Using ANN to assess the mode choice resulting from MATSim

Ruurd Buijs, Thomas Koch, Luk Knapen

Centrum Wiskunde & Informatica (CWI) Amsterdam

Ruurd.Buijs@cwi.nl

Data Preparation

MATSim plans for The Netherlands are generated from daily schedules predicted by an activity-based tool (FEATHERS). Activity locations are predicted only at TAZ (travel analysis zone) level but trip timing, duration, distance and mode are predicted in detail. FEATHERS uses census data, household travel surveys, land-use data and impedance matrices to feed a sequence of discrete choice models to predict relevant aspects of each schedule.
In this project, activity locations are disaggregated by means of the Dutch BAG (Basisregistraties Adressen en Gebouwen) Addresses and Buildings Key Registry which specifies one or more use purposes for each street address in the country. The address sampling (location disaggregation) needs to deliver tours that are consistent with the predicted distances and trip durations. The zoning granularity is fine in the urban parts of the study area but coarse in other parts of the country. Sampling addresses based solely on the given address purpose and the predicted activity type may heavily affect trip timing in areas covered by or located near large TAZ.

Methodology

The idea is to estimate a trip based mode choice model from the MATSim output for a region in Amsterdam for which equivalent mode choice models have been estimated from revealed preference based on GPS traces. The goal is to investigate whether or not such technique can support calibration of MATSim parameters. Instead of comparing MATSim results directly to observable quantities (e.g. flows) we aim to compare the properties of the embedded mode choice mechanism in MATSim (not implemented by an explicit choice model) to a choice model extracted from observations.

After the MATSim model is set up and ran using a population that is equal in size to 10% of the actual population, it needs to be calibrated to ensure it captures mode choice behaviour well. To evaluate whether mode choice behaviour is captured correctly by the model, a mode choice prediction model is drawn up using the MATSim output and choice sets based on the trip data that can be extracted from the plans generated by MATSim. For the choice sets, we can compute alternative route options for transit using routing applications such as R5 [1]. This tool computes for various alternate realistic route options on the transit network based on GTFS timetables of the Dutch network.

The input for choice prediction model will consist of several alternative-related features, most notably trip duration, as well as agent-specific features, such as age category and income. In order to find the most accurate and robust predictive model to describe mode choice, Machine Learning techniques will be used. Several studies [2] [3] have shown the potential of Deep Neural Networks (DNNs) in this domain, which can capture more complex (non-linear) patterns than statistical logit-like models that could also be used in this context. A known drawback of machine-learning models is the fact that they are less robust and difficult to interpret [4]. To account for this, the models are trained using the *belief matching loss function* that allows for the use of a prior distribution to represent the ground truth in a Bayesian setup. This loss function has a regularizing effect, and can improve the model's generalization performance [5].

From the trip data generated by MATSim, a part is withheld as a testing partition, to eventually compare the performance of the DNN mode choice model on this partition with the performance on out-of-sample real world data, which will give an indication of how well the MATSim model replicates mode choice behaviour. The remaining part of the data is used for training and validation of the models. A hyperparameter search is performed to find a configuration which results in the best average (validation) performance in terms of categorical-cross entropy, which is equivalent to the log-likelihood function if one looks at the models in a statistical way. Since some variation in performance is still expected between different training runs of the model, even when a good hyperparameter configuration is found and used, 100 different models are trained, from which the one with the best validation performance is selected to be used for an ultimate comparison.

Once the DNN model has been selected which best describes the mode choice behaviour observed in the MATSim output, the model is fed with choice sets based on tracking data obtained in Amsterdam, as well as the previously withheld choice sets based on the MATSim model. The comparative performance in terms of log-likelihood and accuracy can be used as an indication for the goodness of calibration of the MATSim model, in a way that goes beyond looking merely at the modal split.

Potential Pitfall

Two generative models (FEATHERS and MATSim) are used in a chain. The FEATHERS output is supposed to comply well with the observed schedules which in turn may be adjusted by MATSim by updating trip timing and mode choice. Questions related to this observation (e.g. compliance between schedules generated by FEATHERS and observed data) need further investigation.

[1] Rafael HM Pereira et al. "r5r: rapid realistic routing on multimodal transport networks with r 5 in r". In: *Findings* (2021), p. 21262.

[2] Yixuan Ma and Zhenji Zhang. "Travel Mode Choice Prediction Using Deep Neural Networks With Entity Embeddings". In: *IEEE Access* 8 (2020), pp. 64959–64970.

[3] Shenhao Wang, Baichuan Mo, and Jinhua Zhao. "Deep neural networks for choice analysis: Architecture design with alternative-specific utility functions". In: *Transportation Research Part C: Emerging Technologies* 112 (2020), pp. 234–251.

[4] Shenhao Wang and Jinhua Zhao. *An Empirical Study of Using Deep Neural Network to Analyze Travel Mode Choice with Interpretable Economic Information*. Tech. rep. 2019.

[5] Taejong Joo, Uijung Chung, and Min-Gwan Seo. "Being Bayesian about categorical probability". In: *International Conference on Machine Learning*. PMLR. 2020, pp. 4950–4961.