Weak teachers: Assisted specification of discrete choice models using ensemble learning

Tim Hillel[1], Michel Bierlaire[1], Mohammed Elshafie[2], and Ying Jin[2]

[1]*École Polytechnique Fédérale de Lausanne, Switzerland*
[2]*University of Cambridge, UK*

Abstract

Mode choice modelling has almost exclusively been tackled using Discrete Choice Models (DCMs). This is in part due to their highly interpretable linear structure, which allows the model to be checked for consistency against established behavioural expectations. However, a key drawback of DCMs is that the utility functions must be specified manually in advance of fitting the model, a process that does not scale well with increasing data complexity.

Machine Learning (ML) is increasingly being investigated as an alternative to DCM for modelling mode choice. Whilst ML automates the decision-making process, requiring no utility functions to be specified, it has a crucial limitation in that the resulting models are difficult to interpret and to check for behavioural consistency.

In order to address the limitations of both ML and discrete choice models, we propose an assisted specification procedure, in which the aggregate structure of a fitted Ensemble Learning (EL) model is used to inform the utility functions in a DCM. The resulting models are found to have greatly improved performance over manually specified DCMs, outperforming all but the highest performing ML classifier.

Introduction & background

Solutions used both in industry and academic research for modelling passenger mode choice rely almost exclusively on Discrete Choice Models (DCMs) based on the random-utility framework. There are many features of DCMs which help explain their ubiquitous usage. Most importantly, the linear utility functions used in DCMs are easy to interpret and ensure a high degree of robustness, as the parameter values can be checked for consistency with established behavioural theory. However, a key drawback of DCMs is that the utility functions must be specified in advance of fitting the model. This is a high-dimensional problem which has no exact solution and cannot be tackled using conventional optimisation techniques. The current approach, *manual specification*, relies on a combination of expert knowledge and guesswork, and is expensive in terms of time, and human and computational resources. This effectively limits the complexity of models used in practice. This limit becomes restrictive when using complex datasets, particularly when considering interactions of input variables.

The availability of larger and more complex datasets describing passenger movements has driven an increasing focus on Machine Learning (ML) as an alternative to DCMs for mode choice prediction. There are several applications of ML algorithms in the choice-modelling literature, including Artificial Neural Networks (ANNs) (Lee, Derrible, and Pereira 2018); Decision Trees (DTs) (Tang, Xiong, and Zhang 2015); Ensemble Learning (EL) (Wang and Ross

2018); and Support Vector Machines (SVMs) (Zhang and Xie 2008). These algorithms automate the decision-making process and as such require no manual specification of utility functions, allowing them to work seamlessly with complex datasets. Furthermore, several studies which investigate ML approaches identify substantially higher Out-Of-Sample (OOS) predictive performance on ML algorithms compared to traditional utility-based DCMs (Wang and Ross 2018; Hagenauer and Helbich 2017). However, ML models have a crucial limitation: the lack of a robust behavioural model. This results in ML models being far less interpretable than DCMs and makes it hard to ensure that the model predictions are consistent with behavioural expectations.

In order to address the current limitations of both the discrete choice and ML models currently used to predict passenger mode choice, we introduce a method for *assisted specification* of DCMs using ML. Our approach involves using the structure of a fitted DT ensemble (i.e. an EL model) to inform the utility specification of a DCM. EL models combine several *weak learners* (DTs) to make predictions, such that each individual DT has low impact on the result (Dietterich 2000). By investigating the structure of the DTs at an aggregate level, we use each DT as a *weak teacher*, providing an understanding of the decision-making process of the model. This can provide valuable insights into how to structure a DCM, including high-order variable interactions and non-linear relationships between input variables and mode choice. We test our approach through a comparative study of DCMs, using both manual and assisted specification, with a suite of several ML classifiers.

Ensemble learning and stochastic gradient boosting

DT-based EL algorithms have several features which make them well-suited to assisted specification of DCMs. Firstly, DTs split data using only the rankings (order) of feature values. As such, DTs are independent of feature scaling, or any monotonic transformation of the features. As well as making the models more robust to varying input data, this provides the EL algorithms with the flexibility to approximate any monotonic non-linear function of the features. By analysing the distribution of the split points for each feature, it is possible to identify these non-linear relationships in the model.

Secondly, the information gain from each split in each DT is calculated during model fitting. It is therefore possible to calculate the relative importance of each feature in the ensemble by summing the gain contributed by all splits using that feature.

Finally, due to their hierarchical structure, feature interactions can be easily observed in DTs by analysing the sequential splits using a set of features. By summing the total gain provided by sequential splits over those features, the relative importance of arbitrary nth order feature interactions can be calculated.

In this study we use Gradient Boosting Decision Trees (GBDT) (Friedman 2001; Chen and Guestrin 2016) as the EL meta-algorithm for the assisted specification. Aside from their best-in-class predictive performance (see experimental results), GBDT have another distinct advantage over other EL meta-algorithms (e.g. Random Forest (RF), Extremely randomised Trees (ET), AdaBoost) for this task; the GBDT algorithm uses sequential regression trees, with each tree predicting the residual of the previous trees in the ensemble. As such, each tree directly evaluates the probability distribution over the travel modes. This contrasts with the other EL meta-algorithms, where each tree attempts to *discretely* predict the correct mode. As such, the structure of the trees in the ensemble are likely to have more relevance to the context of a DCM.

