**Global Poverty Analysis Report**

**By Rajib Kumar De**



**As poverty is a distinctly complex and multivariate issue, determining the true underlying causes contributing to it is paramount to addressing it effectively.**

**Executive Summary:**

This document presents an analysis of data concerning socioeconomic indicators of individuals and the probability that those individuals live below the poverty line. The analysis is based on data for 12,600 individuals. The probability of being in poverty was calculated by the Poverty Probability Index, which estimates an individual’s poverty status using 10 questions about a household’s characteristics and asset ownership. The remaining data comes from the Financial Inclusion Insights household surveys conducted by Inter Media.

After exploring the data by calculating summary and descriptive statistics, and by creating visualizations of the data, several potential relationships between socioeconomic indicators and poverty probability were identified. After exploring the data, a regression model was created to predict an individual’s poverty probability.

**Methodology:**

Cleansing, transformation, and analysis of data was performed using Python 3 as needed. Development of machine-learning model was performed in Python 3 ( Jupyter Notebook).

***Process Flow:***

**Model Fine-tuning**

**Model Validation**

**Model Evaluation**

**Model Output**

**Final Model**

**Modeling**

**Variable Reduction**

**Segmentation**

**Variable transformation**

**Sampling**

**Univariate Analysis**

**Collect Data & Identity Y**

**Import libraries**

I generally followed above roadmap for Machine Learning problems (Regression and Classification). For our problem, I followed the steps which required.   
Follow the bellow analysis:

**# Import general useful packages**

import pandas as pd

**# Import all machine learning algorithms**

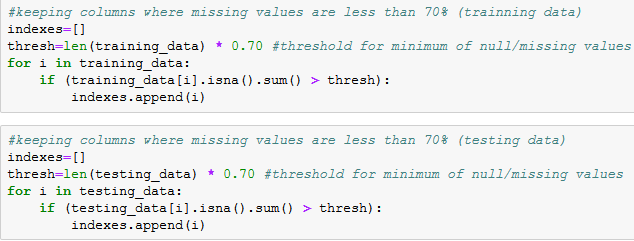
from sklearn.model\_selection import GridSearchCV

from sklearn.ensemble import RandomForestRegressor

from sklearn.ensemble import GradientBoostingRegressor

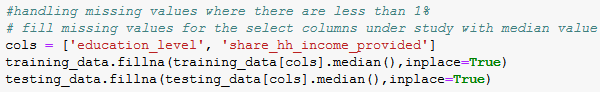
import xgboost as xgb

After importing packages, loaded train and test data and checked shape of the both data set and also checked the missing. YES there were some variables who have missing values and then removed those variables who have > 70% missing for both dataset.

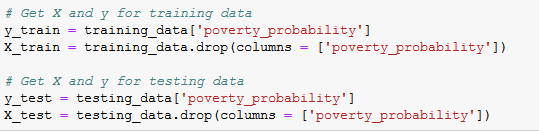


Then again checked missing if there were any missing in the dataset, Yes TWO variables have missing observation: ['education\_level', 'share\_hh\_income\_provided']

Replace missing by MEDIAN as those are categorical observation.

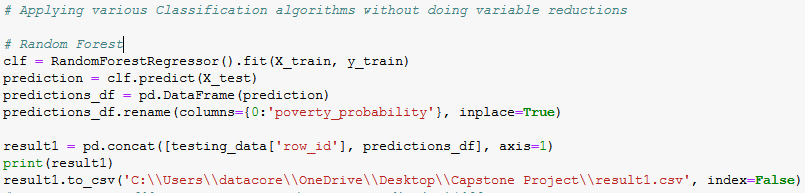


Converted all **Boolean** features **to Integer and separated Independent and dependent feature for both cases.**



