Comparison of advanced imputation algorithms for detection of transportation mode and activity episode using GPS data

Transportation Planning and Technology

ISSN: 0308-1060 (Print) 1029-0354 (Online) Journal homepage: http://www.tandfonline.com/loi/gtpt20

Comparison of advanced imputation algorithms for detection of transportation mode and activity episode using GPS data

Tao Feng & Harry J.P. Timmermans

To cite this article: Tao Feng & Harry J.P. Timmermans (2016) Comparison of advanced imputation algorithms for detection of transportation mode and activity episode using GPS data, Transportation Planning and Technology, 39:2, 180-194, DOI: 10.1080/03081060.2015.1127540

To link to this article: http://dx.doi.org/10.1080/03081060.2015.1127540

Published online: 08 Jan 2016.

TRANSPORTATION PLANNING AND TECHNOLOGY, 2016
VOL. 39, NO. 2, 180–194
http://dx.doi.org/10.1080/03081060.2015.1127540

Routledge
Taylor & Francis Group

Comparison of advanced imputation algorithms for detection of transportation mode and activity episode using GPS data

Tao Feng and Harry J.P. Timmermans

Urban Planning Group, Department of the Built Environment, Eindhoven University of Technology, Eindhoven, The Netherlands

ABSTRACT

Global Positioning System (GPS) technologies have been increasingly considered as an alternative to traditional travel survey methods to collect activity-travel data. Algorithms applied to extract activity-travel patterns vary from informal ad-hoc decision rules to advanced machine learning methods and have different accuracy. This paper systematically compares the relative performance of different algorithms for the detection of transportation modes and activity episodes. In particular, naive Bayesian, Bayesian network, logistic regression, multilayer perceptron, support vector machine, decision table, and C4.5 algorithms are selected and compared for the same data according to their overall error rates and hit ratios. Results show that the Bayesian network has a better performance than the other algorithms in terms of the percentage correctly identified instances and Kappa values for both the training data and test data, in the sense that the Bayesian network is relatively efficient and generalizable in the context of GPS data imputation.

ARTICLE HISTORY
Received 3 October 2014
Accepted 30 July 2015

KEYWORDS
Travel survey; activity-travel data; Global Positioning System (GPS); data imputation; classification algorithm; Bayesian network; decision tree; rules

1. Introduction

The application of Global Positioning System (GPS) technologies in the form of GPS-enabled smart phones and GPS stand-alone devices to collect activity-travel data has increased exponentially in recent years. A portfolio of recent studies from across the world is documented in Rasouli and Timmermans (2014). Whereas most applications have been concerned with collecting one or two day activity-travel data, the goal of some other projects has been much more ambitious in that data were collected for several consecutive weeks (Moiseeva, Jessuren, and Timmermans 2010) or in the contexts of national travel surveys (Marchal and Pham 2013; Feng and Timmermans 2013b). Considering this substantial interest in the application of GPS technology, surprisingly little is known about the relative performance of imputation algorithms for transportation mode and activity episode detection. Different degrees of accuracy have been reported in the literature, but these numbers are difficult to compare because spatial settings have a direct bearing on the discriminatory power of GPS information. Traces in high density,

CONTACT Tao Feng t.feng@tue.nl

congested urban areas with high rise buildings, are more likely to contain errors compared to traces in rural, uncongested areas. Similarly, differences in speeds and accelerations of different transportation modes will be less in congested urban areas. Thus, it is considerably more difficult to detect activity-travel patterns in these urban settings. Although prompted recall surveys have been used to validate imputed activity-travel data, these surveys involve additional effort and, potentially, human errors (Bonsall et al. 2011; Feng and Timmermans 2013a). Therefore, it is essential to improve current algorithms of GPS data imputation and examine their relative performance.

The main purpose of GPS data imputation is to detect transportation modes and/or activity episodes. Different procedures have been proposed in the literature to identify activity-travel patterns. A common approach is to detect activities and trips in separated steps. First, trip ends are detected to divide the full sequence into segments, and then the transportation mode for each trip segment is inferred (Stopher and Wargelin 2010). Dwell time is normally used to identify trip ends. This sequential procedure involves the risk that any error in detecting trip ends may propagate into the process of mode detection. As an alternative, one can detect transportation mode and activity episode simultaneously. Activity episodes have a different pattern from travel episodes because activity episodes are either characterized by close-to-zero speeds (Schönfelder et al. 2005; Tsui and Shalaby 2006) or bundles of GPS points (Zheng and Xie 2008). The methods we compare in this paper simultaneously detect transportation mode and activity episode.

Imputation algorithms reported in the literature vary from informal ad-hoc approaches (Wolf, Guensler, and Bachman 2001; Chung and Shalaby 2005; Du and Aultman-Hall 2007) to advanced machine learning methods, such as neural networks, fuzzy logic regression, support vector machines (SVMs), and Bayesian belief networks (BN) (Byon, Abdulhai, and Shalaby 2009; Schuessler and Axhausen 2009; Moiseeva, Jessuren, and Timmermans 2010; Rudloff and Ray 2010; Byon and Liang 2014). The so-called ad-hoc rule-based approaches in general involve a sequential process based on some revealed pattern or correlations, which are extracted empirically from specific data. One common problem of these methods concerns the question to what extent the rules obtained from one case can be generalized to another. Designing the rules may become an issue with increasing dimensionality of the problem and its complexity. The system might become quite sensitive to any new rule. Moreover, there is no guarantee that optimal rules have been applied, while the rules may not be exclusive and exhaustive (Bohte and Maat 2009). As an alternative, machine learning algorithms are potentially more flexible in handling such complex problems. Some learning algorithms can represent complex nonlinear relationships between input and output variables very well (Mehri 2013).

