Team FitBit - Data Exploration

Rikka, Shiva Ram Reddy (00759744)
srikk1@unh.newhaven.edu

Chilukuri, Gayathri (00755590)
gchil4@unh.newhaven.edu

Potlapelli, Adithya (00761532)
apotl1@unh.newhaven.edu

October 31, 2022

GitHub Repository

1 Introduction

Poverty is one of the topics studied by many economists and data scientists. It is one of the economic issues that most governments wish to address. As one of the world's most powerful economies, the United States faces the challenge of domestic poverty. This project will concentrate on the poverty rate in New York City because it is one of the most representative cities in the United States, as well as having the country's highest income inequality. The project will make use of decision tree model analysis to predict poverty in New York and identify the features that can generate the most accurate predictions.

2 Dataset

The data utilized in this project is derived from the NYCgov poverty measure data, which is generated annually by the poverty research unit of the Mayor's Office of Economic Opportunity. This dataset has 68,644 observations and 79 distinct variables. The dataset includes various characteristics of New York City households (with unique identities), such as education levels, work status, annual income, and so on. Based on these characteristics, a decision is reached on whether or not a particular family is poor.

3 Data Mining - Modeling Technique

A decision tree is a supervised, non-parametric method that uses all available attributes to create the best prediction based on information gain. By computing the impurity of the parent and child nodes, the information gained can tell us the relevance of a specific property. Because we had a relatively large dataset (68644*79), we decided to use the Gini as our cost function for our analysis.

4 Parameters and Hyperparameters

- *criterion:* The function to measure the quality of a split.

- *splitter:* The method used to select the split at each node.
- *max depth:* The name of the hyperparameter max depth refers to the maximum depth to which we allow the tree to grow. The deeper you go, the more complex our model becomes; If we increase the max depth value, the training error will always decrease, but consider the case of testing error; if we set the max depth too high, the decision tree may simply overfit the training data, causing the testing error to increase, but if we set the value of it too low, the decision tree may have little flexibility to capture the patterns and interaction.
- *min samples split:* The minimum number of samples required to split an internal node.
- *min samples leaf:* The minimum number of samples necessary at a leaf node. A split point will be considered at any depth if it leaves at least "min samples leaf" training samples in each of the left and right branches. This has the potential to smooth the model, particularly in regression.
- *min weight fraction leaf:* The minimum weighted fraction of the sum total of weights (of all the input samples) required to be at a leaf node. Samples have equal weight when 'sample weight' is not provided.
- *max features:* The number of features to consider when looking for the best split.
- *max leaf nodes:* Grow a tree with "max leaf nodes" in the best-first fashion. Best nodes are defined as relative reduction in impurity. If None then an unlimited number of leaf nodes.
- *min impurity decrease:* A node will be split if this split induces a decrease of the impurity greater than or equal to this value.
- *class weight:* This actually means that when the algorithm calculates impurity to split at each node, the resulting child node is weighted by class weight by giving the child sample weight, distribution of our classes has been started then the weight of class and then depending on where our tree leans, we can try to increase the weight of the other class so that the algorithm penalizes the sample of one class relative to the other.
- *ccp alpha:* Complexity parameter used for Minimal Cost-Complexity Pruning. The subtree with the largest cost complexity that is smaller than "ccp alpha" will be chosen.

5 Hardware Used

- *Processor:* Intel(R) Core(TM) i5-4210U CPU @ 1.70GHz 2.40 GHz
- *RAM:* 8.00 GB
- *System Type:* 64-bit operating system, x64-based processor

6 Outcomes and Visualization Techniques Used

Before we can create our decision tree, we must ensure that our dataset contains balanced target classes, the appropriate dimensions, and the most representative observations. In our investigation, we used feature selection and instance selection as preprocess approaches. The purpose of feature selection is to remove unnecessary or redundant characteristics in order to

minimize the dimensionality of the dataset. Household Unit ID, Age Category, Poverty Gap, Tax Unit, and other factors are included. We also opted to exclude factors having a significant connection with total income because income and poverty status shared the same feature (low income directly decides whether the person is in poverty or not). If we add total income, the decision tree will select income as a criterion at each node, underestimating the influence of other factors.

