Computational Graph-based Framework for Integrating Econometric Models and Machine Learning Algorithms in Emerging Data-Driven Analytical Environments

Article *in* Transportmetrica A: Transport Science · June 2021

DOI: 10.1080/23249935.2021.1938744

CITATIONS
5

READS
765

3 authors:

Taehooie Kim
UrbanSim
7 PUBLICATIONS **105** CITATIONS
SEE PROFILE

Xuesong Simon Zhou
Arizona State University
220 PUBLICATIONS **7,138** CITATIONS
SEE PROFILE

Ram M. Pendyala
Arizona State University
235 PUBLICATIONS **7,843** CITATIONS
SEE PROFILE

Computational graph-based framework for integrating econometric models and machine learning algorithms in emerging data-driven analytical environments

Taehooie Kim
Arizona State University
School of Sustainable Engineering and the Built Environment
660 S. College Avenue, Tempe, AZ 85287-3005
Email: taehooie.kim@asu.edu

Xuesong (Simon) Zhou
Arizona State University
School of Sustainable Engineering and the Built Environment
660 S. College Avenue, Tempe, AZ 85287-3005
Email: xzhou74@asu.edu

Ram M. Pendyala
Arizona State University
School of Sustainable Engineering and the Built Environment
660 S. College Avenue, Tempe, AZ 85287-3005
Email: ram.pendyala@asu.edu

ABSTRACT

In an era of big data and emergence of new technologies such as app-based ride services, there are growing opportunities for better understanding human mobility patterns from newly available data sources. Statistical models have been mainly utilized to uncover and rigorously calibrate the influence of significant factors; and machine learning algorithms have been used to explore complex patterns through improved computing efficiency for large datasets. Focusing on discrete choice modeling applications, this research aims to introduce an open-source computational graph (CG)-based modeling framework for integrating the strengths of econometric models and machine learning algorithms. In particular, multinomial logit (MNL), nested logit (NL), and integrated choice and latent variable (ICLV) models are selected to demonstrate the performance of the proposed graph-oriented functional representation. Furthermore, the calculation of the gradient in the log-likelihood function and associated Hessian matrix is systematically accomplished using automatic differentiation (AD). Using the 2017 National Household Travel Survey data and an open-source dataset, we compare estimation results from the proposed methods with those obtained from two open-source packages, namely Biogeme and Apollo. The results indicate that the CG-based choice modeling approach can produce consistent estimates of parameters and accurate calculations for the gradients of the estimated parameters with substantial computational efficiency.

Keywords: Computational graphs (CGs), automatic differentiation (AD), multinomial logit (MNL), nested logit (NL), integrated choice and latent variable (ICLV), and gradient calculation.

1. INTRODUCTION

The emergence of massive datasets and widespread internet accessibility across the world have offered valuable opportunities for exploring interconnection between physical/cyber infrastructures and human mobility patterns. This has fostered development of techniques to fuse and analyze multiple data sources such as travel surveys, mobile phone data records, GPS, or sensor data (Hashem et al., 2016; Chen et al., 2016; Wu et al., 2018; Chen and Kwan., 2020). With growing interests to explore available data sources, many scholars have executed machine learning methods to efficiently estimate complex hidden patterns in large-scale datasets. In the field of transportation systems, data-driven approaches have been used to identify patterns of diverse traffic flows as well as assist decision makers to predict future trends (Bhavsar et al., 2017; Chang et al., 2019; Zhao et al., 2020). More recently, the research community has taken further steps to develop interpretable machine learning techniques while significant progress has been made in selecting significant variables that affect travel-related choices, enabling the explanation and testing of predicted results (Ribeiro et al., 2016; Lipton, 2018; Molnar, 2020). These research streams point to a potential paradigm shift in transportation demand modeling.

Transportation planners have also recognized that machine learning methods demonstrate high predictive performance and computing efficiency for large-scale mobility datasets, but those data-driven approaches still need to systematically meet standard requirements and expectations associated with modeling travel data sets (e.g., travel surveys) in transportation planning. The desirable statistics-oriented features include illustrating causal relationships, avoiding overfitted results in relatively small data sets, as well as generating robust standard error estimates for hypothesis testing. If a model estimates only the correlation in a given data set, as pointed out by Mokhtarian (2018), the causation would be eliminated, impeding the ability to answer "why" and "what might happen if" questions. Importantly, incorporating these factors enables researchers and decision makers to deeply fathom the traveler's behavioral patterns. In light of this, statistical modeling approaches have generally been applied in explaining the cause-and-effect relationship and analyzing travel survey data (Paredes et al., 2017; Brathwaite and Walker, 2018b).

In order to bridge the gap between both modeling approaches (i.e., statistical models and machine learning algorithms), this research aims to present a computational framework that can leverage capabilities of existing machine learning platforms to tackle classical estimation problems for discrete choice models. Using a traditional household travel survey dataset and a synthetic dataset available in the Apollo econometric modeling R package, we show how to construct a flexible and efficient modeling framework that utilizes data-driven algorithms in estimating econometric models. The suggested approach could be useful in tackling other estimation problems, such as analyzing multi-dimensional samples from passively collected big data (spatio-temporal dimensions) and enabling real-time updates (predictions) in transportation systems (Nuzzolo and Comi, 2016).

The concept of computational graphs (CGs) is systematically introduced to establish an extended statistical modeling platform capable of covering large-scale datasets and non-linear architectures (e.g., deep neural networks (DNNs)). The computational graph (CG)-based choice models can take full advantage of automatic differentiation (AD) techniques, which have been widely used in machine learning fields (Abadi et al., 2016; Baydin et al., 2017; Paszke et al., 2017). Three different discrete choice models in transportation planning, namely, multinomial logit (MNL), nested logit (NL), and integrated choice and latent variable (ICLV) functions, are reformulated as computational graphs to estimate parameters and associated statistical properties

such as standard errors. These three model forms are chosen because of their widespread use in the field of travel choice modeling. We also examine the flexibility of the modeling structure, and its capability of handling non-concave likelihood functions and simulation-based evaluation of multi-dimensional integrals in latent variable models. Open-source packages, Biogeme (Bierlaire, 2003) and Apollo (Hess and Palma, 2019) are used as test benchmarks, with the publicly accessible National Household Travel Survey (NHTS) 2017 dataset and the synthetic dataset available in the Apollo package serving as use cases.

The remainder of this paper is organized as follows. Section 2 presents the literature review with a particular focus on the integration of statistical models and machine learning methods. Section 3 describes the National Household Travel Survey (NHTS) 2017 and the synthetic datasets. In section 4, the computational graph-based choice models are presented in detail with an emphasis on meeting estimation expectations in planning applications. The estimation and benchmarking results are discussed in section 5.

2. LITERATURE REVIEW

This section addresses three aspects: integration of discrete choice models and machine learning methods, optimization algorithms, and techniques for computing gradients in objective functions. Focusing on the concept of computational graph (CG) and its example, we also provide a discussion of the motivations behind our proposed approach.

2.1. Integration of choice models and machine learning algorithms

Recently, research communities have studied hybrid modelling approaches to integrate strengths of machine learning algorithms into discrete choice models (DCMs). For example, Sifringer et al. (2018) proposed a hybrid modeling framework for combining neural networks and multinomial logistic (MNL) models. Selecting the input features that are relatively uncorrelated with choice alternatives, dense neural network (DNN) learned hidden patterns were derived and the trained information was transmitted into the utility function defined in MNL. This methodology interpreted the specified parameters and led to higher log-likelihood values and improved predictive power. Han et al. (2020) further developed an extended framework to integrate MNL and the constrained data-driven structure (multi-layer perceptron (MLP)). Embedding MLP into the utility function of MNL, their approach demonstrated better predictive performance while maintaining the interpretability and preventing the model from over-fitting. More recently, Sifringer et al. (2020) showed the enhanced choice models by embedding neural networks into the specified utility functions of the MNL and NL models. In a residual logit (ResLogit) model proposed by Wong and Farooq (2019), recursive residual layers were constructed in the utility function of the standard MNL model to capture unobserved heterogeneity. Overall, these above-mentioned modeling efforts aim to resolve overfitting while preserving the econometric interpretability.

Although significant progress has been made to integrate machine learning algorithms in DCM, there are still many challenges to be addressed. First, the existing hybrid models (Sifringer et al., 2018; Han et al., 2020; Sifringer et al., 2020) estimate parameters mainly based on the *Adam* optimizer proposed by Kingma and Ba (2014) or stochastic gradient descent (SGD) (Bottou, 2010).

In terms of optimizing objective functions, the first order-based estimators can be computationally effective to analyze a large-scale dataset and calibrate numerous parameters. However, we have to recognize that there are various model structures in which we are dealing with non-concave functions (e.g., nested logit (NL) model (Williams, 1977)) or simulation-based models involving computation of high-dimensional integrals such as the integrated choice and latent variable (ICLV) model (Ben-Akiva et al., 2002) and the hybrid choice model with a nonlinear utility function (Kim et al., 2016). Second, the first order-based estimation might not be able to provide desirable statistical properties in computing the Hessian matrix. These challenges require a systematic and careful analysis for an effective combination of machine learning techniques and optimization algorithms in the context of statistically-oriented choice models for transportation applications.

2.2. Optimization algorithms for discrete choice models

In the area of discrete choice modeling, maximum likelihood estimation (MLE) is one of the fundamentally important estimation methods. By computing the first order (gradient) and second order (curvature) derivatives of the likelihood function, MLE furnishes values of parameters by maximizing the likelihood function through the use of the Hessian matrix. The derivatives are computed by three approaches: manual/analytical, finite difference, and automatic differentiation (AD) (Bartholomew et al., 2000). Due to the difficulty of embedding/coding highly nonlinear forms in complicated functions, manual differentiation could be used for some very small cases. The numerical differentiation aims to approximate derivatives through the finite differencing, but the solution quality is greatly affected by the potential truncation and round-off errors associated with different finite difference formulas (Wright and Nocedal, 1999). On the other hand, the automatic differentiation (AD) technique utilizes the chain rule-based principle and intermediate variables to evaluate complex derivatives analytically (Wright and Nocedal, 1999; Griewank, and Walther, 2008). Specifically, in the new generation of low-level computational graph libraries such as Tensorflow and PyTorch, the computing architecture can enable modelers to represent the analytical optimization model through a graph of simple elementary operations (i.e., addition, subtraction, multiplication, and division) and elementary functions (e.g., natural logarithm), and further execute a sequential and complex structure of computations easily. In new domain-specific languages (DSLs) for convex optimization such as CVXPY, progress has been made recently to convert standard convex optimization to detailed CG representations with low-level solver interfaces (Agrawal et al. 2018). It should be noted that AD might still encounter the difficulty of computing piecewise rational functions, especially when estimating gradients of non-smooth composite functions (Beck and Fischer, 1994; Nocedal and Wright, 2006).

In the machine learning area, the sequential structure and computational graph approach have been widely applied for large-scale datasets with numerous parameters to be calibrated. These applications have demonstrated the capability of these approaches in computing gradients and Hessians of non-linear optimization formulations efficiently and precisely (Baydin et al., 2017). From a specific system identification perspective, the AD technique has been utilized in the fields of machine learning and econometric modeling to estimate parameters, thanks to its computational efficiency and flexibility of designing diverse composite functions (Sifringer et al., 2018; Wong and Farooq, 2019; Sun et al., 2019; van Kesteren and Oberski, 2019; Han et al., 2020).

Furthermore, in the case of discrete choice modeling (DCM), by carefully selecting the underlying computing algorithms, AD holds the promise for more precise computation of derivatives of the log likelihood with respect to specified parameters through chain rules and back propagation. That is, simply using the popular first order methods (e.g., SGD or *Adam*) is often inadequate in estimating complicated modeling structures (e.g., NL or ICLV). Thus, our research combines the AD technique with quasi-second order methods, e.g., Broyden-Fletcher-Goldfarb-Shanno (BFGS), to calibrate non-concave composite functions and deliver consistent statistical estimates through Hessians.

