Evaluation of a Vehicle Acceleration Behavior through Decision Tree Learning

İhsan Ömür Bucak
Fatih University, Engineering Faculty,
Computer Engineering Department,
34500 Buyukcekmece, Istanbul, Turkiye
ibucak@fatih.edu.tr

Abstract: The faster that a motor vehicle can accelerate to a high velocity is crucial to its performance and handling. The acceleration of the vehicle is important to know because it tells us how the car handles during merging and evasive maneuvering. Decision trees are powerful and popular tools for classification and prediction. The attractiveness of decision trees is due to the fact that, in contrast to neural networks, decision trees represent rules. Rules can readily be expressed so that humans can understand them after a brief explanation. Therefore, the objective of this paper is to develop a systematic method using decision trees of machine learning to evaluate acceleration behavior of motor vehicles based on the forces acting on the vehicle, i.e. vehicle dynamics.

Keywords: Vehicle Acceleration, Vehicle Dynamics, Decision Tree Learning, Machine Learning.

Introduction

A vehicle can go faster if it has more horsepower. In reality, there are many aspects to a vehicle acceleration besides it's horsepower. One other major aspect is vehicle weight. If the vehicle weight is lowered, its acceleration, braking, and handling capabilites will be increased. Given the same power and adequate traction, a light vehicle will accelerate quicker than a heavier vehicle. Figure 1 shows such an acceleration performance, specifically 0-60 mph (=26.66 m/s) passing time, for various vehicle weights from 1100 kg to 1700 kg with all other parameters unchanged on a simple vehicle dynamics model realized on Matlab-Simulink (Matlab, 2008). A very quick street or race car usually combines excellent power with less weight.

The most current vehicle dynamics controllers attempt to ensure stability by keeping lateral acceleration, sometimes, longitudinal acceleration, and yaw within reasonable bounds (Bauer, 1999a; 1999b; Jurgen, 1999; Karri and Butler, 2002). The faster that a vehicle can accelerate to a high velocity is crucial to its performance and handling. The acceleration of a vehicle is important to know because it tells us how the vehicle performs during merging and evasive maneuvering. The launching performances are the acceleration performance with various throttle positions and the transient characteristics of vehicle creep and throttle tip-in. Every time a new or redesigned performance car enters the marketplace, it is accompanied by a number; specifically, the number of seconds it takes to reach 60 miles per hour (26.66 meters per second) from a standing start. So many drivers want to push that pedal all the way to the metal, as much of the time as possible; because they do not want to have trouble merging into an expressway or take endless seconds to pass another vehicle on a two-lane road. Weak acceleration is an issue that needs to be addressed when minicars, including those powered by batteries, begin to emerge into the world market. What matters is not the ability to reach 60 miles per hour in a few seconds. All that counts is the ability to accelerate at midrange speeds: from 30 to 50 mph, or 50 to 70 mph. That is where energetic acceleration has a valid purpose, and is essential for safe motoring. Charts display acceleration times not only from 0 to 60 mph, but for a selection of useful speed ranges. Road tests include a broad set of timed acceleration runs.

Excess acceleration and deceleration need to be detected and mitigated within the required response time particularly in the case of hybrid electric vehicles (HEVs). There are a variety of simulation programs developed to investigate the launching performance in various launching conditions through the use of mathematical models of each driveline component (Kim, 2005). Automotive manufacturers use such performance programs to evaluate their vehicles during product design and development stages so that they can meet the stringent govermental regulations on performance, fuel economy, and emissions before the vehicle is actually launched to the marketplace. Practical evaluation of acceleration behavior through the rulistic expression of decision trees helps to expose the major factors that affect the performance-related design parameters so that the redesign can be made more productively and effectively in order to prevent vehicle recalls and/or customer dissatisfaction.

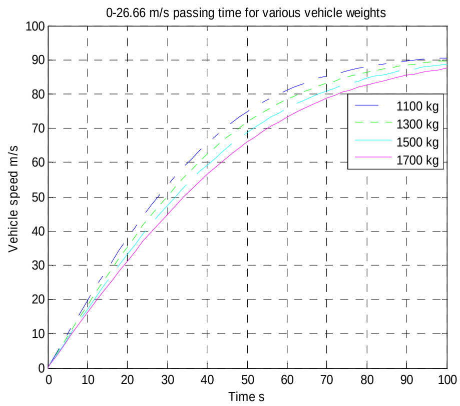

Figure 1. Acceleration performance for various vehicle weights.

