Journal of Classification 34:473-493 (2017)
DOI: 10.1007/s00357-017-9239-5

Analysis of Web Visit Histories, Part II: Predicting Navigation by Nested STUMP Regression Trees

Roberta Siciliano

University of Naples Federico II, Italy

Antonio D'Ambrosio

University of Naples Federico II, Italy

Massimo Aria

University of Naples Federico II, Italy

Sonia Amodio

Leiden University Medical Center, The Netherlands

Abstract: This paper constitutes part II of the contribution to the analysis of web visit histories through a new methodological framework for web usage-structure mining considering association rules theory. The aim is to explore through a tree structure the sequence of direct rules (i.e. paths) that characterize a web navigator who keeps standing longer on a web page with respect to the path characterizing navigators who leave the web earlier. A novel tree-based structure is introduced to take into account that the learning sample changes click by click leaving out navigators who drop off from the web after any click. The response variable at each time point is the remaining number of clicks before leaving the web. The split is induced by the predictors that describe the preferred web sections. The methodology introduced results in a Nested Stump Regression Tree that is an hierarchy of stump trees, where a stump is a tree with only one split or, equivalently, with only two terminal nodes. Suitable properties are outlined. As in first part of the contribution to the analysis of the web visit histories, a methodological description is provided by considering a web portal with a fixed set of web sections, i.e. a data set coming from the UCI Machine Learning Repository.

Keywords: Web path; Sequence rules; Recursive partitioning; Web Usage-Structure Mining.

The authors would like to thank two anonymous reviewers for their valuable comments and suggestions to greatly improve the quality of the paper.

Corresponding Author's Address: R. Siciliano, Department of Industrial Engineering, University of Naples Federic II, Corso Umberto I, 80138 Naples, Italy, email: roberta@unina.it.

Published online : 7 October 2017

1. Introduction

The methodological framework of this paper is the Web Usage-Structure Mining (Pecoraro and Siciliano, 2008; D'Ambrosio, Pecoraro, and Siciliano, 2008) as defined in the first contribution to the analysis of web visit histories (Siciliano, D'Ambrosio, Aria, and Amodio, 2016). The aim is to explore large web data repositories to predict the users' behavior derived from log-files or tracking applications, and especially visits information such as connection time, visited pages, downloaded documents, etc. At the same time, we are interested in detecting the most relevant connections between pages contents (e.g. of various websites or within one web portal). The traditional goal of Web Usage Mining process is just to extract information about how each page is related to the others in terms of navigation behavior.

Web Usage-Structure Mining is essential in defining successful strategies for the personalization of web application in which users' preferences must be supported, i.e. the presentation of news or promotional contents, the launching of new e-marketing campaigns, etc. When dealing with web mining the data can come from different sources, a single web site, a group of them, a server (Etzioni, 1996; Cooley, Mobasher, and Srivastava, 1999; Kosala and Blockeel, 2000; Srivastava, Cooley, Deshpande, and Tans, 2000; Linoff and Berry, 2002; Chakrabarti, 2002; Giudici and Figini, 2009; D'Ambrosio and Pecoraro, 2011). The size of such data is typically huge and their complexity requires a data mining strategy for their statistical analysis.

Part I to the contribution to the analysis of web visit histories (Siciliano et al., 2016) considered extending association rules theory to web data and providing new concepts of web (patterns) association and preference matrices, and of (indirect and direct) sequence rules (the so-called web paths). Distance-based visualization methods have been introduced to detect the most significant rules. In this paper, we focus our attention on *web event history analysis*. In other words, we are interested in identifying a prediction rule to explain the expected number of clicks for each navigation session up to the moment a surfer leaves the portal, click by click. Our goal is exploratory and the main idea is to detect web paths through a tree structure. Specifically, the regression tree framework is considered (Hastie, Tibshirani, and Friedman, 2009; Siciliano and Mola, 2000) where the response variable considered is the number of clicks and the predictors are the web preferences click by click. The main issue is to take into account the fact that at any click some navigators may leave the web while others continue the web navigation session. Therefore, a novel tree-based structure to detect web preferences in the navigation paths at any click is introduced.

The proposed methodology provides a hierarchy of stumps, that is called Nested Stump Regression Tree, where a stump is a tree with only one split or, equivalently, with two terminal nodes (Iba and Langley, 1992). Properties are outlined to describe the main differences of this structure in respect to standard tree structures.

It is worth noting that the most innovative aspect of this structure is the definition of child root nodes. Typically, a tree structure can be interpreted as a recursive partitioning of objects into subgroups that are internally homogeneous and externally heterogeneous with respect to a response variable. In the proposed structure, this is still true but there is a great difference: the groups of objects to be split include those objects which have not been excluded in terms of some predicate to be fixed for the entire structure (i.e. the so-called exclusion property). In our case, this property is specified by considering the active navigators at any click, thus at any level of the tree.

The range of the proposed methodology is a single portal. As in Part I, we consider as a real world case study, a data set coming from the UCI Machine Learning Repository. These data consist of about a million navigation sessions collected in a single day on msnbc.com, an American general purpose portal and from the news related portion of msn.com.

The rest of the paper is structured as follows. Section 2 reviews classification and regression tree structures. In Section 3, some basic definitions of web visit histories are recalled. In Section 4, the proposed methodology with the relevant notation is introduced, and the properties are outlined. Moreover, the real world data set, used as a case study, is presented and the results are shown. Section 5 ends the paper with some concluding remarks.

2. Recall on Tree-Based Methods

A tree is an oriented graph formed by a finite number of nodes departing from each node, as shown in Figure 1. The first node is called the root node. A distinction is made between terminal nodes, denoted by square boxes, and non-terminal nodes, denoted by circles. A non-terminal node is also known as a parent node, yielding two or more child nodes. Standard tree structures typically satisfy two properties: the *shape property*, i.e. each parent node has a fixed number, r, of child nodes (for a binary tree, $r = 2$) and the *heap property*, i.e. each parent node is greater than its own children nodes according to some comparison predicate that is fixed for the entire data structure.

Trees can be used for exploratory analysis of the dependence relation of a response variable either categorical (classification trees) or numeric (regression trees) on a set of predictors of numerical and/or categorical type. A binary tree can be built upon a recursive partitioning of

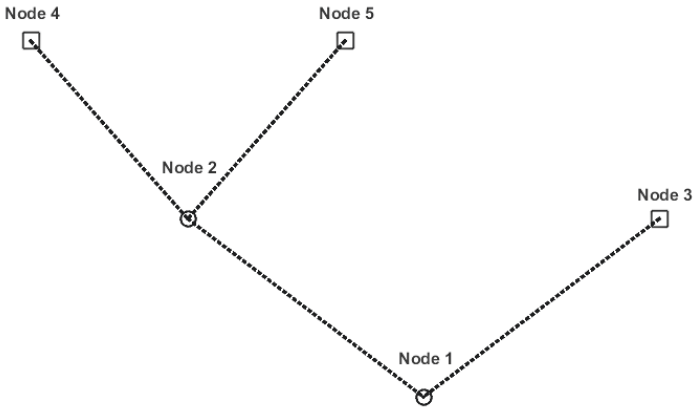

Figure 1. Example of a tree-based structure

a sample of objects into two subgroups to reduce the impurity of the response variable when passing from one node to its child nodes on the basis of the predictor measurements. The impurity can be expressed as a measure of variation/heterogeneity of the response variable distribution for regression/classification trees, respectively.