Several representative machine learning algorithms have been applied or discussed in studies of GPS data imputation, including neural networks (Gonzalez et al. 2008), BN (Moiseeva, Jessuren, and Timmermans 2010; Feng and Timmermans, 2013b), fuzzy logistic regression (LR) (Tsui and Shalaby 2006; Schuessler and Axhausen 2009), and decision tables (DTs) (Zheng and Xie 2008). Byon, Abdulhai, and Shalaby (2007) employed a neural network model to detect four types of transportation modes using variables such as speed, acceleration, average horizontal accuracy of 2d coordinate (HDOP), and average number of satellites. Gonzalez et al. (2008) developed a mobile application based on neural networks. A multilayer perceptron model with a back-propagation algorithm was used to discriminate between car, bus, and walking.

Other researchers have used fuzzy logic algorithms in transportation mode detection. Tsui and Shalaby (2006) applied a fuzzy logic model using GPS-only and a combination of GPS and geographical information system data. Similarly, Schuessler and Axhausen (2009) used a fuzzy logic approach for mode identification and compared their results with the Swiss Microcensus on Travel Behaviour data in terms of trip distance, trip duration, and mode distribution. Biljecki, Ledoux, and Oosterom (2013) also used fuzzy expert systems. A disadvantage of these fuzzy logic-based models is that they require expert rules to infer the probabilities. Therefore, we decided not to include these models in this study.

Algorithms for GPS data imputation have been applied in different contexts, leading to variation in reported prediction accuracy. Most studies report an average accuracy between 70% and 85% (Biljecki, Ledoux, and Oosterom 2013). The difference in predicted accuracy depends not only on the algorithm, but also on the number of identified transportation modes, type of input variables, urban setting, and data used to validate the algorithms. Imputations have been conducted using speed-based indicators, such as speed and acceleration (Schuessler and Axhausen 2009; Rudloff and Ray 2010), spatial location-based variables, such as distance to road and/or to bus stops (Chung and Shalaby 2005; Bohte and Maat 2009), and/or personal profiles (Moiseeva, Jessuren, and Timmermans 2010).

Many of these decisions depend on the specific philosophy underlying the approach. For example, the Trace Annotator system (Moiseeva, Jessuren, and Timmermans 2010) was built under the assumption of a minimum amount of information and fast, on-line processing of uploaded traces so that participants can wait for imputed activity-travel diaries and immediately change them if needed. In principle, one would assume that accuracy is improved by adding detailed specific information.

Studies also vary in terms of the number of transportation modes, ranging from three modes (Gonzalez et al. 2008) to a more complete list of 11 modes (Feng and Timmermans 2013b). Moreover, accuracy depends on how the imputation results were validated. Validation can be based on a comparison of imputed data and either individuals' diaries or historical travel survey data. In case of absence of so-called ground truth (mostly the prompted recall data of the same individual), comparisons with historical survey data at an aggregated level have been made (e.g., Schuessler and Axhausen 2009; Feng and Timmermans 2013b), although various sources of error exist in such data. More importantly, aggregate comparisons do not allow capturing accuracy at the individual level. Therefore, a thorough examination of the performance of different algorithms in a same context remains strictly necessary.

Several papers in the recent literature have touched upon the issue of varying performance of imputation algorithms (Zheng and Xie 2008; Rudloff and Ray 2010; Stenneth et al. 2011). For example, Zheng and Xie (2008) applied a decision tree model to detect four types of modes using cellular phone data because of the superiority of the model relative to three other algorithms. Rudloff and Ray (2010), however, selected a LR model because it generates probabilities for each mode, even if its prediction accuracy is slightly lower than that of other methods. However, these studies have involved either a limited number of transportation modes (Zheng and Xie 2008; Stenneth et al. 2011) or a limited number of input variables (Rudloff and Ray 2010). Therefore, in this paper we systematically evaluate the relative performance of different algorithms for GPS data imputation. Nine types of

transportation modes and activity episode are included. Seven representative algorithms which have been applied and/or discussed in the literature are compared: the naive Bayesian classifier (NB), BN, LR models, multilayer perceptron (MP) networks, SVM, DTs and the C4.5 algorithm (C45). Imputation results of these methods are compared using a sample of GPS data collected in the Netherlands.

The remainder of the paper is organized as follows: Section 2 discusses the basic principles underlying each algorithm. Section 3 then introduces the GPS data source that was used to compare the algorithms. Section 4 presents the results. Finally, Section 5 summarizes and concludes this paper.

2. Algorithms

In general, the imputation of activity episodes and transportation modes can be viewed as a nonlinear classification problem. Many algorithms for classification can be applied. Here, we select seven types of algorithms for comparison. The interrelationship between the input and output variables is established in different ways in these algorithms. The list of algorithms is shown in Table 1. In the following section, we will briefly describe these algorithms.

Because the purpose is to detect transportation modes, the transportation mode is the dependent variable with discrete values, y. We use the vector Y to indicate the dependent variable and X to indicate the N independent variables.

$$Xn = (x_1, x_2, \ldots, x_n), \ n \in N. \tag{1}$$

Assume y has K possible values, expressed as y_k, $k \in K$, where K is the total number of transportation modes. The independent variables are input variables based on the GPS traces and possibly other data sources. Each independent variable may have a different number of categories.