Methodology

The assisted specification approach involves the following steps:

1. Optimise the hyper-parameters of a GBDT model on a (training) dataset
2. Train the optimised GBDT model on the same dataset

3. Investigate the structure of the GBDT model, using it to inform the utility specifications for a DCM
4. Estimate the assisted specification DCM
5. Simplify the DCM by combining parameters where necessary

Steps 3-5 are applied iteratively, with complexity sequentially added, before estimating and simplifying the DCM after each step.

To validate the approach, we compare the predictive performance of the assisted specification DCM with a manual specification DCM, as well as several ML algorithms, using a dataset of historic trips. The following sections describe in turn the dataset, the modelling framework, and how the models are compared.

Dataset

The dataset from Hillel, Elshafie, and Jin (2018) is used for the analysis. It is publicly available online as supporting material of the paper.

The dataset combines individual historic trip records trajectories alongside their corresponding mode-alternatives from an online directions service, and precise estimates of public transport fares and Vehicle Operating Costs (VOCs). The dataset considers four modes: walking, cycling, public transport, and driving. The dataset contains 12 socio-economic/demographic covariates, many of which are continuous or have several classes, and 13 alternative-specific variables.

The first two years of data are used as the training set. The final year of data is used as a holdout test set. This represents a realistic transport simulation task of predicting a future year's trips.

Modelling framework

DCM with manual specification

The baseline DCM is a Multinomial Logit (MNL) model optimised using a manual search using PythonBiogeme (Bierlaire 2016).

The complexity of the dataset means there is very high dimensionality when considering interaction of the variables in the utility specifications. This makes finding optimal variable interactions infeasible using manual specification. As such, the baseline MNL includes only first order interaction of input variables with the utilities. Higher order interactions are instead investigated using the assisted specification.

Even without considering variable interactions, there are still infinite possibilities for how to include the socio-economic/demographic variables in the utility specifications. Five of these variables are either continuous or have many (>5) possible discrete values (trip distance, age, start time, day of week, travel month). To address this, these variables are either binned and included as dummy variables, or included directly in the utility functions as a continuous variable with separate parameters for each mode.

Three bins are used for age, based on the commonly used groupings of child (<18), adult (18-64), and pensioner (65+). The Transport for London (TfL) fare periods are used to define four departure time bins: AM peak (06:30-09:29), inter-peak (09:30-16:29), peak (16:30-19:29), and night (19:30-06:29). The day of the week is grouped into work-days (Monday-Friday), Saturday, and Sunday. The travel month is grouped into winter (December-February), and all other months. Finally, trip distance is included in all models as a continuous variable, with a separate parameter for each mode (fixed to zero for walking). Model performance could be improved further by using more complex strategies for the binned variables, e.g. piecewise linear splines, though this is not explored here.

To conduct the manual search, the full complex initial model is hypothesised, by including all possible parameters for included variables. The model is then simplified sequentially by applying restrictions based on Wald tests of the parameters, one parameter at a time. The utility functions are specified and evaluated using only the training set.

GBDT model

The eXtreme Gradient Boosting (XGB) algorithm (Chen and Guestrin 2016) is used for the GBDT model. We use Sequential Model-Based Optimisation (SMBO) with the training data to select the model hyper-parameters for the GBDT algorithm. The optimisation is performed for 100 iterations of the Tree-structure Parzen Estimator (TPE) algorithm, using the hyperopt library (Bergstra et al. 2015). 10-fold Cross-Validation (CV) is performed grouped by household to estimate model performance for each iteration. The search space for each hyper-parameter is derived from values given by Komer, Bergstra, and Eliasmith (2014). Boosting rounds are performed until the performance does not increase for 50 consecutive iterations. The optimal hyper-parameters are deemed to be those one which achieve the lowest average Cross-Entropy Loss (CEL) over the 10 CV folds.

DCM with assisted specification

An MNL model estimated in PythonBiogeme is used for the assisted specification DCM. Information is extracted from the GBDT ensemble using the*Xgbfir* python library (Kostenko 2018). The library extracts and analyses each DT in the fitted ensemble, identifying the split points and total gain for each feature. The hierarchical structure of the splits in the tree is also analysed to identify second, third, and higher order feature interactions, and rank them according to their importance.

The distribution plots of the split points for the continuous covariate features are analysed to identify underlying non-linear interactions of input features with mode choice. The relative importance of second and third order interactions of input features in the GBDT model are used to identify first and second order interactions of socio-economic/demographic covariates with alternative-specific variables.

Each modification is applied sequentially to the DCM. The model for each iteration of assisted specification is simplified sequentially using Wald tests to identify parameters which should be combined.

Other ML models

For reference, we compare the performance of the two DCMs and the GBDT model with five further machine learning classification algorithms: ANN, Logistic Regression (LR), ET, RF, and SVM. We direct the reader to Hastie, Friedman, and Tibshirani (2008) for an overview of each algorithm. The method used to optimise the models is the same as that used for the GBDT model.

Model comparison

After being fit on the training data, the OOS predictive performance of all models is evaluated on the the holdout set. The models are evaluated on their CEL, where a score closer to zero represents a better fit. For comparison with the results of previous studies, the Discrete Classification Accuracy (DCA) is also provided, where a score closer to one represents a better fit.

As well as holdout validation, the models are also tested using 100 folds of OOS bootstrap validation across the full dataset. This allows distributions of the expected performance to be estimated.

Results & discussion

Model training/specification

Manual specification MNL

The final model parameters for the baseline MNL with manual specification are shown in table 1. The model contains 54 parameters, all of which are significant, and have signs and magnitudes consistent with expected behavioural theory.