The comparison predicate states that when passing from the parent node to its child nodes the impurity measure always reduces. The splitting criterion maximizes the decrease of impurity by partitioning the objects in the parent node into two subgroups. Candidate splits are defined on the basis of the set of all splits of any predictor's modalities. As a matter of fact, it can be shown that maximizing the decrease of impurity is equivalent to maximizing the predictability power of the split. For this reason, known statistical indexes can be considered: the predictability tau index of Goodman and Kruskal for classification and the Pearson correlation coefficient for regression. This idea was introduced by the two-stage splitting criterion and the FAST algorithm (Siciliano and Mola, 1996; Mola and Siciliano, 1997; Siciliano and Mola, 2000). In the two-stage criterion, the predictor plays a global role in the partitioning procedure: the best split is selected on the basis of the set of splits generated by the best predictor that globally predicts the response variable better than other predictors. It is also possible to define a subset of best predictors that are ordered on the basis of their predictability power. The FAST algorithm, on the other hand, iterates the two-stages criterion until the global predictability power of the current best predictor is not greater than the local predictability power of the current best split.

Each terminal node is labeled by the average/modal class of the response variable and by the total node impurity for regression/classification. To pass from the exploratory tree to the decision tree for predicting the re-

sponse value/class for new cases, the CART methodology (Breiman, Friedman, Olshen, and Stone, 1984) suggests a pruning procedure and some selection rule of the final decision tree. The main idea is that the exploratory tree cannot be generalized as universal rule for new cases due to overfitting. The pruning procedure starts by considering the maximal expanded tree and by cutting off recursively the weakest link of the tree, which is the branch of the tree corresponding to the minimum complexity parameter between cost (in terms of increase of impurity) and benefit (in terms of reduced number of terminal nodes). This yields to a sequence of nested trees that are candidates decision trees. The final tree is chosen on the basis of the test sample estimate. Alternatively, ensemble methods, such as Bagging, Boosting and Random Forests (Breiman, 1996; Freund and Schapire, 1997; Dietterich, 2000; Breiman, 2001), define more accurate prediction rules despite the fact that no decision tree can be produced. It is worth noting that trees were also useful considered for missing data imputation and for data fusion (D'Ambrosio, Aria, and Siciliano, 2012).

3. Web Visit Histories: Some Basic Definitions

When dealing with web visit histories, the data set consists of N web navigators, or browsing sessions, which record the web sections visited click by click within the set $W = \{w_1, \ldots, w_j, \ldots, w_J\}$. Let $X = \{X_1, \ldots, X_v, \ldots, X_V\}$ be the set of the V categorical variables describing the web preferences at any click in the web navigation, where X_v is the web preference variable at the v-th click. Typically, the number V of clicks is chosen to be lower than the highest number of clicks during any of the N navigation sessions to have a consistent number of web navigations to analyze.

We consider the **web navigation matrix** $\mathbf{X} = \{x_{lv}\}$ (where $l = 1, \ldots, N$ and $v = 1, \ldots, V$) of N rows and V columns, where the general entry x_{lv} denotes the web page in the set W that is visited in the l-th navigation session at the v-th click, thus it can be equal to any j (for $j = 1, \ldots, J$) and $x_{lv} = 0$ if no web section is visited. By definition, the l-th navigation session includes non zero column entries till the exit from the website, while the remaining column entries are equal to zero.

As an example of the web navigation matrix, we consider a well known data set from the UCI Machine Learning Repository. It consists of about a million navigation sessions collected in a single day on msnbc.com, an American general purpose portal, and from the news related portion of msn.com. The set W includes seventeen main-pages or web sections ($\{Front-page, News, Tech, Local, Opinion, Onair, Misc, Weather, MSN-news, Health, Living, Business, MSN-sports, Sport, Sum-$

Table 1. The web navigation matrix (case study from the UCI Machine Learning Repository)

Session	First section visited	Second section visited	Third section visited	Fourth section visited	Fifth section visited	Sixth section visited	...
1	Frontpage	Sports	News	News	Weather		
2	Frontpage	Opinion	Local	Tech	Opinion	Opinion	Living
3	Weather	Travel	Tech				
4	News	News	News	Local	On-air	Frontpage	
5	BBS	Travel	Business	Travel	Living	Living	Living
6	Frontpage	Sports	Local	Sports	News	Opinion	
...	...	...	...	...	...	...	...

$mary, BBS, Travel\}$). Each navigation session is associated to a web user of the portal. In principle, the same person can access more than once to the same web portal, but each time the navigation session is recorded as distinct. Table 1 describes the structure of the data. Each row describes the clicking path of each navigation session or browsing session, registering, column by column, the visited pages from the entry till the exit from the website.

Being that the number of web pages visited in each navigation session are not fixed, each row may have a different number of entries. For example, the first navigation session goes on till the fifth click on the portal visiting respectively $\{Frontpage\}$, $\{Sport\}$, $\{News\}$, again $\{News\}$ and $\{Weather\}$; the third session goes on till the third click, etc.

We consider the *web click matrix* $\mathbf{Q} = \{q_{lv}\}$ (where $l = 1, \ldots, N$ and $v = 1, \ldots, V$) of N rows and V columns, where the general entry q_{lv} denotes the remaining number of clicks in the l-th navigation session at the v-th click. By definition, $q_{l(v)} = q_{l(v-1)} - 1$, with $q_{l(1)}$ counting the total number of non-zero entries in the l-th row of the web navigation matrix $\mathbf{X}$. At any v there are $N_{(v)}$ active navigators and $L_{(v)} = N - N_{(v)}$ navigators who left the web at the v-th click, the latter corresponds to the number of zero rows of the v-th column of $\mathbf{Q}$.

