Integrating Bayesian networks and decision trees in a sequential rule-based transportation model

Integrating Bayesian Networks and Decision Trees in a Sequential Rule-Based Transportation Model

Davy Janssens

Geert Wets*

Tom Brijs

Koen Vanhoof

***Email*:{davy.janssens;geert.wets;tom.brijs;koen.vanhoof}@luc.ac.be**

Limburgs Universitair Centrum

Transportation Research Institute

Universitaire Campus

Gebouw D

B-3590 Diepenbeek

Belgium

Theo Arentze

Harry Timmermans

***E-mail*: {t.a.arentze; h.j.p.timmermans}@bwk.tue.nl**

Eindhoven University of Technology

Urban Planning Group

PO Box 513

5600 MB Eindhoven

The Netherlands

* Corresponding author. Tel.: +32 (0)11 26.86.49 Fax: +32(0)11 26.87.00

Integrating Bayesian Networks and Decision Trees in a Sequential Rule-Based Transportation Model

ABSTRACT

Several activity-based transportation models are now becoming operational and are entering the stage of application for the modelling of travel demand. Some of these models use decision rules to support its decision making instead of principles of utility maximization. Decision rules can be derived from different modelling approaches. In a previous study, it was shown that Bayesian networks outperform decision trees and that they are better suited to capture the complexity of the underlying decision-making. However, one of the disadvantages is that Bayesian networks are somewhat limited in terms of interpretation and efficiency when rules are derived from the network, while rules derived from decision trees in general have a simple and direct interpretation. Therefore, in this study, the idea of combining decision trees and Bayesian networks was explored in order to maintain the potential advantages of both techniques. The paper reports the findings of a methodological study that was conducted in the context of Albatross, which is a sequential rule based model of activity scheduling behaviour. The results of this study suggest that integrated Bayesian networks and decision trees can be used for modelling the different choice facets of Albatross with better predictive power than CHAID decision trees. Another conclusion is that there are initial indications that the new way of integrating decision trees and Bayesian networks has produced a decision tree that is structurally more stable.

Keywords: Transportation, Activity-based transportation modelling, Bayesian networks, Decision trees, BNT classifier

Topic Area: Transportation Modelling

1. INTRODUCTION

Over the last decade, activity-based transportation models have set the standard for modelling travel demand. The most important characteristic in these models is that travel demand is derived from the activities that individuals and households need or wish to perform. The main advantage is that travel has no longer an isolated existence in these models, but is perceived as a way to perform activities and to realize particular goals in life.

Several activity-based models are now becoming operational and are entering the stage of application in transport planning (e.g. Bhat, *et al.* 2004, Bowman, *et al.* 1998; Arentze and Timmermans 2000). This multitude of modelling attempts seems to converge into two approaches. First, discrete choice utility-maximizing models that were originally developed for trip and tour data, were extended to activity-based models by including more facets. The second approach emphasizes the need for rule-based computational process models, since it is claimed by several scholars that utility-maximizing models do not always reflect the true behavioural mechanisms underlying travel decisions (people may reason more in terms of "if-then" structures than in terms of utility maximizing decisions). For this reason, several studies have shown an increasing interest in computational process models to predict activity-travel patterns. This study contributes to this line of research by narrowing down on one operational computational process model, i.e. the Albatross system (A Learning Based Transportation Oriented Simulation System), developed by Arentze and Timmermans (2000) for the Dutch Ministry of Transport. Albatross is a multi-agent *rule-based* system that predicts which activities are conducted where, when, for how long, with whom and the transport mode involved. It uses decision rules to predict each of those facets (where, when, etc.) and to support scheduling decisions. These decision rules can be derived by various decision tree induction algorithms (C4.5, CHAID, CART, etc.). Comparative studies by Wets, *et al.* (2000) and Moons, *et al*. (2004) found evidence that different kinds of decision tree induction algorithms achieve comparable results. A previous study by Janssens, *et al.* (2004) suggested that Bayesian networks outperform decision trees and that they are better suited to capture the complexity of the underlying decision-making process. Especially, it was found that Bayesian networks are potentially valuable to take into account the many (inter)dependencies among the variables that make up the complex decision-making process. However, the study also revealed that Bayesian networks had some disadvantages. First, in cases where decision rules need to be derived from the Bayesian network, the technique seemed to be somewhat limited. In particular, each decision rule that is derived from the network contains the same number of conditions, resulting in potential sub-optimal decision-making. Second, the interpretation of the rules may be an issue. It should be realized that decision

rules which are derived from decision trees have a simple direct interpretation: condition states are directly related to choices. In contrast, Bayesian networks link more variables in sometimes complex, direct and indirect ways, making interpretation more problematic. Consequently, it may be interesting to explore the possibility of combining these approaches and to examine where advantages can be maintained. In this paper, the results of such a study are reported. To this end, a novel classification technique is proposed in this paper that integrates decision trees and Bayesian networks. The new heuristic is referred to as a Bayesian Network Augmented Tree (BNT) in the reminder of this paper.

The remainder of the paper is organized as follows. First, the conceptual framework underlying the Albatross-system is briefly discussed in order to provide some background information with respect to this transportation model. Next, we will elaborate on the traditional decision tree formalism as it is also used in the original Albatross system. Third, Bayesian network learning will be introduced. In this section, general concepts such as parameter learning, entering evidences and structural learning will be given, along with a problem formulation and a description of the new BNT classification algorithm. Section 5 then describes the design of the experiments that were carried out to validate the new approach and gives an overview of the data that were used. Section 6 provides a detailed quantitative analysis and compares the performance of Bayesian networks and BNT at two levels: the activity pattern level and the trip level. Finally, conclusions and implications for the development and application of future activity-based models of travel demand are reported.

2. THE ALBATROSS SYSTEM

The Albatross system (Arentze and Timmermans 2000) is a computational process model that relies on a set of decision rules to predict activity-travel patterns. Rules are typically extracted from activity diary data. The activity scheduling agent of Albatross is the core of the system which controls the scheduling processes in terms of a sequence of steps. These steps are based on an assumed sequential execution of decision tables to predict activity-travel patterns (see Figure 1). The first step involves for each person decisions about which activities to select, with whom the activity is conducted and the duration of the activity. The order in which (the non-work) activities are evaluated is pre-defined as: daily shopping, services, non-daily shopping, social and leisure activities. The assignment of a scheduling position to each selected activity is the result of the next two steps. After a start time interval is selected for an activity, trip-chaining decisions determine for each activity whether the activity has to be connected with a previous

and/or next activity. Those trip chaining decisions are not only important for timing activities but also for organizing trips into tours. The next steps involve the choice of transport mode for work (referred to as mode1), the choice of transport mode for other purposes (referred to as mode2) and the choice of location. Possible interactions between mode and location choices are taken into account by using location information as conditions of mode selection rules. Each decision in the Albatross system (see oval boxes of Figure 1) is extracted from activity travel diary data using a Chi-squared based technique (hereafter referred to as CHAID, (Kass 1980)). As mentioned above, CHAID is a widely-used decision-tree induction method.