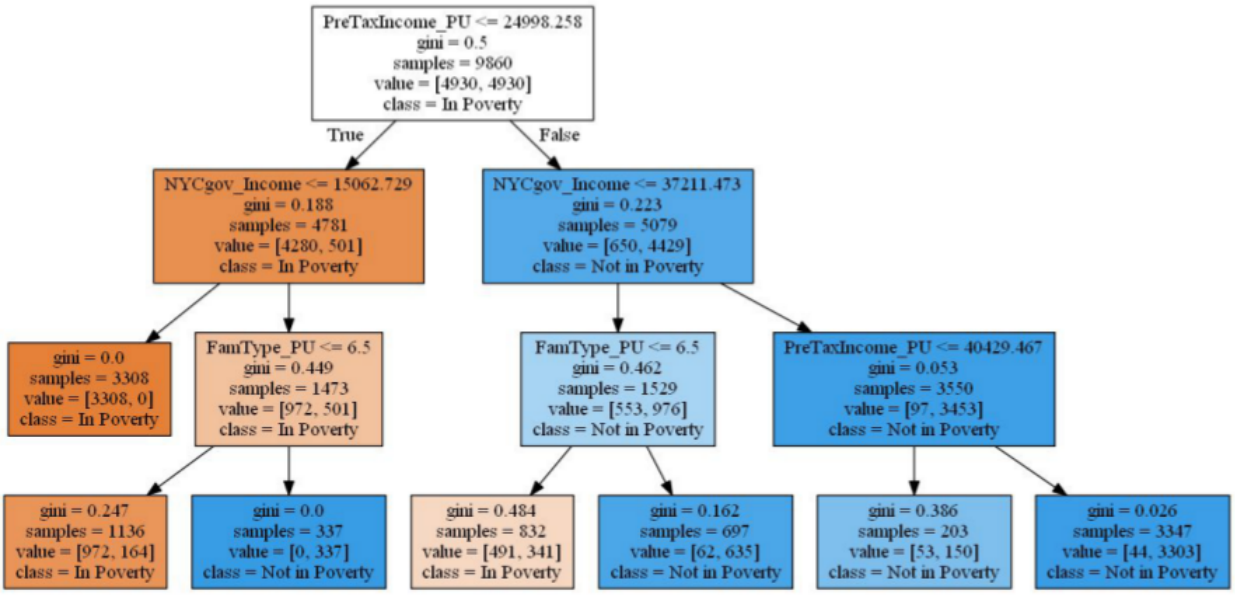

Figure 1: Shows the model will pick Income 4 out of 6 times as the node

The instance selection will choose the most representative observations for the study. Our collection contains information on 70000 people and is organized by household unit. The table below displays the first seven observations from the dataset; observations with the same SERIAL NO indicate that the people belong to the same household. The Poverty Status of each individual is determined by the head of the family (Povunit Rel = 1); if the head of the family is poor, the rest of the family members are poor as well, regardless of other characteristics. In this scenario, we merely used the family head as our unit of measurement.

SERIALNO	AGEP	Povunit_Rel	NYCgov_Pov_Stat
39	51	1	2
55	60	1	2
55	52	2	2
55	26	4	2
55	20	4	2
55	20	4	2
69	39	1	2

Figure 2: Table that shows 7 observations of the dataset

After preprocessing, we built a model with Sklearn and set the max depth to 3 to prevent overfitting. The decision tree is shown in the figure below. The accuracy(measured as the

fraction of correctly classified samples) for the test set is 79%.

