Using Machine Learning and Big Data to Model Car Dependency: Progress Report 2 - Machine Learning

Robin Lovelace and Ilan Fridman-Rojas

2017-04-25

Machine Learning

Progress Report 1 describes the construction of the explanatory variables and definitions of the dependent variable. This second Progress Report describes the work we have done to model the input data using 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. 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 1: 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

[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.

[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.

Model	R^2
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.

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 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.

A note on data cuts and variable forms

One important cut was required to reduce the noise in the dataset. Flows with low numbers of total commuters introduce artifacts in the response variable, such as inflation at values corresponding to exact fractions with a low denominator (e.g. $\frac{1}{2}$, $\frac{1}{3}$, etc.). These artifacts from low sample sizes are not representative of global behaviour and may complicate or skew the model-fitting process, resulting in models with poorer generalisation performance. We therefore chose to drop flows with less than 10 commuters, in doing so removing low-sample artifacts and improving model fit.

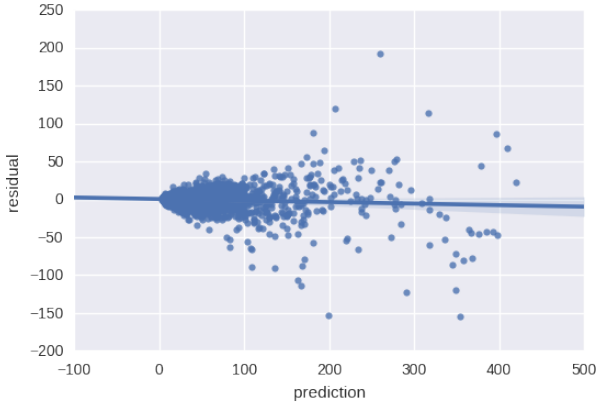

Figure 1: The residuals of the XGBoost regressor on a validation dataset.

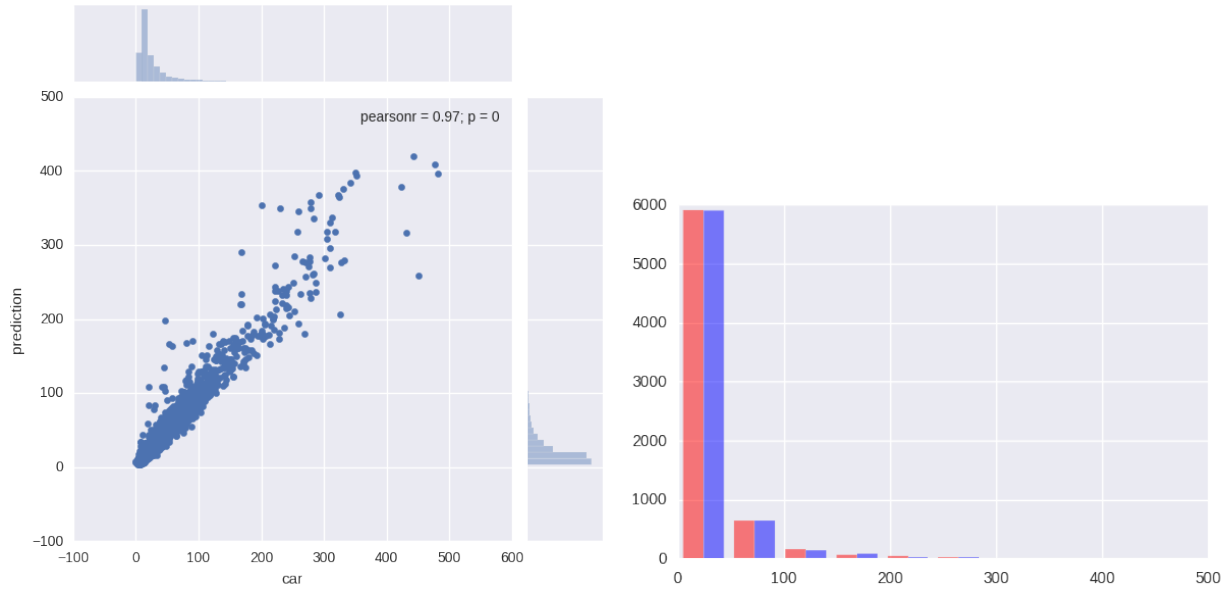

Figure 2: The distribution and correlation of predictions and observed values.

Consideration of both the variables to be included and potential redundancies or multicollinearity between them, as well as low sample-size artifacts and their removal must therefore be carefully undertaken at the earliest stage, ideally at the data preparation stage, or during model selection, as soon as they are identified.

A further important point to make regarding the form of the variables fed into the models is that the use of ratios should be entirely avoided, with the use of raw counts always being preferable. This is due to the fact that using ratios of random variables induces strong spurious correlations (Atchley, Gaskins, and Anderson 1976) which may hurt predictive performance, but more importantly, will skew measures of relative variable importances.

