View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by Apollo

Cite this article
Hillel T, Elshafie MZEB and Jin Y (2018)
Recreating passenger mode choice-sets for transport simulation: A case study of London, UK.
Proceedings of the Institution of Civil Engineers – Smart Infrastructure and Construction
171(1): 29–42, https://doi.org/10.1680/jsmic.17.00018

Research Article
Paper 1700018
Received 30/10/2017; Accepted 11/05/2018

Published with permission by the ICE under the CC-BY 4.0 license.
(http://creativecommons.org/licenses/by/4.0/)

Keywords: mathematical modelling/transport management/transport planning

Smart Infrastructure and Construction

ice publishing

Recreating passenger mode choice-sets for transport simulation: A case study of London, UK

Tim Hillel MEng, MRes
PhD Student, Department of Engineering, University of Cambridge, Cambridge, UK (corresponding author: th389@cam.ac.uk) (Orcid:0000-0001-6872-2235)

Mohammed Z. E. B. Elshafie BSc (Hon), MPhil, PhD
Laing O'Rourke Senior Lecturer in Construction Engineering and Technology, Department of Engineering, University of Cambridge, Cambridge, UK (Orcid:0000-0001-8307-7115)

Ying Jin BArch, PhD
University Senior Lecturer, Department of Architecture, University of Cambridge, Cambridge, UK (Orcid:0000-0003-2683-6829)

Urban transport infrastructure is under increasing pressure from rising travel demand in many cities worldwide. It is no longer sustainable or even economically viable to cope with increased demand by continually adding capacity to transport networks. Instead, travel demand must be managed by encouraging passengers to adapt their travel behaviour. This approach necessitates a significantly deeper understanding of the seemingly random variations of passenger flows than is afforded by the current travel demand modelling techniques. This study presents a new modelling framework for predicting travel mode choice, through recreating and analysing the choice-set faced by the passenger at the time of day of their travel. A new data set has been developed by combining individual trip records from the London Travel Demand Survey (LTDS), with systematically matched trip trajectories alongside their corresponding mode alternatives from an online directions service and detailed estimates of public transport fares and car operating costs. The value of the data set is demonstrated by comparing two models of passenger mode choice based on stochastic gradient boosting trees, one using only the LTDS data and the other with our full data set. The models are then used to identify the key factors driving passenger mode choice.

Notation

C	choice-set
c	index of option from choice set
d_{opt}	predicted driving trip duration for optimistic traffic
d_{pes}	predicted driving trip duration for pessimistic traffic
d_{typ}	predicted driving trip duration for typical traffic
i	index of choice situation
$\mathcal{L}_{\text{ce}}$	multiclass classification error
$\mathcal{L}_{\text{ese}}$	expected simulation error
$\mathcal{L}_{\text{nll}}$	negative log-likelihood loss
N	total number of choice situations
$\widehat{p}_{i,c}$	predicted probability for option c for choice situation i
v	traffic variability
$y_{i,c}$	Boolean choice indicator for option c for choice situation i

1. Introduction

Urban transport networks are facing a number of unprecedented challenges, most notably how to deal with increased transport demand from rapid population increase. London is no exception to this where the population is expected to increase by 25% over the next 25 years (Greater London Authority, 2016). Over the same period, the UK government has committed to a legally binding target to cut greenhouse gas emissions drastically (by 50% of baseline levels by 2027 and 80% by 2050) (Department for Business, Energy and Industrial Strategy, 2017). Addressing these challenges requires the intelligent management of travel demand, combining network improvements with policy and regulatory changes, that encourage passengers to adapt their travel behaviour. Transport models used for infrastructure investment and operations planning conventionally rely on multinomial logit random utility models (RUMs) to predict passenger mode choice (Ben-Akiva and Lerman, 1985). These models define utility equations for each mode, which are used to predict the mode taken by the passenger for a given trip. RUMs have several desirable features that help explain their ubiquitous use within transport modelling: they define a linear function for utility which is easy for modellers and transport stakeholders to interpret; they make efficient use of sparse data; and they can be calibrated to aggregate survey data and passenger/vehicle counts (Train, 2009; Walker and Ben-Akiva, 2002).

There are two fundamental limitations of the RUM approach. The first is that separate parameters for each variable must be trained for each covariate within the utility equation. This increases the model complexity by order of (n^2) for input predictors in terms of the parameters that must be estimated. This severely limits the number of predictors used in a RUM as problems arise with the convergence of parameter estimates in models with high dimensionality (Nyquist, 1991; Zahid and Tutz, 2013). Second, the utility specification must be defined beforehand by the modeller prior to fitting the model. In practice, this restricts modellers to using coarse categorical covariates and only investigating first-order interactions of the covariates with the

Downloaded by [] on [17/08/18]. Published with permission by the ICE under the CC-BY license

input predictors. Therefore, these models do not provide the level of insight into the seemingly random variations of passenger flows required for effective travel demand management.

The adoption of several notable transportation-related technologies, such as live travel information feeds, mobile phone-based location services, contactless smart cards, vehicle tracking cameras and connected vehicles, has driven a step change in the availability of data on passenger movements of several orders of magnitude. These data provide the opportunity to build much richer models of passenger behaviour, which directly infer the relationship between transport and environment conditions, and passenger travel decisions. These models could be used to generate passenger flows with finer spatial, temporal and behavioural granularity than is possible with current techniques. However, there have been limited attempts at integrating these disparate data sources to create cohesive models.

1.1 Applications of machine learning to predicting travel mode choice

Several studies have investigated the use of data-driven machine-learning models as an alternative to RUMs for predicting mode choice. These approaches include applications of artificial neural networks (Cantarella and de Luca, 2003, 2005; Hagenauer and Helbich, 2017; Hensher and Ton, 2000; Omrani, 2015; Omrani *et al*., 2013; Xie *et al*., 2003; Zhao *et al*., 2010), support vector machines (Hagenauer and Helbich, 2017; Omrani, 2015; Xian-Yu, 2011; Zhang and Xie, 2008), decision trees (DTs) (Tang *et al*., 2015; Xie *et al*., 2003) and ensemble methods (Biagioni *et al*., 2009; Hagenauer and Helbich, 2017; Rasouli and Timmermans, 2014).

These studies have made an important contribution in demonstrating the suitability of machine-learning classifiers for predicting passenger mode choice. However, the models are limited by their input data, which are typically the raw data from the trip diaries of travel surveys. These data sets do not include any details of the choice-set faced by a passenger when choosing mode of travel, and so the impacts of attributes of each mode, such as duration and cost, cannot be investigated.

This study identifies four further limitations of previous studies.

(*a*) Machine-learning models are treated as discrete classifiers, predicting one mode with probability 1 and all others with probability 0. The primary use case for mode choice models is probabilistic simulation. As such, it is more beneficial for mode choice models to output well-calibrated choice probabilities.
(*b*) Classification accuracy is often used as the primary metric for assessing predictive power. This is an unsuitable metric for comparing classifiers as it exhibits the accuracy paradox (see Section 2.2.2 and the study by Bruckhaus (2007)).
(*c*) Parameters for the machine-learning classifiers, such as the number of DTs in an ensemble, are not selected in a methodological and unbiased way. Model performance is highly dependent on chosen parameter values, so it is important that a rigorous selection method is used (Hoos *et al*., 2014).
(*d*) While stochastic gradient boosting DTs (GBDTs) have been shown to have best in class performance in a number of similar tasks (Brown and Mues, 2012; Caruana and Niculescu-Mizil, 2006; Chapelle and Chang, 2011; Zhang *et al*., 2017), they have seen limited use for predictive transport modelling.

To address these requirements, this study presents a new modelling framework for predicting mode choice, through recreating and analysing the choice-set faced by the passenger at the time of travel. GBDTs are used to predict mode choice, with parameter values selected using sequential optimisation. Continuous choice probabilities are obtained for all modes, and the model fit is primarily assessed using log-likelihood loss.

2. Data and methods

2.1 The data set

A new data set has been developed by combining individual records of the London Travel Demand Survey (LTDS) with closely matched trip trajectories alongside their corresponding mode alternatives (i.e. the choice-set faced by the passenger at the time of travel) from a directions application programming interface (API) and precise estimates of public transport fares and car operating costs. This represents the most comprehensive and closely tailored travel data set for estimating travel choices in a major metropolitan area. The full dataset is available as supplementary material to this paper on the website of this journal.

