Poverty Level Characterizations via Feature Selection and Machine Learning

Master's Thesis

Jama Hussein MOHAMUD

Eskişehir, 2019

Poverty Level Characterizations via Feature Selection and Machine Learning

Jama Hussein MOHAMUD

MASTER OF SCIENCE THESIS

Program in Telecommunication

Supervisor: Prof. Dr. Ömer Nezih Gerek

ESKİŞEHİR

Anadolu University

Graduate School of Sciences

September 2019

FINAL APPROVAL FOR THESIS

This thesis titled “Poverty Level Characterizations via Feature Selection and Machine Learning” has been prepared and submitted by Jama Hussein MOHAMUD in partial fulfillment of the requirements in “Anadolu University Directive on Graduate Education and Examination” for the Degree of Master of Science (M.Sc.) in Electrical and Electronics Engineering Department and has been examined and approved on 23/09/2019.

Committee Members	**Title - Name Surname**	**Signature**
Member (Advisor)	: Prof. Dr. Ömer Nezih Gerek	.....
Member	: Assoc. Prof. Dr. Tansu FILIK	.....
Member	: Assoc. Prof. Dr. Semih ERGIN	.....

Prof. Dr. Murat TANIŞLI

Director of Institute of Graduate Programs

ÖZET

Yüksek Lisans Tezi

ÖZNİTELİK SEÇİMİ VE MAKİNE ÖĞRENMESİ İLE YOKSULLUK SEVİYE KARAKTERİZASYONU

Jama Hussein MOHAMUD

Anadolu Üniversitesi, Fen Bilimleri Enstitüsü

Elektrik-Elektronik Mühendisliği Anabilim Dalı

Eylül 2019

Danışman: Prof. Dr. Ömer Nezih Gerek

Yoksulluk seviye tespiti hassas, güncel ve güvenilir sosyo-ekonomik hane halkı verisine ihtiyaç duymaktadır. Öte yandan çoğu gelişmekte olan ülkede gelir, tüketim, yaşam türü verilerin güvenilir olarak elde edilmesi zor ve pahalıdır. Bu tür veriler uzun ve detaylı anketler gerektirir. Gelişmekte olan ülkelerin bu tür veri elde etmelerindeki güçlük nedeniyle veri azdır; bu da hane halklarına yönelik iyileştirme politikalarını geliştirme aşamasında zorluklara neden olmaktadır. Zira, ekonomik karakteristiklerin hassas ölçümü toplum politikaları üretmek açısından elzemdir. Bu tür durumlarda makine öğrenmesi yaklaşımlarını ele almak son derece faydalı olabilir. Maalesef makine öğrenmesi algoritmaları genel olarak "kara kutu" formatında olup, gerçekleştirdiği öğrenmenin ve sınıflandırmanın hangi parametrelere ve özniteliklere dayandığı çoğunlukla belirsizdir. Detay vermek gerekirse; bir hanenin yoksul olarak categorize olmasına neden olan niteliklerin neler olduğu konusunda makine öğrenme yöntemleri doğrudan sonuç üretmemektedir. Bu nedenle, bu çalışmada yoksulluk konusuna sadece "gelirin belli seviyenin altında kalması" şeklindeki tek boyutlu yaklaşımı kullanmayarak, bunun yerine çok boyutlu bir perspektif ele alınmaktadır. Yöntemimizin uygulaması ve faydalı olup olmadığı Inter-Amerikan Gelişim Bankası tarafından Kaggle'a yüklenen Kosta Rika veri seti üzerinden değerlendirilmiştir.

Anahtar Kelimeler: Yoksulluk karakterizasyonu, Yoksulluk ölçümü, Çok boyutlu yoksulluk, Öznitelik çıkarımı, Makine öğrenmesi

ABSTRACT

Master of Science Thesis

POVERTY LEVEL CHARACTERIZATIONS VIA FEATURE SELECTION AND MACHINE LEARNING

Jama Hussein MOHAMUD

Department of Electrical and Electronics Engineering

Program in Telecommunication

Graduate School of Sciences, Anadolu University,

September 2019

Supervisor: Prof. Dr. Ömer Nezih Gerek

Targeting poverty requires access to accurate, timely and reliable quantitative data on socio-economic characteristics of households. However, in many developing countries, collecting accurate, timely, and reliable data on household characteristics is expensive, time-consuming, and unreliable, often requiring long and detailed surveys. Reliable data on economic status remain scarce in developing countries, hampering efforts to study these outcomes and to design appropriate policy responses to improve household welfare. In such situations machine learning algorithms can be of a great help. However, these models are normally designed in the form of black boxes; if the model is trained on a certain known data and predicted on unseen data, it doesn't give any information about the features that discriminate between classes. In other words, it is very tough to extract the features indicating that someone falls under specific category of poverty. Moreover, in poverty identification, measurement or classification, it is crucial to know how such features contribute to each class of poverty. Therefore, we designed an approach that extracts a subset of features that best characterize each poverty class, examines how this subset affect the chosen class and finally employ ensemble models to best classify between these classes. Through this approach we look at poverty from a multidimensional perspective contrary to a single dimension perspective defined as living on consumption expenditure of less than a predefined income threshold. The application and usefulness of our proposed framework is tested on a Costa Rican dataset collected from Kaggle website and provided by Inter-American Development Bank.

Keywords: Poverty Characterization, Poverty Measurement, Poverty Identification, Multidimensional Poverty, Feature Extraction, Machine Learning.

ACKNOWLEDGEMENTS

First, I would like to express my eternal gratitude to my advisor Prof. Dr. Ömer Nezih Gerek for his endless, untiring support, enthusiasm and patience throughout my research. Without him it would have been tough to successfully complete my research work. His door was always open whenever I had a question to ask. I couldn't have imagined having better advisor and mentor.

I am equally thankful to Abdikani Hassan, Sanday Amos and Doungahiré Abdoul Karim ZANHOUO for their academic help and mentorship which bolstered my understanding in the field of poverty studies; an area that was new to me as an Electrical and Electronic Engineering graduate student.

I am also thankful to the Turkish Government Scholarship Body (YTB) for giving me this transformative opportunity. Without them it would have been difficult to reach this point. I also owe much gratitude to my fellow classmates whom we spent most of our time exchanging ideas, solving technical problems and all the beautiful moments we shared for last 2-3 years.

I am, moreover, very beholden to my technical advisor at work Dr. Celalettin Bilgen for introducing me to various areas in artificial intelligence. His guidance through every project will be unforgotten. I also would like to express my gratitude to my two language teachers Dr. Nilay Girisen and Dr. Kevser Candemir for teaching me Turkish language; a language that opened my mind and allowed me to meet new people. May you all live longer with joy and happiness.

Finally, I must express my sincere gratitude to my parents, especially my mother for her continued prayers, support, and encouragement throughout my studies. This achievement wouldn't have been imaginable without her.

Jama Hussein Mohamud

STATEMENT OF COMPLIANCE WITH ETHICAL PRINCIPLES AND RULES

I hereby truthfully declare that this thesis is an original work prepared by me; that I have behaved in accordance with the scientific ethical principles and rules throughout the stages of preparation, data collection, analysis, and presentation of my work; that I have cited the sources of all the data and information that could be obtained within the scope of this study, and included these sources in the references section; and that this study has been scanned for plagiarism with "scientific plagiarism detection program" used by Anadolu University, and that "it does not have any plagiarism" whatsoever. I also declare that, if a case contrary to my declaration is detected in my work at any time, I hereby express my consent to all the ethical and legal consequences that are involved.

……

Jama Hussein Mohamud

23/09/2019

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

SYMBOLS AND ABBREVIATIONS

AI	: Artificial Intelligence
ML	: Machine Learning
B40	: Bottom 40 percent
IHDS	: Indian Human Development Survey
PMT	: Proxy Means Test
tSNE	: t-Distributed Stochastic Neighbor Embedding
PCA	: Principle Component Analysis
GBM	: Gradient Boosting Models
HH	: Household
CSFS	: Class Specific Feature Subset Selection
CEFS	: Class-Specific Ensemble Feature Selection
SMOTE	: Synthetic Minority Over-sampling Techniques
CFS4	: Feature subset selection method based on sparse similar samples
SFFS	: Step Forward Feature Selection
SBFS	: Step Backward Feature Selection
DNN	: Deep Neural Networks
LIME	: Local Interpretable Model-agnostic Explanations

1. INTRODUCTION

Poverty is state or situation where an individual or household lacks usual or basic needs for food, clothing, and shelter. Even though people who live in poverty are conventionally classified as poor and non-poor, the nature of poverty is categorized into many forms: extreme poverty, moderate poverty, relative poverty, non-vulnerable etc. Traditionally, organizations such as World Bank define poverty as living on consumption expenditure of less than 1.9 dollars a day for low income countries and 2.5 dollars for advanced countries. However, Amartya Sen's arguments has totally reshaped the way poverty is conceptualized in the literature. According to Sen, poverty has two distinct difficulties; (I) poverty identification, and (II) aggregating indicators of poverty to construct an index for poverty measurement [1]. For long time, income was used to overcome the first issue, but second issue has forever been a long-argued topic in academia and poverty research [2]. Sen and many other researchers acknowledge that poverty is a multidimensional concept that is based on the deprivations of many indicators and dimensions. They argued that defining poverty as unidimensional (i.e. income) is unrealistic and will not accomplish the desired and effective solution to poverty.

This multidimensional view enhanced both theoretical and empirical research in the area of poverty measurement. This has led to the identification of dominant dimensions of poverty. The identification of multiple dimensions of poverty provides an important information for the design and implementation of socioeconomic policies aimed at providing a realistic solution to reduce the degree of poverty globally. Unlike unidimensional framework which depend only on income or expenditure, multidimensional concept requires multidisciplinary analysis.

One of the early multidimensional poverty measures developed is the concept of fuzzy sets. This multidimensional framework analyses a vector of features/indicators that are indicative of deprivation or poverty. They are expressed as N-order vector of variables X = (X1, X2, X3…. XN); these variables include education, healthcare, and basic needs. The choice of these variables was fundamental step in developing a dependable multidimensional framework [3].

Measurement or targeting poverty is very crucial in order to overcome the difficulties faced by poor people. Globally it is recorded that almost half of the human population are living under the poverty line. Many of these are classified as subsisting in extreme poverty. Almost a billion children are living in poverty worldwide. According to

UNICEF, 22,000 children die each day due to poverty. Similarly, [4], according to a recent report in their global multidimensional index, around one and a half billion individuals from 103 countries are poor multidimensionally. These statistics show that 48% of these people live in southern Asia and rest in Sub-Saharan Africa. Following their results 72% of these MPI poor people are resident in middle income countries. And approximately ½ of these people are children.

