Framing Discrete Choice Model as Deep Neural Network with Utility Interpretation

Shenhao Wang, Jinhua Zhao
Department of Urban Studies and Planning
Massachusetts Institute of Technology

Abstract

Deep neural network (DNN) has been increasingly applied to travel demand prediction. However, no study has examined how DNN relates to utility-based discrete choice models (DCM) beyond simple comparison of prediction accuracy. To fill this gap, this paper investigates the relationship between DNN and DCM from a theoretical perspective with three major findings. First, we introduce the utility interpretation to the DNN models and demonstrate that DCM is one special case of DNN with shallow and sparse architecture, identifiable parameters, logistic loss, zero regularization, and domain-knowledge based feature transformation. Second, a sequence of four neural network models illustrate how DNN gradually trade away interpretability for predictability in the context of travel mode choice. High predictability is achieved by DNN’s powerful representation learning and high model capacity; but interpretability is sacrificed through the loss of convex optimization and statistical properties, and non-identification of parameters. Third, the utility interpretation allows us to develop a numerical method of extracting important economic information from DNN including choice probability, elasticity, marginal rate of substitution, and consumer surplus. Overall, this study makes three contributions: theoretically it frames DCM as a special case of DNN and introduces the utility interpretation to DNN; methodologically it demonstrates the interpretability-predictability tradeoff between DCM and DNN and suggests the potential of their joint improvement, and practically it introduces a post-hoc numerical method to extract economic information from DNN and make it interpretable through the utility concept.

Keywords: Deep Neural Network, Discrete Choice Model, Travel Behavior, Utility Interpretation, Machine Learning

1. Introduction

Since the seminal paper by McFadden [1], discrete choice models (DCM) have been widely used to understand how individuals make decisions in economics, marketing, and transportation [2] [3] [4]. But recent studies revealed that deep neural networks (DNN) and generally machine learning classifiers could be used to analyze individual choice scenarios as well. DNN empirically performed better than DCM in many generic empirical studies and specific transportation studies [5] [6] [7]. However, DNN was criticized as a black-box model lacking interpretability, although researchers argued that a model with consistently high prediction accuracy must have captured something [8] [9]. The DCM and DNN models followed very different traditions as the former coming from econometrics and the latter from the machine learning camp [10] [11]. To the best of our
knowledge, no study has examined the underlying relationship between DCM and DNN models beyond simple comparison of prediction accuracy.

To fill this gap, this paper investigates the theoretical relationship between DCM and DNN with focuses on the interpretability-predictability tradeoff and the possibility of interpreting DNN from the utility perspective. We aim to contribute to the frontiers of theory, methodology, and practice of choice modeling. First, we frame DCM as one special case of DNN and introduce utility interpretation to DNN. Second, through a sequence of four neural network (NN) models we illustrate how DNN gradually trade away interpretability for predictability and inform modelers of their methodological options in the context of travel mode choice. Third, based on the utility perspective of DNN, we introduce a post-hoc numerical method to extract economic information from DNN so that researchers can both estimate and interpret the travel behavior models based on DNN. The utility interpretation and the post-hoc numerical method improve DNN interpretability without sacrificing its powerful predictability.

The four NN models are one-layer sparse NN ($A_0$), one-layer dense NN ($A_1$), two-layer dense NN ($A_2$), and multi-layer dense NN ($A_3$). The four models are similar in their shared Softmax layer, based on which we introduced utility interpretation to DNN. The binary choice model (BCM) $A_0$ is the simplest of the four models in terms of model architecture and estimation. As utility specifications from $A_0$ to $A_3$ become more flexible, NN models become stronger in predictability but weaker in interpretability. This process of trading away interpretability for predictability results from the changes in terms of optimization, statistical properties, model dimensionality, and identification of parameters. Instead of sacrificing more interpretability for high predictability, we propose to improve the pareto interpretability-predictability frontier by using the implicit utility interpretation in DNN. We demonstrate this in a simulation experiment in which DCM was assumed to be the true data generating process (DGP) and four NN models ($A_0$ to $A_3$) were used to recover this DGP. We show that more generic models approximate parsimonious DGP well and that we can numerically extract economic information from the DNN models including choice prediction, choice probabilities, elasticity, marginal rates of substitution (MRS), and consumer surplus.

Section 2 synthesizes the relevant literature. Section 3 discusses the generic relationship between DCM and DNN, introduces the four NN architectures, and discusses the interpretability-predictability frontier. Section 4 presents the simulation experiment and the numerical method of extracting economic information from DNN. Section 5 concludes the paper and points out the direction for future research.

2. Literature Review

DCM models serve both interpretation and prediction purposes. Researchers used DCM to both predict travel demand and interpret modeling results by computing value of time, market shares, and elasticity for policy analysis. After McFadden (1974) developed the connection between random utility maximization (RUM) decision rule and logistic regression, researchers used DCM to analyze travel mode choice, travel frequency, travel scheduling, destination and origin choices, route choice, and the choice of using

\[1\] The exact boundary between DNN and NN is unclear because they are not qualitatively very different. We will refer to the four architectures of this study as NN, and only the last NN architecture $A_3$ as DNN
new transit lines, and activity choices from long-run to short-run. The studies aiming to develop DCM models after McFadden (1974) in the transportation and economics fields improved the interpretability and predictability of DCM by making DCM more behaviorally and statistically realistic, such as using decision rules different from RUM, using rank-dependent dataset, incorporating the process of choice formulation, using no choice as one alternative of the choice set, jointly analyzing several datasets, analyzing misspecification of choices, and developing panel regression for choice models.