2.1.1 London Travel Demand Survey

A consistent set of individual records from the LTDS from April 2012 to March 2015 is used to build the data set. This comprises 134 486 trips made by 57 640 London residents within Greater London (TfL, 2015). There are three elements to the survey: a household questionnaire, individual questionnaires and individual trip diaries. Each household is surveyed on one day of the year.

The household questionnaire contains details of household structure and characteristics, as well as household members and vehicle ownership. Each household member over 5 years of age completes the individual questionnaire and trip diary. The individual questionnaire details socioeconomic and travel-related information. The trip diary records details of each stage of every journey made on the survey date, alongside the journey purpose, modes used, trip start time and duration.

The trips in the LTDS are preprocessed before building the new data set. Short trips within a single postcode area are removed. Next, each trip is assigned to one of the four main modes: walking, cycling, public transport, and driving (which includes car passenger, taxi, van and motorbike). For mixed mode journeys, the assigned mode is the mode used to travel the most distance. Trips made by other modes – for example, coach or boat – account for less than 0·5% of journeys and so are omitted.

Finally, each trip is assigned to one of five journey purposes, derived from the origin and destination purposes in the LTDS.

Downloaded by [] on [17/08/18]. Published with permission by the ICE under the CC-BY license

These purposes are employer's business (B), home-based work (HBW), home-based education (HBE), home-based other (HBO) and non-home based other (NHBO).

2.1.2 Directions API

The Google Maps directions API allows for the retrieval of predicted journey information, including optimal route and duration. The calculation of the route and duration depends on the mode specified.

- Walking: The route and duration returned are independent of departure time, with routes prioritising footpaths and pavements.
- Cycling: The route and duration are independent of departure time, with routes prioritising cycle paths and quieter roads where available.
- Public transport: The routes and durations are extracted for specific times of the day and days of the week, matching the surveyed trip. The route and duration are calculated using timetable information. The returned route is broken into separate stages for each walking/bus/rail leg of the journey. Transfers between services represent separate stages.
- Driving: The routes and durations are extracted for specific times of the day and days of the week, matching the surveyed trip. The route and duration are calculated using a traffic model that represents the typical traffic conditions on that day and time. Three traffic levels can be specified which impact both route and duration: optimistic, pessimistic and best guess.

2.1.3 Travel costs

The estimation of travel costs makes full use of the socioeconomic and demographic profiles from the LTDS with the corresponding route and duration data from the directions service. The costs are closely tailored to represent accurately public transport fares, fuel prices and the Central London Congestion Charge.

2.1.3.1 PUBLIC TRANSPORT FARES

Public transport fares are determined for single trips using Oyster card/contactless payment.

The fares for buses and trams are charged per boarding, independent of trip length. There is no peak/off-peak pricing or zoning of fares. The three fare levels – full, half (for reduced bus fare holders or children aged 16–18 who live outside London) or free (for all children under 16, children aged 16–18 living in London, Transport for London (TfL) staff, police or national concession buspass holders) – are applied as per each relevant LTDS journey. These fare types are determined for each passenger from the LTDS information.

Fares for National Rail, London Underground, London Overground and the Docklands Light Railway are dependent on four variables: fare zones (zones 1–9 plus extension fares for specific stations outside these zones), services used (seven fare types for different rail services), time of day (peaks from 06.30 to 09.30 and from 16.30 to 19.30) and passenger fare type (normal, child under 16, 16+, disabled persons' railcard, other discount railcards and free). The TfL Unified API has been used as far as possible to collect the correct fares across the variants above. For rail fares, the fare type is determined by travel stage. For instance, the peak/off-peak classification is determined from the start time of each corresponding stage. As with any major metropolitan area, there are some exceptions to address within the train fare scheme.

First, there are several pairs of stations on the TfL network where a free walking interchange between lines is permitted, when exiting from one station and reentering at a paired station. To allow for this, a data set of all free transfer station pairs on the TfL network has been assembled to identify routes where separate services constitute one continuous journey. Second, there is often more than one route available between two stations, each with different fares. In the real world, these fares are determined either by tapping the contactless payment card on a special interchange card reader at certain stations or by exiting and reentering a station with a free interchange. If more than one fare is available, the TFL Unified API returns a list detailing the available fares and the required transfer stations. To ensure that the correct fare is assigned, a list of transfer stations from the directions result is assembled, and this is used to determine if the required transfer station is passed through.

For complex, multilegged journeys, the total public transport fare is calculated as the sum of the individual fares for each separate part. A new fare is recorded each time a bus is boarded or a journey exits and reenters a station without a free interchange.

2.1.3.2 DRIVING

The driving costs consist of operating cost and the congestion charge cost. Parking costs are not directly included here as there are no data to determine parking rates at the destination. Instead, parking costs are accounted for as part of the residual in model estimation, following Jin *et al*. (2002). The operating cost is calculated using the vehicle operating costs (VOCs) formula presented in the UK Department for Transport (DfT, 2014) *Transport Analysis Guidance* (WebTAG), with the fuel type determined by the vehicle(s) available to the household. The lowest cost fuel type is assumed if there is more than one vehicle in the household. If the household owns no vehicle, an average fuel type is used. The congestion charge is included if the driving route crosses the congestion charge zone. The charge is ignored for journeys on weekends, bank holidays and outside hours, as well as for other exemptions and zero-charge vehicles.

2.1.4 Processing the data set

An automated script has been written to process the data, the overall structure of which is illustrated in Figure 1. This develops the initial work from Hillel *et al*. (2016).

Six directions API requests are made for each LTDS trip: walking; cycling; public transport; and driving under optimistic, pessimistic and best-guess traffic conditions. For the public transport and driving requests, the requested departure time and date are matched by day of week and start time to the original trip, with the departure time set for two weeks after the request

Downloaded by [] on [17/08/18]. Published with permission by the ICE under the CC-BY license

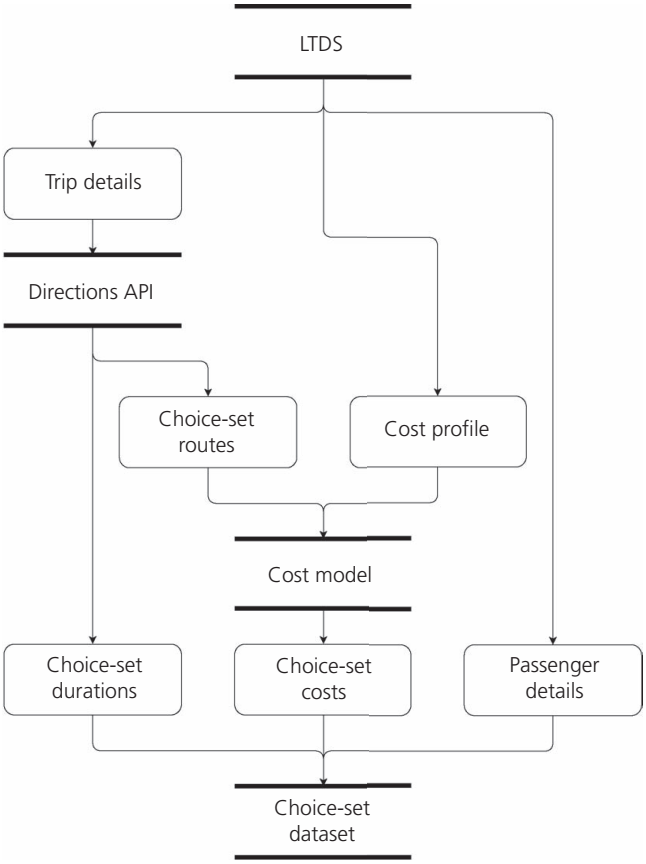

Figure 1. Flow chart of data set building process

date to ensure that the routes are calculated for typical conditions and do not include planned public transport disruptions or real-time traffic. Figure 2 illustrates the routes generated by the directions API for a single trip.

