
Papers

Merits of interactive decision tree building: Part 1

Received (in revised form): 21st May, 2007

Bas van den Berg

is Principal Consultant at the marketing intelligence department of VODW Marketing (www.vodw.com). His core business is helping companies make their marketing activities more efficient and effective based on facts. His fields of interest span: predictive modelling, lifetime value management and retention.

Tom Breur

runs consulting firm XLNT Consulting (www.xlntconsulting.com) dedicated to helping companies make more money with their data. His fields of interest span: data mining, analytics, data quality, IT governance, data warehousing and business models.

Keywords CRM, analytical CRM, data mining, decision tree, targeting, direct marketing

Abstract There is a growing tendency to embed the entire data mining process within a single software solution. These data mining suites have data capture, pre-processing, model building and deployment all integrated. In two articles the authors will discuss the merits of interactive model building. In this first paper the 'why' question will be answered; in the second paper, the authors will demonstrate 'how' to do this. This 'interactivity' basically consists of overriding statistical parameters as they are derived from training data. The authors propose interactive model building as an alternative to automatic model building. First, interactive model building generates more knowledge on customer behaviour and on the structure of the data. Secondly, deliberately influencing the way models are being generated actually leads to better predictions of customer behaviour. The authors illustrate how the context of the business problem can and should be taken into account when developing models.

Journal of Targeting, Measurement and Analysis for Marketing (2007) **15**, 137–145. doi:10.1057/palgrave.jt.5750045

INTRODUCTION

Both within the CRM product space as well as outside it, integrated data mining solutions are dominating the marketplace. Successful execution of CRM strategy depends largely on *delivering* analytic results. For data mining to become effective, ease of model deployment is one of the decisive factors. Because of this, the need for integrated solutions goes even beyond the borders of the data mining process: it extends into the areas of campaign management, personalisation and marketing automation. But this seamless integration of data mining technology in CRM

systems has evoked the illusion that machines can take over the task of (predictive) modelling. Vendors sometimes suggest a future where intelligent machines will automatically learn how to respond to changing customer needs. Expectations of these adaptive Customer Relation Optimisation solutions have been set, and the need for human intervention to develop business intelligence and to manage this knowledge is often severely underestimated.

The authors do not oppose integrated nor automated solutions *per se*. In fact, embeddability is a key requirement to make data mining solutions pay off. Neither do we oppose automatic data mining solutions *per se*. The needed accompanying organisation and knowledge structure surrounding it is, however,

Correspondence: Tom Breur, Principal, XLNT Consulting, Langestraat 8-03, SE Tilburg 5038, The Netherlands.
Tel: +31 6 463 468 75;
E-mail: tombreur@xlntconsulting.com

rarely in place to begin with. Furthermore, it may easily be omitted in the short run, with poor results for the long haul. Such short-sightedness poses a considerable risk as the use of data mining technology is not quite fully mature and established, yet.

The hope for these grandiose solutions has been fuelled, not in the least part, by vendors of end-to-end CRM solutions. In one case, a vendor's claim suggested that implementation of their tool had caused response percentages to rise from 1 to 30 per cent. And this was at a company with ten years experience in direct marketing, notable for its sophisticated use of data mining models! Clearly such claims should foremost cast doubt on the expertise of vendors making such unwarranted claims. But management is easily impressed by these claims, and expectations about the possibilities of data mining do need to be managed. Is it really true that machines can replace humans in this process? What are the advantages and disadvantages of automating the model building process? Is the current state of affairs advanced enough for this, yet? Which requirements need to be met to make this possible?

The structure in which these thoughts are presented in this paper will now be outlined. After discussing the advantages and disadvantages of integrated solutions, we will have a look at automated model building. Then we discuss interactive tree building, as an example of interactive model building, and explain why this leads to more knowledge and better models. Of course 'better' needs to be understood in relation to the application and the business context. In the next paper, the authors go on to explain how interactive tree building is done in practice, with detailed practical guidelines. In the conclusion, we explain under what conditions one can typically expect our proposed approach to be most useful.