2.3. Computational graph (CG)

Understanding computational graph (CG) approach is important for designing flexible modeling structures that integrate choice models and machine learning seamlessly. Using the binary logit model in Eq. (1) as an example, Wu et al., (2018) and Sun et al. (2019) took a few initial steps to illustrate how CG can decompose complex composite functions as follows.

$$\mathrm{P}\,(y=1)=\frac{1}{1+e^{-V}} \tag{1}$$

Eq. (1) indicates the probability of choosing a binary alternative, and the term V is a specified utility function (e.g., $V=\beta_0+\beta_1x_1+\beta_2x_2+\cdots+\beta_nx_n$ where β_n is the unknown parameter associated with the attribute x_n). Using the concept of computational graph (CG), this logistic function is now expressed as a directed graph which consists of nodes (elementary operations) and edges (directions):

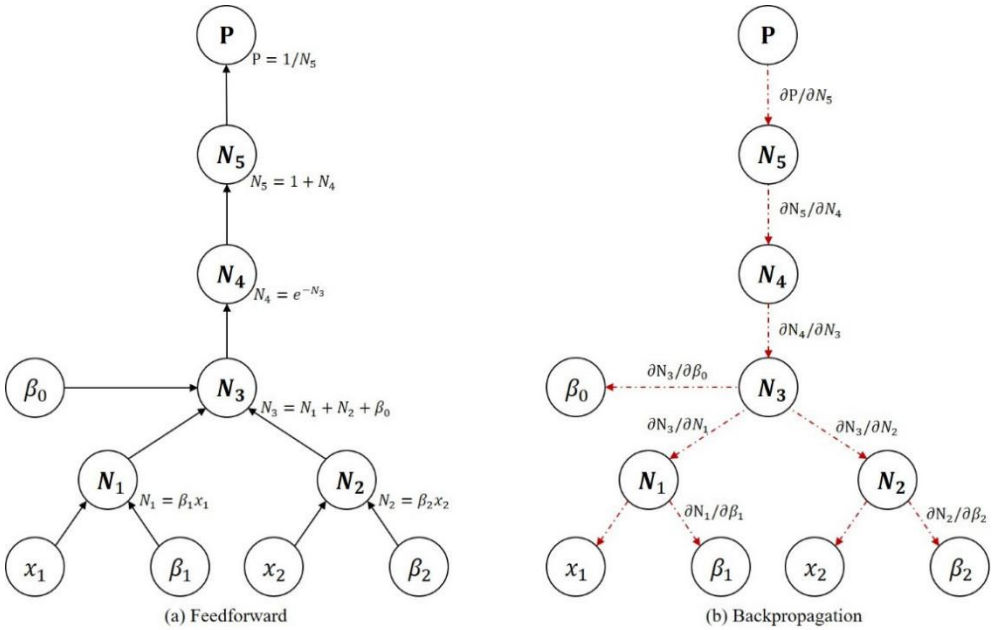

Fig. 1. Computational graph (CG) of the binary logit model

Fig. 1 clearly illustrates the logistic formulation written in Eq. (1) as a sequentially nested structure made up of nodes and edges. In particular, Fig. 1(a) is the process of computing the probability of a given binary alternative, and Fig. 1(b) represents the procedure of estimating parameters. For example, the parameter β_1 is obtained by the defined nodes and links shown in Fig. 1:

$$\frac{\partial \mathrm{P}}{\partial \beta_1} = \frac{\partial \mathrm{P}}{\partial N_5} \cdot \frac{\partial N_5}{\partial N_4} \cdot \frac{\partial N_4}{\partial N_3} \cdot \frac{\partial N_3}{\partial N_1} \cdot \frac{\partial N_1}{\partial \beta_1} = \frac{x_1}{(N_5)^2} \cdot \mathrm{e}^{-N_3} = \frac{x_1}{(1+e^{-V})^2} \cdot \mathrm{e}^{-V} \tag{2}$$

Eq. (2) presents the analytic derivative with respect to the parameter and the description of the chain rule-based computation. Furthermore, applying the gradients in the BFGS optimizer, this computed differentiation offers more precise Hessians. In this context, it is helpful to compare the computed values in Eq. (2) with analytical sensitivities detailed in Koppelman and Bhat (2006) and Train (2009).

To calibrate a broader set of DCMs in transportation planning with rigorously defined standard error estimates, we will tackle three econometric models (i.e., multinomial logit, nested logit , and integrated choice and latent variable) to demonstrate the capability of the enhanced choice modelling framework along three directions: the numerical efficiency of processing a high-dimension survey sample, greater flexibility in employing different composite functions (e.g., deep learning architectures), and realization of desirable statistical properties. A widely used machine learning platform, TensorFlow (Abadi et al., 2016), is selected to implement the proposed CG-based discrete choice models, and the source code can be downloaded at Kim et al. (2021). There are other computational graph-oriented programming platforms such as Theano (Bastien et al., 2012) or Pytorch (Paszke et al., 2017). In addition, to systematically verify the estimated parameters and statistical properties, two leading open-source packages for estimating DCMs, namely Biogeme (Bierlaire, 2003) and Apollo (Hess and Palma, 2019), are used to serve as benchmarks.

It should be noted that the concept of computational graph has been adapted in the pioneering open-source DCM estimation package, Biogeme, in 2000, through the use of chain rule differentiation and analytical gradients. In our proposed domain-specific languages (DSLs) for maximum likelihood estimation of various DCMs, we do not need to build the low-level computational graph manually through a general-purpose language (GPL); instead, we translate the corresponding DCM optimization to forms compatible to the interfaces of recent CG libraries (e.g., TensorFlow). By doing so, our approach can further fully utilize the backpropagation mechanism provided by differentiable optimization layers/pipelines. The DSLs for MLE-DCM helps modelers greatly reduce the computational redundancy by decomposing the computing units in a layered structure and enabling the use of dynamic programming for iteratively finding a solution. The development of domain-specific languages requires a deep understanding of the problem structure and domain knowledge, and we will further highlight the potential for integrating different transportation modeling elements of more complex estimation and planning problems in the conclusion of this paper.

3. DATA PREPARATION

Two datasets are utilized in this research: the 2017 National Household Travel Survey (NHTS) dataset for estimating MNL and NL models, and a synthetic dataset provided by Hess and Palma (2019) to estimate the extended integrated choice and latent variable or ICLV model.

3.1. National Household Travel Survey (NHTS) dataset

The dataset used for the case study is derived from the National Household Travel Survey (NHTS 2017) conducted by the US Department of Transportation. This data set provides information about travel behavior, particularly associated with trip purposes and modes. In the current study, this large-scale dataset with 923,572 trips is explored. To alleviate unobserved taste heterogeneity, we restrict the scope of the trip purpose and time-dimension by selecting commuting trips (home to work trips) departing between 6 and 9 AM.

After filtering the dataset based on criteria and eliminating obviously erroneous observations or those with large amounts of missing data, the final subsample size used for the model estimation is 40,177 observations. Table 1 depicts the travelers' socio-economic and demographic information, as well as travel time and distance variables that are subsequently used as explanatory variables in the specification of the utility function. The five alternatives, namely drive alone (DA), shared ride (SR), transit (TR), bike, and walk, are considered as the choice elements in the proposed MNL and NL choice models. In terms of person characteristics, 84.3 percent of the commuting trips are accounted for by those age 30-74 years. The gender ratio of this subsample is nearly 51 percent male and 49 percent female. In terms of educational attainment, travelers who earned the bachelor's degree and graduate degree account for 29.8 percent and 26.1 percent of the commute tours, respectively. Among household attributes, individuals within the household income categories ($50,000-$124,999 and $125,000 or above) account for 76.7 percent of the commute tours. Two-person households and individuals living with five persons or more account for the highest and lowest proportion of commute tours, respectively. Nearly 79 percent of commuters travel from an urban area. According to travel characteristics, the average commute distance is 12.9 miles with a standard deviation of 15.8 miles, and the average time taken is 27.3 minutes with a standard deviation of 27.9 minutes. The distribution of commute mode choices is 79.3 percent of commute trips by drive alone (DA), 13.8 percent by shared ride, 3.8 percent by transit, and 3.1 percent by bike and walk. This mode choice distribution follows a similar pattern in a prior study by Paleti et al. (2013).

Table 1. Description of the subsample (N=40,177)

Person characteristics	Frequency	Percentage (%)
Age		
Less than 18 years	111	0.3
18-24 years	2,259	5.6
25-29 years	3,549	8.8
30-44 years	11,502	28.6
45-59 years	15,094	37.6
60-74 years	7,270	18.1
75 years or above	392	1.0
Gender		
Male	20,387	50.7
Female	19,790	49.3

Education attainment		
Less than bachelor's degree	17,723	44.1
Bachelor's degree	11,976	29.8
Graduate degree	10,478	26.1
Household characteristics	**Frequency**	**Percentage (%)**
Household income		
Under $25,000	2,812	7.0
$25,000 - $49,999	6,560	16.3
$50,000 - $124,999	10,491	26.1
$125,000 or above	20,314	50.6
Household size		
1 (I am the only person)	6,284	15.6
2 people	17,468	43.5
3 people	7,270	18.1
4 people	6,032	15.0
5 people or more	3,123	7.8
Travel characteristics	**Continuous (average)**	
Trip distance in miles & Trip duration in minutes	12.88 miles & 27.33 minutes	
Endogenous variable	**Frequency**	**Percentage (%)**
Trip mode		
Drive alone (DA)	31,872	79.3
Shared ride (SR)	5,530	13.8
Transit (TR)	1,543	3.8
Bicycle	386	1.0
Walk	846	2.1

3.2. Synthetic dataset

The lack of attitudinal questions in the NHTS dataset renders it unsuitable for constructing ICLV components, i.e., structural models with latent variables and measurement equations. As a result, we utilized an alternative synthetic dataset that accompanies the Apollo package to estimate the ICLV model (instead of using the NHTS dataset). This dataset documents drug choices for 1,000 individuals; four alternative choices, three socio-demographic characteristics, and four attitudinal questions are presented. The explanatory variables to construct the structural equation of a latent variable were binary in nature: regular drug users, university degree attainment, and age 50 years and above. In addition, the attitudinal questions to define the measurement equations followed a Likert scale from 1 (strongly disagree) to 5 (strongly agree). Four attitudinal questions are selected as measurement equation indicators. The detailed description of the drug choice data is well documented in Hess and Palma (2019).

4. MODELING FRAMEWORK AND METHODOLOGY

This section presents the mathematical formulations of MNL, NL, and ICLV models, the computational graph-based modeling frameworks, as well as the stepwise procedure of estimating the proposed graph-oriented functions. Using the travel survey dataset, we develop the systematic utility function and the probability of choice alternatives, namely drive alone (DA), shared ride (SR), transit (TR), bike, and walk, to estimate MNL and NL models. On the other hand, the ICLV components (i.e., the structural equation of the latent variable, measurement indicators, utility

functions, as well as the probability of a drug choice between four alternatives) are constructed using the synthetic dataset.

4.1. Mathematical formulations of the MNL and NL models

With the fundamental assumptions that error components in the utility function are independently and identically distributed according to a Gumbel distribution, the functional formulation of the multinomial logit (MNL) model can be defined clearly. The probability that a decision maker n chooses an alternative mode i among a set of J alternatives (i.e., DA, SR, TR, bike, and walk) is as follows (McFadden, 1974):

$$\mathbf{P}_{n,i} = \frac{e^{V_{n,i}}}{\sum_{j\epsilon J} e^{V_{n,j}}} \tag{3}$$

where $V_{n,i}$ denotes the systematic utility of the alternative mode $i \epsilon J$ selected by the decision maker n, and the structural utility function includes alternative specific constants and observed attributes with their parameters (i.e., $V_{n,i} = ASC_{i,n} + \sum_{k=1}^{K} \beta_{k,i} x_{k,i,n}$). The index J is the set of the specified alternative choices. K represents the number of attributes used as choice predictors.

By reformulating the MNL structure to relax the independence of irrelevant alternatives (IIA) property of MNL, the nested logit (NL) can be specified (Williams 1977; McFadden 1978). In particular, two layered structures are considered in this study. The upper level of NL includes drive alone (DA), shared ride (SR), transit (TR), and the non-motorized group, and the two alternatives (i.e., bike and walk) included in the non-motorized group are located in the lower level.

The functional formula of the choice probability is expressed by the product of the conditional probability and the marginal probability. For instance, the probability that a decision maker n selects an alternative i in the nest m is formulated as:

$$\mathbf{P}_{n,i} = \mathbf{P}_{n,i|J_m} \times \mathbf{P}_{n,J_m} = \frac{e^{V_{n,i}/\lambda_m}}{\sum_{l\in J_m} e^{V_{n,l}/\lambda_m}} \times \frac{e^{(V_{n,m}+\lambda_m \Gamma_{n,m})}}{\sum_{j=1}^{M} e^{(V_{n,j}+\lambda_j \Gamma_{n,j})}} \tag{4}$$

In Eq. (4) the first component is the conditional probability that the decision maker n chooses either a bike or walk mode given that the non-motorized group J_m is selected, and the second component is the marginal probability of choosing between drive alone, shared ride, transit, and the nested group. λ_m is the logsum parameter bounded by zero to one, an indicator of the correlation between bike and walk; the parameter is explained well in Koppelman and Bhat (2006). The inclusive value $\Gamma_{n,m}$ (or often called log-sum term) is defined by $\Gamma_{n,m} = \log\left[\sum_{l\in J_m} e^{V_{n,l}/\lambda_m}\right]$ where this term is associated with the nested group. Readers interested in the derivation of the mathematical formulations can find details in Koppelman and Bhat (2006) and Train (2009).