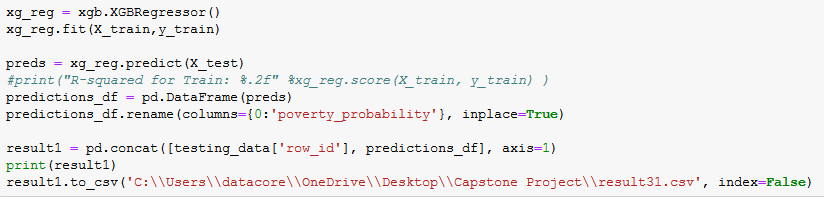
**ML Algorithms:  
  
Here I have applied THREE ML algorithms:**

* **Random Forest**
* **Gradient Boosting**
* **XGBBoosting**

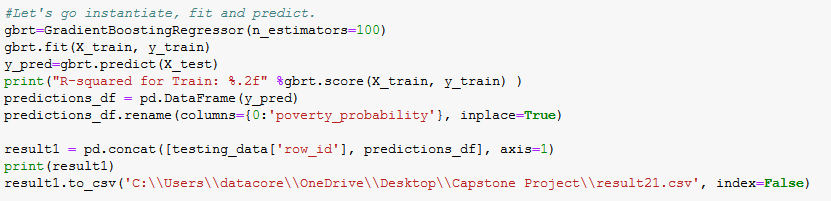
**Using Random Forest:**

Finally, the selected model was modified to be trained on the *entire* training set and then output predicted values of ‘poverty\_probability’ for the test dataset. The model performed suitably in deployment as well, producing a coefficient of determination of **0.3267**.

**Using XGBoosting:**

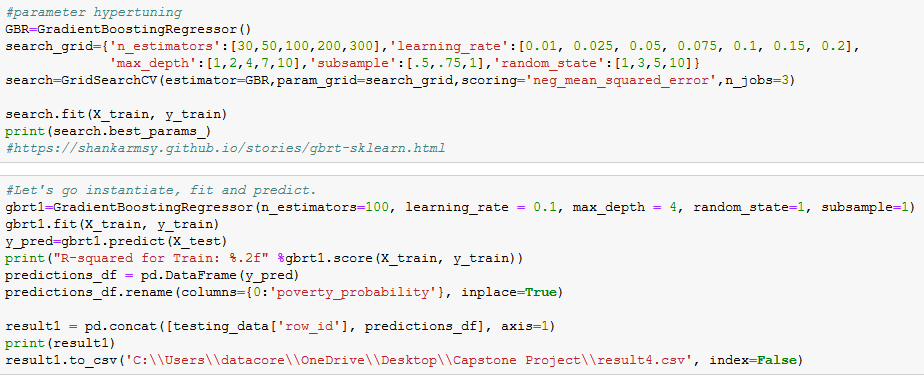


Finally, the selected model was modified to be trained on the *entire* training set and then output predicted values of ‘poverty\_probability’ for the test dataset. The model performed suitably in deployment as well, producing a coefficient of determination of **0.3617**.

Using **Gradient Boosting:**

The selected model was modified to be trained on the *entire* training set and then output predicted values of ‘poverty\_probability’ for the test dataset. The model performed suitably in deployment as well, producing a coefficient of determination of **0.3632**.

**Tune Hyperparameter:**Boosting is a sequential technique which works on the principle of ensemble. It combines a set of weak learners and delivers improved prediction accuracy. At any instant t, the model outcomes are weighed based on the outcomes of previous instant t-1. The outcomes predicted correctly are given a lower weight and the ones miss-classified are weighted higher. This technique is followed for a classification problem



Finally, the selected model was modified to be trained on the *entire* training set and then output predicted values of ‘poverty\_probability’ for the test dataset. The model performed suitably in deployment as well, producing a coefficient of determination of **0.4247.**   
  
Till now I am at No. 7



**Conclusion**

The analysis of the InterMedia survey data and PPI poverty probabilities yielded many interesting findings, and in the end the development of a predictive model determined that just over 40% of the variance in ‘poverty\_probability’ can be determined by characteristic variations within the feature set. While a further investigation into the underlying accuracy of the PPI values themselves may also be worthwhile, the BDT regression model developed in this analysis process is now ready for web-service deployment and could potentially be of use to researchers and activists combatting poverty in these regions.