As machine learning models particularly DNN models permeated into many fields, researchers started to use them to predict travel demand with higher prediction accuracy than DCM models. Karlaftis and Vlahogianni (2011) summarized the transportation fields in which NN models were used, including six fields: 1) traffic operations, 2) infrastructure management and maintenance, 3) transportation planning, 4) environment and transport, 5) safety and human behavior, and 6) air, transit, rail, and freight operations. Notice the literature review was done by 2011, so the number of applications of NN to transportation have been much larger now, given the popularity of DNN in the recent years. However, two problems exist in these applications. First, studies based on DCMs and machine learning models do not connect. Studies based on DCM models typically start with estimation, and then tell stories based on estimated parameters, while those based on machine learning models start with many alternative models, and compare prediction accuracy for model selection. Second, all of the applications of DNN to choice analysis only take advantage of the high prediction accuracy, exploring neither the fundamental relationship between DCM and DNN nor the interpretable components out of DNN. As behavioral and statistical perspectives are two inseparable angles to approach a choice analysis, pure prediction-focused DNN application has discarded many important insights from the classic DCM models.

High prediction accuracy in DNN results from its extraordinary capacity of learning representations. While DCM pre-imposes feature transformation by using domain knowledge, DNN uses generic-purpose architecture to learn representation. Good representations are supposed to be informative in collecting input variations, expressive in fitting labels, and sparse enough to be interpretable. The idea of representation learning with generic NN architecture worked well across domains because NN could extract high-level abstract and generalizable representations. The difference between domain-knowledge-based and generic-purpose representation learning contributes to one key difference between DCM and DNN.

It is understandable that past DNN applications to transportation problems exclusively focused on prediction because DNN is indeed trained to maximize prediction accuracy, and the lack of interpretability in DNN has been a constant challenge in DNN-related studies. For instance, DNN was ranked as the lowest level of interpretability in comparison with decision trees, naive Bayesian models, K nearest neighbors, support vector machines, and rule-based models. Researchers argued that DNN was largely a “black-box” model in many applications. However, it is not true that DNN is totally uninterpretable. Since interpretability is important due to trust, safety, or ethical reasons, recent studies provided many approaches to improve interpretability of DNN. For instance, higher level representations of DNN could disentangle gender and wearing glasses from the image of head portraits. The hidden layers in CNN could identify semantically meaningful objects from different places. Researchers also tried to fit interpretable models to specific instances by LIME and SP-LIME algorithms.
Many other methods could be used to improve the interpretability of DNN [33] [34] [35].

3. Theory

This section starts with the basic setup of DNN and DCM, and then introduces utility interpretation to DNN by identifying the similarity between DCM and DNN. Four models are designed from the simplest DCM to the standard DNN to illustrate their differences in utility specification and consequently in model dimensionality, optimization, and statistical properties. Finally, we summarize the predictability-interpretability tradeoff between DCM and DNN.

3.1. Similarities between DCM and DNN

We will first show that maximum likelihood estimation (MLE) used in DCM is one special case of empirical risk minimization (ERM) problem with logistic loss and zero regularization. The generic ERM problem with $l_2$ penalty is formulated as:

$$\hat{f}_{A_i} = \arg\min_{f \in A_i} \frac{1}{N} \sum_i L(f(w'\Phi(x_i)), y_i) + \lambda ||w||^2$$

(1)

In Equation 1, $y \in \{-1, +1\}$. $l()$ is a generic loss function, and one specific form is the logistic function $l(f(w'\Phi(x_i)), y_i) = \log(1 + e^{-y_iw'\Phi(x_i)})$. $x_i \in \mathbb{R}^d$ is the input, and $\Phi : \mathbb{R}^d \rightarrow \mathbb{R}^p$ is the feature mapping that transforms raw input $x_i$ to richer features in $\mathbb{R}^p$. $w \in \mathbb{R}^p$ is the weights of $\Phi(x_i)$. $\lambda$ is the strength of regularization. When $\lambda \rightarrow 0$ and logistic loss function is used, Equation 1 becomes a more specific form:

$$\hat{f}_{A_i} = \arg\min_{f \in A_i} \frac{1}{N} \sum_i \log(1 + e^{-y_iw'\Phi(x_i)})$$

(2)

While Equations 1 and 2 are both named as ERM, Equation 2 also represents MLE [2], as shown by Equations 3 and 4. Therefore, MLE is one specific ERM with logistic loss and zero regularization.

$$\log(1 + e^{-y_iw'\Phi(x_i)}) = -\log(P(y_i = +1|x_i, w)) = \begin{cases} \frac{-\log(1 + e^{-y_iw'\Phi(x_i)})}{1 + e^{-y_iw'\Phi(x_i)}} & y_i = +1 \\ \frac{-\log(1 + e^{-y_iw'\Phi(x_i)})}{1 + e^{-y_iw'\Phi(x_i)}} & y_i = -1 \end{cases}$$

(3)

$$P(y_i = +1|x_i, w) = \frac{1}{1 + e^{-w'\Phi(x_i)}}$$

$$P(y_i = -1|x_i, w) = \frac{1}{1 + e^{w'\Phi(x_i)}}$$

(4)

Next, we will demonstrate that $w'\Phi(x_i)$ in DNN has an implicit utility interpretation by starting with the derivation of the choice probabilities in DCM framework. Suppose people need to choose between two alternatives 0 and 1 [3], which have the following utility specifications: $U_{i0} = V_{i0} + \epsilon_{i0}$; $U_{i1} = V_{i1} + \epsilon_{i1}$, in which $V$ is the deterministic utility and $\epsilon$ is the random utility term. DCM assumes the decision rule of maximizing utility, and