4.2. Mathematical formulations of the ICLV model

ICLV incorporates a latent variable model into a multinomial discrete choice model. To enable this integrated model, four components are generally required to be specified; a latent variable,

measurement indicators, utility functions, and choice probabilities (Ben-Akiva et al., 2002). First, the latent variable formulated as a function of observable explanatory variables with a stochastic component is given by:

$$X_n^* = \gamma z_n + \eta_n \tag{5}$$

Eq. (5) indicates the structural equation for the latent variable X^* influenced by explanatory variables z_n including three socio-demographic characteristics (in this study) with parameters γ. The stochastic term η_n follows a standard normal distribution $\eta_n \sim N(0,1)$. Second, the probability distribution function of the continuous measurement indicators is expressed as follows:

$$f_n(I_n|z_n, X_n^*; \boldsymbol{\zeta}, \boldsymbol{\sigma}) = \frac{1}{\sqrt{2\pi\sigma_k^2}} e^{-\frac{\left(I_{n,k}-\bar{I}_k-\zeta_k X_n^*\right)^2}{2\sigma_k^2}} \tag{6}$$

where the continuous measurement indicators are defined by $I_{n,k} = \zeta_k X_n^* + v_n$. $I_{n,k}$ represents an indicator associated with an attitude $k \in K$ and the continuous measurement model. $\bar{I}_k$ is the average of the indicator k. Subtracting it from $I_{n,k}$, we avoid estimating the mean of the normal density. ζ_k is the attitudinal coefficient for the latent variable X_n^*, and v_n is the stochastic component characterized by a standard normal distribution $v_n \sim N(0,1)$. Third, the systematic utility function is specified by $V_{n,i} = \sum_{s=1}^{S} \beta_{s,i} x_{s,n,i} + \lambda X_n^*$, where $\beta_{s,i}$ and λ are coefficients of choice predictors and the latent variable, respectively. $V_{n,i}$ represents the utility function of the alternative drug i selected by the decision maker n. Lastly, the probability that a decision maker n chooses a drug i among a set of four products is defined by the multinomial logit formulation. Using the defined components above, we can obtain the joint choice probability as follows (Ben-Akiva et al., 2002; Vij and Walker, 2016):

$$\mathbf{P}_i = \int_{\eta_n} \prod_{k=1}^{K} \frac{1}{\sqrt{2\pi\sigma_k^2}} e^{-\frac{\left(I_{n,k}-\bar{I}_k-\zeta_k X_n^*\right)^2}{2\sigma_k^2}} \times \frac{e^{V_{n,i}}}{\sum_{j\epsilon J} e^{V_{n,j}}} \times \phi(\eta_n) d\eta_n \tag{7}$$

In Eq. (7), the first component is the likelihood of the continuous measurement indicators, the second term is the multinomial logit model, and the third term is derived from the structural equation of the latent variable. Since Eq. (7) has no closed-form solution, this joint choice probability function is conventionally approximated using a Monte Carlo simulation-based approach:

$$\mathbf{P}_i \cong \frac{1}{T} \sum_{t=1}^{T} \prod_{k=1}^{K} \frac{1}{\sqrt{2\pi\sigma_k^2}} e^{-\frac{\left(I_{n,k}-\bar{I}_k-\zeta_k X_{n,t}^*\right)^2}{2\sigma_k^2}} \times \frac{e^{V_{n,i,t}}}{\sum_{j\epsilon J} e^{V_{n,j,t}}} \tag{8}$$

Drawing the standard normal distribution function η_n iteratively, we can simulate the multidimensional integrals, thus deriving Eq. (8); T is the total number of draws. The detailed

description of simulation-based approaches can be found in Train (2009). With the above-derived functions, we now present the procedure of constructing computational graph-based models.

4.3. Illustration of the computational graph-based modeling approach

This subsection presents the CG-based modeling structures for MNL, NL, and ICLV. We present an illustrative example to demonstrate the sequential process of formulating the probability functions associated with mode choices and drug choices in the two datasets respectively. In this description, the probability of choosing the walk mode is exemplified using MNL and NL, and the probability of selecting a drug between four alternatives is illustrated for the ICLV.

4.3.1. CG-based multinomial logit model

Eq. (3) is decomposed and plotted into the directed graph, which includes elementary operations and elementary functions. As shown in Fig. 2, there are 15 input nodes and 16 intermediate nodes to link between input nodes and the output node; input nodes are comprised of the alternative specific constants (ASC) for each alternative and unknown parameters β associated with the attributes x, and the intermediate nodes (N_i where $i = 1, 2, \dots, 16$) play a role of decomposing functions. The output node is the probability of selecting the walk mode $\mathbf{P}_{walk}$. Based on the nodes interconnected by directed edges, we can produce the sequentially nested structure for the probability function so that Eq. (3) can be mapped as follows:

$$\begin{aligned} \mathbf{P}_{walk} &= N_{15}/N_{16} \\ &= \mathrm{e}^{N_{10}}/(N_{11}+N_{12}+N_{13}+N_{14}+N_{15}) \\ &= e^{(N_5+ASC_{walk})}/(e^{N_6}+e^{N_7}+e^{N_8}+e^{N_9}+e^{N_{10}}) \\ &= e^{(\beta_{walk}x_{walk}+ASC_{walk})}/\left(e^{(N_1+ASC_{DA})}+e^{(N_2+ASC_{SR})}+e^{(N_3+ASC_{TR})}+e^{(N_4+ASC_{bike})}+e^{(N_5+ASC_{walk})}\right) \end{aligned} \tag{9}$$

where the nodes from N_{16} to N_5 are used to connect input nodes and the output node, and the index i represents the labels of choice alternatives (DA, SR, TR, Bike, and Walk). It should be noted that in order to simplify the illustration, nodes associated with the availability of the given alternatives are excluded in this graph.

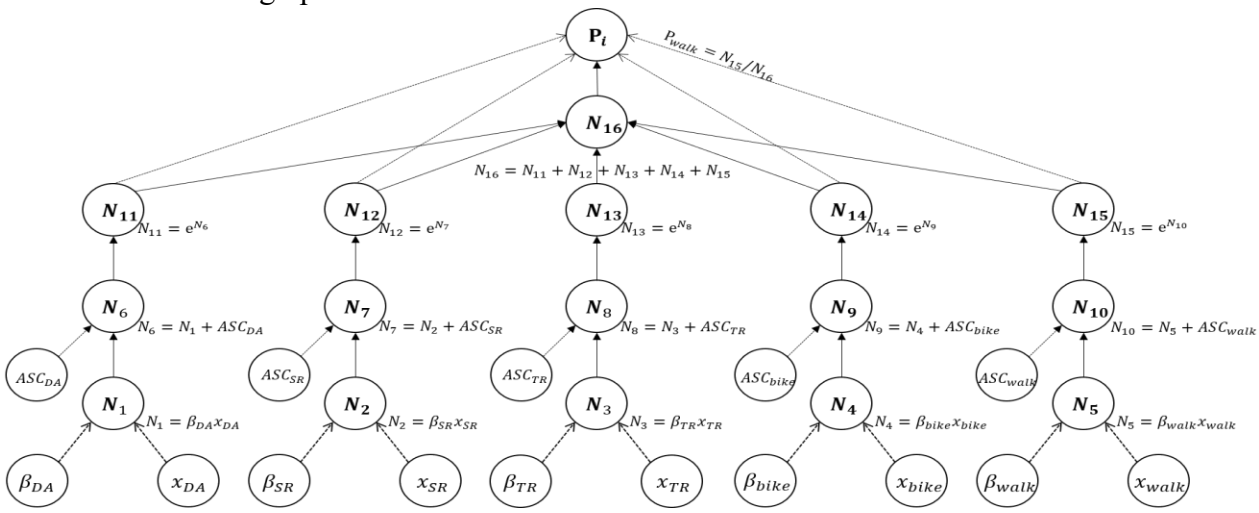

Fig. 2. Illustration of CG-based multinomial logit model

4.3.2 CG-based nested logit model

A two-level nested structure is described in this subsection. Based on Eq. (4), the probability of selecting the walk mode is plotted in Fig. 3. In contrast to the MNL model, this nested model is formulated using the conditional probability and marginal probability to account for the correlation between bike and walk. Fig. 3 denotes 21 input nodes including the nodes used in the MNL computational graph, the log-sum parameter λ_m, as well as the log-sum function Γ_{nm}. In addition, 27 intermediate nodes are embedded to express the decomposed components of NL. With the specified nodes and the directed edges, the product of the conditional probability and the marginal probability can be computed to derive the probability of selecting the walk mode $\mathbf{P}_{walk}$ as follows:

$$\mathbf{P}_{walk} = \mathbf{P}_{walk|non-auto}\mathbf{P}_{non-auto} = N_{26}N_{27} \tag{10}$$

The conditional probability $\mathbf{P}_{walk|non-auto}$ is equal to N_{26}, and the term N_{27} indicates the marginal probability of falling into the non-auto group. To be specific, the sequential steps of mapping the conditional probability $\mathbf{P}_{walk|non-auto}$ are detailed below:

$$\begin{aligned}
\mathbf{P}_{walk|non-auto} &= N_{20}/N_{22} \\
&= e^{N_{15}}/(N_{19}+N_{20}) \\
&= e^{N_9/\lambda_m}/(e^{N_{14}}+e^{N_{15}}) \\
&= e^{(N_4+ASC_{walk})/\lambda_m}/(e^{N_9/\lambda_m}+e^{N_{10}/\lambda_m}) \\
&= e^{(\beta_{walk}x_{walk})/\lambda_m}/(e^{\beta_{bike}x_{bike}/\lambda_m}+e^{\beta_{walk}x_{walk}/\lambda_m})
\end{aligned} \tag{11}$$

Eq. (11) illustrates a stepwise procedure for deriving the conditional probability. The detailed description of the CG nodes and links can be found in Fig. 3. Similarly, the marginal probability $\mathbf{P}_{non-auto}$,which is mapped by the forward propagation of the CG framework, can be written in the following stepwise manner:

$$\begin{aligned}
\mathbf{P}_{non-auto} &= N_{24}/N_{25} \\
&= \lambda_m N_{23}/(N_{21}+N_{24}) \\
&= \lambda_m \log(N_{22})/(N_{16}+N_{17}+N_{18}+\lambda_m N_{23}) \\
&= \lambda_m \log(N_{19}+N_{20})/(e^{N_{11}}+e^{N_{12}}+e^{N_{13}}+\lambda_m N_{23}) \\
&\vdots \\
&= \lambda_m \log\left(e^{N_9/\lambda_m}+e^{N_{10}/\lambda_m}\right)/\left(e^{N_6/\lambda_m}+e^{N_7/\lambda_m}+e^{N_8/\lambda_m}+\lambda_m \log\left(e^{N_9/\lambda_m}+e^{N_{10}/\lambda_m}\right)\right) \\
&= \frac{\lambda_m \log\left(e^{(\beta_{bike}x_{bike})/\lambda_m}+e^{(\beta_{walk}x_{walk})/\lambda_m}\right)}{(e^{N_6/\lambda_m}+e^{N_7/\lambda_m}+e^{N_8/\lambda_m}+\lambda_m log(e^{N_9/\lambda_m}+e^{N_{10}/\lambda_m}))}
\end{aligned} \tag{12}$$

Eq. (12) is the marginal probability of falling into the non-auto nest. The expression $\log\left(e^{(\beta_{bike}x_{bike})/\lambda_m}+e^{(\beta_{walk}x_{walk})/\lambda_m}\right)$ corresponds to the log-sum function Γ_{nm}. By computing the product of Eq. (11) and (12), we can now derive the probability function Eq. (10) through the graph-oriented function. Please note that the utility function V_{nm} shown in Eq. (4) is assumed as zero.