<INSERT FIGURE 1 HERE>

3. DECISION TREES

3.1. General Concepts

Decision trees are state-of-the art techniques, which are used to make decisions from a set of instances. There are two types of nodes in a decision tree: decision nodes and leaves. Leaves are the terminal nodes of the tree and they specify the ultimate decision of the tree. Decision nodes involve testing a particular attribute. Usually, the test at a decision node compares an attribute value with a constant. Ultimately, to classify an unlabeled instance, the case is routed down the tree according to the values of the attributes tested in successive decision nodes and when a leaf is reached, the instance is classified according to the probability distribution over all classification possibilities.

The decision tree is typically constructed by means of a “divide-and-conquer” approach. This means that first an attribute is selected to place at the root node of the tree. This root node splits up and divides the dataset into different subsets, one for every value of the root node. Each value is specified by a branch. Then, the construction of the tree becomes a recursive problem, since the process can be repeated for every branch of the tree. It should be noted that only those instances that actually reach the branch are used in the construction of the tree. In order to determine which attribute to split on, given a set of examples with different classes, different algorithms can be adopted (C4.5, CHAID, CART). The CHAID algorithm in Albatross, starts at a root tree node, dividing into child tree nodes until leaf tree nodes terminate branching. The splits are determined using the Chi Squared test.

After the decision tree is constructed, it is easy to convert the tree into a rule set by deriving a rule for each path in the tree that starts at the root and ends at the leaf node. Decision rules are often represented in a decision table formalism. A decision table represents an exhaustive set of mutual exclusive expressions that

link conditions to particular actions, preferences or decisions. The decision table formalism guarantees that the choice heuristics are exclusive, consistent and complete. A simplified example of a decision tree along with its corresponding decision table is represented in Figure 2.

<INSERT FIGURE 2 HERE>

3.2. Decision trees: problem formulation

Despite their huge popularity, it was already shown in other application domains (Bloemer, *et al.* 2003) that the model structure of decision trees can sometimes be instable. This means that when carrying out multiple tests, mostly the same variables enter the decision tree but the order in which they enter the tree is different. The reason for this is known as "variable masking", i.e. if one variable is highly correlated with another, then a small change in the sample data (given several tests) may shift the split in the tree from one variable to another.

4. BAYESIAN NETWORK LEARNING

4.1. General Concepts

A Bayesian network consists of two components (Pearl, 1988): first, a directed acyclic graph (DAG) in which nodes represent stochastic domain variables and directed arcs represent conditional dependencies between the variables (see definition 1-3) and second, a probability distribution for each node as represented by conditional dependencies captured with the directed acyclic graph (see definition 4). Bayesian networks are powerful representation and visualization tools that enable users to conceptualise the association between variables. However, as will be explained below, Bayesian networks can also be used for making predictions. To formalize, the following definitions are relevant:

Definition 1 A **directed acyclic graph** (DAG) is a directed graph that contains no directed cycles.■

Definition 2 A **directed graph** G can be defined as an ordered pair that consists of a finite set V of vertices or nodes and an adjacency relation E on V. The Graph G is denoted as (V,E). For each $(a, b) \varepsilon E$ (a and b are nodes) there is a directed edge from node a to node b. In this representation, a is called a **parent** of b and b is called a **child** of a. In a graph, this is represented by an arrow which is drawn from node a to node b. For any $a \in V$, $(a,a) \notin E$, which means that an arc cannot have a node as both its start and end point. Each node in a network corresponds to a particular variable of interest. ■

Definition 3 Edges in a Bayesian network represent direct conditional dependencies between the variables. The absence of edges between variables denotes statements of independence. We say that variables *B* and *C* are **independent** given a set of variables *A* if $P(c|b,a)=P(c|a)$ for all values *a*,*b* and *c* of variables *A*, *B* and *C*. Variables *B* and *C* are also said to be **independent conditional** on *A*. ■

Definition 4 A Bayesian network also represents distributions, in addition to representing statements of independence. A distribution is represented by a set of **conditional probability tables (CPT)**. Each node *X* has an associated CPT that describes the conditional distribution of *X* given different assignments of values for its parents. ■

<INSERT FIGURE 3 HERE>

The definitions mentioned above are graphically illustrated in figure 3 by means of a simple hypothetical example. Learning Bayesian networks has traditionally been divided into two categories (Cheng, *et al.* 1997): structural and parameter learning. Since these learning phases are relevant for the new integrated BNT classifier, the following sections elaborate on them into detail.

4.2. Parameter learning

Parameter learning determines the prior CPT of each node of the network, given the link structures and the data. It can therefore be used to examine quantitatively the strength of the identified effect. As mentioned above, a conditional probability table $P(A|B_1 \ldots B_n)$ has to be attached to each variable *A* with parents $B_1, \ldots, B_n$. Note that if *A* has no parents, the table reduces to unconditional probabilities P(*A*). According to this logic, for the example Bayesian network depicted in Figure 3, the prior unconditional and conditional probabilities to specify are: P(Driving License); P(Gender); P(Number of cars); P(Mode choice|Driving License, Gender, Number of cars). Since the variables "Number of cars", "Gender" and "Driving license" are not conditionally dependent on other variables, calculating their prior frequency distribution is straightforward. Calculating the initial probabilities for the "Mode choice" variable is computationally more demanding.