---

2In this study, we will use ERM as the term rather than MLE since ERM is more generic

3The index 0 and 1 are different from previous -1 and +1 due to different traditions in DCM and DNN. In this study, index 0 and -1 refer to the same alternative.
then the probability of choosing alternative 1 is given by Equation 5. It is based on the assumption that $\epsilon$ follows extreme value distribution $EV(0, 1)$ and $\epsilon_0 - \epsilon_1$ follows a logistic distribution.

$$P(y_i = 1|x_i, w) = P(U_{i1} > U_{i0}) = P(\epsilon_{i0} - \epsilon_{i1} < V_{i1} - V_{i0}) = F_{\epsilon_{i0} - \epsilon_{i1}}(V_{i1} - V_{i0}) = \sigma(V_{i1} - V_{i0}) = \frac{1}{1 + e^{-(V_{i1} - V_{i0})}}$$

$$P(y_i = 0|x_i, w) = 1 - P(y_i = +1|x_i, w) = \frac{1}{1 + e^{(V_{i1} - V_{i0})}}$$

Comparing Equations 4 and 5, we could observe the implicit utility interpretation embedded in DNN, and particularly, the inputs into the last layer of softmax function $w^T\Phi(x_i)$ are utilities. Figure 1 shows a more intuitive image that the inputs into the red output neurons in four models are utilities regardless of the complex transformation prior to the red neurons. Note that Equation 4 is shown through ERM in DNN, while Equation 5 is derived from the decision making rule of RUM. Hence DNN could be treated as one extended version of DCM with nonlinear and complex utility specification with an implicit random utility maximization as its decision rule. McFadden (1974) [19] proved that DCM is sufficient and necessary for random utility maximization, so it suggests that random utility maximization naturally leads to the Softmax activation function, and vice versa [19]. To make our argument more clear, let’s match the exponent terms in Equations 4 and 5, $V_{i1} - V_{i0}$ and $w^T\Phi(x_i)$ by assuming:

$$V_{i0}(x_i) = w_0^T\Phi_0(x_i)$$
$$V_{i1}(x_i) = w_1^T\Phi_1(x_i)$$

(6)

in which $\Phi = [\Phi_0^T \Phi_1^T]^T$, $w = [w_0^T w_1^T]^T$. They jointly show the transformation of $x_i$ into the utilities of alternatives $V_{i0}$ and $V_{i1}$. In this study, we assume that $x_i = [x_{i0}^T, x_{i1}^T, x_i^T]^T$, $x_i \in \mathbb{R}^d$, where $x_{i0}$ and $x_{i1}$ are two vectors of alternative-specific variables; $x_i$ is the vector of individual-specific variables. While the derivation above is limited to binary choice, it can easily extend to multinomial logit model with slightly different notations.

3.2. Differences between DCM and DNN: Four Architectures

While DCM and DNN share utility interpretation, the utility specification of DNN is much richer compared to a parsimonious utility specification in DCM. The different levels of richness in utility specification cause important differences in other modeling considerations, such as optimization, statistical properties, and model capacity. To fully illustrate these differences, we design four NN architectures ($A_0, A_1, A_2, A_3$) from a standard DCM ($A_0$) to a standard DNN ($A_3$), as shown in Figure 1. The four NN architectures represent four model families, $A_i = \{Q(x, w), w \in \Lambda_i\}$; $A_0 \subset A_1 \subset A_2 \subset A_3$. In the four figures in Figure 1, the red neurons are the last layer that performs Softmax transformation; grey neurons represent the input layer; the blue neurons represent hidden layers. While the inputs into the red neurons represent utilities as in Equation 6, their difference is how to specify $w_0, w_1, \Phi_0, \Phi_1$, represented by the blue neurons that perform feature transformation.
3.2.1. A0: One-Layer Sparse Neural Network (DCM)

Equations 7 and 8 represent A0 architecture, same as the specification of a typical DCM with identifiable parameters, as visualized in Figure 1a. In Equations 7 and 8, the deterministic utility terms are specified in a linear way similar to common DCM. The parameters \( w_{00}, w_{10} \) and \( w_{11} \) are associated with three vectors of variables. In this case, \( w = [w_{10}^T \ w_{11}^T \ w_{00}^T]^T \) and \( \Phi(x_i) = [x_i^T \ z_i^T \ - x_{i0}^T]^T, \Phi : \mathbb{R}^d \rightarrow \mathbb{R}^d \).

\[
\begin{align*}
V_{i0} &= w_{00}^T \Phi_0(x_i) = w_{00}^T x_{i0} \\
V_{i1} &= w_{10}^T \Phi_1(x_i) = w_{10}^T x_{i1} + w_{11}^T z_i \\
V_{i1} - V_{i0} &= w_{11}^T \Phi_1(x_i) - w_{00}^T \Phi_0(x_i) \\
&= w_{10}^T x_{i1} + w_{11}^T z_i - w_{00}^T x_{i0} \\
&= [w_{10}^T \ w_{11}^T \ w_{00}^T][x_i^T \ z_i^T \ - x_{i0}^T]^T \\
&= w^T \Phi(x_i)
\end{align*}
\] (8)

This standard DCM utility specification allows the coefficients to be identifiable because \( \Phi(X) \) has full rank and the utility difference does not have perfectly correlated variables. We can not add \( z_i \) to \( V_{i0} \) because otherwise infinite groups of coefficients could generate the same utility difference. With this A0 architecture, researchers...
could interpret the model by starting with estimated \( \hat{w} \) and constructing choice predictions \( \hat{y} \) and choice probability predictions \( \hat{P} \). Note that good statistical properties of \( \hat{w} \) in \( A_0 \) are the prerequisites of computing economic information. Since parameters are identifiable, \( A_0 \) model could achieve both interpretation and prediction purposes.