GBDT model

The optimised hyper-parameters for the GBDT model are shown in table 2. The ensemble contains 1472 trees, with a maximum depth of six. This means each tree can model up to fifth-order feature interactions.

Assisted specification MNL

The most important covariates in the GBDT model are (in order): (i) vehicle ownership, (ii) distance, (iii) driving license, and (iv) age. Of these, two are categorical: vehicle ownership (no vehicles, less than one vehicle per adult, one or more vehicles per adult) and driving licence (yes, no); and two are continuous: distance and age.

The categorical variables are already fully interacted directly with the utilities in the baseline DCM. However, in the manual specification, distance is simply included linearly as a continuous variable, and age is included as a binned variable using a-priori bins.

Figure 1 shows the Kernel Density Estimation (KDE) distribution of the binary split values for the straight-line trip distance. The distribution is heavily skewed towards shorter trips, with a long tail towards longer trips. This shape is characteristic of a log-normal distribution and suggests trip choice probabilities are related to log-distance.

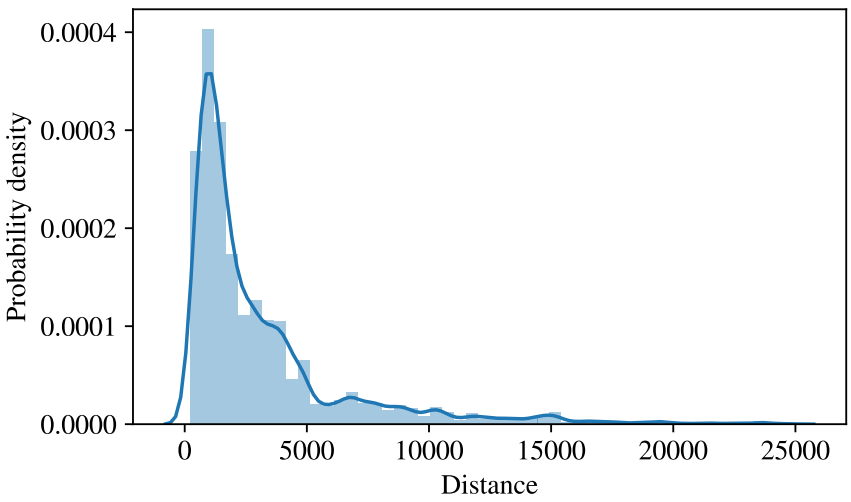

Figure 1: Histogram and KDE plot of split values for straight-line trip distance across all trees in GBDT classifier.

Table 1: Estimation report for baseline MNL with manual specification.

Parameter number	Description	Coeff. estimate	Robust Asympt. std. error	t-stat	p-value
1	ASC_CYCLING	-4.95	0.117	-42.15	0.00
2	ASC_DRIVING	-4.24	0.0838	-50.53	0.00
3	ASC_PT	-3.03	0.0793	-38.24	0.00
4	B_AGE_CHILD_DRIVING	0.774	0.0548	14.12	0.00
5	B_AGE_CHILD_PT	0.245	0.0531	4.61	0.00
6	B_AGE_PENSIONER_CYCLING	-0.447	0.132	-3.37	0.00
7	B_AGE_PENSIONER_DRIVING	0.541	0.0513	10.54	0.00
8	B_AGE_PENSIONER_PT	0.834	0.0537	15.54	0.00
9	B_COST_DRIVE	-0.118	0.00628	-18.79	0.00
10	B_COST_PT	-0.0925	0.0146	-6.33	0.00
11	B_DAY_SAT_CYCLING	-0.338	0.0990	-3.42	0.00
12	B_DAY_SAT_PT	0.204	0.0494	4.13	0.00
13	B_DAY_WEEK_DRIVING	-0.163	0.0370	-4.41	0.00
14	B_DAY_WEEK_PT	0.346	0.0470	7.37	0.00
15	B_DEPARTURE_INTERPEAK_CYCLING	-0.221	0.0751	-2.95	0.00
16	B_DEPARTURE_INTERPEAK_DRIVING	-0.127	0.0340	-3.74	0.00
17	B_DEPARTURE_PMPEAK_CYCLING	0.347	0.0701	4.95	0.00
18	B_DEPARTURE_PMPEAK_DRIVING	0.431	0.0375	11.48	0.00
19	B_DEPARTURE_PMPEAK_PT	0.162	0.0375	4.31	0.00
20	B_DISTANCE_CYCLING	0.405	0.107	3.80	0.00
21	B_DISTANCE_DRIVING	0.654	0.0971	6.74	0.00
22	B_DISTANCE_PT	0.656	0.0974	6.74	0.00
23	B_DRIVINGLICENCE_CYCLING	0.674	0.0702	9.60	0.00
24	B_DRIVINGLICENCE_DRIVING	1.06	0.0451	23.57	0.00
25	B_DRIVINGLICENCE_PT	-0.298	0.0407	-7.31	0.00
26	B_FEMALE_CYCLING	-0.810	0.0636	-12.73	0.00
27	B_FEMALE_DRIVING	0.191	0.0321	5.96	0.00
28	B_FEMALE_PT	0.217	0.0325	6.67	0.00
29	B_PURPOSE_B_CYCLING	1.09	0.131	8.36	0.00
30	B_PURPOSE_B_DRIVING	0.418	0.0860	4.85	0.00
31	B_PURPOSE_B_PT	0.778	0.0916	8.49	0.00
32	B_PURPOSE_HBE_DRIVING	-0.553	0.0493	-11.21	0.00
33	B_PURPOSE_HBE_PT	0.372	0.0541	6.88	0.00
34	B_PURPOSE_HBO_CYCLING	0.400	0.0805	4.97	0.00
35	B_PURPOSE_HBO_PT	0.265	0.0370	7.17	0.00
36	B_PURPOSE_HBW_CYCLING	0.765	0.100	7.63	0.00
37	B_PURPOSE_HBW_DRIVING	-0.686	0.0634	-10.81	0.00
38	B_PURPOSE_HBW_PT	0.279	0.0680	4.11	0.00
39	B_TIME_CYCLING	-2.45	0.612	-4.00	0.00
40	B_TIME_DRIVING	-4.32	0.200	-21.56	0.00
41	B_TIME_ACCESS_PT	-4.41	0.160	-27.62	0.00
42	B_TIME_BUS_PT	-1.92	0.117	-16.50	0.00
43	B_TIME_INTERCHANGEWAIT_PT	-5.02	0.317	-15.83	0.00
44	B_TIME_INTERCHANGEWALK_PT	-2.89	1.01	-2.85	0.00
45	B_TIME_RAIL_PT	-1.51	0.220	-6.88	0.00
46	B_TIME_WALKING	-5.97	0.383	-15.58	0.00
47	B_TRAFFICVARIABILITY_DRIVING	-2.56	0.0846	-30.25	0.00
48	B_VEHICLEOWNERSHIP_1_DRIVING	2.17	0.0453	47.93	0.00
49	B_VEHICLEOWNERSHIP_1_PT	-0.413	0.0380	-10.86	0.00
50	B_VEHICLEOWNERSHIP_2_DRIVING	2.57	0.0509	50.57	0.00
51	B_VEHICLEOWNERSHIP_2_PT	-0.615	0.0485	-12.67	0.00
52	B_VEHICLEOWNERSHIP_CYCLING	-0.138	0.0657	-2.10	0.04
53	B_WINTER_CYCLING	-0.329	0.0817	-4.02	0.00
54	B_WINTER_DRIVING	0.123	0.0315	3.91	0.00

