Trustworthy and Explainable Artificial Neural Networks for Choice Behaviour Analysis

Alwosheel, A.S.A.

DOI
10.4233/uuid:82fcb7b1-153c-4f6f-9d8c-bbdc46cc2d4e

Publication date
2020

Document Version
Final published version

Citation (APA)
Alwosheel, A. S. A. (2020). *Trustworthy and Explainable Artificial Neural Networks for Choice Behaviour Analysis*. [Dissertation (TU Delft), Delft University of Technology]. TRAIL Research School. https://doi.org/10.4233/uuid:82fcb7b1-153c-4f6f-9d8c-bbdc46cc2d4e

Trustworthy and Explainable Artificial Neural Networks for Choice Behaviour Analysis

Ahmad Saleh A Alwosheel

Delft University of Technology

This research has received financial support from King Abdulaziz City for Science and Technology (KACST)

مدينة الملك عبدالعزيز
للعلوم والتقنية KACST

Trustworthy and Explainable Artificial Neural Networks for Choice Behaviour Analysis

Dissertation

for the purpose of obtaining the degree of doctor
at Delft University of Technology
by the authority of Rector Magnificus Prof. dr. ir. T.H.J.J. van der Hagen,
to be defended publicly on
Friday 10 July 2020 at 12:30 o'clock

by

Ahmad Saleh A ALWOSHEEL
Master of Science in Electrical Engineering
University of Southern California
born in Riyadh, Saudi Arabia

This dissertation has been approved by the promotors:
promotor: Prof. dr. ir. C.G. Chorus
copromotor: Dr. ir. S. van Cranenburgh

Composition of the doctoral committee:

Rector Magnificus	chairperson
Prof. dr. ir. C.G. Chorus	Delft University of Technology, promotor
Dr. ir. S. van Cranenburgh	Delft University of Technology, copromotor

Independent members:

Prof. dr. ir. J.W.C. van Lint	Delft University of Technology
Prof. dr. ir. P.H.A.J.M. van Gelder	Delft University of Technology
Prof. dr. F.C. Pereira	Technical University of Denmark
Dr. M.M. de Weerdt	Delft University of Technology
Dr. A. Alahi	Swiss Federal Institute of Technology

TRAIL Thesis Series no. T2020/11, the Netherlands Research School TRAIL

TRAIL
P.O. Box 5017
2600 GA Delft
The Netherlands
E-mail: info@rsTRAIL.nl

ISBN: 978-90-5584-268-1

Printed in the Netherlands

To my parents

To my wife: Bayan

To my boys: Rakan and Sattam

Content

Introduction

1 Research background

“Making decisions is like speaking prose – people do it all the time, knowingly or unknowingly” (Kahneman & Tversky, 2013). A typical day in our life is full of choices, which we make in a variety of contexts, including economical choices (e.g. what to buy from the supermarket), health related choices (e.g. whether to exercise or not), and so on. As such, it is no wonder that choice behaviour is a widely studied topic in fields as diverse as statistics, politics, and economics.

To describe, understand and predict human choice behaviour, Discrete Choice Models (DCMs) have been used for decades in a wide variety of contexts. To name a few examples, they have been used in transportation in order to understand travellers’ behaviour (Hensher & Rose, 2011), in marketing to analyse consumers’ choices (Louviere & Woodworth, 1983), and in an environmental context to estimate environmental values (Bennett & Blamey, 2001). DCMs are used to study choices between different alternatives, to derive the underlying tastes and preferences of individuals. When information regarding the different alternatives are available (e.g. travel times and costs of different modes of travel), DCMs are used to identify the relative weights of attributes assigned by individuals and the decision-making mechanism, providing a valuable understanding of individuals’ choice behaviour. Furthermore, DCMs are used in the evaluation of new products and services to predict future demand.[1]

The field of discrete choice modelling is firmly rooted in economic theory, which is reflected by the fact that its main developer received the Nobel Prize in Economics (McFadden, 2001). Most DCMs are based on the paradigm that decision-makers are assumed to settle for nothing less than the best (McFadden, Machina, & Baron, 1999). The core of standard choice models relies on the assumption that decision-makers, when asked to select an alternative among a set of presented alternatives, make deliberate trade-offs by employing a stable function to assign

[1] For example, DCMs have been useful (and accurate) in predicting demand for new products in the field of transportation (e.g. predicting the demand for a new electric train), see (McFadden, 2002) for an example from the early 1970s.

utility to each alternative, and then select the alternative with the highest utility; hence called utility maximiser.[2] The attributes of the considered alternatives are used to determine the utility they provide, hence utility can be expressed as a function of the attributes (Lancaster, 1966).

How do DCMs work ?

The choice problem consists of observed inputs (i.e. alternative attributes and the decision-maker's characteristics) and outputs (i.e. decisions). Most DCMs are based on utility maximisation, which assumes that the decision-maker selects the alternative with the highest utility. Using the linear-additive random utility framework (McFadden, 1973), the utility function for individual n of i alternative is represented as follows:

$$U_{ni} = V_{ni} + \varepsilon_{ni} = \sum_{m=1}^{M} \beta_m x_{nim} + \varepsilon_{ni} \qquad (1)$$

Where m is the attribute index. V and ε are the deterministic and random parts, respectively. The deterministic part (V) consists of M components representing the observed alternative attributes and the characteristics of the decision-maker (β being the associated parameters to be estimated). The random part is added to take into account the analyst's uncertainty (e.g. unobserved information about the attributes of the alternatives) (Manski, 1977). The so-called logit model is the most-used discrete choice model. It assumes that the random part is independent and identically distributed variable with generalised extreme value type I distribution (of variance $\frac{\pi^2}{6}$), resulting in the closed form probability:

$$P_{ni} = \frac{\exp(V_{ni})}{\sum_{j=1}^{J} \exp(V_{nj})} \qquad (2)$$

J being the number of alternatives considered by the decision-maker. βs are most commonly estimated using maximum likelihood methods.

From a statistical modelling perspective, the standard model (i.e. the random utility maximisation (RUM) model as shown in Equation 1&2) can be seen as a logistic-regression model, with several assumptions purposefully imposed.[3] For instance, the random part is assumed to follow a pre-specified distribution (McFadden, 1973). Furthermore, all decision-makers are assumed to have stable preferences and to consider a fixed set of attributes. As a result of the imposed assumptions, discrete choice models produce closed form probabilities and their parameters provide rigorous economical and behavioural inferences.

There is no doubt that DCMs enjoy popularity across a wide range of fields. This popularity can be attributed to the fact that DCMs offer a *transparent* and *tractable* modelling approach that is deeply rooted in theory. However, there is overwhelming evidence against the rationality assumptions imposed in most choice models (Kahneman & Tversky, 2013). Furthermore, many studies have highlighted that the imposed assumptions may lead to restrictive analysis of human choice behaviour, resulting in biased parameter estimates, lower predictability and incorrect interpretations (Breiman, 2001; Han, Zegras, Pereira, & Ben-Akiva, 2020; Kahneman, 1994; Shmueli, 2010). As such, a recent shift is being made in the choice modelling community to include behavioural and psychological factors and theories that were traditionally ignored. As a result, a wide range of new models that incorporate behavioural and psychological theories

[2] The principle of utility states that behaviours and actions are right, as they promote happiness or pleasure, wrong as they tend to produce unhappiness or pain (White, 2017).

[3] Note that there are a variety of DCMs that are more complex than the linear-additive RUM model (e.g. latent class choice models, see (Train, 2009) for details).

have been developed (e.g. Random Regret Minimisation model (van Cranenburgh, Guevara, & Chorus, 2015)). However, a common feature of all DCMs – traditional and new – is that they are "theory-oriented", in the sense that assumptions (e.g. regarding the choice behaviour) are imposed a priori (based on behavioural theories, for example).

Another way to learn about human choice behaviour can be achieved using approaches that are less theory-reliant and more flexible than discrete choice models. In particular, Artificial Neural Networks (ANNs) surface as an appealing alternative that have gained increasing interest in a wide set of applications. ANNs are mathematical models that are loosely inspired by structural and functional aspects of biological neural systems, and are well-known for being highly effective in solving complex classification and regression problems. Their recent uptake can be attributed to major breakthroughs in ANN research, affecting the daily lives of many people (e.g. in the context of self-driving vehicles, enabling them to recognise traffic signs and navigate routes in complex environments). In particular, the fact that ANNs have the ability to automatically learn and improve from experience (i.e. previous examples), without being explicitly programmed, allows them to achieve impressive results, in some cases better than human experts' performance.[4]

Table 1. Main differences between discrete choice models and artificial neural networks

	DCM	ANN
Philosophy	Data Generating Process is pre-assumed	Data Generating Process is inherently unknown
Goal	Provide insights and inferences	Provide high prediction performance
Model development	Identifies one final solution	Results in multiple (i.e. models are not identifiable)

There are many aspects in which ANNs differ from DCMs, but three main points are highlighted (see Table 1) (Golshani, Shabanpour, Mahmoudifard, Derrible, & Mohammadian, 2018; Karlaftis & Vlahogianni, 2011). The first difference lies in the underlying philosophy of the two approaches. That is, DCMs begin by assuming that the data is generated by a predefined process (e.g. utility maximisation process). In contrast, ANNs' assumption on the data generating process is relaxed (i.e. unknown data generating process is assumed) (Breiman, 2001). The second difference is the goal of each approach: DCMs aim to provide insights and inferences (e.g. by studying elasticities), while the aim of ANNs is to obtain high prediction performance by learning the underlying relationships between independent and dependent variables. The third difference lies in the model development, in which ANNs' flexibility often leads to more than one solution (i.e. models are not identifiable, because the solution space is non-convex) (Goodfellow, Bengio, & Courville, 2016). This is in contrast with DCMs, where models are identifiable (i.e. one final solution is obtained) (Walker, 2001).

[4] For example, a recent study shows that ANN-based models perform better than experts in detecting cancer tumours (Mckinney et al., 2020).

What are the main application fields of discrete choice models ?
DCMs have been successfully used in wide range of fields (see papers cited above). It is, however, worth highlighting that DCMs were originally developed in the context of transportation. For instance, one of the early applications was to estimate the demand for a new transportation service, based on the analysis of individual travel choices (McFadden, 2001). As the travel demand problem is found to be similar to applications such as education and occupation choices, DCMs have been successfully adopted in these applications and beyond, such as marketing and healthcare.
In this PhD thesis, for pragmatic reasons (i.e. to leverage the availability of data, and the expertise of the supervisory team), methods, recommendations and implications have been developed in the context of transportation. It is however important to highlight that the results of this work are not confined to the field of transportation, but are also applicable to other fields where analysing human choice behaviour is needed.

These main differences between the two approaches have encouraged researchers to compare their capabilities, merits and demerits in different contexts, and ideally to look for ways and tactics to merge them in order to get the benefits of both approaches (see section 3 of this chapter for a literature review). However, despite the excitement about the potential of ANN for choice behaviour analysis, many choice behaviour analysts are reluctant to use ANN models mainly because of the lack of trust in them and their deliverables (e.g. predictions). That is, the superior prediction performance of ANNs comes at a cost, this being increasing the complexity of ANNs to a level that makes their reasoning a mystery (i.e. the black-box issue). This leaves the analysts in the dark about whether ANN predictions are based on intuitively correct and expected rationale or not. Without sufficient understanding of how and why a model makes predictions, choice behaviour analysts remain unsure about the extent to which they can trust the trained ANN. As such, the use of ANNs is mainly confined to niche settings where prediction performance is highly valued (e.g. travel route recommendations) and model transparency is not of great importance. However, for many applications of choice behaviour analysis (e.g. a cost-benefit analysis of publicly funded projects), model transparency is considered a prerequisite for justifiable reasons (e.g. transparent governance). Another and perhaps less acknowledged point is that it is unknown what the required sufficient sample size is for training ANNs to deliver reliable results. This is particularly important because ANNs are recognised for consuming large amounts of data (to estimate the model) and are often used in fields where data sets are at the analysts' disposal (e.g. sentiment analysis of social media text), while many datasets used by choice behaviour analysts are considerably smaller.

2 Research goal

Considering the above-mentioned advantages and limitations of using ANNs to analyse choice behaviour, the main goal of this thesis is formulated as follows:

To explore the potentials and limitations of using ANNs for analysing choice behaviour, and to learn from classical ANN application fields (particularly computer vision) about how ANN-based methods can be improved to increase their usefulness in analysing human choice behaviour.

3 ANNs for choice behaviour analysis – A brief literature review

This section aims to identify the main trends in how choice behaviour analysts work with ANNs by presenting a brief literature review of the related studies. To gather research articles for the study, several search engines and databases were used: Google Scholar, ScienceDirect and Scopus. The keywords used in searching were "artificial neural networks" combined with "choice model" and "transportation". The studies reviewed are shown in Table 2 and can be categorised into:

1. **Comparative studies**: A considerable number of the articles reviewed fall under this category, where the focus is to compare ANNs (as well as many other machine learning models) to their counterpart DCMs for choice behaviour analysis. The vast majority of these studies are in the context of transport mode choice behaviour. Most of these studies have highlighted the trade-off relation between prediction performance and model interpretability (i.e. better prediction performance is provided by ANNs at the cost of model interpretability).
2. **Enhancement and hybrid studies**: Under this category, studies aim to either employ ANNs' properties and techniques to enhance/augment DCMs, or to take it a step further by proposing a hybrid ANN-DCM approach. For instance, (Sifringer, Lurkin, & Alahi, 2018) used properties of ANNs to form the utility based choice model and proposed a hybrid approach between ANN and DCMs to increase the model prediction performance, while maintaining the model's interpretability.
3. **Capitalisation studies**: The objective of these studies is to use (or improve the use of) ANNs to analyse aspects of human choice behaviour that were deemed difficult for discrete choice models. The main difference between this category and the second category (i.e. enhancement and hybrid studies) is that ANNs are used directly (or the use of ANNs is improved) to solve challenging problems of choice behaviour analysis (i.e. DCMs are either not used at all or are only used for comparison and validation purposes). For example, (Pereira, 2019) proposed using an ANN-based algorithm for representing travel behaviour variables. Another example is by (Wang, Wang, & Zhao, 2019) where an ANN-based approach was proposed to combine revealed and stated preference data.
4. **Illuminating ANN black-box studies**: The aim of studies under this category is to investigate the ANN black-box issue and propose strategies and solutions to overcome this issue. Despite the fact that the ANN black-box issue is widely reported (e.g. in most studies under the first category), many studies have used (or proposed using) sensitivity analysis to determine the importance of independent variables, for example (Golshani et al., 2018). When studies that proposed the use of sensitivity analysis are excluded, there is no research that attempts to solve this limitation, to the best of our knowledge.

Table 2. Studies in which ANNs are used for choice behavior analysis

Study	Main topic	Category
(Hensher & Ton, 2000; Xie, Lu, & Parkany, 2003)	Comparative study of ANNs and DCMs in the context of commuter mode choice.	1

(Cantarella & de Luca, 2005)	Comparative study of ANNs and DCMs in the context of travel mode choice.	1
(Karlaftis & Vlahogianni, 2011)	Discussing the differences and similarities between ANNs and DCMs.	1
(Omrani, Charif, Gerber, Awasthi, & Trigano, 2013)	Using an ANN-based model for individual travel mode prediction.	1
(Hagenauer & Helbich, 2017)	Comparative study of machine learning methods (including ANNs) and DCMs in the context of travel mode choice.	1
(Lee, Derrible, & Pereira, 2018)	Comparative study of four types of ANNs and DCMs in the context of travel mode choice.	1
(Petersen, Rodrigues, & Pereira, 2019)	Using ANN-based models for bus travel time prediction.	3
(Golshani et al., 2018)	Comparative study of ANNs and DCMs in the context of mode choice behavior and trip departure time.	1
(Wong, Farooq, & Bilodeau, 2018)	Using ANNs for analyzing underlying latent behavior in decision making.	3
(Saadi, Wong, Farooq, Teller, & Cools, 2017)	Using machine learning approaches (including ANNs) for characterizing and forecasting the short-term demand for on-demand ride-hailing services.	3
(Wong & Farooq, 2019)	Integrating an ANN-based model in the random utility maximisation paradigm.	2
(van Cranenburgh & Kouwenhoven, 2019)	An ANN-based approach to Recover the Value-of-Travel-Time Distribution.	3
(Wang et al., 2019)	An ANN-based approach to combine Revealed and Stated preference data.	3
(Wang & Zhao, 2018)	Using an ANN-based approach to analyze travel mode choice with interpretable economic information.	3
(Sifringer et al., 2018)	Enhancing DCMs with neural networks.	2
(Pereira, 2019)	Using an ANN-based algorithm to represent travel behavior variables.	3

(Wang & Zhao, 2019)	Designing a novel ANN structure using behavioral knowledge.	3
(Han et al., 2020)	Developing a neural network embedded choice model to improve the flexibility in modelling taste heterogeneity while keeping model interpretability.	2&3
(Wong & Farooq, 2020)	Examining the use of a generative machine learning approach for analyzing multiple discrete-continuous travel behavior data.	3

Several observations can be made based on Table 2. First, although many of the articles reviewed highlighted the ANNs' black-box issue (i.e. ANNs are difficult to interpret and it is challenging to identify which independent variables are the most important, for example), there is almost no attempt to overcome this issue (except a few studies that used or proposed using a sensitivity analysis based approach). In the classical fields of ANNs (e.g. computer vision), investigating methods and strategies to overcome the black-box issue is an active research trend (see (Olah et al., 2018), for example). It is surprising to see that this line of research is capturing relatively modest interest in the field of choice behaviour analysis where ANNs are increasingly used and a high premium is assigned to model interpretability. Second, a considerable number of the studies focused on the transport mode choice behaviour problem. However, as of 2017, we observe that the number of choice behaviour analysis applications in which ANNs are used has grown significantly (e.g. combining RP and SP data using ANNs by (Wang et al., 2019), see Table 2).

Note that this literature review focuses only on the major trends of how choice behaviour analysts are using ANNs. Readers interested in a recent review on how emerging machine learning methods (including ANNs) are used in one of the main DCMs domains (mode choice behaviour analysis), are referred to (Hillel, Bierlaire, & Jin, 2019).

4 Research focus and methods

To achieved the above-stated goal, the first study of this thesis investigates the minimum sample size required (for an ANN) to reliably learn and capture the relationships between the independent and dependent variables. As the data in the machine learning community are considered to be the entire universe (i.e. data contain independent and dependent variable relationships and the main objective of machine learning models is to learn/capture the relationships directly from data), it is unknown which appropriate sample size is needed for training ANNs in the context of choice behaviour analysis. The second part of this thesis focuses on investigating the black-box issue of ANNs. That is, compared to conventional choice models where the estimation results can be directly and meaningfully interpreted in terms of attribute-weights, elasticities and the like, the interpretability of a trained ANNs weights is very limited. Further, although ANNs' prediction performance is superior to their counterpart choice models, ANNs' predictions cannot be easily understood. As such, two studies (out of four) in this thesis are devoted to this topic. Finally, this thesis tackles the decision rule heterogeneity (which is an aspect of choice behaviour analysis) using a novel ANN structure. Details of each study are as follows:

4.1 Study 1: Sample size requirements when using ANNs for choice behaviour analysis

For reliable and trustworthy ANNs, the dataset (on which the ANN is estimated/trained) needs to be sufficiently large (i.e. consist of a sufficient number of observations). Compared to their counterpart statistical models (e.g. DCMs), ANNs are known for being highly complex in the sense that they are typically constructed of a large number of parameters. As a result, ANNs are expected to consume datasets for training, that are larger in size. In the literature about ANNs, these data requirements have been studied extensively, leading to a series of theoretical results regarding the lower bounds in terms of sample size for a variety of ANN architectures. However, these results rely on a number of assumptions which are very hard to work with in real life applications (Abu-Mostafa, Magdon-Ismail, & Lin, 2012). As such, the ANN community – of scholars and practitioners alike – works with simple rules-of-thumb. In general, these rules-of-thumb are a factor for certain characteristics of the prediction problem. The most widely used rule-of-thumb is that the sample size needs to be at least 10 times the number of weights in the network (Haykin, 2009).

Despite the increasing number of ANN applications to analyse choice behaviour, it is unknown what sample size requirements are appropriate when using ANNs. Therefore, the first research sub-goal of this thesis is:

Research sub-goal no. 1: To investigate the minimum sample size required for reliable implementation of ANNs for choice behaviour analysis

To achieve this goal, the first study of this thesis empirically examines to what extent the widely used "factor 10" rule-of-thumb holds in the context of choice behaviour analysis (and if this rule does not hold, to propose a new rule-of-thumb). To do so, extensive Monte Carlo analyses using a series of different model specifications with different levels of model complexity have been conducted. Furthermore, the analysis of ANNs' data requirements for choice modelling has been extended beyond synthetic data to several real data sets that have been extensively reported in existing literature about choice modelling.

4.2 Study 2: Using prototypical examples to diagnose ANNs for choice behaviour analysis

Many choice modellers are critical about using ANNs, and rightfully so, because they are hard to diagnose. That is, for analysts it is not possible to see whether a trained (estimated) ANN has learned intuitively reasonable relationships, as opposed to spurious, inexplicable or otherwise undesirable ones. As a result, choice modellers often find it difficult to trust an ANN, even if its predictive performance is strong. Therefore, the following research sub-goal has been formulated:

Research sub-goal no. 2: To develop a diagnostic method for trained ANN models

To tackle this issue, inspired by research in the computer vision field, this study pioneers a low-cost and easy-to-implement methodology to diagnose ANNs in the context of choice behaviour analysis. The method involves synthesising prototypical examples after having trained the ANN. These prototypical examples expose the fundamental relationships that the ANN has learned. These, in turn, can be evaluated by the analyst to see whether they make sense and are desirable, or not. In this study we show how to use such prototypical examples in the context of choice data and we discuss practical considerations for successfully diagnosing ANNs.

Furthermore, the main findings are cross-validated using techniques from traditional discrete choice analysis.

4.3 Study 3: Explaining predictions of ANN-based choice behaviour analysis

This study also focuses on the black-box issue of ANNs, but takes a rather different perspective from study 2. Unlike study 2 where the objective is to diagnose the model as a whole, the focus here is on the limited explainability of individual predictions made by trained ANNs. That is, it is very difficult to assess whether or not particular ANNs' predictions are based on intuitively reasonable relationships. As a result, it is difficult for the analyst to trust predictions and act accordingly. Therefore, the following research sub-goal has been formulated:

Research sub-goal no. 3: To develop a method to explain individual predictions made by trained ANNs

To achieve this goal, this study begins by showing that approaches that are often used (i.e. sensitivity analysis) to explain individual predictions are ill-suited for understanding the inner workings of ANNs. Subsequently, we introduce to the domain of travel choice behaviour analysis an alternative method, inspired by recent progress in the field of computer vision. This method is based on a re-conceptualisation of the idea of heat maps to explain the predictions of a trained ANN. To create a heat map, a prediction of an ANN is propagated backward in the ANN towards the input variables, using a technique called Layer-wise Relevance Propagation (LRP). The resulting heat map shows the contribution of each input value. By doing this, the heat map reveals the rationale behind the prediction in a way that is understandable to humans. If the rationale makes sense to the analyst, she or he will gain trust in the prediction. If not, the analyst may choose to adapt or re-train the ANN or decide not to use it.

4.4 Study 4: An ANN-based approach to investigate decision rules

Recent advances in ANNs exhibit unprecedented success at solving complex problems in a variety of fields. To capitalise on the success of ANNs, this research is devoted to studying how ANNs can be used to tackle the decision rule heterogeneity, which is among the challenging problems in choice behaviour analysis. That is, decision rules are the decision mechanisms humans use when making choices, and they are embedded in discrete choice models. Although the vast majority of discrete choice models are built on a single decision rule (predominantly random utility maximisation), there is a growing recognition amongst researchers that decision-makers are heterogeneous in terms of their decision rules. Also, it is increasingly acknowledged that insights into decision rule heterogeneity are crucial for understanding and predicting human choice behaviour. To capture decision rule heterogeneity, choice behaviour analysts often rely on latent class choice models. However, previous studies have shown that a major methodological shortcoming of latent class models lies in their inability to disentangle decision rule heterogeneity from taste heterogeneity. Therefore, the following research sub-goal has been formulated:

Research sub-goal no. 4: To investigate the capabilities of ANNs to capture the decision rules heterogeneity

In this study, a novel ANN-based approach to investigate decision rule heterogeneity has been developed. The developed ANN is trained in such a way that it can recognise the choice patterns of four distinct decision rules: Random Utility Maximisation, Random Regret Minimisation, Lexicographic, and Random. Next, the trained ANN was used to classify the respondents from a recent choice experiment in terms of the decision rule they would most likely employ. Main findings were cross-validated by comparing the results with those from: (1) single class discrete choice models estimated on subsets of the data, and (2) latent class discrete choice models.

5 Thesis outline

The chapters of this thesis are based on journal articles that were either already published or, at the time of writing, they were under review. The text is completely identical to the published work. An overview of the thesis is presented in Figure 1. The chapters of this thesis are structured as follows:

Capitalisation studies: aiming to use (or improve the use of) ANNs to analyse aspects of human choice behaviour that are deemed to be difficult to discrete choice models. Chapters 2 and 5 fall under this category. Chapter 2 contains the empirical study of sample size requirements when using ANNs for choice behaviour analysis. Chapter 5 presents a novel ANN based solution to investigate the decision rule heterogeneity.

Illuminating ANN black-box studies: aiming to investigate the ANN black-box issue and propose strategies and solutions to overcome this issue. Chapters 3 and 4 fall under this category. Chapter 3 presents a method developed to diagnose the rationale of trained ANNs. Moving forward, Chapter 4 contains the study of the explainability of individual predictions made by trained ANNs.

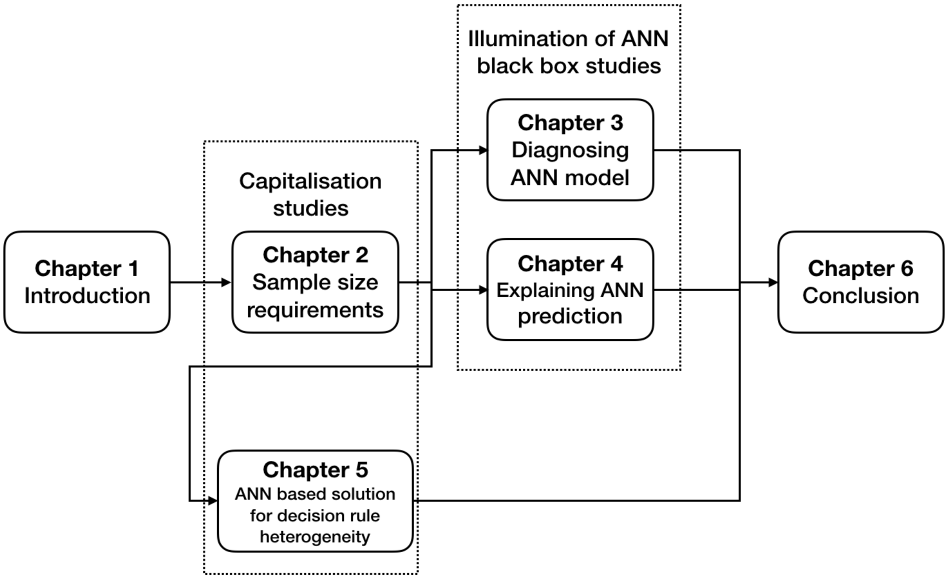

Illumination of ANN black box studies
Capitalisation studies
Chapter 1 Introduction
Chapter 2 Sample size requirements
Chapter 3 Diagnosing ANN model
Chapter 4 Explaining ANN prediction
Chapter 5 ANN based solution for decision rule heterogeneity
Chapter 6 Conclusion

Figure 1. Organisation of the thesis.

References

Abu-Mostafa, Y. S., Magdon-Ismail, M., & Lin, H.-T. (2012). *Learning from data* (Vol. 4): AMLBook New York, NY, USA:.

Bennett, J., & Blamey, R. (2001). *The choice modelling approach to environmental valuation*: Edward Elgar Publishing.

Breiman, L. (2001). Statistical modeling: The two cultures (with comments and a rejoinder by the author). *Statistical science, 16*(3), 199-231.

Cantarella, G. E., & de Luca, S. (2005). Multilayer feedforward networks for transportation mode choice analysis: An analysis and a comparison with random utility models. *Transportation Research Part C: Emerging Technologies, 13*(2), 121-155.

Golshani, N., Shabanpour, R., Mahmoudifard, S. M., Derrible, S., & Mohammadian, A. (2018). Modeling travel mode and timing decisions: Comparison of artificial neural networks and copula-based joint model. *Travel Behaviour and Society, 10*, 21-32.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning* (Vol. 1): MIT press Cambridge.

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

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

Haykin, S. S. (2009). *Neural networks and learning machines* (Vol. 3): Pearson Upper Saddle River.

Hensher, D. A., & Rose, J. (2011). *Choice Modelling: Foundational Contributions*: Edward Elgar Publishing.

Hensher, D. A., & Ton, T. T. (2000). A comparison of the predictive potential of artificial neural networks and nested logit models for commuter mode choice. *Transportation Research Part E: Logistics and Transportation Review, 36*(3), 155-172.

Hillel, T., Bierlaire, M., & Jin, Y. (2019). *A systematic review of machine learning methodologies for modelling passenger mode choice*. Retrieved from

Kahneman, D. (1994). New challenges to the rationality assumption. *Journal of Institutional and Theoretical Economics (JITE)/Zeitschrift für die gesamte Staatswissenschaft*, 18-36.

Kahneman, D., & Tversky, A. (2013). Choices, values, and frames *Handbook of the fundamentals of financial decision making: Part I* (pp. 269-278): World Scientific.

Karlaftis, M. G., & Vlahogianni, E. I. (2011). Statistical methods versus neural networks in transportation research: Differences, similarities and some insights. *Transportation Research Part C: Emerging Technologies, 19*(3), 387-399.

Lancaster, K. J. (1966). A new approach to consumer theory. *Journal of political economy, 74*(2), 132-157.

Lee, D., Derrible, S., & Pereira, F. C. (2018). Comparison of four types of artificial neural network and a multinomial logit model for travel mode choice modeling. *Transportation Research Record, 2672*(49), 101-112.

Louviere, J. J., & Woodworth, G. (1983). Design and analysis of simulated consumer choice or allocation experiments: an approach based on aggregate data. *Journal of marketing research, 20*(4), 350-367.

Manski, C. F. (1977). The structure of random utility models. *Theory and decision, 8*(3), 229.

McFadden, D. (1973). Conditional logit analysis of qualitative choice behavior.

McFadden, D. (2001). Economic choices. *American economic review, 91*(3), 351-378.

McFadden, D., Machina, M. J., & Baron, J. (1999). Rationality for economists? *Elicitation of preferences* (pp. 73-110): Springer.

McFadden, D. L. (2002). The path to discrete-choice models.

Mckinney, S. M., Sieniek, M., Gilbert, F., Godbole, V., Godwin, J., Antropova, N., . . . Corrado, G. C. (2020). International evaluation of an AI system for breast cancer screening. *Nature, 577*, 89-94. doi:10.1038/s41586-019-1799-6

Olah, C., Satyanarayan, A., Johnson, I., Carter, S., Schubert, L., Ye, K., & Mordvintsev, A. (2018). The building blocks of interpretability. *Distill, 3*(3), e10.

Omrani, H., Charif, O., Gerber, P., Awasthi, A., & Trigano, P. (2013). Prediction of individual travel mode with evidential neural network model. *Transportation Research Record, 2399*(1), 1-8.

Pereira, F. C. (2019). Rethinking travel behavior modeling representations through embeddings. *arXiv preprint arXiv:1909.00154*.

Petersen, N. C., Rodrigues, F., & Pereira, F. C. (2019). Multi-output bus travel time prediction with convolutional LSTM neural network. *Expert Systems with Applications, 120*, 426-435.

Saadi, I., Wong, M., Farooq, B., Teller, J., & Cools, M. (2017). An investigation into machine learning approaches for forecasting spatio-temporal demand in ride-hailing service. *arXiv preprint arXiv:1703.02433*.

Shmueli, G. (2010). To explain or to predict? *Statistical science, 25*(3), 289-310.

Sifringer, B., Lurkin, V., & Alahi, A. (2018). *Enhancing Discrete Choice Models with Neural Networks.* Paper presented at the hEART 2018–7th Symposium of the European Association for Research in Transportation conference.

Train, K. E. (2009). *Discrete choice methods with simulation*: Cambridge university press.

van Cranenburgh, S., Guevara, C. A., & Chorus, C. G. (2015). New insights on random regret minimization models. *Transportation Research Part A: Policy and Practice, 74*, 91-109.

van Cranenburgh, S., & Kouwenhoven, M. (2019). *Using Artificial Neural Networks for Recovering the Value-of-Travel-Time Distribution.* Paper presented at the International Work-Conference on Artificial Neural Networks.

Walker, J. L. (2001). *Extended discrete choice models: integrated framework, flexible error structures, and latent variables.* Massachusetts Institute of Technology.

Wang, S., Wang, Q., & Zhao, J. (2019). Multitask Learning Deep Neural Networks to Combine Revealed and Stated Preference Data. *arXiv preprint arXiv:1901.00227*.

Wang, S., & Zhao, J. (2018). Using Deep Neural Network to Analyze Travel Mode Choice With Interpretable Economic Information: An Empirical Example. *arXiv preprint arXiv:1812.04528*.

Wang, S., & Zhao, J. (2019). Deep Neural Networks for Choice Analysis: Architectural Design with Alternative-Specific Utility Functions. *arXiv preprint arXiv:1909.07481*.

White, R. F. (2017). Moral inquiry. *Retrieved January 31st*.

Wong, M., & Farooq, B. (2019). ResLogit: A residual neural network logit model. *arXiv preprint arXiv:1912.10058*.

Wong, M., & Farooq, B. (2020). A bi-partite generative model framework for analyzing and simulating large scale multiple discrete-continuous travel behaviour data. *Transportation Research Part C: Emerging Technologies, 110*, 247-268.

Wong, M., Farooq, B., & Bilodeau, G.-A. (2018). Discriminative conditional restricted Boltzmann machine for discrete choice and latent variable modelling. *Journal of Choice Modelling, 29*, 152-168.

Xie, C., Lu, J., & Parkany, E. (2003). Work travel mode choice modeling with data mining: decision trees and neural networks. *Transportation Research Record, 1854*(1), 50-61.

Sample size requirements when using artificial neural networks for discrete choice analysis

Alwosheel, A., van Cranenburgh, S., & Chorus, C. G. (2018). Is your dataset big enough? Sample size requirements when using artificial neural networks for discrete choice analysis. Journal of choice modelling, 28, 167-182.

Abstract:
Artificial Neural Networks (ANNs) are increasingly used for discrete choice analysis. But, at present, it is unknown what sample size requirements are appropriate when using ANNs in this particular context. This paper fills this knowledge gap: we empirically establish a rule-of-thumb for ANN-based discrete choice analysis based on analyses of synthetic and real data. To investigate the effect of complexity of the data generating process on the minimum required sample size, we conduct extensive Monte Carlo analyses using a series of different model specifications with different levels of model complexity, including RUM and RRM models, with and without random taste parameters. Based on our analyses we advise to use a minimum sample size of fifty times the number of weights in the ANN; it should be noted, that the number of weights is generally much larger than the number of parameters in a discrete choice model. This rule-of-thumb is considerably more conservative than the rule-of-thumb that is most often used in the ANN community, which advises to use at least ten times the number of weights.

1 Introduction

Artificial Neural Networks (ANNs) are receiving an increasing interest from the choice modelling community to analyse choice behaviour in a variety of contexts (e.g., Hagenauer & Helbich, 2017; Hensher & Ton, 2000; Mohammadian & Miller, 2002; van Cranenburgh & Alwosheel, 2019). This recent and profound increase in interest is due to 1) a range of recent innovations in ANN research – leading to improved performance; 2) the availability of “clickn'play" software to work with ANNs; 3) a rapid increase in computational resources, and 4) the increasing volumes and diversity of data which is at the disposal of choice modellers; this latter aspect being the core focus of the current special issue in the Journal of Choice Modelling.

To successfully train ('estimate' in choice modellers' parlance) and use ANNs, the dataset (on which the ANN is trained) needs to be sufficiently large (i.e., consist of a sufficient number of observations). In the ANNs literature such data requirements have extensively been studied (Anthony & Bartlett, 2009; Bartlett & Maass, 2003; Haussler, 1992a), leading to a series of theoretical results regarding lower bounds in terms of data size for a variety of ANNs architectures. However, these results rely on a number of assumptions which are very hard to work with in real life applications (Abu-Mostafa, Magdon-Ismail, & Lin, 2012; Haussler, 1992b). As such, despite that these theoretical results are out there and perhaps because of the fact that in machine learning contexts ample of data are usually available, the ANN community – of scholars and practitioners alike – works with simple rules-of-thumb. In general, these rules-of-thumb are a factor of certain characteristics of the prediction problem. One rule-of-thumb is that the sample size needs to be at least a factor 50 to 1000 times the number of prediction classes (which, in the choice modelling context, is the choice set size) (Cho, Lee, Shin, Choy, & Do, 2015; Cireşan, Meier, & Schmidhuber, 2012). Another rule-of-thumb is that the sample size needs to be at least a factor 10 to 100 times the number of the features (which, in the choice modelling context, is the number of attributes) (Jain & Chandrasekaran, 1982; Kavzoglu & Mather, 2003; Raudys & Jain, 1991).[5] However, the most widely used rule-of-thumb is that the sample size needs to be at least a factor 10 times the number of weights in the network (Abu-Mostafa, 1995; Baum & Haussler, 1989; Haykin, 2009).

Despite the increasing number of applications of ANNs to analyse choice behaviour (see papers cited above, and references cited therein), to the best of the authors' knowledge no study has yet investigated the size of the data that is actually required for meaningful and reliable discrete choice analysis using ANNs. Despite the fact that emerging datasets used for discrete choice analysis tend to be relatively large, many datasets used by choice modellers typically contain somewhere between a couple of hundred and a couple of thousand observations – which is considerably smaller than those sample sizes typically used in the machine learning community. Therefore, it is important to establish what dataset sizes are in fact needed for reliable ANN-based choice modelling efforts, and whether or not conventional dataset sizes used in our community are sufficient in that regard. More specifically, it is important to establish whether the widely used rule-of-thumb to use at least 10 times the number of weights of the network also applies in the context of discrete choice analysis. A related knowledge gap addressed in this paper concerns the effect of the complexity of the data generation process (i.e., the choice model) on the required sample size. Intuitively, it is expected that the more complex (e.g., non-linear) the data generating process is, the more (choice) observations will be needed for the ANN to reliably represent the underlying DGP; but no concrete results are available as of now.[6]

This paper aims to fill the above mentioned knowledge gaps, and as such help pave the way for further and more effective deployment of ANNs for discrete choice analysis, by 1) testing whether the 'factor 10' rule-of-thumb which is used in most ANN-applications is appropriate

[5] Considering the fact that emerging data sets tend to be high dimensional, much effort has been devoted to optimising the data requirements by selecting the most relevant features (Blum & Langley, 1997; Ribeiro, Sung, Suryakumar, & Basnet, 2015). Note that deep neural networks (i.e., deep learning) methods are able to process raw data and automate the feature learning step (see Goodfellow, Bengio, and Courville (2016) for overview)

[6] Note that ANNs are capable of approximating any measurable function, given that sufficient processing neurons are available at the hidden layer and sufficient data is available for training (this property is known as Universal Approximation Theorem (Cybenko, 1989; Hornik, Stinchcombe, & White, 1989)).

in a discrete choice context (and if the answer is 'no', by proposing a new rule-of-thumb); and by 2) studying the relation between the complexity of the choice model's DGP and the size of the dataset that is required for meaningful, reliable discrete choice analysis using ANNs.

To achieve these two contributions to the literature, the remainder of this paper is organised as follows: Section 2 gives a brief theoretical overview of ANNs' sample size requirements, and reviews a selected number of recent applications of ANNs for discrete choice analysis. Section 3 presents a series of Monte Carlo experiments, designed to derive sample size requirements for ANN-based discrete choice analysis. Section 4 provides a cross-validation of obtained preliminary results, in the context of real empirical data. Finally, section 5 draws conclusions and presents potential directions for future research.

2 Sample size requirements for Artificial Neural Networks – Theoritical considerations

ANNs are a class of machine learning algorithms that are inspired by the biological neural system. They are well-known for being highly effective in solving complex classification and regression problems (Bishop, 1995). In the context of discrete choice modelling, various comparison studies between ANNs and choice models have been conducted. For example, Hensher and Ton (2000) found that the prediction performance of ANNs is similar to a nested logit model in the context of commuter mode choice. In contrast, Mohammadian and Miller (2002) concluded that ANNs predictive power outperforms the nested logit model in the context of household automobile choice. A similar conclusion was reported by Cantarella and de Luca (2005), who trained two ANNs with different architectures to model travel mode choices. This conclusion is also confirmed by a recent study by Hagenauer and Helbich (2017), who compared many machine learning tools (including ANNs) and Multinomial Logit (MNL) to model travel mode choice.

An ANN consists of an input layer of neurons, one or more hidden layers, and a final layer of output neurons. The analyst needs to decide upon several factors such as the number of hidden layers, number of neurons at each layers, and the activation functions (see Appendix for more details and a more elaborate introduction to ANNs). Different choices of these factors result in ANNs with different levels of complexity. For example, adding more neurons to a particular hidden layer increases the capacity of the network because it has more degrees of freedom (i.e., a higher number of parameters in the network). However, it is crucial for the analyst to choose the factors so that ANN complexity is in line with the complexity of the underlying data generating process (DGP) of the problem at hand.

2.1 ANN complexity adjustment

The objective of an ANN's training process is to produce a model that approximates the underlying data generating process (DGP) based on previous observations (so-called training data) (see Appendix for more information). A successful approximation of the underlying process implies that the trained network is generalisable, meaning that it maintains a consistent performance in the available data used for training and on future data generated by the same DGP. Importantly, an ANN may fail to deliver such performance consistency if the network is excessively complex compared to the underlying data generating process. In this case, ANN performs very well on the training data, but fails to maintain a similarly strong performance on different data generated by the same DGP, which are used for validation purposes (so-called validation data). This issue is known as overfitting. Another issue that may impact the extent to which a trained ANN's is generalisable is known as underfitting, which means that the ANN is too simple compared to the underlying DGP. As a result, it performs poorly on both training and validation data. In this case, the ANN cannot accurately capture the relation (embodied in the DGP) between input and observed choices. In sum, it is essential for the analyst to consider the relation between complexity and performance (in the ANN-community, this relation is usually framed as a bias-variance dilemma). The above-described concepts of under- and overfitting a learning machine are shown on Fig. 1.