Similarly, poverty is a heterogenous problem that varies through population, geographical location and time. For instance, a poor person in Africa has different deprivation characteristics than one in Asia or America. It has also been observed that children experience poverty in a way that is different from that of adults. Reference [5], quotes children living in poverty, bereaved of the physical, spiritual and emotional resources needed to develop, endure and prosper. They are deprived of their rights to a normal life, making it difficult for them to pursue their dreams as well as not being able to enjoy social equality. Due to age, dependency and vulnerability, children experience poverty in a more severe manner than adults. Additionally, child poverty has different causes and effects than that of a grown person and this can lead to a calamitous effect on the children. In [5], they compared multidimensional child poverty and single dimensional (income) poverty on a country level. Their results show that children experience more deprivations on every aspect (see **Fig 1**).

Technological advances to overcome these miserable conditions are matters of paramount importance. Specifically, a well-designed program that could help aid organizations to address poverty is highly needed.

In general, as mentioned above, there are fundamentally two ways of measuring poverty, the unidimensional method (monetary based) and multidimensional approach (non-monetary based). The monetary method identifies the poor by checking if the income level of the individual drops below a certain threshold. The non-monetary method measures poverty by taking many dimensions (including basic needs, education and healthcare) into consideration. The next section briefly discusses studies based on these measurement methods.

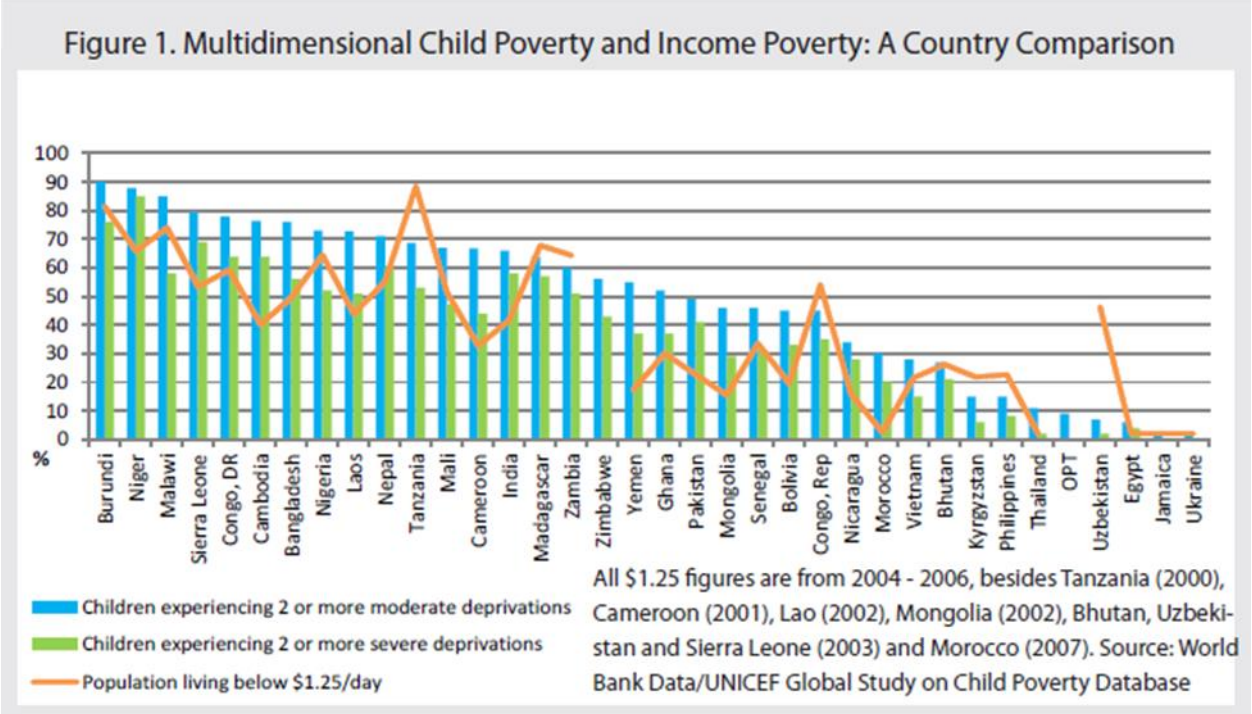


Figure 1: A country comparison of Multidimensional and Unidimensional child poverty (Image taken from [5] report)

1.1 Poverty Measurement

In poverty measurement, economists have long been studying effective ways to measure poverty. Income and consumption expenses were considered as an alternative to estimate household's economic conditions. However, income and consumption expenditure has been widely viewed as unreliable towards poverty measurement [2].

For this reason, most researchers conclude that poverty is a deprivation of many dimensions and that unidimensional measurement of poverty is a form of measurement that provides insufficient information on poverty status. This led practitioners to develop many indices to measure poverty. Some of the common ones include; Bourguignon and Chakravarty family (2003), Fuzzy theoretic approaches, Global multidimensional index (Alkire-Foster) and others [6][2][7][8][4]. The construction of all these measurements was influenced by Sen's capability approach which conceptualizes individual's well-being as a mixture of various functionings [1]. A functioning is an accomplishment of that person; what he/she chooses to do or to be, and projects are a part of the situation of that person. These "functioning" are the elements of an individual's quality of life and measurement must be based on valuing these functioning vectors. In other words, Sen proposes the measure of well-being to be based on dimensions (functioning).

1.2 Machine Learning Application to Poverty

Machine learning is the branch of artificial intelligence (AI) that allows systems to learn automatically from experience without being explicitly programmed. To put it in another way, the ability that lets the systems to think or learn intelligently is called machine learning. Machine learning has been successfully applied to many problems ranging from medical images, weather forecasting, spam classification, cancer analysis recommendation problem, marketing and many more. In this context we are going to discuss how machine learning algorithms could help in targeting poverty.

To help alleviate poverty we first need to recognize the main causes of poverty; these includes, conflicts, security, inaccessibility to social assistance, floods, lack of basic needs (e.g. food, water, education), lack of skills etc. By employing machine learning and artificial intelligence, a system that can handle either one issue at a time, a few at a time or all at once, can be developed. Satellite images, household surveys, social media data, mobile call records and other forms of data are some of the input that machine learning algorithms can adopt and learn from. Traditionally, household data has been frequently used for poverty assessment but recently, researchers utilized satellite images and mobile data to extract information about poverty in specific areas.

In [9], Machine learning algorithms and satellite images are coalesced in order to dispense socioeconomic features of poverty and wealth. In other words, they utilized combination of satellite daytime images and satellite night light images, assuming the areas that are bright at night are richer than those that are not bright. With this assumption they extracted features in the daytime images that are correlated with economic progress. Another promising approach, [10], has utilized mobile data to see if it reflects the individual's socioeconomic status. They used a database consisting of records of billions of interactions on Rwanda's largest telecommunication industry and phone survey data. These approaches have a number of limitations, but they all show that poverty can be examined from different angles.

Machine learning models could also help combat poverty through improvement of education. Humans acquire knowledge in different ways; some people learn through listening, some are visual learners, some learn through reading while others learn best through skill application. The current education system is based on one form which doesn't benefit all students. Luckily, machine learning could help categorize students' learning desires and improve the process.

Another way in which AI applications could help tackle poverty is through improving agriculture. AI experts at Carnegie Mellon University introduced a project using robotics and artificial intelligence to enhance the sustainability of food crops in emerging countries. The researchers have also studied specific types of crops using drone technology, robotics and machine learning models to improve the growth of these crops. Machine Learning accomplishes this by feeding the data collected over the growing season into an AI model that could help predict the best ways for farmers to grow this crop. These studies and many other prove that a machine learning can be employed to help tackle poverty.

In this manuscript we employ machine learning algorithms to study characteristics or dimensions that a household is deprived of. We will also try to best classify the different levels of poverty. Our study is based on a Costa Rican dataset that was obtained from American International. In our study, we are looking at poverty from a multidimensional perspective, which clearly illustrates that "well-being" consists of many dimensions and cannot easily be apprehended based on only an economic measure of income or wealth. We are assuming (like other researchers in this area) that poverty is an indicator of deficient well-being, which depends on both monetary and non-monetary variables. And that income as a sole predictor of poverty is inadequate and should be accompanied by other characteristics e.g., housing, knowledge, lifespan, delivery of public goods and so on.

2. LITERATURE REVIEW

Poverty is a catastrophic situation that needs the usage of all the tools in our disposal to come up with an effective solution. In the literature, several approaches have been proposed to tackle the problem. One of the earlier approaches, as mentioned above, is the concept of unitary measure [11]. This method is very clear as it uses a single dimension for poverty measurement. In other words, poverty is viewed as an economic problem that depends only on income or expenditure and by solving the income issue of the household, could bring an ideal solution. A study made by 12 European countries illustrate that income-based analysis of poverty provides only a fractional insights of poverty state [11]. Other poverty researchers have also been debating this unitary concept for the past few decades. Finally, this brought the concept of multidimensionality to be widely adopted by economists and researchers which led to the developments of several indices for poverty measurement.

Many poverty researchers have insisted on redefining poverty in a multidimensional way rather than unidimensional way. But, yet, not all the researchers have included all the various dimensions of poverty in their measurement indices. Most of the methods used so far consists of aggregating several dimensions into a single index and defining poverty line and related measures on the basis of that index [2]. This thesis [2] proposes another approach that considers the multidimensionality of poverty but constructs index or poverty line for each dimension separately. And if someone falls below one of the poverty lines that person will be considered poor. The paper also talked about a way to combine the various poverty lines and connected dimensional gaps into multidimensional poverty measures.

An Approach by [12] proposes a poverty measure that is additively decomposable in the sense that overall deprivation is a weighted mean of subclass poverty levels. The term decomposable here refers to dividing or breaking down the population in subgroups/classes (e.g. ethnic, geographical). In short, their research proposes (I) a measure that is additively decomposable and leads to a decrease in overall poverty if the level of poverty in one of the subgroups declines. II) an approach that satisfies Sen's capability approach (III) a method that is justified by a relative deprivation concept of poverty [12].

S. Alkire and M. E. Santos [6] proposes a new Multidimensional poverty index to measure poverty. The approach practices the concept of non-unitary poverty

measurement. As mentioned in their study, the index has been tested on over 100 developing countries by using a household data [6]. Quoting from their paper their approach (I) attempts to measure the extent of poverty in the emerging countries. (II) attempts to reduce data limitations and finally (III) has underlying concept of extreme poverty. As can be seen on **Figure 2**, their index is based on three dimensions (living standard, health, education) and these are measured by the aggregations of 10 indicators (as indicated right side of the **figure**). However, since the approach requires a specified number of dimensions and indicators to be present in the survey data, we might need to use a data that has these features in order to deploy the index. In some situations, the nature of the available datasets will not allow us to achieve or extract some of these indicators. But, fortunately, machine learning algorithms can adopt the structure of any dataset, learn from it and be able to predict the unseen conditions of those deprived.

