

Abstract

While ordinal classification problems are common in many situations, induction of ordinal decision trees has not evolved significantly. Conventional trees for regression settings or nominal classification are commonly induced for ordinal classification problems. On the other hand a decision tree consistent with the ordinal setting is often desirable to aid decision making in such situations as credit rating.

In this work we extend a recently proposed strategy based on constraints defined globally over the feature space. We propose a bootstrap technique to improve the accuracy of the baseline solution. Experiments in synthetic and real data show the benefits of our proposal.

1 Introduction

Machine learning is playing a central role in deployment of the so-called intelligent systems, where inductive learning techniques are usually used to induce a general rule from a set of observed instances. Among the wide family set of inductive learning schemes, classification is of fundamental importance. With classification one is interested on finding a mapping from a point (or observation) in \mathbb{R}^d to a value from a finite set. Depending on the problem, this output space can be composed by a set of only two or $K > 2$ elements, the binary or multiclass problem, respectively. The multiclass problem can be further subdivided into the nominal and the ordinal problem. For instance, problems like credit scoring is an example of ordinal problems where data is structured by a “natural” order, consisting on the grading of a customer credit profile in the scale Excellent \succ Good \succ Fair \succ Poor.

Imposing ordinality during the model construction of interpretable learning schemes like Decision Trees (DTs) is not straightforward. In [2] it was proposed to impose ordinality after the training taking place through regions relabelling. Here we recover the work proposed in [2] in order to diminish the over-regularisation issue identified by the authors. Through the usage of ensemble learning techniques, we can fuse the set of resultant trees into a single one. By applying a new formulation for the global constraints in order to impose the order, we can avoid over-regularised output decision regions.

2 Imposing Ordinality on Decision Trees

Different studies have proposed different adaptations to learning schemes to cope with the ordinality setting, ranging from support vector machines, to neural networks or Gaussian Processes. Hierarchical models, like DTs, pose a difficult challenge: the fact that they are designed as a sequence of local decisions raises difficulties when trying to incorporate the information about the order in the learning process. In [2] we paved the way towards a more generic setting for these kind of problems, arguing that the order information is a global property, i.e., it involves a relation between all data, and should therefore be the result of optimising some global function.

2.1 Consistency

Let $f(\mathbf{x})$ be a decision rule that assigns each value of \mathbf{x} to one of the available classes¹. Such a rule will divide the input space into regions \mathcal{R}_k called decision regions, such that all points in \mathcal{R}_k are assigned to class C_k . The boundaries between decision regions are called decision boundaries or decision surfaces. Note that each decision region need not

be contiguous but could comprise any number of disjoint regions. Intuitively, for ordinal data, in a sufficiently small neighbourhood of \mathbf{x} , $\mathcal{V}_\varepsilon(\mathbf{x})$, the decision function should only take at most two consecutive values: $\max f(\mathbf{x}) - \min f(\mathbf{x}) \leq 1$. The motivation for this is that a small change in the input data should not lead to a ‘big jump’ in the output decision. Therefore, we say that a decision function is *consistent* with an ordinal data classification setting in a point \mathbf{x}_0 if $\exists \varepsilon > 0 \forall \mathbf{x} \in \mathcal{V}_\varepsilon(\mathbf{x}_0) \max f(\mathbf{x}) - \min f(\mathbf{x}) \leq 1$. A decision function is consistent in the whole input space if the above condition is verified for every point in the input space: $\forall \mathbf{x}_0 \exists \varepsilon > 0 \forall \mathbf{x} \in \mathcal{V}_\varepsilon(\mathbf{x}_0) \max f(\mathbf{x}) - \min f(\mathbf{x}) \leq 1$.²

2.1.1 Imposing the ordinal constraints in a decision function

Consistency is a global property, i.e., it involves a relation between different decision regions of the space. A key challenge is how to use this information during the design process of a learning algorithm.

It is convenient at this point to define some notation to describe the assignment of labels to different decision regions. Let \mathcal{R}_n , $n = 1, \dots, N$, represent the contiguous decision regions created by some model³. For each region \mathcal{R}_n we introduce a corresponding set of binary indicator variables $x_{n,k} \in \{0, 1\}$, where $k = 1, \dots, K - 1$ describing which of the K ordinal labels is assigned to region \mathcal{R}_n , so that if data points in \mathcal{R}_n are assigned the label k then $x_{n,j} = 1$ for $j < k$, and $x_{n,j} = 0$ otherwise. So, for instance if we have a setting with 5 classes, $K = 5$, and to a particular region happens to be assigned the label 3, then \mathbf{x} will be represented by $\mathbf{x} = [1 \ 1 \ 0 \ 0]^T$. Note that this is different from the often used 1-of- K coding scheme and we find it more convenient for the introduction of the constraints in what follows.

