New tools of the trade? The potential and pitfalls of 'Machine Learning' and 'DAGs' to model origin-destination data

Robin Lovelace*[1], Ilan Fridman-Rojas[2], and Rob Long[3]

[1]Institute for Transport Studies, University of Leeds

*Email: R.Lovelace@leeds.ac.uk

Abstract

This paper explores the potential for emerging methods Machine Learning and Directed Acyclic Graphs (DAGs) to be applied to transport modelling at the origin-destination (OD) level. OD data is inherently spatial and is complex, due to the multitude of ways of allocating geographic attributes to the OD pairs (e.g. buffers and intersections with geographic representations of OD data generated using straight desire lines, shortest path algorithms or probabilistic routing). This makes their analysis an interesting geocomputational challenge, seldom tackled by geographers. The application of Machine Learning and DAG methods, developed in other fields, to this geographical data holds great potential to improve the ability to infer causality in mode split from OD data. However, there are also pitfalls to using these methods which can be black boxes, even if the code is open source, if the analyst does not understand what they are doing with the data. Based on the work we discuss ways to ensure new methods in the field are used wisely and set-out next steps for our own research.

Keywords: Machine Learning, Causal Inference.

1 Introduction

This paper aims to show the potential benefits, and the pitfalls, of machine learning algorithms in analysing transport data. It results from a 3 month 'T-TRIG' (Transport Technology Research Innovation Grant) project funded by the UK's Department for Transport (DfT) entitled "Using Machine Learning and Big Data to Model Car Dependency: an Exploration Using Origin-Destination Data".

Despite the applied title, the prime focus was methodological development: using geocomputation, in the form of new analysis of new open geographical data, to explore a long-standing policy challenge: how to reduce car dependency. As highlighted by the DfT, Machine Learning is relatively new in the transport sector (Hagenauer and Helbich 2017). The research was therefore exploratory and open-ended in its scope. Having completed the first Phase of the work (we will have completed the project in time for the Geocomputation conference), we would like to comunicate some of the findings from the research.

*

By demonstrating previously impossible or inaccessible methods we aim to show how new techniques, combined with new and newly open datasets, can generate a strong evidence base for transport planning and policy. The aspiration is that the work will filter into policies, to make them data-driven, transparent, reproducibible and encouraging of citizen science and innovation.

2 Input data

For simplicity and maximum accessibility this project uses only datasets which are open (publicly available). There are three main input dataset types:

- Origin-destination (OD) commute data from the 2011 Census data.
- Geographically aggregated socio-demographic data from the 2011 Census.
- Geographic variables associated with each OD pair.

The Census was used as the primary input dataset because it provides so many variables relevant to car dependency. To our knowledge, the breadth of input datasets have never been analysed together in a single project.

The case study region used for this project was **West Yorkshire**. This case study was selected due to the wide range of social and geographical environments linked to car dependency found here: it includes rural, urban, deprived and privaledged areas.

2.1 Origin-destination data

The fundamental input dataset was a table reporting the number of people travelling, by main mode of transport, commuting to work between Middle-Super Output Areas (MSOAs, average population: ~7,500). This is an open dataset (available online from the official WICID data portal) (dataset WU03EW). The file was downloaded as `wu03ew_v2.zip`, an 11.8 MB zipfile which when unzipped creates the file 109 MB plain text file `wu03ew_v2.csv`, read-in with the following command:

```
odall = readr::read_csv("data/wu03ew_v2.csv")
```

The result is a data frame the first two columns of which contain a the code for the MSOA of origin and destination, respectively. The subsequent columns report number of people whose main mode of travel to work was:

- work at home (no transport used)
- some form of metro
- train
- bus or coach
- taxi
- motorcycle or scooter
- drive a car or van
- passenger in car or van
- bicycle
- walk
- other

