Tree based ensemble models regularization by convex optimization

Bertrand Cornélusse, Pierre Geurts and Louis Wehenkel
Department of Electrical Engineering and Computer Science
University of Liège
B-4000 Liège, Belgium
{bertrand.cornelusse, p.geurts, louis.wehenkel}@ulg.ac.be

Abstract

Tree based ensemble methods can be seen as a way to learn a kernel from a sample of input-output pairs. This paper proposes a regularization framework to incorporate non-standard information not used in the kernel learning algorithm, so as to take advantage of incomplete information about output values and/or of some prior information about the problem at hand. To this end a generic convex optimization problem is formulated which is first customized into a manifold regularization approach for semi-supervised learning, then as a way to exploit censored output values, and finally as a generic way to exploit prior information about the problem.

1 Motivation

In the standard setting, supervised learning aims at inferring a predictive model mapping an input space $\mathcal{X}$ to an output space $\mathcal{Y}$ given a completely labeled sample of input-output pairs. However, in many applications the available output information for a set of input points is incomplete. For example, in the setting of semi-supervised learning the output is simply unknown for a subset of the provided inputs, but under some assumptions taking into account these unlabeled inputs allows the induction of better predictive models. Between these extreme settings, in the context of censored data some outputs are only partially specified for a subset of the given inputs in the form of a range of possible values (e.g. typically a lower bound on the life-time, in the context of survival data). In other contexts, additional prior knowledge about the target problem is given in the form of hard or soft constraints. In all these cases, one would like to exploit all the available information together with the complete sample of input-output pairs so as to infer better predictive models.

In this paper, a general framework is proposed for the regularization of tree based ensemble models. It exploits a kernel formulation of tree-based predictors and is formulated as a convex optimization problem where the incomplete data and/or prior knowledge is used as extra information to regularize the model. Semi-supervised learning and learning from censored data fit naturally into this general framework. However, other kinds of information can be used, like prior information on the measurement accuracy on the outputs or specific constraints on output values which must be represented in the model. Relations between input-output pairs can also be imposed as well as some other kinds of structural properties about the problem.

The convex optimization formulation is presented in Section 2, as well as the consequences in terms of problem complexity. In Section 3, learning from censored data and manifold regularization for incorporating unlabeled data are cast in the general formulation, and the way to incorporate other types of information is also discussed. Section 4 exposes the related work, while Section 5 concludes.

2 Regularizing an ensemble of regression trees

After an intuitive description of the nature of the problems addressed, the principles of the induction of ensembles of regression trees are recalled and their regularization is formulated as a convex optimization problem which is discussed in terms of modeling capacity and solution complexity.

2.1 Nature of the problem

We consider the supervised learning framework, where we typically seek to infer a function $f(\cdot) : \mathcal{X} \to \mathcal{Y}$ from a completely labeled training sample of input-output observations $\{(x_i, y_i)\}_{i=1}^n$. For convenience, we consider the context of regression, where $\mathcal{Y} \subseteq \mathbb{R}$.

In many cases, additional information can be useful in this inference process. E.g., if for some points $\{(x_i, y_i)\}_{i=n+1}^{n+c}$ the output is censored, for example right censored, we would wish to regularize f such that $f(x_i) \geq y_i$, $\forall i \in \{n+1, ..., n+c\}$. In semi-supervised learning, we have a (typically large) number of input points $\{x_i\}_{i=n+1}^{n+u}$ without their associated outputs. We then would like to exploit regularity assumptions about the input-output relation to bias the learning of the mapping f from the complete data. But it might also happen that the output targeted by the learning process is a priori known to satisfy equality constraints for some particular inputs, or even over some given regions of the input space[1] (e.g. "$f(x) \geq Ax + b$, whenever $Bx \leq c$", cf. [1]). More generally, the additional information at hand might also entail more complex relations involving input-output pairs at several places (e.g. "$f(x_k) \geq f(x'_k), \forall k = 1, \ldots, K$"). Another example is in multiple or structured output prediction if individual models $f^j(\cdot)$ are fitted for each individual output y^j. E.g., if $\mathcal{Y} \subseteq \mathbb{R}^2$ then we may wish to couple the individual models to better respect the known structure of their output relations, so as to satisfy constraints such as "$f^1(x) \geq f^2(x)$".