### 3.2.2. A1: One-Layer Dense Neural Network

The simplest model based on a straightforward NN logic is \( A_1 \), in which the input layer and output layer are fully connected, as visualized in Figure 11. The utility specification of architecture \( A_1 \) is represented in Equations 9 and 10. In this case, \( \Phi(x_i) = [x_{i0}^T x_{i1}^T z_i^T - x_{i0}^T - x_{i1}^T - z_i^T]^T, \Phi : \mathbb{R}^d \rightarrow \mathbb{R}^{2d}. \)

\[
\begin{align*}
V_0 &= w_0^T \phi_0(x_i) = w_{00}^T x_{i0} + w_{01}^T x_{i1} + w_{02}^T z_i \\
V_1 &= w_1^T \phi_1(x_i) = w_{10}^T x_{i0} + w_{11}^T x_{i1} + w_{12}^T z_i \\
V_{11} - V_{01} &= w_1^T \phi_1(x_i) - w_0^T \phi_0(x_i) \\
&= [w_{10}^T w_{11}^T w_{12}^T w_{00}^T w_{01}^T w_{02}^T][x_{i0}^T x_{i1}^T z_i^T - x_{i0}^T - x_{i1}^T - z_i^T]^T \\
&= w^T \phi(x_i)
\end{align*}
\]

The parameters in architecture \( A_1 \) becomes unidentifiable, since the empirical risk function of \( A_1 \) is only weakly convex, as opposed to strongly convex ERM of architecture \( A_1 \). Suppose one group of \( \hat{w} \) could optimize the objective function of \( A_1 \), we could always find other \( \hat{w} \) to achieve the same empirical risk. It can be equivalently seen that Hessian matrix in Equation 10 is not invertable. Clearly the matrix \( \Phi(X) \) is no longer full rank in its columns. In fact, it was exactly this non-identification problem that halted the development of DNNs in the 1990s [20].

The non-identification problem renders statistical methods such as hypothesis testing impossible. When estimating DCM with identifiable parameters, many methods use second order information, such as Newton method \( w^{t+1} = w^{t} - H^{-1} \nabla E(w^t) \). With non-invertable Hessian matrix, it is impossible to run even one iteration. Hence only first order method such as gradient descent could be applied to \( A_1 \), \( A_2 \), and \( A_3 \). Second, classic parameter interpretation is based on consistency: \( \hat{w} \rightarrow p w^* \), where \( w^* \) is the true parameters. Since infinite number of \( \hat{w} \) can achieve the same empirical risk, consistency is impossible to hold. Third, estimators no longer have the asymptotically normal distribution. A Taylor expansion around the \( \hat{w} \) in DCM shows that the estimators follow asymptotically normal distribution with variance equal to \( H^{-1} GH^{-1} \), which is called a sandwich form in econometrics. However, with non-invertable \( H \), we cannot obtain \( H^{-1} \). Hence everything related to this distribution, including p-value and statistical significance, cannot be obtained in even a simple neural network as \( A_1 \), let alone \( A_2 \) and \( A_3 \).

However, from a machine learning perspective, this non-identification is less a problem due to the notion of ”weight space symmetries”, which implies that many groups of parameters can be treated equally as long as they achieve similar prediction accuracy [25]. In Equation 11 there could exist infinite groups of \( \hat{f} \) that achieve the same empirical risk, and it does not matter which one we choose for prediction. Similarly in Equation 10, infinite groups of \( \hat{w} \) could achieve the same utility difference and prediction accuracy. This property of weight space symmetry was conjectured by Hinton (1986) [37] in the 1980s, and further supported in recent review paper by LeCun (2015) [20].

As a result, the non-identification of parameters and the loss of statistical properties in \( A_1 \) weaken the interpretability of NN models. Classic parameter testing cannot be performed due to the loss of statistical consistency and asymptotic normality. Since infi-
nite groups of \( \hat{w} \) can solve the ERM problem, each individual \( \hat{w}_j \) becomes uninterpretable because it is basically an arbitrary number.

### 3.2.3. A2: Two-Layer Dense Neural Network

The utility specification of \( A_2 \) is represented by Equations 11 and 12 as visualized in Figure 1c. \( A_2 \) is a two-layer dense neural network with one hidden layer. This type of two-layer NN was dominant in the machine learning field before 2010.

\[
\begin{align*}
V_{i0} &= w_0^T \Phi_0(x_i) = w_0^T g_1(x_i) \\
V_{i1} &= w_1^T \Phi_1(x_i) = w_1^T g_1(x_i) \\
V_{i1} - V_{i0} &= w_1^T \Phi_1(x_i) - w_0^T \Phi_0(x_i) \\
&= [w_1^T \ w_0^T] [g_1(x_i)^T - g_1(x_i)^T]^T
\end{align*}
\]  
(12)

The hidden layer in \( A_2 \) could be understood through the perspective of representation learning. This representation learning idea is both old and new. On one hand, transportation modelers have been using domain knowledge to learn effective representations for several decades. For instance, when some features represent costs and income, one common transformation is cost/income \cite{4}; when there exists a diminishing marginal impact of income on choice probabilities, squared income as a new feature representation is used. Therefore, we could treat the hidden neurons as cost/income or squared income specified based on domain knowledge. But on the other hand, the representation learning idea is new because NN uses generic purpose functions such as Rectified Linear Unit (ReLU) activation functions for learning representation, rather than using any domain knowledge. In fact, one hidden-layer neural network becomes a universal approximator \cite{38} when the hidden layer is very wide. \( g_1() \) could be polynomial or Gaussian kernels, or simply standard ReLU. NN architectures \( A_0 \) and \( A_1 \) are much smaller model families within the model family represented by \( A_2 \).