Siciliano et al. (2016) introduced suitable definitions and a methodology for web discovery analysis through association and sequence rules (Agrawal and Srikant, 1994; Zhang and Zhang, 2002; Blane and Giudici, 2002). Specifically, an association rule is a statement such as $w_i \Rightarrow w_j$, where w_i is the *Antecedent* web section and w_j is the *Consequent* web section. A **sequence rule of order v** is defined as ${w_i}^{(v')} \Rightarrow {w_j}^{(v)}$ $(v' < v)$, where the *Antecedent* web section ${w_i}^{(v')}$ is preferred at time v' and the *Consequent* web section ${w_j}^{(v)}$ at time v. For $v' = v - 1$ these sequence rules are **direct**, whereas for $v' < v - 1$ they are **indirect sequence rules**. For example, the association rule $\{Frontpage\} \Rightarrow \{News\}$ indicates that if a

navigator clicks on $\{Frontpage\}$ then also $\{News\}$ is preferred; instead the sequence rule $\{Frontpage^{(v')} \Rightarrow News^{(v)}\}$ says that the navigator has chosen $\{Frontpage\}$ at the v'-th click and then $\{News\}$ at the v-th click. In case $v' = v - 1$ the navigator chose the two pages in direct sequence, in the other cases for $v' < v - 1$ the navigator selected other pages in between them.

The web navigation data matrix can be summarized in terms of the **longitudinal web preference matrix**, which describes the presence or absence of each web section preference sequentially (click by click), namely $\tilde{\mathbf{Z}} = \{\mathbf{Z}_1|\ldots|\mathbf{Z}_v|\ldots|\mathbf{Z}_V\}$ of N rows and $J \times V$ columns, where the v-th matrix $\mathbf{Z}_v = [z_{lj(v)}]$ of N rows and J columns (for $v = 1, \ldots, V$) describes in disjoint coding the web preference variable X_v at the v-th click, namely $z_{lj(v)} = 1$ if the web page w_j of the set W has been visited in the l-th navigation session at the v-th click, and $z_{lj(v)} = 0$ otherwise.

The co-occurrences between any couple of web sections in the set W click by click can be recorded by the $J \times V$ square **longitudinal web association matrix** $\tilde{\mathbf{S}} = \tilde{\mathbf{Z}}'\tilde{\mathbf{Z}}$. The upper triangular blocks of the symmetric $\tilde{\mathbf{S}}$ matrix allow to derive the support measures for all *sequence rules between single web sections*, i.e. ${w_i}^{(v')} \Rightarrow {w_j}^{(v)}$, being

$$Sup_{{w_i}^{(v')} \Rightarrow {w_j}^{(v)}} = s_{i(v')j(v)}/N_{(v)}. \tag{1}$$

If we consider only the matrices with $v' = v - 1$, we can derive the support measures for the *direct sequence rules between single web sections click by click*, i.e. ${w_i}^{(v-1)} \Rightarrow {w_j}^{(v)}$. As an example, at click $v = 3$ the entries of the block association matrix $\mathbf{S}_{2,3}$ divided by $N_{(3)}$ provide the support measures of direct rules between the row antecedent web section preferred at second click and the column consequent web section preferred at the third click. Confidence and lift measures can be derived in a straightforward way as

$$Conf_{{w_i}^{(v')} \Rightarrow {w_j}^{(v)}} = s_{i(v')j(v)}/s_{i(v')+(v)}, \tag{2}$$

where $s_{i(v')+(v)} = \sum_j s_{i(v')j(v)}$, and

$$Lift_{{w_i}^{(v')} \Rightarrow {w_j}^{(v)}} = \frac{s_{i(v')j(v)}/s_{i(v')+(v)}}{s_{+(v')j(v)}/N_{(v)}}, \tag{3}$$

where $s_{+(v')j(v)} = \sum_i s_{i(v')j(v)}$.

For further details on the construction of the longitudinal web preference matrix and the longitudnal web association matrix and their properties, we refer to Siciliano et al. (2016, Section 2.5).

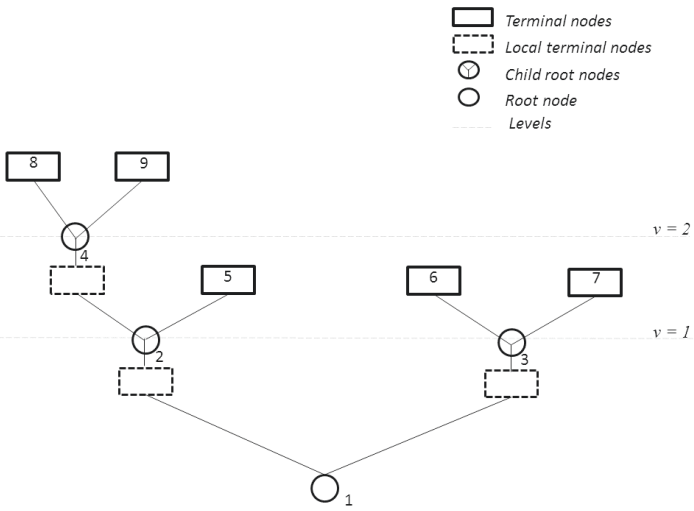

Figure 2. Example of Nested Stump Tree

4. Web Visit Histories: Exploratory Tree-Based Approach

4.1 Nested Stump Tree: The Structure and Its Properties

The case study data set suggested developing a new method for exploratory analysis of web data through a tree structure that is not a standard one. Indeed, when dealing with tree partitioning of web data, one must take into account that at any click some web navigators leave the web. We built up a hierarchy of stumps, where a stump is a tree with only one split, or equivalently with only two terminal nodes. This structure results in the **Nested Stump Tree** as depicted in Figure 2.

The Nested Stump Tree is characterized by levels that correspond to each click time $(v = 1, \ldots, V)$. It describes a recursive partitioning of the objects, i.e. the web navigators, into homogeneous subgroups with respect to a response variable taking into account that, at each level of the tree, some of them leave. Indeed, there are terminal nodes in the structure that generate new nodes, called *child root nodes*, when considering only the active navigators. We can build up a Nested Stump Regression Tree for a numerical response variable and a Nested Stump Classification Tree for a categorical response variable. At the bottom of the structure there is the root node that includes all web navigators that are considered for the analysis. The first stump yields the left and the right terminal nodes according to the split criterion. These nodes can be declared either *final terminal node* or *local terminal node* according to a stopping rule. Any local terminal node

generates a child root node yielding a new stump. The partitioning procedure finishes when all terminal nodes are declared final. Terminal nodes are denoted by square boxes and parent nodes, such as the root node and the child root nodes, by circles. Any parent node t yields the left terminal node numbered $2t$ and the right terminal node numbered $2t + 1$. The size of all terminal nodes is denoted by $\tilde{N}_t$. The size of any child root node t is denoted by N_t. The number of navigators who leave the web page at any local terminal node is denoted by L_t. It can be shown that $N_t = \tilde{N}_t - L_t$. At level v, there are $N_{(v)}$ active navigators partitioned into the child root nodes for $t = 2^v, \ldots, 2^{v+1} - 1$. It can be shown that

$$N_{(v)} = \sum_{t=2^v}^{2^{v+1}-1} \tilde{N}_t \tag{4}$$

for $v = 1, \ldots, V^*$, where V^* is the maximum number of levels of the hierarchy where all terminal nodes are final. Let H^* be the maximum number of final terminal nodes. Furthermore, from level $v - 1$ to level v there are $L_{(v)}$ navigators who leave the web, where $L_{(v)} = N_{(v)} - N_{(v-1)}$ and at any level v it holds

$$N_{(v)} = \sum_{t=2^{(v-1)}}^{2^v-1} N_t, \tag{5}$$

for $v = 2, \ldots, V^*$.