2.2 Tree-based ensemble methods

In this paper, we consider the incorporation of prior knowledge and incompletely labeled samples in the forms suggested in Section 2.1 into tree-based supervised learning methods. The general idea of regression trees is to recursively split the training sample with tests based on the input space description x, trying at each split to reduce as much as possible the variation of the output y in the left and right subsamples of learning cases corresponding to that split. The splitting of a node is stopped when the output y is constant in the subsample of this node or when some other stopping criterion is met (e.g., the size of the local subsample is smaller than $n_{min} \in \mathbb{N}$). To each leaf a label is attached so as to minimize the empirical error, which in least squares regression trees is the local subsample average of outputs. While useful for their interpretability, single trees are usually not competitive with other methods in terms of accuracy, essentially because of their high variance. Thus, ensemble methods have been proposed to reduce variance and thereby improve accuracy. In general, these methods induce an ensemble of M diverse trees and then combine their predictions to yield a final prediction as a weighted average of the predictions of the individual trees.

In the following, trees are indexed by the letter t, l_t is the number of leaves in tree t, $l_{t,i}(x)$ is the leaf indicator function[2] and $n_{t,i}$ is the number of labeled objects reaching leaf i. Let $l_t(x) = \left(l_{t,1}(x)/\sqrt{n_{t,1}}, \ldots, l_{t,l_t}(x)/\sqrt{n_{t,l_t}}\right)^T$, then the prediction of one tree is $\sum_{i=1}^n y_i l_t^T(x_i) l_t(x)$. If we concatenate the vectors of leaves of the M trees, $l(x) = \left(w_1 l_1^T(x), \ldots, w_M l_M^T(x)\right)^T \in \mathbb{R}^p$, where $w_i \geq 0 \; \forall i \in \{1, \ldots, M\}$ are weights associated to the trees such that $\sum_{i=1}^M w_i = 1$, the prediction of the ensemble is $\sum_{i=1}^n y_i l^T(x_i) l(x)$. Equivalently, if we define the kernel $K(x, x') = l^T(x) l(x')$,

$$f(x) = \sum_{i=1}^n y_i K(x_i, x).$$

[1] In what follows, (in)equality relations are to be understood as component-wise when they apply to vectors.

[2] $l_{t,i}(x) = 1$ if x reaches leaf i of tree t, $l_{t,i}(x) = 0$ otherwise.

2.3 Regularization of a tree ensemble model

To incorporate the information contained in the incomplete data and/or prior knowledge, we must modify the model described in Section 2.2. In order to remain as generic as possible, we choose not to modify the tree induction algorithm and the way the kernel function K is computed (e.g. bagging [2], random forests [3], extra-trees [4], boosting [5], etc). Thus we choose not to modify the structure of the trees, i.e. the way splits are selected at their internal nodes, but rather to modify the labels assigned to their leaves. This can be interpreted as a regularization of the model generated by the tree induction algorithm which assigns constant values to regions of the input space, by the correction of these assignments through the resolution of an optimization problem. To this end we consider two possibilities: to modify the vector y of training sample outputs and /or to add a bias to the labels attached to the leaves of the trees. The first way allows to correct the y_i values when they are corrupted by noise and the second way arises when we do not want to modify these values.

To formulate the corresponding regularization problem, we introduce a vector of decision variables to denote the leaf biases $\Delta z \in \mathbb{R}^p$, a vector of modifications to the training sample outputs $\Delta y \in \mathbb{R}^n$ and a vector of auxiliary variables $\nu \in \mathbb{R}^n_+$, and we denote by $K \in \mathbb{R}^{n \times n}$ the gram matrix of the training sample, i.e. $K_{ij} = K(x_i, x_j)$ and by L the sample partitioning matrix of the ensemble:[3]

$$L = \begin{pmatrix} l^T(x_1) & \dots & l^T(x_n) \end{pmatrix}^T \in \mathbb{R}^{n \times p}.$$

We also denote by $\Omega(\cdot, \cdot, \cdot) : \mathbb{R}^n \times \mathbb{R}^p \times \mathbb{R}^n \to \mathbb{R}$ a convex function used to express generically various compromises in terms of regularization, and by $\mathcal{C} \subseteq \mathbb{R}^{n+p+n}$ a convex set used to express hard constraints. Given these notations, we formulate the following optimization problem.

Formulation 1 (General formulation)

$$\begin{aligned} \min \quad & \Omega(\Delta y, \Delta z, \nu) && (1) \\ \textit{s.t.} \quad & -\nu \leq Ky + K\Delta y + L\Delta z - y \leq \nu && (2) \\ & (\Delta y, \Delta z, \nu) \in \mathcal{C}. && (3) \end{aligned}$$

The inequality constraints (2) aim at keeping the prediction error on the training sample low through the vector ν which norm is penalized in (1) and/or constrained in (3). The information of the incomplete data or prior knowledge may be expressed in the constraints (3) and in the objective (1).

