Classifying the decision to perform surgery in MEN1 cancer patients using decision trees

D. Höhle[1], C.R.C. Pieterman [2], G.D. Valk [2], A.R. Hermus [3], H.P.F. Koppeschaar[1] , J-J. Ch. Meyer[1] and R.P.J. de Lange[1]

[1]*Alan Turing Institute Almere, Louis Armstrongweg 84, 1311RL, Almere, The Netherlands*

[2] *University Medical Center Utrecht, Endocrinology, Utrecht, The Netherlands*

[3] *University Medical Center St Radboud, Nijmegen, The Netherlands*

D.Hohle@ati-a.nl, B.delange@ati-a.nl

Abstract

We present a user-friendly decision tree generating algorithm which searches for the best tree from a forest of n trees. We have biased our algorithm to create bigger trees, as they provide insight in the dataset's underlying structure. The algorithm was applied to data from 130 patients, who suffer from an hereditary form of cancer: MEN1. The best of multiple trees was picked based on performance that was evaluated by the algorithm, not only based on classification results of the validation test set, but also on the stability of the train- and validation test set accuracies, discriminative power, tree-width and tree-depth. We present a tree with performance 0.912 on a 0-1 scale, that closely resembles the decision to perform surgery in MEN1 patients by the physician. We also show that even good trees can make medically flawed decisions; that is why they must always be evaluated by health care professionals.

1. Introduction

Multiple endocrine neoplasia type 1 (MEN1) is a relatively rare -incidence in the Netherlands is 2 to 3 per 100,000- hereditary form of cancer originating in different endocrine organs [1]. In almost everyone who carries a MEN1 mutation, tumors will eventually develop in parathyroids, enteropancreatic endocrine tissues, and the anterior pituitary. The parathyroid glands synthesize and secrete parathyroid hormone (PTH), one of the principal calcitropic hormones regulating extracellular calcium concentration. Parathyroid tumors, occur in 95% of MEN1 patients, and hyperparathyroidism with high levels of PTH and the resulting hypercalcaemia is the first manifestation of MEN1 in about 90% of patients. Complaints related to hypercalcaemia are e.g. kidney stones and renal failure, bone brittleness leading to fractures, constipation and nausea, and disorders of the central nervous system.

Surgery is the treatment of choice for hyperparathyroidism in MEN1 patients, but induce the risk of complications such as recurrent laryngeal nerve injury and postoperative hypoparathyroidism.. The ability to predict the ideal moment for surgery would facilitate appropriate management of individual patients with this inherited disorder [2-3]. To assess the optimal parameters and parameter values to classify the decision to perform surgery of the parathyroids in MEN1 patients, we have designed and applied a decision tree making algorithm, to a dataset containing measurements of serum PTH and calcium, results of mutation analysis and from ultrasound imaging, and whether or not a patient had previously undergone parathyroid surgery.

2. Decision trees

Decision trees are used to find relations in datasets. They are a result of an algorithm that is used for classification problems. Research on decision trees goes back more than half a century [4]. Quinlan developed decisions trees further [5]. The basic idea behind decision trees is to create a path from some attributes (or input variables) to a requested target variable. In the example that we describe in this paper, a path from a dataset of clinical parameters, the input variables, is drawn to a target variable: the decision whether or not to perform surgery on the patient. Parameters, or attributes, that may contribute to this decision could be blood values, or results of biopsies or imaging.

Breiman described the bagging predictors algorithm (Bootstrap Aggregation) that creates multiple trees [6]. All trees are used as 'predictor': when supplied with a case to be classified, each tree votes for a class label. The label with the highest number of votes wins. The boosting algorithm [7] is similar, but here the votes are weighed.

Although the bagging and boosting algorithms, at least in comparison with the original C4.5 [7], often provide a smaller number of erroneous predictions, they tend to overfit on the dataset and are less stable, i.e. accuracies differ enormously with different data sets. Furthermore, individual tree depths are usually only two items deep, which may neglect the underlying structure of the data set. The power of decision trees is to find relations – structure – in the data that are not obvious at first glance. Since our main interest is to evaluate the structure of the tree, for experts, we will take a further look in optimizing, but not overfitting, the tree building process and performance measures for individual trees.