Summary statistics

Number of observations = 54766

Number of excluded observations = 0

Number of estimated parameters = 54

$$\mathcal{L}(\beta_0) = -75921.797$$

$$\mathcal{L}(\hat{\beta}) = -37281.766$$

$$-2[\mathcal{L}(\beta_0) - \mathcal{L}(\hat{\beta})] = 77280.062$$

$$\rho^2 = 0.509$$

$$\bar{\rho}^2 = 0.508$$

Table 2: Optimised hyper-parameter values for GBDT model

Hyper-parameter	**Range**
`max_depth`	6
`gamma`	5.439×10^{-3}
`min_child_weight`	36
`max_delta_step`	4
`subsample`	0.65
`colsample_bytree`	0.65
`colsample_bylevel`	0.55
`reg_alpha`	4.823×10^{-4}
`reg_lambda`	2.572
`learning_rate`	0.01
`n_estimators`	1472

The same plot is generated for the natural logarithm of the split points for trip distance, shown in fig. 2. The distribution is approximately symmetrical, reinforcing the suggestion that there is a relationship between the log distance and mode choice.

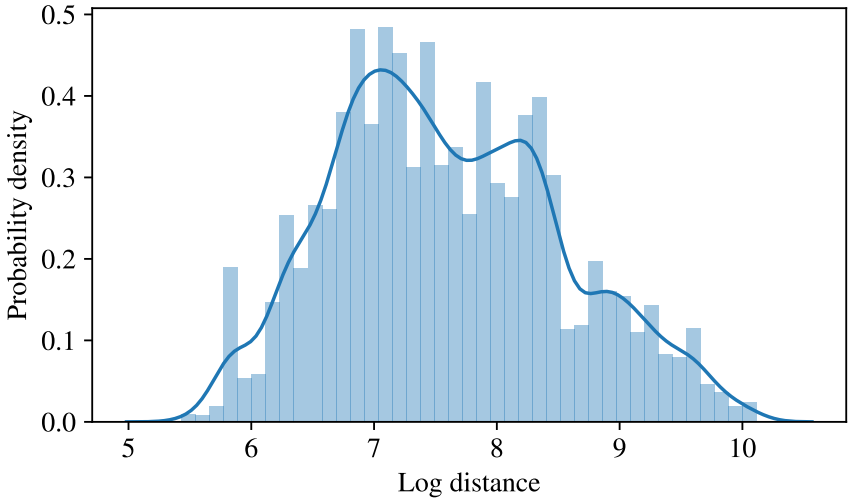

Figure 2: Distribution of split values for natural logarithm of straight-line trip distance across all trees in GBDT classifier.

Based on the distributions in figs. 1 and 2, the log-distance is added (alongside the distance) to the utility specifications for each mode. The resulting simplified model is referred to as the *log-distance* model.

Figure 3 shows a bar chart of the number of splits at each age. Unlike the distribution for the distance splits, there is not a strong skew in the data. Instead, there are three clear modal peaks, at 11.5, 31.5, and 66.5 years. These modal split values define four new heuristic bins to define dummy variables: (i) child (<12), (ii) young adult (12-31), (iii) mature adult (32-66), and (iv) pensioner (67+). These heuristic bins are added to the simplified *log-distance* model specification, in place of the a-priori bins. The resulting simplified model referred to as the *heuristic age* model.