In formulation 1, we express the fact that we want to regularize the model by incorporating the information from the complete training sample, the incomplete data, and the prior knowledge, by assuming that these may be expressed by a finite number of constraints on the vectors $\Delta y, \Delta z$ and ν and/or by an appropriate choice of the objective function. In general, a trade-off must however be defined between the regularization induced by the prior and the incomplete data and the error on the complete training sample. Notice also that without prior knowledge or incomplete data, this formulation allows to globally (re)optimize the leaf labels so as to minimize the error on the training sample without affecting too much the original model, in a way depending on the definition of Ω.

2.4 Problem dimensions and computational complexity

Formulation 1 contains $p + 2n$ variables, and $2n$ linear constraints, without taking into account the constraints defining $\mathcal{C}$. For a balanced tree t built from a finite sample size n, the number of leaves l_t is on the order of n/n_{min}, thus $p \approx Mn/n_{min}$. High dimensional problems formulated as LP can be solved in polynomial time. Anyway if the problem is nonlinear but convex it might still be solved in polynomial time. Depending on the complexity of the ensemble of trees on which the optimization problem is formulated, and on the parameter choices, the problem might not be feasible, or might be feasible only at the price of a significant increase of the error on the training sample. This would be the case if there were not enough degrees of freedom in the model to incorporate the incomplete data. A solution would be to penalize constraint (3) violations in (1).

[3] Up to some normalization, the line i of matrix L essentially indicates the leaves reached by the sample i.

3 Learning from incomplete data

3.1 Censored data

Censored data arise frequently, e.g. in survival analysis where one is interested in the survival time of patients after the inception of a treatment. In this context, it often arises that people leave the study at a given instant t_0 for reasons independent of the disease, i.e. their survival time is only known to be larger than t_0. Here we try to use this partial information, as [6] did using support vector regression for censored data (SVCR). To this end, we learn a tree-based kernel function[4] K on the subset of uncensored data $\{(x_i, y_i)\}_{i=1}^{n}$ and then impose the information contained in the censored data $\{(x_i, y_i)\}_{i=n+1}^{n+c}$ thanks to Formulation 2, a particular case of Formulation 1.

Formulation 2 (Censored data formulation)

$$\begin{aligned} \min \quad & C_1\|\Delta y\| + C_2\|\Delta z\| + C_3\|\nu\| + C_4\|\nu^c\| && (4)\\ \textit{s.t.} \quad & -\nu \leq Ky + K\Delta y + L\Delta z - y \leq \nu && (5)\\ & -\nu^c \leq K^c y + K^c\Delta y + L^c\Delta z - y^c, && (6) \end{aligned}$$

where y^c denotes the vector of censored outputs, $\nu^c \in \mathbb{R}_+^c$ is a vector of auxiliary variables, $K^c \in \mathbb{R}^{c\times n}$ with $K^c_{ij} = K(x_i, x_j),\ \forall i \in \{n+1, \ldots, n+c\}$ and $\forall j \in \{1, \ldots, n\}$, and

$$L^c = \begin{pmatrix} l(x_{n+1}) & \ldots & l(x_{n+c}) \end{pmatrix}^T \in \mathbb{R}^{c\times p}.$$

Constraints (6) with the term $C_4\|\nu^c\|$ of Ω imply that for censored objects an excessive prediction is not penalized. We did not use hard constraints here for the censored data for feasibility reasons. Since in survival data the sample outputs are in principle measured with high accuracy, we penalized $\|\Delta y\|$ very strongly, but we could as well have removed these variables from the formulation.

We compared this approach (Table 1) to unregularized tree-based ensemble methods (denoted by ET and ET*) and to the SVCR algorithm presented in [6]. We analyzed the four real life data sets from the R package "survival" on which SVCR is tested in [6], and evaluated the error measure $MAE = \frac{1}{n+c}\left(\sum_{i=1}^{n} |y_i - f(x_i)| + \sum_{i=n+1}^{n+c} \max(0, y_i - f(x_i))\right)$ by 5-fold cross-validation. ET was used to compute the gram matrix K of formulation 2. In ET* the censored points are included in the training sample and handled as uncensored points. For SVCR we used a Gaussian kernel. The parameters of these methods are tuned by grid search while using only the training samples. We observe that exploiting the censored data via Formulation 2 may indeed improve significantly the quality of the predictors. This is especially remarkable in the "nwtco" data set where the proportion of censored outputs is very high. We also notice that using regularized tree-based predictors actually outperforms, sometimes quite strongly, the SVCR approach to these problems.