Compared to \( A_1 \), one new problem in \( A_2 \) is its high dimensions of its parameter space. As \( \Phi \in R^p, p \to \infty \), models do not explicitly estimate infinite dimensional \( w \). This very high dimensional parameter space further leads to generalization problem. Statistical learning theory suggests that generalizable prediction accuracy cannot be bounded when the ratio between sample size and VC dimension is very small, which could easily happen in \( A_2 \) and \( A_3 \) \cite{39}. It is the typical overfitting problem because models with very high model capacity could literally memorize all training data, and as a result, the model cannot be generalizable \cite{10}. Due to the high dimensionality and representation learning capacity in \( A_2 \), this family of models have much higher predictability and also lower interpretability than \( A_0 \) and \( A_1 \).

### 3.2.4. A3: M-Layer Dense Neural Network

The utility specification of \( A_3 \) is shown in Equations 13 and 14, in which \( g_i(x) \) takes the standard form \( \text{ReLU}(A_i x + b_i) \). In our case, \( A_3 \) is a DNN with eight fully connected layers. After 2010, DNN gradually became dominant in machine learning research.

\footnote{The solution of this problem is through kernel trick: \( \hat{f}(x) = \sum_{i=1}^N k(x, x_i)c_i \), which is shown by the representer theorem. Then the estimation only involves the total number of observations \( N \) as the main dimension to control the computational complexity. With kernel trick, the infinite dimensional \( \Phi \) becomes computationally tractable.}
\[ V_{i0} = w_0^T \Phi_0(x_i) = w_0^T (g_{m-1} \circ g_2 \circ g_1)(x_i) \]
\[ V_{i1} = w_1^T \Phi_1(x_i) = w_1^T (g_{m-1} \circ g_2 \circ g_1)(x_i) \]
\[ V_{i1} - V_{i0} = w_1^T \Phi_1(x_i) - w_0^T \Phi_0(x_i) = w^T \Phi(x_i) \]

(13)

(14)

While NN with one hidden layer \( A_2 \) was proved to be a universal approximator, deeper NN could make the architecture exponentially more efficient \[41\]. Qualitatively, \( A_3 \) is not very different from \( A_2 \), since the previous discussions about non-convex ERM, instability of model parameters, curse of dimensionality, and power of representation learning in \( A_1 \) and \( A_2 \) also apply to \( A_3 \). Theoretical questions concerning \( A_3 \) remain unresolved, such as why we need depth in NN, why this model does not overfit, and how to optimize efficiently. It also remains ambiguous how many layers of NN could be counted as deep.

DNN \( A_3 \) is more flexible than \( A_2 \) due to its depth, particularly suitable for various non-traditional data types such as images and language. For instance, researchers used convolutional neural networks with many layers for image recognition, and CNN architecture simply became deeper and deeper in recent years \[42\], \[43\]. With the depth of NN, researchers could freely create new feature transformation rules as hidden layers. In CNN, the convolutional kernels could be replaced by tiled convolutional layer or variant convolutional kernels, depending on the specific information researchers want to extract \[44\].

3.3. Pareto Frontiers of Interpretability and Predictability

From \( A_0 \) to \( A_3 \), models share the same utility interpretation but are different in their utility specifications. With richer functions for utility specification, models gradually obtain higher predictability, since they have stronger model capacity, higher dimensionality, and more flexibility in representation learning. The findings can be summarized in Figure 2: Figure 2a represents the increasing model capacities from \( A_0 \) to \( A_3 \); Figure 2b represents their tradeoff between interpretability and predictability. As shown in the two figures, model capacity from \( A_0 \) to \( A_3 \) becomes stronger, and predictability becomes higher. The higher predictability of \( A_3 \) can be explained through the interaction between assumed model families and true data generating process (DGP), which can be represented by the four blue dots in Figure 2a. When the blue dot 0 \( (D_0) \) is the true DGP, all four model families could approximate this true model; while when the blue dot 3 \( (D_3) \) is the true DGP, only \( A_3 \) could possibly find the true model. Therefore, unless researchers have a very strong belief in the linear DCM or domain-knowledge based feature transformation, simply for functional approximation purposes, DNN should be used because the true process of human decision-making could be much more complicated than the knowledge researchers have already known.

However, the predictability improvement from \( A_0 \) to \( A_3 \) sacrifices interpretability, by making parameters unidentifiable, losing statistical properties, and using very deep architecture, as shown in Figure 2a. This trajectory from \( A_0 \) to \( A_3 \) is not a pareto improvement of interpretability-predictability. In contrast, the development of DCM, represented by the dash blue arrow in Figure 2b focused mainly on model realism by incorporating more realistic behavioral and statistical components, thus achieving a balanced improvement of interpretability and predictability.
We would argue that choice models in the next generation should be the pareto improvement of the interpretability-predictability frontier based on the powerful prediction-driven DNN models, represented by the solid blue arrow in Figure 2b. Given that $A_3$ is already an efficient universal approximator, we should emphasize obtaining interpretable knowledge from DNN rather than merely focusing on its prediction performance. This goal of improving interpretability becomes possible thanks to the utility interpretation of DNN, as demonstrated below.

