the 5th European Conf. on PKDD. LNAI 2160. (2001)
4. Ženko, B., Džeroski, S., Struyf, J.: Learning predictive clustering rules. In: Proc. of the Workshop on KDID at the 16th ECML. (2005)
5. Dietterich, T.: Ensemble methods in machine learning. In: Proc. of the 1th Int'l Workshop on Multiple Classifier Systems. Volume 1857 of LNCS. (2000) 1–15

6. Breiman, L.: Bagging predictors. Machine Learning **24**(2) (1996) 123–140
7. Karalič, A., Bratko, I.: First order regression. Machine Learning **26** (1997) 147–176
8. Demšar, D., Debeljak, M., Lavigne, C., Džeroski, S.: Modelling pollen dispersal of genetically modified oilseed rape within the field. Abstract presented at The Annual Meeting of the Ecological Society of America. (2005)
9. Džeroski, S., Colbach, N., Messean, A.: Analysing the effect of field characteristics on gene flow between oilseed rape varieties and volunteers with regression trees. In: Proc.of the 2nd Int'l Conference on Co-existence between GM and non-GM based agricultural supply chains. (2005)
10. Demšar, D., Džeroski, S., Larsen, T., Struyf, J., Axelsen, J., Pedersen, M., Krogh, P.: Using multi-objective classification to model communities of soil microarthropods. Ecological Modelling **191**(1) (2006) 131–143
11. Blockeel, H., Džeroski, S., Grbović, J.: Simultaneous prediction of multiple chemical parameters of river water quality with Tilde. In: Proc. of the 3rd European Conference on PKDD. Volume 1704 of LNAI, Springer (1999) 32–40
12. Džeroski, S., Demšar, D., Grbović, J.: Predicting chemical parameters of river water quality from bioindicator data. Applied Intelligence **13**(1) (2000) 7–17
13. Breiman, L.: Random forests. Machine Learning **45**(1) (2001) 5–32
14. Hansen, L., Salamon, P.: Neural network ensembles. IEEE Trans. on Pattern Anal. and Mach. Intell. **12** (1990) 993–1001
15. Freund, Y., Schapire, R.E.: Experiments with a new boosting algorithm. In: Proc. of the 13th ICML, Morgan Kaufmann (1996) 148–156
16. Breiman, L.: Using adaptive bagging to debias regressions. Technical report, Statistics Department, University of California, Berkeley (1999)
17. Ho, T., Hull, J., Srihari, S.: Decision combination in multiple classifier systems. IEEE Trans. on Pattern Anal. and Mach. Intell. **16**(1) (1994) 66–75
18. Kittler, J., Hatef, M., Duin, R., Matas, J.: On combining classifiers. IEEE Trans. on Pattern Anal. and Mach. Intell. **20**(3) (1998) 226–239
19. Breiman, L., Friedman, J., Olshen, R., Stone, C.: Classification and Regression Trees. Wadsworth, Belmont (1984)
20. Blockeel, H., Schietgat, L., Struyf, J., Džeroski, S., Clare, A.: Decision trees for hierarchical multilabel classification: A case study in functional genomics. In: Proc. of the 10th European Conference on PKDD. LNAI vol. 4213. (2006)
21. Clare, A., King, R.: Knowledge discovery in multi-label phenotype data. In: Proc. of the 5th European Conference on PKDD. LNAI vol. 2168. (2001)
22. Struyf, J., Džeroski, S.: Constraint based induction of multi-objective regression trees. In: KDID, 4th Int'l Workshop. LNCS vol. 3933. (2006)
23. Bauer, E., Kohavi, R.: An empirical comparison of voting classification algorithms: Bagging, boosting, and variants. Machine Learning **36** (1999) 105
24. Hettich, S., Blake, C.L., Merz, C.J.: UCI repository of machine learning databases (1998)
25. Wilcoxon, F.: Individual comparisons by ranking methods. Biometrics **1** (1945)
26. Sain, R.S., Carmack, P.S.: Boosting multi-objective regression trees. Computing Science and Statistics **34** (2002) 232–241
27. Banfield, R., Hall, L., Bowyer, K., Kegelmeyer, W.: A comparison of decision tree ensemble creation techniques. IEEE Trans. on Pattern Anal. and Mach. Intell. **29**(1) (2007) 173–180
28. Kocev, D., Džeroski, S., Struyf, J.: Beam search induction and similarity constraints for predictive clustering trees. In: 5th Int'l Workshop on KDID: Revised Selected and Invited Papers. (2007) To appear.