At level v, let Y_v be the response variable and let $\tilde{X}_v$ be the predictor that is measured in each learning sample of the child root nodes and the root node, i.e. $\mathcal{L}_{n_t} = \{y_{n_t v}, \tilde{x}_{n_t v}; n_t = 1, \ldots, N_t\}$ with $t = 2^v, \ldots, 2^{v+1} - 1$. Let $i_{Y_v}(t)$ be the impurity measure of the variable Y_v in the node t satisfying some properties as in CART methodology. The split of the objects can be induced by the split of either a predictor categories into two subgroups (simple split) or a compound predictors categories (multiple split), where a compound predictor is a cross-classification of more predictors categories in order to consider their interaction. In the former, the predictor $\tilde{X}_v$ is one; whereas, in the latter it is a compound variable. For sake of brevity, we will consider only simple splits.

The splitting criterion can be defined by searching for the split s of navigators which maximizes the relative decrease of impurity or the so-called *Impurity Proportional Reduction* when passing from the node t to its child nodes $2t$ and $2t + 1$, namely

$$\gamma_{Y_v|s}(t) = \frac{\Delta i_{Y_v|s}(t)}{i_{Y_v}(t)} = \frac{i_{Y_v}(t) - [i_{Y_v}(2t)p(2t) + i_{Y_v}(2t+1)p(2t+1)]}{i_{Y_v}}, \tag{6}$$

where $p(2t)$ and $p(2t+1)$ are the proportions of navigators falling respectively into the left child, $2t$, and the right child, $2t+1$. The stopping rule can be defined by choosing both a threshold value for the maximum of the impurity proportional reduction and a minimum percentage of objects falling in a terminal node with respect to the root node. Let T^* be the final Nested Stump Tree resulting from the partitioning procedure. The Nested Stump Tree satisfies the shape and heap property, which are standard properties of a tree structure. In particular, the structure is a hierarchy of stumps that are binary trees with only one split (i.e. $r = 2$) and each parent node is greater than or equal to its own child nodes in terms of node size and impurity measure. In addition, two more properties are satisfied, namely the exclusion property and the level property. The *exclusion property* allows to identify the child root nodes, by cutting off those navigators abandoning the web. The *level property* allows to determine the number of stumps at each level of the tree, which is at most two times the number of stumps generated by the previous level. The following statistical ratios within any node t and any click-level v (for $v = 1, \ldots, V^*$) help the interpretation of the Nested Stump Tree:

- the *Impurity Proportional Reduction* (IPR) is the maximum value of the splitting rule (6) at any node, i.e. $IPR(t) = \gamma_{Y_v|s^*}(t)$ where s^* is the best split at node t that discriminates the objects belonging to the left sub-node $2t$ from those belonging to the right sub-node $2t+1$;
- the *Within-Node Exclusion ratio* (WNE) is the relative proportion of objects that leave the partitioning procedure at the local terminal node t, i.e. $WNE(t) = L_t/\tilde{N}_t$;
- the *Within-Level Exclusion ratio* (WLE) is the relative proportion of objects that leave the partitioning procedure at the current level v, i.e. $WLE(v) = L_{(v)}/N_{(v-1)}$;
- the *Global Level Exclusion ratio* (GLE) is the global proportion of objects that leave the partitioning procedure with respect to the root node, at the current level v, i.e. $GLE(v) = L_{(v)}/N_1$;
- the *Tree Impurity* (I(T^*)) is the overall impurity measure of the final tree T^* and it is obtained by summing up the impurities of the final terminal nodes weighted by the proportions of objects falling in each node, i.e. $I(T^*) = \sum_{t \in H^*} i_{Y_v}(t)p(t)$, where H^* is the set of final terminal nodes.

It is also possible to consider a pruning procedure and a decision tree selection to identify a Nested Stump Decision Tree for predicting the response

class/value of new objects. Analogously to the CART pruning, a sequence of nested Nested Stump Trees can be considered by a selective algorithm that cuts off, at each iteration, the weakest link. This weakest link can be chosen as the branch T_t hold by the child root node t which minimizes the cost-benefit parameter value

$$\alpha_t = \frac{\sum_h \Delta i_{Y_v|s^*}(h)}{H_{T_t} - 1}, \tag{7}$$

where h denotes any child root node that is present in the branch T_t and H_{T_t} is the number of terminal nodes in the branch T_t. Rather than CART, the cost is evaluated in terms of impurity decrease (Cappelli, Mola, and Siciliano, 2002). The decision tree selection can be based on considering either an independent test sample of objects or a cross-validation estimate of the Tree Impurity. The main idea is to find the tree of the sequence derived from the pruning procedure to minimize the Tree Impurity Estimation.

4.2 Nested Stump Regression Tree for Web History Mining

The Nested Stump Regression Tree consists in a recursive binary partition of the active navigators at each node of the current level in such a way to obtain homogeneous subgroups of navigators in terms of the number of remaining clicks to be done. The aim is to discover the web navigation paths from one section to another such to identify the user's choices discriminating between those navigators who leave early the web from those who keep standing on the web.

At any click v, the response variable is the total number of remaining clicks, i.e. $Y_v = Q_v$, and the predictor is the web choice, i.e. $\tilde{X}_v = X_v$. Thus, the response variables are by turns the columns of the web click matrix; whereas, the predictors are by turns the columns of the web navigation matrix. At any click v, the feature representing the predictor is always the same, namely the web sections that can be visited (in the case study this feature has seventeen modalities). This variable is nominal and so the total number of candidate splits is equal to $(2^{J-1} - 1)$, where J is the number of categories. It is worth noting that this feature has different measurements depending on the click. Analogously, the total number of remaining clicks varies from click to click.

At any click v, there is a learning sample formed by the active navigators $N_{(v)}$ for which the response variable Y_v and the predictor X_v are measured. Specifically, these variables are the non-zero values in the v-th web click matrix $\mathbf{Q}$ and the corresponding web preferences of the set W in the v-th column of the web navigation data matrix $\mathbf{X}$. The within-node sum of squares of the response variable Y_v taking into account the sample of objects falling in the node t is considered as impurity measure, namely

$$i_{Y_v}(t) = dev(Y_v(t)) = \sum_{n_t=1}^{N_t} (y_{n_t v} - \bar{y}_v(t))^2, \tag{8}$$

where $\bar{y}_v(t)$ is the mean of the response variable Y_v within the node t.