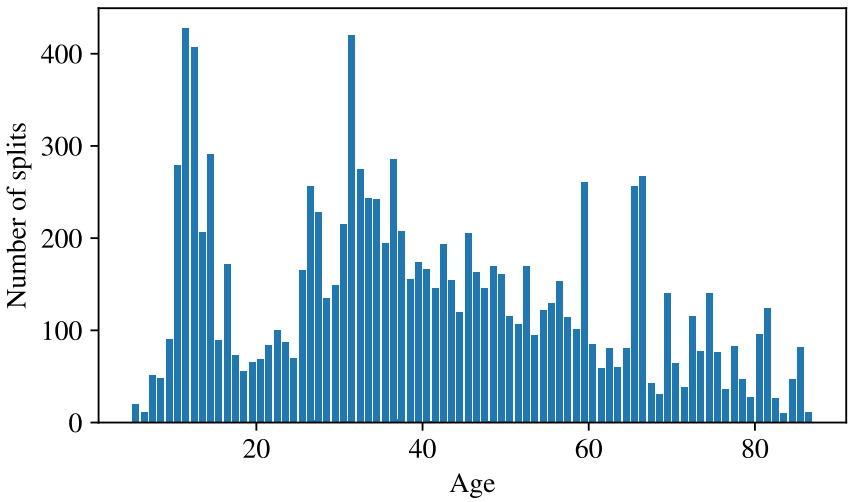

Figure 3: Bar chart of number of splits at each age value across all trees in GBDT classifier.

The relative importance of the second and third order interactions in the GBDT are investigated to identify variable interactions.

Of the 10 most important second order feature interactions, six include vehicle ownership (alongside traffic variability, walking duration, congestion charge, driving duration, straight line distance, and driving licence ownership). This implies that vehicle ownership should be interacted with the other variables in the utility specifications in the assisted specification DCM.

The vehicle ownership covariate is therefore full interacted with all other parameters in the *heuristic age* model, replacing each parameter with three new parameters, with the following suffixes:

- NVO - no vehicles in household
- VO1 - less than one vehicle per adult
- VO2 - one or more vehicle per adult

Where combined during model simplification, the parameters are denoted NVO1 (no cars or less than one car per adult) and VO (household with at least one car). The simplified model is referred to as the *vehicle ownership* model.

Finally, the most important third order interaction which contains at least two socio-economic covariates is vehicle ownership/driving licence/traffic variability. Vehicle ownership/driving licence is also the most important second order feature interaction between socio-economic covariates. As such, the driving licence variable is fully interacted with the parameters from the *vehicle ownership* model, so that each parameter is replaced with two parameters, with the suffixes DL1 and DL0 for having and not having a driving license respectively. The resulting simplified model is referred to as the *full assisted specification* model.

Table 3 shows the estimation results for the four sequential assisted specifications, alongside the baseline manual specification. All the modifications in the assisted specification substantially improve the log-likelihood and Akaike Information Criterion (AIC) during model estimation.

The estimation report for the full assisted specification DCM is given in table 4. The final model has 100 parameters, all of which are significant. One parameter has an unexpected sign; B_COST_FUEL_NCO_DL1 is positive, suggesting that driving licence holders with no vehicles

Table 3: Estimation results for assisted specification MNLs.

Model	**Params**	**Fit time**	**LL**	**AIC**
Manual specification	54	01:40	-37281.77	74671.53
Log-distance	58	02:15	-36513.90	73143.80
Heuristic age	60	04:10	-35976.02	72072.04
Vehicle ownership	87	13:01	-35403.55	70981.10
Full assisted specification	100	42:32	-35082.62	70365.24

in the household have increased utility for driving for increasing fuel costs. As members of households with no available vehicles, driving trips made by this group must either be made by taxi or as a passenger in another household's car. As such, their utility of driving is not directly affected by fuel cost. It is therefore reasonable to remove this parameter from the model.

Table 4: Estimation report for assisted specification MNL.