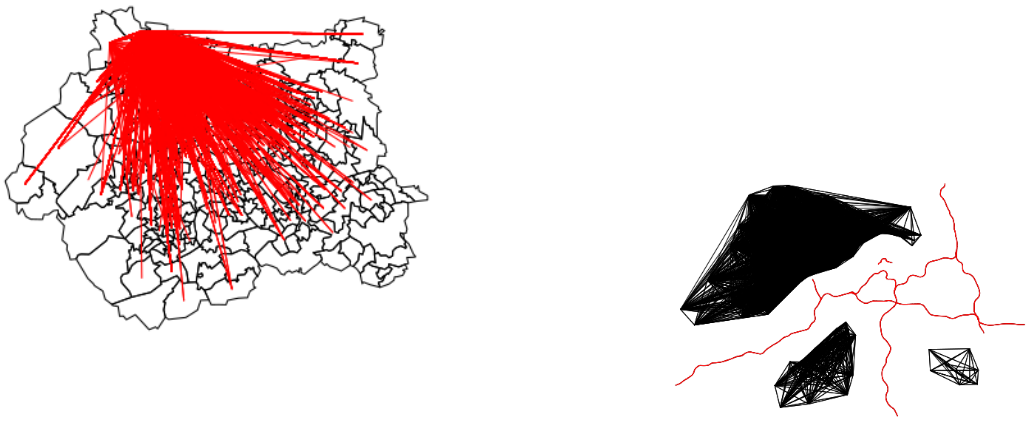

Figure 1: A sample of work commute flows in the West Yorkshire region (left), and a sample of the result of the code which calculates the distance between motorways (red) and flows (black), in this case discarding all flows within a certain distance of the motorway, as an example.

We grouped together people who drive or are passengers to estimate car dependency. Further, we filtered out flows with an origin or destination outside West-Yorkshire, resulting in **53,807** OD pairs in the study region. This amount of data is sufficient for machine learning algorithms to work and extract complex insights, but small enough for experimentation and fast iteration of methods.

To convert the non-geographical OD dataset into geographic data we used the function `od2line()` from the **stplanr** R package. The result is straight lines that can be plotted on the map and about which geographical variables, such as proximity to motorways, can be extracted.

We will hereon refer to each home-work (origin-destination) pair and its corresponding number of commuters, in geographic form, as a *flow* (see Figure 1).

To this base table there are two ways to further increase the data included for modelling: the Census provides numerous other demographic and economic measures and indicators which can be linked to the commuter's home MSOA (available here), and it also provides workplace data which a group of researchers at the University of Southampton have conveniently created a classification of (by work type, available online as well) which can be linked to the workplace MSOA.

2.2 Socio-demographic data

The geodemographic data relevant to each origin (home) MSOA which was deemed relevant and annexed to the flows data consists of the following variables: number of people in particular age brackets, number of people of each gender, car or van availability (including number of homes with 0,1,2,etc. cars), population density, number of economically active and inactive people, general health (number of people with very good, fair, etc. general health), number of people per ethnicity group, number of people by maximum qualification level, and lastly, average number of rooms, bedrooms and fraction of homes with central heating.

In addition to this we included the workplace zone classification which assigns each destination MSOA (workplace) to one of 7 groups (Table 1).

Table 1: The classification of workplace zones.

Workplace zone classification	Code
Retail	1
Top jobs	2
Metro suburbs	3
Suburban services	4
Manufacturing and distribution	5
Rural	6
Servants of society	7

2.3 Spatial data

In addition to this Census data we gathered and computed spatial data as well. We have obtained the location of motorways through the OSM API, and the positions of train stations, coach stations, and bus stops from the NAPTAN dataset. This allowed us to calculate geographical distances between flow lines and motorways, train stations, coach stations, and bus stops. This proximity and accessibility data is arguably vital to best understand and model car usage propensity, and has for the first time become readily available via the OSM platform, and analysis of the type we aim to do next should be cutting-edge data analysis.

2.4 Overview of input data

A sample of the first 20 variables used in the input dataset for the machine learning algorithms is provided in Table 2 (the full list can be seen in a code repository that will accompany this paper).

Table 2: The first 20 variables in the full dataset.

Variable	Description
homeMSOA	MSOA code for origin (place of residence)
workMSOA	MSOA code for destination (place of work)
workhome	Fraction of people who work from home (don't commute)
metro	Fraction of people who use the metro, tram or light train to commute
train	Fraction of people who use the train
bus	Fraction of people who use the bus
taxi	Fraction of people who use a taxi
motorcycle	Fraction of people who use a motorcycle, scooter or moped
car	Fraction of people who use a car or are passengers in a car or van
cycle	Fraction of people who cycle
walk	Fraction of people who walk
othertransp	Fraction of people who use a form of transport not listed above
npeople	Total number of people resident in the homeMSOA
16-24	Fraction of people in the age range 16-24
25-34	Fraction of people in the age range 25-34
35-49	Fraction of people in the age range 35-49
50-54	Fraction of people in the age range 50-54
65-74	Fraction of people in the age range 65-74
75	Fraction of people of age 75+
female	Fraction of people who are female

3 Machine Learning

The first step in this process is that of model selection by cross-validation, and it is this procedure which we will now describe. The first step of model selection is to choose the metric of interest, based on which we will select the best-performing model. Given that this is a regression problem there are multiple possible metrics: mean squared error (MSE), mean absolute error, coefficient of determination (R^2), etc. From the point of view of policy decision-making, which is our ultimate goal, there is no clear choice of best-suited metric. We have chosen the coefficient of determination as it seems to be the metric of interest for our Data Science contact at the Department for Transport, but it must be noted that this metric is ill-suited for the mostly non-linear[1] regression models we consider (see for example (Spiess and Neumeyer 2010)).