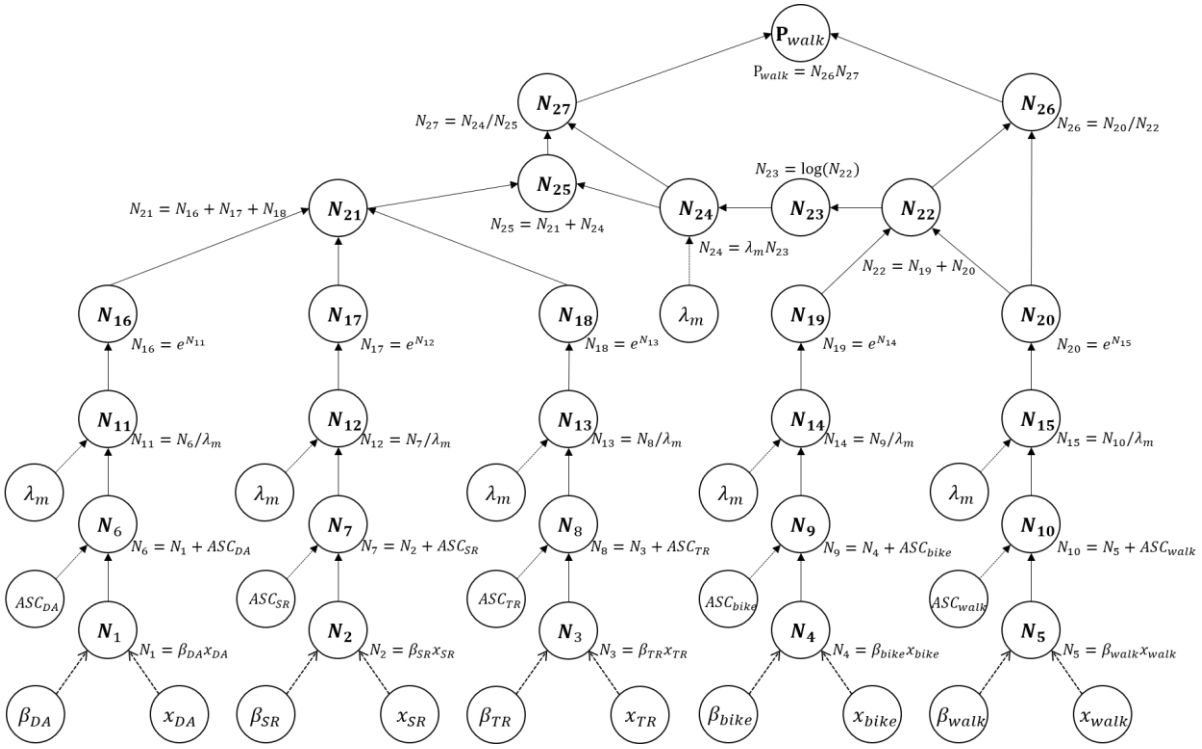

Fig. 3. Illustration of CG-based two-level nested logit model

4.3.3 CG-based integrated choice and latent variable (ICLV) model

In this subsection, the ICLV function comprising of one latent variable, the stochastic term, continuous measurement indicators, as well as the multinomial logit structure is decomposed and plotted in a series of nodes (elementary operations) and edges (directions). According to Fig. 4, 17 input nodes and 22 intermediate nodes are used. N_6, N_{14}, N_{22}, and N_{23} are used to denote the ICLV components, in order to develop the output node $\mathbf{P}_{A1}$ which is the joint choice probability of choosing a drug between four alternatives. With the CG-based structure, the exemplified choice probability can be written in the stepwise manner:

$$\begin{aligned} \mathbf{P}_{A1} &= N_{22} \times N_{23} \\ &= (N_{21}/N_{18}) \times (N_{10}/N_{14}) \\ &= e^{-N_{20}}/\sigma\sqrt{2\pi} \times e^{N_8}/(N_{10} + N_{11} + N_{12} + N_{13}) \\ &\vdots \\ &= \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(I-\zeta N_6)^2}{2\sigma^2}} \times \frac{e^{(N_1+\lambda N_6)}}{(e^{(N_1+\lambda N_6)} + e^{(N_2+\lambda N_6)} + e^{N_3} + e^{N_4})} \end{aligned} \tag{13}$$

Eq. (13) denotes the joint choice probability of falling into drug alternative 1. The first component corresponds to the measurement indicators, while N_6 is the structural equation of the latent variable. The second term is the discrete choice formulation. In order to simplify the illustration shown in Fig. 4, we only show the first iteration of the simulated choice model and exclude nodes associated with the availability of the given alternatives.

With the underlying knowledge of building the forward propagation of the CG-based choice models, the following subsection discusses the automatic differentiation (AD) algorithm to

estimate the proposed CG-based choice models in a backpropagation approach. We describe the backpropagation step by step using the plotted figures.

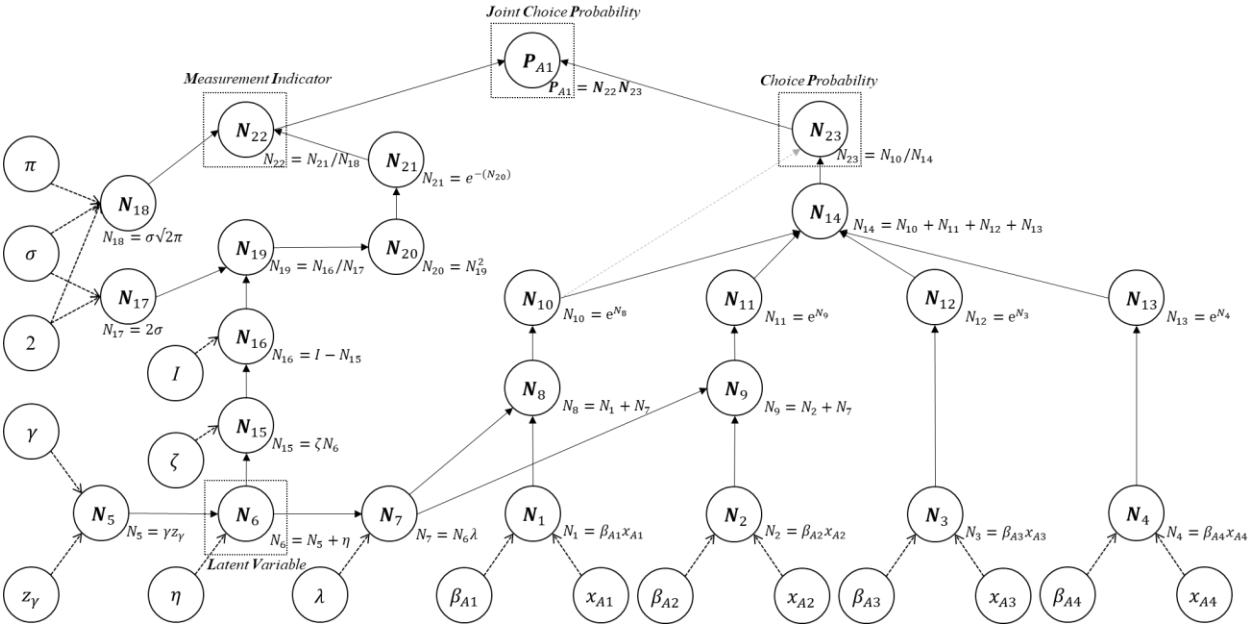

Fig. 4. Illustration of CG-based integrated choice and latent variable (ICLV) model

4.4. Parameter estimation: automatic differentiation (AD) with BFGS

In the CG-based architecture, the unknown parameters specified in Eq. (3), (4), and (8) can be estimated by minimizing the negative log-likelihood function, and the corresponding objective function leads to a particular type of the categorical cross-entropy function proposed by Shannon (1948).

$$H_n(\mathbf{P}_n, \boldsymbol{y}_n) = -\sum_{i \in J} y_{n,i} \ln(\mathrm{P}_{n,i}(\beta)) \tag{14}$$

where $\mathrm{y}_{n,i}$ is the discrete variable that denotes a choice $i \in J$ selected by a decision maker n. Eq. (14) is commonly expressed as $LL(\beta)$, log-likelihood, in the discrete choice field. Using the second-order Taylor's approximation of log-likelihood function $LL(\beta_{k+1})$ in a neighborhood of $LL(\beta_k)$, we can find the optimal value of parameters β_{k+1} to maximize $LL(\beta_{k+1})$ (Train, 2009).

$$\frac{\partial LL(\beta_{k+1})}{\partial \beta_{k+1}} = \frac{\partial LL(\beta_k)}{\partial \beta_k} + B_k(\beta_{t+1} - \beta_t) = 0 \tag{15}$$

The partial derivative of $LL(\beta_k)$ with respect to β_k and the numerically approximated Hessian matrix B_k are determining the best value of β_{k+1}. More specifically, when solving Eq. (15), β_{k+1} can be expressed as $\beta_k + (-B_k)^{-1}(\partial LL(\beta_k)/\partial \beta_k)$. In order to compute the first-order gradients of the objective function with respect to each parameter, we utilize the automatic differentiation (AD) algorithm. By utilizing the derived gradients in the BFGS optimizer, we can calculate the Hessian matrix which is used to evaluate statistical properties of estimated parameters. A detailed

description of computing the numerical Hessian matrix is explained in Nocedal and Wright (2006). As illustrated in the study, the first-order gradient information is valuable for assisting the chain rule-based algorithmic differentiation procedure in deriving the gradients in each choice model.

Consider the estimation of the parameter β_{walk} shown in the equations. The numerical derivative of the parameter in MNL can be derived by the chain rule.

$$\begin{aligned}\frac{\partial LL(\beta_{walk})}{\partial \beta_{walk}} &= \frac{\partial LL(\beta_{walk})}{\partial \mathrm{P}_{walk}}\frac{\partial \mathrm{P}_{walk}}{\partial N_{16}}\frac{\partial N_{16}}{\partial N_{15}}\frac{\partial N_{15}}{\partial N_{10}}\frac{\partial N_{10}}{\partial N_5}\frac{\partial N_5}{\partial \beta_{walk}} \\ &= \frac{1}{\mathrm{P}_{walk}}\frac{(N_{16}-N_{15})}{(N_{16})^2}e^{N_{10}}\boldsymbol{x}_{walk} \\ &= \frac{1}{\mathrm{P}_{walk}}\frac{(e^{N_6}+e^{N_7}+e^{N_8}+e^{N_9})}{(e^{N_6}+e^{N_7}+e^{N_8}+e^{N_9}+e^{N_{10}})^2}e^{N_{10}}\boldsymbol{x}_{walk}\end{aligned} \tag{16}$$

Eq. (16) further details the sequential procedure of computing the partial derivative of $\mathbf{P}_{walk}$ defined in Eq. (9) with respect to the parameter β_{walk}. The description of the intermediate nodes (N_i where i = 6, 7, 8, 9, 10) is illustrated in Fig. 2. The rest of the parameters defined in the CG-based MNL model can be calculated similarly. Now, utilizing the computational graph for the NL model, we introduce the stepwise procedure for computing the partial derivative of the log-likelihood of $\mathbf{P}_{walk}$ with respect to the parameter β_{walk} in Eq. (17).

$$\begin{aligned}\frac{\partial LL(\beta_{walk})}{\partial \beta_{walk}} &= \frac{\partial LL(\beta_{walk})}{\partial \mathrm{P}_{walk}}\frac{\partial \mathrm{P}_{walk}}{\partial N_{26}}\frac{\partial N_{26}}{\partial N_{20}}\frac{\partial N_{20}}{\partial N_{15}}\frac{\partial N_{15}}{\partial N_{10}}\frac{\partial N_{10}}{\partial N_5}\frac{\partial N_5}{\partial \beta_{walk}} \\ &= \frac{1}{\mathrm{P}_{walk}}\frac{N_{27}}{N_{22}}e^{N_{15}}\frac{1}{\lambda_m}\boldsymbol{x}_{walk} \\ &= \frac{1}{\mathrm{P}_{walk}}\frac{\lambda_m N_{23}}{(N_{16}+N_{17}+N_{18}+\lambda_m N_{23})}\frac{e^{N_{15}}}{\lambda_m}\boldsymbol{x}_{walk} \\ &= \frac{1}{\mathrm{P}_{walk}}\frac{\log(e^{N_{14}}+e^{N_{15}})}{(e^{N_{11}}+e^{N_{12}}+e^{N_{13}}+\lambda_m\log(e^{N_{14}}+e^{N_{15}}))}e^{N_{15}}\boldsymbol{x}_{walk}\end{aligned} \tag{17}$$

In the nesting structure, we can observe the log-sum parameter λ_m and inclusive value term as $\log(e^{N_{14}}+e^{N_{15}})$, and the probability of P_{walk} is as shown in Eq. (10). In a similar manner, the stepwise procedure of estimating the partial derivative of the log-likelihood of P_{A1}, Eq. (13), with respect to the parameter β_{A1} in the ICLV model can be expressed as:

$$\begin{aligned}\frac{\partial LL(\beta_{A1})}{\partial \beta_{A1}} &= \frac{\partial LL(\mathrm{P}_{A1})}{\partial \mathrm{P}_{A1}}\frac{\partial \mathrm{P}_{A1}}{\partial N_{23}}\frac{\partial N_{23}}{\partial N_{14}}\frac{\partial N_{14}}{\partial N_{10}}\frac{\partial N_{10}}{\partial N_8}\frac{\partial N_8}{\partial N_1}\frac{\partial N_1}{\partial \beta_{A1}} \\ &= \frac{1}{\mathrm{P}_{A1}}N_{22}\left(-\frac{N_{10}}{(N_{14})^2}\right)e^{N_8}\boldsymbol{x}_{A1} \\ &= \frac{1}{\mathrm{P}_{A1}}N_{22}\left(-\frac{e^{N_8}}{(e^{N_8}+e^{N_9}+e^{N_3}+e^{N_4})^2}\right)e^{N_8}\boldsymbol{x}_{A1} \\ &= \frac{1}{\mathrm{P}_{A1}}\frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{(I-\zeta N_6)^2}{2\sigma^2}}\left(-\frac{e^{N_8}}{(e^{N_8}+e^{N_9}+e^{N_3}+e^{N_4})^2}\right)e^{N_8}\boldsymbol{x}_{A1}\end{aligned} \tag{18}$$

With the computed gradients of the log-likelihood function, the TensorFlow-based program starts from the initial settings of parameters and convergence criteria. Then these numerical tensors are transmitted into the optimizer of BFGS relying on an approximated Hessian matrix, with the

goal of minimizing the negative log-likelihood function defined by the CG-based structure. Based on the iterative algorithm of the optimizer, the inverse of the Hessian matrix $\widehat{H}^{-1}$ is derived such that we can obtain the parameter variance-covariance matrix as follows:

$$\mathrm{SE}(\widehat{\boldsymbol{\beta}}) = \sqrt{\frac{\boldsymbol{\sigma}^2(\widehat{\boldsymbol{\beta}})}{N}} = \sqrt{\frac{(\widehat{\boldsymbol{H}}^{-1})}{N}} = \begin{bmatrix} \sigma^2(\beta_1) & \sigma(\beta_1)\sigma(\beta_2) & \cdots & \sigma(\beta_1)\sigma(\beta_n) \\ \sigma(\beta_2)\sigma(\beta_1) & \sigma^2(\beta_2) & & \sigma(\beta_2)\sigma(\beta_n) \\ \vdots & & \ddots & \vdots \\ \sigma(\beta_n)\sigma(\beta_1) & \sigma(\beta_n)\sigma(\beta_2) & \cdots & \sigma^2(\beta_n) \end{bmatrix}_{n\times n} \tag{19}$$

where $\boldsymbol{\sigma}^2(\widehat{\boldsymbol{\beta}})$ is the variance-covariance matrix of the parameters, $\widehat{H}^{-1}$ is the approximated inverse of the Hessian matrix, and N is the total number of observations. The diagonal elements of $\boldsymbol{\sigma}^2(\widehat{\boldsymbol{\beta}})$ is the variances of parameters. Then, assuming the null hypothesis of β_o=0, t-statistics of each parameter can be obtained.

$$t_{\hat{\beta}_n} = \frac{\hat{\beta}_n - \beta_o}{\mathrm{SE}(\hat{\beta}_n)} \tag{20}$$

Eq. (20) denotes t-statistics of a parameter $\hat{\beta}_n$ and $n \in N$, the total number of estimated parameters. Detailed information on computing the robust t-ratio can be found in the documentation of Biogeme by Bierlaire (2016).