In order to calculate the prior probabilities for the "Mode choice" variable, the conditional probability table for P(Mode Choice| Driving License, Gender, Number of cars) was set up in the first part of Table 1. Again, this is straightforward mathematical calculus. In order to get the prior probabilities for the Mode

Choice variable, we now first have to calculate the joint probability P(Choice, Gender, Number of cars, Driving License) and then marginalize "Number of cars", "Driving License" and "Gender" out. This can be done by applying *Bayes' rule*, which states that: P(Choice, Gender, Number of cars, Driving License)=P(Choice|Gender, Number of cars, Driving License)*P(Gender, Number of cars, Driving License). Since "Gender", "Number of cars" and "Driving License" are independent, the equation can be simplified for this example as: P(Choice, Gender, Number of cars, Driving License)=P(Choice|Gender, Number of cars, Driving License)*P(Gender)*P(Number of cars)*P(Driving License). Note that P(Gender=male; Gender=female)=(0.75; 0.25), P(Driving License=yes; Driving license=no) = (0.6; 0.4) and P(Number of cars=1;Number of cars>1)=(0.2;0.8) which are the prior frequency distributions for those 3 variables. By using this information, the joint probabilities were calculated in the middle part of table 1. Marginalizing "Gender", "Number of cars" and "Driving License" out of P(Choice, Gender, Number of cars, Driving License) yields P(Mode Choice=bike; Mode Choice=car) = (0.506; 0.494). These are the prior probabilities for the "Mode choice" variable. Of course, computations become more complex when "Gender", "Number of cars" and "Driving License" are dependent. Fortunately, in these cases, probabilities can be calculated automatically by means of probabilistic inference algorithms that are implemented in Bayesian network-enabled software.

<INSERT TABLE 1 HERE>

4.3. Entering evidences

In fact, Figure 3 only depicts the prior distributions for each variable. This is useful but not very innovative information. An important strength of Bayesian networks, however, is to compute posterior probability distributions of the variable under consideration, given the fact that values of some other variables are known. In this case, the known states of variables can be entered as evidence in the network. When evidence is entered, this is likely to change the states of other variables as well, since they are conditionally dependent. This is demonstrated by entering the evidence in the network that the "Mode choice" variable is equal to "car". In this case, evidence on "Mode choice" now arrives in the form of P^*(Mode Choice)=(0, 1), where P^* indicates that we are no longer calculating prior probabilities. Then P^*(Choice, Gender, Number of cars, Driving License)= P(Number of cars, Gender, Driving License | Mode choice) * P^*(Mode Choice)= (P(Choice, Gender, Number of cars, Driving License)*P^*(Mode Choice))/P(Mode Choice). This means that the joint probability table for "Choice", "Number of cars", "Driving License" and "Gender" is updated by multiplying by the new distributions and dividing by the old ones. The multiplication consists of

annihilating all entries with “Choice”=“bike”. The division by P(Mode Choice) only has an effect on entries with Mode Choice=”car”, so therefore the division is by P(Mode Choice=”car”). For this simple example, the calculations can be found in the lower part of table 1. The distributions P^*(Number of cars), P^*(Gender) and P^*(Driving License) are calculated through marginalization of P^*(Choice, Gender, Number of cars, Driving License). This means that P^*(Gender=male; Gender=female) = (0.765;0.235); P^*(Number of cars=1; Number of cars>1)= (0.255;0.745) and P^*(Driving License=yes; Driving License=no)= (0.522; 0.478) when evidence was entered that the “Mode choice” variable equals car. Obviously, the calculation of this example is simple, however, in real-life situations it is likely that conditionally dependent relationships between the “choice” variable and other variables exist as well, and as a result the evidence will propagate through the whole network. More information about efficient algorithms for propagation of evidence in Bayesian networks can be found in Pearl (1988) and in Jensen, Lauritzen and Olesen (1990).

4.4. Structural learning

Structural learning determines the dependence and independence of variables and suggests a direction of causation (or association), in other words, the position of the links in the network. Experts can provide the structure of the network using domain knowledge. However, the structure can also be extracted from empirical data. Especially the latter option offers important and interesting opportunities for transportation travel demand modelling because it enables one to visually identify which variable or combination of variables influences the target variable of interest. Structural learning can be divided into two categories: search & scoring methods and dependency analysis methods. Algorithms, belonging to the first category interpret the learning problem as a search for the structure that best fits the data. Different scoring criteria have been suggested to evaluate the structure, such as the Bayesian scoring method (Cooper and Herskovits, 1992; Heckerman, Geiger and Chickering, 1995) and minimum description length (Lam and Bacchus, 1994).

A Bayesian network is essentially a descriptive probabilistic graphical model that is potentially well suited for unsupervised learning. However, the technique can also be tuned so that it becomes suitable for supervised (or classification) learning, just like decision trees, neural networks or for instance support vector machines. A number of Bayesian network classifiers (eg. Naïve Bayes, Tree augmented Naïve Bayes, General Bayesian network) have been developed for this purpose

4.5. Bayesian network classifiers: problem formulation

While Bayesian network classifiers have proven to give accurate and good results in a transportation context (Janssens, *et al*. 2004; Torres and Huber, 2003), Achilles' tendons obviously are the decision rules which can be derived from the Bayesian network. As mentioned above, each decision rule that is derived from the network contains the same number of conditions, resulting in potential sub-optimal decision-making. Second, Bayesian networks link more variables in sometimes complex, direct and indirect ways, making interpretation more problematic.

To illustrate this, the procedure of transforming a Bayesian network into a decision table (i.e. rule based form) is shown in the second part of figure 4. In this figure, probability distributions of the target variable are calculated for every possible combination of states. This can be done by entering evidences for those states in the network (see section 4.3). An example is shown in the middle part of Figure 4.

<INSERT FIGURE 4>

As it can be seen from this figure, every rule contains the same number of condition variables. For the example shown here, this number is equal to 4. Moreover, the number of rules that are derived from the network is fixed and can be determined in advance for a particular network. This number is equal to every possible combination of states (values of the condition variables). Therefore, the total number of rules, which has to be derived from the network shown in Figure 4 is equal to 2*3*2*7=84, assuming that the duration attribute is taken as the class attribute. Especially when more nodes are incorporated, this number is likely to become extremely large. While this does not need to be a problem as such, it is obvious that a number of these decision rules will be redundant as they will never be "fired". This flaw has no influence on the total accuracy of each Bayesian network classifier (see Janssens, *et al*. 2004), but it is clearly a sub-optimal solution, not only because some of the rules will never be used, but also because this large number of conditions do not favour the interpretation.

For both reasons mentioned here, i.e. the possibility of combining the advantage of Bayesian networks (take into account the interdependencies among variables) and the advantage of decision trees (derive easy understandable and flexible (i.e. non-fixed) decision rules), and for the reason mentioned before, i.e. deal with the variable masking problem in decision trees, the idea to integrate both techniques into a new classifier was conceived.

4.6. Towards a new integrated classifier

In the integrated BNT classifier, the idea is proposed to derive a decision tree from a Bayesian network (that is build upon the original data) instead of immediately deriving the tree from the original data. By doing so, it is expected that the structure of the tree is more stable, especially because the variable correlations are already taken into account in the Bayesian network, which may reduce the variable masking problem. To the best of our knowledge, the idea to build decision trees in this way has not been explored before in previous studies.