Table 1: Comparison of unregularized/regularized tree-based predictors with SVCR (first row gives the percentage of censored data in each data set). The values reported are the average MAE over the 5 folds $\pm$ one unit standard deviation; † indicates that SVCR scores are reproduced from [6].

	lung (28%)	veteran (7%)	heart (56%)	nwtco (86%)
ET	144 $\pm$ 11	85 $\pm$ 22	146 $\pm$ 57	1888 $\pm$ 72
ET*	117 $\pm$ 17	84 $\pm$ 19	78 $\pm$ 13	224 $\pm$ 38
ET + Formulation 2	113 $\pm$ 13	81 $\pm$ 34	68 $\pm$ 23	98 $\pm$ 13
SVCR	144 $\pm$ 14	80 $\pm$ 28	138 $\pm$ 48	476†

3.2 Manifold regularization for semi supervised learning

We show how the scheme presented in [7] may be casted in our formulation to yield semi-supervised and/or transductive tree learning algorithms; in [7] an adjacency graph among samples is inferred from similarities of their inputs, and the scheme regularizes predictions according to this graph. Let

[4] Using the "Extremely randomized trees" algorithm described in [4] and named ET in the sequel.

$\{x_i\}_{i=1}^{n+m}$ denote the inputs of our sample, and suppose that we are given the output labels $\{y_i\}_{i=1}^{n}$ only for the first n of them. We suppose also to be given an ensemble of M tree structures and a similarity graph over the whole sample (we denote by $\mathcal{L}$ its $(n+m) \times (n+m)$ Laplacian).

Our objective is to compute ensemble predictions over the complete sample that are "regular" with respect to the similarity graph. To this end we exploit the full set of inputs $\{x_i\}_{i=1}^{n+m}$ in each tree $t \in \{1, \ldots, M\}$ to compute the values $n_{t,i}$, $\forall i = 1, \ldots, l_t$ so as to define the ensemble kernel $K'(\cdot, \cdot)$, from which we compute the gram matrix $K' \in \mathbb{R}^{(n+m)\times(n+m)}$ over $\{x_i\}_{i=1}^{n+m}$:

$$K' = \begin{pmatrix} K'_{\ell\ell} & K'_{\ell u} \\ K'_{\ell u} & K'_{uu} \end{pmatrix},$$

where $K'_{\ell\ell}$ $K'_{\ell u}$ and K'_{uu} are submatrices corresponding respectively to the kernel evaluations between labeled, between labeled and unlabeled, and between unlabeled cases. We use Formulation 3 to compute predictions regularized along the similarity graph:

Formulation 3 (Manifold regularization for semi-supervised learning)

$$\begin{aligned} \min \quad & \|\nu\|_2^2 + C(y + \Delta y)^T K' \mathcal{L} K' (y + \Delta y) && (7) \\ \textit{s.t.} \quad & -\nu \leq (K'_{\ell\ell} | K'_{\ell u})\,(y + \Delta y) - y_\ell \leq \nu && (8) \\ & \Delta y_\ell = 0, && (9) \end{aligned}$$

where $y = (y_\ell^T, y_u^T)^T \in \mathbb{R}^{n+m}$ denotes a vector of output labels obtained by completing the n given labels with m labels equal to zero ($y_u = 0$), and $\Delta y = (\Delta y_\ell^T, \Delta y_u^T)^T$.

Note that in this formulation, we do not allow to adjust the given output labels (9) nor use leaf biases Δz, but in (8) the outputs Δy_u contribute to the predictions for the labeled sample $\{x_i\}_{i=1}^{n}$.

3.3 Other types of prior knowledge and objectives

Obviously, the formulations 2 and 3 could be merged to handle both types of data in a common formulation. However, formulation 1 is not limited to the incorporation of incomplete data. For example, to approximate the dynamics of a non-linear system with known equilibrium points $(x_j^\star, y_j^\star)$ $\forall j \in \mathcal{J}$, it is possible to force the value $f(x)$ for these points by expressing (3) as

$$\sum_{i=1}^{n} K(x_i, x_j^\star)(y_i + \Delta y_i) + l(x_j^\star)^T \Delta z = y_j^\star,\ \forall j \in \mathcal{J}. \tag{10}$$

We have observed experimentally that adding such constraints may significantly enhance the precision of the inferred model. We have also successfully used our framework to incorporate other types of prior knowledge, such as relations between different output space dimensions.