In ordinal data settings, the loss associated with a region \mathcal{R}_n when deciding for class C_k is usually captured with the absolute error, the sum over all points lying in \mathcal{R}_n of the absolute difference between the true class of the point and the predicted class for the region: $c_{n,k} = \sum_{i=1}^K |i - k| p_{n,i}$, where $p_{n,i}$, $n = 1, \dots, N$, $i = 1, \dots, K$ represent the number of observations (from the data used in creating the region by some learning algorithm) from class k satisfying the conditions for region \mathcal{R}_n , (that is, lying inside \mathcal{R}_n). Nevertheless, the following model is generic for any costs $c_{n,k}$.

The optimal labelling of the regions can then be found by minimising the following objective function: $J = \sum_{n=1}^N \sum_{k=1}^{K-1} c_{n,k} (x_{n,k-1} - x_{n,k})$, where the *constants* $x_{n,0} = 1$ and $x_{n,K} = 0$ have been introduced for notational convenience, with the constraints

$$x_{n,k+1} - x_{n,k} \leq 0, k = 1, \dots, K - 2, \quad n = 1, \dots, N \quad (1)$$

and

$$x_{n,k} \in \{0, 1\}, k = 1, \dots, K - 1, \quad n = 1, \dots, N \quad (2)$$

Now, we want to impose that adjacent regions have labels that differ at most by one. Therefore we are led to the optimisation of the loss of the decision function constrained by the consistency of it. Consistency imposes that, for any pair of adjacent regions \mathcal{R}_n and $\mathcal{R}_{n'}$, the following inequalities must be verified:

$$\sum_{k=1}^{K-1} x_{n,k} - \sum_{k=1}^{K-1} x_{n',k} \leq 1, \quad \sum_{k=1}^{K-1} x_{n',k} - \sum_{k=1}^{K-1} x_{n,k} \leq 1 \quad (3)$$

²This definition of consistency precludes decision functions such as $f(x) = 1, x < 0; f(x) = 2, x = 0; f(x) = 3, x > 0$, where the region corresponding to class 2 is a measure-zero set.

³Note the change of notation: so far we have used \mathcal{R}_k to represent the decision region, contiguous or not, corresponding to class C_k . From now on \mathcal{R}_n just represents a continuous region of the space with all points inside that region being assigned the same class. Therefore, different regions \mathcal{R}_n and \mathcal{R}_m may be assigned the same class and the number of regions is likely greater than the number of classes.

¹A remark should be made. Since we are dealing with ordered classes, we shall consider that the output of the decision function is one of the K labels $\{C_1, \dots, C_K\}$ or one number in $\{1, \dots, K\}$ resulting from the bijective map $g: \{C_i\}_{i=1}^K \rightarrow \{1, \dots, K\}$ which assigns the number k to the class C_k , i.e., $g(C_k) = k$. The context should make it clear which of the two output formats is being considered.

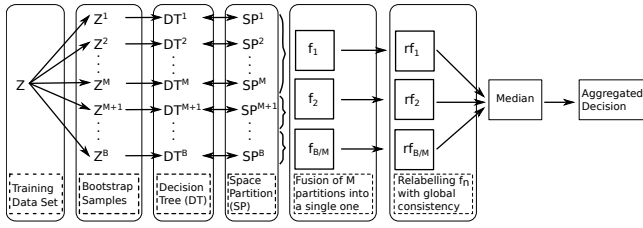


Figure 1: Schematic of the proposed aggregation process.

2.1.2 Avoiding Over-Regularised Decision Spaces

One way to try overcoming the over-regularization problem is to force an over-partition of the space prior to the relabelling for global consistency through resampling techniques. One would expect that the global optimisation would then compensate this initial over-refinement. In here we explore the resampling approach on the context of ensemble learning.

Although the bootstrap technique is a general tool for assessing statistical accuracy, it can also be used to improve the accuracy of a prediction scheme. The basic idea is to randomly draw datasets with replacement from the training data, each sample the same size as the original training set. This is done B times ($B = 100$ say), producing B bootstrap datasets. Then we fit a DT to each of the bootstrap datasets. Typically bootstrap aggregation or bagging would then select the class with the most “votes” from the B DTs. In here we will consider the option of working directly with the partition of the space corresponding to each DT.

Instead of bagging directly the output of the B DTs we propose to group first the B DTs in groups of M DTs and to compute the fusion (intersection) of the M corresponding space partitions, see Figure 1. Each fused partition will then be relabelled according to the consistency optimisation procedure described earlier. Finally, we bag the relabelled models. Since we are dealing with ordinal data, we use the median of the B/M votes as the final decision.