Having chosen the metric of interest, we proceed to shuffle and split the data in preparation for cross-validation. Flows are unordered data and therefore can be shuffled uniformly with no problem.

[1] For clarity, we are referring to non-linearity in the regression parameters, models non-linear in the covariates (e.g. the Elastic Net model we use) are still linear by this definition.

We shuffle the flows in the West Yorkshire dataset, and then split off half of the data for model selection and validation, the other half is left for final model testing to estimate the generalisation error (or parts of it may used for further model selection, e.g. parameter tuning, if needed).

We then perform 10-fold cross-validation on the validation half of the dataset. At this point we do not perform any hyperparameter tuning and use the default parameters set in scikit-learn's implementation (Pedregosa et al. 2011) of these algorithms[2]. The regression models considered and the result of the model selection are shown in the table below.

Table 3: 10-fold cross-validation R^2 scores of the the regression models considered. The `MEAN MODEL` is a model which always predicts the mean of the response variable, and is included as a reference point and benchmark. The RANSAC and Stochastic Gradient Descent (SGD) regressors have clearly not converged but we will not attempt to tune them for now.

Model	R^2
MLP	9.548066e-01
XGB	9.522178e-01
RandomForest	9.411053e-01
ExtraTrees	9.366043e-01
ElasticNetCV	9.268778e-01
PassiveAggressive	8.870397e-01
DecisionTrees	8.736974e-01
TheilSen	8.561763e-01
KNeighbors	8.427571e-01
RANSAC	7.766147e-01
–MEAN MODEL–	0.000000e+00
Dummy	-3.752000e-04
SGD	-1.211319e+23

Without engaging in extensive hyperparameter tuning on a separate dataset, we see that the Multi-Layer Perceptron and the XGBoost regressor (Chen and Guestrin 2016) top the list with $R^2 \approx 0.95$. The difference between the top-performing models is likely not significant, therefore on theoretical grounds we will choose the XGBoost regressor as our tentative final model, as boosting is known to be less likely to overfit (Schapire 1999), and therefore is likely to generalise best to new datasets. We will therefore select it as a tentative final model and now proceed to perform some model validation to ensure its predictions are sensible.

4 Model validation

For a regression task such as the one at hand, the model validation checks which can be performed are perhaps less intuitive than the tests available for classification tests. However some basic plots

[2] Grid search or random search hyperparameter tuning by cross-validation could be carried out at a later date on a separate dataset for a subset of the selected models to extract further gains in predictive performance.

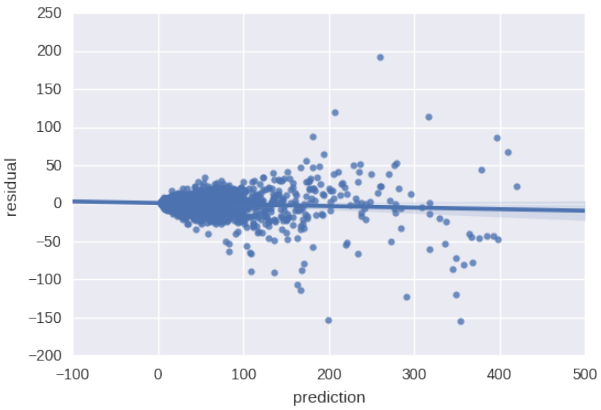

Figure 2: The residuals of the XGBoost regressor on a validation dataset.