INTEGRATED AND AUTOMATED DATA MINING

Integrated data mining tools

Clearly, there are great advantages when analytic data mining solutions can be seamlessly embedded into an operational environment. On the input

side of the data mining process, mapping of input variables tends to be a cumbersome and error prone process. Often this pre-processing requires extensive manual programming, making this an unreliable and brittle part of the architecture. On the output side of the process, error-free deployment is a challenge, for similar reasons. These advantages have greatly and legitimately contributed to the popularity of integrated CRM solutions.

But there are disadvantages, too. When input data are fed into an integrated, automatic model building environment, one may be tempted to 'forget' that scores on a variable are but an abstraction of reality — this link is easily forgotten. Profound insight into the way actual customer behaviour translates into a mapping on an input variable can greatly enhance insight into potential leverage points for influencing customer behaviour.¹ This insight itself is not lost through integration, but for a lack of manual data capture and pre-processing, the quest for profound insight into customer behaviour must be scheduled separately. The workflow process in an integrated data mining solution does not call for this anymore.

Clearly there are big advantages to be gained from integrated data mining workbench tools, too. Process speed is one thing, but certainly a lesser dependence on human intervention is a valuable gain as well. People just are not as consistent and reliable as computers.

Automatic model building

The essence of automatic model building lies exactly in 'automatic'. Automating makes the process less dependent on humans, and can speed certain phases up. When input parameters are set, then the output, the model, can be generated and validated automatically. One push of the enter key and the model is practically ready. No intervention of the miner is called for. With automatic model building, the data miner's contribution comes in the setting of parameters beforehand. This is an advantage of automatic model building, apart from the fact that it can speed up the process.

But the lack of intervention by the miner is also a serious drawback of automatic model building. There is little (or no) opportunity for

the miner to input domain knowledge, and no knowledge is gained from the model building process. The lack of opportunity to gain knowledge is a serious loss because it is an invaluable spin-off from model building that can be used for improvements in the way a company currently does business. Also, such insights could potentially inspire radically innovative future developments.

Apart from the lost opportunity for customer and (meta) data knowledge, there is another important disadvantage of automatic model building: lesser quality results. There are three reasons for this. First, automatic model building comes at a price: in exchange for immediate, run-of-the-mill models one does trade off some predictive accuracy. Careful manual tweaking of the model parameters can improve the prediction quality, sometimes considerably. The second reason for lesser quality results is that there is no built in human check for potential errors in the mining set. This may happen when 'leakers'² or 'anachronistic variables'³ have accidentally entered the data set. Thirdly, automatic model building is not geared towards the specific situation in which the model is to be used. Considerations like the percentage of the population that will be targeted can help in designing the optimal model for a given business problem. Should the model perform optimally for the best 10 per cent, or should the best 30 per cent be optimised? Automatic model building cannot realistically take more than one of these business constraints into account. This is not so much a matter of translating an optimisation criterion into the appropriate algorithm. This, essentially, refers to the business modelling part of the analytical CRM endeavour, which will fundamentally remain a human undertaking for (at least) the foreseeable future.

These disadvantages of automatic model building lead us to generally prefer interactive model building. The interactive model development process takes care of the disadvantages just described. The authors will attempt to demonstrate that 'the human factor' in the analytic process still holds its own. We find that the human element adds indispensable value in the wider context of making the best

possible use of business intelligence in general, and data mining in specific. This holds in particular for business settings where optimal use of customer data is one of the means of capturing a sustainable source of competitive advantage.

WHAT IS INTERACTIVE MODEL BUILDING?

Interactive model building

Interactive model building is characterised by the miner *actively* engaging in the model building process. This 'active' influencing of model development can be done in several ways, and for various reasons. These reasons may have to do with the miner deliberately inputting domain knowledge on either the data (❶ in Figure 1), or on previous experience with models in the same domain (❷ or ❸ in Figure 1). There is still no substitute for experience, and no data mining algorithm is likely to ever replace domain knowledge. External knowledge is especially important when the volume of data is limited, and statistical evidence for predicting target behaviour is scanty.

