Using Random Forests to Predict Supplemental Security Income Marginal Tax Rates

July 2016

Abstract

Using the Current Population Survey, we use a machine learning method called Random Forests for computing the implicit marginal tax rates (MTRs) pertaining to the Supplemental Security Income Program. Traditional MTR estimation can be complicated and difficult. Random Forests, a machine learning technique, can provide a simple and effective alternative for MTR calculation.

1 Introduction

We provide an easy and effective alternative MTR calculation method, called Random Forests, and compare its results with the MTRs derived using the traditional methods of programmed income rules, and linear regressions. More specifically we calculate the MTRs for the Supplemental Security Income (SSI) program: a government program that offers benefits to eligible disabled or old-aged individuals.

Random Forests is one of the many machine learning techniques that may allow for a more effective and easy way to model these complex, nonlinear relationships Varian (2014). When compared to the other two traditional methods, Random Forests can produce accurate and robust MTR calculation results.

Here we provide a brief overview of Random Forests; then we move to calculating the marginal tax rates (MTRs) of the Supplemental Security Income program (SSI) using all the three methods above.

2 Random Forests

Random Forests average the predictions from subordinate models called prediction trees to determine overall prediction results. To help understand this method, imagine a jury about to decide the outcome of a criminal case, with each juror representing a prediction tree. Each juror is given information about similar criminal cases to the one they will decide. The patterns across previous cases help inform them as to whether the accused is guilty or not, should a similar pattern arise in the current case. This learning process represents the training of the Random Forest model. For example, if the current accused was present at the crime scene, and individuals who were present at the crime scene are historically convicted, this could illuminate evidence for conviction. Now, the trained jury considers the case they are to decide by recognizing the patterns from previous cases in the current case. The juror finds that the current case satisfies the patterns of a few of the previous cases, and each juror votes to convict according to the conviction of these previous cases. Notice,

that each member of the jury may give different answers, as the jurors may not think about each case in the same way. Finally, the jury casts their votes; the judge, who represents the aggregative influence of the random forest, considers all the votes and determines the conviction according to a majority rule.

In our particular application, the prediction trees used in our Random Forest model are called regression trees. Each of these regression trees are trained after taking random samples, with replacement, of the original sample space. Each regression tree that constitutes the forest takes that sample and places partitions of that sample into categorical bins: at each node, the tree asks a categorical question about the data. For example, is earned income less than $\$8,000$? The resulting edges place the data points where earned income is less than $\$8,000$ into one bin, and those data where earned income is larger are placed into another bin. This happens recursively until no more useful information can be extracted from the particular set of questions, or until the data set has been spread too thin and is therefore biased Svetnik et al. (2003). At the end of this process each tree will have a collection of leaves, or terminal nodes, that are full of data points that are similar as they fulfilled similar criteria at each node. Lastly, the mean of the variable that will be predicted is taken among the data points in each leaf. Then a set of test data, or data that does not contain the variable we want to predict, is passed into the Random Forest model. The test data then falls into terminal nodes, or leaves, by answering the same questions that the training data were asked. Once in a terminal node, the test data will then be assigned the same mean value of the training data points that fell into that same terminal node. Lastly, this process is repeated using each tree in the forest and the final prediction is the average result of all the trees Breiman (2001). This forest can provide accurate results by using a flexible model of the relationships between many variables Wager and Athey (2015).

The main difference between a linear regression, and a regression tree is that a linear regression is a global model, meaning there is a single predictive formula, being the estimated regression equation, that holds for the entire sample space. This can present problems when the variables have complicated nonlinear interactions.

To make these complicated interactions more manageable, regression trees partition the sample space into smaller, more manageable regions. Then those partitions are partitioned recursively until the complications can be mitigated enough that a simple model can fit the data within each partition. In summary, regression trees recursively partition the global model until a simple model can be fit to each terminal partition.

An advantage of these regression trees is that variables need not be all continuous or all categorical. One can have a mix of these variables and still obtain accurate predictions. Also, predictions are fast since the models at each terminal node are very simple. Moreover, the regression trees give a nonlinear response, thus they can be effective even when the true regression surface is not smooth. However, if the surface is smooth, then the piecewise structure of the regression trees can approximate it arbitrarily closely, so long as there are a sufficient amount of terminal nodes. Shalizi (n.d.)