In order to select a particular decision node in the BNT classifier, the mutual information value that is calculated between two nodes in the Bayesian network is used. This mutual information value is to some extent equivalent with the entropy measure that C4.5 decision trees use. It is defined as the expected entropy reduction of one node due to a finding (observation) related to the other node. The dependent variable is called the query variable (denoted by the symbol *Q*), the independent variables are called findings variables (denoted by the symbol *F*). Therefore, the expected reduction in entropy (measured in bits) of *Q* due to a finding related to *F* can be calculated according the following equation (Pearl, *et al.*, 1988):

$$I(Q,F) = \sum_{q}\sum_{f} p(q,f) * \log\left(\frac{p(q,f)}{p(q)p(f)}\right) \qquad (1)$$

where, p(*q*,*f*) is the posterior probability that a particular state of *Q* (*q*) and a particular state of *F* (*f*) occur together; p(*q*) is the prior probability that a state *q* of *Q* will occur and p(*f*) is the prior probability that a state *f* of *F* will occur. The probabilities are summed across all states of *Q* and across all states of *F*.

The expected reduction in entropy of the dependent variable can be calculated for the various findings variables. The finding variable that obtains the highest reduction in entropy is selected as the root node in the tree. To better illustrate the idea of building a BNT classifier, we consider again the network that was shown in figure 3 by means of example. In this case, the dependent variable is “Mode choice” and the different finding variables are “Driving license”, “Gender” and “Number of cars”. In a first step, we can for instance calculate the expected reduction in entropy between the “Mode choice” and the “Gender” variable.

$$I = P(Mode_{bike},Gender_{male})*\log\frac{P(Mode_{bike},Gender_{male})}{P(Mode_{bike})P(Gender_{male})} +$$

$$P(Mode_{car},Gender_{male})*\log\frac{P(Mode_{car},Gender_{male})}{P(Mode_{car})P(Gender_{male})} +$$

$$P(Mode_{bike},Gender_{female})*\log\frac{P(Mode_{bike},Gender_{female})}{P(Mode_{bike})P(Gender_{female})} +$$

$$P(Mode_{car},Gender_{female})*\log\frac{P(Mode_{car},Gender_{female})}{P(Mode_{car})P(Gender_{female})}$$

The calculation of the joint probabilities $P(Mode_i, Gender_j)$ for i={bike, car} and j={male,female} is completely the same as explained in section 4.2. The calculation of the individual prior probabilities $P(Mode_i)$ and $P(Gender_j)$ is straightforward as well (see section 4.2). As a result, the expected result of formula (1) is: I (Mode choice,Gender) $= 0.372*\log\frac{0.372}{0.506*0.75} + 0.378*\log\frac{0.378}{0.494*0.75} + 0.134*\log\frac{0.134}{0.506*0.25} + 0.116*\log\frac{0.116}{0.494*0.25} = 0.00087$. In a similar way, I (Mode choice, Driving License) = 0.01781 and I (Mode choice, Number of cars)=0.01346 can be calculated. Since I (Mode choice,Driving License) > I (Mode choice, Number of cars) > I (Mode choice, Gender); the variable Driving License is selected as the root node of the tree (see figure 5). Once the root node has been determined, the tree is split up into different branches according to the different states (values) of the root node. To this end, evidences can be entered for each state of the root node in the Bayesian network and the entropy value can be re-calculated for all other combinations between the findings nodes (except for the root node) and the query node. The node, which achieves the highest entropy reduction is taken as the node which is used for splitting up that particular branch of the root node. In our example, the root node "Driving License" has two branches: Driving License=yes and Driving License=no. For the split in the first branch (Driving License=yes), only two variables have to be taken into account (since the root node is excluded): "Number of cars" and "Gender". The way in which the expected reduction in entropy is calculated is the same as shown above, expect for the fact that an evidence needs to be entered for the node "Driving License", i.e. P(Driving License=Yes; Driving License=no)=(1;0) (since we are in the first branch). The procedure for doing this was already described in section 4.3. Again, I (Mode choice, Gender)=0.02282 and I (Mode choice, Number of cars)=0.07630. Since I(Mode choice, Number of cars)>I((Mode choice, Gender); the variable "Number of cars" is selected as the next split in this first branch. Finally, the whole process then becomes recursive and needs to be repeated for all possible branches in the tree. A computer code has been

established to automate the whole process. The final decision tree for this simple Bayesian network is shown in figure 5.

<INSERT FIGURE 5 HERE>

5. DATA AND DESIGN OF THE EXPERIMENTS

5.1. Data

The activity diary data used in this study were collected in the municipalities of Hendrik-Ido-Ambacht and Zwijndrecht in the Netherlands (South Rotterdam region) to develop the Albatross model system (Arentze and Timmermans 2000). The data involve a full activity diary, implying that both in-home and out-of-home activities were reported. The sample covered all seven days of the week, but individual respondents were requested to complete the diaries for two designated consecutive days. Respondents were asked, for each successive activity, to provide information about the nature of the activity, the day, start and end time, the location where the activity took place, the transport mode, the travel time, accompanying individuals and whether the activity was planned or not. A pre-coded scheme was used for activity reporting. After cleaning, a data set of a random sample of 1649 respondents was used in the experiments.

There are some general variables that are used for each choice facet of the Albatross model (i.e. each oval box). These include (among others) household and person characteristics that might be relevant for the segmentation of the sample. Each dimension also has its own extensive list of more specific variables, which are not described here in detail.

5.2 Design of The Experiments

The aim of this study is to examine both the predictive capabilities and the potential advantages of the BNT classifier. To this end, the predictive performance of this integration technique is compared with a decision tree learning algorithm (CHAID) and with original Bayesian network learning.

For the CHAID decision tree approach, experiments were conducted for the full set of decision agents of the Albatross system. First, decision trees were therefore extracted from activity-travel diaries. Hereafter, these decision trees were converted into decision tables as described in section 3.1. Next, the decision tables were successively executed to predict the activity-travel patterns for the randomly selected sample of 1649 respondents.