The generation of decision trees is an abstract mechanism which can be applied to any dataset having multiple attributes with a relation to the class variable, and, for each attribute, multiple (preferably as many as possible) sample data cases.

3. Tree structure

The essence of decision trees is to find relations in a dataset. Relation, here, means that all cases sharing the same value for attribute A and sharing the same value for attribute B (and so on), should classify to the same value for target attribute Y.

The first step in the creation of the decision tree is to calculate the mutual information [8] between each of the input variables and the target variable. The input variable that shares the most information with the target variable, will be used as root node - which is the starting point for the decision tree.

The variable for which the root node is created, has a number of possible values. For example, in the resulting decision tree from our dataset (figure 1), the root node is "iocal_jaar_-1_categorie" (serum ionized calcium levels one year prior to surgery). For each of the possible values a branch is made to another node; the child nodes. In our dataset the branches for the root node are the calcium concentration categories that we had made (see also paragraph 4.2). Each case in the patient dataset can be classified to only one of these branches.

At this point, for each branch and its subset of cases, the mutual information between each of the remaining input variables and the target variable is again calculated. The variable sharing the most information with the target variable becomes the child node for that branch. In our dataset, for example, the 29 patient cases in branch 7) of the root node are best split further using the "PTH_jaar_-3_categorie" parameter mentioned in the node with node id a2 in figure 1 (serum PTH levels three years prior to surgery). The process described above will repeat itself, until a stopping parameter is satisfied.

4. Decision tree design parameters

4.1 Pre- and post-pruning

Stopping parameters are set before building the tree and include, 1) not meeting the minimum necessary amount of information gain (the mutual information given the subsets of the branches), 2) not reaching the required minimum number of cases for a child node with the fewest cases, or 3) exceeding the maximum number of generations - i.e. maximum depth of the tree. These stopping criteria are called pre- pruning methods.

When the decision tree is built entirely, an algorithm will check whether the tree can be pruned. This process is called post pruning. This is done, again, by looking at the mutual information between the current variable (given the filtered dataset) and the target variable. In contrast to pre-pruning, post-pruning starts at the leaf nodes (i.e. the nodes without children). If the mutual information is smaller than the minimum required information gain, then all the child-nodes of this parent are pruned, so that the parent-node becomes a leaf-node itself. This process repeats until nothing can be pruned anymore, with set criteria.

4.2 Categories

Decision trees are not able to handle continuous data. Since each branch is defined as a category and individual cases need to be a member of that category, we need to define appropriate discrete classes.

There are numerous ways to categorize continuous data, one of which is to let the expert decide. This is especially recommended when certain boundaries can be made based on evidence from literature. In our data, for example, we have categorized the PTH and ionized calcium serum levels in a fashion that reflects their sigmoid relationship:

Cat.	Ionized calcium (mmol/l)	PTH (pmol/l)
1	0.01-0.80	0.1-1.0
2	0.81-1.12	1.1-1.6
3	1.13-1.17	1.7-2.4
4	1.18-1.22	2.5-6.0
5	1.23-1.27	6.1-7.0
6	1.28-1.30	7.1-8.0
7	1.31-1.35	
8	1.36-2.00	

Alternatively, our decision tree generator can create categories based on some simple settings. Examples for this are: *X* number of categories, between *X* and *Y* number of categories, flexible boundaries, predefined boundaries, or empty - if already categorized. The term 'boundaries' is used as division value, which implies that the number of categories is the number of boundaries plus one. Where numbers of categories are chosen, each category has, approximately, the same number of samples. For the option 'flexible boundaries' the number of boundaries is fixed, by the number of flexible boundaries. If, for example, the range is between 2.5 and 3.5 then, before the individual tree building process a random value between 2.5 and 3.5 is taken and used for the supplied attribute. Categories are required for each attribute that might be used in the tree building process.