[1]

3 Predicting MTRs for Supplemental Security Income Program

We now turn to how we used Random Forests to predict MTRs for the Supplemental Security Income program (SSI).

Theoretically, SSI MTRs for all individuals who have no earned income should be 0, since the first $65 of earned income each month is not considered when determining the benefit amount. Also, single individuals with earned income greater than zero can have MTR values of -.5 and 0, since half of earned income that exceeds $65 is excluded. Lastly, the possible SSI MTRs for married individuals are -.25 and 0, since the SSI benefit for married couples is split between spouses.

We provide MTR calculation results derived from programming income rules, and regressions in order to gage the efficacy of Random Forests. First, we train our

[1]For more information on Random Forests visit https://www.stat.berkeley.edu/ breiman/RandomForests/cc_home.htm

Random Forest model to predict the SSI benefit amount for individuals in the CPS-based dataset. We use four different types of income reported by each individual, whether or not they are married, and state residency as the predicting variables. We then use the same earned income adjustment method used in our EITC example to calculate SSI MTRs. The table below represents the weighted average of MTRs of all individuals who have no earned income, those who are single with positive income, and those who are married with positive income. We provide the regression and programmed income rules MTRs, though we do not have the results from Tax-Calculator for comparison since the model is not capable of computing SSI MTRs. The results are given in the following table.

Table 1: SSI MTR for Income Rules, Random Forests and Regressions

SSI MTR By Earned Income and Filing Status			
Income Decile	Program MTR	RF MTR	Reg MTR
Income = 0	0	0	0
Single & Income > 0	-0.2686	-0.4052	0.0715
Married & Income > 0	-0.1572	-0.1291	-0.0570

We see that the Random Forest MTR is far from the programmed income rules MTR but closer than the regression. One possible explanation for this is that the Random Forest accounts for both federal and state SSI benefit changes that result from the earned income adjustment, whereas our Program MTR does not: we programmed income rules for only the federal SSI benefit, because the state supplement rules were difficult to define. This means that the state benefit is not factored into the MTRs of each individual in Program MTR.

Since the Random Forest and linear regression models were previously accounting for both state and federal SSI benefits, and the Program MTR was not, we provide results where these models predict only the federal SSI benefit. We train our Random Forest and regression models to predict federal SSI by using our calculated SSI amount from our programmed income rules model. The MTRs associated with each method

are as follows.

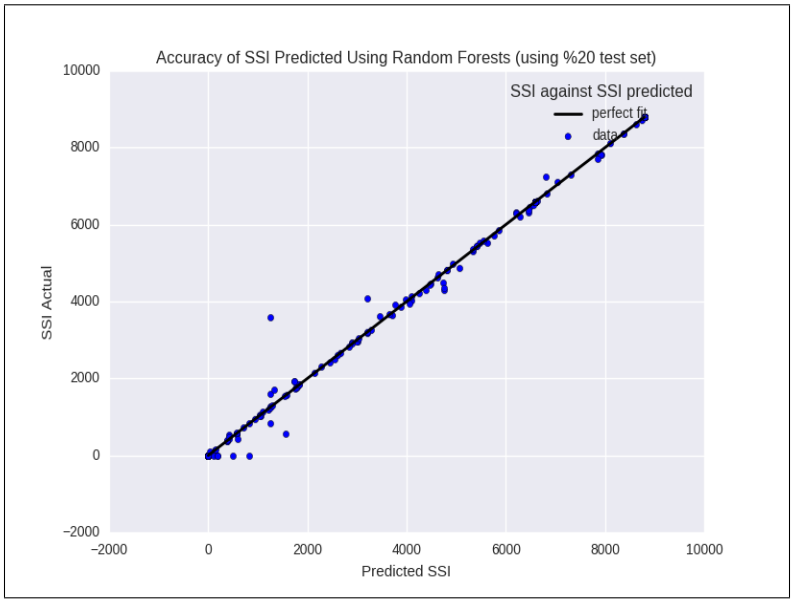
Table 2: SSI MTR for Income Rules, Random Forests and Regressions Using Calculated SSI

SSI MTR By Earned Income and Filing Status			
Income Decile	Program MTR	RF MTR	Reg MTR
Income = 0	0	0	0
Single & Income > 0	-0.2686	-0.2137	-0.1364
Married & Income > 0	-0.1572	-0.1364	-0.1849

We see that the program MTRs and the Random Forest MTRs are much closer in this example, since both are predicting the same SSI value.