For the Bayesian network approach, a Bayesian network was constructed for every decision agent using a structural learning algorithm, developed by Cheng, *et al*. (1997). This implies that the structure of the network was not imposed on the basis of a-priori domain knowledge, but was learned from the data. The structural learning algorithm was also enhanced by adding a pruning stage. This pruning stage aims at reducing the size of the network without resulting in any loss of relevant information or loss of accuracy. This means that nodes, which are not valuable for decision-making, are pruned away. In order to decide which nodes in the network are suitable for pruning, the reduction in entropy between two nodes was calculated using equation (1), shown in section 4.6. Obviously, a huge entropy reduction indicates a potentially important and useful node in the network. An entropy reduction of less than 0.05 bits was used as a threshold to prune the network. Once the pruned network is constructed for every decision agent, the model can be used for prediction. To this end, probability distributions of all the variables in the networks have to be computed. A parameter learning algorithm developed by Lauritzen (1995) was used to calculate these probability distributions. An example of such a distribution can be seen in the left part of Figure 4, where each state in the network is shown with its belief level (probability) expressed as a percentage and a bar chart. The last step is to transform the predictive model to the decision table formalism. The approach for doing so was already described in section 4.5.

For building the BNT classifier, a decision tree is not derived directly from the original data, but from the Bayesian networks that are built in the previous step. The procedure for doing this was explained in section 4.6, while the approach in section 3.1 can be used here as well for converting the tree into a decision table format. Once again, the decision tables are then sequentially executed to predict activity-travel patterns.

In the next section, we report the results of detailed quantitative analyses that were conducted to evaluate the BNT classifier for every decision agent in the Albatross model. The results of the three alternative approaches are validated in terms of accuracy percentages. The techniques are compared at both the activity pattern level and the trip level.

6. RESULTS

6.1 Model Comparison: Accuracy Results

To be able to test the validity of the presented models on a holdout sample, only a subset of the cases is used to build the models (i.e., “training set”). The decline in goodness-of-fit between this “training” set and the

validation set is taken as an indicator of the degree of overfitting. The purpose of the validation test is also to evaluate the predictive ability of the three techniques for a new set of cases. For each decision step, we used a random sample of 75% of the cases to build and optimise the models. The other subset of 25% of the cases was presented as "unseen" data to the models; this part of the data was used as the validation set. The accuracy percentages that indicate the predictive performance of the three models on the training and test sets are presented in Table 2.

It can be seen that the accuracy percentages of the BNT classifier are comparable with the accuracy results of the Bayesian network approach. This should not be surprising of course, because Bayesian networks were used as the underlying structure of the decision trees. More important is the observation that the newly proposed BNT classifier outperformed the CHAID decision trees for all nine decision agents of the Albatross model. This means that by using Bayesian networks as the underlying structure for building decision trees, better results can be obtained than using a traditional CHAID based decision tree approach. In terms of validity of the three models, we can conclude that the degree of over-fitting (i.e., the difference between the training and the validation set) is low for all decision agents. Therefore, we conclude that the transferability of the models to a new set of cases is satisfactory. The next section examines the results of the models at the pattern level.

<INSERT TABLE 2 HERE>

6.2. Activity Pattern Level Analysis

As explained before, the set of decision tables which is derived for each model will predict activity schedules, assuming the sequential execution of the decision tables, depicted in Figure 1. At the activity pattern level, sequence alignment methods (SAM) (Joh, *et al.*, 2001a) were used to calculate the similarity between observed and generated activity schedules. This measure allows users to evaluate the goodness-of-fit. SAM originally stems from work in molecular biology to measure the biological distance between DNA and RNA strings. Later, it was used in time use research (Wilson, 1998). To account for differences in composition as well as sequential order of elements, SAM determines the minimum effort required to make two strings identical using insertion, deletion and substitution operations.

The mean SAM distances between the observed and the predicted schedules are shown in Table 3. SAM distances were separately calculated for the qualitative activity pattern attributes (activity type, with-whom, location and mode). Also, both "UDSAM" and "MDSAM" measures were calculated. UDSAM represents a weighted sum of attribute SAM values, where activity type was given a weight of two units and

the other attributes a weight of one unit. To account for the multidimensionality, which is incorporated in the Albatross model, the MDSAM measure (Joh, *et al.*, 2001b) was used. The lower the SAM measure, the higher the degree of similarity between observed and predicted activity sequences.

<INSERT TABLE 3 HERE>

Obviously, the number of decision rules, which are derived from the Bayesian network is much larger than the number of rules, which are extracted from both decision trees. This difference should be attributed to the different nature of the technique; the number of variables that are incorporated is the same. This larger number of rules has no negative impact on performance; in fact, Table 3 provides evidence on the contrary. However, as said before, Bayesian networks likely suffer from (i) suboptimal decision making because many rules may never be used and (ii) interpretation difficulties. The integration approach developed in this study aimed at solving both potential problems. It can be seen from table 3 that the decision tree using the Bayesian network as the underlying structure, performed better than the state-of-the-art CHAID decision tree approach. Unfortunately, however, compared to Bayesian networks, some of the very good performance is lost. However, the rules which are derived from the integrated decision tree are better understandable and do not contain a fixed number of variables in the conditions of the rules.

Thus, it seems that the problem can be reduced to a trade-off between model complexity and accuracy. Since the CHAID decision tree approach and the BNT classifier are comparable in terms of model complexity (both approaches do contain more or less the same number of decision rules), the BNT classifier is a better way of predicting activity schedules when a pattern level performance measure is used for validation.

6.3. Trip Matrix Level Analysis

The last measure to evaluate the predictive performance is calculated at the trip level. The origins and destinations of each trip, derived from the activity patterns, are used to build OD-matrices. The origin locations are represented in the rows of the matrix and the destination locations in the columns. The number of trips from a certain origin to a certain location is used as a matrix entry. A third dimension was added to the matrix to break down the interactions according to some third variable. The third dimensions considered are day of the week, transport mode and primary activity. The bi-dimensional case (no third dimension) was considered as well. In order to determine the degree of correspondence between predicted and observed matrices, a correlation coefficient was calculated. To this end, cells of the matrix were rearranged into one

array and the calculation of the correlation is based on comparing the corresponding elements of the predicted and the observed array. The results are presented in Table 4.

It can be seen from Table 4 that the Bayesian network model generates higher correlation coefficients between observed and predicted OD matrices than the original CHAID-based Albatross model and the BNT classifier. Unfortunately, the good performance of BNT at the activity pattern level could not be maintained at trip level. The correlation coefficient is especially low for the OD matrix, where the primary activity is taken as the third dimension. Although this should be the subject of additional and future research, it is believed that the integrated approach did predict less activities in the activity schedule than both CHAID and Bayesian networks. This deficiency is less apparent at the pattern level than at the trip level. The smaller number of rules, which are used in the integrated decision tree may be responsible for this.