4.3 Bootstrapping

A requirement for building decision trees is that the data set is divided into two subsets, a train set for building the tree and a test- or validation set for testing the tree. In our example we randomly assign 70% of the cases to the train set and the remaining 30% to the validation set. Patient data sets, especially on MEN1, are often the results of many years of laborious data collection. To optimally use the data, we apply a resampling method called bootstrapping. With bootstrapping a case is drawn randomly from the train set and replaced into the pool. Thus, a single patient case can be drawn multiple times, while others may not be drawn at all. In addition to artificially increasing the number of patient data cases, bootstrapping may decrease data noise when, and only when, the same sets of attributes are used in multiple trees in the forest. Noisy data will lead to a decrease in final performance of such trees.

4.4 Windowing attributes

As mentioned before, the purpose of decision trees, is to produce structure from a dataset. However, it is conceivable that more than one structure can be produced from a single dataset, but that other structures may not be found when we include all the attributes for each tree, since each tree will start with the same attribute (ignoring a small deviation which might be found due to bootstrapping).

To overcome this limitation, a parameter called windowing size is introduced. The windowing size is the maximum number of attributes a tree may use for the building process, preferably a lot smaller than the total number of attributes. Therefore, attributes that, given the whole attribute set, will always be used as root node, will (by chance) be ignored. This will result in examining different tree structures, which might include trees that describe the dataset better than the ones without the windowing parameter.

4.5 Multiple trees

The windowing size parameter may lead to trees with attributes that have no relation at all with one another. It would be time-consuming to examine each possible tree. Therefore, a property is created to generate X number of trees and only return the best one.

But how do we decide which one is the best one? A straightforward measure would be to test the accuracy of the test- or validation set, i.e. percentage of the test cases that are classified correctly. But, if thousands of trees are generated, and only the best one is displayed, is this not over fitting? It probably is. Imagine some random train-set with one root node and two child nodes. If the value of the variable in the root node is 'yes', take the left branch, if 'no' take the right one. When leaf L1 (the left one) is reached, the outcome is 'operate', leaf L2 has outcome 'do not operate'. Now image a test-set that, by accident, exactly fits this tree. Then this tree will be remembered and printed as being a valid and indeed very good tree. Although the tree itself is valid, it will not generalize to the whole population, which is of course exactly what we would like.

4.6 Tree performance

A more appropriate measure of the performance of decision trees was created by Osei-Bryson [9]. Instead of using just a single measure, Osei-Bryson proposed a set of five different measures: the accuracy of the validation set (ACCv in the performance box in figure 1), the stability of the accuracies (STABc) which is calculated from ACCt and ACCv, the discriminatory power of the leafs (DSCPWR), the simplicity of the number of leafs (SIMPLleafs) and the simplicity of the average depth of the tree (SIMPLrule). The combined measure is called the final performance (final perf), this is the measure that is used to evaluate the performance of the trees. The contribution of the five individual measures to the final performance can be weighted to appropriate preferences. The tree from the example above will not be evaluated as a very good one for at least two reasons: firstly, if the train-set does not correspond to the validation set, the corresponding accuracies will deviate resulting in a lower STABc, and secondly, the SIMPLleaf score will be 0 since the minimum number of leafs cannot be lower than e.g. 3

(a parameter set by the user). For more information on performance measures see [9].

4.7 Biased root node

It is conceivable that some attributes may be more interesting to researchers than others. In hypothesis driven investigations, some input variables may, *a priori*, seem more related to the target variable than others. Or, for practical purposes, it may be cheaper to take a patients temperature than to perform a biopsy. Therefore, an option was built, for biasing the root node, so that only a subset of the windowed attributes can be used as starting point. This is also another way to make sure other data-structures are found