To cross-validate the accuracy of the Random Forest's prediction of the calculated SSI value, we consider the following plot of the actual SSI values against the predicted SSI values of the excluded %20 test set.

Figure 1: Predicted SSI Amounts Against Actual SSI Amounts



We see that the Random Forest model predicts the calculated SSI amounts effectively for the the test set respondents: the model's coefficient of determination was

$R^2 = 0.97$

A notable characteristic of Random Forest models is the fluctuation of MTRs with different earned income adjustments. For three different earned income adjustments of $\$1,000$, \$250 and \$1 we derived different MTR results. This effect is not surprising and can be explained by the theoretical behavior of the SSI benefit.

First, since %90 of individuals who receive SSI do not have any earned income, there is a relatively small amount of information pertaining to individuals with positive earned income; this leads to reduced predictive power. Moreover, individuals receiving SSI typically have relatively small amounts of positive earned income.

On the contrary, individuals receiving the EITC benefit must have earned income greater than zero to qualify for the benefit. This creates a richness of information on individuals receiving positive earned income. Thus, for EITC we had the opposing problem. Since all individuals receiving the EITC have positive earned income, a small earned income adjustment produced too small of a change in benefits for the Random Forest to register since Random Forests work on finite piecewise partitions of the data. Thus, obtain accurate implicit MTRs for the EITC by using a larger \$1,000 earned income adjustment. Theoretically, this larger adjustment causes a more detectable change in benefits, which allows the Random Forest to categorize individuals into a different terminal node.

Secondly, large earned income adjustments can affect an individual's earned income exclusions. For example, a large adjustment can potentially push an individual's earned income amount past the \$65 exclusion threshold, which could cause individuals with no earned income to have a theoretically impossible, less than zero MTR. We see this effect in table 3. .

To illustrate these adjustment effects, we provide the Random Forest SSI MTRs calculated using the adjustments of \$1,000, \$250 and \$1.

Table 3: SSI MTR Using Different Adjustment Types For Random Forests

SSI MTR By Earned Income and Filing Status			
Income Decile	RF MTR($1,000)	RF MTR($250)	RF MTR($1)
Income = 0	-0.4009	-0.9005	0
Single & Income > 0	-0.0350	-0.0396	-0.4052
Married & Income > 0	-0.0146	-0.0673	-0.1291

The $1,000 and $250 adjustments produce drastically different results than the $1 adjustment results. As predicted, these two larger adjustments produced the theoretically impossible negative MTR for individuals with no earned income; however, the $1 adjustment produced MTRs that are similar to both the programmed income rules and the theoretical values.

4 Conclusion

Even when datasets are relatively small, Random Forest models can be a simple and accurate alternative to the traditional ways of computing the implicit marginal tax rates associated with government benefit programs.

The SSI MTR results we obtained using Random Forests were more accurate than the regression outcomes, and comparable to the results from programming income rules into a small microsimulation model.

References

Breiman, Leo, "Random Forests," *Machine Learning*, 2001, *45* (1), 5–32.

Shalizi, Cosma, "Lecture 10: Regression Trees," http://www.stat.cmu.edu/~cshalizi/350-2006/lecture-10.pdf.

Svetnik, Vladimir, Andy Liaw, Christopher Tong, J. Christopher Culberson, Robert P. Sheridan, and Bradley P. Feuston, "Random Forest: A Classification and Regression Tool for Compound Classification and QSAR Modeling," *Journal of Chemical Information and Computer Sciences*, 2003, *43* (6), 1947–1958.

Varian, Hal R., "Big Data: New Tricks for Econometrics," *Journal of Economic Perspectives*, March./Apr. 2014, *28* (2), 3–28.

Wager, Stefan and Susan Athey, "Estimation and Inference of Heterogeneous Treatment Effects using Random Forests," *arXiv preprint*, November 2015, *arXiv:1510.04342v*.