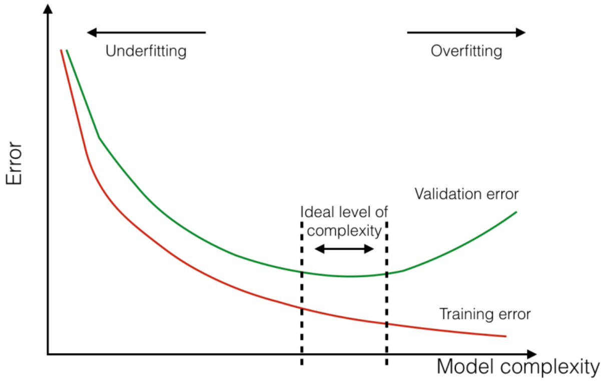


Figure 1. A conceptual representation of the relationship between model complexity and performance. Low model complexity (compared to the underlying DGP) is represented on the left hand side: here, models perform poorly on both training and future data, as they impose too simplistic assumptions on the DGP. In contrast, very complex models are represented on the right hand side. These models perform well on the available data, but fail to obtain a similarly strong performance on validation data generated by the same DGP. The ideal level of complexity is found in the range where the validation error is low, and divergence between training and validation error (thus the vertical distance between the red and green lines) is small.

In this study, the ANN complexity is adjusted by adding/removing hidden neurons. For example, if the underlying DGP is complex, an ANN with very few hidden neurons (in the extreme case: only one hidden neuron) will underfit this DGP. In contrast, using large number of hidden neurons will lead to overfitting. A common approach to test for under- and overfitting is to randomly separate the available data into three subsets: one each for training, validation and testing (Ripley, 2007; Shalev-Shwartz & Ben-David, 2014). Various ANNs with different levels of complexity (i.e., different number of hidden neurons) are estimated using the training set. Then, the performance of each of the estimated ANNs is evaluated on the validation set. The network that has the best performance with respect to the validation set is selected, as its complexity falls in the ideal level of complexity range shown in Fig. 1. Subsequently, to provide an unbiased evaluation of the selected network, ANN performance is further evaluated on the testing set. If the ANN also performs well on the testing data, the analyst can be confident that the network has successfully learned the underlying DGP. The error returned by the selected network on the testing set is an approximation of the so-called generalisation error, which is the key error for assessing an ANN's learning capability, because having an ANN with low generalisation error implies that the underlying data generating process has been well approximated (Abu-Mostafa et al., 2012).[7] A pseudocode of the above-described processes can be found below.

Pseudocode 1: ANN complexity adjustment and testing

```
Input:
    Training set, validation set, testing set, three-layers ANN
Step 1: Initialisation
    $M$ ANNs with different number of hidden neurons (different level of complexity)
Step 2: ANN performance evaluation
    For $m=1,2,\dots,M$
        Train ANN
        Measure the performance on validation set
    Choose the best performing ANN (as it has the optimum level of complexity)
    Measure the performance on testing set
    If satisfactory performance is obtained on testing test, ANN generalises
Output:
    ANN with optimum level of complexity
```

2.2 Theoritical measure of sample size requirements

Although this paper is intended to develop an *empirical* study of sample size requirements for ANN-powered discrete choice analysis, it is nonetheless useful to provide a brief background on theoretical contributions of the problem to the ANN-literature, and show the limited potential for practical application of these theories. As alluded to above, it is clear that the more complex the ANN, the more parameters the network consumes. And, the more parameters it consumes, the more data are needed for training the network. This intuitive relation has motivated scholars to estimate the appropriate training data size needed for reliable ANN (see papers cited in the introduction). To theoretically derive sample size requirements, a quantitative measure of ANN complexity is needed, which can be obtained from statistical learning theory. In particular, Vapnik and Chervonenkis (2015) provide a measure (known as the VC dimension) for the complexity of learning models such as ANNs. The quantification of model complexity using the VC dimension allows the statistical learning theory to provide quantitative predictions regarding the sample size requirements. The most significant outcomes in this regard are that the discrepancy between training and generalisation error is bounded from above by a quantity that grows as the model's VC dimension grows, and shrinks as the number of training examples increases (Goodfellow et al., 2016). However, despite that these outcomes provide a rigorous mathematical framework for studying data requirements, they have led to hardly any application in practice due to the prohibitive difficulty of meaningfully quantifying the VC dimension for complex learning models such as ANNs (Anthony & Bartlett, 2009; Blumer, Ehrenfeucht, Haussler, & Warmuth, 1989; Haussler, 1992a). Therefore, scientists and practitioners alike tend to follow rules-of-thumb when measuring the VC dimension for ANNs,

[7] In some cases, training the ANN and adjusting its complexity may not result in a low generalisation error, which means that the ANN has failed to approximate the underlying DGP to a sufficient extent. One possible reason of this outcome is that the used data are insufficient in size; i.e., when trained on a very small – relative to the number of nodes in the network – dataset, the ANN may end up memorising observations rather than learning the underlying DGP. In this case, it is recommended to use larger datasets. Another possible reason behind a low generalisation error is that the data quality may be poor, e.g., there may be many outliers in the data. A remedy for this is to implement pre-processing techniques in order to limit the randomness of the data.

which is then used for estimating the required sample size. The dominant rules can be summarised as follows: 1) the VC dimension of ANNs is approximately the same as the number of weights (Abu-Mostafa, 1995); 2) the sample size required to train the ANN is roughly 10 times the VC dimension (Baum & Haussler, 1989; Haykin, 2009). In sum, the size of the data that is required for meaningful and reliable ANNs is approximately 10 times the number of weights in the network (Abu-Mostafa, 1995; Baum & Haussler, 1989; Haykin, 2009).

Before we move on to the core of our paper, being the derivation and testing of rules-of-thumb for sample sizes in the context of ANN-based discrete choice analysis, we would like to note the following: ever since the introduction of ANNs, but especially in recent years (e.g., Castelvecchi, 2016), there has been debate about the 'black-box'-nature of ANNs. Indeed, compared to conventional choice models whose estimation results can be directly and meaningfully interpreted in terms of attribute-weights, elasticities and the like, the interpretability of a trained ANN's weights is very limited. Although progress is being made in this regard (see van Cranenburgh and Alwosheel (2019) for an example in a choice modelling context), it remains the case that the use of trained ANNs is currently mostly limited to forecasting, with less to offer in terms of learning about behavioural processes. We consider attempts to 'open the black box' of ANNs and to deploy them for behavioural analyses, as very important directions for further research. However, in the present paper we do not focus on this aspect, nor do we wish to make claims about the (dis-)advantages of ANNs compared to conventional choice models. Our work in this paper is motivated by the increasing use of ANNs for discrete choice analysis, which in our view makes it important to know what sample size requirements apply in this context.

3 Sample size requirements – Monte Carlo experiments

In this section, we aim to put the 'factor 10' rule-of-thumb for sample size requirements to the test in a discrete choice analysis context, and to acquire insights into the relation between the complexity of the DGP (i.e., the choice model) and the model's sample size requirements. To do this, we conduct a series of Monte Carlo experiments, in which the true DGP varies in degrees of complexity which are observable and manageable by the analyst. Furthermore, besides studying the complexity of the DGP we also investigate the effect of random noise in the DGP (which is reflected in variations in parameter sizes, causing variation in rho-square) on sample size requirements.

3.1 Data

Table 1 presents an overview of the (synthetic) DGPs used in this section, including their parameterisations. All data sets consist of three alternatives with two generic attributes: X_1 and X_2. Each data set consists of 1,000 hypothetical respondents. Each decision-maker is confronted with T = 10 choice tasks. Attribute levels are generated using a random number generator drawing values between zero and one. To create the synthetic observations for the Random Utility Maximisation (RUM) Multinomial Logit (MNL) DGPs, the total utility of each alternative is computed and the highest utility alternative is assumed to chosen. Similarly, for the Random Regret Minimisation (RRM) DGPs, the total regret is computed for each alternative and the minimum regret alternative is assumed to be chosen; note that we use the Pure RRM (P-RRM) model introduced in van Cranenburgh, Guevara, and Chorus (2015), which provides the strongest possible level of regret aversion which can be attained in an RRM framework. For the Panel Mixed Logit (ML) DGPs, each respondent is assigned one draw from the associated normal distribution for each β.

Table 1. Data Generating Processes and their specifications

Data no.	DGP	Model specification	Parameterisation
A1	RUM-MNL	$V_{in} = \sum_m \beta_m x_{imn}$	β_1 = -4.3 β_2 = -6.45 ρ^2 = 0.50
A2	RUM-MNL	$V_{in} = \sum_m \beta_m x_{imn}$	β_1 = -2.85 β_2 = -4.28 ρ^2 = 0.35
A3	RUM-MNL	$V_{in} = \sum_m \beta_m x_{imn}$	β_1 = -1.84 β_2 = -2.75 ρ^2 = 0.20
B1	RUM-ML	$V_{in} = \sum_m \beta_m x_{imn}$	β_1 ~N(-4.44, 1) β_2 ~N(-6.66, 1) ρ^2 = 0.50
B2	RUM-ML	$V_{in} = \sum_m \beta_m x_{imn}$	β_1 ~N(-3.07, 1) β_2 ~N(-4.61, 1) ρ^2 = 0.35
B3	RUM-ML	$V_{in} = \sum_m \beta_m x_{imn}$	β_1 ~N(-2.02, 1) β_2 ~N(-3.02, 1) ρ^2 = 0.20
C1	P-RRM-MNL	$R_{in} = \sum_m \beta_m \tilde{x}_{imn}$ $where\ \tilde{x}_{imn} = \sum_{j \neq i} \max(0, x_{jmn} - x_{imn})$	β_1 = -2.88 β_2 = -4.32 ρ^2 = 0.50
C2	P-RRM-MNL	$R_{in} = \sum_m \beta_m \tilde{x}_{imn}$ $where\ \tilde{x}_{imn} = \sum_{j \neq i} \max(0, x_{jmn} - x_{imn})$	β_1 = -1.83 β_2 = -2.74 ρ^2 = 0.35
C3	P-RRM-MNL	$R_{in} = \sum_m \beta_m \tilde{x}_{imn}$ $where\ \tilde{x}_{imn} = \sum_{j \neq i} \max(0, x_{jmn} - x_{imn})$	β_1 = -1.13 β_2 = -1.69 ρ^2 = 0.20

3.2 ANN complexity adjustment process

In this sub-section, we present an example of how ANN complexity is adjusted in practice, following the ANN training procedure as explained in the Appendix. To avoid repetition, we only present the case for dataset A1; the same procedure applies for the other cases as well. Initially, data are randomly divided into three parts: 70% for training, 15% for validation and 15% for testing.[8] Several ANNs with different levels of complexity are subsequently created.

[8] Note that it is possible to end-up with suboptimal ANN performance due to drawing a biased or skewed subsets. One proposed remedy for such issue is to use the so-called k-fold cross validation method. In our data, we did not find different results when using the k-fold cross validation method for ANN complexity adjustment purposes.

These ANNs are then trained on the training data. Fig. 2 shows the relationship between ANN complexity (i.e., the number of hidden neurons) and the Log-Likelihoods (averaged across observations) obtained on both the training and validation set. The network that provides the best performance on the validation data is then selected. Fig. 2 shows that four hidden neurons provide the best performance (on the validation set). Using more than four hidden neurons does not affect the resulting Log-Likelihood, implying that ANN has learned the input/output relationship with four neurons.[9]

When complexity of the underlying DGP is increased, ANNs with more hidden neurons are needed. For example, our analysis shows that the optimum number of hidden neurons for the more non-linear and 'complex' (as it involves a series of max-operations and pairwise comparisons in the regret function) RRM-MNL data is eight, constituting a doubling compared to a linear-in-parameters RUM model (see Table 2 for results for all DGPs).[10] Once the network that provides the best performance is obtained, the number of weights in the network can be observed accordingly.

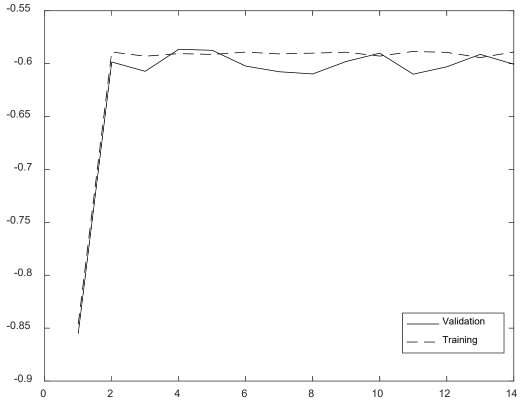


Figure 2. Number of ANN hidden neurons vs average Log-Likelihood values for RUM-MNL data.

3.3 Resulting ANN sample size requirements

To assess, in the context of the testing data, whether the ANN has been trained on a number of choice observations that is large enough to enable a sufficiently accurately learning of the underlying DGP, several approaches have been introduced in different contexts and applications (Cho et al., 2015; Figueroa, Zeng-Treitler, Kandula, & Ngo, 2012; Mukherjee et al., 2003; Sung, Ribeiro, & Liu, 2016). In our study we determine the sample size required for

[9] According to Occam's razor principle, an explanation of a set of data should be limited to the bare minimum that is consistent with the data. In Fig 2, increasing the complexity does not result in better performance. Therefore, the simplest model that describe data is preferred, which in this case is an ANN with four neurons.

[10] Note that different ANN structures (i.e., different number of hidden layers, different activation functions) have been also implemented for this study. We found that adding more hidden layers did not improve prediction performance. Also, we found that using different activation functions for shallow ANN did not result in a different prediction performance. As such, due to space limitations and for the ease of communication, we choose to focus in this study on the single hidden-layer ANN.

accurately learning of the underlying DGP based on the learning curve. More specifically, we inspect the gradient of the learning curve – which represents the size in improvement of the ANN's prediction performance as more training data sets are used. The intuition behind an ANN's learning curve is straightforward: as more observations are used to train the network, a better prediction performance is obtained until the learning curve reaches a saturation point where its learning rate slows and its gradient starts to approach zero, implying that the size of the training dataset has been sufficient for the ANN to learn the DGP (Cortes, Jackel, Solla, Vapnik, & Denker, 1994; Kohavi, 1995). We consider the ANN to have successfully learned the underlying DGP if the gradient of the learning curve is less than 10^{-5}.

Furthermore, given that –in this subsection– we deal with synthetic data, we can also inspect the deviation between the prediction performance of the ANN and the best possible prediction performance (note that no model is capable to outperform the true DGP). Hence, in the context of synthetic data there is a theoretical and observable upper limit of the prediction performance, which is embedded in the true DGP. So, as a cross-check of the learning curve gradient criterion mentioned above, we inspect the difference between the ANN prediction performance and the theoretical upper limit.

Fig. 4 presents the ANN learning curves for the data sets described in Table 1. For each data set, it shows the impact of training data size (on the *x*-axis) on the ANN prediction performance (using metrics presented in Appendix 2A) on testing data (*y*-axis). We present results for both the Log-Likelihood-based measure (top-panels) and the Hit-Rate-measure (bottom-panels); note that while the Hit-Rate measure is popular in the ANN-community, but is only occasionally used in the field of choice modelling, in generally not being recommended).[11] For each data set, we fitted a power function of the form $y = ax^b + c$. Based on this fitted function the gradient is determined.

Note that each data point represents the performance of ANNs trained using k-folds cross validation method, to avoid presenting the result of a particular manifestation of the randomness in the data generating process (Abu-Mostafa et al., 2012). The notion of k-folds cross validation methodology is to partition the data into k equal sized subsamples. A single subsample is then used for testing and the remaining $(k-1)$ are used for training. This process is repeated k times, where each of the k subsamples used only once for testing. The resulted ANN performances are averaged and reported. Also, note that to reflect the difference in levels of noise represented in the underlying DGPs, the ANN performance is normalised with respect to the associated theoretical upper limit. For the Log-Likelihood (LL) measure, ANN prediction performance is normalised as follows:

$$1 - \frac{LL_{ANN}}{LL_{max}} \qquad (1)$$

And for the Hit-Rate (classification accuracy) measure, the following normalisation applies:

$$\frac{HitRate_{ANN}}{HitRate_{max}} \qquad (2)$$

[11] Particularly in a Marketing context, Hit-Rates are often used to assess a choice model's empirical performance (e.g., Huber & Train, 2001; Kalwani, Meyer, & Morrison, 1994; Neelamegham & Jain, 1999). However, its use has been criticized for failing to accurately represent the probabilistic nature of choice models (e.g., Train, 2009). In this paper, we do not wish to express a strong opinion on this matter, but we do note that the mainstream in choice modelling attaches far more importance to likelihood-based measures of model performance than 'correct classification'-based metrics.

Note that two vertical lines represent the data requirements according to: 1) the factor 10 requirement that is the widely adopted rule-of-thumb in the ANN-community (i.e., the data required for ANN training is 10 times the number of weights in the network); 2) the sample size requirement according to the criterion of successful learning mentioned above. For all cases, the difference between the theoretical upper limit and the ANN prediction performance (that has been trained on data of the proposed size) is less than 10%, indicating the strong prediction performance achieved by the ANN. To facilitate inspection of the figures, we only draw this second vertical line for the least noisy DGP within a particular category of DGPs. Results are summarised in Table 2.

Finally, to put our findings in yet more perspective, we also compare the results obtained using the learning curve approach with a recently proposed methodology for big data applications (not focusing on discrete choice analysis-contexts) known as the Critical Sampling Size (CSS) heuristic (Ribeiro et al., 2015; Sung et al., 2016). The CSS heuristic method aims to find the absolute minimal number of observations required to ensure that a learning machine meets a desirable performance (Silva, Ribeiro, & Sung, 2017). The first step of the CSS heuristic method is to partition the data into k clusters. Then, m randomly sampled data-points are selected from each cluster (m is initially set to be fairly small) to form a training data set of size mk. If the performance of the trained ANN (on a separate testing dataset) exceeds a pre-defined threshold value T, then the training data size is considered sufficient for the ANN. Otherwise, the process of sampling is repeated with larger value of m, until a satisfactory performance is achieved. For a more extensive description of this method see Silva et al. (2017). In the context of this study, we set T to be 2% less than the ANN prediction performance (in terms of Log-Likelihood measure) when it has access to the whole dataset.

The CSS heuristic method is executed 50 times. Fig. 3 shows a histogram of the frequency of sample size requirements across 50 runs for dataset A1, which follows a seemingly normal distribution. We can notice that once the ANN has access to a sample size of 2,000, more than two thirds of the 50 runs obtained a performance that exceed the defined threshold T. Further, around half of the 50 runs provide a satisfactory performance with a training data of size 2,000 (see Fig. 3). In this case, we report that the ANN data requirements is 2,000. Results for all datasets are shown in Table 2.

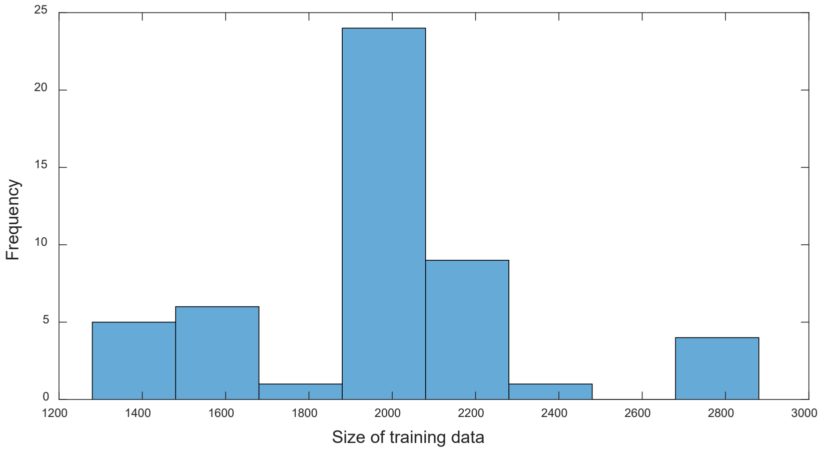


Figure 3. Frequency of sample size requirements using heuristic CSS method.

Table 2. ANN data requirement for synthetic data.

DGP	Rho-square	Hidden nodes	Number of ANN parameters	Data requirement based on 'factor 10' rule of thumb	Data requirement based on the learning curve gradient method	Factor implied by the learning curve gradient method	Factor implied by the CSS heuristic method
(A1) RUM-MNL	0.50	4	43	430	2200	54	47
(A2) RUM-MNL	0.35	4	43	430	2000	47	42
(A3) RUM-MNL	0.20	4	43	430	2000	47	38
(B1) RUM-ML	0.50	5	53	530	2600	50	46
(B2) RUM-ML	0.35	5	53	530	2200	42	42
(B3) RUM-ML	0.20	5	53	530	1800	34	34
(C1) P-RRM-MNL	0.50	8	83	830	3000	37	37
(C2) P-RRM-MNL	0.35	8	83	830	2400	29	32
(C3) P-RRM-MNL	0.20	8	83	830	1800	22	27

3.4 Interpretation of results, and discussion

Based on these results, we are able to establish a number of important observations: first, looking at the ANN learning curves, it is directly seen that for all decision rules the training data size requirement imposed by the 'factor 10' rule of thumb is not conservative enough, especially when considering the Log-Likelihood-based measure of evaluation (which is used considerably more often in the choice modelling field than the Hit-Rate). Clearly, ANN performance significantly enhances as the network has access to larger training dataset, i.e., beyond the size which is advised by the 'factor 10' rule-of-thumb. Table 2 shows the factor (i.e., the ratio between required number of training observations and the number of weights in the network) which is implied when one considers the proposed requirements; it varies, across DGPs, between 22 and 54. Furthermore, the factors obtained using the CSS heuristic methodology are within the same range (see Table 2, last column). Therefore, to be on the safe side, these results – based on synthetic data – suggest the following rule of thumb when using ANNs to analyse discrete choice data and when considering Log-Likelihood-based measures as the appropriate standard for model evaluation: the number of observations in a training dataset needs to be at least 50 times larger than the number of weights in the network to enable a sufficient performance.

Another (and at first sight possibly counterintuitive) point worth noting concerns the effect of the level of noise in the DGP on ANN sample size requirements. Our analysis shows that the ANN requires more training observations, as the DGP becomes less noisy. One likely interpretation of this finding is that as the level of noise in the DGP decreases, the data contains more information worth learning by the ANN. Hence, the network requires more effort (i.e., more data for training) to extract the information.

In sum, from the Monte Carlo experiments we learn that the complexity and the 'noisiness' of the underlying DGP both have an impact on the minimum number of observations required to train an ANN. A 'factor 50' rule of thumb seems to be appropriate and the commonly used 'factor 10' rule of thumb seems too 'optimistic' (especially when using a Log-Likelihood-based metric for model evaluation as is the standard in the choice modelling community). Finally, these results on synthetic data suggest that ANN data requirements are by and large within the range of common dataset sizes used in choice modelling.

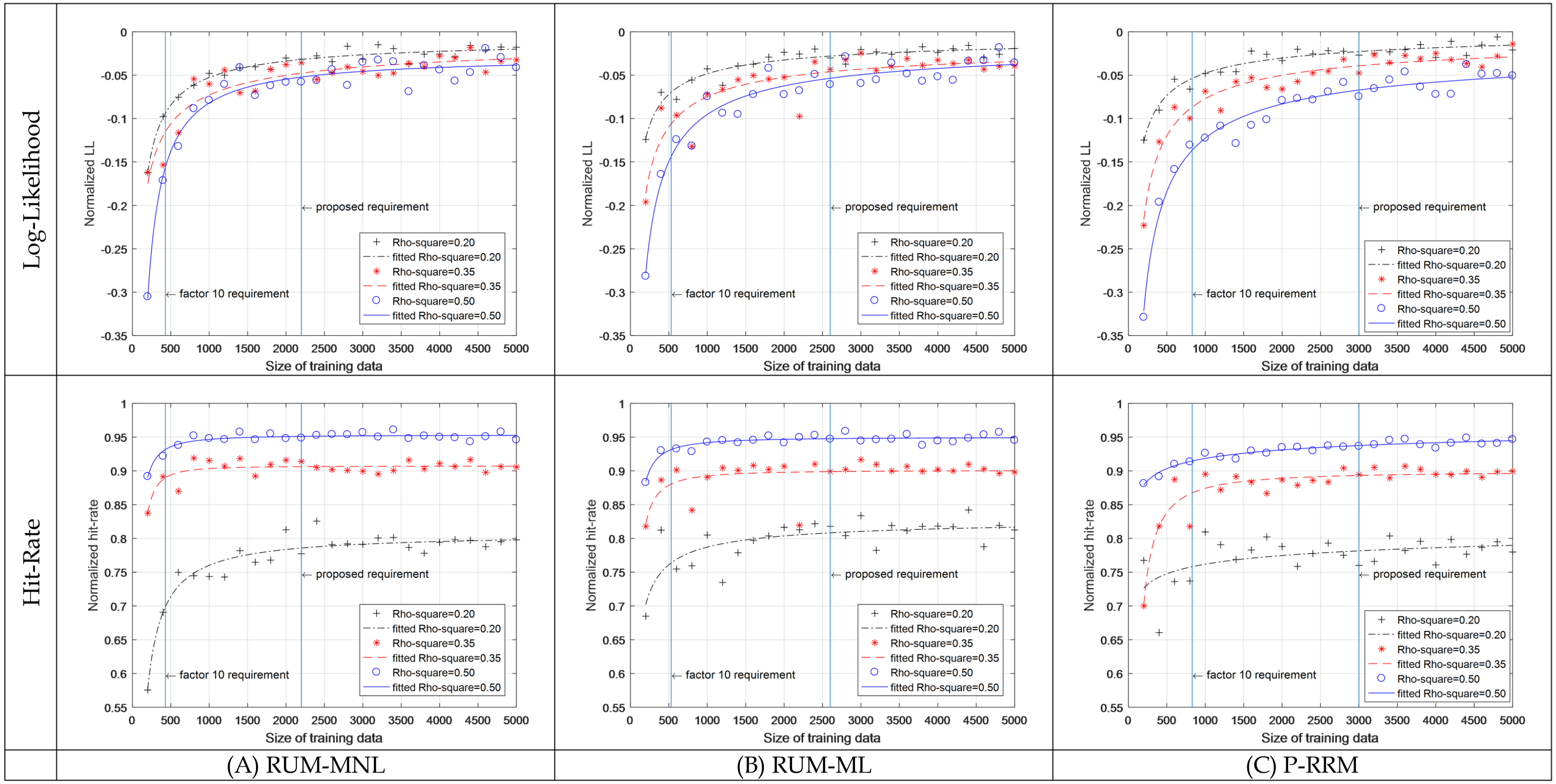


Figure 4. ANN prediction performance on synthetic datasets.

4 Sample size requirements – real data

In this section, we aim to extend our analysis of ANNs data requirements for choice modelling, beyond synthetic data towards several real data sets that have been extensively reported in the choice modelling literature. A brief description of the used data sets can be found in Table 3. To assess whether the ANN has been trained on a sufficient amount of data, the criterion reported in sub-section 3.3 is used. That is, we consider an ANN to have sufficiently accurately learned the underlying DGP once the gradient of the learning curve is less than 10^{-5}. Note that, unlike the Monte Carlo experiments, the true DGP is obviously unknown for these datasets, and consequently the theoretical upper limit prediction performance – which we used to cross-check the derived sample size in the previous section – cannot be determined in this context.

Table 3. Description of real data sets.

	Reference	Number of choice observations	Number of alternatives in the choice set	Number of attributes per alternative
Data set 1	(Bierlaire, Axhausen, & Abay, 2001)	9036	3	2
Data set 2	(Chorus & Bierlaire, 2013)	3510	3	4
Data set 3	(Hague Consulting Group, 1998)	17787	2	2

Fig. 5 shows the ANN learning curves for each of the data sets described in Table 3. As in previous plots, for each data set, the impact of training data size (depicted on the *x*-axis) on two aspects of the ANNs prediction performance is shown: average Log-Likelihood and classification accuracy (Hit-Rate). Note that two vertical lines represent the data requirements according to: 1) the 'factor 10' rule-of-thumb commonly used in the ANN literature, and 2) the data requirements based on the proposed learning curve gradient criterion. Note also that smaller subsets of the full dataset were obtained by randomly removing observations from the mother-dataset. Finally, the sample size requirements obtained using learning curve gradient are compared with those obtained using the CSS heuristic method. A summary of results is shown in Table 4.

Table 4. ANN data requirements: real data.

Data set	Hidden nodes	Number of ANN parameters	Data requirement based on 'factor 10' rule of thumb	Data requirement based on the learning curve gradient method	Factor implied by the learning curve gradient method	Factor implied by the CSS heuristic method
Data set 1	10	133	1330	3400	31	28
Data set 2	5	93	930	2600	28	25
Data set 3	8	106	1060	2800	27	36

The results confirm the insufficiency of the data requirements based on the 'factor 10' rule of thumb: for all data sets, Fig. 5 shows a clear pattern of attaining better predictive performance when the network is trained on larger data sets. Based on the learning curve gradient condition, dataset sizes implying a factor of 27 to 31 times the number of weights in the network appear to be sufficient. Further, the factors obtained using the CSS heuristic method are within the same range (see Table 4, last two columns).

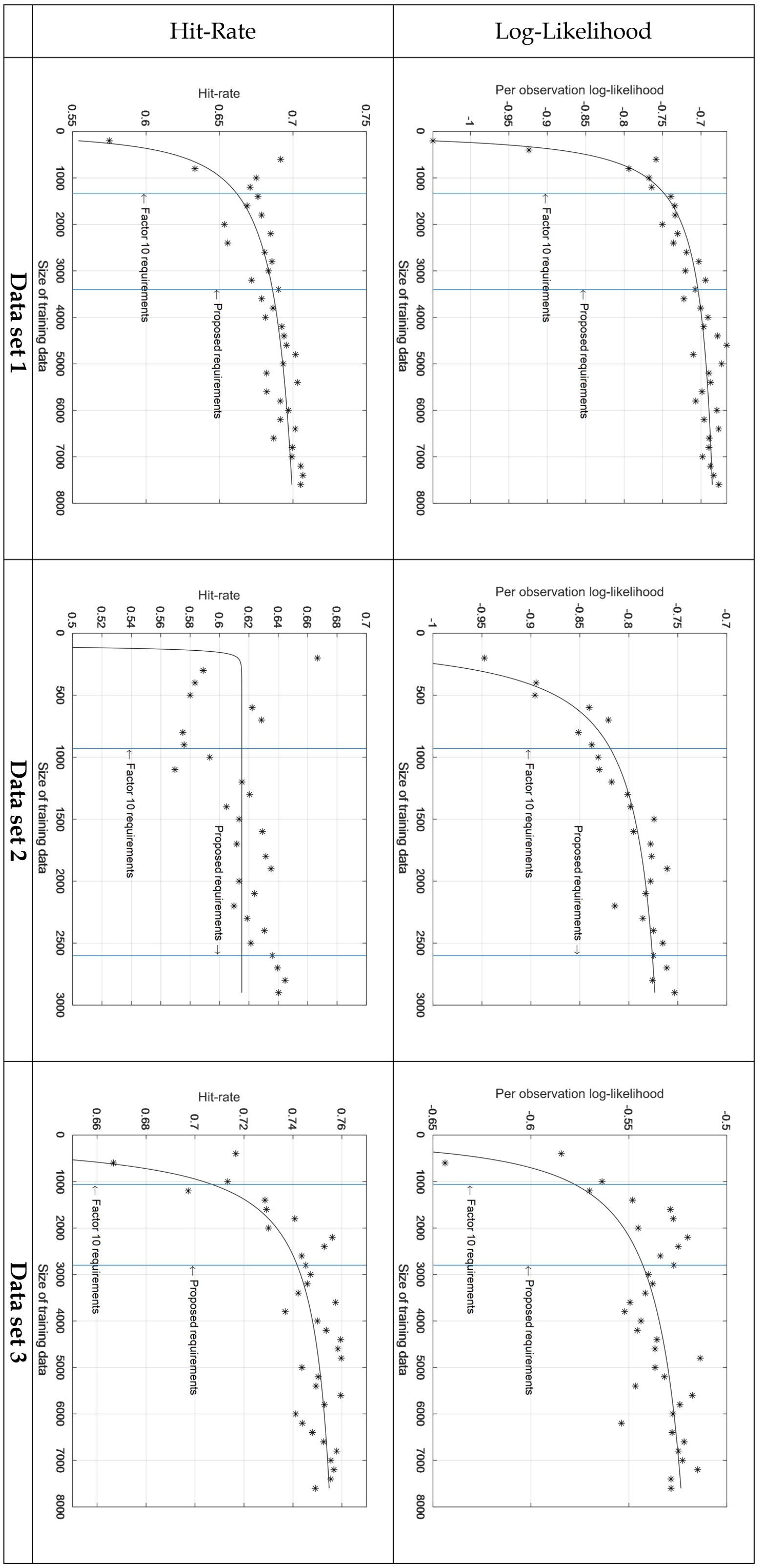


Figure 5. ANN prediction performance for real data

5 Conclusions and recommendations

This study contributes to the rapidly growing literature which focuses on using artificial intelligence (machine learning) techniques for discrete choice analysis, by investigating the size of datasets which is required for reliable representation of discrete choice models using Artificial Neural Networks (ANNs). In particular, using synthetic datasets, we study the sample size that is required for Data Generating Processes with different levels of complexity and 'noisiness'. In addition, we analyse dataset size requirements for ANN-based discrete choice analysis, based on several real data sets that have been used in the literature. For each data set, the complexity of the ANNs (which ultimately determines the required sample size) is optimised using validation methods commonly used in the artificial intelligence (machine learning) community. Using the concept of a learning curve, we are able to establish the number of observations that an ANN needs to obtain a reliable and strong predictive performance on out-of-sample data. Based on our analyses, we are able to draw the following conclusions and recommendations concerning data requirements for ANN-based discrete choice analysis.

First: data requirements based on the 'factor 10' rule-of-thumb which is widely-adopted in the ANN literature appear to be insufficient if one wants to evaluate model performance in terms of Log-Likelihood-based measures (as is the norm in most of the choice modelling community). Based on inspecting results for synthetic and real data sets, and to be conservative, we propose to use a 'factor 50' rule of thumb (i.e., the number of observations needs to be at least 50 times the number of adjustable parameters in the network; where it should be noted that the number of adjustable parameters in an ANN is generally much higher than the number of parameters in a corresponding choice model). If one aims to evaluate model performance on terms of Hit-Rate – or: correctly classified – based metrics, smaller data sets may be used (which may explain the popularity of this rule-of-thumb in the machine learning literature which generally uses Hit-Rates for model evaluation). But also in that case, our analyses suggest that the 'factor 10' rule-of-thumb appears somewhat too 'optimistic'. Second, as an important side result we find that the ANN requires more data as the complexity of the DGP increases and its noisiness decreases. Third, our analysis shows that ANN sample size requirements are roughly within the range of most data set sizes encountered in the field of choice modelling. This finding suggests that indeed there is ample opportunity for using ANNs to analyse discrete choice data, also on existing data sets but particularly so on emerging 'Big-'datasets. Note that these conclusions are derived from shallow ANNs trained using back-propagation approach. We acknowledge that there are various types of ANNs models (i.e., different network structure, activation functions, etc.) that we haven't examined in this study. However, this provides an avenue for further research in the near future.

As a final note, we wish to re-emphasise that the required sample size for ANN-based (discrete choice) analysis depends on the complexity (i.e., number of neurons) of the ANN. Since the complexity of the ANN cannot be determined in advance – see section 2 for a description of the iterative procedure used to determine the optimal number of neurons – this implies that sample size requirements can only be determined after 'estimation'. Three approaches are suggested in this regard: first, the analyst may indeed determine ex post if the sample used for training the ANN has in fact been large enough. Second, the analyst may use a prior study to determine the optimal number of neurons in the ANN, and based on that choose the sample size for the core study.[12] Third, the analyst may build on past work reported in the literature to ex

[12] For highly complex problems (e.g., image processing problems), deep networks are commonly used. Due to the large number of weights and the computing power required for training them, it is commonly practiced to use pre-trained networks, where the structure (i.e., number of hidden layers, and number

ante guess the likely number of neurons needed in the ANN, and work from there. Note that this approach is quite similar to common practice in classical choice modelling, where minimum sample sizes needed to obtain significant parameters can only be determined ex post, or based on prior parameters which can be based on literature or on pilot studies.[13]

Appendix 2A. Choice tasks in the value-of-time choice experiment

ANNs consist of highly interconnected processing elements, called neurons, which communicate together to perform a learning task, such as classification, based on a set of observations. Figure 2A.1 shows the layout of the neuron structure.

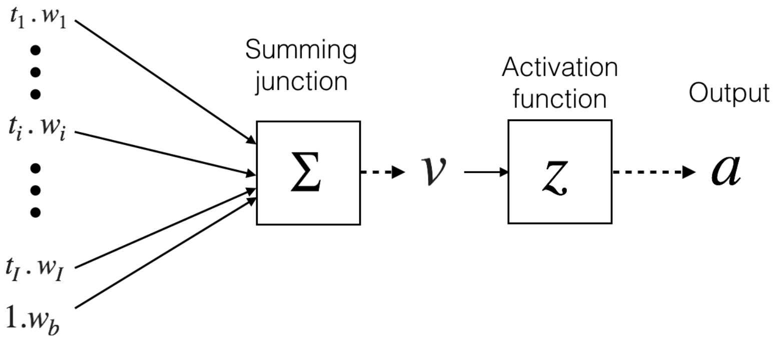


Figure 2A.1. A neuron layout.

Each neuron in the network receives inputs (t_i) multiplied by estimable parameters known as weights (w_i). The weighted inputs are accumulated and added to a constant (called bias, denoted b) to form a single input v for a pre-defined processing function known as activation function $z(.)$. The bias has the effect of increasing or decreasing the net input of the activation function by a constant value, which increases the ANNs flexibility (Haykin, 2009). The activation function $z(.)$ generates one output a that is fanned out to other neurons. The output a can be described as follows:

$a = z(v) = z(\sum_{i=1}^{I} w_i * t_i + w_b)$, where w_b is the weight associated with the bias.

The neurons are connected together to form a network (Bishop, 2006; LeCun, Bengio, & Hinton, 2015). A widely used ANN structure consists of layers of neurons connected

of neurons in each layers) and the weights' values are used (see for example Vedaldi & Lenc, 2015). The network is then trained on the newly presented data.

[13] Sample size requirements have been investigated for Stated Preference (SP) and Revealed Preference (RP) data. Just like machine learning practitioners, SP practitioners have developed several rules-of-thumb. For example, McFadden (1984) proposed that a sample size of thirty responses per alternative. Another widely used rule, which is a mirror-image of the developed rule in this study, is to have at least 30 times the number of adjustable parameters (see Rose and Bliemer (2013) for overview). For RP data, Hensher, Rose, and Greene (2005) proposed to have a minimum sample sizes of 50 decision maker choosing each alternative.

successively, known as multi-layer perceptron (MLP) structure. Typically, the first (input) layer and the output layer depend on the problem at hand. More specifically, input layer neurons represent the independent variables. In the context of choice modelling, these are the alternatives' attributes, characteristics of decision-makers, and contextual factors. The output layer, in a discrete choice context, consists of neurons that provide choice probabilities P for each alternative. Layers in-between are called hidden layers because their inputs and outputs are connected to other neurons and are therefore 'invisible' to the analyst. For illustrative purposes, consider the following hypothetical situation: a person can travel using one of three modes: bus, train, or car; two attributes (travel cost "TC" and travel time "TT") are associated with each alternative. Figure 2A.2 shows this typical choice situation in a three-layer MLP network with four hidden neurons.

Neurons at the hidden and output layers are represented by circles in Figure 2A.2, while input and bias neurons are represented by squares. This is to emphasise that the neurons at the hidden and output layers are processing units, meaning that they receive inputs t and return outputs a according to predefined activation function $z(.)$, as illustrated in Figure 2A.1. Input neurons pass the input signals to the next layer. In Figure 2A.2, the ANN has a total of 7 processing units.

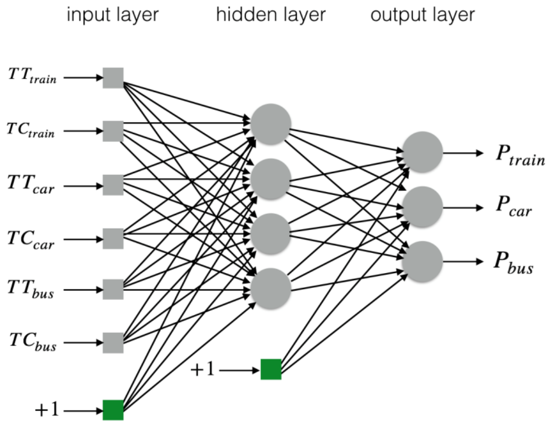


Figure 2A.2. Three-layers Artificial Neural Network.

2A.1. ANN specifications

For a complete MLP structure, three elements need to be defined:

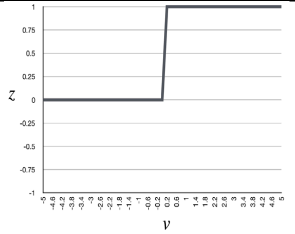
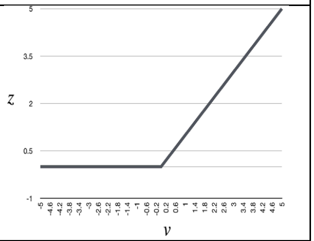
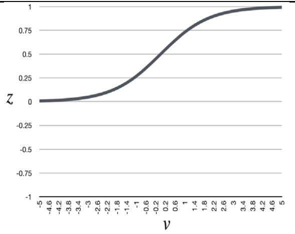
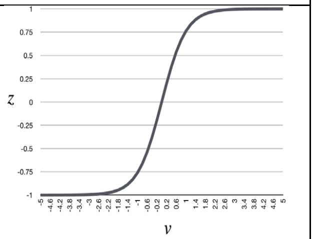
1) Number of hidden layers: a commonly used structure is three-layers MLP: input, output and one hidden layer. A key property of this structure lies in the ability to approximate, with arbitrary level of precision, any measurable function given that a sufficient number of processing neurons are available at the hidden layer; this property is known as the Universal Approximation Theorem (UAT) (Cybenko, 1989; Hornik et al., 1989). The three-layer MLP structure is considered and discussed in more detail further on.

2) Number of neurons for the hidden layer(s): the UAT holds true only if a sufficient number of hidden neurons are available. Intuitively, ANNs with more hidden neurons

have more free parameters (w) and are therefore capable of learning more complex functions.

3) Activation function $z(.)$: As mentioned before, each neuron processes its input via a pre-defined activation function. Neurons at the same layer usually employ identical functions. Examples of commonly used functions in the hidden layers are presented in Table. A1. In the analyses presented in the remainder of this paper, a tangent sigmoidal function has been employed at the hidden layer neurons, as it has been shown to lead to fast training times (LeCun, Bottou, Orr, & Müller, 2012). For the output layer, a so-called softmax function is used (which is essentially a logit) to ensure that the sum of the choice probabilities equals one.

Table 2A.1. Activation functions

	Activation Function Name	**Function**	**Plot**
a	Step Function	$z = \begin{cases} 0 & v \leq 0 \\ 1 & v > 0 \end{cases}$	
b	Rectifier Linear Unit (ReLU) Function	$z = max(0, v)$	
c	Sigmoid Function	$z = \dfrac{1}{1 + \exp(-v)}$	
d	Tangent Sigmoidal Function	$z = \tanh(v)$	

An analyst sets these three elements according to the desired objective of the modelling effort. For example, adding two or more hidden layers serves to create a deep learning network, which has been shown to lead to breakthrough results in fields such as image classification (e.g., Krizhevsky, Sutskever, & Hinton, 2012). In this paper, for reasons of ease of communication and without loss of generic applicability, we limit our focus to the so-called shallow network version of the ANN (i.e., an ANN with single hidden layer). The complexity of such network is adjusted by adding or removing neurons at the hidden layer (called hidden neuron). It is crucial for learning to adjust the number of hidden neurons so that ANN complexity matches the problem at hand (i.e., the underlying DGP). An example that shows how to adjust the number of hidden neurons is presented in subsection 2.1 of Chapter 2.