This paper is based on decision tree learning used to evaluate vehicle dynamic parameters for describing an accelerating vehicle behavior.

The remaining of the paper is arranged as follows. Section 2 briefly explains the basics of the decision tree learning including the definition, the construction, attributes as classifiers, entropy and information gain. Section 3 describes the method. Section 4 presents the numerical experiments and simulations. Finally, conclusions are drawn in Section 5.

Decision Tree Learning

Decision trees are powerful and popular tools for classification and prediction. One of the several advantages of decision trees is that they are simple to understand and interpret. This is mainly due to the fact that, in contrast to neural networks, decision trees represent rules. These rules can readily be expressed so that people can understand them after a brief explanation (Gamberger and Smuc, 2001). Decision tree learning, used in data mining and machine learning, uses a decision tree as a predictive model which maps observations about an item to conclusions about the item's target value. More descriptive names for such tree models are classification tree (discrete outcome) or regression tree (continuous outcome). In these tree structures, leaves represent classifications and branches represent conjunctions of features that lead to those classifications (Breiman et al., 1984; Yuan and Shaw, 1995; Mitchell, 1997; Berikov and Litvinenko, 2003; Menzies and Hu, 2003, Wikipedia, 2009).

Decision tree is a classifier in the form of a tree structure, where each node is either (Gamberger and Smuc, 2001):

- a leaf node - indicates the value of the target attribute (class) of examples, or
- a decision node - specifies some test to be carried out on a single attribute-value, with one branch and sub-tree for each possible outcome of the test.

A decision tree can be used to classify an example by starting at the root of the tree and moving through it until a leaf node, which provides the classification of the instance. Most algorithms that have been developed for learning decision trees are variations on a core algorithm that employs a top-down, greedy search through the space of possible decision trees. Decision tree programs construct a decision tree from a set of training cases (Gamberger and Smuc, 2001).

The estimation criterion in the decision tree algorithm is the selection of an attribute to test at each decision node in the tree. The goal is to select the attribute that is most useful for classifying examples. A good quantitative measure of the worth of an attribute is a statistical property called information gain that measures how well a given attribute separates the training examples according to their target classification. This measure is used to select among the candidate attributes at each step while growing the tree. In order to define information gain precisely, we need to define a measure commonly used in information theory, called entropy, that characterizes the (im)purity of an arbitrary collection of examples. Given a set S, containing only positive and negative examples of some target concept (a 2 class problem), the entropy of set S relative to this simple, binary classification is defined as:

$$Entropy(s) = -p_p \log_2 p_p - p_n \log_2 p_n \,, \qquad (1)$$

where p_p is the proportion of positive examples in S and p_n is the proportion of negative examples in S (Mitchell 1997; Gamberger and Smuc, 2001).

One interpretation of entropy from information theory is that it specifies the minimum number of bits of information needed to encode the classification of an arbitrary member of S (i.e., a member of S drawn at random with uniform probability). For example, if p_p is 1, the receiver knows the drawn example will be positive, so no message need be sent, and the entropy is 0. On the other hand, if p_p is 0.5, one bit is required to indicate whether the drawn example is positive or negative. If p_p is 0.8, then a collection of messages can be encoded using on average less than 1 bit per message by assigning shorter codes to collections of positive examples and longer codes to less likely negative examples (Mitchell 1997; Gamberger and Smuc, 2001).

If the target attribute takes on c different values rather than the special case discussed above where the target classification takes on 2 different values, i.e., binary, then the entropy of S relative to this c-wise classification is defined as

$$Entropy(S) = \sum_{i=1}^{c} - p_i \log_2 p_i \ , \qquad (2)$$

where p_i is the proportion of S belonging to class i. Note the logarithm is still base 2 because entropy is a measure of the expected encoding length measured in bits. Note also that if the target attribute can take on c possible values, the maximum possible entropy is $\log_2 c$ (Mitchell 1997; Gamberger and Smuc, 2001).