Parameter number	Description	Coeff. estimate	Robust Asympt. std. error	t-stat	p-value
1	ASC_CYCLING	-13.5	0.662	-20.37	0.00
2	ASC_DRIVING_DL0	-16.9	0.626	-26.92	0.00
3	ASC_DRIVING_DL1	-14.0	0.515	-27.12	0.00
4	ASC_PT_VO	-23.3	0.581	-40.10	0.00
5	ASC_PT_NVO	-19.4	0.580	-33.42	0.00
6	B_AGE_CHILD_CYCLING_NVO_DL0	-1.21	0.365	-3.32	0.00
7	B_AGE_CHILD_DRIVING_NVO_DL0	0.706	0.129	5.48	0.00
8	B_AGE_CHILD_DRIVING_VO1_DL0	1.70	0.0947	17.91	0.00
9	B_AGE_CHILD_DRIVING_VO2_DL0	2.18	0.0918	23.79	0.00
10	B_AGE_CHILD_PT_VO1_DL0	-0.344	0.107	-3.21	0.00
11	B_AGE_MATUREADULT_CYCLING_DL1	1.10	0.104	10.53	0.00
12	B_AGE_MATUREADULT_DRIVING_NVO_DL0	-1.75	0.122	-14.29	0.00
13	B_AGE_MATUREADULT_DRIVING_NVO_DL1	-0.722	0.210	-3.44	0.00
14	B_AGE_MATUREADULT_DRIVING_VO2_DL1	0.520	0.0601	8.65	0.00
15	B_AGE_MATUREADULT_PT_NVO	-0.918	0.0938	-9.80	0.00
16	B_AGE_MATUREADULT_PT_VO1_DL1	-0.374	0.0583	-6.42	0.00
17	B_AGE_YOUNGADULT_CYCLING_DL0	0.452	0.107	4.24	0.00
18	B_AGE_YOUNGADULT_CYCLING_DL1	-0.363	0.0817	-4.45	0.00
19	B_AGE_YOUNGADULT_DRIVING_NVO2_DL1	-0.137	0.0624	-2.19	0.03
20	B_AGE_YOUNGADULT_DRIVING_VO1_DL0	-0.287	0.0642	-4.47	0.00
21	B_AGE_YOUNGADULT_DRIVING_VO1_DL1	-0.534	0.0509	-10.48	0.00
22	B_AGE_YOUNGADULT_PT_NVO_DL0	-0.203	0.0599	-3.40	0.00
23	B_AGE_YOUNGADULT_PT_NVO_DL1	-0.364	0.0781	-4.67	0.00
24	B_VOST_VONCHARGE_VO	-0.118	0.00641	-18.35	0.00
25	B_VOST_FUEL_NVO_DL1	0.851	0.158	5.39	0.00
26	B_VOST_FUEL_VO2_DL1	-0.268	0.105	-2.56	0.01
27	B_VOST_PT_NVO	-0.271	0.0266	-10.20	0.00
28	B_DAY_SAT_CYCLING_NVO_DL1	-0.816	0.224	-3.65	0.00
29	B_DAY_SAT_PT_VO	0.313	0.0655	4.77	0.00
30	B_DAY_WEEK_DRIVING_NVO1_DL0	-0.396	0.0519	-7.62	0.00
31	B_DAY_WEEK_DRIVING_NVO1_DL1	-0.192	0.0454	-4.21	0.00
32	B_DAY_WEEK_PT_VO	0.510	0.0566	9.01	0.00
33	B_DEPARTURE_INTERPEAK_DRIVING_VO2_DL0	-0.264	0.0872	-3.03	0.00
34	B_DEPARTURE_INTERPEAK_DRIVING_VO2_DL1	-0.124	0.0550	-2.25	0.02
35	B_DEPARTURE_PMPEAK_CYCLING_NVO1	0.288	0.0650	4.43	0.00
36	B_DEPARTURE_PMPEAK_DRIVING_DL0	0.809	0.0513	15.75	0.00
37	B_DEPARTURE_PMPEAK_DRIVING_DL1	0.360	0.0417	8.62	0.00
38	B_DEPARTURE_PMPEAK_PT_VO	0.257	0.0460	5.60	0.00
39	B_DISTANCE_CYCLING_VO_DL0	-0.896	0.139	-6.44	0.00
40	B_DISTANCE_CYCLING_DL1	-1.47	0.142	-10.33	0.00
41	B_DISTANCE_CYCLING_NVO_DL0	-0.819	0.136	-6.03	0.00
42	B_DISTANCE_DRIVING_VO_DL0	-0.671	0.139	-4.84	0.00
43	B_DISTANCE_DRIVING_DL1	-1.27	0.141	-9.01	0.00
44	B_DISTANCE_DRIVING_NVO_DL0	-0.517	0.137	-3.77	0.00
45	B_DISTANCE_PT_VO_DL0	-0.751	0.138	-5.45	0.00
46	B_DISTANCE_PT_DL1	-1.34	0.141	-9.46	0.00

