RESPONSE TO "DEMAND ESTIMATION WITH MACHINE LEARNING AND MODEL COMBINATION"

PAUL SANGREY

1. Introduction

In their recent NBER working paper, "Demand Estimation with Machine Learning and Model Combination" Bajari, Nekipelov, Ryan and Yang present a method of demand estimation using methods taken from the machine learning literature and contrast them with the typical methods used within economics. In recent years, datasets with large numbers of variables and very large numbers of potential regressors have been made available. This poses both unique strengths and unique challenges for estimation. Because of the increase of data, we are able to relax a large variety of assumptions regarding the parametric form of the decision-rules and distributional assumptions on the shocks. On the other hand though, the typical linear models are not identified in situations where there are more potential regressors than data-points – The p ¿ N problem.

This paper tries to develop a procedure that can be used by applied econometricians to estimate these demand functions to better make firm-level decisions regarding pricing, advertising and so on. This is a natural situation for regularized non-parametric estimators as common in the machine learning literature because what practitioners generally care about is demand prediction as a function of regressors as opposed to the marginal effects themselves. They do provide a method for estimating the effect of a discrete change in promotions but because of the non-linear nature of many of their methods this is not the same as the marginal effect.

2. Survey of Results

Their results come in three forms. The first is the one mentioned above – an estimated effect of a discrete change in some of the regressors. The second is theoretical in nature. They provide asymptotic convergence rates of a number different estimators. For the most part, they converge at a rate slower than $\sqrt{N}$, with ${}^{1/3}\sqrt{N}$ occurring rather often, but this is to be expected because they are doing non-parametric estimation. Interestingly, they do show that a combination estimator converges at a $\sqrt{N}$ rate to a Normal distribution, and hence typical bootstrap procedures provide valid inference. Their third contribution is simply a horse race between the various estimators. They do this both in a simulation and using a scanner panel data, where in the estimation case they compare predictive performance on a part of the data saved for that purpose.

The model they study is $\log Q_{jmht} = f(p_{mt}, a_{mt}, X_{mt}, D_{mt}, \epsilon_{jmgt})$, where a is a matrix of advertising and promotional measures, D is a vector of demographics, p is a vector of prices, and ϵ is an idiosyncratic shock. There are J products with observable characteristics X_j. They are sold in market m at time t.

The estimators they compare from the econometrics literature are linear regression using the various possible regressors and their interactions and logit using a BLP-style approach. The machine learning estimators include bagging, random forest, lasso, stepwise regression, stagewise regression, and incremental forward stagewise regression. They also find the optimal linear combination of the models using linear regression with weights chosen using out-of-sample test data.

In their simulation study they compare a basic linear regression model and a highly non-linear functional form based upon sine, log, and identity functions. They also include various junk regressors, and to asses the performance of the various regressors they divide their data into three parts. They use the first part to estimate the model; the second part to estimate weights to form the best linear predictor mentioned above, and compare out-of-sample fits on the third. They use datasets with 400, 6400, and 51200 observations. Bagging and random forests do well in all sample sizes, and stagewise regression has the best fit in the smallest sample size. Linear regression does well in-sample but terribly out-of sample. The random forest, bagging and stagewise regression models receive the greatest weight in the best linear predictor. They then do a very similar procedure using data on a large number of sales, that is scanner data. In that case random forests and support vector machines perform the best.

They also provide theoretical statements about a number of the estimators' rates of convergence. For example, the random forest estimator's mean square error is $O_p(1/\sqrt[3]{N})$. They also show that if the weights converge to the true values fast enough, $N^{1/6} \sup_x ||\hat{w}(x) - w(x)|| \to_p 0$, then the predicted values converge to the true ones at the parametric rate, and are asymptotically normal. As a result, one can use bootstrap procedures to determine the standard errors that you would have to use.

They then use their data to examine the effect of a promotion. They tag a product as promoted if the accompanying price reduction is greater than 5%. They train their models on the control group (no promotion), and then use it to predict the treated group. They then use the difference between the predicted and actual in treatment groups as the effect of the promotion. If everything else is equal between the control group and the treatment group this necessarily follows.

3. Conclusion

They provide a very natural, efficient way to estimate demand dealing both the $p > N$ problem and making very few assumptions about the underlying functional form. Demand estimation lends itself very easily to machine learning methods because the question of interest is how to predict the regressand, there is a large number of potential regressors, and one can find copious amounts of data. This question has been studied in a great more detail in the machine learning literature than it has the economics literature, where treatment effects, small to medium sample sizes, and a few regressors are common. As a result, they provide a very good new method for demand estimation. Furthermore, since they manage to show that the best linear estimator converges at the parametric rate and is asymptotically Gaussian under very general assumptions. It is computationally very difficult to implement, but in areas where a machine learning algorithm might be useful being able to do non-parametric estimation and get correct standard errors using the bootstrap could be very useful.