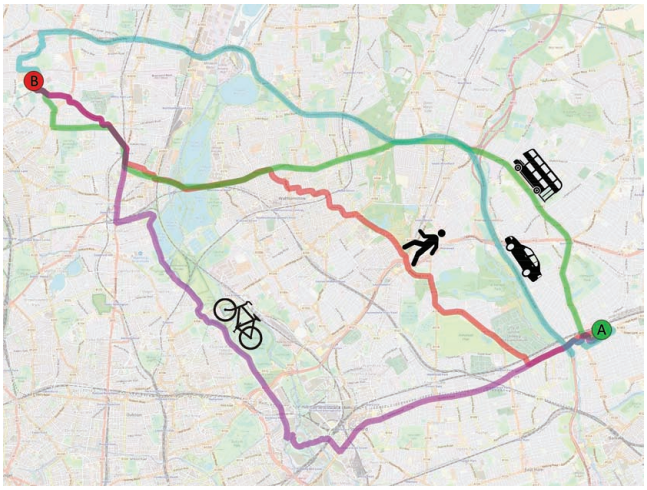

Figure 2. Diagram of routes generated by directions API for a single trip

The durations of each separate stage in the public transport route are analysed to calculate the access duration, interchange duration and on-board durations for bus and rail. A measure of the traffic variability for the driving route is calculated as follows

$$v = \frac{d_{\text{opt}} - d_{\text{pes}}}{d_{\text{typ}}} \quad (1)$$

where d_{opt}, d_{pes} and d_{typ} are the durations for optimistic, pessimistic and best-guess traffic, respectively, as predicted by the directions API.

Each public transport and driving entry is then processed to add the public transport and driving costs.

Finally, the authors remove trips that are out of the scope of the study. The authors retain all trips that meet the following criteria: (*a*) the routes for all modes are completely contained within the bounding box of the combination of the London Boroughs, M25 orbital motorway and all TfL stations; (*b*) all steps within the suggested public transport route use only TfL services and/or stations (this ensures that all fares are correctly estimated); and (*c*) the suggested public transport route has at least one public transport step (i.e. it is not purely walking).

The finished data set contains entries for 81 086 journeys across 3 years (2012/2013–2014/2015). The data set is naturally imbalanced, with different ratios of trips for walking (17·6%), cycling (3·0%), public transport (35·3%) and driving (44·2%).

Figure 3 is generated by projecting the path for the driving option (under best guess traffic) for all trips in the data set, showing the data set's geographical coverage. The trips are projected at low transparency, so the line intensity represents how frequently the

Figure 3. Diagram of driving paths in data set

Downloaded by [] on [17/08/18]. Published with permission by the ICE under the CC-BY license

Table 1. Data set attributes and descriptions

Type	Group	Attribute	Description
ID and context	Context	`trip_id`	Unique ID for each trip
		`travel_mode`	Mode of travel chosen by LTDS trip
		`ori_postcode`	Origin postcode of the trip
		`desti_postcode`	Destination postcode of the trip
Socioeconomic and demographic profile (from LTDS)	Categorical	`purpose`	Journey purpose for trip (B, HBW, HBE, HBO and NHBO)
		`fare_type`	Public transport fare type of passenger (16+, child, disabled, free and full)
		`fuel_type`	Fuel type of passenger's vehicle (diesel/petrol/hybrid car or diesel/petrol LGV)
		`driving_license`	Whether the traveller has a driving licence
		`sex`	Gender of passenger
	Ordered numerical	`age`	Age of passenger in years
		`distance`	Straight line trip distance
		`car_ownership`	Car ownership of household (no cars, less than one car per adult and one or more cars per adult)
		`bus_scale`	Percentage of the full bus fare paid by the passenger
		`start_time`	Start time of trip
		`day_of_week`	Day of the week of travel
		`travel_month`	Month of year of travel
Choice-set data (from directions cost models)	Walking	`dur_walking`	Duration of walking route
	Cycling	`dur_cycling`	Duration of cycling route
	Public transport	`dur_pt:rail`	Duration spent on rail services on public transport route
		`dur_pt:bus`	Duration spent on bus services on public transport route
		`dur_pt:access`	Duration walking to/from first/last stop on public transport route
		`dur_pt:interchange`	Total duration of public transport interchanges interchanges
		`dur_pt:total`	Duration of whole public transport route (`dur_pt:access+ dur_pt:rail+dur_pt:bus+dur_pt:interchange`)
		`cost_pt`	Cost of whole public transport route
		`n_ints`	Total number of public transport interchanges (rail–rail, bus–bus, bus–rail and rail–bus)
	Driving	`dur_driving`	Duration of driving route
		`cost_driving:VOC`	Vehicle operation costs of driving route
		`cost_driving: con_charge`	Congestion charge for driving route
		`traffic_var`	Traffic variability (shown in Equation 2)

LGV, light goods vehicle

underlying road is used in the data set. A summary of the attributes for each trip in the data set is given in Table 1.

2.2 Model training and comparison

The data set provides details of the choice-set faced by the passenger at the time of travel, out of walking, cycling, public transport and driving. Machine-learning models of mode choice are trained in order to predict the likely choices made by the passenger and investigate the relationship between the network and environment conditions and mode choice.

The primary use case for mode choice models is probabilistic simulation of future trips, where the mode choice for each trip is drawn randomly from a probability distribution across each mode. This presents a supervised probabilistic classification problem. A mode choice model must therefore generate a probability distribution for the mode choice for each trip from a feature vector of attributes of the trip.

In order to emulate the use case of predicting future trips, the data set is divided by survey year into a training set (2 years, April 2012–March 2014), used for model optimisation, cross-validation performance estimation, final model training and a holdout test set (1 year, April 2014–March 2015), used for the performance evaluation of the final models.

Two machine-learning models are trained on the data set, one using only the socioeconomic and demographic profile data from the LTDS and the other also using the additional choice-set data from the directions API and cost models (see Table 1 for a list of corresponding attributes). These are referred to as the raw data model and the choice-set model, respectively.

2.2.1 Gradient boosting trees

This paper investigates the use of GBDTs to predict passenger mode choice. GBDTs are a class of ensemble methods, which make predictions by combining the votes of several shallow DT weak learners (Friedman, 2002). GBDTs have been shown to consistently outperform other classification algorithms in a range of tasks (Brown and Mues, 2012; Caruana and Niculescu-Mizil, 2006; Chapelle and Chang, 2011; Zhang *et al.*, 2017). Despite this, they have seen limited use for predictive transport modelling.

Downloaded by [] on [17/08/18]. Published with permission by the ICE under the CC-BY license

As well as their predictive performance, GBDTs have a number of other properties that make them suitable for mode choice prediction.

- They are highly robust: GBDTs are invariant under monotonic transformations of individual input variables, insensitive to long-tail distributions and outliers, can handle missing values, and are robust to the inclusion of highly correlated and/or irrelevant input variables (Friedman, 2001). This limits the need for data preprocessing and feature selection.
- They can be interpreted using relative feature importance: the contributions towards minimising the cost function of each input feature at each split can be summed across the ensemble to provide the relative importance of each feature (Friedman, 2001).
- They produce well-calibrated choice probabilities when trained using a log-likelihood error term (Niculescu-Mizil and Caruana, 2005).

XGBoost (eXtreme Gradient Boosting; see the study by Chen and Guestrin (2016)) is a cross-platform implementation of GBDT, with an interface available with Python. Zhang *et al*. (2017) show XGBoost has the highest accuracy in 42·25% of 71 benchmarks compared with ten alternative state-of-the-art machine-learning classifiers. This is twice the number of the next-highest performing classifier, Random Forests (another ensemble-of-trees classifier).

2.2.2 Assessing model fit

There are many performance metrics that can be used to compare the fit of different classifiers and define the cost function during model training (Seliya *et al*., 2009). Machine-learning mode choice papers in the literature typically use multiclass classification error (CE, $\mathcal{L}_{\text{ce}}$) as the primary performance metric. This is where a discrete label is generated for each trip by selecting the mode with the highest predicted probability. The proportion of trips where the label is not equal to the mode originally taken (i.e. incorrect classification) is then calculated.

There are two primary limitations to this approach. Firstly, the model is treated as a discrete classifier, predicting a single class with probability 1 and all other classes with probability 0. The primary use case for mode choice models is a simulation, where trips are assigned to each mode with a probability p. As such the class probabilities are a more relevant output for mode choice models, and the performance metric used should reflect this. This limitation applies to other performance metrics calculated from the confusion matrix, including false positive/negative rates and F_1 score.

Second, for imbalanced data sets, where the distribution between classes is not equal, high-classification accuracy can be achieved by a trivial classifier, which always predicts the class with the highest prior. This is known as the accuracy paradox (Bruckhaus, 2007). This is a valid issue for mode choice prediction, where certain modes (e.g. cycling) represent a much lower overall share of trips made.