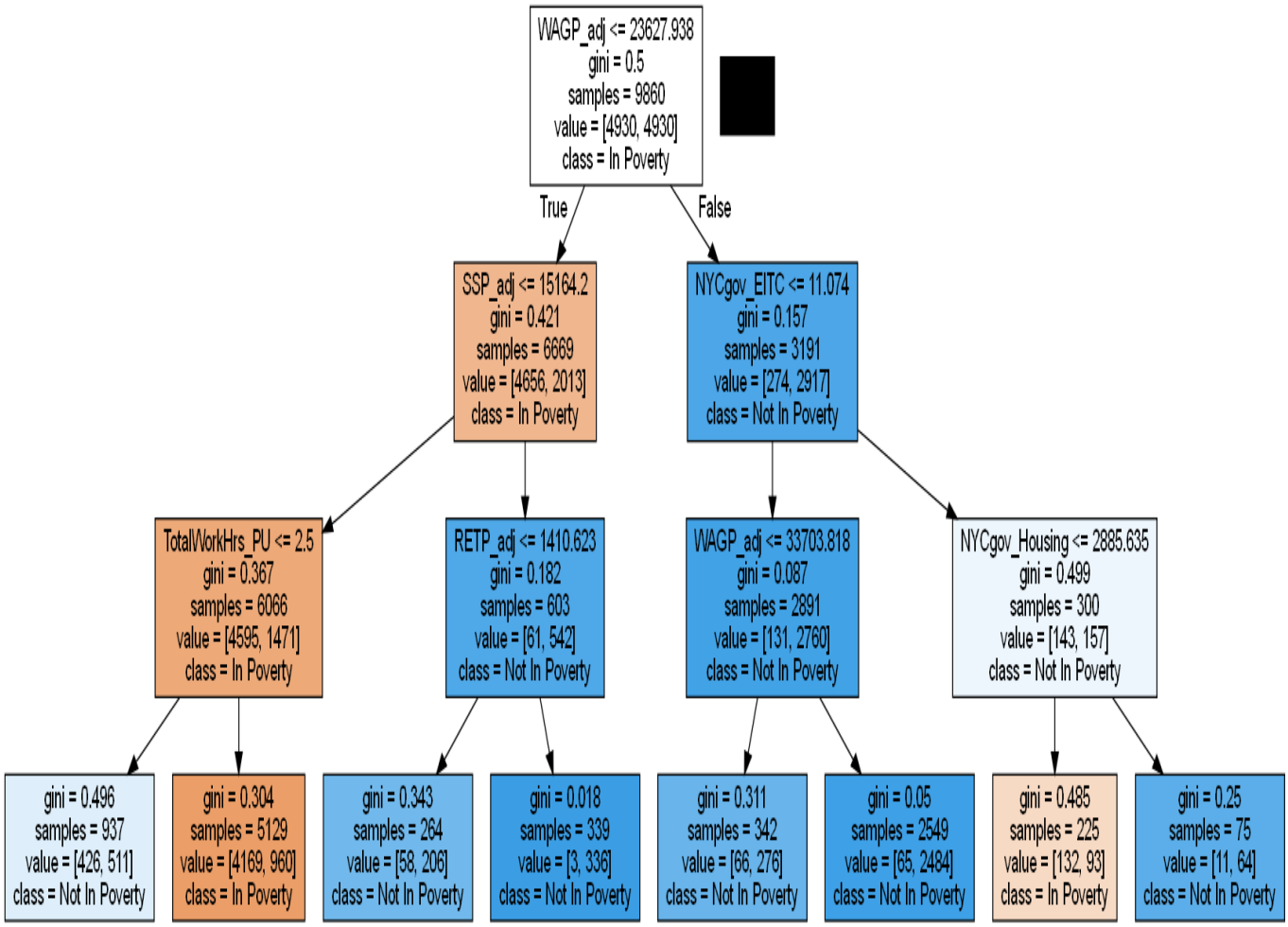

Figure 3: Decision Tree on the dataset

7 Conclusion

It is understood that even if we preprocess the dataset before implementing the decision tree, the variance in the dataset might result in an unstable outcome. Only one tree is insufficient for our research. Instead of just one decision tree, create many decision trees, integrate all of the distinct trees, and base the final prediction on the majority vote to achieve a more robust and significant outcome.