A Deep Learning model

For completeness, we would be remiss if we did not consider fitting a deep learning model, particularly since the Multi-Layer Perceptron performed well during the previous model selection stage, considering a neural network model with further hidden layers is fully justified. We have constructed a 3-hidden layer Multi-Layer Perceptron (MLP) using the Theano (The Theano Development Team et al. 2016) Python package. The MLP implementation in scikit-learn includes only a single hidden layer, therefore there is value in exploring potential predictive performance gains by considering deeper neural networks.

Numerous contemporary training methods were implemented to enable the successful training of this neural network, including unsupervised pre-training of the layers by Stacked Denoising Auto-encoders (Vincent et al. 2010), orthogonal weight initialisation (Saxe, McClelland, and Ganguli 2013), dropout regularisation (Srivastava et al. 2014), L_1 and L_2 regularisation, ReLu activation units, etc. This allowed the neural network to be properly fit over hundred of epochs (passes over the training data), with no sign of significant overfitting, as shown in the training and validation errors in the figure below.

A deep learning model was therefore successfully fit, with a sensible residual plot and distribution of predictions. However its performance was still below that of the XGBoost regressor previously selected, with a coefficient of determination of $R^2 \approx 0.89$, and a Pearson correlation between predicted and observed values of 0.96. One may speculate that with larger training datasets (e.g. a good fraction of the flows in England), this model may dominate. However with the limited training set of part of West Yorkshire, XGBoost remains dominant, and the significant computation overhead of training a Deep Learning model is not justified.

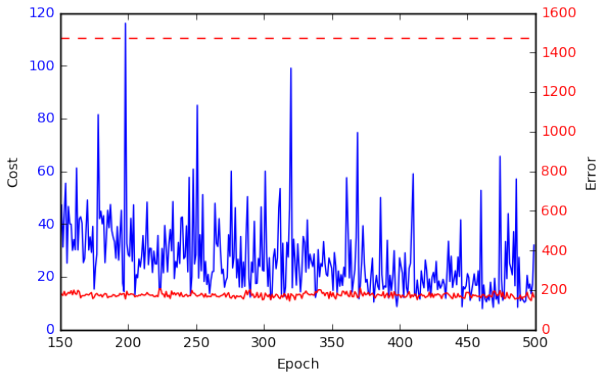

Figure 3: The MSE on the training and validation sets as training of the MLP progresses.

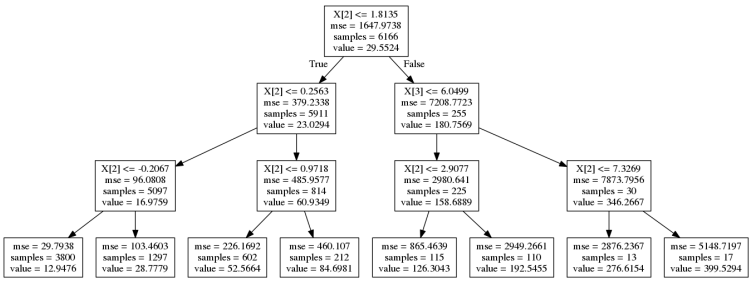

Figure 4: A decision tree fit on the West Yorkshire training data. X is the training data matrix with X[i] denoting its i-th column as the variable that split has been based on.

Interpretable analysis

Thus far we have selected and validated a model which maximises a metric of predictive performance. Should we be interested purely in predictive purposes this is the best model to consider from our analysis thus far. However from a policy perspective the goal is perhaps only partly predictive accuracy (which is of value when considering counterfactual scenarios for future projects, e.g. given the data we have on known flows on particular roads, how heavy a flow can we expect on a hypothetical new road with certain features), but also reliant on gaining a data-driven understanding of the factors which most heavily influence car commuting flows. To this end we must extract interpretable insights from the predictive machine learning models we have fit.

A noteworthy caveat to always be kept in mind when we attempt to extract interpretable knowledge from machine learning models is that the particular set of variables highlighted as most important in making accurate predictions are as much a function of the model itself as they are of the data. Therefore no single set of important features extracted from a single model should be taken without undue caution. Ideally, one may approach the problem as one of consensus, where variables selected by multiple models (each significantly algorithmically different from the rest) would have a stronger evidence-base to claim variable importance.

The simplest visual approach to gauging variable importances is to fit a shallow decision tree, such as the one shown in the figure below.

However decision trees yielded a comparatively poor fit during model selection (see table above), so this gives perhaps a suboptimal understanding of variable importances.

Another tree-based approach however gave much better predictive performance, and can also yield a measure of variable importances based on how often a particular variable was used to create splits in the model's trees: a Random Forest regressor. Extracting variable importances from a Random Forest regressor fit on the