Given entropy as a measure of the impurity in a collection of training examples, we can now define a measure of the effectiveness of an attribute in classifying the training data. The measure we will use, called information gain, is simply the expected reduction in entropy caused by partitioning the examples according to this attribute. More precisely, the information gain, Gain (S, A) of an attribute A, relative to a collection of examples S, is defined as

$$Gain(S,A) = Entropy(S) - \sum_{v \in Value(A)} \frac{|S_v|}{|S|} Entropy(S_v) \qquad (3)$$

where Values(A) is the set of all possible values for attribute A, and S_v is the subset of S for which attribute A has value v (i.e., S_v = {s $\in$ S | A(s) = v}). Note the first term in the equation for Gain is just the entropy of the original collection S and the second term is the expected value of the entropy after S is partitioned using attribute A. Gain (S,A) is therefore the expected reduction in entropy caused by knowing the value of attribute A. Put another way, Gain(S,A) is the information provided about the target attribute value, given the value of some other attribute A. The value of Gain(S,A) is the number of bits saved when encoding the target value of an arbitrary member of S, by knowing the value of attribute A (Mitchell 1997; Gamberger and Smuc, 2001).

The process of selecting a new attribute and partitioning the training examples is now repeated for each non-terminal descendant node, this time using only the training examples associated with that node. Attributes that have been incorporated higher in the tree are excluded, so that any given attribute can appear at most once along any path through the tree. This process continues for each new leaf node until either of two conditions is met:

1. every attribute has already been included along this path through the tree, or
2. the training examples associated with this leaf node all have the same target attribute value (i.e., their entropy is zero).

Practical issues in learning decision trees include determining how deeply to grow the decision tree, handling continuous attributes, choosing an appropriate attribute selection measure, handling training data with missing attribute values, handing attributes with differing costs, and improving computational efficiency. Overfitting is a significant practical difficulty for decision tree learning and many other learning methods. There are several approaches to avoiding overfitting in decision tree learning (Schaffer, 1991; Mitchell 1997; Gamberger and Smuc, 2001).

Method and Data

Vehicle dynamics describes the forces acting on the vehicle that result in its motion. Tractive effort and resistance are the two primary opposing forces that determine the performance characteristics of road vehicles. The engine in the vehicle supplies the tractive-effort force, and the magnitude of this force is restricted by internal friction losses. The difference between the tractive effort and the resulting resisting force is the net thrust F_{net} available for accelerating vehicle. The forces of air resistance, rolling resistance, grade resistance, and friction resistance oppose the engine tractive force and limit the acceleration capability of the vehicle (Wong, 1978; Snare, 2002).

Maximum tractive force, F_{total}, is the maximum amount of force that the engine can supply to the tires of the drive axles. Therefore, the maximum tractive force delivered to the tires of the drive axles in the chosen gear combination can be expressed as follows:

$$F_{total} = (T_m.T_r.R_r.R_n.Gr_x)/T_s \ , \qquad (4)$$

where T_m represents maximum torque (Nm), T_r transfer case or auxiliary gear box ratio, R_r final drive ratio,

R_n drive axles efficiency (%), Gr_1 gear ratio for the first gear (i.e., x=1), and T_s tire size (radius) (m).

Required pull force, F_{req}, is the force required to cause the vehicle to roll. Hence, the required pull force is calculated by the following equation:

$$F_{req} = overcome\ \ friction + Accelerate + Climb\ \ slope \text{ or,} \qquad (5)$$

$$F_{req} = \mu_R . W_g . cos(sl) + (W_g / g).a + W_g . sin(sl),$$

where μ_R represents coefficient of rolling friction, W_g gross vehicle weight (mTon), g gravitation (m/s^2), and sl maximum slope in route (degree).