2.1. Naive bayesian

The Naive Bayes algorithm is a classification algorithm based on Bayes rule which assumes that the probability of output variable Y equals to certain value y_k is dependent on the probability of X, $p(Y = y_k|X)$. The Naive Bayes algorithm assumes that the attributes, $(x_1, x_2, \ldots, x_n)$, are all conditionally independent of one another, given Y. The value of this assumption is that it dramatically reduces the number of parameters to be estimated.

A naive Bayes classifier considers all these features to contribute independently to the probability. The probability model for a classifier is a conditional model over a dependent

Table 1. List of algorithms and parameter settings.

Id	Algorithms
1	Bayesian Network (BN)
2	Naive Bayesian (NB)
3	Logistic Regression (LR)
4	Multilayer Perceptron (MP)
5	Decision Table (DT)
6	Support Vector Machine (SVM)
7	C4.5 (C45)

class variable Y with a small number of outcomes or classes, conditional on several independent variables X_1 through X_n.

$$p(X_1, \ldots, X_n|Y) = p(X_1|Y)p(X_2|Y) \ldots (X_n|Y). \tag{2}$$

The expression for the probability that Y will take on its kth possible value, y_k, according to Bayes rule, is then

$$p(Y = y_k|X) = \frac{p(Y = y_k)p(X|Y = y_k)}{\sum_j p(Y = y_j)p(X|Y = y_j)}. \tag{3}$$

The problem is that if the number of features N is large or when a feature can take on a large number of values, then basing such a model on probability tables is infeasible. Therefore, using Bayes theorem, a more tractable model can be reformulated as follows:

$$p(Y|X_1, \ldots, X_n) = \frac{P(Y)P(X_1, \ldots, X_n|Y)}{P(X_1, \ldots, X_n)}. \tag{4}$$

The parameters in the Naive Bayes model can be estimated using the maximum likelihood method.

2.2. Bayesian network

A BN is a graphical representation of probabilistic causal information incorporating sets of conditional probability tables. It can be considered an enhanced naïve Bayesian model by relaxing the assumption of independent distributions in that BN consider the joint probability of an attribute with its parent attributes, while the naive Bayesian assume all variables are independent. Thus, a BN represents all factors deemed potentially relevant for observing a particular outcome.

The model is described qualitatively by directed acyclic graphs where nodes and edges represent variables and dependencies between variables. The nodes where the edge originates and ends are called the parent and the child, respectively. Because of the statistical characteristics of BN for probabilistic inference, the probability of each value of a node can be computed when the values of the other variables are known. In a Bayesian network, each variable is conditionally independent of its non-descendent given the state of its parents. That is, if X_i is a variable with parents *parents*(X_i), all variables that are not descendants of X_i are conditionally independent of X_i given *parents*(X_i). Since independence among the variables is clearly defined, not all joint probabilities in the Bayesian system need to be calculated, which provides an efficient way to compute the posterior probabilities.

A BN considers the joint probability of an attribute with its parent attributes. Suppose the set of variables in a BN is (X_1, X_2, $\ldots$, X_n) and that parents (X_i) denotes the set of parents of the node X_i in the BN. Then, the joint probability distribution for (X_1, X_2, $\ldots$, X_n) can be calculated from the product of individual probabilities of the nodes:

$$p(X_1, X_2, \ldots, X_n) = \prod_{n=1}^{N} p(X_i|\text{parents}(X_i)) . \tag{5}$$

The network is represented as a directed graph, together with an associated set of probability tables. In our case, the Bayesian network measures the interrelationship between spatial and temporal factors (input), and activity-travel pattern (output), that is, transportation modes and activity episode. All the input variables are considered as child nodes of the MODE, which labels either an activity episode or one of the transportation modes. The parameters are estimated using the maximum likelihood method when the network structure is determined.

2.3. Logistic regression

Logistic regression is a form of regression analysis used for predicting the outcome of a categorical dependent variable based on one or more predictor variables. The probabilities describing the possible outcome of a single trial are modeled as a function of explanatory variables using a logistic function. Logistic regression assumes a parametric form for the distribution $p(Y|X)$, and directly estimates its parameters from the training data.

In the past, different types of models have been developed as an extension of the basic LR model. The multinomial LR model is such a model which generalizes LR by allowing more than two discrete outcomes. That is, it is a model that is used to predict the probabilities of the different possible outcomes of a categorically distributed dependent variable, given a set of independent variables.

In the general case of a linear classification rule, the probability of class k, $k \in K.$, with the exception of the last class, is equal to

$$p(Y = y_k|X) = \frac{\exp(w_{k0} + \sum_{i=1}^{n} w_{ki}X_i)}{1 + \sum_{j=1}^{K-1} \exp(w_{k0} + \sum_{i=1}^{n} w_{ji}X_i)}. \tag{6}$$

where w is the weight parameter to be estimated, $w \in W$, $W = (w_0, w_1, \ldots, w_n)$. The last class has probability

$$1 - \sum_{j=1}^{k-1} p_j(x_i) = \frac{1}{\sum_{j=1}^{k-1} \exp(w_{k0} + \sum_{i=1}^{n} w_{ji}X_i)}. \tag{7}$$

It can be seen that when Y takes on K possible values, $K-1$ different linear expressions are formulated to capture the distributions for the different values of Y. The distribution for the final, Kth, value of Y is calculated as one minus the probabilities of the first $K-1$ values.