Negative log-likelihood loss (NLL, $\mathcal{L}_{\text{nll}}$, also known as cross-entropy loss or logarithmic score) is defined as

$$\mathcal{L}_{\text{nll}} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C} y_{i,c} \ln \widehat{p}_{i,c} \quad (2)$$

where $\widehat{p}_{i,c}$ is the predicted probability of taking option c from the choice-set C for choice i and $y_{i,c}$ is the actual choice made (1 for the chosen option and 0 for all others). Minimising $\mathcal{L}_{\text{nll}}$ is equivalent to maximising the likelihood of the observed data given the model. As such, using $\mathcal{L}_{\text{nll}}$ as the cost function during model training results in maximum likelihood estimation.

As log-likelihood loss is a representation of the overall likelihood of the data given the model, it inherently handles imbalanced data sets and treats mode choice as a stochastic process. Log-likelihood therefore reflects the use case for mode choice models and is more suitable than CE/accuracy for assessing mode choice model fit. However, log-likelihood scores are difficult to interpret physically. As such, the expected simulation error (ESE, $\mathcal{L}_{\text{ese}}$) is also provided, defined as

$$\mathcal{L}_{\text{ese}} = 1 - \frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C} y_{i,c}\widehat{p}_{i,c} \quad (3)$$

This is equivalent to the mean predicted probability for the selected mode subtracted from 1 and provides the expected proportion of incorrect mode assignments when using the model for simulation. Unlike CE, this metric also treats mode choice as a stochastic process. However, it still suffers from the accuracy paradox. As such, it should be interpreted as a secondary metric that demonstrates the impacts of using two different models in simulation.

In order to evaluate the real-world performance of a model, it is necessary to validate the performance metric on data unseen during the model training. Cross-validation splits the data into k folds and trains k separate models, with each model is trained on $k - 1$ folds of the data and tested on the remaining fold. The performance is then averaged over the k models. This provides performance estimates that are more robust to sampling noise than those obtained with a single validation sample and allows the performance to be estimated across all the data available for model training and evaluation. Kohavi (1995) investigated cross-validation and bootstrapping for model selection and found that stratified tenfold cross-validation provides the most accurate estimate of real-world performance.

Varma and Simon (2006) show that cross-validation provides a biased estimate of true performance if cross-validation is also used for model optimisation. For large data sets, a separate holdout test sample not seen during model training can be used to provide an estimate of model performance. This is known as holdout sample testing. When predicting future trips, there are likely to be changes in unobserved explanatory variables not present in the input feature

Downloaded by [] on [17/08/18]. Published with permission by the ICE under the CC-BY license

vector. These changes will further reduce model performance when predicting future trips compared with the predicted cross-validation performance. By testing the model on a holdout sample from a future year that has not been seen during model training, the use case for a model predicting future trips can be emulated.

2.2.3 Hyperparameter optimisation

A classification model is an instance of an algorithm fitted to training data. As with other machine-learning classifiers, GBDTs have a set of hyperparameters, which control how the model fits to input data during training. In order to maximise model performance, it is necessary to select hyperparameter values that minimise the generalisation error resulting from the bias–variance trade-off for the specified task (Hastie *et al*., 2008).

Models with high bias underfit to the training data and fail to account for relevant correlations between input features and mode choice that are present in the real-world test data. Models with high variance overfit to noise in the test data and as such will introduce correlations in the model that are not present in other data samples. Both these scenarios will have two primary impacts with respect to mode choice simulation: (*a*) the increased error will result in lower predictive performance, therefore reducing simulation accuracy, and (*b*) there are more likely to be invalid inferences on the relationship between transport and environment decisions, which can be interpreted from the model.

Model generalisation error is highly dependent on chosen hyperparameter values (Friedman, 2001; Hoos *et al*., 2014). As such, it is important to employ a rigorous and unbiased method for hyperparameter selection. The hyperparameters for XGBoost are presented in Table 2.

`n_estimators` and `learning_rate` are the primary hyperparameters of GBDTs. `n_estimators` limits the overall size of the ensemble, and should be set such that performing another round of boosting does not improve the error score on the data. In XGBoost, this can be set dynamically using the `early_stopping_rounds` variable, which specifies the maximum extra trees that can be added to the model without the error score improving. Once this number of rounds is repeated without the score improving, the training process is terminated. When using this method, `n_estimators` is approximately inversely proportional to the `learning_rate`, which controls overfitting by limiting the weighting of each individual tree's contribution to the ensemble. This slows down the training process, requiring a higher `n_estimators`, and therefore more time for the model to converge, resulting, however, in increased overall performance. Friedman (2001) found that small values of learning rate (≤0·1) dramatically improve model performance.

Optimal values for the remaining hyperparameters must be determined experimentally. `max_depth` limits the order of interaction of input features in all trees in the ensemble. `gamma`, `min_child_weight`, `reg_alpha` and `reg_lambda` also all control the bias–variance trade-off by limiting the complexity of individual. `subsample`, `colsample_bytree` and `colsample_bylevel` introduce randomness to make the training procedure more robust to noise. `max_delta_step` is similar to `learning_rate` except it defines an absolute limit (rather than a multiplier) of the gain from each tree in the ensemble.

Sequential model-based optimisation (SMBO) algorithms, otherwise known as Bayesian optimisation algorithms, can learn from previous hyperparameter selections in order to converge to an optimal solution in an iterative process. SMBO has been shown to outperform other methods of hyperparameter selection, including manual search, grid search and random search (Bergstra *et al*., 2014; Snoek *et al*., 2012). The Hyperopt package (Bergstra *et al*., 2015) provides a Python implementation of the tree-structured Parzen estimator (TPE) algorithm (Bergstra *et al*., 2011), which minimises a cost function by drawing values from prior probability distributions. The cost function and the probability distributions in the search space must be specified by the user, as well as the total number of iterations.

2.3 Experimental methodology

GBDT models are used for both the raw data and the choice-set models, implemented using the XGBoost library. The models are trained to predict the actual mode taken in the LTDS (`travel_mode`). The choice probabilities are calculated for all four modes in each model. The models are trained with NLL as the cost function to minimise. The framework for model training is shown in Figure 4.

Table 2. Description of XGBoost hyperparameters

Hyperparameter	Description
`max_depth`	Maximum limit of the depth (number of levels) of each tree in the ensemble
`Learning_rate`	Multiplicative factor of the information contributed by each tree in the ensemble
`n_estimators`	Number of boosting rounds/trees in the ensemble
`gamma`	Threshold for the minimum loss reduction for a split on a leaf node of each tree in the ensemble
`min_child_weight`	Threshold for the minimum weight of each child node after splitting in each tree in the ensemble
`subsample`	Random subsample ratio of observations (rows) in the data set for training each tree
`colsample_bytree`	Random subsample ratio of features (columns) in the data set which can be considered for splitting in each tree
`colsample_bylevel`	Random subsample ratio of features (columns) in the data set which can be considered for splitting at each level of each tree
`reg_alpha/lambda`	Regularisation terms that penalise tree complexity in cost function calculation at each split
`max_delta_step`	Maximum absolute limit of information contributed to ensemble by an individual tree

Downloaded by [] on [17/08/18]. Published with permission by the ICE under the CC-BY license

The data set is preprocessed prior to training the mode choice models. The categorical data are one-hot encoded, so that an n-class categorical variable is replaced with n binary variables. An additional sine and cosine are calculated for cyclical data (`start_time`, `day_of_week` and `travel_month`) to preserve the cyclical ordering. These are included as features alongside their linear representation. This results in a total of 30 features in the raw data model and 44 in the choice-set model. The processed data sets are then divided by year into a training/evaluation set (2012/2013–2013/2014) and a holdout test sample (2014/2015). The training/evaluation set is used for hyperparameter selection and cross-validation performance estimation and to train the final mode choice models. The final models are then tested on the holdout sample.