Please note that, while finite differences (numerical differentiation) estimate the gradient (the first-order derivative) using the difference between a certain point and the point added by a small value, the chain rule-based differentiation (AD) produces the exact derivative values. That is, the computational graph-based structures can avoid truncation and round-off errors due to numerical differentiation and accordingly improve the computational efficiency (Chapra and Canale, 2010). Table 2 presents the different characteristics of three estimation models.

Table 2. Attributes of two leading estimation packages and CG-based models

	CG-based Models	**Biogeme**	**Apollo**
Objective function	Log-likelihood ($\ln P_{ni}(\boldsymbol{\beta})$)		
Starting values of the parameters (MNL and NL)	$\beta_i = 0$ where $i = 0, 1, 2, \ldots, n$; $\lambda_{NL} = 0.95$		
Starting values of the parameters (ICLV)	$\beta_i = 0$ where $i = 0, 1, 2, \ldots, n$; $\lambda_{ICLV} = 1$; σ_i and $\zeta_i = 1$ where $i = 1, 2, 3, 4$		
Method of computing gradient derivative	Automatic differentiation through integration of domain-specific language and low-level CG layers	Chain rule of differentiation with analytical gradient	Numerical derivative using advanced extrapolation methods such as Richardson extrapolation
Optimization method	BFGS	BFGS	BFGS
Programming language	Python, C++ library	Python, C++ library	R

In general, CG and both open-source packages use the log-likelihood function as the objective function, start from the same initial values for estimation, and implement the BFGS optimizer with an approximate second-order gradient. CG and Biogeme are coded based on the Python language with underlying C++ libraries, and Apollo (0.2.4 version) is written in the R language

(computational environment: Windows Intel(R) Core (TM) i7-9750H CPU @2.60GHz, 6 Core(s), 32 GB RAM, and 500 GB SSD).

5. MODEL ESTIMATION RESULTS

This section provides the estimation results of MNL, NL, and ICLV models, and our focus is on the investigation of the accuracy and performance of computed gradients through various methods. The computational efficiency and numerical accuracy of the CG-based models are systematically compared to two established DCM estimation packages for MNL and NL models. Using the estimation results of ICLV, we demonstrate the ability of the proposed graph-oriented function to construct a simulation-based choice model and compare performance to the Apollo package. This research does not focus on the behavioral interpretation of the parameters (especially because NHTS data does not furnish level of service attributes critical to mode choice model specification, and the synthetic dataset is used solely for validating the CG-based models).

5.1. Estimation of MNL and NL with constants only

In Table 3, Part I shows the estimation results of MNL including alternative specific constants (ASCs) and their statistical properties. It is found that the graph-oriented approach shows identical estimation results when compared to Biogeme and Apollo; as noted earlier, both packages also implement the BFGS algorithm to derive the coefficients.

Part II of Table 3 compares numerical differences between the CG-based NL model and the benchmark packages. The calibrated coefficients (constants) from CG are consistent with the values estimated by the two packages, but the standard errors of the *Walk* constant and the *logsum* parameter $\boldsymbol{\lambda}$ show some numerical inconsistency.

Table 3. Model estimation results for MNL and NL

Part I: MNL	**DSL- based CG**			**Biogeme**			**Apollo**		
	Coef.	*Std.err*	*t-ratio*	*Coef.*	*Std.err*	*t-ratio*	*Coef.*	*Std.err*	*t-ratio*
*Driving Alone (DA; **base**)*	0	NA	NA	0	NA	NA	0	NA	NA
Shared Ride (SR)	-1.36	0.016	-84.402	-1.36	0.016	-84.402	-1.36	0.016	-84.940
Transit (TR)	-2.93	0.044	-66.547	-2.93	0.044	-66.547	-2.93	0.044	-66.510
Bike	-3.40	0.068	-50.066	-3.40	0.068	-50.066	-3.40	0.068	-50.080
Walk	-3.28	0.051	-63.870	-3.28	0.051	-63.870	-3.28	0.051	-63.780
LL (initial)* // *LL (final)	-27031.930 // -16192.126			-27031.930 // -16192.126			-27031.940 // -16192.130		
AIC // BIC	32392.252 // 32426.656			32392.252 // 32426.656			32392.260 // 32426.670		
Part II: NL	**DSL-based CG**			**Biogeme**			**Apollo**		
	Coef.	*Std.err*	*t-ratio*	*Coef.*	*Std.err*	*t-ratio*	*Coef.*	*Std.err*	*t-ratio*
*Driving Alone (DA; **base**)*	0	NA	NA	0	NA	NA	0	NA	NA
Shared Ride (SR)	-1.36	0.016	-84.364	-1.36	0.016	-84.963	-1.36	0.016	-84.960
Transit (TR)	-2.92	0.044	-66.056	-2.92	0.044	-66.581	-2.92	0.044	-66.580
Bike	-3.10	0.072	-43.085	-3.10	0.073	-42.253	-3.10	0.073	-42.250
Walk	-3.12	0.059	-52.523	-3.12	0.062	-50.677	-3.12	0.062	-50.680
Logsum ($\boldsymbol{\lambda}$)	0.46	0.117	3.917	2.21*	0.622	3.556	0.45	0.127	3.560
LL (initial)* // *LL (final)	-27031.94 // -16183.793			-27031.94 // -16183.78			-27031.94 // -16183.78		
AIC // BIC	32377.586 // 32420.592			32377.56 // 32420.57			32377.56 // 32420.57		

**Note*: The calculated $\boldsymbol{\lambda}$ in Biogeme is expressed as the inverse of $\boldsymbol{\lambda}$ (i.e., $1/2.21 \cong 0.45$)

In order to check the source of this inconsistency, we investigate how the packages (Biogeme and Apollo) approximate the Hessian matrix of the log-likelihood function with respect to each parameter. Biogeme aims to approximate the elements of the Hessian matrix based on chain rule differentiation (CRD) and calculate the standard errors of the coefficients.

Unlike the estimation results through CRD, the proposed modeling approach in this paper uses automatic differentiation (AD) to obtain the first order gradient of the log-likelihood function. Both approaches are based on the chain rule-based differentiation, but AD can implement intermediate variables in computing gradients, which enables the proposed model to find the analytic gradients efficiently.

Table 4 compares the numeric gradients extracted from two approaches (Biogeme and CG-based). In Table 4-Part I, we notice the gradients computed through both CRD and AD are approaching zero so that the approximated standard errors were closely identical to each other. However, as the gradient approximated by CRD in Part II (nested logit) is not sufficiently close to zero, the approximated Hessian matrix might yield different standard errors compared to the AD-based result. As shown in Eq. (15), the magnitude of the first-order gradients is a critical indicator for convergence, which is required to assure maximization of the log-likelihood functions (Train 2009). Please note that the approximation issue of CRD has been investigated and discussed by Brathwaite (2017) and Brathwaite and Walker, (2018a). According to Table 4-Part I, the absolute averages of gradients of CRD and AD are 1.32E-05 and 1.78E-09, respectively. Table 4-Part II shows the absolute average of the gradients of CRD is 1.16E-04 while the corresponding value for AD shows 2.83E-07. The gradients produced from both methods are significantly small, and the differences depend on the selection of stopping criteria. In other words, if we use the same stopping criteria for the estimation of gradients in both methods, the discrepancy shown in Table 4 would be vanished.

Table 4. Estimated Gradients computed by chain rule differentiation and analytical gradient (CRD+AG) and automatic differentiation (AD) ***through DSL***

Part I: Gradients of MNL	Chain rule differentiation and analytical gradient CRD+AG	***Automatic Differentiation (AD) through DSL***
Driving Alone (DA; ***base)***	0	0
Shared Ride (SR)	-9.85144E-05	1.86265E-08
Transit (TR)	9.86795E-05	-1.19908E-08
Bike	-3.09112E-05	4.65661E-10
Walk	-2.21991E-05	0
Part II: Gradients of NL	CRD+AG	***AD + DSL***
Driving Alone (DA; ***base)***	0	0
Shared Ride (SR)	1.15E-03	1.86265E-09
Transit (TR)	5.97E-04	4.08152E-07
Bike	-2.57E-05	9.76317E-07
Walk	-1.31E-03	-1.57219E-07
*Logsum (**λ**)*	1.71E-04	1.86265E-07

5.2. Estimation of MNL and NL with constants and explanatory variables

This subsection presents estimation results for a fully specified model including explanatory variables. Specifically, five categorical variables and one continuous variable were included. The utility function of each mode is influenced by the same explanatory variables; age groups, gender, education attainment, household income and size, as well as travel time. There are 33 estimated parameters, and the detailed description of each parameter is provided in Table 5 and Table 6. Based on the log-likelihood values obtained, all methods showed similarity in terms of the estimated coefficients. On the other hand, due to the fact that the two packages used different methods to derive the gradients (numerical differentiation and chain-rule differentiation, respectively) of the parameters while the CG-based structure utilized the analytical approach (i.e., AD), we see differences in the numeric gradients. These differences likely explain the discrepancy in standard errors and t-ratio statistics.

The gradients computed by CRD and AD are presented in Table 7. As expected, the gradients computed by the algorithmic differentiation are significantly closer to zero compared to the counterpart by the chain rule-based approach with different stopping criteria. In terms of the final absolute average of gradients in MNL and NL, CRD provides values of 8.72E-05 in MNL and 1.86E-04 in NL. On the other hand, the estimated gradients using AD are 9.31E-07 in MNL and 1.07E-08 in NL.