At any node t of the level v the best split s_t^* is found by maximizing the *Impurity Proportional Reduction* (6), which is relative decrease in the within-node sum of squares of the response variable Y_v due to any split s_t induced by the $\tilde{X}_v$ predictor modalities. It can be shown that for the impurity measure (8), (6) yields the Pearson's correlation coefficient

$$\eta_t(Y_v|s_t) = \frac{dev_{Y_v}(t) - [dev_{Y_v}(2t)p(2t) + dev_{Y_v}(2t+1)p(2t+1)]}{dev_{Y_v}(t)}. \tag{9}$$

Hence, the best split s_t^* is found by maximizing the between-group deviation within the local terminal nodes or the decrease in the within-group deviation when passing from the child root node to the terminal node of the stump generated by the node t. As for the stopping rule, a node is declared to be terminal if the percentage of the active navigators is below a fixed threshold. The mean value of the variable Y_v at any node is used as assignment rule for the final terminal nodes. This value is an estimate of the expected number of clicks that the visitors, which are still surfing the portal, will perform before stopping their navigation session, with standard deviation measuring the within node impurity. For example, by assuming that Figure 2 shows a Nested Stump Regression Tree, we can deduce that, at the first level a partition of the N_1 navigators yields two subgroups that reduce the internal variation of the response variable Y_1 with respect to the first predictor X_1. Both the terminal nodes of the first stump are not final ones. They generate two child root nodes by considering the learning samples of size N_2 and N_3 respectively, thus omitting those navigators who left the web after the first click. The latter are denoted by L_2 and L_3, with $L_{(2)} = L_2 + L_3$. At the second level, both the binary partitions of each of the active navigators groups N_2 and N_3, respectively within the child root nodes $t = 2$ and $t = 3$, provide two subgroups reducing the variation of the response variable on the basis of the second predictor X_2. The best splits at each child root node at the second level describe the web preferences of distinct navigators' groups, when passing from the first click to the second click, or the so-called web paths, each coming from distinct choices at the first click level. The stump at the third child root node generates two final terminal nodes, whereas the stump at the second child root node yields the final terminal node 5 and another node that is further partitioned. The recursive partitioning continues until each terminal node is declared to be final terminal.

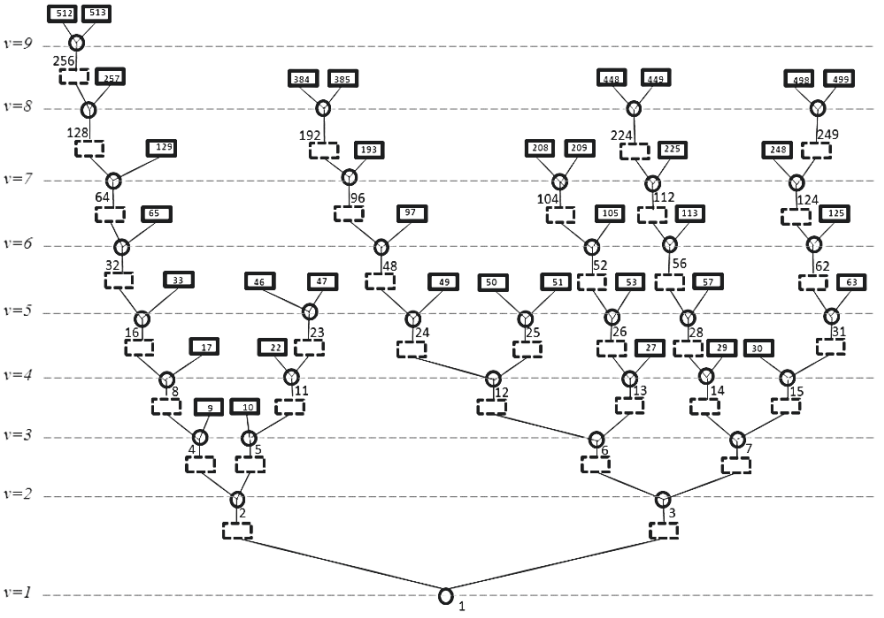

Figure 3. Nested Stump Regression Tree for the case study

4.3 Sequence Rules Visualization and Web Usage Paths Analysis

For the case study, we considered the raw data of users that arrived till the ninth navigation session (about the 90% of the data set). A partial view of the web navigation matrix is shown in Table 1.

Figure 3 shows the Nested Stump Regression Tree. The horizontal lines represents the level of the tree and they correspond to the clicking time. The figure is the graphical representation of Tables 2, 3 and 4, that reports the statistical information of any split at each level. The coding used for the web-sections is: x_1 = *Frontpage,* x_2 = *News* x_3 = *Tech* x_4 = *Local,* x_5 = *Opinion,* x_5 = *On air,* x_7 = *Misc* x_8 = *Weather,* x_9 = *MSN-news,* x_{10} = *Health,* x_{11} = *Living,* x_{12} = *Business,* x_{13} = *MSN-sports,* x_{13} = *Sport,* x_{15} = *Summary,* x_{16} = *BBS,* x_{17} =*Travel.*

The statistical information reported is, at any level, the level impurity reduction and the Within Level Exclusion, and at any node, the node size, the mean and standard deviation of the response variable (i.e. the expected remaining clicks), the IPR at any stump and the Within Node Exclusion (WNE), the splitting rule at the child root node. The terminal node that is not further partitioned is reported as terminal.

Table 2. Nested Stump Regression Tree for the case study: Statistical Information for Levels 1, 2, 3, 4.