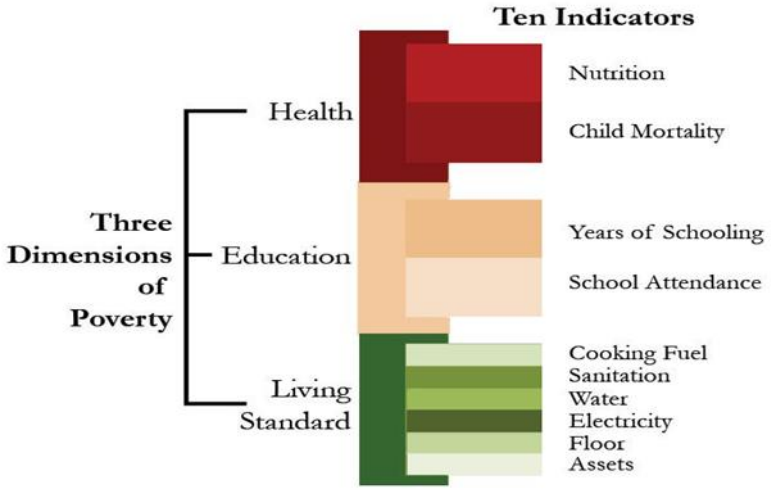


Figure 2: Multidimensional Poverty Index

The concept of fuzzy sets has also been utilized by many researchers to measure poverty and classify the difference between poor and the rich. The fuzzy set approach allows us to (I) measure household's level of deprivation (II) guesstimate the average poverty index of the population and (III) measure how much they are deprived from each dimension or attribute [13]. Another study that utilizes the fuzzy concept also proposes a method to measure the poverty index in a fuzzy environment via a two-step membership function [14]. They used linguistic variables to find the membership values.

N. S. Sani et al. [15], analyses machine learning models (such as Naive Bayes, Decision Tree and k-Nearest Neighbors) that best identifies the B40 – Bottom 40 percent – population. In their study, they categorized the household population that falls under category B40. They performed several data preprocessing before proceeding to modeling. They have also used sampling methods to balance the training data. And their results which is based on 10-fold cross validation show that decision tree performed better among the others.

Another study, [16], shows how different variables describe falling into and escaping from poverty. Their study which is based on a data from Indian Human Development Survey (IHDS) database show that some attributes will trigger some households to fall into poverty and others to escape from it. Since the database consists of data that is collected through different periods of time, machine learning algorithms can easily learn how to categorize the respective strength of each household feature.

PMT (proxy means test) models have also been utilized as an approach to measure inequality. PMT models considers various observable features of the household to measure deprivations of household when income data is not available [17]. Assumptions is made based on the household characteristics, for example, a family living in a brick walled home will probably be having more sustainable life than one living in a house made of clay. In this case the "type of a house" is used as a proxy to measure income.

As stated by [17], PMT solves two problems that are related to the assumptions we mentioned above: (I) there is no proof to the assumed informed guess; (II) even if the assumption is correct, we don't know the degree of poverty between the two families. The PMT overcomes these two complications by using actual quantitative data that is collected from the household. since the data collected has both household characteristics and consumption, statistical methods are used to measure the relationship between household's characteristics and wellbeing. Such statistical methods that may have been utilized include multiple regressions; regressions are models that allows us to estimate the relationship of many variables to a target variable.

In [18], Machine learning algorithms were utilized to boost the performance of PMT models. They argue that effective poverty targeting tools should increase out-of-sample performance. Machine learning models are known to perform well in such cases when there is enough data. They claim to have used stochastic ensemble models and achieved an accuracy improvement of about 2 to 18 percent.

All these studies prove that various tools can be used to tackle the poverty. Unfortunately, neither of those approaches offer a comprehensive solution to poverty, each focusing only on a specific aspect. In our study, we neither endorse nor verify the performance of the above-mentioned methods, however, we assume that the data is labeled through any kind of multidimensional approach. Our framework focuses on obtaining the features that best characterize each class. To the best of our knowledge, such an approach has not been utilized to poverty targeting. Our proposed method can be applicable to any dataset regardless of the model deployed.

3. DATA

Data scarcity is one of the biggest reasons that poverty cannot be easily tackled by researchers and experts. The lack of reliable data in emerging countries has become a major complication towards solving the problem of poverty. Poverty data is tough, scarce and resource intensive to obtain. Obtaining data that has poverty related characteristics would be very significant for accurate measurement, policy making, resource allocations, and intervention purposes. For countries in Africa, the shortage of statistical data is a huge challenge for organizations trying to provide social assistance to poverty-stricken areas.

However, in rich nations, new sources of data such as data collected through social media, internet-of-things, etc, has provided them with new approaches of poverty estimation and measurement [19]. But still, since poverty is a heterogeneous problem, measurement of individual/household poverty levels is tough even in rich countries. Therefore, heterogenous data and a cheap technological approach that considers both households and nation is very momentous towards solving the calamitous problem of poverty.

So far, various datasets have been used for poverty measurement. These datasets include; household-based datasets, satellite imagery datasets, night luminosity datasets and datasets collected through mobile devices and other data as well. Most of the researches based on datasets other than household data view poverty from a single dimension; Income-based. In other words, their studies don't show a clear definition of poverty in the multidimensional perspective. For example, dimensions like education, basic needs and healthcare cannot be extracted using their approach. Assume someone living in Africa has a 100 camels and thousands of goats; measuring poverty using methods that are based on his roof and some other environment-related characteristics would be totally biased. This limitation applies to all studies that utilize night lights and satellite imaging without direct combination of household datasets that have all the other characteristic attributes of deprivations.

In our study, we try to contribute to the existing literature by preserving the multidimensionality concept. We base our research on a dataset that has no income related indicators. Our data does not have household-economy related variables but contains very informative details of household situations. The nature of our data was mainly categorical, having few features that are continuous. It consists of households' observable characteristics and other variables that were asked to the respondents. The dataset is

drawn from Kaggle –which is an online web challenge community of data scientists and machine learners. It is based on Costa Rican dataset and provided by Inter-American Development bank. It comprises of four different classes that include; Non-vulnerable, Vulnerable, Moderate and Extreme poverty. The data shows that different attributes contribute differently to the levels of poverty and that deprivation is a function of multiple variables.

The multidimensional measure of poverty consists of diverse forms of dimensions that all contribute to the household's well-being. Some of these dimensions can be Current assets (education, skills, health), Social capital (social network, trust, relations), Physical capital (infrastructure, technology), Natural resources (such as wood or land) and monetary (income, remittances, savings, credits and debts). However, since we depend on this specific dataset, we can only extract the dimensions available in our data.

3.1 Data Preprocessing

Data preprocessing is one of toughest but very essential steps in machine learning pipeline or Artificial Intelligence in general. In today's real-world, data is highly vulnerable to missing, different type of noise, changeability because of its massiveness, collinearity, unbalancedness and skewness. Big data will always have these properties and overcoming them is very crucial in order to develop reliable models.

Data Preprocessing is a technique of transforming raw data into a comprehensible format. It has been practiced and proven that preprocessing stages can solve many problems in median and big data. The preprocessing steps generally follow five steps; Data cleaning, Data Integration, Data Transformation, Data Reduction, and Data Discretization (see **Fig 3.1**).

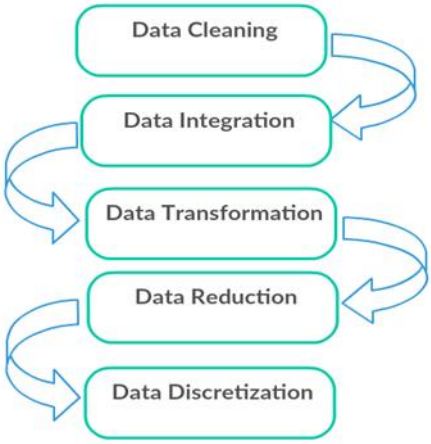


Figure 3.1: Data Preprocessing

3.1.1 Data Cleaning

It is the most important step in data preprocessing. It is the process of dealing with missing values, noise variables, identifying & removing outliers, and resolving disparities. In our case, we had a couple of variables that had missing values (**see Table 1**). Generally, these features can be imputed with mean, mode or medium; or the one that gives you high performance or a combination of all. But first we went back to the documentation and found out why some of these variables had missing values. For example, rez_esc, v18q1 and v2a1 stand for; years behind in school, number of tablets household owns, and monthly rent payment respectively. According to the web challenge discussion where this challenge was hosted; we learn that the first feature that has the highest missing values (rez_esc) is defined only for individuals between the age of 7 and 19. Which means any age that is above or below this range has no years behind in school. Therefore, we assigned any age that is not in this range to 0. Similarly, we learn from the comments that the maximum value of this feature is 5. Therefore, any value above 5 is set to 5, and the rest of the missing values of this feature (which was very small about 3.5 %) is set to the mean value of the feature.

The second feature that has the highest missing value is v18q1 (number of tablets household owns). luckily, we have another feature (v18q) that is 1 if individual owns a tablet and 0 if not. And, fortunately, the v18q has no missing values which solves our problem. which means every household that has a missing for v18q1 does not possess a tablet. Therefore, we filled in this missing value with zero.

The last feature with the highest missing value is v2a1, which represents rent payment, to figure out the reason, we checked “tipovivi_” which shows the ownership &

rent of the home. According to the data description, the feature has 5 different values described as below;

tipovivi1, =1 own and fully paid house

tipovivi2, "=1 own, paying in installments"

tipovivi3, =1 rented

tipovivi4, =1 precarious

tipovivi5, "=1 other (assigned, borrowed)"

So, we found out that families that don't pay the rent mostly own the house. This resulted having small proportion of missing values. Finally, we imputed the rest of the missing values with the mean, medium and mode depending on the type of the feature (i.e. numerical, categorial, float, etc.)

Table 3.1: Missing Values

	Missing Values	% of Total Values
rez_esc	27581	82.5
v18q1	25468	76.2
v2a1	24263	72.6
meaneduc	36	0.1
SQBmeaned	36	0.1

3.2 Data Analysis and Visualization

In this section, we are going to analyze the data, show it is complexity, see the disproportions of the class distributions and many other graphs and tables for envisioning and getting enough intuition of the data we are working on. The dataset is composed of two separate files; one test file and one train file. The train portion has 9557 examples (rows) and 143 features (columns), while the test portion contains 23856 examples and 142 features. Since our data is a supervised multiclass problem, the additional column in the train data consists of the target values (Labels). Each observation embodies one individual and each variable (column) characterizes the individual or the household.

Table 3.2 shows the first five examples and a few features from our train data.

_Table 3.2: First five examples of the train data

	Id	v2a1	hacdor	rooms	hacapo	v14a	refrig	v18q	v18q1	r4h1	r4h2	r4h3	r4m1	r4m2	r4m3	r4t1	r4t2	r4t3	tamhog	tamviv	escolari	rez_esc
0	ID_279628684	190000.0	0	3	0	1	1	0	NaN	0	1	1	0	0	0	0	1	1	1	1	10	NaN
1	ID_f29eb3ddd	135000.0	0	4	0	1	1	1	1.0	0	1	1	0	0	0	0	1	1	1	1	12	NaN
2	ID_68de51c94	NaN	0	8	0	1	1	0	NaN	0	0	0	0	1	1	0	1	1	1	1	11	NaN
3	ID_d671db89c	180000.0	0	5	0	1	1	1	1.0	0	2	2	1	1	2	1	3	4	4	4	11	11.0
4	ID_d56d6f5f5	180000.0	0	5	0	1	1	1	1.0	0	2	2	1	1	2	1	3	4	4	4	11	11.0

Further explanation of all the variables can be found on the Kaggle website but a brief description is given as follows; the Id column denotes a unique identifier for the individuals in the household. The “Idhogar” column is a unique identifier for the household. This feature can further be used for the aggregation and grouping of individuals that share a common household. The “Parentesco” specifies if the person is the head of the family/household. And finally, Target feature is the label and it should be equal for all inhabitants in the household.