4. Experiments

4.1. Experiment Setup

Based on the implicit utility interpretation of DNN, we can extract key economic information for policy and behavioral analysis so that DNN becomes practically interpretable. This section uses a simple simulated experiment to demonstrate this possibility of interpretability improvement and the capacity of $A_3$ to approximate the DGP based on $A_0$. In our experiment, DNN $A_3$ uses eight hidden layers with 100 neurons in each layer, which was used in some past studies \[45\], and more layers may not generate many more insights for our discussion. We only simulated one DGP, which is the $D_0$ in Figure 2a following the simplest $A_0$ structure. We used two input variables $x$. The exact form of our simulation is:

\[
\begin{align*}
U_{i0} &= V_{i0} + \epsilon_{i0} \\
U_{i1} &= V_{i1} + \epsilon_{i1} = w_{10} + w_{11}x_{11} + w_{12}x_{12} + \epsilon_{i1} \\
y_i &= \arg\max_{0,1}\{U_{i0}, U_{i1}\} \\
w_{10} &= w_{11} = w_{12} = 1 \\
x_{11} &\sim N(0, 1); x_{12} \sim N(0, 1);
\end{align*}
\]  

The key economic information elicited from DNN includes choice predictions, choice probabilities, elasticity, marginal rate of substitution (MRS), and consumer surplus (CS).
The second column of Table 1 shows how to numerically compute the five parameters out of DNN, and the third column shows how to analytically compute them in DCM. It is important to note that none of the five key parameters really needs $\hat{w}$ if we start with the initial definitions of these parameters. Even in DCM, we could treat $\hat{w}$ estimation as an intermediate step for model estimation. When we have inputs $x$, outputs $y$ and a black-box model relating inputs and outputs, the five key parameters in DCM could always be computed without explicitly involving $\hat{w}$.

| Parameters              | Numerical Computation | Formula in DCM/DCM |
|-------------------------|------------------------|--------------------|
| 1. Choice Predictions   | $\hat{y}_i$           | $\hat{y}_i$        |
| 2. Choice Probabilities | $\hat{P}_{i1}$        | $\hat{P}_{i1}$     |
| 3. Elasticity           | $\Delta P_{i1}/\Delta x_{i1}$ | $P_{i1}(1 - P_{i1})\hat{w}$ |
| 4. MRS                  | $\Delta P_{i1}/\Delta x_{i2}$ | $w_{i1}/w_{i2}$ |
| 5. Consumer Surplus     | $1/\beta_{i1} \log(e^{V_{i0}} + e^{V_{i1}})$ | $1/\beta_{i1} \log(e^{V_{i0}} + e^{V_{i1}})$ |

Table 1: Formula of Computing Five Parameters

We generated one pair of training and testing sets, and estimated five times based on the same dataset. The final table is the average across five models. The sizes of training and testing sets were both 5,000, which is of the same magnitude of common practices in transportation field. Models were trained by using training set, and all performance measures were computed by using testing set. The training took 10,000 steps and used Adam stochastic gradient descent (SGD) optimization [46].

4.2. Experiment Results

The results of applying four NN architectures to the synthesized dataset are summarized in Table 2 with the average values across five estimates and their errors in parenthesis. All of the four models are very close to the performance of the true model. While the errors of these models seem to deviate more from the true models as models become more complicated, the degree of this deviation is not large. Take DNN $A_3$ as an example. The average prediction of true model should be 0.763 and average probability should be 0.677; as a comparison, the average prediction and prediction errors of $A_3$ are 0.747 and 0.679. Unsurprisingly, the average values of $A_0$ estimators are close to true average values with nearly zero errors. It is because the ratio of model capacity over sample size approaches zero in $A_0$, ERM problem is convex, consistency and asymptotic normality hold, and most importantly the underlying DGP is exactly the same as $A_0$.

Note that since DGP in this experiment is assumed to be $A_0$, simpler models should outperform more complex models; however, we want to emphasize three points here. First, when DGP is outside the $A_0$ model family, which is usually the case in reality, the approximation errors in more complex models start to drop because these models are more likely to incorporate the true DGP, while the estimation errors tend to increase because it is harder to search for the final model in a larger functional space. The construction of this experiment is in the most favor of the DCM because the DCM ($A_0$) specification is exactly correct. When DGP follows $D_1$, $D_2$, or $D_3$ as in Figure 2a, only more complicated models could approximate the true DGP well. But a thorough analysis about the tradeoff between approximation and estimation errors is beyond the scope of our analysis. Second, our experiment is designed to show that it is possible to use numerical method to derive economic information based on utility interpretation across four NN architectures, without delving into the details of this approach’s accuracy and feasibility.
| Average Values of Parameters | True Model | A0 | A1 | A2 | A3 |
|-----------------------------|------------|----|----|----|----|
| Choice Prediction           | 0.763      | 0.762 | 0.766 | 0.765 | 0.747 |
|                             | (-0.001)   | (+0.003) | (+0.002) | (-0.016) |
| Choice Probability          | 0.677      | 0.678 | 0.681 | 0.685 | 0.679 |
|                             | (+0.001)   | (+0.004) | (+0.007) | (+0.002) |
| Average Elasticity          | 0.229      | 0.231 | 0.231 | 0.226 | 0.236 |
|                             | (+0.002)   | (+0.002) | (-0.003) | (+0.007) |
| Average MRS                 | 1.0        | 1.015 | 1.029 | 1.017 | 1.073 |
|                             | (+0.015)   | (+0.029) | (+0.017) | (+0.073) |
| Average CS                  | 1.498      | 1.535 | 1.038 | 0.993 | 1.372 |
|                             | (+0.037)   | (-0.460) | (-0.505) | (-0.126) |

Table 2: Performance of Four Architectures in the Experiment (errors in parenthesis)

Further research is needed to test how viable this method is in real dataset and whether true economic information could be recovered in more simulations. Third, DNN could be dramatically improved given the large number of tools recently developed in deep learning community. We did not use regularization, model ensemble, or any complicated NN architecture to improve $A_3$ performance. $A_3$ has a large potential to be improved.