2A.2. ANN Training

In the discrete choice modelling context, the process of finding values of the model's parameters (w) is known as estimation. In this study, we comply with the language of machine learning community and call it training. The choice data used for training the ANN consists of a set of observations $S = ((\mathbf{x}_1, \boldsymbol{y}_1), (\mathbf{x}_2, \boldsymbol{y}_2), \dots, (\mathbf{x}_n, \boldsymbol{y}_n), \dots, (\mathbf{x}_N, \boldsymbol{y}_N))$. Each n^{th} observation s_n contains a vector of independent variables $\mathbf{x}_n$ that represent the attributes and a *K*-dimensional vector of dependent variables $\boldsymbol{y}_n$ that represent the observed choice (i.e., zeros for the non-chosen alternatives, and a one for the chosen alternative); *K* being the size of the choice set. Since choices are mutually exclusive (i.e., only one alternative can be chosen from the choice set), from a machine learning perspective this is considered a classification problem.

The central goal of ANN training is to model the underlying data generating process (DGP) that has led to the current set of observations, so that the best possible prediction for future observations is achieved (Bishop, 1995). While to estimate the parameter of a choice model the likelihood function is maximised, for ANN training an equivalent so-called error function $J(\mathbf{w})$ is minimised. We define $\mathbf{w}$ as a vector that contains the ANN estimable parameters w. Assuming the data consist of N choice observations across K alternatives, the error function is defined as follows:

$$J(\mathbf{w}) = -\sum_{n=1}^{N}\sum_{k=1}^{K} y_{nk} \ln(P_{nk}) \qquad (2A.1)$$

Where y_{nk} is an indicator which denotes whether alternative k is chosen in observation n, and P_{nk} is the choice probability predicted by the ANN, which is a function of $\mathbf{w}$ and $\mathbf{x}$. To avoid unnecessary semantic confusion, the function in Equation $(2A.1)$ is called negative Log-Likelihood function in the rest of this document.

By training the ANN, the analyst's objective is to find the weight vector $\mathbf{w}$ such that $J(\mathbf{w})$ is minimised, by means of searching through the parameters' space in successive steps. At each step, $J(\mathbf{w})$ is decreased by adjusting the parameters in $\mathbf{w}$. The well-known gradient descent approach is the most widely applied algorithm for this purpose. In short, this process of training an ANN can be described as follows: first, the weights' values w are randomly initialised. The input neurons' values (taken from the training data) are propagated to the output layer through the hidden layer, this process is called forward propagation. Then, the output neurons' values (i.e., choice probabilities) are compared with the observed choices to compute the function $J(\mathbf{w})$ described in Equation $(2A.1)$. The optimisation mechanism is then conducted by propagating J backward to the input layers through the hidden layer. To adjust the weights, the backward propagation process includes taking the partial derivative of the error J with respect to the weights, called the gradient vector $\mathbf{g}$. Along with a learning rate value η, $\mathbf{w}$ values are re-adjusted as follows:

$$\mathbf{w}_{p+1} = \mathbf{w}_p + \eta_p \mathbf{g}_p \qquad (2A.2)$$

Where p represents a step index. The learning rate η determines how fast the learning algorithm is moving toward the optimum $\mathbf{w}$. If η is very large, there is a relatively high possibility to never obtain the optimum $\mathbf{w}$ due to overshooting. In contrast, using a very small η increases the learning time substantially. One commonly used way to overcome this problem is to use adaptive learning rates, iteratively determined during training.

The process of error (forward and backward) propagation is repeated iteratively until a pre-specified stopping criterion is achieved. This training mechanism is known as back-propagation, and constitutes the most popular approach to train neural networks (Rumelhart, Hinton, & Williams, 1988). However, it should be noted that moving toward a local minimum is one of the widely reported risks associated with this back-propagation approach (Iyer & Rhinehart, 1999; Park, Murray, & Chen, 1996). As such, it is always recommended to train the network more than once to minimise the probability of ending up with a sub-optimal trained network. A pseudocode of the ANN training can be found below, and for comprehensive description of ANNs training interested readers are referred to Bishop (2006).

Pseudocode 2A.1: ANN training

```
Step 1: Initialisation
        Set $\mathbf{w}$ values to random numbers
Step 2: Forward propagation
        Propagate the input neuron values $\mathbf{x}$ to output neuron through hidden neurons
        Calculate the ANN output neuron values (ANN probabilities)
        Calculate the error function (Equation $(A1)$)
Step 3: Backward propagation
        Calculate the gradient $\mathbf{g}$ for the network neurons
        Update $\mathbf{w}$ values
        Increase iteration $p$ by one
        Go back to step 2 and repeat the process until the selected error criterion is satisfied
Step 4: Repeat (recommended)
        Go back to step 1, repeat the whole process to minimise the probability of ending up
with a sub-optimal ANN
```

2A.3. Performance metrics for classification

In this section, we define the metrics that are used to evaluate the performance of a trained ANN. The first metric is equivalent to the negative Log-Likelihood measure, presented earlier in Equation $(2A.1)$. More specifically, we modify it slightly to obtain an average (across observations) Log-Likelihood measure (see Table 2A.2). Another metric which is commonly used in the ANN-literature is the classification accuracy measure. This so-called Hit-Rate is computed as follows: the ANN assigns probabilities P to each output neuron (see Figure 2A.2). The classifier output (denoted by $\hat{y}$) is set to one for the alternative which has the highest probability and zero for all others. In case of two choice situation, the classifier thus assigns 1 (i.e., $\widehat{y_1} = 1$) for the first alternative and zero (i.e., $\widehat{y_2} = 0$) for the second one if the predicted probability of choosing the first alternative is greater than 0.5. To measure the classification accuracy, we calculate the percentage of the correctly classified observations. A mathematical representation of the used metrics is shown in Table 2A.2.

Table 2A.2. Performance metrics

Performance metric	Function
Average Log-Likelihood function	$\frac{1}{N}\sum_{n=1}^{N}\sum_{k=1}^{K} y_{nk}\ln(P_{nk})$
Classification accuracy	$\frac{1}{N}\sum_{n=1}^{N}\sum_{k=1}^{K} I$ where $I = \begin{cases} 1 & if\ y_{nk} = \widehat{y_{nk}} \\ 0 & otherwise \end{cases}$

References

Abu-Mostafa, Y. S. (1995). Hints. *Neural computation, 7*(4), 639-671.

Abu-Mostafa, Y. S., Magdon-Ismail, M., & Lin, H.-T. (2012). *Learning from data* (Vol. 4): AMLBook New York, NY, USA:.

Anthony, M., & Bartlett, P. L. (2009). *Neural network learning: Theoretical foundations*: cambridge university press.

Bartlett, P. L., & Maass, W. (2003). Vapnik-Chervonenkis dimension of neural nets. *The handbook of brain theory and neural networks*, 1188-1192.

Baum, E. B., & Haussler, D. (1989). *What size net gives valid generalization?* Paper presented at the Advances in neural information processing systems.

Bierlaire, M., Axhausen, K. W., & Abay, G. (2001). *The acceptance of modal innovation: The case of Swissmetro.* Paper presented at the Swiss Transport Research Conference.

Bishop, C. M. (1995). *Neural networks for pattern recognition*: Oxford university press.

Bishop, C. M. (2006). *Pattern recognition and machine learning*: springer.

Blum, A. L., & Langley, P. (1997). Selection of relevant features and examples in machine learning. *Artificial intelligence, 97*(1-2), 245-271.

Blumer, A., Ehrenfeucht, A., Haussler, D., & Warmuth, M. K. (1989). Learnability and the Vapnik-Chervonenkis dimension. *Journal of the ACM (JACM), 36*(4), 929-965.

Cantarella, G. E., & de Luca, S. (2005). Multilayer feedforward networks for transportation mode choice analysis: An analysis and a comparison with random utility models. *Transportation Research Part C: Emerging Technologies, 13*(2), 121-155.

Castelvecchi, D. (2016). Can we open the black box of AI? *Nature News, 538*(7623), 20.

Cho, J., Lee, K., Shin, E., Choy, G., & Do, S. (2015). How much data is needed to train a medical image deep learning system to achieve necessary high accuracy? *arXiv preprint arXiv:1511.06348*.

Chorus, C. G., & Bierlaire, M. (2013). An empirical comparison of travel choice models that capture preferences for compromise alternatives. *Transportation, 40*(3), 549-562.

Cireşan, D. C., Meier, U., & Schmidhuber, J. (2012). *Transfer learning for Latin and Chinese characters with deep neural networks.* Paper presented at the Neural Networks (IJCNN), The 2012 International Joint Conference on.

Cortes, C., Jackel, L. D., Solla, S. A., Vapnik, V., & Denker, J. S. (1994). *Learning curves: Asymptotic values and rate of convergence.* Paper presented at the Advances in Neural Information Processing Systems.

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals, and Systems (MCSS), 2*(4), 303-314.

Figueroa, R. L., Zeng-Treitler, Q., Kandula, S., & Ngo, L. H. (2012). Predicting sample size required for classification performance. *BMC medical informatics and decision making, 12*(1), 8.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning* (Vol. 1): MIT press Cambridge.

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

Hague Consulting Group. (1998). *The second Netherlands' value of time study: final report*. Report 6089-1 for AVV, HCG, Den Haag.

Haussler, D. (1992a). Decision theoretic generalizations of the PAC model for neural net and other learning applications. *Information and computation, 100*(1), 78-150.

Haussler, D. (1992b). Overview of the Probably Approximately Correct (PAC) Learning Framework. *Information and computation, 100*(1), 78-150.

Haykin, S. S. (2009). *Neural networks and learning machines* (Vol. 3): Pearson Upper Saddle River.

Hensher, D. A., Rose, J. M., & Greene, W. H. (2005). *Applied choice analysis: a primer*: Cambridge University Press.

Hensher, D. A., & Ton, T. T. (2000). A comparison of the predictive potential of artificial neural networks and nested logit models for commuter mode choice. *Transportation Research Part E: Logistics and Transportation Review, 36*(3), 155-172.

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural networks, 2*(5), 359-366.

Huber, J., & Train, K. (2001). On the similarity of classical and Bayesian estimates of individual mean partworths. *Marketing Letters, 12*(3), 259-269.

Iyer, M. S., & Rhinehart, R. R. (1999). A method to determine the required number of neural-network training repetitions. *IEEE Transactions on Neural Networks, 10*(2), 427-432.

Jain, A. K., & Chandrasekaran, B. (1982). 39 Dimensionality and sample size considerations in pattern recognition practice. *Handbook of statistics, 2*, 835-855.

Kalwani, M. U., Meyer, R. J., & Morrison, D. G. (1994). Benchmarks for discrete choice models. *Journal of marketing research*, 65-75.

Kavzoglu, T., & Mather, P. M. (2003). The use of backpropagating artificial neural networks in land cover classification. *International journal of remote sensing, 24*(23), 4907-4938.

Kohavi, R. (1995). *A study of cross-validation and bootstrap for accuracy estimation and model selection.* Paper presented at the Ijcai.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). *Imagenet classification with deep convolutional neural networks.* Paper presented at the Advances in neural information processing systems.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature, 521*(7553), 436-444.

LeCun, Y. A., Bottou, L., Orr, G. B., & Müller, K.-R. (2012). Efficient backprop *Neural networks: Tricks of the trade* (pp. 9-48): Springer.

McFadden, D. L. (1984). Econometric analysis of qualitative response models. *Handbook of econometrics, 2*, 1395-1457.

Mohammadian, A., & Miller, E. (2002). Nested logit models and artificial neural networks for predicting household automobile choices: comparison of performance. *Transportation Research Record: Journal of the Transportation Research Board*(1807), 92-100.

Mukherjee, S., Tamayo, P., Rogers, S., Rifkin, R., Engle, A., Campbell, C., . . . Mesirov, J. P. (2003). Estimating dataset size requirements for classifying DNA microarray data. *Journal of computational biology, 10*(2), 119-142.

Neelamegham, R., & Jain, D. (1999). Consumer choice process for experience goods: An econometric model and analysis. *Journal of marketing research*, 373-386.

Park, Y. R., Murray, T. J., & Chen, C. (1996). Predicting sun spots using a layered perceptron neural network. *IEEE Transactions on Neural Networks, 7*(2), 501-505.

Raudys, S. J., & Jain, A. K. (1991). Small sample size effects in statistical pattern recognition: Recommendations for practitioners. *IEEE Transactions on pattern analysis and machine intelligence, 13*(3), 252-264.

Ribeiro, B., Sung, A. H., Suryakumar, D., & Basnet, R. B. (2015). *The Critical Feature Dimension and Critical Sampling Problems.* Paper presented at the ICPRAM (1).

Ripley, B. D. (2007). *Pattern recognition and neural networks*: Cambridge university press.

Rose, J. M., & Bliemer, M. C. (2013). Sample size requirements for stated choice experiments. *Transportation, 40*(5), 1021-1041.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1988). Learning representations by back-propagating errors. *Cognitive modeling, 5*(3), 1.

Shalev-Shwartz, S., & Ben-David, S. (2014). *Understanding machine learning: From theory to algorithms*: Cambridge university press.

Silva, J., Ribeiro, B., & Sung, A. H. (2017). *Finding the Critical Sampling of Big Datasets.* Paper presented at the Proceedings of the Computing Frontiers Conference.

Sung, A., Ribeiro, B., & Liu, Q. (2016). *Sampling and evaluating the big data for knowledge discovery.* Paper presented at the Proceedings of International Conference on Internet of Things and Big Data (IoTBD 2016), Science and Technology Publications.

Train, K. E. (2009). *Discrete choice methods with simulation*: Cambridge university press.

van Cranenburgh, S., & Alwosheel, A. (2019). An artificial neural network based approach to investigate travellers' decision rules. *Transportation Research Part C: Emerging Technologies, 98*, 152-166.

van Cranenburgh, S., Guevara, C. A., & Chorus, C. G. (2015). New insights on random regret minimization models. *Transportation Research Part A: Policy and Practice, 74*, 91-109.

Vapnik, V. N., & Chervonenkis, A. Y. (2015). On the uniform convergence of relative frequencies of events to their probabilities *Measures of complexity* (pp. 11-30): Springer.

Vedaldi, A., & Lenc, K. (2015). *Matconvnet: Convolutional neural networks for matlab.* Paper presented at the Proceedings of the 23rd ACM international conference on Multimedia.

Using prototypical examples to diagnose artificial neural networks for discrete choice analysis

Alwosheel, A., van Cranenburgh, S., & Chorus, C. G. (2019). 'Computer says no'is not enough: Using prototypical examples to diagnose artificial neural networks for discrete choice analysis. Journal of Choice Modelling, 33, 100186.

Abstract:

Artificial Neural Networks (ANNs) are increasingly used for discrete choice analysis, being appreciated in particular for their strong predictive power. However, many choice modellers are critical – and rightfully so – about using ANNs, for the reason that they are hard to diagnose. That is, for analysts it is hard to see whether a trained (estimated) ANN has learned intuitively reasonable relationships, as opposed to spurious, inexplicable or otherwise undesirable ones. As a result, choice modellers often find it difficult to trust an ANN, even if its predictive performance is strong. Inspired by research from the field of computer vision, this paper pioneers a low-cost and easy-to-implement methodology to diagnose ANNs in the context of choice behaviour analysis. The method involves synthesising prototypical examples after having trained the ANN. These prototypical examples expose the fundamental relationships that the ANN has learned. These, in turn, can be evaluated by the analyst to see whether they make sense and are desirable, or not. In this paper we show how to use such prototypical examples in the context of choice data and we discuss practical considerations for successfully diagnosing ANNs. Furthermore, we cross-validate our findings using techniques from traditional discrete choice analysis. Our results suggest that the proposed method helps build trust in well-functioning ANNs, and is able to flag poorly trained ANNs. As such, it helps choice modellers use ANNs for choice behaviour analysis in a more reliable and effective way.

1 Introduction

Throughout the choice modelling community there is a considerable and increasing interest in using Artificial Neural Networks (ANNs) to analyse and predict choice behaviour. Recently, several papers have emerged which show the added value of ANNs in a variety of settings (Alwosheel, van Cranenburgh, & Chorus, 2018; Golshani, Shabanpour, Mahmoudifard, Derrible, & Mohammadian, 2018; Lee, Derrible, & Pereira, 2018; Sifringer, Lurkin, & Alahi, 2018; van Cranenburgh & Alwosheel, 2019). This recent and rapid increase in interest can be explained by impressive achievements of ANNs in other fields, such as speech recognition and image classification. In particular, ANNs' flexible modelling structure and their ability to work with abundant, complex and highly non-linear data allow them to outperform statistical models (e.g., discrete choice models), most notably in their prediction accuracy (Hagenauer & Helbich, 2017; Karlaftis & Vlahogianni, 2011).

But, despite their often superior prediction performance, many choice modellers remain understandably critical towards ANNs. An important reason for this relates to their proverbial 'black box'-nature, meaning that the trained network itself is hard to interpret; this severely limits the behavioural insights that may be drawn from them, and makes them notoriously hard to diagnose (Karlaftis & Vlahogianni, 2011).[14] Regarding this latter aspect (model diagnosis), ANNs are typically validated by relying on empirical prediction performance. For instance, a widely-used validation approach is to evaluate ANN prediction performance on out-of-sample set of data points (Haykin, 2009). The error returned by the ANN on the out-of-sample set is an approximation of the so-called generalisation error, which is the key metric for assessing an ANN's learning capability. Having a low (high) generalisation error implies that the underlying data generating process has been well (poor) approximated (Abu-Mostafa, Magdon-Ismail, & Lin, 2012; Breiman, 2001).

However, having a low generalisation error does not necessarily mean that the ANN has learned intuitively correct and desirable relationships. Due to the complexity of the learning task, in practice it occasionally happens that ANNs learn counterintuitive, inexplicable, or otherwise undesirable relations between variables. One evocative example in which an ANN learned undesirable relations occurred when it was used to select candidates in a recruitment context, and produced gender bias (i.e. favouring male candidates over female candidates) (Fernández & Fernández, 2019). While the ANN merely reproduced a bias that was implicitly present in the data on which it was trained, the opaqueness of the ANN made that such undesirable relations could stay unnoticed for a long time. Aside from this example, where an undesirable relation was learned by the ANN, in many cases a poorly trained network may have learned spurious or inexplicable relations between input and output data. This too, would create a problem in terms of the model's usefulness for making predictions, and it would not always surface in a process of merely evaluation prediction performance. In some fields trusting that the relationships between the explanatory variables and the predictions are reasonable, is more important than in others (Shmueli, 2010). In fields like natural language processing, the difficulty of diagnosing ANNs does not seem to substantially hinder applications; but, in the field of discrete choice behaviour analysis, having trust in the relations learned by a choice model, even when it is only used for predictions and not for behavioural or economic analysis, is rightfully considered very important.

Lately, the development of alternative techniques for validating and diagnosing ANNs in light of their black box-nature has been the subject of many debate, critiques, and research efforts in a variety of contexts (Lipton, 2016). Notably, in the computer vision field much research effort

[14] In the machine learning community, training is equivalent to estimating in choice modellers' parlance, see Appendix 3A for more information on training ANNs.

has recently been made (Montavon, Samek, & Müller, 2018; Samek, Wiegand, & Müller, 2017; Simonyan, Vedaldi, & Zisserman, 2013). A particularly interesting and easy-to-implement method that has been proposed in that field is based on the synthesis of so-called prototypical examples (Erhan, Bengio, Courville, & Vincent, 2009; Montavon et al., 2018). By synthesising prototypical examples using the ANN (after having trained it), the ANN exposes the fundamental relationships that it has learned. These, in turn, can be evaluated by the analyst to see if they are intuitively correct and desirable, or not. For instance, when creating a prototypical example of a cat using an ANN that is trained to discriminate between cats and dogs (note that this involves 'drawing' a cat, as opposed to selecting a cat from the training data set), we expect to see distinguishable characteristics of cats in the prototypical example, such as whiskers. Hence, (human) interpretation of the prototypical example allows the analyst to diagnose the rationale behind the network predictions by comparing it with his (or her) mental image of a cat. To the best of authors' knowledge, no study has yet pioneered the use of prototypical examples to diagnose ANNs used for choice behaviour analysis.

This paper pioneers the use of prototypical examples to diagnose ANNs for choice behaviour analysis. In particular, we show that the generation and interpretation of prototypical examples is a low-cost and easy-to-implement method to validate the rationale of ANNs for choice behaviour analysis, and as such building trust (or not) in the ANN. To this aim, we first show that the use of prototypical examples for diagnosis is not confined to the domain of visual data, by re-conceptualising this notion towards one that is applicable for choice analysis. Subsequently, we apply this method to a recently collected Revealed Preference (RP) mode choice data set. That is, we train an ANN on these RP data and diagnose it by synthesising and interpreting prototypical examples. Finally, we cross-validate the proposed method, by comparing the relationships that are exposed by the prototypical examples with those obtained from traditional discrete choice models that were estimated on these same data.

Before we proceed, we would like to emphasize that this research effort does not aim to promote using ANNs for discrete choice behaviour analysis. In our view, since the interpretability of ANNs remains limited, even after having diagnosed it with the method proposed in this paper, the natural domain of application of ANNs is forecasting, rather than behavioural and economic analysis. This research effort is motivated by the increasing use of ANNs for discrete choice analysis (and prediction in particular), which in our view presents a strong motivation to offer a low-cost and easy-to-implement method to test whether the relationships learned by an ANN in the training process, are intuitive and desirable. Without such a diagnosis, a choice modeller remains unsure to what extent to trust the ANN, hampering its usefulness as a tool to make predictions.

The remainder of this paper is organised as follows: Section 2 introduces the prototypical example methodology to the choice modelling community. Section 3 provides the empirical case study. It presents the main dataset that we use for our analysis and it discusses the training of the ANN. Section 4 presents the prototypical examples created using our trained ANN. It shows how the methodology can be used to as a tool for diagnosing an ANN trained in the context of choice data. Section 5 provides a cross-validation of the proposed method, using conventional discrete choice analysis techniques. Finally, Section 6 draws conclusions and presents potential directions for future research. A first appendix presents a textbook-level introduction of how to train ANNs. The second appendix provides an additional case study, in which we show how the proposed prototyping method can be used to flag a poorly trained ANN.

2 Methodology

2.1 Model interpretability and diagnosis

Opening the black-box of ANNs has received much attention in a variety of fields (Gunning, 2017). In the literature, several meanings have been attached to the effort of opening an ANN's black-box such as enhancing interpretability, explainability or understandability (Lipton, 2016). In this study, we use the term diagnosis which is closely related to the notion of interpretability; this latter concept is defined as the mapping of an abstract concept (e.g., a predicted mode choice) into a domain that the human can make sense of (Montavon et al., 2017). We consider the process of diagnosing an ANN to consist of the effort to test to what extent the ANN has learned intuitive and desirable relations and hence can be trusted[15]; this is done by interpreting a set of synthesized examples (see further below), although it should be noticed here, that a full interpretation of the ANN is not the aim here.

Nonetheless, movement towards interpretable ANNs has created important tools that may (also) be used to diagnose ANNs (Ribeiro, Singh, & Guestrin, 2016; Samek et al., 2017). This interpretability movement can be classified into ante-hoc and post-hoc approaches. In ante-hoc approaches interpretability is incorporated ('hard-wired') in the model structure. This approach entails designing a model such that its parameters and predictions can be interpreted by the analyst in a meaningful way. For choice modellers this approach is familiar, as the high level of interpretability of discrete choice models can be considered a result of imposing a strong structure as an ante-hoc approach. In other words, choice modellers routinely embed domain knowledge (e.g., theories on choice behaviour) into the model's structure. As a result, the parameters of, for instance, the widely used linear-additive Random Utility Maximisation (RUM) model (McFadden, 1973) can readily be interpreted as marginal utilities.

In post-hoc approaches interpretability comes after having trained a model (Montavon et al., 2018). Post-hoc methods provide bits of interpretation without elucidating or enforcing how the complete model works in full detail. Rather, they take a trained model and generate either a local (i.e., interpretation of a particular prediction) or a global interpretation of the model. Some post-hoc approaches have recently been applied to ANNs used in the choice modelling literature. Chiang, Zhang, and Zhou (2006) and Hagenauer and Helbich (2017) conduct sensitivity analysis to measure the importance of input variables for different types of trained ANNs. Golshani et al. (2018) implemented Garson's algorithm (Garson, 1991), which aims to determine the relative importance of input attributes, for explaining the ANN's predictions. Note that a recent study highlights the drawbacks of using sensitivity analysis for interpretability (Samek et al., 2017).

2.2 Synthesising prototypical examples for diagnosing an ANN

Synthesising prototypical examples is another post-hoc approach to provide interpretation to ANNs. It has been proposed in the computer vision field, where it is difficult to understand exactly how a trained ANN functions due to the large number of interacting and non-linear parts (e.g., an ANN called AlexNet consists of over 60 million parameters) (Krizhevsky, Sutskever, & Hinton, 2012). Synthesising prototypical examples is considered a low-cost method to uncover a sample of the learned patterns in images (Erhan et al., 2009; Simonyan et al., 2013). The idea of this approach is that by synthesising prototypical examples using a

[15] Note that a growing body of literature studies what constitutes a trustworthy model (Kulesza, Burnett, Wong, & Stumpf, 2015; Lipton, 2016; Miller, 2017).

trained ANN (note this means drawing an image, not selecting one from a data set), these prototypes expose fundamental relationships that the trained ANN has learned. As such, the analyst can assess the synthetically generated prototypes to see if they make sense, e.g. contain the expected characteristics. Hence, prototypical examples allow the analyst to diagnose part of the rationale behind the network predictions and build trust on the predictions by comparing it with his or her own mental map. Note that because the set of generated prototypical examples is generally small compared to the number of relations learned by the network, it is not able to generate a complete interpretation of the ANN; nonetheless, its use for diagnosing and building trust in ANNs is well established in the field of computer vision.

Activation maximisation

To synthesise prototypes, a technique called activation maximisation is used. This technique searches for the input that maximises the probability of a particular output label (e.g. a cat or a dog). The inputs that maximise the probability of a certain output label are called 'prototypical examples'; these examples, in the context of image classification, are synthetic drawings of a dog or a cat, rather than a particular image of a dog or a cat selected from the input data. The process of maximising the activation is much akin to the process of training an ANN (a detailed description of this process is presented in the Appendix for interested readers). In the context of choice data, the set of observations is given by $S = ((\mathbf{x}_1, \boldsymbol{y}_1), (\mathbf{x}_2, \boldsymbol{y}_2), \ldots, (\mathbf{x}_n, \boldsymbol{y}_n), \ldots, (\mathbf{x}_N, \boldsymbol{y}_N))$. Each n^{th} observation s_n contains a vector of independent variables $\mathbf{x}_n$ that represent the attributes of alternatives, socio-demographics, and possibly other covariates, and a *K*-dimensional vector of dependent variables $\boldsymbol{y}_n$ that represent the observed choice (i.e., zeros for non-chosen alternatives, and a one for the chosen alternative); *K* being the size of the choice set. The ANN training process aims to find the optimum weight parameters $\mathbf{w}$ such that the error function (which depends on the observations' variables $\mathbf{x}$ and $\boldsymbol{y}$) is minimised.

To maximise the activation within a trained ANN, we aim to find input values (i.e., the prototype) $\mathbf{x}^*$ such that the activation of a particular output neuron (i.e., the choice probability P_i for alternative i) is maximised (equation 1)

$$\mathbf{x}^* = \underset{\mathbf{x}}{\operatorname{argmax}}(P_i(\mathbf{w}, \mathbf{x}) - \lambda \,\|\mathbf{x}^*\|) \qquad (1)$$

where λ represents the regularisation term.[16]

Like the process of training an ANN, the process of finding $\mathbf{x}^*$ is iterative. At the beginning, the $\mathbf{x}$ values are initialised (either randomly or from some predefined point). Then, at each iteration step, P is increased by changing the $\mathbf{x}$ (note that the weights $\mathbf{w}$ are kept fixed during this process). A gradient ascent approach can be used for this purpose (Erhan et al., 2009). This process is repeated until a pre-specified stopping criterion is achieved. Note that equation (1) is non-convex, meaning that there are many local maxima in the search space. As such, it is possible to obtain several solutions, meaning that a single ANN can generate many different prototypical examples.

In case $\mathbf{x}$ is randomly initialised, the gradient ascent process usually is able to produce many prototypical examples. However, previous studies have shown that many of them can be useless, in the sense that they do not resemble meaningful inputs $\mathbf{x}$ (e.g. because they take extreme and unrealistic values (Nguyen, Yosinski, & Clune, 2015)). To overcome this problem, the optimisation process can be constrained to only generate examples that resemble realistic inputs (Nguyen, Dosovitskiy, Yosinski, Brox, & Clune, 2016). In particular, two controlling factors can be applied: First, instead of randomly initialising $\mathbf{x}$ values (as was initially proposed

[16] Regularisation is a frequently used method in machine learning to avoid overfitting by penalising the size the models flexibility (Bishop, 2006).

by Erhan et al. (2009)), **x** can be initialised based on the mean and variance of the attributes levels as they appear in the original data (Nguyen, Yosinski, & Clune, 2016). This is found to substantially improve the quality of the synthesised prototypes. Second, **x** values can be constrained to be within the range of the original data. In the field of image processing this practice is widely used (called clipping, see Yosinski, Clune, Nguyen, Fuchs, and Lipson (2015) for example). A pseudocode of the activation maximisation method can be found below.

```
Pseudocode: Activation Maximisation Method
Step 1: Initialisation
        Set x values (according to the selected initialisation method)
Step 2: Forward propagation
        Propagate the input neuron values x to output neuron through hidden neurons
        Calculate the ANN output neuron values (ANN probabilities)
Step 3: Backward propagation
        Calculate the gradient for the network input neurons
        Update x values (within the range of the original data)
        Go back to step 2 and repeat the process until the selected criterion is satisfied
```

2.3 Prototypical examples – a computer vision illustration

To further clarify the method, in this subsection we provide an illustration of how prototypical examples are used in the computer vision field. An illustration from the computer vision field is taken for two reasons: 1) many of the recent advancements in ANN research have taken place in this field; 2) visual examples are particularly effective in illustrating how the method works. The particular example is taken from Mordvintsev, Olah, and Tyka (2015). Mordvintsev et al. (2015) use an ANN to discriminate between several output classes, including a dumbbell (weight) class. To train this ANN, they presented the ANN with many images, including many images of dumbbells. To check whether the trained network has learned the relevant features of dumbbells (e.g., a dumbbell has a bar and weight plates) and ignores unrelated ones (e.g., weight plates can be of various shapes), prototypes of dumbbells were synthesised.

Fig. 1 shows four different prototypical examples of dumbbells generated by the trained ANN. From Fig 1 we can make a number of observations. First we notice that the four prototypes are not the same. But it is clear that they all present patterns that, once interpreted, can be recognised by the human analyst as dumbbells. Second, the prototypes always show a part of a muscular weightlifter's arm. Depending on the analyst expectations[17], this may either indicate that the trained network has failed to completely learn the features of a dumbbell (as it is mixed with the arm of a muscular weightlifter). Hence, the predictions of the trained network may not be fully trusted, as the trained ANN may confuse muscular weightlifter images with dumbbell images. A possible source of this 'failure' is that the examples used for training contain dumbbells and arms holding them. Most importantly for the purpose of our paper, this example illustrates that prototypical examples can be used to diagnose the rationale of a trained ANN.

[17] Analysts may have different beliefs and expectations. For example, some would expect (and accept) that the muscular weightlifter appears in dumbbells prototypes, whereas others may not. Most importantly, the synthesised prototypes can be used to reveal these relations learned by the ANN (either expected or not), and as such help the analyst to determine whether or not an analyst will trust the ANN.

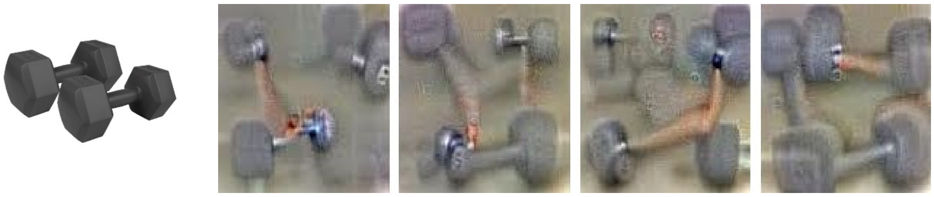


Figure 1. Left: Actual dumbbell. Right: Four prototypical examples of dumbbells (Mordvintsev et al., 2015).

3 Data and ANN training

3.1 Data preparation

For this study, we use revealed preference (RP) data from a study conducted for travel mode choice analysis in London city (Hillel, Elshafie, & Jin, 2018).[18,19] The chosen dataset contains four alternatives and a total of 27 features (i.e., attributes of alternatives and characteristics of decision makers). To prepare these data for our study, we took a number of steps. First, we removed features that were deemed redundant and merged them with others. For instance, rather than using three features to represent car cost (fuel, congestion, and total cost), we merged them into one total cost feature. Table 1 presents statistics on the attribute levels in the data set used for analysis. Second, we noticed that the data set was highly imbalanced in terms of chosen mode distribution: walking (17.6%), cycling (3.0%), public transport (35.3%) and driving (44.2%). Such imbalances could potentially be problematic for training ANNs (Haykin, 2009). In light of the aim of this paper (which does not focus on finding the best predictions of travel behaviour, but rather on testing a method to diagnose the ANN), we deemed dealing with this sort of data imbalances out of scope. Therefore, we 'repaired' this data imbalance by removing the cycling alternative from the data set. However, we consider exploring the use of prototypical examples in the context of imbalanced data sets an interesting avenue for further research. Third, we excluded very short trips (i.e., less than two minutes), as these were deemed not to contain a mode trade-off. The resulting dataset that is used for this study consist of 77,638 mode choice observations.

Table 1. Data statistics

No.	Attribute	Description	Type	Range [min, max]	Mean and standard deviation
1	TC_{Drive}	Estimated cost of driving route, including fuel cost and congestion charge	Float (£)	[0.05, 17.16]	(1.91, 3.48)
2	TC_{PubTr}	Estimated cost of public transport route, accounting for rail and bus fare types	Float (£)	[0, 13.49]	(1.56, 1.55)
3	TT_{Drive}	Predicted duration of driving route	Float (hours)	[0.03, 2.06]	(0.29, 0.25)

[18] The dataset and its description are available online, and can be downloaded from the first author profile at researchgate.net

[19] In addition, we use the well-known stated preference Swiss Metro data, as a second case study, to take a first step towards testing the general applicability of the proposed method. To avoid repetition in the main text, we present this second case study in Appendix 3B.

4	TT_{PubTr}	Predicted duration of public transportation	Float (hours)	[0.03, 2.73]	(0.47, 0.31)
5	TT_{Walk}	Predicted total duration of walking times for interchanges on public transport route	Float (hours)	[0.04, 9.27]	(1.15, 1.13)
6	DIS	Straight line distance between origin and destination	Integer (meters)	[96, 40941]	(4690, 4827)
7	$TRAF$	Predicted traffic variability on driving route	Float	[0, 1]	(0.34, 0.20)
8	$INTER$	Number of interchanges on public transport route from directions API	Integer	[0, 4]	(0.38, 0.62)
9	DL	Boolean identifier of a person making trip: 1 if person has driving license, 0 otherwise	Bool	[0, 1]	(0.62, 0.49)
10	CO	Car ownership of household person belongs to: no cars in household (0), less than one car per adult (1), one or more cars per adult (2)	Integer	[0, 2]	(0.99, 0.75)
11	BS	Bus fare scale of person making trip imputed from socio-economic data: 0 (free bus journeys), 0.5 (half price), 1 (full price)	Float	[0, 1]	(0.64, 0.47)
12	FEM	Boolean identifier of a person making trip: 1 if female, 0 otherwise	Bool	[0, 1]	(0.53, 0.49)
13	AG	Age of person making trip	Integer (years)	[5, 99]	(39.5, 19.3)

3.2 ANN development and training

The ANN is implemented in the Python environment using the open source deep learning library Keras (Chollet, 2015).[20] To train the ANN, the built-in training algorithm (which is used to update weights' values **w**) known as Adam is used (Kingma & Ba, 2014).[21] Prior to training the ANN, the data are normalised to reduce training time and minimize the probability of ending-up with suboptimal solutions. [22] A conventional three layers (input, output and one hidden layer) fully connected ANN structure is used (the used ANN has *M = 173; M* being the number of adjustable parameters).[23] To test the performance of the ANN to predict the travel mode choice, we conducted a so-called *k*-fold cross-validation method, with *k* = 5. The dataset is partitioned into five equally sized folds of (roughly) 15,528 observations. Then, a single fold is used for testing, while the remaining four folds are used for training. This process is repeated 5 times, where each of the five folds is used only once for testing.

[20] Code is available upon request from the first author.

[21] Note that the used training algorithm is more sophisticated than the described training process at the Appendix (i.e., the Adam algorithm implements a version of stochastic gradient descent), but both are based on a backpropagation training mechanism.

[22] Data normalisation is common practice for ANN training. In this study, the minimum and maximum values of data are normalised to -1 to +1.

[23] ANN complexity is adjusted using a cross validation approach (see (Alwosheel et al., 2018) for more details). To avoid overfitting, a commonly used rule-of-thumb in ANNs is that the sample size needs to be (at least) 10 times larger than the number of adjustable parameters in the network (Haykin, 2009). A recent study specifically dealing with sample size requirements for using ANNs in the context of choice models is more conservative, and recommends to use a sample size of (at least) 50 times the number of estimable weights (Alwosheel et al., 2018). Our sample size satisfies this condition and, therefore, we safely avoid overfitting issues.

Table 2 shows several performance metrics for the trained ANN. The reported performance metrics are averaged across the five folds. It shows that ANN achieves a fair prediction performance. For comparison, we also report the performance of a standard linear-additive RUM-MNL model.[24] As expected based on previous literature, the ANN outperforms the discrete choice model by a large margin. Table 3 shows the *k*-folds confusion matrix for the trained ANN. To construct the confusion matrix each observation is assigned to an alternative based on the highest probability as predicted by the ANN. Then, each prediction is compared to the true chosen alternative. The cells on the diagonal show the mean percentage of the observations that are correctly assigned, across the 5 folds. Additionally, the values between parentheses show the average ANN probabilities of the observations that are correctly classified. The off-diagonal cells show the mean percentage of observations that are misclassified, across the 5 folds. Values between parentheses show the average ANN probabilities of these observations. Table 3 shows that the ANN predicts driving choices quite well, while it has some more difficulty with accurately predicting choices for walking and public transport. However, it should be noticed that there can be different degrees of randomness associated with choices for different modes. Hence, it could be the case that it is inherently more difficult to accurately predict choices for public transport than for driving. Another possible explanation could be related to imbalances in the training data (i.e., the driving alternative has 45% share of the data, as mentioned above). It is well known, that under-representation of particular choice alternatives ('output labels' in machine learning parlance) can undermine the reliability of a trained ANN model. Dealing with such problems is beyond the interest of this work, but a plethora of methods and approaches are developed to combat these issues. For example, a commonly used approach is to synthesise more observations of the under-represented alternative (see e.g. Chawla, Bowyer, Hall, and Kegelmeyer (2002)). Another approach is based on penalising the ANN when it makes classification mistakes concerning under-represented alternatives. For further discussions on these methods, interested readers are referred to e.g., He and Garcia (2008) and Batista, Prati, and Monard (2004).

Table 2. Performance of the trained ANN

Performance metric	Function	Null model	ANN	Linear-additive RUM
Final Log-likelihood	$\sum_{n=1}^{N}\sum_{k=1}^{K} y_{nk}\ln(P_{nk})$	-86,625	-43,477	-50,704
Cross-entropy	$\frac{1}{N}\sum_{n=1}^{N}\sum_{k=1}^{K} y_{nk}\ln(P_{nk})$	-1.09	-0.56	-0.65
ρ^2	$\rho = 1 - \frac{LL(\hat{\beta})}{LL(0)}$	0	0.50	0.42

[24] Note that the *k*-fold method is not used for the RUM model. Rather, the RUM model is estimated one time using the whole dataset.

Table 3. Confusion matrix

		ANN Classification			
		Driving	Public Transport	Walking	∑
True chosen alternative	Driving	83.45 (0.68)	10.85 (0.20)	5.68 (0.12)	100% (1)
	Public Transport	21.6 (0.23)	71.52 (0.67)	6.7 (0.10)	100% (1)
	Walking	21.2 (0.24)	9.5 (0.20)	69.3 (0.56)	100% (1)

4 Results: prototypical examples

Now that we know that, as expected, the ANN greatly outperforms a conventional discrete choice model in terms of predictive power, the big question becomes: can we trust the ANN, that is, is its predictive power based on intuitive, explicable and desirable relations between in- and output variables which the network has learned (or not)? Such a conclusion cannot be drawn by simply inspecting prediction outcomes such as the ones presented above; and this is where the proposed method of creating prototypical examples comes in. Using the techniques described in Section 2, we let the trained ANN create typical examples of a choice for a car (driver), for public transport, and for walking. Inspecting of these synthetic examples allows us to determine to what extent we can trust the ANN. Table 4 shows, for each alternative (driving, public transport, walking), five prototypical examples created by the trained ANN. Note that each example is independently synthesised. That is, the initial inputs are independently initialised. To facilitate inspection, we employ a so-called vertical heat-map, where high values are depicted red and low values are depicted blue. For example, the red colour at the last attribute 'age' (AG) indicates high values (between 50 to 53), while blue indicates low values.