<INSERT TABLE 4 HERE>

7. CONCLUSION AND DISCUSSION OF THE RESULTS

Several activity-based models are nowadays becoming operational and are entering the stage of application in transport planning. Some of these models (like Albatross) rely on a set of decision rules that are derived from activity-travel diary data rather than on principles of utility maximization. While the use of rules may have some theoretical advantages, the performance of several rule induction algorithms in models of activity-travel behavior is not well understood. This is unfortunate because there is some empirical evidence that decision tree induction algorithms are relatively sensitive to random fluctuations in the data. To add to the growing literature on the performance of alternate decision-tree induction algorithms, we proposed in this paper a way of combining Bayesian networks and decision trees. The idea was motivated by the results of a previous study (Janssens, *et al.*,2004) which suggested that Bayesian networks outperformed decision trees but that model complexity also increased significantly along with the increase in accuracy. For this reason, this study was designed to examine whether a decision tree (which is implicitly always less complex) that uses the structure of a Bayesian network (referred to as Bayesian network augmented tree classifier, BNT) to select its decision nodes can achieve simultaneously accuracy results comparable to Bayesian networks and an easier and less complex model structure, comparable to traditional decision trees.

In order to test the validity and the transferability to a new set of cases of the proposed approach, datasets were split up into training and validation sets. The predictive performance of the new approach was evaluated at three different levels. The test has shown that the BNT approach indeed achieved comparably

good accuracy results than Bayesian networks and along with the Bayesian networks outperformed CHAID decision trees for all decision agents of the Albatross model. Moreover, the results showed that the decrease in model complexity of the BNT classifier also led to a decrease in performance at the activity pattern level in comparison with Bayesian networks. However, when the BNT approach was compared to the equally complex CHAID decision tree, BNT outperformed CHAID at pattern level.

Finally, at the third level of validation, the trip matrix level, correlation coefficients between observed and predicted origin-destination matrices showed that Bayesian networks outperform both CHAID and BNT. Thus, the good results of the integrated decision tree were not maintained at the trip level. It is believed that the technique did predict fewer activities in the activity schedule than both CHAID and Bayesian networks. While the smaller number rules that are used may be responsible for this, this finding should be the subject of additional research.

In summary then, this study has shown some interesting results. There are some initial indications that the new way of integrating decision trees and Bayesian networks may produce a decision tree that is structurally more stable and less vulnerable to the variable masking problem. Additionally, the results at the activity level and trip level suggest at least for the Albatross data, a trade-off between model accuracy and model complexity. When the main issue is the interpretation and the general understanding of the decision rules, the integrated BNT approach may be favoured above CHAID decision trees when decisions need to be made at pattern level. At a more detailed level, one may benefit from the use of the CHAID approach. However, when the main issue is model accuracy, Bayesian networks should be favoured. Additional and further research should examine the behavior of the three techniques under evaluation on other data sets.

REFERENCES

Arentze, T.A., Timmermans, H.J.P., 2000. Albatross: A Learning-Based Transportation Oriented Simulation System. European Institute of Retailing and Services Studies. Eindhoven, The Netherlands.

Ben-Akiva, M., Bowman, J., Ramming, S., Walker, J., 1998. Behavioral realism in urban transportation planning models. In Transportation Models in the Policy-Making Process: A Symposium in Memory of Greig Harvey, Asilomar Conference Center, California.

Bhat, C.R., Guo, J., Srinivaasan, S., Sivakumar, A., 2004. Comprehensive Econometric Microsimulator for Daily Activity-Travel Patterns, In: Electronic conference proceedings of the 83rd Annual Meeting of the Transportation Research Board (CD-ROM), January 11-15, Washington, D.C.

Bloemer, J.M.M., Brijs, T., Swinnen, G., Vanhoof, K., 2003. Comparing complete and partial classification for identifying customers at risk, International Journal of Research in Marketing 20 (2) 117-131.

Bowman, J.L., M. Bradley, Y. Shiftan, T.K. Lawton, Ben-Akiva, M.E, 1998. Demonstration of an activity-based model system for Portland. Paper presented at the 8th WCTR conference, Antwerp.

Cheng, J., Bell, D., Liu, W., 1997. Learning Bayesian networks from data: an efficient approach based on information theory. In Proceedings of the sixth ACM International Conference on Information and Knowledge Management.

Cooper, G.F., Herskovits, E., 1992. A Bayesian Method for the induction of probabilistic networks from data, Machine Learning 9 309-347.

Heckerman, D., Geiger, D., Chickering, D.M., 1995. Learning Bayesian networks: the combination of knowledge and statistical data, Machine Learning 20 197-243.

Janssens, D., Wets, G., Brijs, T., Vanhoof, K., Timmermans, H.J.P., Arentze, T.A., 2004. Improving the Performance of a Multi-Agent Rule-Based Model for Activity Pattern Decisions Using Bayesian Networks. Forthcoming in Journal of the Transportation Research board, also in: Electronic conference proceedings of the 83rd Annual Meeting of the Transportation Research Board (CD-ROM)

Jensen, F.V., Lauritzen, S.L., Olesen, K.G., 1990. Bayesian updating in causal probabilistic networks by local computations, Computational Statistics Quarterly 4 269-282.

Joh, C.-H, Arentze, T.A., Timmermans, H.J.P, 2001(a). A position-sensitive sequence alignment method illustrated for space-time activity diary data, Environment and Planning A 33 313-338.

Joh, C.-H, Arentze, T.A., Hofman, F., Timmermans, H.J.P., 2001(b). Activity pattern similarity: a multidimensional sequence alignment method, Transportation Research B 36 385-403.

Kass, G.V., 1980. An Exploratory Technique for Investigating Large Quantities of Categorical Data, Applied Statistics 29 119-127.

Lam, W., Bacchus, F., 1994. Learning Bayesian belief networks: An approach based on the MDL principle, Computational Intelligence, 10 (4) 269-293.

Lauritzen, S.L., 1995. The EM algorithm for graphical association models with missing data, Computational Statistics & Data Analysis 19 191-201.

Moons, E. A. L. M. G, Wets, G., Aerts, M., Arentze, T.A. and Timmermans, H.J.P, 2004. The Impact of Simplification in a Sequential Rule-Based Model of Activity Scheduling Behavior, Forthcoming in Environment & Planning A.

Pearl, J., 1988. Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kaufmann Publishers, Palo Alto.

Torres, FJ., Huber, M., 2003. Learning a Causal Model from Household Survey Data Using a Bayesian Belief Network, In: Electronic conference proceedings of the 82nd Annual Meeting of the Transportation Research Board, Washington D.C

Wets, G., Vanhoof, K., Arentze, T.A., Timmermans, H.J.P., 2000. Identifying Decision Structures Underlying Activity Patterns: An Exploration of Data Mining Algorithms, Transportation Research Record 1718 1-9.