Table 5. Model estimation results for Multinomial Logit (MNL) with explanatory variables

Part III: MNL with explanatory variables		DSLCG -based MNL			Biogeme			Apollo		
		Coef.	*Std.err*	*t-ratio*	*Coef.*	*Std.err*	*t-ratio*	*Coef.*	*Std.err*	*t-ratio*
Drive Alone (DA; ***base***)		0.00	NA	NA	0.00	NA	NA	0.00	NA	NA
Shared Ride (SR)		-1.25	0.07	-17.69	-1.25	0.07	-17.41	-1.25	0.07	-17.38
Transit (TR)		-9.55	0.33	-29.02	-9.55	0.33	-29.12	-9.55	0.33	-29.07
Bike		-3.66	0.30	-12.15	-3.67	0.31	-11.88	-3.67	0.31	-11.86
Walk		-0.54	0.16	-3.36	-0.53	0.18	-3.05	-0.53	0.18	-3.05
SR	Gender (Male=1, Female=0)	-0.10	0.03	-3.21	-0.10	0.03	-3.18	-0.10	0.03	-3.17
	Aged 30-44 years (Yes=1, No=0)	0.12	0.04	2.85	0.12	0.04	2.80	0.12	0.04	2.79
	Aged 45-59 years (Yes=1, No=0)	-0.06	0.04	-1.48	-0.06	0.04	-1.47	-0.06	0.04	-1.47
	Education attainment: Graduate degree (Yes=1, No=0)	-0.18	0.04	-4.88	-0.18	0.04	-4.81	-0.18	0.04	-4.82
	Household income: $125,000 or more (Yes=1, No=0)	-0.06	0.03	-1.83	-0.06	0.03	-1.82	-0.06	0.03	-1.82
	Household size: Three-person or more (Yes=1, No=0)	0.11	0.03	3.38	0.11	0.03	3.32	0.11	0.03	3.32
	Natural logarithm of travel time (in minutes)	-0.02	0.05	-0.43	-0.02	0.05	-0.43	-0.02	0.05	-0.43
TR	Gender (Male=1, Female=0)	-0.15	0.04	-3.35	-0.15	0.10	-1.49	-0.15	0.10	-1.48
	Aged 30-44 years (Yes=1, No=0)	0.17	0.08	2.17	0.16	0.13	1.30	0.16	0.13	1.31
	Aged 45-59 years (Yes=1, No=0)	-0.20	0.08	-2.66	-0.21	0.13	-1.63	-0.21	0.13	-1.63
	Education attainment: Graduate degree (Yes=1, No=0)	0.45	0.08	5.54	0.45	0.10	4.32	0.45	0.10	4.32
	Household income: $125,000 or more (Yes=1, No=0)	-0.17	0.09	-1.90	-0.17	0.10	-1.67	-0.17	0.10	-1.67
	Household size: Three-person or more (Yes=1, No=0)	-0.04	0.06	-0.59	-0.04	0.11	-0.34	-0.04	0.11	-0.34
	Natural logarithm of travel time (in minutes)	4.33	0.19	22.99	4.33	0.19	22.44	4.33	0.19	22.47
Bike	Gender (Male=1, Female=0)	0.62	0.11	5.39	0.61	0.15	3.97	0.61	0.15	3.96
	Aged 30-44 years (Yes=1, No=0)	0.13	0.07	1.83	0.13	0.17	0.74	0.13	0.17	0.74
	Aged 45-59 years (Yes=1, No=0)	-0.36	0.07	-5.33	-0.36	0.18	-2.00	-0.36	0.18	-2.01
	Education attainment: Graduate degree (Yes=1, No=0)	0.55	0.08	6.87	0.55	0.14	3.93	0.55	0.14	3.93
	Household income: $125,000 or more (Yes=1, No=0)	0.06	0.08	0.80	0.06	0.14	0.41	0.06	0.14	0.41
	Household size: Three-person or more (Yes=1, No=0)	-0.16	0.04	-4.04	-0.16	0.15	-1.08	-0.16	0.15	-1.08
	Natural logarithm of travel time (in minutes)	-0.25	0.17	-1.46	-0.24	0.20	-1.16	-0.24	0.20	-1.16
Walk	Gender (Male=1, Female=0)	-0.17	0.06	-3.02	-0.17	0.11	-1.60	-0.17	0.11	-1.61
	Aged 30-44 years (Yes=1, No=0)	-0.07	0.07	-1.01	-0.07	0.14	-0.55	-0.07	0.13	-0.55
	Aged 45-59 years (Yes=1, No=0)	-0.46	0.08	-5.45	-0.45	0.13	-3.44	-0.45	0.13	-3.45
	Education attainment: Graduate degree (Yes=1, No=0)	0.27	0.07	3.88	0.26	0.11	2.32	0.26	0.11	2.32
	Household income: $125,000 or more (Yes=1, No=0)	-0.21	0.05	-4.10	-0.22	0.11	-2.00	-0.22	0.11	-2.00
	Household size: Three-person or more (Yes=1, No=0)	-0.19	0.04	-4.28	-0.20	0.12	-1.59	-0.20	0.12	-1.59
	Natural logarithm of travel time (in minutes)	-2.20	0.12	-18.78	-2.20	0.14	-16.18	-2.20	0.14	-16.12
LL (initial) // LL (final)		-27031.94 // -15553.39			-27031.94 // -15553.39			-27031.94 // -15553.39		
AIC // BIC		31170.78 // 31446.02			31170.78 // 31446.02			31170.78 // 31446.02		

Table 6. Model estimation results for Nested Logit (NL) with explanatory variables

Part IV: NL with explanatory variables			DSLCG-based NL			Biogeme			Apollo		
			Coef.	*Std.err*	*t-ratio*	*Coef.*	*Std.err*	*t-ratio*	*Coef.*	*Std.err*	*t-ratio*
Drive Alone (DA; ***base***)			0.00	NA	NA	0.00	NA	NA	0.00	NA	NA
Shared Ride (SR)			-1.25	0.07	-17.61	-1.25	0.07	-17.43	-1.25	0.07	-17.43
Transit (TR)			-9.49	0.33	-29.06	-9.49	0.33	-28.97	-9.49	0.33	-28.97
Bike			-2.70	0.37	-7.32	-2.69	0.37	-7.26	-2.69	0.37	-7.26
Walk			-0.70	0.18	-3.97	-0.70	0.18	-3.96	-0.70	0.18	-3.96
Logsum ($\boldsymbol{\lambda}$)			0.56	0.11	5.00	1.80*	0.37	4.84	0.56	0.12	4.84
SR		Gender (Male=1, Female=0)	-0.10	0.03	-3.22	-0.10	0.03	-3.18	-0.10	0.03	-3.18
		Aged 30-44 years (Yes=1, No=0)	0.12	0.04	2.80	0.12	0.04	2.78	0.12	0.04	2.78
		Aged 45-59 years (Yes=1, No=0)	-0.06	0.04	-1.47	-0.06	0.04	-1.46	-0.06	0.04	-1.46
		Education attainment: Graduate degree (Yes=1, No=0)	-0.18	0.04	-4.93	-0.18	0.04	-4.85	-0.18	0.04	-4.85
		Household income: $125,000 or more (Yes=1, No=0)	-0.06	0.03	-1.83	-0.06	0.03	-1.82	-0.06	0.03	-1.82
		Household size: Three-person or more (Yes=1, No=0)	0.11	0.03	3.37	0.11	0.03	3.33	0.11	0.03	3.33
		Natural logarithm of travel time (in minutes)	-0.02	0.05	-0.40	-0.02	0.05	-0.39	-0.02	0.05	-0.39
TR		Gender (Male=1, Female=0)	-0.15	0.05	-3.23	-0.15	0.10	-1.52	-0.15	0.10	-1.52
		Aged 30-44 years (Yes=1, No=0)	0.16	0.08	2.11	0.16	0.13	1.30	0.16	0.13	1.30
		Aged 45-59 years (Yes=1, No=0)	-0.21	0.09	-2.45	-0.21	0.13	-1.64	-0.21	0.13	-1.64
		Education attainment: Graduate degree (Yes=1, No=0)	0.44	0.05	8.34	0.44	0.10	4.30	0.44	0.10	4.30
		Household income: $125,000 or more (Yes=1, No=0)	-0.18	0.08	-2.22	-0.17	0.10	-1.71	-0.17	0.10	-1.71
		Household size: Three-person or more (Yes=1, No=0)	-0.04	0.04	-0.96	-0.04	0.11	-0.38	-0.04	0.11	-0.38
		Natural logarithm of travel time (in minutes)	4.30	0.18	23.46	4.30	0.19	22.38	4.30	0.19	22.38
Nested Group	**Bike**	Gender (Male=1, Female=0)	0.46	0.09	4.90	0.46	0.13	3.52	0.46	0.13	3.52
		Aged 30-44 years (Yes=1, No=0)	0.06	0.10	0.63	0.06	0.14	0.41	0.06	0.14	0.42
		Aged 45-59 years (Yes=1, No=0)	-0.40	0.12	-3.46	-0.40	0.15	-2.73	-0.40	0.15	-2.73
		Education attainment: Graduate degree (Yes=1, No=0)	0.45	0.08	5.61	0.45	0.12	3.79	0.45	0.12	3.79
		Household income: $125,000 or more (Yes=1, No=0)	0.00	0.08	0.04	0.00	0.12	0.03	0.00	0.12	0.03
		Household size: Three-person or more (Yes=1, No=0)	-0.10	0.07	-1.53	-0.10	0.13	-0.80	-0.10	0.13	-0.80
		Natural logarithm of travel time (in minutes)	-0.68	0.22	-3.12	-0.68	0.22	-3.11	-0.68	0.22	-3.11
	Walk	Gender (Male=1, Female=0)	-0.09	0.07	-1.31	-0.09	0.10	-0.91	-0.09	0.10	-0.91
		Aged 30-44 years (Yes=1, No=0)	-0.01	0.11	-0.07	-0.01	0.12	-0.07	-0.01	0.12	-0.07
		Aged 45-59 years (Yes=1, No=0)	-0.41	0.11	-3.77	-0.41	0.12	-3.39	-0.41	0.12	-3.39
		Education attainment: Graduate degree (Yes=1, No=0)	0.34	0.07	4.78	0.34	0.10	3.25	0.34	0.10	3.25
		Household income: $125,000 or more (Yes=1, No=0)	-0.18	0.08	-2.27	-0.18	0.10	-1.78	-0.18	0.10	-1.78
		Household size: Three-person or more (Yes=1, No=0)	-0.21	0.06	-3.28	-0.21	0.11	-1.88	-0.21	0.11	-1.88
		Natural logarithm of travel time (in minutes)	-2.02	0.13	-15.04	-2.02	0.14	-14.23	-2.02	0.14	-14.23
LL (initial) // LL (final)			-26962.012 // -15547.65			-27107.14 // -15547.65			-26962.02 // -15547.65		
AIC // BIC			31161.29 // 31445.13			31161.29 // 31445.13			31161.29 // 31445.13		

**Note*: The calculated logsum coefficient in Biogeme is expressed as the inverse of $\boldsymbol{\lambda}$ (i.e., $1/1.7968 \cong 0.56$)

Table 7. Gradients estimated by chain rule differentiation (CRD) and automatic differentiation (AD) through **DSL- based CG**

Estimated Gradients	Part III: Gradients of MNL		Part IV: Gradients of NL	
	Chain Rule Differentiation	***AD*+DSL CG**	***Chain Rule Differentiation***	***AD*+DSL CG**
Drive Alone (DA; ***base***)	0	0	0	0
Shared Ride (SR)	-4.54E-04	-3.04E-06	-4.25E-04	3.09E-07
Transit (TR)	-3.47E-03	-2.75E-06	-2.15E-03	8.67E-08
Bike	1.43E-03	4.66E-07	-1.76E-03	1.12E-07
Walk	-8.73E-04	-6.45E-06	-4.12E-04	1.81E-07
Logsum (λ)	NA	NA	1.30E-03	4.11E-07
Shared Ride (SR)				
Gender	5.35E-04	2.84E-07	3.96E-04	1.70E-07
Aged 30-44 years	2.35E-03	1.05E-06	8.39E-04	3.11E-08
Aged 45-59 years	2.48E-03	8.15E-07	4.98E-04	-7.12E-08
Education attainment: Graduate degree	8.36E-04	3.72E-07	9.26E-04	8.35E-08
Household income: $125,000 or more	5.04E-04	2.11E-07	6.06E-04	-6.84E-08
Household size: Three-person or more	-6.85E-04	1.18E-06	3.47E-04	-8.13E-08
Natural logarithm of travel time	-2.93E-03	2.55E-06	1.30E-03	-2.36E-07
Transit (TR)				
Gender	-9.98E-04	2.25E-06	5.82E-04	1.29E-06
Aged 30-44 years	-5.99E-04	8.00E-06	1.39E-03	5.01E-07
Aged 45-59 years	4.21E-04	6.02E-06	-5.89E-04	-1.55E-07
Education attainment: Graduate degree	-3.65E-04	1.22E-06	5.96E-04	-8.60E-07
Household income: $125,000 or more	-9.01E-04	9.93E-06	-5.88E-04	-6.74E-07
Household size: Three-person or more	-3.44E-04	9.02E-07	9.93E-04	-1.31E-06
Natural logarithm of travel time	-2.27E-03	-6.47E-06	-3.75E-04	2.17E-07
Bike				
Gender	8.69E-04	4.04E-06	-7.97E-04	5.82E-07
Aged 30-44 years	1.04E-03	9.75E-08	-3.04E-05	1.63E-06
Aged 45-59 years	-6.08E-04	6.12E-07	-2.67E-03	8.69E-07
Education attainment: Graduate degree	2.07E-04	-3.88E-06	3.00E-03	7.30E-08
Household income: $125,000 or more	-9.84E-04	5.42E-06	-1.25E-03	1.53E-07
Household size: Three-person or more	1.35E-04	5.99E-06	5.50E-04	-2.74E-06
Natural logarithm of travel time	-1.66E-03	-8.02E-06	-1.16E-04	-1.55E-07
Walk				
Gender	1.05E-03	-1.27E-06	5.64E-04	-2.08E-07
Aged 30-44 years	-1.21E-04	6.23E-06	-7.54E-04	7.29E-07
Aged 45-59 years	9.36E-04	-1.25E-05	2.96E-03	5.58E-07
Education attainment: Graduate degree	1.57E-03	4.72E-06	-1.02E-03	-2.80E-07
Household income: $125,000 or more	1.14E-03	1.30E-06	7.14E-04	5.81E-07
Household size: Three-person or more	1.05E-03	8.26E-06	2.43E-04	-1.15E-06
Natural logarithm of travel time	-2.08E-03	2.29E-06	1.26E-03	-2.24E-07

5.3. Computational efficiency: MNL and NL

We now compare the computational efficiency across all methods. As seen in Fig. 5, the CG-based models show the best computational performance, and a slight increase in running time is observed in both Fig. 5 (a) and (b) when more parameters are added. Biogeme, which is written in Python, also provides excellent computational performance to compute a few parameters. However, for a larger number of parameters to be calibrated, the Biogeme package could yield a nonlinear increase in running time, particularly when models involve non-concave functions (two or multiple nested structures). The Apollo package coded in the R language demands significantly more computing resources. For instance, when estimating a large set of parameters (i.e., 89 parameters), the average running time of CG-based MNL and NL is 10.6 seconds. On the other hand, the average computing times for Biogeme and Apollo are 12 minutes and 35 minutes, respectively. In Fig.5 (b), it can be seen that the nested logit models estimated by Biogeme and Apollo packages require substantially more computational time when the set of variables becomes large.