To estimate the parameters of LR models, the (negative) multinomial log-likelihood can be formulated as:

$$L = -\sum_{i=1}^{n}\sum_{j=1}^{k-1} (Y_{ij} \ln (P_j(x_i)) + \left(1 - \sum_{j=1}^{k-1} Y_{ij}\right) \cdot \ln(1 - \left(\sum_{j=1}^{k-1} P_j(x_i)\right). \tag{8}$$

However, in practice over-fitting the training data is a problem that can arise in LR, especially when data are very high dimensional and the training data are sparse. One approach to reducing over-fitting is creating a modified log-likelihood function which

penalizes large values of W. The penalized log-likelihood function can be expressed as:

$$L = -\sum_{i=1}^{n}\sum_{j=1}^{k-1}(Y_{ij}\ln(P_j(x_i)) + \left(1 - \sum_{j=1}^{k-1} Y_{ij}\right) \cdot \ln(1 - \left(\sum_{j=1}^{k-1} P_j(x_i)\right) + \text{ridge} \cdot W^2 \,. \quad (9)$$

The *ridge* is a parameter which needs to be given in advance in the log-likelihood function. In order to find the matrix W for which L is minimized, a Quasi-Newton Method is used to search for the optimized values of the $m^*(k-1)$ variables.

2.4. Multilayer perceptron

A MP is a feedforward artificial neural network model that maps sets of input data onto a set of appropriate output. An MP consists of multiple layers of nodes in a directed graph, with each layer fully connected to the next one. Except for the input nodes, each node is a neuron (or processing element) with a nonlinear activation function. MP utilizes a supervised learning technique called back-propagation for training the network.

The learning occurs in the perceptron by changing connection weights after each piece of data is processed, based on the amount of error in the output compared to the expected result.

$$e_l(n) = d_l(n) - d_l'(n), \quad (10)$$

$$\epsilon(n) = \frac{1}{2}\sum_{l} e_l^2(n). \quad (11)$$

where d is the target value; d' is the value produced by the perceptron (functions); e is the error at the lth iteration; ϵ is the overall amount error which is used to compare with the threshold value.

Due to the fact the neural network with one hidden layer is in principle able to simulate all types of nonlinear problems, we set one hidden layer in the network model. The activation function used the sigmoid function, as follows:

$$\emptyset(y_i) = \frac{1}{(1 + e^{-v_i})}, \quad (12)$$

where y_i is the output of the ith node (neuron) and v_i is the weighted sum of the input synapses.

Since the weights are obtained through an iterated calculation process, some parameters need to be configured in advance. Here, we set the momentum and learning rate as 0.2 and 0.3, respectively. The training time was set as 500, which means that the calculation stops when the number of epochs reaches 500. The final model structure we obtained in this paper has 25 neurons in the hidden layer.

2.5. Decision tables

A DT is a two-dimensional table that shows the action to be taken following a series of related decisions. In general, a DT is composed of rows and columns, presented as a matrix. Each column corresponds to a single rule, with the rows defining the conditions

and actions of the rules. Different algorithms and underlying statistical criteria may be used to construct the DT. In this study, we used an algorithm, which searches the space of attribute subsets by greedy hill-climbing augmented with backtracking. The performance of attribute combinations used in the DT is evaluated based on the overall root mean squared error (RMSE) and the accuracy of different classes.

In this paper, we use the BestFirst algorithm to construct the DT. The BestFirst algorithm was proposed by Kohavi to search the space of attribute subsets by greedy hill-climbing augmented with a backtracking facility. Details can be found in Kohavi (1995).

2.6. Support vector machines

SVMs are supervised learning models with associated learning algorithms that analyze data and recognize patterns. The basic SVM takes a set of input data and predicts, for each given input, which of two possible classes forms the output, making it a non-probabilistic binary linear classifier. Given a set of training examples, each marked as belonging to one of two categories, a SVM training algorithm builds a model that assigns new examples into one category or the other. A SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall on.

SVM performs a nonlinear classification using a kernel trick, mapping inputs into high-dimensional feature spaces. The kernel function normally takes one of the forms: linear, polynomial, radial basis, and sigmoid function. We selected the polynomial function in the subsequent analysis. In addition, we used a sequential minimal optimization algorithm to train a support vector classifier. The algorithm globally replaces all missing values and transforms nominal attributes into binary ones. It also normalizes all attributes, which results in the coefficients in the output based on the normalized data rather than original data. Multi-class problems are solved using pairwise classification. To obtain proper probability estimates, the option that fits LR models to the outputs of the SVM is used. In the multi-class case, the predicted probabilities are coupled using Hastie and Tibshirani's pairwise coupling method (Hall et al. 2009).

2.7. C4.5

C4.5 builds decision trees from a set of training data using the concept of entropy. The training data are a set $S = s_1, s_2, \ldots, s_n$ of already classified samples. Each sample s_i consists of a p-dimensional vector $(x_{1,i}, x_{2,i}, \ldots, x_{p,i})$, where the x_j represents attributes or features of the sample, as well as the class in which s_i falls. At each node of the tree, C4.5 chooses the attribute of the data that most effectively splits its set of samples into subsets enriched in one class or the other. The splitting criterion is the normalized information gain (difference in entropy).

$$\text{entropy}(j|\bar{s}) = \frac{|s_j|}{|\bar{s}|} \log \frac{|s_j|}{|\bar{s}|}, \tag{13}$$

$$\text{entropy}(\bar{s}) = -\sum_{j=1}^{n} \frac{|s_j|}{|\bar{s}|} \log \frac{|s_j|}{|\bar{s}|}, \tag{14}$$

$$\text{Gain}(\bar{s}, j) = \text{entropy}(\bar{s} - \text{entropy}(j|\bar{s})). \tag{15}$$

Then, the attribute with the highest normalized information gain is chosen to make the decision.