Hyperparameter selection is performed in the Hyperopt library using a TPE search with 100 iterations for each model. Stratified tenfold cross-validation is used to estimate the loss in each iteration of hyperparameter selection, as recommended by Kohavi (1995). The folds are grouped by household, so that all members from the same household appear in only one fold. This ensures that linked journeys (return journeys or journeys made together by multiple members of a household) do not appear across folds, which would boost apparent cross-validation performance, overstating it compared with true performance on unseen data. The number of boosting rounds in each iteration of hyperparameter selection is set dynamically using the extra trials variable, as explained in Section 2.2.2. Extra rounds are performed until the log-likelihood loss does not improve for 50 consecutive additional rounds, with a total cap of 6000 rounds. The learning rate is set at 0·01 during the hyperparameter search. This allows all 1000 XGBoost models (100 iterations, ten folds for each iteration) to train within a reasonable time on a modern personal computer (<24 h on an eight-core 3·1 Ghz server). The search spaces for the other hyperparameters are given in Table 3. These values are modified from the default search space for XGBoost in the Hyperopt-sklearn library (Komer *et al.*, 2014).

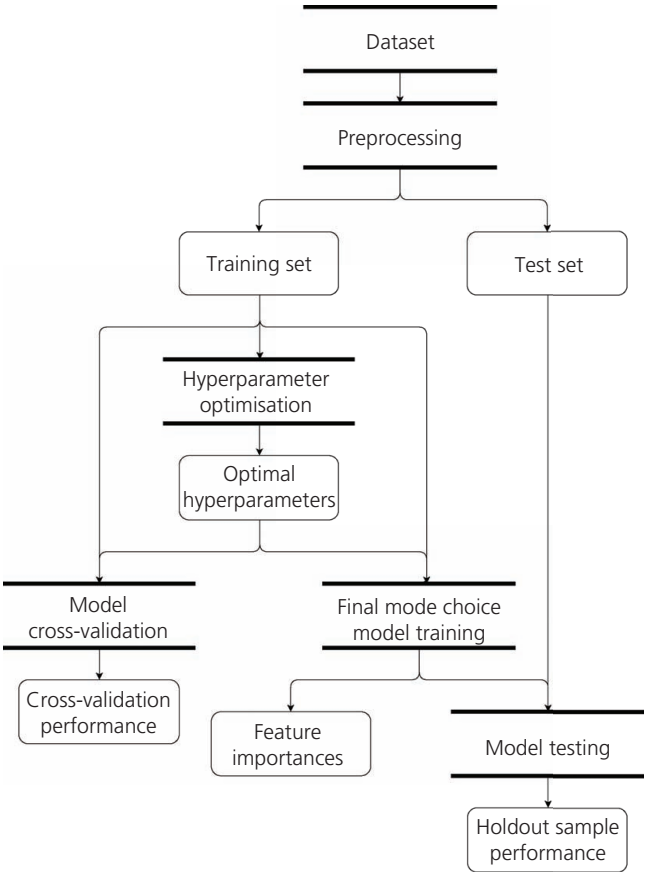

Figure 4. Flow chart of experimental methodology

Cross-validation is then performed for each model with the optimal hyperparameters to provide a cross-validation performance estimate. Again, tenfold cross-validation is used, with the folds grouped by household. The choice probabilities are predicted for each validation fold in cross-validation. The probabilities across all ten folds are used to calculate the log-likelihood loss. The expected simulation accuracy is also provided.

The mode choice models are then trained on all of the training data (2012/2013–2013/2014). The feature importances are calculated from these models using the ensemble gain, which is defined as the total sum of improvement to the log-likelihood loss across all splits in the ensemble. Finally, the models are tested on the future year holdout sample (2014/2015) to assess their real world performance. Both the log-likelihood loss and ESE rate are also provided.

3. Results and discussion

Table 4 shows the average NLL, ESE and CE for each model, for both tenfold cross-validation on the training set and holdout validation of the final models with the test set. The choice-set model achieves better performance than the raw data model across all metrics, for both cross-validation and holdout testing.

The predicted performance in cross-validation is higher than that achieved in holdout-validation for all metrics for both models.

Table 3. XGBoost hyperparameter search spaces

Hyperparameter	Distribution	Range
`max_depth`	Uniform	1:14
`learning_rate`	Fixed	0·01
`n_estimators`	Variable	See text
`gamma`	Log-uniform	0:5
`min_child_weight`	Log-uniform	1:100
`subsample`	Log-uniform	0·5:1
`colsample_bytree`	Log-uniform	0·5:1
`colsample_bylevel`	Log-uniform	0·5:1
`reg_alpha`	Log-uniform	0:1
`reg_lambda`	Log-uniform	1:4
`max_delta_step`	Log-uniform	0:10

Downloaded by [] on [17/08/18]. Published with permission by the ICE under the CC-BY license

Table 4. Performance metrics for each model (lower is better)

Metric	Validation method	Raw data model	Choice-set model
NLL	Cross-validation	0·694	0·635
ESE	Cross-validation	0·390	0·349
CE	Cross-validation	0·276	0·242
NLL	Holdout sample	0·717	0·651
ESE	Holdout sample	0·393	0·351
CE	Holdout sample	0·285	0·252

This demonstrates that cross-validation on the training data overestimates the performance for predicting mode choice for future trips. As such, the holdout sample results are used to indicate real-world performance in the discussion.

GBDTs are non-parametric models. As such, traditional significance tests based on log-likelihood, such as restricted model likelihood ratio tests or Bayesian/Akaike information criterion, cannot be applied. However, as the models are validated on out-of-sample data, overfitting does not need to be accounted for within the test statistics, and, as such, the relative likelihood can be used to compute a confidence interval directly. Over the 26 320 observations in the test data set, the total log-likelihoods are −18 869·7 and −17 140·7 for the raw data model and the choice-set model, respectively. The relative likelihood is therefore e^{-1729}. This demonstrates that adding choice-set information to the data set significantly improves the model's predictive ability at all confidence levels.

The ESE of the choice-set model is 10·6% lower than that of the raw-data model, which illustrates the real-world benefit of adding choice-set information to the feature vector for mode choice models used in transport simulation.

For both models, the CE is significantly lower than the ESE. As such, it initially appears that treating the choice-model as a discrete classifier results in a more accurate representation of mode choice than outputting continuous probabilities for simulation. However, as CE discretises the model predictions, it does not account for uncertainty in model predictions. Crucially, this results in aggregate mode shares, which do not represent those observed in the data. This is critical for aggregate transportation models, where passenger flows are the key model output. Table 5 shows the observed mode shares in the test and train data set, as well as the predicted and expected mode shares from classification and simulation, respectively, for each model. The mode shares from discrete classification differ significantly from the observed mode shares, with the most common mode, driving, being overrepresented in the model results and the less common modes (walking and cycling) being underrepresented. In the case of cycling, there are around 100 times fewer trips predicted by the model in classification than observed in the data. For both models, the expected mode shares from simulation match much more closely to the observed mode shares. These results highlight the issue of using classification accuracy, and other metrics based on the discrete confusion matrix, to evaluate mode choice model performance.

Table 5. Observed aggregate mode shares for the train and test sets, as well as predicted (classification) and expected (simulation) aggregate mode shares for holdout validation set for raw data model and choice-set model

	Walking	Cycling	PT	Driving
		Observed		
Train: %	17·50	2·82	34·88	44·80
Test: %	17·80	3·27	36·10	42·83
		Raw data model		
Classification: %	15·36	0·03	35·46	49·16
Simulation: %	16·96	2·94	36·71	43·39
		Choice-set model		
Classification: %	15·63	0·04	36·51	47·82
Simulation: %	16·98	2·88	36·30	43·84

PT, public transport

Figure 5 shows violin distribution plots of the predicted mode choice probabilities for the selected mode within the holdout test data set for each model – that is, the probability predicted by the model for the mode actually taken by the passenger. The choice-set model distribution is skewed more towards higher probabilities than the raw data model distribution, both for all modes (Figure 5(a)) and for each individual mode (Figure 5(b)). In each case, the median and quartiles are all higher for the choice-set model. This shows that the choice-set model tends to predict the correct mode with higher probability than the raw data model.

Figure 5(b) shows that both models predict highest choice probabilities for the selected mode for driving and public transport trips, followed by walking trips and finally cycling trips. This is a result of the mode shares of trips in the data set, as seen in Table 5.

3.1 Hyperparameter optimisation

Figure 6 shows the cross-validation log-likelihood loss for each iteration of hyperparameter selection for the choice-set and raw-data model. It shows a significant improvement in log-likelihood loss score for both classifiers, with both models converging towards an optimal solution. This improvement is a result of SMBO identifying hyperparameter values, which mininimise the generalisation error from bias and variance.