Table 4. Synthesised prototypical examples. See Table 1 for description of the attributes.[25]

Attribute No.		1	2	3	4	5	6	7	8	9	10	11	12	13
		Travel Cost		Travel time			Trip characteristics			Traveller characteritics				
		TC_{Driv}	TC_{PubTr}	TT_{Driv}	TT_{PubTr}	TT_{Walk}	DIS	TRAF	INTER	DL	CO	BS	FEM	AG
Driving	Ex. 1	6.42	3.81	0.26	2.51	5.15	15953	0.43	2.24	0.41	1.16	0.50	0.52	52
	Ex. 2	6.40	3.70	0.25	2.47	5.15	15381	0.42	2.23	0.42	1.16	0.55	0.55	51
	Ex. 3	6.34	3.74	0.24	2.49	4.90	14249	0.41	2.17	0.46	1.17	0.53	0.51	50
	Ex. 4	6.25	3.79	0.22	2.51	4.65	13006	0.43	2.08	0.52	1.18	0.53	0.54	48
	Ex. 5	6.37	3.63	0.28	2.50	5.27	15919	0.43	2.29	0.45	1.16	0.54	0.56	53
Public Tran	Ex. 1	8.94	7.78	1.37	0.92	3.12	17524	0.54	1.77	0.57	0.68	0.51	0.49	48
	Ex. 2	9.20	7.76	1.35	0.90	3.20	16756	0.54	1.66	0.54	0.68	0.50	0.46	49
	Ex. 3	9.47	7.60	1.34	0.89	3.18	16365	0.53	1.62	0.52	0.69	0.53	0.50	48
	Ex. 4	9.13	7.74	1.31	0.89	3.09	17001	0.56	1.69	0.55	0.67	0.48	0.53	48
	Ex. 5	9.01	7.79	1.35	0.91	3.06	17081	0.53	1.61	0.52	0.69	0.53	0.53	48
Walking	Ex. 1	8.40	8.75	0.52	1.60	0.07	198	0.49	1.69	0.46	0.72	0.47	0.53	40
	Ex. 2	8.79	8.68	0.53	1.63	0.07	179	0.47	1.70	0.46	0.72	0.45	0.51	41
	Ex. 3	8.36	8.78	0.51	1.58	0.06	181	0.50	1.67	0.45	0.71	0.51	0.48	41
	Ex. 4	8.41	8.82	0.51	1.58	0.06	199	0.51	1.72	0.51	0.70	0.50	0.54	41
	Ex. 5	8.66	8.82	0.50	1.60	0.06	198	0.53	1.65	0.44	0.74	0.48	0.45	39

[25] For optimal reading, this table is best shown in screen and coloured printing.

A number of inferences can be made based on Table 4. First, we use the examples to diagnose whether our ANN has learned intuitively correct relationships, as opposed to spurious or otherwise undesirable ones. For instance, Table 4 shows that prototypical examples in which a travel mode is chosen are associated with relatively low travel times for that mode (columns 3 to 5). For instance, in the prototypical examples of a choice to drive, driving has the lowest travel time of available modes. Likewise, we see that a prototypical choice to drive is associated with a high number of interchanges for the public transport alternative (column 8), a high number of owned cars (column 10) and a low variability in traffic conditions (column 7). Furthermore, the prototypical case in which walking is chosen is associated with a relatively low distance to the destination. All these results are all in line with expectations. Second, the synthesised prototypes resemble realistic relations. For instance, distances for driving prototypes are between 13 and 16 km, while they are around 200 m for walking prototypes. Furthermore, a prototypical driving trip takes about 15 minutes (0.25 hours), and a prototypical walking trip takes less than five minutes. Third, it should be noted that the prototypes may also show some unexpected patterns. For example, prototypes for driving are expected to show that a person has a driving license, but they do not. Instead, prototypes for public transport are associated with high driving license ownership. Although this is not as expected, finding such relations can be useful for understanding the data and the model. For instance, they could indicate that an attribute is not relevant for explaining predictions, or could point towards issues related to the data itself. For example, after seeing this result, we found that the driving alternative also includes car passenger, taxi, van and motor bike (see Hillel et al. (2018) for more information). This could explain why prototypical examples for driving alternative do not associate with driving license ownership.

Finally, as alluded to in section 2.2.1 for this method setting λ to an appropriate value is needed. In particular, we find that in case λ is set too small the regularization term will have little effect. As a result, the generated prototypical examples may have extreme attribute levels, while in case λ is set too large the prototypical examples will contain only noise. In this study, we have tested numerous values for λs and found $\lambda = 0.005$ to work best (in the sense that it resulted in the generation of prototypes that are meaningful, i.e. from the perspective of the analyst). This result is in line with work conducted in computer vision, where also $\lambda = 0.005$ is found to work well, see e.g. Nguyen, Dosovitskiy, et al. (2016).[26] As such, we recommend using this value in future work.

5 Cross-validation using dicrete choice models

This section cross-validates the results and interpretations presented in section 4. To do so, we compare our findings with results obtained from a traditional linear-additive RUM-MNL discrete choice model. More specifically, we use this estimated conventional discrete choice model for two purposes: First, to inspect whether the synthesised prototypical examples would also be considered realistic prototypes from the discrete choice model's perspective. In other words, do the prototypical examples obtain high prediction probabilities when fed into the choice model? Second, we use this choice model to put the derived interpretations (in section 4) to the test. Since the estimates of discrete choice models allow for straightforward inference of the importance of attributes, we can use them as a test to check the interpretations on attribute importance derived from the synthesised prototypes. It is very important to note here, that we do not aim to compare the trained ANN and the estimated discrete choice model (DCM) in

[26] Also note that we use the same regularisation value for the additional case study we show in Appendix 3B.

terms of their interpretability; clearly, the DCM beats the ANN in this regard, given its strong and behaviourally intuitive model structure which facilitates rigorous behavioural and economic interpretation. Our aim is different: given the strong predictive power of the ANN (compared to DCM), we use prototypical examples to diagnose and build trust in the model (which is a more modest aim than achieving a full interpretation); in this section we validate our diagnosis method by checking whether the generated prototypical examples are congruent with the estimated DCM, whose model structure we trust a priori.
Table 5 shows the estimation results of the DCM alongside the implied part-worth utilities.[27] As can be seen, and as is expected, all parameters have the intuitively correct sign and are highly significantly different from zero (see Appendix 3C for the model specifications). More interestingly, Table 6 shows that the prototypes created by the ANN can also be considered prototypes from the discrete choice model's perspective. That is, we see that the all prototypes yield very high choice probabilities from the estimated choice model's perspective. This result validates the synthesised examples. Furthermore, Table 5 confirms the interpretations derived in section 4 regarding attribute importance. Specifically, both the prototypes and the part-worth utilities indicate that travel time is a highly important factor the mode choice (i.e., prototypical examples are always associated with the lowest travel time). Additionally, the importance of car ownership and light traffic conditions for the driving alternative are cross-validated by respectively the 2nd and 3th largest associated part-worth utilities.

Table 4. Estimation results for RUM-MNL model.[28]

No. of observations	77,638		
Final LL	-50,704.14		
ρ^2	0.42		
Attribute	*Est.*	*Rob. t-values*	*Part-worth utility*
ASC_Drive	0	fixed	
ASC_PubTr	1.85	59.57	
ASC_Walk	2.62	73.91	
TT	-6.11	-95.48	4.30
TC	-0.121	-7.94	0.27
DL	0.994	44.23	0.994
BS	-0.112	-4.73	0.112
CO	1.39	90.38	2.78
INTER	0.767	38.02	0.767
TRAF	-2.77	-43.00	1.04
AG_TC	-0.121	-3.92	0.09
DIS_TC	0.00840	9.75	0.12
FEM_TC	-0.0348	-4.08	0.05

[27] To calculate the part-worth utility the attribute level range is multiplied by parameter estimation. However, since we work with RP data here we use the 20-80 percentile range (as opposed to the full range). For example, consider an attribute A that consists of uniformly distributed values between 0 and 100. From that, we get the 20-80 percentile range (60 in this case) and multiply it by the parameter estimate.

[28] Note that the model is estimated using the whole dataset (*k*-folds method is not used in this case).

Table 5. Choice probabilities for prototypical examples, based on discrete choice model.

Alternatives	Examples No.	P_{Drive}	P_{PubTr}	P_{Walk}
Alt. 1: Driving	Example 1	0.9999	0.0000	0.0000
	Example 2	0.9999	0.0000	0.0000
	Example 3	0.9999	0.0000	0.0000
	Example 4	0.9999	0.0000	0.0000
	Example 5	0.9999	0.0000	0.0000
Alt. 2: Public Transport	Example 1	0.0025	0.9974	0.0000
	Example 2	0.0027	0.9972	0.0000
	Example 3	0.0027	0.9972	0.0000
	Example 4	0.0030	0.9969	0.0000
	Example 5	0.0032	0.9967	0.0000
Alt. 3: Walking	Example 1	0.0010	0.0000	0.9989
	Example 2	0.0009	0.0000	0.9990
	Example 3	0.0010	0.0000	0.9988
	Example 4	0.0010	0.0000	0.9988
	Example 5	0.0010	0.0000	0.9989

6 Conculsions and recommendations

This study contributes to the growing literature which focuses on using machine learning techniques for choice behaviour analysis, by pioneering a post-hoc methodology for diagnosing trained ANNs (in the context of choice behaviour analysis). We show how the proposed methodology can be easily applied at low cost to build trust in an ANN which was trained to predict (mode) choice behaviour. Based on our encouraging results, we believe that the proposed methodology provides a valuable tool for discrete choice modellers. It is however crucial to mention once more that the proposed method does not entirely open-up the black box of an ANN. As such, in our view of the most natural domain of application of ANNs still lies in forecasting tasks, as opposed to behavioural or economic analysis; our prototypical examples method helps the analyst to determine whether or not to trust predictions made by the trained ANN.

We would like to point out several limitations to this study, providing avenues for future research. Firstly, to avoid synthesising unrealistic prototypical examples, in our study the prototypical examples are randomly initialised according to a pre-defined distribution (normal distribution). In future research, a more accurate initialisation process can be employed. In particular, incorporating generative models (e.g., generative adversarial network as proposed in Goodfellow et al. (2014)) in the initialisation process may produce more reliable prototypes. Secondly, the empirical analyses provided in this paper are based on two datasets (one of which is presented in Appendix 3B). It is advisable to repeat these type of analyses using more datasets having different characteristics, e.g. many attributes, more alternatives, data imbalances, etc.. This will provide a richer view on the extent to which the proposed 'prototypical examples' methodology is a valuable tool to diagnose trained ANNs more generally. Lastly, the method is based on an inherently subjective process on the side of the analyst, when deriving interpretations from synthesised prototypes; that is, the analyst ultimately decided to what extent the generated examples match his or her expectations regarding the phenomenon being modelled (e.g. mode choices). Although we believe that this is certainly not a disadvantage per se, it is worthwhile to develop more objective methods or guidelines to extract and interpret prototypes. This will improve the rigour of this method and ultimately will help analysts to better understand the potential and limitations of using ANNs for discrete choice analysis.

Acknowledgement

We are grateful to three anonymous reviewers for their helpful suggestions on an earlier version of this manuscript. The first author would like to thank King Abdulaziz City for Science and Technology (KACST) for supporting this work. The third author would like to acknowledge funding from the ERC, consolidator grant BEHAVE - 724431.

Appendix 3A. Training of ANNs

ANNs are biologically inspired systems that have proven to be a powerful technique for machine learning. They are well-known for being highly effective in solving complex classification and regression problems (Haykin, 2009). ANNs consist of highly interconnected processing elements, called neurons, which communicate together to perform a learning task, such as classification, based on a set of observations. There are three types of neurons: input neurons, hidden neurons, and output neurons. Input neurons represent the independent variables. In the context of choice modelling, these are the alternatives' attributes, characteristics of decision-makers, and contextual factors. Output neurons contain the dependent variables. In a discrete choice context, these are the choice probabilities for each alternative. Neurons in-between are called hidden neurons because their inputs and outputs are connected to other neurons and are therefore 'invisible' to the analyst.

Each neuron in the network (except input neurons) receives inputs multiplied by estimable parameters known as weights (w_i). The weighted inputs are accumulated and added to a constant (called bias) to form a single input for a pre-defined processing function known as activation function. The bias has the effect of increasing or decreasing the net input of the activation function by a constant value, which increases the ANNs flexibility (Haykin, 2009). The activation function generates one output that is fanned out to other neurons. Commonly used activation functions are tan-sigmoid, softmax and step function.

In this work, we use the widely implemented ANN structure that consists of layers of neurons connected successively, known as multilayer perceptron (MLP) structure. Three elements need to be defined for MLP structure: activation functions, number of layers, and number of neurons at each layer. These three elements are set according to the desired objective of the modelling effort. For example, adding more layers or more hidden neurons increase the complexity of the network. In this paper, we use the so-called shallow version of ANN (i.e., an ANN with input layer, output layer, and one hidden layer). The complexity of which is adjusted by modifying the number of hidden neurons (we found that 20 hidden neurons provides satisfactory performance). The activation functions employed in this paper are tang-sigmoid softmax function for hidden and output neurons, respectively.

Once these three elements are defined, the training process that aims to find the model's parameter $(\mathbf{w})$ is implemented. The choice data that we use for training consist of set of observation observations $S = ((\mathbf{x}_1, \boldsymbol{y}_1), (\mathbf{x}_2, \boldsymbol{y}_2), \ldots, (\mathbf{x}_n, \boldsymbol{y}_n), \ldots, (\mathbf{x}_N, \boldsymbol{y}_N))$. Each n^{th} observation s_n contains a vector of independent variables $\mathbf{x}_n$ that represent the attributes and a *K*-dimensional vector of dependent variables $\boldsymbol{y}_n$ that represent the observed choice (i.e., zeros for the non-chosen alternatives, and a one for the chosen alternative); *K* being the size of the choice set. Since choices are mutually exclusive (i.e., only one alternative can be chosen from the choice set), from a machine learning perspective this is considered a classification problem.

The central goal of training is to model the underlying data generating process (DGP) that has led to the current set of observations, so that the best possible prediction for future observation is achieved (Bishop, 1995). While to estimate the parameter of a choice model the likelihood function is maximised, for ANN training an equivalent so-called error function $J(\mathbf{w})$ is minimised. We define $\mathbf{w}$ as a vector that contains the ANN estimable parameters w. Assuming the data consist of N choice observations across K alternatives, the error function is defined as follows:

$$J(\mathbf{w}) = -\sum_{n=1}^{N}\sum_{k=1}^{K} y_{nk}\ln(P_{nk}) \qquad (3A.1)$$

Where y_{nk} is an indicator which denotes whether alternative k is chosen in observation n, and P_{nk} is the choice probability predicted by the ANN, which is a function of $\mathbf{w}$ and $\mathbf{x}$. By training the ANN, the analyst's objective is to find the weight vector $\mathbf{w}$ such that $J(\mathbf{w})$ is minimised. This process can be described as follows:

$$\mathbf{w}^* = \underset{\mathbf{w}}{\operatorname{argmin}}\ J(\mathbf{w}) \qquad (3A.2)$$

The process of finding optimum weights ($\mathbf{w}^*$) is conducted in successive steps. At each step, $J(\mathbf{w})$ is decreased by adjusting the parameters in $\mathbf{w}$. The well-known gradient descent approach is the widely applied algorithm for this purpose.[29] In short, this process of training an ANN can be described as follows: first, the weights' values w are randomly initialised. The input neurons' values (taken from the training data) are propagated to the output layer through the hidden layer, this process is called forward propagation. Then, the output neurons' values (i.e., choice probabilities) are compared with the observed choices to compute the function $J(\mathbf{w})$ described in Equation $(3A.1)$. The optimisation mechanism is then conducted by propagating J backward to the input layers through the hidden layer. To adjust the weights, the backward propagation process includes taking the partial derivative of the error J with respect to the weights, called the gradient vector $\mathbf{g}$. Along with an adaptive or predefined learning rate value, $\mathbf{w}$ values are re-adjusted.

The process of error (forward and backward) propagation is repeated iteratively until a pre-specified stopping criterion is achieved. This training mechanism is known as back-propagation, and constitutes the most popular approach to train neural networks (Rumelhart, Hinton, & Williams, 1988). However, it should be noted that moving toward a local minimum is one of the widely reported risks associated with this back-propagation approach (Iyer & Rhinehart, 1999; Park, Murray, & Chen, 1996). As such, it is always recommended to train the network more than once to minimise the probability of ending up with a sub-optimal trained network. A pseudocode of the ANN training can be found below, and for comprehensive description of ANNs training interested readers are referred to Bishop (2006).

Appendix 3B. Results of Swiss Metro data

This section presents the outcome of applying the proposed approach on the Swiss Metro data (Bierlaire, Axhausen, & Abay, 2001). The data are pre-processed such that only travel time and

[29] Note that in case of training deeper or more complex ANNs, sophisticated algorithms inspired form the gradient descent (e.g., stochastic gradient descent) are used.

cost attributes are considered.[30] We used the processed data to train a three-layers fully connected network (see Alwosheel et al. (2018) for ANN complexity adjustment for Swiss Metro data). Note that the training process is similar to the process conducted in section 3 (e.g., same training built-in algorithm, and the k-folds cross-validation method).

Table 3B.1. Performance of the trained ANN

Performance metric	Null model	ANN	Linear-additive RUM
Final Log-likelihood	-9,849.2	-6,485.9	-6,714.54
Cross-entropy	-1.09	-0.70	-0.75
ρ^2	0	0.36	0.32

Table 3B.2 shows, for each alternative (car, Swiss Metro, train), five prototypical examples synthesised using the trained ANN. Note that each example is independently synthesised (i.e., the initial inputs are independently initialised). To facilitate inspection, we employ a so-called vertical heat-map, where high values are depicted red and low values are depicted blue.

Table 3B.2. Synthesised prototypical examples

		TC			TT		
		Car	SM	Train	Car	SM	Train
Train	Ex. 1	94.48	252.83	10.72	110.32	99.07	150.23
	Ex. 2	94.48	252.83	10.72	110.32	99.07	150.23
	Ex. 3	94.48	252.83	10.72	110.32	99.07	150.23
	Ex. 4	94.48	252.83	10.72	110.32	99.07	150.23
	Ex. 5	94.48	252.83	10.72	110.32	99.07	150.23
SM	Ex. 1	93.91	0.56	105.36	170.23	74.78	158.22
	Ex. 2	93.91	0.56	105.36	170.23	74.78	158.22
	Ex. 3	93.91	0.56	105.36	170.23	74.78	158.22
	Ex. 4	93.91	0.56	105.36	170.23	74.78	158.22
	Ex. 5	93.91	0.56	105.36	170.23	74.78	158.22
Car	Ex. 1	77.80	129.12	85.66	129.28	139.11	258.43
	Ex. 2	77.80	129.12	85.66	129.28	139.11	258.43
	Ex. 3	77.80	129.12	85.66	129.28	139.11	258.43
	Ex. 4	77.80	129.12	85.66	129.28	139.11	258.43
	Ex. 5	77.80	129.12	85.66	129.28	139.11	258.43

A number of inferences can be made based on Table 3B.2. First, although each example is independently initialised, the synthesised prototypes for each alternative are almost identical, implying that the ANN has less flexibility due to using only two attributes. Second, the synthesised examples show that the ANN has learned the expected relations. For example, the prototypical examples in which a travel mode is chosen are associated with relatively low travel cost for that mode.

The prototypes shown in Table 3B.2 are cross-validated using the standard linear-additive RUM-MNL (estimation results of RUM-MNL model are shown at Table 3B.3). We use this estimated choice model to inspect whether the synthesised prototypical examples would also

[30] The minimum and maximum values of data are normalised to -1 and +1.

be considered prototypical examples from the choice model's perspective. As expected, the estimated choice model returns very high choice probabilities for all prototypes (resulted probabilities are similar to the results reported at Table 6).

Table 3B.3. Estimation results of RUM-MNL model

No. of observations	9,036	
Final LL	-6,714.54	
ρ^2	0.32	
Attribute	*Est.*	*Rob. t-values*
TT	-2.01	-55.7
TC	-1.03	-24.79

Finally, we use the Swiss Metro data to show how the proposed method can be used to detect or flag a (deliberately) poorly trained ANN. More specifically, we would like to present a situation where the synthesised prototypes show patterns that are unexpected by the analyst, signalling problems with the trained ANN. To do so, we deliberately train a far too complex ANN that consists of two hidden layers (with 500 nodes each). The complexity of this ANN is much higher than the complexity of the problem at hand, which results in a poor model that fails to approximate the underlying data generating process (Vapnik, 2013). For this network, we did not apply the k-folds cross-validation method (the data are randomly separated into two parts: 80% of the data is used for training and the remaining 20% is used for testing). The performance of the trained ANN on training and testing data is presented at Table 3B.4. Results show that the trained ANN obtained excellent prediction performance in the training data but very poor performance in the testing data. This result in itself already signals a so-called overfitting problem, which occurs when the ANN is excessively complex comparing to the underlying data generating process. The excellent performance in the training data is obtained because the ANN complexity allows for fitting the training data perfectly (i.e., including noise artefacts). We choose to apply an overfitting scenario because it is a common mistake due to the ANN capability and flexibility (Abu-Mostafa et al., 2012), although clearly for this extreme overfitting situation, inspection of conventional performance metrics would already suggest to the analyst that something is wrong with the ANN.

Table 3B.4. Performance of the trained ANN (with two hidden layers)

Performance metric	Training data	Testing data
Cross-entropy	-0.14	-2.09
Hit-rate	0.94	0.63

Table 3B.5 shows, for each alternative (car, Swiss metro, train), five prototypical examples synthesised using the trained ANN. Note that each example is independently synthesised (i.e., the initial inputs are independently initialised). To facilitate inspection, we employ a so-called vertical heat-map, where high values are depicted red and low values are depicted blue.

Table 3B.5. Synthesised prototypical examples

		Travel Cost (CHF)			Travel Time (min)		
		Car	SM	Train	Car	SM	Train
Train	Ex. 1	86.90	1069.96	463.03	145.21	85.86	162.74
	Ex. 2	86.58	16.27	56.07	142.81	101.38	179.01
	Ex. 3	85.85	812.39	377.17	142.95	86.94	176.58
	Ex. 4	81.63	1195.12	387.19	146.57	67.47	246.98
	Ex. 5	85.85	812.40	377.17	142.95	86.94	176.58
SM	Ex. 1	82.70	12.29	9.77	154.68	118.15	194.99
	Ex. 2	103.07	12.30	280.78	143.56	90.65	152.66
	Ex. 3	91.07	12.34	85.23	117.82	98.20	190.21
	Ex. 4	94.09	523.81	399.89	150.27	90.07	170.77
	Ex. 5	94.10	523.81	399.89	150.27	90.07	170.77
Car	Ex. 1	93.44	474.00	359.49	146.01	92.04	173.51
	Ex. 2	93.44	474.00	359.49	146.01	92.04	173.51
	Ex. 3	93.44	474.01	359.49	146.01	92.05	173.51
	Ex. 4	93.44	474.01	359.49	146.01	92.03	173.51
	Ex. 5	67.60	191.72	49.38	149.23	107.10	137.95

The prototypes shown at Table 3B.5 clearly reveal that the ANN has not really learned the expected patterns. For example, considering the first train prototype, the car alternative is actually more attractive than the train alternative (cheaper and faster). This pattern is not expected by travel behaviour analyst. Therefore, the network in this case cannot be trusted.

Appendix 3C. Specifications of linear additive random utility maximisation model

Table 3C.1 shows the observed utility function for the linear-additive random utility maximisation (RUM) model used in this study (see Table 1 for attributes' name, notation, and description). The model is estimated in Multinomial Logit (MNL) form.

Table 3C.1. Utility function specifications

$$\begin{aligned} V_{Drive} = {} & ASC_{Drive} + \beta_{TT} TT_{Drive} + \beta_{TC} TC_{Drive} + \beta_{AG\_TC}(AG * TC_{Drive}) + \beta_{DIS\_TC}(DIS \\ & * TC_{Drive}) + \beta_{FEM\_TC}(FEM * TC_{Drive}) + \beta_{DL} DL + \beta_{CO} CO + \beta_{TRAF} TRAF \end{aligned}$$

$$\begin{aligned} V_{PubTr} = {} & ASC_{PublTr} + \beta_{TT} TT_{PubTr} + \beta_{TC} TC_{PubTr} + \beta_{AG\_TC}(AG * TC_{PubTr}) \\ & + \beta_{DIS_{TC}}(DIS * TC_{PubTr}) + \beta_{FEM\_TC}(FEM * TC_{PubTr}) + \beta_{BS} BS \\ & + \beta_{INTER} INTER \end{aligned}$$

$$\begin{aligned} V_{Walk} = {} & ASC_{Walk} + \beta_{TT} TT_{Walk} + \beta_{TC} TC_{Walk} + \beta_{AG\_TC}(AG * TC_{Walk}) + \beta_{DIS\_TC}(DIS \\ & * TC_{Walk}) + \beta_{FEM\_TC}(FEM * TC_{Walk}) \end{aligned}$$

Notations

V_i	Observed part of utility of alternative *i*
ASC_i	Specific constant of alternative *i*
β_{TT}	Taste parameter associated with travel time attribute
β_{TC}	Taste parameter associated with travel cost attribute
β_{AG_TC}	Taste parameter associated with interaction between age and travel cost attribute
β_{DIS_TC}	Taste parameter associated with interaction between travel distance and travel cost attribute
β_{FEM_TC}	Taste parameter associated with interaction between gender and travel cost attribute

β_{DL}	Taste parameter associated with driving license attribute
β_{CO}	Taste parameter associated with number of owned car attribute
β_{TRAF}	Taste parameter associated with traffic variability attribute
β_{BS}	Taste parameter associated with bus scale attribute
β_{INTER}	Taste parameter associated with number of interchanges attribute

References

Abu-Mostafa, Y. S., Magdon-Ismail, M., & Lin, H.-T. (2012). *Learning from data* (Vol. 4): AMLBook New York, NY, USA:.

Alwosheel, A., van Cranenburgh, S., & Chorus, C. G. (2018). Is your dataset big enough? Sample size requirements when using artificial neural networks for discrete choice analysis. *Journal of Choice Modelling, 28*, 167-182. doi:https://doi.org/10.1016/j.jocm.2018.07.002

Batista, G. E., Prati, R. C., & Monard, M. C. (2004). A study of the behavior of several methods for balancing machine learning training data. *ACM SIGKDD explorations newsletter, 6*(1), 20-29.

Bierlaire, M., Axhausen, K., & Abay, G. (2001). *The acceptance of modal innovation: The case of Swissmetro.* Paper presented at the Swiss Transport Research Conference.

Bishop, C. M. (1995). *Neural networks for pattern recognition*: Oxford university press.

Bishop, C. M. (2006). *Pattern recognition and machine learning*: springer.

Breiman, L. (2001). Statistical modeling: The two cultures (with comments and a rejoinder by the author). *Statistical science, 16*(3), 199-231.

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: synthetic minority over-sampling technique. *Journal of artificial intelligence research, 16*, 321-357.

Chiang, W.-y. K., Zhang, D., & Zhou, L. (2006). Predicting and explaining patronage behavior toward web and traditional stores using neural networks: a comparative analysis with logistic regression. *Decision Support Systems, 41*(2), 514-531.

Chollet, F. (2015). Keras.

Erhan, D., Bengio, Y., Courville, A., & Vincent, P. (2009). Visualizing higher-layer features of a deep network. *University of Montreal, 1341*(3), 1.

Fernández, C., & Fernández, A. (2019). Ethical and Legal Implications of AI Recruiting Software. *ERCIM NEWS*(116), 22-23.

Garson, G. D. (1991). Interpreting neural-network connection weights. *AI expert, 6*(4), 46-51.

Golshani, N., Shabanpour, R., Mahmoudifard, S. M., Derrible, S., & Mohammadian, A. (2018). Modeling travel mode and timing decisions: Comparison of artificial neural networks and copula-based joint model. *Travel Behaviour and Society, 10*, 21-32.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., . . . Bengio, Y. (2014). *Generative adversarial nets.* Paper presented at the Advances in neural information processing systems.

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

Haykin, S. S. (2009). *Neural networks and learning machines* (Vol. 3): Pearson Upper Saddle River.

He, H., & Garcia, E. A. (2008). Learning from imbalanced data. *IEEE Transactions on Knowledge & Data Engineering*(9), 1263-1284.

Hillel, T., Elshafie, M. Z., & Jin, Y. (2018). Recreating Passenger Mode Choice-Sets for Transport Simulation. *Proceedings of the Institution of Civil Engineers-Smart Infrastructure and Construction*, 1-49.

Iyer, M. S., & Rhinehart, R. R. (1999). A method to determine the required number of neural-network training repetitions. *IEEE Transactions on Neural Networks, 10*(2), 427-432.

Karlaftis, M. G., & Vlahogianni, E. I. (2011). Statistical methods versus neural networks in transportation research: Differences, similarities and some insights. *Transportation Research Part C: Emerging Technologies, 19*(3), 387-399.

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

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). *Imagenet classification with deep convolutional neural networks.* Paper presented at the Advances in neural information processing systems.

Kulesza, T., Burnett, M., Wong, W.-K., & Stumpf, S. (2015). *Principles of explanatory debugging to personalize interactive machine learning.* Paper presented at the Proceedings of the 20th international conference on intelligent user interfaces.

Lee, D., Derrible, S., & Pereira, F. C. (2018). Comparison of Four Types of Artificial Neural Network and a Multinomial Logit Model for Travel Mode Choice Modeling. *Transportation Research Record*, 0361198118796971. doi:10.1177/0361198118796971

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

McFadden, D. (1973). Conditional logit analysis of qualitative choice behavior.

Miller, T. (2017). Explanation in artificial intelligence: insights from the social sciences. *arXiv preprint arXiv:1706.07269*.

Montavon, G., Samek, W., & Müller, K.-R. (2018). Methods for interpreting and understanding deep neural networks. *Digital Signal Processing, 73*, 1-15.

Mordvintsev, A., Olah, C., & Tyka, M. (2015). Inceptionism: Going deeper into neural networks. *Google Research Blog. Retrieved June, 20*(14), 5.

Nguyen, A., Dosovitskiy, A., Yosinski, J., Brox, T., & Clune, J. (2016). *Synthesizing the preferred inputs for neurons in neural networks via deep generator networks.* Paper presented at the Advances in Neural Information Processing Systems.

Nguyen, A., Yosinski, J., & Clune, J. (2015). *Deep neural networks are easily fooled: High confidence predictions for unrecognizable images.* Paper presented at the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

Nguyen, A., Yosinski, J., & Clune, J. (2016). Multifaceted feature visualization: Uncovering the different types of features learned by each neuron in deep neural networks. *arXiv preprint arXiv:1602.03616*.

Park, Y. R., Murray, T. J., & Chen, C. (1996). Predicting sun spots using a layered perceptron neural network. *IEEE Transactions on Neural Networks, 7*(2), 501-505.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1988). Learning representations by back-propagating errors. *Cognitive modeling, 5*(3), 1.

Samek, W., Wiegand, T., & Müller, K.-R. (2017). Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models. *arXiv preprint arXiv:1708.08296*.

Shmueli, G. (2010). To explain or to predict? *Statistical science, 25*(3), 289-310.

Sifringer, B., Lurkin, V., & Alahi, A. (2018). *Enhancing Discrete Choice Models with Neural Networks.* Paper presented at the hEART 2018–7th Symposium of the European Association for Research in Transportation conference.

Simonyan, K., Vedaldi, A., & Zisserman, A. (2013). Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*.

van Cranenburgh, S., & Alwosheel, A. (2019). An artificial neural network based approach to investigate travellers' decision rules. *Transportation Research Part C: Emerging Technologies, 98*, 152-166.

Vapnik, V. (2013). *The nature of statistical learning theory*: Springer science & business media.

Yosinski, J., Clune, J., Nguyen, A., Fuchs, T., & Lipson, H. (2015). Understanding neural networks through deep visualization. *arXiv preprint arXiv:1506.06579*.

Toward explainable artificial neural networks for travel demand analysis

Alwosheel, A., van Cranenburgh, S., & Chorus, C. G. Why did you predict that? Toward explainable artificial neural networks for travel demand analysis. Transportation Research Part C: Emerging Technologies. Manuscript under review.

Abstract:
Artificial Neural Networks (ANNs) are rapidly gaining popularity in transportation research in general and travel demand analysis in particular. While ANNs typically outperform conventional methods in terms of predictive performance, they suffer from limited explainability. That is, it is very difficult to assess whether or not particular predictions made by an ANN are based on intuitively reasonable relationships. As a result, it is difficult for the analyst to gain trust in ANNs. We show that often-used approaches using perturbation (sensitivity analysis) are ill-suited for understanding the inner workings of ANNs. Subsequently, we introduce to the domain of transportation an alternative method, inspired by recent progress in the field of computer vision. This method is based on a re-conceptualisation of the idea of heat maps to explain predictions of a trained ANN. To create a heat map a prediction of an ANN is propagated backward in the ANN towards the input variables, using a technique called Layer-wise Relevance Propagation (LRP). The resulting heat map shows the contribution of each input value; for example the travel time of a certain mode, for a given travel mode choice prediction. By doing this, the heat map reveals the rationale behind the prediction in a way that is understandable to humans. If the rationale makes sense to the analyst, she or he will gain trust in the prediction, and, by extension, in the trained ANN as a whole. If not, the analyst may choose to adapt or re-train the ANN or decide not to use it. We show that by reconceptualising the LRP methodology towards the travel demand analysis context, it can be put to effective use in application domains well beyond computer vision.

1 Introduction

Artificial Neural Networks (ANNs) are emerging as an indispensable tool for many applications in the field of transportation. Recent examples include modelling lane-changing behaviour of drivers (Xie, Fang, Jia, & He, 2019), predicting mode choice behaviour (Sun et al., 2018), predicting traffic flow (Polson & Sokolov, 2017), and investigating travellers' decision rules (Alwosheel, van Cranenburgh, & Chorus, 2017; van Cranenburgh & Alwosheel, 2019). This increase in ANNs' popularity in transportation research is mainly driven by the abundance of data from a variety of emerging sources (Chen, Ma, Susilo, Liu, & Wang, 2016), and the ANN's often impressive predictive performance (Goodfellow, Bengio, & Courville, 2016; Karlaftis & Vlahogianni, 2011; Mckinney et al., 2020).

Although ANNs often obtain superior prediction performance compared to their conventional, more theory-driven counterparts (e.g. discrete choice models in a travel demand analysis context) their opaque nature makes explaining individual predictions which are made by an ANN very difficult. Without sufficient understanding of how and why a model makes a particular prediction, the use of ANNs will mainly be confined to niche settings where prediction performance is highly valued (e.g., short term travel demand predictions) and model transparency is not of great importance. For justifiable reasons, governments and transport planning agencies put a higher premium on model transparency (which is considered a prerequisite for good governance), than on superior empirical prediction performance.

Recently, the development of techniques for opening up and explaining the ANN's black-box has been the subject of many research efforts in a variety of research fields (Lipton, 2016). Notably, in the computer vision field much progress has been made to shed light on the inner workings of trained ANNs (Montavon, Samek, & Müller, 2018; Samek, Wiegand, & Müller, 2017; Simonyan, Vedaldi, & Zisserman, 2013). The Layer-wise Relevance Propagation (LRP) method has emerged as one of the most popular approaches to inspect the rationale behind ANNs' predictions (Adebayo et al., 2018). The LRP method generates a so called heat map. For example, the heat map of an ANN trained to discriminate between dogs and cats based on pictures, highlights which parts of an image (e.g., pixels representing cat whiskers) were most relevant for the produced prediction (in casu: cat). The generated heat map reveals the rationale of a trained ANN and as such allows for intuitive investigation of what made the model produce a prediction; in case the rationale aligns with the mental map of the analyst, this helps to build trust in that prediction. In case the exhibited rationale does not align, this of course still offers valuable information to the analyst. In principle, LRP (and related techniques) could also be of use to analysts working in other contexts than computer vision, including transportation. But, to the best of the authors' knowledge, no studies have yet investigated the use of heat maps for transportation research in general and travel demand analysis in specific; this is possibly due to the fact that the analogy between picture classification (the original domain of LRP) and travel demand predictions is not directly obvious.

This paper re-conceptualises the use of LRP-based heat map generation and pioneers its use in a transportation (travel demand analysis) context. In particular, we show that by properly reconceptualising the notion of heat maps, they can provide meaningful explanations for predictions made by ANNs which were trained for predicting travel mode choices. As such, our paper presents a method to help analysts gain trust in ANNs' predictions in transportation contexts. Furthermore, we show that by carefully selecting predictions to analyse, the process of heat map generation can be used to build trust in the trained ANN as a whole. For the empirical part of our study, we use a recently collected Revealed Preference (RP) mode choice data dataset (Hillel, Elshafie, & Ying, 2018).

The remainder of this paper is organised as follows: Section 2 introduces the used methodology and establishes the analogy between travel mode choice modelling and image classification.

Section 3 presents the dataset used for our analysis and discusses the ANN training procedure. Section 4 presents the results. It shows the heat maps created using LRP. Section 5 draws conclusions and shows directions for future research.

2 Methodology

Before delving into the LRP methodology details, it is useful to present notations and establish the analogy between image classification and discrete (travel mode) choice modelling. Numerous concepts in discrete choice modelling have a counterpart, under a different name, in machine learning. For the reader's convenience Table 1 provides a brief 'translation' table.

In discrete choice analysis the choice data consists of a set of observations $S = ((\mathbf{x}_1, \boldsymbol{y}_1), (\mathbf{x}_2, \boldsymbol{y}_2), \dots, (\mathbf{x}_n, \boldsymbol{y}_n), \dots, (\mathbf{x}_N, \boldsymbol{y}_N))$. Each n^{th} observation s_n contains a vector of independent variables $\mathbf{x}_n$ that represent the attributes and a *K*-dimensional vector of dependent variables $\boldsymbol{y}_n$ that represents the observed choice (i.e., zeros for the non-chosen alternatives, and one for the chosen alternative); *K* being the size of the choice set. Each vector $\mathbf{x}_n$ consists of *I* independent variables (annotated as x_i). Since choices are mutually exclusive (i.e., only one alternative can be chosen from the choice set), from a machine learning perspective this is considered a classification problem.

In image classification problems each observation contains an array of pixels of the image and a Q-dimensional vector that represents the image label; Q being the size of the fixed set of categories. For simplicity we consider the case of a greyscale image where each pixel takes a single value that represents intensity within some range (e.g., from 0 (black) to 255 (white)).[31] The task of image classification is to assign an input data to one label from the fixed set of categories. In this setting, an analogy between image classification and discrete choice modelling can be drawn, where pixels are equivalent to attributes (e.g. travel time), and intensity corresponds to the attribute value (e.g. 25 minutes). Further, similar to the observed choice set, the image label set of size Q is finite, collectively exhaustive and mutually exclusive.

Table 1. Basic image processing terminology and discrete choice modelling equivalent

Image processing	**Discrete choice modelling**
Pixel	Attribute
Intensity	Value
Label/Class	Alternative
Label set	Choice set

When using ANNs for classification, the so-called softmax function is used at the output layer to convert values (processed and forwarded by hidden layers) into probabilities. The softmax is essentially a logit function, see Appendix 4A for a brief description of the ANN methodology. Similar to discrete choice models, ANNs make predictions up to a probability. Since this study is primarily concerned with explaining predictions of ANNs by uncovering the relevance of each independent variable to a particular prediction, the values processed and forwarded to the output layer (i.e., softmax function inputs) are annotated $f(x)$ and are henceforth referred to as

[31] In other image types (e.g., RGB type), each pixels consist of several channels (e.g., red, green and blue channels).

the *relevance*. Note that the notion of relevance in this context can loosely be conceived as utilities in a discrete choice context.

2.1 Model explainability and trust

Opening the black-box of ANNs has received much attention in a variety of fields (Hall & Gill, 2018). In the literature, several meanings have been attached to the effort of opening an ANN's black-box such as enhancing interpretability, explainability and understandability (Doshi-Velez & Kim, 2017; Lipton, 2016; Rosenfeld & Richardson, 2019). In this study, we focus on explainability, which is defined as the ability of the analyst to inspect the contribution of each input (e.g., attributes or image pixels) for a particular example to produce a prediction (Montavon et al., 2017). By explaining a model prediction, we mean presenting a numerical or visual artefact that provides a qualitative understanding of the relationship between independent variables (e.g., attributes) and the model's prediction (Ribeiro, Singh, & Guestrin, 2016). We consider the ability to explain predictions to be critical to build trust between the analyst and the trained ANN model (see further below).

For an analyst to trust a model prediction and take some actions based on it, it is essential to: 1) understand *why* the model has made this prediction (i.e., prediction explainability, henceforth called the Why part); 2) ensure that is based on 'correct', i.e. intuitive and expected relations (this is called the Domain Knowledge part). Obviously, the latter is domain dependent, and the analyst has the "final say" in this regard. For example, consider a black-box model trained to detect tumours from x-ray images. For a doctor to trust a model's prediction, (s)he needs to understand on what basis or factors (e.g., which part of the x-ray) the model made that prediction (the Why part), and whether these are correct, intuitive and expected (based on the doctor's Domain Knowledge). In the remaining part of Section 2, we focus on the Why part (i.e., we present an approach that enables an analyst to answer the Why question). The Domain Knowledge part will be elaborated as part of our discussion of the results of our empirical analysis in Section 4.

To address the Why part, so-called saliency methods have emerged as a popular tool to highlight which independent variables deemed relevant or important for an ANN prediction (Adebayo et al., 2018; Kittley-Davies et al., 2019; Simonyan et al., 2013). These methods can be broadly classified into two categories: perturbation- and backpropagation-based methods (Shrikumar, Greenside, & Kundaje, 2017).

Perturbation-based methods aim to measure the effect of applying small changes in each input (or removing it) on the predictions (or probabilities) produced by the trained ANN (Zeiler & Fergus, 2013; Zintgraf, Cohen, Adel, & Welling, 2017). The underlying principle of perturbation-based methods is that the input whose change or removal affects the ANN output most is the one that has the most relative importance (Ancona, Ceolini, Öztireli, & Gross, 2017). In applications of ANNs for travel choice behaviour modelling, most efforts to answer the Why part have indeed been devoted to perturbation-based approaches. For example, several studies conducted (or suggested using) perturbation-based methods – mostly called sensitivity analysis by transportation researchers – to measure the importance of independent variables for different types of trained ANNs (Chiang, Zhang, & Zhou, 2006; Golshani, Shabanpour, Mahmoudifard, Derrible, & Mohammadian, 2018; Hagenauer & Helbich, 2017; Hensher & Ton, 2000; Lee, Derrible, & Pereira, 2018).

While the perturbation-based methods are widely used to answer the Why part, several studies have highlighted their drawbacks and explained that they are fundamentally inappropriate for this aim. The first, more practical, drawback is that these methods can be computationally inefficient as each change requires a separate forward propagation for the ANN (Shrikumar et al., 2017). This aspect of computational (in-) efficiency becomes more important as the complexity and number of parameters of ANNs grow (e.g., an early version of convolution neural network consists of over 60 million parameters (Krizhevsky, Sutskever, & Hinton, 2012)). The second, and more fundamental, drawback of perturbation-based methods is that upon close inspection, they do not actually provide an answer to the Why-question that analysts are looking for. Instead, because the process is based on alternation of independent variables' values, perturbation based methods answer a different question, being *which* independent variable needs to be altered to make the example belong more/less to the predicted alternative. In other words, perturbation-based methods measure the susceptibility of the output to changes in the input which might not necessarily coincide with those inputs on which the network based its prediction (Böhle, Eitel, Weygandt, & Ritter, 2019; Montavon et al., 2018; Shrikumar et al., 2017). This is indeed a fundamental limitation when answering the Why question. A visual illustration of this fundamental point is presented by Samek et al. (2017) where an image of rooster is correctly predicted by the model (see Fig. 1). Changing the pixels' values of the yellow flowers (that block part of the rooster) in a specific way would reconstruct the covered part of the rooster, which may result in an increase in the probability of predicting a rooster. As such, the result of this perturbation process may lead the analyst to believe that pixels that constructed the yellow flowers were important to the prediction of rooster (which is certainly not correct).