C4.5 has been considered as a very efficient classifier in many other applications (Kotsiantis 2007). We will compare C4.5 with other major algorithms in this paper in the context of transportation mode detection.

3. Data

The data used in this paper were collected from a small group of individuals as reported in a pilot study (Moiseeva, Jessuren, and Timmermans 2010). Eight individuals living in Eindhoven, The Netherlands, carried the GPS logger Bluetooth A + during a 6–8 week period. The GPS devices were configured to record data every 3 s (epoch). In addition, to include more transportation modes, we collected activity and travel data specifically for the trips by tram and metro in the city of Rotterdam. During the survey period, 1554 trips were identified by the system and confirmed by the respondents. The confirmation of the real activity-travel diaries was conducted by face-to-face interview. Participants were invited to see their GPS traces on a map using computers. A list of prepared questions regarding the activity-travel diaries, such as the timing (start time and end time) and locations of activities and trips, transportation mode, and activity types were asked to answer. In this way, data that were confirmed as the ground truth were used in this paper to evaluate the performance of different imputation algorithms.

In total, 53,258 data points were used for model calibration and validation. The data points here meant the GPS traces at the epoch level (every 3 s in this paper). It is different from the ones used in other studies (e.g. Zheng and Xie 2008), which were based on the data at trip level. The imputation at trip level normally involves the requirement of a large size of sample apart from the activity/trip segmentation. In this paper, however, we examine the imputation of transportation mode and activity episode only, without addressing the segmentation issue of activities and trips. In real applications regarding segmentation, we applied merge rules to construct the sequence of activities and trips according to the imputation results at the epoch level. In this sense, therefore, the amount of data is considered sufficient to ensure the quality of model estimation and calibration.

The GPS traces include information such as date, time, longitude, latitude, speed, distance, accuracy of the measurement (like Position accuracy of 3d coordinate (PDOP), HDOP, etc.), and number of satellites. To impute transportation modes, the three-second epoch data were averaged within a time window. Furthermore, additional statistical indicators were generated as input variables of the prediction models. Previous studies have indicated that including more variables, which are relevant to the detection of transportation modes, can help increase the imputation accuracy (Bohte and Maat 2009; Moiseeva, Jessuren, and Timmermans 2010; Stopher and Wargelin 2010). Thus, we included variables with regard to speed, spatial distance to networks, accuracy of the GPS log

measurement, and personal profiles. These variables have been used successfully and discussed in previous studies (e.g. Moiseeva, Jessuren, and Timmermans 2010; Feng and Timmermans 2013b). A detailed list of variables is shown in Table 2.

The same input and output variables were used for all algorithms. Thus, each model is based on 18 input variables and 1 output variable, which is named MODE. The dependent variable differentiates 10 transportation modes and the activity episode.

The data were divided into two sets: training data and test data. We randomly draw 75% of the sample as the training dataset and used the remaining 25% as the test dataset. Table 3 shows the partitions of the sample across different transportation modes.

4. Results and analyses

The relative performance of the different algorithms was assessed in terms of the percentage correctly classified instances (CCI), percentage incorrectly classified instances, Kappa value, and RMSE. These statistics were calculated separately for the training and the test data.

4.1. Correctly identified instances, Kappa values and RMSE

Table 4 presents the results of the predictive accuracy of the various algorithms. The kappa statistic measures the agreement in prediction with the true class, with 0 and 1 signifying complete disagreement and complete agreement, respectively. A larger Kappa value indicates better model performance. Taking the results of the training data as an example, one can see that the Kappa values for C4.5 and BN are higher than for other algorithms, 0.997 and 0.998, respectively. All algorithms, except NB, have a Kappa value higher than 0.9, indicating a good performance of most of algorithms. A similar level of Kappa is achieved for the test data.

The CCI value indicates the overall accuracy of an algorithm. Results show that, for the training data, C4.5 and BN have a higher CCI (99.825% and 99.805%) than the other algorithms, indicating a better prediction accuracy. The LR model (94.865%) results in

Table 2. Attribute variables for GPS data imputation.

	Variable names	Content
Input	STDDEVSPEED	Standard deviation of speed
	AVGSPEED	Average speed
	AVGACC	Average acceleration
	MAXSPEED	Maximum speed
	MAXACC	Maximum acceleration
	ACCUMDISTANCE	Accumulated distance
	RRDIST	Distance to road line
	RTDIST	Distance to tram line
	RMDIST	Distance to metro line
	USEDSAT	Number of used satellites
	VIEWSAT	Number of viewed satellites
	VALID	GPX fix type
	PDOP	Position accuracy of 3d coordinate
	HDOP	Horizontal accuracy of 2d coordinate
	CAROWN	Yes if the respondent has a car, no otherwise
	BIKEOWN	Yes if the respondent has a bike, no otherwise
	MOTORBIKEOWN	Yes if the respondent has a motorbike, no otherwise
Output	MODE	Activity episode, train, walk, bike, car, bus, motorbike, running, tram, and metro

Table 3. Selection of training and test datasets.