The way in which interactive model building is done, depends in part on the technique and software being used. The case for decision trees will be elaborated on later in the second paper. For regression models, for example, one might want to constrain certain parameters. Or a variable might have been removed in a stepwise procedure, yet subsequently re-entered at the cost of another variable that shows higher correlation with the target variable (at least in the training data). A variable that is 'known' to be monotonically increasing, can be fixed to 'correct' anomalies in the data that are assumed to be caused by sampling fluctuations (❹ in Figure 1). For example, a decrease for a certain range of the variable may be corrected by setting the variable to the local maximum. In such cases, the analyst overrides statistical evidence in the data on the basis of experience, or some other insight (knowledge with the data). This justification comes more readily when statistical evidence from the data is not very strong, typically when using

relatively small samples of data. Such decisions should be based on previous experience with substantive phenomena (5 in Figure 1). Such approaches to model building are sometimes referred to as model engineering.

Interactive tree building

Decision tree tools

There are many decision tree tools available in the marketplace, commercial and noncommercial. All of these support automatic tree building. In fact, most tools are mainly geared towards automatic generating of models. The degree in which the user interface supports interactive tree building differs widely.

Not all tools allow the same degree of interactivity. We define level of interactivity here as the influence the miner can potentially exert on setting the values of the eventual model. Pyle³: ‘So, in explaining data, it is very important that the tree tool does not steer the miner, but allows the miner to steer the tool’.

Segment versus continuous predictions

Some tree algorithms can model continuous target variables, but most work exclusively with a

discrete target variable. For the majority of algorithms this is limited to dichotomous classification, very few can deal with a target variable that has multiple categories. The authors have found that trees that simultaneously predict multiple output categories tend to be less transparent and more difficult to interpret.

When used as a predictive modelling tool, decision trees generate an ordered set of segments. In this respect, the output from a decision tree resembles the result from for instance rule-based algorithms, decision lists or association rules.⁴ Many other algorithms generate a continuous prediction. Examples include: Neural Networks, Regression, Nearest Neighbour, Support Vector Machines, etc.

Because trees generate ordered segments as a prediction, this results in stepwise increments in response probabilities. The prediction value is constant within any given segment, or leaf of the tree. Consequently, the cumulative response curves are not smooth lines.

Transparent versus opaque techniques

Data mining algorithms can be ordered along a continuum that ranges from transparent to opaque. Oftentimes Neural Networks are used as