In contrast to perturbation methods, backpropagation-based methods operate by propagating the relevance (i.e., softmax function input $f(x)$) backwards from the output neuron backward through the hidden layers towards the input layer (see Appendix 4A for an overview of ANN structure)) (Springenberg, Dosovitskiy, Brox, & Riedmiller, 2014). One of the most popular of this type of methods in the computer vision field is LRP (Bach et al., 2015). The LRP method leverages the structure of ANNs and the model parameters (i.e., weights) to determine the negative/positive contribution of each independent variable to a particular prediction. It basically asks, for each node, which of the nodes in the preceding layer contributed to what extend to the value in that node. As such, after being applied to the full network, it identifies the independent variables that were pivotal for the ANN's prediction. Thereby, LRP allows the analyst to understand *why* the model has made a particular prediction, given a set of independent variables (Samek, Montavon, Vedaldi, Hansen, & Müller, 2019). Furthermore, as these methods require a single pass to propagate the relevance from the output to the input layer, they are computationally highly efficient (Böhle et al., 2019). Colloquially put, in contrast to perturbation methods which in essence inspect choice probabilities for other than a particular observation (by changing the input variables and looking at changes in choice probabilities), the LRP method only focuses on the particular observation to be explained, studying which input values were particularly crucial for the ANN to arrive at a prediction in the context of the observation.

Before we delve into the technical details, we would like to make a clear distinction between two types of trust: 1) trusting a particular prediction made by an ANN; and 2) trusting the ANN model as a whole. In its core, the LRP method is developed for the former type, but it is worth noting that the method can be also used for the latter type of trust by applying the method to many carefully selected observations (Ribeiro et al., 2016). In this study, we show how to use the method to gain trust regarding multiple ANN predictions (to build trust in each of those predictions). Then, we show a case of how trusting multiple systematically selected ANN predictions can lead to increased levels of trust in the model as a whole (see Section 4).

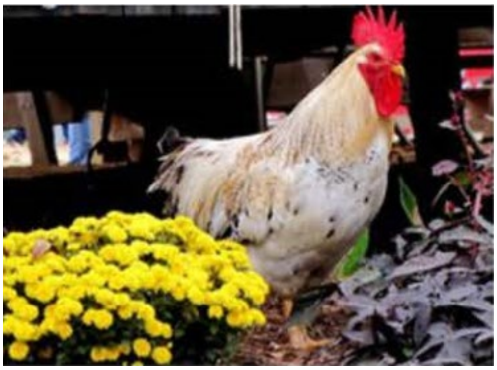


Figure 1. Rooster image (Samek et al., 2017) (For interpretation of the references to colour in this figure legend, the reader is referred to the online version of this article.)

2.2 Layer-wise Relevance Propagation method

LRP operates by propagating the activation strength of the node of interest backward, through hidden layers, to the input layer. In this study, we limit our focus on understanding the ANN prediction; hence, we are mainly concerned with back propagating the activation at the *output* nodes backwards through the hidden layers, using local propagation rules, until it allocates a relevance score R_i to each *input* variable x_i (Samek et al., 2017). Each R_i can be interpreted as the contribution an input x_i has made to a prediction (see Fig. 2). Crucially, each output node can have its own LRP-process; for example, in a travel mode choice context, the ANN assigns a probability to each mode, representing the probability, for a particular case, that the traveller chooses, for instance, the bus, train, or car. LRP can then be used for each of these probabilities, what factors were relevant for that prediction. In other words, LRP can be used to explain the choice probability, predicted by the ANN, for the bus mode, and likewise, for the train and car mode. However, in most cases the LRP method is applied to explain the highest choice probability assigned by the ANN; that is, the method explains why the ANN predicts that a particular mode has a higher probability than the others, of being chosen. In our paper, we use LRP in both ways, and we will clearly indicate when the method is used in which way.

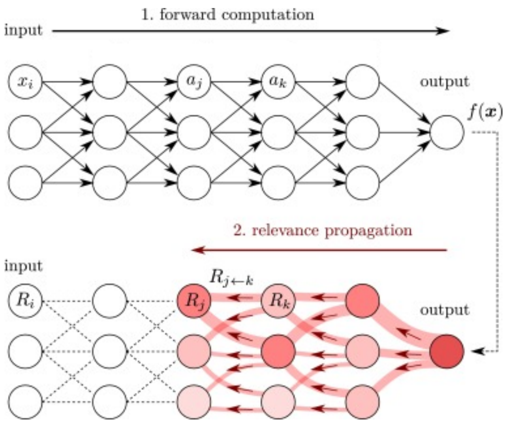


Figure 2. Diagram of the LRP procedure (Montavon et al., 2018) (For interpretation of the references to colour in this figure legend, the reader is referred to the online version of this article.)

The key property of the relevance redistribution process used in LRP is that the total relevance at every layer of the ANN (from the output layer to the input) needs to be maintained; this property is known as relevance conservation and can be described as follows:

$$\sum_i R_i = \cdots = \sum_j R_j = \sum_k R_k = \cdots = f(\mathbf{x}) \qquad (1)$$

where i, j and k are the indices for nodes on the layers, and R_k is the relevance of node k for the relevance $f(\mathbf{x})$. This equation highlights that the method computes the decomposition of $f(\mathbf{x})$ (most right) in terms of the input variables (most left). To ensure Equation (1) holds, two rules need to be imposed:

$$\sum_j R_{j\leftarrow k} = R_k \qquad (2)$$

$$R_j = \sum_k R_{j\leftarrow k} \qquad (3)$$

where $R_{j\leftarrow k}$ is defined as the share of R_k that is redistributed to node j in the lower layer (see Fig. 2). The redistribution of the relevance resembles the process of forward propagation (used to produce predictions). In forward propagation, the activation function $z(.)$ of the node k generates one output a_k that is fanned out to other neurons and can be described as follows (see Appendix 4A for comprehensive description of ANN structure):

$$a_k = z\left(\sum_j w_{jk}a_j + w_k\right) \qquad (4)$$

Where w_{jk}, w_k are the weight and bias parameter of the neuron. The main principle used by LRP to back propagate the relevance is that what has been received by a node should be redistributed to the nodes at the lower layer proportionally. In the literature, different ways in which relevance is back propagated have been proposed. Empirical studies have shown that some of these rules yield better relevance redistribution depending on many factors such as the used activation function and position of the hidden layer (i.e., the layer deepness). In this study, we use the ϵ-rule (as described in (Samek et al., 2019)), which back propagates the relevance to each neuron as follows:

$$R_j = \sum_k \left(\frac{w_{jk} * a_j}{\sum_j \left(w_{jk} * a_j\right) + \epsilon}\right) R_k \tag{5}$$

where ϵ is a fixed constant of small value ($\epsilon = 10^{-7}$) which is added to the denominator to prevent division by zero (not to be confused with the error in discrete choice models). Doing so avoids the relevance values to become too large. This equation shows that the relevance is propagated proportionally depending on: 1) the neuron activation a_j (i.e., more activated neurons receive larger share of relevance), and 2) the strength of the connection w_{jk} (more relevance flows through more strong connection). In this study, we focus only on rule shown in Equation (5), and for more detailed description of LRP and comprehensive discussion on alternative relevance redistribution rules, interested readers are referred to Samek et al. (2019) and Lapuschkin, Binder, Montavon, Müller, and Samek (2016).

2.3 Explaining a prediction using heat map – a computer vision illustration

To further clarify the method, in this subsection we provide a brief illustration of how the LRP method is commonly used in the computer vision field. This particular example is taken from Lapuschkin, Binder, Montavon, Muller, and Samek (2016) whose aim is to explain the predictions of two different machine learning models (these models themselves are not of interest to us in this paper and are not discussed in any detail here). Each of these models is trained using large number of images to discriminate between several output classes, including a horse class. A horse image is presented to the two models, see the left-hand side plot in Fig. 3. Both models produced the correct prediction with high confidence. Then, the prediction is propagated backward using the above-described explanation method (i.e., LRP) to provide an answer to the Why-question (why did the ANN believe that this is a picture of a horse). The analyst can then use the outcome of the LRP process to verify whether the model predictions are based on intuitive and expected rationales (the Domain Knowledge question). To facilitate inspection, the relevance is usually presented as a heat map, where pixels with high positive relevance are shown in red (see colour map on the right side of Fig. 3).

The middle and right-hand side plots in Fig. 3 show the heat maps generated using the LRP method, given the input: the horse image on the left-hand side. Although the predictions produced by both models are correct, the heat maps reveal that the models have a different rationale. For a horse image, we expect (as human analysts with some domain knowledge) a well-trained model to base its prediction on relevant features and distinguishable characteristics of horses, such as e.g. the horse tail. Fig. 3 shows that Model A indeed assigns a high relevance to such horse pixels, while Model B assigns a high relevance to the lower left-hand side corner of the image, where the copyright tag is located. Hence, the heat map reveals that the prediction of model B is largely based on the existence and nature of the copyright tag, rather than the part of the image where distinguishable characteristics of horse are shown. The source of this outcome is that in the training data many horse images were present with the same copyright tag. As a result, Model B has learned that the copyright tag is a good explanatory ‘variable’. This is a clear example of the fact that machine learning methods excel in detecting patterns, regardless of whether these patterns are meaningful, or not (Abu-Mostafa, Magdon-Ismail, & Lin, 2012). Most importantly for the purpose of this paper, this example illustrates that LRP can be used to inspect the model rationale and examine its trustworthiness, using human domain knowledge. In the following, we illustrate how the LRP can be recast and implemented for non-visual contexts, specifically discrete choice analysis (see the analogy between image classification and discrete choice modelling established earlier in this section).

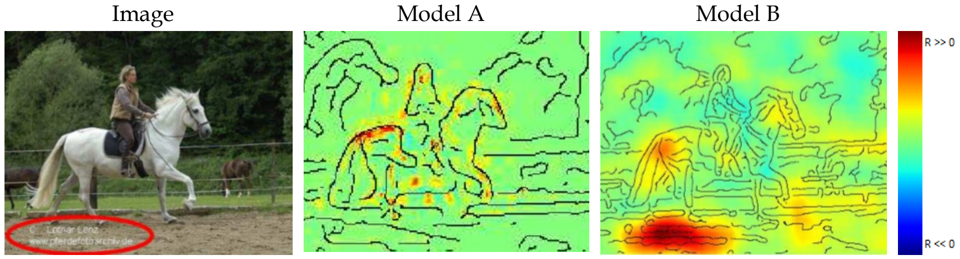


Figure 3. Left: Image of the horse class, presented to two different models. Middle and right: relevance of each pixel is drawn as heat map. (For interpretation of the references to colour in this figure legend, the reader is referred to the online version of this article.)

2.4 Explaining a prediction in travellers' discrete choice context – A re-conceptualisation using Monte Carlo experiments

This subsection conducts a series of Monte Carlo experiments to get a feeling for how heat maps can be re-conceptualised and used in the context of discrete choice data. Table 2 shows the parametrisations of the three synthetic data sets that we generated. Each data set consists of three alternatives and two generic attributes: X_1 and X_2. Parameters have different values across data sets (we use negative, positive and neutral parameter values). Each data set consists of 10,000 hypothetical respondents, each making a single choice. Attribute levels are generated using a random number generator between zero and one. To create the synthetic choices, the total utility of each alternative is computed and the highest utility alternative is assumed to be chosen, following a Logit (RUM-MNL) model where the random part of utility is distributed Extreme Value type I with variance $\pi^2/6$.

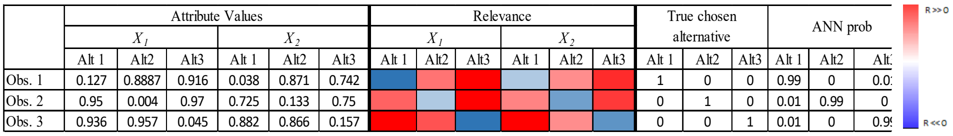
Table 2. Synthetic data specification and parametrisation

Dataset no.	Model specification	Parametrisation	Cross-entropy (RUM-MNL)	ρ^2 (RUM-MNL)	Cross-entropy (ANN)	ρ^2 (ANN)
A1	$V_{un} = \sum_m \beta_m x_{umn}$	$\beta_1 = -6$ $\beta_2 = -4$	-0.53	0.51	-0.54	0.50
A2		$\beta_1 = +6$ $\beta_2 = +4$	-0.53	0.51	-0.54	0.50
A3		$\beta_1 = -6$ $\beta_2 = 0$	-0.59	0.45	-0.61	0.44

For each data set, a three-layers ANN with 4 hidden nodes on the hidden layers is trained. As has also been found in previous studies (e.g., Alwosheel, van Cranenburgh, and Chorus (2018)), the ANNs are able to learn the a RUM-MNL data generating process with high accuracy, in the sense that the prediction performance of the ANN almost matches that of the true underlying data generating process encoded in a corresponding discrete choice model, see Table 2.

For the first data set (A1), the negative sign of the parameters imposes a dislike for higher attribute values (i.e., the lower the attribute values, the more attractive the alternative becomes). Hence, the attribute values of the *chosen* alternative are expected to contribute negatively to the choice probability prediction for that alternative (as reducing the attribute values would increase the attractiveness of the chosen alternative). In contrast, we expect that high attribute values of the *non-chosen* alternatives contribute positively to the prediction, implying that the attractiveness of these non-chosen alternatives increases as these attribute values increase. These expectations are confirmed in Table 3, where we see the relevance of the attribute values that are computed using the LRP method[32], alongside the choice probabilities predicted by the ANN, for three randomly selected observations from the synthetic data. In this Table, we apply the LRP method to explain the choice probability assigned to the chosen alternative – that is, we do not explain choice probabilities assigned to non-chosen alternatives. In the heat map positive relevance values are depicted red; negative relevance values are depicted blue, and neutral relevance values are depicted white. The colour intensity for each observation is normalised to the maximum absolute value.

Table 3. Results of observations randomly selected from A1 data set

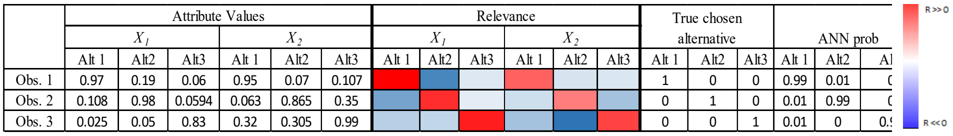


	Attribute Values						Relevance						True chosen alternative			ANN prob		
	X_1			X_2			X_1			X_2								
	Alt 1	Alt2	Alt3	Alt 1	Alt2	Alt3	Alt 1	Alt2	Alt3	Alt 1	Alt2	Alt3	Alt 1	Alt2	Alt3	Alt 1	Alt2	Alt
Obs. 1	0.127	0.8887	0.916	0.038	0.871	0.742	🟦	🟥	🟥	🟦	🟥	🟥	1	0	0	0.99	0	0.0
Obs. 2	0.95	0.004	0.97	0.725	0.133	0.75	🟥	🟦	🟥	🟥	🟦	🟥	0	1	0	0.01	0.99	0
Obs. 3	0.936	0.957	0.045	0.882	0.866	0.157	🟥	🟥	🟦	🟥	🟥	🟦	0	0	1	0.01	0	0.9

Consider the three observations shown in Table 3, where alternative 1 is chosen in the first observation, alternative 2 is chosen in the second observation, and alternative 3 is chosen in the third observation. Note that the ANN predictions are correct with very high confidence as shown by predicted choice probabilities of 0.99 for the chosen alternative in each of the three observations. The blue diagonal values show that the attribute values of the chosen alterative have contributed negatively toward the predicted probability of the alternative being chosen. In contrast, the off-diagonal cells, which here are associated with the non-chosen alternatives, are coloured red. This means that the attribute values of these unattractive alternatives, which are comparatively high, positively contribute to the prediction that alternative 1 is chosen in observation 1, alternative 2 in observation 2, etc. Hence, increasing the attribute values of the non-chosen alternatives would in this situation further increase the probabilities of the chosen alternatives, which is exactly as expected.

Compared to the first data set, in the second data set (A2) the parameters have flipped signs. Hence, lower attribute values are more attractive than higher ones. Table 4 shows the results for three randomly selected observations (again, from the subset of observations that are correctly predicted by the ANN). We use same colour map and intensity as in Table 3. As can be seen, Table 4 reveals the same patterns as shown in Table 3, but colours are flipped, i.e., cells on the diagonal are red, and cells off the diagonal are blue. This is fully in line with expectations, as here an increase (decrease) in the attribute levels of the chosen (non-chosen) alternative positively contributes to the choice probability that is predicted for the chosen alternative.

Table 4. Results of observations randomly selected from A2 data set

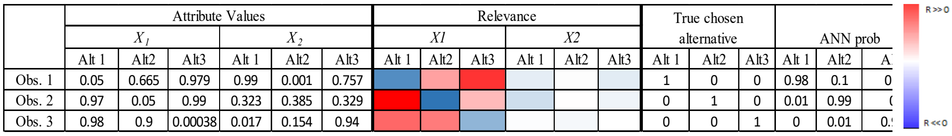


	Attribute Values						Relevance						True chosen alternative			ANN prob		
	X_1			X_2			X_1			X_2								
	Alt 1	Alt2	Alt3	Alt 1	Alt2	Alt3	Alt 1	Alt2	Alt3	Alt 1	Alt2	Alt3	Alt 1	Alt2	Alt3	Alt 1	Alt2	Al
Obs. 1	0.97	0.19	0.06	0.95	0.07	0.107	🟥	🟦	🟦	🟥	🟦	🟦	1	0	0	0.99	0.01	(
Obs. 2	0.108	0.98	0.0594	0.063	0.865	0.35	🟦	🟥	🟦	🟦	🟥	🟦	0	1	0	0.01	0.99	(
Obs. 3	0.025	0.05	0.83	0.32	0.305	0.99	🟦	🟦	🟥	🟦	🟦	🟥	0	0	1	0.01	0	0.

[32] To generate heat maps, the LRP method is used and implemented in the Python environment using the open source library iNNvestigate (Alber et al., 2019).

Lastly, Table 5 presents the results for data set A3. Again, three randomly selected observations from the subset of observations that are correctly assigned by the ANN are shown. In this data set, β_2 is zero. This means that the attribute X_2 does not impact the decision makers' choices. As such, we expect the relevancies for these attribute values to have values that are close to zero. In line with expectation, Table 5 shows that all cells for X_2 are (almost) white – meaning that the values of this attribute do neither positively or negatively contribute to the predicted choice probabilities.

Table 5. Results of observations randomly selected from A3 data set



	Attribute Values						Relevance						True chosen alternative			ANN prob		
	X_1			X_2			$X1$			$X2$								
	Alt 1	Alt2	Alt3	Alt 1	Alt2	Alt3	Alt 1	Alt2	Alt3	Alt 1	Alt2	Alt3	Alt 1	Alt2	Alt3	Alt 1	Alt2	A
Obs. 1	0.05	0.665	0.979	0.99	0.001	0.757	🟦	🟥	🟥	🟦		🟦	1	0	0	0.98	0.1	0
Obs. 2	0.97	0.05	0.99	0.323	0.385	0.329	🟥	🟦	🟥	🟦			0	1	0	0.01	0.99	(
Obs. 3	0.98	0.9	0.00038	0.017	0.154	0.94	🟥	🟥	🟦				0	0	1	0	0.01	0.

R>>0

R<<0

In sum, this application on synthetic data provides a first idea of how the LRP method can be used to inspect the rationale based on which an ANN makes it predictions in a travel mode choice context, and it provides a first sign of face validity of the method. The next sections present an application of the method on a real empirical data set.

3 Empirical data and ANN training

3.1 Data preparation

For this study, we use revealed preference (RP) data from a study conducted for travel mode choice analysis in London city (Hillel, Elshafie, & Jin, 2018).[33] This dataset contains of four alternatives, and a total of 27 features (i.e., attributes of alternatives and characteristics of decision makers). Three processing steps have been executed to prepare the data for this study: First, features that were considered redundant are removed, or merged with others. For instance, rather than using three features to represent car cost (fuel, congestion, and total cost), we merged them into a single one representing the total car cost. Table 6 shows statistics on the attribute levels in the dataset used for this analysis. Second, we noticed that the dataset is highly imbalanced in terms of the chosen mode: walking (17.6%), cycling (3.0%), public transport (35.3%) and driving (44.2%). Such imbalances could affect the reliability of the trained ANNs (Haykin, 2009). As this paper is concerned with explaining ANN predictions (i.e., we do not aim to find the best ANN to predict the mode choices), we considered dealing with this sort of data imbalances out of scope for this paper. Therefore, the data imbalance is 'repaired' by eliminating the cycling alternative from the dataset. Third, we excluded very short trips (i.e., less than two minutes), as these were deemed not to contain a mode trade-off. The resulting dataset that is used for this study consists of 77,638 mode choice observations.

[33] The dataset and its description are available online, and can be downloaded from the first author profile at researchgate.net

Table 6. Data statistics

No.	Attribute	Description	Type	Range [min, max]	Mean and standard deviation
1	TC_{Drive}	Estimated cost of driving route, including fuel cost and congestion charge	Float (£)	[0.05, 17.16]	(1.91, 3.48)
2	TC_{PubTr}	Estimated cost of public transport route, accounting for rail and bus fare types	Float (£)	[0, 13.49]	(1.56, 1.55)
3	TT_{Drive}	Predicted duration of driving route	Float (minutes)	[0.02, 142]	(17, 15)
4	TT_{PubTr}	Predicted duration of public transportation	Float (minutes)	[0.02, 141]	(28, 19)
5	TT_{Walk}	Predicted total duration of walking times for interchanges on public transport route	Float (minutes)	[0.02, 550]	(69, 68)
6	DIS	Straight line distance between origin and destination	Integer (meters)	[96, 40,941]	(4,690, 4,827)
7	$TRAF$	Predicted traffic variability on driving route	Float	[0, 1]	(0.34, 0.20)
8	$INTER$	Number of interchanges on public transport route from directions API	Integer	[0, 4]	(0.38, 0.62)
9	DL	Boolean identifier of a person making trip: 1 if person has driving license, 0 otherwise	Bool	[0, 1]	(0.62, 0.49)
10	CO	Car ownership of household person belongs to: no cars in household (0), less than one car per adult (1), one or more cars per adult (2)	Integer	[0, 2]	(0.99, 0.75)
11	FEM	Boolean identifier of a person making trip: 1 if female, 0 otherwise	Bool	[0, 1]	(0.53, 0.49)
12	AG	Age of person making trip	Integer (years)	[5, 99]	(39.5, 19.3)

3.2 ANN development and training

The ANN is implemented in a Python environment, using the open source deep learning library Keras (Chollet, 2015). To train the ANN, the built-in training algorithm (which is used to update weights' values **w**) known as Adam is used (Kingma & Ba, 2014). Prior to training the ANN, the data are normalised to reduce training time and minimise the likelihood of ending-up with suboptimal local solutions.[34] A conventional three layers (input, output and one hidden layer consist of ten nodes, see Appendix 4A for a similar ANN layout) fully connected ANN structure is used. Unlike the traditional three-layers ANN, we have removed the bias nodes in the hidden and output layer to avoid losing fraction of the relevance values.[35] Note that the bias nodes

[34] Data normalisation is a common practice for ANN training. In this study, the minimum and maximum values of data are normalised to 0 to +1.

[35] ANN complexity is adjusted using a cross validation approach (see (Alwosheel et al., 2018) for more details). To avoid overfitting, a commonly used rule-of-thumb in ANNs is that the sample size needs to be (at least) 10 times larger than the number of adjustable parameters in the network (Haykin, 2009). A recent study specifically dealing with sample size requirements for using ANNs in the context of choice

removal has not impacted the prediction performance of the trained ANN. To train the ANN and test its performance to predict the travel mode choice, we conducted a so-called *k*-fold cross-validation method, with $k = 5$. The data set is randomly partitioned into five equally sized folds of (roughly) 15,528 observations. Then, a single fold is used for testing, while the remaining four folds are used for training. This process is repeated 5 times, where each of the five folds is used only once for testing.

Table 7 shows several performance metrics for the trained ANN. The reported performance metrics are averaged across the five hold-out folds. It shows that ANN achieves a satisfactory prediction performance. For comparison, we also report the performance of a standard linear-additive RUM-MNL model (see Appendix 4B for the model specifications).[36] As expected based on previous literature, the trained ANN outperforms the discrete choice model by a large margin. Table 8 shows the *k*-folds confusion matrix for the trained ANN. To construct the confusion matrix each observation is assigned to an alternative based on the highest probability as predicted by the ANN. Then, each prediction is compared to the true chosen alternative. The cells on the diagonal show the mean percentage of the observations that are correctly assigned, across the 5 folds. Additionally, the values between parentheses show the average ANN probabilities of the observations that are correctly classified. The off-diagonal cells show the mean percentage of observations that are misclassified, across the 5 folds. Values between parentheses show the average ANN probabilities of these observations.

Table 7. Performance of the trained ANN

Performance metric	Function	Null model	ANN	Linear-additive RUM
Final Log-likelihood	$\sum_{n=1}^{N}\sum_{k=1}^{K} y_{nk}\ln(P_{nk})$	-86,625	-43,477	-50,704
Cross-entropy	$\frac{1}{N}\sum_{n=1}^{N}\sum_{k=1}^{K} y_{nk}\ln(P_{nk})$	-1.09	-0.56	-0.65
ρ^2	$\rho = 1 - \frac{LL(\hat{\beta})}{LL(0)}$	0	0.50	0.42

models is more conservative, and recommends to use a sample size of (at least) 50 times the number of estimable weights (Alwosheel et al., 2018). Our sample size satisfies this condition and, therefore, we safely avoid overfitting issues.

[36] Note that the *k*-fold method is not used for the RUM model. Rather, the RUM model is estimated one time using the whole dataset.

Table 8. Confusion matrix

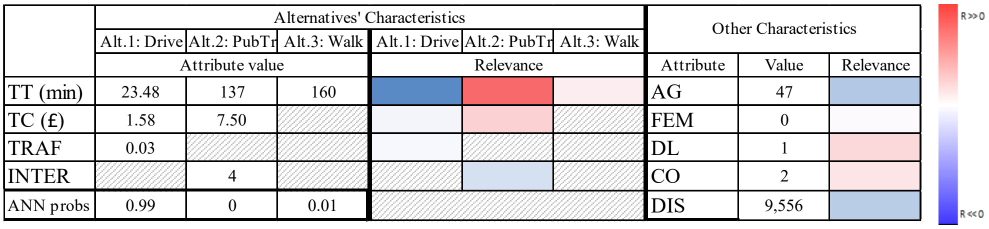
		ANN Classification			
		Driving	Public Transport	Walking	∑
True chosen alternative	Driving	82.55 (0.69)	11.23 (0.19)	6.22 (0.10)	100% (1)
	Public Transport	21.22 (0.25)	72.66 (0.66)	6.12 (0.09)	100% (1)
	Walking	23.72 (0.25)	11.56 (0.17)	64.72 (0.57)	100% (1)

4 Applying the LRP method

4.1 ANN prediction explanation of randomly selected observations

In this subsection, we use the LRP-generated heat maps for multiple observations with the aim to understand *why* the ANN produces certain mode choice predictions. Tables 9 to 11 show the back-propagated relevance extracted for three observations randomly selected from the subset of observations that are correctly assigned by the ANN. It can be seen, that predictions are made with different levels of confidence. In the context of our analyses, a high confidence level means the network assigns a choice probability of more than 0.80 to one the modes, and a low confidence level means that the highest (across travel mode alternatives in the context of a particular observation) predicted choice probability is still below 0.40. As such, for diversification purposes and to build trust in the model as a whole, the three observations are randomly selected as follows: two predictions with high confidence levels and one prediction with low confidence level. Tables 9 to 11 show the ANN probabilities, the attributes' values, and relevancies obtained using LRP for the selected observations. A heat map (using same colour map as in section 2.4 i.e., positive, negative and neutral values are depicted in red, blue and white, respectively) is employed to visualise the relevancies. As in the Monte Carlo analysis, we apply the LRP to the output node with the highest (choice) probability.

Table 9. Results of observation 1 (index: 47,489 and true chosen alternative is Drive)



	Alternatives' Characteristics						Other Characteristics		
	Alt.1: Drive	Alt.2: PubTr	Alt.3: Walk	Alt.1: Drive	Alt.2: PubTr	Alt.3: Walk			
	Attribute value			Relevance			Attribute	Value	Relevance
TT (min)	23.48	137	160	🟦	🟥	🟥	AG	47	🟦
TC (£)	1.58	7.50	▨	⬜	🟥	▨	FEM	0	⬜
TRAF	0.03	▨	▨	⬜	▨	▨	DL	1	🟥
INTER	▨	4	▨	▨	🟦	▨	CO	2	🟥
ANN probs	0.99	0	0.01	▨			DIS	9,556	🟦

Table 10. Results of observation 2 (index: 24,618 and true chosen alternative is Walk)

	Alternatives' Characteristics						Other Characteristics		
	Alt.1: Drive	Alt.2: PubTr	Alt.3: Walk	Alt.1: Drive	Alt.2: PubTr	Alt.3: Walk			
	Attribute value			Relevance			Attribute	Value	Relevance
TT (min)	7.38	4	6	🟥	🟥	🟦	AG	26	
TC (£)	10.70	2.40	▨	🟥	🟥	▨	FEM	1	
TRAF	0.31	▨	▨	🟥	▨	▨	DL	1	
INTER	▨	0	▨	▨		▨	CO	0	
ANN probs	0	0.01	0.99	▨			DIS	368	🟦

Table 9 shows an observation of a middle-aged female, who holds a driver license and owns two cars, who chose the driving alternative, which indeed seems to be the most attractive travel alternative in this case as is the fasted and cheapest alternative. In line with intuition, the ANN predicts a choice for the driving alternative with a very high level of confidence (assigning a 0.99 choice probability to that alternative). The relevance values show that car travel time receives a strong negative relevance, as expected (given that lower travel times are preferred). The relatively long travel times offered by the non-chosen alternatives receive a strong positive relevance, as expected (given that the high travel times of these alternatives makes driving alternative more appealing). Furthermore, the number of owned cars (two) and the driving license availability have a positive relevance. Together, these analyses reveal on what basis the ANN model has predicted that this traveller would choose the driving alternative. From a travel behaviour perspective, all these points are in line with expectations. As such, the analyst equipped with the proper domain knowledge can safely trust this prediction.

Moving forward to Table 10, for this observation, a young female traveller chose the walking alternative. As before, the travel time and cost of the non-chosen alternatives and the relatively high traffic on the driving route have high positive relevance for the predicted choice probability for walking. Furthermore, we see that the travel time of the chosen alternative, and the travelled distance have negative relevance values. All these relations are expected from a travel behaviour perspective; hence, this prediction too can be safely trusted. Lastly, in the Table 11 the alternative with highest predicted probability is walking; however, this mode receives a predicted probability which is only one percentage point higher than that of the other mode, implying that the ANN has low confidence in this prediction. The relevance values show that attribute values with negative relevance for the predicted choice probability for the walking alternative, are the relatively long distance and walking travel time, suggesting that shorter distance and less walking time would have made the walking alternative more attractive. This is expected from a travel behaviour perspective. Further, it can be noticed that the red and blue colours associated in this heat map are less bright, meaning that the ANN is less outspoken about what determined its prediction; this too is to be expected, given that the ANN assigns almost equal choice probabilities to each of the three alternatives.

Table 11. Results of observation 3 (index: 1,621 and true chosen alternative is Walk)

	Alternatives' Characteristics						Other Characteristics		
	Alt.1: Drive	Alt.2: PubTr	Alt.3: Walk	Alt.1: Drive	Alt.2: PubTr	Alt.3: Walk			
	Attribute value			Relevance			Attribute	Value	Relevance
TT (min)	7.23	22	24	🟥		🟦	AG	15	🟥
TC (£)	0.30	0.00	▨			▨	FEM	1	🟥
TRAF	0.44	▨	▨	🟥	▨	▨	DL	0	
INTER	▨	0	▨	▨		▨	CO	1	
ANN probs	0.33	0.33	0.34	▨			DIS	1,271	🟦

4.2 Using RUM-MNL and ANN models to guide observation selection

In this subsection, we use RUM-MNL (a highly robust and well understood model for mode choice analysis) and ANN predictions jointly to guide the process of observations selection, instead of relying on ANN predictions only, as in the previous subsection. This allows us to examine the overall workings and rationale of the trained ANN and decide whether we can trust a trained ANN as a whole. Furthermore, we in this section analyse correct as well as incorrect predictions. Note that while explaining correct predictions made by the black-box is important (as shown in previous subsection), it could even be more important to inspect why an ANN makes wrong predictions. This helps to obtain a higher level understanding on the model. Specifically, we are interested to learn whether, or not, these wrong predictions are based on meaningful intuition, or on counter intuitive or flawed relations learned by the ANN. It goes without saying that a mis-prediction by an trained ANN does not necessarily mean it has learned counterintuitive or flawed relations. But, if the ANN has learned such relations they are particularly to show up in mis-predictions.

To select a diverse set of observations (including observations where the trained ANN makes a wrong prediction), we distinguish between three cases, see Table 12. For each case, we randomly sample one or a few observations to analyse using LRP-generated heat maps.

- Case I: The ANN model predicts the true chosen alternative, while the RUM-MNL model makes the wrong prediction. For this case, we randomly select two observations: one for ANN prediction with high probability, and the other for ANN prediction with low probability (see Tables 13 &14).
- Case II: The RUM-MNL model predicts the true chosen alternative, while the ANN misses it. For this case, we randomly select one observation (see Table 15). As explained in 2.2, relevance back-propagation using LRP method can be implemented for any node at the network. In addition to having the relevance for the predicted alternative, we for this case also compute the relevance for the true chosen alternative (that the ANN misses), which allows for additional examination of the black-box rationale.
- Case III: Both the ANN and the RUM-MNL model mispredicts the correct alternative. Under this category, there are two subcases: ANN and RUM model agree (e.g., an observation where both models predict Driving alternative, and the true chosen alternative is Walking), or ANN and RUM disagree (e.g., an observation where the true chosen alternative is Walking, ANN prediction is Public Transport, RUM prediction is Driving alternative). One observation is selected for each subcase. See Tables 16 &17 for the selected observations details.

Table 12. Cases developed using ANN and RUM-MNL prediction. Values between parenthesis are the total number of observations for each case

		RUM-MNL prediction	
		Correct	Incorrect
ANN prediction	Correct		Case I (6,679)
	Incorrect	Case II (3,588)	Case III (14,792)

Table 13. Results of first observation in Case I (index: 7,011 and true chosen alternative is Walking)

	Alternatives' Characteristics						Other Characteristics		
	Alt.1: Drive	Alt.2: PubTr	Alt.3: Walk	Alt.1: Drive	Alt.2: PubTr	Alt.3: Walk			
	Attribute value			Relevance			Attribute	Value	Relevance
TT (min)	5.32	5	7	🟥		🟦	AG	18	
TC (£)	0.22	1.50	▨			▨	FEM	1	🟥
TRAF	0.02	▨	▨		▨	▨	DL	0	
INTER	▨	0	▨	▨		▨	CO	2	
ANN probs	0.06	0.01	0.93	▨	▨	▨	DIS	403	🟦
RUM probs	0.47	0.14	0.39	▨	▨	▨	▨	▨	▨

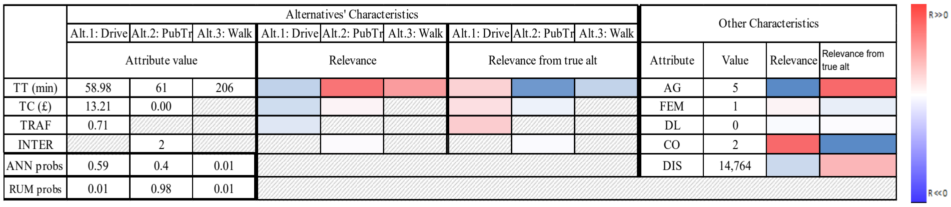
Table 14. Results of second observation in Case I (index: 38,475 and true chosen alternative is Driving)

	Alternatives' Characteristics						Other Characteristics		
	Alt.1: Drive	Alt.2: PubTr	Alt.3: Walk	Alt.1: Drive	Alt.2: PubTr	Alt.3: Walk			
	Attribute value			Relevance			Attribute	Value	Relevance
TT (min)	3.92	9	22	🟦	🟥	🟥	AG	18	🟦
TC (£)	0.21	0.00	▨			▨	FEM	0	
TRAF	0.14	▨	▨		▨	▨	DL	0	
INTER	▨	0	▨	▨		▨	CO	1	
ANN probs	0.35	0.33	0.32	▨	▨	▨	DIS	1,158	🟥
RUM probs	0.3	0.43	0.29	▨	▨	▨	▨	▨	▨

Tables 13 &14 show (for Case I observations) the attribute values, the relevancies produced using LRP method, ANN and RUM-MNL probabilities. Table 13 shows a young female, who owns two cars and chose the walking alternative. For the chosen alternative, the attribute values with negative relevance are the distance and walking travel time. Further, the travel time values of the non-chosen alternatives have resulted in positive relevance values. Both of these points are in line with expectations and have been also noted from the observation shown in Table 11 (where walking alternative was also predicted). For the observation shown in Table 14, the relatively long distance, and longer travel time offered by public transport and walking alternatives have a positive relevance for the prediction of driving, as we expect. Further, also in line with expectations, travel time and cost of the chosen alternative have a negative relevance to the prediction.

It is possible to end up with relevance values that are unexpected or hard to rationalise. For instance, we expect owning a car to have a positive relevance for the choice probability predicted for the driving alternative, but that is not always the case, as the observation shown in Table 14 reveals. Also, the unavailability of driving license only has a negligible contribution to the driving prediction. It should be kept in mind here that, since the ANN itself is a probabilistic technique that is not expected to fit complex data perfectly (Abu-Mostafa et al., 2012), we should not expect the relevancies values that are produced by LRP to always provide a fully accurate description of the contribution of all independent variables on every observation. As such, we advise the analyst to tolerate having some unexpected relevancies' values, but of course (s)he has the 'final say' on deciding to what extent these unexpected relevancies are tolerable or not – leading to a rejection of the trained network in the latter case. For this particular prediction (shown in Table 14), we believe having two unexplainable values (out of 11) is acceptable, given that all other values are in line with expectations based on domain knowledge.[37]

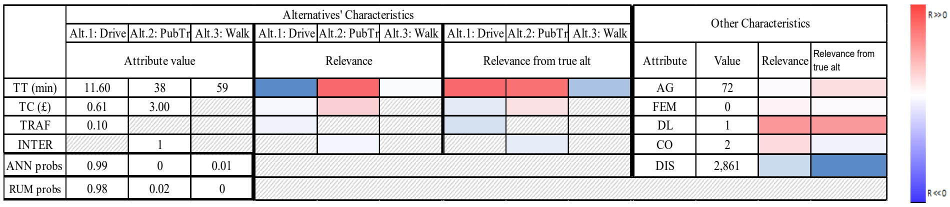
Table 15. Results of Case II observation (index: 32,923 and true chosen alternative is Public Transport)



	Alternatives' Characteristics									Other Characteristics			
	Alt.1: Drive	Alt.2: PubTr	Alt.3: Walk	Alt.1: Drive	Alt.2: PubTr	Alt.3: Walk	Alt.1: Drive	Alt.2: PubTr	Alt.3: Walk				
	Attribute value			Relevance			Relevance from true alt			Attribute	Value	Relevance	Relevance from true alt
TT (min)	58.98	61	206	🟦	🟥	🟥	🟥	🟦	🟦	AG	5	🟦	🟥
TC (£)	13.21	0.00	▨	🟦	🟥	▨	🟥	🟦	▨	FEM	1	🟥	🟦
TRAF	0.71	▨	▨	🟦	▨	▨	🟥	▨	▨	DL	0		
INTER	▨	2	▨	▨		▨	▨		▨	CO	2	🟥	🟦
ANN probs	0.59	0.4	0.01	▨						DIS	14,764	🟦	🟥
RUM probs	0.01	0.98	0.01	▨									

Two back-propagated relevancies are extracted for Case II observation, where RUM-MNL model predicts the true chosen alternative (public transport), and the ANN incorrectly predicts that the driving alternative has the highest choice probability. For a deeper examination of the trained ANN rationale, Table 15 shows the relevance values back-propagated from two output nodes: one from the public transport alternative (the true chosen alternative), and another from the driving alternative (which was predicted by ANN to have the highest choice probability). By doing so, we can obtain a higher level of trust in the model, as we are now able to inspect the model reasoning in the case of "mis-prediction" (i.e., we may come to understand what has led the model to mis-predict, and whether this is still based on an intuitive and defensible rationale). For instance, the relevancies of driving alternative shows that relatively high travel times of the other two alternatives have led to the driving prediction, which is to be expected. Further, the high travel cost and time of the predicted alternative have negative contributions, which is also in line with domain knowledge. Inspecting the relevance extracted from the true chosen alternative (public transport), we observe that the long travel time and the high number of owned cars have contributed negatively to the choice probability assigned to the public transport alternative, highlighting that the probability of choosing public transport would have been higher if these values are lowered, which is as expected.

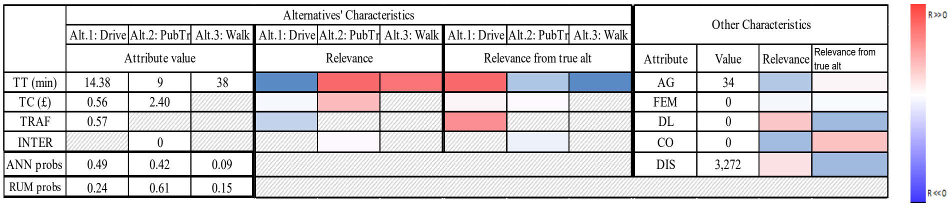
Table 16. Results of Case III first observation (index: 48,999 and true chosen alternative is Walking)



	Alternatives' Characteristics									Other Characteristics			
	Alt.1: Drive	Alt.2: PubTr	Alt.3: Walk	Alt.1: Drive	Alt.2: PubTr	Alt.3: Walk	Alt.1: Drive	Alt.2: PubTr	Alt.3: Walk				
	Attribute value			Relevance			Relevance from true alt			Attribute	Value	Relevance	Relevance from true alt
TT (min)	11.60	38	59	🟦	🟥		🟥	🟥	🟦	AG	72		🟥
TC (£)	0.61	3.00	▨	🟦	🟥	▨	🟦	🟥	▨	FEM	0	🟥	🟥
TRAF	0.10	▨	▨	🟦	▨	▨	🟦	▨	▨	DL	1	🟥	🟥
INTER	▨	1	▨	▨	🟦	▨	▨	🟦	▨	CO	2	🟥	🟦
ANN probs	0.99	0	0.01	▨						DIS	2,861	🟦	🟦
RUM probs	0.98	0.02	0	▨									

[37] A similar pattern can be also noticed in the computer vision example (presented in subsection 2.3). For instance, we can identify small red spots on non-horse pixels (i.e., pixels showing the background, see Fig. 3).