	Count	Percentage
Training data	39,942	75
Test data	13,316	25
Total	53,258	100

a similar CCI as the SVM model (94.667%) for the training data. Except for the BN and C45s, the DT model (98.886%) has a higher CCI than the other models, for both training and test data. This finding is consistent with conclusions in previous research where the DT algorithm normally resulted in higher prediction accuracy. For both datasets, NB has the lowest CCI of all algorithms. This suggests that the NB is unable to represent the complex relationships between input and output variables, due to its assumption of independence. Consistent with the finding for the training dataset, both BN and C4.5 yield a higher CCI than other algorithms for the test data, indicating that both these algorithms outperform others.

The RMSE indicates the difference between predicted values and true values, a smaller value indicating higher accuracy. As presented in Table 4, for both the training and test data, the BN and C4.5 always result in the lowest level of error of all algorithms. SVM however gives larger errors than others. It means that SVM, at least for the present parameter settings, is not a better option than the other algorithms to detect transportation mode using GPS data.

Examining differences in accuracy between two datasets, results show that for all algorithms the Kappa and CCI values for the test data are lower than for the training data. This is understandable from the perspective of generalizability. BN and C4.5 have a similar level CCI for the training data (99.805% vs. 99.825%), while for the test data the BN results in a slightly higher CCI than C4.5. It suggests that the BN may be more robust.

Apart from prediction accuracy, the complexity of algorithms differs. The NB has a simple network structure, while the C45 results in complicated decision trees, with 214 leaves and 413 trees in total. This means that although C4.5 has a good predictability, in practice calculations can get very complex particularly if some of the values are uncertain and/or if many outcomes are linked. It might be that not all rules are representative.

4.2. Hit ratios

The results of hit ratios show the prediction accuracy for each transportation mode and the activity episode. Table 5 presents the results for the training data. It shows that the

Table 4. Prediction accuracy and model performance.

	Training data				Test data			
Algorithms	CCI (%)	ICI (%)	Kappa	RMSE	CCI (%)	ICI (%)	Kappa	RMSE
BN	99.805	0.195	0.997	0.0185	99.474	0.526	0.993	0.0316
NB	86.966	13.034	0.822	0.0152	86.648	13.352	0.818	0.1533
LR	94.865	5.135	0.926	0.0905	94.510	5.490	0.921	0.0925
MP	97.118	2.882	0.958	0.0675	96.816	3.184	0.954	0.0715
DT	98.886	1.114	0.984	0.0428	98.100	1.900	0.973	0.1029
SVM	94.667	5.333	0.923	0.2718	94.458	5.542	0.920	0.2719
C4.5	99.825	0.175	0.998	0.0180	99.309	0.691	0.990	0.0368

Table 5. Hit ratios for training data by transportation mode and activity episode.

	A	B	C	D	E	F	G	H	I	J
BN	0.997	0.997	0.999	1	0.999	0.999	1	0.999	1	1
NB	0.848	0.969	0.934	0.799	0.836	0.926	0.949	0.98	1	0.983
LR	0.989	0.991	0.818	0.928	0.891	0.758	0.947	0.76	1	1
MP	0.998	0.974	0.916	0.926	0.965	0.743	0.989	0.985	1	1
DT	0.999	0.971	0.958	0.985	0.979	0.99	0.991	0.974	0.982	0.98
SVM	0.987	0.999	0.76	0.925	0.876	0.888	0.971	0.654	1	1
C4.5	1	0.999	0.993	0.997	0.997	0.994	0.998	0.999	0.996	0.99

Note: A-Activity episode; B-Train; C-Walking; D-Bike; E-Car; F-Bus; G-Motorbike; H-Running; I-Tram; J-Metro.

BN predicts four transportation modes – bike, motorbike, tram, and metro – with 100% accuracy, while others modes are predicted with an accuracy equal to or higher than 99.7%. The hit ratio of the BN classifier is comparable to that of the other algorithms.

The C4.5 and DT also achieve a high accuracy, but not as good as the BN model. Other algorithms perform less well. For example, the NB model has a low hit ratio for bike (0.799), while the SVM has a low hit ratio for running (0.654) and walking (0.76). In addition, both the LR (0.758) and the MP (0.743) result in low predictive accuracy for the bus mode, while the LR did not predict well the running mode (0.76). Table 6 presents the results for the test data. Similar conclusions may be drawn for this data. The comparison of the hit ratios between the training and test data shows that the hit ratios for each facet (activity episode or transportation mode) do not decrease much across all algorithms (as shown in Table 7). However, if we average the value difference (hit ratios) between training and test data for each algorithm across all transportation modes, some interesting results can be found. First, BN yields the least average difference (0.0063), indicating the BN is relatively stable than other algorithms. In another words, the BN model may be more generalizable than other algorithms in real predictions. In addition, C4.5 results in a larger level of difference (0.0095) than BN, although, as presented above, both algorithms have relatively higher accuracy than others. This means the C4.5 is less generalizable than the BN.

On the other hand, the DT has the largest result of difference (0.0212) among all algorithms, indicating that DT is less generalizable than other algorithms. The major portion of such a large difference is attributed to the walking mode (0.971 for training data and 0.948 for test data). This means that DT may result in less accurate predictions, especially in the detection of the walking mode. Considering the fact that several existing studies attempted to use DT to detect transportation modes, one should note that DT may not be a good option considering the accuracy of real predictions.