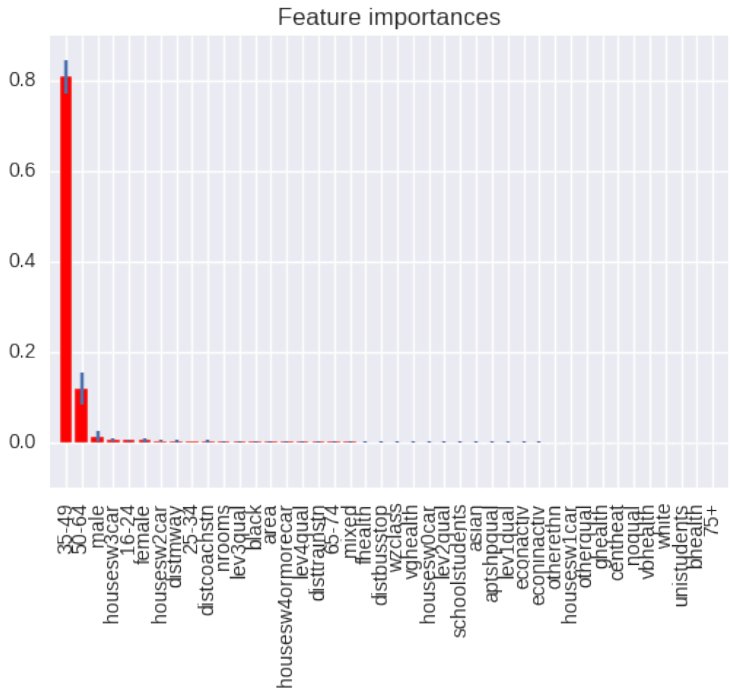

Figure 5: Variable importances from a Random Forest model.

training set gives the variable importances shown in the figure below.

A different, linear, model which yielded decent predictive performance and which is interpretable, is the Elastic Net regression model we fit as part of the model selection. The regression coefficients of this model give both a measure of variable importances, as well as directionality of their effect, as shown in the table below.

Table 2: The regression coefficients of the Elastic Net model, ordered by descending absolute value.

Variable	Coefficient
35-49	16.1887695
50-64	13.7989016
male	9.3765921
female	3.2682303
25-34	-3.1827790
16-24	-3.0933285
65-74	2.0803315
housesw0car	-1.7695871
econactiv	1.5624934
lev4qual	-1.2248944
75+	0.9260478
bhealth	0.8581431
housesw2car	0.8252421
housesw3car	0.8228936
black	-0.6374966

We have also considered the Boruta (Kursa, Jankowski, and Rudnicki 2010) feature selection algorithm, which is based on Random Forest regressors to yield the following set of important variables:

```
['16-24', '25-34', '35-49', '50-64', 'male', 'female', 'housesw0car', 'housesw2car', 'housesw3car', 'hou
 'area', 'distmway']
```

The top 10 variables by importance according to the Elastic Net regressor, the Random Forest model, and the 8 variables identified by the Boruta algorithm are therefore as follows:

```
{'male', 'housesw0car', '35-49', '25-34', 'lev4qual', '16-24', '65-74', 'econactiv', 'female', '50-64'}
{'distcoachstn', 'distmway', 'housesw3car', 'male', '35-49', '25-34', '16-24', 'housesw2car', 'female',
{'distmway', 'housesw3car', 'housesw4ormorecar', 'male', 'black', 'housesw0car', '35-49', 'area', '25-34
```

Choosing the common subset of these variables as the set of variables on which we have a strong consensus evidence-base of their importance, we therefore come to the following set of variables, which we conclude have a heavy influence on predicting the response variable:

{'16-24', '25-34', '35-49', '50-64', 'female', 'male'}

Interpretability of individual predictions

So far we have focussed on extracting the set of variables which are globally important in predicting the response variable. However often what is of interest is being able to extract some understanding of why individual predictions have been made. To this end we have included the recently devised LIME algorithm (Ribeiro, Singh, and Guestrin 2016) as implemented by one of the authors of the paper and made available online.

A typical 'explanation' for a given prediction then looks as follows:

```
[('distmway > 0.38', 1.5658305905564187),
 ('wzclass > 0.64', -0.65392769537881634),
 ('75+ <= -0.28', -0.50537077658088714),
 ('65-74 <= -0.47', 0.15784334084586701),
 ('50-64 <= -0.45', -0.11138315818592245),
 ('-0.66 < distcoachstn <= -0.28', -0.093203330715291008),
 ('-0.32 < distbusstop <= 0.09', 0.066183230441780636),
 ('-0.21 < 16-24 <= -0.02', 0.063091145326841958),
 ('-0.41 < asian <= -0.08', -0.057087600296320297),
 ('housesw1car <= -0.75', 0.054789659285169511),
 ('-0.68 < fhealth <= -0.20', 0.04912114269573125),
 ('econactiv <= -0.77', -0.048831449650547552),
 ('35-49 <= -0.44', -0.048015501255921647),
 ('-0.17 < housesw0car <= 0.54', -0.046159925942110741),
 ('lev2qual <= -0.68', -0.042291716940468689)]
```

The reliability, interpretation, and further uses of the LIME algorithm in our present case study remain to be fully examined, but its potential is clear.

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

Atchley, William R, Charles T Gaskins, and Dwane Anderson. 1976. "Statistical Properties of Ratios. I. Empirical Results." *Systematic Zoology*. JSTOR, 137–48.

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.

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.

Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin. 2016. "Why Should I Trust You?: Explaining the Predictions of Any Classifier." In *Proceedings of the 22nd Acm Sigkdd International Conference on Knowledge Discovery and Data Mining*, 1135–44. ACM.

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.