5. Conclusion and Discussion

This study explores the theoretical relationship between DCM and DNN, with an emphasis on their similarity through utility interpretation, and differences in terms of utility specifications. We firstly argued the similar utility interpretation between DCM and DNN, and showed their different utility specifications and accordingly their differences in terms of optimization, statistical properties, model capacity, and representation learning capacity. Finally, a simulation-based experiment was conducted to compare the four NN architectures based on the simplest DGP assumption. We obtained three groups of insights.

First, we introduce the utility interpretation to DNN and demonstrate that the DCM model is one special case of DNN with shallow and sparse architecture, and identifiable parameter specification. MLE of DCM is one special case of ERM with logistic cost function and no regularization. The DCM way of crafting features based on domain knowledge is one specific idea of general representation learning.

Second, a sequence of the four models with different utility specifications show the tradeoff between interpretability and predictability from DCM to DNN models. High predictability is achieved through DNN’s powerful representation learning capacity, complicated feature engineering functions, large model families and even universal approximators, and high dimensionality of statistical models. Interpretability is sacrificed through the loss of convexity in optimization, non-identification of model parameters, and the loss of statistical properties and tests. We call for a pareto improvement of the interpretability-predictability frontier for the next generation of choice modeling in the transportation field.

Third, the implicit utility interpretation enables researchers to numerically construct key parameters commonly used in travel demand analysis, including choice prediction,
choice probabilities, elasticity, marginal rate of substitution, and consumer surplus. Instead of relying on the estimation of individual \( \hat{\omega} \) as an intermediate step, the numerical approach directly computes economic information. In the most restricted DGP assumed in our experiment, DNN \((A_3)\) on average approximates the highest possible prediction accuracy, true choice probabilities, elasticity, and MRS, although we also find that errors tend to propagate along the complexity of models.

Compared to other machine learning models, such as decision trees, K-nearest neighbors classifiers, and naive Bayesian models, DNN seems special to share so many similarities with DCM in terms of implicit utility interpretation, model family relationships, diverse utility specifications, and numerical method for interpretation. While other machine learning classifiers could have other behavioral interpretations, DNN serves as a natural extension of DCM due to their striking similarities and DNN’s exceptional modeling power and flexibility. It also seems more likely to incorporate domain knowledge of choice modeling in DNN than other machine learning models.

The framing of DCM as DNN raises questions about the source of interpretable information and the importance of convex optimization framework in future choice modeling. In the DCM framework, interpretable information relies on accurately estimated individual parameters and convex optimization. However, we have shown that the valuable economic information such as elasticity and MRS can be obtained in an end-to-end manner without involving individual parameters as an intermediate step. If this is the case, convex optimization of ERM will become less important in future choice modeling. DCM is sometimes overly restrictive due to the convex optimization framework. On the other hand, the flexible utility specification in DNN provides researchers with greater opportunities to capture the richness of human behavior than the DCM approaches.

We highlight the opportunities for future studies in three directions. In terms of theory, while we have examined the relationship between DCM and DNN using simple logit model as the example, future studies can explore whether other DCM models such as nested logit, mixed logit, and generalized extreme value (GEV) models can be framed from the DNN theoretical perspective. We introduced the utility interpretation of DNN as a starting point, and future studies can bring substantive utility theories to DNN so that the machine learning models can be closely connected to the microeconomic behavioral theories. In fact, the DNN model \((A_3)\) in our study did not incorporate any domain-specific knowledge about travel choice, and potentially by introducing domain knowledge accumulated in the field of choice modeling, researchers can design more interesting and efficient DNN models. In terms of methodology, future research can work on the improvement of the pareto frontier of predictability and interpretability with an emphasis on interpretability because of its practical importance and the disregard so far. While DNN has a much larger model family than DCM, it is unclear how DNN can approximate mixture models or data with non-IID structure. Even in the comparison between a simple multinomial logit (MNL) and DNN, many difficult questions remain such as computational complexity, sample size requirement, stability of estimation errors, tradeoff between approximation and estimation errors, and other aspects that machine learning models emphasize while statistical models don’t. In terms of practice, the utility interpretation creates the opportunity of numerically eliciting economic information from DNN models. Future studies should test the robustness of the numerical method we introduced, to examine whether this method can provide valuable information with real world datasets and recover the true information as it did in the simulated experiment. We believe these future directions are important and promising because they respond
to the urgent call for reconciling machine learning models, econometrics models, behavioral theories, and practical needs of better understanding individual behaviors in the transportation field.

6. Acknowledgement

The research is supported by the National Research Foundation (NRF), Prime Minister’s Office, Singapore, under CREATE programme, Singapore-MIT Alliance for Research and Technology (SMART) Centre.

References

[1] D. McFadden, Conditional logit analysis of qualitative choice behavior (1974).

[2] K. Train, A structured logit model of auto ownership and mode choice, The Review of Economic Studies 47 (1980) 357–370.

[3] M. Ben-Akiva, J. Zhao, Behavior economics in transportation, 2014.

[4] K. Small, C. Winston, The demand for transportation: models and applications, in: Essays in Transportation Economics and Policy, 1998.

[5] M. Fernández-Delgado, E. Cernadas, S. Barro, D. Amorim, Do we need hundreds of classifiers to solve real world classification problems, Journal of Machine Learning Research 15 (2014) 3133–3181.

[6] J. Hagenauer, M. Helbich, A comparative study of machine learning classifiers for modeling travel mode choice, Expert Systems with Applications 78 (2017) 273–282.