Wilson, C., 1998. Activity pattern analysis by means of sequence-alignment methods, Environment and Planning 30A 1017-1038.

List of Tables and Figures

Table 1 Prior and Posterior Probability Tables for the Transport Mode Choice Variable

Conditional Prior Probability Table specifying P(Choice|Gender, Driving License, Ncar)

Gender	Male				Female			
Driving License	Yes		No		Yes		No	
Number of cars	1	>1	1	>1	1	>1	1	>1
Mode Choice bike	0.2	0.6	0.7	0.4	0.4	0.8	0.1	0.3
Mode Choice car	0.8	0.4	0.3	0.6	0.6	0.2	0.9	0.7

Joint Prior Probability Table for P(Choice,Gender,Ncar, Driving License)

Gender	Male				Female			
Driving License	Yes		No		Yes		No	
Number of cars	1	>1	1	>1	1	>1	1	>1
Mode Choice bike	0.018	0.216	0.042	0.096	0.012	0.096	0.002	0.024
Mode Choice car	0.072	0.144	0.018	0.144	0.018	0.024	0.018	0.056

The Calculation of P^*(Choice, Gender, Ncar,Driving License) = P(Choice, Gender,Ncar, Driving Licence|Mode Choice=car)

Gender	Male				Female			
Driving License	Yes		No		Yes		No	
Number of cars	1	>1	1	>1	1	>1	1	>1
Mode Choice bike	0	0	0	0	0	0	0	0
Mode Choice car	0.146	0.291	0.036	0.291	0.036	0.049	0.036	0.113

Table 2 Comparison of accuracy percentages for CHAID decision trees, Bayesian networks and integrated BNT classifier

Decision Agent	Decision making based on CHAID decision trees		Decision making based on Bayesian networks		Decision making based on integrated BNT classifier	
	Training Set	Validation Set	Training Set	Validation Set	Training Set	Validation Set
Selection	0.724	0.716	0.791	0.792	0.790	0.791
With Whom	0.509	0.484	0.577	0.534	0.577	0.535
Duration	0.413	0.388	0.409	0.405	0.410	0.402
Start time	0.398	0.354	0.477	0.380	0.423	0.393
Trip Chain	0.833	0.809	0.831	0.823	0.831	0.825
Mode for work	0.648	0.667	0.769	0.779	0.770	0.783
Mode other	0.528	0.495	0.583	0.521	0.583	0.521
Location 1	0.575	0.589	0.696	0.679	0.694	0.685
Location 2	0.354	0.326	0.473	0.420	0.473	0.419
Average	*0.553*	*0.536*	*0.622*	*0.592*	*0.617*	*0.595*

Table 3 SAM distance measures for activity pattern level analysis

SAM distance measure	Decision making based on CHAID decision trees	Decision making based on Bayesian networks	Decision making based on integrated BNT classifier
SAM activity-type	2.86	0.061	2.151
SAM with	3.225	0.062	2.6
SAM location	3.181	2.12	2.144
SAM mode	4.599	0.063	3.784
UDSAM	16.725	2.366	12.83
MDSAM	8.457	1.584	6.25

Table 4 Correlation coefficients between OD-matrices for trip level analysis

	Decision making based on CHAID decision trees	Decision making based on Bayesian networks	Decision making based on integrated BNT classifier
None	0.953814	0.963992	0.9542192
Mode	0.876937	0.954744	0.8495603
Day	0.960293	0.970894	0.9504108
Primary Activity	0.889740	0.935173	0.8281859

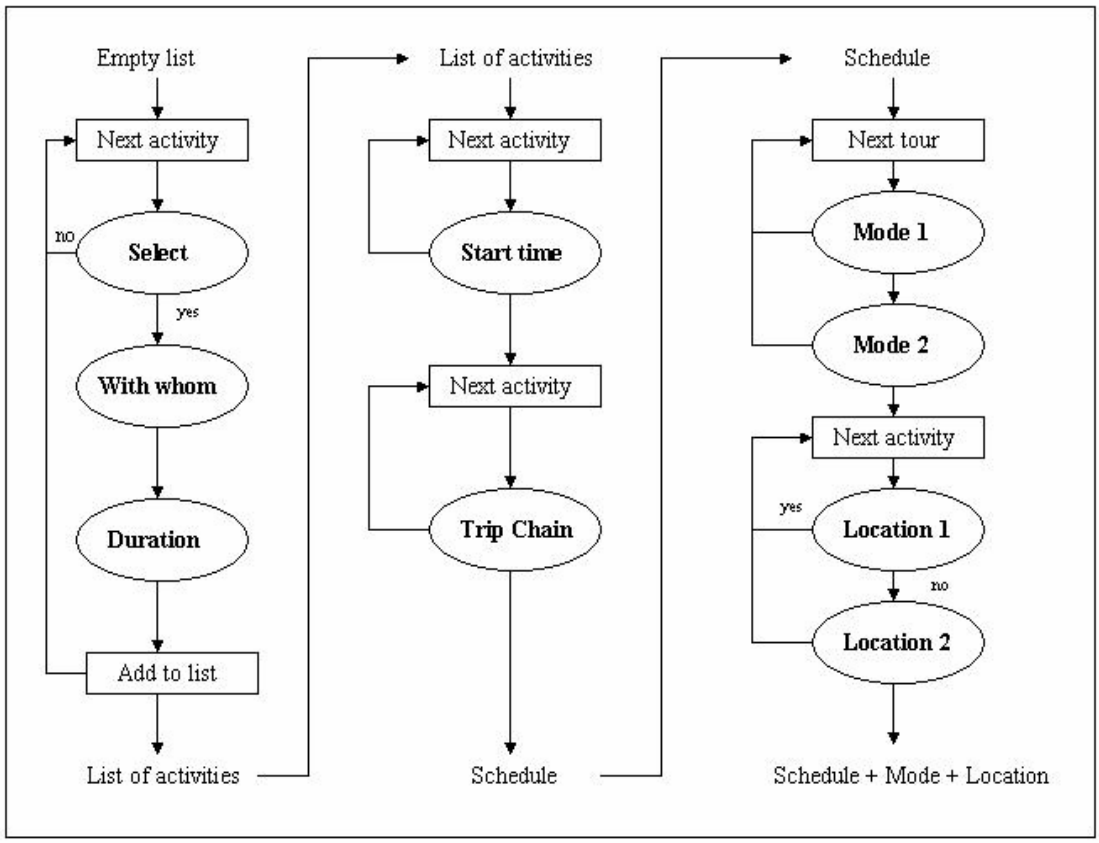
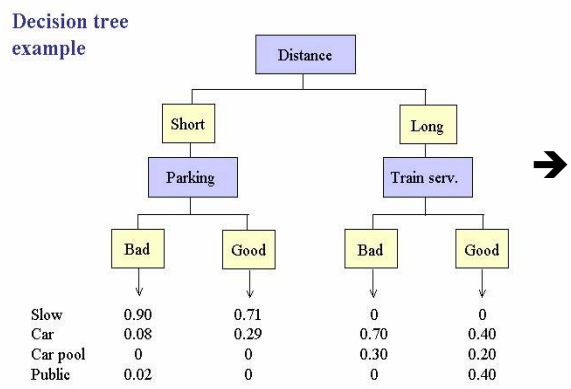


FIGURE 1 Albatross' scheduling engine.