To get a very good intuition of the data we are working on, we are going to perform some exploratory data analysis in order to inspect if there are patterns, inclinations, correlations, variances in our data. After we get enough insight from our data, we will demonstrate some of the feature engineering techniques we have used in building a consistent model. Feature engineering is a technique that is considered the most important part of any machine learning pipeline or problem. So, performing very good feature engineering tactics will always improve the performance of machine learning models.

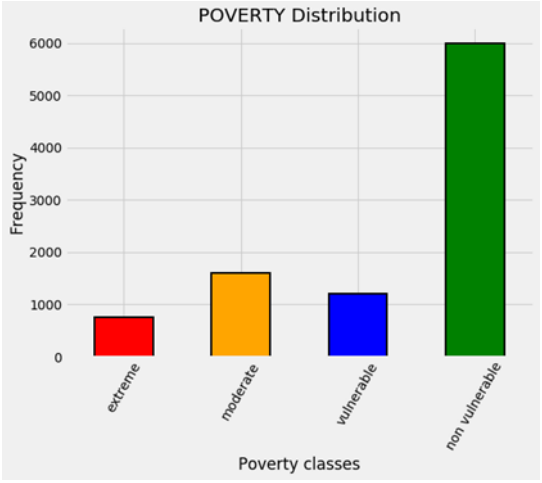


Figure 3.2: Distribution of Poverty Classes

The dataset is very skewed where minority classes are less represented (see **Fig 3.2**, showing the frequency distribution of the data). This leads to an untrustworthy situation where models might overfit the data. Data disparity is one of the main problems in data science and machine learning. This problem occurs when classes are not represented equally in the dataset. For example, assuming you have a binary classification problem with 200 examples; and 180 of those examples are labeled in one class, while the rest 20 examples are in the other class. This makes approximately a ratio of 8:1 making it difficult for the model to discriminate between the classes. In our problem – the poverty problem, the disproportion of the data is very critical if we want to develop realistic models. There are couple of ways to tackle the imbalanced problem; some of them include; obtaining more data where the classes are equally proportional, changing the evaluation metric or performing sampling.

Another complexity, other than data disparity, is the class memberships. **Fig 3.3** show tSNE plot of the data; a data visualization tool that projects high dimensional data into a low dimensional space. is a non-linear representation of the data, unlike PCA which is a linear projection. It utilizes the **local relationships** between feature points to create a low dimensional mapping. This allows it to capture **non-linear structure [20]**. Due to complexities in the distribution of the data, some regression models will fail to produce a classifier that perfectly suits our data.

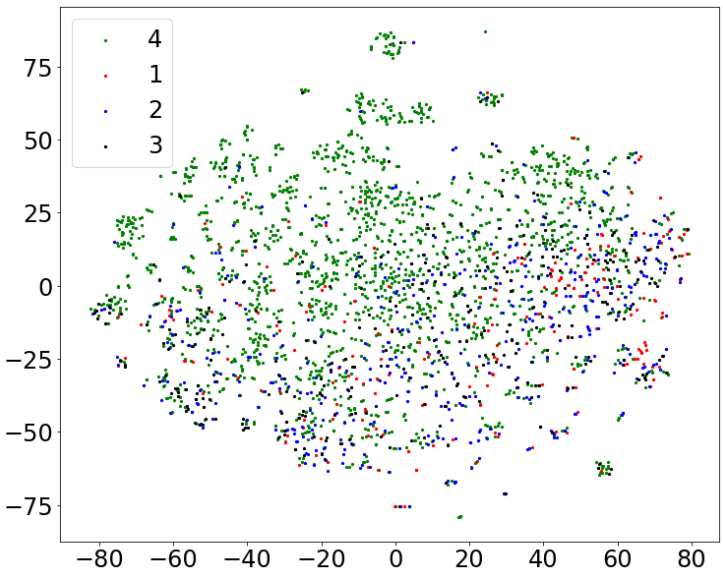


Figure 3.3: t-SNE Visualization of poverty classes (1 = Extreme, 2 = Moderate)

Table 3.3: Most and least correlated features

Most Positive Correlated features:		Most Negative Correlated features:	
edjefe	0.243215	hogar_nin	-0.328199
SQBedjefe	0.246368	r4t1	-0.316745
etecho3	0.257378	SQBhogar_nin	-0.311186
paredblolad	0.261274	overcrowding	-0.289110
v2a1	0.273559	SQBovercrowding	-0.258744
SQBmeaned	0.276620	r4m1	-0.253163
pisomoscer	0.280284	r4h1	-0.229889
epared3	0.292451	eviv1	-0.208038
eviv3	0.294222	pisocemento	-0.205439
SQBescolari	0.296577	epared1	-0.203025
escolari	0.302305	dependency	-0.194402
cielorazo	0.304421	hacdor	-0.191714
meaneduc	0.335203	etecho1	-0.190837
Target	1.000000	eviv2	-0.179421
elimbasu5	NaN	epared2	-0.177334

Another very important thing to check while analyzing data is the correlation of the features on the target label. In **Table 3.4**, we took the 15 most and least correlated features on our target label. We viewed these correlations without performing any feature engineering or any other data pre-processing. We see that features edjefe (years of education of male head), etecho3 (if roof is good or not), paredblolad (if the material on the outside wall is block or brick) and others are the most correlated features on the target. Similarly, features like epared2 (if walls are regular), eviv2 (if floor are regular) and etecho1 (if roof are bad) seems to be the least correlated ones.

To further understand how the highly correlated features influence the target variable we plotted the density plot. Density plots are a type of distribution that visualizes the data in a continuous interval or time period. It is a variation of histogram that uses kernel smoothing to plot values, allowing for smoother distributions by leaving out the noise. The peaks of plots aid in demonstrating where the values are concentrated over the interval. The main benefit of variation of histograms that deploy the kernel smoothing is that they better describe the shape of distribution as they are not influenced by the number of bins used. For example, a histogram containing only 5 bins might not yield noticeable enough shape of distribution as compared to 30 bin histograms.

In **Fig 3.4**, we can spot the variation of some of the highly correlated features. In the plot you can also see how variables vary in different target levels. In the second-row column one we see that the correlation of the “dependency” feature to “non-vulnerable class” is very high as compared to other classes. Density plots helped us extract a lot of information from our crude data. We also used density plots to verify if the distribution of the train variables and test variables are close or similar. It was one of the main reasons we continued working on this specific data. If we had seen a big variation between the test and train data, we would have had trouble working on the data, as our model would overfit on the train data and perform badly on the test data.

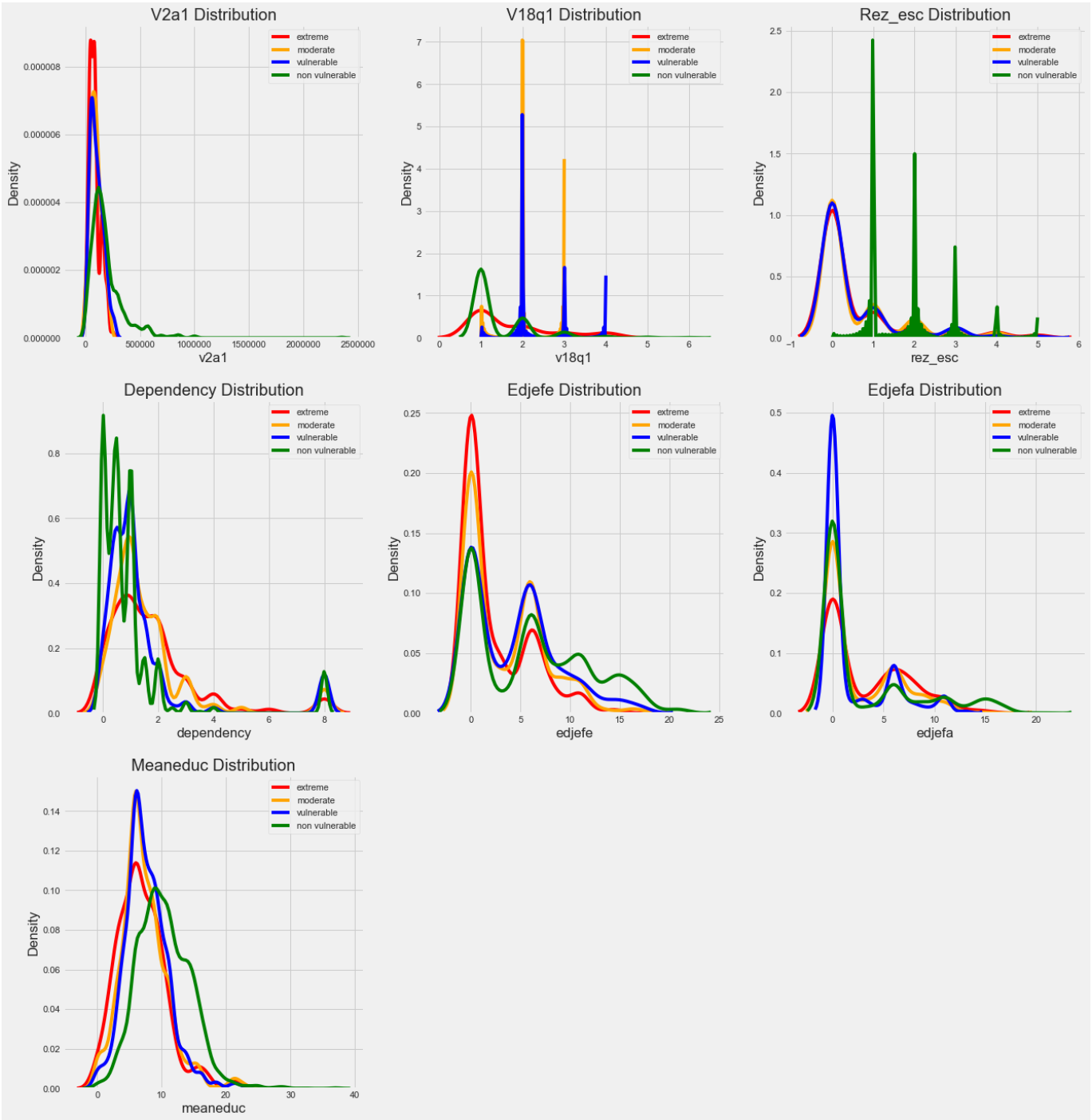


Figure 3.4: Density plots

4. METHODOLOGY

4.1. Feature Engineering and Selection

Before we started on our methodology, we went through various feature selection methods. We engineered new features from the features we had in our data. Even though the data was too mixed (Making it impossible for some classifiers to get a better decision boundary), we have discovered slight performance improvement with feature engineering/selection. Some of the features we created are shown in **Table 4.1**.