In this paper acceleration behavior is evaluated based on the 37500 data, which are formed by the combination of the maximum torque, weight, gear ratio, final drive ratio, maximum slope and tire size variables. These variables, also known as attributes, are expressed in the form of one-dimensional arrays. The same number of elements of each attribute array represents a dynamic parameters of a unique vehicle. Therefore the number of different vehicle types in the evaluation process is determined by the size of the arrays, all in equal length. Drive axles efficiency is taken constant that is set at 90%. The efficiency slightly reduces due to internal friction although the absolute traction between tires and road surface increases. Coefficient of rolling friction is also considered constant, and is set at 0.03 valid for most ordinary car tires on asphalt pavements. The other constant parameter is transfer case. Transfer case or auxiliary gear is intended to select two wheel drive or four wheel drive operations and may contain one or more sets of low range gears. Low range gears slow down the vehicle and increase the torque available at the axles. Therefore, they are used during slow speed or extreme off road maneuvers. Although on all drive sports cars this feature is absent, we still consider the equations that include the transfer case ratio, but we decide to set the ratio 1:1 for all the vehicle types considered. However, transfer case ratio has had no effect on our conclusions when we set it at 2:1 for the purpose to exhibit and observe its role in describing the complete acceleration behavior. It is assumed that the first gear is engaged as the lowest gear and used as such in the entire calculations. The limits of acceleration variable for the evaluations have been determined through acceleration vs speed charts (Snare, 2002). The speed calculations have not taken into account any environmental forces such as wind or state (and incline) of the road.

Numerical Experiments and Simulations

In this study there are seven determined attributes used to evaluate the vehicle acceleration behavior. These attributes are maximum acceleration, weight, torque, gear ratio, final drive ratio, tire size, and slope. The attributes are used to calculate the leaf and decision nodes, and the branches in the tree are formed by the attribute values, which are simply one-dimensional array elements. Each element holds a unique branch value. For each and every acceleration value in the acceleration array, the difference between the total tractive force and the required pull force is checked whether the resulting difference between the forces is sufficient enough to accelerate the vehicle when needed on the road. If the difference is positive, the further analysis is done to determine the effects of the variables mentioned previously and hence evaluate acceleration behavior (usually for maximum acceleration) as the acceleration is varied between its predetermined minimum and maximum limits. Therefore, the objective of this paper is to develop a systematic method using decision trees of machine learning to evaluate acceleration behavior of personal motor vehicles based on the forces acting on the vehicle, i.e. vehicle dynamics.

In this study we use the ID3 algorithm (Quinlan, 1986) to learn the decision tree by constructing them topdown, beginning with the root node of the tree. The best variable (attribute) has been selected and used as the test at the root node of the tree. A descendant of the root node is then created for each possible value of this attribute, and the training examples are sorted to the appropriate descendant node. The entire process is then repeated using the training examples associated with each descendant node to select the best attribute to test at that point in the tree (greedy search policy) (Mitchell, 1997). We must note that the gain can be negative. A negative gain indicates that the cost of using the statistical information is more than the cost of determining the path at each node (Rontogiannis and Dimopoulos, 1995). The decision tree in Figure 2 shows the entire tree to classify by sorting the problem through the tree to the appropriate leaf node, then turning classification associated with this leaf (in this case Yes or No). Figure 3 shows the 30% post pruned decision tree (Mitchell, 1997; Esposito, Malerba, and Semeraro, 1997). Both figure sets the depth of the tree to three to view the tree better since the branch numbers get intermingled as the depth is increased. Yes in the tree indicates the value of the target attribute (class) of examples, which, also means that the difference between the total tractive force and the required pull force is sufficient enough to accelerate the vehicle when needed on the road, and the magnitude of this difference will always be positive. No indicates that no such force for a given acceleration can be produced by the engine and therefore vehicle cannot accelerate for a given conditions, and the magnitude of this difference will always be zero or negative.

The leaf and decision nodes receive their values, which, in this case, are shown in numbers by the program, according to the variable organization sequence during the software loop execution and represent attributes. Therefore, '1' represents maximum acceleration, '2' maximum slope, '3' gross vehicle weight, '4' maximum torque, '5' gear ratio, '6' final drive ratio, and finally '7' tire size. The numbered branches indicate the values in the attribute array in the order from left to the right that the specific branches belong to as shown in Table 1.

	1	2	3	4	5
Accel (m/s^2)	0.5	1.0	1.5	2.0	
W (mTon)	1.0728	1.696	1.999	1.379	1.192
sl (deg)	arctan(0.01)	arctan(0.03)	atctan(0.05)		
Tm (Nm)	145.1	332.2	375.6	244.1	173.6
Gr1	3.615	4.484	3.06	3.50	3.143
Rr	4.056	3.16	3.42	3.812	4.765
Ts (m)	0.2997	0.34544	0.3746	0.3327	0.3175

Table 1. Branch numbers and their values for the selected attributes.