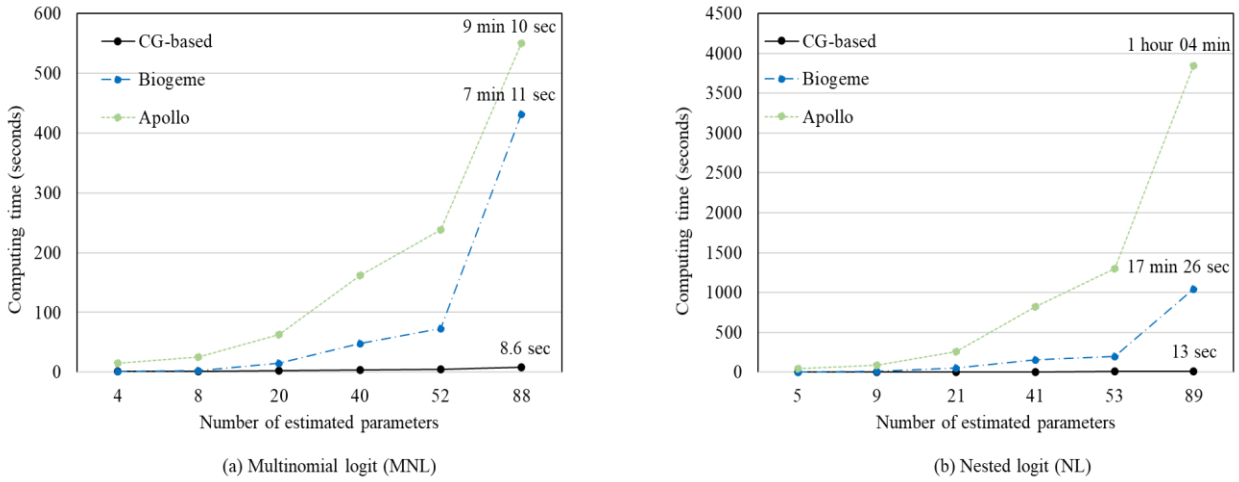

Fig. 5. Comparison of computation time between CG-based models, Biogeme, and Apollo

5.4. ICLV model estimation and computational efficiency

In this subsection, experimental results for the ICLV model are presented. The graph-oriented model and Apollo use the Monte Carlo simulation-based approach to numerically compute the ICLV function. By generating random numbers from a normal distribution, we can run the program 500 times. The specified utility function is defined by two explanatory variables and one latent variable constructed by the structural equation where it is defined by three socio-demographic characteristics. As we assume the indicators as continuous variables, components required in the normal distribution function are estimated. Table 8 demonstrates the ability of the CG-based approach to construct the simulation-based choice model, yielding simulated coefficients. Because the estimation involves the random sampling procedure and different methods to derive coefficients' gradients, we observe slightly different results between the CG-based ICLV, Biogeme, and Apollo. For instance, the initial log-likelihood of CG-based ICLV

displays -8405.706 while Biogeme and Apollo show values of -8404.603 and -8404.237, respectively.

Table 8. Model estimation results for ICLV: Monte Carlo experiment

ICLV	DSLCG-based ICLV			Biogeme			Apollo		
	Coef.	*Std.err*	*t-ratio*	*Coef.*	*Std.err*	*t-ratio*	*Coef.*	*Std.err*	*t-ratio*
Parameters in the utility specification									
Drug: side-effect	-0.002	0.0002	-11.03	-0.002	0.0002	-11.1	-0.002	0.0002	-11.05
Drug: price	-0.173	0.032	-5.45	-0.173	0.032	-5.42	-0.173	0.032	-5.42
λ_{latent}	0.567	0.089	6.33	0.565	0.089	6.37	0.569	0.089	6.39
Parameters in the structural equation									
Regular user (Yes=1, No=0)	-0.677	0.072	-9.47	-0.678	0.087	-7.78	-0.677	0.087	-7.81
Education attainment: Bachelor's degree (Yes=1, No=0)	-0.253	0.054	-4.707	-0.249	0.079	-3.15	-0.248	0.079	-3.14
Aged 50 or above (Yes=1, No=0)	0.675	0.076	8.92	0.677	0.085	8.01	0.674	0.084	7.99
Parameters in measurement indicators									
ζ_{Quality}	0.562	0.044	12.7	0.557	0.045	12.3	0.564	0.046	12.4
$\zeta_{\text{Ingredients}}$	-0.565	0.043	-13.3	-0.564	0.046	-12.2	-0.564	0.046	-12.16
ζ_{Patent}	0.613	0.047	13.1	0.608	0.047	13	0.609	0.047	12.89
$\zeta_{\text{Dominance}}$	-0.400	0.036	-11.21	-0.40	0.041	-9.78	-0.401	0.041	-9.78
$\sigma_{Quality}$	1.053	0.032	33.13	1.05	0.03	34.6	1.051	0.031	34.29
$\sigma_{Ingredients}$	1.08	0.030	37.4	1.08	0.031	34.8	1.079	0.031	34.89
σ_{Patent}	1.091	0.033	32.74	1.09	0.033	33.6	1.093	0.033	33.51
$\sigma_{Dominance}$	1.047	0.025	41.57	1.05	0.027	39.5	1.047	0.027	39.48
LL (initial) // LL (final)	-8405.706 // -7552.271			-8404.603 // -7553.033			-8404.237 // -7552.271		
AIC // BIC	15132.434 // 15201.143			15134.07 // 15202.77			15132.54 // 15201.25		

In Fig. 6, the CG-based ICLV shows the best computational performance in running Monte Carlo simulation for estimating ICLV, when compared to Biogeme and Apollo. The above limited experiments show that, when the number of simulation runs increases, the two-open source packages take more computational time than the CG-based approach using DSL.

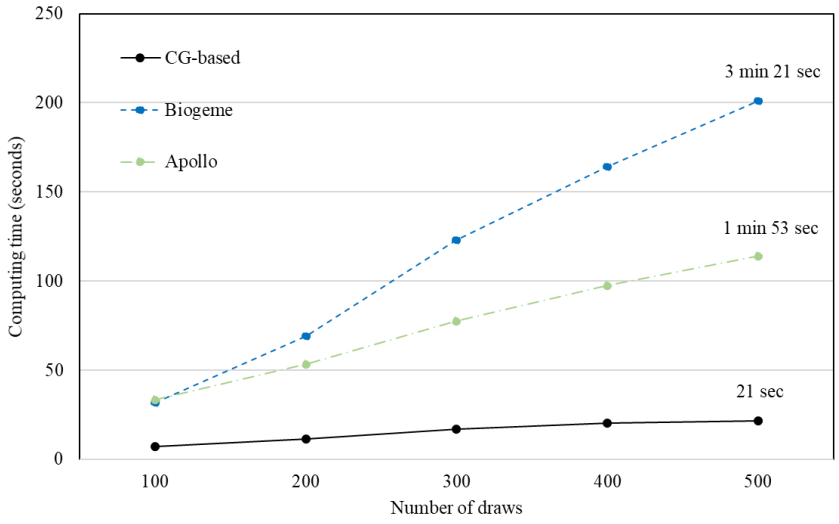

Fig. 6. Comparison of simulation running time for ICLV Estimation between DSLCG-model, Biogeme, and Apollo

6. CONCLUSIONS

As the influx of real-time streaming data and new mobility technologies appears in the field of transportation, transportation planning communities are very interested in systematically integrating data-driven models and econometric models. In this paper, to bridge the gap between both methods, the functional formulation of discrete choice models is examined in a computational graph framework, which is less known in the areas of discrete choice modeling and transportation planning, but has been widely used as underlying building blocks for deep learning packages. We hope to clearly show an implementable path to empower DCM estimation with the automatic differentiation algorithm embedded in CG, through three key findings below.

(a) A computational graph-based framework offers a highly flexible modeling method for applying the emerging techniques of deep learning in econometric methods, especially for a wide class of discrete choice models. Furthermore, CG can cover a wide range of elementary operations in its graph-oriented model representation such that researchers can easily integrate standard econometric models with machine learning algorithms that deal effectively with large amounts of time series data.

(b) In particular, for MNL and NL models, we demonstrate that CG-based learning process produces consistent estimation results compared to two leading packages, namely Biogeme and Apollo. In terms of estimating t-statistics, the *chain rule* of AD provides a robust analytical derivation, leading to converging computed gradients toward the optimality conditions. Compared to the other approximated gradient methods, the proposed approach generates high-quality estimators through a more precise Hessian matrix. Furthermore, by demonstrating the capability in the context of the ICLV modeling structure, we also show CG can be used as an effective framework in implementing extended choice models.

(c) For emerging transportation planning applications with high-dimensional survey samples and real-time big data streams, the proposed methodology holds the promise of achieving computational efficiency in handling large-scale datasets and producing rapid model updates in a cloud computing environment.

The computational graph-based architectures demonstrate the flexibility of decomposing diverse composite functions and redesigning the functions with a new functional form. In the application areas of transportation planning, researchers and planners can further use this method to improve the accuracy and time of computing/estimating systematic utility functions. As a representative example, one can better calculate the logsum term, which is widely used in practice to calculate a broad set of accessibility-oriented planning applications (Miller, 2018). One can further extend conventional modeling structures such as joint-choice models for modeling travelers' multi-dimensional choice decision-making process.

On the one hand, by building choice models through computational graph-based domain-specific languages, modelers can integrate such models easily with external deep learning architectures, leading to enhanced representation of travelers' complex activity patterns. In Fig. 7,

we provide the conceptual framework to illustrate the integration workflow of choice models and deep learning techniques.

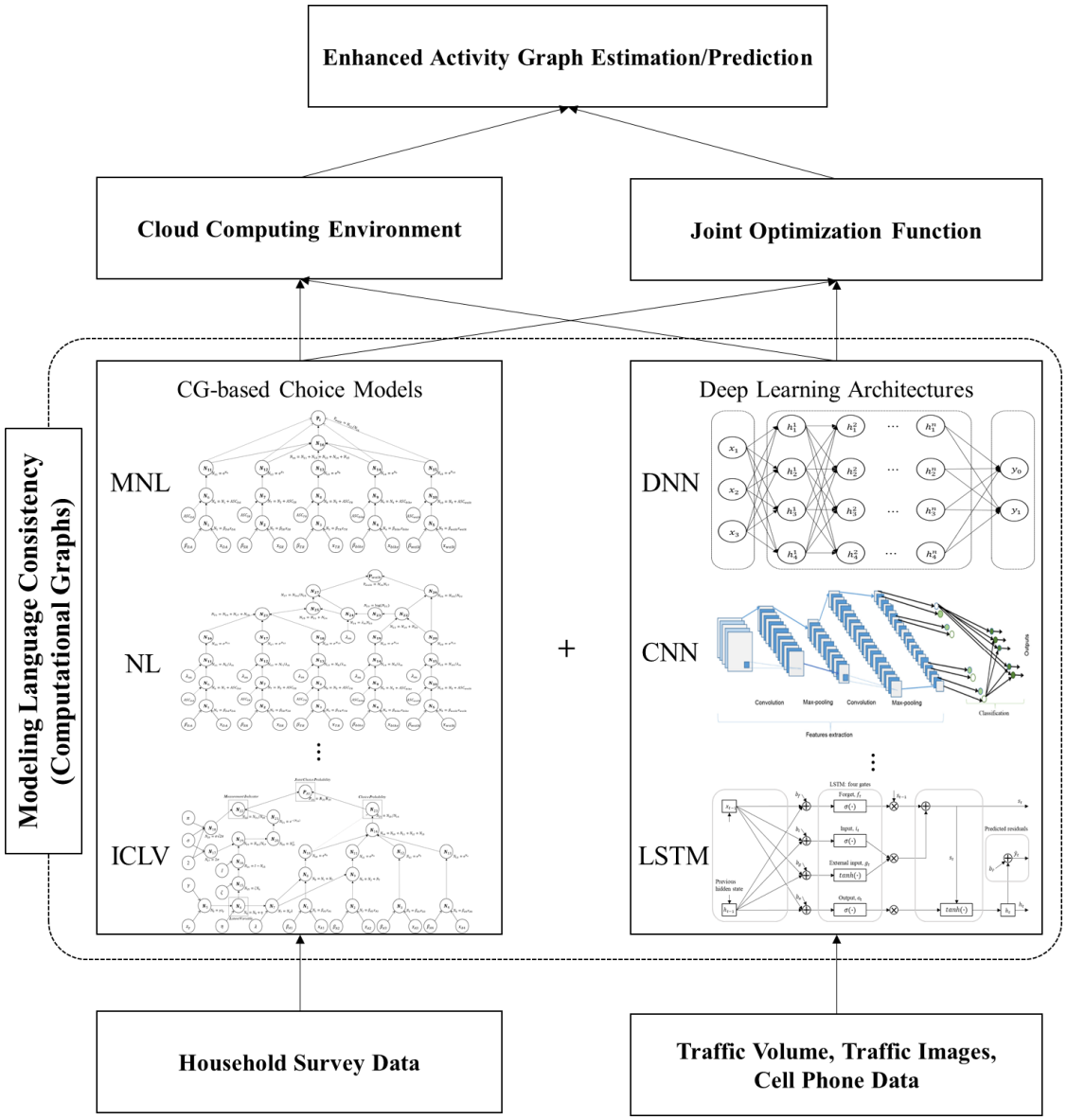

Fig. 7. Illustration of developing a consistent modeling structure between choice models and deep learning (Using examples from CNN in Alom et al., (2019) and LSTM in Kim et al., (2020))

With modeling structures capable of handling different data sources, computational graph-based modeling tools facilitate the estimation of more complex model structures, possibly improving interpretability and predictability. More precisely, the efficiency of the CG-based structures can help to rapidly estimate models that can be applied to synthetic population datasets, which are generated by microsamples and census-based marginal distributions (Ye et al., 2009; Sun et al., 2018). Additionally, since the graph-based structure can facilitate tensor decomposition (TD) efficiently, planners are able to utilize the synthesized data and different large datasets (e.g.,

mobility trajectories or smart-card records), for a better understanding of travelling patterns (Sun and Axhausen, 2016).