All these features and many other features that we created, contributed a lot to the performance of our framework and models. Initially, we had 142 features, with our feature engineering, we had gone up to 400 features. Some of the features were created through aggregation (mean, standard deviation, minimum, maximum and summation aggregation). To remove the features that contribute less to our model, we have utilized several feature selection methods.

In machine learning problems, the representation of data varies a lot, only a few variables may be correlated or related to the target variable. In this situation, feature extraction is of paramount importance, both to speed up learning algorithm and to improve the performance of the classifier. In **Table 4.1**, we show that some of the features that we extracted (column 1) and how they were extracted (column 2). The process of extracting requires deep understanding of the data and might consume a lot of time.

One of the promising ones being a gradient boosting models (GBM); a machine learning algorithm for regression and classification problems. GBM provides a value or score that indicates the importance of features, this makes it one of the best characteristics in ensemble models. The data fed to the GBM model and removed all features that are not important.

We also discovered some collinear features, features that are highly correlated to each other in our data. To speed up the learning process, we chose one feature from each pool of collinear features and eliminated the rest, specifically those that have above 0.98 correlation to each other.

Table 4.1: Extracted Features

No_roof or roof_waste_material	O is returned if the below features are 0, else 1. Techozinc (material on the roof is metal foil or Zink: 0/1) Techoentrepiso (material on the roof is fiber cement, mezzanine: 0/1) Techocane (material on the roof is natural fibers: 0/1) Techootro (if the material on the roof is other: 0/1)
No_electricity	O is returned if the below features are 0, else 1. Public (electricity from CNFL, ICE, ESPH/JASEC: 0/1) Planpri (electricity from private plant: 0/1) Noelec (no electricity in the dwelling: 0/1) Coopele (electricity from cooperative: 0/1)
HH_owner_adult	If "Age" is less than 18, we return 0, else 1.
adult	Individuals of age between 18-65
dependency_count	Individuals of age less than 19 and greater than 65
Overcrowding_room_bedroom	Overcrowding of both room and bedroom
room_per_person_household	Rooms per person in the household
tablet_per_person_household	Tablets per person
no_appliances	If No refrigerator, computer or television in HH
phone_per_person_household	Phone per person
escolari_age	Years of schooling divide by the age
rez_esc_age	Years behind the school divide by the age

Another model that was exploited to select high discriminative features is random forest. We first computed feature interactions for only numerical variables, then using feature importance provided by random forest we chose those that provide the highest importance degree (**see Fig. 4.1.1, 4.1.2, 4.1.3**). As you can see from the figures, the interactions of some of the features are very high. We took **100** features from each interaction (addition, subtraction, and multiplication) and observed a performance improvement.

In our study, we see feature extraction more like feature engineering, but since it generates too many features, adding it to our initial features will lead to model overfitting. Therefore, some kind of feature selection is needed to reduce the dimensionality of the

features and leave the “only” features that will give us the best accuracy from our model. Feature selection is a helpful tool that positively impacts the performance of the machine learning models.

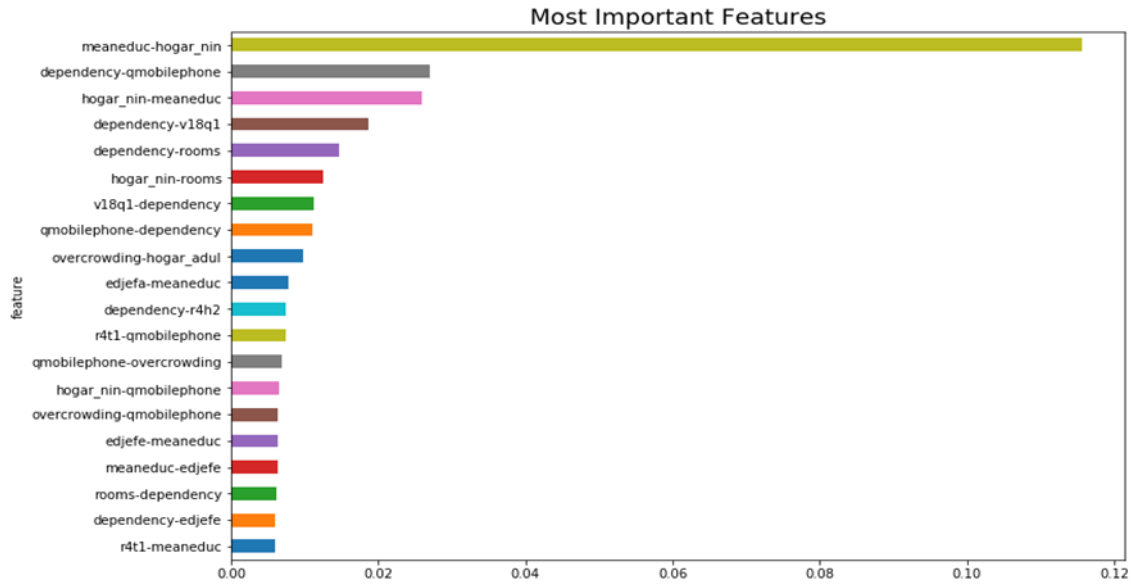


Figure 4.1.1: Feature interaction via subtraction

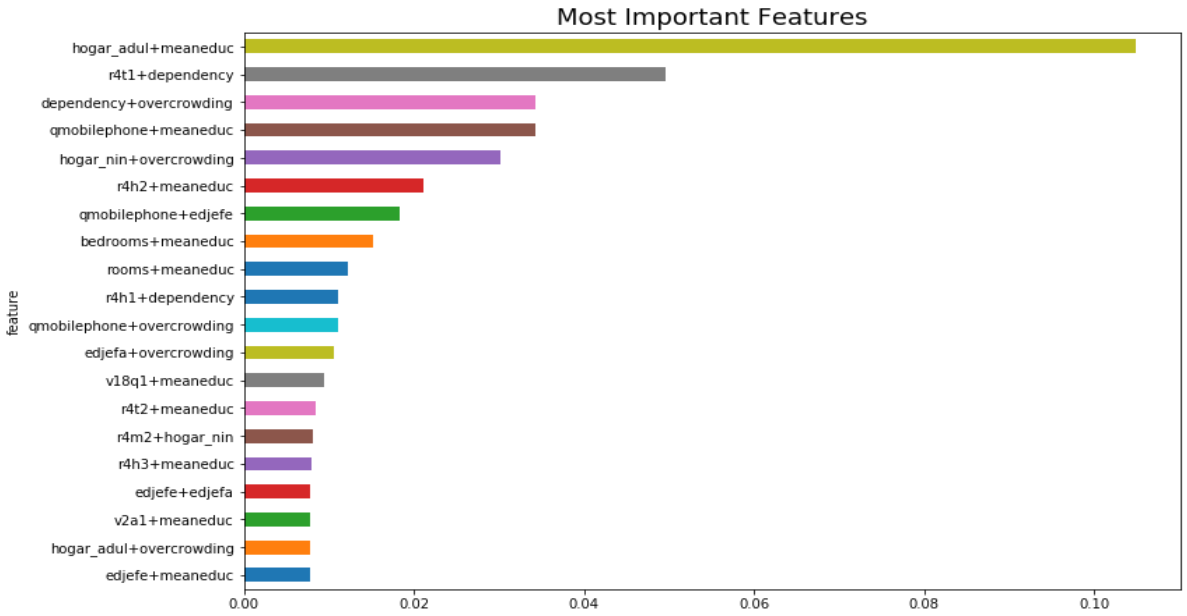


Figure 4.1.2: Feature Interaction via addition

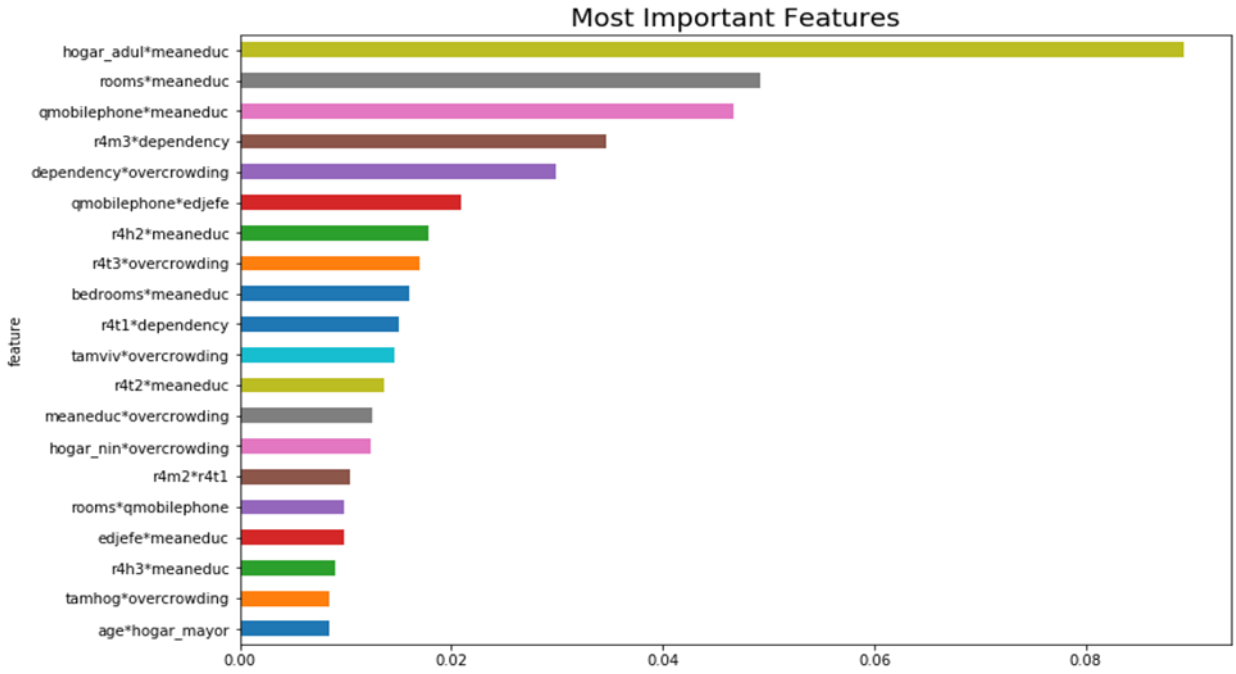


Figure 4.1.3: Feature Interaction via Multiplication

4.2. General Framework for Feature Contribution Identification

When performing feature selection, a single subset of feature is commonly selected for all classes. This may not be the best representation of the poverty status of an individual or household because different features may have different influences in different levels of poverty [21]. Therefore, a new form feature extraction technique is needed for poverty assessment. Specifically, methods that will allow us to differentiate between single-class and multi-class problems. Such techniques will determine and select a feature-set that is suitable for representing or discriminating for all the available classes.