Conclusions

In this paper the most effective parameters or variables for describing an accelerating vehicle behavior have been assessed by using decision tree learning. Having analyzed the 37500 data by the fully complete and post pruned decision tree, we conclude that the maximum torque that the engine can produce is the main significant factor in determining an accelerating vehicle behavior and always ends up at the root node of the tree regardless of the several trials with different initial parameters. Moving down the tree branch, the maximum acceleration comes up as the second significant variable to describe an accelerating vehicle behavior. In the third subtree level, the vehicle weight, and in the fourth, the final drive ratio seem to emerge other most significant ones. Lastly, gear ratio, tire size, and maximum slope in route are the least significant ones, depending on their defined ranges, as compared to others. These conclusions have been obtained after many runs with different initial set-ups for mainly train size, test size, and prune size parameters of the decision tree algorithm. Of the 37500 data, 50% to 80% has been chosen as train size, 3% to 10% as test size, and 30% to 50% as prune size in different occasions to come to the above conclusion.

For a vehicle to be really quick, these conclusions suggest that it is important to pay attention to variables like torque, horsepower, desired maximum acceleration, gear ratios, transmission selection, traction, weight. Specifically, maximum performance in longitudinal acceleration of a motor vehicle may be determined by tire traction limit at low speeds other than engine power which may be accounted at high speeds (Gillespie, 1992). All of these factors work in harmony with each other to create a signature acceleration rate. Once the major components are in place, the next thing would be to tune the combination to create an optimum acceleration.

References

Bauer, H. (1999a) *ESP Electronic Stability Program, Technical Instruction*, Robert Bosch GmbH, Society of Automotive Engineers.

Bauer, H. (1999b) *Driving Safety Systems*, Robert Bosch GmbH, Society of Automotive Engineers.

Berikov, V. and Litvinenko, A. (2003) *Methods for statistical data analysis with decision trees*, Sobolev Institute of Mathematics, Novosibirsk.

Breiman, L., Friedman, J., Olshen, R. A. and Stone, C. J. (1984) *Classification and regression trees*, Wadsworth.

Esposito, F., Malerba, D. and Semeraro, G. (1997) 'A Comparative Analysis of Methods for Pruning Decision Trees', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19, pp. 476-491.

Gamberger D. and Šmuc, T. (2001) *Data Mining Server* Available at *http://dms.irb.hr/*, Zagreb, Croatia: Rudjer Boskovic Institute, Laboratory for Information Systems.

Gillespie, T. D. (1992) *Fundamentals of Vehicle Dynamics*, Society of Automotive Engineers, Inc.

Jurgen, R. K. (1999) *Automotive Electronics Handbook*, 2nd Edition, Society of Automotive Engineers.

Karri, V. And Butler, D. (2002) 'Using artificial neural networks to predict vehicle acceleration and yaw angles', In: Wang, L., Rajapakse, J.C., Fukishima, K., Lee, S. And Yao, X. (eds.), *Proceedings of the 9th International Conference on Neural Information Processing (ICONIP'02)* , Vol. 4.

Kim, J. (2005) 'Launching Performance Analysis of a Continuously Variable Transmission Vehicle With Different Torsional Couplings', *Journal of Mechanical Design*, 127, pp. 295-301.

Matlab (2008), Matlab Inc. R2008b.

Menzies, T. and Hu, Y. (2003) 'Data Mining For Very Busy People', *IEEE Computer*, pp. 18-25.

Mitchell, T. M. (1997) *Machine Learning*, McGraw-Hill International Editions.

Quinlan, J. R. (1986) 'Induction of decision trees', *Machine Learning*, 1, pp. 81-106.

Rontogiannis A. and Dimopoulos, N. J., (1995) 'A probabilistic approach for reducing the search cost in binary decision trees', *IEEE Transactions on Systems, Man, and Cybernetics*, 24, pp. 362-370.

Schaffer, C. (1991) 'When does overfitting decrease prediction accuracy in induced decision trees and rule sets? Machine Learning' — *EWSL-91*, Springer Berlin/Heidelberg, pp. 192-205.

Snare, M. C. (2002) *Dynamics model for predicting maximum and typical acceleration rates of passenger vehicles*, M.Sc. Thesis, Virginia Polytechnic Institute and State University, USA.

Wikipedia (2009). *Decision tree learning.* Available at *http://en.wikipedia.org/wiki/Decision_tree_learning* [accessed 31/3/2009].