5. The MEN1 dataset

One hundred and thirty cases, from the hyperparathyroidism subdatabase of the larger Dutch MEN1 database were included in the dataset. All patients were treated at either Utrecht or Nijmegen University Medical Centres or seen at both sites. The hyperparathyroidism subdatabase contains data on laboratory values (serum ionized calcium, total protein bound calcium, albumin, PTH, vitamin D), imaging of tumors (ultrasound imaging, computed tomography imaging, MRI imaging, parathyroid scanning), surgery, medication, MEN1 mutation, MEN1 mutation category, family, date of birth, the hospital at which the patient was seen and pathology. For this paper, we have only considered database entries from 2006 onwards, since the decision to perform surgery was re-evaluated at that time. To be able to compare patients who had had surgery to patients who have not had surgery until their last visit and database entry, we transformed the data as follows. For patients who underwent surgery between the first quarter of 2006 and the first quarter of 2011, the quarter (date) of surgery was set as time = 0 and averages for calcium and PTH were calculated for each of the three years prior to their operation. For patients who had not had surgery by the first quarter of 2011 (or the last time they were seen between 2006 and 2011), their last visit was set as time t = 0 and averages for calcium and PTH were calculated for each of the three years prior to this last visit. Averaging values over a year's period also helped to reduce the number of missing values. Missing values were still present in the dataset after transformation and were always represented as "99999".

Ionized calcium and PTH serum levels were categorized based on their sigmoid relationship in healthy individuals. There were too many categories for the different MEN1 mutations and for the different families to be used in the decision tree algorithm (more than fifty). Only ultrasound imaging data were readily available for the majority of the cases, so it was decided to ignore the other imaging data.

6. Results

The decision tree shown in figure 1 was the best of 1000 trees to represent the MEN1 dataset with the following decision tree building parameters set: maximum number of attributes used to build the tree = 6; percentage from dataset used as train set = 70; use bootstrap, with replacement, bootstrap percentage = 300; minimum number of values for node selection = 9; maximum tree depth = 6; performance weights (add up to 1): weight_ACCv = 0.25, weight_STABc = 0.25, weight_DSCPWR = 0.25, weight_SIMPLrule = 0.125, weight_SIMPLleaf = 0.125.

The tree was visualized from left to right for readability. On the left, below the root node there are three boxes with information. The first box shows which variables or attributes were assigned by the user to be allowed to be used in the decision tree. We have, e.g. chosen not to use the hospital at which the patient was seen or year of birth in this particular run. The third box shows which of the attributes was the target attribute ("chirurgiena2006" or whether or not a patient had undergone surgery since 2006); how many test and train samples were used in the tree building process, the pre- and post-pruning parameter settings, and the maximum number of attributes and tree depth.

The second box contains the important information on the tree quality: accuracy of representing the train set (ACCt = 0.912), accuracy of representing the validation set (ACCv = 0.949), the stability of the accuracies (STABc = 0.961), the discriminatory power of the leafs (DSCPWR = 0.911), the simplicity of the average depth of the tree (SIMPLrule = 1.0), the simplicity of the number of leafs (SIMPLleafs = 0.652) and the final performance (final perf = 0.912).

The root node is " iocal_jaar_-1_categorie": serum ionized calcium level one year prior to operation. This means that this variable has the highest mutual information with the target variable. The 273 train cases are split into eight branches from the root node. For example, at the top of figure 1 it is shown that 5 cases have category 3 for the root node, serum ionized calcium level of 1.13-1.17 mmol/l one year prior to operation. For categories 2, 3, 4, 5, 6 and 9999 no further information was required: all cases are classified to outcome "no surgery".

Outcome "surgery" may follow from the root node branch with category 7, that then splits by PTH levels three years prior to surgery, or from the root node

branch with category 8, that then splits by the next most informative variable was "iocal_jaar_-3_categorie" (serum ionized calcium levels three years prior to surgery). For the latter variable, for example, category 4 leads to outcome "no surgery", category 5, 6 and 7 lead to outcome "surgery", whereas category 8 is further split by PTH levels one year prior to surgery.

Each of the 273 train cases may be similarly traced from root node to outcome in the same way. At each node the information gain is plotted. The ACCt performance is reflected by the frequency % at each of the nodes: the tree classifies 91% of the 273 train cases correctly to surgery or no surgery (and 94.9% of the 39 test cases, hence the 0.961 score on stability of the accuracies).