Continued on next page

Table 4 – continued from previous page

Parameter number	Description	Coeff. estimate	Robust Asympt. std. error	t-stat	p-value
47	B_DISTANCE_PT_NVO_DL0	-0.658	0.135	-4.88	0.00
48	B_FEMALE_CYCLING	-0.853	0.0593	-14.38	0.00
49	B_FEMALE_DRIVING_DL0	0.458	0.0541	8.46	0.00
50	B_FEMALE_PT_DL0	0.261	0.0466	5.60	0.00
51	B_FEMALE_PT_DL1	0.140	0.0335	4.17	0.00
52	B_LOGDISTANCE_CYCLING	1.44	0.0995	14.45	0.00
53	B_LOGDISTANCE_DRIVING_VO_DL0	2.33	0.0961	24.26	0.00
54	B_LOGDISTANCE_DRIVING_VO_DL1	2.21	0.0818	26.99	0.00
55	B_LOGDISTANCE_DRIVING_NVO_DL0	2.33	0.0977	23.87	0.00
56	B_LOGDISTANCE_DRIVING_NVO_DL1	1.86	0.0863	21.59	0.00
57	B_LOGDISTANCE_PT_VO	3.12	0.0893	34.91	0.00
58	B_LOGDISTANCE_PT_NVO	2.88	0.0909	31.71	0.00
59	B_PURPOSE_B_CYCLING	0.871	0.101	8.65	0.00
60	B_PURPOSE_B_DRIVING_VO2	0.729	0.109	6.69	0.00
61	B_PURPOSE_B_PT_VO_DL0	0.995	0.144	6.89	0.00
62	B_PURPOSE_B_PT_VO_DL1	0.523	0.0783	6.68	0.00
63	B_PURPOSE_HBE_DRIVING_VO_DL0	-0.737	0.0696	-10.58	0.00
64	B_PURPOSE_HBE_DRIVING_NVO	-1.23	0.166	-7.41	0.00
65	B_PURPOSE_HBE_DRIVING_VO1_DL1	-0.299	0.0900	-3.32	0.00
66	B_PURPOSE_HBE_PT_DL0	0.355	0.0594	5.97	0.00
67	B_PURPOSE_HBE_PT_VO2_DL1	-0.492	0.174	-2.83	0.00
68	B_PURPOSE_HBO_CYCLING_VO2	0.611	0.0970	6.30	0.00
69	B_PURPOSE_HBO_DRIVING_VO_DL1	-0.401	0.0419	-9.56	0.00
70	B_PURPOSE_HBO_PT_NVO_DL0	0.196	0.0578	3.38	0.00
71	B_PURPOSE_HBW_CYCLING_NVO1	0.723	0.0728	9.94	0.00
72	B_PURPOSE_HBW_CYCLING_VO2_DL1	1.55	0.144	10.72	0.00
73	B_PURPOSE_HBW_DRIVING_NVO	-1.60	0.145	-11.01	0.00
74	B_PURPOSE_HBW_DRIVING_VO1_DL0	-1.32	0.114	-11.53	0.00
75	B_PURPOSE_HBW_DRIVING_VO1_DL1	-0.842	0.0612	-13.76	0.00
76	B_PURPOSE_HBW_PT_VO2_DL0	1.33	0.322	4.14	0.00
77	B_PURPOSE_HBW_PT_VO2_DL1	0.315	0.0858	3.67	0.00
78	B_TIME_ACCESS_PT_VO	-5.57	0.197	-28.25	0.00
79	B_TIME_ACCESS_PT_NVO_DL0	-6.56	0.359	-18.26	0.00
80	B_TIME_ACCESS_PT_NVO_DL1	-6.55	0.449	-14.57	0.00
81	B_TIME_BUS_PT_NVO1_DL0	-2.12	0.160	-13.25	0.00
82	B_TIME_BUS_PT_NVO1_DL1	-2.94	0.147	-19.97	0.00
83	B_TIME_BUS_PT_VO2_DL0	-2.80	0.283	-9.89	0.00
84	B_TIME_BUS_PT_VO2_DL1	-4.00	0.203	-19.67	0.00
85	B_TIME_DRIVING_NVO	-5.37	0.504	-10.66	0.00
86	B_TIME_DRIVING_VO1_DL0	-3.86	0.431	-8.96	0.00
87	B_TIME_DRIVING_VO1_DL1	-4.10	0.262	-15.63	0.00
88	B_TIME_DRIVING_VO2_DL0	-3.98	0.494	-8.05	0.00
89	B_TIME_DRIVING_VO2_DL1	-2.63	0.353	-7.45	0.00
90	B_TIME_INTERCHANGEWAIT_PT_VO	-6.36	0.336	-18.92	0.00
91	B_TIME_INTERCHANGEWAIT_PT_NVO	-4.06	0.618	-6.57	0.00
92	B_TIME_INTERCHANGEWALK_PT_VO1	-4.69	1.25	-3.77	0.00
93	B_TIME_RAIL	-1.86	0.209	-8.87	0.00
94	B_TIME_WALKING_DL0	-4.99	0.525	-9.50	0.00
95	B_TIME_WALKING_DL1	-7.14	0.567	-12.60	0.00
96	B_TRAFFICVARIABILITY_DL0	-1.63	0.151	-10.78	0.00
97	B_TRAFFICVARIABILITY_NVO1_DL1	-2.36	0.125	-18.94	0.00
98	B_TRAFFICVARIABILITY_VO2_DL1	-3.24	0.156	-20.75	0.00
99	B_WINTER_CYCLING_VO_DL1	-0.417	0.121	-3.44	0.00
100	B_WINTER_DRIVING_VO	0.163	0.0355	4.59	0.00

Summary statistics
Number of observations = 54766
Number of excluded observations = 0
Number of estimated parameters = 100

$$\begin{aligned} \mathcal{L}(\beta_0) &= -75921.797 \\ \mathcal{L}(\hat{\beta}) &= -35082.618 \\ -2[\mathcal{L}(\beta_0) - \mathcal{L}(\hat{\beta})] &= 81678.357 \\ \rho^2 &= 0.538 \\ \bar{\rho}^2 &= 0.537 \end{aligned}$$

Comparison with ML models

The holdout validation results for all models are shown in table 5. The GBDT model performs best, achieving both the lowest CEL and highest DCA. The manual specification DCM is the

lowest performing model. The assisted specification DCM is the second highest performing model, outperforming all ML classifiers except the GBDT model. This shows the model performance has been substantially improved using assisted specification.

Table 5: Holdout-validation results for optimised ML classifiers.

	Score		Rank	
Model	**CEL**	**DCA**	**CEL**	**DCA**
MNL - Manual	0.7012	0.7297	8	8
MNL - Assisted	0.6702	0.7434	2	2
GBDT	0.6511	0.7484	1	1
LR	0.6931	0.7356	7	5
FFNN	0.6881	0.7347	5	6
RF	0.6769	0.7416	3	3
ET	0.6798	0.7412	4	4
SVM	0.6920	0.7316	6	7

The distributions of the CEL estimated from the 100 iterations of OOS bootstrap validation for each model are shown in fig. 4. This figure highlights the significant jump in performance from the manual specification MNL (MS-MNL), and the assisted specification MNL (AS-MNL). Using a paired t-test with the bootstrap results, the assisted specification MNL is shown to significantly outperform all other classifiers except GBDT at the $2.5\,\%$ significance level.