Since feature extraction methods don't discriminate between single & multiclass problems, we are interested in selecting a distinct feature subset for each class of our classification problem. And to obtain these distinctive subsets we should utilize class specific feature subset selection (CSFS) methods. There are number of studies about CSFS that all claim that they obtain better accuracies than using other feature extraction methods. But, in our case "the poverty problem" we are not only interested in the accuracy but also in extracting information and implications hidden in these subsets of features.

For example, if education variable has high weight in one class, we can say that "education" contributes a lot to that class.

For this reason, in this research, we proposed a framework to extract feature contributions. Similar approaches have been mentioned in the literature with the intention of improving classification accuracy or other purposes[22]–[26]. These methods are generally called class-specific feature selection, but our purpose in this study is not feature selection to better accuracy rather we need to visualize features which have a high discriminative power and are able to differentiate a class of a problem from the others.

4.3. CSFS literature review

Studies about class specific subset selection have used different methods to attain a distinctive subset of features that could separate one class of problem from the others. A new CSFS method is suggested called Class-specific Ensemble Feature Selection (CEFS) [27]. As stated in their paper, it selects a subset of variables that is optimal to each classification class. Each subset is then merged with a classifier which is then utilized to estimate unseen instances. Another research selects the variables that are strongly pertinent to a class from high resolution remote sensing images [28]. To achieve this, they proposed a class specific feature subset selection method based on sparse similar samples (CFS4). Their CFS4 contains local geometric structure and discriminative info about the data.

In a different study, another similar class specific feature selection approach is proposed that utilizes clustering method (in this case K-means) [29]. It was developed for supervised interval values variables. The method takes care the selection of subsets through interval K-means clustering. And the K-means kernel is modified to adapt such interval valued data [29].

Our method follows four stages similar to study [22] but is a little bit more extended (**see Fig 4.4.5**). In other words, in our method, we added features explanations: which explains how the retrieved features of each class vary or effect the chosen class. The main reason we deploy such a method is that it is suitable or supports the use of all traditional feature extraction methods.

The four stages are as follows:

- Binarization
- SMOTE (Synthetic Minority Over-sampling Technique)
- Class Specific Subset Selection
- Feature Explanations

4.4. Binarization

Class binarization is a way of transforming a k-class classification problem into several binary problems, this allows each class to be compared against all others, or all classes compared against one another [30]. In our framework we chose to use a one-versus-all class binarization in order to turn a 4-class problem into a two-class problem. These are formed by taking the samples of one class as positive and the samples of the rest of class as negative [30].

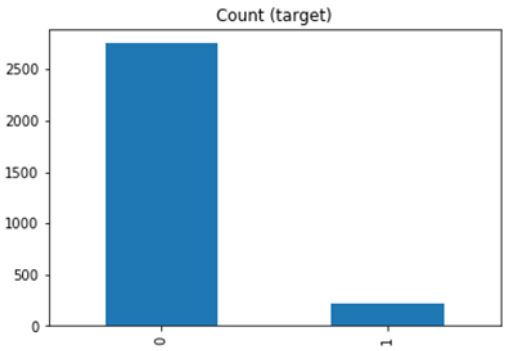


Figure 4.4.1: Class 1 vs rest

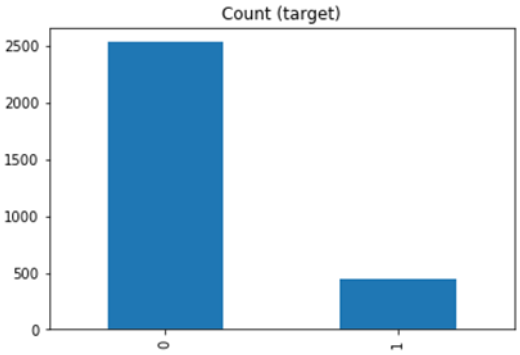


Figure 4.4.2: Class 2 vs rest

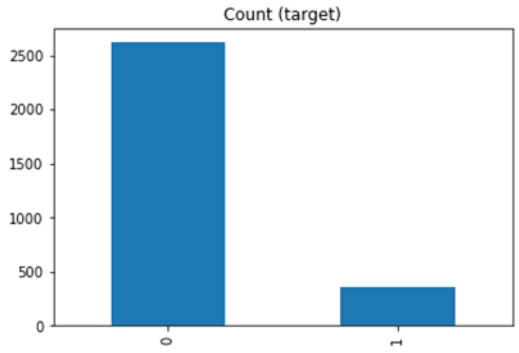


Figure 4.4.3: Class 3 vs rest

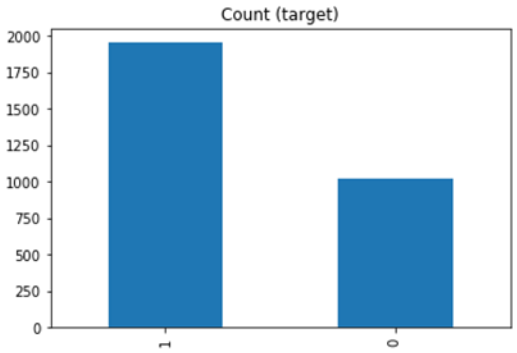


Figure 4.4.4: Class 4 vs rest

Figures 4.4.1- 4.4.4 demonstrate how our data transformed after binarization. As can be seen, this raises a big problem and feeding such data to our models will lead to overfitting. Therefore, a solution is needed in order to balance our data. And this is where SMOTE (Synthetic Minority Over-sampling Technique) comes in.

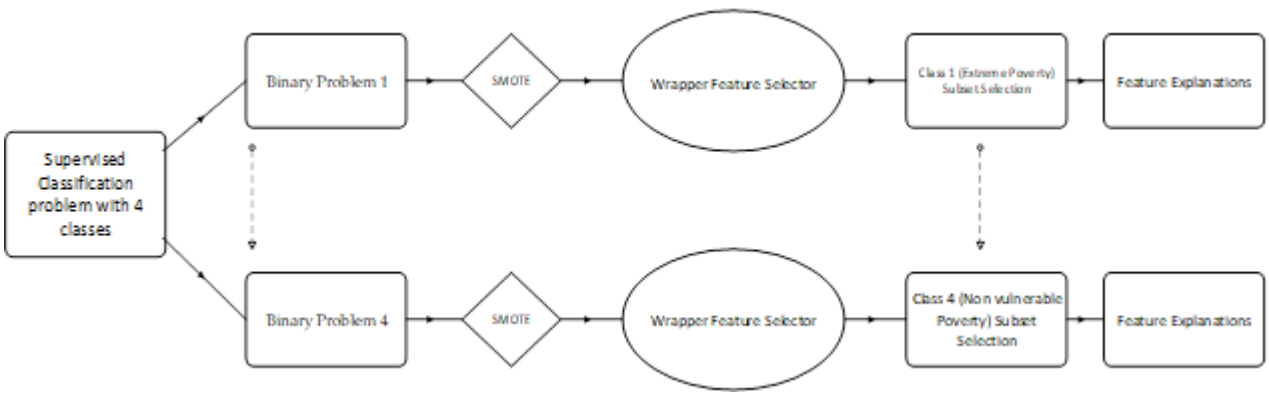


Figure 4.4.5: Proposed framework for each poverty class feature subset selection

4.5. SMOTE

To overcome the problem introduced by binarization we had to deploy some sampling techniques. There are number of sampling methods but choosing the right one for our data is essential. Promising methods include under-sampling and over-sampling. As the name implies, under-sampling is a popular approach of dealing with the class imbalance problem, where a subset of majority class is down sampled. On the other hand, over-sampling is a method of over-sampling the majority class, by creating new artificial examples from the less represented class. However, oversampling the minority class can lead to model overfitting, since it will introduce replica examples by extracting from a pool of samples that is already small. Likewise, under-sampling the majority can lead to eliminating important instances that provide perfect discrimination between the classes.

To avoid overfitting and to overcome the "imbalance problem" we used SMOTE (Synthetic Minority Over-sampling Technique) which provided the best accuracy in our case. SMOTE over-samples the minority class by creating synthetic minority class instances. In other words, the minority class is over-sampled by taking each minority class example and introducing artificial instances along the line segments linking all the k minority class nearest neighbors [31]. In their paper, they also indicate that "a combination of their method of over-sampling the minority (abnormal) class and under-sampling the majority (normal) class can attain improved classifier accuracy (in ROC space) than only under-sampling the majority class." [31]. **Figure 4.5** shows the class distribution after we applied SMOTE. As seen, each class has a balanced number of instances which improved our performance compared to when we had un-evenly distributed classes.

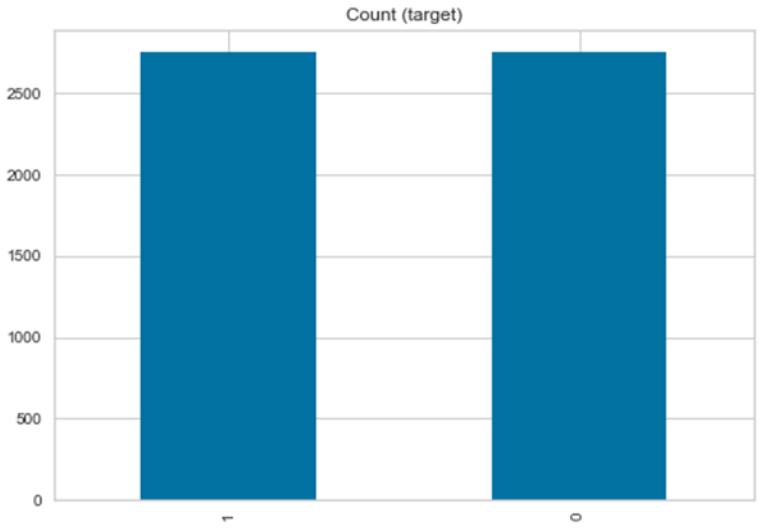


Figure 4.5: Class 1 vs Rest distribution after SMOTE

4.6. Class Specific Subset Selection

After Binarization we obtained 4 distinctive over-sampled binary classes i.e. class one against all, class two versus rest and so on. Our mission was to find a subset of features that best characterizes each class. Therefore, we used common feature selection techniques and found subsets that best isolate each class from the rest of the classes. Specifically, we exploited wrapper feature selectors and retrieved the variables that led each household to fall under this class category. Wrappers are greedy search based algorithms that select a subset of features that obtain best accuracy for a given machine learning algorithm [32].

4.6.1 Wrappers: a popular Feature extraction method

Popular feature selection methods include Filters, Wrappers and Embedded feature extraction methods. Wrappers measure the performance of the classifier and the combination of features that performs the best is chosen [32], Wrapper methods are computationally expensive since they check each combination of variables. Filter methods select the relevance of the attributes based on statistical measurements. Though, wrappers are computationally expensive, their performance is promising as compared to filter methods.