Also, ϵ-insensitive formulations may be handled by the incorporation of appropriate constraints in the set $\mathcal{C}$. These may be used to inject prior knowledge about the accuracy of the sensors used to measure the output values or to trade-off empirical accuracy with generalization performance.

4 Related work

Many developments in supervised learning can be considered as the incorporation of (more or less explicit) constraints on the learned input-output map. Model regularization imposes global constraints on the smoothness of input-output maps and semi-supervised learning [8] imposes local constraints among the predictions at nearby samples derived from similarity measures.

In this paper, we have focused on the regularization of tree-based models by the incorporation of incomplete data (and possibly other sources of additional prior knowledge about the problem) into predictive models. To the best of our knowledge, there is no related work using tree-based learning algorithms. In support vector machines, the explicit incorporation of constraints has already received a lot of attention (see [9, 10] and the references therein). The definition of the model derived from these methods as the solution of a convex (quadratic or linear) optimization problem indeed makes the incorporation of regularization terms and additional constraints natural. At first sight, one main

advantage of these approaches with respect to tree-based ones is the simultaneous handling of both fitting the training data and satisfying the constraints, whereas, in our case, the optimization only acts as a corrector for the tree-based learning algorithm. However, our Formulation 1 incorporates the quality of the fitting of the training data, meaning that it could be able to learn a useful model even if the initial tree model is not determined from the training data (but for example randomly built). Furthermore, the tree-based ensemble methods allow to learn a kernel over the input-space from the data in a supervised way, contrary to many approaches which assume that this kernel is given. Additionally, exploiting a learned tree model may take benefit of the main advantages of tree-based methods, such as for example their embedded feature selection mechanism.

5 Conclusion and further work

We have proposed a generic extension of tree-based ensemble methods which allows to incorporate incomplete data but also prior knowledge about the problem. The framework is based on a convex optimization problem allowing to regularize a tree based ensemble model by adjusting either (or both) the labels attached to the leaves of an ensemble of regression trees or the outputs of the observations of a training sample. It allows to incorporate weak additional information in the form of partial information about output labels (like in censored data or semi-supervised learning) or – more generally – to cope with observations of varying degree of precision, or strong priors in the form of structural knowledge about the sought model. In addition to enhancing the precision by exploiting additional information, the proposed approach may be used to produce models which naturally comply with feasibility constraints which need to be satisfied in many practical decision making problems, specially in contexts where the output space is of high-dimension and/or structured by invariances, symmetries and other kinds of constraints.

Further work will aim at validating these ideas on practical problems and incorporating them within the algorithms used to grow ensembles of trees.

Acknowledgements

This paper presents research results of the Belgian Network BIOMAGNET (Bioinformatics and Modeling: from Genomes to Networks), funded by the Interuniversity Attraction Poles Programme, initiated by the Belgian State, Science Policy Office. BC thanks FRIA (Belgian Fund for Research in Industry and Agriculture) for allowing him to carry out this research. PG is a research associate of the F.R.S.-FNRS. The scientific responsibility rests with the authors.

References

[1] O.L. Mangasarian, J.W. Shavlik, and E.W. Wild. Knowledge-based kernel approximation. *Journal of Machine Learning Research*, 5:1127–1141, 2004.

[2] L. Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, 1996.

[3] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, October 2001.

[4] P. Geurts, D. Ernst, and L. Wehenkel. Extremely randomized trees. *Machine Learning*, 36(1):3–42, 2006.

[5] Y. Freund and R. E. Schapire. Experiments with a new boosting algorithm. In *Proceedings of the Thirteenth International Conference on Machine Learning*, pages 148–156, 1996.

[6] P.K. Shivaswamy, Wei Chu, and M. Jansche. A support vector approach to censored targets. In *Seventh IEEE International Conference on Data Mining*, pages 655–660, Oct. 2007.

[7] M. Belkin, P. Niyogi, and V. Sindhwani. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *JMLR*, 7:2399–2434, 2006.

[8] O. Chapelle, B. Schölkopf, and A. Zien, editors. *Semi-Supervised Learning*. MIT Press, Cambridge, MA, 2006.

[9] F. Lauer and G. Bloch. Incorporating prior knowledge in support vector regression. *Machine Learning*, 70:89–118, 2008.

[10] F. Lauer and G. Bloch. Incorporating prior knowledge in support vector machines for classification: A review. *Neurocomputing*, 71(7-9):1578 – 1594, 2008.