Table 17. Results of Case III second observation (index: 46,900 and true chosen alternative is Walking)



	Alternatives' Characteristics									Other Characteristics			
	Alt.1: Drive	Alt.2: PubTr	Alt.3: Walk	Alt.1: Drive	Alt.2: PubTr	Alt.3: Walk	Alt.1: Drive	Alt.2: PubTr	Alt.3: Walk				
	Attribute value			Relevance			Relevance from true alt			Attribute	Value	Relevance	Relevance from true alt
TT (min)	14.38	9	38	🟦	🟥	🟥	🟥	🟦	🟦	AG	34	🟦	⬜
TC (£)	0.56	2.40	▨	⬜	🟥	▨	⬜	⬜	▨	FEM	0	⬜	⬜
TRAF	0.57	▨	▨	🟦	▨	▨	🟥	▨	▨	DL	0	🟥	🟦
INTER	▨	0	▨	▨	⬜	▨	▨	🟦	▨	CO	0	🟦	🟥
ANN probs	0.49	0.42	0.09	▨						DIS	3,272	🟥	🟦
RUM probs	0.24	0.61	0.15	▨									

Finally, two back-propagated relevancies are presented for the two scenarios of Case III (Table 16 shows when ANN and RUM models agree, and Table 17 when the two models disagree). In Table 16 (ANN and RUM have predicted driving alternative with high probability), the chosen alternative is walking, despite that the travelled distance is relatively long (around 3km). This indicates that in this case, the actual choice for the walking alternative deviates from expectations regarding the length of the average walking trip. The produced relevancies for the ANN are actually as expected. For example, the relevance computed for from the actually chosen alternative (walking) shows that walking travel time and distance have the highest negative relevance, indicating that the walking probability would have been higher if lower walking travel time and distance were lower – this makes sense. A similar point can be also noted for relevancies shown in Table 17 (when RUM and ANN disagree). The true chosen alternative is walking, despite the relatively long travel time and distance. The back-propagated relevancies of these two attributes from the walking node are negative, explaining the reasons for this mis-prediction which turn out to be in line with common sense and domain knowledge.

5 Conclusions and recommendations

This study re-conceptualises the use of heat maps, generated using a Layer-wise Relevance Propagation method, to explain predictions of Artificial Neural Networks in the context of travel demand analysis. We show how heat maps can be applied to provide explanation for the predictions of a trained ANN, thereby helping an analyst to build trust in predictions made by ANNs. Furthermore, we show that by carefully selecting a set of observations, this method can ultimately help building trust in a trained ANN as a whole (or not, in which the ANN can be retrained or adapted).

We would like to point out several limitations to this study, providing potential directions for future research. Firstly, to generate heat maps, this study implemented the widely used ϵ-LRP rule. Several alternative variations to this rule have been proposed in the literature, and some are found to provide better outcomes in specific domains (e.g., natural language processing). Investigating the performance of these alternative rules in the context of transport applications is a fruitful direction for further research. Possibly, this could lead to the discovery of new rules that particularly work well for transportations contexts. Secondly, additional to explaining ANN predictions, it is possible to use the LRP technique in hidden nodes to reveal what concepts and features have been learned by the trained ANN (several researches (e.g., (Olah et al., 2018)) have investigated this in computer vision context). We believe doing so in travel choice behaviour context may provide a deeper understanding of the workings of ANNs and perhaps of the decision-making processes of travellers. Thirdly, the empirical analyses provided in this paper are based on a single empirical data set and a Monte Carlo analysis on synthetic data. It is advisable to repeat these analyses on more data sets with different characteristics (e.g., more and different attributes and alternatives). This will provide a richer view on the generalisability of the proposed method to explain ANNs' predictions. Lastly, to build trust in the ANN model as a whole, predictions of RUM-MNL and ANN models are used to guide the observation selection process. Although we believe this process is very helpful to select diverse observations, it might be rewarding to develop alternative selection strategies that may result in a better representation of the data set.

To conclude, while our analysis suggests that the proposed LRP-based heat map methodology provides a valuable tool to understand the rationale behind ANN predictions in the context of travel choice behaviour, it is important to acknowledge that the proposed method does not completely solve the ANNs' black-box puzzle as it will never completely explain the inner workings of the network. As such, in our view and despite ongoing advances in explainable ANNs, their most natural domain of use in transportation still remains forecasting applications, where complete model transparency is not a prerequisite.

Appendix 4A. Artificial Neural Networks – An overview

ANNs consist of highly interconnected processing elements, called neurons, which communicate together to perform a learning task, such as classification, based on a set of observations. Fig. A1 shows the layout of the neuron structure.

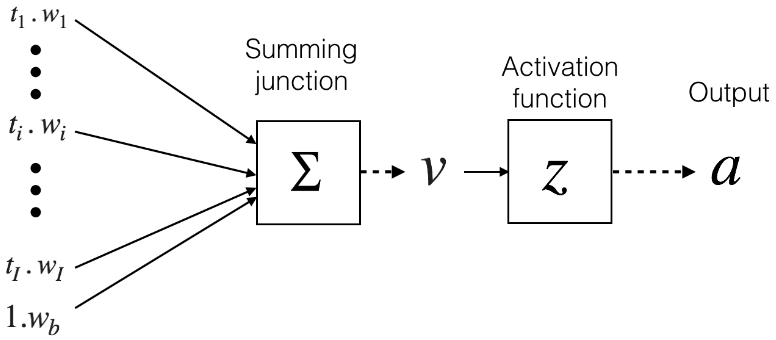


Figure 4A.1. A neuron layout.

Each neuron in the network receives inputs (t_m) multiplied by estimable parameters known as weights (w_m). The weighted inputs are accumulated and added to a constant (called bias, denoted b) to form a single input v for a pre-defined processing function known as activation function $z(.)$. The bias has the effect of increasing or decreasing the net input of the activation function by a constant value, which increases the ANNs flexibility (Haykin, 2009). The activation function $z(.)$ generates one output a that is fanned out to other neurons. The output a can be described as follows:

$a = z(v) = z(\sum_{m=1}^{M} w_m * t_m + w_b)$, where w_b is the weight associated with the bias.

The neurons are connected together to form a network (Bishop, 2006; LeCun, Bengio, & Hinton, 2015). A widely used ANN structure consists of layers of neurons connected successively, known as multi-layer perceptron (MLP) structure. Typically, the first (input) layer and the output layer depend on the problem at hand. More specifically, input layer neurons represent the independent variables. In the context of choice modelling, these are the alternatives' attributes, characteristics of decision-makers, and contextual factors. The output layer, in a discrete choice context, consists of neurons that provide choice probabilities P for each alternative. Layers in-between are called hidden layers because their inputs and outputs are connected to other neurons and are therefore 'invisible' to the analyst. For illustrative purposes, consider the following hypothetical situation: a person can travel using one of three modes: bus, train, or car; two attributes (travel cost "TC" and travel time "TT") are associated with each alternative. Fig. A2 shows this typical choice situation in a three-layer MLP network with four hidden neurons.

Neurons at the hidden and output layers are represented by circles in Fig. A2, while input and bias neurons are represented by squares. This is to emphasise that the neurons at the hidden and output layers are processing units, meaning that they receive inputs t and return outputs a according to predefined activation function $z(.)$, as illustrated in Fig. A1. Input neurons pass the input signals to the next layer. In Fig. A2, the ANN has a total of 7 processing units.

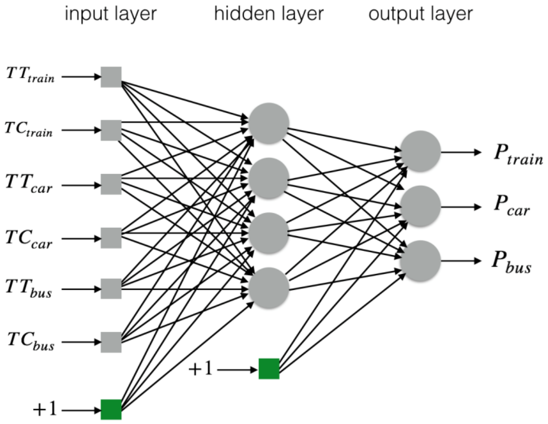


Figure 4A.2. Three-layers Artificial Neural Network.

For a complete MLP structure, three elements need to be defined:

1) Number of hidden layers: a commonly used structure is three-layers MLP: input, output and one hidden layer. A key property of this structure lies in the ability to approximate, with arbitrary level of precision, any measurable function given that a sufficient number of processing neurons are available at the hidden layer; this property is known as the Universal Approximation Theorem (UAT) (Cybenko, 1989; Hornik, Stinchcombe, & White, 1989).
2) Number of neurons for the hidden layer(s): the UAT holds true only if a sufficient number of hidden neurons are available. Intuitively, ANNs with more hidden neurons have more free parameters (w) and are therefore capable of learning more complex functions.
3) Activation function $z(.)$: As mentioned before, each neuron processes its input via a pre-defined activation function. Neurons at the same layer usually employ identical functions. In the analyses presented in the remainder of this paper, a tangent sigmoidal function has been employed at the hidden layer neurons, as it has been shown to lead to fast training times (LeCun, Bottou, Orr, & Müller, 2012). For the output layer, a so-

called softmax function is used (which is essentially a logit) to ensure that the sum of the choice probabilities equals one.

Appendix 4B. Specifications of linear additive random utility maximisation model

Table A1 shows the estimation results of the linear-additive random utility maximisation (RUM) model used in this study (see Table 1 for attributes' name, notation, and description). The model is estimated in Multinomial Logit (MNL) form.. As can be seen, and as is expected, all parameters have the intuitively correct sign and are highly significantly different from zero. Table A2 shows the observed utility function for RUM model.

Table 4B.1. Estimation results for RUM-MNL model

No. of observations	77,638	
Final LL	-50,716.29	
ρ^2	0.41	
Attribute	*Est.*	*Rob. t-values*
ASC_Drive	0	fixed
ASC_PubTr	1.81	59.60
ASC_Walk	2.64	74.47
TT	-6.10	-95.31
TC	-0.135	-8.85
DL	1.02	46.26
CO	1.38	89.96
INTER	0.765	37.99
TRAF	-2.70	-43.44
AG_TC	-0.128	-4.04
DIS_TC	0.00937	11.59
FEM_TC	-0.0377	-4.31

Table 4B.2. Utility function specifications

$$V_{Drive} = ASC_{Drive} + \beta_{TT}TT_{Drive} + \beta_{TC}TC_{Drive} + \beta_{AG\_TC}(AG * TC_{Drive}) + \beta_{DIS\_TC}(DIS * TC_{Drive}) + \beta_{FEM\_TC}(FEM * TC_{Drive}) + \beta_{DL}DL + \beta_{CO}CO + \beta_{TRAF}TRAF$$

$$V_{PubTr} = ASC_{PublTr} + \beta_{TT}TT_{PubTr} + \beta_{TC}TC_{PubTr} + \beta_{AG\_TC}(AG * TC_{PubTr}) + \beta_{DIS_{TC}}(DIS * TC_{PubTr}) + \beta_{FEM\_TC}(FEM * TC_{PubTr}) + \beta_{INTER}INTER$$

$$V_{Walk} = ASC_{Walk} + \beta_{TT}TT_{Walk} + \beta_{TC}TC_{Walk} + \beta_{AG\_TC}(AG * TC_{Walk}) + \beta_{DIS\_TC}(DIS * TC_{Walk}) + \beta_{FEM\_TC}(FEM * TC_{Walk})$$

Notations

V_i Observed part of utility of alternative *i*

ASC_i Specific constant of alternative *i*

β_{TT}	Taste parameter associated with travel time attribute
β_{TC}	Taste parameter associated with travel cost attribute
β_{AG_TC}	Taste parameter associated with interaction between age and travel cost attribute
β_{DIS_TC}	Taste parameter associated with interaction between travel distance and travel cost attribute
β_{FEM_TC}	Taste parameter associated with interaction between gender and travel cost attribute
β_{DL}	Taste parameter associated with driving license attribute
β_{CO}	Taste parameter associated with number of owned car attribute
β_{TRAF}	Taste parameter associated with traffic variability attribute
β_{INTER}	Taste parameter associated with number of interchanges attribute

References

Abu-Mostafa, Y. S., Magdon-Ismail, M., & Lin, H.-T. (2012). *Learning from data* (Vol. 4): AMLBook New York, NY, USA:.

Adebayo, J., Gilmer, J., Muelly, M., Goodfellow, I., Hardt, M., & Kim, B. (2018). *Sanity checks for saliency maps.* Paper presented at the Advances in Neural Information Processing Systems.

Alber, M., Lapuschkin, S., Seegerer, P., Hägele, M., Schütt, K. T., Montavon, G., . . . Kindermans, P.-J. (2019). iNNvestigate neural networks! *Journal of Machine Learning Research, 20*(93), 1-8.

Alwosheel, A., van Cranenburgh, S., & Chorus, C. (2017). *Artificial neural networks as a means to accommodate decision rules in choice models.* Paper presented at the International Choice Modelling Conference 2017.

Alwosheel, A., van Cranenburgh, S., & Chorus, C. G. (2018). Is your dataset big enough? Sample size requirements when using artificial neural networks for discrete choice analysis. *Journal of Choice Modelling, 28*, 167-182. doi:https://doi.org/10.1016/j.jocm.2018.07.002

Ancona, M., Ceolini, E., Öztireli, C., & Gross, M. (2017). *A unified view of gradient-based attribution methods for deep neural networks.* Paper presented at the NIPS 2017-Workshop on Interpreting, Explaining and Visualizing Deep Learning.

Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K.-R., & Samek, W. (2015). On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one, 10*(7), e0130140.

Bishop, C. M. (2006). *Pattern recognition and machine learning*: springer.

Böhle, M., Eitel, F., Weygandt, M., & Ritter, K. (2019). Layer-wise relevance propagation for explaining deep neural network decisions in MRI-based Alzheimer's disease classification. *Frontiers in aging neuroscience, 11*, 194.

Chen, C., Ma, J., Susilo, Y., Liu, Y., & 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*, 285-299.

Chiang, W.-y. K., Zhang, D., & Zhou, L. (2006). Predicting and explaining patronage behavior toward web and traditional stores using neural networks: a comparative analysis with logistic regression. *Decision Support Systems, 41*(2), 514-531.

Chollet, F. (2015). Keras.

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals, and Systems (MCSS), 2*(4), 303-314.

Doshi-Velez, F., & Kim, B. (2017). Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*.

Golshani, N., Shabanpour, R., Mahmoudifard, S. M., Derrible, S., & Mohammadian, A. (2018). Modeling travel mode and timing decisions: Comparison of artificial neural networks and copula-based joint model. *Travel Behaviour and Society, 10*, 21-32.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning* (Vol. 1): MIT press Cambridge.

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

Hall, P., & Gill, N. (2018). *An Introduction to Machine Learning Interpretability-Dataiku Version*: O'Reilly Media, Incorporated.

Haykin, S. S. (2009). *Neural networks and learning machines* (Vol. 3): Pearson Upper Saddle River.

Hensher, D. A., & Ton, T. T. (2000). A comparison of the predictive potential of artificial neural networks and nested logit models for commuter mode choice. *Transportation Research Part E: Logistics and Transportation Review, 36*(3), 155-172.

Hillel, T., Elshafie, M., & Ying, J. (2018). Recreating passenger mode choice-sets for transport simulation: A case study of London, UK. *Proceedings of the Institution of Civil Engineers - Smart Infrastructure and Construction, 171*(1), 29-42. doi:10.1680/jsmic.17.00018

Hillel, T., Elshafie, M. Z., & Jin, Y. (2018). Recreating Passenger Mode Choice-Sets for Transport Simulation. *Proceedings of the Institution of Civil Engineers-Smart Infrastructure and Construction*, 1-49.

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural networks, 2*(5), 359-366.

Karlaftis, M. G., & Vlahogianni, E. I. (2011). Statistical methods versus neural networks in transportation research: Differences, similarities and some insights. *Transportation Research Part C: Emerging Technologies, 19*(3), 387-399.

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

Kittley-Davies, J., Alqaraawi, A., Yang, R., Costanza, E., Rogers, A., & Stein, S. (2019). *Evaluating the effect of feedback from different computer vision processing stages: a comparative lab study.* Paper presented at the Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). *Imagenet classification with deep convolutional neural networks.* Paper presented at the Advances in neural information processing systems.

Lapuschkin, S., Binder, A., Montavon, G., Muller, K.-R., & Samek, W. (2016). *Analyzing classifiers: Fisher vectors and deep neural networks.* Paper presented at the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

Lapuschkin, S., Binder, A., Montavon, G., Müller, K.-R., & Samek, W. (2016). The LRP toolbox for artificial neural networks. *The Journal of Machine Learning Research, 17*(1), 3938-3942.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature, 521*(7553), 436-444.

LeCun, Y. A., Bottou, L., Orr, G. B., & Müller, K.-R. (2012). Efficient backprop *Neural networks: Tricks of the trade* (pp. 9-48): Springer.

Lee, D., Derrible, S., & Pereira, F. C. (2018). Comparison of four types of artificial neural network and a multinomial logit model for travel mode choice modeling. *Transportation Research Record, 2672*(49), 101-112.

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

Mckinney, S. M., Sieniek, M., Gilbert, F., Godbole, V., Godwin, J., Antropova, N., . . . Corrado, G. C. (2020). International evaluation of an AI system for breast cancer screening. *Nature, 577*, 89-94. doi:10.1038/s41586-019-1799-6

Montavon, G., Samek, W., & Müller, K.-R. (2018). Methods for interpreting and understanding deep neural networks. *Digital Signal Processing, 73*, 1-15.

Olah, C., Satyanarayan, A., Johnson, I., Carter, S., Schubert, L., Ye, K., & Mordvintsev, A. (2018). The building blocks of interpretability. *Distill, 3*(3), e10.

Polson, N. G., & Sokolov, V. O. (2017). Deep learning for short-term traffic flow prediction. *Transportation Research Part C: Emerging Technologies, 79*, 1-17.

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). *Why should i trust you?: Explaining the predictions of any classifier.* Paper presented at the Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining.

Rosenfeld, A., & Richardson, A. (2019). Explainability in human–agent systems. *Autonomous Agents and Multi-Agent Systems*, 1-33.

Samek, W., Montavon, G., Vedaldi, A., Hansen, L. K., & Müller, K.-R. (2019). *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*: Springer Nature.

Samek, W., Wiegand, T., & Müller, K.-R. (2017). Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models. *arXiv preprint arXiv:1708.08296*.

Shrikumar, A., Greenside, P., & Kundaje, A. (2017). *Learning important features through propagating activation differences.* Paper presented at the Proceedings of the 34th International Conference on Machine Learning-Volume 70.

Simonyan, K., Vedaldi, A., & Zisserman, A. (2013). Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*.

Springenberg, J. T., Dosovitskiy, A., Brox, T., & Riedmiller, M. (2014). Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*.

Sun, Y., Jiang, Z., Gu, J., Zhou, M., Li, Y., & Zhang, L. (2018). Analyzing high speed rail passengers' train choices based on new online booking data in China. *Transportation Research Part C: Emerging Technologies, 97*, 96-113.

van Cranenburgh, S., & Alwosheel, A. (2019). An artificial neural network based approach to investigate travellers' decision rules. *Transportation Research Part C: Emerging Technologies, 98*, 152-166.

Xie, D.-F., Fang, Z.-Z., Jia, B., & He, Z. (2019). A data-driven lane-changing model based on deep learning. *Transportation Research Part C: Emerging Technologies, 106*, 41-60.

Zeiler, M. D., & Fergus, R. (2013). Stochastic pooling for regularization of deep convolutional neural networks. *arXiv preprint arXiv:1301.3557*.

Zintgraf, L. M., Cohen, T. S., Adel, T., & Welling, M. (2017). Visualizing deep neural network decisions: Prediction difference analysis. *arXiv preprint arXiv:1702.04595*.

An artificial neural network based approach to investigate travellers' decision rule

van Cranenburgh, S., & Alwosheel, A. (2019). An artificial neural network based approach to investigate travellers' decision rules. Transportation Research Part C: Emerging Technologies, 98, 152-166.[38]

Abstract:
This study develops a novel Artificial Neural Network (ANN) based approach to investigate decision rule heterogeneity amongst travellers. This complements earlier work on decision rule heterogeneity based on Latent Class discrete choice models. We train our ANN to recognise the choice patterns of four distinct decision rules: Random Utility Maximisation, Random Regret Minimisation, Lexicographic, and Random. Next, we apply our trained ANN to classify the respondents from a recent Value-of-Time Stated Choice experiment in terms of their most likely employed decision rule. We cross-validate our findings by comparing our results with those from: (1) single class discrete choice models estimated on subsets of the data, and (2) latent class discrete choice models. The cross-validations provide strong support for the notion that ANNs can be used to identify underlying decision rules in choice data. As such, we believe that ANNs provide a valuable addition to the toolbox of analysts who wish to investigate decision rule heterogeneity. The substantive contribution of this study is that we provide strong empirical evidence for the presence of decision rule heterogeneity amongst travellers.

[38] My personal contributions in this body of research are developing the network architecture, optimising codes, reflecting on and making comments to the manuscript (at several stages), and the revision and the replies to the reviewers.

1 Introduction

Artificial Neural Networks (ANNs) are gaining increasing popularity in many research fields, including in transportation (e.g. Alwosheel, Van Cranenburgh, & Chorus, 2017; Borysov, Lourenço, Rodrigues, Balatsky, & Pereira, 2016; Hensher & Ton, 2000; Mohammadian & Miller, 2002; Omrani, Charif, Gerber, Awasthi, & Trigano, 2013; van Lint, Hoogendoorn, & van Zuylen, 2005; Wong, Farooq, & Bilodeau, 2017). ANNs are mathematical models which are loosely inspired by the structure and functional aspects of biological neural systems. Their recent uptake can be explained by major breakthroughs in ANN research, affecting the daily lives of many people (such as e.g. in the context of natural language processing or facial recognition), in combination with the rise of emerging data (Chen, Ma, Susilo, Liu, & Wang, 2016; Vlahogianni, Park, & van Lint, 2015). Particularly, the versatile architecture of ANNs makes them well-equipped to deal with large volumes of (unstructured) emerging data (Maren, Harston, & Pap, 2014).

However, despite the general excitement in the field of transportation about the potential of ANN (and other data-oriented techniques), the number of areas of application of ANNs in transport, and in particular in the travel behaviour research subfield, is still fairly limited. Currently, ANNs are predominantly used to analyse observed movement patterns and to make short-term travel demand predictions. However, a recent paper in this journal (Chen et al., 2016) advocates to move beyond this state, and use these data-oriented techniques to improve understanding of the travel behaviour underlying human mobility patterns. In particular, Chen et al. suggest using data-oriented methods to identify factors explaining travel decisions and to uncover underlying decision-rules.

This paper answers to this call: it develops an ANN based approach to investigate travellers' decision-rules. Decision-rules are the decision mechanisms humans use when making choices (Payne, Bettman, & Johnson, 1993). Decision-rules are widespread in transportation research as they are embedded in discrete choice models. Although the vast majority of discrete choice models are built on a single decision-rule (random utility maximisation), there is a growing recognition amongst transport researchers that travellers are heterogeneous in terms of their decision-rules. Also, it is increasingly acknowledged that insights on decision-rule heterogeneity are crucial for understanding and predicting travel behaviour. In this context, a number of recent studies have explored decision-rule heterogeneity amongst travellers (using traditional discrete choice models) (Balbontin, Hensher, & Collins, 2017; Boeri & Longo, 2017; Gonzalez-Valdes & Raveau, 2018; Hess & Chorus, 2015; Hess, Stathopoulos, & Daly, 2012; Leong & Hensher, 2012).

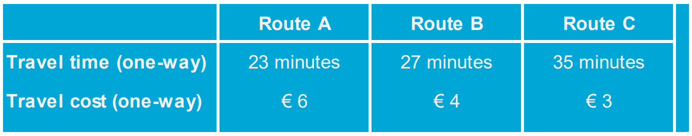
Specifically, in this study we develop a novel pattern recognition Artificial Neural Network (ANN) to classify travellers in terms of their most likely employed decision-rules based on observed sequence of choices. By doing so, this study aims to shed new light on decision-rule heterogeneity amongst travellers. In order to detect patterns in the data ANNs need to be trained based on so-called training data, which includes correct classifications. However, in the context of decision-rules the 'true' decision-rule is inherently unknown. In fact, decision-rules should rather be perceived as quintessential models explaining choice behaviour in a parsimonious way than as models that accurately reflect the complex decision-making processes (Chorus, 2014). Therefore, in the absence of real-world training data consisting of the 'correct' classifications, we train our ANN using synthetic data. The synthetic decision-makers which we created for training are heterogeneous not only in terms of their employed decision-rules, but also in terms of their preferences. By doing so, the ANN is trained to classify travellers in terms of their decision-rules in the realistic condition in which besides decision-rule heterogeneity also taste heterogeneity and heteroscedasticity are present. Finally, we apply our trained ANN to classify the respondents from a recent Value-of-Time (VoT) Stated Choice (SC) experiment, and cross-validate its classifications using traditional discrete choice analysis. The methodological contribution of this paper is that it is the first to show how ANNs can be employed to investigate travellers' decision-rule heterogeneity. We present a novel ANN topology that is particularly suited to deal with sequences of choice observations, and show how to train it using synthetic data. As such, this research complements earlier work on decision-rule heterogeneity based on Latent Class discrete choice models. The substantive contribution of this study is that we provide new, strong, empirical evidence for the presence of decision-rule heterogeneity amongst travellers.

The remainder of this paper is organised as follows. Section 2 first presents the empirical data that we aim to analyse using the ANN based approach. Section 3 develops the ANN and applies it to the empirical data set. In section 4 we cross-validate our results obtained from the ANN by comparing them with results obtained using traditional LC discrete choice models. Finally, section 5 draws conclusions and provides a discussion.

2 Data

For this study we use data from a relatively small VoT SC experiment[39]. We choose this data set because of its simplicity. Its simplicity provides a degree of tractability on the underlying decision-making mechanisms used by the respondents. In these data we can, for instance, straightforwardly identify the compromise alternative[40] and we can easily detect non-trading behaviour. This allows us latter on to cross-validate the results of our ANN.

Figure 1 shows a screenshot of the first choice tasks presented to respondents in the SC experiment. Choice tasks in this experiment consist of three unlabelled route alternatives, each consisting of two generic attributes: Travel Cost (TC) and Travel Time (TT). Attribute levels are selected as follows: the range of the travel times was chosen such that they are in consonance with the range of the travel times presented in previous European VoT SC-experiments, which is usually in the order of 10 to 15 minutes. The minimum travel time is set at 23 minutes, and the maximum at 35 minutes, with equally spaced 4 minutes intervals. In this experiment respondents were presented $T = 10$ choice tasks. To optimise the statistical efficiency of the experiment, a so-called D-efficient design is used. The statistical efficiency of the experimental design was optimised for a combination of RUM and P-RRM models, see Van Cranenburgh, Rose, and Chorus (2018) for more details. The complete experimental design can be found in Appendix 5A.



	Route A	Route B	Route C
Travel time (one-way)	23 minutes	27 minutes	35 minutes
Travel cost (one-way)	€ 6	€ 4	€ 3

Figure 1. Screenshot of 1st choice task (translated to Englsih).

[39] To avoid any misunderstanding, we note upfront that these data are not used for the training ANNs. These data would be too small to do so.

[40] Compromise alternatives have an intermediate performance on each or most attributes (relative to other alternatives in the choice set) rather than having a poor performance on some attributes and a strong performance.

2.1 Data collection

The data collection took place in The Netherlands in May 2016. To recruit respondents a panel company was used (TNS-NIPO). Only car commuters were admitted to conduct the choice experiment. In total 106 respondents completed the full survey. A relatively balanced sample has been obtained in terms of gender, age, education and income. The sample statistics can be found in Appendix 5B. See Van Cranenburgh et al. (2018) for more details on the data collection.

2.2 Decision rules

Close inspection of the choice data allows us to derive indications on the decision-rules that may have been used by the respondents in this route choice experiment. In each choice task respondents could choose between a 'fast and expensive', a compromise, and a 'slow and cheap' alterative, although they were not explicitly labelled as such in the SC experiment. Across all choice observations, in 30 per cent of the cases the 'fast and expensive' alternative is chosen; in 38 per cent of the cases the compromise alternative is chosen; and in 32 per cent of the cases the 'slow and cheap' alternative is chosen. The observation that the compromise alternative acquires the highest market share provides an indication that an RRM decision-rule may have been used by a considerable share of the respondents. After all, RRM models predict a market share bonus for the compromise alternative when compared to Random Utility Maximisation (RUM) (Chorus & Bierlaire, 2013; Guevara & Fukushi, 2016).

The stacked bar graph in Figure 2 provides more insights on the distribution of respondents' choices. It visualises for each of the 106 respondents in the data set the number of times the 'fast and expensive', the compromise and the 'slow and cheap' alternative has been chosen (note that we sorted the respondents on the *x*-axis for the sake of clarity). The bar graph reveals that 9 respondents consistently choose for the 'fast and expensive' alternative (full blue bars on the left) and 16 respondents who consistently chose for the 'slow and cheap' alternative (full yellow bars on the right). This suggests that a substantial share of the respondents opted for a lexicographic decision-rule.[41]

A substantial share of the respondents (77) switched between alternatives across the ten choice tasks. Of these 77 respondents, 29 respondents consistently chose either the 'fast and expensive' alternative or the compromise alternative (blue-green bars) and 13 respondents consistently chose either the 'slow and cheap' alternative or the compromise alternative (yellow-green bars). This signals a degree of rationality underlying the trade-offs, which seems consistent with RUM. Thirty-three respondents chose all three alternatives at least once (tri-coloured bars). Close inspection of the choices of these respondents using the half-space method (Rouwendal, de Blaeij, Rietveld, & Verhoef, 2010) reveals that their choices do not suggest a stable underlying VoT – at least not from a RUM modelling perspective. In addition, two respondents even chose either the 'fast and expensive' alternative or the 'slow and cheap' alternative. This observation suggests that a substantial share of respondents made (seemingly) random choices.

[41] Under a lexicographic decision-rule a decision-maker first evaluates the alternatives, then identifies the most important attribute and subsequently chooses the alternative that performs best in terms of this particular attribute.

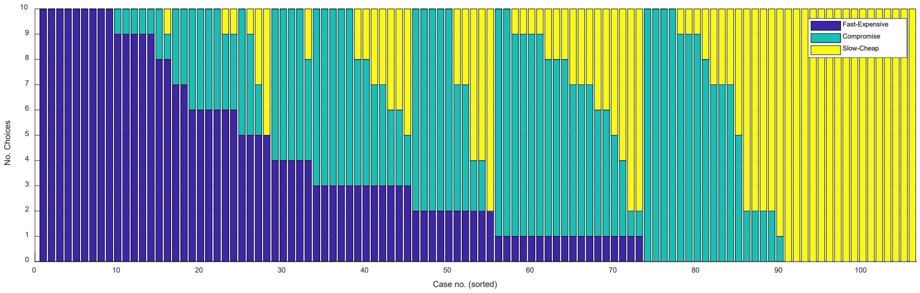


Figure 2. Choices of respondents.

Based on these descriptive analyses we obtain first indications that the following four decision-rules may be present in our data: RUM, RRM, Lexicographic, and Random. In the remaining part of this paper we will focus on these four decision-rules. The latter decision-rule 'Random' may seem a bit odd, in the sense that random behaviour is typically not considered a decision-rule and is therefore typically not explicitly accounted for in discrete choice studies. However, seasoned SC researchers know that in SC data typically a considerable share of respondents makes seemingly random choices. Therefore, we treat random choice behaviour as a separate decision-rule in the context of this study. Table 1 shows the mathematical formulations of these decision-rules as well as their implementation in a discrete choice modelling framework. Note that the RRM model we use in this study is the so-called P-RRM model (S. Van Cranenburgh, C. A. Guevara, & C. G. Chorus, 2015). This model is increasingly used in the RRM literature as yields the strongest regret minimization behaviour, i.e., the highest level of regret aversion which is possible, within the RRM modelling framework. As such, this RRM model postulates choice behaviour which is strongly different from RUM – which intuitively should make it easier to distinguish between these two decision-rules.

Table 1. Decision rules.

Decision rule	Choice rule	Total utility/regret	Value function	MNL choice probability
Utiliti maximisation	$y_i = 1 \Leftrightarrow U_i \geq U_j \forall j \in C$	$U_i = V_i + \varepsilon_i$	$V_i = \sum_m \beta_m + x_m$	$P_i = \frac{e^{V_i}}{\sum_{j \in C} e^{V_j}}$
P-RRM	$y_i = 1 \Leftrightarrow RR_i \leq RR_j \forall j \in C$	$RR_i = R_i + \varepsilon_i$	$R_i = \sum_{j \neq i} \sum_m \max(0, \beta_m [x_{jm} - x_{im}])$	$P_i = \frac{e^{-R_i}}{\sum_{j \in C} e^{-R_j}}$
Lexicographic	$y_i = 1 \Leftrightarrow x_{i\bar{m}} \geq x_{j\bar{m}} \forall j \in C$	N/A		N/A
Random	$y_i = 1 \Leftrightarrow U_i \geq U_j \forall j \in C$	$U_i = V_i + \varepsilon_i$	$V_i = 0 \ \forall i \in C$	$P_i = \frac{1}{J}$

Notation:

C	Set of alternatives
i, j	Choice alternatives in C
y_i	Indicator which denotes whether alternative *i* is chosen
U_i	Total utility of alternative *i*

V_i	Observed part of utility of alternative *i*
ε_i	Unobserved part of utility or regret of alternative *i*
β_m	Taste parameter associated with attribute *m*
x_{mi}	Attribute level of the m'th attribute of alternative *i*
P_i	Choice probability of alternative *i*
RR_i	Total regret of alternative *i*
R_i	Observed part of regret of alternative *i*
$\bar{m}$	The most important of the *M* attributes
J	Cardinality of the choice set
$x_{j\bar{m}}$	Attribute level of alternative *j* for the attribute $\bar{m}$

3 An artificial neural network based appraoch

Section 3.1 and Section 3.2 discuss the development of the ANN for decision rule classification. Next, 3.3 Training data, 3.4 Performance and cross validation present the training data and the performance of the trained network. Finally, in Section 3.5 we elaborate on how to employ the trained network to classify travellers in empirical data, and apply it to our empirical VoT data to classify the respondents.

3.1 Artificial neural networks

Artificial neural networks are inspired by the structure and functional aspects of biological neural systems. ANNs originate from the field of neuro and computer sciences, but are currently rapidly spreading out to other research disciplines (Maren et al., 2014). Underlying this rapid expansion is the emergence of so-called Big Data. The combination of large volumes of (unstructured) data on the one hand and the versatile architecture of ANNs on the other hand has led to numerous ground-breaking results in a variety of disciplines, such as speech recognition, gene detection of autism and natural language processing.

Computations in ANNs are structured in terms of interconnected groups of artificial neurons, processing information using a so-called connectionist approach (Bishop, 1995). ANNs are composed of nodes. Three types of nodes are commonly distinguished: input nodes, hidden nodes and output nodes. The input nodes contain the explanatory variables. In the context of choice models, these typically concern the attribute levels of the alternatives. The output nodes contain the dependent variables. In the context of choice models, e.g. i.e. when predicting choices, the output nodes consist of the choice probabilities. The signals propagate in forward direction, through the links which connect the nodes. The links have a numeric weight w, which needs to be learned from the data. At each node the weights are multiplied with the input value from the previous nodes and summed. Then the signal is propagated to the next layer using an activation function. Commonly used activation functions are tan-sigmoid, softmax and purelin. See Bishop (1995) for an extensive overview of ANNs and their characteristics.

Despite the fact that an extensive variety of ANNs have been developed (Maren et al., 2014) to tackle all sorts of (classification) problems – each of which with strengths particular to their application –, to the best of the authors' knowledge no type of ANN has been put forward that is particularly suited to investigate decision rule heterogeneity. Our classification problem is somewhat unconventional as we aim to make a classification based on an unordered sequence of correlated (choice) observations. Note that this is conceptually different from the more commonly encountered ordered sequence-to-sequence classification problem. To classify a traveller in terms of his or her (most likely) employed decision rule we need to assess a sequence of choice observations made by a traveller. After all, based on a single choice observation of a traveller it is virtually impossible to make such a classification, as any single choice could be driven by any decision rule (considering some randomness is present). In contrast, a sequence of choice observations may provide a 'fingerprint' of what is the most likely employed decision rule. However, for our classification problem the order of the observations within the sequence is irrelevant. Essentially, what we want is a network that classifies decision-makers to decision rules, having seen the full sequence of observations of that a decision-maker, regardless of the order in which the observations are presented to the network. Therefore, we cannot use ANN types that are specifically designed to capture serial correlations, such as the Recurrent ANN. In the absence of a suitable 'off-the-shelf' ANN, the next subsection proposes a new ANN topology that is particularly designed for classification of travellers' decision rules (and the data set presented in Section 2).

3.2 An artificial neural network for decision rule classification

ANNs are often pitched as a generic method that can be used to model any problem. However, this is a misconception: much of the art of machine learning is determining how to incorporate problem specific knowledge into the ANN via its topology, performance function, activation functions, etc. (Maren et al., 2014).

To develop an ANN capable to classify decision-makers to decision rules, we have tested different types of ANNs (Multi-layer Perceptron and LSTM) and experimented with different topologies, number of hidden layers, activation functions as well as the degree of connectivity between hidden layers. We compared the networks in terms of their classification performance, while considering their complexity (in terms of e.g. number of layers and number of nodes at each layer).

Figure 3 shows the topology of our best performing ANN.[42] The ANN is a Multi-Layer Perceptron (MLP). A key characteristic of the ANN topology is that it processes sequences of choice tasks made by a decision-maker as one chunk of input data (note that the input layer contains all ten choice tasks including the observed choice). Hence, the model treats a sequence of choices of a traveller as one independent observation. This is crucial as the choice observations belonging to the same individual are correlated and therefore cannot be treated as independent observations.

The ANN's topology is behaviourally informed in the sense that behavioural intuition is added to the network to help it grasp the data structure and classification problem. In particular, the ANN is deliberately sparsely connected, i.e., many nodes are not connected to one another. For instance, the input nodes of one choice task are not connected with the hidden nodes of other choice tasks. Behaviourally, this makes sense because the attributes (input nodes) presented to a decision-maker in e.g. choice task $t = 10$ simply cannot affect the choice in choice task $t = 1$ (i.e., unless the respondent is allowed to revise his or her choices, which is usually not the case in SC experiments). On the other hand, the choice in choice task $t = 1$ could affect the choice in e.g. the next choice task. However, our aim in this paper is to uncover decision rules

[42] Note that the topology of the ANN in Figure 3 is specifically tweaked in terms of the number of input nodes and the number of decision-rules (output nodes) to match the structure of the empirical data that we aim to analyse (see Section 2). However, it can easily be adjusted to fit other data sets.

based on a sequence of choice observations of an individual, regardless of the order in which they are presented to the network. Therefore, we actually want to prevent the network to pick up learning and inertia effects (if present). Furthermore, in a sense, the first hidden layer takes care of assigning ‘values’ to alternatives and combining these with the observed choice, which are then passed on to the second hidden layer where the sequence as a whole is processed to make the decision rule classification. From this perspective, the full connection between the second hidden layer and the output layer is intuitive as it is the sequence of choice observations that is informative on the employed decision rule.

On a more technical level, our network uses two hidden layers. We find that adding more hidden layers does not improve the performance, while removing one layer deteriorates the performance drastically. At the first hidden layer (for each choice task) as well as at the second hidden layer we use four nodes. This is found to result in the best performance. Decreasing the number of nodes (either at the first or second hidden layer) decreases the classification performance noticeably, while increasing the number of nodes beyond four does not seem to improve performance (while consuming considerably larger number of weights than needed). Also note that no bias nodes are present in the network (see Figure 3). During training the order of the alternatives and the order of the choice sets are fully shuffled (see Section 3.3), meaning that bias nodes become superfluous. The total number of weights of the network is 496. Furthermore, our network uses tan-sigmoid activation functions (LeCun, Bottou, Orr, & Müller, 2012) at the nodes of the hidden layers. Other types of activation function are found to perform worse, or result in longer training times. At the nodes of the output layer we however use softmax activation functions. This ensures that the sum of the decision rule classification probabilities adds up to 1.

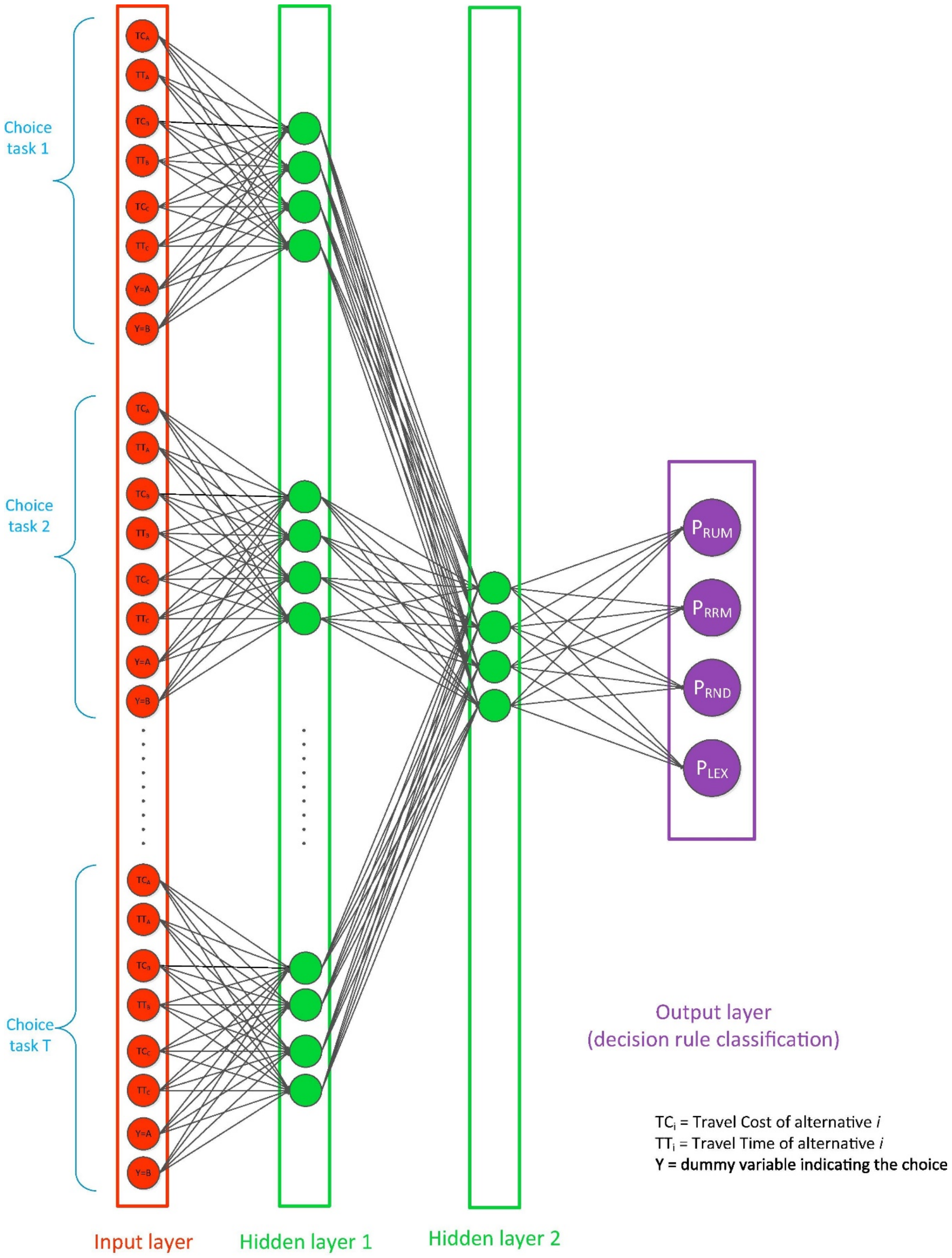


Figure 3. ANN topology.