The decision tree that is made by the algorithm must always be interpreted by the user. A tree may fit a dataset perfectly, yet make little sense from a medical point of view. We have deliberately left out the variable "ultrasound imaging after 2006" to draw the tree presented in figure 1. When we left this variable in, using the same parameters, we obtained a tree with an even higher final performance. In this tree, the branch from the root node with category 8 was best further split by the ultrasound variable. Yet when we looked at the branching at the ultrasound variable (figure 2), we were surprised to find that performing ultrasound no matter if abnormalities were found *or not* (categories 2 and *1*, respectively) would always lead to outcome "surgery", whereas not performing ultrasound would always lead to outcome "no surgery". Although this might properly fit these data, this is of course no sound basis for the medical decision to perform surgery.

7. Discussion

We have designed and applied a decision tree making algorithm to a real-life data set of one hundred and thirty cases of MEN1 cancer patients. Physicians base their decision to perform parathyroid surgery in MEN1 patients on ionized serum calcium levels, serum PTH levels, results from imaging techniques and results from exploratory or prior surgery. We have shown here that our algorithm can draw a decision tree that reflects their decision making process very well. Patients that do not classify correctly may be studied further to re-evaluate if the decision to perform surgery on them (or not), was justified.

The user of the algorithm can alter multiple settings that affect the final decision tree: he may choose to set category boundaries, alter pre- and post-pruning settings, apply bootstrapping, alter the weights of the performance measures, and bias the root node decision. For our dataset and with the settings we have described above, the algorithm is ready in seconds, fast enough to allow testing with several combinations of settings and including or excluding variables from the dataset. Though we should stress that one should not infer causal relationships from the tree-path to the target variable, the decision trees may lend meaningful insights to the user.

We are confident that our decision tree algorithm may be applied equally usefully to other datasets and welcome collaborations in the health care setting.

8. References

[1] Bassett, J. H., Forbes, S. A., Pannett, A. A., Lloyd, S. E., Christie, P. T., Wooding, C., Harding, B., et al., "Characterization of mutations in patients with multiple endocrine neoplasia type 1", *American Journal of Human Genetics*, 1998, 62(2), 232-244

[2] Pieterman, C.R.C., Schreinemakers, J.M.J., Koppeschaar, H.P.F., Vriens, M.R., Rinkes, I.H.M.B., Zonnenberg, B.A., Luijt, R.B. van der, and Valk, G.D., "Multiple endocrine neoplasia type 1 (MEN1): its manifestations and effect of genetic screening on clinical outcome", *Clinical Endocrinology*, 2009, 70(4), pp. 575-581.

[3] Pieterman, C. R. C., Vriens, M. R., Dreijerink, K. M. A., Luijt, R. B. van der, and Valk, G. D., "Care for patients with multiple endocrine neoplasia type 1: the current evidence base.", *Familial Cancer*, 2011, 10(1), 157-171.

[4] Feigenbaum, E.A., and Simon, H.A. (1963). "Performance of a Reading Task by an Elementary perceiving and Memorizing program", *Behavioral Science*, 1963, p 8.