Table 6. Hit ratios for test data by transportation mode and activity episode.

	A	B	C	D	E	F	G	H	I	J
BN	0.996	0.993	0.988	0.997	0.994	0.977	0.999	1	1	0.983
NB	0.849	0.964	0.942	0.789	0.826	0.9	0.946	0.963	1	0.975
LR	0.99	0.994	0.815	0.915	0.882	0.733	0.935	0.752	1	1
MP	0.998	0.976	0.896	0.926	0.962	0.708	0.987	0.974	1	1
DT	0.998	0.948	0.939	0.973	0.97	0.973	0.982	0.963	0.892	0.959
SVM	0.987	0.998	0.763	0.931	0.869	0.844	0.968	0.641	0.985	1
C4.5	0.998	0.998	0.974	0.992	0.987	0.98	0.991	0.956	1	0.992

Note: A-Activity episode; B-Train; C-Walking; D-Bike; E-Car; F-Bus; G-Motorbike; H-Running; I-Tram; J-Metro.

Table 7. Difference of hit ratios between training and test data.

	A	B	C	D	E	F	G	H	I	J	Average
BN	0.001	0.004	0.011	0.003	0.005	0.022	0.001	−0.001	0	0.017	0.0063
NB	−0.001	0.005	−0.008	0.01	0.01	0.026	0.003	0.017	0	0.008	0.007
LR	−0.001	−0.003	0.003	0.013	0.009	0.025	0.012	0.008	0	0	0.0066
MP	0	−0.002	0.02	0	0.003	0.035	0.002	0.011	0	0	0.0069
DT	0.001	0.023	0.019	0.012	0.009	0.017	0.009	0.011	0.09	0.021	0.0212
SVM	0	0.001	−0.003	−0.006	0.007	0.044	0.003	0.013	0.015	0	0.0074
C4.5	0.002	0.001	0.019	0.005	0.01	0.014	0.007	0.043	−0.004	−0.002	0.0095

5. Conclusions

The use of GPS technologies to collect activity-travel data and reduce respondent burden ultimately depends on the accuracy of imputation algorithms. Various algorithms have been suggested in the literature, including the traditional ad hoc rules and machine learning algorithms. It is difficult to assess the relative performance of these algorithms because reported accuracy depends on the number of input variables, the number of transportation modes, the data used for validation, and perhaps most importantly on urban context. Little is known about the relative performance of different algorithms. Although several studies touch upon this issue, none addressed the performance of different algorithms in a sufficient and systematic manner.

In this paper, we selected several representative machine learning algorithms and evaluated their relative performance for GPS data imputation. In particular, we compared the naive Bayesian, Bayesian network, LR, MP, SVM, DT, and C45. We implemented all algorithms in the same context and simultaneously detected transportation modes and activity episode.

Results show that the Bayesian network outperforms the other algorithms in terms of the percentage correctly identified instances and Kappa for both the training and test data. C4.5 results in a similar level of prediction accuracy. However, the Bayesian network seems more robust. The BN also has the highest hit ratio.

The superiority of the Bayesian network model in this study suggests that it should be a serious candidate in any application. Besides the imputation accuracy, BN has a flexible network structure, which is based on the conditional probabilities between different variables. Different from the imputations in a way of black-box, BN allows identify the level of probabilities in the imputation. The imputation output for each transportation mode therefore has a certain level of probability associated, providing a way of measuring the level of uncertainty.

It goes without saying that, although this study has systematically compared the performance of different imputation algorithms for the same data set, all data relate to the same area. Additional comparative studies are thus needed to better understand the relative performance of different algorithms in different spatial settings and assess the generalizability of our results. One may also examine the performance using a large dataset with a special focus on the detection of activity episode because the activity pattern can be different in the context of indoor and outdoor movement. To that end, a larger sample of the ground truth data associated with GPS traces is needed.

In addition, other algorithms that have not been explored yet may be included. For example, the relatively simple heuristic algorithms may be used regarding their popularity.

Moreover, the fuzzy logic-based models seem valuable to examine. One can apply the expert rules, which were identified successfully in existing studies (e.g. Tsui and Shalaby 2006), into the same context we implemented for the other algorithms. Apart from that, improvements of existing algorithms can be in the hybrid use of different classifiers, such as random forests (Breiman, 2001) that rely on ensembles of predictors.

Acknowledgements

The research leading to these results has received funding from the European Research Council under the European Community's 7th Framework Programme (FP7/2007-2013)/ERC grant agreement no. 230517 (U4IA project). The views and opinions expressed in this paper represent those of the authors only. The ERC and European Community are not liable for any use that may be made of the information in this publication.

Disclosure statement

No potential conflict of interest was reported by the authors.

References

Biljecki, F., H. Ledoux, and P. V. Oosterom. 2013. "Transportation Mode-Based Segmentation and Classification of Movement Trajectories." *International Journal of Geographical Information Science* 27: 385–407.

Bohte, W., and K. Maat. 2009. "Deriving and Validating Trip Purpose and Travel Modes for Multiday GPS-Based Surveys: A Large-Scale Application in the Netherlands." *Transportation Research Part C: Emerging Technologies* 17: 285–297.

Bonsall, P., J. Schade, L. Roessger, and B. Lythgoe. 2011. "Can We Believe What They Tell Us? Factors Affecting People's Engagement with Survey Tasks." Paper presented at the 9th international conference on transport survey methods, Termas de Puyehue, Chile, November 14–18.