	Node	Size	Expected clicks Mean	Expected clicks Std	IPR	WNE	Splitting rule
v1. Level impurity reduction: 6.377%							
Split 1	1	888056	2.951	2.415	6.377%	-	$X \neq x_1, x_5, x_7, x_8, x_{13}, x_{15}, x_{16}$
	2	466952	2.372	2.101		53.313%	
	3	421104	3.593	2.573		27.704%	
v2. Level impurity reduction: 4.610%. WLE: 41.170%. GLE=41.170%.							
Split 2	2	218006	2.938	2.203	6.202%	-	$X \neq x_7$
	4	206287	2.808	2.146		38.422%	
	5	11719	5.237	1.902		3.021%	
Split 3	3	304440	3.586	2.364	3.470%	-	$X = x_1, x_3, x_4, x_9, x_{12}, x_{13}, x_{15}, x_{16}$
	6	175310	3.292	2.306		29.051%	
	7	129130	3.986	2.345		17.890%	
v3. Level impurity reduction: 1.876%. WLE: 29.409%. GLE=58.471%							
Split 4	4	127027	2.936	2.042	2.978%	-	$X \neq x_7$
	8	122960	2.872	2.017		34.502%	
	9	4067	4.865	1.833		Terminal	
Split 5	5	11365	4.370	1.776	5.657%	-	$X \neq x_7, x_{11}, x_{14}, x_{15}, x_{17}$
	10	2176	3.505	2.079		Terminal	
	11	9189	4.574	1.630		0.965%	
Split 6	6	124381	3.231	2.112	1.143%	-	$X = x_1, x_3, x_4, , x_{15}$
	12	96210	3.109	2.085		1.143%	
	13	28171	3.648	2.150		19.247%	
Split 7	7	106029	3.636	2.135	1.011%	-	$X \neq x_4, x_5, x_7, x_8, x_{16}$
	14	59570	3.442	2.158		23.843%	
	15	46459	3.884	2.081		13.849%	
v4. Level IPR: 1.102%. WLE: 26.748%. GLE=70.094%							
Split 8	8	80536	2.858	1.842	1.666%	-	$X \neq x_7$
	16	75946	2.800	1.822		32.681%	
	17	4590	3.820	1.901		Terminal	
Split 11	11	9104	3.608	1.601	0.462%	-	$X \neq x_7, x_{11}, x_{14}, x_{15}, x_{17}$
	22	195	2.918	1.823		Terminal	
	33	8909	3.623	1.592		3.031%	
Split 12	12	67803	2.993	1.877	0.746%	-	$X = x_1, x_3, x_4, x_{15}$
	24	58491	2.927	1.865		30.649%	
	25	9312	3.407	1.901		19.534%	
Split 13	13	22749	3.279	1.912	0.948%	-	$X \neq x_4, x_5, x_7, x_8, x_{16}$
	26	19688	3.205	1.914		24.528%	
	27	3061	3.759	1.830		Terminal	
Split 15	14	45367	3.207	1.914	0.773%	-	$X = x_1, x_3, x_4, x_{15}$
	28	43727	3.174	1.911		25.463%	
	29	1640	4.080	1.785		Terminal	
Split 15	15	40025	3.348	1.862	1.177%	-	$X \neq x_4, x_5, x_7, x_8, x_{16}$
	30	6586	2.888	1.901		Terminal	
	31	33439	3.439	1.841		15.847%	

Table 3. Nested Stump Regression Tree for the case study: Statistical Information for Levels 5, 6.

	Node	Size	Expected clicks Mean	Expected clicks Std	IPR	WNE	Splitting rule
v5. Level impurity reduction: 0.847%. WLE: 26.491%. GLE=79.346%							
Split 16	16	51127	2.674	1.610	1.095%	-	$X \neq x_7$
	32	50290	2.251	1.604		32.591%	
	33	837	4.031	1.420		Terminal	
Split 23	23	8639	2.705	1.547	0.301%	-	$X \neq x_1, x_4, x_7, x_{10}, x_{13}, x_{15}, x_{16}, x_{17}$
	46	6069	2.649	1.513		Terminal	
	47	2570	2.837	1.617		Terminal	
Split 24	24	40564	2.778	1.628	0.690%	-	$X \neq x_2, x_7, x_8, x_{10}, x_{11}, x_{14}, x_{16}, x_{17}$
	48	35679	2.730	1.618		30.598%	
	49	4885	3.133	1.654		Terminal	
Split 25	25	7493	2.991	1.656	0.952%	-	$X \neq x_5, x_7, x_{16}$
	50	6647	2.932	1.661		Terminal	
	51	846	3.462	1.543		Terminal	
Split 26	26	14859	2.921	1.661	0.793%	-	$X \neq x_7$
	52	14279	2.892	1.657		27.257%	
	53	580	3.640	1.589		Terminal	
Split 28	28	32593	2.917	1.653	0.758%	-	$X \neq x_7$
	56	31888	2.895	1.651		27.258%	
	57	705	3.909	1.440		Terminal	
Split 31	31	28140	2.898	1.642	0.892%	-	$X \neq x_5, x_7$
	62	25013	2.842	1.628		26.601%	
	63	3127	3.347	1.687		Terminal	
v6. Level impurity reduction: 0.690%. WLE: 29.618%. GLE=87.546%							
Split 32	32	33900	2.450	1.364	0.865%	-	$X \neq x_7$
	64	33464	2.436	1.359		33.908%	
	65	436	3.505	1.286		Terminal	
Split 48	48	24762	2.492	1.368	0.388%	-	$X \neq x_2, x_4, x_5, x_7, x_{10}, x_{11}, x_{14}, x_{16}$
	96	20862	2.445	1.363		33.808%	
	97	3900	2.744	1.365		3.031%	
Split 52	52	10387	2.601	1.390	0.523%	-	$X \neq x_7$
	104	10154	2.544	1.389		29.762%	
	105	233	3.313	1.266		Terminal	
Split 56	56	23196	2.606	1.377	0.716%	-	$X \neq x_7$
	112	22813	2.590	1.375		29.119%	
	113	383	3.520	1.184		Terminal	
Split 62	62	18357	2.509	1.391	0.435%	-	$X \neq x_7, x_{10}$
	124	17911	2.497	1.389		33.549%	
	125	446	3.004	1.346		Terminal	

In each table, within each level of v, the sum of the size of terminal nodes is equal to the size of the respective parent node. For each local terminal node, the within node exclusion ratio (WNE) is reported. If the terminal node is not considered final terminal, at level $v+1$ the within-node size,

Table 4 Nested Stump Regression Tree for the case study: Statistical Information for Levels 7, 8, 9.

	Node	Size	Expected clicks Mean	Expected clicks Std	IPR	WNE	Splitting rule
v7. Level impurity reduction: 0.675%. WLE: 32,388%. GLE=91.990%							
Split 64	64	22117	2.173	1.093	0.750%	-	$X \neq x_7$
	128	21896	2.164	1.090		36.354%	
	129	221	3.090	0.973		Terminal	
Split 96	96	13809	2.183	1.093	0.736%	-	$X \neq x_1, x_4, x_7, x_{10}, x_{13}, x_{15}, x_{16}, x_{17}$
	192	12654	2.154	1.084		36.479%	
	193	1155	2.509	1.142		Terminal	
Split 104	104	7132	2.256	1.109	0.676%	-	$X \neq x_2, x_7, x_8, x_{10}, x_{11}, x_{14}, x_{16}, x_{17}$
	208	3253	2.155	1.115		Terminal	
	209	3879	2.341	1.097		Terminal	
Split 112	112	16170	2.243	1.097	0.613%	-	$X \neq x_5, x_7, x_{16}$
	224	15985	2.234	1.095		33.438%	
	225	185	3.049	0.968		Terminal	
Split 124	124	11902	2.253	1.097	0.546%	-	$X \neq x_7$
	248	748	1.961	1.079		Terminal	
	249	11154	2.272	1.095		31.782%	
v8. Level impurity reduction: 0.069%. WLE: 34.797%. GLE=95.471%							
Split 128	128	13936	1.828	0.807	0.069%	-	$X \neq x_7$
	256	13875	1.826	0.807		42.775%	
	257	61	2.246	0.741		Terminal	
Split 192	192	8038	1.816	0.804	0.375%	-	$X \neq x_2, x_4, x_5, x_7, x_{10}, x_{11}, x_{14}, x_{16}$
	384	7745	1.805	0.801		Terminal	
	385	293	2.119	0.841		Terminal	
Split 224	224	10640	1.854	0.807	0.445%	-	$X \neq x_7$
	448	4241	1.794	0.804		Terminal	
	449	6399	1.894	0.806		Terminal	
Split 249	249	7609	1.865	0.807	0.229%	-	$X \neq x_7$
	498	7948	1.859	0.807		Terminal	
	499	111	2.279	0.741		Terminal	
v9. Level impurity reduction: 0.300%. WLE: 42.775%. GLE=99.106%							
Split 256	256	7940	1.443	0.497	0.300%	-	$X \neq x_7, x_{10}$
	512	7905	1.441	0.496		Terminal	
	513	35	1.857	0.349		Terminal	
Tree impurity: 3.488%							