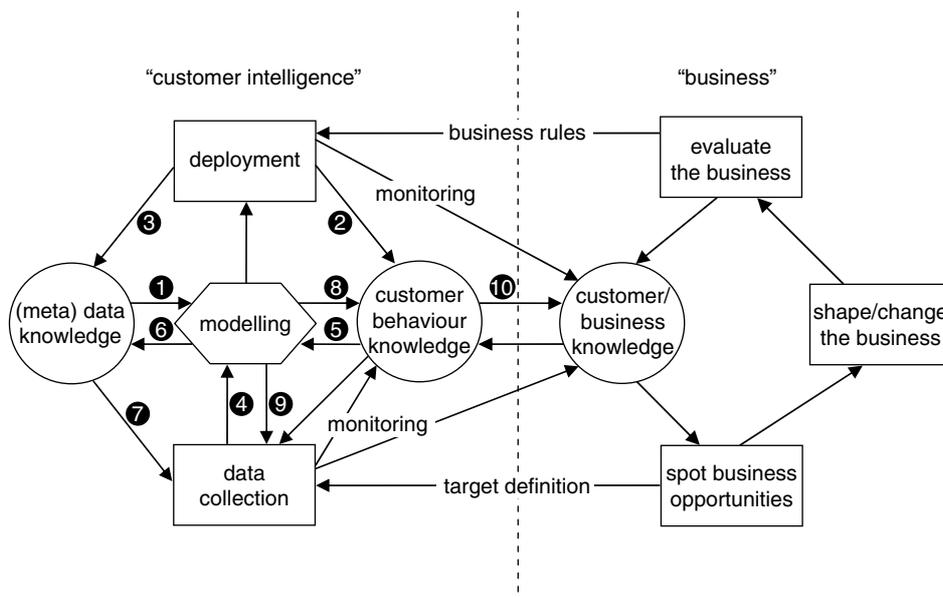


Figure 1: Knowledge loops from interactive modelling.

the classic example of an opaque technique: the algebraic classification rules are basically incomprehensible to human beings. Within this spectrum, decision trees can be considered as one of the most transparent techniques available. The big advantage of transparent data mining techniques is that these show the miner most about the data and more readily offer insight (⑥ in Figure 1). Decision tree tools have proven intuitive to both analysts, but also to business owners.

The accuracy of a tree model may not be the highest possible, and in fact it often is not. Tree models are, however, easy and fast to create, even when dealing with very large numbers of variables. They require limited effort in terms of pre-processing. And in the process, they generate a lot of insight in the data and customer behaviour. In the long run, this knowledge spin-off will help to derive and generate new and better variables that pave the way for more profound insight, and higher long-term potential for creating predictive accuracy (⑦ in Figure 1). For modelling environments with continuity, this long-term goal will reap the most rewards. Getting to the best possible models hinges on amassing a large body of strong potential predictor variables, specific for each business domain. Using the practical guidelines we will put forward later in our second paper, trees have proven remarkably resistant to the curse of dimensionality (having a large number of variables relative to the number of rows).

More knowledge is gained from interactive tree building

The largest drawback of automatic tree building is that there is less opportunity to gather knowledge and insight from the model building process. There is little opportunity to *input* domain knowledge (⑤ in Figure 1), and less knowledge is *gained from* the model building process (⑧ in Figure 1).

The data mining process is cyclical in nature. By making the model building process as interactive as it can be, this will support several development cycles in Figure 1 of which some examples are: developing more consistent data

(⑦ in Figure 1), enhancing the data set by deriving new variables (⑨ in Figure 1), checking the data set for possible problems like sampling aberrations (① in Figure 1), etc. This description of the interactive model building process serves to illustrate why *knowledge* is the central driving factor. Knowledge is both input for the process as well as essential output for both short-term and long-term improvement cycles of the data. Note how *both* the process on the left ('customer intelligence'), as well as the right ('business') are characterised by circumventing Plan-Do-Check-Act cycles.

Learning about customer behaviour

Whether models are specifically developed to provide insight, or 'merely' to predict future response, *all* models require an explanation.³ Business stakeholders will always benefit from describing the relationships in the data in the most comprehensible way. Therefore, the more parsimonious a tree, the better it is in this respect. Sometimes for reasons of explanation, it may be beneficial to deliberately *include* a particular variable in the tree model.

Knowledge gleaned from the mining process may very well be useful outside the realm of predictive modelling for targeting. New business opportunities can be discovered during the modelling process, which may then trigger a change in the way business is done (⑩ in Figure 1).

For example, one of the authors was involved in a mortgage modelling project, when it was discovered that a distinct segment of customer would *only* close their second mortgage through a direct phone channel. This was noteworthy because these customers were typically affluent, did not hold their primary mortgage with the bank and apart from this segment few other customers closed their mortgages over the phone. When this insight was fed back to the marketers, initially nobody could come up with a compelling explanation. Upon further (internal) research, the team found that this call centre team (with the highest experience and skill levels) were allowed to negotiate interest rates with customers. It turned out that these price savvy affluent customers typically held their primary account

elsewhere, and went ‘shopping around’ for the best deal on their second mortgage. In all likelihood, given their wealth, they were among the few consumers who were experienced enough to dare to close their mortgage entirely over the phone. This insight triggered an entirely new marketing campaign targeted at this segment.