The primary hyperparameters (`n_estimators` and `learning_rate`) are set on each iteration to achieve the minimum log-likelihood loss, and so all improvement in loss is achieved by optimising the remaining hyperparameters. Both models have found a solution close to the optimum after around 40 iterations. The worst-case guess for the models result in a log-likelihood loss of 0·715 and 0·643 for the raw data model and choice-set model, respectively. This represents a substantial

Downloaded by [] on [17/08/18]. Published with permission by the ICE under the CC-BY license

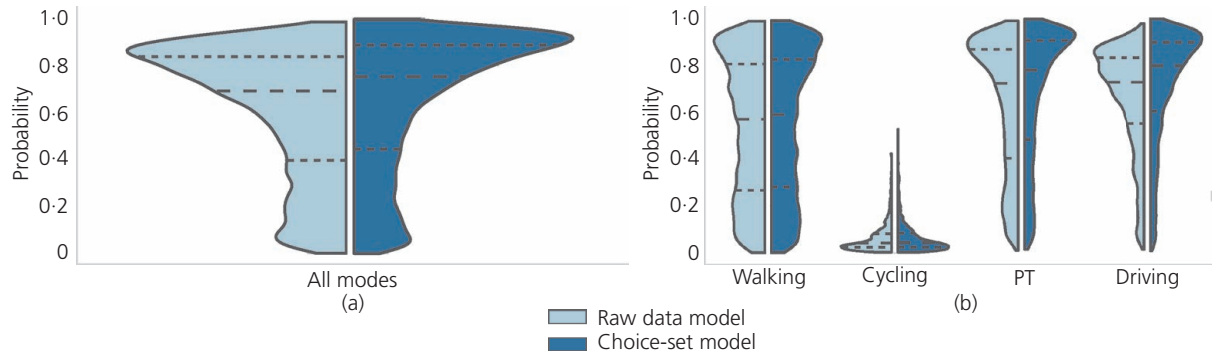

Figure 5. Violin frequency plots of predicted mode choice probabilities for raw data and choice-set modes for (a) all modes combined (predicted probability of chosen mode) and (b) separated by transport mode (predicted probability of chosen mode). Dashed lines mark median and interquartile ranges of each distribution. PT, public transport

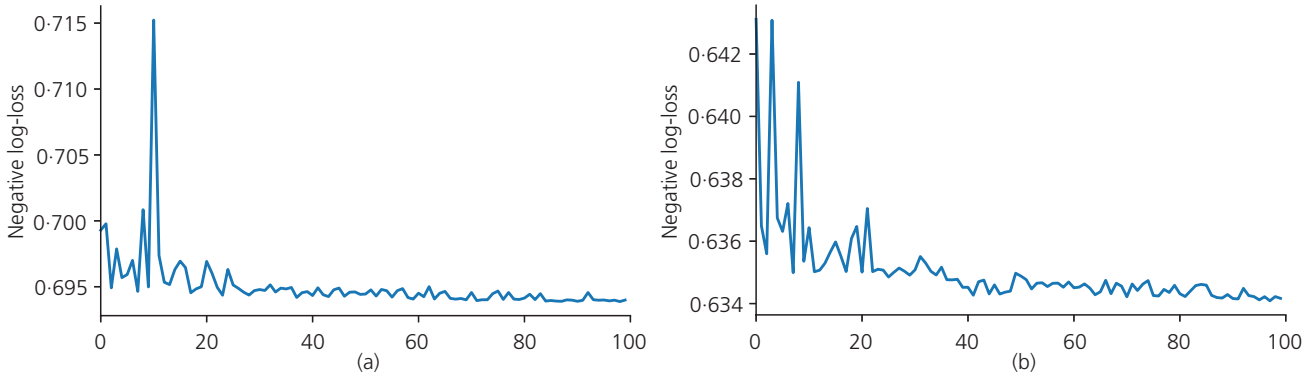

Figure 6. Cross-validation log-likelihood loss at each iteration of hyperparameter optimisation for (a) raw data model and (b) choice-set model

difference in performance compared with the optimal values of 0·694 and 0·634. In particular when considering that the primary hyperparameters, which have the highest impact on predictive performance, are optimised separately for each iteration. This highlights the need for a rigorous hyperparameter selection procedure when comparing machine-learning models.

Table 6 shows the optimised hyperparameters for the raw data and choice-set models. In both cases, the optimal solution is an ensemble with a large number (>1000) of DTs of limited depth (≤6). This depth allows for up to fourth-order interactions of input features in each tree of the raw data model and fifth order in the choice-set model.

3.2 Feature importances

Figure 7 shows the ranked relative feature importances of each feature in each model, calculated using the ensemble gain. Classes of features are grouped by category to form compound features, as shown in Figures 7(c) and 7(d). This includes the categorical features that were one-hot encoded (`faretype`, `fueltype` and `purpose`) and cyclical data, which had a sine and cosine added (`start_time`, `day_of_week` and `travel_month`) for both models, as well as the different subcategories of the public transport duration and driving cost (`dur_pt` and `cost_driving`) in the choice-set model.

In general, the choice-set model has more balanced feature importances than the raw data model. In the raw data model, the two features with the highest importance (`distance` and

Table 6. Optimised hyperparameters for tuned models (stated to four significant figures)

Hyperparameter	Raw data model	Choice-set model
`max_depth`	5	6
`learning_rate`	0·01	0·01
`n_estimators`	1340	1440
`gamma`	0·02379	0·2871
`min_child_weight`	21	31
`subsample`	0·7	0·55
`colsample_bytree`	0·75	0·6
`colsample_bylevel`	0·55	0·8
`reg_alpha`	0·0044 85	0·0068 64
`reg_lambda`	2·369	2·057
`max_delta_step`	6	1

Downloaded by [] on [17/08/18]. Published with permission by the ICE under the CC-BY license