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

Byon, Y. J., B. Abdulhai, and A. S. Shalaby. 2007. "Impact of Sampling Rate of GPS-Enabled Cell Phones on Mode Detection and GIS Map Matching Performance." Paper presented at the 86th annual meeting of the Transportation Research Board, Washington DC, January 21–25.

Byon, Y. J., B. Abdulhai, and A. Shalaby. 2009. "Real-Time Transportation Mode Detection via Tracking Global Positioning System Mobile Devices." *Journal of Intelligent Transportation Systems* 13 (4): 161–170.

Byon, Y. J., and S. Liang. (2014) "Real-Time Transportation Mode Detection Using Smartphones and Artificial Neural Networks: Emphasis on Performance Comparisons Between Smartphone and Conventional GPS Sensor." *Journal of Intelligent Transportation Systems*. doi:10.1080/15472450.2013.824762

Chung, E. H., and A. Shalaby. 2005. "A Trip Reconstruction Tool for GPS-Based Personal Travel Surveys." *Transportation Planning and Technology* 28: 381–401.

Du, J., and L. Aultman-Hall. 2007. "Increasing the Accuracy of Trip Rate Information from Passive Multi-Day GPS Travel Datasets: Automatic Trip End Identification Issues." *Transportation Research Part A: Policy and Practice* 41: 220–232.

Feng, T., and H. J. P. Timmermans. 2013a. "Analysis of Error in Prompted Recall Surveys." Paper presented at the XII NECTAR international conference, São Miguel Island, Azores, June 16–18.

Feng, T., and H. J. P. Timmermans. 2013b. "Transportation Mode Recognition Using GPS and Accelerometer Data." *Transportation Research Part C* 37: 118–130.

Gonzalez, P. A., J. S. Weinstein, S. J. Barbeau, M. A. Labrador, P. L. Winters, N. L. Georggi, and R Perez. 2008. "Automating Mode Detection Using Neural Networks and Assisted Data Collected

Using GPS-Enabled Mobile Phones." Paper presented at the 15th world congress on intelligent transportation systems, New York, November 16–20.
Hall, M., E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. 2009. "The WEKA Data Mining Software: An Update." *SIGKDD Explorations* 11 (1): 10–18.
Kohavi, R. 1995. "The Power of Decision Tables." *Lecture Notes in Computer Science* 21: 174–189.
Kotsiantis, S. B. 2007. *Supervised Machine Learning: A Review of Classification Techniques. Frontiers in Artificial Intelligence and Applications*. Amsterdam: IOS Press.
Marchal, P., and T. Pham. 2013. "Comparison of Conventional Versus GPS-Based Data Collection in the French National Travel Survey. "Paper presented at the international conference of the new techniques and technologies for Statistics, Brussels, March 5–7.
Mehri, M. 2013. "A Comparison of Neural Network Models, Fuzzy Logic, and Multiple Linear Regression for Prediction of Hatchability." *Poultry Science* 92: 1138–1142.
Moiseeva, A., J. Jessuren, and H. J. P. Timmermans. 2010. "Semiautomatic Imputation of Activity Travel Diaries: Use of Global Positioning System Traces, Prompted Recall, and Context-Sensitive Learning algorithms." *Transportation Research Record: Journal of the Transportation Research Board* 2183: 60–68.
Rasouli, S., and Timmermans, H. J. P. eds. 2014. *Mobile Technologies for Activity-travel Data Collection and Analysis*. New York: IGI Publishers.
Rudloff, C., and M. Ray. 2010. "Detecting Travel Modes and Profiling Commuter Habits Solely Based on GPS Data." Paper presented at the 89th annual meeting of the transportation research board, Washington, DC, January 10–14.
Schönfelder, S., H. Li, R. Guensler, J. Ogle, and K. W. Axhausen. 2005. "Analysis of Commute Atlanta Vehicle Instrumented GPS Data: Destination Choice Behavior and Activity Spaces." *Arbeitsberichte Verkehrs- und Raumplanung* 303, Report, 24 pp.
Schuessler, N., and K. W. Axhausen. 2009. "Processing Raw Data from Global Positioning Systems Without Additional Information." *Transportation Research Record: Journal of the Transportation Research Board* 2105: 28–36.
Stenneth, L., O. Wolfson, P. S. Yu, and B. Xu. 2011. "Transportation Mode Detection Using Mobile Phones and GIS Information." Paper presented at the proceedings of the 19th ACM SIGSPATIAL international conference on advances in geographic information systems. Chicago, IL, USA, November 1–4.
Stopher, P. R., and L. Wargelin. 2010. "Conducting a Household Travel Survey with GPS: Reports on a Pilot Study." Paper presented at the 12th world conference on transportation research, Lisboa, Portugal, July 11–15.
Tsui, S. Y. A., and A. S. Shalaby. 2006. "Enhanced System for Link and Mode Identification for Personal Travel Surveys Based on Global Positioning Systems." *Transportation Research Record: Journal of the Transportation Research Board* 1972: 38–45.
Wolf, J., R. Guensler, and W. Bachman. 2001. "Elimination of the Travel Diary: An Experiment to Derive Trip Purpose from GPS Travel Data." Paper presented at the 80th annual meeting of the transportation research board, Washington DC, USA, January 7–11.
Zheng, Y., and X. Xie. 2008. "Learning Transportation Mode from Raw GPS Data for Geographic Applications on the Web." Paper presented at the 17th World Wide Web conference, Beijing, China, April 21–25.