Learning to improve the data

A second way in which knowledge from the model building process can be put to use, is to improve the data. This might be improving data quality (eg removing errors), or enhancing data by adding derived features (⑨ in Figure 1).

Examples:

- Some variables may lead to more stable tree segments than others. This will become evident by comparing the distribution of train- and test set. Another way to test model stability is by comparing the mining set with variable distributions at run time (when the model gets deployed). When these distributions do not match, it is possible that the mining set is not fully representative of the population, or that variables in the model are unstable. These two sources of instability may be impossible to distinguish in most practical cases. Either the mining sample may be too small to be fully representative, or the relations between variables may be subject to change over time. When relations between variables are monitored over time, and are found to be unstable (③ in Figure 1), it is questionable whether to use them when developing models. We label these effects as being ‘not robust’ (more on this in our second paper).
- When developing a tree model, it can become apparent that not all missing values on a variable are ‘equal’, even though this group of records technically all have the same value ‘missing’. Interrelations with other variables in the data set may show that there are ‘subgroups’ to be found within records that are missing. By delving further into this, it may be possible to codify the different types of missing into a new, purpose developed variable (⑦ in Figure 1). This new variable

can then be added as an extra explanatory variable. This practice goes beyond Pyle’s² recommendation of adding an indicator before replacing missing values with an appropriately distributed value (*never* the mean or mode!). In general, most tree algorithms can deal with missing values well, and oftentimes support missing values as a separate class of their own. As an example: a variable can be missing because it is impossible to calculate this field’s entry if the customer’s tenure is too short. But the same field could also be missing due to some other source of error. It is generally useful to distinguish between different types of missing fields within a variable.

- By their nature, tree tools are insensitive to interactions between input variables. At each node the data are split, one variable at a time. The interactions between variables in the input battery play no role when deciding on each individual split. If interactions between input variables do play a role, interaction variables need to be added to the mining set beforehand (⑦ in Figure 1).

How interactive tree building leads to better models

We will discuss two reasons why interactive tree building can lead to better models. The first reason has to do with lift. Automatically built trees may contain undesirable splits.⁵ Rebuilding the entire model, as would be customary in an automatic tree building procedure, can only purge these splits. But the new model may then contain other undesirable splits. When building trees automatically, it is practically impossible to get rid of such undesirable splits altogether. The second reason for lesser quality models is that automatic tree building is not geared towards the specific situation (implementation specification) in which the model is to be used. For example, one of the considerations to take into account might be reasonable segment sizes (around the cut-off point) that will allow for monitoring of model performance with sufficient statistical power.

Automatic tree building does not take these considerations into account. Only the loss

function, particular to the algorithm, is minimised *at each split*. Decision trees, by the very nature of the algorithm, will seek to choose the best split for any subset of the data that arrives at the given node. Trees have no ‘memory’, and they cannot ‘look forward’ to see how the subsequent splits will work out. It is not the overall tree that is being optimised, just a particular split. And sometimes this leads to sub-optimal models.

Better lift from interactive tree building

Idiosyncrasies of train/test fluctuations. As we have discussed, an interactive tree building procedure can help to deal with problems that arise from not having a sufficiently large data sample. How large is large enough is a difficult statistical question. In most practical cases, however, the miner finds himself with a data set that he is simply asked to make the best of. This means handling the problematic signal to noise ratio by employing a procedure that is as robust as possible. The challenge typically is to not over-specify the model. For tree building this implies trying to limit the number of leaves from the tree, as each split will use up a scarce resource: degrees of freedom in the training set.

The practical way to come up with a tree that works well in both the training as well as the test data set, is to run over as many candidate splits as possible. Also, the eventual outcome of a split needs to be reconsidered after one or more additional layers in the tree are generated. So the miner should expect to find himself trying promising splits, but subsequently pruning them again if the result turns out to be not so fruitful after another layer of the tree is built. More on this topic in our next paper on practical guidelines for interactive tree building.