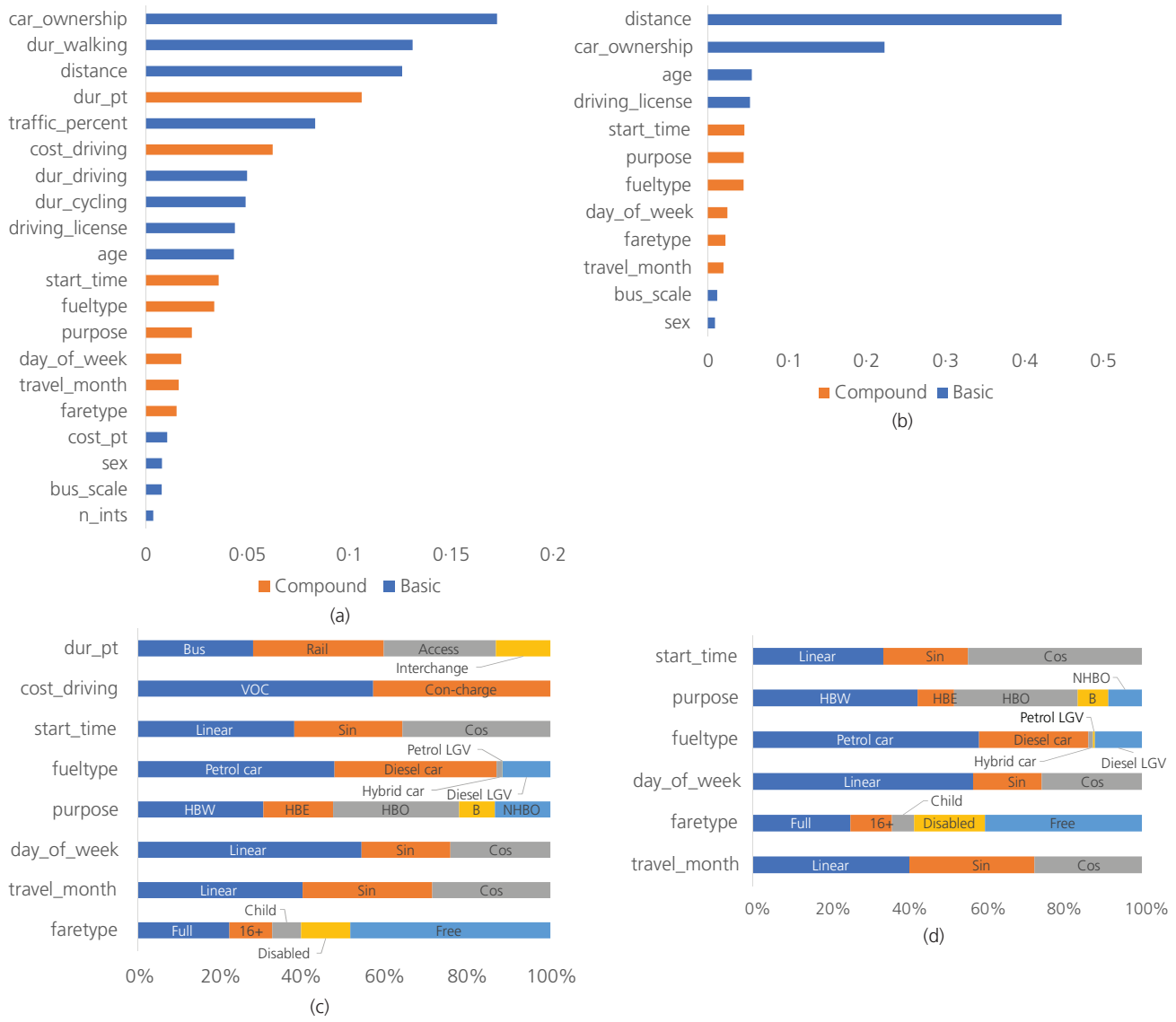

Figure 7. Relative feature importance (ensemble gain) with compound features for (a) choice-set model (relative feature importances) and (b) raw data model (relative feature importances). Subfeature labels and proportions for compound features are given in (c) for the choice-set model (compound features) and (d) for the raw data model (compound features)

`car_ownership`) account for 67% of the total information in the model, whereas in the choice-set model, the two features with the highest importance (`car_ownership` and `dur_walking`) only account for 29%. Figure 7(a) shows that the duration features added to the choice-set model from the directions API have high relative importance, as does `traffic_percent`. Within the public transport duration, the on-board bus, on-board rail and access durations are all of similar importance. The impacts of these additional features on aggregate passenger flows can be investigated when using the choice-set model for transport simulation. In both models, `sex` and `travel month` are of low relative importance, suggesting that there are not strong month-by-month or gender variations in the mode distributions.

4. Predictive framework

The methodology to build the data set and train the mode choice model presented in Sections 2.1.4 and 2.3 form a general framework for building a predictive mode choice model, that can be applied to any major city. The model can then be used to predict the mode choice for a previously unseen journey within a larger transportation simulation model.

The results of this study highlight the advantages of this approach over existing techniques. Adding detailed choice-set information to the feature vector significantly improves the predictive ability of the model, in terms of both overall log-likelihood and ESE. This improves the accuracy of passenger flows generated by the model. Additionally, the model has added flexibility, as the

Downloaded by [] on [17/08/18]. Published with permission by the ICE under the CC-BY license

impacts of the additional features on mode choice can be investigated to determine experimentally the expected passenger flows under changing conditions.

The automated model optimisation and training process allows the mode choice model to be developed without specification and testing of the utility function or feature interactions beforehand. The maximum tree depth of the choice-set model allows for fifth-order interactions of input features in each tree, which is outside the scope of what could be modelled within a RUM.

Overall, the improvements in mode choice prediction open up the opportunity for city-scale transport network simulation at high temporal and spatial resolution, by simulating the choices of individuals on the network in an agent-based approach. These simulations could provide transport network operators with a significantly deeper understanding of passenger flow variations and allow for reliable quantitative analysis of network improvements, policy and regulatory changes and potential disruptions.

A script has been written that automates the process of recreating the choice-set for new trips for mode choice prediction and can be used to batch process a data set of trips from an origin–destination matrix.

(*a*) Predict/collate trip origin, destination, departure time, and travel day-of-week for the trip(s).
(*b*) Generate directions API requests for walking; cycling; public transport; and driving under optimistic, pessimistic and best-guess traffic using information from step (*a*). Use the result of step (*b*) to determine walking, cycling, public transport and driving durations, number of interchanges and traffic variability.
(*c*) Predict/collate fare type, bus-scale and fuel-type. Assume default values if they are not available.
(*d*) Use the cost model with information from steps (*c*) and (*d*) to determine public transport fare, driving VOCs and congestion charge.
(*e*) Predict/collate as many as possible from purpose, driving license, sex, age, car ownership and travel month.
(*f*) Use choice-set model to predict mode choice probabilities from feature vector defined from information from steps (*a*) and (*c*)–(*f*).

4.1 Missing data

GBDTs inherently handle missing values during prediction with an optimal direction for missing values at each split determined during training. As such, the modelling framework is robust to missing values when predicting mode choice.

As a minimum, the trip origin, destination, departure time and travel day-of-week need to be specified. This allows the choice-set to be recreated from the directions API using accurate timetable and traffic information for the public transport and driving trips. Within the cost model, default values are assumed for the cost profile (`fueltype`: average car; `faretype`: full; `bus_scale`: 1).

The remaining attributes in the feature vector are not critical, and a prediction can be obtained even if no values are present in the data.

5. Conclusions

This study presents a new data fusion approach for recreating choice-sets faced by passengers at the time of day of their travel. This methodology has been used in this paper to create a comprehensive, closely tailored travel data set, developed from trip diaries from the LTDS from 2012 to 2015. This data set is used to train GBDT models of passenger mode choice. The models are capable of predicting well-calibrated choice probabilities, which make them useful in understanding people's travel choices whilst accounting for the diversity and variability of human behaviour.

The performance of the models is evaluated using NLL, estimated on the data using *k*-fold cross-validation, with the folds grouped by household. The results demonstrate that adding choice-set information to the model input data significantly improves the mode choice model's predictive capability, in terms of both log-likelihood loss and overall classification accuracy.

SBMO is used for hyperparameter selection for the mode choice models. The results demonstrate the impact of hyperparameter selection on model performance and highlight the requirement for rigorous hyperparameter selection procedures when comparing machine-learning models.

Adding choice-set information to the travel choice model allows the impacts of the additional variables on passenger flows to be determined experimentally. Additionally, the flexible nature of GBDT models allows for complex feature interactions without prior specification of a utility function.

The methodology in this paper has been used to form a general framework for creating new mode choice models for transport simulation for infrastructure investment and operations management, which can be applied to any city with similar travel surveys worldwide. These models present the opportunity for a significantly deeper understanding of passenger flow variations.

Acknowledgements

This research was undertaken as part of Tim Hillel's PhD programme within the Future Infrastructure and Built Environment Centre for Doctoral Training at the University of Cambridge, with supervision by the two co-authors. The Centre for Doctoral Training is funded by the UK Engineering and Physical Sciences Research Council (EP/L016095/1). Additionally, Ying Jin would like to thank the funding he has received from the Cambridge-Berkeley-National University of Singapore University Alliance funded Smart Design project, and that from the EPSRC Managing Air for Green Inner Cities project (EP/N010221/1).

The authors thank Transport of London for their provision of the London Travel Demand Survey data. However, the authors are solely responsible for the views or any remaining errors.

Downloaded by [] on [17/08/18]. Published with permission by the ICE under the CC-BY license

REFERENCES

Ben-Akiva ME and Lerman SR (1985) *Discrete Choice Analysis: Theory and Application to Travel Demand*. MIT Press, Cambridge, MA, USA.

Bergstra JS, Bardenet R, Bengio Y and Kégl B (2011) Algorithms for hyperparameter optimization. *Proceedings of the 24th International Conference on Neural Information Processing Systems, Granada, Spain*, pp. 2546–2554.

Bergstra J, Komer B, Eliasmith C and Warde-Farley D (2014) Preliminary evaluation of hyperopt algorithms on HPOLib. *ICML Workshop on AutoML, Beijing, China.*

Bergstra J, Komer B, Eliasmith C, Yamins D and Cox DD (2015) Hyperopt: a Python library for model selection and hyperparameter optimization. *Computational Science & Discovery* **8(1)**: 014008, https://doi.org/10.1088/1749-4699/8/1/014008.

Biagioni JP, Szczurek PM, Nelson PC and Mohammadian A (2009) Tour-based mode choice modeling: using an ensemble of conditional and unconditional data mining classifiers. *Transportation Research Board 88th Annual Meeting* **312(2009)**: 09–3281.

Brown I and Mues C (2012) An experimental comparison of classification algorithms for imbalanced credit scoring data sets. *Expert Systems with Applications* **39(3)**: 3446–3453, https://doi.org/10.1016/j.eswa.2011.09.033.