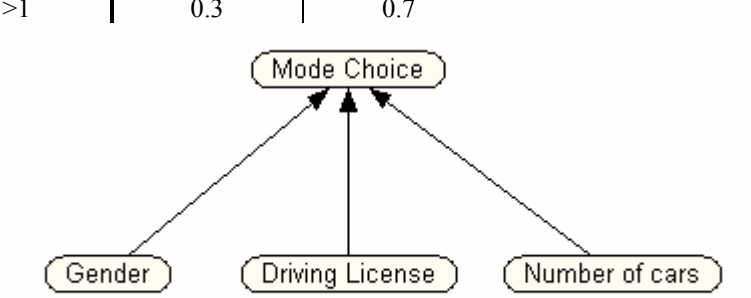


Distance	Short	Short	Long	Long
Parking	Bad	Good	-	-
Train Serv.	-	-	Bad	Good
Transport mode : Slow	0.90	0.71	0	0
Transport mode : Car	0.08	0.29	0.70	0.40
Transport mode : Car pool	0	0	0.30	0.20
Transport mode : Public	0.02	0	0	0.40

FIGURE 2 Decision tree example with its corresponding decision table.

Gender	Driving License	Number of cars	*P* (mode choice = bike)	*P* (mode hoice = car)
Male	Yes	1	0.2	0.8
Male	Yes	>1	0.6	0.4
Male	No	1	0.7	0.3
Male	No	>1	0.4	0.6
Female	Yes	1	0.4	0.6
Female	Yes	>1	0.8	0.2
Female	No	1	0.1	0.9
Female	No	>1	0.3	0.7

CPT Mode Choice



P(Gender = male)	*P* (Gender = female)
0.75	0.25

CPT Gender

P (Driving License=Yes)	*P* (Driving License=No)
0.6	0.4

CPT Driving License

P (Number of cars=1)	*P* (Number of cars>1)
0.2	0.8

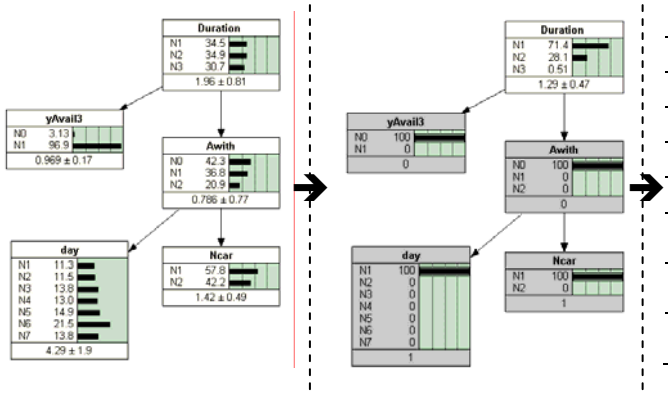
CPT Number of Cars

FIGURE 3 A small Bayesian network with its CPT.

The pruned network

Entering evidences (for every combination)

A decision table with probability distributions



yAvail3	N0	N0	N0	N1	…	N1
Awith	N0	N1	N2	N0	…	N2
Day	N1	N1	N1	N1	…	N7
Ncar	N1	N1	N1	N1	…	N2
Duration: N1	71.4%	56.4%	58.9%	43.7%	…	25.9%
Duration: N2	28.1%	42.6%	39.9%	32.9%	…	33.5%
Duration: N3	0.51%	1.03%	1.24%	23.4%	…	40.6%

FIGURE 4 Calculating probability distributions and entering them in a decision table.

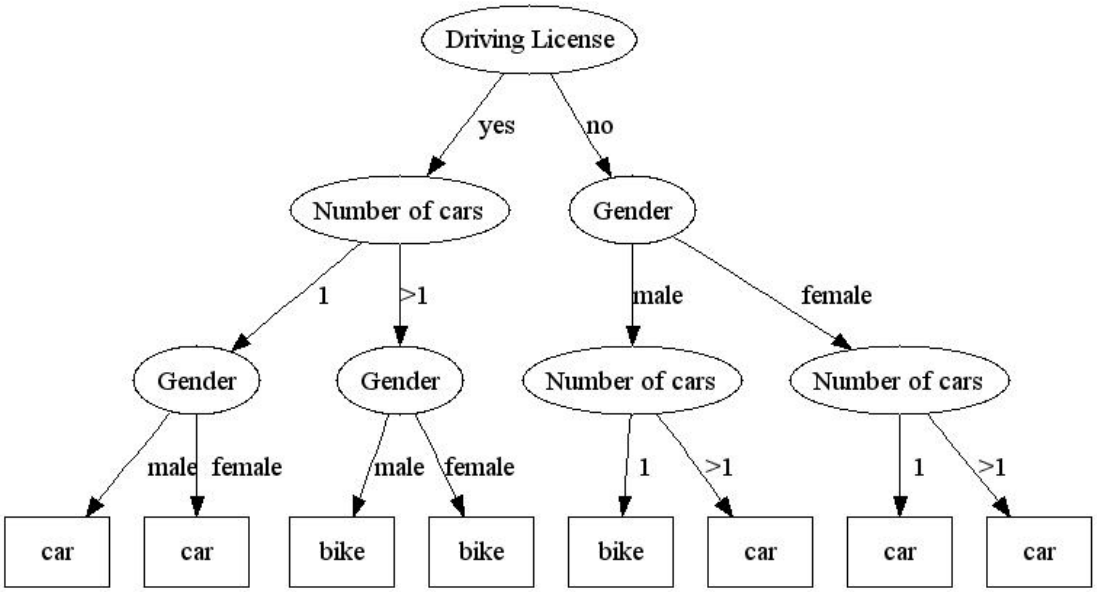


FIGURE 5 The final integrated BNT decision tree classifier (example).