In data mining, if an estimate of the predictive accuracy of the resulting model is wanted, one (nearly always⁶) needs to split the mining set three ways: a training-, a test- and an evaluation-set. The evaluation-set is locked away at the start of the project, and never again looked at, until the final model is delivered. This process is the *only* way to guarantee an unbiased estimate of the

prediction accuracy. The fundamental issue here is that possible over-specification threatens to produce a model that does well (too good, actually) on the train data, but does not perform well on the evaluation data.⁷

Choosing enduring relationships in data to prolong model use. A second reason why interactive tree building tends to lead to models with better lift characteristics is because knowledge from previous deployment (and evaluations) can be used when choosing among candidate splits. As the miner gains experience, he may develop a ‘feel’ for which (combinations of) variables hold up their predictive power over time, and which do not.

Often time the consideration to choose among splitting variables involves a trade-off. Is a tree with maximal lift preferred, but with the potential risk of a performance that might decay rapidly? Or, stated otherwise, the decision is whether the miner should opt for splitting variables that do not quite give the very highest predictive accuracy in the short run, but are more likely to remain accurate predictors in the long run? The choice between these alternatives revolves around the cost of building and implementing a model, and the complexity cost for updating models.

The authors have found that — in general — variables that are derived from transient behavioural activity tend to be both strong(er) predictors as well as less ‘stable’ variables. Therefore, depending on both the required lift and wanted halftime for reusability of the model, some compromise needs to be sought.

Better problem-model fit from interactive tree building

In this section, we will have a look at some business considerations that can have an impact on how model building will proceed. Taking knowledge about the business context into account while developing a model is where interactive model building really holds its own.

Maximum lift at cut-off point. The anticipated targeting depth can play a role when developing a model. This holds in particular for interactive tree

methods. For example, if only the best 10 per cent of the population are to be targeted, there is little point in refining splits with very low response probabilities. These customers will never be contacted. When only a small proportion of the population will be targeted, it is acceptable to begin by identifying some small leaves with very high response probabilities. The downside might then be that a relatively large part of the mining set is left over later, with very few candidate splitting variables. Clearly this would be a problem when lift should be optimised at the 50 per cent mark. More on this in the section on 'to deal with "clusters" in the data' in the second part of this paper. Knowledge on the intended targeting depth can, and should play a role when building trees interactively.

Anticipate model monitoring. One of the important considerations to take into account when incorporating the business context in the model development, is how monitoring of model accuracy is going to take place. To monitor the stability of tree segments, one needs to re-establish random response rates every time the model is deployed. In order to test the response percentages in all segments, the random selection needs to be of sufficient size. How large is large enough here depends on two things: the random response rate, and the size of the tree segments.⁸ A tree with more segments leads to smaller leaves, and therefore a larger required random sample.

The business dilemma here is that the random sample has a lower response rate than, say, the best 10 per cent that are targeted. Therefore, this random group represents some opportunity cost.⁹ On the one hand, one wants to be assured that the model keeps performing as expected. On the other hand, one would like to limit the opportunity cost for this random group. These are two contradictory goals. The size of tree segments can be adjusted to meet the needs for monitoring. When the final leaves of the tree are very close in size to the minimal size required for monitoring, one can manually 'adjust' their size by adjusting ranges of a variable that appear from a split. If merging of adjacent leaves is not

feasible, however, we employ a different procedure.

When the end leaves of a fully grown tree turn out to be smaller than the required size, we use a procedure that allows nonadjacent leaves to be merged. We append a field to the original mining set that indicates membership class of a certain end leaf. Next we build a tree with the original target variable, and the newly created 'leaf membership' variable as the *only* input variable. The end result is segments that may be composed of either one or several of the original end leaves. These newly created segments then carry the restriction of minimum size, as required to attain sufficient statistical power for monitoring. From a programmer's perspective, the final model takes the form of sets of '... and ... and ...' rules interspersed with 'or' clauses that indicate which leaves have been merged together. This procedure has the added advantage that merging of leaves is done on the basis of statistical criteria from which the tree itself is built.