The third type, embedded methods, are functionally close to the wrappers method as they are also used to optimize the objective function. Embedded methods diverge from

other selection methods in the way selection and learning interact. They don't separate the learning from the selection part. An example includes decision trees algorithms where the algorithms learn from the data and at the same time choose the best performing features [32].

Below are some examples of these different feature selection methods:

Filter Extractors:

- Chi-square test
- Correlation coefficient

Wrapper Extractors:

- Step forward feature selectors
- Step backwards feature selectors
- Exhaustive feature selectors

Embedded Extractors:

- L1 Regularization
- Decision Tree

These methods do have benefits and drawbacks but to mention few of them Filter Methods are strong against overfitting but may fail to choose the best features. In contrary, Wrapper Methods can find the best optimal features, but they are vulnerable to overfitting, they are also computationally expensive. On the other hand, Embedded Methods are Less computationally expensive and Less vulnerable to overfitting

In our study, we used wrapper methods which seemed promising and provided us with good accuracy. As mentioned above, wrappers are classified in to three main categories.

1. Step Forward Feature Selection (SFFS):

   In the first step of SFFS, the performance of the model is evaluated against each feature. And the feature that performs the best is kept. Next, the feature is combined with all other variables and the combination of two features that provide the best performance is chosen. The operation continues until the subset of features that perform the best is selected [32].

2. Step backward feature selection (SBFS):

   SBFS, as the name implies is the exact the opposite of SFFS. It selects the attributes in a round-robin fashion where one feature is removed from a pool of features and the performance of the remained subset is computed. The process continues thus until an optimal subset is selected.

3. Exhaustive Feature Selection

   Unlike SFFS and SBFS, in exhaustive feature selection the performance of the model is computed against the combination of all variables in the data. And the subset with the highest accuracy is preserved. Unfortunately, this is the most expensive wrapper method as it evaluates all the feature combinations.

We have observed that when the data is small, running an exhaustive search is the best choice. But if the data is quite big the step forward and backward feature selection methods are the preferred wrapper methods. in this research, due to the high dimensionality of our data, we exploited Step Forward Feature Selection. Finally, at the end of class specific subset selection, we retrieved the characteristics that cause an individual/household to fall into a poverty level (see **Table 5**).

4.7. Model Explanations

Machine learning models are designed in the form of a black box, where you don't understand the reasons behind predictions. **Fig 4.7** demonstrates a clear example of how ML models work i.e. they take input and provide outputs. What is happening in the box and how It chooses the features that provide such output is unknown. Determining the factors behind predictions is significant when a model is used for policymaking. Particularly in human's poverty status, predictions cannot be acted upon on their own as the penalties may be dangerous. In such cases having model explanation methods is very essential.

These methods will help us understand why the model has made such decisions. For instance, you are developing machine learning models for credit risk analysis. And one of your costumers has asked you to give an explanation in case of negative credit decision. A similar case is poverty, where most of the aid organizations would not only be interested to know whether an individual is poor or not, but rather what led him/her to be poor/non-poor. In such a situation the only way to provide clarification would be to

use model explanation methods. The commonly used approaches are mainly examining model features by looking at feature importance's and correlations.

Feature importance's do provide intuitions about what the model is learning or the variables that are important. Yet, this is unreliable if the variables are correlated. **Figs 4.1.1- 4.1.3** show feature importance's of our data. It can be seen from the figures that there are quite good insights about the data, even though there is no correlation information. If we could use deep neural networks (DNN) for model explanation, we could check the weights as they hold the information about the variables. However, this would be a complicated task since the information is compressed, and examining next layers even gets tougher as the network grows.

Figure 4.7: Machine learning model structure

Therefore, after a suitable subset of feature that best describes each class was obtained from our framework, we validated our results with LIME (Local Interpretable Model-agnostic Explanations). LIME is a model explanation method that is used to explain the outcome of a machine learning algorithms. It provides textual or visual artifacts that give qualitative understanding [33]. LIME data helps us to validate the effect of the features we retrieved from our framework on the samples we are investigating. In other words, It provides qualitative interpretation of the relationship between the instance's variables and the model's prediction [33].

4.8 Classification

Machine learning algorithms are designed to deal with many problems that vary from regression, classification, detections and many more. Our problem was a classification problem, therefore, we deployed classification algorithms. Classification is a supervised learning approach where the model learns from the input patterns and then uses this knowledge to classify observations into separate categories. There are a number of classifications algorithms and choosing the right one for your data is crucial. Some of them are listed below.

- Logistic regression
- K-nearest Neighbor
- Support vector machines
- Random forest
- Decision tree
- Neural networks

4.8.1 Random forest classifier

Random forest classifier is built/based on decision trees. A decision tree is a ML algorithm that uses a tree-like structure model. Each node represents a test of an attribute where the attribute is split. For example, assume that you have education indicative feature in your data, the decision tree will likely split into “Education” and “No education” depending on the values in the variable. **See Fig 4.8.1** illustrating how decision tree works. The responses to the predictions of the next split depend on the number of available split possibilities in the feature.

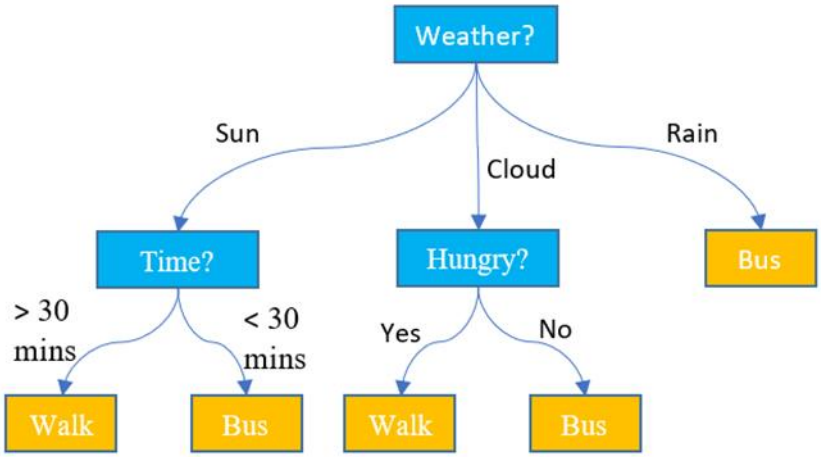


Figure 4.8.1: Decision tree (image taken from DisplayR blog post)

Random Forest, as the name implies, is a combination of decision trees (**see Fig 4.8.2**). The underlying idea behind random forest is the fact that many uncorrelated trees operating as one group will outperform individual decision trees. The generalization error depends on the correlation strength between the trees. Where the error decreases as the number of decision trees grow [34].

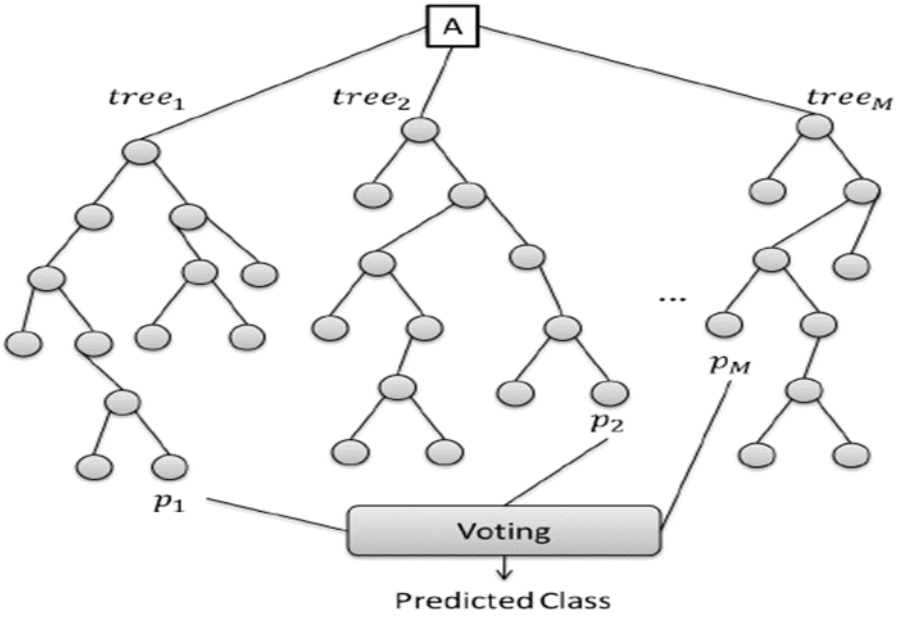


Figure 4.8.2: Random Forest structure

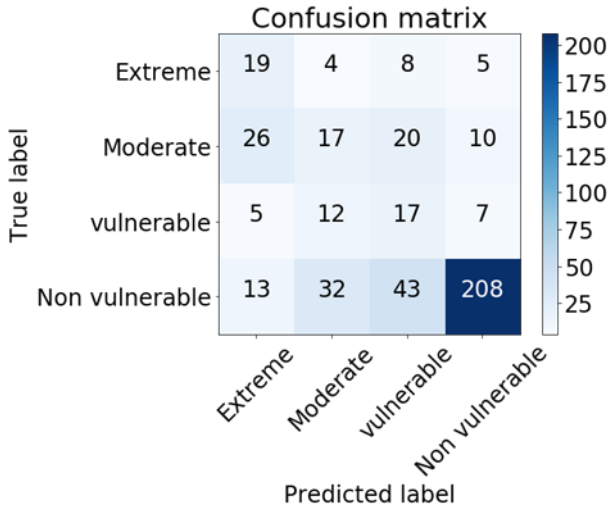
To get a good performance, we went through numerous preprocessing stages such as imputing missing values, aggregation of household characteristics, computing feature interactions, performing feature transformations and so on. A process that is both time

intense and ambiguous in building a consistent model. Then we trained on a random forest model on our clean data. The smallness and unbalancedness of the data made it hard for the classifier to best separate between classes. The non-vulnerable class (class 4) is more over-represented than other classes. If we are to separate class 4 from all other classes, our decision boundary will be able to separate between these two classes. However, the classifier fails to identify a perfect decision boundary that can easily isolate between the other three classes (Extreme, Moderate, Vulnerable). Even though, with such small and imbalanced data we identified the features that best characterize these classes, the classifiers find it challenging to achieve a high accuracy score. Therefore, the combination of our framework and LIME seems to be a promising move to depict each poverty class.

5. DISCUSSION & RESULTS

The results of this study provide interesting and promising insights that would trigger further research in this area. For demonstration purposes, we chose six features from a set of features that best describe/characterize each class (**see Table 5.2**). The first column of the table indicates the classes of our classification data. The second column are the 6 features we chose for the illustration. The third column is the descriptions of features, and finally the next two columns are feature dimensions and LIME explanations respectively.

Table 5.1: Confusion Matrix



The extracted features contribute differently on the classes. For instance, dimensions like current assets (**Table 5.2**), and standard of living (housing related feature) best contribute to class 1 (Extreme Poverty), whereas they are not among the features that best discriminate class 3 from other classes. Likewise, it can be seen that different levels of education contribute differently to the classes. For instance, deprivation of basic education is a characteristic of extreme poverty, while good education is an indicator of the non-vulnerable class. Bearing in mind that we don't have any features that are characteristic of income; this further strengthens the theory that poverty is a multidimensional concept.