To further illustrate our overarching modeling approach, we use the conceptual framework in in Fig. 7 to highlight the needed consistency of modeling language to build behavioral models and machine learning architectures. We hope this CG-oriented perspective could allow us to seamlessly integrate traditional econometric traveler behavior models with new and emerging data-driven approaches. Overall, the proposed graph-based modeling framework not only offers the flexibility of expanding conventional modeling approaches but also enables planners and policy makers to estimate the system-wide utility more precisely for different projects and demand management alternatives, potentially leading to better decisions for improved transportation systems.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge anonymous reviewers and Dr. Michel Bierlaire for providing constructive comments and informative suggestions. This research was supported by the Center for Teaching Old Models New Tricks (TOMNET) (Grant No. 69A3551747116), which is Tier 1 University Transportation Centers sponsored by the US Department of Transportation. The second author is partially supported by NSF Grant No. CMMI 1663657 "Real-time Management of Large Fleets of Self-Driving Vehicles Using Virtual Cyber Tracks".

AUTHOR STATEMENT

The authors confirm contribution to the paper as follows; study conception and design: T. Kim, X. Zhou, R. Pendyala; data preparation: T. Kim; analysis and interpretation of results: T. Kim, X. Zhou, R. Pendyala; draft manuscript preparation: T. Kim, X. Zhou, R. Pendyala. All authors reviewed the results and approved the final version of the manuscript.

REFERENCES

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M. and Kudlur, M., 2016. Tensorflow: A system for large-scale machine learning. In 12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16) (pp. 265-283).

Agrawal, A., Verschueren, R., Diamond, S. and Boyd, S., 2018. A rewriting system for convex optimization problems. Journal of Control and Decision, *5*(1), pp.42-60.

Alom, M.Z., Taha, T.M., Yakopcic, C., Westberg, S., Sidike, P., Nasrin, M.S., Hasan, M., Van Essen, B.C., Awwal, A.A. and Asari, V.K., 2019. A state-of-the-art survey on deep learning theory and architectures. Electronics, 8(3), p.292.

Bartholomew-Biggs, M., Brown, S., Christianson, B. and Dixon, L., 2000. Automatic differentiation of algorithms. Journal of Computational and Applied Mathematics, 124(1-2), pp.171-190.

Bastien, F., Lamblin, P., Pascanu, R., Bergstra, J., Goodfellow, I., Bergeron, A., Bouchard, N., Warde-Farley, D. and Bengio, Y., 2012. Theano: new features and speed improvements. arXiv preprint arXiv:1211.5590.

Baydin, A.G., Pearlmutter, B.A., Radul, A.A. and Siskind, J.M., 2017. Automatic differentiation in machine learning: a survey. The Journal of Machine Learning Research, 18(1), pp.5595-5637.

Beck, T. and Fischer, H., 1994. The if-problem in automatic differentiation. Journal of Computational and Applied Mathematics, 50(1-3), pp.119-131.

Ben-Akiva, M., Walker, J., Bernardino, A.T., Gopinath, D.A., Morikawa, T. and Polydoropoulou, A., 2002. Integration of choice and latent variable models. Perpetual motion: Travel behaviour research opportunities and application challenges, pp.431-470.

Bhavsar, P., Safro, I., Bouaynaya, N., Polikar, R. and Dera, D., 2017. Machine learning in transportation data analytics. In Data analytics for intelligent transportation systems (pp. 283-307). Elsevier.

Bierlaire, M., 2016. PythonBiogeme: a short introduction (Report TRANSP-OR 160706, Ecole Polytechnique F´ed´erale de Lausanne).

Bierlaire, M., 2003. BIOGEME: A free package for the estimation of discrete choice models. In Swiss Transport Research Conference (No. CONF).

Bottou, L., 2010. Large-scale machine learning with stochastic gradient descent. In Proceedings of COMPSTAT'2010 (pp. 177-186). Physica-Verlag HD.

Brathwaite, T. and Walker, J.L., 2018a. Asymmetric, closed-form, finite-parameter models of multinomial choice. Journal of choice modelling, 29, pp.78-112.

Brathwaite, T., & Walker, J. L., 2018b. Causal inference in travel demand modeling (and the lack thereof). Journal of choice modelling, 26, 1-18.

Brathwaite, T., 2017. GitHub repository, https://github.com/timothyb0912/pylogit/

Chang, X., Wu, J., Liu, H., Yan, X., Sun, H. and Qu, Y., 2019. Travel mode choice: a data fusion model using machine learning methods and evidence from travel diary survey data. Transportmetrica A: Transport Science, 15(2), pp.1587-1612.

Chapra, S. C., & Canale, R. P., 2010. Numerical methods for engineers. Boston: McGraw-Hill Higher Education.

Chen, B.Y. and Kwan, M.P., 2020. Special Issue on Spatiotemporal Big Data Analytics for Transportation Applications, Transportmetrica A: Transport Science, 16:1, 1-4.

Chen, C., Ma, J., Susilo, Y., Liu, Y. and Wang, M., 2016. The promises of big data and small data for travel behavior (aka human mobility) analysis. Transportation research part C: emerging technologies, 68, pp.285-299.

Griewank, A. and Walther, A., 2008. Evaluating derivatives: principles and techniques of algorithmic differentiation. Society for Industrial and Applied Mathematics.

Han, Y., Zegras, C., Pereira, F.C. and Ben-Akiva, M., 2020. A Neural-embedded Choice Model: TasteNet-MNL Modeling Taste Heterogeneity with Flexibility and Interpretability. arXiv preprint arXiv:2002.00922.

Hashem, I.A.T., Chang, V., Anuar, N.B., Adewole, K., Yaqoob, I., Gani, A., Ahmed, E. and Chiroma, H., 2016. The role of big data in smart city. International Journal of Information Management, 36(5), pp.748-758.

Hess, S. and Palma, D., 2019. Apollo: A flexible, powerful and customisable freeware package for choice model estimation and application. Journal of choice modelling, 32, p.100170.

Kim, J., Rasouli, S. and Timmermans, H., 2016. A hybrid choice model with a nonlinear utility function and bounded distribution for latent variables: application to purchase intention decisions of electric cars. Transportmetrica A: Transport Science, 12(10), pp.909-932.

Kim, T., Sharda, S., Zhou, X. and Pendyala, R.M., 2020. A stepwise interpretable machine learning framework using linear regression (LR) and long short-term memory (LSTM): City-wide demand-side prediction of yellow taxi and for-hire vehicle (FHV) service. Transportation Research Part C: Emerging Technologies, 120, p.102786.

Kim, T., Zhou, X. and Pendyala, R.M., 2021. GitHub repository, https://github.com/Taehooie/CGChoice

Kingma, D.P. and Ba, J., 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

Koppelman, F.S. and Bhat, C., 2006. A self instructing course in mode choice modeling: multinomial and nested logit models.

Lipton, Z.C., 2018. The mythos of model interpretability. Queue, 16(3), pp.31-57.

McFadden, D., 1974. Conditional logit analysis of qualitative choice behavior. In: Frontiers in Econometrics, vols. 105&142. Academic Press, New York.

McFadden, D., 1978. Modeling the choice of residual location. Transp. Res. Rec. 673.v

Miller, Eric J., 2018. Accessibility: measurement and application in transportation planning, Transport Reviews, 38:5, 551-555

Mokhtarian, P., 2018, August. Why travel surveys matter in the age of big data?. In 2018 National Household Travel Survey Workshop (p. 2).

Molnar, C., 2020. Interpretable machine learning. Lulu. com.

Nocedal, J., Wright, S., 2006. Numerical optimization. Springer Science & Business Media.

Nuzzolo, A. and Comi, A., 2016. Advanced public transport and intelligent transport systems: new modelling challenges. Transportmetrica A: Transport Science, 12(8), pp.674-699.

Paleti, R., Bhat, C.R. and Pendyala, R.M., 2013. Integrated model of residential location, work location, vehicle ownership, and commute tour characteristics. Transportation research record, 2382(1), pp.162-172.

Paredes, M., Hemberg, E., O'Reilly, U. M., & Zegras, C., 2017, Machine learning or discrete choice models for car ownership demand estimation and prediction?. In 2017 5th IEEE International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS) (pp. 780-785). IEEE.

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L. and Lerer, A., 2017. Automatic differentiation in pytorch.

Ribeiro, M.T., Singh, S. and Guestrin, C., 2016, August. " Why should I trust you?" Explaining the predictions of any classifier. In Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining (pp. 1135-1144).

Shannon, C. E., 1948. A mathematical theory of communication. The Bell system technical journal, 27(3), 379-423.

Sifringer, B., Lurkin, V. and Alahi, A., 2018, May. Enhancing discrete choice models with neural networks. In Proceedings of the 18th Swiss Transport Research Conference (STRC), Monte Verità/Ascona, Switzerland (pp. 16-18).

Sifringer, B., Lurkin, V. and Alahi, A., 2020. Enhancing discrete choice models with representation learning. Transportation Research Part B: Methodological, 140, pp.236-261.

Sun, L. and Axhausen, K.W., 2016. Understanding urban mobility patterns with a probabilistic tensor factorization framework. Transportation Research Part B: Methodological, 91, pp.511-524.

Sun, L., Erath, A. and Cai, M., 2018. A hierarchical mixture modeling framework for population synthesis. Transportation Research Part B: Methodological, 114, pp.199-212.

Sun, J., Guo, J., Wu, X., Zhu, Q., Wu, D., Xian, K. and Zhou, X., 2019. Analyzing the Impact of Traffic Congestion Mitigation: From an Explainable Neural Network Learning Framework to Marginal Effect Analyses. Sensors, 19(10), p.2254.

Train, K.E., 2009. Discrete choice methods with simulation. Cambridge university press.

van Kesteren, E.J. and Oberski, D.L., 2019. Structural equation models as computation graphs. arXiv preprint arXiv:1905.04492.

Vij, A. and Walker, J.L., 2016. How, when and why integrated choice and latent variable models are latently useful. Transportation Research Part B: Methodological, 90, pp.192-217.

Williams, H.C., 1977. On the formation of travel demand models and economic evaluation measures of user benefit. Environment and planning A, 9(3), pp.285-344.

Wong, M. and Farooq, B., 2019. ResLogit: A residual neural network logit model. arXiv preprint arXiv:1912.10058.

Wright, S. and Nocedal, J., 1999. Numerical optimization. Springer Science, 35(67-68), p.7.

Wu, X., Guo, J., Xian, K. and Zhou, X., 2018. Hierarchical travel demand estimation using multiple data sources: A forward and backward propagation algorithmic framework on a layered computational graph. Transportation Research Part C: Emerging Technologies, 96, pp.321-346.

Ye, X., Konduri, K., Pendyala, R.M., Sana, B. and Waddell, P., 2009, January. A methodology to match distributions of both household and person attributes in the generation of synthetic populations. In 88th Annual Meeting of the Transportation Research Board, Washington, DC.

Zhao, P., Liu, X., Kwan, M.P. and Shi, W., 2020. Unveiling cabdrivers' dining behavior patterns for site selection of 'taxi canteen'using taxi trajectory data. Transportmetrica A: Transport Science, 16(1), pp.137-160.