Global consistency with empty regions

The fusion mechanism is likely to produce empty regions, i.e., regions without instances from the training set. A direct consequence is that the optimisation procedure provided early becomes ill-defined, in the sense that there are multiple optimal labellings. In fact, any relabelling of the empty regions that is still consistent does not change the value of the objective function. We set additional constraints on the labels of the empty regions so that the optimisation problem becomes again well defined. The constraints given in Equation (3) are re-written for pairs of regions involving empty regions as in Equation (4):

$$\sum_{k=1}^{K-1} x_{n,k} - \sum_{k=1}^{K-1} x_{n',k} \leq \delta_{(n,n')} \quad \forall (n,n') \in \Delta \quad (4)$$

$$\sum_{k=1}^{K-1} x_{n',k} - \sum_{k=1}^{K-1} x_{n,k} \leq \delta_{(n,n')} \quad \forall (n,n') \in \Delta \quad (5)$$

$$\delta_{(n,n')} \in \{0, 1\} \quad \forall (n,n') \in \Delta$$

where Δ contains all empty adjacent regions.

Intuitively, empty regions adjoin with non-empty regions should share the label of the non-empty region. The rationale is similar to the margin maximisation of other learning schemes, putting the transition between labels further away from the data points. Therefore, pairs of empty regions should have a lower penalty than pairs which have exactly one empty region. Letting Δ_1 be the set containing only pairs of empty regions and Δ_2 the set of pairs which have exactly one empty region⁴, we penalise differently the deviation of the aforementioned objective:

$$J = \sum_{n=1}^N \left\{ c_{n,1} + \sum_{k=1}^{K-1} x_{n,k} (c_{n,k+1} - c_{n,k}) \right\} + C_1 \sum_{(n,n') \in \Delta_1} \delta_{(n,n')} + C_2 \sum_{(n,n') \in \Delta_2} \delta_{(n,n')}, \quad (6)$$

with $C_2 > C_1 > 0$. Both C_1 and C_2 controls the tradeoff between the smoothness over the labels of the empty regions. We defined C_1 with

⁴ $\Delta = \Delta_1 \cup \Delta_2$ and $\Delta_1 \cap \Delta_2 = \emptyset$.

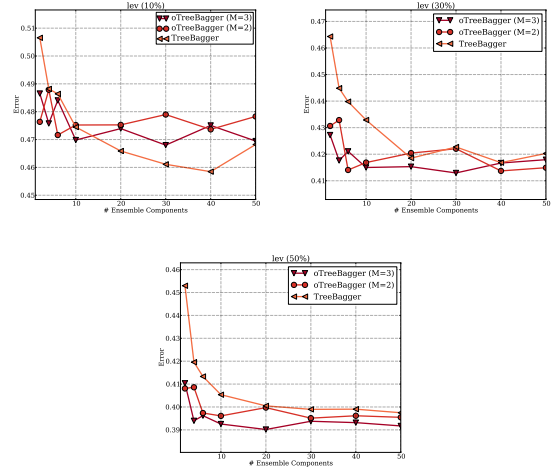


Figure 2: Results for real dataset. Models trained with 10%, 30% and 50% of the 1000 instances in the left, centre and right plots, respectively.

value of $1/(N(K-1))$ and C_2 with $1/(N(K-1)0.9)$. The factor 0.9 was set empirically. The formulation presented in Equation (6) constrained to (1), (2), (3), (4) and (5) in conjugation with the aggregation approach represented in Figure 1, results in our proposal titled *oTreeBagger*.

3 Experiments

Our experiments were conducted in a real ordinal data, LEV [1]. LEV dataset contains examples of anonymous lecturer evaluations, taken at the end of MBA courses and is composed by 4 features and 5 classes.

The baseline method (*TreeBagger*) used in our experiments consisted on the bagging approach with decision trees available in MatlabTM Statistical Toolbox. We opted to use the Gini index as splitting criterion. The grouping size M was evaluated from 1 to 5. The results presented in Figure 2 show only the performance for a subset of these values for easier interpretation of the results. In these figures it is also clear the evolution of the learners throughout the increasing number of ensemble components. Due to the sensibility of these learners in regards to the number of training instances used, we conducted our experiments in 10%, 30% and 50% of training data. Our proposal outperformed the standard ensemble learner obtaining considerable gains in terms of performance. Logically, when the number of training instances increases this gain is more subtle, though.

4 Conclusion

Learning on ordinal data has challenged many researchers to unfold the natural structure of the problem which, at the end, could lead to better performance results when compared with standard learning mechanisms. Despite the literature already presenting a rich collection in what concerns to this problem, there still exists a gap related to some classical methods. Decision trees are one example of it. Being well known and widely used within the machine learning community, as well the advantage of the interpretable capability, it is not straightforward its mapping towards ordinal data problems. In this work we proposed an improvement of [2] in order to reduce the over-regularised decision regions artifact through the usage of ensemble learning techniques. Results shown the benefits of our proposal in terms of accuracy gained when compared to a standard ensemble learning technique. Further studies will be taken to reduce the number of variables and constrains towards complexity diminution.

References

- [1] Arie Ben-David and Leon Sterling. Generating rules from examples of human multiattribute decision making should be simple. *Expert Systems with Applications*, 31(2):390 – 396, 2006.
- [2] Jaime S. Cardoso and Ricardo Sousa. Classification models with global constraints for ordinal data. In *Proceedings of The Ninth International Conference on Machine Learning and Applications*, 2010.