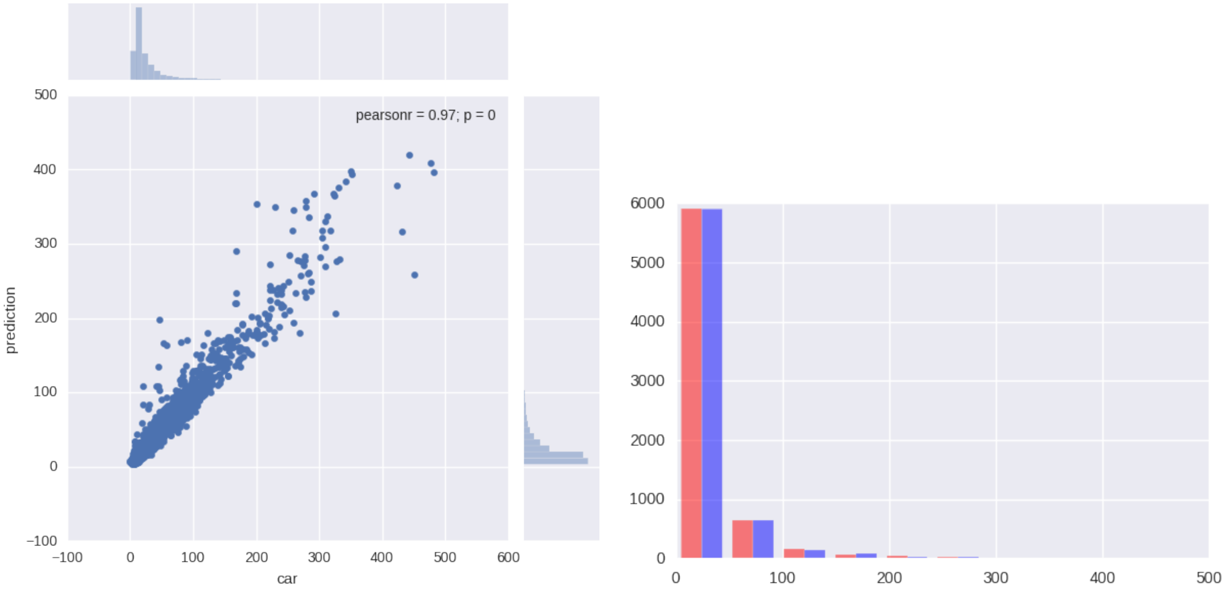

Figure 3: The distribution and correlation of predictions and observed values.

of the model's predictions and checks of the model's residuals and the correlation of the model's predictions with observed values can and should be carried out.

The figure below shows the residuals of the XGBoost regressor on a validation dataset, showing no obvious trend or asymmetry which would be indicative of a poorly-fit model.

We can further inspect the distribution and correlation of the predicted and observed values, as well as a histogram of their distribution.

5 Next steps

There is one noteworthy caveat which applies to all studies using Machine Learning (and more generally regression on observational data without Randomised Controlled Trials) to inform decision making: both Machine Learning approaches and standard regression approaches when carried out appropriate can identify correlations, but evidence for these correlations is tentative and correlations by themselves, without evidence of a causal link, are not a sound basis for decision-making or interventions.

The patterns found are therefore a very solid first step in finding potential causal links which could inform decision-making and policy interventions, but they must be informed by causal modelling. To this end, in the next stage we will explore introducing causal information into the machine learning modelling, as well as using standard causal analysis via the use of Directed Acyclic Graphs (DAGs) to tease out which correlations may or may not be causal.

The results so far should thus be taken as tentative as they are in the process of being tested further to guarantee as much as possible their validity and potential for informed, data-driven decision making.

References

Chen, Tianqi, and Carlos Guestrin. 2016. "XGBoost: A Scalable Tree Boosting System." In *Proceedings of the 22Nd Acm Sigkdd International Conference on Knowledge Discovery and Data Mining*, 785–94. KDD '16. New York, NY, USA: ACM. doi:10.1145/2939672.2939785.

Hagenauer, Julian, and Marco Helbich. 2017. "A Comparative Study of Machine Learning Classifiers for Modeling Travel Mode Choice." *Expert Systems with Applications* 78 (July): 273–82. doi:10.1016/j.eswa.2017.01.057.

Kursa, Miron B, Aleksander Jankowski, and Witold R Rudnicki. 2010. "Boruta–a System for Feature Selection." *Fundamenta Informaticae* 101 (4). IOS Press: 271–85.

Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, et al. 2011. "Scikit-Learn: Machine Learning in Python." *Journal of Machine Learning Research* 12: 2825–30.

Saxe, A. M., J. L. McClelland, and S. Ganguli. 2013. "Exact solutions to the nonlinear dynamics of learning in deep linear neural networks." *ArXiv E-Prints*, December.

Schapire, Robert E. 1999. "A Brief Introduction to Boosting." In *Ijcai*, 99:1401–6.

Spiess, Andrej-Nikolai, and Natalie Neumeyer. 2010. "An Evaluation of R2 as an Inadequate Measure for Nonlinear Models in Pharmacological and Biochemical Research: A Monte Carlo Approach." *BMC Pharmacology* 10 (1). Springer: 6.

Srivastava, Nitish, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. "Dropout: A Simple Way to Prevent Neural Networks from Overfitting." *Journal of Machine Learning Research* 15 (1): 1929–58.

The Theano Development Team, R. Al-Rfou, G. Alain, A. Almahairi, C. Angermueller, D. Bahdanau, N. Ballas, et al. 2016. "Theano: A Python framework for fast computation of mathematical expressions." *ArXiv E-Prints*, May.

Vincent, Pascal, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. 2010. "Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion." *Journal of Machine Learning Research* 11 (Dec): 3371–3408.