3.3 Training data

Training the ANN involves exposing it to data containing the correct classifications. However, as noted before, in the context of human choice behaviour the 'true' decision rule is inherently unknown. In fact, decision rules should rather be seen as quintessential mathematical models representing highly complex decision processes. To deal with the fact that we do not have real data containing the 'true' decision rule, we train our ANN using synthetic data. These data are created using the decision rules shown in Table 1.

The training data set consists of 40,000 synthetic decision-makers; 10,000 decision-makers for each decision rule. Hence, the training data set is fully balanced. In consonance with the VoT data that we aim to analyse (see Section 2) each synthetic decision-maker is confronted with $T = 10$ choice tasks. The choice tasks are the same as those used in the empirical data collection (see Appendix 5A).

Given that synthetic data are in unlimited supply, the number of synthetic decision-makers is deliberately set high. A commonly used rule-of-thumb in machine learning is that the sample size needs to be (at least) 10 times larger than the number of estimable weights in the network (Haykin, 2009). A recent study specifically dealing with sample size requirements for using ANNs in the context of choice models is more conservative, and recommends to use a sample size of (at least) 50 times the number of estimable weights (Alwosheel, van Cranenburgh, & Chorus, 2018). By using 40,000 observations in our study, our sample size is a comfortable 80 times larger than the number of estimable weights. Thereby, we safely avoid overfitting issues.

Importantly, besides decision rule heterogeneity, in the empirical data that we aim to analyse there are two other potential sources of correlation across the sequence of choice observations of individuals. Firstly, decision-makers can be heterogeneous in their preferences. Such taste heterogeneity creates correlation in the sequence of choices of an individual because they are generated using the same set of individual specific parameters. Secondly, decision-makers may learn from their earlier made choices, creating serial correlation across the observations. For instance, the respondent could choose the fast and expensive alternative in, say, choice task 5 because he/she also chose the fast and expensive alternative in the choice tasks before, and he/she would like to stick to that choice.

During training we need to account for these sources of correlation, such that the trained network is able to detect the correlation specifically caused by the decision rule (and becomes capable to accurately classify decision-makers in terms of their employed decision rule). To account for learning effects, during the training stage we shuffle the order of the choice tasks and alternatives in the training data. Thereby, we prevent the network from learning (1) the (fixed) structure of the data and (2) ordering effects caused by learning or inertia (if present). Our synthetic decision-makers do not have the ability to learn, meaning that there are no learning effects to capture during training. Still, shuffling the order of alternatives and choice sets at the training stage is important as not doing so would preclude doing so in the application stage (where it is important). Then, when we apply the network to classify respondents in the empirical data set (Section 3.5), we also shuffle – at the level of the individual – the order of the choice tasks and the order of the alternatives. By doing so, serial-correlation in the empirical data is removed (if present).

To account for heterogeneity in preferences, we take a different approach. We created synthetic decision-makers which are heterogeneous in their preferences. By doing so, the ANN is trained to classify travellers in terms of their decision rules in the realistic condition in which both taste heterogeneity and heteroscedasticity are present. In the context of the decision rules considered in this study, taste heterogeneity is only relevant for RUM and RRM decision rules, as these decision rules contain taste parameters governing the decision-making process. Specifically, for both RUM and RRM decision-makers we assume that tastes for travel cost and travel time are symmetrically triangular distributed across decision-makers. Other distributions (normal and uniform) have been tested, but gave by and large similar results.

Accordingly, to generate the RUM and RRM choices every synthetic decision-maker is attributed two independent draws from the associated triangular densities for the marginal utilities/regrets of cost and time. Pseudo-random draws are generated from the Extreme Value Type I distribution for every alternative in the choice task. For RUM decision-makers, the alternative with the highest total utility is chosen; for RRM decision-makers the alternative with the minimum total regret is chosen. Forβ_{Cost} the lower bound of the distribution is set at $a = -3.2$, the mean at -1.6 and the upper bound is set at $c = 0$; for β_{Time} the lower bound is set at $a = -0.8$, the mean at -0.6 and the upper bound is set at $c = 0$. The means of β_{Cost} and β_{Time} are chosen based results from a LC discrete choice analysis conducted prior to training the ANN. However, we also tested larger and smaller values and found that these do not substantially influence results. For the RRM model a choice set size correction factor of 2/3 is applied to the parameters, see (Van Cranenburgh, Prato, & Chorus, 2015). Thereby, it is ensured that the using the same parameterisation for RUM and RRM corresponds to approximately the same degree of choice consistency. This avoids that the ANN learns that relatively deterministic choice behaviour is specific for RUM while relatively Random choice behaviour is specific for RRM, or vice versa.

To generate the Lexicographic choices, for 5000 decision-makers we set the choice to be the fastest route, and for the other 5000 decision-makers we set the choice to be the cheapest route. For this decision rule no randomness is added. To generate the Random choices, we simply took a draw z from the standard uniform distribution for each choice task. After all, random choice behaviour implies equal choice probabilities across all alternatives. In case $z < 1/3$ then the choice is set to alternative A, in case $\frac{1}{3} < z < \frac{2}{3}$ the choice is set to alternative B, and in case $z > 2/3$ the choice is set to alternative C.

3.4 Performance and cross validation

The ANN is implemented in MATLAB2017. Prior to training, the data are normalised to minimise training time and reduce the probability of ending-up with suboptimal solutions. To train the ANN Levenberg-Marquardt backpropagation is used. This training algorithm is built-in and found to work well. Particular advantages of this algorithm over other training algorithms are that it is computationally relatively fast and requires relatively little memory. Training the ANN takes a few minutes using a desktop PC with six CPUs.

To test the capability of the ANN to classify decision-makers we use the so-called k-fold cross-validation method, with $k = 10$. With 40,000 synthetic decision-makers, the data set is split into 10 folds of 4000 randomly selected decision-makers, although we made sure that in each fold the four decision rules are equally represented. In each repetition of the training and testing, the data of 36,000 decision-makers are used for training and the data of 4000 decision-makers are used as a holdout set for testing. Holdout sets are selected such that their union over all repetitions is the entire training set. By doing so, the data of every decision-maker is guaranteed to be part of the training and testing data. The trained network is eventually applied in Section 3.5 to classify the respondents in the empirical data.

Table 2 shows the k-fold confusion matrix. Using the trained network each decision-maker in the test data sets are assigned to a decision rule based on the highest classification probability. This assignment is then compared to the true classification. More specifically, the cells on the diagonal show the mean percentage of the decision-makers that are correctly classified, across the 10 folds. The off-diagonal cells show the mean percentage of decision-makers that are misclassified into a certain decision rule, across the 10 folds. In parenthesis below the means values, the standard deviation is reported.

Table 2. Classification of ANN based on highest likelihood.

		ANN Classification [%]				
		RUM	RRM	Lexicographic	Random	∑
True DGP	RUM	54.6 (3.2)	16.1 (1.9)	9.7 (1.1)	19.7 (3.4)	100%
	RRM	18.7 (2.3)	67 (1.9)	1.1 (0.3)	13.2 (1.7)	100%
	Lexicographi c	0.2 (0.2)	0.0 (0.0)	99.8 (0.2)	0.1 (0.1)	100%
	Random	12.5 (1.4)	9.9 (1.4)	0.9 (0.5)	76.8 (2.5)	100%

Based on Table 2 a number of inferences can be made. Firstly, the percentage correctly classified decision-makers is fairly good. It ranges between 54 (for RUM) to 99.8 per cent (for Lexicographic). Altogether, across the 10 folds between 73 and 76 per cent of the decision-makers are correctly classified by the trained ANN (with an average of 75 per cent). Finally, the small standard deviations show that the classification is relatively stable across the folds.

The fact that not all decision-makers are correctly classified may, at first sight, seem not very promising. However, this was actually to be expected for two reasons. One reason is that when generating choices[43] we explicitly add randomness to account for unobserved factors on the side of the analyst. Therefore, in principle, any generated sequence of choices may just by coincidence appear to be generated by a certain decision rule, while it was generated by another. Another reason is that for certain parameterisations of the decision rules the implied choice behaviour becomes indistinguishable. Specifically, the taste parameters for RUM and RRM are drawn from symmetric triangular distributions. These distributions have mass very close to zero, meaning that individuals can have taste parameters that will generate choice behaviour that is virtually indistinguishable from Random choice behaviour. Likewise, the taste parameters may have been drawn such that one attribute is relatively far more important than the other, for instance, in case $\beta_{Cost} = -3$ and $\beta_{Time} = -0.01$. The resulting choice behaviour is then virtually indistinguishable from lexicographic choice behaviour. After all, with extreme ratios of parameters a decision-maker will always choose for either the cheapest or the fastest alternative. On a more general note, the fact that not all decision-makers are correctly classified highlights that classification of decision-makers to decision rules based on small finite sequence of choices – in the presence of randomness and taste heterogeneity – is inherently a hard task.

Close inspection of Table 2 shows that the ANN has most difficulty with distinguishing between RUM and RRM decision rules. 16.1 per cent of the RUM decision-makers are misclassified as RRM and 18.7 per cent of the RRM decision-makers are misclassified as RUM. This highlights that RUM and RRM differ from one another in rather subtle ways, even though we used the P-RRM model – which imposes very strong regret minimization behaviour.

[43] Except for lexicographic choice behaviour.

3.5 Application to empirical data

Next, we use our trained ANN to classify the 106 respondents in the data set presented in Section 2. To do so, we use a resampling approach. That is, for each respondent we shuffle the order of the sequence of choice observations and the order of the alternatives (left, middle, right) and apply the trained network. The respondent is then classified to the decision rule that attains the highest probability. But, rather than classifying each respondent just once, we classify each respondent 1000 times based on 1000 reshuffles of the order of the sequence of choice observations. After that, each respondent is classified to the decision rule that attains the highest number of 'votes' across the 1000 trials. We find that some respondents are very consistently classified by the network to one particular decision rule – regardless of how the data are shuffled. For other respondents the particular manifestation of the data does affect the decision rule classification. However, by classifying each respondent 1000 times, we obtain stable results in terms of what is the most likely decision rule employed for each respondent in our data.

Table 3 shows the final classifications. Based on Table 3 the following observations can be made. Firstly, and we consider this a noticeable substantive finding, we see that only 22 out of the 106 respondents are classified as random utility maximisers. This finding may spark further debate on the dominance of RUM models in discrete choice modelling practices – although we certainly do not want to claim that these market shares are one-to-one transferable to other choice contexts. Secondly, we see that the largest number of respondents (51) is classified as random regret minimisers. Thirdly, respectively 27 and 6 respondents are classified as Lexicographic and Random decision-makers.

Table 3. Classification based on highest likelihood N=106.

Decision-rule	RUM	RRM	Lexicographic	Random
No. respondents	22	51	27	6

The design of the SC experiment itself may partly explain the obtained markets shares, for a number of reasons. Firstly, the relatively high market share of the RRM decision rule may (in part) be attributed to the fact that the compromise alternative is very easy to identify for respondents. Therefore, it seems possible that the design of the SC experiment may actually have triggered an RRM like decision rule on the side of the respondent. Secondly, the relatively high market share of the Lexicographic decision rule may be due to the fact that the experimental design consists of just two attributes (possibly in combination with the ranges of the attribute levels). This makes it fairly easy for respondents to use a lexicographic decision rule; a respondent may first decide on the most important attribute (either travel cost or travel time) and then choose consistently the best alternative based on that attribute.

The proposed method is rather flexible in application. It can be applied to SC as well as to Revealed Preference (RP) data. Applying the method to SC data seems however most natural since SC data are generally less noisy, which is an important asset since differences between decision rules can be subtle. We have applied the method in the context of a straightforward three alternative, two attribute SC data set. However, the method can essentially be applied to data sets which involve any number of attributes or alternatives. Finally, it is worthwhile to note that since the actual training is conducted on synthetic data, the size of the empirical data set is never a limiting factor to apply the method (this in contrast to discrete choice based methods). This makes the method also particularly suited to investigate decision rule heterogeneity in small data sets.

4 Cross-validation using discrete choice models

This section aims to cross-validate the classification results of Section 3.5. Although the performance of the ANN on the test data set is encouraging, it is good to keep in mind that we used synthetic data to train our ANN (Section 3.3) while in Section 3.5 we ultimately apply the ANN to real empirical data (Section 3.5). Due to this discrepancy between training and application additional analyses are needed to further examine the capability of the ANN to classify travellers to decision rules. The next two subsections present these analyses.

4.1 Model fit based on subsets

One way to cross-validate the capability of the ANN to classify respondents to decision rules is by estimating discrete choice models based on carefully created subsamples of the data. In particular, we split the 106 respondents in the data set into four subsets based on the highest classification probability as predicted by the ANN. That is, subset 1 consists of the 22 respondents classified by the ANN as random utility maximisers; subset 2 consists of the 51 respondents classified by the ANN as random regret minimisers, and so on. For these subsets we formulate the following conjectures:

If the ANN has accurately classified respondents to decision rules, then

1. The RUM model outperforms the P-RRM model in subset 1.
2. The P-RRM model outperforms the RUM model in subset 2.
3. Both the RUM and the P-RRM model have a very poor model fit in subset 4.
4. The respondents allocated to subset 3 include those 25 respondents which in Section 2.2 are identified to have consistently chosen for the fast and expensive or the slow and cheap alternative across all choice tasks.

On these subsets we estimate three types of discrete choice models: a linear-additive RUM model, a P-RRM model and a μRRM model (Sander van Cranenburgh, Cristian Angelo Guevara, & Caspar G Chorus, 2015).[44] The latter model is a very flexible type of RRM model, which holds both the linear-additive RUM and the P-RRM model as special cases. The μRRM model consists of one additional parameter (as compared to linear-additive RUM and P-RRM). This 'regret aversion parameter' μ captures the degree of regret minimisation behaviour, where $\mu \rightarrow 0$ implies a very strong degree of regret aversion (i.e. a P–RRM decision rule) and $\mu \rightarrow \infty$ implies no regret aversion (i.e. a linear-additive RUM decision rule). As such, this model allows to empirically establish the underlying decision rule (RUM or RRM with varying degrees of regret aversion). All models are estimated in Multinomial Logit (MNL) form.

Table 4 shows estimation results.[45] First we look at the results for subset 1. We see that the RUM model very strongly outperforms the P-RRM model on this subset. The difference in final log-likelihood when put to the (Ben-Akiva & Swait, 1986) test for nonnested models, is very significant at $p < 0.000$. In fact, the very low ρ^2 for the P-RRM model indicates this model is hardly able to describe the data generating process in any meaningful manner. Despite this, we see that the signs are recovered. Looking at the results for the μRRM model, we see that the regret aversion parameter μ hits the upper bound, which is set at $\mu = 100$ in the estimation. Hence, the μRRM model collapses to the linear-additive RUM model, implying that imposing any degree of regret minimisation behaviour would worsen the model fit. Therefore, conjecture (1) is supported by these discrete choice analyses. As such, we are rather confident that the

[44] A choice set size correction factor is applied to the RRM models to be fully consistent with Section 3. This has no impact on model fit; it merely scales the estimates by a fix factor.

[45] Estimating results based on the full data set can be found in Appendix 5C.

ANN has been successful in identifying those travellers whose choice behaviour is best described by RUM.

Table 4. Estimation results.

Subset 1: Respondents identifified as random utility maximisers

Model	RUM MNL			P-RRM MNL			muRRM MNL		
Number of observations	220			220			220		
Number of individuals	22			22			22		
Null Log-likelihood	-241.7			-241.7			-241.7		
Final Log-likelihood	-212.2			-238.0			-212.2		
ρ^2	0.12			0.02			0.12		
Parameters	*Est*	*Std error*	*t-val*	*Est*	*Std error*	*t-val*	*Est*	*Std error*	*t-val*
β_{cost}	-1.26	0.176	-7.13	-0.29	0.115	-2.57	-1.24	0.174	-7.11
β_{time}	-0.30	0.046	-6.61	-0.05	0.027	-1.82	-0.30	0.045	-6.58
µ							100		

Subset 2: Respondents identifified as random regret minimisers

Model	RUM MNL			P-RRM MNL			muRRM MNL		
Number of observations	510			510			510		
Number of individuals	51			51			51		
Null Log-likelihood	-560.3			-560.3			-560.3		
Final Log-likelihood	-510.0			-383.2			-383.2		
ρ^2	0.09			0.32			0.32		
Parameters	*Est*	*Std error*	*t-val*	*Est*	*Std error*	*t-val*	*Est*	*Std error*	*t-val*
β_{cost}	-0.88	0.110	-8.01	-1.63	0.130	-12.57	-1.63	0.130	-12.57
β_{time}	-0.28	0.030	-9.19	-0.50	0.037	-13.29	-0.50	0.037	-13.29
µ							0.01		

Subset 4: Respondents identifified to choose at random

Model	RUM MNL			P-RRM MNL			muRRM MNL		
Number of observations	60			60			60		
Number of individuals	6			6			6		
Null Log-likelihood	-65.9			-65.9			-65.9		
Final Log-likelihood	-61.1			-61.2			-61.1		
ρ^2	0.07			0.07			0.07		
Parameters	*Est*	*Std error*	*t-val*	*Est*	*Std error*	*t-val*	*Est*	*Std error*	*t-val*
β_{cost}	0.37	0.309	1.19	0.27	0.210	1.28	0.37	0.309	1.19
β_{time}	0.00	0.080	-0.01	-0.02	0.058	-0.40	0.00	0.080	-0.01
µ							100		

Looking at the results for subset 2 we see that the P-RRM model very strongly outperforms the RUM model. The difference in final log-likelihood – which is over 120 LL points – is highly significant. Furthermore, the signs of the parameters are all in the expected directions. The µRRM model indicates that the choice behaviour of the respondents in this subset is best described by very strong regret minimisation behaviour. The regret aversion parameter µ in the µRRM model hits the lower bound, which is set at $\mu = 0.01$ in the estimation, implying that the µRRM model collapses to a P-RRM model. Therefore, also conjecture (2) is supported by these discrete choice analyses. This gives us confidence that the ANN has also successfully identified those travellers whose choice behaviour is best described by RRM. Furthermore, it is encouraging to see that the ratios of the parameter estimates are roughly constant across the subsets. In a RUM modelling framework, the ratios of parameters represent the marginal rate of substitution, which – in this context – translate into VoTs. The implied VoTs by the RUM models in subset 1 and 2 are respectively $^{\beta_{Time}}/_{\beta_{Cost}} = 0.24$ €/minute and $^{\beta_{Time}}/_{\beta_{Cost}} = 0.30$ €/minute. These results provide some support for that the ANN has classified travellers based on decision rules, rather than based on taste heterogeneity – as is a major methodological problem when using a LC discrete choice modelling approach to investigate decision rule heterogeneity (Hess et al., 2012).

Looking at the results for subset 4 we see that the RUM, P-RRM and the μRRM model all fit the data very poorly ($\rho^2 \leq 0.08$). This means that none of these models is able to describe the data generating process meaningfully. The small (and insignificant) taste parameters in all three models confirm this inference. These tell us that these models basically predict rather random choices. Therefore, conjecture (3) is supported.

Finally, we analyse subset 3. The ANN classified 27 respondents to the Lexicographic decision rule. To examine this classification we do not need to estimate discrete choice models. After all, lexicographic behaviour can fairly easily be detected by inspection of the data.[46] As expected, we find that all 25 respondents who consistently chose for the 'fast and expensive' alternative (9) and the 'slow and cheap' alternative (16) are classified as lexicographic by the ANN. In addition, two respondents which chose twice the 'fast and expensive' route and 8 times the 'slow and cheap' route are 'falsely' allocated to this class. Nonetheless, despite these misclassifications, it is very reasonable to say that also the final conjecture (4) is supported. Taken together, these results encourage us to believe that the ANN is well capable to classify travellers in terms of their underlying decision rules.

4.2 Latent class modelling appraoch

Another way to learn about the capability of ANNs to classify travellers in terms of their underlying decision rule involves comparing the classification of the ANN with those obtained from a traditional Latent Class discrete choice modelling approach. To do so, we estimate – in consonance with the ANN – LC discrete choice models with three predefined classes: a RUM, a P-RRM and a Random class.[47] Specifically, we estimate a discrete mixture LC model as well as a so-called mixed-mixed logit model (Keane & Wasi, 2013). This latter model is a discrete mixture of mixed logit models. This model allows for taste heterogeneity within decision rule classes. For the random parameters different distributions are tested (normal, triangular, uniform). Uniform distributions are found to give the best performance in terms of model fit. Furthermore, we have estimated 'generic' sigmas, in the sense that they are shared across decision rule classes, as we find this specification to outperform a specification in which class-specific sigmas are estimated (when taking the model parsimony into account).

Latent class models are estimated using Pythonbiogeme (Bierlaire, 2016). To avoid getting stuck in local maxima, estimations are repeated 20 times using randomly drawn starting values between -1 and 1. For these LC analyses we use subsets 1, 2 and 4. Together these subsets comprise of 79 respondents. Subset 3 – consisting of respondents classified as Lexicographic and also identified as such in Section 2.2 – is excluded from this analysis because discrete choice models are not well-equipped to deal with lexicographic choice behaviour (Hess, Rose, & Polak, 2010).

Table 5 shows the LC estimation results. First we look at the results for the discrete mixture LC model. Looking at the market shares of the decision rules, we see that these are rather similar

[46] Although we acknowledge the extreme preferences may also lead to seemingly lexicographic choice behaviour.

[47] For the sake of completeness we also tested LC models with more and less than 3 classes. 3-class models are found to obtain the lowest BIC values.

to those predicted by the ANN[48] (RUM: 27%, P-RRM: 65%, RND: 8%). Both models find that P-RRM is the most common decision rule in this sample, with between 60% and 70% market share. However, whereas the ANN classified the remaining decision-makers mostly as RUM (27%), in the LC model the Random decision rule attains the second highest market share. Next, we look at the ratios of the parameter estimates, and compare these to those presented in Table 4. The ratios of the time and cost parameter in Table 5 are respectively $\beta_{time}^{RUM}/\beta_{cost}^{RUM} = 0.55$ and $\beta_{time}^{P-RRM}/\beta_{cost}^{P-RRM} = 0.26$, for the RUM and P-RRM class. In Table 4 we however find a ratio of $\beta_{time}^{RUM}/\beta_{cost}^{RUM} = 0.24$ for decision-makers classified as RUM, and a ratio of βtimeP-$\beta_{time}^{P-RRM}/\beta_{cost}^{P-RRM} = 0.30$ for decision-makers classified as P-RRM. Hence, particularly the ratio of the RUM parameters seems 'outside' from what we would expect based on Table 4. This signals that the LC discrete mixture model has also captured taste heterogeneity – aside from decision rule heterogeneity. This exposes the methodological shortcoming of LC models to study decision rule heterogeneity.

Looking at the results for the mixed-mixed logit model we see that accounting for taste heterogeneity within the decision rule classes substantially improves model fit. With regard to the predicted market shares we see a moderate change as compared to the discrete mixture LC model market shares. In particular, the mixed-mixed logit model predicts larger shares for RUM and P-RRM at the expense of the Random decision rule class. Noteworthy, the market shares predicted by the mixed-mixed logit model and those predicted by the ANN are even closer to one another, than are the ones predicted by the discrete mixture LC model and those predicted by the ANN.

Finally, we investigate the consensus between the ANN and mixed-mixed logit model in terms of their classifications of individual respondents. A strong consensus between the two modelling approaches would provide confidence in both methods for their capability to investigate decision rule heterogeneity. To compute the classification probability of individual respondents for the mixed-mixed model, first we simulate the choice probabilities to obtain their expected values. Then, we apply Bayes rule. That is, we compute the likelihood of the model – which in this context is the decision rule – given the observed sequence of choices of each respondent. Respondents are allocated to the decision rule (i.e. the class) with the highest classification probability, just like is done for the ANN classification in Section 3.4.

Table 6 shows the results in a cross-table. The rows contain the ANN classification; the columns contain the LC model classification. The cells on the diagonal show the number of cases in which the same respondents are classified to the same decision rules by both modelling approaches. The off-diagonal cells indicate disagreements between the two modelling approaches in terms of what is the most likely underlying decision rule of a respondent.

[48] Market shares excluding the respondents classified by the ANN as lexicographic

Table 5. LC estimation results.

Subset 1, 2 and 4																		
Model	3-class discrete mixture LC									3-class mixed-mixed logit								
Number of observatio	790									790								
Number of individuals	79									79								
Number of draws	N/A									500								
Null Log-likelihood	-867.9									-867.9								
Final Log-likelihood	-675.8									-617.8								
ρ^2	0.221									0.288								
BIC	1391.6									1289.0								
Decision-rule	*Class 1 RUM [19%]*			*Class 2 P-RRM [61%]*			*Class 3 RND [20%]*			*Class 1 RUM [24%]*			*Class 2 P-RRM [72%]*			*Class 3 RND [4%]*		
Model parameters	*Est*	*Std error*	*t-val*	*Est*	*Std error*	*t-val*	*Est*	*Std error*	*t-val*	*Est*	*Std error*	*t-val*	*Est*	*Std error*	*t-val*	*Est*	*Std error*	*t-val*
β_{cost}	-1.35	0.488	-2.77	-1.87	0.176	-10.65	0	--fixed		-2.84	0.531	-5.35	-1.73	0.210	-8.25	0	--fixed	
β_{time}	-0.74	0.191	-3.87	-0.49	0.047	-10.24	0	--fixed		-0.55	0.095	-5.79	-0.64	0.061	-10.49	0	--fixed	
σ_{cost}										1.55	0.185	8.36						
σ_{time}										0.21	0.106	1.99						
Class allocation parameters																		
	Est	Std error	t-val							Est	Std error	t-val						
s1	0.00	--fixed								0.00	--fixed							
s2	1.17	0.321	3.63							1.12	0.34	3.31						
s3	0.06	0.409	0.15							-1.73	0.787	-2.19						

Table 6. Cross-table ANN and LC classification based on highest classification probability.

		Mixed-mixed logit classification			
		RUM	P-RRM	Random	∑
ANN classification	RUM	15	7	0	22
	P-RRM	3	43	5	51
	Random	0	2	4	6
	∑	18	52	9	79

A number of inferences can be made based on Table 6. Firstly, in the majority of the cases the ANN and the LC model agree on the most likely underlying decision rule: almost 80 per cent of the respondents are allocated to the same decision rule by the ANN and the mixed-mixed logit model. We see a strong agreement between the ANN and mixed-mixed logit classification particularly for the RRM decision rule. Strongest disagreement between the two methods is seen for the Random decision rule. These results show that even though the two methods find very similar market shares for the decision rules, their underlying results can be considerably different. This highlights the added value of having a second method available to investigate decision rule heterogeneity.

All together, we believe that the notion that ANNs can be used to investigate decision rule heterogeneity has convincingly been demonstrated. We have cross-validated the ANN outcomes using two approaches: (1) by estimating discrete choice models based on carefully created subsets of the data and (2) by comparing results with those obtained from LC discrete choice models (discrete mixture and mixed-mixed logit). Both approaches provide strong support for the capability of ANNs to identify underlying decision rules. As such, we believe that ANNs provide a promising addition the toolbox of analysts who wish to investigate decision rule heterogeneity. A potential benefit of ANNs over conventional Latent Class models is that ANNs can be trained to recognise decision rules in the presence of taste heterogeneity. Therefore, and given that ANNs are a sort of LC models on 'steroids', they may be better capable to disentangle decision rule heterogeneity from taste heterogeneity than LC models. However, further research is needed to investigate this conjecture more in depth. A disadvantage of our ANN based approach– as compared to traditional LC modelling – is however that in the latter model the membership function can be used to directly provide insights on what type of traveller is best described by what type of decision rule e.g. in terms of socio-demographic variables. With ANNs this can only be done by means of correlation analysis ex post (due to the fact that the ANN is trained on synthetic data). Furthermore, LC discrete choice models do provide confidence intervals, while ANNs do not.

5 Conclusions and discussion

This study is the first to investigate decision rule heterogeneity amongst traveller using a novel artificial neural networks based approach. We have shown how ANNs can be employed to investigate decision rule heterogeneity amongst travellers. In particular, we have proposed a novel ANN topology which is equipped to deal with the panel structure of SC data and we have shown that the ANN can be trained using synthetic data. Based on the encouraging results we have obtained, we believe that ANNs provide a valuable addition to the toolbox of analysts who wish to investigate decision rule heterogeneity. The substantive contribution is that we enriched the growing body of empirical studies providing evidence for the presence of decision rule heterogeneity amongst travellers.

Finally, we would like to point out several limitations to this study, providing avenues for further research. Firstly, to keep track of our results we have trained our ANN to learn a fairly small number of decision rules (4). In future research ANNs can be trained to learn recognise more types of decision rules, such as Contextual concavity (Kivetz, Netzer, & Srinivasan, 2004), Relative Advantage Maximization (RAM) (Leong & Hensher, 2014), Reference dependent utility maximization (Koszegi & Rabin), Stochastic Satisficing (Gonzalez-Valdes & Raveau, 2018) and models that capture learning effects, such as e.g. Value Learning (Balbontin et al., 2017; McNair, Hensher, & Bennett, 2012). Furthermore, given that our empirical analyses are based on just one data set, it is advisable to repeat these analyses using other data sets. This will provide a richer view on the extent to which ANNs are a valuable tool to investigate decision-rule heterogeneity. Furthermore, new types of ANNs can be developed, possibly inspired by the rapid developments in computer science. Lastly, a well-known limitation of ANNs relates to its black-box nature. Given their complex internal structure, ANNs provide limited insights on underlying causal relations (e.g. by what mechanism does it actually detect the decision-rules?). Future research may be directed to illuminate the black boxes of ANNs (Castelvecchi, 2016). This may help researcher to better understand human decision-making, and perhaps even may lead to the discovery of new decision-rules.

Appendix 5A. Choice tasks in the value-of-time choice experiment

Choice task	Route A		Route B		Route C	
	TT	TC	TT	TC	TT	TC
1	23	6	27	4	35	3
2	27	5	35	4	23	6
3	35	3	23	5	31	4
4	27	4	23	5	35	3
5	35	3	27	6	31	5
6	23	6	27	5	35	3
7	35	3	31	5	23	6
8	27	5	23	6	31	3
9	35	3	31	4	27	6
10	23	6	27	4	35	3

Appendix 5B. Sample statistics

Variable	Sample frequency	Percentage [%] in sample
Gender		
Male	44	42%
Female	45	42%
Missing	17	16%
Age		
18 to 24 yr.	2	2%
25 to 34 yr.	24	23%
35 to 44 yr.	25	24%
45 to 54 yr.	21	20%
55 to 64 yr.	15	14%
65 to 74 yr.	2	2%
Missing	17	16%
Completed education		
No education	0	0%
Elementary school	7	7%
Lower education	5	5%
Middle education	39	37%
Higher education	34	32%
University education	4	4%
Missing	17	16%
Income		
I < €9,400	2	2%
€9,400 ≤ I < €14,700	4	4%
€14,700 ≤ I < €20,600	5	5%
€20,600 ≤ I < €33,500	14	13%
€33,500 ≤ I < €67,000	37	35%
I ≥ €67,000	27	25%
Missing	17	16%

Appendix 5C. Estimation results based on full set

Full data set

Model	RUM MNL			P-RRM MNL			muRRM MNL		
Number of observations	1060			1060			1060		
Number of individuals	106			106			106		
Null Log-likelihood	-1164.5			-1164.5			-1164.5		
Final Log-likelihood	-1123.0			-1128.4			-1118.4		
ρ^2	0.036			0.031			0.040		
Parameters	*Est*	*Std error*	*t-val*	*Est*	*Std error*	*t-val*	*Est*	*Std error*	*t-val*
β_{cost}	-0.64	0.072	-8.85	-0.43	0.053	-8.07	-0.43	0.049	-8.82
β_{time}	-0.16	0.019	-8.58	-0.10	0.013	-7.69	-0.11	0.013	-8.35
μ							1.19	0.511	2.32

References

Alwosheel, A., Van Cranenburgh, S., & Chorus, C. G. (2017). *Artificial neural networks as a means to accommodate decision rules in choice models*. Paper presented at the ICMC2017, Cape Town.

Alwosheel, A., van Cranenburgh, S., & Chorus, C. G. (2018). Is your dataset big enough? Sample size requirements when using artificial neural networks for discrete choice analysis. *Journal of Choice Modelling, 28*, 167-182.

Balbontin, C., Hensher, D. A., & Collins, A. T. (2017). Integrating attribute non-attendance and value learning with risk attitudes and perceptual conditioning. *Transportation Research Part E: Logistics and Transportation Review, 97*, 172-191. doi:https://doi.org/10.1016/j.tre.2016.11.002

Ben-Akiva, M., & Swait, J. (1986). The Akaike likelihood ratio index. *Transportation Science, 20*(2), 133-136.

Bierlaire, M. (2016). PythonBiogeme: A Short Introduction, Technical Report TRANSP-OR 160706.

Bishop, C. M. (1995). *Neural networks for pattern recognition*: Oxford university press.

Boeri, M., & Longo, A. (2017). The importance of regret minimization in the choice for renewable energy programmes: Evidence from a discrete choice experiment. *Energy Economics, 63*, 253-260. doi:http://doi.org/10.1016/j.eneco.2017.03.005

Borysov, S., Lourenço, M., Rodrigues, F., Balatsky, A., & Pereira, F. (2016, 1-4 Nov. 2016). *Using internet search queries to predict human mobility in social events.* Paper presented at the 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC).

Castelvecchi, D. (2016). Can we open the black box of AI? *Nature News, 538*(7623), 20.

Chen, C., Ma, J., Susilo, Y., Liu, Y., & 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*, 285-299.

Chorus, C. G. (2014). Capturing alternative decision rules in travel choice models: a critical discussion. In S. Hess & A. Daly (Eds.), *Handbook of Choice Modelling* (pp. 290-310): Edward Elgar.

Chorus, C. G., & Bierlaire, M. (2013). An empirical comparison of travel choice models that capture preferences for compromise alternatives. *Transportation, 40*(3), 549-562. doi:DOI 10.1007/s11116-012-9444-3

Gonzalez-Valdes, F., & Raveau, S. (2018). Identifying the presence of heterogeneous discrete choice heuristics at an individual level. *Journal of Choice Modelling, 28*, 28-40.

Guevara, C. A., & Fukushi, M. (2016). Modeling the decoy effect with context-RUM Models: Diagrammatic analysis and empirical evidence from route choice SP and mode choice RP case studies. *Transportation Research Part B: Methodological, 93, Part A*, 318-337. doi:http://dx.doi.org/10.1016/j.trb.2016.07.012

Haykin, S. S. (2009). *Neural networks and learning machines* (Vol. 3): Pearson Upper Saddle River.

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

Hess, S., & Chorus, C. G. (2015). Utility Maximisation and Regret Minimisation: A Mixture of a Generalisation. *RASOULI, S. & TIMMERMANS H., Bounded Rational Choice Behaviour: Applications in Transport. Bingley UK: Emerald*, 31-48.

Hess, S., Rose, J. M., & Polak, J. (2010). Non-trading, lexicographic and inconsistent behaviour in stated choice data. *Transportation Research Part D: Transport and Environment, 15*(7), 405-417.

Hess, S., Stathopoulos, A., & Daly, A. (2012). Allowing for heterogeneous decision rules in discrete choice models: an approach and four case studies. *Transportation, 39*(3), 565-591. doi:DOI 10.1007/s11116-011-9365-6

Keane, M., & Wasi, N. (2013). Comparing alternative models of heterogeneity in consumer choice behavior. *Journal of Applied Econometrics, 28*(6), 1018-1045.

Kivetz, R., Netzer, O., & Srinivasan, V. (2004). Alternative models for capturing the compromise effect. *Journal of marketing research, 41*(3), 237-257.

Koszegi, B., & Rabin, M. A model of reference-dependent preferences. 121 (4): 1133–1165, 2006. *Cited on*, 4.

LeCun, Y. A., Bottou, L., Orr, G. B., & Müller, K.-R. (2012). Efficient backprop *Neural networks: Tricks of the trade* (pp. 9-48): Springer.

Leong, W., & Hensher, D. A. (2012). Embedding decision heuristics in discrete choice models: A review. *Transport Reviews, 32*(3), 313-331.

Leong, W., & Hensher, D. A. (2014). Relative advantage maximisation as a model of context dependence for binary choice data. *Journal of Choice Modelling, 11*, 30-42.

Maren, A. J., Harston, C. T., & Pap, R. M. (2014). *Handbook of neural computing applications*: Academic Press.

McNair, B. J., Hensher, D. A., & Bennett, J. (2012). Modelling heterogeneity in response behaviour towards a sequence of discrete choice questions: a probabilistic decision process model. *Environmental and Resource Economics, 51*(4), 599-616.

Mohammadian, A., & Miller, E. (2002). Nested Logit Models and Artificial Neural Networks for Predicting Household Automobile Choices: Comparison of Performance. *Transportation Research Record: Journal of the Transportation Research Board, 1807*, 92-100. doi:doi:10.3141/1807-12

Omrani, H., Charif, O., Gerber, P., Awasthi, A., & Trigano, P. (2013). Prediction of Individual Travel Mode with Evidential Neural Network Model. *Transportation Research Record: Journal of the Transportation Research Board, 2399*, 1-8. doi:doi:10.3141/2399-01

Payne, J. W., Bettman, J. R., & Johnson, E. J. (1993). *The adaptive decision maker*: Cambridge University Press.

Rouwendal, J., de Blaeij, A., Rietveld, P., & Verhoef, E. (2010). The information content of a stated choice experiment: A new method and its application to the value of a statistical life. *Transportation Research Part B: Methodological, 44*(1), 136-151. doi:http://dx.doi.org/10.1016/j.trb.2009.04.006

Van Cranenburgh, S., Guevara, C. A., & Chorus, C. G. (2015). New insights on random regret minimization models. *Transportation Research Part A: Policy and Practice, 74*(0), 91-109. doi:http://dx.doi.org/10.1016/j.tra.2015.01.008

van Cranenburgh, S., Guevara, C. A., & Chorus, C. G. (2015). New insights on random regret minimization models. *Transportation Research Part A: Policy and Practice, 74*, 91-109.

Van Cranenburgh, S., Prato, C. G., & Chorus, C. (2015). Accounting for variation in choice set size in Random Regret Minimization models: working paper.

Van Cranenburgh, S., Rose, J. M., & Chorus, C. G. (2018). On the robustness of efficient experimental designs towards the underlying decision rule. *Transportation Research Part A: Policy and Practice, 109*, 50-64. doi:https://doi.org/10.1016/j.tra.2018.01.001

van Lint, J. W. C., Hoogendoorn, S. P., & van Zuylen, H. J. (2005). Accurate freeway travel time prediction with state-space neural networks under missing data. *Transportation Research Part C: Emerging Technologies, 13*(5–6), 347-369. doi:http://dx.doi.org/10.1016/j.trc.2005.03.001

Vlahogianni, E. I., Park, B. B., & van Lint, J. W. C. (2015). Big data in transportation and traffic engineering. *Transportation Research Part C: Emerging Technologies, 58, Part B*, 161. doi:http://dx.doi.org/10.1016/j.trc.2015.08.006

Wong, M., Farooq, B., & Bilodeau, G. A. (2017). *Latent behaviour modelling using discriminative restricted Boltzmann machines.* Paper presented at the ICMC 2017, Cape Town.

Conclusions, implications and future research

This thesis has proposed solutions to improve the usefulness of ANNs for human choice analysis. The main research goal is to *explore the potentials and limitations of using ANNs for analysing choice behaviour, and to learn from classical ANN application fields (particularly computer vision) about how ANN-based methods can be improved to increase their usefulness in analysing human choice behaviour*. The first study investigated the sample size required for successful and reliable ANN training (Chapter 2). The next two studies were conducted to use classical ANN application fields (i.e. computer vision) and to learn techniques and strategies for illuminating ANNs' black-box from them (Chapter 3 and 4). The last study proposed an ANN-based solution to address the decision rule heterogeneity (Chapter 5). Below, the main findings of this thesis are summarised, each set out under a research sub-goal. Following that, an overall conclusion of this thesis is drawn. Then, policy and strategy implications are derived. Finally, this chapter closes with directions and recommendations for future research.

1 Conclusions of study 1: Sample size requirements when using ANNs for choice behaviour analysis

Research sub-goal no. 1: To investigate the minimum sample size required for reliable implementation of ANNs for choice behaviour analysis

The first study is devoted to investigating sample size requirements in the context of choice behaviour analysis. The ANNs' sample size requirements have been studied extensively through statistical learning theory, leading to a series of theoretical recommendations regarding what is the minimum number of observations needed. However, those recommendations are of limited use in practice. As such, this research studied the ANNs' sample size requirements empirically using synthetic and real datasets. More specifically, synthetic datasets, with data generating processes of different levels of complexity and randomness, were used. Additionally, the sample size requirements for ANN-based choice behaviour analysis were studied using several real datasets that have been widely reported in literature about choice behaviour analysis. To assess whether the trained ANNs were trained using data that was sufficient in size, the concept of the learning curve was used in the context of out-of-sample data. Based on the analyses on synthetic and real datasets, the following conclusions are drawn:

- Sample size requirements based on the widely adopted rule (i.e. minimum sample size should be 10 times the number of parameters in the network) is not sufficient for reliable and trustworthy ANN in the context of choice behaviour analysis.
- Instead, this study proposes a more conservative rule-of-thumb: sample size should be at least 50 times the number of parameters in the network.
- It is advised to use an increasingly larger sample size for training ANNs as the complexity of the underlying data generating process increases.
- Also, using a larger sample size for training ANNs is needed as noise levels in the data decreases.
- Although ANNs need a much larger sample size for training (compared to discrete choice models), the required sample size is still within the range of most existing datasets in the field of choice modelling.