CONCLUSION

In this paper (the first of a series of two), the authors have outlined *why* interactive model building has compelling merits, in particular for companies that regard model building as part of ongoing exploitation of their data. In the second paper, the authors will provide practical guidelines on *how* to do interactive model building, as demonstrated using decision trees.

The tendency in the CRM industry is to integrate data mining tools in comprehensive, end-to-end suites. The authors acknowledge certain advantages this presents, but stress the fact that automation risks a loss of knowledge as gleaned from the modelling process. Another drawback from automated model building is that anomalies in the data can easily go unnoticed, and may lead to awkward results.

Interactive model building is characterised by the analyst willingly overriding statistical parameters as they show up in the modelling process. This is sometimes referred to as model engineering. Based on modelling experience and familiarity with substantive behavioural

phenomena, the analyst sometimes chooses to influence model parameters.

The authors conclude that interacting with the data through the modelling process allows for valuable insight in the data and the events one is trying to model. As a secondary benefit, this can inspire the purpose developed data that can improve predictive accuracy significantly. Such sustainable long-term gains are enormously valuable in environments where modellers can work to improve their set of mining variables.

Yet another advantage of interactive model building is that one tends to arrive at models with better lift. This happens both because sampling fluctuations in the data are handled better, but also because one can aim to develop models with longer halftimes (eg that can be reused over extended periods of time).

Interactive model building ensures a better fit between the chosen model and the business problem. Typically one can choose from a large range of alternative models with comparable short-term performance. The choice between such a wide range of models can hardly be automated, as this is a multidimensional optimisation problem, with criteria that are nearly impossible to quantify. One of these considerations is to facilitate eventual model monitoring when the model gets reused, which in itself is based on sophisticated statistics.

These advantages, taken together, generally outweigh the advantages of an automatic model

building process. This holds in particular for modellers who can exert influence on the variables available to them for predicting. Feeding such 'learning loops' within the organisation, in our experience, holds the most promise in the medium to long run when attempting to arrive at the best possible predictive models.

References and Notes

- 1 Berry, M. and Linoff, G. (2000) 'Mastering Data Mining', Wiley, New York, NY.
- 2 Pyle, D. (1999) 'Preparing Data for Data Mining', Morgan Kaufman, San Francisco, CA.
- 3 Pyle, D. (2003) 'Business Modelling and Data Mining', Morgan Kaufman, San Francisco, CA.
- 4 Witten, I. H. and Frank, E. (2000) 'Data Mining', Morgan Kaufman, San Francisco, CA.
- 5 How these undesirable splits can be recognised will be discussed in our next paper on practical guidelines for interactive tree building.
- 6 The only exception to this rule comes from certain, as yet, new Information Theoretic approaches to data mining. With these revolutionary new Information Theoretic approaches *both* signal and noise are measured from the same data set (partition). With these approaches only a two-way partition is necessary. The requirement of a three-way split of the data, however, certainly *does* hold for Information Theoretic decision trees (eg ID3, C4.5, C5.0, etc).
- 7 Exactly because of this repeated alternating between training- and test-data, only the evaluation data set can provide an unbiased estimate of the lift of the model. The lift, as measured from the test-data, would provide an overly optimistic estimate of model performance.
- 8 In principal, there is also a third consideration, the accuracy of the conclusion, the desired statistical significance.
- 9 Breur, T. (2007) 'How to evaluate campaign response — The relative contribution of data mining models and marketing execution', *Journal of Targeting, Measurement and Analysis for Marketing*, Vol.15, No. 2, pp. 103–112.

Copyright of *Journal of Targeting, Measurement & Analysis for Marketing* is the property of Palgrave Macmillan Ltd. and its content may not be copied or emailed to multiple sites or posted to a listserv without the copyright holder's express written permission. However, users may print, download, or email articles for individual use.