adjusted for the ratio of leaving navigators, is reported. For example, the first *stump* at node 1 at level $v = 1$ provides two terminal nodes which are not declared to be final terminal. Within the left local terminal node 2 the WNE is equal to 53.31% of navigators leaving the web navigation; as a result, at the level $v = 2$ the child root node 2 includes $\tilde{N}_2 = 218006$ active navigators with the expected number of clicks equal to 2.938 and standard

deviation equal to 2.203. The child root node 2 provides a new *stump* for which the navigators that do not choose the seventh web section ($\{Misc\}$) go to the left child node 4 and the others to the right node 5.

A group of discriminant web sections, e.g. $\{Misc\}$, $\{Opinion\}$, $\{Weather\}$, $\{Health\}$, $\{BBS\}$), brings the surfers to leave the web. For example, in the stump coded as Split 4 the navigators who choose $\{Misc\}$ go into the right child node 9 which is declared to be terminal, all the others keep standing on the web. This prediction rule is valid till the last level of the same path, and this is also true for all the other paths involving the above-mentioned discriminant web sections. Table 4 reports in the last row the relative impurity measure of the Nested Stump Regression Tree as the ratio between the overall impurity measure and the impurity in the root node.

The tree-structure is also useful to inquire the direct sequence rules through the paths identified by the nested stump regression tree. Tables from 5 to 7 show an example of how the direct sequence rules can be associated with a tree. In the computation of the sequence rules, we considered a minimum support equal to 0.01 and a lift measure larger than 1.

Tables 5, 6 and 7 show the direct sequence rules from node 1 to node 2, from node 2 to node 4 and from node 5 to node 10 respectively. Antecedent items are the web sections that are visited at the previous level with respect to level v, and the consequent items are the web sections visited at level $v + 1$.

Data analyses were performed with our own program written in MatLab language in a Computer Intel Core i5-3317U 1.70 GHz and 4GB of RAM. It can be noted that, in general, users tend to stay on the same pages visited the previous time. On the other hand the sequence rules are consistent with the splitting rules, even if these measures are not connected in any way. For example, the direct sequence rules detected in Table 5 confirm the splitting rule of Split 1 in Table 2. In other words, once the sample was partitioned Table 5 hows that significant sequence rules in passing from node 1 to node 2 involve as antecedents web sections visited by surfers moving from node 1 to node 2. We get the same conclusion by comparing the direct sequence rules showed in Tables 6 and 7 with the rules governing the split of the stumps.

5. Concluding Remarks

This paper considers a new methodological framework for web usage-structure mining. Namely, it is possible to explore web navigation behavior through sequence rules. The most interesting paths can be discovered by considering the support, the confidence and the lift measures.

A novel tree-based structure, namely the Nested Stump Tree, is introduced to deal with both numerical and categorical responses, yielding to

Table 5. Direct sequence rules for the case study: node 1 - node 2

rule	support	confidence	lift
$x_2 \Rightarrow x_2$	0.117	0.814	4.790
$x_3 \Rightarrow x_3$	0.061	0.606	5.801
$x_4 \Rightarrow x_4$	0.076	0.683	5.823
$x_6 \Rightarrow x_9$	0.018	0.428	1.503
$x_{10} \Rightarrow x_{10}$	0.017	0.588	21.312
$x_{11} \Rightarrow x_{11}$	0.015	0.544	18.256
$x_{14} \Rightarrow x_{14}$	0.076	0.785	6.043
$x_6 \Rightarrow x_{15}$	0.024	0.741	2.602

Table 6. Direct sequence rules for the case study: node 2 - node 4

rule	support	confidence	lift
$x_1 \Rightarrow x_1$	0.027	0.601	10.717
$x_2 \Rightarrow x_2$	0.117	0.764	4.966
$x_3 \Rightarrow x_3$	0.059	0.650	7.518
$x_4 \Rightarrow x_4$	0.084	0.748	6.401
$x_6 \Rightarrow x_6$	0.079	0.691	5.196
$x_6 \Rightarrow x_7$	0.018	0.552	4.152
$x_8 \Rightarrow x_8$	0.018	0.654	29.843
$x_9 \Rightarrow x_9$	0.023	0.528	12.073
$x_{10} \Rightarrow x_{10}$	0.024	0.578	17.236
$x_{11} \Rightarrow x_{11}$	0.016	0.494	17.982
$x_{12} \Rightarrow x_{12}$	0.073	0.760	7.498
$x_{13} \Rightarrow x_{13}$	0.012	0.563	29.811
$x_{14} \Rightarrow x_{14}$	0.140	0.860	5.407

Table 7. Direct sequence rules for the case study: node 5 - node 10

rule	support	confidence	lift
$x_4 \Rightarrow x_2$	0.021	0.692	2.222
$x_4 \Rightarrow x_4$	0.142	0.903	2.901
$x_6 \Rightarrow x_6$	0.141	0.948	1.552
$x_6 \Rightarrow x_7$	0.414	0.761	1.257
$x_9 \Rightarrow x_9$	0.011	0.526	31.053
$x_{10} \Rightarrow x_{10}$	0.010	0.642	39.283
$x_6 \Rightarrow x_{15}$	0.026	0.938	1.536

Nested Stump Regression Tree and Nested Stump Classification Tree respectively. The proposed tree-based structure is formed by a hierarchy of stumps and it is characterized by the shape and the heap properties which hold for standard trees, in addition it satisfies new properties, namely the exclusion and the level properties. The main issue is to consider, at each node and at each level of the structure, a learning sample where some of the objects leave the partitioning procedure for a given criterion. In web mining, these objects are the users who may leave the web at any click. A general definition of CART-like splitting criterion and of a stopping rule are defined to build up the exploratory tree. It is worth nothing that it is not feasible to define a measure of goodness (or badness) of fit of the entire tree-based structure. At any level the statistical units are independent. A way to consider the informative overall power of the structure is looking at the global decrease of the impurity measure. For the case study, and in general in the web mining framework, the statistical unit of the web navigation matrix is always the navigation session. Indeed, it is only relevant to understand the web pattern preferences, no matter who is the subject being in the navigation session. If a unique identifier for subjects can be retrieved, it could be possible to use further variables, such as country, gender, age, etc. in the analysis. In our case study this information was not available. On the contrary, a possible strategy of data analysis in presence of personal information could involve longitudinal tree-based methods approaches (Fu and Simonoff, 2015; Fokkema et al., 2015) or ensemble methods dealing with longitudinal data (Vezzoli, 2011).