Bruckhaus T (2007) The business impact of predictive analytics. In *Knowledge Discovery and Data Mining: Challenges and Realities* (Zhu X and Davidson I (eds)). Igi Global, Hershey, PA, USA, pp. 114–138.

Cantarella GE and de Luca S (2003) Modeling transportation mode choice through artificial neural networks. In *Fourth International Symposium on Uncertainty Modeling and Analysis (ISUMA 2003), College Park, MD, USA, 21–24 September*, pp. 84–90.

Cantarella GE and de Luca S (2005) Multilayer feedforward networks for transportation mode choice analysis: an analysis and a comparison with random utility models. *Transportation Research Part C: Emerging Technologies* **13(2)**: 121–155, https://doi.org/10.1016/j.trc.2005.04.002.

Caruana R and Niculescu-Mizil A (2006) An empirical comparison of supervised learning algorithms. *Proceedings of the 23rd International Conference on Machine Learning, Pittsburgh, Pennsylvania, USA, 25–29 June*, pp. 161–168.

Chapelle O and Chang Y (2011) Yahoo! learning to rank challenge overview. *Proceedings of the 2010 International Conference on Yahoo! Learning to Rank Challenge, Haifa, Israel*, vol. 14, pp. 1–24.

Chen T and Guestrin C (2016) XGBoost: a scalable tree boosting system. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, California, USA*, pp. 785–794.

Department for Business, Energy and Industrial Strategy (2017) *Government Response to the Committee on Climate Change 2017 Report to Parliament – Meeting Carbon Budgets*. Department for Business, Energy and Environmental Strategy, London, UK.

DfT (Department for Transport) (2014) *Transport Analysis Guidance (TAG) Unit A1. 3 – User and Provider Impacts*. DfT, London, UK.

Friedman JH (2001) Greedy function approximation: a gradient boosting machine. *Annals of Statistics* **29(5)**: 1189–1232, https://doi.org/10.1214/aos/1013203451.

Friedman JH (2002) Stochastic gradient boosting. *Computational Statistics & Data Analysis* **38(4)**: 367–378, https://doi.org/10.1016/S0167-9473(01)00065-2.

Greater London Authority (2016) *2016-Based Population Projections*. Greater London Authority, London, UK.

Hagenauer J and Helbich M (2017) A comparative study of machine learning classifiers for modeling travel mode choice. *Expert Systems with Applications* **78**: 273–282, https://doi.org/10.1016/j.eswa.2017.01.057.

Hastie T, Friedman J and Tibshirani R (2008) *The Elements of Statistical Learning*, 2nd edn., Springer Series in Statistics, Springer, New York, NY, USA.

Hensher DA and Ton TT (2000) A comparison of the predictive potential of artificial neural networks and nested logit models for commuter mode choice. *Transportation Research Part E: Logistics and Transportation Review* **36(3)**: 155–172, https://doi.org/10.1016/S1366-5545(99)00030-7.

Hillel T, Guthrie P, Elshafie MZE and Jin Y (2016) Assessing the discrepancies between recorded and commonly assumed journey times in London. In *Transforming the Future of Infrastructure through Smarter Information – Proceedings of the International Conference on Smart Infrastructure and Construction* (Mair RJ, Soga K, Jin Y, Parlikad AK and Schooling JM (eds)). ICE Publishing, London, UK, pp. 759–764.

Hoos H, Ca UBC, Leyton-Brown K and Hutter F (2014) An efficient approach for assessing hyperparameter importance. *Proceedings of the 31st International Conference on International Conference on Machine Learning, Beijing, China*, vol. 32, pp. 754–762.

Jin Y, Williams IN and Shahkarami M (2002) A new land use and transport interaction model for London and its surrounding regions. *Proceedings of the AET European Transport Conference, Cambridge, UK.*

Kohavi R (1995) A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence* (Mellish CS (ed.)). Montréal, Québec, Canada, vol. 2, pp. 1137–1145.

Komer B, Bergstra J and Eliasmith C (2014) Hyperopt-sklearn: automatic hyperparameter configuration for scikit-learn. *Proceedings of the 13th Python in Science Conference, Austin, Texas, USA*, pp. 34–40.

Niculescu-Mizil A and Caruana R (2005) Predicting good probabilities with supervised learning. *Proceedings of the 22nd International Conference on Machine Learning, Bonn, Germany*, pp. 625–632.

Nyquist H (1991) Restricted estimation of generalized linear models. *Journal of the Royal Statistical Society Series C (Applied Statistics)* **40(1)**: 133–141, https:/doi.org/10.2307/2347912.

Omrani H (2015) Predicting travel mode of individuals by machine learning. *Transportation Research Procedia* **10**: 840–849, https://doi.org/10.1016/j.trpro.2015.09.037.

Omrani H, Charif O, Gerber P, Awasthi A and Trigano P (2013) Prediction of individual travel mode with evidential neural network model. *Transportation Research Record* **2399(1)**: 1–8, https://doi.org/10.3141/2399-01.

Rasouli S and Timmermans H (2014) Using ensembles of decision trees to predict transport mode choice decisions: effects on predictive success and uncertainty estimates. *European Journal of Transport and Infrastructure Research* **14(4)**: 412–424.

Seliya N, Khoshgoftaar TM and Hulse JV (2009) A study on the relationships of classifier performance metrics. *21st International Conference on Tools with Artificial Intelligence*. IEEE, Newark, New Jersey, USA, pp. 59–66.

Snoek J, Larochelle H and Adams RP (2012) Practical Bayesian optimization of machine learning algorithms. *Proceedings of the 25th International Conference on Neural Information Processing Systems (NIPS 2012), Lake Tahoe, Nevada*, vol. 2, pp. 2951–2959.

Tang L, Xiong C and Zhang L (2015) Decision tree method for modeling travel mode switching in a dynamic behavioral process. *Transportation Planning and Technology* **38(8)**: 833–85, https://doi.org/10.1080/03081060.2015.1079385.

TfL (Transport for London) (2015) *London Travel Demand Survey (LTDS) – Summary Report*. TfL, London, UK.

Train KE (2009) *Discrete Choice Methods with Simulation*. Cambridge University Press, Cambridge, UK.

Varma S and Simon R (2006) Bias in error estimation when using cross-validation for model selection. *BMC Bioinformatics* **7**: 91, https://doi.org/10.1186/1471-2105-7-91.

Downloaded by [] on [17/08/18]. Published with permission by the ICE under the CC-BY license

Walker J and Ben-Akiva M (2002) Generalized random utility model. *Mathematical Social Sciences* **43(3)**: 303–343, https://doi.org/10.1016/S0165-4896(02)00023-9.

Xian-Yu JC (2011) Travel mode choice analysis using support vector machines. *11th International Conference of Chinese Transportation Professionals (ICCTP), Nanjing, China*, pp. 360–371.

Xie C, Lu J and Parkany E (2003) Work travel mode choice modeling with data mining: decision trees and neural networks. *Transportation Research Record* **1854**: 50–61, https://doi.org/10.3141/1854-06.

Zahid FM and Tutz G (2013) Multinomial logit models with implicit variable selection. *Advances in Data Analysis and Classification* **7(4)**: 393–416, https://doi.org/10.1007/s11634-013-0136-4.

Zhang Y and Xie Y (2008) Travel mode choice modeling with support vector machines. *Transportation Research Record* **2076(1)**: 141–150, https://doi.org/10.3141/2076-16.

Zhang C, Liu C, Zhang X and Almpanidis G (2017) An up-to-date comparison of state-of-the-art classification algorithms. *Expert Systems with Applications* **82**: 128–150, https://doi.org/10.1016/j.eswa.2017.04.003.

Zhao D, Shao C, Li J, Dong C and Liu Y (2010) Travel mode choice modeling based on improved probabilistic neural network. *Seventh International Conference on Traffic and Transportation Studies, Kunming, China*, pp. 685–695.

How can you contribute?

To discuss this paper, please email up to 500 words to the editor at journals@ice.org.uk. Your contribution will be forwarded to the author(s) for a reply and, if considered appropriate by the editorial board, it will be published as discussion in a future issue of the journal.

Proceedings journals rely entirely on contributions from the civil engineering profession (and allied disciplines). Information about how to submit your paper online is available at www.icevirtuallibrary.com/page/authors, where you will also find detailed author guidelines.

Downloaded by [] on [17/08/18]. Published with permission by the ICE under the CC-BY license