2 Conclusions of study 2: Diagnosis of ANN-based choice behaviour analysis using prototypical examples

Research sub-goal no. 2: To develop a diagnostic method for trained ANN models

This study focused on one of the main criticisms against the use of ANNs for choice behaviour analysis: the lack of trust in ANNs' results caused by the ANNs' black-box nature. That is, although ANNs' predictive power is strong, their opaque nature make it difficult to understand the rationale behind these predictions. As such, the study pioneered a computationally cheap and easy-to-use method to diagnose ANNs in the context of choice behaviour analysis. Inspired by research from the field of computer vision, the proposed method involves synthesising prototypical examples (using the activation maximisation method) after having trained the ANN. These prototypical examples expose the fundamental relationships that the ANN has learned, which can be evaluated by the analyst to inspect the trained ANN rationale. This study discussed how the synthesised prototypical examples can be used in the context of choice data and presented the practical considerations needed for successfully diagnosing ANNs. The main findings of this study were cross-validated using techniques from traditional discrete choice analysis. Based on the analyses and results, the following conclusions are drawn:

- Synthesising prototypical examples is an effective tool to understand the rationale of ANNs trained to analyse mode choice behaviour.
- The ANN black-box puzzle has not been completely solved by the proposed method. As such, we still believe that the natural domain for using ANNs remains in forecasting (as opposed to economic inference, for example). The synthesised prototypical examples help the analyst to determine whether or not to trust predictions made by the trained ANN.
- While the activation maximisation method has been proven to be useful to validate a trained ANN, it also has the potential to be used to provide a better (and perhaps new) understanding of human choice behaviour and/or how ANNs deal with choice

behaviour data.[49] By having an ANN trained with a large choice behaviour data, the network may observe patterns in the data that are hard for analysts. As such, by applying the activation maximisation method to ANN nodes (either output or hidden nodes), we can extract this distilled knowledge from the network in order to acquire new insights.[50]

3 Conclusions of study 3: Explaining the predictions of ANNs-based choice behaviour analysis

Research sub-goal no. 3: To develop a method to explain individual predictions made by trained ANNs

This study also addressed the ANNs' black-box issue, but from a different perspective. Rather than deriving a global understanding of the rationale of trained ANNs (as in study 2), this study re-conceptualised and pioneered the use of heat maps to explain the (individual) predictions of ANNs in the context of (travel) choice behaviour analysis. This study showed that the approach that is often used to explain ANNs' predictions (i.e. sensitivity analysis) is inappropriate for this purpose. Then, we showed how heat maps (generated using the Layer-wise Relevance Propagation (LRP) method) can be applied to provide an explanation for the trained ANNs' predictions; hence allowing an analyst to build trust in predictions. Furthermore, we showed how using heat maps to explain multiple, carefully selected predictions (using guidance from traditional discrete choice models) can be applied to build trust in the trained ANN as a whole. Based on our results, the following conclusions are drawn:

- Sensitivity analysis-based approaches are ill-suited for explaining the predictions of individual ANNs.
- Heat map generation is an effective tool to understand the rationale of the individual predictions of ANNs, trained to analyse mode choice behaviour; hence, helping the analyst to establish trust in the predictions.
- Heat maps can also be helpful in establishing trust in the trained ANN as a whole, by explaining (and establishing trust in) several carefully selected predictions.
- Heat maps were generated using a single re-distribution rule of the LRP method. As such, there is room to explore (and develop) alternative rules to better explain the predictions of ANNs in the context of choice behaviour analysis.

4 Conclusions of study 4: An ANN-based approach to investigate decision rule

Research sub-goal no. 4: To investigate the capabilities of ANNs to capture the decision rules heterogeneity

This study investigated the decision rules heterogeneity amongst decision-makers. Decision rule is the process used by a decision-maker to evaluate alternatives and determine a choice.

[49] Knowledge extraction using machine learning techniques is attracting increasing attention, particularly in the field of physics, see (Iten, Metger, Wilming, Del Rio, & Renner, 2020) for a recent example.

[50] Note that here, we do not mean to use the proposed method for validation and knowledge extraction simultaneously.

The majority of discrete choice models maintain the assumption of homogeneous decision rule. Yet, in recent years, the fact that travellers are heterogeneous in terms of their decision rules has attracted an increasing amount of attention within the choice modelling community. It is also acknowledged that insights into decision rule heterogeneity are crucial for understanding and predicting choice behaviour. To investigate decision rule heterogeneity, the analysts in the choice modelling community rely on latent class (LC) choice models. However, a major limitation of the LC models is their inability to distinguish the decision rule heterogeneity from taste heterogeneity. This research tackled this issue by proposing a novel ANN topology that can be trained using synthetic data, and then used to recognise the decision rule(s) of certain stated choice experiments. The proposed approach was trained so that four distinct decision rules can be recognised. The main findings of this study were cross-validated by comparing the results with those of traditional discrete choice models. Based on the analyses and results, the following conclusions are drawn:

- The proposed ANN-based solution can distinguish decision rule heterogeneity without confounding it with taste heterogeneity. As such, it provides a valuable tool for choice behaviour analysts who want to investigate decision rule heterogeneity.
- This study provided an example of how ANNs' flexibility can be capitalised by incorporating behavioural knowledge into the structure of ANNs (hence, leading to a behaviourally-informed topology). Such a powerful approach can be used further to solve choice behaviour problems.[51]
- This study showed empirically how synthesised data can be used to overcome the issue of small training data sets (or in some cases the unavailability of training data) in the context of choice behaviour analysis.

5 Overall conclusions

This thesis contributed to the existing research about facilitating the use of ANNs for choice behaviour analysis by 1) capitalising and improving the use of ANNs; 2) illuminating the black-box of ANNs. In addition to the conclusions associated with each of the four individual studies, several overall points are worth mentioning here:

- With regard to the ANN black-box issue, despite the effort put into this research to illuminate the ANN black-box (particularly in Chapters 3 and 4), we are far from achieving models that are fully interpretable and explainable. As such, the foremost insight derived from this thesis is that most natural domain of how ANNs can be used remains in forecasting (e.g. traffic prediction) and recommendation (e.g. travel time departure) applications, where complete model transparency is not a must. It is however worth noting that illuminating the ANN black-box is the focus of a growing number of research in a variety of contexts (e.g. image processing in medical field). Thus, in the near future we may witness the birth of a solution (or solutions) to this limitation. Having said that, however, one caveat must be noted. We do not expect these solutions to make ANNs as transparent as discrete choice models, but transparent enough to be used in many choice behaviour analysis problems.
- So-called shallow ANN architecture (i.e. three layers network) has been found to perform well on most datasets encountered throughout the PhD project. These datasets represent different problems with different levels of complexity (e.g.

[51] An example of this can be found in (Sifringer, Lurkin, & Alahi, 2018), where an ANN structure to form the utility based choice model was proposed.

different number of independent variables).[52] We did not find that using deeper ANN architecture led to achieving better prediction performance, particularly in travel mode choice problems. It is however important to highlight that alternative topologies can be beneficial for solving some aspects of choice behaviour analysis (as shown in Chapter 5).

- A major difference between ANNs and choice behaviour communities lies in the culture of openness and sharing. That is, in ANNs community (and machine learning in general), researchers and practitioners tend to make their codes (and datasets) publicly available. We think this culture has contributed to: 1) accelerate the development of machine learning methods and techniques; 2) help researchers to avoid re-inventing the wheel (i.e. by collecting new data to validate a hypothesis); and 3) improve the accuracy of research.[53]
- For most of the stated preference datasets encountered throughout this PhD project, we found that the prediction performance of ANN-based models is not much better than the prediction performance of traditional DCMs.[54] This is in contrast to revealed preference data, where using ANNs usually yields much better prediction performance.

6 Policy and strategy implications

The methodological contributions of this thesis have policy and strategy implications for decision-makers in a variety of domains. However, as the datasets and developed methods used are all in the context of transportation, this section follows suit and investigates the policy and strategy implications in the context of transportation.[55] In particular, we investigate the policy and strategy implications from the angle of disruption, due to the nature of ANNs and machine learning methods (i.e. in many domains, machine learning methods brush aside the approach they replace, mainly because of their superior prediction performance). To do so, we list the main transportation applications in which human choice behaviour is widely analysed. Then, we assess to what extent each of the listed applications is vulnerable to disruption by ANNs in general and the results of this thesis in particular.

Indeed, it is impossible to create an exhaustive list of all types of transportation applications within the available space. Instead, we cover the spectrum by classifying the transportation applications into three main types, as follows:

- Type I: Recommendation: where the objective is to suggest items of interest to a user from a much larger set to overcome the issue of information overload (Danaf, Becker, Song, Atasoy, & Ben-Akiva, 2019). Recommendation services are primarily offered for commercial application (e.g. recommending Netflix movies). In the transportation

[52] For instance, the analyses of 15 (synthesised and real) datasets were reported in this thesis. In addition, we have not reported the analyses of over a dozen datasets.

[53] The culture of openness and sharing has also started to be adopted in the community of choice behaviour analysis. For instance, codes and data of Random Regret Minimisation Models (van Cranenburgh, Guevara, & Chorus, 2015) can be found at: https://www.advancedrrmmodels.com/. Also, a Python library that allows for developing discrete choice models with machine learning was developed by (Wong & Farooq, 2020) and can be found at: https://github.com/mwong009/genome.

[54] It is possible that this is caused by the optimisation of choice experiments for certain decision rules, as observed by (van Cranenburgh, Rose, & Chorus, 2018).

[55] This does not mean that the implications are only confined to the transportation domain, but they are also extendable to other fields.

context, recommendation services are increasingly used by private companies, to recommend to travellers a mode(s) or time(s) of travel, for example.

- Type II: Prediction: where the goal is to predict aspects of transportation that are useful to both analysts and users. Prediction services can be offered by private companies (e.g. provide short-term traffic prediction based on the analysis of GPS data) or public sector (e.g. 10 years transportation demand forecasting).
- Type III: Policy assessment and evaluation: where the goal is to learn aspect(s) of transportation that can be used (by analysts) to evaluate and assess policies. In the transportation context, discrete choice models have been widely used for this purpose. For example, discrete choice models are used for travel time valuation, which is an important input to the cost-benefit analysis of publicly funded transportation projects (Mackie, Jara-Dıaz, & Fowkes, 2001).

For each of these types, we list three examples (and the owner or actor of each example) that we believe are most common. To assess the application vulnerability to disruption, two factors are considered: the data availability (i.e. whether an abundance of cheap data is at the analyst's disposal or not)[56] and the level of transparency required (i.e. high, medium or low level). Table 1 shows the application types, the examples and the factors considered.

Table 1. Main types of transportation applications

Application type	Example	Actor/owner	Data availability	Required level of transparency	Readiness for disruption
Recommendation	Travel route	Private sector (e.g. Google Maps)	High	Low	High
	Departure time	Private sector (e.g. Google Maps)	High	Low	High
	Travel mode(s)	Private sector (e.g. SkedGo)	High	Low	High
Prediction	Short term traffic flow	Private sector (e.g. Google Maps)	High	Low	High
	Travel time (duration)	Private sector (e.g. Google Maps)	High	Low	High
	Demand for new transportation services	Public sector (e.g. transportation authority)	Low	High	Low
Policy assessment and evaluation	Welfare policies	Public sector (e.g. transportation authority)	Medium	High	Low
	Travel time valuation	Public sector (e.g. transportation authority)	Medium	High	Medium

[56] We would like to acknowledge that with the recent shift toward privately owned data, the data ownership and accessibility may end up being one of the main challenges facing the adoption of data-driven decision making.

	New infrastructure projects	Public sector (e.g. transportation authority)	Low	High	Low

Several observations can be made, based on Table 1. First, the readiness for disruption can be formulated as a function of data availability and the required level of transparency. That is, a particular application is more prone to disruption by ANNs (and machine learning methods in general) as more data are available and a lower level of transparency is required. For instance, to recommend a travel route (as many private companies such as Google Maps do), plenty of data are available for training and validating models. Further, as the cost of giving an incorrect recommendation is low, the required level of model transparency is low. Hence, this application is scored high for readiness for disruption (actually, data-driven models are already widely adopted). This is in contrast to applications such as the prediction of the demand for a new transportation service, where data are scarce (i.e. very little data is collected using stated preference choice experiments) and model transparency is a prerequisite for decision-makers (because a publicly funded transportation project depends on this prediction, for example). Second, among the types of applications, recommendation applications are highly prone to disruption, mainly because of the abundance of data available and the low level of transparency required. This is in contrast to policy evaluation and assessment applications, where the data scarcity and the required level of transparency make these applications less vulnerable to disruption caused by ANNs. Third, for some applications (e.g. predicting the demand for a new transportation service), the results of this thesis (particularly the techniques presented in Chapters 3 and 4) can be used to increase the level of transparency to the meet the requirements. This can be achieved by explaining the prediction of carefully selected observations, for example, or synthesising prototypical examples for the transportation service of interest.

7 Recommendations for future research

Several Chapters in this thesis concluded with a description of research directions that were deemed worth pursuing in the future (Chapters 3, 4 and 5). These will not be reiterated here. However, the following interesting directions, can be added to the list:

- Throughout this thesis, we extensively used multiple synthesised and real datasets (most real datasets were collected in the context of transportation). A strategic decision, made at the beginning of this PhD project, was to only use real datasets that are accessible to researchers (hence some datasets had already been used more than once).[57] In contrast to the tradition in the machine learning community, there is a limited number of "freely and easily accessible" datasets in the community of choice modelling. As has already been shown in many fields (e.g. medical research), benchmark datasets that are freely and easily accessible are important for researchers to objectively measure the performance of their models. As such, one of the main recommendations here is to encourage researchers and choice behaviour analysts to develop (discrete) choice behaviour benchmark datasets (both stated and revealed preference data with different levels of complexity). We believe this step is essential for both method development and performance assessment, particularly in the context of using emerging machine learning models.[58]

[57] As part of the Findable, Accessible, Interoperable and Reusable (FAIR) data initiative at Delft University of Technology.

[58] For the interested researcher, a list of choice behaviour datasets can be found at: https://biogeme.epfl.ch/data.html.

- With regard to illuminating the ANN black-box (the studies in Chapters 3 & 4), two different approaches were used separately (activation maximisation to provide a global view of model rationale and Layer-wise Relevance Propagation to explain individual predictions). A promising line of research would be to use the two approaches together to understand what each neuron/layer of the network has learned and how each neuron/layer works, which may lead to discovering new relations or developing a new understanding of human choice behaviour.[59] For instance, in the context of computer vision, multiple approaches to illuminate the black-box were applied to understand how ANNs see the world. An interesting finding was that the individual neurons extract features in more of a local manner, rather than a distributed manner, with each neuron corresponding to a specific pattern (Qin, Yu, Liu, & Chen, 2018).
- The main stream of ANN use for choice behaviour analysis focused on searching for the parameter values of an ANN architecture (set by an analyst) with the goal of maximising (as far as possible) the prediction performance. While this approach has been undoubtfully useful, a promising line of research is to shift the focus toward searching for ANN architectures that incorporate the knowledge of the choice modelling domain (e.g. the fourth research sub-goal of this thesis has led to developing a behaviourally-informed ANN architecture). Pursuing this line of research may lead to finding an architecture that approximates a certain decision rule(s), for example, or discovering ANN layers/activation functions that are more appropriate for choice behaviour problems (e.g. convolutional networks are found to be specially suited for image processing (Cohen & Shashua, 2016). This direction of research was influenced by the research on evolutionary computing (e.g. (Yao & Liu, 1998)), and there are recent promising developments (see (Gaier & Ha, 2019), for example).
- Similar to traditional machine learning, independent variables in the field of choice behaviour analysis are typically pre-specified and engineered by analysts. However, the breakthrough of ANN-based models (i.e. convolution neural networks) in the field of computer vision (e.g. image recognition tasks) is due to the fact that independent variables are not engineered by experts, but rather learned automatically from previous examples (i.e. end-to-end learning). As such, an interesting future line of research would be to follow course and seek to automate (or train a model to learn) the task of selecting and processing independent variables (see (Pereira, 2019) for a recent example in this direction).
- In this thesis, the main research line was to learn techniques/methods from classic ANNs fields to improve the usefulness of ANNs for choice behaviour analysis. It is worth noting, however, that the field of choice modelling has much to offer to the machine learning community (due to its rich theories on decision-making). For instance, in a reinforcement learning context (i.e. an area of machine learning concerned with how agents ought to take action in an environment in order to maximise the notion of cumulative reward), choice behaviour theories have the potential to enrich this field by incorporating the behaviour theories in the model's reward function to increase the agent's moral awareness, for example.
- In the field of choice behaviour analysis, the use of machine learning methods has been dominated by the so-called supervised learning approach (i.e. the machine learning task of learning a function that maps an input to an output, based on

[59] Note that there is a growing interest in knowledge discovery using ANNs (and other machine learning methods) in various fields (e.g. see (Iten, Metger, Wilming, Del Rio, & Renner, 2020) in physics).

example input-output pairs). Alternative learning methods (i.e. unsupervised and reinforcement learning) have been shown to be very useful in a variety of contexts (e.g. unsupervised learning to reduce input dimensionality or to discover rules that describe data). We believe that these alternatives have the potential to also be used in the contexts of choice behaviour. For example, an analogy between agents in reinforcement learning (that take actions to maximise rewards) and utility maximisers can be established and modelled. Doing so may open up new pathways to enrich and increase our understanding of human choice behaviour.

- Lastly, we believe developing concrete definitions and measures of descriptive words such as transparent, trustable and interpretable is a very important direction for future work. Currently, these descriptive words are widely used but not accurately defined (Doshi-Velez & Kim, 2017; Lipton, 2016; Rosenfeld & Richardson, 2019). We believe that doing this research (to develop comprehensive definitions and measures) should be interdisciplinary, cutting across the various disciplines (e.g. computer science, social science).

References

Danaf, M., Becker, F., Song, X., Atasoy, B., & Ben-Akiva, M. (2019). Online discrete choice models: Applications in personalized recommendations. *Decision Support Systems, 119*, 35-45.

Doshi-Velez, F., & Kim, B. (2017). Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*.

Iten, R., Metger, T., Wilming, H., Del Rio, L., & Renner, R. (2020). Discovering physical concepts with neural networks. *Physical Review Letters, 124*(1), 010508.

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

Mackie, P. J., Jara-Dıaz, S., & Fowkes, A. (2001). The value of travel time savings in evaluation. *Transportation Research Part E: Logistics and Transportation Review, 37*(2-3), 91-106.

Pereira, F. C. (2019). Rethinking travel behavior modeling representations through embeddings. *arXiv preprint arXiv:1909.00154*.

Qin, Z., Yu, F., Liu, C., & Chen, X. (2018). How convolutional neural network see the world-A survey of convolutional neural network visualization methods. *arXiv preprint arXiv:1804.11191*.

Rosenfeld, A., & Richardson, A. (2019). Explainability in human–agent systems. *Autonomous Agents and Multi-Agent Systems*, 1-33.

Sifringer, B., Lurkin, V., & Alahi, A. (2018). *Enhancing Discrete Choice Models with Neural Networks.* Paper presented at the hEART 2018–7th Symposium of the European Association for Research in Transportation conference.

van Cranenburgh, S., Guevara, C. A., & Chorus, C. G. (2015). New insights on random regret minimization models. *Transportation Research Part A: Policy and Practice, 74*, 91-109.

van Cranenburgh, S., Rose, J. M., & Chorus, C. G. (2018). On the robustness of efficient experimental designs towards the underlying decision rule. *Transportation Research Part A: Policy and Practice, 109*, 50-64.

Wong, M., & Farooq, B. (2020). A bi-partite generative model framework for analyzing and simulating large scale multiple discrete-continuous travel behaviour data. *Transportation Research Part C: Emerging Technologies, 110*, 247-268.

Summary

For decades, Discrete Choice Models (DCMs) have been used to describe, understand and predict human choice behaviour in a wide variety of contexts including transportation, healthcare and marketing. The field of discrete choice modelling is firmly rooted in economic theory, and most DCMs are based on the assumption that decision-makers, when asked to select an alternative among a set of presented alternatives, make deliberate trade-offs by employing a stable function to assign utility to each alternative, and then select the alternative with the highest utility.

There is no doubt that DCMs enjoy popularity across a wide range of fields. This popularity can be attributed to the fact that DCMs offer a transparent and tractable modelling approach that is deeply rooted in theory. However, it has been highlighted in the literature that the imposed assumptions may lead to restrictive analysis of human choice behaviour, resulting in biased parameter estimates, lower predictability and incorrect interpretations. As such, a recent shift is being made in the choice modelling community to include behavioural and psychological factors and theories that were traditionally ignored. As a result, a wide range of new models that incorporate behavioural and psychological theories have been developed. However, a common feature of all DCMs – traditional and new – is that they are “theory-oriented”, in the sense that assumptions are imposed a priori.

Another way to learn about human choice behaviour can be achieved using approaches that are less theory-reliant and more flexible than discrete choice models. In particular, Artificial Neural Networks (ANNs) surface as an appealing alternative that have gained increasing interest in a wide set of applications. ANNs are mathematical models that are loosely inspired by structural and functional aspects of biological neural systems, and are well-known for being highly effective in solving complex classification and regression problems. Their recent uptake can be attributed to major breakthroughs in ANN research, affecting the daily lives of many people (e.g. in the context of self-driving vehicles, enabling them to recognise traffic signs and navigate routes in complex environments). In particular, the fact that ANNs have the ability to automatically learn and improve from experience (i.e. previous examples), without being

explicitly programmed, allows them to achieve impressive results, in some cases better than human experts' performance.

However, despite the excitement about the potential of ANN for choice behaviour analysis, many choice behaviour analysts are reluctant to use ANN models mainly because their superior prediction performance comes at a cost, this being increasing the complexity of ANNs to a level that makes their reasoning a mystery (i.e. the black-box issue). This leaves the analysts in the dark about whether ANN predictions are based on intuitively correct and expected rationale or not. Without sufficient understanding of how and why a model makes predictions, choice behaviour analysts remain unsure about the extent to which they can trust the trained ANN. As such, the use of ANNs is mainly confined to niche settings where prediction performance is highly valued (e.g. travel route recommendations) and model transparency is not of great importance. However, for many applications of choice behaviour analysis (e.g. a cost-benefit analysis of publicly funded projects), model transparency is considered a prerequisite for justifiable reasons (e.g. transparent governance).

Considering the mentioned advantages and limitations of using ANNs to analyse choice behaviour, this thesis aims to explore the potentials and limitations of using ANNs for analysing choice behaviour, and to learn from classical ANN application fields (particularly computer vision field) about how ANN-based methods can be improved to increase their usefulness in analysing human choice behaviour.

In chapter 2, we investigate the sample size requirements when using ANNs for discrete choice analysis. For reliable and trustworthy ANNs, the dataset (on which the ANN is estimated/trained) needs to be sufficiently large (i.e. consist of a sufficient number of observations). Compared to their counterpart statistical models (e.g. DCMs), ANNs are known for consuming datasets for training, that are larger in size. The ANNs' sample size requirements have been studied extensively through statistical learning theory, leading to a series of theoretical recommendations regarding what is the minimum number of observations needed. However, those recommendations are of limited use in practice. As such, this research studied the ANNs' sample size requirements empirically using several synthetic and real datasets. Based on our analyses, sample size requirements based on the widely adopted rule (i.e. minimum sample size should be 10 times the number of parameters in the network) is not sufficient for reliable ANN in the context of choice behaviour analysis. Instead, this study proposes a more conservative rule-of-thumb: sample size should be at least 50 times the number of parameters in the network. Also, we find that the ANN requires more data as the complexity of the underlying data generating process increases and its noisiness decreases.

Chapter 3 focuses on the lack of trust in ANNs' results caused by the ANNs' black-box nature. That is, although ANNs' predictive power is strong, their opaque nature make it difficult to understand the rationale behind these predictions. As such, this study pioneers a computationally cheap and easy-to-use method to diagnose ANNs in the context of choice behaviour analysis. Inspired by research from the field of computer vision, the proposed method involves synthesising prototypical examples (using the activation maximisation method) after having trained the ANN. These prototypical examples expose the fundamental relationships that the ANN has learned, which can be evaluated by the analyst to inspect the trained ANN rationale. This study discusses how the synthesised prototypical examples can be used in the context of choice data and presents the practical considerations needed for successfully diagnosing ANNs. The main findings of this study were cross-validated using techniques from traditional discrete choice analysis.

Moving forward, chapter 4 also addresses the ANNs' black-box issue, but from a different perspective. Rather than deriving a global understanding of the rationale of trained ANNs, this study re-conceptualises the use of heat maps to explain the (individual) predictions of ANNs in the context of (travel) choice behaviour analysis. This study shows that the approach that is often used to explain ANNs' predictions (i.e. sensitivity analysis) is inappropriate for this purpose. Then, we show how heat maps (generated using the Layer-wise Relevance Propagation (LRP) method) can be applied to provide an explanation for the trained ANNs' predictions; hence allowing an analyst to build trust in predictions. Furthermore, we show how using heat maps to explain multiple, carefully selected predictions (using guidance from traditional discrete choice models) can be applied to build trust in the trained ANN as a whole.

In chapter 5, we investigate the decision rules heterogeneity amongst decision-makers. Decision rule is the process used by a decision-maker to evaluate alternatives and determine a choice. The majority of discrete choice models maintain the assumption of homogeneous decision rule. Yet, in recent years, the fact that travellers are heterogeneous in terms of their decision rules has attracted an increasing amount of attention within the choice modelling community. It is also acknowledged that insights into decision rule heterogeneity are crucial for understanding and predicting choice behaviour. To investigate decision rule heterogeneity, the analysts in the choice modelling community rely on latent class (LC) choice models. However, a major limitation of the LC models is their inability to distinguish the decision rule heterogeneity from taste heterogeneity. This research tackle this issue by proposing a novel ANN topology that can be trained using synthetic data, and then used to recognise the decision rule(s) of certain stated choice experiments. The proposed ANN-based solution can distinguish decision rule heterogeneity without confounding it with taste heterogeneity. As such, it provides a valuable tool for choice behaviour analysts who want to investigate decision rule heterogeneity. The main findings of this study were cross-validated by comparing the results with those of traditional discrete choice models.

In conclusion, this thesis has contributed to the growing research about using (and facilitating the use of) ANNs for choice behaviour analysis. It has been able to do so by developing methods and strategies to increase the choice behaviour analysts' trust in ANN-based models and their deliverables (e.g. by alleviating the black-box). Furthermore, we show how ANNs can be used to solve aspects of choice behaviour analysis that are deemed difficult to discrete choice models. Hence, the contributions of this thesis motivate future research into leveraging ANNs capabilities to derive better understanding and prediction of human choice behaviour.

Samenvatting

Decennialang zijn discrete keuzemodellen (DKM) gebruikt om het menselijke keuzegedrag te beschrijven, begrijpen en voorspellen in een wijde variëteit aan contexten zoals transport, medische zorg en marketing. Het onderzoeksveld van DKM vindt zijn oorsprong in de economische theorie en de meeste DKM zijn op de assumptie gebaseerd dat besluitvormers, wanneer men deze vraagt om een keuze te maken tussen een set van alternatieven, weloverwogen afwegingen maken door middel van het gebruiken van functies waarin het nut voor elk alternatief is toebedeeld. De keuze wordt dan gemaakt door het alternatief te kiezen met het maximale nut.

Er heerst geen twijfel dat DKM erg populair is en wordt gebruikt in een wijde range aan toepassingsgebieden. Deze populariteit is te danken aan het feit dat DKM een transparante en volgzame modelleertechniek is met een diepgewortelde theorie. Echter, in de literatuur is benadrukt dat de opgelegde aannames kunnen leiden tot een restrictieve analyse van menselijk keuzegedrag, resulterend in bevooroordeelde parameterschattingen, lagere voorspelbaarheid en onjuiste interpretaties. Als zodanig vindt er een recente verandering plaats in de gemeenschap van keuzemodellen om gedrags- en psychologische factoren en theorieën op te nemen die traditioneel werden genegeerd. Als gevolg hiervan is een breed scala aan nieuwe modellen ontwikkeld die gedrags- en psychologische theorieën incorporeren. Een gemeenschappelijk kenmerk van alle DKM - traditioneel en nieuw - is echter dat ze 'theoriegericht' zijn, in de zin dat aannames a priori worden opgelegd.

Een andere manier om meer te leren over menselijk keuzegedrag kan is door middel van benaderingen die minder theorie-afhankelijk en flexibeler zijn dan DKM. Met name kunstmatige neurale netwerken (KNN) komen naar voren als een aantrekkelijk alternatief dat steeds meer aandacht heeft gekregen in een wijde range aan toepassingen. KNN zijn wiskundige modellen die geïnspireerd zijn door structurele en functionele aspecten van biologische neurale systemen, en staan erom bekend dat ze zeer effectief zijn bij het oplossen van complexe classificatie- en regressieproblemen. Hun recente introductie kan worden toegeschreven aan grote doorbraken in KNN-onderzoeken, die het dagelijks leven van veel mensen beïnvloeden (bijv. in de context van zelfrijdende voertuigen, welke verkeersborden kunnen herkennen en zich door routes in complexe omgevingen weten te navigeren). Met name het feit dat KNN de mogelijkheid hebben om automatisch te leren en zich kunnen verbeteren uit ervaring (ofwel eerdere voorbeelden), zonder hiervoor expliciet te zijn geprogrammeerd, stelt hen in staat indrukwekkende resultaten te behalen, in sommige gevallen beter dan de prestaties van menselijke experts.

Ondanks de hoge verwachtingen over de potentie van KNN voor analyse van keuzegedrag, zijn veel analisten van keuzegedrag terughoudend om KNN-modellen te gebruiken, vooral omdat de superieure voorspellingsprestaties nadelen met zich meebrengen, waardoor de complexiteit van KNN toeneemt tot een niveau dat hun redenering tot een mysterie maakt, de zogenaamde black-box. Dit laat de analisten in het ongewisse of de voorspellingen van KNN zijn gebaseerd op intuïtief correcte en verwachte rationale of niet. Zonder voldoende inzicht te hebben in hoe en waarom een model voorspellingen doet, blijven analisten van keuzegedrag onzeker over de mate waarin ze de getrainde KNN kunnen vertrouwen. Als zodanig is het gebruik van KNN voornamelijk beperkt tot niche-gevallen waar hoge waarde aan voorspellingsprestaties wordt gehecht en modeltransparantie van minder groot belang is (bijv. aanbevelingen voor reisroutes). Echter, voor veel toepassingen van analyse van keuzegedrag (bijv. een kosten-batenanalyse van door de overheid gefinancierde projecten) wordt modeltransparantie om redenen van rechtvaardiging als een voorwaarde beschouwd, namelijk transparant bestuur.

Gezien de genoemde voordelen en beperkingen van het gebruik van KNN om keuzegedrag te analyseren, is dit proefschrift bedoeld om de mogelijkheden en beperkingen van het gebruik van KNN voor het analyseren van keuzegedrag te onderzoeken. Het doel van de thesis om te leren van de klassieke KNN-toepassingsgebieden (met name computervisie) over hoe KNN-gebaseerde methoden kunnen worden verbeterd om hun bruikbaarheid bij het analyseren van menselijk keuzegedrag te vergroten.

In hoofdstuk 2 onderzoeken we de steekproefvereisten bij het gebruik van KNN voor discrete keuzeanalyse. Voor betrouwbare KNN moet de dataset, waarop de KNN wordt geschat/getraind, voldoende groot zijn (d.w.z. bestaan uit een voldoende aantal waarnemingen). In vergelijking met de tegenhanger in statistische modellen (bijv. DKM), staan KNN bekend om het consumeren van grotere datasets voor training. De vereisten voor de steekproefomvang van de KNN zijn uitgebreid bestudeerd door middel van de statistische leertheorie, wat heeft geleid tot een reeks theoretische aanbevelingen met betrekking tot het minimumaantal observaties dat nodig is. Deze aanbevelingen zijn in de praktijk echter van beperkt nut. Als zodanig bestudeerde dit onderzoek de vereisten voor de steekproefgrootte van KNN empirisch met behulp van verschillende synthetische en echte datasets. Op basis van onze analyses zijn vereisten voor de steekproefomvang op basis van de algemeen aanvaarde regel, waar de minimale steekproefomvang moet 10 keer het aantal parameters in het netwerk zijn, niet voldoende voor betrouwbare KNN in de context van analyse van keuzegedrag. In plaats daarvan stelt deze studie een meer conservatieve vuistregel voor: de steekproefomvang moet minstens 50 keer het aantal parameters in het netwerk zijn. Ook hebben we vastgesteld dat de KNN meer data nodig hebben naarmate de complexiteit van het onderliggende data-generatieproces toeneemt en de ruis afneemt.

Hoofdstuk 3 richt zich op het gebrek aan vertrouwen in de resultaten van KNN als gevolg van de blackbox-aard van KNN. Dat wil zeggen, hoewel de voorspellende kracht van KNN sterk is, maakt hun ondoorzichtige karakter het moeilijk om de grondgedachte achter deze voorspellingen te begrijpen. Als zodanig pioniert deze studie een qua rekenkracht goedkope en eenvoudig te gebruiken methode om KNN te diagnosticeren in de context van analyse van keuzegedrag. Geïnspireerd door onderzoek op het gebied van computervisie, omvat de voorgestelde methode het synthetiseren van prototypische voorbeelden (met behulp van de activeringsmaximalisatiemethode) na het trainen van de KNN. Deze prototypische voorbeelden leggen de fundamentele relaties bloot die de ANN heeft geleerd, die door de analist kunnen worden geëvalueerd om de getrainde KNN-grondgedachte te inspecteren. Deze studie bespreekt hoe de gesynthetiseerde prototypische voorbeelden kunnen worden gebruikt in de context van keuzedata en presenteert de praktische overwegingen die nodig zijn voor het succesvol diagnosticeren van KNN. De belangrijkste bevindingen van deze studie werden kruislings gevalideerd met behulp van technieken uit traditionele discrete keuzeanalyse.

Hoofdstuk 4 gaat ook in op de blackbox-kwestie van KNN, maar vanuit een ander perspectief. In plaats van een globaal inzicht te verwerven in de grondgedachte van getrainde KNN, herconceptualiseert dit onderzoek het gebruik van heatmaps om de (individuele) voorspellingen van KNN te verklaren in de context van (reis)keuzegedragsanalyse. Dit onderzoek toont aan dat de aanpak die vaak wordt gebruikt om de voorspellingen van KNN uit te leggen (d.w.z. gevoeligheidsanalyse), hiervoor niet geschikt is. Vervolgens laten we zien hoe heatmaps (gegenereerd met de Layer-wise Relevance Propagation (LRP)-methode) kunnen worden toegepast om een verklaring te geven voor de voorspellingen van de getrainde KNN; waardoor een analist vertrouwen in voorspellingen kan opbouwen. Verder laten we zien hoe het gebruik van heatmaps, om meerdere zorgvuldig geselecteerde voorspellingen (met behulp van traditionele discrete keuzemodellen) uit te leggen, kan worden toegepast om vertrouwen op te bouwen in de getrainde KNN als geheel.

In hoofdstuk 5 onderzoeken we de heterogeniteit van de beslissingsregels bij besluitvormers. Een beslissingsregel is het proces dat een besluitvormer gebruikt om alternatieven te evalueren en een keuze te maken. Bij de meeste discrete keuzemodellen wordt de aanname gemaakt van een homogene beslissingsregel. Toch heeft het feit dat reizigers heterogeen zijn in termen van hun beslissingsregels de laatste jaren steeds meer de aandacht getrokken binnen de keuzemodelleringsgemeenschap. Het is ook erkend dat inzichten in heterogeniteit van beslissingsregels cruciaal zijn voor het begrijpen en voorspellen van keuzegedrag. Om heterogeniteit in beslisregels te onderzoeken, vertrouwen de analisten in de keuzemodelleringsgemeenschap op latente klasse (LK) keuzemodellen. Een belangrijke beperking van de LK-modellen is echter hun onvermogen om de beslisregelsheterogeniteit te onderscheiden van smaakheterogeniteit. Dit onderzoek pakt dit probleem aan door een nieuwe KNN-topologie voor te stellen die kan worden getraind met synthetische data en vervolgens kan worden gebruikt om de beslissingsregel(s) van bepaalde stated choice experimenten te herkennen. De voorgestelde KNN-gebaseerde oplossing kan heterogeniteit in beslissingsregels onderscheiden zonder deze te verwarren met heterogeniteit in smaak. Zodanig is het een waardevol hulpmiddel voor analisten van keuzegedrag die heterogeniteit in beslissingsregels willen onderzoeken. De belangrijkste bevindingen van dit onderzoek werden gevalideerd door de resultaten te vergelijken met die van traditionele discrete keuzemodellen.

Concluderend, heeft deze thesis bijgedragen aan het groeiende onderzoek naar het gebruik (en het vergemakkelijken van het gebruik van) KNN voor de analyse van keuzegedrag. Dit is gelukt door methoden en strategieën te ontwikkelen om het vertrouwen van analisten in keuzegedrag in op KNN gebaseerde modellen en hun resultaten te vergroten (bijvoorbeeld door de black-box op te heffen). Verder laten we zien hoe KNN kunnen worden gebruikt om aspecten van keuzegedragsanalyse op te lossen die als moeilijk worden beschouwd voor discrete keuzemodellen. De bijdragen van dit proefschrift motiveren dus toekomstig onderzoek om de mogelijkheden van KNN te benutten om beter begrip en voorspelling van menselijk keuzegedrag af te leiden.

About the author

Ahmad Alwosheel was born on the 22nd of October 1986 in Riyadh, Saudi Arabia. He obtained bachelor and master degrees in electrical engineering from King Saud Unversity and University of Southern California. His work experience includes working for Intel corporations (Santa Clara, California) as a telelcommunication trainee. From 2016 to 2020, he was a PhD candidate at Delft University of Technology. During his time at Delft TU, he joined Boston Consulting Group as a visiting consultant (summer 2019).

List of publications

Peer reviewed journal

Alwosheel, A., van Cranenburgh, S., & Chorus, C. G. (2018). Is your dataset big enough? sample size requirements when using artificial neural networks for discrete choice analysis. *Journal of choice modelling*, *28*, 167-182.
van Cranenburgh, S., & Alwosheel, A. (2019). An artificial neural network based approach to investigate travellers' decision rules. *Transportation Research Part C: Emerging Technologies*, *98*, 152-166.
Alwosheel, A., van Cranenburgh, S., & Chorus, C. G. (2019). 'Computer says no'is not enough: Using prototypical examples to diagnose artificial neural networks for discrete choice analysis. *Journal of Choice Modelling*, *33*, 100186.

Peer reviewed journal (under review)

Alwosheel, A., van Cranenburgh, S., & Chorus, C. G. Why did you predict that? Toward explainable artificial neural networks for travel demand analysis. Transportation Research Part C: Emerging Technologies. Manuscript under review.

TRAIL Thesis Series

The following list contains the most recent dissertations in the TRAIL Thesis Series. For a complete overview of more than 250 titles see the TRAIL website: www.rsTRAIL.nl.

The TRAIL Thesis Series is a series of the Netherlands TRAIL Research School on transport, infrastructure and logistics.

Alwosheel, A.S.A., *Trustworthy and Explainable Artificial Neural Networks for choice Behaviour Analysis,* T2020/11, July 2020, TRAIL Thesis Series, the Netherlands

Zeng, Q., *A New Composite Indicator of Company Performance Measurement from Economic and Environmental Perspectives for Motor Vehicle Manufacturers,* T2020/10, May 2020, TRAIL Thesis Series, the Netherlands

Mirzaei, M., *Advanced Storage and Retrieval Policies in Automated Warehouses,* T2020/9, April 2020, TRAIL Thesis Series, the Netherlands

Nordhoff, S., *User Acceptance of Automated Vehicles in Public Transport,* T2020/8, April 2020, TRAIL Thesis Series, the Netherlands

Winter, M.K.E., *Providing Public Transport by Self-Driving Vehicles: User preferences, fleet operation, and parking management,* T2020/7, April 2020, TRAIL Thesis Series, the Netherlands

Mullakkal-Babu, F.A., *Modelling Safety Impacts of Automated Driving Systems in Multi-Lane Traffic,* T2020/6, March 2020, TRAIL Thesis Series, the Netherlands

Krishnakumari, P.K., *Multiscale Pattern Recognition of Transport Network Dynamics and its Applications: A bird's eye view on transport,* T2020/5, February 2020, TRAIL Thesis Series, the Netherlands

Wolbertus, R, *Evaluating Electric Vehicle Charging Infrastructure Policies,* T2020/4, February 2020, TRAIL Thesis Series, the Netherlands

Yap, M.D., *Measuring, Predicting and Controlling Disruption Impacts for Urban Public Transport*, T2020/3, February 2020, TRAIL Thesis Series, the Netherlands

Luo, D., *Data-driven Analysis and Modeling of Passenger Flows and Service Networks for Public Transport Systems,* T2020/2, February 2020, TRAIL Thesis Series, the Netherlands

Erp, P.B.C. van, *Relative Flow Data: New opportunities for traffic state estimation,* T2020/1, February 2020, TRAIL Thesis Series, the Netherlands

Zhu, Y., *Passenger-Oriented Timetable Rescheduling in Railway Disruption Management,* T2019/16, December 2019, TRAIL Thesis Series, the Netherlands

Chen, L., *Cooperative Multi-Vessel Systems for Waterborne Transport,* T2019/15, November 2019, TRAIL Thesis Series, the Netherlands

Kerkman, K.E., *Spatial Dependence in Travel Demand Models: Causes, implications, and solutions,* T2019/14, October 2019, TRAIL Thesis Series, the Netherlands

Liang, X., *Planning and Operation of Automated Taxi Systems,* T2019/13, September 2019, TRAIL Thesis Series, the Netherlands

Ton, D., *Unravelling Mode and Route Choice Behaviour of Active Mode Users*, T2019/12, September 2019, TRAIL Thesis Series, the Netherlands

Shu, Y., *Vessel Route Choice Model and Operational Model Based on Optimal Control,* T2019/11, September 2019, TRAIL Thesis Series, the Netherlands

Luan, X., *Traffic Management Optimization of Railway Networks,* T2019/10, July 2019, TRAIL Thesis Series, the Netherlands

Hu, Q., *Container Transport inside the Port Area and to the Hinterland,* T2019/9, July 2019, TRAIL Thesis Series, the Netherlands

Andani, I.G.A., *Toll Roads in Indonesia: transport system, accessibility, spatial and equity impacts,* T2019/8, June 2019, TRAIL Thesis Series, the Netherlands

Ma, W., *Sustainability of Deep Sea Mining Transport Plans,* T2019/7, June 2019, TRAIL Thesis Series, the Netherlands

Alemi, A., *Railway Wheel Defect Identification,* T2019/6, January 2019, TRAIL Thesis Series, the Netherlands

Liao, F., *Consumers, Business Models and Electric Vehicles,* T2019/5, May 2019, TRAIL Thesis Series, the Netherlands

Tamminga, G., *A Novel Design of the Transport Infrastructure for Traffic Simulation Models,* T2019/4, March 2019, TRAIL Thesis Series, the Netherlands

Lin, X., *Controlled Perishable Goods Logistics: Real-time coordination for fresher products,* T2019/3, January 2019, TRAIL Thesis Series, the Netherlands

Dafnomilis, I., *Green Bulk Terminals: A strategic level approach to solid biomass terminal design*, T2019/2, January 2019, TRAIL Thesis Series, the Netherlands

Feng, Fan, *Information Integration and Intelligent Control of Port Logistics System,* T2019/1, January 2019, TRAIL Thesis Series, the Netherlands

Beinum, A.S. van, *Turbulence in Traffic at Motorway Ramps and its Impact on Traffic Operations and Safety,* T2018/12, December 2018, TRAIL Thesis Series, the Netherlands

Bellsolà Olba, X., *Assessment of Capacity and Risk: A Framework for Vessel Traffic in Ports,* T2018/11, December 2018, TRAIL Thesis Series, the Netherlands

Knapper, A.S., *The Effects of using Mobile Phones and Navigation Systems during Driving,* T2018/10, December 2018, TRAIL Thesis Series, the Netherlands