Table 5.2: Contribution of features to each poverty target class

Levels of Poverty	Features	Feature Descriptions	Dimension	LIME Explanations
Extreme Poverty	v18q1	Number of Tablets	Current Assets	Deprived
	epared3	IF Walls good	standard of living	Deprived
	instlevel1	No Level of Education	Education	Yes
	rent_per_room	Rent Per Room	-	0 (mostly don't pay rent)
	instlevel9	Postgraduate H. education	Education	Mostly lack Postgraduate higher Education
	pisonatur	IF Floor is natural Material	standard of living	Not Natural Material (Deprived)
	hogar_adul	Number of Adults in the Household	-	Adults (1,2,3,4,5) are indicative to fall into this class
	hogar_mayor	Individuals Age > 65	-	Almost 30 % of this class, Age > 65
Moderate	age_std	Age Standard deviation	-	Probability of moderate class increases if std > 22
	paredmad	Material on the outside wall is wood	Physical capital	About 80% in this class walls are not wood
	paredzinc	Material on the outside wall is zinc	Physical capital	Deprived (Almost 98% wall is not zinc)
	Escolari_mean	Average years Education	Education	[6.75 - 10]
	hogar_nin	Number of Children (0-19) in household	-	1
	instlevel2	Incomplete Primary Education	Education	Yes
Vulnerable	pisocemento	Material on the floor is cement	Physical capital	(around 17 % have cement on the floor)
	meaneduc	Average years of Education for Adults	Education	6
	sanitario1	IF no Toilet in the Dwelling	Physical capital	0 (Almost Every HH has Toilet in the dwelling)
	epared3	IF Walls good	Physical capital	Walls good (not deprived)
	phone_per_person_household	Phone per person in household	Physical capital	1 (indicates almost every person has telephone)
	television	Television	Physical capital	1 (indicates HH has Television)
Non-Vulnerable	etecho3	IF roof is good	Physical capital	1 (HH has a good roof)
	dependence	Dependence Rate	Social capital	Mostly dependence rate is 0
	escolari_mean	Average years of schooling	Education	>= 13 (higher the more likely to be this class
	eviv3	IF floor is good	Physical capital	1 (indicating floor is good)

The performance of our random forest classifier is evaluated using F1-macro score. We chose this metric for two reasons; I) our classes are distributed unproportionally, II)

the provided metric from Kaggle website was F1-macro. Additionally, we haven't had the labels of our test samples, so we split our train data in into 85% train and 15% validation portions. After training was done, we computed the confusion matrix (**see Table 5.1**) of validation set. The results show that apart from class 4 the rest are strongly mixed.

5.1 Conclusion

In this study we proposed a new poverty characterization method. A method that provides the features indicating a household to be considered to be in a specific class of poverty. We used LIME model explanation technique to validate if the features extracted from our framework affect the chosen class. We then trained random forest classifier to our data to best classify between classes.

We have discovered that different features contribute differently to the classes. This might be an easy guess when talking about poverty but extracting from a black box machine learning model is the challenging part. Moreover, too many missing values, data disproportion, and small train data made it much harder for the random classifier to achieve a very high accuracy. However, using our framework we can easily depict the causality of each poverty class.

As a continuation of this study, we recommend obtaining enough multidimensionally labeled data. We also suggest better feature explanation methods to be developed for tabular data, since LIME might be misleading sometimes. Another possible improvement could be the combination of many datasets such as household (survey) data, day and night satellite data, and any data that constitutes observable characteristics of poverty. Since only one type of data cannot reflect the majority of humankind's poverty status, machine learning algorithms could be developed in such a way that it learns from various datasets at the same time.

After the model is built, the indicators that led someone to fall into a class are already known. So, when someone is predicted to be in this class, you can provide support according to the features extracted from our framework. for example, if that person is deprived of health, we can provide health services.

Thus, this proves that Machine learning and Feature selection are good use of tackling poverty and our research is a good example. Deprivation is a horrifying situation that affects most of our community from different angles. Therefore, poverty

characterization and looking at poverty from a multidimensional perspective is very essential in policymaking and this study sheds light on the prospect of developing a mechanism that utilizes machine learning and hence contributes to efforts of making the causes of poverty more easily understandable.

REFERENCES

[1] A. Sen, “Poverty: An Ordinal Approach to Measurement,” *Econometrica*, vol. 44, no. 2, p. 219, 1976.

[2] F. Bourguignon and S. R. Chakravarty, “The Measurement of Multidimensional Poverty,” *J. Econ. Inequal.*, vol. 1225, no. February, pp. 41–42, 2003.

[3] B. Belhadj and M. Limam, “Unidimensional and multidimensional fuzzy poverty measures: New approach,” *Econ. Model.*, vol. 29, no. 4, pp. 995–1002, 2012.

[4] N. Na¨ıri and N. Quinn, “Alkire-Foster Method The Global MPI Policy Use Public Communication The Global Multidimensional Poverty Index,” no. November, 2017.

[5] P. W. Briefs, “Multidimensional Child poverty UNICEF,” no. February, 2011.

[6] S. Alkire and M. E. Santos, “Measuring Acute Poverty in the Developing World: Robustness and Scope of the Multidimensional Poverty Index,” *World Dev.*, vol. 59, pp. 251–274, 2014.

[7] S. Alkire and M. E. Santos, “Multidimensional Poverty Index,” *Oxford Poverty Hum. Dev. Initiat.*, no. July, pp. 1–8, 2010.

[8] S. Alkire and S. Seth, “Multidimensional Poverty Reduction in India between 1999 and 2006: Where and How?,” *World Dev.*, vol. 72, pp. 93–108, 2015.

[9] M. Xie, N. Jean, M. Burke, D. Lobell, and S. Ermon, “Transfer Learning from Deep Features for Remote Sensing and Poverty Mapping,” 2015.

[10] M. Hernandez, L. Hong, V. Frías-Martínez, A. Whitby, and E. Frías-Martínez, “Estimating Poverty Using Cell Dhone data: Evidence from Guatemala,” *World Bank Group, Policy Res. Work. Pap.*, no. 7969, 2017.

[11] M. Costa, “IRISS at CEPS / INSTEAD A COMPARISON BETWEEN UNIDIMENSIONAL AND by An Integrated Research Infrastructure in the,” 2003.

[12] J. Foster, J. Greer, and E. Thorbecke, “A Class of Decomposable Poverty Indices A Class of Decomposable Poverty Measures,” no. February 1984, 2015.

[13] M. Costa and L. De Angelis, “where n is the cardinality of the set A . In the case of a census , A contains all the households of a population , while , if A is a representative sample of a population ,” 2008.

[14] A. C. S. M. S. Kar, “Poverty Level of Households: A Multidimensional Approach Based on Fuzzy Mathematics,” vol. 463–487, 2014.

[15] N. S. Sani, M. A. Rahman, A. A. Bakar, S. Sahran, and H. Mohd, “Machine Learning Approach for Bottom 40 Percent Households (B40) Poverty Classification,” vol. 8, no. 4, pp. 1698–1705, 2018.

[16] S. Narendranath, S. Khare, D. Gupta, and A. Jyotishi, “Characteristics of ’ Escaping ’ and ’ Falling into ’ Poverty in India : An Analysis of IHDS Panel Data using machine learning approach,” *2018 Int. Conf. Adv. Comput. Commun. Informatics*, pp. 1391–1397, 2018.

[17] World bank, “Measuring income and poverty using Proxy Means Tests.”

[18] L. Mcbride and A. Nichols, “Improved poverty targeting through machine learning: An application to the USAID Poverty Assessment Tools,” p. 24, 2015.

[19] J. Blumenstock, G. Cadamuro, and R. On, “Predicting poverty and wealth from mobile phone metadata,” *Science (80-.).*, vol. 350, no. 6264, pp. 1073–1076, 2015.

[20] Laurens van der Maaten and H. Geoffrey E., “Visualizing Data using t-SNE,” *J. Mach. Learn. Res.*, vol. 164, no. 2210, p. 10, 2008.

[21] J. H. Mohamud, “Poverty Level Characterization via Feature Selection and Machine Learning.”

[22] B. B. Pineda-Bautista, J. A. Carrasco-Ochoa, and J. F. Martínez-Trinidad, “General framework for class-specific feature selection,” *Expert Systems with Applications*, vol. 38, no. 8. pp. 10018–10024, 2011.

[23] W. Zhou and J. A. Dickerson, “A novel class dependent feature selection method for cancer biomarker discovery,” *Computers in Biology and Medicine*, vol. 47, no. 1. Elsevier, pp. 66–75, 2014.

[24] D. Kumar, “Class Specific Feature Selection for Identity Validation using Dynamic Signatures,” *Journal of Biometrics & Biostatistics*, vol. 04, no. 02. 2013.

[25] A. Roy, P. D. Mackin, and S. Mukhopadhyay, “Methods for pattern selection, class-specific feature selection and classification for automated learning,” *Neural Networks*, vol. 41. Elsevier Ltd, pp. 113–129, 2013.

[26] A. M. P. Canuto, K. M. O. Vale, A. Feitos, and A. Signoretti, “ReinSel: A class-based mechanism for feature selection in ensemble of classifiers,” *Applied Soft Computing Journal*, vol. 12, no. 8. Elsevier B.V., pp. 2517–2529, 2012.

[27] C. Soares, P. Williams, J. Gilbert, and G. Dozier, “A class-specific ensemble

feature selection approach for classification problems," *Proceedings of the 48th Annual Southeast Regional Conference*. p. 33, 2010.

[28] X. Chen and Y. Gu, "Class-Specific Feature Selection With Local Geometric Structure and Discriminative Information Based on Sparse Similar Samples," *IEEE Geoscience and Remote Sensing Letters*, vol. 12, no. 7. IEEE, pp. 1392–1396, 2015.

[29] D. S. Guru and N. Vinay Kumar, "Class specific feature selection for interval valued data through interval K-means clustering," *Communications in Computer and Information Science*, vol. 709. pp. 228–239, 2017.

[30] Y. S. Erin L. Allwein, Robert E. Schapire, "Reducing Multiclass to Binary," *J. Mach. Learn. Res.*, vol. 1, pp. 113–141, 2000.

[31] W. P. K. Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, "Synthetic Minority Over-sampling Technique (SMOTE)," *J. Artif. Intell. Res.*, vol. 16, pp. 321–357, 2002.

[32] L. A. Z. (eds. . Isabelle Guyon, Steve Gunn, Masoud Nikravesh, *Feature Extraction: Foundations and Applications*, vol. 91. 2017.

[33] and C. G. M. T. Ribeiro, S. Singh, "Why should i trust you?: Explaining the predictions of any classifier," in *International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 1135–1144.

[34] L. Breiman, "RANDOM FORESTS," pp. 1–33, 2001.