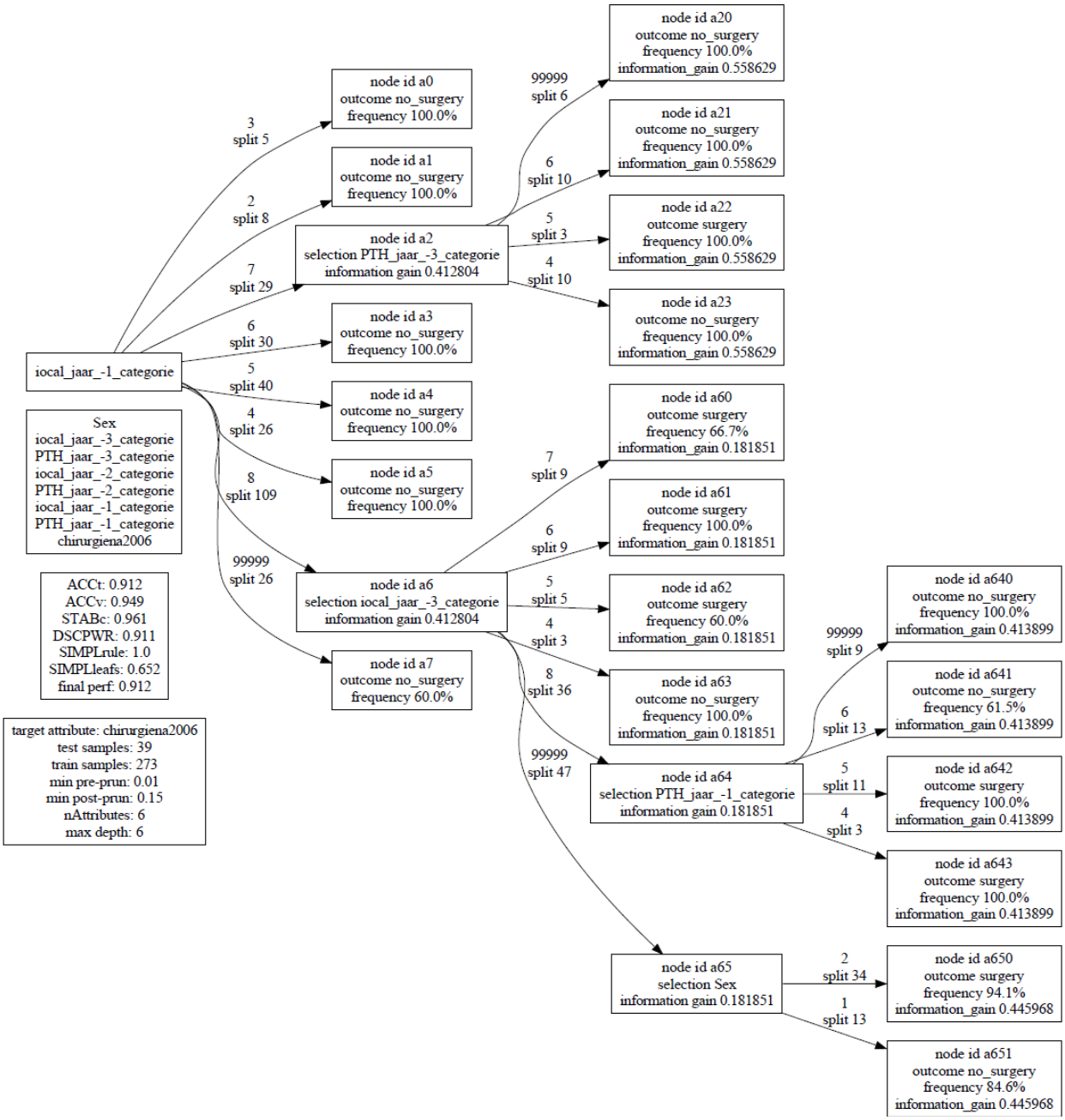
[5] Quinlan, J.R., "Induction of decision trees", *Machine Learning*, 1986, 1, pp. 81-106.

[6] Breiman, L., "Bagging Predictors", *Machine Learning*, 1996, 24, 123-140.

[7] Quinlan, J.R., "Bagging, Boosting, and C4.5", *In: 'Proceedings of the Thirteenth National Conference on Artificial Intelligence'*, AAAI Press and the MIT Press, (1996). pp. 725-730.

[8] Shannon, C.E., "A Mathematical Theory of Communication", *The Bell System Technical Journal*, 1948, . 27, pp. 379-423 and 623-656.

[9] Bryson-Osei, K., "Evaluation of Decision Trees: A Multi Criteria Approach.", *Computers and Operational Research*, 2004, 31, pp. 1933-1945.



Figures 1(above) and 2 (below) . See text for further explanation