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

[8] S. B. Kotsiantis, I. Zaharakis, P. Pintelas, Supervised machine learning: A review of classification techniques, Emerging artificial intelligence applications in computer engineering 160 (2007) 3–24.

[9] S. Mullainathan, J. Spiess, Machine learning: an applied econometric approach, Journal of Economic Perspectives 31 (2017) 87–106.

[10] M. G. Karlaftis, E. I. Vlahogianni, Statistical methods versus neural networks in transportation research: Differences, similarities and some insights, Transportation Research Part C: Emerging Technologies 19 (2011) 387–399.

[11] M. Ben-Akiva, M. Bierlaire, D. McFadden, J. Walker, Discrete Choice Analysis, 2014.

[12] K. A. Small, E. T. Verhoef, R. Lindsey, Travel demand, in: The economics of urban transportation, volume 2, Routledge, 2007.

[13] J. De Dios Ortuzar, L. G. Willumsen, Modelling transport, John Wiley and Sons, 2011.
[14] M. Ben-Akiva, J. L. Bowman, D. Gopinath, Travel demand model system for the information era, Transportation 23 (1996) 241–266.

[15] C. G. Chorus, T. A. Arentze, H. J. Timmermans, A random regret-minimization model of travel choice, Transportation Research Part B: Methodological 42 (2008) 1–18.

[16] J. A. Hausman, J. Abrevaya, F. M. Scott-Morton, Misclassification of the dependent variable in a discrete-response setting, Journal of Econometrics 87 (1998) 239–269.

[17] S. Beggs, S. Cardell, J. Hausman, Assessing the potential demand for electric cars, Journal of econometrics 17 (1981) 1–19.

[18] C. R. Bhat, V. Pulugurta, A comparison of two alternative behavioral choice mechanisms for household auto ownership decisions, Transportation Research Part B: Methodological 32 (1998) 61–75.

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

[20] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (2015) 436–444.

[21] N. Jean, M. Burke, M. Xie, W. M. Davis, D. B. Lobell, S. Ermon, Combining satellite imagery and machine learning to predict poverty, Science 353 (2016) 790–794.

[22] E. L. Glaeser, S. D. Kominers, M. Luca, N. Naik, Big data and big cities: The promises and limitations of improved measures of urban life, Economic Inquiry 56 (2018) 114–137.

[23] S. Pulugurta, A. Arun, M. Errampalli, Use of artificial intelligence for mode choice analysis and comparison with traditional multinomial logit model, Procedia-Social and Behavioral Sciences 104 (2013) 583–592.

[24] H. Omrani, Predicting travel mode of individuals by machine learning, Transportation Research Procedia 10 (2015) 840–849.

[25] C. M. Bishop, Pattern recognition and machine learning, springer, 2006.

[26] Y. Bengio, A. Courville, P. Vincent, Representation learning: A review and new perspectives, IEEE transactions on pattern analysis and machine intelligence 35 (2013) 1798–1828.

[27] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, A. Torralba, Object detectors emerge in deep scene cnns, arXiv preprint arXiv:1412.6856 (2014).

[28] M. D. Zeiler, R. Fergus, Visualizing and understanding convolutional networks, in: European conference on computer vision, Springer, 2014, pp. 818–833.

[29] P. W. Koh, P. Liang, Understanding black-box predictions via influence functions, arXiv preprint arXiv:1703.04730 (2017).

[30] R. Brauneis, E. P. Goodman, Algorithmic transparency for the smart city (2017).
[31] A. A. Freitas, Comprehensible classification models: a position paper, ACM SIGKDD explorations newsletter 15 (2014) 1–10.

[32] M. T. Ribeiro, S. Singh, C. Guestrin, 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, ACM, 2016, pp. 1135–1144.

[33] Z. C. Lipton, The mythos of model interpretability, arXiv preprint arXiv:1606.03490 (2016).

[34] G. Hinton, O. Vinyals, J. Dean, Distilling the knowledge in a neural network, arXiv preprint arXiv:1503.02531 (2015).

[35] F. Doshi-Velez, B. Kim, Towards a rigorous science of interpretable machine learning (2017).

[36] M. E. Ben-Akiva, S. R. Lerman, Discrete choice analysis: theory and application to travel demand, volume 9, MIT press, 1985.

[37] D. E. Rumelhart, G. E. Hinton, R. J. Williams, Learning representations by back-propagating errors, nature 323 (1986) 533.

[38] K. Hornik, M. Stinchcombe, H. White, Multilayer feedforward networks are universal approximators, Neural networks 2 (1989) 359–366.

[39] V. N. Vapnik, An overview of statistical learning theory, IEEE transactions on neural networks 10 (1999) 988–999.

[40] C. Zhang, S. Bengio, M. Hardt, B. Recht, O. Vinyals, Understanding deep learning requires rethinking generalization, arXiv preprint arXiv:1611.03530 (2016).

[41] I. Goodfellow, Y. Bengio, A. Courville, Y. Bengio, Deep learning, volume 1, MIT press Cambridge, 2016.

[42] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks, in: Advances in neural information processing systems, pp. 1097–1105.

[43] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, Cvpr, 2015.

[44] N. Kalchbrenner, E. Grefenstette, P. Blunsom, A convolutional neural network for modelling sentences, arXiv preprint arXiv:1404.2188 (2014).

[45] J. Yosinski, J. Clune, Y. Bengio, H. Lipson, How transferable are features in deep neural networks?, in: Advances in neural information processing systems, pp. 3320–3328.

[46] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, arXiv preprint arXiv:1412.6980 (2014).