Wong, J. Y. (1978) *Theory of Ground Vehicles*, John Wiley & Sons Inc.

Yuan, Y. and Shaw, M. J. (1995), 'Induction of fuzzy decision trees', *Fuzzy Sets and Systems*, 69, pp. 125–139.

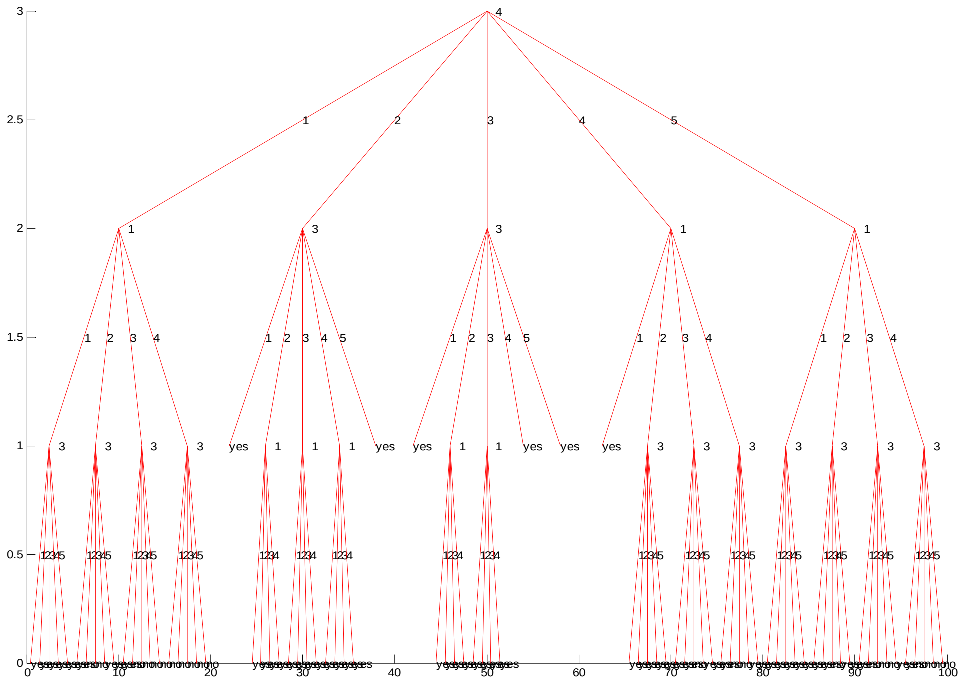

Figure 2. The decison tree output for an accelerating vehicle.

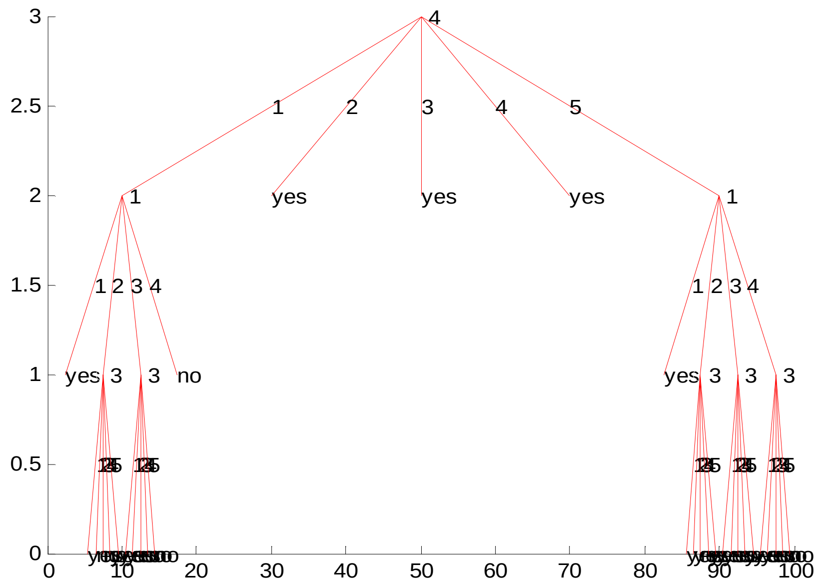

Figure 3. The output of the 30% pruned decison tree for an accelerating vehicle.