In this paper, the regression case has been exploited. As case study, a well known data set from the UCI Machine Learning Repository has been considered to show the application of the proposed approach for exploring web navigation behavior. Nested Stump Regression Trees are introduced to investigate web paths preferences that lead the navigators to leave earlier the web with respect to others who keep standing on the web. The remaining number of clicks before leaving is the response variable and the web preference at the current click is the predictor according to which the best split is selected. Once the tree structure is built it is possible to interpret the tree paths through direct sequence rules and also to derive their strength in terms of support, confidence and lift measures. Classification Trees will be considered in another paper.

References

AGRAWAL, R., and SRIKANT, R. (1994), "Fast Algorithms for Mining Association Rules", *Proceedings of the 20th International Conference on Very Large Data Bases, VLDB*, Vol. 1215, pp. 487–499.

BLANC, E., and GIUDICI, P. (2002), "Sequence Rules for Web Clickstream Analysis", in *Advances in Data Mining*, Berlin, Heidelberg: Springer, pp. 1–14.

BREIMAN, L. (1996), "Bagging Predictors", *Machine Learning, 24(2)*, 123–140.

BREIMAN, L. (2001), "Random Forests", *Machine Learning, 45(1)*, 5–32.

BREIMAN, L., FRIEDMAN, J., OLSHEN, R.A. and STONE, C.J. (1984), *Classification and Regression Trees*, Boca Raton: CRC Press.

CAPPELLI, C., MOLA, F., and SICILIANO, R. (2002), "A Statistical Approach to Growing a Reliable Honest Tree", *Computational Statistics and Data Analysis, 38(3)*, 285–299.

CHAKRABARTI, S. (2002), *Mining the Web: Discovering Knowledge from Hypertext Data*, The Netherlands: Elsevier.

COOLEY, R., MOBASHER, B., and SRIVASTAVA, J. (1999), "Data Preparation for Mining World Wide Web Browsing Patterns", *Knowledge and Information Systems, 1(1)*, 5–32.

D'AMBROSIO, A., ARIA, M., and SICILIANO, R. (2012), "Accurate Tree-Based Missing Data Imputation and Data Fusion Within the Statistical Learning Paradigm", *Journal of Classification, 29(2)*, 227–258.

D'AMBROSIO, A., and PECORARO, M. (2011), "Multidimensional Scaling as Visualization Tool of Web Sequence Rules", in *Classification and Multivariate Analysis for Complex Data Structures*, Berlin, Heidelberg: Springer, pp. 309–316.

D'AMBROSIO, A., PECORARO, M., and SICILIANO, R. (2008), "Web Preferences Visualization Through Multidimensional Scaling and Trees", in *DATAVIZ VI International Conference: Statistical Graphics: Data and Information Visualization in Today's Multimedia Society*, Bremen, June 25–28, 2008.

DIETTERICH, T.G. (2000), "Ensemble Methods in Machine Learning", in *Multiple Classifier Systems*, Berlin: Springer, pp. 1–15.

ETZIONI, O. (1996), "The World-Wide Web: Quagmire or Gold Mine?", *Communications of the ACM, 39(11)*, 65–68.

FOKKEMA, M., SMITS, N., ZEILEIS, A., HOTHORN, T., and KELDERMAN, H. (2015), "Detecting Treatment-Subgroup Interactions in Clustered Data with Generalized Linear Mixed-Effects Model Trees", Working Papers, Faculty of Economics and Statistics, University of Innsbruck, ftp://ftp.repec.org/opt/ReDIF/RePEc/inn/wpaper/2015-10.pdf.

FREUND, Y., and SCHAPIRE, R.E. (1997), "A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting", *Journal of Computer and System Sciences, 55(1)*, 119–139.

FU, W., and SIMONOFF, J.S. (2015), "Unbiased Regression Trees for Longitudinal and Clustered Data", *Computational Statistics and Data Analysis, 88*, 53–74.

GIUDICI, P., and FIGINI, S. (2009), *Applied Data Mining: Statistical Methods for Business and Industry*, New York: John Wiley and Sons.

HASTIE, T., TIBSHIRANI, R., and FRIEDMAN, J. (2009), *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, Berlin: Springer.

IBA, W., and LANGLEY, P. (1992), "Induction of One-Level Decision Trees", in *Proceedings of the Ninth International Conference on Machine Learning*, pp. 233–240.

KOSALA, R., and BLOCKEEL, H. (2000), "Web Mining Research: A Survey", *ACM SIGKDD Explorations, 2*, 1–15.

LINOFF, G.S, and BERRY, M.J. (2001), *Mining the Web: Transforming Customer Data into Customer Value*, New York: John Wiley and Sons, Inc.

MOLA, F., and SICILIANO, R. (1997), "A Fast Splitting Procedure for Classification and Regression Trees", *Statistics and Computing, 7*, 208–216.

PECORARO, M., and SICILIANO, R. (2008), "Statistical Methods for User Profiling in Web Usage Mining", in *Handbook of Research on Text and Web Mining Technologies*, eds. M. Song and Y.B. Wu, Hershey PA: Idea Group Inc., pp. 359–368.

SICILIANO, R., D'AMBROSIO, A., ARIA, M., and AMODIO, S. (2016), "Analysis of Web Visit Histories, Part I: Distance-Based Visualization of Sequence Rules", *Journal of Classification, 33(2)*, 298–324.

SICILIANO, R., and MOLA, F. (1996), "A Fast Regression Tree Procedure", in *Proceedings of the 11th International Workshop on Statistical Modeling*, eds. A. Forcina, G.M. Marchetti, R. Hatzinger, and G. Galmacci, Citta' di Castello IT: Graphos, pp. 332–340.

SICILIANO, R., and MOLA, F. (2000), "Multivariate Data Analysis Through Classification and Regression Trees", *Computational Statistics and Data Analysis, 32*, 285–301.

SRIVASTAVA, J., COOLEY, R., DESHPANDE, M., and PANG-NING T., (2000), "Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data", *ACM SIGKDD Explorations Newsletter, 1(2)*, 12–23.

VEZZOLI, M. (2011), "Exploring the Facets of Overall Job Satisfaction Through a Novel Ensemble Learning", *Electronic Journal of Applied Statistical Analysis, 4(1)*, 23–38.

ZHANG, C., and ZHANG, S. (2002), *Association Rule Mining: Models and Algorithms*, Heidelberg: Springer.