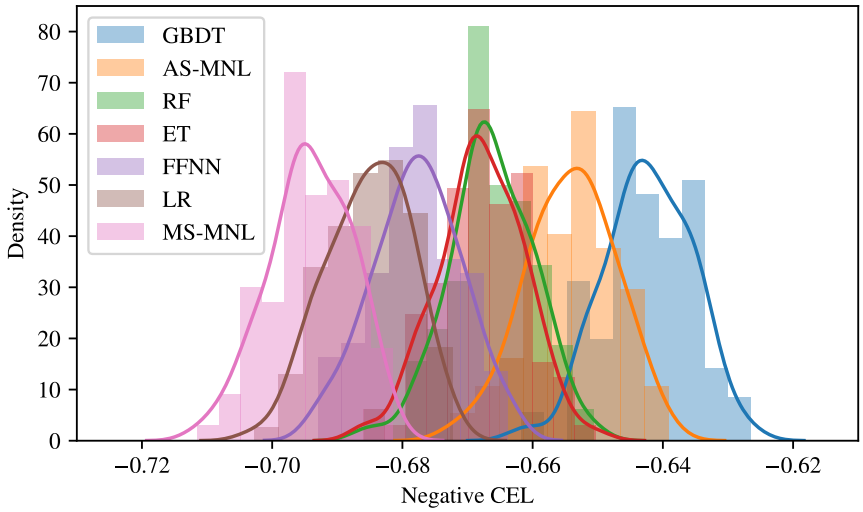

Figure 4: KDE plots and histograms of out-of-sample CEL for 100 iterations of bootstrapping.

Conclusions & future work

In this study, we introduce a new approach for assisted specification of DCMs using the structure of a fitted EL model. The approach is tested against a DCM using manual specification, as well as several ML classifiers. The results show that the assisted specification substantially improves

model performance compared to manual specification, with the resulting model outperforming all ML models except GBDT.

Despite significant improvement in predictive performance, the assisted specification DCM still maintains an interpretable linear behavioural model, with parameter values which are consistent with expected behaviour. This is a substantial advantage over the GBDT model.

Planned future work for this research includes: (i) further testing and case studies for the assisted specification approach, (ii) investigating advanced DCM structures, including nesting and piecewise linear splines, and (iii) formalising the approach in an automated process for automated specification of DCMs.

References

Bergstra, James et al. (2015). "Hyperopt: A Python Library for Model Selection and Hyperparameter Optimization". In: *Computational Science & Discovery* 8.1, p. 014008.

Bierlaire, Michel (2016). *PythonBiogeme: A Short Introduction*. TRANSP-OR 160706. Switzerland: Transport and Mobility Laboratory, School of Architecture, Civil and Environmental Engineering, Ecole Polytechnique Fédérale de Lausanne, p. 19.

Chen, Tianqi and Carlos Guestrin (2016). "XGBoost: A Scalable Tree Boosting System". In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, pp. 785–794.

Dietterich, Thomas G. (2000). "Ensemble Methods in Machine Learning". In: *Multiple Classifier Systems*. Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 1–15.

Friedman, Jerome H. (2001). "Greedy Function Approximation: A Gradient Boosting Machine". In: *The Annals of Statistics* 29.5, pp. 1189–1232.

Hagenauer, Julian and Marco Helbich (2017). "A Comparative Study of Machine Learning Classifiers for Modeling Travel Mode Choice". In: *Expert Systems with Applications* 78, pp. 273–282.

Hastie, Trevor, Jerome Friedman, and Robert Tibshirani (2008). *The Elements of Statistical Learning*. 2nd ed. Springer Series in Statistics. New York: Springer. 745 pp.

Hillel, Tim, Mohammed Z E B Elshafie, and Ying Jin (2018). "Recreating Passenger Mode Choice-Sets for Transport Simulation: A Case Study of London, UK". In: *Proceedings of the Institution of Civil Engineers - Smart Infrastructure and Construction* 171.1, pp. 29–42.

Komer, Brent, James Bergstra, and Chris Eliasmith (2014). "Hyperopt-Sklearn: Automatic Hyperparameter Configuration for Scikit-Learn". In: *Proceedings of the 13th Python in Science Conference*, pp. 34–40.

Kostenko, Boris (Dec. 22, 2018). *XGBoost Feature Interactions Reshaped*. URL: `https://github.com/limexp/xgbfir` (visited on 12/23/2018).

Lee, Dongwoo, Sybil Derrible, and Francisco Camara Pereira (2018). "Comparison of Four Types of Artificial Neural Networks and a Multinomial Logit Model for Travel Mode Choice Modeling". In: *Transportation Research Board 97th Annual Meeting*. Washington DC, USA: Transportation Research Board.

Tang, Liang, Chenfeng Xiong, and Lei Zhang (2015). "Decision Tree Method for Modeling Travel Mode Switching in a Dynamic Behavioral Process". In: *Transportation Planning and Technology* 38.8, pp. 833–850.

Wang, Fangru and Catherine L. Ross (2018). "Machine Learning Travel Mode Choices: Comparing the Performance of an Extreme Gradient Boosting Model with a Multinomial Logit Model". In: *Transportation Research Record* Advanced online publication, pp. 1–11.

Zhang, Yunlong and Yuanchang Xie (2008). "Travel Mode Choice Modeling with Support Vector Machines". In: *Transportation Research Record* 2076, pp. 141–150.